

1	%%	1
2	%% This is file `expl3-code.tex',	2
3	%% generated with the docstrip utility.	3
4	%%	4
5	%% The original source files were:	5
6	%%	6
7	%% expl3.dtx (with options: `code')	7
8	%% l3bootstrap.dtx (with options: `code')	8
9	%% l3names.dtx (with options: `code')	9
10	%% l3basics.dtx (with options: `code')	10
11	%% l3expan.dtx (with options: `code')	11
12	%% l3quark.dtx (with options: `code')	12
13	%% l3tl.dtx (with options: `code')	13
14	%% l3tl-build.dtx (with options: `code')	14
15	%% l3str.dtx (with options: `code')	15
16	%% l3seq.dtx (with options: `code')	16
17	%% l3int.dtx (with options: `code')	17
18	%% l3flag.dtx (with options: `code')	18
19	%% l3prg.dtx (with options: `code')	19
20	%% l3sys.dtx (with options: `code')	20
21	%% l3clist.dtx (with options: `code')	21
22	%% l3token.dtx (with options: `code')	22
23	%% l3prop.dtx (with options: `code')	23
24	%% l3msg.dtx (with options: `code')	24
25	%% l3file.dtx (with options: `code')	25
26	%% l3skip.dtx (with options: `code')	26
27	%% l3keys.dtx (with options: `code')	27
28	%% l3intarray.dtx (with options: `code')	28
29	%% l3fp.dtx (with options: `code')	29
30	%% l3fp-aux.dtx (with options: `code')	30
31	%% l3fp-traps.dtx (with options: `code')	31
32	%% l3fp-round.dtx (with options: `code')	32
33	%% l3fp-parse.dtx (with options: `code')	33
34	%% l3fp-assign.dtx (with options: `code')	34
35	%% l3fp-logic.dtx (with options: `code')	35
36	%% l3fp-basics.dtx (with options: `code')	36
37	%% l3fp-extended.dtx (with options: `code')	37
38	%% l3fp-expo.dtx (with options: `code')	38
39	%% l3fp-trig.dtx (with options: `code')	39
40	%% l3fp-convert.dtx (with options: `code')	40
41	%% l3fp-random.dtx (with options: `code')	41
42	%% l3fp-types.dtx (with options: `code')	42
43	%% l3fp-symbolic.dtx (with options: `code')	43
44	%% l3fp-functions.dtx (with options: `code')	44
45	%% l3fparray.dtx (with options: `code')	45
46	%% l3bitset.dtx (with options: `code')	46

47	%% l3cctab.dtx (with options: `code')	47
48	%% l3sort.dtx (with options: `code')	48
49	%% l3str-convert.dtx (with options: `code')	49
50	%% l3tl-analysis.dtx (with options: `code')	50
51	%% l3benchmark.dtx (with options: `code')	51
52	%% l3regex.dtx (with options: `code')	52
53	%% l3box.dtx (with options: `code')	53
54	%% l3color.dtx (with options: `code')	54
55	%% l3graphics.dtx (with options: `code')	55
56	%% l3opacity.dtx (with options: `code')	56
57	%% l3pdf.dtx (with options: `code')	57
58	%% l3coffins.dtx (with options: `code')	58
59	%% l3luatex.dtx (with options: `code')	59
60	%% l3unicode.dtx (with options: `code')	60
61	%% l3text.dtx (with options: `code')	61
62	%% l3text-case.dtx (with options: `code')	62
63	%% l3text-map.dtx (with options: `code')	63
64	%% l3text-purify.dtx (with options: `code')	64
65	%% l3legacy.dtx (with options: `code')	65
66	%% l3deprecation.dtx (with options: `code')	66
67	%%	67
68	%% Copyright (C) 1990-2025 The LaTeX Project	68
69	%%	69
70	%% It may be distributed and/or modified under the conditions of	70
71	%% the LaTeX Project Public License (LPPL), either version 1.3c of	71
72	%% this license or (at your option) any later version. The latest	72
73	%% version of this license is in the file:	73
74	%%	74
75	%% https://www.latex-project.org/lppl.txt	75
76	%%	76
77	%% This file is part of the "l3kernel bundle" (The Work in LPPL)	77
78	%% and all files in that bundle must be distributed together.	78
79	%%	79
80	%% File: expl3.dtx	80
81	<u>\def\ExplFileDate{2025-04-29}%</u>	81
82	<u>\beginingroup</u>	82
83	<u>\def\next{\endgroup}%</u>	83
84	<u>\expandafter\ifx\csname PackageError\endcsname\relax</u>	84
85	<u>\beginingroup</u>	85
86	<u>\def\next{\endgroup\endgroup}%</u>	86
87	<u>\def\PackageError#1#2#3%</u>	87
88	<u>{%</u>	88
89	<u>\endgroup</u>	89
90	<u>\errhelp{#3}%</u>	90
91	<u>\errmessage{#1 Error: #2!}%</u>	91
92	<u>}%</u>	92

93	<u>\fi</u>	93
94	<u>\expandafter\ifx\csname</u> ExplLoaderFileDate <u>\endcsname\relax</u>	94
95	<u>\def\next</u>	95
96	{%	96
97	<u>\PackageError{expl3}{No expl3 loader detected}</u>	97
98	{%	98
99	You have attempted to use the expl3 code directly rather than using	99
100	the correct loader. Loading of expl3 will abort.	100
101	}%	101
102	<u>\endgroup</u>	102
103	<u>\endinput</u>	103
104	}	104
105	<u>\else</u>	105
106	<u>\ifx\ExplLoaderFileDate\ExplFileDate</u>	106
107	<u>\else</u>	107
108	<u>\def\next</u>	108
109	{%	109
110	<u>\PackageError{expl3}{Mismatched expl3 files detected}</u>	110
111	{%	111
112	You have attempted to load expl3 with mismatched files:	112
113	probably you have one or more files 'locally installed' which	113
114	are in conflict. Loading of expl3 will abort.	114
115	}%	115
116	<u>\endgroup</u>	116
117	<u>\endinput</u>	117
118	}%	118
119	<u>\fi</u>	119
120	<u>\fi</u>	120
121	<u>\next</u>	121
122	<u>\beginingroup\expandafter\expandafter\expandafter\endgroup</u>	122
123	<u>\expandafter\ifx\csname</u> ver@expl3-code.tex <u>\endcsname\relax</u>	123
124	<u>\expandafter\edef\csname</u> ver@expl3-code.tex <u>\endcsname</u>	124
125	{%	125
126	<u>\ExplFileDate\space</u>	126
127	L3 programming layer	127
128	}%	128
129	<u>\else</u>	129
130	<u>\expandafter\endinput</u>	130
131	<u>\fi</u>	131
132	<u>\immediate\write-1</u> %	132
133	{%	133
134	Package: expl3	134
135	<u>\ExplFileDate\space</u>	135
136	L3 programming layer (code)%	136
137	}%	137
138	%% File: l3bootstrap.dtx	138

139	<u>\beginngroup\expandafter\expandafter\expandafter\endgroup</u>	139
140	<u>\expandafter\ifx\csname</u> pdfstrcmp <u>\endcsname\relax</u>	140
141	<u>\let\pdfstrcmp\strcmp</u>	141
142	<u>\fi</u>	142
143	<u>\beginngroup\expandafter\expandafter\expandafter\endgroup</u>	143
144	<u>\expandafter\ifx\csname</u> directlua <u>\endcsname\relax</u>	144
145	<u>\else</u>	145
146	<u>\ifnum\luatexversion</u> <110 %	146
147	<u>\else</u>	147
148	<u>\beginngroup\expandafter\expandafter\expandafter\endgroup</u>	148
149	<u>\expandafter\ifx\csname</u> newcatcodetable <u>\endcsname\relax</u>	149
150	<u>\input{ltluatex}%</u>	150
151	<u>\fi</u>	151
152	<u>\beginngroup\expandafter\expandafter\expandafter\endgroup</u>	152
153	<u>\expandafter\ifx\csname</u> newluabytecode <u>\endcsname\relax</u>	153
154	<u>\else</u>	154
155	<u>\newluabytecode\@expl@luadata@bytecode</u>	155
156	<u>\fi</u>	156
157	<u>\directlua{require("expl3")}%</u>	157
158	<u>\ifnum</u> 0%	158
159	<u>\directlua{</u>	159
160	if status.ini_version then	160
161	tex.write("1")	161
162	end	162
163	<u>}>0 %</u>	163
164	<u>\everyjob\expandafter{%</u>	164
165	<u>\the\expandafter\everyjob</u>	165
166	<u>\csname\detokenize{lua_now:n}\endcsname{require("expl3")}%</u>	166
167	<u>}%</u>	167
168	<u>\fi</u>	168
169	<u>\fi</u>	169
170	<u>\fi</u>	170
171	<u>\beginngroup</u>	171
172	<u>\def\next{\endgroup}%</u>	172
173	<u>\def\ShortText{Required primitives not found}%</u>	173
174	<u>\def\LongText%</u>	174
175	<u>{%</u>	175
176	The L3 programming layer requires the e-TeX primitives and the	176
177	<u>\LineBreak</u> 'pdfTeX utilities' as described in the README file.	177
178	<u>\LineBreak</u>	178
179	These are available in the engines <u>\LineBreak</u>	179
180	- pdfTeX v1.40.20 <u>\LineBreak</u>	180
181	- XeTeX v0.999991 <u>\LineBreak</u>	181
182	- LuaTeX v1.10 <u>\LineBreak</u>	182
183	- e-(u)pTeX v3.8.2 <u>\LineBreak</u>	183
184	- Prote (2021) <u>\LineBreak</u>	184

```

185 or later.\LineBreak
186 \LineBreak
187 }%
188 \ifnum0%
189 \expandafter\ifx\csname luatexversion\endcsname\relax
190 \expandafter\ifx\csname expanded\endcsname\relax\else 1\fi
191 \else
192 \ifnum\luatexversion<110 \else 1\fi
193 \fi
194 =0 %
195 \newlinechar`^^J %
196 \def\LineBreak{\noexpand\MessageBreak}%
197 \expandafter\ifx\csname PackageError\endcsname\relax
198 \def\LineBreak{^^J}%
199 \begingroup
200 \lccode`~=`\lccode`~}%
201 \lccode`T=`T\lccode`H=`H%
202 \catcode`\_ =11 %
203 \lowercase{\endgroup\def\PackageError#1#2#3{%
204 \begingroup\errorcontextlines-1\immediate\write0{\errhelp{#3}\def%
205 \_ {#1 Error: #2.^^J^^J
206 Type H <return> for immediate help}\def~{\errmessage{%
207 \_ }}~\endgroup}}%
208 \fi
209 \edef\next
210 {%
211 \noexpand\PackageError{expl3}{\ShortText}
212 {\LongText Loading of expl3 will abort!}%
213 \endgroup
214 \noexpand\endinput
215 }%
216 \fi
217 \next
218 \protected\edef\ExplSyntaxOff
219 {%
220 \protected\def\noexpand\ExplSyntaxOff{}%
221 \catcode 9 = \the\catcode 9\relax
222 \catcode 32 = \the\catcode 32\relax
223 \catcode 34 = \the\catcode 34\relax
224 \catcode 58 = \the\catcode 58\relax
225 \catcode 94 = \the\catcode 94\relax
226 \catcode 95 = \the\catcode 95\relax
227 \catcode 124 = \the\catcode 124\relax
228 \catcode 126 = \the\catcode 126\relax
229 \endlinechar = \the\endlinechar\relax
230 \chardef\csname\detokenize{1\_kernel\_expl\_bool}\endcsname = 0\relax

```

231	}%	231
232	<u>\catcode</u> 9 = 9 <u>\relax</u>	232
233	<u>\catcode</u> 32 = 9 <u>\relax</u>	233
234	<u>\catcode</u> 34 = 12 <u>\relax</u>	234
235	<u>\catcode</u> 58 = 11 <u>\relax</u>	235
236	<u>\catcode</u> 94 = 7 <u>\relax</u>	236
237	<u>\catcode</u> 95 = 11 <u>\relax</u>	237
238	<u>\catcode</u> 124 = 12 <u>\relax</u>	238
239	<u>\catcode</u> 126 = 10 <u>\relax</u>	239
240	<u>\endlinechar</u> = 32 <u>\relax</u>	240
241	<u>\global\chardef</u> \l__kernel_expl_bool = 1 <u>\relax</u>	241
242	<u>\protected \def</u> \ExplSyntaxOn	242
243	{	243
244	\bool_if:NF \l__kernel_expl_bool	244
245	{	245
246	\cs_set_protected:Npe \ExplSyntaxOff	246
247	{	247
248	\char_set_catcode:nn { 9 } { \char_value_catcode:n { 9 } }	248
249	\char_set_catcode:nn { 32 } { \char_value_catcode:n { 32 } }	249
250	\char_set_catcode:nn { 34 } { \char_value_catcode:n { 34 } }	250
251	\char_set_catcode:nn { 58 } { \char_value_catcode:n { 58 } }	251
252	\char_set_catcode:nn { 94 } { \char_value_catcode:n { 94 } }	252
253	\char_set_catcode:nn { 95 } { \char_value_catcode:n { 95 } }	253
254	\char_set_catcode:nn { 124 } { \char_value_catcode:n { 124 } }	254
255	\char_set_catcode:nn { 126 } { \char_value_catcode:n { 126 } }	255
256	\tex_endlinechar:D =	256
257	\tex_the:D \tex_endlinechar:D \scan_stop:	257
258	\bool_set_false:N \l__kernel_expl_bool	258
259	\cs_set_protected:Npn \ExplSyntaxOff { }	259
260	}	260
261	}	261
262	\char_set_catcode_ignore:n { 9 } % tab	262
263	\char_set_catcode_ignore:n { 32 } % space	263
264	\char_set_catcode_other:n { 34 } % double quote	264
265	\char_set_catcode_letter:n { 58 } % colon	265
266	\char_set_catcode_math_superscript:n { 94 } % circumflex	266
267	\char_set_catcode_letter:n { 95 } % underscore	267
268	\char_set_catcode_other:n { 124 } % pipe	268
269	\char_set_catcode_space:n { 126 } % tilde	269
270	\tex_endlinechar:D = 32 \scan_stop:	270
271	\bool_set_true:N \l__kernel_expl_bool	271
272	}	272
273	%% File: l3names.dtx	273
274	<u>\let</u> \tex_global:D \global	274
275	<u>\let</u> \tex_let:D \let	275
276	<u>\begingroup</u>	276

277	<code>\long \def _kernel_primitive:NN #1#2</code>	277
278	<code>{ \tex_global:D \tex_let:D #2 #1 }</code>	278
279	<code>_kernel_primitive:NN \</code>	<code>\tex_space:D</code> 279
280	<code>_kernel_primitive:NN \/</code>	<code>\tex_italiccorrection:D</code> 280
281	<code>_kernel_primitive:NN \-</code>	<code>\tex_hyphen:D</code> 281
282	<code>_kernel_primitive:NN \above</code>	<code>\tex_above:D</code> 282
283	<code>_kernel_primitive:NN \abovedisplayshortskip</code>	<code>\tex_abovedisplayshortskip:D</code> 283
284	<code>_kernel_primitive:NN \abovedisplayskip</code>	<code>\tex_abovedisplayskip:D</code> 284
285	<code>_kernel_primitive:NN \abovewithdelims</code>	<code>\tex_abovewithdelims:D</code> 285
286	<code>_kernel_primitive:NN \accent</code>	<code>\tex_accent:D</code> 286
287	<code>_kernel_primitive:NN \adjdemerits</code>	<code>\tex_adjdemerits:D</code> 287
288	<code>_kernel_primitive:NN \advance</code>	<code>\tex_advance:D</code> 288
289	<code>_kernel_primitive:NN \afterassignment</code>	<code>\tex_afterassignment:D</code> 289
290	<code>_kernel_primitive:NN \aftergroup</code>	<code>\tex_aftergroup:D</code> 290
291	<code>_kernel_primitive:NN \atop</code>	<code>\tex_atop:D</code> 291
292	<code>_kernel_primitive:NN \atopwithdelims</code>	<code>\tex_atopwithdelims:D</code> 292
293	<code>_kernel_primitive:NN \badness</code>	<code>\tex_badness:D</code> 293
294	<code>_kernel_primitive:NN \baselineskip</code>	<code>\tex_baselineskip:D</code> 294
295	<code>_kernel_primitive:NN \batchmode</code>	<code>\tex_batchmode:D</code> 295
296	<code>_kernel_primitive:NN \begingroup</code>	<code>\tex_begingroup:D</code> 296
297	<code>_kernel_primitive:NN \belowdisplayshortskip</code>	<code>\tex_belowdisplayshortskip:D</code> 297
298	<code>_kernel_primitive:NN \belowdisplayskip</code>	<code>\tex_belowdisplayskip:D</code> 298
299	<code>_kernel_primitive:NN \binoppenalty</code>	<code>\tex_binoppenalty:D</code> 299
300	<code>_kernel_primitive:NN \botmark</code>	<code>\tex_botmark:D</code> 300
301	<code>_kernel_primitive:NN \box</code>	<code>\tex_box:D</code> 301
302	<code>_kernel_primitive:NN \boxmaxdepth</code>	<code>\tex_boxmaxdepth:D</code> 302
303	<code>_kernel_primitive:NN \brokenpenalty</code>	<code>\tex_brokenpenalty:D</code> 303
304	<code>_kernel_primitive:NN \catcode</code>	<code>\tex_catcode:D</code> 304
305	<code>_kernel_primitive:NN \char</code>	<code>\tex_char:D</code> 305
306	<code>_kernel_primitive:NN \chardef</code>	<code>\tex_chardef:D</code> 306
307	<code>_kernel_primitive:NN \cleaders</code>	<code>\tex_cleaders:D</code> 307
308	<code>_kernel_primitive:NN \closein</code>	<code>\tex_closein:D</code> 308
309	<code>_kernel_primitive:NN \closeout</code>	<code>\tex_closeout:D</code> 309
310	<code>_kernel_primitive:NN \clubpenalty</code>	<code>\tex_clubpenalty:D</code> 310
311	<code>_kernel_primitive:NN \copy</code>	<code>\tex_copy:D</code> 311
312	<code>_kernel_primitive:NN \count</code>	<code>\tex_count:D</code> 312
313	<code>_kernel_primitive:NN \countdef</code>	<code>\tex_countdef:D</code> 313
314	<code>_kernel_primitive:NN \cr</code>	<code>\tex_cr:D</code> 314
315	<code>_kernel_primitive:NN \crcr</code>	<code>\tex_crcr:D</code> 315
316	<code>_kernel_primitive:NN \csname</code>	<code>\tex_csname:D</code> 316
317	<code>_kernel_primitive:NN \day</code>	<code>\tex_day:D</code> 317
318	<code>_kernel_primitive:NN \deadcycles</code>	<code>\tex_deadcycles:D</code> 318
319	<code>_kernel_primitive:NN \def</code>	<code>\tex_def:D</code> 319
320	<code>_kernel_primitive:NN \defaultthyphenchar</code>	<code>\tex_defaultthyphenchar:D</code> 320
321	<code>_kernel_primitive:NN \defaultskewchar</code>	<code>\tex_defaultskewchar:D</code> 321
322	<code>_kernel_primitive:NN \delcode</code>	<code>\tex_delcode:D</code> 322

323	_kernel_primitive:NN	<u>\delimiter</u>	\tex_delimiter:D	323
324	_kernel_primitive:NN	<u>\delimiterfactor</u>	\tex_delimiterfactor:D	324
325	_kernel_primitive:NN	<u>\delimitershortfall</u>	\tex_delimitershortfall:D	325
326	_kernel_primitive:NN	<u>\dimen</u>	\tex_dimen:D	326
327	_kernel_primitive:NN	<u>\dimendef</u>	\tex_dimendef:D	327
328	_kernel_primitive:NN	<u>\discretionary</u>	\tex_discretionary:D	328
329	_kernel_primitive:NN	<u>\displayindent</u>	\tex_displayindent:D	329
330	_kernel_primitive:NN	<u>\displaylimits</u>	\tex_displaylimits:D	330
331	_kernel_primitive:NN	<u>\displaystyle</u>	\tex_displaystyle:D	331
332	_kernel_primitive:NN	<u>\displaywidowpenalty</u>	\tex_displaywidowpenalty:D	332
333	_kernel_primitive:NN	<u>\displaywidth</u>	\tex_displaywidth:D	333
334	_kernel_primitive:NN	<u>\divide</u>	\tex_divide:D	334
335	_kernel_primitive:NN	<u>\doublehyphendemerits</u>	\tex_doublehyphendemerits:D	335
336	_kernel_primitive:NN	<u>\dp</u>	\tex_dp:D	336
337	_kernel_primitive:NN	<u>\dump</u>	\tex_dump:D	337
338	_kernel_primitive:NN	<u>\edef</u>	\tex_edef:D	338
339	_kernel_primitive:NN	<u>\else</u>	\tex_else:D	339
340	_kernel_primitive:NN	<u>\emergencystretch</u>	\tex_emergencystretch:D	340
341	_kernel_primitive:NN	<u>\end</u>	\tex_end:D	341
342	_kernel_primitive:NN	<u>\endcsname</u>	\tex_endcsname:D	342
343	_kernel_primitive:NN	<u>\endgroup</u>	\tex_endgroup:D	343
344	_kernel_primitive:NN	<u>\endinput</u>	\tex_endinput:D	344
345	_kernel_primitive:NN	<u>\endlinechar</u>	\tex_endlinechar:D	345
346	_kernel_primitive:NN	<u>\eqno</u>	\tex_eqno:D	346
347	_kernel_primitive:NN	<u>\errhelp</u>	\tex_errhelp:D	347
348	_kernel_primitive:NN	<u>\errmessage</u>	\tex_errmessage:D	348
349	_kernel_primitive:NN	<u>\errorcontextlines</u>	\tex_errorcontextlines:D	349
350	_kernel_primitive:NN	<u>\errorstopmode</u>	\tex_errorstopmode:D	350
351	_kernel_primitive:NN	<u>\escapechar</u>	\tex_escapechar:D	351
352	_kernel_primitive:NN	<u>\everycr</u>	\tex_everycr:D	352
353	_kernel_primitive:NN	<u>\everydisplay</u>	\tex_everydisplay:D	353
354	_kernel_primitive:NN	<u>\everyhbox</u>	\tex_everyhbox:D	354
355	_kernel_primitive:NN	<u>\everyjob</u>	\tex_everyjob:D	355
356	_kernel_primitive:NN	<u>\everymath</u>	\tex_everymath:D	356
357	_kernel_primitive:NN	<u>\everypar</u>	\tex_everypar:D	357
358	_kernel_primitive:NN	<u>\everyvbox</u>	\tex_everyvbox:D	358
359	_kernel_primitive:NN	<u>\exhyphenpenalty</u>	\tex_exhyphenpenalty:D	359
360	_kernel_primitive:NN	<u>\expandafter</u>	\tex_expandafter:D	360
361	_kernel_primitive:NN	<u>\fam</u>	\tex_fam:D	361
362	_kernel_primitive:NN	<u>\fi</u>	\tex_fi:D	362
363	_kernel_primitive:NN	<u>\finalhyphendemerits</u>	\tex_finalhyphendemerits:D	363
364	_kernel_primitive:NN	<u>\firstmark</u>	\tex_firstmark:D	364
365	_kernel_primitive:NN	<u>\floatingpenalty</u>	\tex_floatingpenalty:D	365
366	_kernel_primitive:NN	<u>\font</u>	\tex_font:D	366
367	_kernel_primitive:NN	<u>\fontdimen</u>	\tex_fontdimen:D	367
368	_kernel_primitive:NN	<u>\fontname</u>	\tex_fontname:D	368

369	_kernel_primitive:NN	<u>\futurelet</u>	\tex_futurelet:D	369
370	_kernel_primitive:NN	<u>\gdef</u>	\tex_gdef:D	370
371	_kernel_primitive:NN	<u>\global</u>	\tex_global:D	371
372	_kernel_primitive:NN	<u>\globaldefs</u>	\tex_globaldefs:D	372
373	_kernel_primitive:NN	<u>\halign</u>	\tex_halign:D	373
374	_kernel_primitive:NN	<u>\hangafter</u>	\tex_hangafter:D	374
375	_kernel_primitive:NN	<u>\hangindent</u>	\tex_hangindent:D	375
376	_kernel_primitive:NN	<u>\hbadness</u>	\tex_hbadness:D	376
377	_kernel_primitive:NN	<u>\hbox</u>	\tex_hbox:D	377
378	_kernel_primitive:NN	<u>\hfil</u>	\tex_hfil:D	378
379	_kernel_primitive:NN	<u>\hfill</u>	\tex_hfill:D	379
380	_kernel_primitive:NN	<u>\hfilneg</u>	\tex_hfilneg:D	380
381	_kernel_primitive:NN	<u>\hfuzz</u>	\tex_hfuzz:D	381
382	_kernel_primitive:NN	<u>\hoffset</u>	\tex_hoffset:D	382
383	_kernel_primitive:NN	<u>\holdinginserts</u>	\tex_holdinginserts:D	383
384	_kernel_primitive:NN	<u>\hrule</u>	\tex_hrule:D	384
385	_kernel_primitive:NN	<u>\hsize</u>	\tex_hsize:D	385
386	_kernel_primitive:NN	<u>\hskip</u>	\tex_hskip:D	386
387	_kernel_primitive:NN	<u>\hss</u>	\tex_hss:D	387
388	_kernel_primitive:NN	<u>\ht</u>	\tex_ht:D	388
389	_kernel_primitive:NN	<u>\hyphenation</u>	\tex_hyphenation:D	389
390	_kernel_primitive:NN	<u>\hyphenchar</u>	\tex_hyphenchar:D	390
391	_kernel_primitive:NN	<u>\hyphenpenalty</u>	\tex_hyphenpenalty:D	391
392	_kernel_primitive:NN	<u>\if</u>	\tex_if:D	392
393	_kernel_primitive:NN	<u>\ifcase</u>	\tex_ifcase:D	393
394	_kernel_primitive:NN	<u>\ifcat</u>	\tex_ifcat:D	394
395	_kernel_primitive:NN	<u>\ifdim</u>	\tex_ifdim:D	395
396	_kernel_primitive:NN	<u>\ifeof</u>	\tex_ifeof:D	396
397	_kernel_primitive:NN	<u>\iffalse</u>	\tex_iffalse:D	397
398	_kernel_primitive:NN	<u>\ifhbox</u>	\tex_ifhbox:D	398
399	_kernel_primitive:NN	<u>\ifhmode</u>	\tex_ifhmode:D	399
400	_kernel_primitive:NN	<u>\ifinner</u>	\tex_ifinner:D	400
401	_kernel_primitive:NN	<u>\ifmmode</u>	\tex_ifmmode:D	401
402	_kernel_primitive:NN	<u>\ifnum</u>	\tex_ifnum:D	402
403	_kernel_primitive:NN	<u>\ifodd</u>	\tex_ifodd:D	403
404	_kernel_primitive:NN	<u>\iftrue</u>	\tex_iftrue:D	404
405	_kernel_primitive:NN	<u>\ifvbox</u>	\tex_ifvbox:D	405
406	_kernel_primitive:NN	<u>\ifvmode</u>	\tex_ifvmode:D	406
407	_kernel_primitive:NN	<u>\ifvoid</u>	\tex_ifvoid:D	407
408	_kernel_primitive:NN	<u>\ifx</u>	\tex_ifx:D	408
409	_kernel_primitive:NN	<u>\ignorespaces</u>	\tex_ignorespaces:D	409
410	_kernel_primitive:NN	<u>\immediate</u>	\tex_immediate:D	410
411	_kernel_primitive:NN	<u>\indent</u>	\tex_indent:D	411
412	_kernel_primitive:NN	<u>\input</u>	\tex_input:D	412
413	_kernel_primitive:NN	<u>\inputlineno</u>	\tex_inputlineno:D	413
414	_kernel_primitive:NN	<u>\insert</u>	\tex_insert:D	414

415	<code>__kernel_primitive:NN</code>	<code>\insertpenalties</code>	<code>\tex_insertpenalties:D</code>	415
416	<code>__kernel_primitive:NN</code>	<code>\interlinepenalty</code>	<code>\tex_interlinepenalty:D</code>	416
417	<code>__kernel_primitive:NN</code>	<code>\jobname</code>	<code>\tex_jobname:D</code>	417
418	<code>__kernel_primitive:NN</code>	<code>\kern</code>	<code>\tex_kern:D</code>	418
419	<code>__kernel_primitive:NN</code>	<code>\language</code>	<code>\tex_language:D</code>	419
420	<code>__kernel_primitive:NN</code>	<code>\lastbox</code>	<code>\tex_lastbox:D</code>	420
421	<code>__kernel_primitive:NN</code>	<code>\lastkern</code>	<code>\tex_lastkern:D</code>	421
422	<code>__kernel_primitive:NN</code>	<code>\lastpenalty</code>	<code>\tex_lastpenalty:D</code>	422
423	<code>__kernel_primitive:NN</code>	<code>\lastskip</code>	<code>\tex_lastskip:D</code>	423
424	<code>__kernel_primitive:NN</code>	<code>\lccode</code>	<code>\tex_lccode:D</code>	424
425	<code>__kernel_primitive:NN</code>	<code>\leaders</code>	<code>\tex_leaders:D</code>	425
426	<code>__kernel_primitive:NN</code>	<code>\left</code>	<code>\tex_left:D</code>	426
427	<code>__kernel_primitive:NN</code>	<code>\lefthyphenmin</code>	<code>\tex_lefthyphenmin:D</code>	427
428	<code>__kernel_primitive:NN</code>	<code>\leftskip</code>	<code>\tex_leftskip:D</code>	428
429	<code>__kernel_primitive:NN</code>	<code>\leqno</code>	<code>\tex_leqno:D</code>	429
430	<code>__kernel_primitive:NN</code>	<code>\let</code>	<code>\tex_let:D</code>	430
431	<code>__kernel_primitive:NN</code>	<code>\limits</code>	<code>\tex_limits:D</code>	431
432	<code>__kernel_primitive:NN</code>	<code>\linepenalty</code>	<code>\tex_linepenalty:D</code>	432
433	<code>__kernel_primitive:NN</code>	<code>\lineskip</code>	<code>\tex_lineskip:D</code>	433
434	<code>__kernel_primitive:NN</code>	<code>\lineskiplimit</code>	<code>\tex_lineskiplimit:D</code>	434
435	<code>__kernel_primitive:NN</code>	<code>\long</code>	<code>\tex_long:D</code>	435
436	<code>__kernel_primitive:NN</code>	<code>\looseness</code>	<code>\tex_looseness:D</code>	436
437	<code>__kernel_primitive:NN</code>	<code>\lower</code>	<code>\tex_lower:D</code>	437
438	<code>__kernel_primitive:NN</code>	<code>\lowercase</code>	<code>\tex_lowercase:D</code>	438
439	<code>__kernel_primitive:NN</code>	<code>\mag</code>	<code>\tex_mag:D</code>	439
440	<code>__kernel_primitive:NN</code>	<code>\mark</code>	<code>\tex_mark:D</code>	440
441	<code>__kernel_primitive:NN</code>	<code>\mathaccent</code>	<code>\tex_mathaccent:D</code>	441
442	<code>__kernel_primitive:NN</code>	<code>\mathbin</code>	<code>\tex_mathbin:D</code>	442
443	<code>__kernel_primitive:NN</code>	<code>\mathchar</code>	<code>\tex_mathchar:D</code>	443
444	<code>__kernel_primitive:NN</code>	<code>\mathchardef</code>	<code>\tex_mathchardef:D</code>	444
445	<code>__kernel_primitive:NN</code>	<code>\mathchoice</code>	<code>\tex_mathchoice:D</code>	445
446	<code>__kernel_primitive:NN</code>	<code>\mathclose</code>	<code>\tex_mathclose:D</code>	446
447	<code>__kernel_primitive:NN</code>	<code>\mathcode</code>	<code>\tex_mathcode:D</code>	447
448	<code>__kernel_primitive:NN</code>	<code>\mathinner</code>	<code>\tex_mathinner:D</code>	448
449	<code>__kernel_primitive:NN</code>	<code>\mathop</code>	<code>\tex_mathop:D</code>	449
450	<code>__kernel_primitive:NN</code>	<code>\mathopen</code>	<code>\tex_mathopen:D</code>	450
451	<code>__kernel_primitive:NN</code>	<code>\mathord</code>	<code>\tex_mathord:D</code>	451
452	<code>__kernel_primitive:NN</code>	<code>\mathpunct</code>	<code>\tex_mathpunct:D</code>	452
453	<code>__kernel_primitive:NN</code>	<code>\mathrel</code>	<code>\tex_mathrel:D</code>	453
454	<code>__kernel_primitive:NN</code>	<code>\mathsurround</code>	<code>\tex_mathsurround:D</code>	454
455	<code>__kernel_primitive:NN</code>	<code>\maxdeadcycles</code>	<code>\tex_maxdeadcycles:D</code>	455
456	<code>__kernel_primitive:NN</code>	<code>\maxdepth</code>	<code>\tex_maxdepth:D</code>	456
457	<code>__kernel_primitive:NN</code>	<code>\meaning</code>	<code>\tex_meaning:D</code>	457
458	<code>__kernel_primitive:NN</code>	<code>\medmuskip</code>	<code>\tex_medmuskip:D</code>	458
459	<code>__kernel_primitive:NN</code>	<code>\message</code>	<code>\tex_message:D</code>	459
460	<code>__kernel_primitive:NN</code>	<code>\mkern</code>	<code>\tex_mkern:D</code>	460

461	_kernel_primitive:NN	<u>\month</u>	\tex_month:D	461
462	_kernel_primitive:NN	<u>\moveleft</u>	\tex_moveleft:D	462
463	_kernel_primitive:NN	<u>\moveright</u>	\tex_moveright:D	463
464	_kernel_primitive:NN	<u>\mskip</u>	\tex_mskip:D	464
465	_kernel_primitive:NN	<u>\multiply</u>	\tex_multiply:D	465
466	_kernel_primitive:NN	<u>\muskip</u>	\tex_muskip:D	466
467	_kernel_primitive:NN	<u>\muskipdef</u>	\tex_muskipdef:D	467
468	_kernel_primitive:NN	<u>\newlinechar</u>	\tex_newlinechar:D	468
469	_kernel_primitive:NN	<u>\noalign</u>	\tex_noalign:D	469
470	_kernel_primitive:NN	<u>\noboundary</u>	\tex_noboundary:D	470
471	_kernel_primitive:NN	<u>\noexpand</u>	\tex_noexpand:D	471
472	_kernel_primitive:NN	<u>\noindent</u>	\tex_noindent:D	472
473	_kernel_primitive:NN	<u>\nolimits</u>	\tex_nolimits:D	473
474	_kernel_primitive:NN	<u>\nonscript</u>	\tex_nonscript:D	474
475	_kernel_primitive:NN	<u>\nonstopmode</u>	\tex_nonstopmode:D	475
476	_kernel_primitive:NN	<u>\nulldelimiterspace</u>	\tex_nulldelimiterspace:D	476
477	_kernel_primitive:NN	<u>\nullfont</u>	\tex_nullfont:D	477
478	_kernel_primitive:NN	<u>\number</u>	\tex_number:D	478
479	_kernel_primitive:NN	<u>\omit</u>	\tex_omit:D	479
480	_kernel_primitive:NN	<u>\openin</u>	\tex_openin:D	480
481	_kernel_primitive:NN	<u>\openout</u>	\tex_openout:D	481
482	_kernel_primitive:NN	<u>\or</u>	\tex_or:D	482
483	_kernel_primitive:NN	<u>\outer</u>	\tex_outer:D	483
484	_kernel_primitive:NN	<u>\output</u>	\tex_output:D	484
485	_kernel_primitive:NN	<u>\outputpenalty</u>	\tex_outputpenalty:D	485
486	_kernel_primitive:NN	<u>\over</u>	\tex_over:D	486
487	_kernel_primitive:NN	<u>\overfullrule</u>	\tex_overfullrule:D	487
488	_kernel_primitive:NN	<u>\overline</u>	\tex_overline:D	488
489	_kernel_primitive:NN	<u>\overwithdelims</u>	\tex_overwithdelims:D	489
490	_kernel_primitive:NN	<u>\pagedepth</u>	\tex_pagedepth:D	490
491	_kernel_primitive:NN	<u>\pagefilllstretch</u>	\tex_pagefilllstretch:D	491
492	_kernel_primitive:NN	<u>\pagefillstretch</u>	\tex_pagefillstretch:D	492
493	_kernel_primitive:NN	<u>\pagefilstretch</u>	\tex_pagefilstretch:D	493
494	_kernel_primitive:NN	<u>\pagegoal</u>	\tex_pagegoal:D	494
495	_kernel_primitive:NN	<u>\pageshrink</u>	\tex_pageshrink:D	495
496	_kernel_primitive:NN	<u>\pagestretch</u>	\tex_pagestretch:D	496
497	_kernel_primitive:NN	<u>\pagetotal</u>	\tex_pagetotal:D	497
498	_kernel_primitive:NN	<u>\par</u>	\tex_par:D	498
499	_kernel_primitive:NN	<u>\parfillskip</u>	\tex_parfillskip:D	499
500	_kernel_primitive:NN	<u>\parindent</u>	\tex_parindent:D	500
501	_kernel_primitive:NN	<u>\parshape</u>	\tex_parshape:D	501
502	_kernel_primitive:NN	<u>\parskip</u>	\tex_parskip:D	502
503	_kernel_primitive:NN	<u>\patterns</u>	\tex_patterns:D	503
504	_kernel_primitive:NN	<u>\pausing</u>	\tex_pausing:D	504
505	_kernel_primitive:NN	<u>\penalty</u>	\tex_penalty:D	505
506	_kernel_primitive:NN	<u>\postdisplaypenalty</u>	\tex_postdisplaypenalty:D	506

507	_kernel_primitive:NN	<u>\predisplaypenalty</u>	\tex_predisplaypenalty:D	507
508	_kernel_primitive:NN	<u>\predisplaysize</u>	\tex_predisplaysize:D	508
509	_kernel_primitive:NN	<u>\pretolerance</u>	\tex_pretolerance:D	509
510	_kernel_primitive:NN	<u>\prevdepth</u>	\tex_prevdepth:D	510
511	_kernel_primitive:NN	<u>\prevgraf</u>	\tex_prevgraf:D	511
512	_kernel_primitive:NN	<u>\radical</u>	\tex_radical:D	512
513	_kernel_primitive:NN	<u>\raise</u>	\tex_raise:D	513
514	_kernel_primitive:NN	<u>\read</u>	\tex_read:D	514
515	_kernel_primitive:NN	<u>\relax</u>	\tex_relax:D	515
516	_kernel_primitive:NN	<u>\relpenalty</u>	\tex_relpenalty:D	516
517	_kernel_primitive:NN	<u>\right</u>	\tex_right:D	517
518	_kernel_primitive:NN	<u>\righthyphenmin</u>	\tex_righthyphenmin:D	518
519	_kernel_primitive:NN	<u>\rightskip</u>	\tex_rightskip:D	519
520	_kernel_primitive:NN	<u>\romannumeral</u>	\tex_romannumeral:D	520
521	_kernel_primitive:NN	<u>\scriptfont</u>	\tex_scriptfont:D	521
522	_kernel_primitive:NN	<u>\scriptscriptfont</u>	\tex_scriptscriptfont:D	522
523	_kernel_primitive:NN	<u>\scriptscriptstyle</u>	\tex_scriptscriptstyle:D	523
524	_kernel_primitive:NN	<u>\scriptspace</u>	\tex_scriptspace:D	524
525	_kernel_primitive:NN	<u>\scriptstyle</u>	\tex_scriptstyle:D	525
526	_kernel_primitive:NN	<u>\scrollmode</u>	\tex_scrollmode:D	526
527	_kernel_primitive:NN	<u>\setbox</u>	\tex_setbox:D	527
528	_kernel_primitive:NN	<u>\setlanguage</u>	\tex_setlanguage:D	528
529	_kernel_primitive:NN	<u>\sfcode</u>	\tex_sfcode:D	529
530	_kernel_primitive:NN	<u>\shipout</u>	\tex_shipout:D	530
531	_kernel_primitive:NN	<u>\show</u>	\tex_show:D	531
532	_kernel_primitive:NN	<u>\showbox</u>	\tex_showbox:D	532
533	_kernel_primitive:NN	<u>\showboxbreadth</u>	\tex_showboxbreadth:D	533
534	_kernel_primitive:NN	<u>\showboxdepth</u>	\tex_showboxdepth:D	534
535	_kernel_primitive:NN	<u>\showlists</u>	\tex_showlists:D	535
536	_kernel_primitive:NN	<u>\showthe</u>	\tex_showthe:D	536
537	_kernel_primitive:NN	<u>\skewchar</u>	\tex_skewchar:D	537
538	_kernel_primitive:NN	<u>\skip</u>	\tex_skip:D	538
539	_kernel_primitive:NN	<u>\skipdef</u>	\tex_skipdef:D	539
540	_kernel_primitive:NN	<u>\spacefactor</u>	\tex_spacefactor:D	540
541	_kernel_primitive:NN	<u>\spaceskip</u>	\tex_spaceskip:D	541
542	_kernel_primitive:NN	<u>\span</u>	\tex_span:D	542
543	_kernel_primitive:NN	<u>\special</u>	\tex_special:D	543
544	_kernel_primitive:NN	<u>\splitbotmark</u>	\tex_splitbotmark:D	544
545	_kernel_primitive:NN	<u>\splitfirstmark</u>	\tex_splitfirstmark:D	545
546	_kernel_primitive:NN	<u>\splitmaxdepth</u>	\tex_splitmaxdepth:D	546
547	_kernel_primitive:NN	<u>\splittopskip</u>	\tex_splittopskip:D	547
548	_kernel_primitive:NN	<u>\string</u>	\tex_string:D	548
549	_kernel_primitive:NN	<u>\tabskip</u>	\tex_tabskip:D	549
550	_kernel_primitive:NN	<u>\textfont</u>	\tex_textfont:D	550
551	_kernel_primitive:NN	<u>\textstyle</u>	\tex_textstyle:D	551
552	_kernel_primitive:NN	<u>\the</u>	\tex_the:D	552

553	_kernel_primitive:NN	<u>\thickmuskip</u>	\tex_thickmuskip:D	553
554	_kernel_primitive:NN	<u>\thinmuskip</u>	\tex_thinmuskip:D	554
555	_kernel_primitive:NN	<u>\time</u>	\tex_time:D	555
556	_kernel_primitive:NN	<u>\toks</u>	\tex_toks:D	556
557	_kernel_primitive:NN	<u>\toksdef</u>	\tex_toksdef:D	557
558	_kernel_primitive:NN	<u>\tolerance</u>	\tex_tolerance:D	558
559	_kernel_primitive:NN	<u>\topmark</u>	\tex_topmark:D	559
560	_kernel_primitive:NN	<u>\topskip</u>	\tex_topskip:D	560
561	_kernel_primitive:NN	<u>\tracingcommands</u>	\tex_tracingcommands:D	561
562	_kernel_primitive:NN	<u>\tracinglostchars</u>	\tex_tracinglostchars:D	562
563	_kernel_primitive:NN	<u>\tracingmacros</u>	\tex_tracingmacros:D	563
564	_kernel_primitive:NN	<u>\tracingonline</u>	\tex_tracingonline:D	564
565	_kernel_primitive:NN	<u>\tracingoutput</u>	\tex_tracingoutput:D	565
566	_kernel_primitive:NN	<u>\tracingpages</u>	\tex_tracingpages:D	566
567	_kernel_primitive:NN	<u>\tracingparagraphs</u>	\tex_tracingparagraphs:D	567
568	_kernel_primitive:NN	<u>\tracingrestores</u>	\tex_tracingrestores:D	568
569	_kernel_primitive:NN	<u>\tracingstats</u>	\tex_tracingstats:D	569
570	_kernel_primitive:NN	<u>\uccode</u>	\tex_uccode:D	570
571	_kernel_primitive:NN	<u>\uchyph</u>	\tex_uchyph:D	571
572	_kernel_primitive:NN	<u>\underline</u>	\tex_underline:D	572
573	_kernel_primitive:NN	<u>\unhbox</u>	\tex_unhbox:D	573
574	_kernel_primitive:NN	<u>\unhcopy</u>	\tex_unhcopy:D	574
575	_kernel_primitive:NN	<u>\unkern</u>	\tex_unkern:D	575
576	_kernel_primitive:NN	<u>\unpenalty</u>	\tex_unpenalty:D	576
577	_kernel_primitive:NN	<u>\unskip</u>	\tex_unskip:D	577
578	_kernel_primitive:NN	<u>\unvbox</u>	\tex_unvbox:D	578
579	_kernel_primitive:NN	<u>\unvcopy</u>	\tex_unvcopy:D	579
580	_kernel_primitive:NN	<u>\uppercase</u>	\tex_uppercase:D	580
581	_kernel_primitive:NN	<u>\vadjust</u>	\tex_vadjust:D	581
582	_kernel_primitive:NN	<u>\valign</u>	\tex_valign:D	582
583	_kernel_primitive:NN	<u>\vbadness</u>	\tex_vbadness:D	583
584	_kernel_primitive:NN	<u>\vbox</u>	\tex_vbox:D	584
585	_kernel_primitive:NN	<u>\vcenter</u>	\tex_vcenter:D	585
586	_kernel_primitive:NN	<u>\vfil</u>	\tex_vfil:D	586
587	_kernel_primitive:NN	<u>\vfill</u>	\tex_vfill:D	587
588	_kernel_primitive:NN	<u>\vfilneg</u>	\tex_vfilneg:D	588
589	_kernel_primitive:NN	<u>\vfuzz</u>	\tex_vfuzz:D	589
590	_kernel_primitive:NN	<u>\voffset</u>	\tex_voffset:D	590
591	_kernel_primitive:NN	<u>\vrule</u>	\tex_vrule:D	591
592	_kernel_primitive:NN	<u>\vsize</u>	\tex_vsize:D	592
593	_kernel_primitive:NN	<u>\vskip</u>	\tex_vskip:D	593
594	_kernel_primitive:NN	<u>\vsplit</u>	\tex_vsplit:D	594
595	_kernel_primitive:NN	<u>\vss</u>	\tex_vss:D	595
596	_kernel_primitive:NN	<u>\vtop</u>	\tex_vtop:D	596
597	_kernel_primitive:NN	<u>\wd</u>	\tex_wd:D	597
598	_kernel_primitive:NN	<u>\widowpenalty</u>	\tex_widowpenalty:D	598

599	_kernel_primitive:NN	<u>\write</u>	\tex_write:D	599
600	_kernel_primitive:NN	<u>\xdef</u>	\tex_xdef:D	600
601	_kernel_primitive:NN	<u>\xleaders</u>	\tex_xleaders:D	601
602	_kernel_primitive:NN	<u>\xspaceskip</u>	\tex_xspaceskip:D	602
603	_kernel_primitive:NN	<u>\year</u>	\tex_year:D	603
604	_kernel_primitive:NN	<u>\beginL</u>	\tex_beginL:D	604
605	_kernel_primitive:NN	<u>\beginR</u>	\tex_beginR:D	605
606	_kernel_primitive:NN	<u>\botmarks</u>	\tex_botmarks:D	606
607	_kernel_primitive:NN	<u>\clubpenalties</u>	\tex_clubpenalties:D	607
608	_kernel_primitive:NN	<u>\currentgrouplevel</u>	\tex_currentgrouplevel:D	608
609	_kernel_primitive:NN	<u>\currentgrouptype</u>	\tex_currentgrouptype:D	609
610	_kernel_primitive:NN	<u>\currentifbranch</u>	\tex_currentifbranch:D	610
611	_kernel_primitive:NN	<u>\currentiflevel</u>	\tex_currentiflevel:D	611
612	_kernel_primitive:NN	<u>\currentifttype</u>	\tex_currentifttype:D	612
613	_kernel_primitive:NN	<u>\detokenize</u>	\tex_detokenize:D	613
614	_kernel_primitive:NN	<u>\dimexpr</u>	\tex_dimexpr:D	614
615	_kernel_primitive:NN	<u>\displaywidowpenalties</u>	\tex_displaywidowpenalties:D	615
616	_kernel_primitive:NN	<u>\endL</u>	\tex_endL:D	616
617	_kernel_primitive:NN	<u>\endR</u>	\tex_endR:D	617
618	_kernel_primitive:NN	<u>\eTeXrevision</u>	\tex_eTeXrevision:D	618
619	_kernel_primitive:NN	<u>\eTeXversion</u>	\tex_eTeXversion:D	619
620	_kernel_primitive:NN	<u>\everyeof</u>	\tex_everyeof:D	620
621	_kernel_primitive:NN	<u>\firstmarks</u>	\tex_firstmarks:D	621
622	_kernel_primitive:NN	<u>\fontchardp</u>	\tex_fontchardp:D	622
623	_kernel_primitive:NN	<u>\fontcharht</u>	\tex_fontcharht:D	623
624	_kernel_primitive:NN	<u>\fontcharic</u>	\tex_fontcharic:D	624
625	_kernel_primitive:NN	<u>\fontcharwd</u>	\tex_fontcharwd:D	625
626	_kernel_primitive:NN	<u>\glueexpr</u>	\tex_glueexpr:D	626
627	_kernel_primitive:NN	<u>\glueshrink</u>	\tex_glueshrink:D	627
628	_kernel_primitive:NN	<u>\glueshrinkorder</u>	\tex_glueshrinkorder:D	628
629	_kernel_primitive:NN	<u>\gluestretch</u>	\tex_gluestretch:D	629
630	_kernel_primitive:NN	<u>\gluestretchorder</u>	\tex_gluestretchorder:D	630
631	_kernel_primitive:NN	<u>\gluetomu</u>	\tex_gluetomu:D	631
632	_kernel_primitive:NN	<u>\ifcsname</u>	\tex_ifcsname:D	632
633	_kernel_primitive:NN	<u>\ifdefined</u>	\tex_ifdefined:D	633
634	_kernel_primitive:NN	<u>\iffontchar</u>	\tex_iffontchar:D	634
635	_kernel_primitive:NN	<u>\interactionmode</u>	\tex_interactionmode:D	635
636	_kernel_primitive:NN	<u>\interlinepenalties</u>	\tex_interlinepenalties:D	636
637	_kernel_primitive:NN	<u>\lastlinefit</u>	\tex_lastlinefit:D	637
638	_kernel_primitive:NN	<u>\lastnodetype</u>	\tex_lastnodetype:D	638
639	_kernel_primitive:NN	<u>\marks</u>	\tex_marks:D	639
640	_kernel_primitive:NN	<u>\middle</u>	\tex_middle:D	640
641	_kernel_primitive:NN	<u>\muexpr</u>	\tex_muexpr:D	641
642	_kernel_primitive:NN	<u>\mutoglua</u>	\tex_mutoglua:D	642
643	_kernel_primitive:NN	<u>\numexpr</u>	\tex_numexpr:D	643
644	_kernel_primitive:NN	<u>\pagediscards</u>	\tex_pagediscards:D	644

645	_kernel_primitive:NN	<u>\parshapedimen</u>	\tex_parshapedimen:D	645
646	_kernel_primitive:NN	<u>\parshapeindent</u>	\tex_parshapeindent:D	646
647	_kernel_primitive:NN	<u>\parshapelength</u>	\tex_parshapelength:D	647
648	_kernel_primitive:NN	<u>\predisplaydirection</u>	\tex_predisplaydirection:D	648
649	_kernel_primitive:NN	<u>\protected</u>	\tex_protected:D	649
650	_kernel_primitive:NN	<u>\readline</u>	\tex_readline:D	650
651	_kernel_primitive:NN	<u>\savinghyphcodes</u>	\tex_savinghyphcodes:D	651
652	_kernel_primitive:NN	<u>\savingvdiscards</u>	\tex_savingvdiscards:D	652
653	_kernel_primitive:NN	<u>\scantokens</u>	\tex_scantokens:D	653
654	_kernel_primitive:NN	<u>\showgroups</u>	\tex_showgroups:D	654
655	_kernel_primitive:NN	<u>\showifs</u>	\tex_showifs:D	655
656	_kernel_primitive:NN	<u>\showtokens</u>	\tex_showtokens:D	656
657	_kernel_primitive:NN	<u>\splitbotmarks</u>	\tex_splitbotmarks:D	657
658	_kernel_primitive:NN	<u>\splitdiscards</u>	\tex_splitdiscards:D	658
659	_kernel_primitive:NN	<u>\splitfirstmarks</u>	\tex_splitfirstmarks:D	659
660	_kernel_primitive:NN	<u>\TeXXeTstate</u>	\tex_TeXXeTstate:D	660
661	_kernel_primitive:NN	<u>\topmarks</u>	\tex_topmarks:D	661
662	_kernel_primitive:NN	<u>\tracingassigns</u>	\tex_tracingassigns:D	662
663	_kernel_primitive:NN	<u>\tracinggroups</u>	\tex_tracinggroups:D	663
664	_kernel_primitive:NN	<u>\tracingifs</u>	\tex_tracingifs:D	664
665	_kernel_primitive:NN	<u>\tracingnesting</u>	\tex_tracingnesting:D	665
666	_kernel_primitive:NN	<u>\tracingscantokens</u>	\tex_tracingscantokens:D	666
667	_kernel_primitive:NN	<u>\unexpanded</u>	\tex_unexpanded:D	667
668	_kernel_primitive:NN	<u>\unless</u>	\tex_unless:D	668
669	_kernel_primitive:NN	<u>\widowpenalties</u>	\tex_widowpenalties:D	669
670	_kernel_primitive:NN	<u>\pdfannot</u>	\tex_pdfannot:D	670
671	_kernel_primitive:NN	<u>\pdfcatalog</u>	\tex_pdfcatalog:D	671
672	_kernel_primitive:NN	<u>\pdfcompresslevel</u>	\tex_pdfcompresslevel:D	672
673	_kernel_primitive:NN	<u>\pdfcolorstack</u>	\tex_pdfcolorstack:D	673
674	_kernel_primitive:NN	<u>\pdfcolorstackinit</u>	\tex_pdfcolorstackinit:D	674
675	_kernel_primitive:NN	<u>\pdfdecimaldigits</u>	\tex_pdfdecimaldigits:D	675
676	_kernel_primitive:NN	<u>\pdfdest</u>	\tex_pdfdest:D	676
677	_kernel_primitive:NN	<u>\pdfdestmargin</u>	\tex_pdfdestmargin:D	677
678	_kernel_primitive:NN	<u>\pdfendlink</u>	\tex_pdfendlink:D	678
679	_kernel_primitive:NN	<u>\pdfendthread</u>	\tex_pdfendthread:D	679
680	_kernel_primitive:NN	<u>\pdffakespace</u>	\tex_pdffakespace:D	680
681	_kernel_primitive:NN	<u>\pdffontattr</u>	\tex_pdffontattr:D	681
682	_kernel_primitive:NN	<u>\pdffontname</u>	\tex_pdffontname:D	682
683	_kernel_primitive:NN	<u>\pdffontobjnum</u>	\tex_pdffontobjnum:D	683
684	_kernel_primitive:NN	<u>\pdfgamma</u>	\tex_pdfgamma:D	684
685	_kernel_primitive:NN	<u>\pdfgentounicode</u>	\tex_pdfgentounicode:D	685
686	_kernel_primitive:NN	<u>\pdfglyphtounicode</u>	\tex_pdfglyphtounicode:D	686
687	_kernel_primitive:NN	<u>\pdfhorigin</u>	\tex_pdfhorigin:D	687
688	_kernel_primitive:NN	<u>\pdfimageapplygamma</u>	\tex_pdfimageapplygamma:D	688
689	_kernel_primitive:NN	<u>\pdfimagegamma</u>	\tex_pdfimagegamma:D	689
690	_kernel_primitive:NN	<u>\pdfimagehicolor</u>	\tex_pdfimagehicolor:D	690

691	_kernel_primitive:NN	<u>\pdfimageresolution</u>	\tex_pdfimageresolution:D	691
692	_kernel_primitive:NN	<u>\pdfincludechars</u>	\tex_pdfincludechars:D	692
693	_kernel_primitive:NN	<u>\pdfinclusioncopyfonts</u>	\tex_pdfinclusioncopyfonts:D	693
694	_kernel_primitive:NN	<u>\pdfinclusionerrorlevel</u>		694
695		\tex_pdfinclusionerrorlevel:D		695
696	_kernel_primitive:NN	<u>\pdfinfo</u>	\tex_pdfinfo:D	696
697	_kernel_primitive:NN	<u>\pdfinfoomitdate</u>	\tex_pdfinfoomitdate:D	697
698	_kernel_primitive:NN	<u>\pdfinterwordsoff</u>	\tex_pdfinterwordsoff:D	698
699	_kernel_primitive:NN	<u>\pdfinterwordspaceon</u>	\tex_pdfinterwordspaceon:D	699
700	_kernel_primitive:NN	<u>\pdflastannot</u>	\tex_pdflastannot:D	700
701	_kernel_primitive:NN	<u>\pdflastlink</u>	\tex_pdflastlink:D	701
702	_kernel_primitive:NN	<u>\pdflastobj</u>	\tex_pdflastobj:D	702
703	_kernel_primitive:NN	<u>\pdflastxform</u>	\tex_pdflastxform:D	703
704	_kernel_primitive:NN	<u>\pdflastximage</u>	\tex_pdflastximage:D	704
705	_kernel_primitive:NN	<u>\pdflastximagecolordepth</u>		705
706		\tex_pdflastximagecolordepth:D		706
707	_kernel_primitive:NN	<u>\pdflastximagepages</u>	\tex_pdflastximagepages:D	707
708	_kernel_primitive:NN	<u>\pdflinkmargin</u>	\tex_pdflinkmargin:D	708
709	_kernel_primitive:NN	<u>\pdfliteral</u>	\tex_pdfliteral:D	709
710	_kernel_primitive:NN	<u>\pdfmapfile</u>	\tex_pdfmapfile:D	710
711	_kernel_primitive:NN	<u>\pdfmapline</u>	\tex_pdfmapline:D	711
712	_kernel_primitive:NN	<u>\pdfmajorversion</u>	\tex_pdfmajorversion:D	712
713	_kernel_primitive:NN	<u>\pdfminorversion</u>	\tex_pdfminorversion:D	713
714	_kernel_primitive:NN	<u>\pdfnames</u>	\tex_pdfnames:D	714
715	_kernel_primitive:NN	<u>\pdfnobuiltintounicode</u>	\tex_pdfnobuiltintounicode:D	715
716	_kernel_primitive:NN	<u>\pdfobj</u>	\tex_pdfobj:D	716
717	_kernel_primitive:NN	<u>\pdfobjcompresslevel</u>	\tex_pdfobjcompresslevel:D	717
718	_kernel_primitive:NN	<u>\pdfomitcharset</u>	\tex_pdfomitcharset:D	718
719	_kernel_primitive:NN	<u>\pdfoutline</u>	\tex_pdfoutline:D	719
720	_kernel_primitive:NN	<u>\pdfoutput</u>	\tex_pdfoutput:D	720
721	_kernel_primitive:NN	<u>\pdfpageattr</u>	\tex_pdfpageattr:D	721
722	_kernel_primitive:NN	<u>\pdfpagebox</u>	\tex_pdfpagebox:D	722
723	_kernel_primitive:NN	<u>\pdfpageref</u>	\tex_pdfpageref:D	723
724	_kernel_primitive:NN	<u>\pdfpageresources</u>	\tex_pdfpageresources:D	724
725	_kernel_primitive:NN	<u>\pdfpagesattr</u>	\tex_pdfpagesattr:D	725
726	_kernel_primitive:NN	<u>\pdfptexuseunderscore</u>	\tex_pdfptexuseunderscore:D	726
727	_kernel_primitive:NN	<u>\pdfrefobj</u>	\tex_pdfrefobj:D	727
728	_kernel_primitive:NN	<u>\pdfrefxform</u>	\tex_pdfrefxform:D	728
729	_kernel_primitive:NN	<u>\pdfrefximage</u>	\tex_pdfrefximage:D	729
730	_kernel_primitive:NN	<u>\pdfrestore</u>	\tex_pdfrestore:D	730
731	_kernel_primitive:NN	<u>\pdfretval</u>	\tex_pdfretval:D	731
732	_kernel_primitive:NN	<u>\pdfrunninglinkoff</u>	\tex_pdfrunninglinkoff:D	732
733	_kernel_primitive:NN	<u>\pdfrunninglinkon</u>	\tex_pdfrunninglinkon:D	733
734	_kernel_primitive:NN	<u>\pdfsave</u>	\tex_pdfsave:D	734
735	_kernel_primitive:NN	<u>\pdfsetmatrix</u>	\tex_pdfsetmatrix:D	735
736	_kernel_primitive:NN	<u>\pdfstartlink</u>	\tex_pdfstartlink:D	736

737	<code>__kernel_primitive:NN</code>	<code>\pdfstartthread</code>	<code>\tex_pdfstartthread:D</code>	737
738	<code>__kernel_primitive:NN</code>	<code>\pdfsuppressptexinfo</code>	<code>\tex_pdfsuppressptexinfo:D</code>	738
739	<code>__kernel_primitive:NN</code>	<code>\pdfsuppresswarningdupdest</code>		739
740		<code>\tex_pdfsuppresswarningdupdest:D</code>		740
741	<code>__kernel_primitive:NN</code>	<code>\pdfsuppresswarningdupmap</code>		741
742		<code>\tex_pdfsuppresswarningdupmap:D</code>		742
743	<code>__kernel_primitive:NN</code>	<code>\pdfsuppresswarningpagegroup</code>		743
744		<code>\tex_pdfsuppresswarningpagegroup:D</code>		744
745	<code>__kernel_primitive:NN</code>	<code>\pdfthread</code>	<code>\tex_pdfthread:D</code>	745
746	<code>__kernel_primitive:NN</code>	<code>\pdfthreadmargin</code>	<code>\tex_pdfthreadmargin:D</code>	746
747	<code>__kernel_primitive:NN</code>	<code>\pdftrailer</code>	<code>\tex_pdftrailer:D</code>	747
748	<code>__kernel_primitive:NN</code>	<code>\pdftrailerid</code>	<code>\tex_pdftrailerid:D</code>	748
749	<code>__kernel_primitive:NN</code>	<code>\pdfuniqueresname</code>	<code>\tex_pdfuniqueresname:D</code>	749
750	<code>__kernel_primitive:NN</code>	<code>\pdfvorigin</code>	<code>\tex_pdfvorigin:D</code>	750
751	<code>__kernel_primitive:NN</code>	<code>\pdfxform</code>	<code>\tex_pdfxform:D</code>	751
752	<code>__kernel_primitive:NN</code>	<code>\pdfxformname</code>	<code>\tex_pdfxformname:D</code>	752
753	<code>__kernel_primitive:NN</code>	<code>\pdfximage</code>	<code>\tex_pdfximage:D</code>	753
754	<code>__kernel_primitive:NN</code>	<code>\pdfximagebbox</code>	<code>\tex_pdfximagebbox:D</code>	754
755	<code>__kernel_primitive:NN</code>	<code>\ifpdfabsdim</code>	<code>\tex_ifabsdim:D</code>	755
756	<code>__kernel_primitive:NN</code>	<code>\ifpdfabsnum</code>	<code>\tex_ifabsnum:D</code>	756
757	<code>__kernel_primitive:NN</code>	<code>\ifpdfprimitive</code>	<code>\tex_ifprimitive:D</code>	757
758	<code>__kernel_primitive:NN</code>	<code>\pdfadjustinterwordglue</code>		758
759		<code>\tex_adjustinterwordglue:D</code>		759
760	<code>__kernel_primitive:NN</code>	<code>\pdfadjustspacing</code>	<code>\tex_adjustspacing:D</code>	760
761	<code>__kernel_primitive:NN</code>	<code>\pdfappendkern</code>	<code>\tex_appendkern:D</code>	761
762	<code>__kernel_primitive:NN</code>	<code>\pdfcopyfont</code>	<code>\tex_copyfont:D</code>	762
763	<code>__kernel_primitive:NN</code>	<code>\pdfcreationdate</code>	<code>\tex_creationdate:D</code>	763
764	<code>__kernel_primitive:NN</code>	<code>\pdfdraftmode</code>	<code>\tex_draftmode:D</code>	764
765	<code>__kernel_primitive:NN</code>	<code>\pdfeachlinedepth</code>	<code>\tex_eachlinedepth:D</code>	765
766	<code>__kernel_primitive:NN</code>	<code>\pdfeachlineheight</code>	<code>\tex_eachlineheight:D</code>	766
767	<code>__kernel_primitive:NN</code>	<code>\pdfelapsedtime</code>	<code>\tex_elapsedtime:D</code>	767
768	<code>__kernel_primitive:NN</code>	<code>\pdfescapehex</code>	<code>\tex_escapehex:D</code>	768
769	<code>__kernel_primitive:NN</code>	<code>\pdfescapename</code>	<code>\tex_escapename:D</code>	769
770	<code>__kernel_primitive:NN</code>	<code>\pdfescapestring</code>	<code>\tex_escapestring:D</code>	770
771	<code>__kernel_primitive:NN</code>	<code>\pdffirstlineheight</code>	<code>\tex_firstlineheight:D</code>	771
772	<code>__kernel_primitive:NN</code>	<code>\pdffontexpand</code>	<code>\tex_fontexpand:D</code>	772
773	<code>__kernel_primitive:NN</code>	<code>\pdffontsize</code>	<code>\tex_fontsize:D</code>	773
774	<code>__kernel_primitive:NN</code>	<code>\pdfignoreddimen</code>	<code>\tex_ignoreddimen:D</code>	774
775	<code>__kernel_primitive:NN</code>	<code>\pdfinsertht</code>	<code>\tex_insertht:D</code>	775
776	<code>__kernel_primitive:NN</code>	<code>\pdflastlinedepth</code>	<code>\tex_lastlinedepth:D</code>	776
777	<code>__kernel_primitive:NN</code>	<code>\pdflastmatch</code>	<code>\tex_lastmatch:D</code>	777
778	<code>__kernel_primitive:NN</code>	<code>\pdflastxpos</code>	<code>\tex_lastxpos:D</code>	778
779	<code>__kernel_primitive:NN</code>	<code>\pdflastypos</code>	<code>\tex_lastypos:D</code>	779
780	<code>__kernel_primitive:NN</code>	<code>\pdfmatch</code>	<code>\tex_match:D</code>	780
781	<code>__kernel_primitive:NN</code>	<code>\pdfnoligatures</code>	<code>\tex_noligatures:D</code>	781
782	<code>__kernel_primitive:NN</code>	<code>\pdfnormaldeviate</code>	<code>\tex_normaldeviate:D</code>	782

783	_kernel_primitive:NN	<u>\pdfpageheight</u>	\tex_pageheight:D	783
784	_kernel_primitive:NN	<u>\pdfpagewidth</u>	\tex_pagewidth:D	784
785	_kernel_primitive:NN	<u>\pdfpkmode</u>	\tex_pkmode:D	785
786	_kernel_primitive:NN	<u>\pdfpkresolution</u>	\tex_pkresolution:D	786
787	_kernel_primitive:NN	<u>\pdfprimitive</u>	\tex_primitive:D	787
788	_kernel_primitive:NN	<u>\pdfprependkern</u>	\tex_prependkern:D	788
789	_kernel_primitive:NN	<u>\pdfprotrudechars</u>	\tex_protrudechars:D	789
790	_kernel_primitive:NN	<u>\pdfpxdimen</u>	\tex_pxdimen:D	790
791	_kernel_primitive:NN	<u>\pdfrandomseed</u>	\tex_randomseed:D	791
792	_kernel_primitive:NN	<u>\pdfresettimer</u>	\tex_resettimer:D	792
793	_kernel_primitive:NN	<u>\pdfsavepos</u>	\tex_savepos:D	793
794	_kernel_primitive:NN	<u>\pdfsetrandomseed</u>	\tex_setrandomseed:D	794
795	_kernel_primitive:NN	<u>\pdfshellescape</u>	\tex_shellescape:D	795
796	_kernel_primitive:NN	<u>\pdftracingfonts</u>	\tex_tracingfonts:D	796
797	_kernel_primitive:NN	<u>\pdfunescapehex</u>	\tex_unescapehex:D	797
798	_kernel_primitive:NN	<u>\pdfuniformdeviate</u>	\tex_uniformdeviate:D	798
799	_kernel_primitive:NN	<u>\pdftexbanner</u>	\tex_pdftexbanner:D	799
800	_kernel_primitive:NN	<u>\pdftexrevision</u>	\tex_pdftexrevision:D	800
801	_kernel_primitive:NN	<u>\pdftexversion</u>	\tex_pdftexversion:D	801
802	_kernel_primitive:NN	<u>\efcode</u>	\tex_efcode:D	802
803	_kernel_primitive:NN	<u>\ifincsname</u>	\tex_ifincsname:D	803
804	_kernel_primitive:NN	<u>\knaccode</u>	\tex_knaccode:D	804
805	_kernel_primitive:NN	<u>\knbccode</u>	\tex_knbccode:D	805
806	_kernel_primitive:NN	<u>\knbscode</u>	\tex_knbscode:D	806
807	_kernel_primitive:NN	<u>\leftmarginkern</u>	\tex_leftmarginkern:D	807
808	_kernel_primitive:NN	<u>\letterspacefont</u>	\tex_letterspacefont:D	808
809	_kernel_primitive:NN	<u>\lpcode</u>	\tex_lpcode:D	809
810	_kernel_primitive:NN	<u>\quitvmode</u>	\tex_quitvmode:D	810
811	_kernel_primitive:NN	<u>\rightmarginkern</u>	\tex_rightmarginkern:D	811
812	_kernel_primitive:NN	<u>\rpcode</u>	\tex_rpcode:D	812
813	_kernel_primitive:NN	<u>\shbscode</u>	\tex_shbscode:D	813
814	_kernel_primitive:NN	<u>\stbscode</u>	\tex_stbscode:D	814
815	_kernel_primitive:NN	<u>\synctex</u>	\tex_synctex:D	815
816	_kernel_primitive:NN	<u>\tagcode</u>	\tex_tagcode:D	816
817	\tex_long:D \tex_def:D	\use_ii:nn #1#2 {#2}		817
818	\tex_long:D \tex_def:D	\use_none:n #1 { }		818
819	\tex_long:D \tex_def:D	_kernel_primitive:NN #1#2		819
820		{		820
821		\tex_ifdefined:D #1		821
822		\tex_expandafter:D \use_ii:nn		822
823		\tex_fi:D		823
824		\use_none:n { \tex_global:D \tex_let:D #2 #1 }		824
825		}		825
826	_kernel_primitive:NN	<u>\pdfstrcmp</u>	\tex_strcmp:D	826
827	_kernel_primitive:NN	<u>\pdffilesize</u>	\tex_filesize:D	827
828	_kernel_primitive:NN	<u>\pdfmdfivesum</u>	\tex_mdfivesum:D	828

829	_kernel_primitive:NN	<u>\pdffilemoddate</u>	\tex_filemoddate:D	829
830	_kernel_primitive:NN	<u>\pdffiledump</u>	\tex_filedump:D	830
831	_kernel_primitive:NN	<u>\suppressfontnotfounderror</u>		831
832		\tex_suppressfontnotfounderror:D		832
833	_kernel_primitive:NN	<u>\XeTeXcharclass</u>	\tex_XeTeXcharclass:D	833
834	_kernel_primitive:NN	<u>\XeTeXcharglyph</u>	\tex_XeTeXcharglyph:D	834
835	_kernel_primitive:NN	<u>\XeTeXcountfeatures</u>	\tex_XeTeXcountfeatures:D	835
836	_kernel_primitive:NN	<u>\XeTeXcountglyphs</u>	\tex_XeTeXcountglyphs:D	836
837	_kernel_primitive:NN	<u>\XeTeXcountselectors</u>	\tex_XeTeXcountselectors:D	837
838	_kernel_primitive:NN	<u>\XeTeXcountvariations</u>	\tex_XeTeXcountvariations:D	838
839	_kernel_primitive:NN	<u>\XeTeXdefaultencoding</u>	\tex_XeTeXdefaultencoding:D	839
840	_kernel_primitive:NN	<u>\XeTeXdashbreakstate</u>	\tex_XeTeXdashbreakstate:D	840
841	_kernel_primitive:NN	<u>\XeTeXfeaturecode</u>	\tex_XeTeXfeaturecode:D	841
842	_kernel_primitive:NN	<u>\XeTeXfeaturename</u>	\tex_XeTeXfeaturename:D	842
843	_kernel_primitive:NN	<u>\XeTeXfindfeaturebyname</u>		843
844		\tex_XeTeXfindfeaturebyname:D		844
845	_kernel_primitive:NN	<u>\XeTeXfindselectorbyname</u>		845
846		\tex_XeTeXfindselectorbyname:D		846
847	_kernel_primitive:NN	<u>\XeTeXfindvariationbyname</u>		847
848		\tex_XeTeXfindvariationbyname:D		848
849	_kernel_primitive:NN	<u>\XeTeXfirstfontchar</u>	\tex_XeTeXfirstfontchar:D	849
850	_kernel_primitive:NN	<u>\XeTeXfonttype</u>	\tex_XeTeXfonttype:D	850
851	_kernel_primitive:NN	<u>\XeTeXgenerateactualtext</u>		851
852		\tex_XeTeXgenerateactualtext:D		852
853	_kernel_primitive:NN	<u>\XeTeXglyph</u>	\tex_XeTeXglyph:D	853
854	_kernel_primitive:NN	<u>\XeTeXglyphbounds</u>	\tex_XeTeXglyphbounds:D	854
855	_kernel_primitive:NN	<u>\XeTeXglyphindex</u>	\tex_XeTeXglyphindex:D	855
856	_kernel_primitive:NN	<u>\XeTeXglyphname</u>	\tex_XeTeXglyphname:D	856
857	_kernel_primitive:NN	<u>\XeTeXinputencoding</u>	\tex_XeTeXinputencoding:D	857
858	_kernel_primitive:NN	<u>\XeTeXinputnormalization</u>		858
859		\tex_XeTeXinputnormalization:D		859
860	_kernel_primitive:NN	<u>\XeTeXinterchartokenstate</u>		860
861		\tex_XeTeXinterchartokenstate:D		861
862	_kernel_primitive:NN	<u>\XeTeXinterchartoks</u>	\tex_XeTeXinterchartoks:D	862
863	_kernel_primitive:NN	<u>\XeTeXisdefaultselector</u>		863
864		\tex_XeTeXisdefaultselector:D		864
865	_kernel_primitive:NN	<u>\XeTeXisexclusivefeature</u>		865
866		\tex_XeTeXisexclusivefeature:D		866
867	_kernel_primitive:NN	<u>\XeTeXlastfontchar</u>	\tex_XeTeXlastfontchar:D	867
868	_kernel_primitive:NN	<u>\XeTeXlinebreakskip</u>	\tex_XeTeXlinebreakskip:D	868
869	_kernel_primitive:NN	<u>\XeTeXlinebreaklocale</u>	\tex_XeTeXlinebreaklocale:D	869
870	_kernel_primitive:NN	<u>\XeTeXlinebreakpenalty</u>	\tex_XeTeXlinebreakpenalty:D	870
871	_kernel_primitive:NN	<u>\XeTeXOTcountfeatures</u>	\tex_XeTeXOTcountfeatures:D	871
872	_kernel_primitive:NN	<u>\XeTeXOTcountlanguages</u>	\tex_XeTeXOTcountlanguages:D	872
873	_kernel_primitive:NN	<u>\XeTeXOTcountscripts</u>	\tex_XeTeXOTcountscripts:D	873
874	_kernel_primitive:NN	<u>\XeTeXOTfeaturetag</u>	\tex_XeTeXOTfeaturetag:D	874

875	_kernel_primitive:NN	<u>\XeTeXOTlanguage</u> tag	\tex_XeTeXOTlanguage:D	875
876	_kernel_primitive:NN	<u>\XeTeXOTscript</u> tag	\tex_XeTeXOTscript:D	876
877	_kernel_primitive:NN	<u>\XeTeXpdf</u> file	\tex_XeTeXpdf:D	877
878	_kernel_primitive:NN	<u>\XeTeXpdf</u> pagecount	\tex_XeTeXpdfpagecount:D	878
879	_kernel_primitive:NN	<u>\XeTeXpic</u> file	\tex_XeTeXpic:D	879
880	_kernel_primitive:NN	<u>\XeTeXre</u> vision	\tex_XeTeXreversion:D	880
881	_kernel_primitive:NN	<u>\XeTeXse</u> lectorname	\tex_XeTeXselectorname:D	881
882	_kernel_primitive:NN	<u>\XeTeXtr</u> acingfonts	\tex_XeTeXtracingfonts:D	882
883	_kernel_primitive:NN	<u>\XeTeXup</u> wardsmode	\tex_XeTeXupwardsmode:D	883
884	_kernel_primitive:NN	<u>\XeTeXuse</u> glyphmetrics	\tex_XeTeXuseglyphmetrics:D	884
885	_kernel_primitive:NN	<u>\XeTeXv</u> ariation	\tex_XeTeXvariation:D	885
886	_kernel_primitive:NN	<u>\XeTeXv</u> ariationdefault	\tex_XeTeXvariationdefault:D	886
887	_kernel_primitive:NN	<u>\XeTeXv</u> ariationmax	\tex_XeTeXvariationmax:D	887
888	_kernel_primitive:NN	<u>\XeTeXv</u> ariationmin	\tex_XeTeXvariationmin:D	888
889	_kernel_primitive:NN	<u>\XeTeXv</u> ariationname	\tex_XeTeXvariationname:D	889
890	_kernel_primitive:NN	<u>\XeTeXv</u> ersion	\tex_XeTeXversion:D	890
891	_kernel_primitive:NN	<u>\XeTeXse</u> lectorcode	\tex_XeTeXselectorcode:D	891
892	_kernel_primitive:NN	<u>\XeTeXinter</u> wordspaceshaping		892
893		\tex_XeTeXinterwordspaceshaping:D		893
894	_kernel_primitive:NN	<u>\XeTeXhy</u> phenatablelength		894
895		\tex_XeTeXhyphenatablelength:D		895
896	_kernel_primitive:NN	<u>\creation</u> date	\tex_creationdate:D	896
897	_kernel_primitive:NN	<u>\elapsed</u> time	\tex_elapsedtime:D	897
898	_kernel_primitive:NN	<u>\file</u> dump	\tex_filedump:D	898
899	_kernel_primitive:NN	<u>\file</u> moddate	\tex_filemoddate:D	899
900	_kernel_primitive:NN	<u>\file</u> size	\tex_filesize:D	900
901	_kernel_primitive:NN	<u>\mdfiv</u> esum	\tex_mdfivesum:D	901
902	_kernel_primitive:NN	<u>\if</u> primitive	\tex_ifprimitive:D	902
903	_kernel_primitive:NN	<u>\p</u> rimitive	\tex_primitive:D	903
904	_kernel_primitive:NN	<u>\reset</u> timer	\tex_resettimer:D	904
905	_kernel_primitive:NN	<u>\shel</u> lescape	\tex_shellescape:D	905
906	_kernel_primitive:NN	<u>\XeTeXp</u> rotrudechars	\tex_protrudechars:D	906
907	_kernel_primitive:NN	<u>\align</u> mark	\tex_alignmark:D	907
908	_kernel_primitive:NN	<u>\align</u> tab	\tex_aligntab:D	908
909	_kernel_primitive:NN	<u>\attrib</u> ute	\tex_attribute:D	909
910	_kernel_primitive:NN	<u>\attrib</u> utedef	\tex_attributedef:D	910
911	_kernel_primitive:NN	<u>\automatic</u> discretionary		911
912		\tex_automaticdiscretionary:D		912
913	_kernel_primitive:NN	<u>\automatic</u> hyphenmode	\tex_automatichyphenmode:D	913
914	_kernel_primitive:NN	<u>\automatic</u> hyphenpenalty		914
915		\tex_automatichyphenpenalty:D		915
916	_kernel_primitive:NN	<u>\begin</u> csname	\tex_begincsname:D	916
917	_kernel_primitive:NN	<u>\body</u> dir	\tex_bodydir:D	917
918	_kernel_primitive:NN	<u>\body</u> direction	\tex_bodydirection:D	918
919	_kernel_primitive:NN	<u>\bound</u> ary	\tex_boundary:D	919
920	_kernel_primitive:NN	<u>\box</u> dir	\tex_boxdir:D	920

921	_kernel_primitive:NN	<u>\boxdirection</u>	\tex_boxdirection:D	921
922	_kernel_primitive:NN	<u>\breakafterdirmode</u>	\tex_breakafterdirmode:D	922
923	_kernel_primitive:NN	<u>\catcodetable</u>	\tex_catcodetable:D	923
924	_kernel_primitive:NN	<u>\clearmarks</u>	\tex_clearmarks:D	924
925	_kernel_primitive:NN	<u>\crampeddisplaystyle</u>	\tex_crampeddisplaystyle:D	925
926	_kernel_primitive:NN	<u>\crampedscriptscriptstyle</u>		926
927		\tex_crampedscriptscriptstyle:D		927
928	_kernel_primitive:NN	<u>\crampedscriptstyle</u>	\tex_crampedscriptstyle:D	928
929	_kernel_primitive:NN	<u>\crampedtextstyle</u>	\tex_crampedtextstyle:D	929
930	_kernel_primitive:NN	<u>\csstring</u>	\tex_csstring:D	930
931	_kernel_primitive:NN	<u>\deferred</u>	\tex_deferred:D	931
932	_kernel_primitive:NN	<u>\discretionaryligaturemode</u>		932
933		\tex_discretionaryligaturemode:D		933
934	_kernel_primitive:NN	<u>\directlua</u>	\tex_directlua:D	934
935	_kernel_primitive:NN	<u>\dviextension</u>	\tex_dviextension:D	935
936	_kernel_primitive:NN	<u>\dvifedback</u>	\tex_dvifedback:D	936
937	_kernel_primitive:NN	<u>\dvivariable</u>	\tex_dvivariable:D	937
938	_kernel_primitive:NN	<u>\eTeXglueshrinkorder</u>	\tex_eTeXglueshrinkorder:D	938
939	_kernel_primitive:NN	<u>\eTeXgluestretchorder</u>	\tex_eTeXgluestretchorder:D	939
940	_kernel_primitive:NN	<u>\endlocalcontrol</u>	\tex_endlocalcontrol:D	940
941	_kernel_primitive:NN	<u>\etoksapp</u>	\tex_etoksapp:D	941
942	_kernel_primitive:NN	<u>\etokspre</u>	\tex_etokspre:D	942
943	_kernel_primitive:NN	<u>\exceptionpenalty</u>	\tex_exceptionpenalty:D	943
944	_kernel_primitive:NN	<u>\exhyphenchar</u>	\tex_exhyphenchar:D	944
945	_kernel_primitive:NN	<u>\explicitthyphenpenalty</u>	\tex_explicitthyphenpenalty:D	945
946	_kernel_primitive:NN	<u>\expanded</u>	\tex_expanded:D	946
947	_kernel_primitive:NN	<u>\explicitdiscretionary</u>	\tex_explicitdiscretionary:D	947
948	_kernel_primitive:NN	<u>\firstvalidlanguage</u>	\tex_firstvalidlanguage:D	948
949	_kernel_primitive:NN	<u>\fontid</u>	\tex_fontid:D	949
950	_kernel_primitive:NN	<u>\formatname</u>	\tex_formatname:D	950
951	_kernel_primitive:NN	<u>\hjcode</u>	\tex_hjcode:D	951
952	_kernel_primitive:NN	<u>\hpack</u>	\tex_hpack:D	952
953	_kernel_primitive:NN	<u>\hyphenationbounds</u>	\tex_hyphenationbounds:D	953
954	_kernel_primitive:NN	<u>\hyphenationmin</u>	\tex_hyphenationmin:D	954
955	_kernel_primitive:NN	<u>\hyphenpenaltymode</u>	\tex_hyphenpenaltymode:D	955
956	_kernel_primitive:NN	<u>\gleaders</u>	\tex_gleaders:D	956
957	_kernel_primitive:NN	<u>\glet</u>	\tex_glet:D	957
958	_kernel_primitive:NN	<u>\glyphdimensionsmode</u>	\tex_glyphdimensionsmode:D	958
959	_kernel_primitive:NN	<u>\gtoksapp</u>	\tex_gtoksapp:D	959
960	_kernel_primitive:NN	<u>\gtokspre</u>	\tex_gtokspre:D	960
961	_kernel_primitive:NN	<u>\ifcondition</u>	\tex_ifcondition:D	961
962	_kernel_primitive:NN	<u>\immediateassigned</u>	\tex_immediateassigned:D	962
963	_kernel_primitive:NN	<u>\immediateassignment</u>	\tex_immediateassignment:D	963
964	_kernel_primitive:NN	<u>\initcatcodetable</u>	\tex_initcatcodetable:D	964
965	_kernel_primitive:NN	<u>\lastnamedcs</u>	\tex_lastnamedcs:D	965
966	_kernel_primitive:NN	<u>\latelua</u>	\tex_latelua:D	966

967	__kernel_primitive:NN	<u>\lateluafunction</u>	\tex_lateluafunction:D	967
968	__kernel_primitive:NN	<u>\leftghost</u>	\tex_leftghost:D	968
969	__kernel_primitive:NN	<u>\letcharcode</u>	\tex_letcharcode:D	969
970	__kernel_primitive:NN	<u>\linedir</u>	\tex_linedir:D	970
971	__kernel_primitive:NN	<u>\linedirection</u>	\tex_linedirection:D	971
972	__kernel_primitive:NN	<u>\localbrokenpenalty</u>	\tex_localbrokenpenalty:D	972
973	__kernel_primitive:NN	<u>\localinterlinepenalty</u>	\tex_localinterlinepenalty:D	973
974	__kernel_primitive:NN	<u>\luabytecode</u>	\tex_luabytecode:D	974
975	__kernel_primitive:NN	<u>\luabytecodecall</u>	\tex_luabytecodecall:D	975
976	__kernel_primitive:NN	<u>\luacopyinputnodes</u>	\tex_luacopyinputnodes:D	976
977	__kernel_primitive:NN	<u>\luadef</u>	\tex_luadef:D	977
978	__kernel_primitive:NN	<u>\localleftbox</u>	\tex_localleftbox:D	978
979	__kernel_primitive:NN	<u>\localrightbox</u>	\tex_localrightbox:D	979
980	__kernel_primitive:NN	<u>\luaescapestring</u>	\tex_luaescapestring:D	980
981	__kernel_primitive:NN	<u>\luafunction</u>	\tex_luafunction:D	981
982	__kernel_primitive:NN	<u>\luafunctioncall</u>	\tex_luafunctioncall:D	982
983	__kernel_primitive:NN	<u>\luatexbanner</u>	\tex_luatexbanner:D	983
984	__kernel_primitive:NN	<u>\luatexrevision</u>	\tex_luatexrevision:D	984
985	__kernel_primitive:NN	<u>\luatexversion</u>	\tex_luatexversion:D	985
986	__kernel_primitive:NN	<u>\mathdefaultsmode</u>	\tex_mathdefaultsmode:D	986
987	__kernel_primitive:NN	<u>\mathdelimitersmode</u>	\tex_mathdelimitersmode:D	987
988	__kernel_primitive:NN	<u>\mathdir</u>	\tex_mathdir:D	988
989	__kernel_primitive:NN	<u>\mathdirection</u>	\tex_mathdirection:D	989
990	__kernel_primitive:NN	<u>\mathdisplayskipmode</u>	\tex_mathdisplayskipmode:D	990
991	__kernel_primitive:NN	<u>\matheqdirmode</u>	\tex_matheqdirmode:D	991
992	__kernel_primitive:NN	<u>\matheqnogapstep</u>	\tex_matheqnogapstep:D	992
993	__kernel_primitive:NN	<u>\mathemptydisplaymode</u>	\tex_mathemptydisplaymode:D	993
994	__kernel_primitive:NN	<u>\mathflattenmode</u>	\tex_mathflattenmode:D	994
995	__kernel_primitive:NN	<u>\mathitalicsmode</u>	\tex_mathitalicsmode:D	995
996	__kernel_primitive:NN	<u>\mathnolimitsmode</u>	\tex_mathnolimitsmode:D	996
997	__kernel_primitive:NN	<u>\mathoption</u>	\tex_mathoption:D	997
998	__kernel_primitive:NN	<u>\mathpenaltiesmode</u>	\tex_mathpenaltiesmode:D	998
999	__kernel_primitive:NN	<u>\mathrulesfam</u>	\tex_mathrulesfam:D	999
1000	__kernel_primitive:NN	<u>\mathscriptsmode</u>	\tex_mathscriptsmode:D	1000
1001	__kernel_primitive:NN	<u>\mathscriptboxmode</u>	\tex_mathscriptboxmode:D	1001
1002	__kernel_primitive:NN	<u>\mathscriptcharmode</u>	\tex_mathscriptcharmode:D	1002
1003	__kernel_primitive:NN	<u>\mathstyle</u>	\tex_mathstyle:D	1003
1004	__kernel_primitive:NN	<u>\mathsurroundmode</u>	\tex_mathsurroundmode:D	1004
1005	__kernel_primitive:NN	<u>\mathsurroundskip</u>	\tex_mathsurroundskip:D	1005
1006	__kernel_primitive:NN	<u>\nohrule</u>	\tex_nohrule:D	1006
1007	__kernel_primitive:NN	<u>\nokerns</u>	\tex_nokerns:D	1007
1008	__kernel_primitive:NN	<u>\noligs</u>	\tex_noligs:D	1008
1009	__kernel_primitive:NN	<u>\nospaces</u>	\tex_nospaces:D	1009
1010	__kernel_primitive:NN	<u>\novrule</u>	\tex_novrule:D	1010
1011	__kernel_primitive:NN	<u>\outputbox</u>	\tex_outputbox:D	1011
1012	__kernel_primitive:NN	<u>\pagebottomoffset</u>	\tex_pagebottomoffset:D	1012

1013	_kernel_primitive:NN	<u>\pagedir</u>	\tex_pagedir:D	1013
1014	_kernel_primitive:NN	<u>\pagedirection</u>	\tex_pagedirection:D	1014
1015	_kernel_primitive:NN	<u>\pageleftoffset</u>	\tex_pageleftoffset:D	1015
1016	_kernel_primitive:NN	<u>\pagerightoffset</u>	\tex_pagerightoffset:D	1016
1017	_kernel_primitive:NN	<u>\pagetopoffset</u>	\tex_pagetopoffset:D	1017
1018	_kernel_primitive:NN	<u>\pardir</u>	\tex_pardir:D	1018
1019	_kernel_primitive:NN	<u>\pardirection</u>	\tex_pardirection:D	1019
1020	_kernel_primitive:NN	<u>\pdfextension</u>	\tex_pdfextension:D	1020
1021	_kernel_primitive:NN	<u>\pdffeedback</u>	\tex_pdffeedback:D	1021
1022	_kernel_primitive:NN	<u>\pdfvariable</u>	\tex_pdfvariable:D	1022
1023	_kernel_primitive:NN	<u>\postexhyphenchar</u>	\tex_postexhyphenchar:D	1023
1024	_kernel_primitive:NN	<u>\posthyphenchar</u>	\tex_posthyphenchar:D	1024
1025	_kernel_primitive:NN	<u>\prebinoppenalty</u>	\tex_prebinoppenalty:D	1025
1026	_kernel_primitive:NN	<u>\predisplaygapfactor</u>	\tex_predisplaygapfactor:D	1026
1027	_kernel_primitive:NN	<u>\preexhyphenchar</u>	\tex_preexhyphenchar:D	1027
1028	_kernel_primitive:NN	<u>\prehyphenchar</u>	\tex_prehyphenchar:D	1028
1029	_kernel_primitive:NN	<u>\prerelpenalty</u>	\tex_prerelpenalty:D	1029
1030	_kernel_primitive:NN	<u>\protrusionboundary</u>	\tex_protrusionboundary:D	1030
1031	_kernel_primitive:NN	<u>\rightghost</u>	\tex_rightghost:D	1031
1032	_kernel_primitive:NN	<u>\savecatcodetable</u>	\tex_savecatcodetable:D	1032
1033	_kernel_primitive:NN	<u>\scantextokens</u>	\tex_scantextokens:D	1033
1034	_kernel_primitive:NN	<u>\setfontid</u>	\tex_setfontid:D	1034
1035	_kernel_primitive:NN	<u>\shapemode</u>	\tex_shapemode:D	1035
1036	_kernel_primitive:NN	<u>\suppressifcsnameerror</u>	\tex_suppressifcsnameerror:D	1036
1037	_kernel_primitive:NN	<u>\suppresslongerror</u>	\tex_suppresslongerror:D	1037
1038	_kernel_primitive:NN	<u>\suppressmathparerror</u>	\tex_suppressmathparerror:D	1038
1039	_kernel_primitive:NN	<u>\suppressoutererror</u>	\tex_suppressoutererror:D	1039
1040	_kernel_primitive:NN	<u>\suppressprimitiveerror</u>		1040
1041		\tex_suppressprimitiveerror:D		1041
1042	_kernel_primitive:NN	<u>\textdir</u>	\tex_textdir:D	1042
1043	_kernel_primitive:NN	<u>\textdirection</u>	\tex_textdirection:D	1043
1044	_kernel_primitive:NN	<u>\toksapp</u>	\tex_toksapp:D	1044
1045	_kernel_primitive:NN	<u>\tokspre</u>	\tex_tokspre:D	1045
1046	_kernel_primitive:NN	<u>\tpack</u>	\tex_tpack:D	1046
1047	_kernel_primitive:NN	<u>\variablefam</u>	\tex_variablefam:D	1047
1048	_kernel_primitive:NN	<u>\vpack</u>	\tex_vpack:D	1048
1049	_kernel_primitive:NN	<u>\wordboundary</u>	\tex_wordboundary:D	1049
1050	_kernel_primitive:NN	<u>\xtoksapp</u>	\tex_xtoksapp:D	1050
1051	_kernel_primitive:NN	<u>\xtokspre</u>	\tex_xtokspre:D	1051
1052	_kernel_primitive:NN	<u>\adjustspacing</u>	\tex_adjustspacing:D	1052
1053	_kernel_primitive:NN	<u>\copyfont</u>	\tex_copyfont:D	1053
1054	_kernel_primitive:NN	<u>\draftmode</u>	\tex_draftmode:D	1054
1055	_kernel_primitive:NN	<u>\expandglyphsinfont</u>	\tex_fontexpand:D	1055
1056	_kernel_primitive:NN	<u>\ifabsdim</u>	\tex_ifabsdim:D	1056
1057	_kernel_primitive:NN	<u>\ifabsnum</u>	\tex_ifabsnum:D	1057
1058	_kernel_primitive:NN	<u>\ignoreligaturesinfont</u>	\tex_ignoreligaturesinfont:D	1058

1059	_kernel_primitive:NN	<u>\insertht</u>	\tex_insertht:D	1059
1060	_kernel_primitive:NN	<u>\lastsavedboxresourceindex</u>		1060
1061		\tex_pdflastxform:D		1061
1062	_kernel_primitive:NN	<u>\lastsavedimageresourceindex</u>		1062
1063		\tex_pdflastximage:D		1063
1064	_kernel_primitive:NN	<u>\lastsavedimageresourcepages</u>		1064
1065		\tex_pdflastximagepages:D		1065
1066	_kernel_primitive:NN	<u>\lastxpos</u>	\tex_lastxpos:D	1066
1067	_kernel_primitive:NN	<u>\lastypos</u>	\tex_lastypos:D	1067
1068	_kernel_primitive:NN	<u>\normaldeviate</u>	\tex_normaldeviate:D	1068
1069	_kernel_primitive:NN	<u>\outputmode</u>	\tex_pdfoutput:D	1069
1070	_kernel_primitive:NN	<u>\pageheight</u>	\tex_pageheight:D	1070
1071	_kernel_primitive:NN	<u>\pagewidth</u>	\tex_pagewidth:D	1071
1072	_kernel_primitive:NN	<u>\protrudechars</u>	\tex_protrudechars:D	1072
1073	_kernel_primitive:NN	<u>\pxdimen</u>	\tex_pxdimen:D	1073
1074	_kernel_primitive:NN	<u>\randomseed</u>	\tex_randomseed:D	1074
1075	_kernel_primitive:NN	<u>\useboxresource</u>	\tex_pdfrefxform:D	1075
1076	_kernel_primitive:NN	<u>\useimageresource</u>	\tex_pdfrefximage:D	1076
1077	_kernel_primitive:NN	<u>\savepos</u>	\tex_savepos:D	1077
1078	_kernel_primitive:NN	<u>\saveboxresource</u>	\tex_pdfxform:D	1078
1079	_kernel_primitive:NN	<u>\saveimageresource</u>	\tex_pdfximage:D	1079
1080	_kernel_primitive:NN	<u>\setrandomseed</u>	\tex_setrandomseed:D	1080
1081	_kernel_primitive:NN	<u>\tracingfonts</u>	\tex_tracingfonts:D	1081
1082	_kernel_primitive:NN	<u>\uniformdeviate</u>	\tex_uniformdeviate:D	1082
1083	_kernel_primitive:NN	<u>\Uchar</u>	\tex_Uchar:D	1083
1084	_kernel_primitive:NN	<u>\Ucharcat</u>	\tex_Ucharcat:D	1084
1085	_kernel_primitive:NN	<u>\Udelcode</u>	\tex_Udelcode:D	1085
1086	_kernel_primitive:NN	<u>\Udelcodenum</u>	\tex_Udelcodenum:D	1086
1087	_kernel_primitive:NN	<u>\Udelimiter</u>	\tex_Udelimiter:D	1087
1088	_kernel_primitive:NN	<u>\Udelimiterover</u>	\tex_Udelimiterover:D	1088
1089	_kernel_primitive:NN	<u>\Udelimiterunder</u>	\tex_Udelimiterunder:D	1089
1090	_kernel_primitive:NN	<u>\Uhextensible</u>	\tex_Uhextensible:D	1090
1091	_kernel_primitive:NN	<u>\Uleft</u>	\tex_Uleft:D	1091
1092	_kernel_primitive:NN	<u>\Umathaccent</u>	\tex_Umathaccent:D	1092
1093	_kernel_primitive:NN	<u>\Umathaxis</u>	\tex_Umathaxis:D	1093
1094	_kernel_primitive:NN	<u>\Umathbinbinspacing</u>	\tex_Umathbinbinspacing:D	1094
1095	_kernel_primitive:NN	<u>\Umathbinclosespacing</u>	\tex_Umathbinclosespacing:D	1095
1096	_kernel_primitive:NN	<u>\Umathbininnerspacing</u>	\tex_Umathbininnerspacing:D	1096
1097	_kernel_primitive:NN	<u>\Umathbinopenspacing</u>	\tex_Umathbinopenspacing:D	1097
1098	_kernel_primitive:NN	<u>\Umathbinopspacing</u>	\tex_Umathbinopspacing:D	1098
1099	_kernel_primitive:NN	<u>\Umathbinordspacing</u>	\tex_Umathbinordspacing:D	1099
1100	_kernel_primitive:NN	<u>\Umathbinpunctspacing</u>	\tex_Umathbinpunctspacing:D	1100
1101	_kernel_primitive:NN	<u>\Umathbinrelspacing</u>	\tex_Umathbinrelspacing:D	1101
1102	_kernel_primitive:NN	<u>\Umathchar</u>	\tex_Umathchar:D	1102
1103	_kernel_primitive:NN	<u>\Umathcharclass</u>	\tex_Umathcharclass:D	1103
1104	_kernel_primitive:NN	<u>\Umathchardef</u>	\tex_Umathchardef:D	1104

1105	_kernel_primitive:NN	<u>\Umathcharfam</u>	\tex_Umathcharfam:D	1105
1106	_kernel_primitive:NN	<u>\Umathcharnum</u>	\tex_Umathcharnum:D	1106
1107	_kernel_primitive:NN	<u>\Umathcharnumdef</u>	\tex_Umathcharnumdef:D	1107
1108	_kernel_primitive:NN	<u>\Umathcharslot</u>	\tex_Umathcharslot:D	1108
1109	_kernel_primitive:NN	<u>\Umathclosebinspacing</u>	\tex_Umathclosebinspacing:D	1109
1110	_kernel_primitive:NN	<u>\Umathcloseclosespacing</u>		1110
1111		\tex_Umathcloseclosespacing:D		1111
1112	_kernel_primitive:NN	<u>\Umathcloseinnerspacing</u>		1112
1113		\tex_Umathcloseinnerspacing:D		1113
1114	_kernel_primitive:NN	<u>\Umathcloseopenspacing</u>	\tex_Umathcloseopenspacing:D	1114
1115	_kernel_primitive:NN	<u>\Umathcloseopspacing</u>	\tex_Umathcloseopspacing:D	1115
1116	_kernel_primitive:NN	<u>\Umathcloseordspacing</u>	\tex_Umathcloseordspacing:D	1116
1117	_kernel_primitive:NN	<u>\Umathclosepunctspacing</u>		1117
1118		\tex_Umathclosepunctspacing:D		1118
1119	_kernel_primitive:NN	<u>\Umathcloserelspacing</u>	\tex_Umathcloserelspacing:D	1119
1120	_kernel_primitive:NN	<u>\Umathcode</u>	\tex_Umathcode:D	1120
1121	_kernel_primitive:NN	<u>\Umathcodenum</u>	\tex_Umathcodenum:D	1121
1122	_kernel_primitive:NN	<u>\Umathconnectoroverlapmin</u>		1122
1123		\tex_Umathconnectoroverlapmin:D		1123
1124	_kernel_primitive:NN	<u>\Umathfractiondelsize</u>	\tex_Umathfractiondelsize:D	1124
1125	_kernel_primitive:NN	<u>\Umathfractiondenomdown</u>		1125
1126		\tex_Umathfractiondenomdown:D		1126
1127	_kernel_primitive:NN	<u>\Umathfractiondenomvgap</u>		1127
1128		\tex_Umathfractiondenomvgap:D		1128
1129	_kernel_primitive:NN	<u>\Umathfractionnumup</u>	\tex_Umathfractionnumup:D	1129
1130	_kernel_primitive:NN	<u>\Umathfractionnumvgap</u>	\tex_Umathfractionnumvgap:D	1130
1131	_kernel_primitive:NN	<u>\Umathfractionrule</u>	\tex_Umathfractionrule:D	1131
1132	_kernel_primitive:NN	<u>\Umathinnerbinspacing</u>	\tex_Umathinnerbinspacing:D	1132
1133	_kernel_primitive:NN	<u>\Umathinnerclosespacing</u>		1133
1134		\tex_Umathinnerclosespacing:D		1134
1135	_kernel_primitive:NN	<u>\Umathinnerinnerspacing</u>		1135
1136		\tex_Umathinnerinnerspacing:D		1136
1137	_kernel_primitive:NN	<u>\Umathinneropenspacing</u>	\tex_Umathinneropenspacing:D	1137
1138	_kernel_primitive:NN	<u>\Umathinneropspacing</u>	\tex_Umathinneropspacing:D	1138
1139	_kernel_primitive:NN	<u>\Umathinnerordspacing</u>	\tex_Umathinnerordspacing:D	1139
1140	_kernel_primitive:NN	<u>\Umathinnerpunctspacing</u>		1140
1141		\tex_Umathinnerpunctspacing:D		1141
1142	_kernel_primitive:NN	<u>\Umathinnerrelspacing</u>	\tex_Umathinnerrelspacing:D	1142
1143	_kernel_primitive:NN	<u>\Umathlimitabovebgap</u>	\tex_Umathlimitabovebgap:D	1143
1144	_kernel_primitive:NN	<u>\Umathlimitabovekern</u>	\tex_Umathlimitabovekern:D	1144
1145	_kernel_primitive:NN	<u>\Umathlimitabovevgap</u>	\tex_Umathlimitabovevgap:D	1145
1146	_kernel_primitive:NN	<u>\Umathlimitbelowbgap</u>	\tex_Umathlimitbelowbgap:D	1146
1147	_kernel_primitive:NN	<u>\Umathlimitbelowkern</u>	\tex_Umathlimitbelowkern:D	1147
1148	_kernel_primitive:NN	<u>\Umathlimitbelowvgap</u>	\tex_Umathlimitbelowvgap:D	1148
1149	_kernel_primitive:NN	<u>\Umathnolimitsubfactor</u>	\tex_Umathnolimitsubfactor:D	1149
1150	_kernel_primitive:NN	<u>\Umathnolimitsupfactor</u>	\tex_Umathnolimitsupfactor:D	1150

1151	_kernel_primitive:NN	<u>\Umathopbinspacing</u>	\tex_Umathopbinspacing:D	1151
1152	_kernel_primitive:NN	<u>\Umathopclosespacing</u>	\tex_Umathopclosespacing:D	1152
1153	_kernel_primitive:NN	<u>\Umathopenbinspacing</u>	\tex_Umathopenbinspacing:D	1153
1154	_kernel_primitive:NN	<u>\Umathopenclosespacing</u>	\tex_Umathopenclosespacing:D	1154
1155	_kernel_primitive:NN	<u>\Umathopeninnerspacing</u>	\tex_Umathopeninnerspacing:D	1155
1156	_kernel_primitive:NN	<u>\Umathopenopenspacing</u>	\tex_Umathopenopenspacing:D	1156
1157	_kernel_primitive:NN	<u>\Umathopenopspacing</u>	\tex_Umathopenopspacing:D	1157
1158	_kernel_primitive:NN	<u>\Umathopenordspacing</u>	\tex_Umathopenordspacing:D	1158
1159	_kernel_primitive:NN	<u>\Umathopenpunctspacing</u>	\tex_Umathopenpunctspacing:D	1159
1160	_kernel_primitive:NN	<u>\Umathopenrelspacing</u>	\tex_Umathopenrelspacing:D	1160
1161	_kernel_primitive:NN	<u>\Umathoperatorsize</u>	\tex_Umathoperatorsize:D	1161
1162	_kernel_primitive:NN	<u>\Umathopinnerspacing</u>	\tex_Umathopinnerspacing:D	1162
1163	_kernel_primitive:NN	<u>\Umathopopenspacing</u>	\tex_Umathopopenspacing:D	1163
1164	_kernel_primitive:NN	<u>\Umathopopspacing</u>	\tex_Umathopopspacing:D	1164
1165	_kernel_primitive:NN	<u>\Umathopordspacing</u>	\tex_Umathopordspacing:D	1165
1166	_kernel_primitive:NN	<u>\Umathoppunctspacing</u>	\tex_Umathoppunctspacing:D	1166
1167	_kernel_primitive:NN	<u>\Umathoprelspacing</u>	\tex_Umathoprelspacing:D	1167
1168	_kernel_primitive:NN	<u>\Umathordbinspacing</u>	\tex_Umathordbinspacing:D	1168
1169	_kernel_primitive:NN	<u>\Umathordclosespacing</u>	\tex_Umathordclosespacing:D	1169
1170	_kernel_primitive:NN	<u>\Umathordinnerspacing</u>	\tex_Umathordinnerspacing:D	1170
1171	_kernel_primitive:NN	<u>\Umathordopenspacing</u>	\tex_Umathordopenspacing:D	1171
1172	_kernel_primitive:NN	<u>\Umathordopspacing</u>	\tex_Umathordopspacing:D	1172
1173	_kernel_primitive:NN	<u>\Umathordordspacing</u>	\tex_Umathordordspacing:D	1173
1174	_kernel_primitive:NN	<u>\Umathordpunctspacing</u>	\tex_Umathordpunctspacing:D	1174
1175	_kernel_primitive:NN	<u>\Umathordrelspacing</u>	\tex_Umathordrelspacing:D	1175
1176	_kernel_primitive:NN	<u>\Umathoverbarkern</u>	\tex_Umathoverbarkern:D	1176
1177	_kernel_primitive:NN	<u>\Umathoverbarrule</u>	\tex_Umathoverbarrule:D	1177
1178	_kernel_primitive:NN	<u>\Umathoverbarvgap</u>	\tex_Umathoverbarvgap:D	1178
1179	_kernel_primitive:NN	<u>\Umathoverdelimiterbgap</u>		1179
1180		\tex_Umathoverdelimiterbgap:D		1180
1181	_kernel_primitive:NN	<u>\Umathoverdelimitervgap</u>		1181
1182		\tex_Umathoverdelimitervgap:D		1182
1183	_kernel_primitive:NN	<u>\Umathpunctbinspacing</u>	\tex_Umathpunctbinspacing:D	1183
1184	_kernel_primitive:NN	<u>\Umathpunctclosespacing</u>		1184
1185		\tex_Umathpunctclosespacing:D		1185
1186	_kernel_primitive:NN	<u>\Umathpunctinnerspacing</u>		1186
1187		\tex_Umathpunctinnerspacing:D		1187
1188	_kernel_primitive:NN	<u>\Umathpunctopenspacing</u>	\tex_Umathpunctopenspacing:D	1188
1189	_kernel_primitive:NN	<u>\Umathpunctopspacing</u>	\tex_Umathpunctopspacing:D	1189
1190	_kernel_primitive:NN	<u>\Umathpunctordspacing</u>	\tex_Umathpunctordspacing:D	1190
1191	_kernel_primitive:NN	<u>\Umathpunctpunctspacing</u>		1191
1192		\tex_Umathpunctpunctspacing:D		1192
1193	_kernel_primitive:NN	<u>\Umathpunctrelspacing</u>	\tex_Umathpunctrelspacing:D	1193
1194	_kernel_primitive:NN	<u>\Umathquad</u>	\tex_Umathquad:D	1194
1195	_kernel_primitive:NN	<u>\Umathradicaldegreeafter</u>		1195
1196		\tex_Umathradicaldegreeafter:D		1196

1197	<code>__kernel_primitive:NN</code>	<code>\Umathradicaldegreebefore</code>		1197
1198		<code>\tex_Umathradicaldegreebefore:D</code>		1198
1199	<code>__kernel_primitive:NN</code>	<code>\Umathradicaldegreeraise</code>		1199
1200		<code>\tex_Umathradicaldegreeraise:D</code>		1200
1201	<code>__kernel_primitive:NN</code>	<code>\Umathradicalkern</code>	<code>\tex_Umathradicalkern:D</code>	1201
1202	<code>__kernel_primitive:NN</code>	<code>\Umathradicalrule</code>	<code>\tex_Umathradicalrule:D</code>	1202
1203	<code>__kernel_primitive:NN</code>	<code>\Umathradicalvgap</code>	<code>\tex_Umathradicalvgap:D</code>	1203
1204	<code>__kernel_primitive:NN</code>	<code>\Umathrelbinspacing</code>	<code>\tex_Umathrelbinspacing:D</code>	1204
1205	<code>__kernel_primitive:NN</code>	<code>\Umathrelclosespacing</code>	<code>\tex_Umathrelclosespacing:D</code>	1205
1206	<code>__kernel_primitive:NN</code>	<code>\Umathrelinnerspacing</code>	<code>\tex_Umathrelinnerspacing:D</code>	1206
1207	<code>__kernel_primitive:NN</code>	<code>\Umathrelopenspacing</code>	<code>\tex_Umathrelopenspacing:D</code>	1207
1208	<code>__kernel_primitive:NN</code>	<code>\Umathrelopspacing</code>	<code>\tex_Umathrelopspacing:D</code>	1208
1209	<code>__kernel_primitive:NN</code>	<code>\Umathrelordspacing</code>	<code>\tex_Umathrelordspacing:D</code>	1209
1210	<code>__kernel_primitive:NN</code>	<code>\Umathrelpunctspacing</code>	<code>\tex_Umathrelpunctspacing:D</code>	1210
1211	<code>__kernel_primitive:NN</code>	<code>\Umathrelrelspacing</code>	<code>\tex_Umathrelrelspacing:D</code>	1211
1212	<code>__kernel_primitive:NN</code>	<code>\Umathskewedfractionhgap</code>		1212
1213		<code>\tex_Umathskewedfractionhgap:D</code>		1213
1214	<code>__kernel_primitive:NN</code>	<code>\Umathskewedfractionvgap</code>		1214
1215		<code>\tex_Umathskewedfractionvgap:D</code>		1215
1216	<code>__kernel_primitive:NN</code>	<code>\Umathspaceafterscript</code>	<code>\tex_Umathspaceafterscript:D</code>	1216
1217	<code>__kernel_primitive:NN</code>	<code>\Umathstackdenomdown</code>	<code>\tex_Umathstackdenomdown:D</code>	1217
1218	<code>__kernel_primitive:NN</code>	<code>\Umathstacknumup</code>	<code>\tex_Umathstacknumup:D</code>	1218
1219	<code>__kernel_primitive:NN</code>	<code>\Umathstackvgap</code>	<code>\tex_Umathstackvgap:D</code>	1219
1220	<code>__kernel_primitive:NN</code>	<code>\Umathsubshiftdown</code>	<code>\tex_Umathsubshiftdown:D</code>	1220
1221	<code>__kernel_primitive:NN</code>	<code>\Umathsubshiftdrop</code>	<code>\tex_Umathsubshiftdrop:D</code>	1221
1222	<code>__kernel_primitive:NN</code>	<code>\Umathsubsupshiftdown</code>	<code>\tex_Umathsubsupshiftdown:D</code>	1222
1223	<code>__kernel_primitive:NN</code>	<code>\Umathsubsupvgap</code>	<code>\tex_Umathsubsupvgap:D</code>	1223
1224	<code>__kernel_primitive:NN</code>	<code>\Umathsubtopmax</code>	<code>\tex_Umathsubtopmax:D</code>	1224
1225	<code>__kernel_primitive:NN</code>	<code>\Umathsupbottommin</code>	<code>\tex_Umathsupbottommin:D</code>	1225
1226	<code>__kernel_primitive:NN</code>	<code>\Umathsupshiftdrop</code>	<code>\tex_Umathsupshiftdrop:D</code>	1226
1227	<code>__kernel_primitive:NN</code>	<code>\Umathsupshiftup</code>	<code>\tex_Umathsupshiftup:D</code>	1227
1228	<code>__kernel_primitive:NN</code>	<code>\Umathsupsubbottommax</code>	<code>\tex_Umathsupsubbottommax:D</code>	1228
1229	<code>__kernel_primitive:NN</code>	<code>\Umathunderbarkern</code>	<code>\tex_Umathunderbarkern:D</code>	1229
1230	<code>__kernel_primitive:NN</code>	<code>\Umathunderbarrule</code>	<code>\tex_Umathunderbarrule:D</code>	1230
1231	<code>__kernel_primitive:NN</code>	<code>\Umathunderbarvgap</code>	<code>\tex_Umathunderbarvgap:D</code>	1231
1232	<code>__kernel_primitive:NN</code>	<code>\Umathunderdelimiterbgap</code>		1232
1233		<code>\tex_Umathunderdelimiterbgap:D</code>		1233
1234	<code>__kernel_primitive:NN</code>	<code>\Umathunderdelimitervgap</code>		1234
1235		<code>\tex_Umathunderdelimitervgap:D</code>		1235
1236	<code>__kernel_primitive:NN</code>	<code>\Umiddle</code>	<code>\tex_Umiddle:D</code>	1236
1237	<code>__kernel_primitive:NN</code>	<code>\Unosubscript</code>	<code>\tex_Unosubscript:D</code>	1237
1238	<code>__kernel_primitive:NN</code>	<code>\Unosuperscript</code>	<code>\tex_Unosuperscript:D</code>	1238
1239	<code>__kernel_primitive:NN</code>	<code>\Uoverdelimiter</code>	<code>\tex_Uoverdelimiter:D</code>	1239
1240	<code>__kernel_primitive:NN</code>	<code>\Uradical</code>	<code>\tex_Uradical:D</code>	1240
1241	<code>__kernel_primitive:NN</code>	<code>\Uright</code>	<code>\tex_Uright:D</code>	1241
1242	<code>__kernel_primitive:NN</code>	<code>\Uroot</code>	<code>\tex_Uroot:D</code>	1242

1243	__kernel_primitive:NN	<u>\Uskewed</u>	\tex_Uskewed:D	1243
1244	__kernel_primitive:NN	<u>\Uskewedwithdelims</u>	\tex_Uskewedwithdelims:D	1244
1245	__kernel_primitive:NN	<u>\Ustack</u>	\tex_Ustack:D	1245
1246	__kernel_primitive:NN	<u>\Ustartdisplaymath</u>	\tex_Ustartdisplaymath:D	1246
1247	__kernel_primitive:NN	<u>\Ustartmath</u>	\tex_Ustartmath:D	1247
1248	__kernel_primitive:NN	<u>\Ustopdisplaymath</u>	\tex_Ustopdisplaymath:D	1248
1249	__kernel_primitive:NN	<u>\Ustopmath</u>	\tex_Ustopmath:D	1249
1250	__kernel_primitive:NN	<u>\Usubscript</u>	\tex_Usubscript:D	1250
1251	__kernel_primitive:NN	<u>\Usuperscript</u>	\tex_Usuperscript:D	1251
1252	__kernel_primitive:NN	<u>\Uunderdelimater</u>	\tex_Uunderdelimater:D	1252
1253	__kernel_primitive:NN	<u>\Uvextensible</u>	\tex_Uvextensible:D	1253
1254	__kernel_primitive:NN	<u>\autospacing</u>	\tex_autospacing:D	1254
1255	__kernel_primitive:NN	<u>\autoxspacing</u>	\tex_autoxspacing:D	1255
1256	__kernel_primitive:NN	<u>\currentcjktoken</u>	\tex_currentcjktoken:D	1256
1257	__kernel_primitive:NN	<u>\currentspacingmode</u>	\tex_currentspacingmode:D	1257
1258	__kernel_primitive:NN	<u>\currentxspacingmode</u>	\tex_currentxspacingmode:D	1258
1259	__kernel_primitive:NN	<u>\disinhibitglue</u>	\tex_disinhibitglue:D	1259
1260	__kernel_primitive:NN	<u>\dtou</u>	\tex_dtou:D	1260
1261	__kernel_primitive:NN	<u>\epTeXinputencoding</u>	\tex_epTeXinputencoding:D	1261
1262	__kernel_primitive:NN	<u>\epTeXversion</u>	\tex_epTeXversion:D	1262
1263	__kernel_primitive:NN	<u>\euc</u>	\tex_euc:D	1263
1264	__kernel_primitive:NN	<u>\hfi</u>	\tex_hfi:D	1264
1265	__kernel_primitive:NN	<u>\ifdbox</u>	\tex_ifdbox:D	1265
1266	__kernel_primitive:NN	<u>\ifddir</u>	\tex_ifddir:D	1266
1267	__kernel_primitive:NN	<u>\ifjfont</u>	\tex_ifjfont:D	1267
1268	__kernel_primitive:NN	<u>\ifmbox</u>	\tex_ifmbox:D	1268
1269	__kernel_primitive:NN	<u>\ifmdir</u>	\tex_ifmdir:D	1269
1270	__kernel_primitive:NN	<u>\iftbox</u>	\tex_iftbox:D	1270
1271	__kernel_primitive:NN	<u>\iftfont</u>	\tex_iftfont:D	1271
1272	__kernel_primitive:NN	<u>\iftdir</u>	\tex_iftdir:D	1272
1273	__kernel_primitive:NN	<u>\ifybox</u>	\tex_ifybox:D	1273
1274	__kernel_primitive:NN	<u>\ifydir</u>	\tex_ifydir:D	1274
1275	__kernel_primitive:NN	<u>\inhibitglue</u>	\tex_inhibitglue:D	1275
1276	__kernel_primitive:NN	<u>\inhibitxspcode</u>	\tex_inhibitxspcode:D	1276
1277	__kernel_primitive:NN	<u>\jcharwidowpenalty</u>	\tex_jcharwidowpenalty:D	1277
1278	__kernel_primitive:NN	<u>\jfam</u>	\tex_jfam:D	1278
1279	__kernel_primitive:NN	<u>\jfont</u>	\tex_jfont:D	1279
1280	__kernel_primitive:NN	<u>\jis</u>	\tex_jis:D	1280
1281	__kernel_primitive:NN	<u>\kanjiskip</u>	\tex_kanjiskip:D	1281
1282	__kernel_primitive:NN	<u>\kansuji</u>	\tex_kansuji:D	1282
1283	__kernel_primitive:NN	<u>\kansujichar</u>	\tex_kansujichar:D	1283
1284	__kernel_primitive:NN	<u>\kcatcode</u>	\tex_kcatcode:D	1284
1285	__kernel_primitive:NN	<u>\kuten</u>	\tex_kuten:D	1285
1286	__kernel_primitive:NN	<u>\lastnodechar</u>	\tex_lastnodechar:D	1286
1287	__kernel_primitive:NN	<u>\lastnodefont</u>	\tex_lastnodefont:D	1287
1288	__kernel_primitive:NN	<u>\lastnodesubtype</u>	\tex_lastnodesubtype:D	1288

1289	_kernel_primitive:NN	<u>\noautospaceing</u>	\tex_noautospaceing:D	1289
1290	_kernel_primitive:NN	<u>\noautoxspaceing</u>	\tex_noautoxspaceing:D	1290
1291	_kernel_primitive:NN	<u>\pagefistretch</u>	\tex_pagefistretch:D	1291
1292	_kernel_primitive:NN	<u>\postbreakpenalty</u>	\tex_postbreakpenalty:D	1292
1293	_kernel_primitive:NN	<u>\prebreakpenalty</u>	\tex_prebreakpenalty:D	1293
1294	_kernel_primitive:NN	<u>\ptexfontname</u>	\tex_ptexfontname:D	1294
1295	_kernel_primitive:NN	<u>\ptexlineendmode</u>	\tex_lineendmode:D	1295
1296	_kernel_primitive:NN	<u>\ptexminorversion</u>	\tex_ptexminorversion:D	1296
1297	_kernel_primitive:NN	<u>\ptexrevision</u>	\tex_ptexrevision:D	1297
1298	_kernel_primitive:NN	<u>\ptextracingfonts</u>	\tex_ptextracingfonts:D	1298
1299	_kernel_primitive:NN	<u>\ptexversion</u>	\tex_ptexversion:D	1299
1300	_kernel_primitive:NN	<u>\readpapersizespecial</u>	\tex_readpapersizespecial:D	1300
1301	_kernel_primitive:NN	<u>\scriptbaselineshiftfactor</u>		1301
1302		\tex_scriptbaselineshiftfactor:D		1302
1303	_kernel_primitive:NN	<u>\scriptscriptbaselineshiftfactor</u>		1303
1304		\tex_scriptscriptbaselineshiftfactor:D		1304
1305	_kernel_primitive:NN	<u>\showmode</u>	\tex_showmode:D	1305
1306	_kernel_primitive:NN	<u>\sjis</u>	\tex_sjis:D	1306
1307	_kernel_primitive:NN	<u>\tate</u>	\tex_tate:D	1307
1308	_kernel_primitive:NN	<u>\tbaselineshift</u>	\tex_tbaselineshift:D	1308
1309	_kernel_primitive:NN	<u>\textbaselineshiftfactor</u>		1309
1310		\tex_textbaselineshiftfactor:D		1310
1311	_kernel_primitive:NN	<u>\tfont</u>	\tex_tfont:D	1311
1312	_kernel_primitive:NN	<u>\tojis</u>	\tex_tojis:D	1312
1313	_kernel_primitive:NN	<u>\toucs</u>	\tex_toucs:D	1313
1314	_kernel_primitive:NN	<u>\ucs</u>	\tex_ucs:D	1314
1315	_kernel_primitive:NN	<u>\xkanjiskip</u>	\tex_xkanjiskip:D	1315
1316	_kernel_primitive:NN	<u>\xspcode</u>	\tex_xspcode:D	1316
1317	_kernel_primitive:NN	<u>\ybaselineshift</u>	\tex_ybaselineshift:D	1317
1318	_kernel_primitive:NN	<u>\yoko</u>	\tex_yoko:D	1318
1319	_kernel_primitive:NN	<u>\vfi</u>	\tex_vfi:D	1319
1320	_kernel_primitive:NN	<u>\disablecjktoken</u>	\tex_disablecjktoken:D	1320
1321	_kernel_primitive:NN	<u>\enablecjktoken</u>	\tex_enablecjktoken:D	1321
1322	_kernel_primitive:NN	<u>\forcecjktoken</u>	\tex_forcecjktoken:D	1322
1323	_kernel_primitive:NN	<u>\kchar</u>	\tex_kchar:D	1323
1324	_kernel_primitive:NN	<u>\kchardef</u>	\tex_kchardef:D	1324
1325	_kernel_primitive:NN	<u>\uptexrevision</u>	\tex_uptexrevision:D	1325
1326	_kernel_primitive:NN	<u>\uptexversion</u>	\tex_uptexversion:D	1326
1327	_kernel_primitive:NN	<u>\odelcode</u>	\tex_odelcode:D	1327
1328	_kernel_primitive:NN	<u>\odelimiter</u>	\tex_odelimiter:D	1328
1329	_kernel_primitive:NN	<u>\omathaccent</u>	\tex_omathaccent:D	1329
1330	_kernel_primitive:NN	<u>\omathchar</u>	\tex_omathchar:D	1330
1331	_kernel_primitive:NN	<u>\omathchardef</u>	\tex_omathchardef:D	1331
1332	_kernel_primitive:NN	<u>\omathcode</u>	\tex_omathcode:D	1332
1333	_kernel_primitive:NN	<u>\oradical</u>	\tex_oradical:D	1333
1334	_kernel_primitive:NN	<u>\ignoreprimitiveerror</u>	\tex_ignoreprimitiveerror:D	1334


```

1335 \__kernel_primitive:NN \partokencontext \tex_partokencontext:D 1335
1336 \__kernel_primitive:NN \partokenname \tex_partokenname:D 1336
1337 \__kernel_primitive:NN \showstream \tex_showstream:D 1337
1338 \__kernel_primitive:NN \tracingstacklevels \tex_tracingstacklevels:D 1338
1339 \tex_endgroup:D 1339
1340 \tex_ifdefined:D \@@end 1340
1341 \tex_let:D \tex_end:D \@@end 1341
1342 \tex_let:D \tex_input:D \@@input 1342
1343 \tex_fi:D 1343
1344 \tex_ifdefined:D \@@hyph 1344
1345 \tex_let:D \tex_everydisplay:D \frozen@everydisplay 1345
1346 \tex_let:D \tex_everymath:D \frozen@everymath 1346
1347 \tex_let:D \tex_hyphen:D \@@hyph 1347
1348 \tex_let:D \tex_italiccorrection:D \@@italiccorr 1348
1349 \tex_let:D \tex_underline:D \@@underline 1349
1350 \tex_ifdefined:D \@@shipout 1350
1351 \tex_let:D \tex_shipout:D \@@shipout 1351
1352 \tex_fi:D 1352
1353 \tex_begingroup:D 1353
1354 \tex_edef:D \l_tmpa_tl { \tex_string:D \shipout } 1354
1355 \tex_edef:D \l_tmpb_tl { \tex_meaning:D \shipout } 1355
1356 \tex_ifx:D \l_tmpa_tl \l_tmpb_tl 1356
1357 \tex_else:D 1357
1358 \tex_expandafter:D \@tfor \tex_expandafter:D \@tempa \tex_string:D := 1358
1359 \CROP@shipout 1359
1360 \dup@shipout 1360
1361 \GPTorg@shipout 1361
1362 \LL@shipout 1362
1363 \mem@oldshipout 1363
1364 \opem@shipout 1364
1365 \pgfpages@originalshipout 1365
1366 \pr@shipout 1366
1367 \Shipout 1367
1368 \verso@orig@shipout 1368
1369 \do 1369
1370 { 1370
1371 \tex_edef:D \l_tmpb_tl 1371
1372 { \tex_expandafter:D \tex_meaning:D \@tempa } 1372
1373 \tex_ifx:D \l_tmpa_tl \l_tmpb_tl 1373
1374 \tex_global:D \tex_expandafter:D \tex_let:D 1374
1375 \tex_expandafter:D \tex_shipout:D \@tempa 1375
1376 \tex_fi:D 1376
1377 } 1377
1378 \tex_fi:D 1378
1379 \tex_endgroup:D 1379
1380 \tex_let:D \tex_tracingfonts:D \tex_undefined:D 1380

```

1381	\tex_ifdefined:D \pdftracingfonts	1381
1382	\tex_let:D \tex_tracingfonts:D \pdftracingfonts	1382
1383	\tex_else:D	1383
1384	\tex_ifdefined:D \tex_directlua:D	1384
1385	\tex_directlua:D { tex.enableprimitives("@@", {"tracingfonts"}) }	1385
1386	\tex_let:D \tex_tracingfonts:D \@tracingfonts	1386
1387	\tex_fi:D	1387
1388	\tex_fi:D	1388
1389	\tex_fi:D	1389
1390	\tex_ifnum:D 0	1390
1391	\tex_ifdefined:D \tex_pdftexversion:D 1 \tex_fi:D	1391
1392	\tex_ifdefined:D \tex luatexversion:D 1 \tex_fi:D	1392
1393	= 0 %	1393
1394	\tex_let:D \tex_pdfmapfile:D \tex_undefined:D	1394
1395	\tex_let:D \tex_pdfmapline:D \tex_undefined:D	1395
1396	\tex_fi:D	1396
1397	\tex_begingroup:D	1397
1398	\tex_edef:D \l_tmpa_tl { \tex_meaning:D \tex_time:D }	1398
1399	\tex_edef:D \l_tmpb_tl { \tex_string:D \time }	1399
1400	\tex_ifx:D \l_tmpa_tl \l_tmpb_tl	1400
1401	\tex_else:D	1401
1402	\tex_global:D \tex_let:D \tex_time:D \tex_undefined:D	1402
1403	\tex_fi:D	1403
1404	\tex_edef:D \l_tmpa_tl { \tex_meaning:D \tex_day:D }	1404
1405	\tex_edef:D \l_tmpb_tl { \tex_string:D \day }	1405
1406	\tex_ifx:D \l_tmpa_tl \l_tmpb_tl	1406
1407	\tex_else:D	1407
1408	\tex_global:D \tex_let:D \tex_day:D \tex_undefined:D	1408
1409	\tex_fi:D	1409
1410	\tex_edef:D \l_tmpa_tl { \tex_meaning:D \tex_month:D }	1410
1411	\tex_edef:D \l_tmpb_tl { \tex_string:D \month }	1411
1412	\tex_ifx:D \l_tmpa_tl \l_tmpb_tl	1412
1413	\tex_else:D	1413
1414	\tex_global:D \tex_let:D \tex_month:D \tex_undefined:D	1414
1415	\tex_fi:D	1415
1416	\tex_edef:D \l_tmpa_tl { \tex_meaning:D \tex_year:D }	1416
1417	\tex_edef:D \l_tmpb_tl { \tex_string:D \year }	1417
1418	\tex_ifx:D \l_tmpa_tl \l_tmpb_tl	1418
1419	\tex_else:D	1419
1420	\tex_global:D \tex_let:D \tex_year:D \tex_undefined:D	1420
1421	\tex_fi:D	1421
1422	\tex_endgroup:D	1422
1423	\tex_ifdefined:D \orieveryjob	1423
1424	\tex_let:D \tex_everyjob:D \orieveryjob	1424
1425	\tex_fi:D	1425
1426	\tex_ifdefined:D \oripdfoutput	1426

1427	\tex_let:D \tex_pdfoutput:D \oripdfoutput	1427
1428	\tex_fi:D	1428
1429	\tex_ifdefined:D \normalend	1429
1430	\tex_let:D \tex_end:D \normalend	1430
1431	\tex_let:D \tex_everyjob:D \normaleveryjob	1431
1432	\tex_let:D \tex_input:D \normalinput	1432
1433	\tex_let:D \tex_language:D \normallanguage	1433
1434	\tex_let:D \tex_mathop:D \normalmathop	1434
1435	\tex_let:D \tex_month:D \normalmonth	1435
1436	\tex_let:D \tex_outer:D \normalouter	1436
1437	\tex_let:D \tex_over:D \normalover	1437
1438	\tex_let:D \tex_vcenter:D \normalvcenter	1438
1439	\tex_let:D \tex_unexpanded:D \normalunexpanded	1439
1440	\tex_let:D \tex_expanded:D \normalexpanded	1440
1441	\tex_fi:D	1441
1442	\tex_ifdefined:D \normalitaliccorrection	1442
1443	\tex_let:D \tex_hoffset:D \normalhoffset	1443
1444	\tex_let:D \tex_italiccorrection:D \normalitaliccorrection	1444
1445	\tex_let:D \tex_voffset:D \normalvoffset	1445
1446	\tex_let:D \tex_showtokens:D \normalshowtokens	1446
1447	\tex_let:D \tex_bodydir:D \spac_directions_normal_body_dir	1447
1448	\tex_let:D \tex_pagedir:D \spac_directions_normal_page_dir	1448
1449	\tex_fi:D	1449
1450	\tex_ifdefined:D \normalleft	1450
1451	\tex_let:D \tex_left:D \normalleft	1451
1452	\tex_let:D \tex_middle:D \normalmiddle	1452
1453	\tex_let:D \tex_right:D \normalright	1453
1454	\tex_fi:D	1454
1455	%% File: l3basics.dtx	1455
1456	\tex_global:D \tex_let:D \if_true: \tex_iftrue:D	1456
1457	\tex_global:D \tex_let:D \if_false: \tex_iffalse:D	1457
1458	\tex_global:D \tex_let:D \or: \tex_or:D	1458
1459	\tex_global:D \tex_let:D \else: \tex_else:D	1459
1460	\tex_global:D \tex_let:D \fi: \tex_fi:D	1460
1461	\tex_global:D \tex_let:D \reverse_if:N \tex_unless:D	1461
1462	\tex_global:D \tex_let:D \if:w \tex_if:D	1462
1463	\tex_global:D \tex_let:D \if_charcode:w \tex_if:D	1463
1464	\tex_global:D \tex_let:D \if_catcode:w \tex_ifcat:D	1464
1465	\tex_global:D \tex_let:D \if_meaning:w \tex_ifx:D	1465
1466	\tex_global:D \tex_let:D \if_bool:N \tex_ifodd:D	1466
1467	\tex_global:D \tex_let:D \if_mode_math: \tex_ifmmode:D	1467
1468	\tex_global:D \tex_let:D \if_mode_horizontal: \tex_ifhmode:D	1468
1469	\tex_global:D \tex_let:D \if_mode_vertical: \tex_ifvmode:D	1469
1470	\tex_global:D \tex_let:D \if_mode_inner: \tex_ifinner:D	1470
1471	\tex_global:D \tex_let:D \if_cs_exist:N \tex_ifdefined:D	1471
1472	\tex_global:D \tex_let:D \if_cs_exist:w \tex_ifcsname:D	1472

```

1473 \tex_global:D \tex_let:D \cs:w \tex_csname:D 1473
1474 \tex_global:D \tex_let:D \cs_end: \tex_endcsname:D 1474
1475 \tex_global:D \tex_let:D \exp_after:wN \tex_expandafter:D 1475
1476 \tex_global:D \tex_let:D \exp_not:N \tex_noexpand:D 1476
1477 \tex_global:D \tex_let:D \exp_not:n \tex_unexpanded:D 1477
1478 \tex_global:D \tex_let:D \exp:w \tex_romannumeral:D 1478
1479 \tex_global:D \tex_chardef:D \exp_end: = 0 ~ 1479
1480 \tex_global:D \tex_let:D \token_to_meaning:N \tex_meaning:D 1480
1481 \tex_global:D \tex_let:D \cs_meaning:N \tex_meaning:D 1481
1482 \tex_global:D \tex_let:D \tl_to_str:n \tex_detokenize:D 1482
1483 \tex_global:D \tex_let:D \token_to_str:N \tex_string:D 1483
1484 \tex_global:D \tex_let:D \__kernel_tl_to_str:w \tex_detokenize:D 1484
1485 \tex_global:D \tex_let:D \scan_stop: \tex_relax:D 1485
1486 \tex_global:D \tex_let:D \group_begin: \tex_begingroup:D 1486
1487 \tex_global:D \tex_let:D \group_end: \tex_endgroup:D 1487
1488 \tex_global:D \tex_let:D \if_int_compare:w \tex_ifnum:D 1488
1489 \tex_global:D \tex_let:D \__int_to_roman:w \tex_romannumeral:D 1489
1490 \tex_global:D \tex_let:D \group_insert_after:N \tex_aftergroup:D 1490
1491 \tex_long:D \tex_gdef:D \exp_args:Nc #1#2 1491
1492 { \exp_after:wN #1 \cs:w #2 \cs_end: } 1492
1493 \tex_long:D \tex_gdef:D \exp_args:cc #1#2 1493
1494 { \cs:w #1 \exp_after:wN \cs_end: \cs:w #2 \cs_end: } 1494
1495 \tex_gdef:D \token_to_str:c { \exp_args:Nc \token_to_str:N } 1495
1496 \tex_long:D \tex_gdef:D \cs_meaning:c #1 1496
1497 { 1497
1498 \if_cs_exist:w #1 \cs_end: 1498
1499 \exp_after:wN \use_i:nn 1499
1500 \else: 1500
1501 \exp_after:wN \use_ii:nn 1501
1502 \fi: 1502
1503 { \exp_args:Nc \cs_meaning:N {#1} } 1503
1504 { \tl_to_str:n {undefined} } 1504
1505 } 1505
1506 \tex_global:D \tex_let:D \token_to_meaning:c = \cs_meaning:c 1506
1507 \tex_global:D \tex_chardef:D \c_zero_int = 0 ~ 1507
1508 \tex_ifdefined:D \tex_luatexversion:D 1508
1509 \tex_global:D \tex_chardef:D \c_max_register_int = 65 535 ~ 1509
1510 \tex_else:D 1510
1511 \tex_ifdefined:D \tex_omathchardef:D 1511
1512 \tex_global:D \tex_omathchardef:D \c_max_register_int = 65535 ~ 1512
1513 \tex_else:D 1513
1514 \tex_global:D \tex_mathchardef:D \c_max_register_int = 32767 ~ 1514
1515 \tex_fi:D 1515
1516 \tex_fi:D 1516
1517 \tex_global:D \tex_let:D \cs_gset_nopar:Npn \tex_gdef:D 1517
1518 \tex_global:D \tex_let:D \cs_gset_nopar:Npe \tex_xdef:D 1518

```

1519	\tex_global:D \tex_let:D \cs_gset_nopar:Npx	\tex_xdef:D	1519
1520	\tex_protected:D \tex_long:D \tex_gdef:D \cs_gset:Npn		1520
1521	{ \tex_long:D \tex_gdef:D }		1521
1522	\tex_protected:D \tex_long:D \tex_gdef:D \cs_gset:Npe		1522
1523	{ \tex_long:D \tex_xdef:D }		1523
1524	\tex_global:D \tex_let:D \cs_gset:Npx \cs_gset:Npe		1524
1525	\tex_protected:D \tex_long:D \tex_gdef:D \cs_gset_protected_nopar:Npn		1525
1526	{ \tex_protected:D \tex_gdef:D }		1526
1527	\tex_protected:D \tex_long:D \tex_gdef:D \cs_gset_protected_nopar:Npe		1527
1528	{ \tex_protected:D \tex_xdef:D }		1528
1529	\tex_global:D \tex_let:D \cs_gset_protected_nopar:Npx \cs_gset_protected_nopar:Npe		1529
1530	\tex_protected:D \tex_long:D \tex_gdef:D \cs_gset_protected:Npn		1530
1531	{ \tex_protected:D \tex_long:D \tex_gdef:D }		1531
1532	\tex_protected:D \tex_long:D \tex_gdef:D \cs_gset_protected:Npe		1532
1533	{ \tex_protected:D \tex_long:D \tex_xdef:D }		1533
1534	\tex_global:D \tex_let:D \cs_gset_protected:Npx \cs_gset_protected:Npe		1534
1535	\tex_global:D \tex_let:D \cs_set_nopar:Npn	\tex_def:D	1535
1536	\tex_global:D \tex_let:D \cs_set_nopar:Npe	\tex_edef:D	1536
1537	\tex_global:D \tex_let:D \cs_set_nopar:Npx	\tex_edef:D	1537
1538	\cs_gset_protected:Npn \cs_set:Npn		1538
1539	{ \tex_long:D \tex_def:D }		1539
1540	\cs_gset_protected:Npn \cs_set:Npe		1540
1541	{ \tex_long:D \tex_edef:D }		1541
1542	\tex_global:D \tex_let:D \cs_set:Npx \cs_set:Npe		1542
1543	\cs_gset_protected:Npn \cs_set_protected_nopar:Npn		1543
1544	{ \tex_protected:D \tex_def:D }		1544
1545	\cs_gset_protected:Npn \cs_set_protected_nopar:Npe		1545
1546	{ \tex_protected:D \tex_edef:D }		1546
1547	\tex_global:D \tex_let:D \cs_set_protected_nopar:Npx \cs_set_protected_nopar:Npe		1547
1548	\cs_gset_protected:Npn \cs_set_protected:Npn		1548
1549	{ \tex_protected:D \tex_long:D \tex_def:D }		1549
1550	\cs_gset_protected:Npn \cs_set_protected:Npe		1550
1551	{ \tex_protected:D \tex_long:D \tex_edef:D }		1551
1552	\tex_global:D \tex_let:D \cs_set_protected:Npx \cs_set_protected:Npe		1552
1553	\cs_gset_nopar:Npn \l__exp_internal_tl { }		1553
1554	\cs_gset:Npn \use:c #1 { \cs:w #1 \cs_end: }		1554
1555	\cs_gset_protected:Npn \use:x #1		1555
1556	{		1556
1557	\cs_set_nopar:Npx \l__exp_internal_tl {#1}		1557
1558	\l__exp_internal_tl		1558
1559	}		1559
1560	\cs_gset:Npn \use:e #1 { \tex_expanded:D {#1} }		1560
1561	\cs_gset:Npn \use:n #1 {#1}		1561
1562	\cs_gset:Npn \use:nn #1#2 {#1#2}		1562
1563	\cs_gset:Npn \use:nnn #1#2#3 {#1#2#3}		1563
1564	\cs_gset:Npn \use:nnnn #1#2#3#4 {#1#2#3#4}		1564

1565	\cs_gset:Npn \use_i:nn	#1#2 {#1}	1565
1566	\cs_gset:Npn \use_ii:nn	#1#2 {#2}	1566
1567	\cs_gset:Npn \use_i:nnn	#1#2#3 {#1}	1567
1568	\cs_gset:Npn \use_ii:nnn	#1#2#3 {#2}	1568
1569	\cs_gset:Npn \use_iii:nnn	#1#2#3 {#3}	1569
1570	\cs_gset:Npn \use_i:nnnn	#1#2#3#4 {#1}	1570
1571	\cs_gset:Npn \use_ii:nnnn	#1#2#3#4 {#2}	1571
1572	\cs_gset:Npn \use_iii:nnnn	#1#2#3#4 {#3}	1572
1573	\cs_gset:Npn \use_iv:nnnn	#1#2#3#4 {#4}	1573
1574	\cs_gset:Npn \use_i:nnnnn	#1#2#3#4#5 {#1}	1574
1575	\cs_gset:Npn \use_ii:nnnnn	#1#2#3#4#5 {#2}	1575
1576	\cs_gset:Npn \use_iii:nnnnn	#1#2#3#4#5 {#3}	1576
1577	\cs_gset:Npn \use_iv:nnnnn	#1#2#3#4#5 {#4}	1577
1578	\cs_gset:Npn \use_v:nnnnn	#1#2#3#4#5 {#5}	1578
1579	\cs_gset:Npn \use_i:nnnnnn	#1#2#3#4#5#6 {#1}	1579
1580	\cs_gset:Npn \use_ii:nnnnnn	#1#2#3#4#5#6 {#2}	1580
1581	\cs_gset:Npn \use_iii:nnnnnn	#1#2#3#4#5#6 {#3}	1581
1582	\cs_gset:Npn \use_iv:nnnnnn	#1#2#3#4#5#6 {#4}	1582
1583	\cs_gset:Npn \use_v:nnnnnn	#1#2#3#4#5#6 {#5}	1583
1584	\cs_gset:Npn \use_vi:nnnnnn	#1#2#3#4#5#6 {#6}	1584
1585	\cs_gset:Npn \use_i:nnnnnnn	#1#2#3#4#5#6#7 {#1}	1585
1586	\cs_gset:Npn \use_ii:nnnnnnn	#1#2#3#4#5#6#7 {#2}	1586
1587	\cs_gset:Npn \use_iii:nnnnnnn	#1#2#3#4#5#6#7 {#3}	1587
1588	\cs_gset:Npn \use_iv:nnnnnnn	#1#2#3#4#5#6#7 {#4}	1588
1589	\cs_gset:Npn \use_v:nnnnnnn	#1#2#3#4#5#6#7 {#5}	1589
1590	\cs_gset:Npn \use_vi:nnnnnnn	#1#2#3#4#5#6#7 {#6}	1590
1591	\cs_gset:Npn \use_vii:nnnnnnn	#1#2#3#4#5#6#7 {#7}	1591
1592	\cs_gset:Npn \use_i:nnnnnnnn	#1#2#3#4#5#6#7#8 {#1}	1592
1593	\cs_gset:Npn \use_ii:nnnnnnnn	#1#2#3#4#5#6#7#8 {#2}	1593
1594	\cs_gset:Npn \use_iii:nnnnnnnn	#1#2#3#4#5#6#7#8 {#3}	1594
1595	\cs_gset:Npn \use_iv:nnnnnnnn	#1#2#3#4#5#6#7#8 {#4}	1595
1596	\cs_gset:Npn \use_v:nnnnnnnn	#1#2#3#4#5#6#7#8 {#5}	1596
1597	\cs_gset:Npn \use_vi:nnnnnnnn	#1#2#3#4#5#6#7#8 {#6}	1597
1598	\cs_gset:Npn \use_vii:nnnnnnnn	#1#2#3#4#5#6#7#8 {#7}	1598
1599	\cs_gset:Npn \use_viii:nnnnnnnn	#1#2#3#4#5#6#7#8 {#8}	1599
1600	\cs_gset:Npn \use_i:nnnnnnnnn	#1#2#3#4#5#6#7#8#9 {#1}	1600
1601	\cs_gset:Npn \use_ii:nnnnnnnnn	#1#2#3#4#5#6#7#8#9 {#2}	1601
1602	\cs_gset:Npn \use_iii:nnnnnnnnn	#1#2#3#4#5#6#7#8#9 {#3}	1602
1603	\cs_gset:Npn \use_iv:nnnnnnnnn	#1#2#3#4#5#6#7#8#9 {#4}	1603
1604	\cs_gset:Npn \use_v:nnnnnnnnn	#1#2#3#4#5#6#7#8#9 {#5}	1604
1605	\cs_gset:Npn \use_vi:nnnnnnnnn	#1#2#3#4#5#6#7#8#9 {#6}	1605
1606	\cs_gset:Npn \use_vii:nnnnnnnnn	#1#2#3#4#5#6#7#8#9 {#7}	1606
1607	\cs_gset:Npn \use_viii:nnnnnnnnn	#1#2#3#4#5#6#7#8#9 {#8}	1607
1608	\cs_gset:Npn \use_ix:nnnnnnnnn	#1#2#3#4#5#6#7#8#9 {#9}	1608
1609	\cs_gset:Npn \use_i_ii:nnn	#1#2#3 {#1#2}	1609
1610	\cs_gset:Npn \use_ii_i:nn	#1#2 { #2 #1 }	1610

```
1611 \cs_gset:Npn \use_none_delimit_by_q_nil:w #1 \q_nil { } 1611
1612 \cs_gset:Npn \use_none_delimit_by_q_stop:w #1 \q_stop { } 1612
1613 \cs_gset:Npn \use_none_delimit_by_q_recursion_stop:w #1 \q_recursion_stop { } 1613
1614 \cs_gset:Npn \use_i_delimit_by_q_nil:nw #1#2 \q_nil {#1} 1614
1615 \cs_gset:Npn \use_i_delimit_by_q_stop:nw #1#2 \q_stop {#1} 1615
1616 \cs_gset:Npn \use_i_delimit_by_q_recursion_stop:nw 1616
1617 #1#2 \q_recursion_stop {#1} 1617
1618 \cs_gset:Npn \use_none:n #1 { } 1618
1619 \cs_gset:Npn \use_none:nn #1#2 { } 1619
1620 \cs_gset:Npn \use_none:nnn #1#2#3 { } 1620
1621 \cs_gset:Npn \use_none:nnnn #1#2#3#4 { } 1621
1622 \cs_gset:Npn \use_none:nnnnn #1#2#3#4#5 { } 1622
1623 \cs_gset:Npn \use_none:nnnnnn #1#2#3#4#5#6 { } 1623
1624 \cs_gset:Npn \use_none:nnnnnnn #1#2#3#4#5#6#7 { } 1624
1625 \cs_gset:Npn \use_none:nnnnnnnn #1#2#3#4#5#6#7#8 { } 1625
1626 \cs_gset:Npn \use_none:nnnnnnnnn #1#2#3#4#5#6#7#8#9 { } 1626
1627 \cs_gset_protected:Npn \__kernel_if_debug:TF #1#2 {#2} 1627
1628 \cs_gset_protected:Npn \debug_on:n #1 1628
1629 { 1629
1630 \sys_load_debug: 1630
1631 \cs_if_exist:NT \__debug_all_on: 1631
1632 { \debug_on:n {#1} } 1632
1633 } 1633
1634 \cs_gset_protected:Npn \debug_off:n #1 1634
1635 { 1635
1636 \sys_load_debug: 1636
1637 \cs_if_exist:NT \__debug_all_on: 1637
1638 { \debug_off:n {#1} } 1638
1639 } 1639
1640 \cs_gset_protected:Npn \debug_suspend: { } 1640
1641 \cs_gset_protected:Npn \debug_resume: { } 1641
1642 \cs_gset_nopar:Npn \g__debug_deprecation_on_tl { } 1642
1643 \cs_gset_nopar:Npn \g__debug_deprecation_off_tl { } 1643
1644 \cs_gset_protected:Npn \__kernel_deprecation_code:nn #1#2 1644
1645 { 1645
1646 \tl_gput_right:Nn \g__debug_deprecation_on_tl {#1} 1646
1647 \tl_gput_right:Nn \g__debug_deprecation_off_tl {#2} 1647
1648 } 1648
1649 \cs_gset:Npn \prg_return_true: 1649
1650 { \exp_after:wN \use_i:nn \exp:w } 1650
1651 \cs_gset:Npn \prg_return_false: 1651
1652 { \exp_after:wN \use_ii:nn \exp:w } 1652
1653 \cs_gset:Npn \__prg_use_none_delimit_by_q_recursion_stop:w 1653
1654 #1 \q_prg_recursion_stop { } 1654
1655 \cs_gset_protected:Npn \prg_set_conditional:Npnn 1655
1656 { \__prg_generate_conditional_parm:NNNpnn \cs_set:Npn e } 1656
```



```
1657 \cs_gset_protected:Npn \prg_gset_conditional:Npnn 1657
1658 { \__prg_generate_conditional_parm:NNNpnn \cs_gset:Npn e } 1658
1659 \cs_gset_protected:Npn \prg_new_conditional:Npnn 1659
1660 { \__prg_generate_conditional_parm:NNNpnn \cs_new:Npn e } 1660
1661 \cs_gset_protected:Npn \prg_set_protected_conditional:Npnn 1661
1662 { \__prg_generate_conditional_parm:NNNpnn \cs_set_protected:Npn p } 1662
1663 \cs_gset_protected:Npn \prg_gset_protected_conditional:Npnn 1663
1664 { \__prg_generate_conditional_parm:NNNpnn \cs_gset_protected:Npn p } 1664
1665 \cs_gset_protected:Npn \prg_new_protected_conditional:Npnn 1665
1666 { \__prg_generate_conditional_parm:NNNpnn \cs_new_protected:Npn p } 1666
1667 \cs_gset_protected:Npn \__prg_generate_conditional_parm:NNNpnn #1#2#3#4# 1667
1668 { 1668
1669 \use:e 1669
1670 { 1670
1671 \__prg_generate_conditional:nnNNNnnn 1671
1672 \cs_split_function:N #3 1672
1673 } 1673
1674 #1 #2 {#4} 1674
1675 } 1675
1676 \cs_gset_protected:Npn \prg_set_conditional:Nnn 1676
1677 { \__prg_generate_conditional_count:NNNnn \cs_set:Npn e } 1677
1678 \cs_gset_protected:Npn \prg_gset_conditional:Nnn 1678
1679 { \__prg_generate_conditional_count:NNNnn \cs_gset:Npn e } 1679
1680 \cs_gset_protected:Npn \prg_new_conditional:Nnn 1680
1681 { \__prg_generate_conditional_count:NNNnn \cs_new:Npn e } 1681
1682 \cs_gset_protected:Npn \prg_set_protected_conditional:Nnn 1682
1683 { \__prg_generate_conditional_count:NNNnn \cs_set_protected:Npn p } 1683
1684 \cs_gset_protected:Npn \prg_gset_protected_conditional:Nnn 1684
1685 { \__prg_generate_conditional_count:NNNnn \cs_gset_protected:Npn p } 1685
1686 \cs_gset_protected:Npn \prg_new_protected_conditional:Nnn 1686
1687 { \__prg_generate_conditional_count:NNNnn \cs_new_protected:Npn p } 1687
1688 \cs_gset_protected:Npn \__prg_generate_conditional_count:NNNnn #1#2#3 1688
1689 { 1689
1690 \use:e 1690
1691 { 1691
1692 \__prg_generate_conditional_count:nnNNNnn 1692
1693 \cs_split_function:N #3 1693
1694 } 1694
1695 #1 #2 1695
1696 } 1696
1697 \cs_gset_protected:Npn \__prg_generate_conditional_count:nnNNNnn #1#2#3#4#5 1697
1698 { 1698
1699 \__kernel_cs_parm_from_arg_count:nnF 1699
1700 { \__prg_generate_conditional:nnNNNnnn {#1} {#2} #3 #4 #5 } 1700
1701 { \tl_count:n {#2} } 1701
1702 { 1702
```

```

1703         \msg_error:nnee { kernel } { bad-number-of-arguments }
1704         { \token_to_str:c { #1 : #2 } }
1705         { \tl_count:n {#2} }
1706     \use_none:nn
1707 }
1708 }
1709 \cs_gset_protected:Npn \__prg_generate_conditional:nnNNnnn #1#2#3#4#5#6#7#8
1710 {
1711     \if_meaning:w \c_false_bool #3
1712         \msg_error:nne { kernel } { missing-colon }
1713         { \token_to_str:c {#1} }
1714         \exp_after:wN \use_none:nn
1715     \fi:
1716     \use:e
1717     {
1718         \exp_not:N \__prg_generate_conditional:NNnnnnNw
1719         \exp_not:n { #4 #5 {#1} {#2} {#6} }
1720         \__prg_generate_conditional_test:w
1721             #8 \s__prg_mark
1722             \__prg_generate_conditional_fast:nw
1723             \prg_return_true: \else: \prg_return_false: \fi: \s__prg_mark
1724             \use_none:n
1725             \exp_not:n { {#8} \use_i_ii:nnn }
1726             \tl_to_str:n {#7}
1727             \exp_not:n { , \q__prg_recursion_tail , \q__prg_recursion_stop }
1728     }
1729 }
1730 \cs_gset:Npn \__prg_generate_conditional_test:w
1731     #1 \prg_return_true: \else: \prg_return_false: \fi: \s__prg_mark #2
1732     { #2 {#1} }
1733 \cs_gset:Npn \__prg_generate_conditional_fast:nw #1#2 \exp_not:n #3
1734     { \exp_not:n { {#1} \use_i:nn } }
1735 \cs_gset_protected:Npn \__prg_generate_conditional:NNnnnnNw #1#2#3#4#5#6#7#8 ,
1736 {
1737     \if_meaning:w \q__prg_recursion_tail #8
1738         \exp_after:wN \__prg_use_none_delimit_by_q_recursion_stop:w
1739     \fi:
1740     \use:c { __prg_generate_ #8 _form:wNNnnnnN }
1741     \tl_if_empty:nF {#8}
1742     {
1743         \msg_error:nnee
1744             { kernel } { conditional-form-unknown }
1745             {#8} { \token_to_str:c { #3 : #4 } }
1746     }
1747     \use_none:nnnnnnnn
1748     \s__prg_stop

```

```
1749 #1 #2 {#3} {#4} {#5} {#6} #7 1749
1750 \__prg_generate_conditional:NNnnnnNw #1 #2 {#3} {#4} {#5} {#6} #7 1750
1751 } 1751
1752 \cs_gset_protected:Npn \__prg_generate_p_form:wNNnnnnN 1752
1753 #1 \s__prg_stop #2#3#4#5#6#7#8 1753
1754 { 1754
1755 \if_meaning:w e #3 1755
1756 \exp_after:wN \use_i:nn 1756
1757 \else: 1757
1758 \exp_after:wN \use_ii:nn 1758
1759 \fi: 1759
1760 { 1760
1761 #8 1761
1762 { \exp_args:Nc #2 { #4 _p: #5 } #6 } 1762
1763 { { #7 \exp_end: \c_true_bool \c_false_bool } } 1763
1764 { #7 \__prg_p_true:w \fi: \c_false_bool } 1764
1765 } 1765
1766 { 1766
1767 \msg_error:nne { kernel } { protected-predicate } 1767
1768 { \token_to_str:c { #4 _p: #5 } } 1768
1769 } 1769
1770 } 1770
1771 \cs_gset_protected:Npn \__prg_generate_T_form:wNNnnnnN 1771
1772 #1 \s__prg_stop #2#3#4#5#6#7#8 1772
1773 { 1773
1774 #8 1774
1775 { \exp_args:Nc #2 { #4 : #5 T } #6 } 1775
1776 { { #7 \exp_end: \use:n \use_none:n } } 1776
1777 { #7 \__prg_T_true:w \fi: \use_none:n } 1777
1778 } 1778
1779 \cs_gset_protected:Npn \__prg_generate_F_form:wNNnnnnN 1779
1780 #1 \s__prg_stop #2#3#4#5#6#7#8 1780
1781 { 1781
1782 #8 1782
1783 { \exp_args:Nc #2 { #4 : #5 F } #6 } 1783
1784 { { #7 \exp_end: { } } } 1784
1785 { #7 \__prg_F_true:w \fi: \use:n } 1785
1786 } 1786
1787 \cs_gset_protected:Npn \__prg_generate_TF_form:wNNnnnnN 1787
1788 #1 \s__prg_stop #2#3#4#5#6#7#8 1788
1789 { 1789
1790 #8 1790
1791 { \exp_args:Nc #2 { #4 : #5 TF } #6 } 1791
1792 { { #7 \exp_end: } } 1792
1793 { #7 \__prg_TF_true:w \fi: \use_ii:nn } 1793
1794 } 1794
```

1795	\cs_gset:Npn __prg_p_true:w \fi: \c_false_bool { \fi: \c_true_bool }	1795
1796	\cs_gset:Npn __prg_T_true:w \fi: \use_none:n { \fi: \use:n }	1796
1797	\cs_gset:Npn __prg_F_true:w \fi: \use:n { \fi: \use_none:n }	1797
1798	\cs_gset:Npn __prg_TF_true:w \fi: \use_ii:nn { \fi: \use_i:nn }	1798
1799	\cs_gset_protected:Npn \prg_set_eq_conditional:NNn	1799
1800	{ __prg_set_eq_conditional:NNNn \cs_set_eq:cc }	1800
1801	\cs_gset_protected:Npn \prg_gset_eq_conditional:NNn	1801
1802	{ __prg_set_eq_conditional:NNNn \cs_gset_eq:cc }	1802
1803	\cs_gset_protected:Npn \prg_new_eq_conditional:NNn	1803
1804	{ __prg_set_eq_conditional:NNNn \cs_new_eq:cc }	1804
1805	\cs_gset_protected:Npn __prg_set_eq_conditional:NNNn #1#2#3#4	1805
1806	{	1806
1807	\use:e	1807
1808	{	1808
1809	\exp_not:N __prg_set_eq_conditional:nnNnnNNw	1809
1810	\cs_split_function:N #2	1810
1811	\cs_split_function:N #3	1811
1812	\exp_not:N #1	1812
1813	\tl_to_str:n {#4}	1813
1814	\exp_not:n { , \q__prg_recursion_tail , \q__prg_recursion_stop }	1814
1815	}	1815
1816	}	1816
1817	\cs_gset_protected:Npn __prg_set_eq_conditional:nnNnnNNw #1#2#3#4#5#6	1817
1818	{	1818
1819	\if_meaning:w \c_false_bool #3	1819
1820	\msg_error:nne { kernel } { missing-colon }	1820
1821	{ \token_to_str:c {#1} }	1821
1822	\exp_after:wN __prg_use_none_delimit_by_q_recursion_stop:w	1822
1823	\fi:	1823
1824	\if_meaning:w \c_false_bool #6	1824
1825	\msg_error:nne { kernel } { missing-colon }	1825
1826	{ \token_to_str:c {#4} }	1826
1827	\exp_after:wN __prg_use_none_delimit_by_q_recursion_stop:w	1827
1828	\fi:	1828
1829	__prg_set_eq_conditional_loop:nnnnNw {#1} {#2} {#4} {#5}	1829
1830	}	1830
1831	\cs_gset_protected:Npn __prg_set_eq_conditional_loop:nnnnNw #1#2#3#4#5#6 ,	1831
1832	{	1832
1833	\if_meaning:w \q__prg_recursion_tail #6	1833
1834	\exp_after:wN __prg_use_none_delimit_by_q_recursion_stop:w	1834
1835	\fi:	1835
1836	\use:c { __prg_set_eq_conditional_ #6 _form:wNnnnn }	1836
1837	\tl_if_empty:nF {#6}	1837
1838	{	1838
1839	\msg_error:nnee	1839
1840	{ kernel } { conditional-form-unknown }	1840

```

1841      {#6} {\token_to_str:c { #1 : #2 } }
1842    }
1843    \use_none:nnnnnn
1844    \s__prg_stop
1845    #5 {#1} {#2} {#3} {#4}
1846    \__prg_set_eq_conditional_loop:nnnnNw {#1} {#2} {#3} {#4} #5
1847  }
1848  \cs_gset:Npn \__prg_set_eq_conditional_p_form:wNnnnn #1 \s__prg_stop #2#3#4#5#6
1849    { #2 { #3 _p : #4      }      { #5 _p : #6      } }
1850  \cs_gset:Npn \__prg_set_eq_conditional_TF_form:wNnnnn #1 \s__prg_stop #2#3#4#5#6
1851    { #2 { #3      : #4 TF }      { #5      : #6 TF } }
1852  \cs_gset:Npn \__prg_set_eq_conditional_T_form:wNnnnn #1 \s__prg_stop #2#3#4#5#6
1853    { #2 { #3      : #4 T  }      { #5      : #6 T  } }
1854  \cs_gset:Npn \__prg_set_eq_conditional_F_form:wNnnnn #1 \s__prg_stop #2#3#4#5#6
1855    { #2 { #3      : #4 F  }      { #5      : #6 F  } }
1856  \tex_global:D \tex_chardef:D \c_true_bool  = 1 ~
1857  \tex_global:D \tex_chardef:D \c_false_bool = 0 ~
1858  \cs_gset:Npn \cs_to_str:N
1859  {
1860    \tex_romannumeral:D
1861    \if:w \token_to_str:N \__\__cs_to_str:w \fi:
1862    \exp_after:wN \__cs_to_str:N \token_to_str:N
1863  }
1864  \cs_gset:Npn \__cs_to_str:N #1 { \c_zero_int }
1865  \cs_gset:Npn \__cs_to_str:w #1 \__cs_to_str:N
1866    { - \int_value:w \fi: \exp_after:wN \c_zero_int }
1867  \cs_gset_protected:Npn \__cs_tmp:w #1
1868  {
1869    \cs_gset:Npn \cs_split_function:N ##1
1870    {
1871      \exp_after:wN \exp_after:wN \exp_after:wN
1872      \__cs_split_function_auxi:w
1873      \cs_to_str:N ##1 \s__cs_mark \c_true_bool
1874      #1 \s__cs_mark \c_false_bool \s__cs_stop
1875    }
1876    \cs_gset:Npn \__cs_split_function_auxi:w
1877      ##1 #1 ##2 \s__cs_mark ##3##4 \s__cs_stop
1878      { \__cs_split_function_auxii:w ##1 \s__cs_mark \s__cs_stop {##2} ##3 }
1879    \cs_gset:Npn \__cs_split_function_auxii:w ##1 \s__cs_mark ##2 \s__cs_stop
1880      { {##1} }
1881  }
1882  \exp_after:wN \__cs_tmp:w \token_to_str:N :
1883  \prg_gset_conditional:Npnn \cs_if_exist:N #1 { p , T , F , TF }
1884  {
1885    \if_meaning:w #1 \scan_stop:
1886    \use_i:nnnn

```



```

1887 \else: 1887
1888 \fi: 1888
1889 \if_cs_exist:N #1 1889
1890 \prg_return_true: 1890
1891 \else: 1891
1892 \prg_return_false: 1892
1893 \fi: 1893
1894 } 1894
1895 \cs_if_exist:NTF \tex_lastnamedcs:D 1895
1896 { 1896
1897 \prg_gset_conditional:Npnn \cs_if_exist:c #1 { p , T , F , TF } 1897
1898 { 1898
1899 \if_cs_exist:w #1 \cs_end: 1899
1900 \__cs_if_exist_c_aux: 1900
1901 \prg_return_true: 1901
1902 \else: 1902
1903 \prg_return_false: 1903
1904 \fi: 1904
1905 } 1905
1906 \cs_gset:Npn \__cs_if_exist_c_aux: 1906
1907 { \fi: \exp_after:wN \if_meaning:w \tex_lastnamedcs:D \scan_stop: \else: } 1907
1908 } 1908
1909 { 1909
1910 \prg_gset_conditional:Npnn \cs_if_exist:c #1 { p , T , F , TF } 1910
1911 { 1911
1912 \if_cs_exist:w #1 \cs_end: 1912
1913 \__cs_if_exist_c_aux:w 1913
1914 \fi: 1914
1915 \use_none:n {#1} 1915
1916 \if_false: 1916
1917 \prg_return_true: 1917
1918 \else: 1918
1919 \prg_return_false: 1919
1920 \fi: 1920
1921 } 1921
1922 \cs_gset:Npn \__cs_if_exist_c_aux:w \fi: \use_none:n #1 \if_false: 1922
1923 { \fi: \exp_after:wN \if_meaning:w \cs:w #1 \cs_end: \scan_stop: \else: } 1923
1924 } 1924
1925 \prg_gset_conditional:Npnn \cs_if_free:N #1 { p , T , F , TF } 1925
1926 { 1926
1927 \if_cs_exist:N #1 1927
1928 \else: 1928
1929 \use_none:nnnn 1929
1930 \fi: 1930
1931 \if_meaning:w #1 \scan_stop: 1931
1932 \prg_return_true: 1932

```

```

1933     \else:
1934         \prg_return_false:
1935     \fi:
1936 }
1937 \cs_if_exist:NTF \tex_lastnamedcs:D
1938 {
1939     \prg_gset_conditional:Npnn \cs_if_free:c #1 { p , T , F , TF }
1940     {
1941         \if_cs_exist:w #1 \cs_end:
1942         \__cs_if_free_c_aux:w
1943         \fi:
1944         \if_true:
1945             \prg_return_true:
1946         \else:
1947             \prg_return_false:
1948         \fi:
1949     }
1950     \cs_gset:Npn \__cs_if_free_c_aux:w \fi: \if_true:
1951     { \fi: \exp_after:wN \if_meaning:w \tex_lastnamedcs:D \scan_stop: }
1952 }
1953 {
1954     \prg_gset_conditional:Npnn \cs_if_free:c #1 { p , T , F , TF }
1955     {
1956         \if_cs_exist:w #1 \cs_end:
1957         \__cs_if_free_c_aux:w
1958         \fi:
1959         \use_none:n {#1}
1960         \if_true:
1961             \prg_return_true:
1962         \else:
1963             \prg_return_false:
1964         \fi:
1965     }
1966     \cs_gset:Npn \__cs_if_free_c_aux:w \fi: \use_none:n #1 \if_true:
1967     { \fi: \exp_after:wN \if_meaning:w \cs:w #1 \cs_end: \scan_stop: }
1968 }
1969 \cs_gset:Npn \cs_if_exist_use:NTF #1#2
1970 { \cs_if_exist:NTF #1 { #1 #2 } }
1971 \cs_gset:Npn \cs_if_exist_use:NF #1
1972 { \cs_if_exist:NTF #1 #1 }
1973 \cs_gset:Npn \cs_if_exist_use:NT #1 #2
1974 { \cs_if_exist:NT #1 { #1 #2 } }
1975 \cs_gset:Npn \cs_if_exist_use:N #1
1976 { \cs_if_exist:NT #1 #1 }
1977 \cs_if_exist:NTF \tex_lastnamedcs:D
1978 {

```

1979	\cs_gset:Npn \cs_if_exist_use:cTF #1	1979
1980	{	1980
1981	\if_cs_exist:w #1 \cs_end:	1981
1982	__cs_if_exist_use_aux:w	1982
1983	\fi:	1983
1984	\use_ii:nn	1984
1985	}	1985
1986	\cs_gset:Npn __cs_if_exist_use_aux:w \fi: \use_ii:nn	1986
1987	{ \fi: \exp_after:wN __cs_if_exist_use_aux:Nnn \tex_lastnamedcs:D }	1987
1988	}	1988
1989	{	1989
1990	\cs_gset:Npn \cs_if_exist_use:cTF #1	1990
1991	{	1991
1992	\if_cs_exist:w #1 \cs_end:	1992
1993	__cs_if_exist_use_aux:w	1993
1994	\fi:	1994
1995	\use_iii:nnn {#1}	1995
1996	}	1996
1997	\cs_gset:Npn __cs_if_exist_use_aux:w \fi: \use_iii:nnn #1	1997
1998	{ \fi: \exp_after:wN __cs_if_exist_use_aux:Nnn \cs:w #1 \cs_end: }	1998
1999	}	1999
2000	\cs_gset:Npn __cs_if_exist_use_aux:Nnn #1#2	2000
2001	{	2001
2002	\if_meaning:w #1 \scan_stop:	2002
2003	\exp_after:wN \use_iii:nnn	2003
2004	\fi:	2004
2005	\use_i:nn { #1 #2 }	2005
2006	}	2006
2007	\cs_gset:Npn \cs_if_exist_use:cF #1	2007
2008	{ \cs_if_exist_use:cTF {#1} {} }	2008
2009	\cs_gset:Npn \cs_if_exist_use:cT #1#2	2009
2010	{ \cs_if_exist_use:cTF {#1} {#2} {} }	2010
2011	\cs_gset:Npn \cs_if_exist_use:c #1	2011
2012	{ \cs_if_exist_use:cTF {#1} {} {} }	2012
2013	\cs_gset_protected:Npn \msg_error:nnee #1#2#3#4	2013
2014	{	2014
2015	\tex_newlinechar:D = ``^^J \scan_stop:	2015
2016	\tex_errmessage:D	2016
2017	{	2017
2018	!!~! ^^J	2018
2019	Argh,~internal~LaTeX3~error! ^^J ^^J	2019
2020	Module ~ #1 , ~ message~name~"#2": ^^J	2020
2021	Arguments~'#3'~and~'#4' ^^J ^^J	2021
2022	This~is~one~for~The~LaTeX3~Project::~bailing~out	2022
2023	}	2023
2024	\tex_end:D	2024

```

2025 }
2026 \cs_gset_protected:Npn \msg_error:nne #1#2#3
2027 { \msg_error:nnee {#1} {#2} {#3} { } }
2028 \cs_gset_protected:Npn \msg_error:nn #1#2
2029 { \msg_error:nnee {#1} {#2} { } { } }
2030 \cs_gset:Npn \msg_line_context:
2031 { on~line~ \tex_the:D \tex_inputlineno:D }
2032 \cs_gset_protected:Npn \iow_log:e
2033 { \tex_immediate:D \tex_write:D -1 }
2034 \cs_gset_protected:Npn \iow_term:e
2035 { \tex_immediate:D \tex_write:D 16 }
2036 \cs_gset_protected:Npn \__kernel_chk_if_free_cs:N #1
2037 {
2038   \cs_if_free:NF #1
2039   {
2040     \msg_error:nnee { kernel } { command-already-defined }
2041     { \token_to_str:N #1 } { \token_to_meaning:N #1 }
2042   }
2043 }
2044 \cs_gset_protected:Npn \__kernel_chk_if_free_cs:c
2045 { \exp_args:Nc \__kernel_chk_if_free_cs:N }
2046 \cs_set:Npn \__cs_tmp:w #1#2
2047 {
2048   \cs_gset_protected:Npn #1 ##1
2049   {
2050     \__kernel_chk_if_free_cs:N ##1
2051     #2 ##1
2052   }
2053 }
2054 \__cs_tmp:w \cs_new_nopar:Npn \cs_gset_nopar:Npn
2055 \__cs_tmp:w \cs_new_nopar:Npe \cs_gset_nopar:Npe
2056 \__cs_tmp:w \cs_new_nopar:Npx \cs_gset_nopar:Npx
2057 \__cs_tmp:w \cs_new:Npn \cs_gset:Npn
2058 \__cs_tmp:w \cs_new:Npe \cs_gset:Npe
2059 \__cs_tmp:w \cs_new:Npx \cs_gset:Npx
2060 \__cs_tmp:w \cs_new_protected_nopar:Npn \cs_gset_protected_nopar:Npn
2061 \__cs_tmp:w \cs_new_protected_nopar:Npe \cs_gset_protected_nopar:Npe
2062 \__cs_tmp:w \cs_new_protected_nopar:Npx \cs_gset_protected_nopar:Npx
2063 \__cs_tmp:w \cs_new_protected:Npn \cs_gset_protected:Npn
2064 \__cs_tmp:w \cs_new_protected:Npe \cs_gset_protected:Npe
2065 \__cs_tmp:w \cs_new_protected:Npx \cs_gset_protected:Npx
2066 \cs_set:Npn \__cs_tmp:w #1#2
2067 { \cs_new_protected_nopar:Npn #1 { \exp_args:Nc #2 } }
2068 \__cs_tmp:w \cs_set_nopar:cpn \cs_set_nopar:Npn
2069 \__cs_tmp:w \cs_set_nopar:cpe \cs_set_nopar:Npe
2070 \__cs_tmp:w \cs_set_nopar:cpx \cs_set_nopar:Npx

```

2071	_cs_tmp:w \cs_gset_nopar:cpn \cs_gset_nopar:Npn	2071
2072	_cs_tmp:w \cs_gset_nopar:cpe \cs_gset_nopar:Npe	2072
2073	_cs_tmp:w \cs_gset_nopar:cpx \cs_gset_nopar:Npx	2073
2074	_cs_tmp:w \cs_new_nopar:cpn \cs_new_nopar:Npn	2074
2075	_cs_tmp:w \cs_new_nopar:cpe \cs_new_nopar:Npe	2075
2076	_cs_tmp:w \cs_new_nopar:cpx \cs_new_nopar:Npx	2076
2077	_cs_tmp:w \cs_set:cpn \cs_set:Npn	2077
2078	_cs_tmp:w \cs_set:cpe \cs_set:Npe	2078
2079	_cs_tmp:w \cs_set:cpx \cs_set:Npx	2079
2080	_cs_tmp:w \cs_gset:cpn \cs_gset:Npn	2080
2081	_cs_tmp:w \cs_gset:cpe \cs_gset:Npe	2081
2082	_cs_tmp:w \cs_gset:cpx \cs_gset:Npx	2082
2083	_cs_tmp:w \cs_new:cpn \cs_new:Npn	2083
2084	_cs_tmp:w \cs_new:cpe \cs_new:Npe	2084
2085	_cs_tmp:w \cs_new:cpx \cs_new:Npx	2085
2086	_cs_tmp:w \cs_set_protected_nopar:cpn \cs_set_protected_nopar:Npn	2086
2087	_cs_tmp:w \cs_set_protected_nopar:cpe \cs_set_protected_nopar:Npe	2087
2088	_cs_tmp:w \cs_set_protected_nopar:cpx \cs_set_protected_nopar:Npx	2088
2089	_cs_tmp:w \cs_gset_protected_nopar:cpn \cs_gset_protected_nopar:Npn	2089
2090	_cs_tmp:w \cs_gset_protected_nopar:cpe \cs_gset_protected_nopar:Npe	2090
2091	_cs_tmp:w \cs_gset_protected_nopar:cpx \cs_gset_protected_nopar:Npx	2091
2092	_cs_tmp:w \cs_new_protected_nopar:cpn \cs_new_protected_nopar:Npn	2092
2093	_cs_tmp:w \cs_new_protected_nopar:cpe \cs_new_protected_nopar:Npe	2093
2094	_cs_tmp:w \cs_new_protected_nopar:cpx \cs_new_protected_nopar:Npx	2094
2095	_cs_tmp:w \cs_set_protected:cpn \cs_set_protected:Npn	2095
2096	_cs_tmp:w \cs_set_protected:cpe \cs_set_protected:Npe	2096
2097	_cs_tmp:w \cs_set_protected:cpx \cs_set_protected:Npx	2097
2098	_cs_tmp:w \cs_gset_protected:cpn \cs_gset_protected:Npn	2098
2099	_cs_tmp:w \cs_gset_protected:cpe \cs_gset_protected:Npe	2099
2100	_cs_tmp:w \cs_gset_protected:cpx \cs_gset_protected:Npx	2100
2101	_cs_tmp:w \cs_new_protected:cpn \cs_new_protected:Npn	2101
2102	_cs_tmp:w \cs_new_protected:cpe \cs_new_protected:Npe	2102
2103	_cs_tmp:w \cs_new_protected:cpx \cs_new_protected:Npx	2103
2104	\cs_new_protected:Npn \cs_set_eq:NN #1 { \tex_let:D #1 =~ }	2104
2105	\cs_new_protected:Npn \cs_set_eq:cN { \exp_args:Nc \cs_set_eq:NN }	2105
2106	\cs_new_protected:Npn \cs_set_eq:Nc { \exp_args:NNc \cs_set_eq:NN }	2106
2107	\cs_new_protected:Npn \cs_set_eq:cc { \exp_args:Ncc \cs_set_eq:NN }	2107
2108	\cs_new_protected:Npn \cs_gset_eq:NN { \tex_global:D \cs_set_eq:NN }	2108
2109	\cs_new_protected:Npn \cs_gset_eq:Nc { \exp_args:NNc \cs_gset_eq:NN }	2109
2110	\cs_new_protected:Npn \cs_gset_eq:cN { \exp_args:Nc \cs_gset_eq:NN }	2110
2111	\cs_new_protected:Npn \cs_gset_eq:cc { \exp_args:Ncc \cs_gset_eq:NN }	2111
2112	\cs_new_protected:Npn \cs_new_eq:NN #1	2112
2113	{	2113
2114	_kernel_chk_if_free_cs:N #1	2114
2115	\tex_global:D \cs_set_eq:NN #1	2115
2116	}	2116

2117	\cs_new_protected:Npn \cs_new_eq:cN { \exp_args:Nc \cs_new_eq:NN }	2117
2118	\cs_new_protected:Npn \cs_new_eq:Nc { \exp_args:NNc \cs_new_eq:NN }	2118
2119	\cs_new_protected:Npn \cs_new_eq:cc { \exp_args:Ncc \cs_new_eq:NN }	2119
2120	\cs_new_protected:Npn \cs_undefine:N #1	2120
2121	{ \cs_gset_eq:NN #1 \tex_undefined:D }	2121
2122	\cs_new_protected:Npn \cs_undefine:c #1	2122
2123	{	2123
2124	\if_cs_exist:w #1 \cs_end:	2124
2125	\else:	2125
2126	\use_i:nnnn	2126
2127	\fi:	2127
2128	\exp_args:Nc \cs_undefine:N {#1}	2128
2129	}	2129
2130	\cs_new_protected:Npn __kernel_cs_parm_from_arg_count:nnF #1#2	2130
2131	{	2131
2132	\exp_args:Ne __cs_parm_from_arg_count_test:nnF	2132
2133	{	2133
2134	\exp_after:wN \exp_not:n	2134
2135	\if_case:w \int_eval:n {#2}	2135
2136	{ }	2136
2137	\or: { ##1 }	2137
2138	\or: { ##1##2 }	2138
2139	\or: { ##1##2##3 }	2139
2140	\or: { ##1##2##3##4 }	2140
2141	\or: { ##1##2##3##4##5 }	2141
2142	\or: { ##1##2##3##4##5##6 }	2142
2143	\or: { ##1##2##3##4##5##6##7 }	2143
2144	\or: { ##1##2##3##4##5##6##7##8 }	2144
2145	\or: { ##1##2##3##4##5##6##7##8##9 }	2145
2146	\else: { \c_false_bool }	2146
2147	\fi:	2147
2148	}	2148
2149	{#1}	2149
2150	}	2150
2151	\cs_new_protected:Npn __cs_parm_from_arg_count_test:nnF #1#2	2151
2152	{	2152
2153	\if_meaning:w \c_false_bool #1	2153
2154	\exp_after:wN \use_ii:nn	2154
2155	\else:	2155
2156	\exp_after:wN \use_i:nn	2156
2157	\fi:	2157
2158	{ #2 {#1} }	2158
2159	}	2159
2160	\cs_new:Npn __cs_count_signature:N #1	2160
2161	{ \exp_args:Nf __cs_count_signature:n { \cs_split_function:N #1 } }	2161
2162	\cs_new:Npn __cs_count_signature:n #1	2162

```
2163 { \int_eval:n { \__cs_count_signature:nnN #1 } } 2163
2164 \cs_new:Npn \__cs_count_signature:nnN #1#2#3 2164
2165 { 2165
2166   \if_meaning:w \c_true_bool #3 2166
2167   \tl_count:n {#2} 2167
2168   \else: 2168
2169   -1 2169
2170   \fi: 2170
2171 } 2171
2172 \cs_new:Npn \__cs_count_signature:c 2172
2173 { \exp_args:Nc \__cs_count_signature:N } 2173
2174 \cs_new_protected:Npn \cs_generate_from_arg_count:NNnn #1#2#3#4 2174
2175 { 2175
2176   \__kernel_cs_parm_from_arg_count:nnF { \use:nnn #2 #1 } {#3} 2176
2177   { 2177
2178     \msg_error:nnee { kernel } { bad-number-of-arguments } 2178
2179     { \token_to_str:N #1 } { \int_eval:n {#3} } 2179
2180     \use_none:n 2180
2181   } 2181
2182   {#4} 2182
2183 } 2183
2184 \cs_new_protected:Npn \cs_generate_from_arg_count:cNnn 2184
2185 { \exp_args:Nc \cs_generate_from_arg_count:NNnn } 2185
2186 \cs_new_protected:Npn \cs_generate_from_arg_count:Ncnn 2186
2187 { \exp_args:NNc \cs_generate_from_arg_count:NNnn } 2187
2188 \cs_set:Npn \__cs_tmp:w #1#2#3 2188
2189 { 2189
2190   \cs_new_protected:cpx { cs_ #1 : #2 } 2190
2191   { 2191
2192     \exp_not:N \__cs_generate_from_signature:NNn 2192
2193     \exp_after:wN \exp_not:N \cs:w cs_ #1 : #3 \cs_end: 2193
2194   } 2194
2195 } 2195
2196 \cs_new_protected:Npn \__cs_generate_from_signature:NNn #1#2 2196
2197 { 2197
2198   \use:e 2198
2199   { 2199
2200     \__cs_generate_from_signature:nnNNNn 2200
2201     \cs_split_function:N #2 2201
2202   } 2202
2203   #1 #2 2203
2204 } 2204
2205 \cs_new_protected:Npn \__cs_generate_from_signature:nnNNNn #1#2#3#4#5#6 2205
2206 { 2206
2207   \bool_if:NTF #3 2207
2208   { 2208
```

```
2209 \cs_set_nopar:Npx \__cs_tmp:w 2209
2210 { \tl_map_function:nN {#2} \__cs_generate_from_signature:n } 2210
2211 \tl_if_empty:oF \__cs_tmp:w 2211
2212 { 2212
2213 \msg_error:nneee { kernel } { non-base-function } 2213
2214 { \token_to_str:N #5 } {#2} { \__cs_tmp:w } 2214
2215 } 2215
2216 \cs_generate_from_arg_count:NNnn 2216
2217 #5 #4 { \tl_count:n {#2} } {#6} 2217
2218 } 2218
2219 { 2219
2220 \msg_error:nne { kernel } { missing-colon } 2220
2221 { \token_to_str:N #5 } 2221
2222 } 2222
2223 } 2223
2224 \cs_new:Npn \__cs_generate_from_signature:n #1 2224
2225 { 2225
2226 \if:w n #1 \else: \if:w N #1 \else: 2226
2227 \if:w T #1 \else: \if:w F #1 \else: #1 \fi: \fi: \fi: 2227
2228 } 2228
2229 \__cs_tmp:w { set } { Nn } { Npn } 2229
2230 \__cs_tmp:w { set } { Ne } { Npe } 2230
2231 \__cs_tmp:w { set } { Nx } { Npx } 2231
2232 \__cs_tmp:w { set_nopar } { Nn } { Npn } 2232
2233 \__cs_tmp:w { set_nopar } { Ne } { Npe } 2233
2234 \__cs_tmp:w { set_nopar } { Nx } { Npx } 2234
2235 \__cs_tmp:w { set_protected } { Nn } { Npn } 2235
2236 \__cs_tmp:w { set_protected } { Ne } { Npe } 2236
2237 \__cs_tmp:w { set_protected } { Nx } { Npx } 2237
2238 \__cs_tmp:w { set_protected_nopar } { Nn } { Npn } 2238
2239 \__cs_tmp:w { set_protected_nopar } { Ne } { Npe } 2239
2240 \__cs_tmp:w { set_protected_nopar } { Nx } { Npx } 2240
2241 \__cs_tmp:w { gset } { Nn } { Npn } 2241
2242 \__cs_tmp:w { gset } { Ne } { Npe } 2242
2243 \__cs_tmp:w { gset } { Nx } { Npx } 2243
2244 \__cs_tmp:w { gset_nopar } { Nn } { Npn } 2244
2245 \__cs_tmp:w { gset_nopar } { Ne } { Npe } 2245
2246 \__cs_tmp:w { gset_nopar } { Nx } { Npx } 2246
2247 \__cs_tmp:w { gset_protected } { Nn } { Npn } 2247
2248 \__cs_tmp:w { gset_protected } { Ne } { Npe } 2248
2249 \__cs_tmp:w { gset_protected } { Nx } { Npx } 2249
2250 \__cs_tmp:w { gset_protected_nopar } { Nn } { Npn } 2250
2251 \__cs_tmp:w { gset_protected_nopar } { Ne } { Npe } 2251
2252 \__cs_tmp:w { gset_protected_nopar } { Nx } { Npx } 2252
2253 \__cs_tmp:w { new } { Nn } { Npn } 2253
2254 \__cs_tmp:w { new } { Ne } { Npe } 2254
```

2255	_cs_tmp:w { new }	{ Nx }	{ Npx }	2255
2256	_cs_tmp:w { new_nopar }	{ Nn }	{ Npn }	2256
2257	_cs_tmp:w { new_nopar }	{ Ne }	{ Npe }	2257
2258	_cs_tmp:w { new_nopar }	{ Nx }	{ Npx }	2258
2259	_cs_tmp:w { new_protected }	{ Nn }	{ Npn }	2259
2260	_cs_tmp:w { new_protected }	{ Ne }	{ Npe }	2260
2261	_cs_tmp:w { new_protected }	{ Nx }	{ Npx }	2261
2262	_cs_tmp:w { new_protected_nopar }	{ Nn }	{ Npn }	2262
2263	_cs_tmp:w { new_protected_nopar }	{ Ne }	{ Npe }	2263
2264	_cs_tmp:w { new_protected_nopar }	{ Nx }	{ Npx }	2264
2265	\\cs_set:Npn _cs_tmp:w #1#2			2265
2266	{			2266
2267	\\cs_new_protected:cpx { cs_ #1 : c #2 }			2267
2268	{			2268
2269	\\exp_not:N \\exp_args:Nc			2269
2270	\\exp_after:wN \\exp_not:N \\cs:w cs_ #1 : N #2 \\cs_end:			2270
2271	}			2271
2272	}			2272
2273	_cs_tmp:w { set }	{ n }		2273
2274	_cs_tmp:w { set }	{ e }		2274
2275	_cs_tmp:w { set }	{ x }		2275
2276	_cs_tmp:w { set_nopar }	{ n }		2276
2277	_cs_tmp:w { set_nopar }	{ e }		2277
2278	_cs_tmp:w { set_nopar }	{ x }		2278
2279	_cs_tmp:w { set_protected }	{ n }		2279
2280	_cs_tmp:w { set_protected }	{ e }		2280
2281	_cs_tmp:w { set_protected }	{ x }		2281
2282	_cs_tmp:w { set_protected_nopar }	{ n }		2282
2283	_cs_tmp:w { set_protected_nopar }	{ e }		2283
2284	_cs_tmp:w { set_protected_nopar }	{ x }		2284
2285	_cs_tmp:w { gset }	{ n }		2285
2286	_cs_tmp:w { gset }	{ e }		2286
2287	_cs_tmp:w { gset }	{ x }		2287
2288	_cs_tmp:w { gset_nopar }	{ n }		2288
2289	_cs_tmp:w { gset_nopar }	{ e }		2289
2290	_cs_tmp:w { gset_nopar }	{ x }		2290
2291	_cs_tmp:w { gset_protected }	{ n }		2291
2292	_cs_tmp:w { gset_protected }	{ e }		2292
2293	_cs_tmp:w { gset_protected }	{ x }		2293
2294	_cs_tmp:w { gset_protected_nopar }	{ n }		2294
2295	_cs_tmp:w { gset_protected_nopar }	{ e }		2295
2296	_cs_tmp:w { gset_protected_nopar }	{ x }		2296
2297	_cs_tmp:w { new }	{ n }		2297
2298	_cs_tmp:w { new }	{ e }		2298
2299	_cs_tmp:w { new }	{ x }		2299
2300	_cs_tmp:w { new_nopar }	{ n }		2300

2301	_cs_tmp:w { new_nopar }	{ e }	2301
2302	_cs_tmp:w { new_nopar }	{ x }	2302
2303	_cs_tmp:w { new_protected }	{ n }	2303
2304	_cs_tmp:w { new_protected }	{ e }	2304
2305	_cs_tmp:w { new_protected }	{ x }	2305
2306	_cs_tmp:w { new_protected_nopar }	{ n }	2306
2307	_cs_tmp:w { new_protected_nopar }	{ e }	2307
2308	_cs_tmp:w { new_protected_nopar }	{ x }	2308
2309	\\prg_new_conditional:Npnn \\cs_if_eq:NN #1#2 { p , T , F , TF }		2309
2310	{		2310
2311	\\if_meaning:w #1#2		2311
2312	\\prg_return_true: \\else: \\prg_return_false: \\fi:		2312
2313	}		2313
2314	\\cs_new:Npn \\cs_if_eq_p:cN { \\exp_args:Nc \\cs_if_eq_p:NN }		2314
2315	\\cs_new:Npn \\cs_if_eq:cNTF { \\exp_args:Nc \\cs_if_eq:NNTF }		2315
2316	\\cs_new:Npn \\cs_if_eq:cNT { \\exp_args:Nc \\cs_if_eq:NNT }		2316
2317	\\cs_new:Npn \\cs_if_eq:cNF { \\exp_args:Nc \\cs_if_eq:NNF }		2317
2318	\\cs_new:Npn \\cs_if_eq_p:Nc { \\exp_args:NNc \\cs_if_eq_p:NN }		2318
2319	\\cs_new:Npn \\cs_if_eq:NcTF { \\exp_args:NNc \\cs_if_eq:NNTF }		2319
2320	\\cs_new:Npn \\cs_if_eq:NcT { \\exp_args:NNc \\cs_if_eq:NNT }		2320
2321	\\cs_new:Npn \\cs_if_eq:NcF { \\exp_args:NNc \\cs_if_eq:NNF }		2321
2322	\\cs_new:Npn \\cs_if_eq_p:cc { \\exp_args:Ncc \\cs_if_eq_p:NN }		2322
2323	\\cs_new:Npn \\cs_if_eq:ccTF { \\exp_args:Ncc \\cs_if_eq:NNTF }		2323
2324	\\cs_new:Npn \\cs_if_eq:ccT { \\exp_args:Ncc \\cs_if_eq:NNT }		2324
2325	\\cs_new:Npn \\cs_if_eq:ccF { \\exp_args:Ncc \\cs_if_eq:NNF }		2325
2326	\\cs_new_protected:Npn __kernel_chk_defined:NT #1#2		2326
2327	{		2327
2328	\\cs_if_exist:NTF #1		2328
2329	{#2}		2329
2330	{		2330
2331	\\msg_error:nne { kernel } { variable-not-defined }		2331
2332	{ \\token_to_str:N #1 }		2332
2333	}		2333
2334	}		2334
2335	\\cs_new_protected:Npn __kernel_register_show:N		2335
2336	{ __kernel_register_show_aux:NN \\tl_show:n }		2336
2337	\\cs_new_protected:Npn __kernel_register_show:c		2337
2338	{ \\exp_args:Nc __kernel_register_show:N }		2338
2339	\\cs_new_protected:Npn __kernel_register_log:N		2339
2340	{ __kernel_register_show_aux:NN \\tl_log:n }		2340
2341	\\cs_new_protected:Npn __kernel_register_log:c		2341
2342	{ \\exp_args:Nc __kernel_register_log:N }		2342
2343	\\cs_new_protected:Npn __kernel_register_show_aux:NN #1#2		2343
2344	{		2344
2345	__kernel_chk_defined:NT #2		2345
2346	{		2346


```

2347 \exp_args:No \__kernel_register_show_aux:nNN 2347
2348 { \tex_the:D #2 } #2 #1 2348
2349 } 2349
2350 } 2350
2351 \cs_new_protected:Npn \__kernel_register_show_aux:nNN #1#2#3 2351
2352 { \exp_args:No #3 { \token_to_str:N #2 = #1 } } 2352
2353 \cs_new_protected:Npn \cs_show:N { \__kernel_show:NN \tl_show:n } 2353
2354 \cs_new_protected:Npn \cs_show:c 2354
2355 { \group_begin: \exp_args:Nnc \group_end: \cs_show:N } 2355
2356 \cs_new_protected:Npn \cs_log:N { \__kernel_show:NN \tl_log:n } 2356
2357 \cs_new_protected:Npn \cs_log:c 2357
2358 { \group_begin: \exp_args:Nnc \group_end: \cs_log:N } 2358
2359 \cs_new_protected:Npn \__kernel_show:NN #1#2 2359
2360 { 2360
2361 \group_begin: 2361
2362 \int_set:Nn \tex_escapechar:D { `` } 2362
2363 \exp_args:NNe 2363
2364 \group_end: 2364
2365 #1 { \token_to_str:N #2 = \cs_meaning:N #2 } 2365
2366 } 2366
2367 \cs_new_protected:Npn \group_show_list: 2367
2368 { \__kernel_group_show:NN \use_none:n 1 } 2368
2369 \cs_new_protected:Npn \group_log_list: 2369
2370 { \__kernel_group_show:NN \int_gzero:N 0 } 2370
2371 \cs_new_protected:Npn \__kernel_group_show:NN #1#2 2371
2372 { 2372
2373 \use:e 2373
2374 { 2374
2375 #1 \tex_interactionmode:D 2375
2376 \int_set:Nn \tex_tracingonline:D {#2} 2376
2377 \int_set:Nn \tex_errorcontextlines:D { -1 } 2377
2378 \exp_not:N \exp_after:wN \scan_stop: 2378
2379 \tex_showgroups:D 2379
2380 \int_gset:Nn \tex_interactionmode:D 2380
2381 { \int_use:N \tex_interactionmode:D } 2381
2382 \int_set:Nn \tex_tracingonline:D 2382
2383 { \int_use:N \tex_tracingonline:D } 2383
2384 \int_set:Nn \tex_errorcontextlines:D 2384
2385 { \int_use:N \tex_errorcontextlines:D } 2385
2386 } 2386
2387 } 2387
2388 \use:e 2388
2389 { 2389
2390 \exp_not:n { \cs_new:Npn \__kernel_prefix_arg_replacement:wN #1 } 2390
2391 \tl_to_str:n { macro : } \exp_not:n { #2 -> #3 \s__kernel_stop #4 } 2391
2392 } 2392

```

2393	{ #4 {#1} {#2} {#3} }	2393
2394	\cs_new:Npn \cs_prefix_spec:N #1	2394
2395	{	2395
2396	\token_if_macro:NTF #1	2396
2397	{	2397
2398	\exp_after:wN __kernel_prefix_arg_replacement:wN	2398
2399	\token_to_meaning:N #1 \s__kernel_stop \use_i:nnn	2399
2400	}	2400
2401	{ \scan_stop: }	2401
2402	}	2402
2403	\cs_new:Npn \cs_parameter_spec:N #1	2403
2404	{	2404
2405	\token_if_macro:NTF #1	2405
2406	{	2406
2407	\exp_after:wN __kernel_prefix_arg_replacement:wN	2407
2408	\token_to_meaning:N #1 \s__kernel_stop \use_ii:nnn	2408
2409	}	2409
2410	{ \scan_stop: }	2410
2411	}	2411
2412	\cs_new:Npn \cs_replacement_spec:N #1	2412
2413	{	2413
2414	\token_if_macro:NTF #1	2414
2415	{	2415
2416	\exp_after:wN __kernel_prefix_arg_replacement:wN	2416
2417	\token_to_meaning:N #1 \s__kernel_stop \use_iii:nnn	2417
2418	}	2418
2419	{ \scan_stop: }	2419
2420	}	2420
2421	\cs_new:Npn \prg_do_nothing: { }	2421
2422	\cs_new_eq:NN \prg_break_point:Nn \use_ii:nn	2422
2423	\cs_new:Npn \prg_map_break:Nn #1#2#3 \prg_break_point:Nn #4#5	2423
2424	{	2424
2425	#5	2425
2426	\if_meaning:w #1 #4	2426
2427	\exp_after:wN \use_iii:nnn	2427
2428	\fi:	2428
2429	\prg_map_break:Nn #1 {#2}	2429
2430	}	2430
2431	\cs_new_eq:NN \prg_break_point: \prg_do_nothing:	2431
2432	\cs_new:Npn \prg_break: #1 \prg_break_point: { }	2432
2433	\cs_new:Npn \prg_break:n #1#2 \prg_break_point: {#1}	2433
2434	\cs_new_protected:Npn \mode_leave_vertical:	2434
2435	{	2435
2436	\if_mode_vertical:	2436
2437	\exp_after:wN \tex_indent:D	2437
2438	\fi:	2438

```

2439 }
2440 %% File: l3expan.dtx
2441 \cs_new:Npn \__exp_arg_next:nnn #1#2#3 { #2 \::: { #3 {#1} } }
2442 \cs_new:Npn \__exp_arg_next:Nnn #1#2#3 { #2 \::: { #3 #1 } }
2443 \cs_new:Npn \::: #1 {#1}
2444 \cs_new:Npn \::n #1 \::: #2#3 { #1 \::: { #2 {#3} } }
2445 \cs_new:Npn \::N #1 \::: #2#3 { #1 \::: {#2#3} }
2446 \cs_new:Npn \::p #1 \::: #2#3# { #1 \::: {#2#3} }
2447 \cs_new:Npn \::c #1 \::: #2#3
2448 { \exp_after:wN \__exp_arg_next:Nnn \cs:w #3 \cs_end: {#1} {#2} }
2449 \cs_new:Npn \::o #1 \::: #2#3
2450 { \exp_after:wN \__exp_arg_next:nnn \exp_after:wN {#3} {#1} {#2} }
2451 \cs_new:Npn \::e #1 \::: #2#3
2452 { \tex_expanded:D { \exp_not:n { #1 \::: } { \exp_not:n {#2} {#3} } } }
2453 \cs_new:Npn \::f #1 \::: #2#3
2454 {
2455   \exp_after:wN \__exp_arg_next:nnn
2456   \exp_after:wN { \exp:w \exp_end_continue_f:w #3 }
2457   {#1} {#2}
2458 }
2459 \use:nn { \cs_new_eq:NN \exp_stop_f: } { ~ }
2460 \cs_new_protected:Npn \::x #1 \::: #2#3
2461 {
2462   \cs_set_nopar:Npe \l__exp_internal_tl
2463   { \exp_not:n { #1 \::: } { \exp_not:n {#2} {#3} } }
2464   \l__exp_internal_tl
2465 }
2466 \cs_new:Npn \::V #1 \::: #2#3
2467 {
2468   \exp_after:wN \__exp_arg_next:nnn
2469   \exp_after:wN { \exp:w \__exp_eval_register:N #3 }
2470   {#1} {#2}
2471 }
2472 \cs_new:Npn \::v #1 \::: #2#3
2473 {
2474   \exp_after:wN \__exp_arg_next:nnn
2475   \exp_after:wN { \exp:w \__exp_eval_register:c {#3} }
2476   {#1} {#2}
2477 }
2478 \cs_new:Npn \__exp_eval_register:N #1
2479 {
2480   \exp_after:wN \if_meaning:w \exp_not:N #1 #1
2481   \if_meaning:w \scan_stop: #1
2482   \__exp_eval_error_msg:w
2483   \fi:
2484   \else:

```

```

2485 \exp_after:wN \use_i_ii:nnn
2486 \fi:
2487 \exp_after:wN \exp_end: \tex_the:D #1
2488 }
2489 \cs_new:Npn \__exp_eval_register:c #1
2490 { \exp_after:wN \__exp_eval_register:N \cs:w #1 \cs_end: }
2491 \cs_new:Npn \__exp_eval_error_msg:w #1 \tex_the:D #2
2492 {
2493     \fi:
2494     \fi:
2495     \msg_expandable_error:nnn { kernel } { bad-variable } {#2}
2496     \exp_end:
2497 }
2498 \cs_new:Npn \exp_args:NNc #1#2#3
2499 { \exp_after:wN #1 \exp_after:wN #2 \cs:w # 3\cs_end: }
2500 \cs_new:Npn \exp_args:Ncc #1#2#3
2501 { \exp_after:wN #1 \cs:w #2 \exp_after:wN \cs_end: \cs:w #3 \cs_end: }
2502 \cs_new:Npn \exp_args:Nccc #1#2#3#4
2503 {
2504     \exp_after:wN #1
2505     \cs:w #2 \exp_after:wN \cs_end:
2506     \cs:w #3 \exp_after:wN \cs_end:
2507     \cs:w #4 \cs_end:
2508 }
2509 \cs_new:Npn \exp_args:No #1#2 { \exp_after:wN #1 \exp_after:wN {#2} }
2510 \cs_new:Npn \exp_args:NNo #1#2#3
2511 { \exp_after:wN #1 \exp_after:wN #2 \exp_after:wN {#3} }
2512 \cs_new:Npn \exp_args:NNNo #1#2#3#4
2513 { \exp_after:wN #1 \exp_after:wN#2 \exp_after:wN #3 \exp_after:wN {#4} }
2514 \cs_new:Npn \exp_args:Ne #1#2
2515 { \exp_after:wN #1 \tex_expanded:D { {#2} } }
2516 \cs_new:Npn \exp_args:Nf #1#2
2517 { \exp_after:wN #1 \exp_after:wN { \exp:w \exp_end_continue_f:w #2 } }
2518 \cs_new:Npn \exp_args:Nv #1#2
2519 {
2520     \exp_after:wN #1 \exp_after:wN
2521     { \exp:w \__exp_eval_register:c {#2} }
2522 }
2523 \cs_new:Npn \exp_args:NV #1#2
2524 {
2525     \exp_after:wN #1 \exp_after:wN
2526     { \exp:w \__exp_eval_register:N #2 }
2527 }
2528 \cs_new:Npn \exp_args:NNV #1#2#3
2529 {
2530     \exp_after:wN #1

```

2531	\exp_after:wN #2	2531
2532	\exp_after:wN { \exp:w __exp_eval_register:N #3 }	2532
2533	}	2533
2534	\cs_new:Npn \exp_args:NNv #1#2#3	2534
2535	{	2535
2536	\exp_after:wN #1	2536
2537	\exp_after:wN #2	2537
2538	\exp_after:wN { \exp:w __exp_eval_register:c {#3} }	2538
2539	}	2539
2540	\cs_new:Npn \exp_args:NNe #1#2#3	2540
2541	{	2541
2542	\exp_after:wN #1	2542
2543	\exp_after:wN #2	2543
2544	\tex_expanded:D { {#3} }	2544
2545	}	2545
2546	\cs_new:Npn \exp_args:NNf #1#2#3	2546
2547	{	2547
2548	\exp_after:wN #1	2548
2549	\exp_after:wN #2	2549
2550	\exp_after:wN { \exp:w \exp_end_continue_f:w #3 }	2550
2551	}	2551
2552	\cs_new:Npn \exp_args:Nco #1#2#3	2552
2553	{	2553
2554	\exp_after:wN #1	2554
2555	\cs:w #2 \exp_after:wN \cs_end:	2555
2556	\exp_after:wN {#3}	2556
2557	}	2557
2558	\cs_new:Npn \exp_args:NcV #1#2#3	2558
2559	{	2559
2560	\exp_after:wN #1	2560
2561	\cs:w #2 \exp_after:wN \cs_end:	2561
2562	\exp_after:wN { \exp:w __exp_eval_register:N #3 }	2562
2563	}	2563
2564	\cs_new:Npn \exp_args:Ncv #1#2#3	2564
2565	{	2565
2566	\exp_after:wN #1	2566
2567	\cs:w #2 \exp_after:wN \cs_end:	2567
2568	\exp_after:wN { \exp:w __exp_eval_register:c {#3} }	2568
2569	}	2569
2570	\cs_new:Npn \exp_args:Ncf #1#2#3	2570
2571	{	2571
2572	\exp_after:wN #1	2572
2573	\cs:w #2 \exp_after:wN \cs_end:	2573
2574	\exp_after:wN { \exp:w \exp_end_continue_f:w #3 }	2574
2575	}	2575
2576	\cs_new:Npn \exp_args:NVV #1#2#3	2576

2577	{	2577
2578	\exp_after:wN #1	2578
2579	\exp_after:wN { \exp:w \exp_after:wN	2579
2580	__exp_eval_register:N \exp_after:wN #2 \exp_after:wN }	2580
2581	\exp_after:wN { \exp:w __exp_eval_register:N #3 }	2581
2582	}	2582
2583	\cs_new:Npn \exp_args:NNNV #1#2#3#4	2583
2584	{	2584
2585	\exp_after:wN #1	2585
2586	\exp_after:wN #2	2586
2587	\exp_after:wN #3	2587
2588	\exp_after:wN { \exp:w __exp_eval_register:N #4 }	2588
2589	}	2589
2590	\cs_new:Npn \exp_args:NNNv #1#2#3#4	2590
2591	{	2591
2592	\exp_after:wN #1	2592
2593	\exp_after:wN #2	2593
2594	\exp_after:wN #3	2594
2595	\exp_after:wN { \exp:w __exp_eval_register:c {#4} }	2595
2596	}	2596
2597	\cs_new:Npn \exp_args:NNNe #1#2#3#4	2597
2598	{	2598
2599	\exp_after:wN #1	2599
2600	\exp_after:wN #2	2600
2601	\exp_after:wN #3	2601
2602	\tex_expanded:D { {#4} }	2602
2603	}	2603
2604	\cs_new:Npn \exp_args:NcNc #1#2#3#4	2604
2605	{	2605
2606	\exp_after:wN #1	2606
2607	\cs:w #2 \exp_after:wN \cs_end:	2607
2608	\exp_after:wN #3	2608
2609	\cs:w #4 \cs_end:	2609
2610	}	2610
2611	\cs_new:Npn \exp_args:NcNo #1#2#3#4	2611
2612	{	2612
2613	\exp_after:wN #1	2613
2614	\cs:w #2 \exp_after:wN \cs_end:	2614
2615	\exp_after:wN #3	2615
2616	\exp_after:wN {#4}	2616
2617	}	2617
2618	\cs_new:Npn \exp_args:Ncco #1#2#3#4	2618
2619	{	2619
2620	\exp_after:wN #1	2620
2621	\cs:w #2 \exp_after:wN \cs_end:	2621
2622	\cs:w #3 \exp_after:wN \cs_end:	2622

```

2623 \exp_after:wN {#4}
2624 }
2625 \cs_new_protected:Npn \exp_args:Nx #1#2
2626 { \use:x { \exp_not:N #1 {#2} } }
2627 \cs_new:Npn \__exp_arg_last_unbraced:nn #1#2 { #2#1 }
2628 \cs_new:Npn \::o_unbraced \::: #1#2
2629 { \exp_after:wN \__exp_arg_last_unbraced:nn \exp_after:wN {#2} {#1} }
2630 \cs_new:Npn \::V_unbraced \::: #1#2
2631 {
2632 \exp_after:wN \__exp_arg_last_unbraced:nn
2633 \exp_after:wN { \exp:w \__exp_eval_register:N #2 } {#1}
2634 }
2635 \cs_new:Npn \::v_unbraced \::: #1#2
2636 {
2637 \exp_after:wN \__exp_arg_last_unbraced:nn
2638 \exp_after:wN { \exp:w \__exp_eval_register:c {#2} } {#1}
2639 }
2640 \cs_new:Npn \::e_unbraced \::: #1#2
2641 { \tex_expanded:D { \exp_not:n {#1} #2 } }
2642 \cs_new:Npn \::f_unbraced \::: #1#2
2643 {
2644 \exp_after:wN \__exp_arg_last_unbraced:nn
2645 \exp_after:wN { \exp:w \exp_end_continue_f:w #2 } {#1}
2646 }
2647 \cs_new_protected:Npn \::x_unbraced \::: #1#2
2648 {
2649 \cs_set_nopar:Npe \l__exp_internal_tl { \exp_not:n {#1} #2 }
2650 \l__exp_internal_tl
2651 }
2652 \cs_new:Npn \exp_last_unbraced:No #1#2 { \exp_after:wN #1 #2 }
2653 \cs_new:Npn \exp_last_unbraced:NV #1#2
2654 { \exp_after:wN #1 \exp:w \__exp_eval_register:N #2 }
2655 \cs_new:Npn \exp_last_unbraced:Nv #1#2
2656 { \exp_after:wN #1 \exp:w \__exp_eval_register:c {#2} }
2657 \cs_new:Npn \exp_last_unbraced:Ne #1#2
2658 { \exp_after:wN #1 \tex_expanded:D {#2} }
2659 \cs_new:Npn \exp_last_unbraced:Nf #1#2
2660 { \exp_after:wN #1 \exp:w \exp_end_continue_f:w #2 }
2661 \cs_new:Npn \exp_last_unbraced:NNo #1#2#3
2662 { \exp_after:wN #1 \exp_after:wN #2 #3 }
2663 \cs_new:Npn \exp_last_unbraced:NNV #1#2#3
2664 {
2665 \exp_after:wN #1
2666 \exp_after:wN #2
2667 \exp:w \__exp_eval_register:N #3
2668 }

```

2669	\cs_new:Npn \exp_last_unbraced:NNf #1#2#3	2669
2670	{	2670
2671	\exp_after:wN #1	2671
2672	\exp_after:wN #2	2672
2673	\exp:w \exp_end_continue_f:w #3	2673
2674	}	2674
2675	\cs_new:Npn \exp_last_unbraced:Nco #1#2#3	2675
2676	{ \exp_after:wN #1 \cs:w #2 \exp_after:wN \cs_end: #3 }	2676
2677	\cs_new:Npn \exp_last_unbraced:NcV #1#2#3	2677
2678	{	2678
2679	\exp_after:wN #1	2679
2680	\cs:w #2 \exp_after:wN \cs_end:	2680
2681	\exp:w __exp_eval_register:N #3	2681
2682	}	2682
2683	\cs_new:Npn \exp_last_unbraced:NNNo #1#2#3#4	2683
2684	{ \exp_after:wN #1 \exp_after:wN #2 \exp_after:wN #3 #4 }	2684
2685	\cs_new:Npn \exp_last_unbraced:NNNV #1#2#3#4	2685
2686	{	2686
2687	\exp_after:wN #1	2687
2688	\exp_after:wN #2	2688
2689	\exp_after:wN #3	2689
2690	\exp:w __exp_eval_register:N #4	2690
2691	}	2691
2692	\cs_new:Npn \exp_last_unbraced:NNNf #1#2#3#4	2692
2693	{	2693
2694	\exp_after:wN #1	2694
2695	\exp_after:wN #2	2695
2696	\exp_after:wN #3	2696
2697	\exp:w \exp_end_continue_f:w #4	2697
2698	}	2698
2699	\cs_new:Npn \exp_last_unbraced:Nno { \::n \::o_unbraced \::: }	2699
2700	\cs_new:Npn \exp_last_unbraced:Nnf { \::n \::f_unbraced \::: }	2700
2701	\cs_new:Npn \exp_last_unbraced:Noo { \::o \::o_unbraced \::: }	2701
2702	\cs_new:Npn \exp_last_unbraced:Nfo { \::f \::o_unbraced \::: }	2702
2703	\cs_new:Npn \exp_last_unbraced:NnNo { \::n \::N \::o_unbraced \::: }	2703
2704	\cs_new:Npn \exp_last_unbraced:NNNNo #1#2#3#4#5	2704
2705	{ \exp_after:wN #1 \exp_after:wN #2 \exp_after:wN #3 \exp_after:wN #4 #5 }	2705
2706	\cs_new:Npn \exp_last_unbraced:NNNNf #1#2#3#4#5	2706
2707	{	2707
2708	\exp_after:wN #1	2708
2709	\exp_after:wN #2	2709
2710	\exp_after:wN #3	2710
2711	\exp_after:wN #4	2711
2712	\exp:w \exp_end_continue_f:w #5	2712
2713	}	2713
2714	\cs_new_protected:Npn \exp_last_unbraced:Nx { \::x_unbraced \::: }	2714

```

2715 \cs_new:Npn \exp_last_two_unbraced:Noo #1#2#3 2715
2716 { \exp_after:wN \exp_last_two_unbraced:noN \exp_after:wN {#3} {#2} #1 } 2716
2717 \cs_new:Npn \exp_last_two_unbraced:noN #1#2#3 2717
2718 { \exp_after:wN #3 #2 #1 } 2718
2719 \cs_new_eq:NN \__kernel_exp_not:w \tex_unexpanded:D 2719
2720 \cs_new:Npn \exp_not:c #1 { \exp_after:wN \exp_not:N \cs:w #1 \cs_end: } 2720
2721 \cs_new:Npn \exp_not:o #1 { \__kernel_exp_not:w \exp_after:wN {#1} } 2721
2722 \cs_new:Npn \exp_not:e #1 2722
2723 { \__kernel_exp_not:w \tex_expanded:D { {#1} } } 2723
2724 \cs_new:Npn \exp_not:f #1 2724
2725 { \__kernel_exp_not:w \exp_after:wN { \exp:w \exp_end_continue_f:w #1 } } 2725
2726 \cs_new:Npn \exp_not:V #1 2726
2727 { 2727
2728 \__kernel_exp_not:w \exp_after:wN 2728
2729 { \exp:w \exp_eval_register:N #1 } 2729
2730 } 2730
2731 \cs_new:Npn \exp_not:v #1 2731
2732 { 2732
2733 \__kernel_exp_not:w \exp_after:wN 2733
2734 { \exp:w \exp_eval_register:c {#1} } 2734
2735 } 2735
2736 \group_begin: 2736
2737 \tex_catcode:D \^^@ = 13 2737
2738 \cs_new_protected:Npn \exp_end_continue_f:w { \^^@ } 2738
2739 \if_cs_exist:N \^^@ 2739
2740 \else: 2740
2741 \cs_new:Npn \^^@ 2741
2742 { \msg_expandable_error:nn { kernel } { bad-exp-end-f } } 2742
2743 \fi: 2743
2744 \cs_new:Npn \exp_end_continue_f:nw #1 { \^^@ #1 } 2744
2745 \group_end: 2745
2746 \cs_new_eq:NN \s__cs_mark \scan_stop: 2746
2747 \cs_new_eq:NN \s__cs_stop \scan_stop: 2747
2748 \cs_new:Npn \q__cs_recursion_stop { \q__cs_recursion_stop } 2748
2749 \cs_new:Npn \__cs_use_none_delimit_by_s_stop:w #1 \s__cs_stop { } 2749
2750 \cs_new:Npn \__cs_use_i_delimit_by_s_stop:nw #1 #2 \s__cs_stop {#1} 2750
2751 \cs_new:Npn \__cs_use_none_delimit_by_q_recursion_stop:w 2751
2752 #1 \q__cs_recursion_stop { } 2752
2753 \cs_new_protected:Npn \cs_generate_variant:Nn #1#2 2753
2754 { 2754
2755 \__cs_generate_variant:N #1 2755
2756 \use:e 2756
2757 { 2757
2758 \__cs_generate_variant:nnNN 2758
2759 \cs_split_function:N #1 2759
2760 \exp_not:N #1 2760

```

```

2761         \tl_to_str:n {#2} ,
2762         \exp_not:N \scan_stop: ,
2763         \exp_not:N \q__cs_recursion_stop
2764     }
2765 }
2766 \cs_new_protected:Npn \cs_generate_variant:cn
2767 { \exp_args:Nc \cs_generate_variant:Nn }
2768 \cs_new_protected:Npe \__cs_generate_variant:N #1
2769 {
2770     \exp_not:N \exp_after:wN \exp_not:N \if_meaning:w
2771     \exp_not:N \exp_not:N #1 #1
2772     \cs_set_eq:NN \exp_not:N \__cs_tmp:w \cs_new_protected:Npe
2773     \exp_not:N \else:
2774     \exp_not:N \exp_after:wN \exp_not:N \__cs_generate_variant:ww
2775     \exp_not:N \token_to_meaning:N #1 \tl_to_str:n { ma }
2776     \s__cs_mark
2777     \s__cs_mark \cs_new_protected:Npe
2778     \tl_to_str:n { pr }
2779     \s__cs_mark \cs_new:Npe
2780     \s__cs_stop
2781     \exp_not:N \fi:
2782 }
2783 \exp_last_unbraced:NNNNNo
2784 \cs_new_protected:Npn \__cs_generate_variant:ww
2785 #1 { \tl_to_str:n { ma } } #2 \s__cs_mark
2786 { \__cs_generate_variant:wwNw #1 }
2787 \exp_last_unbraced:NNNNNo
2788 \cs_new_protected:Npn \__cs_generate_variant:wwNw
2789 #1 { \tl_to_str:n { pr } } #2 \s__cs_mark #3 #4 \s__cs_stop
2790 { \cs_set_eq:NN \__cs_tmp:w #3 }
2791 \cs_new_protected:Npn \__cs_generate_variant:nnNN #1#2#3#4
2792 {
2793     \if_meaning:w \c_false_bool #3
2794     \msg_error:nne { kernel } { missing-colon }
2795     { \token_to_str:c {#1} }
2796     \exp_after:wN \__cs_use_none_delimit_by_q_recursion_stop:w
2797     \fi:
2798     \__cs_generate_variant:Nnnw #4 {#1}{#2}
2799 }
2800 \cs_new_protected:Npn \__cs_generate_variant:Nnnw #1#2#3#4 ,
2801 {
2802     \if_meaning:w \scan_stop: #4
2803     \exp_after:wN \__cs_use_none_delimit_by_q_recursion_stop:w
2804     \fi:
2805     \use:e
2806     {

```


2807	\exp_not:N __cs_generate_variant:wwNN	2807
2808	__cs_generate_variant_loop:nNwN { }	2808
2809	#4	2809
2810	__cs_generate_variant_loop_end:nwwwNNnn	2810
2811	\s__cs_mark	2811
2812	#3 ~	2812
2813	{ ~ { } \fi: __cs_generate_variant_loop_long:wNNnn } ~	2813
2814	{ }	2814
2815	\s__cs_stop	2815
2816	\exp_not:N #1 {#2} {#4}	2816
2817	}	2817
2818	__cs_generate_variant:Nnnw #1 {#2} {#3}	2818
2819	}	2819
2820	\cs_new:Npn __cs_generate_variant_loop:nNwN #1#2#3 \s__cs_mark #4	2820
2821	{	2821
2822	\if:w #2 #4	2822
2823	\exp_after:wN __cs_generate_variant_loop_same:w	2823
2824	\else:	2824
2825	\if:w #4 __cs_generate_variant_loop_base:N #2 \else:	2825
2826	\if:w 0	2826
2827	\if:w N #4 \else: \if:w n #4 \else: 1 \fi: \fi:	2827
2828	\if:w \scan_stop: __cs_generate_variant_loop_base:N #2 1 \fi:	2828
2829	0	2829
2830	__cs_generate_variant_loop_special:NNwNNnn #4#2	2830
2831	\else:	2831
2832	__cs_generate_variant_loop_invalid:NNwNNnn #4#2	2832
2833	\fi:	2833
2834	\fi:	2834
2835	\fi:	2835
2836	#1	2836
2837	\prg_do_nothing:	2837
2838	#2	2838
2839	__cs_generate_variant_loop:nNwN { } #3 \s__cs_mark	2839
2840	}	2840
2841	\cs_new:Npn __cs_generate_variant_loop_base:N #1	2841
2842	{	2842
2843	\if:w c #1 N \else:	2843
2844	\if:w o #1 n \else:	2844
2845	\if:w V #1 n \else:	2845
2846	\if:w v #1 n \else:	2846
2847	\if:w f #1 n \else:	2847
2848	\if:w e #1 n \else:	2848
2849	\if:w x #1 n \else:	2849
2850	\if:w n #1 n \else:	2850
2851	\if:w N #1 N \else:	2851
2852	\scan_stop:	2852

2853	\fi:	2853
2854	\fi:	2854
2855	\fi:	2855
2856	\fi:	2856
2857	\fi:	2857
2858	\fi:	2858
2859	\fi:	2859
2860	\fi:	2860
2861	\fi:	2861
2862	}	2862
2863	\cs_new:Npn __cs_generate_variant_loop_same:w	2863
2864	#1 \prg_do_nothing: #2#3#4	2864
2865	{ #3 { #1 __cs_generate_variant_same:N #2 } }	2865
2866	\cs_new:Npn __cs_generate_variant_loop_end:nwwwNNnn	2866
2867	#1#2 \s__cs_mark #3 ~ #4 \s__cs_stop #5#6#7#8	2867
2868	{	2868
2869	\scan_stop: \scan_stop: \fi:	2869
2870	\s__cs_mark \s__cs_stop	2870
2871	\exp_not:N #6	2871
2872	\exp_not:c { #7 : #8 #1 #3 }	2872
2873	}	2873
2874	\cs_new:Npn __cs_generate_variant_loop_long:wNNnn #1 \s__cs_stop #2#3#4#5	2874
2875	{	2875
2876	\exp_not:n	2876
2877	{	2877
2878	\s__cs_mark	2878
2879	\msg_error:nnee { kernel } { variant-too-long }	2879
2880	{#5} { \token_to_str:N #3 }	2880
2881	\use_none:nnn	2881
2882	\s__cs_stop	2882
2883	#3	2883
2884	#3	2884
2885	}	2885
2886	}	2886
2887	\cs_new:Npn __cs_generate_variant_loop_invalid:NNwNNnn	2887
2888	#1#2 \fi: \fi: \fi: #3 \s__cs_stop #4#5#6#7	2888
2889	{	2889
2890	\fi: \fi: \fi:	2890
2891	\exp_not:n	2891
2892	{	2892
2893	\s__cs_mark	2893
2894	\msg_error:nneeee { kernel } { invalid-variant }	2894
2895	{#7} { \token_to_str:N #5 } {#1} {#2}	2895
2896	\use_none:nnn	2896
2897	\s__cs_stop	2897
2898	#5	2898

```
2899         #5
2900     }
2901 }
2902 \cs_new:Npn \__cs_generate_variant_loop_special:NNwNNnn
2903   #1#2#3 \s__cs_stop #4#5#6#7
2904 {
2905     #3 \s__cs_stop #4 #5 {#6} {#7}
2906     \exp_not:n
2907     {
2908         \msg_error:nneeee
2909         { kernel } { deprecated-variant }
2910         {#7} { \token_to_str:N #5 } {#1} {#2}
2911     }
2912 }
2913 \cs_new:Npn \__cs_generate_variant_same:N #1
2914 {
2915     \if:w N #1 #1 \else:
2916         \if:w p #1 #1 \else:
2917             \token_to_str:N n
2918             \if:w n #1 \else:
2919                 \__cs_generate_variant_loop_special:NNwNNnn #1#1
2920             \fi:
2921         \fi:
2922     \fi:
2923 }
2924 \cs_new_protected:Npn \__cs_generate_variant:wwNN
2925   #1 \s__cs_mark #2 \s__cs_stop #3#4
2926 {
2927     #2
2928     \cs_if_free:NT #4
2929     {
2930         \group_begin:
2931             \__cs_generate_internal_variant:n {#1}
2932             \__cs_tmp:w #4 { \exp_not:c { exp_args:N #1 } \exp_not:N #3 }
2933         \group_end:
2934     }
2935 }
2936 \cs_new_protected:Npe \__cs_generate_internal_variant:n #1
2937 {
2938     \exp_not:N \__cs_generate_internal_variant:wwnNwn
2939     #1 \s__cs_mark
2940     { \cs_set_eq:NN \exp_not:N \__cs_tmp:w \cs_new_protected:Npe }
2941     \cs_new_protected:cpn
2942     \use:x
2943     \token_to_str:N x \s__cs_mark
2944     { }
```

```

2945         \cs_new:cpn                                2945
2946         \exp_not:N \tex_expanded:D                    2946
2947     \s__cs_stop                                2947
2948     {#1}                                2948
2949 }                                2949
2950 \exp_last_unbraced:NNNNNo                                2950
2951     \cs_new_protected:Npn \__cs_generate_internal_variant:wwnNwn #1 2951
2952     { \token_to_str:N x } #2 \s__cs_mark #3#4#5#6 \s__cs_stop #7 2952
2953 {                                2953
2954     #3                                2954
2955     \cs_if_free:cT { exp_args:N #7 }                2955
2956     { \__cs_generate_internal_variant:NNn #4 #5 {#7} } 2956
2957 }                                2957
2958 \cs_set_protected:Npn \__cs_tmp:w #1                2958
2959 {                                2959
2960     \cs_new_protected:Npn \__cs_generate_internal_variant:NNn ##1##2##3 2960
2961     {                                2961
2962         \if_catcode:w X \use_none:nnnnnnnn ##3        2962
2963         \prg_do_nothing: \prg_do_nothing: \prg_do_nothing: 2963
2964         \prg_do_nothing: \prg_do_nothing: \prg_do_nothing: 2964
2965         \prg_do_nothing: \prg_do_nothing: X            2965
2966         \exp_after:wN \__cs_generate_internal_test:Nw \exp_after:wN ##2 2966
2967     \else:                                2967
2968         \exp_after:wN \__cs_generate_internal_test_aux:w \exp_after:wN #1 2968
2969     \fi:                                2969
2970     ##3                                2970
2971     \s__cs_mark                                2971
2972     {                                2972
2973         \use:e                                2973
2974         {                                2974
2975             ##1 { exp_args:N ##3 }                2975
2976             { \__cs_generate_internal_variant_loop:n ##3 { : \use_i:nn } } 2976
2977         }                                2977
2978     }                                2978
2979     #1                                2979
2980     \s__cs_mark                                2980
2981     { \exp_not:n { \__cs_generate_internal_one_go:NNn ##1 ##2 {##3} } } 2981
2982     \s__cs_stop                                2982
2983 }                                2983
2984 \cs_new_protected:Npn \__cs_generate_internal_test_aux:w 2984
2985     ##1 #1 ##2 \s__cs_mark ##3 ##4 \s__cs_stop {##3} 2985
2986 \cs_new_eq:NN \__cs_generate_internal_test:Nw            2986
2987     \__cs_generate_internal_test_aux:w                2987
2988 }                                2988
2989 \exp_args:No \__cs_tmp:w { \token_to_str:N p } 2989
2990 \cs_new_protected:Npn \__cs_generate_internal_one_go:NNn #1#2#3 2990

```

```
2991 {
2992     \__cs_generate_internal_loop:nwnnw
2993     { \exp_not:N ##1 } 1 . { } { }
2994     #3 { ? \__cs_generate_internal_end:w } X ;
2995     23456789 { ? \__cs_generate_internal_long:w } ;
2996     #1 #2 {#3}
2997 }
2998 \cs_new_protected:Npn \__cs_generate_internal_loop:nwnnw #1#2 . #3#4#5#6 ; #7
2999 {
3000     \use_none:n #5
3001     \use_none:n #7
3002     \cs_if_exist_use:cF { __cs_generate_internal_#5:NN }
3003     { \__cs_generate_internal_other:NN }
3004     #5 #7
3005     #7 .
3006     { #3 #1 } { #4 ## #2 }
3007     #6 ;
3008 }
3009 \cs_new_protected:Npn \__cs_generate_internal_N:NN #1#2
3010 { \__cs_generate_internal_loop:nwnnw { \exp_not:N ###2 } }
3011 \cs_new_protected:Npn \__cs_generate_internal_c:NN #1#2
3012 { \exp_args:No \__cs_generate_internal_loop:nwnnw { \exp_not:c {###2} } }
3013 \cs_new_protected:Npn \__cs_generate_internal_n:NN #1#2
3014 { \__cs_generate_internal_loop:nwnnw { { \exp_not:n {###2} } } }
3015 \cs_new_protected:Npn \__cs_generate_internal_x:NN #1#2
3016 { \__cs_generate_internal_loop:nwnnw { {###2} } }
3017 \cs_new_protected:Npn \__cs_generate_internal_other:NN #1#2
3018 {
3019     \exp_args:No \__cs_generate_internal_loop:nwnnw
3020     {
3021         \exp_after:wN
3022         {
3023             \exp:w \exp_args:NNc \exp_after:wN \exp_end:
3024             { exp_not:#1 } {###2}
3025         }
3026     }
3027 }
3028 \cs_new_protected:Npn \__cs_generate_internal_end:w #1 . #2#3#4 ; #5 ; #6#7#8
3029 { #6 { exp_args:N #8 } #3 { #7 {#2} } }
3030 \cs_new_protected:Npn \__cs_generate_internal_long:w #1 N #2#3 . #4#5#6#
3031 {
3032     \exp_args:Nx \__cs_generate_internal_long:nnnNNn
3033     { \__cs_generate_internal_variant_loop:n #2 #6 { : \use_i:nn } }
3034     {#4} {#5}
3035 }
3036 \cs_new:Npn \__cs_generate_internal_long:nnnNNn #1#2#3#4 ; ; #5#6#7
```



```
3037 { #5 { exp_args:N #7 } #3 { #6 { \exp_not:n {#1} {#2} } } } 3037
3038 \cs_new:Npn \__cs_generate_internal_variant_loop:n #1 3038
3039 { 3039
3040 \exp_after:wN \exp_not:N \cs:w :: #1 \cs_end: 3040
3041 \__cs_generate_internal_variant_loop:n 3041
3042 } 3042
3043 \cs_new_protected:Npn \prg_generate_conditional_variant:Nnn #1 3043
3044 { 3044
3045 \use:e 3045
3046 { 3046
3047 \__cs_generate_variant:nnNnn 3047
3048 \cs_split_function:N #1 3048
3049 } 3049
3050 } 3050
3051 \cs_new_protected:Npn \__cs_generate_variant:nnNnn #1#2#3#4#5 3051
3052 { 3052
3053 \if_meaning:w \c_false_bool #3 3053
3054 \msg_error:nne { kernel } { missing-colon } 3054
3055 { \token_to_str:c {#1} } 3055
3056 \__cs_use_i_delimit_by_s_stop:nw 3056
3057 \fi: 3057
3058 \exp_after:wN \__cs_generate_variant:w 3058
3059 \tl_to_str:n {#5} , \scan_stop: , \q__cs_recursion_stop 3059
3060 \__cs_use_none_delimit_by_s_stop:w \s__cs_mark {#1} {#2} {#4} \s__cs_stop 3060
3061 } 3061
3062 \cs_new_protected:Npn \__cs_generate_variant:w 3062
3063 #1 , #2 \s__cs_mark #3#4#5 3063
3064 { 3064
3065 \if_meaning:w \scan_stop: #1 \scan_stop: 3065
3066 \if_meaning:w \q__cs_nil #1 \q__cs_nil 3066
3067 \use_i:nnn 3067
3068 \fi: 3068
3069 \exp_after:wN \__cs_use_none_delimit_by_q_recursion_stop:w 3069
3070 \else: 3070
3071 \cs_if_exist_use:cTF { __cs_generate_variant_#1_form:nnn } 3071
3072 { {#3} {#4} {#5} } 3072
3073 { 3073
3074 \msg_error:nnee 3074
3075 { kernel } { conditional-form-unknown } 3075
3076 {#1} { \token_to_str:c { #3 : #4 } } 3076
3077 } 3077
3078 \fi: 3078
3079 \__cs_generate_variant:w #2 \s__cs_mark {#3} {#4} {#5} 3079
3080 } 3080
3081 \cs_new_protected:Npn \__cs_generate_variant_p_form:nnn #1#2 3081
3082 { \__cs_generate_variant_check:nn { #1 _p : #2 } } 3082
```

```
3083 \cs_new_protected:Npn \__cs_generate_variant_T_form:nnn #1#2 3083
3084 { \__cs_generate_variant_check:nn { #1 : #2 T } } 3084
3085 \cs_new_protected:Npn \__cs_generate_variant_F_form:nnn #1#2 3085
3086 { \__cs_generate_variant_check:nn { #1 : #2 F } } 3086
3087 \cs_new_protected:Npn \__cs_generate_variant_TF_form:nnn #1#2 3087
3088 { \__cs_generate_variant_check:nn { #1 : #2 TF } } 3088
3089 \cs_new_protected:Npn \__cs_generate_variant_check:nn #1#2 3089
3090 { 3090
3091   \cs_if_exist:cTF {#1} 3091
3092   { \cs_generate_variant:cn {#1} {#2} } 3092
3093   { 3093
3094     \msg_error:nne 3094
3095     { kernel } { conditional-base-undefined } 3095
3096     { \token_to_str:c {#1} } 3096
3097   } 3097
3098 } 3098
3099 \cs_new_protected:Npn \exp_args_generate:n #1 3099
3100 { 3100
3101   \exp_args:No \clist_map_inline:nn { \tl_to_str:n {#1} } 3101
3102   { 3102
3103     \str_map_inline:nn {##1} 3103
3104     { 3104
3105       \str_if_in:nnF { NnpcofeVvx } {####1} 3105
3106       { 3106
3107         \msg_error:nnnn { kernel } { invalid-exp-args } 3107
3108         {####1} {##1} 3108
3109         \str_map_break:n { \use_none:nn } 3109
3110       } 3110
3111     } 3111
3112     \__cs_generate_internal_variant:n {##1} 3112
3113   } 3113
3114 } 3114
3115 \cs_set_protected:Npn \__cs_tmp:w #1 3115
3116 { 3116
3117   \group_begin: 3117
3118   \exp_args:No \__cs_generate_internal_variant:n 3118
3119   { \tl_to_str:n {#1} } 3119
3120   \group_end: 3120
3121 } 3121
3122 \__cs_tmp:w { nc } 3122
3123 \__cs_tmp:w { no } 3123
3124 \__cs_tmp:w { nV } 3124
3125 \__cs_tmp:w { nv } 3125
3126 \__cs_tmp:w { ne } 3126
3127 \__cs_tmp:w { nf } 3127
3128 \__cs_tmp:w { oc } 3128
```

3129	__cs_tmp:w { oo }	3129
3130	__cs_tmp:w { of }	3130
3131	__cs_tmp:w { Vo }	3131
3132	__cs_tmp:w { fo }	3132
3133	__cs_tmp:w { ff }	3133
3134	__cs_tmp:w { ee }	3134
3135	__cs_tmp:w { Nx }	3135
3136	__cs_tmp:w { cx }	3136
3137	__cs_tmp:w { nx }	3137
3138	__cs_tmp:w { ox }	3138
3139	__cs_tmp:w { xo }	3139
3140	__cs_tmp:w { xx }	3140
3141	__cs_tmp:w { Ncf }	3141
3142	__cs_tmp:w { Nno }	3142
3143	__cs_tmp:w { NnV }	3143
3144	__cs_tmp:w { Noo }	3144
3145	__cs_tmp:w { NVV }	3145
3146	__cs_tmp:w { cno }	3146
3147	__cs_tmp:w { cnV }	3147
3148	__cs_tmp:w { coo }	3148
3149	__cs_tmp:w { cVV }	3149
3150	__cs_tmp:w { nnc }	3150
3151	__cs_tmp:w { nno }	3151
3152	__cs_tmp:w { nnf }	3152
3153	__cs_tmp:w { nff }	3153
3154	__cs_tmp:w { ooo }	3154
3155	__cs_tmp:w { oof }	3155
3156	__cs_tmp:w { ffo }	3156
3157	__cs_tmp:w { eee }	3157
3158	__cs_tmp:w { NNx }	3158
3159	__cs_tmp:w { Nnx }	3159
3160	__cs_tmp:w { Nox }	3160
3161	__cs_tmp:w { nnx }	3161
3162	__cs_tmp:w { nox }	3162
3163	__cs_tmp:w { ccx }	3163
3164	__cs_tmp:w { cnx }	3164
3165	__cs_tmp:w { oox }	3165
3166	\cs_generate_variant:Nn \cs_generate_from_arg_count:NNnn { NNno }	3166
3167	\cs_generate_variant:Nn \cs_replacement_spec:N { c }	3167
3168	%% File: l3quark.dtx	3168
3169	\cs_new_protected:Npn \quark_new:N #1	3169
3170	{	3170
3171	__kernel_chk_if_free_cs:N #1	3171
3172	\cs_gset_nopar:Npn #1 {#1}	3172
3173	}	3173
3174	\quark_new:N \q_nil	3174

3175	\quark_new:N \q_mark	3175
3176	\quark_new:N \q_no_value	3176
3177	\quark_new:N \q_stop	3177
3178	\quark_new:N \q_recursion_tail	3178
3179	\quark_new:N \q_recursion_stop	3179
3180	\cs_new_eq:NN \s__quark \scan_stop:	3180
3181	\quark_new:N \q__quark_nil	3181
3182	\cs_new:Npn \quark_if_recursion_tail_stop:N #1	3182
3183	{	3183
3184	\if_meaning:w \q_recursion_tail #1	3184
3185	\exp_after:wN \use_none_delimit_by_q_recursion_stop:w	3185
3186	\fi:	3186
3187	}	3187
3188	\cs_new:Npn \quark_if_recursion_tail_stop_do:Nn #1	3188
3189	{	3189
3190	\if_meaning:w \q_recursion_tail #1	3190
3191	\exp_after:wN \use_i_delimit_by_q_recursion_stop:nw	3191
3192	\else:	3192
3193	\exp_after:wN \use_none:n	3193
3194	\fi:	3194
3195	}	3195
3196	\cs_new:Npn \quark_if_recursion_tail_stop:n #1	3196
3197	{	3197
3198	\tl_if_empty:oTF	3198
3199	{ __quark_if_recursion_tail:w {} #1 {} ?! \q_recursion_tail ??? }	3199
3200	{ \use_none_delimit_by_q_recursion_stop:w }	3200
3201	{ }	3201
3202	}	3202
3203	\cs_new:Npn \quark_if_recursion_tail_stop_do:nn #1	3203
3204	{	3204
3205	\tl_if_empty:oTF	3205
3206	{ __quark_if_recursion_tail:w {} #1 {} ?! \q_recursion_tail ??? }	3206
3207	{ \use_i_delimit_by_q_recursion_stop:nw }	3207
3208	{ \use_none:n }	3208
3209	}	3209
3210	\cs_new:Npn __quark_if_recursion_tail:w	3210
3211	#1 \q_recursion_tail #2 ? #3 ?! { #1 #2 }	3211
3212	\cs_generate_variant:Nn \quark_if_recursion_tail_stop:n { o }	3212
3213	\cs_generate_variant:Nn \quark_if_recursion_tail_stop_do:nn { o }	3213
3214	\cs_new:Npn \quark_if_recursion_tail_break:NN #1#2	3214
3215	{	3215
3216	\if_meaning:w \q_recursion_tail #1	3216
3217	\exp_after:wN #2	3217
3218	\fi:	3218
3219	}	3219
3220	\cs_new:Npn \quark_if_recursion_tail_break:nN #1#2	3220

3221	{	3221
3222	\tl_if_empty:oT	3222
3223	{ _quark_if_recursion_tail:w {} #1 {} ?! \q_recursion_tail ??? }	3223
3224	{#2}	3224
3225	}	3225
3226	\prg_new_conditional:Npnn \quark_if_nil:N #1 { p, T , F , TF }	3226
3227	{	3227
3228	\if_meaning:w \q_nil #1	3228
3229	\prg_return_true:	3229
3230	\else:	3230
3231	\prg_return_false:	3231
3232	\fi:	3232
3233	}	3233
3234	\prg_new_conditional:Npnn \quark_if_no_value:N #1 { p, T , F , TF }	3234
3235	{	3235
3236	\if_meaning:w \q_no_value #1	3236
3237	\prg_return_true:	3237
3238	\else:	3238
3239	\prg_return_false:	3239
3240	\fi:	3240
3241	}	3241
3242	\prg_generate_conditional_variant:Nnn \quark_if_no_value:N	3242
3243	{ c } { p , T , F , TF }	3243
3244	\prg_new_conditional:Npnn \quark_if_nil:n #1 { p, T , F , TF }	3244
3245	{	3245
3246	_quark_if_empty_if:o	3246
3247	{ _quark_if_nil:w {} #1 {} ? ! \q_nil ? ? ! }	3247
3248	\prg_return_true:	3248
3249	\else:	3249
3250	\prg_return_false:	3250
3251	\fi:	3251
3252	}	3252
3253	\cs_new:Npn _quark_if_nil:w #1 \q_nil #2 ? #3 ? ! { #1 #2 }	3253
3254	\prg_new_conditional:Npnn \quark_if_no_value:n #1 { p, T , F , TF }	3254
3255	{	3255
3256	_quark_if_empty_if:o	3256
3257	{ _quark_if_no_value:w {} #1 {} ? ! \q_no_value ? ? ! }	3257
3258	\prg_return_true:	3258
3259	\else:	3259
3260	\prg_return_false:	3260
3261	\fi:	3261
3262	}	3262
3263	\cs_new:Npn _quark_if_no_value:w #1 \q_no_value #2 ? #3 ? ! { #1 #2 }	3263
3264	\prg_generate_conditional_variant:Nnn \quark_if_nil:n	3264
3265	{ V , o } { p , TF , T , F }	3265
3266	\cs_new:Npn _quark_if_empty_if:o #1	3266

```
3267 { 3267
3268     \exp_after:wN \if_meaning:w \exp_after:wN \q_nil 3268
3269     \__kernel_tl_to_str:w \exp_after:wN {#1} \q_nil 3269
3270 } 3270
3271 \cs_new_protected:Npn \__kernel_quark_new_test:N #1 3271
3272 { \__quark_new_test_aux:Ne #1 { \__quark_module_name:N #1 } } 3272
3273 \cs_new_protected:Npn \__quark_new_test_aux:Nn #1 #2 3273
3274 { 3274
3275     \if_meaning:w \q_nil #2 \q_nil 3275
3276     \msg_error:nne { quark } { invalid-function } 3276
3277     { \token_to_str:N #1 } 3277
3278 \else: 3278
3279     \__quark_new_test:Nccn #1 3279
3280     { q__#2_recursion_tail } { q__#2_recursion_stop } { __#2 } 3280
3281 \fi: 3281
3282 } 3282
3283 \cs_generate_variant:Nn \__quark_new_test_aux:Nn { Ne } 3283
3284 \cs_new_protected:Npn \__quark_new_test:NNNn #1 3284
3285 { 3285
3286     \exp_last_unbraced:Nf \__quark_new_test_aux:nnNNnnnn 3286
3287     { \cs_split_function:N #1 } 3287
3288     #1 { test } 3288
3289 } 3289
3290 \cs_generate_variant:Nn \__quark_new_test:NNNn { Ncc } 3290
3291 \cs_new_protected:Npn \__kernel_quark_new_conditional:Nn #1 3291
3292 { 3292
3293     \__quark_new_conditional:Neen #1 3293
3294     { \__quark_quark_conditional_name:N #1 } 3294
3295     { \__quark_module_name:N #1 } 3295
3296 } 3296
3297 \cs_new_protected:Npn \__quark_new_conditional:Nnnn #1#2#3#4 3297
3298 { 3298
3299     \if_meaning:w \q_nil #2 \q_nil 3299
3300     \msg_error:nne { quark } { invalid-function } 3300
3301     { \token_to_str:N #1 } 3301
3302 \else: 3302
3303     \if_meaning:w \q_nil #3 \q_nil 3303
3304     \msg_error:nne { quark } { invalid-function } 3304
3305     { \token_to_str:N #1 } 3305
3306 \else: 3306
3307     \exp_last_unbraced:Nf \__quark_new_test_aux:nnNNnnnn 3307
3308     { \cs_split_function:N #1 } 3308
3309     #1 { conditional } 3309
3310     {#2} {#3} {#4} 3310
3311 \fi: 3311
3312 \fi: 3312
```



```
3313 } 3313
3314 \cs_generate_variant:Nn \__quark_new_conditional:Nnnn { Nee } 3314
3315 \cs_new_protected:Npn \__quark_new_test_aux:nnNNnnnn #1 #2 #3 #4 #5 3315
3316 { 3316
3317     \cs_if_exist_use:cTF { \__quark_new_#5_#2:Nnnn } { #4 } 3317
3318     { 3318
3319         \msg_error:nnee { quark } { invalid-function } 3319
3320         { \token_to_str:N #4 } {#2} 3320
3321         \use_none:nnn 3321
3322     } 3322
3323 } 3323
3324 \cs_new_protected:Npn \__quark_new_test_n:Nnnn #1 #2 #3 #4 3324
3325 { 3325
3326     \__quark_new_test_aux_do:nNNnnnnNNn {#4} #2 #3 { none } { } { } { } 3326
3327     \__quark_new_test_define_tl:nNnNNn #1 { } 3327
3328 } 3328
3329 \cs_new_protected:Npn \__quark_new_test_nn:Nnnn #1 #2 #3 #4 3329
3330 { 3330
3331     \__quark_new_test_aux_do:nNNnnnnNNn {#4} #2 #3 { i } { n } {##1} {##2} 3331
3332     \__quark_new_test_define_tl:nNnNNn #1 { \use_none:n } 3332
3333 } 3333
3334 \cs_new_protected:Npn \__quark_new_test_nN:Nnnn #1 #2 #3 #4 3334
3335 { 3335
3336     \__quark_new_test_aux_do:nNNnnnnNNn {#4} #2 #3 { i } { n } {##1} {##2} 3336
3337     \__quark_new_test_define_break_tl:nNNNNn #1 { } 3337
3338 } 3338
3339 \cs_new_protected:Npn \__quark_new_test_N:Nnnn #1 #2 #3 #4 3339
3340 { 3340
3341     \__quark_new_test_aux_do:nNNnnnnNNn {#4} #2 #3 { none } { } { } { } { } 3341
3342     \__quark_new_test_define_ifx:nNnNNn #1 { } 3342
3343 } 3343
3344 \cs_new_protected:Npn \__quark_new_test_Nn:Nnnn #1 #2 #3 #4 3344
3345 { 3345
3346     \__quark_new_test_aux_do:nNNnnnnNNn {#4} #2 #3 { i } { n } {##1} {##2} 3346
3347     \__quark_new_test_define_ifx:nNnNNn #1 3347
3348     { \else: \exp_after:wN \use_none:n } 3348
3349 } 3349
3350 \cs_new_protected:Npn \__quark_new_test_NN:Nnnn #1 #2 #3 #4 3350
3351 { 3351
3352     \__quark_new_test_aux_do:nNNnnnnNNn {#4} #2 #3 { i } { n } {##1} {##2} 3352
3353     \__quark_new_test_define_break_ifx:nNNNNn #1 { } 3353
3354 } 3354
3355 \cs_new_protected:Npn \__quark_new_test_aux_do:nNNnnnnNNn #1 #2 #3 #4 #5 3355
3356 { 3356
3357     \exp_args:Ncc \__quark_test_define_aux:NNNNnnNNn 3357
3358     { #1 _quark_recursion_tail:w } 3358
```

```
3359 { #1 _use_ #4 _delimit_by_q_recursion_stop: #5 w } 3359
3360 #2 #3 3360
3361 } 3361
3362 \cs_new_protected:Npn \__quark_test_define_aux:NNNNnnNNn #1 #2 #3 #4 #5 #6 #7 3362
3363 { 3363
3364 \cs_gset:Npn #1 ##1 #3 ##2 ? ##3 ?! { ##1 ##2 } 3364
3365 \cs_gset:Npn #2 ##1 #6 #4 {#5} 3365
3366 #7 {##1} #1 #2 #3 3366
3367 } 3367
3368 \cs_new_protected:Npn \__quark_new_test_define_tl:nNnNNn #1 #2 #3 #4 #5 #6 3368
3369 { 3369
3370 \cs_new:Npn #5 #1 3370
3371 { 3371
3372 \tl_if_empty:oTF 3372
3373 { #2 {} ##1 {} ?! #4 ??! } 3373
3374 {#3} {#6} 3374
3375 } 3375
3376 } 3376
3377 \cs_new_protected:Npn \__quark_new_test_define_ifx:nNnNNn #1 #2 #3 #4 #5 #6 3377
3378 { 3378
3379 \cs_new:Npn #5 #1 3379
3380 { 3380
3381 \if_meaning:w #4 ##1 3381
3382 \exp_after:wN #3 3382
3383 #6 3383
3384 \fi: 3384
3385 } 3385
3386 } 3386
3387 \cs_new_protected:Npn \__quark_new_test_define_break_tl:nNNNNn #1 #2 #3 3387
3388 { \__quark_new_test_define_tl:nNnNNn {##1##2} #2 {##2} } 3388
3389 \cs_new_protected:Npn \__quark_new_test_define_break_ifx:nNNNNn #1 #2 #3 3389
3390 { \__quark_new_test_define_ifx:nNnNNn {##1##2} #2 {##2} } 3390
3391 \cs_new_protected:Npn \__quark_new_conditional_n:Nnnn #1 #2 #3 3391
3392 { 3392
3393 \exp_args:Ncc \__quark_new_conditional_n_aux:NNNn 3393
3394 { __ #3 _if_quark_ #2 :w } { q__ #3 _ #2 } #1 3394
3395 } 3395
3396 \cs_new_protected:Npn \__quark_new_conditional_N:Nnnn #1 #2 #3 3396
3397 { 3397
3398 \exp_args:NNc \__quark_new_conditional_N_aux:NNNn 3398
3399 \prg_do_nothing: { q__ #3 _ #2 } #1 3399
3400 } 3400
3401 \cs_new_protected:Npn \__quark_new_conditional_n_aux:NNNn #1 #2 #3 #4 3401
3402 { 3402
3403 \cs_gset:Npn #1 ##1 #2 ##2 ? ##3 ?! { ##1##2 } 3403
3404 \prg_new_conditional:Npnn #3 ##1 {#4} 3404
```

```

3405 {
3406     \__quark_if_empty_if:o { #1 {} ##1 {} ?! #2 ??! }
3407     \prg_return_true:
3408 \else:
3409     \prg_return_false:
3410 \fi:
3411 }
3412 }
3413 \cs_new_protected:Npn \__quark_new_conditional_N_aux:NNNn #1 #2 #3 #4
3414 {
3415     \prg_new_conditional:Npnn #3 ##1 {#4}
3416     {
3417         \if_meaning:w #2 ##1
3418         \prg_return_true:
3419 \else:
3420         \prg_return_false:
3421 \fi:
3422     }
3423 }
3424 \cs_set:Npn \__quark_tmp:w #1#2
3425 {
3426     \cs_new:Npn \__quark_module_name:N ##1
3427     {
3428         \exp_last_unbraced:Nf \__quark_module_name:w
3429         { \cs_to_str:N ##1 } #1 \s__quark
3430     }
3431     \cs_new:Npn \__quark_module_name:w ##1 #1 ##2 \s__quark
3432     { \__quark_module_name_loop:w ##1 #2 \use_none:n { } #2 \s__quark }
3433     \cs_new:Npn \__quark_module_name_loop:w ##1 #2
3434     {
3435         \use_i_ii:nnn \if_meaning:w \prg_do_nothing:
3436         ##1 \prg_do_nothing: \prg_do_nothing:
3437         \exp_after:wN \__quark_module_name_loop:w
3438 \else:
3439         \__quark_module_name_end:w ##1
3440 \fi:
3441     }
3442     \cs_new:Npn \__quark_module_name_end:w
3443     ##1 \fi: ##2 \s__quark { \fi: ##1 }
3444 }
3445 \exp_after:wN \__quark_tmp:w \tl_to_str:n { : _ }
3446 \cs_set:Npn \__quark_tmp:w #1 #2 \s__quark
3447 {
3448     \cs_new:Npn \__quark_quark_conditional_name:N ##1
3449     {
3450         \exp_last_unbraced:Nf \__quark_quark_conditional_name:w

```

```

3451         { \cs_to_str:N ##1 } #1 #2 #1 \s__quark 3451
3452     } 3452
3453     \cs_new:Npn \__quark_quark_conditional_name:w 3453
3454         ##1 #2 ##2 #1 ##3 \s__quark {##2} 3454
3455 } 3455
3456 \exp_after:wN \__quark_tmp:w \tl_to_str:n { : _quark_if_ } \s__quark 3456
3457 \cs_new_protected:Npn \scan_new:N #1 3457
3458 { 3458
3459     \tl_if_in:NnTF \g__scan_marks_tl { #1 } 3459
3460     { 3460
3461         \msg_error:nne { scanmark } { already-defined } 3461
3462         { \token_to_str:N #1 } 3462
3463     } 3463
3464     { 3464
3465         \tl_gput_right:Nn \g__scan_marks_tl {#1} 3465
3466         \cs_new_eq:NN #1 \scan_stop: 3466
3467     } 3467
3468 } 3468
3469 \cs_new_eq:NN \s_stop \scan_stop: 3469
3470 \cs_gset_nopar:Npn \g__scan_marks_tl 3470
3471 { 3471
3472     \s_stop 3472
3473     \s__quark 3473
3474     \s__cs_mark 3474
3475     \s__cs_stop 3475
3476 } 3476
3477 \cs_new:Npn \use_none_delimit_by_s_stop:w #1 \s_stop { } 3477
3478 %% File: l3tl.dtx 3478
3479 \cs_new_eq:NN \__kernel_tl_set:Nx \cs_set_nopar:Npe 3479
3480 \cs_new_eq:NN \__kernel_tl_gset:Nx \cs_gset_nopar:Npe 3480
3481 \cs_new_protected:Npn \tl_new:N #1 3481
3482 { 3482
3483     \__kernel_chk_if_free_cs:N #1 3483
3484     \cs_gset_eq:NN #1 \c_empty_tl 3484
3485 } 3485
3486 \cs_generate_variant:Nn \tl_new:N { c } 3486
3487 \cs_new_protected:Npn \tl_const:Nn #1#2 3487
3488 { 3488
3489     \__kernel_chk_if_free_cs:N #1 3489
3490     \cs_gset_nopar:Npe #1 { \__kernel_exp_not:w {#2} } 3490
3491 } 3491
3492 \cs_generate_variant:Nn \tl_const:Nn { NV , Ne , c , cV , ce } 3492
3493 \cs_generate_variant:Nn \tl_const:Nn { Nx , cx } 3493
3494 \cs_new_protected:Npn \tl_clear:N #1 3494
3495     { \tex_let:D #1 = ~ \c_empty_tl } 3495
3496 \cs_new_protected:Npn \tl_gclear:N #1 3496

```

3497	{ \tex_global:D \tex_let:D #1 ~ \c_empty_tl }	3497
3498	\cs_generate_variant:Nn \tl_clear:N { c }	3498
3499	\cs_generate_variant:Nn \tl_gclear:N { c }	3499
3500	\cs_new_protected:Npn \tl_clear_new:N #1	3500
3501	{ \tl_if_exist:NTF #1 { \tl_clear:N #1 } { \tl_new:N #1 } }	3501
3502	\cs_new_protected:Npn \tl_gclear_new:N #1	3502
3503	{ \tl_if_exist:NTF #1 { \tl_gclear:N #1 } { \tl_new:N #1 } }	3503
3504	\cs_generate_variant:Nn \tl_clear_new:N { c }	3504
3505	\cs_generate_variant:Nn \tl_gclear_new:N { c }	3505
3506	\cs_new_protected:Npn \tl_set_eq:NN #1#2	3506
3507	{ \tex_let:D #1 = ~ #2 }	3507
3508	\cs_new_protected:Npn \tl_gset_eq:NN #1#2	3508
3509	{ \tex_global:D \tex_let:D #1 = ~ #2 }	3509
3510	\cs_generate_variant:Nn \tl_set_eq:NN { cN, Nc, cc }	3510
3511	\cs_generate_variant:Nn \tl_gset_eq:NN { cN, Nc, cc }	3511
3512	\cs_new_protected:Npn \tl_concat:NNN #1#2#3	3512
3513	{	3513
3514	__kernel_tl_set:Nx #1	3514
3515	{	3515
3516	__kernel_exp_not:w \exp_after:wN {#2}	3516
3517	__kernel_exp_not:w \exp_after:wN {#3}	3517
3518	}	3518
3519	}	3519
3520	\cs_new_protected:Npn \tl_gconcat:NNN #1#2#3	3520
3521	{	3521
3522	__kernel_tl_gset:Nx #1	3522
3523	{	3523
3524	__kernel_exp_not:w \exp_after:wN {#2}	3524
3525	__kernel_exp_not:w \exp_after:wN {#3}	3525
3526	}	3526
3527	}	3527
3528	\cs_generate_variant:Nn \tl_concat:NNN { ccc }	3528
3529	\cs_generate_variant:Nn \tl_gconcat:NNN { ccc }	3529
3530	\prg_new_eq_conditional:NNn \tl_if_exist:N \cs_if_exist:N { TF , T , F , p }	3530
3531	\prg_new_eq_conditional:NNn \tl_if_exist:c \cs_if_exist:c { TF , T , F , p }	3531
3532	\tl_const:Nn \c_empty_tl { }	3532
3533	\group_begin:	3533
3534	\tex_catcode:D `~ = 11 ~	3534
3535	\tl_const:Ne \c_novalue_tl { - NoValue \token_to_str:N - }	3535
3536	\group_end:	3536
3537	\tl_const:Nn \c_space_tl { ~ }	3537
3538	\cs_new_protected:Npn \tl_set:Nn #1#2	3538
3539	{ __kernel_tl_set:Nx #1 { __kernel_exp_not:w {#2} } }	3539
3540	\cs_new_protected:Npn \tl_set:No #1#2	3540
3541	{ __kernel_tl_set:Nx #1 { __kernel_exp_not:w \exp_after:wN {#2} } }	3541
3542	\cs_new_protected:Npn \tl_gset:Nn #1#2	3542

3543 { _kernel_tl_gset:Nx #1 { _kernel_exp_not:w {#2} } } 3543
3544 \cs_new_protected:Npn \tl_gset:No #1#2 3544
3545 { _kernel_tl_gset:Nx #1 { _kernel_exp_not:w \exp_after:wN {#2} } } 3545
3546 \cs_generate_variant:Nn \tl_set:Nn { NV , Nv , Ne , Nf } 3546
3547 \cs_generate_variant:Nn \tl_set:Nn { c, cV , cv , ce , cf } 3547
3548 \cs_generate_variant:Nn \tl_set:No { c } 3548
3549 \cs_generate_variant:Nn \tl_set:Nn { Nx , cx } 3549
3550 \cs_generate_variant:Nn \tl_gset:Nn { NV , Nv , Ne , Nf } 3550
3551 \cs_generate_variant:Nn \tl_gset:Nn { c, cV , cv , ce , cf } 3551
3552 \cs_generate_variant:Nn \tl_gset:No { c } 3552
3553 \cs_generate_variant:Nn \tl_gset:Nn { Nx , cx } 3553
3554 \cs_new_protected:Npn \tl_put_left:Nn #1#2 3554
3555 { 3555
3556 _kernel_tl_set:Nx #1 3556
3557 { _kernel_exp_not:w {#2} _kernel_exp_not:w \exp_after:wN {#1} } 3557
3558 } 3558
3559 \cs_new_protected:Npn \tl_put_left:NV #1#2 3559
3560 { 3560
3561 _kernel_tl_set:Nx #1 3561
3562 { \exp_not:V #2 _kernel_exp_not:w \exp_after:wN {#1} } 3562
3563 } 3563
3564 \cs_new_protected:Npn \tl_put_left:Nv #1#2 3564
3565 { 3565
3566 _kernel_tl_set:Nx #1 3566
3567 { \exp_not:v {#2} _kernel_exp_not:w \exp_after:wN {#1} } 3567
3568 } 3568
3569 \cs_new_protected:Npn \tl_put_left:Ne #1#2 3569
3570 { 3570
3571 _kernel_tl_set:Nx #1 3571
3572 { 3572
3573 _kernel_exp_not:w \tex_expanded:D { {#2} } 3573
3574 _kernel_exp_not:w \exp_after:wN {#1} 3574
3575 } 3575
3576 } 3576
3577 \cs_new_protected:Npn \tl_put_left:No #1#2 3577
3578 { 3578
3579 _kernel_tl_set:Nx #1 3579
3580 { 3580
3581 _kernel_exp_not:w \exp_after:wN {#2} 3581
3582 _kernel_exp_not:w \exp_after:wN {#1} 3582
3583 } 3583
3584 } 3584
3585 \cs_new_protected:Npn \tl_gput_left:Nn #1#2 3585
3586 { 3586
3587 _kernel_tl_gset:Nx #1 3587
3588 { _kernel_exp_not:w {#2} _kernel_exp_not:w \exp_after:wN {#1} } 3588


```

3589 }
3590 \cs_new_protected:Npn \tl_gput_left:NV #1#2
3591 {
3592   \__kernel_tl_gset:Nx #1
3593   { \exp_not:V #2 \__kernel_exp_not:w \exp_after:wN {#1} }
3594 }
3595 \cs_new_protected:Npn \tl_gput_left:Nv #1#2
3596 {
3597   \__kernel_tl_gset:Nx #1
3598   { \exp_not:v {#2} \__kernel_exp_not:w \exp_after:wN {#1} }
3599 }
3600 \cs_new_protected:Npn \tl_gput_left:Ne #1#2
3601 {
3602   \__kernel_tl_gset:Nx #1
3603   {
3604     \__kernel_exp_not:w \tex_expanded:D { {#2} }
3605     \__kernel_exp_not:w \exp_after:wN {#1}
3606   }
3607 }
3608 \cs_new_protected:Npn \tl_gput_left:No #1#2
3609 {
3610   \__kernel_tl_gset:Nx #1
3611   {
3612     \__kernel_exp_not:w \exp_after:wN {#2}
3613     \__kernel_exp_not:w \exp_after:wN {#1}
3614   }
3615 }
3616 \cs_generate_variant:Nn \tl_put_left:Nn { c }
3617 \cs_generate_variant:Nn \tl_put_left:NV { c }
3618 \cs_generate_variant:Nn \tl_put_left:Nv { c }
3619 \cs_generate_variant:Nn \tl_put_left:Ne { c }
3620 \cs_generate_variant:Nn \tl_put_left:No { c }
3621 \cs_generate_variant:Nn \tl_put_left:Nn { Nx, cx }
3622 \cs_generate_variant:Nn \tl_gput_left:Nn { c }
3623 \cs_generate_variant:Nn \tl_gput_left:NV { c }
3624 \cs_generate_variant:Nn \tl_gput_left:Nv { c }
3625 \cs_generate_variant:Nn \tl_gput_left:Ne { c }
3626 \cs_generate_variant:Nn \tl_gput_left:No { c }
3627 \cs_generate_variant:Nn \tl_gput_left:Nn { Nx , cx }
3628 \cs_new_protected:Npn \tl_put_right:Nn #1#2
3629 { \__kernel_tl_set:Nx #1 { \__kernel_exp_not:w \exp_after:wN { #1 #2 } } }
3630 \cs_new_protected:Npn \tl_put_right:NV #1#2
3631 {
3632   \__kernel_tl_set:Nx #1
3633   { \__kernel_exp_not:w \exp_after:wN {#1} \exp_not:V #2 }
3634 }

```

3635	\cs_new_protected:Npn \tl_put_right:Nv #1#2	3635
3636	{	3636
3637	__kernel_tl_set:Nx #1	3637
3638	{ __kernel_exp_not:w \exp_after:wN {#1} \exp_not:v {#2} }	3638
3639	}	3639
3640	\cs_new_protected:Npn \tl_put_right:Ne #1#2	3640
3641	{	3641
3642	__kernel_tl_set:Nx #1	3642
3643	{	3643
3644	__kernel_exp_not:w \exp_after:wN {#1}	3644
3645	__kernel_exp_not:w \tex_expanded:D { {#2} }	3645
3646	}	3646
3647	}	3647
3648	\cs_new_protected:Npn \tl_put_right:No #1#2	3648
3649	{	3649
3650	__kernel_tl_set:Nx #1	3650
3651	{	3651
3652	__kernel_exp_not:w \exp_after:wN {#1}	3652
3653	__kernel_exp_not:w \exp_after:wN {#2}	3653
3654	}	3654
3655	}	3655
3656	\cs_new_protected:Npn \tl_gput_right:Nn #1#2	3656
3657	{ __kernel_tl_gset:Nx #1 { __kernel_exp_not:w \exp_after:wN { #1 #2 } } }	3657
3658	\cs_new_protected:Npn \tl_gput_right:NV #1#2	3658
3659	{	3659
3660	__kernel_tl_gset:Nx #1	3660
3661	{ __kernel_exp_not:w \exp_after:wN {#1} \exp_not:V #2 }	3661
3662	}	3662
3663	\cs_new_protected:Npn \tl_gput_right:Nv #1#2	3663
3664	{	3664
3665	__kernel_tl_gset:Nx #1	3665
3666	{ __kernel_exp_not:w \exp_after:wN {#1} \exp_not:v {#2} }	3666
3667	}	3667
3668	\cs_new_protected:Npn \tl_gput_right:Ne #1#2	3668
3669	{	3669
3670	__kernel_tl_gset:Nx #1	3670
3671	{	3671
3672	__kernel_exp_not:w \exp_after:wN {#1}	3672
3673	__kernel_exp_not:w \tex_expanded:D { {#2} }	3673
3674	}	3674
3675	}	3675
3676	\cs_new_protected:Npn \tl_gput_right:No #1#2	3676
3677	{	3677
3678	__kernel_tl_gset:Nx #1	3678
3679	{	3679
3680	__kernel_exp_not:w \exp_after:wN {#1}	3680

```
3681 \__kernel_exp_not:w \exp_after:wN {#2} 3681
3682 } 3682
3683 } 3683
3684 \cs_generate_variant:Nn \tl_put_right:Nn { c } 3684
3685 \cs_generate_variant:Nn \tl_put_right:NV { c } 3685
3686 \cs_generate_variant:Nn \tl_put_right:Nv { c } 3686
3687 \cs_generate_variant:Nn \tl_put_right:Ne { c } 3687
3688 \cs_generate_variant:Nn \tl_put_right:No { c } 3688
3689 \cs_generate_variant:Nn \tl_put_right:Nn { Nx , cx } 3689
3690 \cs_generate_variant:Nn \tl_gput_right:Nn { c } 3690
3691 \cs_generate_variant:Nn \tl_gput_right:NV { c } 3691
3692 \cs_generate_variant:Nn \tl_gput_right:Nv { c } 3692
3693 \cs_generate_variant:Nn \tl_gput_right:Ne { c } 3693
3694 \cs_generate_variant:Nn \tl_gput_right:No { c } 3694
3695 \cs_generate_variant:Nn \tl_gput_right:Nn { Nx, cx } 3695
3696 \quark_new:N \q__tl_nil 3696
3697 \quark_new:N \q__tl_mark 3697
3698 \quark_new:N \q__tl_stop 3698
3699 \quark_new:N \q__tl_recursion_tail 3699
3700 \quark_new:N \q__tl_recursion_stop 3700
3701 \__kernel_quark_new_test:N \__tl_if_recursion_tail_break:nN 3701
3702 \__kernel_quark_new_conditional:Nn \__tl_quark_if_nil:n { TF } 3702
3703 \tl_const:Ne \c__tl_rescan_marker_tl { : \token_to_str:N : } 3703
3704 \cs_new_protected:Npn \tl_rescan:nn #1#2 3704
3705 { 3705
3706 \tl_set_rescan:Nnn \l__tl_internal_a_tl {#1} {#2} 3706
3707 \exp_after:wN \__tl_rescan_aux: 3707
3708 \l__tl_internal_a_tl 3708
3709 } 3709
3710 \cs_generate_variant:Nn \tl_rescan:nn { nV } 3710
3711 \exp_args:NNo \cs_new_protected:Npn \__tl_rescan_aux: 3711
3712 { \tl_clear:N \l__tl_internal_a_tl } 3712
3713 \cs_new_protected:Npn \tl_set_rescan:Nnn 3713
3714 { \__tl_set_rescan:NNnn \tl_set:No } 3714
3715 \cs_new_protected:Npn \tl_gset_rescan:Nnn 3715
3716 { \__tl_set_rescan:NNnn \tl_gset:No } 3716
3717 \cs_new_protected:Npn \__tl_set_rescan:NNnn #1#2#3#4 3717
3718 { 3718
3719 \group_begin: 3719
3720 \if_false: { \fi: 3720
3721 \int_set_eq:NN \tex_tracingnesting:D \c_zero_int 3721
3722 \int_compare:nNnT \tex_endlinechar:D = { 32 } 3722
3723 { \int_set:Nn \tex_endlinechar:D { -1 } } 3723
3724 \int_set_eq:NN \tex_newlinechar:D \tex_endlinechar:D 3724
3725 #3 \scan_stop: 3725
3726 \exp_args:No \__tl_set_rescan:nNN { \tl_to_str:n {#4} } #1 #2 3726
```

```

3727 \if_false: } \fi:
3728 }
3729 \cs_new_protected:Npn \__tl_set_rescan_multi:nNN #1#2#3
3730 {
3731   \tex_everyeof:D \exp_after:wN { \c__tl_rescan_marker_tl }
3732   \exp_after:wN \__tl_rescan:NNw
3733   \exp_after:wN #2
3734   \exp_after:wN #3
3735   \exp_after:wN \prg_do_nothing:
3736   \tex_scantokens:D {#1}
3737 }
3738 \exp_args:Nno \use:nn
3739 { \cs_new:Npn \__tl_rescan:NNw #1#2#3 } \c__tl_rescan_marker_tl
3740 {
3741   \group_end:
3742   #1 #2 {#3}
3743 }
3744 \cs_generate_variant:Nn \tl_set_rescan:Nnn { NnV , Nne , c , cnV , cne }
3745 \cs_generate_variant:Nn \tl_set_rescan:Nnn { Nno , Nnx , cno , cnx }
3746 \cs_generate_variant:Nn \tl_gset_rescan:Nnn { NnV , Nne , c , cnV , cne }
3747 \cs_generate_variant:Nn \tl_gset_rescan:Nnn { Nno , Nnx , cno , cnx }
3748 \cs_new_protected:Npn \__tl_set_rescan:nNN #1
3749 {
3750   \int_compare:nNnTF \tex_newlinechar:D < 0
3751   { \use_ii:nn }
3752   {
3753     \exp_args:Nnf \tl_if_in:nnTF {#1}
3754     { \char_generate:nn { \tex_newlinechar:D } { 12 } }
3755   }
3756   { \__tl_set_rescan_multi:nNN }
3757   {
3758     \int_set:Nn \tex_endlinechar:D { -1 }
3759     \__tl_set_rescan_single:nNN { `` }
3760   }
3761   {#1}
3762 }
3763 \cs_new_protected:Npn \__tl_set_rescan_single:nNN #1
3764 {
3765   \int_compare:nNnTF
3766   { \char_value_catcode:n {#1} / 2 } = 6
3767   {
3768     \exp_args:Nof \__tl_set_rescan_single_aux:nnnNN
3769     \c__tl_rescan_marker_tl
3770     { \char_generate:nn {#1} { \char_value_catcode:n {#1} } }
3771   }
3772   {

```

```

3773 \int_compare:nNnTF {#1} < { `~ } 3773
3774 { 3774
3775 \exp_args:Nf \__tl_set_rescan_single:nnNN 3775
3776 { \int_eval:n { #1 + 1 } } 3776
3777 } 3777
3778 { \__tl_set_rescan_multi:nnN } 3778
3779 } 3779
3780 } 3780
3781 \cs_new_protected:Npn \__tl_set_rescan_single_aux:nnnNN #1#2#3#4#5 3781
3782 { 3782
3783 \tex_everyeof:D 3783
3784 { 3784
3785 #1 \use_none:n 3785
3786 #2 #1 { \exp:w \__tl_set_rescan_single_aux:w } 3786
3787 \s__tl_stop 3787
3788 } 3788
3789 \cs_set:Npn \__tl_rescan:NNw ##1##2##3 #2 #1 ##4 ##5 \s__tl_stop 3789
3790 { 3790
3791 \group_end: 3791
3792 ##1 ##2 { ##4 ##3 } 3792
3793 } 3793
3794 \exp_after:wN \__tl_rescan:NNw 3794
3795 \exp_after:wN #4 3795
3796 \exp_after:wN #5 3796
3797 \tex_scantokens:D { #2 #3 #2 } 3797
3798 } 3798
3799 \exp_args:Nno \use:nn 3799
3800 { \cs_new:Npn \__tl_set_rescan_single_aux:w #1 } 3800
3801 \c__tl_rescan_marker_tl #2 3801
3802 { \use_i:nn \exp_end: #1 } 3802
3803 \cs_new_protected:Npn \tl_replace_once:Nnn 3803
3804 { \__tl_replace:NnNNNnn \q__tl_mark ? \__tl_replace_wrap:w \__kernel_tl_set:Nx } 3804
3805 \cs_new_protected:Npn \tl_greplace_once:Nnn 3805
3806 { \__tl_replace:NnNNNnn \q__tl_mark ? \__tl_replace_wrap:w \__kernel_tl_gset:Nx } 3806
3807 \cs_new_protected:Npn \tl_replace_all:Nnn 3807
3808 { \__tl_replace:NnNNNnn \q__tl_mark ? \__tl_replace_next:w \__kernel_tl_set:Nx } 3808
3809 \cs_new_protected:Npn \tl_greplace_all:Nnn 3809
3810 { \__tl_replace:NnNNNnn \q__tl_mark ? \__tl_replace_next:w \__kernel_tl_gset:Nx } 3810
3811 \cs_generate_variant:Nn \tl_replace_once:Nnn 3811
3812 { NnV , Nne , NV , Ne , Nee , c , cnV , cne , cV , ce , cee } 3812
3813 \cs_generate_variant:Nn \tl_replace_once:Nnn 3813
3814 { Nx , Nnx , Nxx , cxn , cnx , cxx } 3814
3815 \cs_generate_variant:Nn \tl_greplace_once:Nnn 3815
3816 { NnV , Nne , NV , Ne , Nee , c , cnV , cne , cV , ce , cee } 3816
3817 \cs_generate_variant:Nn \tl_greplace_once:Nnn 3817
3818 { Nx , Nnx , Nxx , cxn , cnx , cxx } 3818

```

```
3819 \cs_generate_variant:Nn \tl_replace_all:Nnn 3819
3820 { NnV , Nne , NV , Ne , Nee , c , cnV , cne , cV , ce , cee } 3820
3821 \cs_generate_variant:Nn \tl_replace_all:Nnn 3821
3822 { Nx , Nnx , Nxx , cxn , cnx , cxx } 3822
3823 \cs_generate_variant:Nn \tl_greplace_all:Nnn 3823
3824 { NnV , Nne , NV , Ne , Nee , c , cnV , cne , cV , ce , cee } 3824
3825 \cs_generate_variant:Nn \tl_greplace_all:Nnn 3825
3826 { Nx , Nnx , Nxx , cxn , cnx , cxx } 3826
3827 \cs_new_protected:Npn \__tl_replace:NnNNNnn #1#2#3#4#5#6#7 3827
3828 { 3828
3829 \tl_if_empty:nTF {#6} 3829
3830 { 3830
3831 \msg_error:nne { kernel } { empty-search-pattern } 3831
3832 { \tl_to_str:n {#7} } 3832
3833 } 3833
3834 { 3834
3835 \tl_if_in:onTF { #5 #6 } {#1} 3835
3836 { 3836
3837 \tl_if_in:nnTF {#6} {#1} 3837
3838 { \exp_args:Nc \__tl_replace:NnNNNnn {#2} {#2?} } 3838
3839 { 3839
3840 \__tl_quark_if_nil:nTF {#6} 3840
3841 { \__tl_replace_auxi:NnnNNNnn #5 {#1} { #1 \q__tl_stop } } 3841
3842 { \__tl_replace_auxi:NnnNNNnn #5 {#1} { #1 \q__tl_nil } } 3842
3843 } 3843
3844 } 3844
3845 { \__tl_replace_auxii:nNNNnn {#1} } 3845
3846 #3#4#5 {#6} {#7} 3846
3847 } 3847
3848 } 3848
3849 \cs_new_protected:Npn \__tl_replace_auxi:NnnNNNnn #1#2#3 3849
3850 { 3850
3851 \tl_if_in:NnTF #1 { #2 #3 #3 } 3851
3852 { \__tl_replace_auxi:NnnNNNnn #1 { #2 #3 } {#2} } 3852
3853 { \__tl_replace_auxii:nNNNnn { #2 #3 #3 } } 3853
3854 } 3854
3855 \cs_new_protected:Npn \__tl_replace_auxii:nNNNnn #1#2#3#4#5#6 3855
3856 { 3856
3857 \group_align_safe_begin: 3857
3858 \cs_set:Npn \__tl_replace_wrap:w ##1 #1 ##2 3858
3859 { \__kernel_exp_not:w \exp_after:wN { \use_none:nn ##1 } ##2 } 3859
3860 \cs_set:Npe \__tl_replace_next:w ##1 #5 3860
3861 { 3861
3862 \exp_not:N \__tl_replace_wrap:w ##1 3862
3863 \exp_not:n { #1 } 3863
3864 \exp_not:n { \exp_not:n {#6} } 3864
```



```
3865 \exp_not:n { #2 { } { } } 3865
3866 } 3866
3867 #3 #4 3867
3868 { 3868
3869 \exp_after:wN \__tl_replace_next_aux:w 3869
3870 #4 3870
3871 #1 3871
3872 { 3872
3873 \if_false: { \fi: } 3873
3874 \exp_after:wN \use_none:n \exp_after:wN { \if_false: } \fi: 3874
3875 } 3875
3876 #5 3876
3877 } 3877
3878 \group_align_safe_end: 3878
3879 } 3879
3880 \cs_new:Npn \__tl_replace_next_aux:w { \__tl_replace_next:w { } { } } 3880
3881 \cs_new_eq:NN \__tl_replace_wrap:w ? 3881
3882 \cs_new_eq:NN \__tl_replace_next:w ? 3882
3883 \cs_new_protected:Npn \tl_regex_replace_once:Nnn #1#2#3 3883
3884 { \regex_replace_once:nnN {#2} {#3} #1 } 3884
3885 \cs_generate_variant:Nn \tl_regex_replace_once:Nnn { c } 3885
3886 \cs_new_protected:Npn \tl_regex_replace_once:NNn #1#2#3 3886
3887 { \regex_replace_once:NnN #2 {#3} #1 } 3887
3888 \cs_generate_variant:Nn \tl_regex_replace_once:NNn { c } 3888
3889 \cs_new_protected:Npn \tl_regex_replace_all:Nnn #1#2#3 3889
3890 { \regex_replace_all:nnN {#2} {#3} #1 } 3890
3891 \cs_generate_variant:Nn \tl_regex_replace_all:Nnn { c } 3891
3892 \cs_new_protected:Npn \tl_regex_replace_all:NNn #1#2#3 3892
3893 { \regex_replace_all:NnN #2 {#3} #1 } 3893
3894 \cs_generate_variant:Nn \tl_regex_replace_all:NNn { c } 3894
3895 \group_begin: 3895
3896 \cs_set_protected:Npn \__tl_tmp:w #1#2#3 3896
3897 { 3897
3898 \cs_new_protected:cpe { tl_regex_greplace_ #1 :N #2 n } ##1##2##3 3898
3899 { 3899
3900 \group_begin: 3900
3901 \tl_set_eq:NN \exp_not:N \l__tl_internal_a_tl ##1 3901
3902 \exp_not:c { regex_replace_ #1 :Nn #2 } 3902
3903 #3 {##2} {##3} \exp_not:N \l__tl_internal_a_tl 3903
3904 \tl_gset_eq:NN ##1 \exp_not:N \l__tl_internal_a_tl 3904
3905 \group_end: 3905
3906 } 3906
3907 \cs_generate_variant:cn { tl_regex_greplace_ #1 :N #2 n } { c } 3907
3908 } 3908
3909 \__tl_tmp:w { once } n { } 3909
3910 \__tl_tmp:w { once } N \use:n 3910
```

```
3911 \tl_tmp:w { all } n { }
3912 \tl_tmp:w { all } N \use:n
3913 \group_end:
3914 \cs_new_protected:Npn \tl_remove_once:Nn #1#2
3915 { \tl_replace_once:Nnn #1 {#2} { } }
3916 \cs_new_protected:Npn \tl_gremove_once:Nn #1#2
3917 { \tl_greplace_once:Nnn #1 {#2} { } }
3918 \cs_generate_variant:Nn \tl_remove_once:Nn { NV , Ne , c , cV , ce }
3919 \cs_generate_variant:Nn \tl_gremove_once:Nn { NV , Ne , c , cV , ce }
3920 \cs_new_protected:Npn \tl_remove_all:Nn #1#2
3921 { \tl_replace_all:Nnn #1 {#2} { } }
3922 \cs_new_protected:Npn \tl_gremove_all:Nn #1#2
3923 { \tl_greplace_all:Nnn #1 {#2} { } }
3924 \cs_generate_variant:Nn \tl_remove_all:Nn { NV , Ne , c , cV , ce }
3925 \cs_generate_variant:Nn \tl_remove_all:Nn { Nx , cx }
3926 \cs_generate_variant:Nn \tl_gremove_all:Nn { NV , Ne , c , cV , ce }
3927 \cs_generate_variant:Nn \tl_gremove_all:Nn { Nx , cx }
3928 \prg_new_conditional:Npnn \tl_if_empty:N #1 { p , T , F , TF }
3929 {
3930 \if_meaning:w #1 \c_empty_tl
3931 \prg_return_true:
3932 \else:
3933 \prg_return_false:
3934 \fi:
3935 }
3936 \prg_generate_conditional_variant:Nnn \tl_if_empty:N
3937 { c } { p , T , F , TF }
3938 \prg_new_conditional:Npnn \tl_if_empty:n #1 { p , TF , T , F }
3939 {
3940 \if:w \scan_stop: \tl_to_str:n {#1} \scan_stop:
3941 \prg_return_true:
3942 \else:
3943 \prg_return_false:
3944 \fi:
3945 }
3946 \prg_generate_conditional_variant:Nnn \tl_if_empty:n
3947 { V , e } { p , TF , T , F }
3948 \cs_new:Npn \__tl_if_empty_if:o #1
3949 {
3950 \if:w \scan_stop: \__kernel_tl_to_str:w \exp_after:wN {#1} \scan_stop:
3951 }
3952 \exp_args:Nno \use:n
3953 { \prg_new_conditional:Npnn \tl_if_empty:o #1 { p , TF , T , F } }
3954 {
3955 \__tl_if_empty_if:o {#1}
3956 \prg_return_true:
```

```

3957 \else:
3958 \prg_return_false:
3959 \fi:
3960 }
3961 \exp_args:Nno \use:n
3962 { \prg_new_conditional:Npnn \tl_if_blank:n #1 { p , T , F , TF } }
3963 {
3964 \__tl_if_empty_if:o { \use_none:n #1 ? }
3965 \prg_return_true:
3966 \else:
3967 \prg_return_false:
3968 \fi:
3969 }
3970 \prg_generate_conditional_variant:Nnn \tl_if_blank:n
3971 { e , V , o } { p , T , F , TF }
3972 \prg_new_eq_conditional:NNn \tl_if_eq:NN \cs_if_eq:NN { p , T , F , TF }
3973 \prg_generate_conditional_variant:Nnn \tl_if_eq:NN
3974 { Nc , c , cc } { p , TF , T , F }
3975 \tl_new:N \l__tl_internal_a_tl
3976 \tl_new:N \l__tl_internal_b_tl
3977 \prg_new_protected_conditional:Npnn \tl_if_eq:Nn #1#2 { T , F , TF }
3978 {
3979 \group_begin:
3980 \tl_set:Nn \l__tl_internal_b_tl {#2}
3981 \exp_after:wN
3982 \group_end:
3983 \if_meaning:w #1 \l__tl_internal_b_tl
3984 \prg_return_true:
3985 \else:
3986 \prg_return_false:
3987 \fi:
3988 }
3989 \prg_generate_conditional_variant:Nnn \tl_if_eq:Nn { c } { TF , T , F }
3990 \prg_new_protected_conditional:Npnn \tl_if_eq:nn #1#2 { T , F , TF }
3991 {
3992 \group_begin:
3993 \tl_set:Nn \l__tl_internal_a_tl {#1}
3994 \tl_set:Nn \l__tl_internal_b_tl {#2}
3995 \exp_after:wN
3996 \group_end:
3997 \if_meaning:w \l__tl_internal_a_tl \l__tl_internal_b_tl
3998 \prg_return_true:
3999 \else:
4000 \prg_return_false:
4001 \fi:
4002 }

```

```

4003 \prg_generate_conditional_variant:Nnn \tl_if_eq:nn
4004 { nV , ne , nx , V, e , ee , x , xx }
4005 { TF , T , F }
4006 \cs_new_protected:Npn \tl_if_in:NnT { \exp_args:No \tl_if_in:nnT }
4007 \cs_new_protected:Npn \tl_if_in:NnF { \exp_args:No \tl_if_in:nnF }
4008 \cs_new_protected:Npn \tl_if_in:NnTF { \exp_args:No \tl_if_in:nnTF }
4009 \prg_generate_conditional_variant:Nnn \tl_if_in:Nn
4010 { NV , No , c , cV , co } { T , F , TF }
4011 \prg_new_protected_conditional:Npnn \tl_if_in:nn #1#2 { T , F , TF }
4012 {
4013   \scan_stop:
4014   \if_false: { \fi:
4015     \cs_set:Npn \__tl_tmp:w ##1 #2 { }
4016     \tl_if_empty:oTF { \__tl_tmp:w #1 {} {} #2 }
4017     { \prg_return_false: } { \prg_return_true: }
4018     \if_false: } \fi:
4019   }
4020 \prg_generate_conditional_variant:Nnn \tl_if_in:nn
4021 { V , VV , o , oo , nV , no } { T , F , TF }
4022 \cs_set_protected:Npn \__tl_tmp:w #1
4023 {
4024   \prg_new_conditional:Npnn \tl_if_novalue:n ##1
4025   { p , T , F , TF }
4026   {
4027     \__tl_if_empty_if:o { \__tl_if_novalue:w {} ##1 {} ? ! #1 ? ? ! }
4028     \prg_return_true:
4029     \else:
4030     \prg_return_false:
4031     \fi:
4032   }
4033   \cs_new:Npn \__tl_if_novalue:w ##1 #1 ##2 ? ##3 ? ! { ##1 ##2 }
4034 }
4035 \exp_args:No \__tl_tmp:w { \c_novalue_tl }
4036 \cs_new:Npn \tl_if_single_p:N { \exp_args:No \tl_if_single_p:n }
4037 \cs_new:Npn \tl_if_single:NT { \exp_args:No \tl_if_single:nT }
4038 \cs_new:Npn \tl_if_single:NF { \exp_args:No \tl_if_single:nF }
4039 \cs_new:Npn \tl_if_single:NTF { \exp_args:No \tl_if_single:nTF }
4040 \prg_generate_conditional_variant:Nnn \tl_if_single:N {c} { p , T , F , TF }
4041 \prg_new_conditional:Npnn \tl_if_single:n #1 { p , T , F , TF }
4042 {
4043   \if:w \scan_stop: \exp_after:wN \__tl_if_single:nnw
4044     \__kernel_tl_to_str:w
4045     \exp_after:wN { \use_none:nn #1 ?? } \scan_stop: ? \s__tl_stop
4046     \prg_return_true:
4047   \else:
4048     \prg_return_false:

```

```
4049 \fi:
4050 }
4051 \cs_new:Npn \__tl_if_single:nnw #1#2#3 \s_tl_stop {#2}
4052 \prg_new_conditional:Npnn \tl_if_single_token:n #1 { p , T , F , TF }
4053 {
4054   \tl_if_head_is_N_type:nTF {#1}
4055   { \__tl_if_empty_if:o { \use_none:n #1 } }
4056   {
4057     \tl_if_empty:nTF {#1}
4058     { \if_false: }
4059     { \__tl_if_empty_if:o { \exp:w \exp_end_continue_f:w #1 } }
4060   }
4061   \prg_return_true:
4062 \else:
4063   \prg_return_false:
4064 \fi:
4065 }
4066 \prg_new_protected_conditional:Npnn \tl_if_regex_match:nn #1#2 { TF , T , F }
4067 {
4068   \regex_match:nnTF {#2} {#1}
4069   \prg_return_true: \prg_return_false:
4070 }
4071 \prg_generate_conditional_variant:Nnn \tl_if_regex_match:nn
4072 { V } { TF , T , F }
4073 \prg_new_protected_conditional:Npnn \tl_if_regex_match:nN #1#2 { TF , T , F }
4074 {
4075   \regex_match:nNTF {#2} #1
4076   \prg_return_true: \prg_return_false:
4077 }
4078 \prg_generate_conditional_variant:Nnn \tl_if_regex_match:nN
4079 { V } { TF , T , F }
4080 \cs_new:Npn \tl_map_function:nN #1#2
4081 {
4082   \__tl_map_function:Nnnnnnnnn #2 #1
4083   \s_tl_stop \s_tl_stop \s_tl_stop \s_tl_stop
4084   \s_tl_stop \s_tl_stop \s_tl_stop \s_tl_stop
4085   \prg_break_point:Nn \tl_map_break: { }
4086 }
4087 \cs_generate_variant:Nn \tl_map_function:nN { e }
4088 \cs_new:Npn \tl_map_function:NN
4089 { \exp_args:No \tl_map_function:nN }
4090 \cs_generate_variant:Nn \tl_map_function:NN { c }
4091 \cs_new:Npn \__tl_map_function:Nnnnnnnnn #1#2#3#4#5#6#7#8#9
4092 {
4093   \__tl_use_none_delimit_by_s_stop:w
4094   #9 \__tl_map_function_end:w \s_tl_stop
```

```
4095 #1 {#2} #1 {#3} #1 {#4} #1 {#5} #1 {#6} #1 {#7} #1 {#8} #1 {#9} 4095
4096 \__tl_map_function:Nnnnnnnnn #1 4096
4097 } 4097
4098 \cs_new:Npn \__tl_map_function_end:w \s__tl_stop #1#2 4098
4099 { 4099
4100 \__tl_use_none_delimit_by_s_stop:w #2 \tl_map_break: \s__tl_stop 4100
4101 #1 {#2} 4101
4102 \__tl_map_function_end:w \s__tl_stop 4102
4103 } 4103
4104 \cs_new:Npn \__tl_use_none_delimit_by_s_stop:w #1 \s__tl_stop { } 4104
4105 \cs_new_protected:Npn \tl_map_inline:nn #1#2 4105
4106 { 4106
4107 \int_gincr:N \g__kernel_prg_map_int 4107
4108 \cs_gset_protected:cpn 4108
4109 { __tl_map_ \int_use:N \g__kernel_prg_map_int :w } ##1 {#2} 4109
4110 \exp_args:Nc \__tl_map_function:Nnnnnnnnn 4110
4111 { __tl_map_ \int_use:N \g__kernel_prg_map_int :w } 4111
4112 #1 4112
4113 \s__tl_stop \s__tl_stop \s__tl_stop \s__tl_stop 4113
4114 \s__tl_stop \s__tl_stop \s__tl_stop \s__tl_stop 4114
4115 \prg_break_point:Nn \tl_map_break: 4115
4116 { \int_gdecr:N \g__kernel_prg_map_int } 4116
4117 } 4117
4118 \cs_new_protected:Npn \tl_map_inline:Nn 4118
4119 { \exp_args:No \tl_map_inline:nn } 4119
4120 \cs_generate_variant:Nn \tl_map_inline:Nn { c } 4120
4121 \cs_new:Npn \tl_map_tokens:nn #1#2 4121
4122 { 4122
4123 \__tl_map_tokens:nnnnnnnn {#2} #1 4123
4124 \s__tl_stop \s__tl_stop \s__tl_stop \s__tl_stop 4124
4125 \s__tl_stop \s__tl_stop \s__tl_stop \s__tl_stop 4125
4126 \prg_break_point:Nn \tl_map_break: { } 4126
4127 } 4127
4128 \cs_new:Npn \tl_map_tokens:Nn 4128
4129 { \exp_args:No \tl_map_tokens:nn } 4129
4130 \cs_generate_variant:Nn \tl_map_tokens:Nn { c } 4130
4131 \cs_new:Npn \__tl_map_tokens:nnnnnnnn #1#2#3#4#5#6#7#8#9 4131
4132 { 4132
4133 \__tl_use_none_delimit_by_s_stop:w 4133
4134 #9 \__tl_map_tokens_end:w \s__tl_stop 4134
4135 \use:n {#1} {#2} \use:n {#1} {#3} \use:n {#1} {#4} \use:n {#1} {#5} 4135
4136 \use:n {#1} {#6} \use:n {#1} {#7} \use:n {#1} {#8} \use:n {#1} {#9} 4136
4137 \__tl_map_tokens:nnnnnnnn {#1} 4137
4138 } 4138
4139 \cs_new:Npn \__tl_map_tokens_end:w \s__tl_stop \use:n #1#2 4139
4140 { 4140
```


4141	_tl_use_none_delimit_by_s_stop:w #2 \tl_map_break: \s_tl_stop	4141
4142	#1 {#2}	4142
4143	_tl_map_tokens_end:w \s_tl_stop	4143
4144	}	4144
4145	\cs_new_protected:Npn \tl_map_variable:nNn #1#2#3	4145
4146	{ \tl_map_tokens:nn {#1} { _tl_map_variable:Nnn #2 {#3} } }	4146
4147	\cs_new_protected:Npn _tl_map_variable:Nnn #1#2#3	4147
4148	{ \tl_set:Nn #1 {#3} #2 }	4148
4149	\cs_new_protected:Npn \tl_map_variable:NNn	4149
4150	{ \exp_args:No \tl_map_variable:nNn }	4150
4151	\cs_generate_variant:Nn \tl_map_variable:NNn { c }	4151
4152	\cs_new:Npn \tl_map_break:	4152
4153	{ \prg_map_break:Nn \tl_map_break: { } }	4153
4154	\cs_new:Npn \tl_map_break:n	4154
4155	{ \prg_map_break:Nn \tl_map_break: }	4155
4156	\cs_generate_variant:Nn \tl_to_str:n { o , V , v , e }	4156
4157	\cs_new:Npn \tl_to_str:N #1 { _kernel_tl_to_str:w \exp_after:wN {#1} }	4157
4158	\cs_generate_variant:Nn \tl_to_str:N { c }	4158
4159	\cs_new:Npn \tl_use:N #1	4159
4160	{	4160
4161	\tl_if_exist:NTF #1 {#1}	4161
4162	{	4162
4163	\msg_expandable_error:nnn	4163
4164	{ kernel } { bad-variable } {#1}	4164
4165	}	4165
4166	}	4166
4167	\cs_generate_variant:Nn \tl_use:N { c }	4167
4168	\cs_new:Npn \tl_count:n #1	4168
4169	{	4169
4170	\int_eval:n	4170
4171	{ 0 \tl_map_function:nN {#1} _tl_count:n }	4171
4172	}	4172
4173	\cs_new:Npn \tl_count:N #1	4173
4174	{	4174
4175	\int_eval:n	4175
4176	{ 0 \tl_map_function:NN #1 _tl_count:n }	4176
4177	}	4177
4178	\cs_new:Npn _tl_count:n #1 { + 1 }	4178
4179	\cs_generate_variant:Nn \tl_count:n { V , v , e , o }	4179
4180	\cs_generate_variant:Nn \tl_count:N { c }	4180
4181	\cs_new:Npn \tl_count_tokens:n #1	4181
4182	{	4182
4183	\int_eval:n	4183
4184	{	4184
4185	_tl_act:NNNn	4185
4186	_tl_act_count_normal:N	4186

```
4187         \__tl_act_count_group:n 4187
4188         \__tl_act_count_space: 4188
4189         {#1} 4189
4190     } 4190
4191 } 4191
4192 \cs_new:Npn \__tl_act_count_normal:N #1 { 1 + } 4192
4193 \cs_new:Npn \__tl_act_count_space: { 1 + } 4193
4194 \cs_new:Npn \__tl_act_count_group:n #1 { 2 + \tl_count_tokens:n {#1} + } 4194
4195 \cs_new:Npn \tl_reverse_items:n #1 4195
4196 { 4196
4197     \__tl_reverse_items:nwNwn #1 ? 4197
4198     \s__tl_mark \__tl_reverse_items:nwNwn 4198
4199     \s__tl_mark \__tl_reverse_items:wn 4199
4200     \s__tl_stop { } 4200
4201 } 4201
4202 \cs_new:Npn \__tl_reverse_items:nwNwn #1 #2 \s__tl_mark #3 #4 \s__tl_stop #5 4202
4203 { 4203
4204     #3 #2 4204
4205     \s__tl_mark \__tl_reverse_items:nwNwn 4205
4206     \s__tl_mark \__tl_reverse_items:wn 4206
4207     \s__tl_stop { {#1} #5 } 4207
4208 } 4208
4209 \cs_new:Npn \__tl_reverse_items:wn #1 \s__tl_stop #2 4209
4210 { \__kernel_exp_not:w \exp_after:wN { \use_none:nn #2 } } 4210
4211 \cs_new:Npn \tl_trim_spaces:n 4211
4212 { \__tl_trim_spaces:nn { \__kernel_exp_not:w \exp_after:wN } } 4212
4213 \cs_new:Npn \tl_trim_left_spaces:n 4213
4214 { \__tl_trim_left_spaces:nn { \__kernel_exp_not:w \exp_after:wN } } 4214
4215 \cs_new:Npn \tl_trim_right_spaces:n 4215
4216 { \__tl_trim_right_spaces:nn { \__kernel_exp_not:w \exp_after:wN } } 4216
4217 \cs_generate_variant:Nn \tl_trim_spaces:n { V , v , e , o } 4217
4218 \cs_generate_variant:Nn \tl_trim_left_spaces:n { V , v , e , o } 4218
4219 \cs_generate_variant:Nn \tl_trim_right_spaces:n { V , v , e , o } 4219
4220 \cs_new:Npn \tl_trim_spaces_apply:nN #1#2 4220
4221 { \__tl_trim_spaces:nn { \exp_args:No #2 } { #1 } } 4221
4222 \cs_new:Npn \tl_trim_left_spaces_apply:nN #1#2 4222
4223 { \__tl_trim_left_spaces:nn { \exp_args:No #2 } { #1 } } 4223
4224 \cs_new:Npn \tl_trim_right_spaces_apply:nN #1#2 4224
4225 { \__tl_trim_right_spaces:nn { \exp_args:No #2 } { #1 } } 4225
4226 \cs_generate_variant:Nn \tl_trim_spaces_apply:nN { o } 4226
4227 \cs_generate_variant:Nn \tl_trim_left_spaces_apply:nN { o } 4227
4228 \cs_generate_variant:Nn \tl_trim_right_spaces_apply:nN { o } 4228
4229 \cs_new_protected:Npn \tl_trim_spaces:N #1 4229
4230 { \__kernel_tl_set:Nx #1 { \exp_args:No \tl_trim_spaces:n {#1} } } 4230
4231 \cs_new_protected:Npn \tl_trim_left_spaces:N #1 4231
4232 { \__kernel_tl_set:Nx #1 { \exp_args:No \tl_trim_left_spaces:n {#1} } } 4232
```

```
4233 \cs_new_protected:Npn \tl_trim_right_spaces:N #1 4233
4234 { \__kernel_tl_set:Nx #1 { \exp_args:No \tl_trim_right_spaces:n {#1} } } 4234
4235 \cs_new_protected:Npn \tl_gtrim_spaces:N #1 4235
4236 { \__kernel_tl_gset:Nx #1 { \exp_args:No \tl_trim_spaces:n {#1} } } 4236
4237 \cs_new_protected:Npn \tl_gtrim_left_spaces:N #1 4237
4238 { \__kernel_tl_gset:Nx #1 { \exp_args:No \tl_trim_left_spaces:n {#1} } } 4238
4239 \cs_new_protected:Npn \tl_gtrim_right_spaces:N #1 4239
4240 { \__kernel_tl_gset:Nx #1 { \exp_args:No \tl_trim_right_spaces:n {#1} } } 4240
4241 \cs_generate_variant:Nn \tl_trim_spaces:N { c } 4241
4242 \cs_generate_variant:Nn \tl_trim_left_spaces:N { c } 4242
4243 \cs_generate_variant:Nn \tl_trim_right_spaces:N { c } 4243
4244 \cs_generate_variant:Nn \tl_gtrim_spaces:N { c } 4244
4245 \cs_generate_variant:Nn \tl_gtrim_left_spaces:N { c } 4245
4246 \cs_generate_variant:Nn \tl_gtrim_right_spaces:N { c } 4246
4247 \cs_set_protected:Npn \__tl_tmp:w #1 4247
4248 { 4248
4249 \cs_new:Npn \__tl_trim_spaces:nn ##1##2 4249
4250 { 4250
4251 \__tl_trim_spaces_auxi:w 4251
4252 \__tl_trim_mark: ##2 \s__tl_nil 4252
4253 \__tl_trim_mark: \__tl_trim_spaces_auxi:w 4253
4254 \__tl_trim_mark: #1 4254
4255 \__tl_trim_mark: \__tl_trim_spaces_auxii:w 4255
4256 {##1} 4256
4257 } 4257
4258 \cs_new:Npn \__tl_trim_left_spaces:nn ##1##2 4258
4259 { 4259
4260 \__tl_trim_spaces_auxi:w 4260
4261 \__tl_trim_mark: ##2 \s__tl_nil 4261
4262 \__tl_trim_mark: \__tl_trim_spaces_auxi:w 4262
4263 \__tl_trim_mark: #1 4263
4264 \__tl_trim_mark: \__tl_trim_spaces_auxv:w 4264
4265 {##1} 4265
4266 } 4266
4267 \cs_new:Npn \__tl_trim_right_spaces:nn ##1##2 4267
4268 { 4268
4269 \__tl_trim_spaces_auxiii:w 4269
4270 \__tl_trim_mark: ##2 \s__tl_nil \__tl_trim_spaces_auxiii:w 4270
4271 #1 \s__tl_nil \__tl_trim_spaces_auxiv:w 4271
4272 {##1} 4272
4273 } 4273
4274 \cs_new:Npn \__tl_trim_spaces_auxi:w 4274
4275 ##1 \__tl_trim_mark: #1 ##2 \__tl_trim_mark: ##3 4275
4276 { ##3 ##1 \__tl_trim_mark: ##2 \__tl_trim_mark: \__tl_trim_spaces_auxi:w } 4276
4277 \cs_new:Npn \__tl_trim_spaces_auxii:w 4277
4278 \__tl_trim_mark: ##1 \__tl_trim_mark: ##2 \__tl_trim_spaces_auxi:w 4278
```

```

4279 \__tl_trim_mark: \__tl_trim_mark: \__tl_trim_spaces_auxi:w 4279
4280 { 4280
4281 \__tl_trim_spaces_auxiii:w 4281
4282 \__tl_trim_mark: ##1 \__tl_trim_spaces_auxiii:w 4282
4283 #1 \s__tl_nil \__tl_trim_spaces_auxiv:w 4283
4284 } 4284
4285 \cs_new:Npn \__tl_trim_spaces_auxiii:w ##1 #1 \s__tl_nil ##2 4285
4286 { ##2 ##1 \s__tl_nil \__tl_trim_spaces_auxiii:w } 4286
4287 \cs_new:Npn \__tl_trim_spaces_auxiv:w 4287
4288 ##1 \s__tl_nil 4288
4289 \__tl_trim_spaces_auxiii:w \s__tl_nil \__tl_trim_spaces_auxiii:w 4289
4290 ##2 4290
4291 { ##2 {##1} } 4291
4292 \cs_new:Npn \__tl_trim_spaces_auxv:w 4292
4293 ##1 \s__tl_nil 4293
4294 \__tl_trim_mark: \__tl_trim_spaces_auxi:w \__tl_trim_mark: 4294
4295 \__tl_trim_mark: \__tl_trim_spaces_auxi:w ##2 4295
4296 { ##2 {##1} } 4296
4297 \cs_new:Npn \__tl_trim_mark: {} 4297
4298 } 4298
4299 \__tl_tmp:w { ~ } 4299
4300 \cs_new:Npn \tl_head:n #1 4300
4301 { 4301
4302 \__kernel_exp_not:w \tex_expanded:D 4302
4303 { { \if_false: { \fi: \__tl_head_aux:n #1 { } } } } 4303
4304 } 4304
4305 \cs_new:Npn \__tl_head_aux:n #1 4305
4306 { 4306
4307 \__kernel_exp_not:w {#1} 4307
4308 \exp_after:wN \use_none:n \exp_after:wN { \if_false: } \fi: 4308
4309 } 4309
4310 \cs_generate_variant:Nn \tl_head:n { V , v , f , e } 4310
4311 \cs_new:Npn \tl_head:w #1#2 \q_stop {#1} 4311
4312 \cs_new:Npn \__tl_tl_head:w #1#2 \s__tl_stop {#1} 4312
4313 \cs_new:Npn \tl_head:N { \exp_args:No \tl_head:n } 4313
4314 \exp_args:Nno \use:n { \cs_new:Npn \tl_tail:n #1 } 4314
4315 { 4315
4316 \exp_after:wN \__kernel_exp_not:w 4316
4317 \tl_if_blank:nTF {#1} 4317
4318 { { } } 4318
4319 { \exp_after:wN { \use_none:n #1 } } 4319
4320 } 4320
4321 \cs_generate_variant:Nn \tl_tail:n { V , v , f , e } 4321
4322 \cs_new:Npn \tl_tail:N { \exp_args:No \tl_tail:n } 4322
4323 \prg_new_conditional:Npnn \tl_if_head_eq_charcode:nN #1#2 { p , T , F , TF } 4323
4324 { 4324

```

4325	\if_charcode:w	4325
4326	\tl_if_head_is_N_type:nTF { #1 ? }	4326
4327	{ __tl_head_exp_not:w #1 { ^ __tl_if_head_eq_empty_arg:w } \s__tl_stop }	4327
4328	{ \str_head:n {#1} }	4328
4329	\exp_not:N #2	4329
4330	\prg_return_true:	4330
4331	\else:	4331
4332	\prg_return_false:	4332
4333	\fi:	4333
4334	}	4334
4335	\prg_generate_conditional_variant:Nnn \tl_if_head_eq_charcode:nN	4335
4336	{ V , e , f } { p , TF , T , F }	4336
4337	\prg_new_conditional:Npnn \tl_if_head_eq_catcode:nN #1 #2 { p , T , F , TF }	4337
4338	{	4338
4339	\if_catcode:w	4339
4340	\tl_if_head_is_N_type:nTF { #1 ? }	4340
4341	{ __tl_head_exp_not:w #1 { ^ __tl_if_head_eq_empty_arg:w } \s__tl_stop }	4341
4342	{	4342
4343	\tl_if_head_is_group:nTF {#1}	4343
4344	\c_group_begin_token	4344
4345	\c_space_token	4345
4346	}	4346
4347	\exp_not:N #2	4347
4348	\prg_return_true:	4348
4349	\else:	4349
4350	\prg_return_false:	4350
4351	\fi:	4351
4352	}	4352
4353	\prg_generate_conditional_variant:Nnn \tl_if_head_eq_catcode:nN	4353
4354	{ V , e , o } { p , TF , T , F }	4354
4355	\prg_new_conditional:Npnn \tl_if_head_eq_meaning:nN #1#2 { p , T , F , TF }	4355
4356	{	4356
4357	\tl_if_head_is_N_type:nTF { #1 ? }	4357
4358	__tl_if_head_eq_meaning_normal:nN	4358
4359	__tl_if_head_eq_meaning_special:nN	4359
4360	{#1} #2	4360
4361	}	4361
4362	\prg_generate_conditional_variant:Nnn \tl_if_head_eq_meaning:nN	4362
4363	{ V , e } { p , TF , T , F }	4363
4364	\cs_new:Npn __tl_if_head_eq_meaning_normal:nN #1 #2	4364
4365	{	4365
4366	\exp_after:wN \if_meaning:w	4366
4367	__tl_tl_head:w #1 { ?? \use_none:nnn } \s__tl_stop #2	4367
4368	\prg_return_true:	4368
4369	\else:	4369
4370	\prg_return_false:	4370

```

4371 \fi:
4372 }
4373 \cs_new:Npn \__tl_if_head_eq_meaning_special:nN #1 #2
4374 {
4375   \if_charcode:w \str_head:n {#1} \exp_not:N #2
4376     \exp_after:wN \use_ii:nn
4377   \else:
4378     \prg_return_false:
4379   \fi:
4380   \use_none:n
4381   {
4382     \if_catcode:w \exp_not:N #2
4383       \tl_if_head_is_group:nTF {#1}
4384         { \c_group_begin_token }
4385         { \c_space_token }
4386     \prg_return_true:
4387   \else:
4388     \prg_return_false:
4389   \fi:
4390 }
4391 }
4392 \cs_new:Npn \__tl_head_exp_not:w #1 #2 \s__tl_stop
4393 { \exp_not:N #1 }
4394 \cs_new:Npn \__tl_if_head_eq_empty_arg:w \exp_not:N #1
4395 { ? }
4396 \prg_new_conditional:Npnn \tl_if_head_is_N_type:n #1 { p , T , F , TF }
4397 {
4398   \if:w
4399     \if_false: { \fi: \__tl_if_head_is_N_type_auxi:w #1 ~ }
4400     { \exp_after:wN { \token_to_str:N #1 } }
4401     \scan_stop: \scan_stop:
4402     \prg_return_true:
4403   \else:
4404     \prg_return_false:
4405   \fi:
4406 }
4407 \exp_args:Nno \use:n { \cs_new:Npn \__tl_if_head_is_N_type_auxi:w #1 ~ }
4408 {
4409   \tl_if_empty:nTF {#1}
4410     { f \exp_after:wN \use_none:nn }
4411     { \exp_after:wN \__tl_if_head_is_N_type_auxii:n }
4412   \exp_after:wN { \if_false: } \fi:
4413 }
4414 \cs_new:Npn \__tl_if_head_is_N_type_auxii:n #1
4415 { \exp_after:wN \use_none:n \exp_after:wN }
4416 \prg_new_conditional:Npnn \tl_if_head_is_group:n #1 { p , T , F , TF }

```



```

4417 {
4418     \if:w
4419         \exp_after:wN \use_none:n
4420         \exp_after:wN { \exp_after:wN { \token_to_str:N #1 ? } }
4421         \scan_stop: \scan_stop:
4422         \__tl_if_head_is_group_fi_false:w
4423     \fi:
4424     \if_true:
4425         \prg_return_true:
4426     \else:
4427         \prg_return_false:
4428     \fi:
4429 }
4430 \cs_new:Npn \__tl_if_head_is_group_fi_false:w \fi: \if_true: { \fi: \if_false: }
4431 \prg_new_conditional:Npnn \tl_if_head_is_space:n #1 { p , T , F , TF }
4432 {
4433     \if:w
4434         \if_false: { \fi: \__tl_if_head_is_space:w \prg_do_nothing: #1 ? ~ }
4435         \scan_stop: \scan_stop:
4436         \prg_return_true:
4437     \else:
4438         \prg_return_false:
4439     \fi:
4440 }
4441 \exp_args:Nno \use:n { \cs_new:Npn \__tl_if_head_is_space:w #1 ~ }
4442 {
4443     \__tl_if_empty_if:o {#1} \else: f \fi:
4444     \exp_after:wN \use_none:n \exp_after:wN { \if_false: } \fi:
4445 }
4446 \scan_new:N \s__tl_act_stop
4447 \cs_set_protected:Npn \__tl_tmp:w #1
4448 {
4449     \cs_new:Npn \__tl_act_if_head_is_space:nTF ##1
4450     {
4451         \__tl_act_if_head_is_space:w
4452         \s__tl_act_stop ##1 \s__tl_act_stop \__tl_act_if_head_is_space_true:w
4453         \s__tl_act_stop #1 \s__tl_act_stop \use_ii:nn
4454     }
4455     \cs_new:Npn \__tl_act_if_head_is_space:w
4456         ##1 \s__tl_act_stop #1 ##2 \s__tl_act_stop
4457     {}
4458     \cs_new:Npn \__tl_act_if_head_is_space_true:w
4459         \s__tl_act_stop #1 \s__tl_act_stop \use_ii:nn ##1 ##2
4460         {##1}
4461 }
4462 \__tl_tmp:w { ~ }

```

```
4463 \exp_args:Nne \use:n { \cs_new:Npn \__tl_act_loop:w #1 \s__tl_act_stop } 4463
4464 { 4464
4465 \exp_not:o { \__tl_act_if_head_is_space:nTF {#1} } 4465
4466 \exp_not:N \__tl_act_space:wwNNN 4466
4467 { 4467
4468 \exp_not:o { \tl_if_head_is_group:nTF {#1} } 4468
4469 \exp_not:N \__tl_act_group:nwNNN 4469
4470 \exp_not:N \__tl_act_normal:NwNNN 4470
4471 } 4471
4472 \exp_not:n {#1} \s__tl_act_stop 4472
4473 } 4473
4474 \cs_undefine:N \__tl_act_if_head_is_space:nTF 4474
4475 \cs_new:Npn \__tl_act_normal:NwNNN #1 #2 \s__tl_act_stop #3 4475
4476 { 4476
4477 #3 #1 4477
4478 \__tl_act_loop:w #2 \s__tl_act_stop 4478
4479 #3 4479
4480 } 4480
4481 \cs_new:Npn \__tl_use_none_delimit_by_s_act_stop:w #1 \s__tl_act_stop { } 4481
4482 \cs_new:Npn \__tl_act_end:wn #1 \__tl_act_result:n #2 4482
4483 { \group_align_safe_end: \exp_end: #2 } 4483
4484 \cs_new:Npn \__tl_act_group:nwNNN #1 #2 \s__tl_act_stop #3#4#5 4484
4485 { 4485
4486 \__tl_use_none_delimit_by_s_act_stop:w #1 \__tl_act_end:wn \s__tl_act_stop 4486
4487 #5 {#1} 4487
4488 \__tl_act_loop:w #2 \s__tl_act_stop 4488
4489 #3 #4 #5 4489
4490 } 4490
4491 \exp_last_unbraced:NNo 4491
4492 \cs_new:Npn \__tl_act_space:wwNNN \c_space_tl #1 \s__tl_act_stop #2#3 4492
4493 { 4493
4494 #3 4494
4495 \__tl_act_loop:w #1 \s__tl_act_stop 4495
4496 #2 #3 4496
4497 } 4497
4498 \cs_new:Npn \__tl_act:NNNn #1#2#3#4 4498
4499 { 4499
4500 \group_align_safe_begin: 4500
4501 \__tl_act_loop:w #4 { \s__tl_act_stop } ? \s__tl_act_stop 4501
4502 #1 #3 #2 4502
4503 \__tl_act_result:n { } 4503
4504 } 4504
4505 \cs_new:Npn \__tl_act_output:n #1 #2 \__tl_act_result:n #3 4505
4506 { #2 \__tl_act_result:n { #3 #1 } } 4506
4507 \cs_new:Npn \__tl_act_reverse_output:n #1 #2 \__tl_act_result:n #3 4507
4508 { #2 \__tl_act_result:n { #1 #3 } } 4508
```

```
4509 \cs_new:Npn \tl_reverse:n #1
4510 {
4511     \__kernel_exp_not:w \exp_after:wN
4512     {
4513         \exp:w
4514         \__tl_act:NNNn
4515         \__tl_reverse_normal:N
4516         \__tl_reverse_group_preserve:n
4517         \__tl_reverse_space:
4518         {#1}
4519     }
4520 }
4521 \cs_generate_variant:Nn \tl_reverse:n { o , V , f , e }
4522 \cs_new:Npn \__tl_reverse_normal:N
4523 { \__tl_act_reverse_output:n }
4524 \cs_new:Npn \__tl_reverse_group_preserve:n #1
4525 { \__tl_act_reverse_output:n { {#1} } }
4526 \cs_new:Npn \__tl_reverse_space:
4527 { \__tl_act_reverse_output:n { ~ } }
4528 \cs_new_protected:Npn \tl_reverse:N #1
4529 { \__kernel_tl_set:Nx #1 { \exp_args:No \tl_reverse:n { #1 } } }
4530 \cs_new_protected:Npn \tl_greverse:N #1
4531 { \__kernel_tl_gset:Nx #1 { \exp_args:No \tl_reverse:n { #1 } } }
4532 \cs_generate_variant:Nn \tl_reverse:N { c }
4533 \cs_generate_variant:Nn \tl_greverse:N { c }
4534 \cs_new:Npn \tl_item:nn #1#2
4535 {
4536     \exp_args:Nf \__tl_item:nn
4537     { \exp_args:Nf \__tl_item_aux:nn { \int_eval:n {#2} } {#1} }
4538     #1
4539     \q__tl_recursion_tail
4540     \prg_break_point:
4541 }
4542 \cs_new:Npn \__tl_item_aux:nn #1#2
4543 {
4544     \int_compare:nNnTF {#1} < 0
4545     { \int_eval:n { \tl_count:n {#2} + 1 + #1 } }
4546     {#1}
4547 }
4548 \cs_new:Npn \__tl_item:nn #1#2
4549 {
4550     \__tl_if_recursion_tail_break:nN {#2} \prg_break:
4551     \int_compare:nNnTF {#1} = 1
4552     { \prg_break:n { \exp_not:n {#2} } }
4553     { \exp_args:Nf \__tl_item:nn { \int_eval:n { #1 - 1 } } }
4554 }
```

```

4555 \cs_new:Npn \tl_item:Nn { \exp_args:No \tl_item:nn }
4556 \cs_generate_variant:Nn \tl_item:Nn { c }
4557 \cs_new:Npn \tl_rand_item:n #1
4558 {
4559     \tl_if_blank:nF {#1}
4560     { \tl_item:nn {#1} { \int_rand:nn { 1 } { \tl_count:n {#1} } } }
4561 }
4562 \cs_new:Npn \tl_rand_item:N { \exp_args:No \tl_rand_item:n }
4563 \cs_generate_variant:Nn \tl_rand_item:N { c }
4564 \cs_new_eq:NN \__kernel_int_sep: \tex_right:D
4565 \cs_new_eq:NN \__tl_sep: \__kernel_int_sep:
4566 \cs_new:Npn \tl_range:Nnn { \exp_args:No \tl_range:nnn }
4567 \cs_generate_variant:Nn \tl_range:Nnn { c }
4568 \cs_new:Npn \tl_range:nnn { \__tl_range:Nnnn \__tl_range:w }
4569 \cs_new:Npn \__tl_range:Nnnn #1#2#3#4
4570 {
4571     \tl_head:f
4572     {
4573         \exp_args:Nf \__tl_range:nnnNn
4574         { \tl_count:n {#2} } {#3} {#4} #1 {#2}
4575     }
4576 }
4577 \cs_new:Npn \__tl_range:nnnNn #1#2#3
4578 {
4579     \exp_args:Nff \__tl_range:nnNn
4580     {
4581         \exp_args:Nf \__tl_range_normalize:nn
4582         { \int_eval:n { #2 - 1 } } {#1}
4583     }
4584     {
4585         \exp_args:Nf \__tl_range_normalize:nn
4586         { \int_eval:n {#3} } {#1}
4587     }
4588 }
4589 \cs_new:Npn \__tl_range:nnNn #1#2#3#4
4590 {
4591     \if_int_compare:w #2 > #1 \exp_stop_f: \else:
4592         \exp_after:wN { \exp_after:wN }
4593     \fi:
4594     \exp_after:wN #3
4595     \int_value:w \int_eval:n { #2 - #1 } \exp_after:wN \__tl_sep:
4596     \exp_after:wN { \exp:w \__tl_range_skip:w #1 \__tl_sep: { } #4 }
4597 }
4598 \cs_new:Npn \__tl_range_skip:w #1 \__tl_sep: #2
4599 {
4600     \if_int_compare:w #1 > \c_zero_int

```

```

4601     \exp_after:wN \__tl_range_skip:w
4602     \int_value:w \int_eval:n { #1 - 1 } \exp_after:wN \__tl_sep:
4603     \else:
4604     \exp_after:wN \exp_end:
4605     \fi:
4606 }
4607 \cs_new:Npn \__tl_range:w #1 \__tl_sep: #2
4608 {
4609     \exp_args:Nf \__tl_range_collect:nn
4610     { \__tl_range_skip_spaces:n {#2} } {#1}
4611 }
4612 \cs_new:Npn \__tl_range_skip_spaces:n #1
4613 {
4614     \tl_if_head_is_space:nTF {#1}
4615     { \exp_args:Nf \__tl_range_skip_spaces:n {#1} }
4616     { { } #1 }
4617 }
4618 \cs_new:Npn \__tl_range_collect:nn #1#2
4619 {
4620     \int_compare:nNnTF {#2} = 0
4621     {#1}
4622     {
4623         \exp_args:No \tl_if_head_is_space:nTF { \use_none:n #1 }
4624         {
4625             \exp_args:Nf \__tl_range_collect:nn
4626             { \__tl_range_collect_space:nw #1 }
4627             {#2}
4628         }
4629         {
4630             \__tl_range_collect:ff
4631             {
4632                 \exp_args:No \tl_if_head_is_N_type:nTF { \use_none:n #1 }
4633                 { \__tl_range_collect_N:nN }
4634                 { \__tl_range_collect_group:nn }
4635                 #1
4636             }
4637             { \int_eval:n { #2 - 1 } }
4638         }
4639     }
4640 }
4641 \cs_new:Npn \__tl_range_collect_space:nw #1 ~ { { #1 ~ } }
4642 \cs_new:Npn \__tl_range_collect_N:nN #1#2 { { #1 #2 } }
4643 \cs_new:Npn \__tl_range_collect_group:nn #1#2 { { #1 {#2} } }
4644 \cs_generate_variant:Nn \__tl_range_collect:nn { ff }
4645 \cs_new:Npn \__tl_range_normalize:nn #1#2
4646 {

```

```

4647 \int_eval:n
4648 {
4649 \if_int_compare:w #1 < \c_zero_int
4650 \if_int_compare:w #1 < -#2 \exp_stop_f:
4651 0
4652 \else:
4653 #1 + #2 + 1
4654 \fi:
4655 \else:
4656 \if_int_compare:w #1 < #2 \exp_stop_f:
4657 #1
4658 \else:
4659 #2
4660 \fi:
4661 \fi:
4662 }
4663 }
4664 \cs_new_protected:Npn \tl_show:N { \__tl_show:NN \tl_show:n }
4665 \cs_generate_variant:Nn \tl_show:N { c }
4666 \cs_new_protected:Npn \tl_log:N { \__tl_show:NN \tl_log:n }
4667 \cs_generate_variant:Nn \tl_log:N { c }
4668 \cs_new_protected:Npn \__tl_show:NN #1#2
4669 {
4670 \__kernel_chk_defined:NT #2
4671 {
4672 \exp_args:Nf \tl_if_empty:nTF
4673 { \cs_prefix_spec:N #2 \cs_parameter_spec:N #2 }
4674 {
4675 \exp_args:Ne #1
4676 { \token_to_str:N #2 = \__kernel_exp_not:w \exp_after:wN {#2} }
4677 }
4678 {
4679 \msg_error:nneee { kernel } { bad-type }
4680 { \token_to_str:N #2 } { \token_to_meaning:N #2 } { tl }
4681 }
4682 }
4683 }
4684 \cs_new_protected:Npn \tl_show:n #1
4685 { \iow_wrap:nnnN { >~ \tl_to_str:n {#1} . } { } { } { } \__tl_show:n }
4686 \cs_generate_variant:Nn \tl_show:n { e , x }
4687 \cs_new_protected:Npn \__tl_show:n #1
4688 {
4689 \tl_set:Nf \l__tl_internal_a_tl { \__tl_show:w #1 \s__tl_stop }
4690 \__kernel_iow_with:Nnn \tex_newlinechar:D { 10 }
4691 {
4692 \__kernel_iow_with:Nnn \tex_errorcontextlines:D { -1 }

```



```

4693         {
4694             \tex_showtokens:D \exp_after:wN \exp_after:wN \exp_after:wN
4695                 { \exp_after:wN \l__tl_internal_a_tl }
4696         }
4697     }
4698 }
4699 \cs_new:Npn \__tl_show:w #1 > #2 . \s__tl_stop {#2}
4700 \cs_new_protected:Npn \tl_log:n #1
4701     { \iow_wrap:nnnN { > ~ \tl_to_str:n {#1} . } { } { } { } \iow_log:n }
4702 \cs_generate_variant:Nn \tl_log:n { e , x }
4703 \cs_new_protected:Npn \__kernel_chk_tl_type:NnnT #1#2#3#4
4704     {
4705         \__kernel_chk_defined:NT #1
4706         {
4707             \exp_args:Nf \tl_if_empty:nTF
4708                 { \cs_prefix_spec:N #1 \cs_parameter_spec:N #1 }
4709                 {
4710                     \tl_set:Ne \l__tl_internal_a_tl {#3}
4711                     \tl_if_eq:NNTF #1 \l__tl_internal_a_tl
4712                         {#4}
4713                     {
4714                         \msg_error:nneeee { kernel } { bad-type }
4715                         { \token_to_str:N #1 } { \tl_to_str:N #1 }
4716                         {#2} { \tl_to_str:N \l__tl_internal_a_tl }
4717                     }
4718                 }
4719             {
4720                 \msg_error:nneeee { kernel } { bad-type }
4721                 { \token_to_str:N #1 } { \token_to_meaning:N #1 } {#2}
4722             }
4723         }
4724     }
4725 \scan_new:N \s__tl_nil
4726 \scan_new:N \s__tl_mark
4727 \scan_new:N \s__tl_stop
4728 \tl_new:N \g_tmpa_tl
4729 \tl_new:N \g_tmpb_tl
4730 \tl_new:N \l_tmpa_tl
4731 \tl_new:N \l_tmpb_tl
4732 \cs_undefine:N \__tl_tmp:w
4733 %% File: l3tl-build.dtx
4734 \cs_new_protected:Npn \tl_build_begin:N #1
4735     { \__tl_build_begin:NN \cs_set_nopar:Npe #1 }
4736 \cs_new_protected:Npn \tl_build_gbegin:N #1
4737     { \__tl_build_begin:NN \cs_gset_nopar:Npe #1 }
4738 \cs_new_protected:Npn \__tl_build_begin:NN #1#2

```

```
4739 { \exp_args:Nc \__tl_build_begin:NNN { \cs_to_str:N #2 ' } #2 #1 } 4739
4740 \cs_new_protected:Npn \__tl_build_begin:NNN #1#2#3 4740
4741 { 4741
4742     #3 #1 { } 4742
4743     #3 #2 4743
4744     { 4744
4745         \exp_not:n { \exp_end: \exp_end: \exp_end: \exp_end: } 4745
4746         \exp_not:n { \__tl_build_last:NNn #3 #1 { } } 4746
4747     } 4747
4748 } 4748
4749 \cs_new_protected:Npn \tl_build_put_right:Nn #1#2 4749
4750 { 4750
4751     \cs_set_nopar:Npe #1 4751
4752     { \__kernel_exp_not:w \exp_after:wN { \exp:w #1 #2 } } 4752
4753 } 4753
4754 \cs_generate_variant:Nn \tl_build_put_right:Nn { Ne , Nx } 4754
4755 \cs_new_protected:Npn \tl_build_gput_right:Nn #1#2 4755
4756 { 4756
4757     \cs_gset_nopar:Npe #1 4757
4758     { \__kernel_exp_not:w \exp_after:wN { \exp:w #1 #2 } } 4758
4759 } 4759
4760 \cs_generate_variant:Nn \tl_build_gput_right:Nn { Ne , Nx } 4760
4761 \cs_new_protected:Npn \__tl_build_last:NNn #1#2 4761
4762 { 4762
4763     \if_false: { { \fi: 4763
4764         \exp_end: \exp_end: \exp_end: \exp_end: \exp_end: 4764
4765         \__tl_build_last:NNn #1 #2 { } 4765
4766     } 4766
4767 } 4767
4768 \if_meaning:w \c_empty_tl #2 4768
4769     \__tl_build_begin:NN #1 #2 4769
4770 \fi: 4770
4771 #1 #2 4771
4772 { 4772
4773     \__kernel_exp_not:w \exp_after:wN 4773
4774     { 4774
4775         \exp:w \if_false: } } \fi: 4775
4776     \exp_after:wN \__tl_build_put:nn \exp_after:wN {#2} 4776
4777 } 4777
4778 \cs_new_protected:Npn \__tl_build_put:nn #1#2 { \__tl_build_put:nw {#2} #1 } 4778
4779 \cs_new_protected:Npn \__tl_build_put:nw #1#2 \__tl_build_last:NNn #3#4#5 4779
4780     { #2 \__tl_build_last:NNn #3 #4 { #1 #5 } } 4780
4781 \cs_new_protected:Npn \tl_build_put_left:Nn #1 4781
4782     { \__tl_build_put_left:NNn \cs_set_nopar:Npe #1 } 4782
4783 \cs_generate_variant:Nn \tl_build_put_left:Nn { Ne , Nx } 4783
4784 \cs_new_protected:Npn \tl_build_gput_left:Nn #1 4784
```

```
4785 { \__tl_build_put_left:NNn \cs_gset_nopar:Npe #1 } 4785
4786 \cs_generate_variant:Nn \tl_build_gput_left:Nn { Ne , Nx } 4786
4787 \cs_new_protected:Npn \__tl_build_put_left:NNn #1#2#3 4787
4788 { 4788
4789     #1 #2 4789
4790     { 4790
4791         \__kernel_exp_not:w \exp_after:wN 4791
4792         { 4792
4793             \exp:w \exp_after:wN \__tl_build_put:nn 4793
4794             \exp_after:wN {#2} {#3} 4794
4795         } 4795
4796     } 4796
4797 } 4797
4798 \cs_new_protected:Npn \tl_build_end:N #1 4798
4799 { 4799
4800     \__tl_build_get:NNN \__kernel_tl_set:Nx #1 #1 4800
4801     \exp_args:Nc \__tl_build_end_loop:NN { \cs_to_str:N #1 ' } \tl_clear:N 4801
4802 } 4802
4803 \cs_new_protected:Npn \tl_build_gend:N #1 4803
4804 { 4804
4805     \__tl_build_get:NNN \__kernel_tl_gset:Nx #1 #1 4805
4806     \exp_args:Nc \__tl_build_end_loop:NN { \cs_to_str:N #1 ' } \tl_gclear:N 4806
4807 } 4807
4808 \cs_new_protected:Npn \__tl_build_end_loop:NN #1#2 4808
4809 { 4809
4810     \if_meaning:w \c_empty_tl #1 4810
4811     \exp_after:wN \use_none:nnnnnn 4811
4812     \fi: 4812
4813     #2 #1 4813
4814     \exp_args:Nc \__tl_build_end_loop:NN { \cs_to_str:N #1 ' } #2 4814
4815 } 4815
4816 \cs_new_protected:Npn \tl_build_get_intermediate:NN 4816
4817 { \__tl_build_get:NNN \__kernel_tl_set:Nx } 4817
4818 \cs_new_protected:Npn \__tl_build_get:NNN #1#2#3 4818
4819 { #1 #3 { \if_false: { \fi: \exp_after:wN \__tl_build_get:w #2 } } } 4819
4820 \cs_new:Npn \__tl_build_get:w #1 \__tl_build_last:NNn #2#3#4 4820
4821 { 4821
4822     \exp_not:n {#4} 4822
4823     \if_meaning:w \c_empty_tl #3 4823
4824     \exp_after:wN \__tl_build_get_end:w 4824
4825     \fi: 4825
4826     \exp_after:wN \__tl_build_get:w #3 4826
4827 } 4827
4828 \cs_new:Npn \__tl_build_get_end:w #1#2#3 4828
4829 { \__kernel_exp_not:w \exp_after:wN { \if_false: } \fi: } 4829
4830 %% File: l3str.dtx 4830
```

```
4831 \scan_new:N \s__str_mark 4831
4832 \scan_new:N \s__str_stop 4832
4833 \cs_new:Npn \__str_use_none_delimit_by_s_stop:w #1 \s__str_stop { } 4833
4834 \cs_new:Npn \__str_use_i_delimit_by_s_stop:nw #1 #2 \s__str_stop {#1} 4834
4835 \quark_new:N \q__str_recursion_tail 4835
4836 \quark_new:N \q__str_recursion_stop 4836
4837 \__kernel_quark_new_test:N \__str_if_recursion_tail_break:NN 4837
4838 \__kernel_quark_new_test:N \__str_if_recursion_tail_stop_do:Nn 4838
4839 \group_begin: 4839
4840 \cs_set_protected:Npn \__str_tmp:n #1 4840
4841 { 4841
4842 \tl_if_blank:nF {#1} 4842
4843 { 4843
4844 \cs_new_eq:cc { str_ #1 :N } { tl_ #1 :N } 4844
4845 \exp_args:Nc \cs_generate_variant:Nn { str_ #1 :N } { c } 4845
4846 \__str_tmp:n 4846
4847 } 4847
4848 } 4848
4849 \__str_tmp:n 4849
4850 { new } 4850
4851 { use } 4851
4852 { clear } 4852
4853 { gclear } 4853
4854 { clear_new } 4854
4855 { gclear_new } 4855
4856 { } 4856
4857 \group_end: 4857
4858 \cs_new_eq:NN \str_set_eq:NN \tl_set_eq:NN 4858
4859 \cs_new_eq:NN \str_gset_eq:NN \tl_gset_eq:NN 4859
4860 \cs_generate_variant:Nn \str_set_eq:NN { c , Nc , cc } 4860
4861 \cs_generate_variant:Nn \str_gset_eq:NN { c , Nc , cc } 4861
4862 \cs_new_eq:NN \str_concat:NNN \tl_concat:NNN 4862
4863 \cs_new_eq:NN \str_gconcat:NNN \tl_gconcat:NNN 4863
4864 \cs_generate_variant:Nn \str_concat:NNN { ccc } 4864
4865 \cs_generate_variant:Nn \str_gconcat:NNN { ccc } 4865
4866 \cs_new_protected:Npn \str_set:Nn #1#2 4866
4867 { \__kernel_tl_set:Nx #1 { \__kernel_tl_to_str:w {#2} } } 4867
4868 \cs_gset_protected:Npn \str_gset:Nn #1#2 4868
4869 { \__kernel_tl_gset:Nx #1 { \__kernel_tl_to_str:w {#2} } } 4869
4870 \cs_new_protected:Npn \str_const:Nn #1#2 4870
4871 { 4871
4872 \__kernel_chk_if_free_cs:N #1 4872
4873 \cs_gset_nopar:Npe #1 { \__kernel_tl_to_str:w {#2} } 4873
4874 } 4874
4875 \cs_new_protected:Npn \str_put_left:Nn #1#2 4875
4876 { 4876
```

```
4877     \__kernel_tl_set:Nx #1
4878     { \__kernel_tl_to_str:w {#2} \__kernel_exp_not:w \exp_after:wN {#1} }
4879 }
4880 \cs_new_protected:Npn \str_gput_left:Nn #1#2
4881 {
4882     \__kernel_tl_gset:Nx #1
4883     { \__kernel_tl_to_str:w {#2} \__kernel_exp_not:w \exp_after:wN {#1} }
4884 }
4885 \cs_new_protected:Npn \str_put_right:Nn #1#2
4886 {
4887     \__kernel_tl_set:Nx #1
4888     { \__kernel_exp_not:w \exp_after:wN {#1} \__kernel_tl_to_str:w {#2} }
4889 }
4890 \cs_new_protected:Npn \str_gput_right:Nn #1#2
4891 {
4892     \__kernel_tl_gset:Nx #1
4893     { \__kernel_exp_not:w \exp_after:wN {#1} \__kernel_tl_to_str:w {#2} }
4894 }
4895 \cs_generate_variant:Nn \str_set:Nn { NV , Ne , Nx , c , cV , ce , cx }
4896 \cs_generate_variant:Nn \str_gset:Nn { NV , Ne , Nx , c , cV , ce , cx }
4897 \cs_generate_variant:Nn \str_const:Nn { NV , Ne , Nx , c , cV , ce , cx }
4898 \cs_generate_variant:Nn \str_put_left:Nn { NV , Ne , Nx , c , cV , ce , cx }
4899 \cs_generate_variant:Nn \str_gput_left:Nn { NV , Ne , Nx , c , cV , ce , cx }
4900 \cs_generate_variant:Nn \str_put_right:Nn { NV , Ne , Nx , c , cV , ce , cx }
4901 \cs_generate_variant:Nn \str_gput_right:Nn { NV , Ne , Nx , c , cV , ce , cx }
4902 \cs_new_protected:Npn \str_replace_once:Nnn
4903 { \__str_replace:NNNnn \prg_do_nothing: \__kernel_tl_set:Nx }
4904 \cs_new_protected:Npn \str_greplace_once:Nnn
4905 { \__str_replace:NNNnn \prg_do_nothing: \__kernel_tl_gset:Nx }
4906 \cs_new_protected:Npn \str_replace_all:Nnn
4907 { \__str_replace:NNNnn \__str_replace_next:w \__kernel_tl_set:Nx }
4908 \cs_new_protected:Npn \str_greplace_all:Nnn
4909 { \__str_replace:NNNnn \__str_replace_next:w \__kernel_tl_gset:Nx }
4910 \cs_generate_variant:Nn \str_replace_once:Nnn { c }
4911 \cs_generate_variant:Nn \str_greplace_once:Nnn { c }
4912 \cs_generate_variant:Nn \str_replace_all:Nnn { c }
4913 \cs_generate_variant:Nn \str_greplace_all:Nnn { c }
4914 \cs_new_protected:Npn \__str_replace:NNNnn #1#2#3#4#5
4915 {
4916     \tl_if_empty:nTF {#4}
4917     {
4918         \msg_error:nne { kernel } { empty-search-pattern } {#5}
4919     }
4920     {
4921         \use:e
4922     }
```

```

4923         \exp_not:n { \__str_replace_aux:NNNnnn #1 #2 #3 }
4924         { \tl_to_str:N #3 }
4925         { \tl_to_str:n {#4} } { \tl_to_str:n {#5} }
4926     }
4927 }
4928 }
4929 \cs_new_protected:Npn \__str_replace_aux:NNNnnn #1#2#3#4#5#6
4930 {
4931     \cs_set:Npn \__str_replace_next:w ##1 #5 { ##1 #6 #1 }
4932     #2 #3
4933     {
4934         \__str_replace_next:w
4935         #4
4936         \__str_use_none_delimit_by_s_stop:w
4937         #5
4938         \s__str_stop
4939     }
4940 }
4941 \cs_new_eq:NN \__str_replace_next:w ?
4942 \cs_new_protected:Npn \str_remove_once:Nn #1#2
4943 { \str_replace_once:Nnn #1 {#2} { } }
4944 \cs_new_protected:Npn \str_gremove_once:Nn #1#2
4945 { \str_greplace_once:Nnn #1 {#2} { } }
4946 \cs_generate_variant:Nn \str_remove_once:Nn { c }
4947 \cs_generate_variant:Nn \str_gremove_once:Nn { c }
4948 \cs_new_protected:Npn \str_remove_all:Nn #1#2
4949 { \str_replace_all:Nnn #1 {#2} { } }
4950 \cs_new_protected:Npn \str_gremove_all:Nn #1#2
4951 { \str_greplace_all:Nnn #1 {#2} { } }
4952 \cs_generate_variant:Nn \str_remove_all:Nn { c }
4953 \cs_generate_variant:Nn \str_gremove_all:Nn { c }
4954 \prg_new_eq_conditional:NNn \str_if_exist:N \tl_if_exist:N
4955 { p , T , F , TF }
4956 \prg_new_eq_conditional:NNn \str_if_exist:c \tl_if_exist:c
4957 { p , T , F , TF }
4958 \prg_new_eq_conditional:NNn \str_if_empty:N \tl_if_empty:N
4959 { p , T , F , TF }
4960 \prg_new_eq_conditional:NNn \str_if_empty:c \tl_if_empty:c
4961 { p , T , F , TF }
4962 \prg_new_eq_conditional:NNn \str_if_empty:n \tl_if_empty:n
4963 { p , T , F , TF }
4964 \cs_new_eq:NN \__str_if_eq:nn \tex_strcmp:D
4965 \prg_new_conditional:Npnn \str_compare:nNn #1#2#3 { p , T , F , TF }
4966 {
4967     \if_int_compare:w
4968         \__str_if_eq:nn { \exp_not:n {#1} } { \exp_not:n {#3} }

```



```

4969         #2 \c_zero_int
4970         \prg_return_true: \else: \prg_return_false: \fi:
4971     }
4972 \prg_new_conditional:Npnn \str_compare:eNe #1#2#3 { p , T , F , TF }
4973 {
4974     \if_int_compare:w \__str_if_eq:nn {#1} {#3} #2 \c_zero_int
4975     \prg_return_true: \else: \prg_return_false: \fi:
4976 }
4977 \prg_new_conditional:Npnn \str_if_eq:nn #1#2 { p , T , F , TF }
4978 {
4979     \if:w 0 \__str_if_eq:nn { \exp_not:n {#1} } { \exp_not:n {#2} }
4980     \prg_return_true: \else: \prg_return_false: \fi:
4981 }
4982 \prg_generate_conditional_variant:Nnn \str_if_eq:nn
4983 { V , v , o , nV , no , VV , nv } { p , T , F , TF }
4984 \prg_new_conditional:Npnn \str_if_eq:ee #1#2 { p , T , F , TF }
4985 {
4986     \if:w 0 \__str_if_eq:nn {#1} {#2}
4987     \prg_return_true: \else: \prg_return_false: \fi:
4988 }
4989 \prg_new_conditional:Npnn \str_if_eq:NN #1#2 { p , TF , T , F }
4990 {
4991     \if:w 0 \__str_if_eq:nn { \tl_to_str:N #1 } { \tl_to_str:N #2 }
4992     \prg_return_true: \else: \prg_return_false: \fi:
4993 }
4994 \prg_generate_conditional_variant:Nnn \str_if_eq:NN
4995 { c , Nc , cc } { T , F , TF , p }
4996 \prg_new_protected_conditional:Npnn \str_if_in:Nn #1#2 { T , F , TF }
4997 {
4998     \use:e
4999     { \tl_if_in:nnTF { \tl_to_str:N #1 } { \tl_to_str:n {#2} } }
5000     { \prg_return_true: } { \prg_return_false: }
5001 }
5002 \prg_generate_conditional_variant:Nnn \str_if_in:Nn
5003 { c } { T , F , TF }
5004 \prg_new_protected_conditional:Npnn \str_if_in:nn #1#2 { T , F , TF }
5005 {
5006     \use:e
5007     { \tl_if_in:nnTF { \tl_to_str:n {#1} } { \tl_to_str:n {#2} } }
5008     { \prg_return_true: } { \prg_return_false: }
5009 }
5010 \cs_new:Npn \str_case:nn #1#2
5011 {
5012     \exp:w
5013     \__str_case:nnTF {#1} {#2} { } { }
5014 }

```

5015	\cs_new:Npn \str_case:nnT #1#2#3	5015
5016	{	5016
5017	\exp:w	5017
5018	__str_case:nnTF {#1} {#2} {#3} { }	5018
5019	}	5019
5020	\cs_new:Npn \str_case:nnF #1#2	5020
5021	{	5021
5022	\exp:w	5022
5023	__str_case:nnTF {#1} {#2} { }	5023
5024	}	5024
5025	\cs_new:Npn \str_case:nnTF #1#2	5025
5026	{	5026
5027	\exp:w	5027
5028	__str_case:nnTF {#1} {#2}	5028
5029	}	5029
5030	\cs_new:Npn __str_case:nnTF #1#2#3#4	5030
5031	{ __str_case:nw {#1} #2 {#1} { } \s__str_mark {#3} \s__str_mark {#4} \s__str_stop }	5031
5032	\cs_generate_variant:Nn \str_case:nn { V , o , e , nV , nv , ne }	5032
5033	\prg_generate_conditional_variant:Nnn \str_case:nn	5033
5034	{ V , o , e , nV , nv , ne } { T , F , TF }	5034
5035	\cs_new_eq:NN \str_case:Nn \str_case:Vn	5035
5036	\cs_new_eq:NN \str_case:NnT \str_case:VnT	5036
5037	\cs_new_eq:NN \str_case:NnF \str_case:VnF	5037
5038	\cs_new_eq:NN \str_case:NnTF \str_case:VnTF	5038
5039	\cs_new:Npn __str_case:nw #1#2#3	5039
5040	{	5040
5041	\str_if_eq:nnTF {#1} {#2}	5041
5042	{ __str_case_end:nw {#3} }	5042
5043	{ __str_case:nw {#1} }	5043
5044	}	5044
5045	\cs_new:Npn \str_case_e:nn #1#2	5045
5046	{	5046
5047	\exp:w	5047
5048	__str_case_e:nnTF {#1} {#2} { } { }	5048
5049	}	5049
5050	\cs_new:Npn \str_case_e:nnT #1#2#3	5050
5051	{	5051
5052	\exp:w	5052
5053	__str_case_e:nnTF {#1} {#2} {#3} { }	5053
5054	}	5054
5055	\cs_new:Npn \str_case_e:nnF #1#2	5055
5056	{	5056
5057	\exp:w	5057
5058	__str_case_e:nnTF {#1} {#2} { }	5058
5059	}	5059
5060	\cs_new:Npn \str_case_e:nnTF #1#2	5060

```
5061 { 5061
5062     \exp:w 5062
5063     \__str_case_e:nnTF {#1} {#2} 5063
5064 } 5064
5065 \cs_new:Npn \__str_case_e:nnTF #1#2#3#4 5065
5066 { \__str_case_e:nw {#1} #2 {#1} { } \s__str_mark {#3} \s__str_mark {#4} \s__str_stop } 5066
5067 \cs_generate_variant:Nn \str_case_e:nn { e } 5067
5068 \prg_generate_conditional_variant:Nnn \str_case_e:nn { e } { T , F , TF } 5068
5069 \cs_new:Npn \__str_case_e:nw #1#2#3 5069
5070 { 5070
5071     \str_if_eq:eeTF {#1} {#2} 5071
5072     { \__str_case_end:nw {#3} } 5072
5073     { \__str_case_e:nw {#1} } 5073
5074 } 5074
5075 \cs_new:Npn \__str_case_end:nw #1#2#3 \s__str_mark #4#5 \s__str_stop 5075
5076 { \exp_end: #1 #4 } 5076
5077 \cs_new:Npn \str_map_function:nN #1#2 5077
5078 { 5078
5079     \exp_after:wN \__str_map_function:w 5079
5080     \exp_after:wN \__str_map_function:nn \exp_after:wN #2 5080
5081     \__kernel_tl_to_str:w {#1} 5081
5082     \q__str_recursion_tail ? ~ 5082
5083     \prg_break_point:Nn \str_map_break: { } 5083
5084 } 5084
5085 \cs_new:Npn \str_map_function:NN 5085
5086 { \exp_args:No \str_map_function:nN } 5086
5087 \cs_new:Npn \__str_map_function:w #1 ~ 5087
5088 { #1 { ~ { ~ } } \__str_map_function:w } } 5088
5089 \cs_new:Npn \__str_map_function:nn #1#2 5089
5090 { 5090
5091     \if_meaning:w \q__str_recursion_tail #2 5091
5092     \exp_after:wN \str_map_break: 5092
5093     \fi: 5093
5094     #1 #2 \__str_map_function:nn {#1} 5094
5095 } 5095
5096 \cs_generate_variant:Nn \str_map_function:NN { c } 5096
5097 \cs_new_protected:Npn \str_map_inline:nn #1#2 5097
5098 { 5098
5099     \int_gincr:N \g__kernel_prg_map_int 5099
5100     \cs_gset_protected:cpn 5100
5101     { __str_map_ \int_use:N \g__kernel_prg_map_int :w } ##1 {#2} 5101
5102     \use:e 5102
5103     { 5103
5104         \exp_not:N \__str_map_inline:NN 5104
5105         \exp_not:c { __str_map_ \int_use:N \g__kernel_prg_map_int :w } 5105
5106         \__kernel_str_to_other_fast:n {#1} 5106
```

5107	}	5107
5108	\q__str_recursion_tail	5108
5109	\prg_break_point:Nn \str_map_break:	5109
5110	{ \int_gdecr:N \g__kernel_prg_map_int }	5110
5111	}	5111
5112	\cs_new_protected:Npn \str_map_inline:Nn	5112
5113	{ \exp_args:No \str_map_inline:nn }	5113
5114	\cs_generate_variant:Nn \str_map_inline:Nn { c }	5114
5115	\cs_new:Npn __str_map_inline:NN #1#2	5115
5116	{	5116
5117	__str_if_recursion_tail_break:NN #2 \str_map_break:	5117
5118	\exp_args:No #1 { \token_to_str:N #2 }	5118
5119	__str_map_inline:NN #1	5119
5120	}	5120
5121	\cs_new_protected:Npn \str_map_variable:nNn #1#2#3	5121
5122	{	5122
5123	\use:e	5123
5124	{	5124
5125	\exp_not:n { __str_map_variable:NnN #2 {#3} }	5125
5126	__kernel_str_to_other_fast:n {#1}	5126
5127	}	5127
5128	\q__str_recursion_tail	5128
5129	\prg_break_point:Nn \str_map_break: { }	5129
5130	}	5130
5131	\cs_new_protected:Npn \str_map_variable:NNn	5131
5132	{ \exp_args:No \str_map_variable:nNn }	5132
5133	\cs_new_protected:Npn __str_map_variable:NnN #1#2#3	5133
5134	{	5134
5135	__str_if_recursion_tail_break:NN #3 \str_map_break:	5135
5136	\str_set:Nn #1 {#3}	5136
5137	\use:n {#2}	5137
5138	__str_map_variable:NnN #1 {#2}	5138
5139	}	5139
5140	\cs_generate_variant:Nn \str_map_variable:NNn { c }	5140
5141	\cs_new:Npn \str_map_break:	5141
5142	{ \prg_map_break:Nn \str_map_break: { } }	5142
5143	\cs_new:Npn \str_map_break:n	5143
5144	{ \prg_map_break:Nn \str_map_break: }	5144
5145	\cs_new:Npn \str_map_tokens:nn #1#2	5145
5146	{	5146
5147	\exp_args:Nno \use:nn	5147
5148	{ __str_map_function:w __str_map_function:nn {#2} }	5148
5149	{ __kernel_tl_to_str:w {#1} }	5149
5150	\q__str_recursion_tail ? ~	5150
5151	\prg_break_point:Nn \str_map_break: { }	5151
5152	}	5152

```
5153 \cs_new:Npn \str_map_tokens:Nn { \exp_args:No \str_map_tokens:nn } 5153
5154 \cs_generate_variant:Nn \str_map_tokens:Nn { c } 5154
5155 \cs_new:Npn \__kernel_str_to_other:n #1 5155
5156 { 5156
5157   \exp_after:wN \__str_to_other_loop:w 5157
5158   \tl_to_str:n {#1} ~ A ~ A ~ A ~ A ~ A ~ A ~ A ~ A ~ \s__str_mark \s__str_stop 5158
5159 } 5159
5160 \group_begin: 5160
5161 \tex_lccode:D \* = \_ % 5161
5162 \tex_lccode:D \A = \A % 5162
5163 \tex_lowercase:D 5163
5164 { 5164
5165   \group_end: 5165
5166   \cs_new:Npn \__str_to_other_loop:w 5166
5167     #1 ~ #2 ~ #3 ~ #4 ~ #5 ~ #6 ~ #7 ~ #8 ~ #9 \s__str_stop 5167
5168   { 5168
5169     \if_meaning:w A #8 5169
5170     \__str_to_other_end:w 5170
5171     \fi: 5171
5172     \__str_to_other_loop:w 5172
5173     #9 #1 * #2 * #3 * #4 * #5 * #6 * #7 * #8 * \s__str_stop 5173
5174   } 5174
5175   \cs_new:Npn \__str_to_other_end:w \fi: #1 \s__str_mark #2 * A #3 \s__str_stop 5175
5176   { \fi: #2 } 5176
5177 } 5177
5178 \cs_new:Npn \__kernel_str_to_other_fast:n #1 5178
5179 { 5179
5180   \exp_after:wN \__str_to_other_fast_loop:w \tl_to_str:n {#1} ~ 5180
5181   A ~ A ~ A ~ A ~ A ~ A ~ A ~ A ~ A ~ \s__str_stop 5181
5182 } 5182
5183 \group_begin: 5183
5184 \tex_lccode:D \* = \_ % 5184
5185 \tex_lccode:D \A = \A % 5185
5186 \tex_lowercase:D 5186
5187 { 5187
5188   \group_end: 5188
5189   \cs_new:Npn \__str_to_other_fast_loop:w 5189
5190     #1 ~ #2 ~ #3 ~ #4 ~ #5 ~ #6 ~ #7 ~ #8 ~ #9 ~ 5190
5191   { 5191
5192     \if_meaning:w A #9 5192
5193     \__str_to_other_fast_end:w 5193
5194     \fi: 5194
5195     #1 * #2 * #3 * #4 * #5 * #6 * #7 * #8 * #9 5195
5196     \__str_to_other_fast_loop:w * 5196
5197   } 5197
5198   \cs_new:Npn \__str_to_other_fast_end:w #1 * A #2 \s__str_stop {#1} 5198
```

5199	}	5199
5200	\cs_new_eq:NN __str_sep: __kernel_int_sep:	5200
5201	\cs_new:Npn \str_item:Nn { \exp_args:No \str_item:nn }	5201
5202	\cs_generate_variant:Nn \str_item:Nn { c }	5202
5203	\cs_new:Npn \str_item:nn #1#2	5203
5204	{	5204
5205	\exp_args:Nf \tl_to_str:n	5205
5206	{	5206
5207	\exp_args:Nf __str_item:nn	5207
5208	{ __kernel_str_to_other:n {#1} } {#2}	5208
5209	}	5209
5210	}	5210
5211	\cs_new:Npn \str_item_ignore_spaces:nn #1	5211
5212	{ \exp_args:No __str_item:nn { \tl_to_str:n {#1} } }	5212
5213	\cs_new:Npn __str_item:nn #1#2	5213
5214	{	5214
5215	\exp_after:wN __str_item:w	5215
5216	\int_value:w \int_eval:n {#2} \exp_after:wN __str_sep:	5216
5217	\int_value:w __str_count:n {#1} __str_sep:	5217
5218	#1 \s__str_stop	5218
5219	}	5219
5220	\cs_new:Npn __str_item:w #1 __str_sep: #2 __str_sep:	5220
5221	{	5221
5222	\int_compare:nNnTF {#1} < 0	5222
5223	{	5223
5224	\int_compare:nNnTF {#1} < {-#2}	5224
5225	{ __str_use_none_delimit_by_s_stop:w }	5225
5226	{	5226
5227	\exp_after:wN __str_use_i_delimit_by_s_stop:nw	5227
5228	\exp:w \exp_after:wN __str_skip_exp_end:w	5228
5229	\int_value:w \int_eval:n { #1 + #2 } __str_sep:	5229
5230	}	5230
5231	}	5231
5232	{	5232
5233	\int_compare:nNnTF {#1} > {#2}	5233
5234	{ __str_use_none_delimit_by_s_stop:w }	5234
5235	{	5235
5236	\exp_after:wN __str_use_i_delimit_by_s_stop:nw	5236
5237	\exp:w __str_skip_exp_end:w #1 __str_sep: { }	5237
5238	}	5238
5239	}	5239
5240	}	5240
5241	\cs_new:Npn __str_skip_exp_end:w #1 __str_sep:	5241
5242	{	5242
5243	\if_int_compare:w #1 > 8 \exp_stop_f:	5243
5244	\exp_after:wN __str_skip_loop:wNNNNNNNN	5244


```
5245 \else: 5245
5246 \exp_after:wN \__str_skip_end:w 5246
5247 \int_value:w \int_eval:w 5247
5248 \fi: 5248
5249 #1 \__str_sep: 5249
5250 } 5250
5251 \cs_new:Npn \__str_skip_loop:wNNNNNNNN #1 \__str_sep: #2#3#4#5#6#7#8#9 5251
5252 { 5252
5253 \exp_after:wN \__str_skip_exp_end:w 5253
5254 \int_value:w \int_eval:n { #1 - 8 } \__str_sep: 5254
5255 } 5255
5256 \cs_new:Npn \__str_skip_end:w #1 \__str_sep: 5256
5257 { 5257
5258 \exp_after:wN \__str_skip_end:NNNNNNNN 5258
5259 \if_case:w #1 \exp_stop_f: \or: \or: \or: \or: \or: \or: \or: 5259
5260 } 5260
5261 \cs_new:Npn \__str_skip_end:NNNNNNNN #1#2#3#4#5#6#7#8 { \fi: \exp_end: } 5261
5262 \cs_new:Npn \str_range:Nnn { \exp_args:No \str_range:nnn } 5262
5263 \cs_generate_variant:Nn \str_range:Nnn { c } 5263
5264 \cs_new:Npn \str_range:nnn #1#2#3 5264
5265 { 5265
5266 \exp_args:Nf \tl_to_str:n 5266
5267 { 5267
5268 \exp_args:Nf \__str_range:nnn 5268
5269 { \__kernel_str_to_other:n {#1} } {#2} {#3} 5269
5270 } 5270
5271 } 5271
5272 \cs_new:Npn \str_range_ignore_spaces:nnn #1 5272
5273 { \exp_args:No \__str_range:nnn { \tl_to_str:n {#1} } } 5273
5274 \cs_new:Npn \__str_range:nnn #1#2#3 5274
5275 { 5275
5276 \exp_after:wN \__str_range:w 5276
5277 \int_value:w \__str_count:n {#1} \exp_after:wN \__str_sep: 5277
5278 \int_value:w \int_eval:n { (#2) - 1 } \exp_after:wN \__str_sep: 5278
5279 \int_value:w \int_eval:n {#3} \__str_sep: 5279
5280 #1 \s__str_stop 5280
5281 } 5281
5282 \cs_new:Npn \__str_range:w #1 \__str_sep: #2 \__str_sep: #3 \__str_sep: 5282
5283 { 5283
5284 \exp_args:Nf \__str_range:nnw 5284
5285 { \__str_range_normalize:nn {#2} {#1} } 5285
5286 { \__str_range_normalize:nn {#3} {#1} } 5286
5287 } 5287
5288 \cs_new:Npn \__str_range:nnw #1#2 5288
5289 { 5289
5290 \exp_after:wN \__str_collect_delimit_by_q_stop:w 5290
```

5291	\int_value:w \int_eval:n { #2 - #1 } \exp_after:wN __str_sep:	5291
5292	\exp:w __str_skip_exp_end:w #1 __str_sep:	5292
5293	}	5293
5294	\cs_new:Npn __str_range_normalize:nn #1#2	5294
5295	{	5295
5296	\int_eval:n	5296
5297	{	5297
5298	\if_int_compare:w #1 < \c_zero_int	5298
5299	\if_int_compare:w #1 < -#2 \exp_stop_f:	5299
5300	0	5300
5301	\else:	5301
5302	#1 + #2 + 1	5302
5303	\fi:	5303
5304	\else:	5304
5305	\if_int_compare:w #1 < #2 \exp_stop_f:	5305
5306	#1	5306
5307	\else:	5307
5308	#2	5308
5309	\fi:	5309
5310	\fi:	5310
5311	}	5311
5312	}	5312
5313	\cs_new:Npn __str_collect_delimit_by_q_stop:w #1 __str_sep:	5313
5314	{ __str_collect_loop:wn #1 __str_sep: { } }	5314
5315	\cs_new:Npn __str_collect_loop:wn #1 __str_sep:	5315
5316	{	5316
5317	\if_int_compare:w #1 > 7 \exp_stop_f:	5317
5318	\exp_after:wN __str_collect_loop:wnNNNNNNN	5318
5319	\else:	5319
5320	\exp_after:wN __str_collect_end:wn	5320
5321	\fi:	5321
5322	#1 __str_sep:	5322
5323	}	5323
5324	\cs_new:Npn __str_collect_loop:wnNNNNNNN #1 __str_sep: #2 #3#4#5#6#7#8#9	5324
5325	{	5325
5326	\exp_after:wN __str_collect_loop:wn	5326
5327	\int_value:w \int_eval:n { #1 - 7 } __str_sep:	5327
5328	{ #2 #3#4#5#6#7#8#9 }	5328
5329	}	5329
5330	\cs_new:Npn __str_collect_end:wn #1 __str_sep:	5330
5331	{	5331
5332	\exp_after:wN __str_collect_end:nnnnnnnnw	5332
5333	\if_case:w \if_int_compare:w #1 > \c_zero_int	5333
5334	#1 \else: 0 \fi: \exp_stop_f:	5334
5335	\or: \or: \or: \or: \or: \or: \fi:	5335
5336	}	5336

```
5337 \cs_new:Npn \__str_collect_end:nnnnnnnnnw #1#2#3#4#5#6#7#8 #9 \s__str_stop 5337
5338 { #1#2#3#4#5#6#7#8 } 5338
5339 \cs_new:Npn \str_count_spaces:N 5339
5340 { \exp_args:No \str_count_spaces:n } 5340
5341 \cs_generate_variant:Nn \str_count_spaces:N { c } 5341
5342 \cs_new:Npn \str_count_spaces:n #1 5342
5343 { 5343
5344 \int_eval:n 5344
5345 { 5345
5346 \exp_after:wN \__str_count_spaces_loop:w 5346
5347 \tl_to_str:n {#1} ~ 5347
5348 X 7 ~ X 6 ~ X 5 ~ X 4 ~ X 3 ~ X 2 ~ X 1 ~ X 0 ~ X -1 ~ 5348
5349 \s__str_stop 5349
5350 } 5350
5351 } 5351
5352 \cs_new:Npn \__str_count_spaces_loop:w #1~#2~#3~#4~#5~#6~#7~#8~#9~ 5352
5353 { 5353
5354 \if_meaning:w X #9 5354
5355 \__str_use_i_delimit_by_s_stop:nw 5355
5356 \fi: 5356
5357 9 + \__str_count_spaces_loop:w 5357
5358 } 5358
5359 \cs_new:Npn \str_count:N { \exp_args:No \str_count:n } 5359
5360 \cs_generate_variant:Nn \str_count:N { c } 5360
5361 \cs_new:Npn \str_count:n #1 5361
5362 { 5362
5363 \__str_count_aux:n 5363
5364 { 5364
5365 \str_count_spaces:n {#1} 5365
5366 + \exp_after:wN \__str_count_loop:NNNNNNNNN \tl_to_str:n {#1} 5366
5367 } 5367
5368 } 5368
5369 \cs_new:Npn \__str_count:n #1 5369
5370 { 5370
5371 \__str_count_aux:n 5371
5372 { \__str_count_loop:NNNNNNNNN #1 } 5372
5373 } 5373
5374 \cs_new:Npn \str_count_ignore_spaces:n #1 5374
5375 { 5375
5376 \__str_count_aux:n 5376
5377 { \exp_after:wN \__str_count_loop:NNNNNNNNN \tl_to_str:n {#1} } 5377
5378 } 5378
5379 \cs_new:Npn \__str_count_aux:n #1 5379
5380 { 5380
5381 \int_eval:n 5381
5382 { 5382
```

```
5383         #1 5383
5384         { X 8 } { X 7 } { X 6 } 5384
5385         { X 5 } { X 4 } { X 3 } 5385
5386         { X 2 } { X 1 } { X 0 } 5386
5387         \s__str_stop 5387
5388     } 5388
5389 } 5389
5390 \cs_new:Npn \__str_count_loop:NNNNNNNNN #1#2#3#4#5#6#7#8#9 5390
5391 { 5391
5392     \if_meaning:w X #9 5392
5393     \exp_after:wN \__str_use_none_delimit_by_s_stop:w 5393
5394     \fi: 5394
5395     9 + \__str_count_loop:NNNNNNNNN 5395
5396 } 5396
5397 \cs_new:Npn \str_head:N { \exp_args:No \str_head:n } 5397
5398 \cs_generate_variant:Nn \str_head:N { c } 5398
5399 \cs_new:Npn \str_head:n #1 5399
5400 { 5400
5401     \exp_after:wN \__str_head:w 5401
5402     \tl_to_str:n {#1} 5402
5403     { { } } ~ \s__str_stop 5403
5404 } 5404
5405 \cs_new:Npn \__str_head:w #1 ~ % 5405
5406 { \__str_use_i_delimit_by_s_stop:nw #1 { ~ } } 5406
5407 \cs_new:Npn \str_head_ignore_spaces:n #1 5407
5408 { 5408
5409     \exp_after:wN \__str_use_i_delimit_by_s_stop:nw 5409
5410     \tl_to_str:n {#1} { } \s__str_stop 5410
5411 } 5411
5412 \cs_new:Npn \str_tail:N { \exp_args:No \str_tail:n } 5412
5413 \cs_generate_variant:Nn \str_tail:N { c } 5413
5414 \cs_new:Npn \str_tail:n #1 5414
5415 { 5415
5416     \exp_after:wN \__str_tail_auxi:w 5416
5417     \reverse_if:N \if_charcode:w 5417
5418     \scan_stop: \tl_to_str:n {#1} X X \s__str_stop 5418
5419 } 5419
5420 \cs_new:Npn \__str_tail_auxi:w #1 X #2 \s__str_stop { \fi: #1 } 5420
5421 \cs_new:Npn \str_tail_ignore_spaces:n #1 5421
5422 { 5422
5423     \exp_after:wN \__str_tail_auxii:w 5423
5424     \tl_to_str:n {#1} \s__str_mark \s__str_mark \s__str_stop 5424
5425 } 5425
5426 \cs_new:Npn \__str_tail_auxii:w #1 #2 \s__str_mark #3 \s__str_stop { #2 } 5426
5427 \cs_new:Npn \str_casefold:n #1 { \__str_change_case:nn {#1} { casefold } } 5427
5428 \cs_new:Npn \str_lowercase:n #1 { \__str_change_case:nn {#1} { lowercase } } 5428
```

```

5429 \cs_new:Npn \str_uppercase:n #1 { \__str_change_case:nn {#1} { uppercase } }
5430 \cs_generate_variant:Nn \str_casefold:n { V }
5431 \cs_generate_variant:Nn \str_lowercase:n { f }
5432 \cs_generate_variant:Nn \str_uppercase:n { f }
5433 \cs_new:Npn \__str_change_case:nn #1
5434 {
5435     \exp_after:wN \__str_change_case_aux:nn \exp_after:wN
5436     { \tl_to_str:n {#1} }
5437 }
5438 \cs_new:Npn \__str_change_case_aux:nn #1#2
5439 {
5440     \__str_change_case_loop:nw {#2} #1 \q__str_recursion_tail \q__str_recursion_stop
5441     \__str_change_case_result:n { }
5442 }
5443 \cs_new:Npn \__str_change_case_output:nw #1#2 \__str_change_case_result:n #3
5444 { #2 \__str_change_case_result:n { #3 #1 } }
5445 \cs_generate_variant:Nn \__str_change_case_output:nw { f }
5446 \cs_new:Npn \__str_change_case_end:wn #1 \__str_change_case_result:n #2
5447 { \tl_to_str:n {#2} }
5448 \cs_new:Npn \__str_change_case_loop:nw #1#2 \q__str_recursion_stop
5449 {
5450     \tl_if_head_is_space:nTF {#2}
5451     { \__str_change_case_space:n }
5452     { \__str_change_case_char:nN
5453       {#1} #2 \q__str_recursion_stop
5454     }
5455 \exp_last_unbraced:NNNNNo
5456 \cs_new:Npn \__str_change_case_space:n #1 \c_space_tl
5457 {
5458     \__str_change_case_output:nw { ~ }
5459     \__str_change_case_loop:nw {#1}
5460 }
5461 \cs_new:Npn \__str_change_case_char:nN #1#2
5462 {
5463     \__str_if_recursion_tail_stop_do:Nn #2
5464     { \__str_change_case_end:wn }
5465     \__str_change_case_codepoint:nN {#1} #2
5466 }
5467 \if_int_compare:w 0
5468 \cs_if_exist:NT \tex_XeTeXversion:D { 1 }
5469 \cs_if_exist:NT \tex luatexversion:D { 1 }
5470 > 0 \exp_stop_f:
5471 \cs_new:Npn \__str_change_case_codepoint:nN #1#2
5472 { \__str_change_case_char:fnn { \int_eval:n {`#2} } {#1} {#2} }
5473 \else:
5474 \cs_new:Npe \__str_change_case_codepoint:nN #1#2

```

```

5475 { 5475
5476 \exp_not:N \int_compare:nNnTF {\`#2} > { "80 } 5476
5477 { 5477
5478 \cs_if_exist:NTF \tex_pdfversion:D 5478
5479 { \exp_not:N \__str_change_case_char_auxi:nN } 5479
5480 { 5480
5481 \exp_not:N \int_compare:nNnTF {\`#2} > { "FF } 5481
5482 { \exp_not:N \__str_change_case_char_auxii:nN } 5482
5483 { \exp_not:N \__str_change_case_char_auxi:nN } 5483
5484 } 5484
5485 } 5485
5486 { \exp_not:N \__str_change_case_char_auxii:nN } 5486
5487 {\#1} #2 5487
5488 } 5488
5489 \cs_new:Npn \__str_change_case_char_auxi:nN #1#2 5489
5490 { 5490
5491 \int_compare:nNnTF {\`#2} < { "E0 } 5491
5492 { \__str_change_case_codepoint:nNN } 5492
5493 { 5493
5494 \int_compare:nNnTF {\`#2} < { "F0 } 5494
5495 { \__str_change_case_codepoint:nNNN } 5495
5496 { \__str_change_case_codepoint:nNNNNN } 5496
5497 } 5497
5498 {\#1} #2 5498
5499 } 5499
5500 \cs_new:Npn \__str_change_case_char_auxii:nN #1#2 5500
5501 { \__str_change_case_char:fnn { \int_eval:n {\`#2} } {\#1} {\#2} } 5501
5502 \cs_new:Npn \__str_change_case_codepoint:nNN #1#2#3 5502
5503 { 5503
5504 \__str_change_case_char:fnn 5504
5505 { \int_eval:n { (\`#2 - "C0) * "40 + \`#3 - "80 } } 5505
5506 {\#1} {\#2#3} 5506
5507 } 5507
5508 \cs_new:Npn \__str_change_case_codepoint:nNNN #1#2#3#4 5508
5509 { 5509
5510 \__str_change_case_char:fnn 5510
5511 { 5511
5512 \int_eval:n 5512
5513 { (\`#2 - "E0) * "1000 + (\`#3 - "80) * "40 + \`#4 - "80 } 5513
5514 } 5514
5515 {\#1} {\#2#3#4} 5515
5516 } 5516
5517 \cs_new:Npn \__str_change_case_codepoint:nNNNN #1#2#3#4#5 5517
5518 { 5518
5519 \__str_change_case_char:fnn 5519
5520 { 5520

```



```

5521         \int_eval:n
5522     {
5523         (`#2 - "F0) * "40000
5524         + (`#3 - "80) * "1000
5525         + (`#4 - "80) * "40
5526         + `#5 - "80
5527     }
5528 }
5529 {#1} {#2#3#4#5}
5530 }
5531 \fi:
5532 \cs_new:Npn \__str_change_case_char:nnn #1#2#3
5533 {
5534     \__str_change_case_output:fw
5535     {
5536         \exp_args:Ne \__str_change_case_char_aux:nnn
5537             { \__kernel_codepoint_case:nn {#2} {#1} } {#1} {#3}
5538     }
5539     \__str_change_case_loop:nw {#2}
5540 }
5541 \cs_generate_variant:Nn \__str_change_case_char:nnn { f }
5542 \cs_new:Npn \__str_change_case_char_aux:nnn #1#2#3
5543 {
5544     \use:e { \__str_change_case_char:nnnnn #1 {#2} {#3} }
5545 }
5546 \cs_new:Npn \__str_change_case_char:nnnnn #1#2#3#4#5
5547 {
5548     \int_compare:nNnTF {#1} = {#4}
5549     { \tl_to_str:n {#5} }
5550     {
5551         \codepoint_str_generate:n {#1}
5552         \tl_if_blank:nF {#2}
5553         {
5554             \codepoint_str_generate:n {#2}
5555             \tl_if_blank:nF {#3}
5556             { \codepoint_str_generate:n {#3} }
5557         }
5558     }
5559 }
5560 \cs_new:Npn \str_mdfive_hash:n #1 { \tex_mdffivesum:D { \tl_to_str:n {#1} } }
5561 \cs_new:Npn \str_mdfive_hash:e #1 { \tex_mdffivesum:D {#1} }
5562 \str_const:Ne \c_ampersand_str { \cs_to_str:N \& }
5563 \str_const:Ne \c_atsign_str { \cs_to_str:N \@ }
5564 \str_const:Ne \c_backslash_str { \cs_to_str:N \ }
5565 \str_const:Ne \c_left_brace_str { \cs_to_str:N \{ }
5566 \str_const:Ne \c_right_brace_str { \cs_to_str:N \} }

```

```
5567 \str_const:Ne \c_circumflex_str { \cs_to_str:N \^ } 5567
5568 \str_const:Ne \c_colon_str { \cs_to_str:N \: } 5568
5569 \str_const:Ne \c_dollar_str { \cs_to_str:N \$ } 5569
5570 \str_const:Ne \c_hash_str { \cs_to_str:N \# } 5570
5571 \str_const:Ne \c_percent_str { \cs_to_str:N \% } 5571
5572 \str_const:Ne \c_tilde_str { \cs_to_str:N \~ } 5572
5573 \str_const:Ne \c_underscore_str { \cs_to_str:N \_ } 5573
5574 \str_const:Ne \c_zero_str { 0 } 5574
5575 \cs_new_eq:NN \c_empty_str \c_empty_tl 5575
5576 \str_new:N \l_tmpa_str 5576
5577 \str_new:N \l_tmpb_str 5577
5578 \str_new:N \g_tmpa_str 5578
5579 \str_new:N \g_tmpb_str 5579
5580 \cs_new_eq:NN \str_show:n \tl_show:n 5580
5581 \cs_new_protected:Npn \str_show:N #1 5581
5582 { 5582
5583     \__kernel_chk_tl_type:NnnT #1 { str } { \tl_to_str:N #1 } 5583
5584     { \tl_show:N #1 } 5584
5585 } 5585
5586 \cs_generate_variant:Nn \str_show:N { c } 5586
5587 \cs_new_eq:NN \str_log:n \tl_log:n 5587
5588 \cs_new_protected:Npn \str_log:N #1 5588
5589 { 5589
5590     \__kernel_chk_tl_type:NnnT #1 { str } { \tl_to_str:N #1 } 5590
5591     { \tl_log:N #1 } 5591
5592 } 5592
5593 \cs_generate_variant:Nn \str_log:N { c } 5593
5594 %% File: l3seq.dtx 5594
5595 \scan_new:N \s__seq 5595
5596 \scan_new:N \s__seq_mark 5596
5597 \scan_new:N \s__seq_stop 5597
5598 \cs_new:Npn \__seq_item:n 5598
5599 { 5599
5600     \msg_expandable_error:nn { seq } { misused } 5600
5601     \use_none:n 5601
5602 } 5602
5603 \tl_new:N \l__seq_internal_a_tl 5603
5604 \tl_new:N \l__seq_internal_b_tl 5604
5605 \cs_new_eq:NN \__seq_tmp:w ? 5605
5606 \tl_const:Nn \c_empty_seq { \s__seq } 5606
5607 \cs_new_protected:Npn \seq_new:N #1 5607
5608 { 5608
5609     \__kernel_chk_if_free_cs:N #1 5609
5610     \cs_gset_eq:NN #1 \c_empty_seq 5610
5611 } 5611
5612 \cs_generate_variant:Nn \seq_new:N { c } 5612
```

5613	\cs_new_protected:Npn \seq_clear:N #1	5613
5614	{ \seq_set_eq:NN #1 \c_empty_seq }	5614
5615	\cs_generate_variant:Nn \seq_clear:N { c }	5615
5616	\cs_new_protected:Npn \seq_gclear:N #1	5616
5617	{ \seq_gset_eq:NN #1 \c_empty_seq }	5617
5618	\cs_generate_variant:Nn \seq_gclear:N { c }	5618
5619	\cs_new_protected:Npn \seq_clear_new:N #1	5619
5620	{ \seq_if_exist:NTF #1 { \seq_clear:N #1 } { \seq_new:N #1 } }	5620
5621	\cs_generate_variant:Nn \seq_clear_new:N { c }	5621
5622	\cs_new_protected:Npn \seq_gclear_new:N #1	5622
5623	{ \seq_if_exist:NTF #1 { \seq_gclear:N #1 } { \seq_new:N #1 } }	5623
5624	\cs_generate_variant:Nn \seq_gclear_new:N { c }	5624
5625	\cs_new_eq:NN \seq_set_eq:NN \tl_set_eq:NN	5625
5626	\cs_new_eq:NN \seq_set_eq:Nc \tl_set_eq:Nc	5626
5627	\cs_new_eq:NN \seq_set_eq:cN \tl_set_eq:cN	5627
5628	\cs_new_eq:NN \seq_set_eq:cc \tl_set_eq:cc	5628
5629	\cs_new_eq:NN \seq_gset_eq:NN \tl_gset_eq:NN	5629
5630	\cs_new_eq:NN \seq_gset_eq:Nc \tl_gset_eq:Nc	5630
5631	\cs_new_eq:NN \seq_gset_eq:cN \tl_gset_eq:cN	5631
5632	\cs_new_eq:NN \seq_gset_eq:cc \tl_gset_eq:cc	5632
5633	\cs_new_protected:Npn \seq_set_from_clist:NN #1#2	5633
5634	{	5634
5635	__kernel_tl_set:Nx #1	5635
5636	{ \s__seq \clist_map_function:NN #2 __seq_wrap_item:n }	5636
5637	}	5637
5638	\cs_new_protected:Npn \seq_set_from_clist:Nn #1#2	5638
5639	{	5639
5640	__kernel_tl_set:Nx #1	5640
5641	{ \s__seq \clist_map_function:nN {#2} __seq_wrap_item:n }	5641
5642	}	5642
5643	\cs_new_protected:Npn \seq_gset_from_clist:NN #1#2	5643
5644	{	5644
5645	__kernel_tl_gset:Nx #1	5645
5646	{ \s__seq \clist_map_function:NN #2 __seq_wrap_item:n }	5646
5647	}	5647
5648	\cs_new_protected:Npn \seq_gset_from_clist:Nn #1#2	5648
5649	{	5649
5650	__kernel_tl_gset:Nx #1	5650
5651	{ \s__seq \clist_map_function:nN {#2} __seq_wrap_item:n }	5651
5652	}	5652
5653	\cs_generate_variant:Nn \seq_set_from_clist:NN { Nc }	5653
5654	\cs_generate_variant:Nn \seq_set_from_clist:NN { c , cc }	5654
5655	\cs_generate_variant:Nn \seq_set_from_clist:Nn { c }	5655
5656	\cs_generate_variant:Nn \seq_gset_from_clist:NN { Nc }	5656
5657	\cs_generate_variant:Nn \seq_gset_from_clist:NN { c , cc }	5657
5658	\cs_generate_variant:Nn \seq_gset_from_clist:Nn { c }	5658

```

5659 \cs_new_protected:Npn \seq_const_from_clist:Nn #1#2 5659
5660 { 5660
5661     \tl_const:Nc #1 5661
5662     { \s__seq \clist_map_function:nN {#2} \__seq_wrap_item:n } 5662
5663 } 5663
5664 \cs_generate_variant:Nn \seq_const_from_clist:Nn { c } 5664
5665 \cs_new_protected:Npn \seq_set_split:Nnn 5665
5666 { \__seq_set_split:NNNnn \__kernel_tl_set:Nx \tl_trim_spaces:n } 5666
5667 \cs_new_protected:Npn \seq_gset_split:Nnn 5667
5668 { \__seq_set_split:NNNnn \__kernel_tl_gset:Nx \tl_trim_spaces:n } 5668
5669 \cs_new_protected:Npn \seq_set_split_keep_spaces:Nnn 5669
5670 { \__seq_set_split:NNNnn \__kernel_tl_set:Nx \exp_not:n } 5670
5671 \cs_new_protected:Npn \seq_gset_split_keep_spaces:Nnn 5671
5672 { \__seq_set_split:NNNnn \__kernel_tl_gset:Nx \exp_not:n } 5672
5673 \cs_new_protected:Npn \__seq_set_split:NNNnn #1#2#3#4#5 5673
5674 { 5674
5675     \tl_if_empty:nTF {#4} 5675
5676     { 5676
5677         \tl_set:Nn \l__seq_internal_a_tl 5677
5678         { \tl_map_function:nN {#5} \__seq_wrap_item:n } 5678
5679     } 5679
5680     { 5680
5681         \tl_set:Nn \l__seq_internal_a_tl 5681
5682         { 5682
5683             \__seq_set_split:Nw #2 \prg_do_nothing: 5683
5684             #5 5684
5685             \__seq_set_split_end: 5685
5686         } 5686
5687         \tl_replace_all:Nnn \l__seq_internal_a_tl {#4} 5687
5688         { 5688
5689             \__seq_set_split_end: 5689
5690             \__seq_set_split:Nw #2 \prg_do_nothing: 5690
5691         } 5691
5692         \__kernel_tl_set:Nx \l__seq_internal_a_tl { \l__seq_internal_a_tl } 5692
5693     } 5693
5694     #1 #3 { \s__seq \l__seq_internal_a_tl } 5694
5695 } 5695
5696 \cs_new:Npn \__seq_set_split:Nw #1#2 \__seq_set_split_end: 5696
5697 { 5697
5698     \exp_not:N \__seq_set_split:w 5698
5699     \exp_args:No #1 {#2} 5699
5700     \exp_not:N \__seq_set_split_end: 5700
5701 } 5701
5702 \cs_new:Npn \__seq_set_split:w #1 \__seq_set_split_end: 5702
5703 { \__seq_wrap_item:n {#1} } 5703
5704 \cs_generate_variant:Nn \seq_set_split:Nnn { NV , NnV , NVV , Nne , Nee } 5704

```

```
5705 \cs_generate_variant:Nn \seq_set_split:Nnn { Nnx , Nxx } 5705
5706 \cs_generate_variant:Nn \seq_gset_split:Nnn { NV , NnV , NVV , Nne , Nee } 5706
5707 \cs_generate_variant:Nn \seq_gset_split:Nnn { Nnx , Nxx } 5707
5708 \cs_generate_variant:Nn \seq_set_split_keep_spaces:Nnn { NnV } 5708
5709 \cs_generate_variant:Nn \seq_gset_split_keep_spaces:Nnn { NnV } 5709
5710 \cs_new_protected:Npn \seq_set_filter:NNn 5710
5711 { \__seq_set_filter:NNNn \__kernel_tl_set:Nx } 5711
5712 \cs_new_protected:Npn \seq_gset_filter:NNn 5712
5713 { \__seq_set_filter:NNNn \__kernel_tl_gset:Nx } 5713
5714 \cs_new_protected:Npn \__seq_set_filter:NNNn #1#2#3#4 5714
5715 { 5715
5716 \__seq_push_item_def:n { \bool_if:nT {#4} { \__seq_wrap_item:n {##1} } } 5716
5717 #1 #2 { #3 } 5717
5718 \__seq_pop_item_def: 5718
5719 } 5719
5720 \cs_new_protected:Npn \seq_set_regex_extract_once:Nnn #1#2#3 5720
5721 { \regex_extract_once:nnN {#2} {#3} #1 } 5721
5722 \cs_generate_variant:Nn \seq_set_regex_extract_once:Nnn { c } 5722
5723 \cs_new_protected:Npn \seq_set_regex_extract_once:Nnn #1#2#3 5723
5724 { \regex_extract_once:NnN #2 {#3} #1 } 5724
5725 \cs_generate_variant:Nn \seq_set_regex_extract_once:Nnn { c } 5725
5726 \cs_new_protected:Npn \seq_set_regex_extract_all:Nnn #1#2#3 5726
5727 { \regex_extract_all:nnN {#2} {#3} #1 } 5727
5728 \cs_generate_variant:Nn \seq_set_regex_extract_all:Nnn { c } 5728
5729 \cs_new_protected:Npn \seq_set_regex_extract_all:Nnn #1#2#3 5729
5730 { \regex_extract_all:NnN #2 {#3} #1 } 5730
5731 \cs_generate_variant:Nn \seq_set_regex_extract_all:Nnn { c } 5731
5732 \cs_new_protected:Npn \seq_set_regex_split:Nnn #1#2#3 5732
5733 { \regex_split:nnN {#2} {#3} #1 } 5733
5734 \cs_generate_variant:Nn \seq_set_regex_split:Nnn { c } 5734
5735 \cs_new_protected:Npn \seq_set_regex_split:Nnn #1#2#3 5735
5736 { \regex_split:NnN #2 {#3} #1 } 5736
5737 \cs_generate_variant:Nn \seq_set_regex_split:Nnn { c } 5737
5738 \group_begin: 5738
5739 \cs_set_protected:Npn \__seq_tmp:w #1#2#3 5739
5740 { 5740
5741 \cs_new_protected:cpe { seq_gset_regex_ #1 :N #2 n } ##1##2##3 5741
5742 { 5742
5743 \group_begin: 5743
5744 \seq_set_eq:NN \exp_not:N \l__seq_tmp_seq ##1 5744
5745 \exp_not:c { regex_ #1 :Nn #2 } 5745
5746 #3 {##2} {##3} \exp_not:N \l__seq_tmp_seq 5746
5747 \seq_gset_eq:NN ##1 \exp_not:N \l__seq_tmp_seq 5747
5748 \group_end: 5748
5749 } 5749
5750 \cs_generate_variant:cn { seq_gset_regex_ #1 :N #2 n } { c } 5750
```

```

5751 }
5752 \__seq_tmp:w { extract_once } n { }
5753 \__seq_tmp:w { extract_once } N \use:n
5754 \__seq_tmp:w { extract_all } n { }
5755 \__seq_tmp:w { extract_all } N \use:n
5756 \__seq_tmp:w { split } n { }
5757 \__seq_tmp:w { split } N \use:n
5758 \group_end:
5759 \cs_new_protected:Npn \seq_concat:NNN #1#2#3
5760 { \tl_set:Nf #1 { \exp_after:wN \use_i:nn \exp_after:wN #2 #3 } }
5761 \cs_new_protected:Npn \seq_gconcat:NNN #1#2#3
5762 { \tl_gset:Nf #1 { \exp_after:wN \use_i:nn \exp_after:wN #2 #3 } }
5763 \cs_generate_variant:Nn \seq_concat:NNN { ccc }
5764 \cs_generate_variant:Nn \seq_gconcat:NNN { ccc }
5765 \prg_new_eq_conditional:NNn \seq_if_exist:N \cs_if_exist:N
5766 { TF , T , F , p }
5767 \prg_new_eq_conditional:NNn \seq_if_exist:c \cs_if_exist:c
5768 { TF , T , F , p }
5769 \cs_new_protected:Npn \seq_put_left:Nn #1#2
5770 {
5771 \__kernel_tl_set:Nx #1
5772 {
5773 \exp_not:n { \s__seq \__seq_item:n {#2} }
5774 \exp_not:f { \exp_after:wN \__seq_put_left_aux:w #1 }
5775 }
5776 }
5777 \cs_new_protected:Npn \seq_gput_left:Nn #1#2
5778 {
5779 \__kernel_tl_gset:Nx #1
5780 {
5781 \exp_not:n { \s__seq \__seq_item:n {#2} }
5782 \exp_not:f { \exp_after:wN \__seq_put_left_aux:w #1 }
5783 }
5784 }
5785 \cs_new:Npn \__seq_put_left_aux:w \s__seq { \exp_stop_f: }
5786 \cs_generate_variant:Nn \seq_put_left:Nn { NV , Nv , Ne , No , Nx }
5787 \cs_generate_variant:Nn \seq_put_left:Nn { c , cV , cv , ce , co , cx }
5788 \cs_generate_variant:Nn \seq_gput_left:Nn { NV , Nv , Ne , No , Nx }
5789 \cs_generate_variant:Nn \seq_gput_left:Nn { c , cV , cv , ce , co , cx }
5790 \cs_new_protected:Npn \seq_put_right:Nn #1#2
5791 { \tl_put_right:Nn #1 { \__seq_item:n {#2} } }
5792 \cs_new_protected:Npn \seq_gput_right:Nn #1#2
5793 { \tl_gput_right:Nn #1 { \__seq_item:n {#2} } }
5794 \cs_generate_variant:Nn \seq_put_right:Nn { NV , Nv , Ne , No , Nx }
5795 \cs_generate_variant:Nn \seq_put_right:Nn { c , cV , cv , ce , co , cx }
5796 \cs_generate_variant:Nn \seq_gput_right:Nn { NV , Nv , Ne , No , Nx }

```



```
5797 \cs_generate_variant:Nn \seq_gput_right:Nn { c , cV , cv , ce , co , cx } 5797
5798 \cs_new:Npn \__seq_wrap_item:n #1 { \exp_not:n { \__seq_item:n {#1} } } 5798
5799 \seq_new:N \l__seq_tmp_seq 5799
5800 \cs_new_protected:Npn \seq_remove_duplicates:N 5800
5801 { \__seq_remove_duplicates:NN \seq_set_eq:NN } 5801
5802 \cs_new_protected:Npn \seq_gremove_duplicates:N 5802
5803 { \__seq_remove_duplicates:NN \seq_gset_eq:NN } 5803
5804 \cs_new_protected:Npn \__seq_remove_duplicates:NN #1#2 5804
5805 { 5805
5806 \seq_clear:N \l__seq_tmp_seq 5806
5807 \seq_map_inline:Nn #2 5807
5808 { 5808
5809 \seq_if_in:NnF \l__seq_tmp_seq {##1} 5809
5810 { \seq_put_right:Nn \l__seq_tmp_seq {##1} } 5810
5811 } 5811
5812 #1 #2 \l__seq_tmp_seq 5812
5813 } 5813
5814 \cs_generate_variant:Nn \seq_remove_duplicates:N { c } 5814
5815 \cs_generate_variant:Nn \seq_gremove_duplicates:N { c } 5815
5816 \cs_new_protected:Npn \seq_remove_all:Nn 5816
5817 { \__seq_remove_all_aux:NNn \__kernel_tl_set:Nx } 5817
5818 \cs_new_protected:Npn \seq_gremove_all:Nn 5818
5819 { \__seq_remove_all_aux:NNn \__kernel_tl_gset:Nx } 5819
5820 \cs_new_protected:Npn \__seq_remove_all_aux:NNn #1#2#3 5820
5821 { 5821
5822 \__seq_push_item_def:n 5822
5823 { 5823
5824 \str_if_eq:nnT {##1} {#3} 5824
5825 { 5825
5826 \if_false: { \fi: } 5826
5827 \tl_set:Nn \l__seq_internal_b_tl {##1} 5827
5828 #1 #2 5828
5829 { \if_false: } \fi: 5829
5830 \exp_not:o {#2} 5830
5831 \tl_if_eq:NNT \l__seq_internal_a_tl \l__seq_internal_b_tl 5831
5832 { \use_none:nn } 5832
5833 } 5833
5834 \__seq_wrap_item:n {##1} 5834
5835 } 5835
5836 \tl_set:Nn \l__seq_internal_a_tl {#3} 5836
5837 #1 #2 {#2} 5837
5838 \__seq_pop_item_def: 5838
5839 } 5839
5840 \cs_generate_variant:Nn \seq_remove_all:Nn { NV , Ne , c , cV , ce } 5840
5841 \cs_generate_variant:Nn \seq_remove_all:Nn { Nx , cx } 5841
5842 \cs_generate_variant:Nn \seq_gremove_all:Nn { NV , Ne , c , cV , ce } 5842
```

```
5843 \cs_generate_variant:Nn \seq_gremove_all:Nn { Nx , cx } 5843
5844 \cs_new_eq:NN \__seq_int_eval:w \tex_numexpr:D 5844
5845 \cs_new_protected:Npn \seq_set_item:Nnn #1#2#3 5845
5846 { \__seq_set_item:NnnNN #1 {#2} {#3} \__kernel_tl_set:Nx \use_i:nn } 5846
5847 \cs_new_protected:Npn \seq_gset_item:Nnn #1#2#3 5847
5848 { \__seq_set_item:NnnNN #1 {#2} {#3} \__kernel_tl_gset:Nx \use_i:nn } 5848
5849 \cs_generate_variant:Nn \seq_set_item:Nnn { c } 5849
5850 \cs_generate_variant:Nn \seq_gset_item:Nnn { c } 5850
5851 \prg_new_protected_conditional:Npnn \seq_set_item:Nnn #1#2#3 { TF , T , F } 5851
5852 { \__seq_set_item:NnnNN #1 {#2} {#3} \__kernel_tl_set:Nx \use_ii:nn } 5852
5853 \prg_new_protected_conditional:Npnn \seq_gset_item:Nnn #1#2#3 { TF , T , F } 5853
5854 { \__seq_set_item:NnnNN #1 {#2} {#3} \__kernel_tl_gset:Nx \use_ii:nn } 5854
5855 \prg_generate_conditional_variant:Nnn \seq_set_item:Nnn { c } { TF , T , F } 5855
5856 \prg_generate_conditional_variant:Nnn \seq_gset_item:Nnn { c } { TF , T , F } 5856
5857 \cs_new_protected:Npn \__seq_set_item:NnnNN #1#2#3 5857
5858 { 5858
5859 \tl_set:Nn \l__seq_internal_a_tl { \__seq_item:n {#3} } 5859
5860 \exp_args:Nff \__seq_set_item:nnNNNN 5860
5861 { \int_eval:n {#2} } { \seq_count:N #1 } #1 \use_none:nn 5861
5862 } 5862
5863 \cs_new_protected:Npn \__seq_set_item:nnNNNN #1#2 5863
5864 { 5864
5865 \int_compare:nNnTF {#1} > 0 5865
5866 { \int_compare:nNnF {#1} > {#2} { \__seq_set_item:nNnnNNNN { #1 - 1 } } } 5866
5867 { 5867
5868 \int_compare:nNnF {#1} < {-#2} 5868
5869 { 5869
5870 \int_compare:nNnF {#1} = 0 5870
5871 { \__seq_set_item:nNnnNNNN { #2 + #1 } } 5871
5872 } 5872
5873 } 5873
5874 \__seq_set_item_false:nnNNNN {#1} {#2} 5874
5875 } 5875
5876 \cs_new_protected:Npn \__seq_set_item_false:nnNNNN #1#2#3#4#5#6 5876
5877 { 5877
5878 #6 5878
5879 { 5879
5880 \msg_error:nneee { seq } { item-too-large } 5880
5881 { \token_to_str:N #3 } {#2} {#1} 5881
5882 } 5882
5883 { \prg_return_false: } 5883
5884 } 5884
5885 \cs_new_protected:Npn \__seq_set_item:nNnnNNNN #1#2#3#4#5#6#7#8 5885
5886 { 5886
5887 #7 #5 5887
5888 { 5888
```

```

5889         \s__seq                                     5889
5890         \exp_after:wN \__seq_set_item:wn           5890
5891         \int_value:w \__seq_int_eval:w #1          5891
5892         #5 \s__seq_stop #6                          5892
5893     }                                                5893
5894     #8 { } { \prg_return_true: }                    5894
5895 }                                                    5895
5896 \cs_new:Npn \__seq_set_item:wn #1 \__seq_item:n #2 5896
5897 {                                                    5897
5898     \if_meaning:w 0 #1 \__seq_set_item_end:w \fi:  5898
5899     \exp_not:n { \__seq_item:n {#2} }              5899
5900     \exp_after:wN \__seq_set_item:wn               5900
5901     \int_value:w \__seq_int_eval:w #1 - 1 \s__seq  5901
5902 }                                                    5902
5903 \cs_new:Npn \__seq_set_item_end:w #1 \exp_not:n #2 #3 \s__seq #4 \s__seq_stop #5 5903
5904 {                                                    5904
5905     #1                                               5905
5906     \exp_not:o \l__seq_internal_a_tl               5906
5907     \exp_not:n {#4}                                5907
5908     #5 #2                                           5908
5909 }                                                    5909
5910 \cs_new_protected:Npn \seq_reverse:N                5910
5911 { \__seq_reverse:NN \__kernel_tl_set:Nx }          5911
5912 \cs_new_protected:Npn \seq_greverse:N               5912
5913 { \__seq_reverse:NN \__kernel_tl_gset:Nx }         5913
5914 \cs_new_protected:Npn \__seq_reverse:NN #1 #2       5914
5915 {                                                    5915
5916     \cs_set_eq:NN \__seq_tmp:w \__seq_item:n       5916
5917     \cs_set_eq:NN \__seq_item:n \__seq_reverse_item:nwn 5917
5918     #1 #2 { #2 \exp_not:n { } }                    5918
5919     \cs_set_eq:NN \__seq_item:n \__seq_tmp:w       5919
5920 }                                                    5920
5921 \cs_new:Npn \__seq_reverse_item:nwn #1 #2 \exp_not:n #3 5921
5922 {                                                    5922
5923     #2                                               5923
5924     \exp_not:n { \__seq_item:n {#1} #3 }           5924
5925 }                                                    5925
5926 \cs_generate_variant:Nn \seq_reverse:N { c }       5926
5927 \cs_generate_variant:Nn \seq_greverse:N { c }      5927
5928 \prg_new_conditional:Npnn \seq_if_empty:N #1 { p , T , F , TF } 5928
5929 {                                                    5929
5930     \if_meaning:w #1 \c_empty_seq                  5930
5931     \prg_return_true:                              5931
5932     \else:                                          5932
5933     \prg_return_false:                             5933
5934     \fi:                                            5934

```

```

5935 }
5936 \prg_generate_conditional_variant:Nnn \seq_if_empty:N
5937 { c } { p , T , F , TF }
5938 \seq_new:N \g__seq_internal_seq
5939 \cs_new_protected:Npn \seq_shuffle:N { \__seq_shuffle:NN \seq_set_eq:NN }
5940 \cs_new_protected:Npn \seq_gshuffle:N { \__seq_shuffle:NN \seq_gset_eq:NN }
5941 \cs_new_protected:Npn \__seq_shuffle:NN #1#2
5942 {
5943   \int_compare:nNnTF { \seq_count:N #2 } > \c_max_register_int
5944   {
5945     \msg_error:nne { seq } { shuffle-too-large }
5946     { \token_to_str:N #2 }
5947   }
5948   {
5949     \group_begin:
5950     \int_zero:N \l__seq_internal_a_int
5951     \__seq_push_item_def:
5952     \cs_gset_eq:NN \__seq_item:n \__seq_shuffle_item:n
5953     #2
5954     \__seq_pop_item_def:
5955     \seq_gclear:N \g__seq_internal_seq
5956     \int_step_inline:nn \l__seq_internal_a_int
5957     {
5958       \seq_gput_right:Ne \g__seq_internal_seq
5959       { \tex_the:D \tex_toks:D ##1 }
5960     }
5961     \group_end:
5962     #1 #2 \g__seq_internal_seq
5963     \seq_gclear:N \g__seq_internal_seq
5964   }
5965 }
5966 \cs_new_protected:Npn \__seq_shuffle_item:n
5967 {
5968   \int_incr:N \l__seq_internal_a_int
5969   \int_set:Nn \l__seq_internal_b_int
5970   { 1 + \tex_uniformdeviate:D \l__seq_internal_a_int }
5971   \tex_toks:D \l__seq_internal_a_int
5972   = \tex_toks:D \l__seq_internal_b_int
5973   \tex_toks:D \l__seq_internal_b_int
5974 }
5975 \cs_generate_variant:Nn \seq_shuffle:N { c }
5976 \cs_generate_variant:Nn \seq_gshuffle:N { c }
5977 \prg_new_protected_conditional:Npnn \seq_if_in:Nn #1#2
5978 { T , F , TF }
5979 {
5980   \group_begin:

```

```

5981      \tl_set:Nn \l__seq_internal_a_tl {#2}
5982      \cs_set_protected:Npn \__seq_item:n ##1
5983      {
5984          \tl_set:Nn \l__seq_internal_b_tl {##1}
5985          \if_meaning:w \l__seq_internal_a_tl \l__seq_internal_b_tl
5986              \exp_after:wN \__seq_if_in:
5987          \fi:
5988      }
5989      #1
5990  \group_end:
5991  \prg_return_false:
5992  \prg_break_point:
5993  }
5994  \cs_new:Npn \__seq_if_in:
5995      { \prg_break:n { \group_end: \prg_return_true: } }
5996  \prg_generate_conditional_variant:Nnn \seq_if_in:Nn
5997      { NV , Nv , Ne , No , Nx , c , cV , cv , ce , co , cx } { T , F , TF }
5998  \cs_new_protected:Npn \__seq_pop:NNNN #1#2#3#4
5999      {
6000          \if_meaning:w #3 \c_empty_seq
6001              \tl_set:Nn #4 { \q_no_value }
6002          \else:
6003              #1#2#3#4
6004          \fi:
6005      }
6006  \cs_new_protected:Npn \__seq_pop_TF:NNNN #1#2#3#4
6007      {
6008          \if_meaning:w #3 \c_empty_seq
6009              % \tl_set:Nn #4 { \q_no_value }
6010              \prg_return_false:
6011          \else:
6012              #1#2#3#4
6013              \prg_return_true:
6014          \fi:
6015      }
6016  \cs_new_protected:Npn \seq_get_left:NN #1#2
6017      {
6018          \__kernel_tl_set:Nx #2
6019          {
6020              \exp_after:wN \__seq_get_left:wnw
6021              #1 \__seq_item:n { \q_no_value } \s__seq_stop
6022          }
6023      }
6024  \cs_new:Npn \__seq_get_left:wnw #1 \__seq_item:n #2#3 \s__seq_stop
6025      { \exp_not:n {#2} }
6026  \cs_generate_variant:Nn \seq_get_left:NN { c }

```

6027	\cs_new_protected:Npn \seq_pop_left:NN	6027
6028	{ __seq_pop:NNNN __seq_pop_left:NNN \tl_set:Nn }	6028
6029	\cs_new_protected:Npn \seq_gpop_left:NN	6029
6030	{ __seq_pop:NNNN __seq_pop_left:NNN \tl_gset:Nn }	6030
6031	\cs_new_protected:Npn __seq_pop_left:NNN #1#2#3	6031
6032	{ \exp_after:wN __seq_pop_left:wnwNNN #2 \s__seq_stop #1#2#3 }	6032
6033	\cs_new_protected:Npn __seq_pop_left:wnwNNN	6033
6034	#1 __seq_item:n #2#3 \s__seq_stop #4#5#6	6034
6035	{	6035
6036	#4 #5 { #1 #3 }	6036
6037	\tl_set:Nn #6 {#2}	6037
6038	}	6038
6039	\cs_generate_variant:Nn \seq_pop_left:NN { c }	6039
6040	\cs_generate_variant:Nn \seq_gpop_left:NN { c }	6040
6041	\cs_new_protected:Npn \seq_get_right:NN #1#2	6041
6042	{	6042
6043	__kernel_tl_set:Nx #2	6043
6044	{	6044
6045	\exp_after:wN \use_i_ii:nnn	6045
6046	\exp_after:wN __seq_get_right_loop:nw	6046
6047	\exp_after:wN \q_no_value	6047
6048	#1	6048
6049	__seq_get_right_end:NnN __seq_item:n	6049
6050	}	6050
6051	}	6051
6052	\cs_new:Npn __seq_get_right_loop:nw #1#2 __seq_item:n	6052
6053	{	6053
6054	#2 \use_none:n {#1}	6054
6055	__seq_get_right_loop:nw	6055
6056	}	6056
6057	\cs_new:Npn __seq_get_right_end:NnN #1#2#3 { \exp_not:n {#2} }	6057
6058	\cs_generate_variant:Nn \seq_get_right:NN { c }	6058
6059	\cs_new_protected:Npn \seq_pop_right:NN	6059
6060	{ __seq_pop:NNNN __seq_pop_right:NNN __kernel_tl_set:Nx }	6060
6061	\cs_new_protected:Npn \seq_gpop_right:NN	6061
6062	{ __seq_pop:NNNN __seq_pop_right:NNN __kernel_tl_gset:Nx }	6062
6063	\cs_new_protected:Npn __seq_pop_right:NNN #1#2#3	6063
6064	{	6064
6065	\cs_set_eq:NN __seq_tmp:w __seq_item:n	6065
6066	\cs_set_eq:NN __seq_item:n \scan_stop:	6066
6067	#1 #2	6067
6068	{ \if_false: } \fi: \s__seq	6068
6069	\exp_after:wN \use_i:nnn	6069
6070	\exp_after:wN __seq_pop_right_loop:nn	6070
6071	#2	6071
6072	{	6072


```

6073         \if_false: { \fi: }
6074         \__kernel_tl_set:Nx #3
6075     }
6076     { } \use_none:nn
6077     \cs_set_eq:NN \__seq_item:n \__seq_tmp:w
6078 }
6079 \cs_new:Npn \__seq_pop_right_loop:nn #1#2
6080 {
6081     #2 { \exp_not:n {#1} }
6082     \__seq_pop_right_loop:nn
6083 }
6084 \cs_generate_variant:Nn \seq_pop_right:NN { c }
6085 \cs_generate_variant:Nn \seq_gpop_right:NN { c }
6086 \prg_new_protected_conditional:Npnn \seq_get_left:NN #1#2 { T , F , TF }
6087 { \__seq_pop_TF:NNNN \prg_do_nothing: \seq_get_left:NN #1#2 }
6088 \prg_new_protected_conditional:Npnn \seq_get_right:NN #1#2 { T , F , TF }
6089 { \__seq_pop_TF:NNNN \prg_do_nothing: \seq_get_right:NN #1#2 }
6090 \prg_generate_conditional_variant:Nnn \seq_get_left:NN
6091 { c } { T , F , TF }
6092 \prg_generate_conditional_variant:Nnn \seq_get_right:NN
6093 { c } { T , F , TF }
6094 \prg_new_protected_conditional:Npnn \seq_pop_left:NN #1#2
6095 { T , F , TF }
6096 { \__seq_pop_TF:NNNN \__seq_pop_left:NNN \tl_set:Nn #1 #2 }
6097 \prg_new_protected_conditional:Npnn \seq_gpop_left:NN #1#2
6098 { T , F , TF }
6099 { \__seq_pop_TF:NNNN \__seq_pop_left:NNN \tl_gset:Nn #1 #2 }
6100 \prg_new_protected_conditional:Npnn \seq_pop_right:NN #1#2
6101 { T , F , TF }
6102 { \__seq_pop_TF:NNNN \__seq_pop_right:NNN \__kernel_tl_set:Nx #1 #2 }
6103 \prg_new_protected_conditional:Npnn \seq_gpop_right:NN #1#2
6104 { T , F , TF }
6105 { \__seq_pop_TF:NNNN \__seq_pop_right:NNN \__kernel_tl_gset:Nx #1 #2 }
6106 \prg_generate_conditional_variant:Nnn \seq_pop_left:NN { c }
6107 { T , F , TF }
6108 \prg_generate_conditional_variant:Nnn \seq_gpop_left:NN { c }
6109 { T , F , TF }
6110 \prg_generate_conditional_variant:Nnn \seq_pop_right:NN { c }
6111 { T , F , TF }
6112 \prg_generate_conditional_variant:Nnn \seq_gpop_right:NN { c }
6113 { T , F , TF }
6114 \cs_new:Npn \seq_item:Nn #1
6115 { \exp_after:wN \__seq_item:wNn #1 \s__seq_stop #1 }
6116 \cs_new:Npn \__seq_item:wNn \s__seq #1 \s__seq_stop #2#3
6117 {
6118     \exp_args:Nf \__seq_item:nwn

```

```
6119 { \exp_args:Nf \__seq_item:nN { \int_eval:n {#3} } #2 } 6119
6120 #1 6120
6121 \prg_break: \__seq_item:n { } 6121
6122 \prg_break_point: 6122
6123 } 6123
6124 \cs_new:Npn \__seq_item:nN #1#2 6124
6125 { 6125
6126   \int_compare:nNnTF {#1} < 0 6126
6127     { \int_eval:n { \seq_count:N #2 + 1 + #1 } } 6127
6128     {#1} 6128
6129   } 6129
6130 \cs_new:Npn \__seq_item:nwn #1#2 \__seq_item:n #3 6130
6131 { 6131
6132   #2 6132
6133   \int_compare:nNnTF {#1} = 1 6133
6134     { \prg_break:n { \exp_not:n {#3} } } 6134
6135     { \exp_args:Nf \__seq_item:nwn { \int_eval:n { #1 - 1 } } } 6135
6136   } 6136
6137 \cs_generate_variant:Nn \seq_item:Nn { NV , Ne , c , cV , ce } 6137
6138 \cs_new:Npn \seq_rand_item:N #1 6138
6139 { 6139
6140   \seq_if_empty:NF #1 6140
6141     { \seq_item:Nn #1 { \int_rand:nn { 1 } { \seq_count:N #1 } } } 6141
6142   } 6142
6143 \cs_generate_variant:Nn \seq_rand_item:N { c } 6143
6144 \cs_new:Npn \seq_map_break: 6144
6145   { \prg_map_break:Nn \seq_map_break: { } } 6145
6146 \cs_new:Npn \seq_map_break:n 6146
6147   { \prg_map_break:Nn \seq_map_break: } 6147
6148 \cs_new:Npn \seq_map_function:NN #1#2 6148
6149 { 6149
6150   \exp_after:wN \use_i_ii:nnn 6150
6151   \exp_after:wN \__seq_map_function:Nw 6151
6152   \exp_after:wN #2 6152
6153   #1 6153
6154   \prg_break: 6154
6155   \__seq_item:n { } \__seq_item:n { } \__seq_item:n { } \__seq_item:n { } 6155
6156   \prg_break_point: 6156
6157   \prg_break_point:Nn \seq_map_break: { } 6157
6158   } 6158
6159 \cs_new:Npn \__seq_map_function:Nw #1 6159
6160   #2 \__seq_item:n #3 6160
6161   #4 \__seq_item:n #5 6161
6162   #6 \__seq_item:n #7 6162
6163   #8 \__seq_item:n #9 6163
6164   { 6164
```

6165	#2 #1 {#3}	6165
6166	#4 #1 {#5}	6166
6167	#6 #1 {#7}	6167
6168	#8 #1 {#9}	6168
6169	__seq_map_function:Nw #1	6169
6170	}	6170
6171	\cs_generate_variant:Nn \seq_map_function:NN { c }	6171
6172	\cs_new_protected:Npn __seq_push_item_def:n	6172
6173	{	6173
6174	__seq_push_item_def:	6174
6175	\cs_gset:Npn __seq_item:n ##1	6175
6176	}	6176
6177	\cs_new_protected:Npn __seq_push_item_def:e	6177
6178	{	6178
6179	__seq_push_item_def:	6179
6180	\cs_gset:Npe __seq_item:n ##1	6180
6181	}	6181
6182	\cs_new_protected:Npn __seq_push_item_def:	6182
6183	{	6183
6184	\int_gincr:N \g__kernel_prg_map_int	6184
6185	\cs_gset_eq:cN { __seq_map_ \int_use:N \g__kernel_prg_map_int :w }	6185
6186	__seq_item:n	6186
6187	}	6187
6188	\cs_new_protected:Npn __seq_pop_item_def:	6188
6189	{	6189
6190	\cs_gset_eq:Nc __seq_item:n	6190
6191	{ __seq_map_ \int_use:N \g__kernel_prg_map_int :w }	6191
6192	\int_gdecr:N \g__kernel_prg_map_int	6192
6193	}	6193
6194	\cs_new_protected:Npn \seq_map_inline:Nn #1#2	6194
6195	{	6195
6196	__seq_push_item_def:n {#2}	6196
6197	#1	6197
6198	\prg_break_point:Nn \seq_map_break: { __seq_pop_item_def: }	6198
6199	}	6199
6200	\cs_generate_variant:Nn \seq_map_inline:Nn { c }	6200
6201	\cs_new:Npn \seq_map_tokens:Nn #1#2	6201
6202	{	6202
6203	\exp_last_unbraced:Nno	6203
6204	\use_i:nn { __seq_map_tokens:nw {#2} } #1	6204
6205	\prg_break:	6205
6206	__seq_item:n { } __seq_item:n { } __seq_item:n { } __seq_item:n { }	6206
6207	\prg_break_point:	6207
6208	\prg_break_point:Nn \seq_map_break: { }	6208
6209	}	6209
6210	\cs_generate_variant:Nn \seq_map_tokens:Nn { c }	6210

6211	\cs_new:Npn __seq_map_tokens:nw #1	6211
6212	#2 __seq_item:n #3	6212
6213	#4 __seq_item:n #5	6213
6214	#6 __seq_item:n #7	6214
6215	#8 __seq_item:n #9	6215
6216	{	6216
6217	#2 \use:n {#1} {#3}	6217
6218	#4 \use:n {#1} {#5}	6218
6219	#6 \use:n {#1} {#7}	6219
6220	#8 \use:n {#1} {#9}	6220
6221	__seq_map_tokens:nw {#1}	6221
6222	}	6222
6223	\cs_new_protected:Npn \seq_map_variable:NNn #1#2#3	6223
6224	{	6224
6225	__seq_push_item_def:e	6225
6226	{	6226
6227	\tl_set:Nn \exp_not:N #2 {##1}	6227
6228	\exp_not:n {#3}	6228
6229	}	6229
6230	#1	6230
6231	\prg_break_point:Nn \seq_map_break: { __seq_pop_item_def: }	6231
6232	}	6232
6233	\cs_generate_variant:Nn \seq_map_variable:NNn { Nc }	6233
6234	\cs_generate_variant:Nn \seq_map_variable:NNn { c , cc }	6234
6235	\cs_new_eq:NN __seq_sep: __kernel_int_sep:	6235
6236	\cs_new:Npn \seq_map_indexed_function:NN #1#2	6236
6237	{	6237
6238	__seq_map_indexed:NN #1#2	6238
6239	\prg_break_point:Nn \seq_map_break: { }	6239
6240	}	6240
6241	\cs_new_protected:Npn \seq_map_indexed_inline:Nn #1#2	6241
6242	{	6242
6243	\int_gincr:N \g__kernel_prg_map_int	6243
6244	\cs_gset_protected:cpn	6244
6245	{ __seq_map_ \int_use:N \g__kernel_prg_map_int :w } ##1##2 {#2}	6245
6246	\exp_args:NNc __seq_map_indexed:NN #1	6246
6247	{ __seq_map_ \int_use:N \g__kernel_prg_map_int :w }	6247
6248	\prg_break_point:Nn \seq_map_break:	6248
6249	{ \int_gdecr:N \g__kernel_prg_map_int }	6249
6250	}	6250
6251	\cs_new:Npn __seq_map_indexed:NN #1#2	6251
6252	{	6252
6253	\exp_after:wN __seq_map_indexed:Nw	6253
6254	\exp_after:wN #2	6254
6255	\int_value:w 1	6255
6256	\exp_after:wN \use_i:nn	6256

6257	\exp_after:wN __seq_sep:	6257
6258	#1	6258
6259	\prg_break: __seq_item:n { } \prg_break_point:	6259
6260	}	6260
6261	\cs_new:Npn __seq_map_indexed:Nw #1#2 __seq_sep: #3 __seq_item:n #4	6261
6262	{	6262
6263	#3	6263
6264	#1 {#2} {#4}	6264
6265	\exp_after:wN __seq_map_indexed:Nw	6265
6266	\exp_after:wN #1	6266
6267	\int_value:w \int_eval:w 1 + #2 __seq_sep:	6267
6268	}	6268
6269	\cs_new:Npn \seq_map_pairwise_function:NNN #1#2#3	6269
6270	{ \exp_after:wN __seq_map_pairwise_function:wNN #2 \s__seq_stop #1 #3 }	6270
6271	\cs_new:Npn __seq_map_pairwise_function:wNN \s__seq #1 \s__seq_stop #2#3	6271
6272	{	6272
6273	\exp_after:wN __seq_map_pairwise_function:wNw #2 \s__seq_stop #3	6273
6274	#1 { ? \prg_break: } { }	6274
6275	\prg_break_point:	6275
6276	\prg_break_point:Nn \seq_map_break: { }	6276
6277	}	6277
6278	\cs_new:Npn __seq_map_pairwise_function:wNw \s__seq #1 \s__seq_stop #2	6278
6279	{	6279
6280	__seq_map_pairwise_function:Nnnwnn #2	6280
6281	#1 { ? \prg_break: } { }	6281
6282	\s__seq_stop	6282
6283	}	6283
6284	\cs_new:Npn __seq_map_pairwise_function:Nnnwnn #1#2#3#4 \s__seq_stop #5#6	6284
6285	{	6285
6286	\use_none:n #2	6286
6287	\use_none:n #5	6287
6288	#1 {#3} {#6}	6288
6289	__seq_map_pairwise_function:Nnnwnn #1 #4 \s__seq_stop	6289
6290	}	6290
6291	\cs_generate_variant:Nn \seq_map_pairwise_function:NNN { Nc , c , cc }	6291
6292	\cs_new_protected:Npn \seq_set_map_e:NNn	6292
6293	{ __seq_set_map_e:NNNn __kernel_tl_set:Nx }	6293
6294	\cs_new_protected:Npn \seq_gset_map_e:NNn	6294
6295	{ __seq_set_map_e:NNNn __kernel_tl_gset:Nx }	6295
6296	\cs_new_protected:Npn __seq_set_map_e:NNNn #1#2#3#4	6296
6297	{	6297
6298	__seq_push_item_def:n { \exp_not:N __seq_item:n {#4} }	6298
6299	#1 #2 { #3 }	6299
6300	__seq_pop_item_def:	6300
6301	}	6301
6302	\cs_new_protected:Npn \seq_set_map:NNn	6302

```

6303 { \__seq_set_map:NNNn \__kernel_tl_set:Nx }
6304 \cs_new_protected:Npn \seq_gset_map:NNn
6305 { \__seq_set_map:NNNn \__kernel_tl_gset:Nx }
6306 \cs_new_protected:Npn \__seq_set_map:NNNn #1#2#3#4
6307 {
6308     \__seq_push_item_def:n { \exp_not:n { \__seq_item:n {#4} } }
6309     #1 #2 { #3 }
6310     \__seq_pop_item_def:
6311 }
6312 \cs_new:Npn \seq_count:N #1
6313 {
6314     \int_eval:n
6315     {
6316         \exp_after:wN \use_i:nn
6317         \exp_after:wN \__seq_count:w
6318         #1
6319         \__seq_count_end:w \__seq_item:n 7
6320         \__seq_count_end:w \__seq_item:n 6
6321         \__seq_count_end:w \__seq_item:n 5
6322         \__seq_count_end:w \__seq_item:n 4
6323         \__seq_count_end:w \__seq_item:n 3
6324         \__seq_count_end:w \__seq_item:n 2
6325         \__seq_count_end:w \__seq_item:n 1
6326         \__seq_count_end:w \__seq_item:n 0
6327         \prg_break_point:
6328     }
6329 }
6330 \cs_new:Npn \__seq_count:w
6331     #1 \__seq_item:n #2 \__seq_item:n #3 \__seq_item:n #4 \__seq_item:n
6332     #5 \__seq_item:n #6 \__seq_item:n #7 \__seq_item:n #8 #9 \__seq_item:n
6333     { #9 8 + \__seq_count:w }
6334 \cs_new:Npn \__seq_count_end:w 8 + \__seq_count:w #1#2 \prg_break_point: {#1}
6335 \cs_generate_variant:Nn \seq_count:N { c }
6336 \cs_new:Npn \seq_use:Nnnn #1#2#3#4
6337 {
6338     \seq_if_exist:NTF #1
6339     {
6340         \int_case:nnF { \seq_count:N #1 }
6341         {
6342             { 0 } { }
6343             { 1 } { \exp_after:wN \__seq_use:NNnNnn #1 ? { } { } }
6344             { 2 } { \exp_after:wN \__seq_use:NNnNnn #1 {#2} }
6345         }
6346         {
6347             \exp_after:wN \__seq_use_setup:w #1 \__seq_item:n
6348             \s_seq_mark { \__seq_use:nwwwnwn {#3} }

```


6349	\s__seq_mark { __seq_use:nwwn {#4} }	6349
6350	\s__seq_stop { }	6350
6351	}	6351
6352	}	6352
6353	{	6353
6354	\msg_expandable_error:nnn	6354
6355	{ kernel } { bad-variable } {#1}	6355
6356	}	6356
6357	}	6357
6358	\cs_generate_variant:Nn \seq_use:Nnnn { c }	6358
6359	\cs_new:Npn __seq_use:NNnNnn #1#2#3#4#5#6 { \exp_not:n { #3 #6 #5 } }	6359
6360	\cs_new:Npn __seq_use_setup:w \s__seq { __seq_use:nwwwwnwn { } }	6360
6361	\cs_new:Npn __seq_use:nwwwwnwn	6361
6362	#1 __seq_item:n #2 __seq_item:n #3 __seq_item:n #4#5	6362
6363	\s__seq_mark #6#7 \s__seq_stop #8	6363
6364	{	6364
6365	#6 __seq_item:n {#3} __seq_item:n {#4} #5	6365
6366	\s__seq_mark {#6} #7 \s__seq_stop { #8 #1 #2 }	6366
6367	}	6367
6368	\cs_new:Npn __seq_use:nwwn #1 __seq_item:n #2 #3 \s__seq_stop #4	6368
6369	{ \exp_not:n { #4 #1 #2 } }	6369
6370	\cs_new:Npn \seq_use:Nn #1#2	6370
6371	{ \seq_use:Nnnn #1 {#2} {#2} {#2} }	6371
6372	\cs_generate_variant:Nn \seq_use:Nn { c }	6372
6373	\cs_new_eq:NN \seq_push:Nn \seq_put_left:Nn	6373
6374	\cs_generate_variant:Nn \seq_push:Nn { NV , Nv , Ne , c , cV , cv , ce }	6374
6375	\cs_generate_variant:Nn \seq_push:Nn { No , Nx , co , cx }	6375
6376	\cs_new_eq:NN \seq_gpush:Nn \seq_gput_left:Nn	6376
6377	\cs_generate_variant:Nn \seq_gpush:Nn { NV , Nv , Ne , c , cV , cv , ce }	6377
6378	\cs_generate_variant:Nn \seq_gpush:Nn { No , Nx , co , cx }	6378
6379	\cs_new_eq:NN \seq_get:NN \seq_get_left:NN	6379
6380	\cs_new_eq:NN \seq_get:cN \seq_get_left:cN	6380
6381	\cs_new_eq:NN \seq_pop:NN \seq_pop_left:NN	6381
6382	\cs_new_eq:NN \seq_pop:cN \seq_pop_left:cN	6382
6383	\cs_new_eq:NN \seq_gpop:NN \seq_gpop_left:NN	6383
6384	\cs_new_eq:NN \seq_gpop:cN \seq_gpop_left:cN	6384
6385	\prg_new_eq_conditional:NNn \seq_get:NN \seq_get_left:NN { T , F , TF }	6385
6386	\prg_new_eq_conditional:NNn \seq_get:cN \seq_get_left:cN { T , F , TF }	6386
6387	\prg_new_eq_conditional:NNn \seq_pop:NN \seq_pop_left:NN { T , F , TF }	6387
6388	\prg_new_eq_conditional:NNn \seq_pop:cN \seq_pop_left:cN { T , F , TF }	6388
6389	\prg_new_eq_conditional:NNn \seq_gpop:NN \seq_gpop_left:NN { T , F , TF }	6389
6390	\prg_new_eq_conditional:NNn \seq_gpop:cN \seq_gpop_left:cN { T , F , TF }	6390
6391	\cs_new_protected:Npn \seq_show:N { __seq_show:NN \msg_show:nneeee }	6391
6392	\cs_generate_variant:Nn \seq_show:N { c }	6392
6393	\cs_new_protected:Npn \seq_log:N { __seq_show:NN \msg_log:nneeee }	6393
6394	\cs_generate_variant:Nn \seq_log:N { c }	6394

```

6395 \cs_new_protected:Npn \__seq_show:NN #1#2
6396 {
6397     \__kernel_chk_tl_type:NnnT #2 { seq }
6398     {
6399         \s__seq
6400         \exp_after:wN \use_i:nn \exp_after:wN \__seq_show_validate:nn #2
6401         \q_recursion_tail \q_recursion_tail \q_recursion_stop
6402     }
6403     {
6404         #1 { seq } { show }
6405         { \token_to_str:N #2 }
6406         { \seq_map_function:NN #2 \msg_show_item:n }
6407         { } { }
6408     }
6409 }
6410 \cs_new:Npn \__seq_show_validate:nn #1#2
6411 {
6412     \quark_if_recursion_tail_stop:n {#2}
6413     \__seq_wrap_item:n {#2}
6414     \__seq_show_validate:nn
6415 }
6416 \seq_new:N \l_tmpa_seq
6417 \seq_new:N \l_tmpb_seq
6418 \seq_new:N \g_tmpa_seq
6419 \seq_new:N \g_tmpb_seq
6420 %% File: l3int.dtx
6421 \cs_new_eq:NN \int_value:w \tex_number:D
6422 \cs_new_eq:NN \__int_eval:w \tex_numexpr:D
6423 \cs_new_eq:NN \__int_eval_end: \tex_relax:D
6424 \cs_new_eq:NN \if_int_odd:w \tex_ifodd:D
6425 \cs_new_eq:NN \if_case:w \tex_ifcase:D
6426 \scan_new:N \s__int_mark
6427 \scan_new:N \s__int_stop
6428 \cs_new:Npn \__int_use_none_delimit_by_s_stop:w #1 \s__int_stop { }
6429 \quark_new:N \q__int_recursion_tail
6430 \quark_new:N \q__int_recursion_stop
6431 \__kernel_quark_new_test:N \__int_if_recursion_tail_stop_do:Nn
6432 \__kernel_quark_new_test:N \__int_if_recursion_tail_stop:N
6433 \cs_new:Npn \int_eval:n #1
6434 { \tex_the:D \__int_eval:w #1 \__int_eval_end: }
6435 \cs_new:Npn \int_eval:w { \tex_the:D \__int_eval:w }
6436 \cs_new_eq:NN \__int_sep: \__kernel_int_sep:
6437 \cs_new:Npn \int_sign:n #1
6438 {
6439     \int_value:w \exp_after:wN \__int_sign:Nw
6440     \int_value:w \__int_eval:w #1 \__int_eval_end: \__int_sep:

```

6441	\exp_stop_f:	6441
6442	}	6442
6443	\cs_new:Npn __int_sign:Nw #1#2 __int_sep:	6443
6444	{	6444
6445	\if_meaning:w 0 #1	6445
6446	0	6446
6447	\else:	6447
6448	\if_meaning:w - #1 - \fi: 1	6448
6449	\fi:	6449
6450	}	6450
6451	\cs_new:Npn \int_abs:n #1	6451
6452	{	6452
6453	\int_value:w \exp_after:wN __int_abs:N	6453
6454	\int_value:w __int_eval:w #1 __int_eval_end:	6454
6455	\exp_stop_f:	6455
6456	}	6456
6457	\cs_new:Npn __int_abs:N #1	6457
6458	{ \if_meaning:w - #1 \else: \exp_after:wN #1 \fi: }	6458
6459	\cs_new:Npn \int_max:nn #1#2	6459
6460	{	6460
6461	\int_value:w \exp_after:wN __int_maxmin:wwN	6461
6462	\int_value:w __int_eval:w #1 \exp_after:wN __int_sep:	6462
6463	\int_value:w __int_eval:w #2 __int_sep:	6463
6464	>	6464
6465	\exp_stop_f:	6465
6466	}	6466
6467	\cs_new:Npn \int_min:nn #1#2	6467
6468	{	6468
6469	\int_value:w \exp_after:wN __int_maxmin:wwN	6469
6470	\int_value:w __int_eval:w #1 \exp_after:wN __int_sep:	6470
6471	\int_value:w __int_eval:w #2 __int_sep:	6471
6472	<	6472
6473	\exp_stop_f:	6473
6474	}	6474
6475	\cs_new:Npn __int_maxmin:wwN #1 __int_sep: #2 __int_sep: #3	6475
6476	{	6476
6477	\if_int_compare:w #1 #3 #2 ~	6477
6478	#1	6478
6479	\else:	6479
6480	#2	6480
6481	\fi:	6481
6482	}	6482
6483	\cs_new:Npn \int_div_truncate:nn #1#2	6483
6484	{	6484
6485	\int_value:w __int_eval:w	6485
6486	\exp_after:wN __int_div_truncate:NwNw	6486

```
6487 \int_value:w \__int_eval:w #1 \exp_after:wN \__int_sep: 6487
6488 \int_value:w \__int_eval:w #2 \__int_sep: 6488
6489 \__int_eval_end: 6489
6490 } 6490
6491 \cs_new:Npn \__int_div_truncate:NwNw #1#2 \__int_sep: #3#4 \__int_sep: 6491
6492 { 6492
6493 \if_meaning:w 0 #1 6493
6494 0 6494
6495 \else: 6495
6496 ( 6496
6497 #1#2 6497
6498 \if_meaning:w - #1 + \else: - \fi: 6498
6499 ( \if_meaning:w - #3 - \fi: #3#4 - 1 ) / 2 6499
6500 ) 6500
6501 \fi: 6501
6502 / #3#4 6502
6503 } 6503
6504 \cs_new:Npn \int_div_round:nn #1#2 6504
6505 { \int_value:w \__int_eval:w ( #1 ) / ( #2 ) \__int_eval_end: } 6505
6506 \cs_new:Npn \int_mod:nn #1#2 6506
6507 { 6507
6508 \int_value:w \__int_eval:w \exp_after:wN \__int_mod:ww 6508
6509 \int_value:w \__int_eval:w #1 \exp_after:wN \__int_sep: 6509
6510 \int_value:w \__int_eval:w #2 \__int_sep: 6510
6511 \__int_eval_end: 6511
6512 } 6512
6513 \cs_new:Npn \__int_mod:ww #1 \__int_sep: #2 \__int_sep: 6513
6514 { #1 - ( \__int_div_truncate:NwNw #1 \__int_sep: #2 \__int_sep: ) * #2 } 6514
6515 \cs_new:Npn \__kernel_int_add:nnn #1#2#3 6515
6516 { 6516
6517 \int_value:w \__int_eval:w #1 6517
6518 \if_int_compare:w #2 < \c_zero_int \exp_after:wN \reverse_if:N \fi: 6518
6519 \if_int_compare:w #1 < \c_zero_int + #2 + #3 \else: + #3 + #2 \fi: 6519
6520 \__int_eval_end: 6520
6521 } 6521
6522 \cs_new_protected:Npn \int_new:N #1 6522
6523 { 6523
6524 \__kernel_chk_if_free_cs:N #1 6524
6525 \cs:w newcount \cs_end: #1 6525
6526 } 6526
6527 \cs_generate_variant:Nn \int_new:N { c } 6527
6528 \cs_new_protected:Npn \int_const:Nn #1#2 6528
6529 { \__int_const:eN { \int_eval:n {#2} } #1 } 6529
6530 \cs_generate_variant:Nn \int_const:Nn { c } 6530
6531 \cs_new_protected:Npn \__int_const:nN #1#2 6531
6532 { 6532
```

6533	\int_compare:nNnTF {#1} < \c_zero_int	6533
6534	{	6534
6535	\int_new:N #2	6535
6536	\tex_global:D	6536
6537	}	6537
6538	{	6538
6539	\int_compare:nNnTF {#1} > \c__int_max_constdef_int	6539
6540	{	6540
6541	\int_new:N #2	6541
6542	\tex_global:D	6542
6543	}	6543
6544	{	6544
6545	__kernel_chk_if_free_cs:N #2	6545
6546	\tex_global:D __int_constdef:Nw	6546
6547	}	6547
6548	}	6548
6549	#2 = __int_eval:w #1 __int_eval_end:	6549
6550	}	6550
6551	\cs_generate_variant:Nn __int_const:nN { e }	6551
6552	\if_int_odd:w 0	6552
6553	\cs_if_exist:NT \tex_luatexversion:D { 1 }	6553
6554	\cs_if_exist:NT \tex_omathchardef:D { 1 }	6554
6555	\cs_if_exist:NT \tex_XeTeXversion:D { 1 } ~	6555
6556	\cs_if_exist:NTF \tex_omathchardef:D	6556
6557	{ \cs_new_eq:NN __int_constdef:Nw \tex_omathchardef:D }	6557
6558	{ \cs_new_eq:NN __int_constdef:Nw \tex_chardef:D }	6558
6559	\tex_global:D __int_constdef:Nw \c__int_max_constdef_int 1114111 ~	6559
6560	\else:	6560
6561	\cs_new_eq:NN __int_constdef:Nw \tex_mathchardef:D	6561
6562	\tex_global:D __int_constdef:Nw \c__int_max_constdef_int 32767 ~	6562
6563	\fi:	6563
6564	\cs_new_protected:Npn \int_zero:N #1 { #1 = \c_zero_int }	6564
6565	\cs_new_protected:Npn \int_gzero:N #1 { \tex_global:D #1 = \c_zero_int }	6565
6566	\cs_generate_variant:Nn \int_zero:N { c }	6566
6567	\cs_generate_variant:Nn \int_gzero:N { c }	6567
6568	\cs_new_protected:Npn \int_zero_new:N #1	6568
6569	{ \int_if_exist:NTF #1 { \int_zero:N #1 } { \int_new:N #1 } }	6569
6570	\cs_new_protected:Npn \int_gzero_new:N #1	6570
6571	{ \int_if_exist:NTF #1 { \int_gzero:N #1 } { \int_new:N #1 } }	6571
6572	\cs_generate_variant:Nn \int_zero_new:N { c }	6572
6573	\cs_generate_variant:Nn \int_gzero_new:N { c }	6573
6574	\cs_new_protected:Npn \int_set_eq:NN #1#2 { #1 = #2 }	6574
6575	\cs_generate_variant:Nn \int_set_eq:NN { c , Nc , cc }	6575
6576	\cs_new_protected:Npn \int_gset_eq:NN #1#2 { \tex_global:D #1 = #2 }	6576
6577	\cs_generate_variant:Nn \int_gset_eq:NN { c , Nc , cc }	6577
6578	\prg_new_eq_conditional:NNn \int_if_exist:N \cs_if_exist:N	6578

```

6579 { TF , T , F , p }
6580 \prg_new_eq_conditional:NNn \int_if_exist:c \cs_if_exist:c
6581 { TF , T , F , p }
6582 \cs_new_protected:Npn \int_add:Nn #1#2
6583 { \tex_advance:D #1 \__int_eval:w #2 \__int_eval_end: }
6584 \cs_new_protected:Npn \int_sub:Nn #1#2
6585 { \tex_advance:D #1 - \__int_eval:w #2 \__int_eval_end: }
6586 \cs_new_protected:Npn \int_gadd:Nn #1#2
6587 { \tex_global:D \tex_advance:D #1 \__int_eval:w #2 \__int_eval_end: }
6588 \cs_new_protected:Npn \int_gsub:Nn #1#2
6589 { \tex_global:D \tex_advance:D #1 - \__int_eval:w #2 \__int_eval_end: }
6590 \cs_generate_variant:Nn \int_add:Nn { c }
6591 \cs_generate_variant:Nn \int_gadd:Nn { c }
6592 \cs_generate_variant:Nn \int_sub:Nn { c }
6593 \cs_generate_variant:Nn \int_gsub:Nn { c }
6594 \cs_new_protected:Npn \int_incr:N #1
6595 { \tex_advance:D #1 \c_one_int }
6596 \cs_new_protected:Npn \int_decr:N #1
6597 { \tex_advance:D #1 - \c_one_int }
6598 \cs_new_protected:Npn \int_gincr:N #1
6599 { \tex_global:D \tex_advance:D #1 \c_one_int }
6600 \cs_new_protected:Npn \int_gdecr:N #1
6601 { \tex_global:D \tex_advance:D #1 - \c_one_int }
6602 \cs_generate_variant:Nn \int_incr:N { c }
6603 \cs_generate_variant:Nn \int_decr:N { c }
6604 \cs_generate_variant:Nn \int_gincr:N { c }
6605 \cs_generate_variant:Nn \int_gdecr:N { c }
6606 \cs_new_protected:Npn \int_set:Nn #1#2
6607 { #1 = \__int_eval:w #2 \__int_eval_end: }
6608 \cs_new_protected:Npn \int_gset:Nn #1#2
6609 { \tex_global:D #1 = \__int_eval:w #2 \__int_eval_end: }
6610 \cs_generate_variant:Nn \int_set:Nn { NV , c , cV }
6611 \cs_generate_variant:Nn \int_gset:Nn { NV , c , cV }
6612 \cs_new_protected:Npn \int_set_regex_count:Nnn #1#2#3
6613 { \regex_count:nnN {#2} {#3} #1 }
6614 \cs_generate_variant:Nn \int_set_regex_count:Nnn { c }
6615 \cs_new_protected:Npn \int_gset_regex_count:Nnn #1#2#3
6616 {
6617   \group_begin:
6618     \int_set_eq:NN \l__int_internal_a_int #1
6619     \regex_count:nnN {#2} {#3} \l__int_internal_a_int
6620     \int_gset_eq:NN #1 \l__int_internal_a_int
6621   \group_end:
6622 }
6623 \cs_generate_variant:Nn \int_gset_regex_count:Nnn { c }
6624 \cs_new_protected:Npn \int_set_regex_count:NNn #1#2#3

```



```

6625 { \regex_count:NnN #2 {#3} #1 } 6625
6626 \cs_generate_variant:Nn \int_set_regex_count:Nnn { c } 6626
6627 \cs_new_protected:Npn \int_gset_regex_count:NNn #1#2#3 6627
6628 { 6628
6629 \group_begin: 6629
6630 \int_set_eq:NN \l__int_internal_a_int #1 6630
6631 \regex_count:NnN #2 {#3} \l__int_internal_a_int 6631
6632 \int_gset_eq:NN #1 \l__int_internal_a_int 6632
6633 \group_end: 6633
6634 } 6634
6635 \cs_generate_variant:Nn \int_gset_regex_count:NNn { c } 6635
6636 \cs_new_eq:NN \int_use:N \tex_the:D 6636
6637 \cs_new:Npn \int_use:c #1 { \tex_the:D \cs:w #1 \cs_end: } 6637
6638 \cs_new_protected:Npn \__int_compare_error: 6638
6639 { 6639
6640 \if_int_compare:w \c_zero_int \c_zero_int \fi: 6640
6641 = 6641
6642 \__int_compare_error: 6642
6643 } 6643
6644 \cs_new:Npn \__int_compare_error:Nw 6644
6645 #1#2 \s__int_stop 6645
6646 { 6646
6647 { } 6647
6648 \c_zero_int \fi: 6648
6649 \msg_expandable_error:nnn 6649
6650 { kernel } { unknown-comparison } {#1} 6650
6651 \prg_return_false: 6651
6652 } 6652
6653 \prg_new_conditional:Npnn \int_compare:n #1 { p , T , F , TF } 6653
6654 { 6654
6655 \exp_after:wN \__int_compare:w 6655
6656 \int_value:w \__int_eval:w #1 \__int_compare_error: 6656
6657 } 6657
6658 \cs_new:Npn \__int_compare:w #1 \__int_compare_error: 6658
6659 { 6659
6660 \exp_after:wN \if_false: \int_value:w 6660
6661 \__int_compare:Nw #1 e { = nd_ } \s__int_stop 6661
6662 } 6662
6663 \cs_new:Npn \__int_compare:Nw #1#2 \s__int_stop 6663
6664 { 6664
6665 \exp_after:wN \__int_compare:NNw 6665
6666 \__int_to_roman:w - 0 #2 \s__int_mark 6666
6667 #1#2 \s__int_stop 6667
6668 } 6668
6669 \cs_new:Npn \__int_compare:NNw #1#2#3 \s__int_mark 6669
6670 { 6670

```

```

6671 \__kernel_exp_not:w
6672 \use:c
6673 {
6674     __int_compare_ \token_to_str:N #1
6675     \if_meaning:w = #2 = \fi:
6676     :NNw
6677 }
6678 \__int_compare_error:Nw #1
6679 }
6680 \cs_new:cpn { __int_compare_end=:NNw } #1#2#3 e #4 \s__int_stop
6681 {
6682     {#3} \exp_stop_f:
6683     \prg_return_false: \else: \prg_return_true: \fi:
6684 }
6685 \cs_new:Npn \__int_compare:nnN #1#2#3
6686 {
6687     {#2} \exp_stop_f:
6688     \prg_return_false: \exp_after:wN \__int_use_none_delimit_by_s_stop:w
6689     \fi:
6690     #1 #2 #3 \exp_after:wN \__int_compare:Nw \int_value:w \__int_eval:w
6691 }
6692 \cs_new:cpn { __int_compare=:NNw } #1#2#3 =
6693 { \__int_compare:nnN { \reverse_if:N \if_int_compare:w } {#3} = }
6694 \cs_new:cpn { __int_compare<:NNw } #1#2#3 <
6695 { \__int_compare:nnN { \reverse_if:N \if_int_compare:w } {#3} < }
6696 \cs_new:cpn { __int_compare>:NNw } #1#2#3 >
6697 { \__int_compare:nnN { \reverse_if:N \if_int_compare:w } {#3} > }
6698 \cs_new:cpn { __int_compare==:NNw } #1#2#3 ==
6699 { \__int_compare:nnN { \reverse_if:N \if_int_compare:w } {#3} = }
6700 \cs_new:cpn { __int_compare!=:NNw } #1#2#3 !=
6701 { \__int_compare:nnN { \if_int_compare:w } {#3} = }
6702 \cs_new:cpn { __int_compare<=:NNw } #1#2#3 <=
6703 { \__int_compare:nnN { \if_int_compare:w } {#3} > }
6704 \cs_new:cpn { __int_compare>=:NNw } #1#2#3 >=
6705 { \__int_compare:nnN { \if_int_compare:w } {#3} < }
6706 \prg_new_conditional:Npnn \int_compare:nNn #1#2#3 { p , T , F , TF }
6707 {
6708     \if_int_compare:w \__int_eval:w #1 #2 \__int_eval:w #3 \__int_eval_end:
6709     \prg_return_true:
6710     \else:
6711     \prg_return_false:
6712     \fi:
6713 }
6714 \prg_new_conditional:Npnn \int_if_zero:n #1 { p , T , F , TF }
6715 {
6716     \if_int_compare:w \__int_eval:w #1 = \c_zero_int

```

6717	\prg_return_true:	6717
6718	\else:	6718
6719	\prg_return_false:	6719
6720	\fi:	6720
6721	}	6721
6722	\cs_new:Npn \int_case:nnTF #1	6722
6723	{	6723
6724	\exp:w	6724
6725	\exp_args:Nf __int_case:nnTF { \int_eval:n {#1} }	6725
6726	}	6726
6727	\cs_new:Npn \int_case:nnT #1#2#3	6727
6728	{	6728
6729	\exp:w	6729
6730	\exp_args:Nf __int_case:nnTF { \int_eval:n {#1} } {#2} {#3} { }	6730
6731	}	6731
6732	\cs_new:Npn \int_case:nnF #1#2	6732
6733	{	6733
6734	\exp:w	6734
6735	\exp_args:Nf __int_case:nnTF { \int_eval:n {#1} } {#2} { }	6735
6736	}	6736
6737	\cs_new:Npn \int_case:nn #1#2	6737
6738	{	6738
6739	\exp:w	6739
6740	\exp_args:Nf __int_case:nnTF { \int_eval:n {#1} } {#2} { } { }	6740
6741	}	6741
6742	\cs_new:Npn __int_case:nnTF #1#2#3#4	6742
6743	{ __int_case:nw {#1} #2 {#1} { } \s__int_mark {#3} \s__int_mark {#4} \s__int_stop }	6743
6744	\cs_new:Npn __int_case:nw #1#2#3	6744
6745	{	6745
6746	\int_compare:nNnTF {#1} = {#2}	6746
6747	{ __int_case_end:nw {#3} }	6747
6748	{ __int_case:nw {#1} }	6748
6749	}	6749
6750	\cs_new:Npn __int_case_end:nw #1#2#3 \s__int_mark #4#5 \s__int_stop	6750
6751	{ \exp_end: #1 #4 }	6751
6752	\prg_new_conditional:Npnn \int_if_odd:n #1 { p , T , F , TF }	6752
6753	{	6753
6754	\if_int_odd:w __int_eval:w #1 __int_eval_end:	6754
6755	\prg_return_true:	6755
6756	\else:	6756
6757	\prg_return_false:	6757
6758	\fi:	6758
6759	}	6759
6760	\prg_new_conditional:Npnn \int_if_even:n #1 { p , T , F , TF }	6760
6761	{	6761
6762	\reverse_if:N \if_int_odd:w __int_eval:w #1 __int_eval_end:	6762

6763	\prg_return_true:	6763
6764	\else:	6764
6765	\prg_return_false:	6765
6766	\fi:	6766
6767	}	6767
6768	\cs_new:Npn \int_while_do:nn #1#2	6768
6769	{	6769
6770	\int_compare:nT {#1}	6770
6771	{	6771
6772	#2	6772
6773	\int_while_do:nn {#1} {#2}	6773
6774	}	6774
6775	}	6775
6776	\cs_new:Npn \int_until_do:nn #1#2	6776
6777	{	6777
6778	\int_compare:nF {#1}	6778
6779	{	6779
6780	#2	6780
6781	\int_until_do:nn {#1} {#2}	6781
6782	}	6782
6783	}	6783
6784	\cs_new:Npn \int_do_while:nn #1#2	6784
6785	{	6785
6786	#2	6786
6787	\int_compare:nT {#1}	6787
6788	{ \int_do_while:nn {#1} {#2} }	6788
6789	}	6789
6790	\cs_new:Npn \int_do_until:nn #1#2	6790
6791	{	6791
6792	#2	6792
6793	\int_compare:nF {#1}	6793
6794	{ \int_do_until:nn {#1} {#2} }	6794
6795	}	6795
6796	\cs_new:Npn \int_while_do:nNnn #1#2#3#4	6796
6797	{	6797
6798	\int_compare:nNnT {#1} #2 {#3}	6798
6799	{	6799
6800	#4	6800
6801	\int_while_do:nNnn {#1} #2 {#3} {#4}	6801
6802	}	6802
6803	}	6803
6804	\cs_new:Npn \int_until_do:nNnn #1#2#3#4	6804
6805	{	6805
6806	\int_compare:nNnF {#1} #2 {#3}	6806
6807	{	6807
6808	#4	6808

6809	\int_until_do:nNnn {#1} #2 {#3} {#4}	6809
6810	}	6810
6811	}	6811
6812	\cs_new:Npn \int_do_while:nNnn #1#2#3#4	6812
6813	{	6813
6814	#4	6814
6815	\int_compare:nNnT {#1} #2 {#3}	6815
6816	{ \int_do_while:nNnn {#1} #2 {#3} {#4} }	6816
6817	}	6817
6818	\cs_new:Npn \int_do_until:nNnn #1#2#3#4	6818
6819	{	6819
6820	#4	6820
6821	\int_compare:nNnF {#1} #2 {#3}	6821
6822	{ \int_do_until:nNnn {#1} #2 {#3} {#4} }	6822
6823	}	6823
6824	\cs_new:Npn \int_step_function:nnnN #1#2#3	6824
6825	{	6825
6826	\exp_after:wN __int_step:w	6826
6827	\int_value:w __int_eval:w #1 \exp_after:wN __int_sep:	6827
6828	\int_value:w __int_eval:w #2 \exp_after:wN __int_sep:	6828
6829	\int_value:w __int_eval:w #3 __int_sep:	6829
6830	}	6830
6831	\cs_new:Npn __int_step:w #1 __int_sep: #2 __int_sep: #3 __int_sep: #4	6831
6832	{	6832
6833	\int_compare:nNnTF {#2} > \c_zero_int	6833
6834	{ __int_step:Nw > }	6834
6835	{	6835
6836	\if_meaning:w 0 #2	6836
6837	\exp_after:wN \use_ii:nn	6837
6838	\fi:	6838
6839	\use_none:n	6839
6840	{	6840
6841	\msg_expandable_error:nnn	6841
6842	{ kernel } { zero-step } {#4}	6842
6843	\prg_break:	6843
6844	}	6844
6845	__int_step:Nw <	6845
6846	}	6846
6847	#1 __int_sep: {#2} {#3} {#4}	6847
6848	\prg_break_point:	6848
6849	}	6849
6850	\cs_new:Npn __int_step:Nw #1#2 __int_sep: #3#4#5	6850
6851	{	6851
6852	\if_int_compare:w #2 #1 #4 \exp_stop_f:	6852
6853	\prg_break:n	6853
6854	\fi:	6854

6855	#5 {#2}	6855
6856	\exp_after:wN __int_step:Nw	6856
6857	\exp_after:wN #1	6857
6858	\int_value:w __int_eval:w #2 + #3 __int_sep: {#3} {#4} {#5}	6858
6859	}	6859
6860	\cs_new:Npn \int_step_function:nN	6860
6861	{ \int_step_function:nnnN \c_one_int \c_one_int }	6861
6862	\cs_new:Npn \int_step_function:nnN #1	6862
6863	{ \int_step_function:nnnN {#1} \c_one_int }	6863
6864	\cs_new_eq:NN \int_step_tokens:nn \int_step_function:nN	6864
6865	\cs_new_eq:NN \int_step_tokens:nnn \int_step_function:nnN	6865
6866	\cs_new_eq:NN \int_step_tokens:nnnn \int_step_function:nnnN	6866
6867	\cs_new_protected:Npn \int_step_inline:nn	6867
6868	{ \int_step_inline:nnnn { 1 } { 1 } }	6868
6869	\cs_new_protected:Npn \int_step_inline:nnn #1	6869
6870	{ \int_step_inline:nnnn {#1} { 1 } }	6870
6871	\cs_new_protected:Npn \int_step_inline:nnnn	6871
6872	{	6872
6873	\int_gincr:N \g__kernel_prg_map_int	6873
6874	\exp_args:NNc __int_step:NNnnnn	6874
6875	\cs_gset_protected:Npn	6875
6876	{ __int_map_ \int_use:N \g__kernel_prg_map_int :w }	6876
6877	}	6877
6878	\cs_new_protected:Npn \int_step_variable:nNn	6878
6879	{ \int_step_variable:nnnNn { 1 } { 1 } }	6879
6880	\cs_new_protected:Npn \int_step_variable:nnNn #1	6880
6881	{ \int_step_variable:nnnNn {#1} { 1 } }	6881
6882	\cs_new_protected:Npn \int_step_variable:nnnNn #1#2#3#4#5	6882
6883	{	6883
6884	\int_gincr:N \g__kernel_prg_map_int	6884
6885	\exp_args:NNc __int_step:NNnnnn	6885
6886	\cs_gset_protected:Npe	6886
6887	{ __int_map_ \int_use:N \g__kernel_prg_map_int :w }	6887
6888	{#1}{#2}{#3}	6888
6889	{	6889
6890	\tl_set:Nn \exp_not:N #4 {##1}	6890
6891	\exp_not:n {#5}	6891
6892	}	6892
6893	}	6893
6894	\cs_new_protected:Npn __int_step:NNnnnn #1#2#3#4#5#6	6894
6895	{	6895
6896	#1 #2 ##1 {#6}	6896
6897	\int_step_function:nnnN {#3} {#4} {#5} #2	6897
6898	\prg_break_point:Nn \scan_stop: { \int_gdecr:N \g__kernel_prg_map_int }	6898
6899	}	6899
6900	\cs_new_eq:NN \int_to_arabic:n \int_eval:n	6900


```
6901 \cs_generate_variant:Nn \int_to_arabic:n { v } 6901
6902 \cs_new:Npn \int_to_symbols:nnn #1#2#3 6902
6903 { 6903
6904   \int_compare:nNnTF {#1} > {#2} 6904
6905   { 6905
6906     \__int_to_symbols:ennn 6906
6907     { 6907
6908       \int_case:nn 6908
6909         { 1 + \int_mod:nn { #1 - 1 } {#2} } 6909
6910         {#3} 6910
6911       } 6911
6912       {#1} {#2} {#3} 6912
6913     } 6913
6914     { \int_case:nn {#1} {#3} } 6914
6915   } 6915
6916 \cs_new:Npn \__int_to_symbols:nnnn #1#2#3#4 6916
6917 { 6917
6918   \exp_args:Nf \int_to_symbols:nnn 6918
6919   { \int_div_truncate:nn { #2 - 1 } {#3} } {#3} {#4} 6919
6920   #1 6920
6921 } 6921
6922 \cs_generate_variant:Nn \__int_to_symbols:nnnn { e } 6922
6923 \cs_new:Npn \int_to_alph:n #1 6923
6924 { 6924
6925   \int_to_symbols:nnn {#1} { 26 } 6925
6926   { 6926
6927     { 1 } { a } 6927
6928     { 2 } { b } 6928
6929     { 3 } { c } 6929
6930     { 4 } { d } 6930
6931     { 5 } { e } 6931
6932     { 6 } { f } 6932
6933     { 7 } { g } 6933
6934     { 8 } { h } 6934
6935     { 9 } { i } 6935
6936     { 10 } { j } 6936
6937     { 11 } { k } 6937
6938     { 12 } { l } 6938
6939     { 13 } { m } 6939
6940     { 14 } { n } 6940
6941     { 15 } { o } 6941
6942     { 16 } { p } 6942
6943     { 17 } { q } 6943
6944     { 18 } { r } 6944
6945     { 19 } { s } 6945
6946     { 20 } { t } 6946
```

6947	{ 21 } { u }	6947
6948	{ 22 } { v }	6948
6949	{ 23 } { w }	6949
6950	{ 24 } { x }	6950
6951	{ 25 } { y }	6951
6952	{ 26 } { z }	6952
6953	}	6953
6954	}	6954
6955	\cs_new:Npn \int_to_Alph:n #1	6955
6956	{	6956
6957	\int_to_symbols:nnn {#1} { 26 }	6957
6958	{	6958
6959	{ 1 } { A }	6959
6960	{ 2 } { B }	6960
6961	{ 3 } { C }	6961
6962	{ 4 } { D }	6962
6963	{ 5 } { E }	6963
6964	{ 6 } { F }	6964
6965	{ 7 } { G }	6965
6966	{ 8 } { H }	6966
6967	{ 9 } { I }	6967
6968	{ 10 } { J }	6968
6969	{ 11 } { K }	6969
6970	{ 12 } { L }	6970
6971	{ 13 } { M }	6971
6972	{ 14 } { N }	6972
6973	{ 15 } { O }	6973
6974	{ 16 } { P }	6974
6975	{ 17 } { Q }	6975
6976	{ 18 } { R }	6976
6977	{ 19 } { S }	6977
6978	{ 20 } { T }	6978
6979	{ 21 } { U }	6979
6980	{ 22 } { V }	6980
6981	{ 23 } { W }	6981
6982	{ 24 } { X }	6982
6983	{ 25 } { Y }	6983
6984	{ 26 } { Z }	6984
6985	}	6985
6986	}	6986
6987	\cs_new:Npn \int_to_base:nn #1	6987
6988	{ \exp_args:Nf __int_to_base:nn { \int_eval:n {#1} } }	6988
6989	\cs_new:Npn \int_to_Base:nn #1	6989
6990	{ \exp_args:Nf __int_to_Base:nn { \int_eval:n {#1} } }	6990
6991	\cs_new:Npn __int_to_base:nn #1#2	6991
6992	{	6992

```
6993 \int_compare:nNnTF {#1} < 0 6993
6994 { \exp_args:No \__int_to_base:nnN { \use_none:n #1 } {#2} - } 6994
6995 { \__int_to_base:nnN {#1} {#2} \c_empty_tl } 6995
6996 } 6996
6997 \cs_new:Npn \__int_to_Base:nn #1#2 6997
6998 { 6998
6999 \int_compare:nNnTF {#1} < 0 6999
7000 { \exp_args:No \__int_to_Base:nnN { \use_none:n #1 } {#2} - } 7000
7001 { \__int_to_Base:nnN {#1} {#2} \c_empty_tl } 7001
7002 } 7002
7003 \cs_new:Npn \__int_to_base:nnN #1#2#3 7003
7004 { 7004
7005 \int_compare:nNnTF {#1} < {#2} 7005
7006 { \exp_last_unbraced:Nf #3 { \__int_to_letter:n {#1} } } 7006
7007 { 7007
7008 \exp_args:Nf \__int_to_base:nnnN 7008
7009 { \__int_to_letter:n { \int_mod:nn {#1} {#2} } } 7009
7010 {#1} 7010
7011 {#2} 7011
7012 #3 7012
7013 } 7013
7014 } 7014
7015 \cs_new:Npn \__int_to_base:nnnN #1#2#3#4 7015
7016 { 7016
7017 \exp_args:Nf \__int_to_base:nnN 7017
7018 { \int_div_truncate:nn {#2} {#3} } 7018
7019 {#3} 7019
7020 #4 7020
7021 #1 7021
7022 } 7022
7023 \cs_new:Npn \__int_to_Base:nnN #1#2#3 7023
7024 { 7024
7025 \int_compare:nNnTF {#1} < {#2} 7025
7026 { \exp_last_unbraced:Nf #3 { \__int_to_Letter:n {#1} } } 7026
7027 { 7027
7028 \exp_args:Nf \__int_to_Base:nnnN 7028
7029 { \__int_to_Letter:n { \int_mod:nn {#1} {#2} } } 7029
7030 {#1} 7030
7031 {#2} 7031
7032 #3 7032
7033 } 7033
7034 } 7034
7035 \cs_new:Npn \__int_to_Base:nnnN #1#2#3#4 7035
7036 { 7036
7037 \exp_args:Nf \__int_to_Base:nnN 7037
7038 { \int_div_truncate:nn {#2} {#3} } 7038
```

7039	{#3}	7039
7040	#4	7040
7041	#1	7041
7042	}	7042
7043	\cs_new:Npn __int_to_letter:n #1	7043
7044	{	7044
7045	\exp_after:wN \exp_after:wN	7045
7046	\if_case:w __int_eval:w #1 - 10 __int_eval_end:	7046
7047	a	7047
7048	\or: b	7048
7049	\or: c	7049
7050	\or: d	7050
7051	\or: e	7051
7052	\or: f	7052
7053	\or: g	7053
7054	\or: h	7054
7055	\or: i	7055
7056	\or: j	7056
7057	\or: k	7057
7058	\or: l	7058
7059	\or: m	7059
7060	\or: n	7060
7061	\or: o	7061
7062	\or: p	7062
7063	\or: q	7063
7064	\or: r	7064
7065	\or: s	7065
7066	\or: t	7066
7067	\or: u	7067
7068	\or: v	7068
7069	\or: w	7069
7070	\or: x	7070
7071	\or: y	7071
7072	\or: z	7072
7073	\else: \int_value:w __int_eval:w #1 \exp_after:wN __int_eval_end:	7073
7074	\fi:	7074
7075	}	7075
7076	\cs_new:Npn __int_to_Letter:n #1	7076
7077	{	7077
7078	\exp_after:wN \exp_after:wN	7078
7079	\if_case:w __int_eval:w #1 - 10 __int_eval_end:	7079
7080	A	7080
7081	\or: B	7081
7082	\or: C	7082
7083	\or: D	7083
7084	\or: E	7084

7085	\or: F	7085
7086	\or: G	7086
7087	\or: H	7087
7088	\or: I	7088
7089	\or: J	7089
7090	\or: K	7090
7091	\or: L	7091
7092	\or: M	7092
7093	\or: N	7093
7094	\or: O	7094
7095	\or: P	7095
7096	\or: Q	7096
7097	\or: R	7097
7098	\or: S	7098
7099	\or: T	7099
7100	\or: U	7100
7101	\or: V	7101
7102	\or: W	7102
7103	\or: X	7103
7104	\or: Y	7104
7105	\or: Z	7105
7106	\else: \int_value:w __int_eval:w #1 \exp_after:wN __int_eval_end:	7106
7107	\fi:	7107
7108	}	7108
7109	\cs_new:Npn \int_to_bin:n #1	7109
7110	{ \int_to_base:nn {#1} { 2 } }	7110
7111	\cs_new:Npn \int_to_hex:n #1	7111
7112	{ \int_to_base:nn {#1} { 16 } }	7112
7113	\cs_new:Npn \int_to_Hex:n #1	7113
7114	{ \int_to_Base:nn {#1} { 16 } }	7114
7115	\cs_new:Npn \int_to_oct:n #1	7115
7116	{ \int_to_base:nn {#1} { 8 } }	7116
7117	\cs_new:Npn \int_to_roman:n #1	7117
7118	{	7118
7119	\exp_after:wN __int_to_roman:N	7119
7120	__int_to_roman:w \int_eval:n {#1} Q	7120
7121	}	7121
7122	\cs_new:Npn __int_to_roman:N #1	7122
7123	{	7123
7124	\use:c { __int_to_roman_ #1 :w }	7124
7125	__int_to_roman:N	7125
7126	}	7126
7127	\cs_new:Npn \int_to_Roman:n #1	7127
7128	{	7128
7129	\exp_after:wN __int_to_Roman_aux:N	7129
7130	__int_to_roman:w \int_eval:n {#1} Q	7130

```

7131 }
7132 \cs_new:Npn \__int_to_Roman_aux:N #1
7133 {
7134   \use:c { __int_to_Roman_ #1 :w }
7135   \__int_to_Roman_aux:N
7136 }
7137 \cs_new:Npn \__int_to_roman_i:w { i }
7138 \cs_new:Npn \__int_to_roman_v:w { v }
7139 \cs_new:Npn \__int_to_roman_x:w { x }
7140 \cs_new:Npn \__int_to_roman_l:w { l }
7141 \cs_new:Npn \__int_to_roman_c:w { c }
7142 \cs_new:Npn \__int_to_roman_d:w { d }
7143 \cs_new:Npn \__int_to_roman_m:w { m }
7144 \cs_new:Npn \__int_to_roman_Q:w #1 { }
7145 \cs_new:Npn \__int_to_Roman_i:w { I }
7146 \cs_new:Npn \__int_to_Roman_v:w { V }
7147 \cs_new:Npn \__int_to_Roman_x:w { X }
7148 \cs_new:Npn \__int_to_Roman_l:w { L }
7149 \cs_new:Npn \__int_to_Roman_c:w { C }
7150 \cs_new:Npn \__int_to_Roman_d:w { D }
7151 \cs_new:Npn \__int_to_Roman_m:w { M }
7152 \cs_new:Npn \__int_to_Roman_Q:w #1 { }
7153 \cs_new:Npn \__int_pass_signs:wn #1
7154 {
7155   \if:w + \if:w - \exp_not:N #1 + \fi: \exp_not:N #1
7156   \exp_after:wN \__int_pass_signs:wn
7157   \else:
7158     \exp_after:wN \__int_pass_signs_end:wn
7159     \exp_after:wN #1
7160   \fi:
7161 }
7162 \cs_new:Npn \__int_pass_signs_end:wn #1 \s__int_stop #2 { #2 #1 }
7163 \cs_new:Npn \int_from_alph:n #1
7164 {
7165   \int_eval:n
7166   {
7167     \exp_after:wN \__int_pass_signs:wn \tl_to_str:n {#1}
7168     \s__int_stop { \__int_from_alph:nN { 0 } }
7169     \q__int_recursion_tail \q__int_recursion_stop
7170   }
7171 }
7172 \cs_new:Npn \__int_from_alph:nN #1#2
7173 {
7174   \__int_if_recursion_tail_stop_do:Nn #2 {#1}
7175   \exp_args:Nf \__int_from_alph:nN
7176   { \int_eval:n { #1 * 26 + \__int_from_alph:N #2 } }

```



```
7177 } 7177
7178 \cs_new:Npn \__int_from_alph:N #1 7178
7179 { `#1 - \int_compare:nNnTF { `#1 } < { 91 } { 64 } { 96 } } 7179
7180 \cs_new:Npn \int_from_base:nn #1#2 7180
7181 { 7181
7182 \int_eval:n 7182
7183 { 7183
7184 \exp_after:wN \__int_pass_signs:wn \tl_to_str:n {#1} 7184
7185 \s_int_stop { \__int_from_base:nnN { 0 } {#2} } 7185
7186 \q__int_recursion_tail \q__int_recursion_stop 7186
7187 } 7187
7188 } 7188
7189 \cs_new:Npn \__int_from_base:nnN #1#2#3 7189
7190 { 7190
7191 \__int_if_recursion_tail_stop_do:Nn #3 {#1} 7191
7192 \exp_args:Nf \__int_from_base:nnN 7192
7193 { \int_eval:n { #1 * #2 + \__int_from_base:N #3 } } 7193
7194 {#2} 7194
7195 } 7195
7196 \cs_new:Npn \__int_from_base:N #1 7196
7197 { 7197
7198 \int_compare:nNnTF { `#1 } < { 58 } 7198
7199 {#1} 7199
7200 { `#1 - \int_compare:nNnTF { `#1 } < { 91 } { 55 } { 87 } } 7200
7201 } 7201
7202 \cs_new:Npn \int_from_bin:n #1 7202
7203 { \int_from_base:nn {#1} { 2 } } 7203
7204 \cs_new:Npn \int_from_hex:n #1 7204
7205 { \int_from_base:nn {#1} { 16 } } 7205
7206 \cs_new:Npn \int_from_oct:n #1 7206
7207 { \int_from_base:nn {#1} { 8 } } 7207
7208 \int_const:cn { c__int_from_roman_i_int } { 1 } 7208
7209 \int_const:cn { c__int_from_roman_v_int } { 5 } 7209
7210 \int_const:cn { c__int_from_roman_x_int } { 10 } 7210
7211 \int_const:cn { c__int_from_roman_l_int } { 50 } 7211
7212 \int_const:cn { c__int_from_roman_c_int } { 100 } 7212
7213 \int_const:cn { c__int_from_roman_d_int } { 500 } 7213
7214 \int_const:cn { c__int_from_roman_m_int } { 1000 } 7214
7215 \int_const:cn { c__int_from_roman_I_int } { 1 } 7215
7216 \int_const:cn { c__int_from_roman_V_int } { 5 } 7216
7217 \int_const:cn { c__int_from_roman_X_int } { 10 } 7217
7218 \int_const:cn { c__int_from_roman_L_int } { 50 } 7218
7219 \int_const:cn { c__int_from_roman_C_int } { 100 } 7219
7220 \int_const:cn { c__int_from_roman_D_int } { 500 } 7220
7221 \int_const:cn { c__int_from_roman_M_int } { 1000 } 7221
7222 \cs_new:Npn \int_from_roman:n #1 7222
```

```

7223 {
7224     \int_eval:n
7225     {
7226         (
7227             0
7228             \exp_after:wN \__int_from_roman:NN \tl_to_str:n {#1}
7229             \q__int_recursion_tail \q__int_recursion_tail \q__int_recursion_stop
7230         )
7231     }
7232 }
7233 \cs_new:Npn \__int_from_roman:NN #1#2
7234 {
7235     \__int_if_recursion_tail_stop:N #1
7236     \int_if_exist:cF { c__int_from_roman_ #1 _int }
7237     { \__int_from_roman_error:w }
7238     \__int_if_recursion_tail_stop_do:Nn #2
7239     { + \use:c { c__int_from_roman_ #1 _int } }
7240     \int_if_exist:cF { c__int_from_roman_ #2 _int }
7241     { \__int_from_roman_error:w }
7242     \int_compare:nNnTF
7243     { \use:c { c__int_from_roman_ #1 _int } }
7244     <
7245     { \use:c { c__int_from_roman_ #2 _int } }
7246     {
7247         + \use:c { c__int_from_roman_ #2 _int }
7248         - \use:c { c__int_from_roman_ #1 _int }
7249         \__int_from_roman:NN
7250     }
7251     {
7252         + \use:c { c__int_from_roman_ #1 _int }
7253         \__int_from_roman:NN #2
7254     }
7255 }
7256 \cs_new:Npn \__int_from_roman_error:w #1 \q__int_recursion_stop #2
7257 { #2 * 0 - 1 }
7258 \cs_new_eq:NN \int_show:N \__kernel_register_show:N
7259 \cs_generate_variant:Nn \int_show:N { c }
7260 \cs_new_protected:Npn \int_show:n
7261 { \__kernel_msg_show_eval:Nn \int_eval:n }
7262 \cs_new_eq:NN \int_log:N \__kernel_register_log:N
7263 \cs_generate_variant:Nn \int_log:N { c }
7264 \cs_new_protected:Npn \int_log:n
7265 { \__kernel_msg_log_eval:Nn \int_eval:n }
7266 \int_const:Nn \c_one_int { 1 }
7267 \int_const:Nn \c_max_int { 2 147 483 647 }
7268 \int_const:Nn \c_max_char_int

```

```

7269 {
7270     \if_int_odd:w 0
7271         \cs_if_exist:NT \tex luatexversion:D { 1 }
7272         \cs_if_exist:NT \tex XeTeXversion:D { 1 } ~
7273         "10FFFF
7274     \else:
7275         "FF
7276     \fi:
7277 }
7278 \int_new:N \l_tmpa_int
7279 \int_new:N \l_tmpb_int
7280 \int_new:N \g_tmpa_int
7281 \int_new:N \g_tmpb_int
7282 \int_new:N \l__seq_internal_a_int
7283 \int_new:N \l__seq_internal_b_int
7284 %% File: l3flag.dtx
7285 \cs_new_eq:NN \__flag_sep: \__kernel_int_sep:
7286 \cs_new_protected:Npn \flag_new:N #1
7287 { \cs_new_protected:Npe #1 { \cs_to_str:N #1 } }
7288 \cs_generate_variant:Nn \flag_new:N { c }
7289 \flag_new:N \l_tmpa_flag
7290 \flag_new:N \l_tmpb_flag
7291 \cs_new_protected:Npn \flag_clear:N #1
7292 {
7293     \__flag_clear:wN 0 \__flag_sep: #1
7294     \prg_break_point:
7295 }
7296 \cs_generate_variant:Nn \flag_clear:N { c }
7297 \cs_new_protected:Npn \__flag_clear:wN #1 \__flag_sep: #2
7298 {
7299     \if_cs_exist:w #2 #1 \cs_end: \else:
7300         \prg_break:n
7301     \fi:
7302     \cs_set_eq:cN { #2 #1 } \tex_undefined:D
7303     \exp_after:wN \__flag_clear:wN
7304     \int_value:w \int_eval:w \c_one_int + #1 \__flag_sep: #2
7305 }
7306 \cs_new_protected:Npn \flag_clear_new:N #1
7307 { \flag_if_exist:NTF #1 { \flag_clear:N } { \flag_new:N } #1 }
7308 \cs_generate_variant:Nn \flag_clear_new:N { c }
7309 \cs_new_protected:Npn \flag_show:N { \__flag_show:NN \tl_show:n }
7310 \cs_generate_variant:Nn \flag_show:N { c }
7311 \cs_new_protected:Npn \flag_log:N { \__flag_show:NN \tl_log:n }
7312 \cs_generate_variant:Nn \flag_log:N { c }
7313 \cs_new_protected:Npn \__flag_show:NN #1#2
7314 {

```

7315	_kernel_chk_defined:NT #2	7315
7316	{ \\exp_args:Ne #1 { \\tl_to_str:n { #2 height } = \\flag_height:N #2 } }	7316
7317	}	7317
7318	\\prg_new_eq_conditional:NNn \\flag_if_exist:N \\cs_if_exist:N	7318
7319	{ TF , T , F , p }	7319
7320	\\prg_new_eq_conditional:NNn \\flag_if_exist:c \\cs_if_exist:c	7320
7321	{ TF , T , F , p }	7321
7322	\\prg_new_conditional:Npnn \\flag_if_raised:N #1 { p , T , F , TF }	7322
7323	{	7323
7324	\\if_cs_exist:w #1 0 \\cs_end:	7324
7325	\\prg_return_true:	7325
7326	\\else:	7326
7327	\\prg_return_false:	7327
7328	\\fi:	7328
7329	}	7329
7330	\\prg_generate_conditional_variant:Nnn \\flag_if_raised:N	7330
7331	{ c } { p , T , F , TF }	7331
7332	\\cs_new:Npn \\flag_height:N #1 { __flag_height_loop:wN 0 __flag_sep: #1 }	7332
7333	\\cs_new:Npn __flag_height_loop:wN #1 __flag_sep: #2	7333
7334	{	7334
7335	\\if_cs_exist:w #2 #1 \\cs_end: \\else:	7335
7336	\\exp_after:wN __flag_height_end:wN	7336
7337	\\fi:	7337
7338	\\exp_after:wN __flag_height_loop:wN	7338
7339	\\int_value:w \\int_eval:w \\c_one_int + #1 __flag_sep: #2	7339
7340	}	7340
7341	\\cs_new:Npn __flag_height_end:wN #1 + #2 __flag_sep: #3 {#2}	7341
7342	\\cs_generate_variant:Nn \\flag_height:N { c }	7342
7343	\\cs_new:Npn \\flag_raise:N #1	7343
7344	{ \\exp_after:wN \\use_none:n \\cs:w #1 \\flag_height:N #1 \\cs_end: }	7344
7345	\\cs_generate_variant:Nn \\flag_raise:N { c }	7345
7346	\\cs_new:Npn \\flag_ensure_raised:N #1	7346
7347	{ \\exp_after:wN \\use_none:n \\cs:w #1 0 \\cs_end: }	7347
7348	\\cs_generate_variant:Nn \\flag_ensure_raised:N { c }	7348
7349	\\cs_new_protected:Npn \\flag_new:n #1 { \\flag_new:c { l_#1_flag } }	7349
7350	\\cs_new_protected:Npn \\flag_clear:n #1 { \\flag_clear:c { l_#1_flag } }	7350
7351	\\cs_new_protected:Npn \\flag_clear_new:n #1 { \\flag_clear_new:c { l_#1_flag } }	7351
7352	\\cs_new:Npn \\flag_if_exist_p:n #1 { \\flag_if_exist_p:c { l_#1_flag } }	7352
7353	\\cs_new:Npn \\flag_if_exist:nT #1 { \\flag_if_exist:cT { l_#1_flag } }	7353
7354	\\cs_new:Npn \\flag_if_exist:nF #1 { \\flag_if_exist:cF { l_#1_flag } }	7354
7355	\\cs_new:Npn \\flag_if_exist:nTF #1 { \\flag_if_exist:cTF { l_#1_flag } }	7355
7356	\\cs_new:Npn \\flag_if_raised_p:n #1 { \\flag_if_raised_p:c { l_#1_flag } }	7356
7357	\\cs_new:Npn \\flag_if_raised:nT #1 { \\flag_if_raised:cT { l_#1_flag } }	7357
7358	\\cs_new:Npn \\flag_if_raised:nF #1 { \\flag_if_raised:cF { l_#1_flag } }	7358
7359	\\cs_new:Npn \\flag_if_raised:nTF #1 { \\flag_if_raised:cTF { l_#1_flag } }	7359
7360	\\cs_new:Npn \\flag_height:n #1 { \\flag_height:c { l_#1_flag } }	7360

```
7361 \cs_new:Npn \flag_raise:n #1 { \flag_raise:c { l_#1_flag } } 7361
7362 \cs_new:Npn \flag_ensure_raised:n #1 { \flag_ensure_raised:c { l_#1_flag } } 7362
7363 \cs_new_protected:Npn \flag_show:n { \__flag_show:Nn \tl_show:n } 7363
7364 \cs_new_protected:Npn \flag_log:n { \__flag_show:Nn \tl_log:n } 7364
7365 \cs_new_protected:Npn \__flag_show:Nn #1#2 7365
7366 { 7366
7367   \exp_args:Nc \__kernel_chk_defined:NT { l_#2_flag } 7367
7368   { 7368
7369     \exp_args:Ne #1 7369
7370     { \tl_to_str:n { flag~#2~height } = \flag_height:n {#2} } 7370
7371   } 7371
7372 } 7372
7373 %% File: l3prg.dtx 7373
7374 \cs_new_eq:NN \if_predicate:w \tex_ifodd:D 7374
7375 \cs_new_protected:Npn \bool_new:N #1 { \cs_new_eq:NN #1 \c_false_bool } 7375
7376 \cs_generate_variant:Nn \bool_new:N { c } 7376
7377 \cs_new_protected:Npn \bool_const:Nn #1#2 7377
7378 { 7378
7379   \__kernel_chk_if_free_cs:N #1 7379
7380   \tex_global:D \tex_chardef:D #1 = \bool_if_p:n {#2} 7380
7381 } 7381
7382 \cs_generate_variant:Nn \bool_const:Nn { c } 7382
7383 \cs_new_protected:Npn \bool_set_true:N #1 7383
7384 { \cs_set_eq:NN #1 \c_true_bool } 7384
7385 \cs_new_protected:Npn \bool_set_false:N #1 7385
7386 { \cs_set_eq:NN #1 \c_false_bool } 7386
7387 \cs_new_protected:Npn \bool_gset_true:N #1 7387
7388 { \cs_gset_eq:NN #1 \c_true_bool } 7388
7389 \cs_new_protected:Npn \bool_gset_false:N #1 7389
7390 { \cs_gset_eq:NN #1 \c_false_bool } 7390
7391 \cs_generate_variant:Nn \bool_set_true:N { c } 7391
7392 \cs_generate_variant:Nn \bool_set_false:N { c } 7392
7393 \cs_generate_variant:Nn \bool_gset_true:N { c } 7393
7394 \cs_generate_variant:Nn \bool_gset_false:N { c } 7394
7395 \cs_new_eq:NN \bool_set_eq:NN \tl_set_eq:NN 7395
7396 \cs_new_eq:NN \bool_gset_eq:NN \tl_gset_eq:NN 7396
7397 \cs_generate_variant:Nn \bool_set_eq:NN { Nc, cN, cc } 7397
7398 \cs_generate_variant:Nn \bool_gset_eq:NN { Nc, cN, cc } 7398
7399 \cs_new_protected:Npn \bool_set:Nn #1#2 7399
7400 { 7400
7401   \exp_last_unbraced:NNNf 7401
7402   \tex_chardef:D #1 = { \bool_if_p:n {#2} } 7402
7403 } 7403
7404 \cs_new_protected:Npn \bool_gset:Nn #1#2 7404
7405 { 7405
7406   \exp_last_unbraced:NNNNf 7406
```

```
7407 \tex_global:D\tex_chardef:D #1 = { \bool_if_p:n {#2} } 7407
7408 } 7408
7409 \cs_generate_variant:Nn \bool_set:Nn { c } 7409
7410 \cs_generate_variant:Nn \bool_gset:Nn { c } 7410
7411 \cs_new_protected:Npn \bool_set_inverse:N #1 7411
7412 { \bool_if:NTF #1 { \bool_set_false:N } { \bool_set_true:N } #1 } 7412
7413 \cs_generate_variant:Nn \bool_set_inverse:N { c } 7413
7414 \cs_new_protected:Npn \bool_gset_inverse:N #1 7414
7415 { \bool_if:NTF #1 { \bool_gset_false:N } { \bool_gset_true:N } #1 } 7415
7416 \cs_generate_variant:Nn \bool_gset_inverse:N { c } 7416
7417 \quark_new:N \q__bool_recursion_tail 7417
7418 \quark_new:N \q__bool_recursion_stop 7418
7419 \cs_new:Npn \__bool_use_i_delimit_by_q_recursion_stop:nw 7419
7420 #1 #2 \q__bool_recursion_stop {#1} 7420
7421 \__kernel_quark_new_test:N \__bool_if_recursion_tail_stop_do:nn 7421
7422 \prg_new_conditional:Npnn \bool_if:N #1 { p , T , F , TF } 7422
7423 { 7423
7424 \if_bool:N #1 7424
7425 \prg_return_true: 7425
7426 \else: 7426
7427 \prg_return_false: 7427
7428 \fi: 7428
7429 } 7429
7430 \prg_generate_conditional_variant:Nnn \bool_if:N { c } { p , T , F , TF } 7430
7431 \cs_new:Npe \bool_to_str:N #1 7431
7432 { 7432
7433 \exp_not:N \bool_if:NTF #1 7433
7434 { \tl_to_str:n { true } } { \tl_to_str:n { false } } 7434
7435 } 7435
7436 \cs_generate_variant:Nn \bool_to_str:N { c } 7436
7437 \cs_new:Npe \bool_to_str:n #1 7437
7438 { 7438
7439 \exp_not:N \bool_if:nTF {#1} 7439
7440 { \tl_to_str:n { true } } { \tl_to_str:n { false } } 7440
7441 } 7441
7442 \cs_new_protected:Npn \bool_show:n 7442
7443 { \__kernel_msg_show_eval:Nn \bool_to_str:n } 7443
7444 \cs_new_protected:Npn \bool_log:n 7444
7445 { \__kernel_msg_log_eval:Nn \bool_to_str:n } 7445
7446 \cs_new_protected:Npn \bool_show:N { \__bool_show:NN \tl_show:n } 7446
7447 \cs_generate_variant:Nn \bool_show:N { c } 7447
7448 \cs_new_protected:Npn \bool_log:N { \__bool_show:NN \tl_log:n } 7448
7449 \cs_generate_variant:Nn \bool_log:N { c } 7449
7450 \cs_new_protected:Npn \__bool_show:NN #1#2 7450
7451 { 7451
7452 \__kernel_chk_defined:NT #2 7452
```



```

7453         {
7454             \token_case_meaning:NnF #2
7455             {
7456                 \c_true_bool { \exp_args:Ne #1 { \token_to_str:N #2 = true } }
7457                 \c_false_bool { \exp_args:Ne #1 { \token_to_str:N #2 = false } }
7458             }
7459             {
7460                 \msg_error:nneee { kernel } { bad-type }
7461                 { \token_to_str:N #2 } { \token_to_meaning:N #2 } { bool }
7462             }
7463         }
7464     }
7465     \bool_new:N \l_tmpa_bool
7466     \bool_new:N \l_tmpb_bool
7467     \bool_new:N \g_tmpa_bool
7468     \bool_new:N \g_tmpb_bool
7469     \prg_new_eq_conditional:NNn \bool_if_exist:N \cs_if_exist:N
7470     { TF , T , F , p }
7471     \prg_new_eq_conditional:NNn \bool_if_exist:c \cs_if_exist:c
7472     { TF , T , F , p }
7473     \prg_new_conditional:Npnn \bool_if:n #1 { T , F , TF }
7474     {
7475         \if_predicate:w \bool_if_p:n {#1}
7476         \prg_return_true:
7477         \else:
7478         \prg_return_false:
7479         \fi:
7480     }
7481     \cs_new:Npn \bool_if_p:n { \exp_args:Nf \__bool_if_p:n }
7482     \cs_new:Npn \__bool_if_p:n #1
7483     {
7484         \tl_if_empty:oT { \use_none:nn #1 . } { \__bool_if_p_aux:w }
7485         \group_align_safe_begin:
7486         \exp_after:wN
7487         \group_align_safe_end:
7488         \exp:w \exp_end_continue_f:w % (
7489         \__bool_get_next:NN \use_i:nnnn #1 )
7490     }
7491     \cs_new:Npn \__bool_if_p_aux:w #1 \use_i:nnnn #2#3
7492     { \bool_if:NTF #2 \c_true_bool \c_false_bool }
7493     \cs_new:Npn \__bool_get_next:NN #1#2
7494     {
7495         \use:c
7496         {
7497             __bool_
7498             \if_meaning:w !#2 ! \else: \if_meaning:w (#2 ( \else: p \fi: \fi:

```

```

7499         :Nw
7500     }
7501     #1 #2
7502 }
7503 \cs_new:cpn { __bool_!:Nw } #1#2
7504 {
7505     \exp_after:wN \__bool_get_next:NN
7506     #1 \use_ii:nnnn \use_i:nnnn \use_iii:nnnn \use_iv:nnnn
7507 }
7508 \cs_new:cpn { __bool_(:Nw } #1#2
7509 {
7510     \exp_after:wN \__bool_choose:NNN \exp_after:wN #1
7511     \int_value:w \__bool_get_next:NN \use_i:nnnn
7512 }
7513 \cs_new:cpn { __bool_p:Nw } #1
7514 { \exp_after:wN \__bool_choose:NNN \exp_after:wN #1 \int_value:w }
7515 \cs_new:Npn \__bool_choose:NNN #1#2#3
7516 {
7517     \use:c
7518     {
7519         __bool_ \token_to_str:N #3 _
7520         #1 #2 { \if_meaning:w 0 #2 1 \else: 0 \fi: } 2 0 :
7521     }
7522 }
7523 \cs_new:cpn { __bool_)_0: } { \c_false_bool }
7524 \cs_new:cpn { __bool_)_1: } { \c_true_bool }
7525 \cs_new:cpn { __bool_)_2: } { \c_true_bool }
7526 \cs_new:cpn { __bool_&_0: } & { \__bool_get_next:NN \use_iv:nnnn }
7527 \cs_new:cpn { __bool_&_1: } & { \__bool_get_next:NN \use_i:nnnn }
7528 \cs_new:cpn { __bool_&_2: } & { \__bool_get_next:NN \use_iii:nnnn }
7529 \cs_new:cpn { __bool_|_0: } | { \__bool_get_next:NN \use_i:nnnn }
7530 \cs_new:cpn { __bool_|_1: } | { \__bool_get_next:NN \use_iii:nnnn }
7531 \cs_new:cpn { __bool_|_2: } | { \__bool_get_next:NN \use_iii:nnnn }
7532 \cs_new:Npn \bool_lazy_all_p:n #1
7533 { \__bool_lazy_all:n #1 \q_bool_recursion_tail \q_bool_recursion_stop }
7534 \prg_new_conditional:Npnn \bool_lazy_all:n #1 { T , F , TF }
7535 {
7536     \if_predicate:w \bool_lazy_all_p:n {#1}
7537     \prg_return_true:
7538     \else:
7539     \prg_return_false:
7540     \fi:
7541 }
7542 \cs_new:Npn \__bool_lazy_all:n #1
7543 {
7544     \__bool_if_recursion_tail_stop_do:nn {#1} { \c_true_bool }

```

7545	\bool_if:nF {#1}	7545
7546	{ __bool_use_i_delimit_by_q_recursion_stop:nw { \c_false_bool } }	7546
7547	__bool_lazy_all:n	7547
7548	}	7548
7549	\prg_new_conditional:Npnn \bool_lazy_and:nn #1#2 { p , T , F , TF }	7549
7550	{	7550
7551	\if_predicate:w	7551
7552	\bool_if:nTF {#1} { \bool_if_p:n {#2} } { \c_false_bool }	7552
7553	\prg_return_true:	7553
7554	\else:	7554
7555	\prg_return_false:	7555
7556	\fi:	7556
7557	}	7557
7558	\cs_new:Npn \bool_lazy_any_p:n #1	7558
7559	{ __bool_lazy_any:n #1 \q_bool_recursion_tail \q_bool_recursion_stop }	7559
7560	\prg_new_conditional:Npnn \bool_lazy_any:n #1 { T , F , TF }	7560
7561	{	7561
7562	\if_predicate:w \bool_lazy_any_p:n {#1}	7562
7563	\prg_return_true:	7563
7564	\else:	7564
7565	\prg_return_false:	7565
7566	\fi:	7566
7567	}	7567
7568	\cs_new:Npn __bool_lazy_any:n #1	7568
7569	{	7569
7570	__bool_if_recursion_tail_stop_do:nn {#1} { \c_false_bool }	7570
7571	\bool_if:nT {#1}	7571
7572	{ __bool_use_i_delimit_by_q_recursion_stop:nw { \c_true_bool } }	7572
7573	__bool_lazy_any:n	7573
7574	}	7574
7575	\prg_new_conditional:Npnn \bool_lazy_or:nn #1#2 { p , T , F , TF }	7575
7576	{	7576
7577	\if_predicate:w	7577
7578	\bool_if:nTF {#1} { \c_true_bool } { \bool_if_p:n {#2} }	7578
7579	\prg_return_true:	7579
7580	\else:	7580
7581	\prg_return_false:	7581
7582	\fi:	7582
7583	}	7583
7584	\cs_new:Npn \bool_not_p:n #1 { \bool_if_p:n { ! (#1) } }	7584
7585	\prg_new_conditional:Npnn \bool_xor:nn #1#2 { p , T , F , TF }	7585
7586	{	7586
7587	\bool_if:nT {#1} \reverse_if:N	7587
7588	\if_predicate:w \bool_if_p:n {#2}	7588
7589	\prg_return_true:	7589
7590	\else:	7590

7591	\prg_return_false:	7591
7592	\fi:	7592
7593	}	7593
7594	\cs_new:Npn \bool_while_do:Nn #1#2	7594
7595	{ \bool_if:NT #1 { #2 \bool_while_do:Nn #1 {#2} } }	7595
7596	\cs_new:Npn \bool_until_do:Nn #1#2	7596
7597	{ \bool_if:NF #1 { #2 \bool_until_do:Nn #1 {#2} } }	7597
7598	\cs_generate_variant:Nn \bool_while_do:Nn { c }	7598
7599	\cs_generate_variant:Nn \bool_until_do:Nn { c }	7599
7600	\cs_new:Npn \bool_do_while:Nn #1#2	7600
7601	{ #2 \bool_if:NT #1 { \bool_do_while:Nn #1 {#2} } }	7601
7602	\cs_new:Npn \bool_do_until:Nn #1#2	7602
7603	{ #2 \bool_if:NF #1 { \bool_do_until:Nn #1 {#2} } }	7603
7604	\cs_generate_variant:Nn \bool_do_while:Nn { c }	7604
7605	\cs_generate_variant:Nn \bool_do_until:Nn { c }	7605
7606	\cs_new:Npn \bool_while_do:nn #1#2	7606
7607	{	7607
7608	\bool_if:nT {#1}	7608
7609	{	7609
7610	#2	7610
7611	\bool_while_do:nn {#1} {#2}	7611
7612	}	7612
7613	}	7613
7614	\cs_new:Npn \bool_do_while:nn #1#2	7614
7615	{	7615
7616	#2	7616
7617	\bool_if:nT {#1} { \bool_do_while:nn {#1} {#2} }	7617
7618	}	7618
7619	\cs_new:Npn \bool_until_do:nn #1#2	7619
7620	{	7620
7621	\bool_if:nF {#1}	7621
7622	{	7622
7623	#2	7623
7624	\bool_until_do:nn {#1} {#2}	7624
7625	}	7625
7626	}	7626
7627	\cs_new:Npn \bool_do_until:nn #1#2	7627
7628	{	7628
7629	#2	7629
7630	\bool_if:nF {#1} { \bool_do_until:nn {#1} {#2} }	7630
7631	}	7631
7632	\scan_new:N \s__bool_mark	7632
7633	\scan_new:N \s__bool_stop	7633
7634	\cs_new:Npn \bool_case:nTF	7634
7635	{ \exp:w __bool_case:nTF }	7635
7636	\cs_new:Npn \bool_case:nT #1#2	7636

[illegible]

```
7683 \cs_new:cpn { __prg_replicate_8:n } #1 7683
7684   { \cs_end: {#1#1#1#1#1#1#1#1#1#1} #1#1#1#1#1#1#1#1#1 } 7684
7685 \cs_new:cpn { __prg_replicate_9:n } #1 7685
7686   { \cs_end: {#1#1#1#1#1#1#1#1#1#1} #1#1#1#1#1#1#1#1#1 } 7686
7687 \cs_new:cpn { __prg_replicate_first_-:n } #1 7687
7688   { 7688
7689     \exp_end: 7689
7690     \msg_expandable_error:nn { prg } { negative-replication } 7690
7691   } 7691
7692 \cs_new:cpn { __prg_replicate_first_0:n } #1 { \exp_end: } 7692
7693 \cs_new:cpn { __prg_replicate_first_1:n } #1 { \exp_end: #1 } 7693
7694 \cs_new:cpn { __prg_replicate_first_2:n } #1 { \exp_end: #1#1 } 7694
7695 \cs_new:cpn { __prg_replicate_first_3:n } #1 { \exp_end: #1#1#1 } 7695
7696 \cs_new:cpn { __prg_replicate_first_4:n } #1 { \exp_end: #1#1#1#1 } 7696
7697 \cs_new:cpn { __prg_replicate_first_5:n } #1 { \exp_end: #1#1#1#1#1 } 7697
7698 \cs_new:cpn { __prg_replicate_first_6:n } #1 { \exp_end: #1#1#1#1#1#1 } 7698
7699 \cs_new:cpn { __prg_replicate_first_7:n } #1 { \exp_end: #1#1#1#1#1#1#1 } 7699
7700 \cs_new:cpn { __prg_replicate_first_8:n } #1 { \exp_end: #1#1#1#1#1#1#1#1 } 7700
7701 \cs_new:cpn { __prg_replicate_first_9:n } #1 7701
7702   { \exp_end: #1#1#1#1#1#1#1#1#1#1 } 7702
7703 \prg_new_conditional:Nppn \mode_if_vertical: { p , T , F , TF } 7703
7704   { \if_mode_vertical: \prg_return_true: \else: \prg_return_false: \fi: } 7704
7705 \prg_new_conditional:Nppn \mode_if_horizontal: { p , T , F , TF } 7705
7706   { \if_mode_horizontal: \prg_return_true: \else: \prg_return_false: \fi: } 7706
7707 \prg_new_conditional:Nppn \mode_if_inner: { p , T , F , TF } 7707
7708   { \if_mode_inner: \prg_return_true: \else: \prg_return_false: \fi: } 7708
7709 \prg_new_conditional:Nppn \mode_if_math: { p , T , F , TF } 7709
7710   { \if_mode_math: \prg_return_true: \else: \prg_return_false: \fi: } 7710
7711 \group_begin: 7711
7712 \tex_catcode:D \^^@ = 2 \exp_stop_f: 7712
7713 \cs_new:Npn \group_align_safe_begin: 7713
7714   { \exp:w \if_false: { \fi: ^^@ \exp_stop_f: } 7714
7715 \tex_catcode:D \^^@ = 1 \exp_stop_f: 7715
7716 \cs_new:Npn \group_align_safe_end: 7716
7717   { \exp:w ^^@ \if_false: } \fi: \exp_stop_f: } 7717
7718 \group_end: 7718
7719 \int_new:N \g__kernel_prg_map_int 7719
7720 % File: l3sys.dtx 7720
7721 \tl_new:N \l__sys_tmp_tl 7721
7722 \cs_new_protected:Npn \__sys_const:nn #1#2 7722
7723   { 7723
7724     \bool_if:nTF {#2} 7724
7725     { 7725
7726       \cs_new_eq:cN { #1 :T } \use:n 7726
7727       \cs_new_eq:cN { #1 :F } \use_none:n 7727
7728       \cs_new_eq:cN { #1 :TF } \use_i:nn 7728
```



```

7729 \cs_new_eq:cN { #1 _p: } \c_true_bool 7729
7730 } 7730
7731 { 7731
7732 \cs_new_eq:cN { #1 :T } \use_none:n 7732
7733 \cs_new_eq:cN { #1 :F } \use:n 7733
7734 \cs_new_eq:cN { #1 :TF } \use_ii:nn 7734
7735 \cs_new_eq:cN { #1 _p: } \c_false_bool 7735
7736 } 7736
7737 } 7737
7738 \str_const:Ne \c_sys_engine_str 7738
7739 { 7739
7740 \cs_if_exist:NT \tex luatexversion:D { luatex } 7740
7741 \cs_if_exist:NT \tex pdftexversion:D { pdftex } 7741
7742 \cs_if_exist:NT \tex kanjiskip:D 7742
7743 { 7743
7744 \cs_if_exist:NTF \tex enablecjktoken:D 7744
7745 { uptex } 7745
7746 { ptex } 7746
7747 } 7747
7748 \cs_if_exist:NT \tex XeTeXversion:D { xetex } 7748
7749 } 7749
7750 \tl_map_inline:nn { { luatex } { pdftex } { ptex } { uptex } { xetex } } 7750
7751 { 7751
7752 \__sys_const:nn { sys_if_engine_ #1 } 7752
7753 { \str_if_eq_p:Vn \c_sys_engine_str {#1} } 7753
7754 } 7754
7755 \__sys_const:nn 7755
7756 { sys_if_engine_opentype } 7756
7757 { \cs_if_exist_p:N \tex_Umathcode:D } 7757
7758 \group_begin: 7758
7759 \cs_set_eq:NN \lua_now:e \tex_directlua:D 7759
7760 \str_const:Ne \c_sys_engine_exec_str 7760
7761 { 7761
7762 \sys_if_engine_pdftex:T { pdf } 7762
7763 \sys_if_engine_xetex:T { xe } 7763
7764 \sys_if_engine_ptex:T { ep } 7764
7765 \sys_if_engine_uptex:T { eup } 7765
7766 \sys_if_engine_luatex:T 7766
7767 { 7767
7768 lua \lua_now:e 7768
7769 { 7769
7770 if (pcall(require, 'luaharfbuzz')) then ~ 7770
7771 tex.print("hb") ~ 7771
7772 end 7772
7773 } 7773
7774 } 7774

```

```
7775 tex 7775
7776 } 7776
7777 \group_end: 7777
7778 \str_const:Ne \c_sys_engine_format_str 7778
7779 { 7779
7780 \cs_if_exist:NTF \fmtname 7780
7781 { 7781
7782 \bool_lazy_or:nnTF 7782
7783 { \str_if_eq_p:Vn \fmtname { plain } } 7783
7784 { \str_if_eq_p:Vn \fmtname { LaTeX2e } } 7784
7785 { 7785
7786 \sys_if_engine_pdftex:T 7786
7787 { \int_compare:nNnT { \tex_pdfoutput:D } = { 1 } { pdf } } 7787
7788 \sys_if_engine_xetex:T { xe } 7788
7789 \sys_if_engine_ptex:T { p } 7789
7790 \sys_if_engine_uptex:T { up } 7790
7791 \sys_if_engine luatex:T 7791
7792 { 7792
7793 \int_compare:nNnT { \tex_pdfoutput:D } = { 0 } { dvi } 7793
7794 lua 7794
7795 } 7795
7796 \str_if_eq:VnTF \fmtname { LaTeX2e } 7796
7797 { latex } 7797
7798 { 7798
7799 \bool_lazy_and:nnT 7799
7800 { \sys_if_engine_pdftex_p: } 7800
7801 { \int_compare_p:nNn { \tex_pdfoutput:D } = { 0 } } 7801
7802 { e } 7802
7803 tex 7803
7804 } 7804
7805 } 7805
7806 { \fmtname } 7806
7807 } 7807
7808 { unknown } 7808
7809 } 7809
7810 \str_const:Ne \c_sys_engine_version_str 7810
7811 { 7811
7812 \str_case:on \c_sys_engine_str 7812
7813 { 7813
7814 { pdftex } 7814
7815 { 7815
7816 \int_div_truncate:nn { \tex_pdftexversion:D } { 100 } 7816
7817 . 7817
7818 \int_mod:nn { \tex_pdftexversion:D } { 100 } 7818
7819 . 7819
7820 \tex_pdftexrevision:D 7820
```

7821	}	7821
7822	{ ptex }	7822
7823	{	7823
7824	\cs_if_exist:NT \tex_ptexversion:D	7824
7825	{	7825
7826	p	7826
7827	\int_use:N \tex_ptexversion:D	7827
7828	.	7828
7829	\int_use:N \tex_ptexminorversion:D	7829
7830	\tex_ptexrevision:D	7830
7831	-	7831
7832	\int_use:N \tex_epTeXversion:D	7832
7833	}	7833
7834	}	7834
7835	{ luatex }	7835
7836	{	7836
7837	\int_div_truncate:nn { \tex_luatexversion:D } { 100 }	7837
7838	.	7838
7839	\int_mod:nn { \tex_luatexversion:D } { 100 }	7839
7840	.	7840
7841	\tex_luatexrevision:D	7841
7842	}	7842
7843	{ uptex }	7843
7844	{	7844
7845	\cs_if_exist:NT \tex_ptexversion:D	7845
7846	{	7846
7847	p	7847
7848	\int_use:N \tex_ptexversion:D	7848
7849	.	7849
7850	\int_use:N \tex_ptexminorversion:D	7850
7851	\tex_ptexrevision:D	7851
7852	-	7852
7853	u	7853
7854	\int_use:N \tex_uptexversion:D	7854
7855	\tex_uptexrevision:D	7855
7856	-	7856
7857	\int_use:N \tex_epTeXversion:D	7857
7858	}	7858
7859	}	7859
7860	{ xetex }	7860
7861	{	7861
7862	\int_use:N \tex_XeTeXversion:D	7862
7863	\tex_XeTeXrevision:D	7863
7864	}	7864
7865	}	7865
7866	}	7866

```

7867 \cs_new_protected:Npn \sys_load_backend:n #1
7868 {
7869     \sys_finalise:
7870     \str_if_exist:NTF \c_sys_backend_str
7871     {
7872         \str_if_eq:VnF \c_sys_backend_str {#1}
7873         { \msg_error:nn { sys } { backend-set } }
7874     }
7875     {
7876         \tl_if_blank:nF {#1}
7877         { \tl_gset:Nn \g__sys_backend_tl {#1} }
7878         \__sys_load_backend_check:N \g__sys_backend_tl
7879         \str_const:Ne \c_sys_backend_str { \g__sys_backend_tl }
7880         \__kernel_sys_configuration_load:n
7881         { l3backend- \c_sys_backend_str }
7882     }
7883 }
7884 \cs_new_protected:Npn \__sys_load_backend_check:N #1
7885 {
7886     \sys_if_engine_xetex:TF
7887     {
7888         \str_case:VnF #1
7889         {
7890             { dvisvgm } { }
7891             { xdvipdfmx } { \tl_gset:Nn #1 { xetex } }
7892             { xetex } { }
7893         }
7894         {
7895             \msg_error:nnee { sys } { wrong-backend }
7896             #1 { xetex }
7897             \tl_gset:Nn #1 { xetex }
7898         }
7899     }
7900     {
7901         \sys_if_output_pdf:TF
7902         {
7903             \str_if_eq:VnTF #1 { pdfmode }
7904             {
7905                 \sys_if_engine luatex:TF
7906                 { \tl_gset:Nn #1 { luatex } }
7907                 { \tl_gset:Nn #1 { pdftex } }
7908             }
7909             {
7910                 \bool_lazy_or:nnF
7911                 { \str_if_eq_p:Vn #1 { luatex } }
7912                 { \str_if_eq_p:Vn #1 { pdftex } }

```

```

7913         {
7914             \msg_error:nnee { sys } { wrong-backend }
7915             #1 { \sys_if_engine luatex:TF { luatex } { pdftex } }
7916             \sys_if_engine luatex:TF
7917                 { \tl_gset:Nn #1 { luatex } }
7918                 { \tl_gset:Nn #1 { pdftex } }
7919         }
7920     }
7921 }
7922 {
7923     \str_case:VnF #1
7924     {
7925         { dvipdfmx } { }
7926         { dvips } { }
7927         { dvisvgm } { }
7928     }
7929     {
7930         \msg_error:nnee { sys } { wrong-backend }
7931         #1 { dvips }
7932         \tl_gset:Nn #1 { dvips }
7933     }
7934 }
7935 }
7936 }
7937 \cs_new_protected:Npn \sys_ensure_backend:
7938 {
7939     \str_if_exist:NF \c_sys_backend_str
7940     { \sys_load_backend:n { } }
7941 }
7942 \bool_new:N \g__sys_debug_bool
7943 \cs_new_protected:Npn \sys_load_debug:
7944 {
7945     \bool_if:NF \g__sys_debug_bool
7946     { \__kernel_sys_configuration_load:n { l3debug } }
7947     \bool_gset_true:N \g__sys_debug_bool
7948 }
7949 \cs_if_exist:NT \@expl@finalise@setup@@
7950 {
7951     \tl_gput_right:Nn \@expl@finalise@setup@@
7952     {
7953         \tl_gput_right:Nn \@kernel@after@begindocument
7954         {
7955             \cs_gset_protected:Npn \sys_load_debug:
7956             { \msg_error:nn { sys } { load-debug-in-preamble } }
7957         }
7958     }

```

```

7959 }
7960 \tl_new:N \l__sys_internal_tl
7961 \tl_const:Nc \c__sys_marker_tl { : \token_to_str:N : }
7962 \cs_new_protected:Npn \sys_get_shell:nnN #1#2#3
7963 {
7964   \sys_get_shell:nnNF {#1} {#2} #3
7965   { \tl_set:Nn #3 { \q_no_value } }
7966 }
7967 \prg_new_protected_conditional:Npnn \sys_get_shell:nnN #1#2#3 { T , F , TF }
7968 {
7969   \sys_if_shell:TF
7970   { \exp_args:No \__sys_get:nnN { \tl_to_str:n {#1} } {#2} #3 }
7971   { \prg_return_false: }
7972 }
7973 \cs_new_protected:Npn \__sys_get:nnN #1#2#3
7974 {
7975   \tl_if_in:nnTF {#1} { " }
7976   {
7977     \msg_error:nne
7978     { kernel } { quote-in-shell } {#1}
7979     \prg_return_false:
7980   }
7981   {
7982     \group_begin:
7983     \if_false: { \fi:
7984       \int_set_eq:NN \tex_tracingnesting:D \c_zero_int
7985       \exp_args:No \tex_everyeof:D { \c__sys_marker_tl }
7986       #2 \scan_stop:
7987       \exp_after:wN \__sys_get_do:Nw
7988       \exp_after:wN #3
7989       \exp_after:wN \prg_do_nothing:
7990       \tex_input:D | "#1" \scan_stop:
7991       \if_false: } \fi:
7992       \prg_return_true:
7993     }
7994   }
7995   \exp_args:Nno \use:nn
7996   { \cs_new_protected:Npn \__sys_get_do:Nw #1#2 }
7997   { \c__sys_marker_tl }
7998   {
7999     \group_end:
8000     \tl_set:Nc #1 {#2}
8001   }
8002   \sys_if_engine luatex:F
8003   { \int_const:Nn \c__sys_shell_stream_int { 18 } }
8004   \sys_if_engine luatex:TF

```



```

8005 {
8006     \cs_new_protected:Npn \sys_shell_now:n #1
8007         { \__sys_shell_now:e { \exp_not:n {#1} } }
8008 }
8009 {
8010     \cs_new_protected:Npn \sys_shell_now:n #1
8011         { \iow_now:Nn \c__sys_shell_stream_int {#1} }
8012 }
8013 \cs_generate_variant:Nn \sys_shell_now:n { e , x }
8014 \sys_if_engine luatex:TF
8015 {
8016     \cs_new_protected:Npn \sys_shell_shipout:n #1
8017         { \__sys_shell_shipout:e { \exp_not:n {#1} } }
8018 }
8019 {
8020     \cs_new_protected:Npn \sys_shell_shipout:n #1
8021         { \iow_shipout:Nn \c__sys_shell_stream_int {#1} }
8022 }
8023 \cs_generate_variant:Nn \sys_shell_shipout:n { e , x }
8024 \cs_new_protected:Npn \__kernel_sys_everyjob:
8025 {
8026     \tl_use:N \g__sys_everyjob_tl
8027     \tl_gclear:N \g__sys_everyjob_tl
8028 }
8029 \cs_new_protected:Npn \__sys_everyjob:n #1
8030     { \tl_gput_right:Nn \g__sys_everyjob_tl {#1} }
8031 \tl_new:N \g__sys_everyjob_tl
8032 \__sys_everyjob:n
8033     { \cs_new_eq:NN \c_sys_jobname_str \tex_jobname:D }
8034 \__sys_everyjob:n
8035 {
8036     \group_begin:
8037     \cs_set:Npn \__sys_tmp:w #1
8038         {
8039             \str_if_eq:eeTF { \cs_meaning:N #1 } { \token_to_str:N #1 }
8040                 { #1 }
8041                 {
8042                     \cs_if_exist:NTF \tex_primitive:D
8043                         {
8044                             \bool_lazy_and:nnTF
8045                                 { \sys_if_engine_xetex_p: }
8046                                 {
8047                                     \int_compare_p:nNn
8048                                         { \exp_after:wN \use_none:n \tex_XeTeXrevision:D }
8049                                         < { 99999 }
8050                                 }

```

```

8051         { 0 }
8052         { \tex_primitive:D #1 }
8053     }
8054     { 0 }
8055 }
8056 }
8057 \int_const:Nn \c_sys_minute_int
8058 { \int_mod:nn { \__sys_tmp:w \time } { 60 } }
8059 \int_const:Nn \c_sys_hour_int
8060 { \int_div_truncate:nn { \__sys_tmp:w \time } { 60 } }
8061 \int_const:Nn \c_sys_day_int { \__sys_tmp:w \day }
8062 \int_const:Nn \c_sys_month_int { \__sys_tmp:w \month }
8063 \int_const:Nn \c_sys_year_int { \__sys_tmp:w \year }
8064 \group_end:
8065 }
8066 \__sys_everyjob:n
8067 {
8068     \str_const:Ne \c_sys_timestamp_str
8069     {
8070         \cs_if_exist:NTF \tex_directlua:D
8071         { \tex_directlua:D { tex.print(pdf.getcreationdate()) } }
8072         { \tex_creationdate:D }
8073     }
8074 }
8075 \__sys_everyjob:n
8076 {
8077     \cs_new:Npn \sys_rand_seed: { \tex_the:D \tex_randomseed:D }
8078 }
8079 \__sys_everyjob:n
8080 {
8081     \cs_new_protected:Npn \sys_gset_rand_seed:n #1
8082     { \tex_setrandomseed:D \int_eval:n {#1} \exp_stop_f: }
8083 }
8084 \cs_new:Npe \sys_timer:
8085 {
8086     \sys_if_engine luatex:TF
8087     { \exp_not:N \__sys_elapsedtime: }
8088     { \exp_not:N \int_value:w \exp_not:N \tex_elapsedtime:D }
8089 }
8090 \__sys_everyjob:n
8091 {
8092     \int_const:Nn \c_sys_shell_escape_int
8093     {
8094         \sys_if_engine luatex:TF
8095         {
8096             \tex_directlua:D

```

```
8097         { tex.sprint(status.shell_escape~or~os.execute()) } 8097
8098     } 8098
8099     { \tex_shellescape:D } 8099
8100 } 8100
8101 } 8101
8102 \__sys_everyjob:n 8102
8103 { 8103
8104     \__sys_const:nn { sys_if_shell } 8104
8105     { \int_compare_p:nNn \c_sys_shell_escape_int > 0 } 8105
8106     \__sys_const:nn { sys_if_shell_unrestricted } 8106
8107     { \int_compare_p:nNn \c_sys_shell_escape_int = 1 } 8107
8108     \__sys_const:nn { sys_if_shell_restricted } 8108
8109     { \int_compare_p:nNn \c_sys_shell_escape_int = 2 } 8109
8110 } 8110
8111 \cs_new_protected:Npn \sys_get_query:nN #1#2 8111
8112 { \sys_get_query:nnnN {#1} { } { } #2 } 8112
8113 \cs_new_protected:Npn \sys_get_query:nnN #1#2#3 8113
8114 { \sys_get_query:nnnN {#1} { } {#2} #3 } 8114
8115 \cs_new_protected:Npn \sys_get_query:nnnN #1#2#3#4 8115
8116 { 8116
8117     \tl_clear:N #4 8117
8118     \__sys_get_query_auxi:neeN {#1} {#2} {#3} #4 8118
8119 } 8119
8120 \cs_new:Npn \__sys_get_query_auxi:nnnN #1#2#3#4 8120
8121 { 8121
8122     \__sys_get_query_auxii:neeN {#1} 8122
8123     { \tl_if_blank:nF {#2} { \tl_to_str:n { ~ #2 } } } 8123
8124     { 8124
8125         \tl_if_blank:nF {#3} 8125
8126         { 8126
8127             \c_space_tl 8127
8128             \sys_if_shell_restricted:F ' 8128
8129             \tl_to_str:n {#3} 8129
8130             \sys_if_shell_restricted:F ' 8130
8131         } 8131
8132     } 8132
8133     #4 8133
8134 } 8134
8135 \cs_generate_variant:Nn \__sys_get_query_auxi:nnnN { nee } 8135
8136 \cs_new_protected:Npn \__sys_get_query_auxii:nnnN #1#2#3#4 8136
8137 { 8137
8138     \sys_if_shell:T 8138
8139     { 8139
8140         \sys_get_shell:nnN 8140
8141         { l3sys-query~#1 #2 #3 } 8141
8142     } 8142
```

```

8143 \int_step_inline:nnn { 0 } { `A - 1 } 8143
8144 { \char_set_catcode_other:n {##1} } 8144
8145 \int_step_inline:nnn { `Z + 1 } { `a - 1 } 8145
8146 { \char_set_catcode_other:n {##1} } 8146
8147 \int_step_inline:nnn { `z + 1 } { 127 } 8147
8148 { \char_set_catcode_other:n {##1} } 8148
8149 \char_set_catcode_active:n { `\_ } 8149
8150 \tex_endlinechar:D 13 \scan_stop: 8150
8151 } 8151
8152 \l__sys_tmp_tl 8152
8153 \tl_if_empty:NF \l__sys_tmp_tl 8153
8154 { 8154
8155 \exp_after:wN \__sys_get_query:Nw \exp_after:wN #4 8155
8156 \l__sys_tmp_tl \q_stop 8156
8157 } 8157
8158 } 8158
8159 } 8159
8160 \cs_generate_variant:Nn \__sys_get_query_auxii:nnnN { nee } 8160
8161 \group_begin: 8161
8162 \tex_lccode:D `* = 13 \scan_stop: 8162
8163 \tex_lowercase:D 8163
8164 { 8164
8165 \group_end: 8165
8166 \cs_new_protected:Npn \__sys_get_query:Nw #1#2 * \q_stop 8166
8167 } 8167
8168 { \tl_set:Nn #1 {#2} } 8168
8169 \cs_new_protected:Npn \sys_split_query:nN #1#2 8169
8170 { \sys_split_query:nnnN {#1} { } { } #2 } 8170
8171 \cs_new_protected:Npn \sys_split_query:nnN #1#2#3 8171
8172 { \sys_split_query:nnnN {#1} { } {#2} #3 } 8172
8173 \group_begin: 8173
8174 \tex_lccode:D `* = 13 \scan_stop: 8174
8175 \tex_lowercase:D 8175
8176 { 8176
8177 \group_end: 8177
8178 \cs_new_protected:Npn \sys_split_query:nnnN #1#2#3#4 8178
8179 { 8179
8180 \seq_clear:N #4 8180
8181 \sys_get_query:nnnN {#1} {#2} {#3} \l__sys_tmp_tl 8181
8182 \tl_if_empty:NF \l__sys_tmp_tl 8182
8183 { \seq_set_split:NnV #4 * \l__sys_tmp_tl } 8183
8184 } 8184
8185 } 8185
8186 \__sys_everyjob:n 8186
8187 { \cs_gset_eq:NN \g_file_curr_name_str \tex_jobname:D } 8187
8188 \cs_new_protected:Npn \sys_finalise: 8188

```

```

8189 {
8190     \__kernel_sys_everyjob:
8191     \tl_use:N \g__sys_finalise_tl
8192     \tl_gclear:N \g__sys_finalise_tl
8193 }
8194 \cs_new_protected:Npn \__sys_finalise:n #1
8195 { \tl_gput_right:Nn \g__sys_finalise_tl {#1} }
8196 \tl_new:N \g__sys_finalise_tl
8197 \__sys_finalise:n
8198 {
8199     \str_const:Ne \c_sys_output_str
8200     {
8201         \int_compare:nNnTF
8202             { \cs_if_exist_use:NF \tex_pdfoutput:D { 0 } } > { 0 }
8203             { pdf }
8204             { dvi }
8205     }
8206     \__sys_const:nn { sys_if_output_dvi }
8207     { \str_if_eq_p:Vn \c_sys_output_str { dvi } }
8208     \__sys_const:nn { sys_if_output_pdf }
8209     { \str_if_eq_p:Vn \c_sys_output_str { pdf } }
8210 }
8211 \tl_new:N \g__sys_backend_tl
8212 \__sys_finalise:n
8213 {
8214     \__kernel_tl_gset:Nx \g__sys_backend_tl
8215     {
8216         \sys_if_engine_xetex:TF
8217         { xetex }
8218         {
8219             \sys_if_output_pdf:TF
8220             {
8221                 \sys_if_engine_pdftex:TF
8222                 { pdftex }
8223                 { luatex }
8224             }
8225             { dvips }
8226         }
8227     }
8228 }
8229 \__sys_finalise:n
8230 {
8231     \cs_if_exist:NT \@classoptionslist
8232     {
8233         \cs_if_eq:NNF \@classoptionslist \scan_stop:
8234         {

```

```

8235 \clist_map_inline:Nn \@classoptionslist
8236 {
8237     \str_case:nnT {#1}
8238     {
8239         { dvipdfmx }
8240         { \tl_gset:Nn \g__sys_backend_tl { dvipdfmx } }
8241         { dvips }
8242         { \tl_gset:Nn \g__sys_backend_tl { dvips } }
8243         { dvisvgm }
8244         { \tl_gset:Nn \g__sys_backend_tl { dvisvgm } }
8245         { pdftex }
8246         { \tl_gset:Nn \g__sys_backend_tl { pdfmode } }
8247         { xetex }
8248         { \tl_gset:Nn \g__sys_backend_tl { xdvipdfmx } }
8249     }
8250     { \clist_remove_all:Nn \@unusedoptionlist {#1} }
8251 }
8252 }
8253 }
8254 }
8255 %% File: l3clist.dtx
8256 \cs_new_eq:NN \c_empty_clist \c_empty_tl
8257 \tl_new:N \l__clist_internal_clist
8258 \scan_new:N \s__clist_mark
8259 \scan_new:N \s__clist_stop
8260 \cs_new:Npn \__clist_use_none_delimit_by_s_mark:w #1 \s__clist_mark { }
8261 \cs_new:Npn \__clist_use_none_delimit_by_s_stop:w #1 \s__clist_stop { }
8262 \cs_new:Npn \__clist_use_i_delimit_by_s_stop:nw #1 #2 \s__clist_stop {#1}
8263 \cs_new_protected:Npn \__clist_tmp:w { }
8264 \cs_new:Npn \__clist_trim_next:w #1 ,
8265 {
8266     \tl_if_empty:oTF { \use_none:nn #1 ? }
8267     { \__clist_trim_next:w \prg_do_nothing: }
8268     { \tl_trim_spaces_apply:oN {#1} \exp_end: }
8269 }
8270 \cs_new:Npn \__clist_sanitize:n #1
8271 {
8272     \exp_after:wN \__clist_sanitize:Nn \exp_after:wN \c_empty_tl
8273     \exp:w \__clist_trim_next:w \prg_do_nothing:
8274     #1 , \s__clist_stop \prg_break: , \prg_break_point:
8275 }
8276 \cs_new:Npn \__clist_sanitize:Nn #1#2
8277 {
8278     \__clist_use_none_delimit_by_s_stop:w #2 \s__clist_stop
8279     #1 \__clist_wrap_item:w #2 ,
8280     \exp_after:wN \__clist_sanitize:Nn \exp_after:wN ,

```


8281	\exp:w __clist_trim_next:w \prg_do_nothing:	8281
8282	}	8282
8283	\prg_new_conditional:Npnn __clist_if_wrap:n #1 { TF }	8283
8284	{	8284
8285	\tl_if_empty:oTF	8285
8286	{	8286
8287	__clist_if_wrap:w	8287
8288	\s__clist_mark ? #1 ~ \s__clist_mark ? ~ #1	8288
8289	\s__clist_mark , ~ \s__clist_mark #1 ,	8289
8290	}	8290
8291	{	8291
8292	\tl_if_head_is_group:nTF { #1 { } }	8292
8293	{	8293
8294	\tl_if_empty:nTF {#1}	8294
8295	{ \prg_return_true: }	8295
8296	{	8296
8297	\tl_if_empty:oTF { \use_none:n #1}	8297
8298	{ \prg_return_true: }	8298
8299	{ \prg_return_false: }	8299
8300	}	8300
8301	}	8301
8302	{ \prg_return_false: }	8302
8303	}	8303
8304	{ \prg_return_true: }	8304
8305	}	8305
8306	\cs_new:Npn __clist_if_wrap:w #1 \s__clist_mark ? ~ #2 ~ \s__clist_mark #3 , { }	8306
8307	\cs_new:Npn __clist_wrap_item:w #1 ,	8307
8308	{ __clist_if_wrap:nTF {#1} { \exp_not:n { {#1} } } { \exp_not:n {#1} } }	8308
8309	\cs_new_eq:NN \clist_new:N \tl_new:N	8309
8310	\cs_new_eq:NN \clist_new:c \tl_new:c	8310
8311	\cs_new_protected:Npn \clist_const:Nn #1#2	8311
8312	{ \tl_const:Ne #1 { __clist_sanitize:n {#2} } }	8312
8313	\cs_generate_variant:Nn \clist_const:Nn { Ne , c , ce }	8313
8314	\cs_generate_variant:Nn \clist_const:Nn { Nx , cx }	8314
8315	\cs_new_eq:NN \clist_clear:N \tl_clear:N	8315
8316	\cs_new_eq:NN \clist_clear:c \tl_clear:c	8316
8317	\cs_new_eq:NN \clist_gclear:N \tl_gclear:N	8317
8318	\cs_new_eq:NN \clist_gclear:c \tl_gclear:c	8318
8319	\cs_new_eq:NN \clist_clear_new:N \tl_clear_new:N	8319
8320	\cs_new_eq:NN \clist_clear_new:c \tl_clear_new:c	8320
8321	\cs_new_eq:NN \clist_gclear_new:N \tl_gclear_new:N	8321
8322	\cs_new_eq:NN \clist_gclear_new:c \tl_gclear_new:c	8322
8323	\cs_new_eq:NN \clist_set_eq:NN \tl_set_eq:NN	8323
8324	\cs_new_eq:NN \clist_set_eq:Nc \tl_set_eq:Nc	8324
8325	\cs_new_eq:NN \clist_set_eq:cN \tl_set_eq:cN	8325
8326	\cs_new_eq:NN \clist_set_eq:cc \tl_set_eq:cc	8326

```
8327 \cs_new_eq:NN \clist_gset_eq:NN \tl_gset_eq:NN 8327
8328 \cs_new_eq:NN \clist_gset_eq:Nc \tl_gset_eq:Nc 8328
8329 \cs_new_eq:NN \clist_gset_eq:cN \tl_gset_eq:cN 8329
8330 \cs_new_eq:NN \clist_gset_eq:cc \tl_gset_eq:cc 8330
8331 \cs_new_protected:Npn \clist_set_from_seq:NN 8331
8332 { \__clist_set_from_seq:NNNN \clist_clear:N \__kernel_tl_set:Nx } 8332
8333 \cs_new_protected:Npn \clist_gset_from_seq:NN 8333
8334 { \__clist_set_from_seq:NNNN \clist_gclear:N \__kernel_tl_gset:Nx } 8334
8335 \cs_new_protected:Npn \__clist_set_from_seq:NNNN #1#2#3#4 8335
8336 { 8336
8337   \seq_if_empty:NTF #4 8337
8338   { #1 #3 } 8338
8339   { 8339
8340     #2 #3 8340
8341     { 8341
8342       \exp_after:wN \use_none:n \exp:w \exp_end_continue_f:w 8342
8343       \seq_map_function:NN #4 \__clist_set_from_seq:n 8343
8344     } 8344
8345   } 8345
8346 } 8346
8347 \cs_new:Npn \__clist_set_from_seq:n #1 8347
8348 { 8348
8349   , 8349
8350   \__clist_if_wrap:nTF {#1} 8350
8351   { \exp_not:n { {#1} } } 8351
8352   { \exp_not:n {#1} } 8352
8353 } 8353
8354 \cs_generate_variant:Nn \clist_set_from_seq:NN { Nc } 8354
8355 \cs_generate_variant:Nn \clist_set_from_seq:NN { c , cc } 8355
8356 \cs_generate_variant:Nn \clist_gset_from_seq:NN { Nc } 8356
8357 \cs_generate_variant:Nn \clist_gset_from_seq:NN { c , cc } 8357
8358 \cs_new_protected:Npn \clist_concat:NNN 8358
8359 { \__clist_concat:NNNN \__kernel_tl_set:Nx } 8359
8360 \cs_new_protected:Npn \clist_gconcat:NNN 8360
8361 { \__clist_concat:NNNN \__kernel_tl_gset:Nx } 8361
8362 \cs_new_protected:Npn \__clist_concat:NNNN #1#2#3#4 8362
8363 { 8363
8364   #1 #2 8364
8365   { 8365
8366     \exp_not:o #3 8366
8367     \clist_if_empty:NF #3 { \clist_if_empty:NF #4 { , } } 8367
8368     \exp_not:o #4 8368
8369   } 8369
8370 } 8370
8371 \cs_generate_variant:Nn \clist_concat:NNN { ccc } 8371
8372 \cs_generate_variant:Nn \clist_gconcat:NNN { ccc } 8372
```

```
8373 \prg_new_eq_conditional:NNn \clist_if_exist:N \cs_if_exist:N 8373
8374 { TF , T , F , p } 8374
8375 \prg_new_eq_conditional:NNn \clist_if_exist:c \cs_if_exist:c 8375
8376 { TF , T , F , p } 8376
8377 \cs_new_protected:Npn \clist_set:Nn #1#2 8377
8378 { \__kernel_tl_set:Nx #1 { \__clist_sanitiz:n {#2} } } 8378
8379 \cs_new_protected:Npn \clist_gset:Nn #1#2 8379
8380 { \__kernel_tl_gset:Nx #1 { \__clist_sanitiz:n {#2} } } 8380
8381 \cs_generate_variant:Nn \clist_set:Nn { NV , Ne , c , cV , ce } 8381
8382 \cs_generate_variant:Nn \clist_set:Nn { No , Nx , co , cx } 8382
8383 \cs_generate_variant:Nn \clist_gset:Nn { NV , Ne , c , cV , ce } 8383
8384 \cs_generate_variant:Nn \clist_gset:Nn { No , Nx , co , cx } 8384
8385 \cs_new_protected:Npn \clist_put_left:Nn 8385
8386 { \__clist_put_left:NNNn \clist_concat:NNN \clist_set:Nn } 8386
8387 \cs_new_protected:Npn \clist_gput_left:Nn 8387
8388 { \__clist_put_left:NNNn \clist_gconcat:NNN \clist_set:Nn } 8388
8389 \cs_new_protected:Npn \__clist_put_left:NNNn #1#2#3#4 8389
8390 { 8390
8391 #2 \l__clist_internal_clist {#4} 8391
8392 #1 #3 \l__clist_internal_clist #3 8392
8393 } 8393
8394 \cs_generate_variant:Nn \clist_put_left:Nn { NV , Nv , Ne , c , cV , cv , ce } 8394
8395 \cs_generate_variant:Nn \clist_put_left:Nn { No , Nx , co , cx } 8395
8396 \cs_generate_variant:Nn \clist_gput_left:Nn { NV , Nv , Ne , c , cV , cv , ce } 8396
8397 \cs_generate_variant:Nn \clist_gput_left:Nn { No , Nx , co , cx } 8397
8398 \cs_new_protected:Npn \clist_put_right:Nn 8398
8399 { \__clist_put_right:NNNn \clist_concat:NNN \clist_set:Nn } 8399
8400 \cs_new_protected:Npn \clist_gput_right:Nn 8400
8401 { \__clist_put_right:NNNn \clist_gconcat:NNN \clist_set:Nn } 8401
8402 \cs_new_protected:Npn \__clist_put_right:NNNn #1#2#3#4 8402
8403 { 8403
8404 #2 \l__clist_internal_clist {#4} 8404
8405 #1 #3 #3 \l__clist_internal_clist 8405
8406 } 8406
8407 \cs_generate_variant:Nn \clist_put_right:Nn 8407
8408 { NV , Nv , Ne , c , cV , cv , ce } 8408
8409 \cs_generate_variant:Nn \clist_put_right:Nn 8409
8410 { No , Nx , co , cx } 8410
8411 \cs_generate_variant:Nn \clist_gput_right:Nn 8411
8412 { NV , Nv , Ne , c , cV , cv , ce } 8412
8413 \cs_generate_variant:Nn \clist_gput_right:Nn 8413
8414 { No , Nx , co , cx } 8414
8415 \cs_new_protected:Npn \clist_get:NN #1#2 8415
8416 { 8416
8417 \if_meaning:w #1 \c_empty_clist 8417
8418 \tl_set:Nn #2 { \q_no_value } 8418
```

```

8419         \else:
8420             \exp_after:wN \__clist_get:wN #1 , \s__clist_stop #2
8421         \fi:
8422     }
8423 \cs_new_protected:Npn \__clist_get:wN #1 , #2 \s__clist_stop #3
8424 { \tl_set:Nn #3 {#1} }
8425 \cs_generate_variant:Nn \clist_get:NN { c }
8426 \cs_new_protected:Npn \clist_pop:NN
8427 { \__clist_pop:NNN \__kernel_tl_set:Nx }
8428 \cs_new_protected:Npn \clist_gpop:NN
8429 { \__clist_pop:NNN \__kernel_tl_gset:Nx }
8430 \cs_new_protected:Npn \__clist_pop:NNN #1#2#3
8431 {
8432     \if_meaning:w #2 \c_empty_clist
8433         \tl_set:Nn #3 { \q_no_value }
8434     \else:
8435         \exp_after:wN \__clist_pop:wwNNN #2 , \s__clist_mark \s__clist_stop #1#2#3
8436     \fi:
8437 }
8438 \cs_new_protected:Npn \__clist_pop:wwNNN #1 , #2 \s__clist_stop #3#4#5
8439 {
8440     \tl_set:Nn #5 {#1}
8441     #3 #4
8442     {
8443         \__clist_pop:wN \prg_do_nothing:
8444         #2 \exp_not:o
8445         , \s__clist_mark \use_none:n
8446         \s__clist_stop
8447     }
8448 }
8449 \cs_new:Npn \__clist_pop:wN #1 , \s__clist_mark #2 #3 \s__clist_stop { #2 {#1} }
8450 \cs_generate_variant:Nn \clist_pop:NN { c }
8451 \cs_generate_variant:Nn \clist_gpop:NN { c }
8452 \prg_new_protected_conditional:Npnn \clist_get:NN #1#2 { T , F , TF }
8453 {
8454     \if_meaning:w #1 \c_empty_clist
8455         \prg_return_false:
8456     \else:
8457         \exp_after:wN \__clist_get:wN #1 , \s__clist_stop #2
8458         \prg_return_true:
8459     \fi:
8460 }
8461 \prg_generate_conditional_variant:Nnn \clist_get:NN { c } { T , F , TF }
8462 \prg_new_protected_conditional:Npnn \clist_pop:NN #1#2 { T , F , TF }
8463 { \__clist_pop_TF:NNN \__kernel_tl_set:Nx #1 #2 }
8464 \prg_new_protected_conditional:Npnn \clist_gpop:NN #1#2 { T , F , TF }

```

```
8465 { \__clist_pop_TF:NNN \__kernel_tl_gset:Nx #1 #2 } 8465
8466 \cs_new_protected:Npn \__clist_pop_TF:NNN #1#2#3 8466
8467 { 8467
8468   \if_meaning:w #2 \c_empty_clist 8468
8469   \prg_return_false: 8469
8470   \else: 8470
8471     \exp_after:wN \__clist_pop:wwNNN #2 , \s__clist_mark \s__clist_stop #1#2#3 8471
8472     \prg_return_true: 8472
8473   \fi: 8473
8474 } 8474
8475 \prg_generate_conditional_variant:Nnn \clist_pop:NN { c } { T , F , TF } 8475
8476 \prg_generate_conditional_variant:Nnn \clist_gpop:NN { c } { T , F , TF } 8476
8477 \cs_new_eq:NN \clist_push:Nn \clist_put_left:Nn 8477
8478 \cs_generate_variant:Nn \clist_push:Nn { NV , No , Nx , c , cV , co , cx } 8478
8479 \cs_new_eq:NN \clist_gpush:Nn \clist_gput_left:Nn 8479
8480 \cs_generate_variant:Nn \clist_gpush:Nn { NV , No , Nx , c , cV , co , cx } 8480
8481 \clist_new:N \l__clist_internal_remove_clist 8481
8482 \seq_new:N \l__clist_internal_remove_seq 8482
8483 \cs_new_protected:Npn \clist_remove_duplicates:N 8483
8484 { \__clist_remove_duplicates:NN \clist_set_eq:NN } 8484
8485 \cs_new_protected:Npn \clist_gremove_duplicates:N 8485
8486 { \__clist_remove_duplicates:NN \clist_gset_eq:NN } 8486
8487 \cs_new_protected:Npn \__clist_remove_duplicates:NN #1#2 8487
8488 { 8488
8489   \clist_clear:N \l__clist_internal_remove_clist 8489
8490   \clist_map_inline:Nn #2 8490
8491   { 8491
8492     \clist_if_in:NnF \l__clist_internal_remove_clist {##1} 8492
8493     { 8493
8494       \tl_put_right:Ne \l__clist_internal_remove_clist 8494
8495       { 8495
8496         \clist_if_empty:NF \l__clist_internal_remove_clist { , } 8496
8497         \__clist_if_wrap:nTF {##1} { \exp_not:n { {##1} } } { \exp_not:n {##1} } 8497
8498       } 8498
8499     } 8499
8500   } 8500
8501   #1 #2 \l__clist_internal_remove_clist 8501
8502 } 8502
8503 \cs_generate_variant:Nn \clist_remove_duplicates:N { c } 8503
8504 \cs_generate_variant:Nn \clist_gremove_duplicates:N { c } 8504
8505 \cs_new_protected:Npn \clist_remove_all:Nn 8505
8506 { \__clist_remove_all:NNNn \clist_set_from_seq:NN \__kernel_tl_set:Nx } 8506
8507 \cs_new_protected:Npn \clist_gremove_all:Nn 8507
8508 { \__clist_remove_all:NNNn \clist_gset_from_seq:NN \__kernel_tl_gset:Nx } 8508
8509 \cs_new_protected:Npn \__clist_remove_all:NNNn #1#2#3#4 8509
8510 { 8510
```

```

8511 \__clist_if_wrap:nTF {#4}
8512 {
8513     \seq_set_from_clist:NN \l__clist_internal_remove_seq #3
8514     \seq_remove_all:Nn \l__clist_internal_remove_seq {#4}
8515     #1 #3 \l__clist_internal_remove_seq
8516 }
8517 {
8518     \cs_set:Npn \__clist_tmp:w ##1 , #4 ,
8519     {
8520         ##1
8521         , \s__clist_mark , \__clist_use_none_delimit_by_s_stop:w ,
8522         \__clist_remove_all:
8523     }
8524     #2 #3
8525     {
8526         \exp_after:wN \__clist_remove_all:
8527         #3 , \s__clist_mark , #4 , \s__clist_stop
8528     }
8529     \clist_if_empty:NF #3
8530     {
8531         #2 #3
8532         {
8533             \exp_args:No \exp_not:o
8534             { \exp_after:wN \use_none:n #3 }
8535         }
8536     }
8537 }
8538 }
8539 \cs_new:Npn \__clist_remove_all:
8540 { \exp_after:wN \__clist_remove_all:w \__clist_tmp:w , }
8541 \cs_new:Npn \__clist_remove_all:w #1 , \s__clist_mark , #2 , { \exp_not:n {#1} }
8542 \cs_generate_variant:Nn \clist_remove_all:Nn { c , NV , cV , Ne , ce }
8543 \cs_generate_variant:Nn \clist_gremove_all:Nn { c , NV , cV , Ne , ce }
8544 \cs_new_protected:Npn \clist_reverse:N #1
8545 { \__kernel_tl_set:Nx #1 { \exp_args:No \clist_reverse:n {#1} } }
8546 \cs_new_protected:Npn \clist_greverse:N #1
8547 { \__kernel_tl_gset:Nx #1 { \exp_args:No \clist_reverse:n {#1} } }
8548 \cs_generate_variant:Nn \clist_reverse:N { c }
8549 \cs_generate_variant:Nn \clist_greverse:N { c }
8550 \cs_new:Npn \clist_reverse:n #1
8551 {
8552     \__clist_reverse:wwNww ? #1 ,
8553     \s__clist_mark \__clist_reverse:wwNww ! ,
8554     \s__clist_mark \__clist_reverse_end:ww
8555     \s__clist_stop ? \s__clist_mark
8556 }

```


8557	\cs_new:Npn __clist_reverse:wwNww	8557
8558	#1 , #2 \s__clist_mark #3 #4 \s__clist_stop ? #5 \s__clist_mark	8558
8559	{ #3 ? #2 \s__clist_mark #3 #4 \s__clist_stop #1 , #5 \s__clist_mark }	8559
8560	\cs_new:Npn __clist_reverse_end:ww #1 ! #2 , \s__clist_mark	8560
8561	{ \exp_not:o { \use_none:n #2 } }	8561
8562	\prg_new_eq_conditional:NNn \clist_if_empty:N \tl_if_empty:N	8562
8563	{ p , T , F , TF }	8563
8564	\prg_new_eq_conditional:NNn \clist_if_empty:c \tl_if_empty:c	8564
8565	{ p , T , F , TF }	8565
8566	\prg_new_conditional:Npnn \clist_if_empty:n #1 { p , T , F , TF }	8566
8567	{	8567
8568	__clist_if_empty_n:w ? #1	8568
8569	, \s__clist_mark \prg_return_false:	8569
8570	, \s__clist_mark \prg_return_true:	8570
8571	\s__clist_stop	8571
8572	}	8572
8573	\cs_new:Npn __clist_if_empty_n:w #1 ,	8573
8574	{	8574
8575	\tl_if_empty:oTF { \use_none:nn #1 ? }	8575
8576	{ __clist_if_empty_n:w ? }	8576
8577	{ __clist_if_empty_n:wNw }	8577
8578	}	8578
8579	\cs_new:Npn __clist_if_empty_n:wNw #1 \s__clist_mark #2#3 \s__clist_stop {#2}	8579
8580	\prg_new_protected_conditional:Npnn \clist_if_in:Nn #1#2 { T , F , TF }	8580
8581	{	8581
8582	\exp_args:No __clist_if_in_return:nnN #1 {#2} #1	8582
8583	}	8583
8584	\prg_new_protected_conditional:Npnn \clist_if_in:nn #1#2 { T , F , TF }	8584
8585	{	8585
8586	\clist_set:Nn \l__clist_internal_clist {#1}	8586
8587	\exp_args:No __clist_if_in_return:nnN \l__clist_internal_clist {#2}	8587
8588	\l__clist_internal_clist	8588
8589	}	8589
8590	\cs_new_protected:Npn __clist_if_in_return:nnN #1#2#3	8590
8591	{	8591
8592	__clist_if_wrap:nTF {#2}	8592
8593	{	8593
8594	\cs_set:Npe __clist_tmp:w ##1	8594
8595	{	8595
8596	\exp_not:N \tl_if_eq:nnT {##1}	8596
8597	\exp_not:n	8597
8598	{	8598
8599	{#2}	8599
8600	{ \clist_map_break:n { \prg_return_true: \use_none:n } }	8600
8601	}	8601
8602	}	8602

```

8603         \clist_map_function:NN #3 \__clist_tmp:w      8603
8604         \prg_return_false:                             8604
8605     }                                                  8605
8606     {                                                  8606
8607         \cs_set:Npn \__clist_tmp:w ##1 ,#2, { }      8607
8608         \tl_if_empty:oTF                               8608
8609             { \__clist_tmp:w ,#1, {} {} ,#2, }      8609
8610             { \prg_return_false: } { \prg_return_true: } 8610
8611     }                                                  8611
8612 }                                                  8612
8613 \prg_generate_conditional_variant:Nnn \clist_if_in:Nn 8613
8614     { NV , No , c , cV , co } { T , F , TF }      8614
8615 \prg_generate_conditional_variant:Nnn \clist_if_in:nn 8615
8616     { nV , no } { T , F , TF }                    8616
8617 \cs_new:Npn \clist_map_function:NN #1#2            8617
8618     {                                                8618
8619         \clist_if_empty:NF #1                      8619
8620         {                                           8620
8621             \exp_after:wN \__clist_map_function:Nw \exp_after:wN #2 #1 , 8621
8622             \s__clist_stop , \s__clist_stop , \s__clist_stop , \s__clist_stop , 8622
8623             \s__clist_stop , \s__clist_stop , \s__clist_stop , \s__clist_stop , 8623
8624             \prg_break_point:Nn \clist_map_break: { } 8624
8625         }                                           8625
8626     }                                               8626
8627 \cs_new:Npn \__clist_map_function:Nw #1 #2, #3, #4, #5, #6, #7, #8, #9, 8627
8628     {                                               8628
8629         \__clist_use_none_delimit_by_s_stop:w      8629
8630         #9 \__clist_map_function_end:w \s__clist_stop 8630
8631         #1 {#2} #1 {#3} #1 {#4} #1 {#5} #1 {#6} #1 {#7} #1 {#8} #1 {#9} 8631
8632         \__clist_map_function:Nw #1                8632
8633     }                                               8633
8634 \cs_new:Npn \__clist_map_function_end:w \s__clist_stop #1#2 8634
8635     {                                               8635
8636         \__clist_use_none_delimit_by_s_stop:w #2 \clist_map_break: \s__clist_stop 8636
8637         #1 {#2}                                     8637
8638         \__clist_map_function_end:w \s__clist_stop 8638
8639     }                                               8639
8640 \cs_generate_variant:Nn \clist_map_function:NN { c } 8640
8641 \cs_new:Npn \clist_map_function:nN #1#2            8641
8642     {                                                8642
8643         \exp_after:wN \__clist_map_function_n:Nn \exp_after:wN #2      8643
8644         \exp:w \__clist_trim_next:w \prg_do_nothing: #1 ,              8644
8645         \s__clist_stop \clist_map_break: ,                             8645
8646         \prg_break_point:Nn \clist_map_break: { }                      8646
8647     }                                               8647
8648 \cs_generate_variant:Nn \clist_map_function:nN { e } 8648

```

```
8649 \cs_new:Npn \__clist_map_function_n:Nn #1 #2 8649
8650 { 8650
8651     \__clist_use_none_delimit_by_s_stop:w #2 \s__clist_stop 8651
8652     \__clist_map_unbrace:wn #2 , #1 8652
8653     \exp_after:wN \__clist_map_function_n:Nn \exp_after:wN #1 8653
8654     \exp:w \__clist_trim_next:w \prg_do_nothing: 8654
8655 } 8655
8656 \cs_new:Npn \__clist_map_unbrace:wn #1, #2 { #2 {#1} } 8656
8657 \cs_new_protected:Npn \clist_map_inline:Nn #1#2 8657
8658 { 8658
8659     \clist_if_empty:NF #1 8659
8660     { 8660
8661         \int_gincr:N \g__kernel_prg_map_int 8661
8662         \cs_gset_protected:cpn 8662
8663         { __clist_map_ \int_use:N \g__kernel_prg_map_int :w } ##1 {#2} 8663
8664         \exp_last_unbraced:Nco \__clist_map_function:Nw 8664
8665         { __clist_map_ \int_use:N \g__kernel_prg_map_int :w } 8665
8666         #1 , 8666
8667         \s__clist_stop , \s__clist_stop , \s__clist_stop , \s__clist_stop , 8667
8668         \s__clist_stop , \s__clist_stop , \s__clist_stop , \s__clist_stop , 8668
8669         \prg_break_point:Nn \clist_map_break: 8669
8670         { \int_gdecr:N \g__kernel_prg_map_int } 8670
8671     } 8671
8672 } 8672
8673 \cs_new_protected:Npn \clist_map_inline:nn #1 8673
8674 { 8674
8675     \clist_set:Nn \l__clist_internal_clist {#1} 8675
8676     \clist_map_inline:Nn \l__clist_internal_clist 8676
8677 } 8677
8678 \cs_generate_variant:Nn \clist_map_inline:Nn { c } 8678
8679 \cs_new_protected:Npn \clist_map_variable:NNn #1#2#3 8679
8680 { \clist_map_tokens:Nn #1 { \__clist_map_variable:Nnn #2 {#3} } } 8680
8681 \cs_generate_variant:Nn \clist_map_variable:NNn { c } 8681
8682 \cs_new_protected:Npn \__clist_map_variable:Nnn #1#2#3 8682
8683 { \tl_set:Nn #1 {#3} #2 } 8683
8684 \cs_new_protected:Npn \clist_map_variable:nNn #1 8684
8685 { 8685
8686     \clist_set:Nn \l__clist_internal_clist {#1} 8686
8687     \clist_map_variable:NNn \l__clist_internal_clist 8687
8688 } 8688
8689 \cs_new:Npn \clist_map_tokens:Nn #1#2 8689
8690 { 8690
8691     \clist_if_empty:NF #1 8691
8692     { 8692
8693         \exp_last_unbraced:Nno \__clist_map_tokens:nw {#2} #1 , 8693
8694         \s__clist_stop , \s__clist_stop , \s__clist_stop , \s__clist_stop , 8694
```

```
8695 \s__clist_stop , \s__clist_stop , \s__clist_stop , \s__clist_stop , 8695
8696 \prg_break_point:Nn \clist_map_break: { } 8696
8697 } 8697
8698 } 8698
8699 \cs_new:Npn \__clist_map_tokens:nw #1 #2, #3, #4, #5, #6, #7, #8, #9, 8699
8700 { 8700
8701 \__clist_use_none_delimit_by_s_stop:w 8701
8702 #9 \__clist_map_tokens_end:w \s__clist_stop 8702
8703 \use:n {#1} {#2} \use:n {#1} {#3} \use:n {#1} {#4} \use:n {#1} {#5} 8703
8704 \use:n {#1} {#6} \use:n {#1} {#7} \use:n {#1} {#8} \use:n {#1} {#9} 8704
8705 \__clist_map_tokens:nw {#1} 8705
8706 } 8706
8707 \cs_new:Npn \__clist_map_tokens_end:w \s__clist_stop \use:n #1#2 8707
8708 { 8708
8709 \__clist_use_none_delimit_by_s_stop:w #2 \clist_map_break: \s__clist_stop 8709
8710 #1 {#2} 8710
8711 \__clist_map_tokens_end:w \s__clist_stop 8711
8712 } 8712
8713 \cs_generate_variant:Nn \clist_map_tokens:Nn { c } 8713
8714 \cs_new:Npn \clist_map_tokens:nn #1#2 8714
8715 { 8715
8716 \__clist_map_tokens_n:nw {#2} 8716
8717 \prg_do_nothing: #1 , \s__clist_stop \clist_map_break: , 8717
8718 \prg_break_point:Nn \clist_map_break: { } 8718
8719 } 8719
8720 \cs_new:Npn \__clist_map_tokens_n:nw #1#2 , 8720
8721 { 8721
8722 \tl_if_empty:oF { \use_none:nn #2 ? } 8722
8723 { 8723
8724 \__clist_use_none_delimit_by_s_stop:w #2 \s__clist_stop 8724
8725 \tl_trim_spaces_apply:oN {#2} \use_ii_i:nn 8725
8726 \__clist_map_unbrace:wn , {#1} 8726
8727 } 8727
8728 \__clist_map_tokens_n:nw {#1} \prg_do_nothing: 8728
8729 } 8729
8730 \cs_new:Npn \clist_map_break: 8730
8731 { \prg_map_break:Nn \clist_map_break: { } } 8731
8732 \cs_new:Npn \clist_map_break:n 8732
8733 { \prg_map_break:Nn \clist_map_break: } 8733
8734 \cs_new:Npn \clist_count:N #1 8734
8735 { 8735
8736 \int_eval:n 8736
8737 { 8737
8738 0 8738
8739 \clist_map_function:NN #1 \__clist_count:n 8739
8740 } 8740
```

8741	}	8741
8742	\cs_generate_variant:Nn \clist_count:N { c }	8742
8743	\cs_new:Npn __clist_count:n #1 { + 1 }	8743
8744	\cs_set_protected:Npn __clist_tmp:w #1	8744
8745	{	8745
8746	\cs_new:Npn \clist_count:n ##1	8746
8747	{	8747
8748	\int_eval:n	8748
8749	{	8749
8750	0	8750
8751	__clist_count:w #1	8751
8752	##1 , \s__clist_stop \prg_break: , \prg_break_point:	8752
8753	}	8753
8754	}	8754
8755	\cs_new:Npn __clist_count:w ##1 ,	8755
8756	{	8756
8757	__clist_use_none_delimit_by_s_stop:w ##1 \s__clist_stop	8757
8758	\tl_if_blank:nF {##1} { + 1 }	8758
8759	__clist_count:w #1	8759
8760	}	8760
8761	}	8761
8762	\exp_args:No __clist_tmp:w \c_space_tl	8762
8763	\cs_generate_variant:Nn \clist_count:n { e }	8763
8764	\cs_new:Npn \clist_use:Nnnn #1#2#3#4	8764
8765	{	8765
8766	\clist_if_exist:NTF #1	8766
8767	{	8767
8768	\int_case:nnF { \clist_count:N #1 }	8768
8769	{	8769
8770	{ 0 } { }	8770
8771	{ 1 } { \exp_after:wN __clist_use:wwn #1 , , { } }	8771
8772	{ 2 } { \exp_after:wN __clist_use:wwn #1 , {#2} }	8772
8773	}	8773
8774	{	8774
8775	\exp_after:wN __clist_use:nwwwwnwn	8775
8776	\exp_after:wN { \exp_after:wN } #1 ,	8776
8777	\s__clist_mark , { __clist_use:nwwwwnwn {#3} }	8777
8778	\s__clist_mark , { __clist_use:nwwn {#4} }	8778
8779	\s__clist_stop { }	8779
8780	}	8780
8781	}	8781
8782	{	8782
8783	\msg_expandable_error:nnn	8783
8784	{ kernel } { bad-variable } {#1}	8784
8785	}	8785
8786	}	8786

8787	\cs_generate_variant:Nn \clist_use:Nnnn { c }	8787
8788	\cs_new:Npn __clist_use:wwn #1 , #2 , #3 { \exp_not:n { #1 #3 #2 } }	8788
8789	\cs_new:Npn __clist_use:nwwwnwn	8789
8790	#1#2 , #3 , #4 , #5 \s__clist_mark , #6#7 \s__clist_stop #8	8790
8791	{ #6 {#3} , {#4} , #5 \s__clist_mark , {#6} #7 \s__clist_stop { #8 #1 #2 } }	8791
8792	\cs_new:Npn __clist_use:nwwn #1#2 , #3 \s__clist_stop #4	8792
8793	{ \exp_not:n { #4 #1 #2 } }	8793
8794	\cs_new:Npn \clist_use:Nn #1#2	8794
8795	{ \clist_use:Nnnn #1 {#2} {#2} {#2} }	8795
8796	\cs_generate_variant:Nn \clist_use:Nn { c }	8796
8797	\cs_new_eq:NN \clist_use:N \tl_use:N	8797
8798	\cs_generate_variant:Nn \clist_use:N { c }	8798
8799	\cs_new:Npn \clist_use:nnnn #1#2#3#4	8799
8800	{	8800
8801	__clist_use:Nw __clist_use_none_delimit_by_s_stop:w	8801
8802	__clist_use:Nw __clist_use_one:w	8802
8803	__clist_use:Nw __clist_use_end:w	8803
8804	__clist_use_more:w ;	8804
8805	{#2} {#3} {#4} ;	8805
8806	\prg_do_nothing: #1 , \s__clist_mark ,	8806
8807	\s__clist_stop	8807
8808	}	8808
8809	\cs_new:Npn __clist_use:Nw #1#2 ; #3 ; #4 ,	8809
8810	{	8810
8811	\tl_if_empty:oTF { \use_none:nn #4 ? }	8811
8812	{ __clist_use:Nw #1#2 ; }	8812
8813	{	8813
8814	__clist_use_none_delimit_by_s_mark:w #4 #1 \s__clist_mark	8814
8815	\tl_trim_spaces_apply:oN {#4} \use_ii_i:nn	8815
8816	__clist_map_unbrace:wn , { #2 ; }	8816
8817	}	8817
8818	#3 ; \prg_do_nothing:	8818
8819	}	8819
8820	\cs_new:Npn __clist_use_one:w \s__clist_mark #1 , #2#3#4 \s__clist_stop	8820
8821	{ \exp_not:n {#3} }	8821
8822	\cs_new:Npn __clist_use_end:w	8822
8823	\s__clist_mark #1 , #2#3#4#5#6 \s__clist_stop	8823
8824	{ \exp_not:n { #4 #5 #3 } }	8824
8825	\cs_new:Npn __clist_use_more:w ; #1#2#3#4#5#6 ;	8825
8826	{	8826
8827	\exp_not:n { #3 #5 }	8827
8828	__clist_use:Nw __clist_use_end:w __clist_use_more:w ;	8828
8829	{#1} {#2} {#6} {#5} {#6} ;	8829
8830	}	8830
8831	\cs_new:Npn \clist_use:nn #1#2 { \clist_use:nnnn {#1} {#2} {#2} {#2} }	8831
8832	\cs_new:Npn \clist_item:Nn #1#2	8832


```

8833 {
8834     \__clist_item:ffoN
8835     { \clist_count:N #1 }
8836     { \int_eval:n {#2} }
8837     #1
8838     \__clist_item_N_loop:nw
8839 }
8840 \cs_new:Npn \__clist_item:nnnN #1#2#3#4
8841 {
8842     \int_compare:nNnTF {#2} < 0
8843     {
8844         \int_compare:nNnTF {#2} < { - #1 }
8845         { \__clist_use_none_delimit_by_s_stop:w }
8846         { \exp_args:Nf #4 { \int_eval:n { #2 + 1 + #1 } } }
8847     }
8848     {
8849         \int_compare:nNnTF {#2} > {#1}
8850         { \__clist_use_none_delimit_by_s_stop:w }
8851         { #4 {#2} }
8852     }
8853     { } , #3 , \s__clist_stop
8854 }
8855 \cs_generate_variant:Nn \__clist_item:nnnN { ffo, ff }
8856 \cs_new:Npn \__clist_item_N_loop:nw #1 #2,
8857 {
8858     \int_compare:nNnTF {#1} = 0
8859     { \__clist_use_i_delimit_by_s_stop:nw { \exp_not:n {#2} } }
8860     { \exp_args:Nf \__clist_item_N_loop:nw { \int_eval:n { #1 - 1 } } }
8861 }
8862 \cs_generate_variant:Nn \clist_item:Nn { c }
8863 \cs_new:Npn \clist_item:nn #1#2
8864 {
8865     \__clist_item:ffnN
8866     { \clist_count:n {#1} }
8867     { \int_eval:n {#2} }
8868     {#1}
8869     \__clist_item_n:nw
8870 }
8871 \cs_generate_variant:Nn \clist_item:nn { e }
8872 \cs_new:Npn \__clist_item_n:nw #1
8873 { \__clist_item_n_loop:nw {#1} \prg_do_nothing: }
8874 \cs_new:Npn \__clist_item_n_loop:nw #1 #2,
8875 {
8876     \exp_args:No \tl_if_blank:nTF {#2}
8877     { \__clist_item_n_loop:nw {#1} \prg_do_nothing: }
8878     {

```

```

8879         \int_compare:nNnTF {#1} = 0
8880         { \exp_args:No \__clist_item_n_end:n {#2} }
8881         {
8882             \exp_args:Nf \__clist_item_n_loop:nw
8883             { \int_eval:n { #1 - 1 } }
8884             \prg_do_nothing:
8885         }
8886     }
8887 }
8888 \cs_new:Npn \__clist_item_n_end:n #1 #2 \s__clist_stop
8889 { \tl_trim_spaces_apply:nN {#1} \__clist_item_n_strip:n }
8890 \cs_new:Npn \__clist_item_n_strip:n #1 { \__clist_item_n_strip:w #1 , }
8891 \cs_new:Npn \__clist_item_n_strip:w #1 , { \exp_not:n {#1} }
8892 \cs_new:Npn \clist_rand_item:n #1
8893 { \exp_args:Nf \__clist_rand_item:nn { \clist_count:n {#1} } {#1} }
8894 \cs_new:Npn \__clist_rand_item:nn #1#2
8895 {
8896     \int_compare:nNnF {#1} = 0
8897     { \clist_item:nn {#2} { \int_rand:nn { 1 } {#1} } }
8898 }
8899 \cs_new:Npn \clist_rand_item:N #1
8900 {
8901     \clist_if_empty:NF #1
8902     { \clist_item:Nn #1 { \int_rand:nn { 1 } { \clist_count:N #1 } } }
8903 }
8904 \cs_generate_variant:Nn \clist_rand_item:N { c }
8905 \cs_new_protected:Npn \clist_show:N { \__clist_show:NN \msg_show:nneeee }
8906 \cs_generate_variant:Nn \clist_show:N { c }
8907 \cs_new_protected:Npn \clist_log:N { \__clist_show:NN \msg_log:nneeee }
8908 \cs_generate_variant:Nn \clist_log:N { c }
8909 \cs_new_protected:Npn \__clist_show:NN #1#2
8910 {
8911     \__kernel_chk_tl_type:NnnT #2 { clist } { \exp_not:o #2 }
8912     {
8913         \int_compare:nNnTF { \clist_count:N #2 }
8914         = { \exp_args:No \clist_count:n #2 }
8915         {
8916             #1 { clist } { show }
8917             { \token_to_str:N #2 }
8918             { \clist_map_function:NN #2 \msg_show_item:n }
8919             { } { }
8920         }
8921         {
8922             \msg_error:nnee { clist } { non-clist }
8923             { \token_to_str:N #2 } { \tl_to_str:N #2 }
8924         }

```

```
8925     }
8926 }
8927 \cs_new_protected:Npn \clist_show:n { \__clist_show:Nn \msg_show:nneeee }
8928 \cs_new_protected:Npn \clist_log:n { \__clist_show:Nn \msg_log:nneeee }
8929 \cs_new_protected:Npn \__clist_show:Nn #1#2
8930 {
8931     #1 { clist } { show }
8932     { } { \clist_map_function:nN {#2} \msg_show_item:n } { } { }
8933 }
8934 \clist_new:N \l_tmpa_clist
8935 \clist_new:N \l_tmpb_clist
8936 \clist_new:N \g_tmpa_clist
8937 \clist_new:N \g_tmpb_clist
8938 %% File: l3token.dtx
8939 \scan_new:N \s__char_stop
8940 \quark_new:N \q__char_no_value
8941 \__kernel_quark_new_conditional:Nn \__char_quark_if_no_value:N { TF }
8942 \cs_new_protected:Npn \char_set_catcode:nn #1#2
8943 { \tex_catcode:D \int_eval:n {#1} = \int_eval:n {#2} \exp_stop_f: }
8944 \cs_new:Npn \char_value_catcode:n #1
8945 { \tex_the:D \tex_catcode:D \int_eval:n {#1} \exp_stop_f: }
8946 \cs_new_protected:Npn \char_show_value_catcode:n #1
8947 { \exp_args:Nf \tl_show:n { \char_value_catcode:n {#1} } }
8948 \cs_new_protected:Npn \char_set_catcode_escape:N #1
8949 { \char_set_catcode:nn {`#1} { 0 } }
8950 \cs_new_protected:Npn \char_set_catcode_group_begin:N #1
8951 { \char_set_catcode:nn {`#1} { 1 } }
8952 \cs_new_protected:Npn \char_set_catcode_group_end:N #1
8953 { \char_set_catcode:nn {`#1} { 2 } }
8954 \cs_new_protected:Npn \char_set_catcode_math_toggle:N #1
8955 { \char_set_catcode:nn {`#1} { 3 } }
8956 \cs_new_protected:Npn \char_set_catcode_alignment:N #1
8957 { \char_set_catcode:nn {`#1} { 4 } }
8958 \cs_new_protected:Npn \char_set_catcode_end_line:N #1
8959 { \char_set_catcode:nn {`#1} { 5 } }
8960 \cs_new_protected:Npn \char_set_catcode_parameter:N #1
8961 { \char_set_catcode:nn {`#1} { 6 } }
8962 \cs_new_protected:Npn \char_set_catcode_math_superscript:N #1
8963 { \char_set_catcode:nn {`#1} { 7 } }
8964 \cs_new_protected:Npn \char_set_catcode_math_subscript:N #1
8965 { \char_set_catcode:nn {`#1} { 8 } }
8966 \cs_new_protected:Npn \char_set_catcode_ignore:N #1
8967 { \char_set_catcode:nn {`#1} { 9 } }
8968 \cs_new_protected:Npn \char_set_catcode_space:N #1
8969 { \char_set_catcode:nn {`#1} { 10 } }
8970 \cs_new_protected:Npn \char_set_catcode_letter:N #1
```

8971	{ \char_set_catcode:nn { `#1 } { 11 } }	8971
8972	\cs_new_protected:Npn \char_set_catcode_other:N #1	8972
8973	{ \char_set_catcode:nn { `#1 } { 12 } }	8973
8974	\cs_new_protected:Npn \char_set_catcode_active:N #1	8974
8975	{ \char_set_catcode:nn { `#1 } { 13 } }	8975
8976	\cs_new_protected:Npn \char_set_catcode_comment:N #1	8976
8977	{ \char_set_catcode:nn { `#1 } { 14 } }	8977
8978	\cs_new_protected:Npn \char_set_catcode_invalid:N #1	8978
8979	{ \char_set_catcode:nn { `#1 } { 15 } }	8979
8980	\cs_new_protected:Npn \char_set_catcode_escape:n #1	8980
8981	{ \char_set_catcode:nn {#1} { 0 } }	8981
8982	\cs_new_protected:Npn \char_set_catcode_group_begin:n #1	8982
8983	{ \char_set_catcode:nn {#1} { 1 } }	8983
8984	\cs_new_protected:Npn \char_set_catcode_group_end:n #1	8984
8985	{ \char_set_catcode:nn {#1} { 2 } }	8985
8986	\cs_new_protected:Npn \char_set_catcode_math_toggle:n #1	8986
8987	{ \char_set_catcode:nn {#1} { 3 } }	8987
8988	\cs_new_protected:Npn \char_set_catcode_alignment:n #1	8988
8989	{ \char_set_catcode:nn {#1} { 4 } }	8989
8990	\cs_new_protected:Npn \char_set_catcode_end_line:n #1	8990
8991	{ \char_set_catcode:nn {#1} { 5 } }	8991
8992	\cs_new_protected:Npn \char_set_catcode_parameter:n #1	8992
8993	{ \char_set_catcode:nn {#1} { 6 } }	8993
8994	\cs_new_protected:Npn \char_set_catcode_math_superscript:n #1	8994
8995	{ \char_set_catcode:nn {#1} { 7 } }	8995
8996	\cs_new_protected:Npn \char_set_catcode_math_subscript:n #1	8996
8997	{ \char_set_catcode:nn {#1} { 8 } }	8997
8998	\cs_new_protected:Npn \char_set_catcode_ignore:n #1	8998
8999	{ \char_set_catcode:nn {#1} { 9 } }	8999
9000	\cs_new_protected:Npn \char_set_catcode_space:n #1	9000
9001	{ \char_set_catcode:nn {#1} { 10 } }	9001
9002	\cs_new_protected:Npn \char_set_catcode_letter:n #1	9002
9003	{ \char_set_catcode:nn {#1} { 11 } }	9003
9004	\cs_new_protected:Npn \char_set_catcode_other:n #1	9004
9005	{ \char_set_catcode:nn {#1} { 12 } }	9005
9006	\cs_new_protected:Npn \char_set_catcode_active:n #1	9006
9007	{ \char_set_catcode:nn {#1} { 13 } }	9007
9008	\cs_new_protected:Npn \char_set_catcode_comment:n #1	9008
9009	{ \char_set_catcode:nn {#1} { 14 } }	9009
9010	\cs_new_protected:Npn \char_set_catcode_invalid:n #1	9010
9011	{ \char_set_catcode:nn {#1} { 15 } }	9011
9012	\cs_new_protected:Npn \char_set_mathcode:nn #1#2	9012
9013	{ \tex_mathcode:D \int_eval:n {#1} = \int_eval:n {#2} \exp_stop_f: }	9013
9014	\cs_new:Npn \char_value_mathcode:n #1	9014
9015	{ \tex_the:D \tex_mathcode:D \int_eval:n {#1} \exp_stop_f: }	9015
9016	\cs_new_protected:Npn \char_show_value_mathcode:n #1	9016

```
9017 { \exp_args:Nf \tl_show:n { \char_value_mathcode:n {#1} } } 9017
9018 \cs_new_protected:Npn \char_set_lccode:nn #1#2 9018
9019 { \tex_lccode:D \int_eval:n {#1} = \int_eval:n {#2} \exp_stop_f: } 9019
9020 \cs_new:Npn \char_value_lccode:n #1 9020
9021 { \tex_the:D \tex_lccode:D \int_eval:n {#1} \exp_stop_f: } 9021
9022 \cs_new_protected:Npn \char_show_value_lccode:n #1 9022
9023 { \exp_args:Nf \tl_show:n { \char_value_lccode:n {#1} } } 9023
9024 \cs_new_protected:Npn \char_set_uccode:nn #1#2 9024
9025 { \tex_uccode:D \int_eval:n {#1} = \int_eval:n {#2} \exp_stop_f: } 9025
9026 \cs_new:Npn \char_value_uccode:n #1 9026
9027 { \tex_the:D \tex_uccode:D \int_eval:n {#1} \exp_stop_f: } 9027
9028 \cs_new_protected:Npn \char_show_value_uccode:n #1 9028
9029 { \exp_args:Nf \tl_show:n { \char_value_uccode:n {#1} } } 9029
9030 \cs_new_protected:Npn \char_set_sfcode:nn #1#2 9030
9031 { \tex_sfcode:D \int_eval:n {#1} = \int_eval:n {#2} \exp_stop_f: } 9031
9032 \cs_new:Npn \char_value_sfcode:n #1 9032
9033 { \tex_the:D \tex_sfcode:D \int_eval:n {#1} \exp_stop_f: } 9033
9034 \cs_new_protected:Npn \char_show_value_sfcode:n #1 9034
9035 { \exp_args:Nf \tl_show:n { \char_value_sfcode:n {#1} } } 9035
9036 \seq_new:N \l_char_special_seq 9036
9037 \seq_set_split:Nnn \l_char_special_seq { } 9037
9038 { \_ \" \# \$ \% \& \\ \^ \_ \{ \} \~ } 9038
9039 \seq_new:N \l_char_active_seq 9039
9040 \seq_set_split:Nnn \l_char_active_seq { } 9040
9041 { \" \$ \% \^ \_ \~ } 9041
9042 \group_begin: 9042
9043 \char_set_catcode_active:N \^^@ 9043
9044 \cs_set_protected:Npn \__char_tmp:nN #1#2 9044
9045 { 9045
9046 \cs_new_protected:cpn { #1 :nN } ##1 9046
9047 { 9047
9048 \group_begin: 9048
9049 \char_set_lccode:nn { \^^@ } { ##1 } 9049
9050 \tex_lowercase:D { \group_end: #2 ^^@ } 9050
9051 } 9051
9052 \cs_new_protected:cpe { #1 :NN } ##1 9052
9053 { \exp_not:c { #1 : nN } { \^^@ } } 9053
9054 } 9054
9055 \__char_tmp:nN { char_set_active_eq } \cs_set_eq:NN 9055
9056 \__char_tmp:nN { char_gset_active_eq } \cs_gset_eq:NN 9056
9057 \group_end: 9057
9058 \cs_generate_variant:Nn \char_set_active_eq:NN { Nc } 9058
9059 \cs_generate_variant:Nn \char_gset_active_eq:NN { Nc } 9059
9060 \cs_generate_variant:Nn \char_set_active_eq:nN { nc } 9060
9061 \cs_generate_variant:Nn \char_gset_active_eq:nN { nc } 9061
9062 \cs_new_eq:NN \__char_int_to_roman:w \tex_romannumeral:D 9062
```

```

9063 \cs_new_eq:NN \__char_sep: \__kernel_int_sep: 9063
9064 \cs_new:Npn \char_generate:nn #1#2 9064
9065 { 9065
9066     \exp:w \exp_after:wN \__char_generate_aux:w 9066
9067     \int_value:w \int_eval:n {#1} \exp_after:wN \__char_sep: 9067
9068     \int_value:w \int_eval:n {#2} \__char_sep: 9068
9069 } 9069
9070 \cs_new:Npn \__char_generate_aux:w #1 \__char_sep: #2 \__char_sep: 9070
9071 { 9071
9072     \if_int_odd:w 0 9072
9073         \if_int_compare:w #2 < 1 \exp_stop_f: 1 \fi: 9073
9074         \if_int_compare:w #2 = 5 \exp_stop_f: 1 \fi: 9074
9075         \if_int_compare:w #2 = 9 \exp_stop_f: 1 \fi: 9075
9076         \if_int_compare:w #2 > 13 \exp_stop_f: 1 \fi: \exp_stop_f: 9076
9077         \msg_expandable_error:nn { char } 9077
9078         { invalid-catcode } 9078
9079     \else: 9079
9080         \if_int_odd:w 0 9080
9081             \if_int_compare:w #1 < \c_zero_int 1 \fi: 9081
9082             \if_int_compare:w #1 > \c_max_char_int 1 \fi: \exp_stop_f: 9082
9083             \msg_expandable_error:nn { char } 9083
9084             { out-of-range } 9084
9085         \else: 9085
9086             \if_int_compare:w #2#1 = 100 \exp_stop_f: 9086
9087             \msg_expandable_error:nn { char } { null-space } 9087
9088             \else: 9088
9089                 \__char_generate_aux:nnw {#1} {#2} 9089
9090             \fi: 9090
9091             \fi: 9091
9092             \fi: 9092
9093             \exp_end: 9093
9094         } 9094
9095 \tl_new:N \l__char_tmp_tl 9095
9096 \group_begin: 9096
9097     \char_set_catcode_active:N ^^L 9097
9098     \cs_set:Npn ^^L { } 9098
9099     \if_cs_exist:N \tex_Ucharcat:D 9099
9100         \cs_new:Npn \__char_generate_aux:nnw #1#2#3 \exp_end: 9100
9101         { 9101
9102             #3 9102
9103             \exp_after:wN \exp_end: 9103
9104             \tex_Ucharcat:D #1 \exp_stop_f: #2 \exp_stop_f: 9104
9105         } 9105
9106     \else: 9106
9107         \char_set_catcode_active:n { 0 } 9107
9108         \tl_set:Nn \l__char_tmp_tl { \exp_not:N ^^@ \exp_not:N \or: } 9108

```



```

9109 \char_set_catcode_other:n { 0 } 9109
9110 \tl_put_right:Nn \l__char_tmp_tl { ^^@ \exp_not:N \or: } 9110
9111 \char_set_catcode_letter:n { 0 } 9111
9112 \tl_put_right:Nn \l__char_tmp_tl { ^^@ \exp_not:N \or: } 9112
9113 \tl_put_right:Nn \l__char_tmp_tl { \use:n { ~ } \exp_not:N \or: } 9113
9114 \tl_put_right:Nn \l__char_tmp_tl { \exp_not:N \or: } 9114
9115 \char_set_catcode_math_subscript:n { 0 } 9115
9116 \tl_put_right:Nn \l__char_tmp_tl { ^^@ \exp_not:N \or: } 9116
9117 \char_set_catcode_math_superscript:n { 0 } 9117
9118 \tl_put_right:Nn \l__char_tmp_tl { ^^@ \exp_not:N \or: } 9118
9119 \char_set_catcode_parameter:n { 0 } 9119
9120 \tl_put_right:Nn \l__char_tmp_tl { ^^@ \exp_not:N \or: } 9120
9121 \tl_put_right:Nn \l__char_tmp_tl { { \if_false: } \fi: \exp_not:N \or: } 9121
9122 \char_set_catcode_alignment:n { 0 } 9122
9123 \tl_put_right:Nn \l__char_tmp_tl { ^^@ \exp_not:N \or: } 9123
9124 \char_set_catcode_math_toggle:n { 0 } 9124
9125 \tl_put_right:Nn \l__char_tmp_tl { ^^@ \exp_not:N \or: } 9125
9126 \char_set_catcode_group_end:n { 0 } 9126
9127 \tl_put_right:Nn \l__char_tmp_tl { \if_false: { \fi: ^^@ \exp_not:N \or: } % } 9127
9128 \char_set_catcode_group_begin:n { 0 } % { 9128
9129 \tl_put_right:Nn \l__char_tmp_tl { ^^@ \exp_not:N \or: } } 9129
9130 \cs_set_protected:Npn \__char_tmp:n #1 9130
9131 { 9131
9132 \char_set_lccode:nn { 0 } {#1} 9132
9133 \char_set_lccode:nn { 32 } {#1} 9133
9134 \exp_args:Ne \tex_lowercase:D 9134
9135 { 9135
9136 \tl_const:Ne 9136
9137 \exp_not:c { c__char_ \__char_int_to_roman:w #1 _tl } 9137
9138 { \exp_not:o \l__char_tmp_tl } 9138
9139 } 9139
9140 } 9140
9141 \int_step_function:nnN { 0 } { 255 } \__char_tmp:n 9141
9142 \cs_new:Npn \__char_generate_aux:nnw #1#2#3 \exp_end: 9142
9143 { 9143
9144 #3 9144
9145 \if_false: { \fi: 9145
9146 \exp_after:wN \exp_after:wN \exp_after:wN \exp_end: 9146
9147 \exp_after:wN \exp_after:wN 9147
9148 \if_case:w \tex_numexpr:D 13 - #2 9148
9149 \exp_after:wN \exp_after:wN \exp_after:wN \exp_after:wN 9149
9150 \exp_after:wN \exp_after:wN \exp_after:wN \exp_after:wN \scan_stop: 9150
9151 \exp_after:wN \exp_after:wN \exp_after:wN \exp_after:wN \exp_not:N 9151
9152 \cs:w c__char_ \__char_int_to_roman:w #1 _tl \cs_end: 9152
9153 } 9153
9154 \fi: 9154

```

```
9155     }
9156   \fi:
9157 \group_end:
9158 \group_begin:
9159   \char_set_catcode_active:N *
9160   \char_set_lccode:n { `* } { `\_}
9161   \tex_lowercase:D { \tl_const:Nn \c_catcode_active_space_tl { * } }
9162 \group_end:
9163 \tl_const:Ne \c_catcode_other_space_tl { \char_generate:nn { `\_ } { 12 } }
9164 \scan_new:N \s__token_mark
9165 \scan_new:N \s__token_stop
9166 \cs_new:Npn \token_to_catcode:N
9167   { \int_value:w \group_align_safe_begin: \_token_to_catcode:N }
9168 \cs_new:Npn \_token_to_catcode:N #1
9169   {
9170     \if_catcode:w \exp_not:N #1 \c_catcode_letter_token
9171       11
9172     \else:
9173       \if_catcode:w \exp_not:N #1 \c_catcode_other_token
9174         12
9175       \else:
9176         \if_catcode:w \exp_not:N #1 \c_math_toggle_token
9177           3
9178         \else:
9179           \if_catcode:w \exp_not:N #1 \c_alignment_token
9180             4
9181           \else:
9182             \if_catcode:w \exp_not:N #1 ##
9183               6
9184             \else:
9185               \if_catcode:w \exp_not:N #1 \c_math_superscript_token
9186                 7
9187             \else:
9188               \if_catcode:w \exp_not:N #1 \c_math_subscript_token
9189                 8
9190             \else:
9191               \if_catcode:w \exp_not:N #1 \c_group_begin_token
9192                 1
9193             \else:
9194               \if_catcode:w \exp_not:N #1 \c_group_end_token
9195                 2
9196             \else:
9197               \if_catcode:w \exp_not:N #1 \c_space_token
9198                 10
9199             \else:
9200               \token_if_cs:NTF #1 { 16 } { 13 }
```

```

9201         \fi:
9202         \fi:
9203         \fi:
9204         \fi:
9205         \fi:
9206         \fi:
9207         \fi:
9208         \fi:
9209         \fi:
9210         \fi:
9211         \group_align_safe_end:
9212         \exp_stop_f:
9213     }
9214 \group_begin:
9215     \__kernel_chk_if_free_cs:N \c_group_begin_token
9216     \tex_global:D \tex_let:D \c_group_begin_token {
9217     \__kernel_chk_if_free_cs:N \c_group_end_token
9218     \tex_global:D \tex_let:D \c_group_end_token }
9219     \char_set_catcode_math_toggle:N \*
9220     \cs_new_eq:NN \c_math_toggle_token *
9221     \char_set_catcode_alignment:N \*
9222     \cs_new_eq:NN \c_alignment_token *
9223     \cs_new_eq:NN \c_parameter_token #
9224     \cs_new_eq:NN \c_math_superscript_token ^
9225     \char_set_catcode_math_subscript:N \*
9226     \cs_new_eq:NN \c_math_subscript_token *
9227     \__kernel_chk_if_free_cs:N \c_space_token
9228     \use:n { \tex_global:D \tex_let:D \c_space_token = ~ } ~
9229     \cs_new_eq:NN \c_catcode_letter_token a
9230     \cs_new_eq:NN \c_catcode_other_token 1
9231 \group_end:
9232 \group_begin:
9233     \char_set_catcode_active:N \*
9234     \tl_const:Nn \c__token_active_tl { \exp_not:N * }
9235 \group_end:
9236 \prg_new_conditional:Npnn \token_if_group_begin:N #1 { p , T , F , TF }
9237 {
9238     \if_catcode:w \exp_not:N #1 \c_group_begin_token
9239     \prg_return_true: \else: \prg_return_false: \fi:
9240 }
9241 \prg_new_conditional:Npnn \token_if_group_end:N #1 { p , T , F , TF }
9242 {
9243     \if_catcode:w \exp_not:N #1 \c_group_end_token
9244     \prg_return_true: \else: \prg_return_false: \fi:
9245 }
9246 \prg_new_conditional:Npnn \token_if_math_toggle:N #1 { p , T , F , TF }

```

9247	{	9247
9248	\if_catcode:w \exp_not:N #1 \c_math_toggle_token	9248
9249	\prg_return_true: \else: \prg_return_false: \fi:	9249
9250	}	9250
9251	\prg_new_conditional:Npnn \token_if_alignment:N #1 { p , T , F , TF }	9251
9252	{	9252
9253	\if_catcode:w \exp_not:N #1 \c_alignment_token	9253
9254	\prg_return_true: \else: \prg_return_false: \fi:	9254
9255	}	9255
9256	\group_begin:	9256
9257	\cs_set_eq:NN \c_parameter_token \scan_stop:	9257
9258	\prg_new_conditional:Npnn \token_if_parameter:N #1 { p , T , F , TF }	9258
9259	{	9259
9260	\if_catcode:w \exp_not:N #1 \c_parameter_token	9260
9261	\prg_return_true: \else: \prg_return_false: \fi:	9261
9262	}	9262
9263	\group_end:	9263
9264	\prg_new_conditional:Npnn \token_if_math_superscript:N #1	9264
9265	{ p , T , F , TF }	9265
9266	{	9266
9267	\if_catcode:w \exp_not:N #1 \c_math_superscript_token	9267
9268	\prg_return_true: \else: \prg_return_false: \fi:	9268
9269	}	9269
9270	\prg_new_conditional:Npnn \token_if_math_subscript:N #1 { p , T , F , TF }	9270
9271	{	9271
9272	\if_catcode:w \exp_not:N #1 \c_math_subscript_token	9272
9273	\prg_return_true: \else: \prg_return_false: \fi:	9273
9274	}	9274
9275	\prg_new_conditional:Npnn \token_if_space:N #1 { p , T , F , TF }	9275
9276	{	9276
9277	\if_catcode:w \exp_not:N #1 \c_space_token	9277
9278	\prg_return_true: \else: \prg_return_false: \fi:	9278
9279	}	9279
9280	\prg_new_conditional:Npnn \token_if_letter:N #1 { p , T , F , TF }	9280
9281	{	9281
9282	\if_catcode:w \exp_not:N #1 \c_catcode_letter_token	9282
9283	\prg_return_true: \else: \prg_return_false: \fi:	9283
9284	}	9284
9285	\prg_new_conditional:Npnn \token_if_other:N #1 { p , T , F , TF }	9285
9286	{	9286
9287	\if_catcode:w \exp_not:N #1 \c_catcode_other_token	9287
9288	\prg_return_true: \else: \prg_return_false: \fi:	9288
9289	}	9289
9290	\prg_new_conditional:Npnn \token_if_active:N #1 { p , T , F , TF }	9290
9291	{	9291
9292	\if_catcode:w \exp_not:N #1 \c__token_active_tl	9292

```
9293 \prg_return_true: \else: \prg_return_false: \fi: 9293
9294 } 9294
9295 \prg_new_eq_conditional:NNn \token_if_eq_meaning:NN \cs_if_eq:NN 9295
9296 { p , T , F , TF } 9296
9297 \prg_new_conditional:Npnn \token_if_eq_catcode:NN #1#2 { p , T , F , TF } 9297
9298 { 9298
9299 \if_catcode:w \exp_not:N #1 \exp_not:N #2 9299
9300 \prg_return_true: \else: \prg_return_false: \fi: 9300
9301 } 9301
9302 \prg_new_conditional:Npnn \token_if_eq_charcode:NN #1#2 { p , T , F , TF } 9302
9303 { 9303
9304 \if_charcode:w \exp_not:N #1 \exp_not:N #2 9304
9305 \prg_return_true: \else: \prg_return_false: \fi: 9305
9306 } 9306
9307 \use:e 9307
9308 { 9308
9309 \prg_new_conditional:Npnn \exp_not:N \token_if_macro:N #1 9309
9310 { p , T , F , TF } 9310
9311 { 9311
9312 \exp_not:N \exp_after:wN \exp_not:N \__token_if_macro_p:w 9312
9313 \exp_not:N \token_to_meaning:N #1 \tl_to_str:n { ma : } 9313
9314 \s__token_stop 9314
9315 } 9315
9316 \cs_new:Npn \exp_not:N \__token_if_macro_p:w 9316
9317 #1 \tl_to_str:n { ma } #2 \c_colon_str #3 \s__token_stop 9317
9318 } 9318
9319 { 9319
9320 \str_if_eq:nnTF { #2 } { cro } 9320
9321 { \prg_return_true: } 9321
9322 { \prg_return_false: } 9322
9323 } 9323
9324 \prg_new_conditional:Npnn \token_if_cs:N #1 { p , T , F , TF } 9324
9325 { 9325
9326 \if_catcode:w \exp_not:N #1 \scan_stop: 9326
9327 \prg_return_true: \else: \prg_return_false: \fi: 9327
9328 } 9328
9329 \prg_new_conditional:Npnn \token_if_expandable:N #1 { p , T , F , TF } 9329
9330 { 9330
9331 \exp_after:wN \if_meaning:w \exp_not:N #1 #1 9331
9332 \prg_return_false: 9332
9333 \else: 9333
9334 \if_cs_exist:N #1 9334
9335 \prg_return_true: 9335
9336 \else: 9336
9337 \prg_return_false: 9337
9338 \fi: 9338
```

```

9339 \fi: 9339
9340 } 9340
9341 \group_begin: 9341
9342 \cs_set_protected:Npn \__token_tmp:w #1 9342
9343 { 9343
9344 \use:e 9344
9345 { 9345
9346 \cs_new:Npn \exp_not:c { __token_delimit_by_ #1 :w } 9346
9347 ##1 \tl_to_str:n {#1} ##2 \s__token_stop 9347
9348 { ##1 \tl_to_str:n {#1} } 9348
9349 } 9349
9350 } 9350
9351 \__token_tmp:w { char" } 9351
9352 \__token_tmp:w { count } 9352
9353 \__token_tmp:w { dimen } 9353
9354 \__token_tmp:w { ~ font } 9354
9355 \__token_tmp:w { macro } 9355
9356 \__token_tmp:w { muskip } 9356
9357 \__token_tmp:w { skip } 9357
9358 \__token_tmp:w { toks } 9358
9359 \group_end: 9359
9360 \group_begin: 9360
9361 \cs_set_protected:Npn \__token_tmp:w #1#2#3 9361
9362 { 9362
9363 \use:e 9363
9364 { 9364
9365 \prg_new_conditional:Npnn \exp_not:c { token_if_ #1 :N } ##1 9365
9366 { p , T , F , TF } 9366
9367 { 9367
9368 \cs_if_exist:cT { tex_ #2 :D } 9368
9369 { 9369
9370 \exp_not:N \if_meaning:w ##1 \exp_not:c { tex_ #2 :D } 9370
9371 \exp_not:N \prg_return_false: 9371
9372 \exp_not:N \else: 9372
9373 \exp_not:N \if_meaning:w ##1 \exp_not:c { tex_ #2 def:D } 9373
9374 \exp_not:N \prg_return_false: 9374
9375 \exp_not:N \else: 9375
9376 } 9376
9377 \exp_not:N \str_if_eq:eeTF 9377
9378 { 9378
9379 \exp_not:N \exp_after:wN 9379
9380 \exp_not:c { __token_delimit_by_ #2 :w } 9380
9381 \exp_not:N \token_to_meaning:N ##1 9381
9382 ? \tl_to_str:n {#2} \s__token_stop 9382
9383 } 9383
9384 { \exp_not:n {#3} } 9384

```

```

9385         { \exp_not:N \prg_return_true: }
9386         { \exp_not:N \prg_return_false: }
9387     \cs_if_exist:cT { tex_ #2 :D }
9388     {
9389         \exp_not:N \fi:
9390         \exp_not:N \fi:
9391     }
9392 }
9393 }
9394 }
9395 \__token_tmp:w { chardef } { char" } { \token_to_str:N \char" }
9396 \__token_tmp:w { mathchardef } { char" } { \token_to_str:N \mathchar" }
9397 \__token_tmp:w { long_macro } { macro } { \tl_to_str:n { \long } macro }
9398 \__token_tmp:w { protected_macro } { macro }
9399     { \tl_to_str:n { \protected } macro }
9400 \__token_tmp:w { protected_long_macro } { macro }
9401     { \token_to_str:N \protected \tl_to_str:n { \long } macro }
9402 \__token_tmp:w { font_selection } { ~ font } { select ~ font }
9403 \__token_tmp:w { dim_register } { dimen } { \token_to_str:N \dimen }
9404 \__token_tmp:w { int_register } { count } { \token_to_str:N \count }
9405 \__token_tmp:w { muskip_register } { muskip } { \token_to_str:N \muskip }
9406 \__token_tmp:w { skip_register } { skip } { \token_to_str:N \skip }
9407 \__token_tmp:w { toks_register } { toks } { \token_to_str:N \toks }
9408 \group_end:
9409 \sys_if_engine luatex:TF
9410 {
9411     \prg_new_conditional:Npnn \token_if_primitive:N #1 { p , T , F , TF }
9412     {
9413         \__token_if_primitive_lua:N #1
9414     }
9415 }
9416 {
9417     \tex_global:D \tex_chardef:D \c__token_A_int = `A ~ %
9418     \use:e
9419     {
9420         \prg_new_conditional:Npnn \exp_not:N \token_if_primitive:N #1
9421         { p , T , F , TF }
9422         {
9423             \exp_not:N \token_if_macro:NTF #1
9424             \exp_not:N \prg_return_false:
9425             {
9426                 \exp_not:N \exp_after:wN \exp_not:N \__token_if_primitive:NNw
9427                 \exp_not:N \token_to_meaning:N #1
9428                 \tl_to_str:n { : : : } \s__token_stop #1
9429             }
9430         }
9431     }

```



```
9431 \cs_new:Npn \exp_not:N \__token_if_primitive:NNw 9431
9432 #1#2 #3 \c_colon_str #4 \s__token_stop 9432
9433 { 9433
9434 \exp_not:N \tl_if_empty:oTF 9434
9435 { \exp_not:N \__token_if_primitive_space:w #3 ~ } 9435
9436 { 9436
9437 \exp_not:N \__token_if_primitive_loop:N #3 9437
9438 \c_colon_str \s__token_stop 9438
9439 } 9439
9440 { \exp_not:N \__token_if_primitive_nullfont:N } 9440
9441 } 9441
9442 } 9442
9443 \cs_new:Npn \__token_if_primitive_space:w #1 ~ { } 9443
9444 \cs_new:Npn \__token_if_primitive_nullfont:N #1 9444
9445 { 9445
9446 \if_meaning:w \tex_nullfont:D #1 9446
9447 \prg_return_true: 9447
9448 \else: 9448
9449 \prg_return_false: 9449
9450 \fi: 9450
9451 } 9451
9452 \cs_new:Npn \__token_if_primitive_loop:N #1 9452
9453 { 9453
9454 \if_int_compare:w `#1 < \c__token_A_int % 9454
9455 \exp_after:wN \__token_if_primitive:Nw 9455
9456 \exp_after:wN #1 9456
9457 \else: 9457
9458 \exp_after:wN \__token_if_primitive_loop:N 9458
9459 \fi: 9459
9460 } 9460
9461 \cs_new:Npn \__token_if_primitive:Nw #1 #2 \s__token_stop 9461
9462 { 9462
9463 \if:w : #1 9463
9464 \exp_after:wN \__token_if_primitive_undefined:N 9464
9465 \else: 9465
9466 \prg_return_false: 9466
9467 \exp_after:wN \use_none:n 9467
9468 \fi: 9468
9469 } 9469
9470 \cs_new:Npn \__token_if_primitive_undefined:N #1 9470
9471 { 9471
9472 \if_cs_exist:N #1 9472
9473 \prg_return_true: 9473
9474 \else: 9474
9475 \prg_return_false: 9475
9476 \fi: 9476
```

9477	}	9477
9478	}	9478
9479	\cs_new:Npn \token_case_catcode:Nn #1#2	9479
9480	{ \exp:w __token_case:NNnTF \token_if_eq_catcode:NNTF #1 {#2} { } { } }	9480
9481	\cs_new:Npn \token_case_catcode:NnT #1#2#3	9481
9482	{ \exp:w __token_case:NNnTF \token_if_eq_catcode:NNTF #1 {#2} {#3} { } }	9482
9483	\cs_new:Npn \token_case_catcode:NnF #1#2	9483
9484	{ \exp:w __token_case:NNnTF \token_if_eq_catcode:NNTF #1 {#2} { } }	9484
9485	\cs_new:Npn \token_case_catcode:NnTF	9485
9486	{ \exp:w __token_case:NNnTF \token_if_eq_catcode:NNTF }	9486
9487	\cs_new:Npn \token_case_charcode:Nn #1#2	9487
9488	{ \exp:w __token_case:NNnTF \token_if_eq_charcode:NNTF #1 {#2} { } { } }	9488
9489	\cs_new:Npn \token_case_charcode:NnT #1#2#3	9489
9490	{ \exp:w __token_case:NNnTF \token_if_eq_charcode:NNTF #1 {#2} {#3} { } }	9490
9491	\cs_new:Npn \token_case_charcode:NnF #1#2	9491
9492	{ \exp:w __token_case:NNnTF \token_if_eq_charcode:NNTF #1 {#2} { } }	9492
9493	\cs_new:Npn \token_case_charcode:NnTF	9493
9494	{ \exp:w __token_case:NNnTF \token_if_eq_charcode:NNTF }	9494
9495	\cs_new:Npn \token_case_meaning:Nn #1#2	9495
9496	{ \exp:w __token_case:NNnTF \token_if_eq_meaning:NNTF #1 {#2} { } { } }	9496
9497	\cs_new:Npn \token_case_meaning:NnT #1#2#3	9497
9498	{ \exp:w __token_case:NNnTF \token_if_eq_meaning:NNTF #1 {#2} {#3} { } }	9498
9499	\cs_new:Npn \token_case_meaning:NnF #1#2	9499
9500	{ \exp:w __token_case:NNnTF \token_if_eq_meaning:NNTF #1 {#2} { } }	9500
9501	\cs_new:Npn \token_case_meaning:NnTF	9501
9502	{ \exp:w __token_case:NNnTF \token_if_eq_meaning:NNTF }	9502
9503	\cs_new:Npn __token_case:NNnTF #1#2#3#4#5	9503
9504	{	9504
9505	__token_case:NNw #1 #2 #3 #2 { }	9505
9506	\s__token_mark {#4}	9506
9507	\s__token_mark {#5}	9507
9508	\s__token_stop	9508
9509	}	9509
9510	\cs_new:Npn __token_case:NNw #1#2#3#4	9510
9511	{	9511
9512	#1 #2 #3	9512
9513	{ __token_case_end:nw {#4} }	9513
9514	{ __token_case:NNw #1 #2 }	9514
9515	}	9515
9516	\cs_new:Npn __token_case_end:nw #1#2#3 \s__token_mark #4#5 \s__token_stop	9516
9517	{ \exp_end: #1 #4 }	9517
9518	\cs_new_eq:NN \l_peek_token ?	9518
9519	\cs_new_eq:NN \g_peek_token ?	9519
9520	\cs_new_eq:NN \l_peek_search_token ?	9520
9521	\tl_new:N \l_peek_search_tl	9521
9522	\cs_new:Npn __peek_true:w { }	9522

```

9523 \cs_new:Npn \__peek_true_aux:w { }
9524 \cs_new:Npn \__peek_false:w { }
9525 \cs_new:Npn \__peek_tmp:w { }
9526 \scan_new:N \s__peek_mark
9527 \scan_new:N \s__peek_stop
9528 \cs_new:Npn \__peek_use_none_delimit_by_s_stop:w #1 \s__peek_stop { }
9529 \cs_new_protected:Npn \peek_after:Nw
9530 { \tex_futurelet:D \l_peek_token }
9531 \cs_new_protected:Npn \peek_gafter:Nw
9532 { \tex_global:D \tex_futurelet:D \g_peek_token }
9533 \cs_new_protected:Npn \__peek_true_remove:w
9534 {
9535     \tex_afterassignment:D \__peek_true_aux:w
9536     \cs_set_eq:NN \__peek_tmp:w
9537 }
9538 \cs_new_protected:Npn \peek_remove_spaces:n #1
9539 {
9540     \cs_set:Npe \__peek_false:w { \exp_not:n {#1} }
9541     \group_align_safe_begin:
9542     \cs_set:Npn \__peek_true_aux:w { \peek_after:Nw \__peek_remove_spaces: }
9543     \__peek_true_aux:w
9544 }
9545 \cs_new_protected:Npn \__peek_remove_spaces:
9546 {
9547     \if_meaning:w \l_peek_token \c_space_token
9548     \exp_after:wN \__peek_true_remove:w
9549 \else:
9550     \group_align_safe_end:
9551     \exp_after:wN \__peek_false:w
9552 \fi:
9553 }
9554 \cs_new_protected:Npn \peek_remove_filler:n #1
9555 {
9556     \cs_set:Npn \__peek_true_aux:w { \__peek_remove_filler:w }
9557     \cs_set:Npe \__peek_false:w
9558     {
9559         \exp_not:N \group_align_safe_end:
9560         \exp_not:n {#1}
9561     }
9562     \group_align_safe_begin:
9563     \__peek_remove_filler:w
9564 }
9565 \cs_new_protected:Npn \__peek_remove_filler:w
9566 {
9567     \exp_after:wN \peek_after:Nw \exp_after:wN \__peek_remove_filler:
9568     \exp:w \exp_end_continue_f:w

```

```
9569 } 9569
9570 \cs_new_protected:Npn \__peek_remove_filler: 9570
9571 { 9571
9572 \if_catcode:w \exp_not:N \l_peek_token \c_space_token 9572
9573 \exp_after:wN \__peek_true_remove:w 9573
9574 \else: 9574
9575 \if_meaning:w \l_peek_token \scan_stop: 9575
9576 \exp_after:wN \exp_after:wN \exp_after:wN 9576
9577 \__peek_true_remove:w 9577
9578 \else: 9578
9579 \exp_after:wN \exp_after:wN \exp_after:wN 9579
9580 \__peek_remove_filler_expand:w 9580
9581 \fi: 9581
9582 \fi: 9582
9583 } 9583
9584 \cs_new_protected:Npn \__peek_remove_filler_expand:w 9584
9585 { 9585
9586 \exp_after:wN \if_meaning:w \exp_not:N \l_peek_token \l_peek_token 9586
9587 \exp_after:wN \__peek_false:w 9587
9588 \else: 9588
9589 \exp_after:wN \__peek_remove_filler:w 9589
9590 \fi: 9590
9591 } 9591
9592 \cs_new_protected:Npn \__peek_token_generic_aux:NNNTF #1#2#3#4#5 9592
9593 { 9593
9594 \group_align_safe_begin: 9594
9595 \cs_set_eq:NN \l__peek_search_token #3 9595
9596 \tl_set:Nn \l__peek_search_tl {#3} 9596
9597 \cs_set:Npe \__peek_true_aux:w 9597
9598 { 9598
9599 \exp_not:N \group_align_safe_end: 9599
9600 \exp_not:n {#4} 9600
9601 } 9601
9602 \cs_set_eq:NN \__peek_true:w #1 9602
9603 \cs_set:Npe \__peek_false:w 9603
9604 { 9604
9605 \exp_not:N \group_align_safe_end: 9605
9606 \exp_not:n {#5} 9606
9607 } 9607
9608 \peek_after:Nw #2 9608
9609 } 9609
9610 \cs_new_protected:Npn \__peek_token_generic:NNTF 9610
9611 { \__peek_token_generic_aux:NNNTF \__peek_true_aux:w } 9611
9612 \cs_new_protected:Npn \__peek_token_generic:NNT #1#2#3 9612
9613 { \__peek_token_generic:NNTF #1 #2 {#3} { } } 9613
9614 \cs_new_protected:Npn \__peek_token_generic:NNF #1#2#3 9614
```

```
9615 { \__peek_token_generic:NNTF #1 #2 { } {#3} } 9615
9616 \cs_new_protected:Npn \__peek_token_remove_generic:NNTF 9616
9617 { \__peek_token_generic_aux:NNNTF \__peek_true_remove:w } 9617
9618 \cs_new_protected:Npn \__peek_token_remove_generic:NNT #1#2#3 9618
9619 { \__peek_token_remove_generic:NNTF #1 #2 {#3} { } } 9619
9620 \cs_new_protected:Npn \__peek_token_remove_generic:NNF #1#2#3 9620
9621 { \__peek_token_remove_generic:NNTF #1 #2 { } {#3} } 9621
9622 \cs_new:Npn \__peek_execute_branches_meaning: 9622
9623 { 9623
9624 \if_meaning:w \l_peek_token \l__peek_search_token 9624
9625 \exp_after:wN \__peek_true:w 9625
9626 \else: 9626
9627 \exp_after:wN \__peek_false:w 9627
9628 \fi: 9628
9629 } 9629
9630 \cs_new:Npn \__peek_execute_branches_catcode: 9630
9631 { \if_catcode:w \__peek_execute_branches_catcode_aux: } 9631
9632 \cs_new:Npn \__peek_execute_branches_charcode: 9632
9633 { \if_charcode:w \__peek_execute_branches_catcode_aux: } 9633
9634 \cs_new:Npn \__peek_execute_branches_catcode_aux: 9634
9635 { 9635
9636 \if_catcode:w \exp_not:N \l_peek_token \scan_stop: 9636
9637 \exp_after:wN \exp_after:wN 9637
9638 \exp_after:wN \__peek_execute_branches_catcode_auxii:N 9638
9639 \exp_after:wN \exp_not:N 9639
9640 \else: 9640
9641 \exp_after:wN \__peek_execute_branches_catcode_auxiii: 9641
9642 \fi: 9642
9643 } 9643
9644 \cs_new:Npn \__peek_execute_branches_catcode_auxii:N #1 9644
9645 { 9645
9646 \exp_not:N #1 9646
9647 \exp_after:wN \exp_not:N \l__peek_search_tl 9647
9648 \exp_after:wN \__peek_true:w 9648
9649 \else: 9649
9650 \exp_after:wN \__peek_false:w 9650
9651 \fi: 9651
9652 #1 9652
9653 } 9653
9654 \cs_new:Npn \__peek_execute_branches_catcode_auxiii: 9654
9655 { 9655
9656 \exp_not:N \l_peek_token 9656
9657 \exp_after:wN \exp_not:N \l__peek_search_tl 9657
9658 \exp_after:wN \__peek_true:w 9658
9659 \else: 9659
9660 \exp_after:wN \__peek_false:w 9660
```

```

9661 \fi:
9662 }
9663 \tl_map_inline:nn { { catcode } { charcode } { meaning } }
9664 {
9665   \tl_map_inline:nn { { } { _remove } }
9666   {
9667     \tl_map_inline:nn { { TF } { T } { F } }
9668     {
9669       \cs_new_protected:cpe { peek_ #1 ##1 :N ####1 }
9670       {
9671         \exp_not:c { __peek_token ##1 _generic:NN ####1 }
9672         \exp_not:c { __peek_execute_branches_ #1 : }
9673       }
9674     }
9675   }
9676 }
9677 \group_begin:
9678   \cs_set_protected:Npn \__peek_tmp:w #1 \s__peek_stop
9679   {
9680     \cs_new_protected:Npn \__peek_execute_branches_N_type:
9681     {
9682       \if_int_odd:w
9683         \if_catcode:w \exp_not:N \l_peek_token { \c_zero_int \fi:
9684         \if_catcode:w \exp_not:N \l_peek_token } \c_zero_int \fi:
9685         \if_meaning:w \l_peek_token \c_space_token \c_zero_int \fi:
9686         \c_one_int
9687       \exp_after:wN \__peek_N_type:w
9688       \token_to_meaning:N \l_peek_token
9689       \s__peek_mark \__peek_N_type_aux:nnw
9690       #1 \s__peek_mark \__peek_use_none_delimit_by_s_stop:w
9691       \s__peek_stop
9692       \exp_after:wN \__peek_true:w
9693     \else:
9694       \exp_after:wN \__peek_false:w
9695     \fi:
9696   }
9697   \cs_new_protected:Npn \__peek_N_type:w ##1 #1 ##2 \s__peek_mark ##3
9698   { ##3 {##1} {##2} }
9699 }
9700 \exp_after:wN \__peek_tmp:w \tl_to_str:n { outer } \s__peek_stop
9701 \group_end:
9702 \cs_new_protected:Npn \__peek_N_type_aux:nnw #1 #2 #3 \fi:
9703 {
9704   \fi:
9705   \tl_if_in:noTF {#1} { \tl_to_str:n {ma} }
9706   { \__peek_true:w }

```



```
9707 { \tl_if_empty:nTF {#2} { \__peek_true:w } { \__peek_false:w } } 9707
9708 } 9708
9709 \cs_new_protected:Npn \peek_N_type:TF 9709
9710 { 9710
9711 \__peek_token_generic:NNTF 9711
9712 \__peek_execute_branches_N_type: \scan_stop: 9712
9713 } 9713
9714 \cs_new_protected:Npn \peek_N_type:T 9714
9715 { \__peek_token_generic:NNT \__peek_execute_branches_N_type: \scan_stop: } 9715
9716 \cs_new_protected:Npn \peek_N_type:F 9716
9717 { \__peek_token_generic:NNF \__peek_execute_branches_N_type: \scan_stop: } 9717
9718 %% File: l3prop.dtx 9718
9719 \cs_new_eq:NN \__prop_tmp:w ? 9719
9720 \tl_new:N \l__prop_internal_tl 9720
9721 \scan_new:N \s__prop_mark 9721
9722 \scan_new:N \s__prop_stop 9722
9723 \quark_new:N \q__prop_recursion_tail 9723
9724 \quark_new:N \q__prop_recursion_stop 9724
9725 \__kernel_quark_new_test:N \__prop_if_recursion_tail_stop:n 9725
9726 \cs_generate_variant:Nn \__prop_if_recursion_tail_stop:n { o } 9726
9727 \scan_new:N \s__prop 9727
9728 \cs_new_protected:Npn \__prop_chk:w { \__prop_chk_loop:nw { } } 9728
9729 \cs_new_protected:Npn \__prop_chk_loop:nw #1 9729
9730 { 9730
9731 \peek_meaning:NNTF \__prop_pair:wn 9731
9732 { \__prop_chk_get:nw {#1} } 9732
9733 { \msg_error:nne { prop } { misused } {#1} } 9733
9734 } 9734
9735 \cs_new_protected:Npn \__prop_chk_get:nw #1 \__prop_pair:wn #2 \s__prop #3 9735
9736 { \__prop_chk_loop:nw { #1 , ~ {#2} = { \tl_to_str:n {#3} } } } 9736
9737 \cs_new:Npn \__prop_pair:wn #1 \s__prop #2 { } 9737
9738 \cs_new_protected:Npn \__prop_flatten:w #1 \s__prop #2#3 9738
9739 { \use:e { \__prop_flatten_aux:N #3 } } 9739
9740 \cs_new:Npn \__prop_flatten:N #1 9740
9741 { \exp_after:wN \__prop_flatten_aux:w #1 } 9741
9742 \cs_new:Npn \__prop_flatten_aux:w #1 \s__prop #2 { \__prop_flatten_aux:N } 9742
9743 \cs_new:Npn \__prop_flatten_aux:N #1 9743
9744 { 9744
9745 \s__prop \__prop_chk:w 9745
9746 \exp_after:wN \__prop_flatten_loop:w #1 9746
9747 \use_none:n \__prop_pair:wn \s__prop { } 9747
9748 } 9748
9749 \cs_new:Npn \__prop_flatten_loop:w #1#2#3 \__prop_pair:wn #4 \s__prop #5 9749
9750 { 9750
9751 #3 9751
9752 \exp_not:n { \__prop_pair:wn #4 \s__prop {#5} } 9752
```



```
9753 \exp_after:wN \__prop_flatten_loop:w 9753
9754 } 9754
9755 \int_new:N \g__prop_prefix_int 9755
9756 \int_const:Nn \c__prop_basis_int { \c_max_char_int - `!\ } 9756
9757 \cs_new_protected:Npn \__prop_next_prefix: 9757
9758 { 9758
9759 \tl_set:Ne \l__prop_internal_tl 9759
9760 { \__prop_to_prefix:n { \g__prop_prefix_int } } 9760
9761 \int_gincr:N \g__prop_prefix_int 9761
9762 } 9762
9763 \cs_new:Npn \__prop_to_prefix:n #1 9763
9764 { 9764
9765 \int_compare:nNnTF {#1} > \c__prop_basis_int 9765
9766 { 9766
9767 \exp_args:Nf \__prop_to_prefix:n 9767
9768 { \int_div_truncate:nn {#1} \c__prop_basis_int } 9768
9769 \exp_args:Nf \__prop_to_prefix:n 9769
9770 { \int_mod:nn {#1} \c__prop_basis_int } 9770
9771 } 9771
9772 { \char_generate:nn { `!\ + #1 } { 12 } } 9772
9773 } 9773
9774 \cs_new:Npn \__prop_if_flat:NTF #1 9774
9775 { 9775
9776 \prop_if_exist:NT #1 9776
9777 \exp_after:wN \__prop_if_flat_aux:w #1 9777
9778 \s__prop_mark \use_ii:nn 9778
9779 \__prop_flatten:w \s__prop_mark \use_i:nn \s__prop_stop 9779
9780 } 9780
9781 \cs_new:Npn \__prop_if_flat_aux:w 9781
9782 #1 \__prop_flatten:w #2 \s__prop_mark #3 #4 \s__prop_stop {#3} 9782
9783 \tl_const:Nn \c_empty_prop { \s__prop \__prop_chk:w } 9783
9784 \cs_new_protected:Npn \prop_new:N #1 9784
9785 { 9785
9786 \__kernel_chk_if_free_cs:N #1 9786
9787 \cs_gset_eq:NN #1 \c_empty_prop 9787
9788 } 9788
9789 \cs_generate_variant:Nn \prop_new:N { c } 9789
9790 \cs_new_protected:Npn \prop_new_linked:N #1 9790
9791 { 9791
9792 \__kernel_chk_if_free_cs:N #1 9792
9793 \__prop_new_linked:N #1 9793
9794 } 9794
9795 \cs_new_protected:Npn \__prop_new_linked:N #1 9795
9796 { 9796
9797 \__prop_next_prefix: 9797
9798 \cs_gset_nopar:Npe #1 9798
```

```

9799 { 9799
9800 \__prop_flatten:w 9800
9801 \exp_not:c { __prop ~ \l__prop_internal_tl } 9801
9802 \s__prop { \l__prop_internal_tl } 9802
9803 \exp_not:c { __prop ~ \l__prop_internal_tl } 9803
9804 } 9804
9805 \cs_gset_nopar:cpe { __prop ~ \l__prop_internal_tl } 9805
9806 { 9806
9807 \exp_not:N \use_none:n 9807
9808 \exp_not:N #1 9808
9809 } 9809
9810 } 9810
9811 \cs_generate_variant:Nn \prop_new_linked:N { c } 9811
9812 \cs_new_protected:Npn \prop_clear:N 9812
9813 { \__prop_clear:NNN \cs_set_eq:NN \cs_set_nopar:Npe } 9813
9814 \cs_generate_variant:Nn \prop_clear:N { c } 9814
9815 \cs_new_protected:Npn \prop_gclear:N 9815
9816 { \__prop_clear:NNN \cs_gset_eq:NN \cs_gset_nopar:Npe } 9816
9817 \cs_generate_variant:Nn \prop_gclear:N { c } 9817
9818 \cs_new_protected:Npn \__prop_clear:NNN #1#2#3 9818
9819 { 9819
9820 \__prop_if_flat:NTF #3 9820
9821 { #1 #3 \c_empty_prop } 9821
9822 { \exp_after:wN \__prop_clear:wNNN #3 #1 #2 #3 } 9822
9823 } 9823
9824 \cs_new_protected:Npn \__prop_clear:wNNN 9824
9825 \__prop_flatten:w #1 \s__prop #2#3#4#5#6 9825
9826 { 9826
9827 \__prop_clear_entries:NN #4 #3 9827
9828 #5 #6 { \exp_not:n { \__prop_flatten:w #1 \s__prop {#2} #1 } } 9828
9829 #5 #1 { \exp_not:n { \use_none:n #6 } } 9829
9830 } 9830
9831 \cs_new_protected:Npn \__prop_clear_entries:NN #1#2 9831
9832 { 9832
9833 \exp_after:wN \__prop_clear_loop:Nw \exp_after:wN #1 #2 9833
9834 \use_none:nnnn \__prop_pair:wn \s__prop { } 9834
9835 } 9835
9836 \cs_new_protected:Npn \__prop_clear_loop:Nw 9836
9837 #1#2#3#4 \__prop_pair:wn #5 \s__prop #6 9837
9838 { 9838
9839 #1 #3 \tex_undefined:D 9839
9840 #4 9840
9841 \exp_after:wN \__prop_clear_loop:Nw 9841
9842 \exp_after:wN #1 9842
9843 } 9843
9844 \cs_new_protected:Npn \prop_clear_new:N #1 9844

```

```
9845 { \prop_if_exist:NTF #1 { \prop_clear:N #1 } { \prop_new:N #1 } } 9845
9846 \cs_generate_variant:Nn \prop_clear_new:N { c } 9846
9847 \cs_new_protected:Npn \prop_gclear_new:N #1 9847
9848 { \prop_if_exist:NTF #1 { \prop_gclear:N #1 } { \prop_new:N #1 } } 9848
9849 \cs_generate_variant:Nn \prop_gclear_new:N { c } 9849
9850 \cs_new_protected:Npn \prop_clear_new_linked:N #1 9850
9851 { \prop_if_exist:NTF #1 { \prop_clear:N #1 } { \prop_new_linked:N #1 } } 9851
9852 \cs_generate_variant:Nn \prop_clear_new_linked:N { c } 9852
9853 \cs_new_protected:Npn \prop_gclear_new_linked:N #1 9853
9854 { \prop_if_exist:NTF #1 { \prop_gclear:N #1 } { \prop_new_linked:N #1 } } 9854
9855 \cs_generate_variant:Nn \prop_gclear_new_linked:N { c } 9855
9856 \cs_new_protected:Npn \prop_set_eq:NN 9856
9857 { \__prop_set_eq:NNNN \cs_set_eq:NN \cs_set_nopar:Npe } 9857
9858 \cs_generate_variant:Nn \prop_set_eq:NN { Nc , cN , cc } 9858
9859 \cs_new_protected:Npn \prop_gset_eq:NN 9859
9860 { \__prop_set_eq:NNNN \cs_gset_eq:NN \cs_gset_nopar:Npe } 9860
9861 \cs_generate_variant:Nn \prop_gset_eq:NN { Nc , cN , cc } 9861
9862 \cs_new_protected:Npn \__prop_set_eq:NNNN #1#2#3#4 9862
9863 { 9863
9864 \cs_if_eq:NNF #3#4 9864
9865 { 9865
9866 \__prop_if_flat:NTF #3 9866
9867 { 9867
9868 \__prop_if_flat:NTF #4 9868
9869 { #1 #3 #4 } 9869
9870 { #2 #3 { \__prop_flatten:N #4 } } 9870
9871 } 9871
9872 { \exp_after:wN \__prop_set_eq:wNNNN #3 #1#2#3#4 } 9872
9873 } 9873
9874 } 9874
9875 \cs_new_protected:Npn \__prop_set_eq:wNNNN 9875
9876 \__prop_flatten:w #1 \s__prop #2#3#4#5#6#7 9876
9877 { 9877
9878 \__prop_clear_entries:NN #4 #3 9878
9879 \exp_args:Nf \__prop_set_eq:nNnNN {#7} #1 {#2} #5 #6 9879
9880 } 9880
9881 \cs_new_protected:Npn \__prop_set_eq:nNnNN #1#2#3#4#5 9881
9882 { 9882
9883 \use_i:nnn 9883
9884 { 9884
9885 \__prop_set_eq_loop:NNnw #5 #4 {#3} 9885
9886 \__prop_flatten:w #2 \s__prop {#3} 9886
9887 } 9887
9888 #1 9888
9889 \use_none:n \__prop_pair:wn ? \s__prop 9889
9890 } 9890
```

```

9891 \cs_new_protected:Npn \__prop_set_eq_loop:NNnw 9891
9892   #1#2#3#4 \s__prop #5#6 \__prop_pair:wn #7 \s__prop 9892
9893 { 9893
9894   \tl_set:Nc \l__prop_internal_tl { \exp_not:c { __prop ~ #3 #6 ~ #7 } } 9894
9895   #2 #1 { \exp_not:n { #4 \s__prop {#5} } \exp_not:o \l__prop_internal_tl } 9895
9896   \use_none:n #6 \__prop_set_eq_end:w 9896
9897   \exp_after:wN \__prop_set_eq_loop:NNnw \l__prop_internal_tl #2 {#3} 9897
9898   \use_none:n #1 \__prop_pair:wn #7 \s__prop 9898
9899 } 9899
9900 \cs_new_protected:Npn \__prop_set_eq_end:w 9900
9901   \exp_after:wN \__prop_set_eq_loop:NNnw #1#2#3 9901
9902   \use_none:n #4#5 \s__prop 9902
9903 { 9903
9904   \exp_after:wN #2 \l__prop_internal_tl { \exp_not:n { \use_none:n #4 } } 9904
9905 } 9905
9906 \cs_new_protected:Npn \prop_make_flat:N #1 9906
9907 { 9907
9908   \int_compare:nNnTF { \tex_currentgrouplevel:D } = 0 9908
9909   { 9909
9910     \__prop_if_flat:NTF #1 { } 9910
9911     { \exp_args:NNf \__prop_make_flat:Nn #1 {#1} } 9911
9912   } 9912
9913   { 9913
9914     \msg_error:nnee { prop } { inner-make } 9914
9915     { \token_to_str:N \prop_make_flat:N } { \token_to_str:N #1 } 9915
9916   } 9916
9917 } 9917
9918 \cs_generate_variant:Nn \prop_make_flat:N { c } 9918
9919 \cs_new_protected:Npn \__prop_make_flat:Nn #1#2 9919
9920 { 9920
9921   \exp_after:wN \__prop_clear:wNNN #1 \cs_set_eq:NN \cs_set_nopar:Npe #1 9921
9922   \cs_set_nopar:Npe #1 { \exp_not:n {#2} } 9922
9923 } 9923
9924 \cs_new_protected:Npn \prop_make_linked:N #1 9924
9925 { 9925
9926   \int_compare:nNnTF { \tex_currentgrouplevel:D } = 0 9926
9927   { 9927
9928     \__prop_if_flat:NTF #1 9928
9929     { \exp_args:NNo \__prop_make_linked:Nn #1 {#1} } { } 9929
9930   } 9930
9931   { 9931
9932     \msg_error:nnee { prop } { inner-make } 9932
9933     { \token_to_str:N \prop_make_linked:N } { \token_to_str:N #1 } 9933
9934   } 9934
9935 } 9935
9936 \cs_generate_variant:Nn \prop_make_linked:N { c } 9936

```

```
9937 \cs_new_protected:Npn \__prop_make_linked:Nn #1#2 9937
9938 { 9938
9939 \__prop_new_linked:N #1 9939
9940 \tl_set:Nn \l__prop_internal_tl {#2} 9940
9941 \exp_after:wN \__prop_set_eq:wNNNN #1 9941
9942 \cs_set_eq:NN \cs_set_nopar:Npe #1 \l__prop_internal_tl 9942
9943 } 9943
9944 \prop_new:N \l_tmpa_prop 9944
9945 \prop_new:N \l_tmpb_prop 9945
9946 \prop_new:N \g_tmpa_prop 9946
9947 \prop_new:N \g_tmpb_prop 9947
9948 \cs_new_protected:Npn \prop_concat:NNN 9948
9949 { \__prop_concat:NNNNN \cs_set_eq:NN \cs_set_nopar:Npe } 9949
9950 \cs_generate_variant:Nn \prop_concat:NNN { ccc } 9950
9951 \cs_new_protected:Npn \prop_gconcat:NNN 9951
9952 { \__prop_concat:NNNNN \cs_gset_eq:NN \cs_gset_nopar:Npe } 9952
9953 \cs_generate_variant:Nn \prop_gconcat:NNN { ccc } 9953
9954 \cs_new_protected:Npn \__prop_concat:NNNNN #1#2#3#4#5 9954
9955 { 9955
9956 \cs_if_eq:NNTF #3 #5 9956
9957 { \__prop_concat:nNNN \use_none:nnn #2 #3 #4 } 9957
9958 { 9958
9959 \__prop_set_eq:NNNN #1 #2 #3 #4 9959
9960 \__prop_concat:nNNN { } #2 #3 #5 9960
9961 } 9961
9962 } 9962
9963 \cs_new_protected:Npn \__prop_concat:nNNN #1#2#3#4 9963
9964 { 9964
9965 \cs_gset_eq:NN \__prop_tmp:w \__prop_pair:wn 9965
9966 \cs_gset_protected:Npn \__prop_pair:wn ##1 \s__prop 9966
9967 { \__prop_put:nNNnn {#1} #2 #3 {##1} } 9967
9968 \exp_last_unbraced:Nf \use_none:nn #4 9968
9969 \cs_gset_eq:NN \__prop_pair:wn \__prop_tmp:w 9969
9970 } 9970
9971 \cs_new_protected:Npn \prop_put_from_keyval:Nn #1 9971
9972 { \__prop_from_keyval:nn { \__prop_put:nNNnn { } \cs_set_nopar:Npe #1 } } 9972
9973 \cs_generate_variant:Nn \prop_put_from_keyval:Nn { c } 9973
9974 \cs_new_protected:Npn \prop_gput_from_keyval:Nn #1 9974
9975 { \__prop_from_keyval:nn { \__prop_put:nNNnn { } \cs_gset_nopar:Npe #1 } } 9975
9976 \cs_generate_variant:Nn \prop_gput_from_keyval:Nn { c } 9976
9977 \cs_new_protected:Npn \__prop_from_keyval:nn 9977
9978 { 9978
9979 \bool_if:NTF \l__kernel_keyval_allow_blank_keys_bool 9979
9980 { \__prop_from_keyval:Nnn \c_true_bool } 9980
9981 { \__prop_from_keyval:Nnn \c_false_bool } 9981
9982 } 9982
```

```
9983 \cs_new_protected:Npn \__prop_from_keyval:Nnn #1#2#3
9984 {
9985     \bool_set_eq:NN \l__kernel_keyval_allow_blank_keys_bool \c_true_bool
9986     \keyval_parse:nnn \__prop_missing_eq:n {#2} {#3}
9987     \bool_set_eq:NN \l__kernel_keyval_allow_blank_keys_bool #1
9988 }
9989 \cs_new_protected:Npn \__prop_missing_eq:n
9990 { \msg_error:nnn { prop } { prop-keyval } }
9991 \cs_new_protected:Npn \prop_set_from_keyval:Nn #1
9992 {
9993     \__prop_clear:NNN \cs_set_eq:NN \cs_set_nopar:Npe #1
9994     \prop_put_from_keyval:Nn #1
9995 }
9996 \cs_generate_variant:Nn \prop_set_from_keyval:Nn { c }
9997 \cs_new_protected:Npn \prop_gset_from_keyval:Nn #1
9998 {
9999     \__prop_clear:NNN \cs_gset_eq:NN \cs_gset_nopar:Npe #1
10000     \prop_gput_from_keyval:Nn #1
10001 }
10002 \cs_generate_variant:Nn \prop_gset_from_keyval:Nn { c }
10003 \cs_new_protected:Npn \prop_const_from_keyval:Nn #1
10004 {
10005     \prop_new:N #1
10006     \__prop_from_keyval:nn { \__prop_put:nNNnn { } \cs_gset_nopar:Npe #1 }
10007 }
10008 \cs_generate_variant:Nn \prop_const_from_keyval:Nn { c }
10009 \cs_new_protected:Npn \prop_const_linked_from_keyval:Nn #1
10010 {
10011     \prop_new_linked:N #1
10012     \__prop_from_keyval:nn { \__prop_put:nNNnn { } \cs_gset_nopar:Npe #1 }
10013 }
10014 \cs_generate_variant:Nn \prop_const_linked_from_keyval:Nn { c }
10015 \cs_new_protected:Npn \__prop_split:NnTFn #1#2
10016 {
10017     \exp_after:wN \__prop_split_aux:nNTFn
10018     \exp_after:wN { \tl_to_str:n {#2} } #1
10019 }
10020 \cs_new_protected:Npn \__prop_split_aux:nNTFn #1#2#3
10021 {
10022     \cs_set:Npn \__prop_split_flat:w \__prop_split_linked:w ##1
10023     \__prop_pair:wn #1 \s__prop ##2 ##3 \s__prop_mark ##4 ##5 \s__prop_stop
10024     { ##4 {#3} }
10025     \exp_after:wN \__prop_split_test:wn #2 \s__prop_mark \use_i:nnn
10026     \__prop_pair:wn #1 \s__prop { ? \fi: \__prop_split_wrong:Nw #2 }
10027     \s__prop_mark \use_ii:nnn
10028     \s__prop_stop
```



```

0029 }
0030 \cs_new:Npn \__prop_split_flat:w { }
0031 \cs_new_protected:Npn \__prop_split_test:wn #1 \s__prop #2
0032 {
0033     \if_meaning:w \__prop_chk:w #2 \exp_after:wN \__prop_split_flat:w \fi:
0034     \__prop_split_linked:w
0035 }
0036 \cs_new_protected:Npn \__prop_split_linked:w #1 \s__prop_stop #2#3 {#3}
0037 \cs_new_protected:Npn \__prop_split_wrong:Nw #1#2 \s__prop_stop #3#4
0038 {
0039     \prop_show:N #1
0040     \cs_gset_eq:NN #1 \c_empty_prop
0041     #3
0042 }
0043 \cs_new_protected:Npn \prop_get:NnN #1#2#3
0044 {
0045     \__prop_get:NnnTF #1 {#2}
0046     { \tl_set:Nn #3 { } } { \tl_set:Nn #3 { \q_no_value } }
0047 }
0048 \cs_generate_variant:Nn \prop_get:NnN { NV , Nv , Ne , c , cV , cv , ce }
0049 \cs_generate_variant:Nn \prop_get:NnN { No , Nx , co , cx }
0050 \cs_generate_variant:Nn \prop_get:NnN { cnc }
0051 \prg_new_protected_conditional:Npnn \prop_get:NnN #1#2#3 { T , F , TF }
0052 {
0053     \__prop_get:NnnTF #1 {#2}
0054     { \tl_set:Nn #3 { } } \prg_return_true: \prg_return_false:
0055 }
0056 \prg_generate_conditional_variant:Nnn \prop_get:NnN
0057 { NV , Nv , Ne , c , cV , cv , ce } { T , F , TF }
0058 \prg_generate_conditional_variant:Nnn \prop_get:NnN
0059 { No , Nx , co , cx } { T , F , TF }
0060 \prg_generate_conditional_variant:Nnn \prop_get:NnN
0061 { cnc } { T , F , TF }
0062 \cs_new_protected:Npn \__prop_get:NnnTF #1#2#3#4#5
0063 {
0064     \__prop_split:NnTFn #1 {#2}
0065     { #3 {##2} #4 }
0066     {#5}
0067     { \exp_after:wN \__prop_get_linked:w #1 {#2} {#3} {#4} {#5} }
0068 }
0069 \cs_new:Npn \__prop_get_linked:w
0070     \__prop_flatten:w #1 \s__prop #2#3#4#5#6#7
0071 {
0072     \if_cs_exist:w __prop ~ #2 ~ \tl_to_str:n {#4} \cs_end:
0073     \exp_after:wN \exp_after:wN \exp_after:wN \__prop_get_linked_aux:w
0074     \cs:w __prop ~ #2 ~ \tl_to_str:n {#4} \exp_after:wN \cs_end:

```

```
0075 \else:
0076 \exp_after:wN \__prop_get_linked_aux:w
0077 \fi:
0078 \s__prop_mark {#5} {#6}
0079 \s__prop { } \s__prop_mark \use_none:n {#7}
0080 \s__prop_stop
0081 }
0082 \cs_new:Npn \__prop_get_linked_aux:w
0083 #1 \s__prop #2 #3 \s__prop_mark #4 #5 #6 \s__prop_stop { #4 {#2} #5 }
0084 \cs_new:Npn \prop_item:Nn #1#2
0085 {
0086 \__prop_if_flat:NTF #1
0087 {
0088 \exp_args:NNo \prop_map_tokens:Nn #1
0089 {
0090 \exp_after:wN \__prop_item:nnn
0091 \exp_after:wN { \tl_to_str:n {#2} }
0092 }
0093 }
0094 { \exp_after:wN \__prop_get_linked:w #1 {#2} \exp_not:n { } { } }
0095 }
0096 \cs_new:Npn \__prop_item:nnn #1#2#3
0097 {
0098 \str_if_eq:eeT {#1} {#2}
0099 { \prop_map_break:n { \exp_not:n {#3} } }
0100 }
0101 \cs_generate_variant:Nn \prop_item:Nn { NV , No , Ne , c , cV , co , ce }
0102 \cs_new_protected:Npn \__prop_pop:NnNnTF #1#2#3#4#5#6#7
0103 {
0104 \__prop_split:NnTFn #1 {#2}
0105 {
0106 #4 #1 { \exp_not:n { \s__prop \__prop_chk:w ##1 ##3 } }
0107 #5 {##2}
0108 #6
0109 }
0110 {#7}
0111 {
0112 \exp_after:wN \__prop_pop_linked:wnNnTF #1 {#2}
0113 #3 #4 {#5} {#6} {#7}
0114 }
0115 }
0116 \cs_new_protected:Npn \__prop_pop_linked:wnNnTF
0117 \__prop_flatten:w #1 \s__prop #2#3#4#5#6#7
0118 {
0119 \if_cs_exist:w __prop ~ #2 ~ \tl_to_str:n {#4} \cs_end:
0120 \exp_after:wN \__prop_pop_linked:NNNn
```

```
0121 \cs:w __prop ~ #2 ~ \tl_to_str:n {#4} \cs_end:
0122 #5 #6 {#7}
0123 \else:
0124 \exp_after:wN \use_iii:nnn
0125 \fi:
0126 \use_i:nn
0127 }
0128 \cs_new_protected:Npn \__prop_pop_linked:NNNn #1#2#3#4
0129 {
0130 \if_meaning:w \scan_stop: #1
0131 \exp_after:wN \exp_after:wN \exp_after:wN \use_iii:nnn
0132 \else:
0133 \exp_after:wN \__prop_pop_linked:w #1 #1 #2 #3 {#4}
0134 \fi:
0135 }
0136 \cs_new_protected:Npn \__prop_pop_linked:w
0137 \use_none:n #1#2 \s__prop #3#4#5#6#7#8
0138 {
0139 #6 #5 \tex_undefined:D
0140 #7 #1
0141 {
0142 \exp_after:wN \__prop_pop_linked_prev:w #1
0143 \exp_not:N #4
0144 }
0145 #7 #4
0146 {
0147 \exp_not:n { \use_none:n #1 }
0148 \exp_not:f { \exp_after:wN \__prop_pop_linked_next:w #4 }
0149 }
0150 #8 {#3}
0151 }
0152 \cs_new:Npn \__prop_pop_linked_prev:w #1 \s__prop #2#3
0153 { \exp_not:n { #1 \s__prop {#2} } }
0154 \cs_new:Npn \__prop_pop_linked_next:w \use_none:n #1 { \exp_stop_f: }
0155 \cs_new_protected:Npn \prop_remove:Nn #1#2
0156 {
0157 \__prop_pop:NnNNnTF #1 {#2}
0158 \cs_set_eq:NN \cs_set_nopar:Npe
0159 \use_none:n { } { }
0160 }
0161 \cs_new_protected:Npn \prop_gremove:Nn #1#2
0162 {
0163 \__prop_pop:NnNNnTF #1 {#2}
0164 \cs_gset_eq:NN \cs_gset_nopar:Npe
0165 \use_none:n { } { }
0166 }
```

```
0167 \cs_generate_variant:Nn \prop_remove:Nn { NV , Ne , c , cV , ce }
0168 \cs_generate_variant:Nn \prop_gremove:Nn { NV , Ne , c , cV , ce }
0169 \cs_new_protected:Npn \prop_pop:NnN #1#2#3
0170 {
0171     \__prop_pop:NnNnTF #1 {#2}
0172     \cs_set_eq:NN \cs_set_nopar:Npe
0173     { \tl_set:Nn #3 } { } { \tl_set:Nn #3 { \q_no_value } }
0174 }
0175 \cs_new_protected:Npn \prop_gpop:NnN #1#2#3
0176 {
0177     \__prop_pop:NnNnTF #1 {#2}
0178     \cs_gset_eq:NN \cs_gset_nopar:Npe
0179     { \tl_set:Nn #3 } { } { \tl_set:Nn #3 { \q_no_value } }
0180 }
0181 \cs_generate_variant:Nn \prop_pop:NnN { NV , No }
0182 \cs_generate_variant:Nn \prop_pop:NnN { c , cV , co }
0183 \cs_generate_variant:Nn \prop_gpop:NnN { NV , No }
0184 \cs_generate_variant:Nn \prop_gpop:NnN { c , cV , co }
0185 \prg_new_protected_conditional:Npnn \prop_pop:NnN #1#2#3 { T , F , TF }
0186 {
0187     \__prop_pop:NnNnTF #1 {#2}
0188     \cs_set_eq:NN \cs_set_nopar:Npe
0189     { \tl_set:Nn #3 } \prg_return_true: \prg_return_false:
0190 }
0191 \prg_new_protected_conditional:Npnn \prop_gpop:NnN #1#2#3 { T , F , TF }
0192 {
0193     \__prop_pop:NnNnTF #1 {#2}
0194     \cs_gset_eq:NN \cs_gset_nopar:Npe
0195     { \tl_set:Nn #3 } \prg_return_true: \prg_return_false:
0196 }
0197 \prg_generate_conditional_variant:Nnn \prop_pop:NnN
0198 { NV , No , c , cV , co } { T , F , TF }
0199 \prg_generate_conditional_variant:Nnn \prop_gpop:NnN
0200 { NV , No , c , cV , co } { T , F , TF }
0201 \cs_new_protected:Npn \prop_put:Nnn
0202 { \__prop_put:nNNnn { } \cs_set_nopar:Npe }
0203 \cs_new_protected:Npn \prop_gput:Nnn
0204 { \__prop_put:nNNnn { } \cs_gset_nopar:Npe }
0205 \cs_new_protected:Npn \prop_put_if_not_in:Nnn
0206 { \__prop_put:nNNnn \use_none:nnn \cs_set_nopar:Npe }
0207 \cs_new_protected:Npn \prop_gput_if_not_in:Nnn
0208 { \__prop_put:nNNnn \use_none:nnn \cs_gset_nopar:Npe }
0209 \cs_generate_variant:Nn \prop_put:Nnn
0210 {
0211     NnV , Nnv , Nne , NV , NVV , NVv , NVe ,
0212     Nv , NvV , Nvv , Nve , Ne , NeV , Nev , Nee
```

```
0213 }
0214 \cs_generate_variant:Nn \prop_put:Nnn
0215 { Nno , No , Noo , Nnx , NVx , NxV , Nxx }
0216 \cs_generate_variant:Nn \prop_put:Nnn
0217 {
0218   c , cnV , cnv , cne , cV , cVV , cVv , cVe ,
0219   cv , cvV , cvv , cve , ce , ceV , cev , cee
0220 }
0221 \cs_generate_variant:Nn \prop_put:Nnn
0222 { cno , co , coo , cnx , cVx , cxV , cxx }
0223 \cs_generate_variant:Nn \prop_gput:Nnn
0224 {
0225   NnV , Nnv , Nne , NV , NVV , NVv , NVe ,
0226   Nv , NvV , Nvv , Nve , Ne , NeV , Nev , Nee
0227 }
0228 \cs_generate_variant:Nn \prop_gput:Nnn
0229 { Nno , No , Noo , Nnx , NVx , NxV , Nxx }
0230 \cs_generate_variant:Nn \prop_gput:Nnn
0231 {
0232   c , cnV , cnv , cne , cV , cVV , cVv , cVe ,
0233   cv , cvV , cvv , cve , ce , ceV , cev , cee
0234 }
0235 \cs_generate_variant:Nn \prop_gput:Nnn
0236 { cno , co , coo , cnx , cVx , cxV , cxx }
0237 \cs_generate_variant:Nn \prop_put_if_not_in:Nnn
0238 {
0239   NnV , Nnv , Nne , NV , NVV , NVv , NVe ,
0240   Nv , NvV , Nvv , Nve , Ne , NeV , Nev , Nee ,
0241   c , cnV , cnv , cne , cV , cVV , cVv , cVe ,
0242   cv , cvV , cvv , cve , ce , ceV , cev , cee
0243 }
0244 \cs_generate_variant:Nn \prop_gput_if_not_in:Nnn
0245 {
0246   NnV , Nnv , Nne , NV , NVV , NVv , NVe ,
0247   Nv , NvV , Nvv , Nve , Ne , NeV , Nev , Nee ,
0248   c , cnV , cnv , cne , cV , cVV , cVv , cVe ,
0249   cv , cvV , cvv , cve , ce , ceV , cev , cee
0250 }
0251 \cs_new_protected:Npn \__prop_put:nNNnn #1#2#3#4#5
0252 {
0253   \tl_set:Nn \l__prop_internal_tl
0254   {
0255     \exp_not:N \__prop_pair:wn \tl_to_str:n {#4}
0256     \s__prop { \exp_not:n {#5} }
0257   }
0258   \__prop_split:NnTFn #3 {#4}
```

```
0259 {
0260     #1 #2 #3
0261     {
0262         \s__prop \__prop_chk:w \exp_not:n {##1}
0263         \l__prop_internal_tl \exp_not:n {##3}
0264     }
0265 }
0266 { #2 #3 { \exp_not:o {#3} \l__prop_internal_tl } }
0267 { \exp_after:wN \__prop_put_linked:wnnN #3 {#4} {#1} #2 }
0268 }
0269 \cs_new_protected:Npn \__prop_put_linked:wnnN
0270     \__prop_flatten:w #1 \s__prop #2#3#4
0271 {
0272     \exp_after:wN \__prop_put_linked:NNnN
0273     \cs:w __prop ~ #2 ~ \tl_to_str:n {#4} \cs_end:
0274     #1
0275 }
0276 \cs_new_protected:Npn \__prop_put_linked:NNnN #1#2#3#4
0277 {
0278     \if_meaning:w \scan_stop: #1
0279     \exp_after:wN \__prop_put_linked_new:w #2 #1 #2 #4
0280     \else:
0281     \exp_after:wN \__prop_put_linked_old:w #1 { #3 #4 #1 }
0282     \fi:
0283 }
0284 \cs_new_protected:Npn \__prop_put_linked_new:w
0285     \use_none:n #1#2#3#4
0286 {
0287     #4 #1
0288     {
0289         \exp_after:wN \__prop_pop_linked_prev:w #1
0290         \exp_not:N #2
0291     }
0292     #4 #2
0293     {
0294         \exp_not:n { \use_none:n #1 }
0295         \l__prop_internal_tl
0296         \exp_not:N #3
0297     }
0298     #4 #3 { \exp_not:n { \use_none:n #2 } }
0299 }
0300 \cs_new_protected:Npn \__prop_put_linked_old:w
0301     \use_none:n #1#2 \s__prop #3#4#5
0302 {
0303     #5
0304     {
```



```
0305         \exp_not:n { \use_none:n #1 }
0306         \l__prop_internal_tl
0307         \exp_not:N #4
0308     }
0309 }
0310 \prg_new_eq_conditional:NNn \prop_if_exist:N \cs_if_exist:N
0311 { TF , T , F , p }
0312 \prg_new_eq_conditional:NNn \prop_if_exist:c \cs_if_exist:c
0313 { TF , T , F , p }
0314 \prg_new_conditional:Npnn \prop_if_empty:N #1 { p , T , F , TF }
0315 {
0316     \if_meaning:w #1 \c_empty_prop
0317     \prg_return_true:
0318 \else:
0319     \exp_after:wN \__prop_if_empty_return:w #1
0320     \__prop_flatten:w 2 \s__prop 34 \s__prop_stop
0321 \fi:
0322 }
0323 \cs_new:Npn \__prop_if_empty_return:w
0324 #1 \__prop_flatten:w #2 \s__prop #3#4#5 \s__prop_stop
0325 {
0326     \if_meaning:w #2 #4
0327     \prg_return_true:
0328 \else:
0329     \prg_return_false:
0330 \fi:
0331 }
0332 \prg_generate_conditional_variant:Nnn \prop_if_empty:N
0333 { c } { p , T , F , TF }
0334 \prg_new_conditional:Npnn \prop_if_in:Nn #1#2 { p , T , F , TF }
0335 {
0336     \__prop_if_flat:NTF #1
0337     {
0338         \exp_after:wN \prop_map_tokens:Nn \exp_after:wN #1
0339         {
0340             \exp_after:wN \__prop_if_in_flat:nnn
0341             \exp_after:wN { \tl_to_str:n {#2} }
0342         }
0343         \prg_return_false:
0344     }
0345     {
0346         \exp_after:wN \__prop_get_linked:w #1 {#2}
0347         \use_none:n \prg_return_true: \prg_return_false:
0348     }
0349 }
0350 \cs_new:Npn \__prop_if_in_flat:nnn #1#2#3
```

```
0351 {
0352     \str_if_eq:eeT {#1} {#2}
0353     { \prop_map_break:n { \use_i:nn \prg_return_true: } }
0354 }
0355 \prg_generate_conditional_variant:Nnn \prop_if_in:Nn
0356 { NV , Ne , No , c , cV , ce , co } { p , T , F , TF }
0357 \cs_new:Npn \prop_map_function:NN #1#2
0358 {
0359     \exp_last_unbraced:Nnf
0360     \use_i:nnn { \__prop_map_function:Nw #2 } #1
0361     \__prop_pair:wn \fi: \prop_map_break: \s__prop { }
0362     \__prop_pair:wn \fi: \prop_map_break: \s__prop { }
0363     \__prop_pair:wn \fi: \prop_map_break: \s__prop { }
0364     \__prop_pair:wn \fi: \prop_map_break: \s__prop { }
0365     \prg_break_point:Nn \prop_map_break: { }
0366 }
0367 \cs_new:Npn \__prop_map_function:Nw #1
0368     \__prop_pair:wn #2 \s__prop #3
0369     \__prop_pair:wn #4 \s__prop #5
0370     \__prop_pair:wn #6 \s__prop #7
0371     \__prop_pair:wn #8 \s__prop #9
0372 {
0373     \if_false: #2 \fi: #1 {#2} {#3}
0374     \if_false: #4 \fi: #1 {#4} {#5}
0375     \if_false: #6 \fi: #1 {#6} {#7}
0376     \if_false: #8 \fi: #1 {#8} {#9}
0377     \__prop_map_function:Nw #1
0378 }
0379 \cs_generate_variant:Nn \prop_map_function:NN { Nc , c , cc }
0380 \cs_new_protected:Npn \prop_map_inline:Nn #1#2
0381 {
0382     \cs_gset_eq:cN
0383     { __prop_map_ \int_use:N \g__kernel_prg_map_int :wn } \__prop_pair:wn
0384     \int_gincr:N \g__kernel_prg_map_int
0385     \cs_gset_protected:Npn \__prop_pair:wn ##1 \s__prop ##2 {#2}
0386     \exp_last_unbraced:Nf \use_none:nn #1
0387     \prg_break_point:Nn \prop_map_break:
0388     {
0389         \int_gdecr:N \g__kernel_prg_map_int
0390         \cs_gset_eq:Nc \__prop_pair:wn
0391         { __prop_map_ \int_use:N \g__kernel_prg_map_int :wn }
0392     }
0393 }
0394 \cs_generate_variant:Nn \prop_map_inline:Nn { c }
0395 \cs_new:Npn \prop_map_tokens:Nn #1#2
0396 {
```

```
0397 \exp_last_unbraced:Nnf
0398 \use_i:nnn { \__prop_map_tokens:nw {#2} } #1
0399 \__prop_pair:wn \fi: \prop_map_break: \s__prop { }
0400 \__prop_pair:wn \fi: \prop_map_break: \s__prop { }
0401 \__prop_pair:wn \fi: \prop_map_break: \s__prop { }
0402 \__prop_pair:wn \fi: \prop_map_break: \s__prop { }
0403 \prg_break_point:Nn \prop_map_break: { }
0404 }
0405 \cs_new:Npn \__prop_map_tokens:nw #1
0406 \__prop_pair:wn #2 \s__prop #3
0407 \__prop_pair:wn #4 \s__prop #5
0408 \__prop_pair:wn #6 \s__prop #7
0409 \__prop_pair:wn #8 \s__prop #9
0410 {
0411 \if_false: #2 \fi: \use:n {#1} {#2} {#3}
0412 \if_false: #4 \fi: \use:n {#1} {#4} {#5}
0413 \if_false: #6 \fi: \use:n {#1} {#6} {#7}
0414 \if_false: #8 \fi: \use:n {#1} {#8} {#9}
0415 \__prop_map_tokens:nw {#1}
0416 }
0417 \cs_generate_variant:Nn \prop_map_tokens:Nn { c }
0418 \cs_new:Npn \prop_map_break:
0419 { \prg_map_break:Nn \prop_map_break: { } }
0420 \cs_new:Npn \prop_map_break:n
0421 { \prg_map_break:Nn \prop_map_break: }
0422 \cs_new:Npn \prop_count:N #1
0423 {
0424 \int_eval:n
0425 {
0426 0
0427 \prop_map_function:NN #1 \__prop_count:nn
0428 }
0429 }
0430 \cs_new:Npn \__prop_count:nn #1#2 { + 1 }
0431 \cs_generate_variant:Nn \prop_count:N { c }
0432 \cs_new:Npn \prop_to_keyval:N #1
0433 {
0434 \__kernel_exp_not:w
0435 \prop_if_empty:NTF #1
0436 { {} }
0437 {
0438 \exp_after:wN \exp_after:wN \exp_after:wN
0439 {
0440 \tex_expanded:D
0441 {
0442 \exp_not:N \use_none:n
```

```
0443         \prop_map_function:NN #1 \__prop_to_keyval:nn
0444     }
0445 }
0446 }
0447 }
0448 \cs_new:Npn \__prop_to_keyval:nn #1#2
0449 { , ~ {#1} =~ { \__kernel_exp_not:w {#2} } }
0450 \cs_new_protected:Npn \prop_show:N { \__prop_show:NN \msg_show:nneeee }
0451 \cs_generate_variant:Nn \prop_show:N { c }
0452 \cs_new_protected:Npn \prop_log:N { \__prop_show:NN \msg_log:nneeee }
0453 \cs_generate_variant:Nn \prop_log:N { c }
0454 \cs_new_protected:Npn \__prop_show:NN #1#2
0455 {
0456     \__kernel_chk_tl_type:NnnT #2 { prop }
0457     {
0458         \__prop_if_flat:NTF #2
0459         {
0460             \s__prop \__prop_chk:w
0461             \exp_after:wN \__prop_show_flat:w #2
0462             \s__prop { }
0463             \__prop_pair:wn \q__prop_recursion_tail \s__prop { }
0464             \q__prop_recursion_stop
0465         }
0466         { \exp_after:wN \__prop_show_linked:w #2 \s__prop ! ? \s__prop_stop }
0467     }
0468     {
0469         \__prop_if_flat:NTF #2
0470         { \__prop_show_finally:NNn #1 #2 { flat } }
0471         {
0472             \tl_set:Nn \l__prop_internal_tl { #1 #2 }
0473             \exp_after:wN \__prop_show_prepare:w #2 #2
0474         }
0475     }
0476 }
0477 \cs_new:Npn \__prop_show_flat:w #1 \__prop_pair:wn #2 \s__prop #3
0478 {
0479     \__prop_if_recursion_tail_stop:n {#2}
0480     \exp_not:N \__prop_pair:wn \tl_to_str:n {#2} \s__prop \exp_not:n { {#3} }
0481     \__prop_show_flat:w
0482 }
0483 \cs_new:Npn \__prop_show_linked:w #1 \s__prop #2#3#4 \s__prop_stop
0484 {
0485     \exp_not:N \__prop_flatten:w
0486     \exp_not:c { __prop ~ \tl_to_str:n {#2} }
0487     \s__prop { \tl_to_str:n {#2} }
0488     \exp_not:n {#3}
```

```
0489 }
0490 \cs_new_protected:Npn \__prop_show_finally:NNn #1#2#3
0491 {
0492   #1 { prop } { show }
0493   { \token_to_str:N #2 }
0494   { \prop_map_function:NN #2 \msg_show_item:nn }
0495   {#3} { }
0496 }
0497 \cs_new_protected:Npn \__prop_show_prepare:w
0498   \__prop_flatten:w #1 \s__prop #2#3#4
0499 {
0500   \use:e
0501   {
0502     \cs_set_nopar:Npn \exp_not:N \__prop_tmp:w
0503       ##1 \token_to_str:N #1 ##2 \s__prop_mark ##3 \s__prop_stop
0504       {
0505         \exp_not:N \tl_if_empty:nTF {##1}
0506         {
0507           \exp_not:N \tl_if_head_is_space:nTF {##2}
0508           { \exp_not:N \exp_args:Nf \__prop_show_loop_key:wNNN }
0509           { \exp_not:N \tl_if_empty:nTF }
0510           {##2}
0511         }
0512         { \exp_not:N \use_ii:nn }
0513         \__prop_show_end:NNN
0514         \__prop_show_bad_name:NNN
0515       }
0516     }
0517   \exp_last_unbraced:NNNo \__prop_show_loop:NNw #1 #4 #4
0518 }
0519 \cs_new_protected:Npn \__prop_show_loop:NNw #1#2 #3 \s__prop #4#5
0520 {
0521   \exp_last_two_unbraced:Noo \__prop_tmp:w
0522   { \token_to_str:N #5 \s__prop_mark }
0523   { \token_to_str:N #1 \s__prop_mark \s__prop_stop }
0524   #1 #2 #5
0525 }
0526 \cs_new_protected:Npn \__prop_show_bad_name:NNN #1#2#3
0527 {
0528   \msg_error:nneeee { prop } { bad-link }
0529   { \tl_tail:N \l__prop_internal_tl }
0530   { \token_to_str:N #2 }
0531   { \token_to_str:N #3 }
0532   { \token_to_str:N #1 }
0533 }
0534 \cs_new_protected:Npn \__prop_show_end:NNN #1#2#3
```

```

0535 {
0536 \__kernel_chk_tl_type:NnnT #3
0537 { \tl_tail:N \l__prop_internal_tl prop~entry }
0538 { \exp_not:n { \use_none:n #2 } }
0539 {
0540 \exp_after:wN \__prop_show_finally:NNn
0541 \l__prop_internal_tl { linked }
0542 }
0543 }
0544 \cs_new_protected:Npn \__prop_show_loop_key:wNNN #1#2#3#4#5#6
0545 {
0546 \__kernel_chk_tl_type:NnnT #6
0547 { \tl_tail:N \l__prop_internal_tl prop~entry }
0548 {
0549 \exp_not:n { \use_none:n #5 }
0550 \exp_after:wN \__prop_show_flat:w #6 \s__prop { }
0551 \__prop_pair:wn \q__prop_recursion_tail \s__prop { }
0552 \q__prop_recursion_stop
0553 \tl_item:Nn #6 { -1 }
0554 }
0555 { \exp_last_unbraced:NNNo \__prop_show_loop:NNw #4 #6 #6 }
0556 }
0557 %% File: l3msg.dtx
0558 \tl_new:N \l__msg_internal_tl
0559 \str_new:N \l__msg_name_str
0560 \str_new:N \l__msg_text_str
0561 \scan_new:N \s__msg_mark
0562 \scan_new:N \s__msg_stop
0563 \cs_new:Npn \__msg_use_none_delimit_by_s_stop:w #1 \s__msg_stop { }
0564 \tl_const:Nn \c__msg_text_prefix_tl { msg~text~>~ }
0565 \tl_const:Nn \c__msg_more_text_prefix_tl { msg~extra~text~>~ }
0566 \prg_new_conditional:Npnn \msg_if_exist:nn #1#2 { p , T , F , TF }
0567 {
0568 \cs_if_exist:cTF { \c__msg_text_prefix_tl #1 / #2 }
0569 { \prg_return_true: } { \prg_return_false: }
0570 }
0571 \cs_new_protected:Npn \__msg_chk_free:nn #1#2
0572 {
0573 \msg_if_exist:nnT {#1} {#2}
0574 {
0575 \msg_error:nnnn { msg } { already-defined }
0576 {#1} {#2}
0577 }
0578 }
0579 \cs_new_protected:Npn \msg_new:nnnn #1#2#3#4
0580 {

```



```
0581 \_msg_chk_free:nn {#1} {#2} 1058
0582 \cs_gset:cpn { \c__msg_text_prefix_tl #1 / #2 } 1058
0583 ##1##2##3##4 {#3} 1058
0584 \cs_gset:cpn { \c__msg_more_text_prefix_tl #1 / #2 } 1058
0585 ##1##2##3##4 {#4} 1058
0586 } 1058
0587 \cs_generate_variant:Nn \msg_new:nnnn { nnee , nxxx } 1058
0588 \cs_new_protected:Npn \msg_new:nnn #1#2#3 1058
0589 { \msg_new:nnnn {#1} {#2} {#3} { } } 1058
0590 \cs_generate_variant:Nn \msg_new:nnn { nne , nnx } 1059
0591 \cs_new_protected:Npn \msg_set:nnnn #1#2#3#4 1059
0592 { 1059
0593 \cs_set:cpn { \c__msg_text_prefix_tl #1 / #2 } 1059
0594 ##1##2##3##4 {#3} 1059
0595 \cs_set:cpn { \c__msg_more_text_prefix_tl #1 / #2 } 1059
0596 ##1##2##3##4 {#4} 1059
0597 } 1059
0598 \cs_new_protected:Npn \msg_set:nnn #1#2#3 1059
0599 { \msg_set:nnnn {#1} {#2} {#3} { } } 1059
0600 \tl_const:Nn \c__msg_coding_error_text_tl 1060
0601 { 1060
0602 This~is~a~coding~error. 1060
0603 \\ \\ 1060
0604 } 1060
0605 \tl_const:Nn \c__msg_continue_text_tl 1060
0606 { Type~<return>~to~continue } 1060
0607 \tl_const:Nn \c__msg_critical_text_tl 1060
0608 { Reading~the~current~file~'\g_file_curr_name_str'~will~stop. } 1060
0609 \tl_const:Nn \c__msg_fatal_text_tl 1060
0610 { This~is~a~fatal~error:~LaTeX~will~abort. } 1061
0611 \tl_const:Nn \c__msg_help_text_tl 1061
0612 { For~immediate~help~type~H~<return> } 1061
0613 \tl_const:Nn \c__msg_no_info_text_tl 1061
0614 { 1061
0615 LaTeX~does~not~know~anything~more~about~this~error,~sorry. 1061
0616 \c__msg_return_text_tl 1061
0617 } 1061
0618 \tl_const:Nn \c__msg_on_line_text_tl { on~line } 1061
0619 \tl_const:Nn \c__msg_return_text_tl 1061
0620 { 1062
0621 \\ \\ 1062
0622 Try~typing~<return>~to~proceed. 1062
0623 \\ 1062
0624 If~that~doesn't~work,~type~X~<return>~to~quit. 1062
0625 } 1062
0626 \tl_const:Nn \c__msg_trouble_text_tl 1062
```

```
0627 {
0628     \\ \\
0629     More~errors~will~almost~certainly~follow: \\
0630     the~LaTeX~run~should~be~aborted.
0631 }
0632 \cs_new:Npn \msg_line_number: { \int_use:N \tex_inputlineno:D }
0633 \cs_gset:Npn \msg_line_context:
0634 {
0635     \c__msg_on_line_text_tl
0636     \c_space_tl
0637     \msg_line_number:
0638 }
0639 \cs_new_protected:Npn \__msg_interrupt:NnnnN #1#2#3#4#5
0640 {
0641     \str_set:Ne \l__msg_text_str { #1 {#2} }
0642     \str_set:Ne \l__msg_name_str { \msg_module_name:n {#2} }
0643     \cs_if_eq:cNTF
0644         { \c__msg_more_text_prefix_tl #2 / #3 }
0645         \__msg_no_more_text:nnnn
0646     {
0647         \__msg_interrupt_wrap:nnn
0648         { \use:c { \c__msg_text_prefix_tl #2 / #3 } #4 }
0649         { \c__msg_continue_text_tl }
0650         {
0651             \c__msg_no_info_text_tl
0652             \tl_if_empty:NF #5
0653             { \\ \\ #5 }
0654         }
0655     }
0656     {
0657         \__msg_interrupt_wrap:nnn
0658         { \use:c { \c__msg_text_prefix_tl #2 / #3 } #4 }
0659         { \c__msg_help_text_tl }
0660         {
0661             \use:c { \c__msg_more_text_prefix_tl #2 / #3 } #4
0662             \tl_if_empty:NF #5
0663             { \\ \\ #5 }
0664         }
0665     }
0666 }
0667 \cs_new:Npn \__msg_no_more_text:nnnn #1#2#3#4 { }
0668 \cs_new_protected:Npn \__msg_interrupt_wrap:nnn #1#2#3
0669 {
0670     \iow_wrap:nnnN { \\ #3 } { } { } \__msg_interrupt_more_text:n
0671     \group_begin:
0672     \int_sub:Nn \l_iow_line_count_int { 2 }
```

```
0673 \iow_wrap:n:nN { \l__msg_text_str : ~ #1 }
0674 {
0675     ( \l__msg_name_str )
0676     \prg_replicate:nn
0677     {
0678         \str_count:N \l__msg_text_str
0679         - \str_count:N \l__msg_name_str
0680         + 2
0681     }
0682     { ~ }
0683 }
0684 { } \__msg_interrupt_text:n
0685 \iow_wrap:nnnN { \l__msg_internal_tl \\ \\ #2 } { } { }
0686 \__msg_interrupt:n
0687 }
0688 \cs_new_protected:Npn \__msg_interrupt_text:n #1
0689 {
0690     \group_end:
0691     \tl_set:Nn \l__msg_internal_tl {#1}
0692 }
0693 \cs_new_protected:Npn \__msg_interrupt_more_text:n #1
0694 { \exp_args:Ne \tex_errhelp:D { #1 \iow_newline: } }
0695 \group_begin:
0696     \char_set_lccode:nn { 38 } { 32 } % &
0697     \char_set_lccode:nn { 46 } { 32 } % .
0698     \char_set_lccode:nn { 123 } { 32 } % {
0699     \char_set_lccode:nn { 125 } { 32 } % }
0700     \char_set_catcode_active:N \&
0701 \tex_lowercase:D
0702 {
0703     \group_end:
0704     \cs_new_protected:Npn \__msg_interrupt:n #1
0705     {
0706         \iow_term:n { }
0707         \__kernel_iow_with:Nnn \tex_newlinechar:D { ``^^J }
0708         {
0709             \__kernel_iow_with:Nnn \tex_errorcontextlines:D { -1 }
0710             {
0711                 \group_begin:
0712                 \cs_set_protected:Npn &
0713                 {
0714                     \tex_errmessage:D
0715                     {
0716                         #1
0717                         \use_none:n
0718                         { ..... }
```

```
0719         }
0720     }
0721     \exp_after:wN
0722     \group_end:
0723     &
0724 }
0725 }
0726 }
0727 }
0728 \int_gset:Nn \tex_errorcontextlines:D { -1 }
0729 \cs_new:Npn \msg_fatal_text:n #1
0730 {
0731     Fatal ~
0732     \msg_error_text:n {#1}
0733 }
0734 \cs_new:Npn \msg_critical_text:n #1
0735 {
0736     Critical ~
0737     \msg_error_text:n {#1}
0738 }
0739 \cs_new:Npn \msg_error_text:n #1
0740 { \__msg_text:nn {#1} { Error } }
0741 \cs_new:Npn \msg_warning_text:n #1
0742 { \__msg_text:nn {#1} { Warning } }
0743 \cs_new:Npn \msg_info_text:n #1
0744 { \__msg_text:nn {#1} { Info } }
0745 \cs_new:Npn \__msg_text:nn #1#2
0746 {
0747     \exp_args:Nf \__msg_text:n { \msg_module_type:n {#1} }
0748     \exp_args:Nf \__msg_text:n { \msg_module_name:n {#1} }
0749     #2
0750 }
0751 \cs_new:Npn \__msg_text:n #1
0752 {
0753     \tl_if_blank:nF {#1}
0754     { #1 ~ }
0755 }
0756 \prop_new:N \g_msg_module_name_prop
0757 \prop_new:N \g_msg_module_type_prop
0758 \prop_gput:Nnn \g_msg_module_type_prop { LaTeX } { }
0759 \cs_new:Npn \msg_module_type:n #1
0760 {
0761     \prop_if_in:NnTF \g_msg_module_type_prop {#1}
0762     { \prop_item:Nn \g_msg_module_type_prop {#1} }
0763     { Package }
0764 }
```

```
0765 \cs_new:Npn \msg_module_name:n #1
0766 {
0767   \prop_if_in:NnTF \g_msg_module_name_prop {#1}
0768     { \prop_item:Nn \g_msg_module_name_prop {#1} }
0769     {#1}
0770 }
0771 \cs_new:Npn \msg_see_documentation_text:n #1
0772 {
0773   See~the~ \msg_module_name:n {#1} ~
0774   documentation~for~further~information.
0775 }
0776 \group_begin:
0777   \cs_set_protected:Npn \__msg_class_new:nn #1#2
0778   {
0779     \prop_new:c { l__msg_redirect_ #1 _prop }
0780     \cs_new_protected:cpn { __msg_ #1 _code:nnnnnn }
0781       ##1##2##3##4##5##6 {#2}
0782     \cs_new_protected:cpn { msg_ #1 :nnnnnn } ##1##2##3##4##5##6
0783       {
0784         \use:e
0785         {
0786           \exp_not:n { \__msg_use:nnnnnnn {#1} {##1} {##2} }
0787           { \tl_to_str:n {##3} } { \tl_to_str:n {##4} }
0788           { \tl_to_str:n {##5} } { \tl_to_str:n {##6} }
0789         }
0790       }
0791     \cs_new_protected:cpe { msg_ #1 :nnnnn } ##1##2##3##4##5
0792       { \exp_not:c { msg_ #1 :nnnnnn } {##1} {##2} {##3} {##4} {##5} { } }
0793     \cs_new_protected:cpe { msg_ #1 :nnnn } ##1##2##3##4
0794       { \exp_not:c { msg_ #1 :nnnnnn } {##1} {##2} {##3} {##4} { } { } }
0795     \cs_new_protected:cpe { msg_ #1 :nnn } ##1##2##3
0796       { \exp_not:c { msg_ #1 :nnnnnn } {##1} {##2} {##3} { } { } { } }
0797     \cs_new_protected:cpe { msg_ #1 :nn } ##1##2
0798       { \exp_not:c { msg_ #1 :nnnnnn } {##1} {##2} { } { } { } { } }
0799     \cs_generate_variant:cn { msg_ #1 :nnn }
0800       { nnV , nne , nnx }
0801     \cs_generate_variant:cn { msg_ #1 :nnnn }
0802       { nnVV , nnVn , nnnV , nnne , nnnx , nnee , nnxx }
0803     \cs_generate_variant:cn { msg_ #1 :nnnnn }
0804       { nnnee , nnnxx , nneee , nnxxx }
0805     \cs_generate_variant:cn { msg_ #1 :nnnnnn } { nneeee , nnxxxx }
0806   }
0807   \__msg_class_new:nn { fatal }
0808   {
0809     \__msg_interrupt:NnnnN
0810     \msg_fatal_text:n {#1} {#2}
```

```
0811         { {#3} {#4} {#5} {#6} }
0812         \c__msg_fatal_text_tl
0813         \__msg_fatal_exit:
0814     }
0815     \cs_new_protected:Npn \__msg_fatal_exit:
0816     {
0817         \tex_batchmode:D
0818         \tex_read:D -1 to \l__msg_internal_tl
0819     }
0820     \__msg_class_new:nn { critical }
0821     {
0822         \__msg_interrupt:NnnnN
0823         \msg_critical_text:n {#1} {#2}
0824         { {#3} {#4} {#5} {#6} }
0825         \c__msg_critical_text_tl
0826         \tex_endinput:D
0827     }
0828     \cs_undefine:N \msg_error:nnee
0829     \cs_undefine:N \msg_error:nne
0830     \cs_undefine:N \msg_error:nn
0831     \__msg_class_new:nn { error }
0832     {
0833         \__msg_interrupt:NnnnN
0834         \msg_error_text:n {#1} {#2}
0835         { {#3} {#4} {#5} {#6} }
0836         \c_empty_tl
0837     }
0838     \cs_new_protected:Npn \__msg_info_aux:NNnnnnnn #1#2#3#4#5#6#7#8
0839     {
0840         \str_set:Ne \l__msg_text_str { #2 {#3} }
0841         \str_set:Ne \l__msg_name_str { \msg_module_name:n {#3} }
0842         #1 { }
0843         \iow_wrap:nenN
0844         {
0845             \l__msg_text_str : ~
0846             \use:c { \c__msg_text_prefix_tl #3 / #4 } {#5} {#6} {#7} {#8}
0847         }
0848         {
0849             ( \l__msg_name_str )
0850             \prg_replicate:nn
0851             {
0852                 \str_count:N \l__msg_text_str
0853                 - \str_count:N \l__msg_name_str
0854             }
0855             { ~ }
0856         }
```

```
0857     { } #1
0858     #1 { }
0859 }
0860 \__msg_class_new:nn { warning }
0861 {
0862     \__msg_info_aux:NNnnnnnn \iow_term:n \msg_warning_text:n
0863     {#1} {#2} {#3} {#4} {#5} {#6}
0864 }
0865 \__msg_class_new:nn { note }
0866 {
0867     \__msg_info_aux:NNnnnnnn \iow_term:n \msg_info_text:n
0868     {#1} {#2} {#3} {#4} {#5} {#6}
0869 }
0870 \__msg_class_new:nn { info }
0871 {
0872     \__msg_info_aux:NNnnnnnn \iow_log:n \msg_info_text:n
0873     {#1} {#2} {#3} {#4} {#5} {#6}
0874 }
0875 \__msg_class_new:nn { log }
0876 {
0877     \iow_wrap:nnnN
0878     { \use:c { \c__msg_text_prefix_tl #1 / #2 } {#3} {#4} {#5} {#6} }
0879     { } { } \iow_log:n
0880 }
0881 \__msg_class_new:nn { term }
0882 {
0883     \iow_wrap:nnnN
0884     { \use:c { \c__msg_text_prefix_tl #1 / #2 } {#3} {#4} {#5} {#6} }
0885     { } { } \iow_term:n
0886 }
0887 \__msg_class_new:nn { none } { }
0888 \__msg_class_new:nn { show }
0889 {
0890     \iow_wrap:nnnN
0891     { \use:c { \c__msg_text_prefix_tl #1 / #2 } {#3} {#4} {#5} {#6} }
0892     { } { } \__msg_show:n
0893 }
0894 \cs_new_protected:Npn \__msg_show:n #1
0895 {
0896     \tl_if_in:nnTF { ^^J #1 } { ^^J > ~ }
0897     {
0898         \tl_if_in:nnTF { #1 \s__msg_mark } { . \s__msg_mark }
0899         { \__msg_show_dot:w } { \__msg_show:w }
0900         ^^J #1 \s__msg_stop
0901     }
0902     { \__msg_show:nn { ? #1 } { } }
```



```
0903 }
0904 \cs_new:Npn \__msg_show_dot:w #1 ^^J > ~ #2 . \s__msg_stop
0905 { \__msg_show:nn {#1} {#2} }
0906 \cs_new:Npn \__msg_show:w #1 ^^J > ~ #2 \s__msg_stop
0907 { \__msg_show:nn {#1} {#2} }
0908 \cs_new_protected:Npn \__msg_show:nn #1#2
0909 {
0910   \tl_if_empty:nF {#1}
0911   { \exp_args:No \iow_term:n { \use_none:n #1 } }
0912   \tl_set:Nn \l__msg_internal_tl {#2}
0913   \__kernel_iow_with:Nnn \tex_newlinechar:D { 10 }
0914   {
0915     \__kernel_iow_with:Nnn \tex_errorcontextlines:D { -1 }
0916     {
0917       \tex_showtokens:D \exp_after:wN \exp_after:wN \exp_after:wN
0918       { \exp_after:wN \l__msg_internal_tl }
0919     }
0920   }
0921 }
0922 \group_end:
0923 \cs_new:Npe \msg_show_item:n #1
0924 { ^^J > ~ \c_space_tl \exp_not:N \tl_to_str:n { {#1} } }
0925 \cs_new:Npe \msg_show_item_unbraced:n #1
0926 { ^^J > ~ \c_space_tl \exp_not:N \tl_to_str:n {#1} }
0927 \cs_new:Npe \msg_show_item:nn #1#2
0928 {
0929   ^^J > \use:nn { ~ } { ~ }
0930   \exp_not:N \tl_to_str:n { {#1} }
0931   \use:nn { ~ } { ~ } => \use:nn { ~ } { ~ }
0932   \exp_not:N \tl_to_str:n { {#2} }
0933 }
0934 \cs_new:Npe \msg_show_item_unbraced:nn #1#2
0935 {
0936   ^^J > \use:nn { ~ } { ~ }
0937   \exp_not:N \tl_to_str:n {#1}
0938   \use:nn { ~ } { ~ } => \use:nn { ~ } { ~ }
0939   \exp_not:N \tl_to_str:n {#2}
0940 }
0941 \cs_new:Npn \__msg_class_chk_exist:nT #1
0942 {
0943   \cs_if_free:cTF { __msg_ #1 _code:nnnnnn }
0944   { \msg_error:nnn { msg } { class-unknown } {#1} }
0945 }
0946 \tl_new:N \l__msg_class_tl
0947 \tl_new:N \l__msg_current_class_tl
0948 \prop_new:N \l__msg_redirect_prop
```

```
0949 \seq_new:N \l__msg_hierarchy_seq
0950 \seq_new:N \l__msg_class_loop_seq
0951 \cs_new_protected:Npn \__msg_use:nnnnnnn #1#2#3#4#5#6#7
0952 {
0953   \cs_if_exist_use:N \conditionally@traceoff
0954   \msg_if_exist:nnTF {#2} {#3}
0955   {
0956     \__msg_class_chk_exist:nT {#1}
0957     {
0958       \tl_set:Nn \l__msg_current_class_tl {#1}
0959       \cs_set_protected:Npe \__msg_use_code:
0960       {
0961         \exp_not:n
0962         {
0963           \use:c { __msg_ \l__msg_class_tl _code:nnnnnn }
0964           {#2} {#3} {#4} {#5} {#6} {#7}
0965         }
0966       }
0967       \__msg_use_redirect_name:n { #2 / #3 }
0968     }
0969   }
0970   { \msg_error:nnnn { msg } { unknown } {#2} {#3} }
0971   \cs_if_exist_use:N \conditionally@tracelon
0972 }
0973 \cs_new_protected:Npn \__msg_use_code: { }
0974 \cs_new_protected:Npn \__msg_use_redirect_name:n #1
0975 {
0976   \prop_get:NnNTF \l__msg_redirect_prop { / #1 } \l__msg_class_tl
0977   { \__msg_use_code: }
0978   {
0979     \seq_clear:N \l__msg_hierarchy_seq
0980     \__msg_use_hierarchy:nwWN { }
0981     #1 \s__msg_mark \__msg_use_hierarchy:nwWN
0982     / \s__msg_mark \__msg_use_none_delimit_by_s_stop:w
0983     \s__msg_stop
0984     \__msg_use_redirect_module:n { }
0985   }
0986 }
0987 \cs_new_protected:Npn \__msg_use_hierarchy:nwWN #1#2 / #3 \s__msg_mark #4
0988 {
0989   \seq_put_left:Nn \l__msg_hierarchy_seq {#1}
0990   #4 { #1 / #2 } #3 \s__msg_mark #4
0991 }
0992 \cs_new_protected:Npn \__msg_use_redirect_module:n #1
0993 {
0994   \seq_map_inline:Nn \l__msg_hierarchy_seq
```

```
0095 {
0096     \prop_get:cnNTF { l__msg_redirect_ \l__msg_current_class_tl _prop }
0097     {##1} \l__msg_class_tl
0098     {
0099         \seq_map_break:n
0100         {
0101             \tl_if_eq:NNTF \l__msg_current_class_tl \l__msg_class_tl
0102             { \__msg_use_code: }
0103             {
0104                 \tl_set_eq:NN \l__msg_current_class_tl \l__msg_class_tl
0105                 \__msg_use_redirect_module:n {##1}
0106             }
0107         }
0108     }
0109     {
0110         \str_if_eq:nnT {##1} {#1}
0111         {
0112             \tl_set_eq:NN \l__msg_class_tl \l__msg_current_class_tl
0113             \seq_map_break:n { \__msg_use_code: }
0114         }
0115     }
0116 }
0117 }
0118 \cs_new_protected:Npn \msg_redirect_name:nnn #1#2#3
0119 {
0120     \tl_if_empty:nTF {#3}
0121     { \prop_remove:Nn \l__msg_redirect_prop { / #1 / #2 } }
0122     {
0123         \__msg_class_chk_exist:nT {#3}
0124         { \prop_put:Nnn \l__msg_redirect_prop { / #1 / #2 } {#3} }
0125     }
0126 }
0127 \cs_new_protected:Npn \msg_redirect_class:nn
0128 { \__msg_redirect:nnn { } }
0129 \cs_new_protected:Npn \msg_redirect_module:nnn #1
0130 { \__msg_redirect:nnn { / #1 } }
0131 \cs_new_protected:Npn \__msg_redirect:nnn #1#2#3
0132 {
0133     \__msg_class_chk_exist:nT {#2}
0134     {
0135         \tl_if_empty:nTF {#3}
0136         { \prop_remove:cn { l__msg_redirect_ #2 _prop } {#1} }
0137         {
0138             \__msg_class_chk_exist:nT {#3}
0139             {
0140                 \prop_put:cnn { l__msg_redirect_ #2 _prop } {#1} {#3}
```

```

1041         \tl_set:Nn \l__msg_current_class_tl {#2}
1042         \seq_clear:N \l__msg_class_loop_seq
1043         \__msg_redirect_loop_chk:nnn {#2} {#3} {#1}
1044     }
1045 }
1046 }
1047 }
1048 \cs_new_protected:Npn \__msg_redirect_loop_chk:nnn #1#2#3
1049 {
1050     \seq_put_right:Nn \l__msg_class_loop_seq {#1}
1051     \prop_get:cnNT { l__msg_redirect_ #1 _prop } {#3} \l__msg_class_tl
1052     {
1053         \str_if_eq:VnF \l__msg_class_tl {#1}
1054         {
1055             \tl_if_eq:NNTF \l__msg_class_tl \l__msg_current_class_tl
1056             {
1057                 \prop_put:cnn { l__msg_redirect_ #2 _prop } {#3} {#2}
1058                 \msg_warning:nneeee
1059                 { msg } { redirect-loop }
1060                 { \seq_item:Nn \l__msg_class_loop_seq { 1 } }
1061                 { \seq_item:Nn \l__msg_class_loop_seq { 2 } }
1062                 {#3}
1063                 {
1064                     \seq_map_function:NN \l__msg_class_loop_seq
1065                     \__msg_redirect_loop_list:n
1066                     { \seq_item:Nn \l__msg_class_loop_seq { 1 } }
1067                 }
1068             }
1069             { \__msg_redirect_loop_chk:onn \l__msg_class_tl {#2} {#3} }
1070         }
1071     }
1072 }
1073 \cs_generate_variant:Nn \__msg_redirect_loop_chk:nnn { o }
1074 \cs_new:Npn \__msg_redirect_loop_list:n #1 { {#1} ~ => ~ }
1075 \cs_new_protected:Npn \__kernel_msg_show_eval:Nn #1#2
1076 { \exp_args:Nf \__msg_show_eval:nnN { #1 {#2} } {#2} \tl_show:n }
1077 \cs_new_protected:Npn \__kernel_msg_log_eval:Nn #1#2
1078 { \exp_args:Nf \__msg_show_eval:nnN { #1 {#2} } {#2} \tl_log:n }
1079 \cs_new_protected:Npn \__msg_show_eval:nnN #1#2#3 { #3 { #2 = #1 } }
1080 \cs_new_protected:Npn \__kernel_msg_new:nnnn #1
1081 { \msg_new:nnnn { LaTeX / #1 } }
1082 \cs_new_protected:Npn \__kernel_msg_new:nnn #1
1083 { \msg_new:nnn { LaTeX / #1 } }
1084 \cs_new_protected:Npn \__kernel_msg_info:nnxx #1
1085 { \msg_info:nnee { LaTeX / #1 } }
1086 \cs_new_protected:Npn \__kernel_msg_warning:nnx #1

```

```

1087 { \msg_warning:nne { LaTeX / #1 } }
1088 \cs_new_protected:Npn \__kernel_msg_warning:nnxx #1
1089 { \msg_warning:nnee { LaTeX / #1 } }
1090 \cs_new_protected:Npn \__kernel_msg_error:nnx #1
1091 { \msg_error:nne { LaTeX / #1 } }
1092 \cs_new_protected:Npn \__kernel_msg_error:nnxx #1
1093 { \msg_error:nnee { LaTeX / #1 } }
1094 \cs_new_protected:Npn \__kernel_msg_error:nnxxx #1
1095 { \msg_error:nneee { LaTeX / #1 } }
1096 \cs_new:Npn \__kernel_msg_expandable_error:nnn #1
1097 { \msg_expandable_error:nnn { LaTeX / #1 } }
1098 \cs_new:Npn \__kernel_msg_expandable_error:nnf #1
1099 { \msg_expandable_error:nnf { LaTeX / #1 } }
1100 \cs_new:Npn \__kernel_msg_expandable_error:nnff #1
1101 { \msg_expandable_error:nnff { LaTeX / #1 } }
1102 \msg_new:nnnn { msg } { already-defined }
1103 { Message~'#2'~for~module~'#1'~already~defined. }
1104 {
1105   \c_msg_coding_error_text_tl
1106   LaTeX~was~asked~to~define~a~new~message~called~'#2'\\
1107   by~the~module~'#1':~this~message~already~exists.
1108   \c_msg_return_text_tl
1109 }
1110 \msg_new:nnnn { msg } { unknown }
1111 { Unknown~message~'#2'~for~module~'#1'. }
1112 {
1113   \c_msg_coding_error_text_tl
1114   LaTeX~was~asked~to~display~a~message~called~'#2'\\
1115   by~the~module~'#1':~this~message~does~not~exist.
1116   \c_msg_return_text_tl
1117 }
1118 \msg_new:nnnn { msg } { class-unknown }
1119 { Unknown~message~class~'#1'. }
1120 {
1121   LaTeX~has~been~asked~to~redirect~messages~to~a~class~'#1':\\
1122   this~was~never~defined.
1123   \c_msg_return_text_tl
1124 }
1125 \msg_new:nnnn { msg } { redirect-loop }
1126 {
1127   Message~redirection~loop~caused~by~ {#1} ~=>~ {#2}
1128   \tl_if_empty:nF {#3} { ~for~module~' \use_none:n #3 ' } .
1129 }
1130 {
1131   Adding~the~message~redirection~ {#1} ~=>~ {#2}
1132   \tl_if_empty:nF {#3} { ~for~the~module~' \use_none:n #3 ' } ~

```

```
1133     created~an~infinite~loop~\\
1134     \iow_indent:n { #4 \\ }
1135 }
1136 \msg_new:nnnn { kernel } { bad-number-of-arguments }
1137 { Function~'#1'~cannot~be~defined~with~#2~arguments. }
1138 {
1139     \c__msg_coding_error_text_tl
1140     LaTeX~has~been~asked~to~define~a~function~'#1'~with~
1141     #2~arguments.~
1142     TeX~allows~between~0~and~9~arguments~for~a~single~function.
1143 }
1144 \msg_new:nnnn { kernel } { command-already-defined }
1145 { Control~sequence~#1~already~defined. }
1146 {
1147     \c__msg_coding_error_text_tl
1148     LaTeX~has~been~asked~to~create~a~new~control~sequence~'#1'~
1149     but~this~name~has~already~been~used~elsewhere. \\ \\
1150     The~current~meaning~is:\\
1151     \\ #2
1152 }
1153 \msg_new:nnnn { kernel } { command-not-defined }
1154 { Control~sequence~#1~undefined. }
1155 {
1156     \c__msg_coding_error_text_tl
1157     LaTeX~has~been~asked~to~use~a~control~sequence~'#1':\\
1158     this~has~not~been~defined~yet.
1159 }
1160 \msg_new:nnnn { kernel } { empty-search-pattern }
1161 { Empty~search~pattern. }
1162 {
1163     \c__msg_coding_error_text_tl
1164     LaTeX~has~been~asked~to~replace~an~empty~pattern~by~'#1':~that~
1165     would~lead~to~an~infinite~loop!
1166 }
1167 \msg_new:nnnn { kernel } { non-base-function }
1168 { Function~'#1'~is~not~a~base~function }
1169 {
1170     \c__msg_coding_error_text_tl
1171     Functions~defined~through~\iow_char:N\\cs_new:Nn~must~have~
1172     a~signature~consisting~of~only~normal~arguments~'N'~and~'n'.~
1173     The~signature~'#2'~of~'#1'~contains~other~arguments~'#3'.~
1174     To~define~variants~use~\iow_char:N\\cs_generate_variant:Nn~
1175     and~to~define~other~functions~use~\iow_char:N\\cs_new:Npn.
1176 }
1177 \msg_new:nnnn { kernel } { missing-colon }
1178 { Function~'#1'~contains~no~':'. }
```

```
1179 {
1180   \c__msg_coding_error_text_tl
1181   Code-level~functions~must~contain~':'~to~separate~the~
1182   argument~specification~from~the~function~name.~This~is~
1183   needed~when~defining~conditionals~or~variants,~or~when~building~a~
1184   parameter~text~from~the~number~of~arguments~of~the~function.
1185 }
1186 \msg_new:nnnn { kernel } { overflow }
1187 { Integers~larger~than~2^{30}-1~cannot~be~stored~in~arrays. }
1188 {
1189   An~attempt~was~made~to~store~#3~
1190   \tl_if_empty:nF {#2} { at~position~#2~ } in~the~array~'#1'.~
1191   The~largest~allowed~value~#4~will~be~used~instead.
1192 }
1193 \msg_new:nnnn { kernel } { out-of-bounds }
1194 { Access~to~an~entry~beyond~an~array's~bounds. }
1195 {
1196   An~attempt~was~made~to~access~or~store~data~at~position~#2~of~the~
1197   array~'#1',~but~this~array~has~entries~at~positions~from~1~to~#3.
1198 }
1199 \msg_new:nnnn { kernel } { protected-predicate }
1200 { Predicate~'#1'~must~be~expandable. }
1201 {
1202   \c__msg_coding_error_text_tl
1203   LaTeX~has~been~asked~to~define~'#1'~as~a~protected~predicate.~
1204   Only~expandable~tests~can~have~a~predicate~version.
1205 }
1206 \msg_new:nnn { kernel } { randint-backward-range }
1207 { Wrong~order~of~bounds~in~\iow_char:N\\int_rand:nn{#1}{#2}. }
1208 \msg_new:nnnn { kernel } { conditional-base-undefined }
1209 { Undefined~conditional~base~function~'#1'. }
1210 {
1211   \c__msg_coding_error_text_tl
1212   LaTeX~has~been~asked~to~define~a~variant~of~the~conditional~'#1',~
1213   but~the~latter~is~not~defined.
1214 }
1215 \msg_new:nnnn { kernel } { conditional-form-unknown }
1216 { Conditional~form~'#1'~for~function~'#2'~unknown. }
1217 {
1218   \c__msg_coding_error_text_tl
1219   LaTeX~has~been~asked~to~define~the~conditional~form~'#1'~of~
1220   the~function~'#2',~but~only~'TF',~'T',~'F',~and~'p'~forms~exist.
1221 }
1222 \msg_new:nnnn { kernel } { variant-too-long }
1223 { Variant~form~'#1'~longer~than~base~signature~of~'#2'. }
1224 {
```



```
1225 \c__msg_coding_error_text_tl 11225
1226 LaTeX~has~been~asked~to~create~a~variant~of~the~function~'#2'~ 11226
1227 with~a~signature~starting~with~'#1',~but~that~is~longer~than~ 11227
1228 the~signature~(part~after~the~colon)~of~'#2'. 11228
1229 } 11229
1230 \msg_new:nnnn { kernel } { invalid-variant } 11230
1231 { Variant~form~'#1'~invalid~for~base~form~'#2'. } 11231
1232 { 11232
1233 \c__msg_coding_error_text_tl 11233
1234 LaTeX~has~been~asked~to~create~a~variant~of~the~function~'#2'~ 11234
1235 with~a~signature~starting~with~'#1',~but~cannot~change~an~argument~ 11235
1236 from~type~'#3'~to~type~'#4'. 11236
1237 } 11237
1238 \msg_new:nnnn { kernel } { invalid-exp-args } 11238
1239 { Invalid~variant~specifier~'#1'~in~'#2'. } 11239
1240 { 11240
1241 \c__msg_coding_error_text_tl 11241
1242 LaTeX~has~been~asked~to~create~an~\iow_char:N\\exp_args:N...~ 11242
1243 function~with~signature~'N#2'~but~'#1'~is~not~a~valid~argument~ 11243
1244 specifier. 11244
1245 } 11245
1246 \msg_new:nnn { kernel } { deprecated-variant } 11246
1247 { 11247
1248 Variant~form~'#1'~deprecated~for~base~form~'#2'.~ 11248
1249 One~should~not~change~an~argument~from~type~'#3'~to~type~'#4' 11249
1250 \str_case:nnF {#3} 11250
1251 { 11251
1252 { n } { :~use~a~'\token_if_eq_charcode:NNTF #4 c v V'~variant? } 11252
1253 { N } { :~base~form~only~accepts~a~single~token~argument. } 11253
1254 {#4} { :~base~form~is~already~a~variant. } 11254
1255 } { . } 11255
1256 } 11256
1257 \msg_new:nnn { char } { active } 11257
1258 { Cannot~generate~active~chars. } 11258
1259 \msg_new:nnn { char } { invalid-catcode } 11259
1260 { Invalid~catcode~for~char~generation. } 11260
1261 \msg_new:nnn { char } { null-space } 11261
1262 { Cannot~generate~null~char~as~a~space. } 11262
1263 \msg_new:nnn { char } { out-of-range } 11263
1264 { Charcode~requested~out~of~engine~range. } 11264
1265 \msg_new:nnn { dim } { zero-unit } 11265
1266 { Zero~unit~in~conversion. } 11266
1267 \msg_new:nnnn { kernel } { quote-in-shell } 11267
1268 { Quotes~in~shell~command~'#1'. } 11268
1269 { Shell~commands~cannot~contain~quotes~("). } 11269
1270 \msg_new:nnnn { keys } { no-property } 11270
```

```

1271 { No~property~given~in~definition~of~key~'#1'. }
1272 {
1273     \c__msg_coding_error_text_tl
1274     Inside~\keys_define:nn each~key~name~
1275     needs~a~property:  \ \ \
1276     \iow_indent:n { #1 .<property> } \ \ \
1277     LaTeX~did~not~find~a~'.'~to~indicate~the~start~of~a~property.
1278 }
1279 \msg_new:nnnn { keys } { property-boolean-values-only }
1280 { The~property~'#1'~accepts~boolean~values~only. }
1281 {
1282     \c__msg_coding_error_text_tl
1283     The~property~'#1'~only~accepts~the~values~'true'~and~'false'.
1284 }
1285 \msg_new:nnnn { keys } { property-requires-value }
1286 { The~property~'#1'~requires~a~value. }
1287 {
1288     \c__msg_coding_error_text_tl
1289     LaTeX~was~asked~to~set~property~'#1'~for~key~'#2'. \
1290     No~value~was~given~for~the~property,~and~one~is~required.
1291 }
1292 \msg_new:nnnn { keys } { property-unknown }
1293 { The~key~property~'#1'~is~unknown. }
1294 {
1295     \c__msg_coding_error_text_tl
1296     LaTeX~has~been~asked~to~set~the~property~'#1'~for~key~'#2':~
1297     this~property~is~not~defined.
1298 }
1299 \msg_new:nnnn { quark } { invalid-function }
1300 { Quark~test~function~'#1'~is~invalid. }
1301 {
1302     \c__msg_coding_error_text_tl
1303     LaTeX~has~been~asked~to~create~quark~test~function~'#1'~
1304     \tl_if_empty:nTF {#2}
1305     { but~that~name~ }
1306     { with~signature~'#2',~but~that~signature~ }
1307     is~not~valid.
1308 }
1309 \__kernel_msg_new:nnn { quark } { invalid }
1310 { Invalid~quark~variable~'#1'. }
1311 \msg_new:nnnn { scanmark } { already-defined }
1312 { Scan~mark~'#1'~already~defined. }
1313 {
1314     \c__msg_coding_error_text_tl
1315     LaTeX~has~been~asked~to~create~a~new~scan~mark~'#1'~
1316     but~this~name~has~already~been~used~for~a~scan~mark.

```

```

1317 }
1318 \msg_new:nnnn { seq } { item-too-large }
1319 { Sequence~'#1'~does~not~have~an~item~#3 }
1320 {
1321   An~attempt~was~made~to~push~or~pop~the~item~at~position~#3~
1322   of~'#1',~but~this~
1323   \int_compare:nTF { #3 = 0 }
1324     { position~does~not~exist. }
1325     { sequence~only~has~#2~item \int_compare:nF { #2 = 1 } {s}. }
1326 }
1327 \msg_new:nnnn { seq } { shuffle-too-large }
1328 { The~sequence~#1~is~too~long~to~be~shuffled~by~TeX. }
1329 {
1330   TeX~has~ \int_eval:n { \c_max_register_int + 1 } ~
1331   toks~registers:~this~only~allows~to~shuffle~up~to~
1332   \int_use:N \c_max_register_int \_items.~
1333   The~list~will~not~be~shuffled.
1334 }
1335 \msg_new:nnnn { kernel } { variable-not-defined }
1336 { Variable~#1~undefined. }
1337 {
1338   \c__msg_coding_error_text_tl
1339   LaTeX~has~been~asked~to~show~a~variable~#1,~but~this~has~not~
1340   been~defined~yet.
1341 }
1342 \msg_new:nnnn { kernel } { bad-type }
1343 { Variable~'#1'~is~not~a~valid~#3. }
1344 {
1345   \c__msg_coding_error_text_tl
1346   The~variable~'#1'~with~\tl_if_empty:nTF {#4} {meaning} {value}\\\\
1347   \iow_indent:n {#2}\\\\
1348   should~be~a~#3~variable,~but~
1349   \tl_if_empty:nTF {#4}
1350     { it~is~not \str_if_eq:nnF {#3} { bool } { ~a~short~macro } . }
1351     {
1352       it~does~not~have~the~correct~
1353       \str_if_eq:nnTF {#2} {#4}
1354         { category~codes. }
1355         { internal~structure:\\\\\iow_indent:n {#4} }
1356     }
1357 }
1358 \msg_new:nnnn { prop } { bad-link }
1359 { Variable~'#1'~is~not~a~valid~(linked)~prop. }
1360 {
1361   \c__msg_coding_error_text_tl
1362   The~variable~'#1'~has~an~incorrect~internal~structure.~

```

```
1363     Its~internal~entry~'#2'~points~to~'#3',~whose~name~is~not~of~the~
1364     form~'#4~<key>'.
1365 }
1366 \msg_new:nnnn { clist } { non-clist }
1367 { Variable~'#1'~is~not~a~valid~clist. }
1368 {
1369     \c__msg_coding_error_text_tl
1370     The~variable~'#1'~with~value\\\\
1371     \iow_indent:n {#2}\\\\
1372     should~be~a~clist~variable,~but~it~includes~empty~or~blank~items~
1373     without~braces.
1374 }
1375 \msg_new:nnnn { prop } { misused }
1376 { A~property~list~was~misused. }
1377 {
1378     \c__msg_coding_error_text_tl
1379     A~property~list~variable~was~used~without~an~accessor~function.~
1380     It~
1381     \tl_if_empty:nTF {#1}
1382     { is~empty. }
1383     { contains~the~key-value~pairs \use_none:n #1 . }
1384 }
1385 \msg_new:nnnn { prop } { inner-make }
1386 { '#1'~ cannot~ be~ used~ in~ a~ group. }
1387 {
1388     \c__msg_coding_error_text_tl
1389     The~ command~ '#1'~ was~ applied~ to~ the~ property~ list~
1390     variable~ '#2', but~ the~ storage~ type~ can~ only~ be~ changed~
1391     at~ the~ outermost~ group~ level.
1392 }
1393 \msg_new:nnn { kernel } { bad-exp-end-f }
1394 { Misused~\exp_end_continue_f:w or~:nw }
1395 \msg_new:nnn { kernel } { bad-variable }
1396 { Erroneous~variable~#1 used! }
1397 \msg_new:nnn { seq } { misused }
1398 { A~sequence~was~misused. }
1399 \msg_new:nnn { prg } { negative-replication }
1400 { Negative~argument~for~\iow_char:N\prg_replicate:nn. }
1401 \msg_new:nnn { prop } { prop-keyval }
1402 { Missing~'='~in~'#1'~(in~'..._keyval:Nn') }
1403 \msg_new:nnn { kernel } { unknown-comparison }
1404 { Relation~'#1'~not~among~=,<,>,==,!=,<=,>=. }
1405 \msg_new:nnn { kernel } { zero-step }
1406 { Zero~step~size~for~function~#1. }
1407 \msg_new:nnn { clist } { show }
1408 {
```

```

1409 The~comma~list~\tl_if_empty:nF {#1} { #1 ~ }
1410 \tl_if_empty:nTF {#2}
1411 { is~empty \>~ . }
1412 { contains~the~items~(without~outer~braces): #2 . }
1413 }
1414 \msg_new:nnn { intarray } { show }
1415 { The~integer~array~#1~contains~#2~items: \> #3 . }
1416 \msg_new:nnn { prop } { show }
1417 {
1418 The~ \str_if_eq:nnF {#3} { flat } { #3~ }
1419 property~list~#1~
1420 \tl_if_empty:nTF {#2}
1421 { is~empty \>~ . }
1422 { contains~the~pairs~(without~outer~braces): #2 . }
1423 }
1424 \msg_new:nnn { seq } { show }
1425 {
1426 The~sequence~#1~
1427 \tl_if_empty:nTF {#2}
1428 { is~empty \>~ . }
1429 { contains~the~items~(without~outer~braces): #2 . }
1430 }
1431 \msg_new:nnn { kernel } { show-streams }
1432 {
1433 \tl_if_empty:nTF {#2} { No~ } { The~following~ }
1434 \str_case:nn {#1}
1435 {
1436 { ior } { input ~ }
1437 { iow } { output ~ }
1438 }
1439 streams~are~
1440 \tl_if_empty:nTF {#2} { open } { in~use: #2 . }
1441 }
1442 \msg_new:nnnn { sys } { backend-set }
1443 { Backend~configuration~already~set. }
1444 {
1445 Run~time~backend~selection~may~only~be~carried~out~once~during~a~run.~
1446 This~second~attempt~to~set~them~will~be~ignored.
1447 }
1448 \msg_new:nnnn { sys } { load-debug-in-preamble }
1449 { Load~debug~support~in~the~preamble. }
1450 {
1451 Debugging~requires~support~loaded~in~the~preamble: \>
1452 Use~\sys_load_debug:~before~\begin{document}.
1453 }
1454 \msg_new:nnnn { sys } { wrong-backend }

```

```
1455 { Backend~request~inconsistent~with~engine::~using~'#2'~backend. } 11455
1456 { 11456
1457     You~have~requested~backend~'#1',~but~this~is~not~suitable~for~use~with~the~ 11457
1458     active~engine.~LaTeX~will~use~the~'#2'~backend~instead. 11458
1459 } 11459
1460 \cs_set_protected:Npn \__msg_tmp:w #1 11460
1461 { 11461
1462     \cs_new:Npn #1 ? { } 11462
1463     \cs_new:Npn \__msg_expandable_error:nn ##1##2 11463
1464     { 11464
1465         \exp_after:wN \exp_after:wN 11465
1466         \exp_after:wN \__msg_use_none_delimit_by_s_stop:w 11466
1467         \use:n { #1 ~ ! ~ ##2 : ~ ##1 } \s__msg_stop 11467
1468     } 11468
1469 } 11469
1470 \exp_args:Nc \__msg_tmp:w { ??? } 11470
1471 \exp_args_generate:n { oooo } 11471
1472 \cs_new:Npn \msg_expandable_error:nnnnnn #1#2#3#4#5#6 11472
1473 { 11473
1474     \exp_args:Nee \__msg_expandable_error:nn 11474
1475     { 11475
1476         \exp_args:Nc \exp_args:Noooo 11476
1477         { \c__msg_text_prefix_tl #1 / #2 } 11477
1478         { \tl_to_str:n {#3} } 11478
1479         { \tl_to_str:n {#4} } 11479
1480         { \tl_to_str:n {#5} } 11480
1481         { \tl_to_str:n {#6} } 11481
1482     } 11482
1483     { \msg_error_text:n {#1} } 11483
1484 } 11484
1485 \cs_new:Npn \msg_expandable_error:nnnnn #1#2#3#4#5 11485
1486 { \msg_expandable_error:nnnnnn {#1} {#2} {#3} {#4} {#5} { } } 11486
1487 \cs_new:Npn \msg_expandable_error:nnnn #1#2#3#4 11487
1488 { \msg_expandable_error:nnnnnn {#1} {#2} {#3} {#4} { } { } } 11488
1489 \cs_new:Npn \msg_expandable_error:nnn #1#2#3 11489
1490 { \msg_expandable_error:nnnnnn {#1} {#2} {#3} { } { } { } } 11490
1491 \cs_new:Npn \msg_expandable_error:nn #1#2 11491
1492 { \msg_expandable_error:nnnnnn {#1} {#2} { } { } { } { } } 11492
1493 \cs_generate_variant:Nn \msg_expandable_error:nnnnnn { nnffff } 11493
1494 \cs_generate_variant:Nn \msg_expandable_error:nnnnn { nnfff } 11494
1495 \cs_generate_variant:Nn \msg_expandable_error:nnnn { nnff } 11495
1496 \cs_generate_variant:Nn \msg_expandable_error:nnn { nnf } 11496
1497 \prop_gput:Nnn \g_msg_module_name_prop { kernel } { LaTeX } 11497
1498 \prop_gput:Nnn \g_msg_module_type_prop { kernel } { } 11498
1499 \clist_map_inline:nn 11499
1500 { 11500
```

```
1501 char , clist , coffin , debug , deprecation , dim, msg ,
1502 quark , prg , prop , scanmark , seq , sys
1503 }
1504 {
1505   \prop_gput:Nnn \g_msg_module_name_prop {#1} { LaTeX }
1506   \prop_gput:Nnn \g_msg_module_type_prop {#1} { }
1507 }
1508 \prop_gput:Nnn \g_msg_module_name_prop { LaTeX / cmd } { LaTeX }
1509 \prop_gput:Nnn \g_msg_module_type_prop { LaTeX / cmd } { }
1510 \prop_gput:Nnn \g_msg_module_name_prop { LaTeX / ltcmd } { LaTeX }
1511 \prop_gput:Nnn \g_msg_module_type_prop { LaTeX / ltcmd } { }
1512 %% File: l3file.dtx
1513 \tl_new:N \l__ior_internal_tl
1514 \int_const:Nn \c__ior_term_ior { 16 }
1515 \seq_new:N \g__ior_streams_seq
1516 \tl_new:N \l__ior_stream_tl
1517 \prop_new:N \g__ior_streams_prop
1518 \int_step_inline:nnn
1519 { 0 }
1520 {
1521   \cs_if_exist:NTF \contextversion
1522     { \tex_count:D 38 ~ }
1523     {
1524       \tex_count:D 16 ~ %
1525       \cs_if_exist:NT \loccount { - 1 }
1526     }
1527 }
1528 {
1529   \prop_gput:Nnn \g__ior_streams_prop {#1} { Reserved~by~format }
1530 }
1531 \cs_new_protected:Npn \ior_new:N #1 { \cs_new_eq:NN #1 \c__ior_term_ior }
1532 \cs_generate_variant:Nn \ior_new:N { c }
1533 \ior_new:N \g_tmpa_ior
1534 \ior_new:N \g_tmpb_ior
1535 \cs_new_protected:Npn \ior_open:Nn #1#2
1536 { \ior_open:NnF #1 {#2} { \__kernel_file_missing:n {#2} } }
1537 \cs_generate_variant:Nn \ior_open:Nn { c }
1538 \tl_new:N \l__ior_file_name_tl
1539 \prg_new_protected_conditional:Npnn \ior_open:Nn #1#2 { T , F , TF }
1540 {
1541   \file_get_full_name:nNTF {#2} \l__ior_file_name_tl
1542   {
1543     \__kernel_ior_open:No #1 \l__ior_file_name_tl
1544     \prg_return_true:
1545   }
1546   { \prg_return_false: }
```



```

1547 }
1548 \prg_generate_conditional_variant:Nnn \ior_open:Nn { c } { T , F , TF }
1549 \exp_args:NNf \cs_new_protected:Npn \__ior_new:N
1550 { \exp_args:NNc \exp_after:wN \exp_stop_f: { newread } }
1551 \cs_if_exist:NT \contextversion
1552 {
1553   \cs_new_eq:NN \__ior_new_aux:N \__ior_new:N
1554   \cs_gset_protected:Npn \__ior_new:N #1
1555   {
1556     \cs_undefine:N #1
1557     \__ior_new_aux:N #1
1558   }
1559 }
1560 \cs_new_protected:Npn \__kernel_ior_open:Nn #1#2
1561 {
1562   \ior_close:N #1
1563   \seq_gpop:NNTF \g__ior_streams_seq \l__ior_stream_tl
1564   { \__ior_open_stream:Nn #1 {#2} }
1565   {
1566     \__ior_new:N #1
1567     \__kernel_tl_set:Nx \l__ior_stream_tl { \int_eval:n {#1} }
1568     \__ior_open_stream:Nn #1 {#2}
1569   }
1570 }
1571 \cs_generate_variant:Nn \__kernel_ior_open:Nn { No }
1572 \cs_new_protected:Npe \__ior_open_stream:Nn #1#2
1573 {
1574   \tex_global:D \tex_chardef:D #1 = \exp_not:N \l__ior_stream_tl \scan_stop:
1575   \prop_gput:NVn \exp_not:N \g__ior_streams_prop #1 {#2}
1576   \tex_openin:D #1
1577   \sys_if_engine luatex:TF
1578   { {#2} }
1579   { \exp_not:N \__kernel_file_name_quote:n {#2} \scan_stop: }
1580 }
1581 \cs_new_protected:Npn \ior_shell_open:Nn #1#2
1582 {
1583   \sys_if_shell:TF
1584   { \__ior_shell_open:oN { \tl_to_str:n {#2} } #1 }
1585   { \msg_error:nn { kernel } { pipe-failed } }
1586 }
1587 \cs_new_protected:Npn \__ior_shell_open:nN #1#2
1588 {
1589   \tl_if_in:nnTF {#1} { " }
1590   {
1591     \msg_error:nne
1592     { kernel } { quote-in-shell } {#1}

```

```

1593     }
1594     { \_kernel_ior_open:Nn #2 { |#1 } }
1595 }
1596 \cs_generate_variant:Nn \_ior_shell_open:nN { o }
1597 \msg_new:nnnn { kernel } { pipe-failed }
1598 { Cannot~run~piped~system~commands. }
1599 {
1600     LaTeX~tried~to~call~a~system~process~but~this~was~not~possible.\
1601     Try~the~"--shell-escape"~(or~"--enable-pipes")~option.
1602 }
1603 \cs_new_protected:Npn \ior_close:N #1
1604 {
1605     \int_compare:nT { -1 < #1 < \c__ior_term_ior }
1606     {
1607         \tex_closein:D #1
1608         \prop_gremove:NV \g__ior_streams_prop #1
1609         \seq_if_in:NVF \g__ior_streams_seq #1
1610             { \seq_gpush:NV \g__ior_streams_seq #1 }
1611         \cs_gset_eq:NN #1 \c__ior_term_ior
1612     }
1613 }
1614 \cs_generate_variant:Nn \ior_close:N { c }
1615 \cs_new_protected:Npn \ior_show:N { \_ior_show:NN \tl_show:n }
1616 \cs_generate_variant:Nn \ior_show:N { c }
1617 \cs_new_protected:Npn \ior_log:N { \_ior_show:NN \tl_log:n }
1618 \cs_generate_variant:Nn \ior_log:N { c }
1619 \cs_new_protected:Npn \_ior_show:NN #1#2
1620 {
1621     \_kernel_chk_defined:NT #2
1622     {
1623         \prop_get:NVNTF \g__ior_streams_prop #2 \l__ior_internal_tl
1624         {
1625             \exp_args:Ne #1
1626                 { \token_to_str:N #2 ~ open: ~ \l__ior_internal_tl }
1627             }
1628         { \exp_args:Ne #1 { \token_to_str:N #2 ~ closed } }
1629     }
1630 }
1631 \cs_new_protected:Npn \ior_show_list: { \_ior_list:N \msg_show:nneeee }
1632 \cs_new_protected:Npn \ior_log_list: { \_ior_list:N \msg_log:nneeee }
1633 \cs_new_protected:Npn \_ior_list:N #1
1634 {
1635     #1 { kernel } { show-streams }
1636     { ior }
1637     {
1638         \prop_map_function:NN \g__ior_streams_prop

```

```
1639         \msg_show_item_unbraced:nn
1640     }
1641     { } { }
1642 }
1643 \cs_new_eq:NN \if_eof:w \tex_ifeof:D
1644 \prg_new_conditional:Npnn \ior_if_eof:N #1 { p , T , F , TF }
1645 {
1646     \if_int_compare:w -1 < #1
1647         \if_int_compare:w #1 < \c__ior_term_ior
1648             \if_eof:w #1
1649                 \prg_return_true:
1650             \else:
1651                 \prg_return_false:
1652             \fi:
1653         \else:
1654             \prg_return_true:
1655         \fi:
1656     \else:
1657         \prg_return_true:
1658     \fi:
1659 }
1660 \cs_new_protected:Npn \ior_get:NN #1#2
1661 { \ior_get:NMF #1 #2 { \tl_set:Nn #2 { \q_no_value } } }
1662 \cs_new_protected:Npn \__ior_get:NN #1#2
1663 { \tex_read:D #1 to #2 }
1664 \prg_new_protected_conditional:Npnn \ior_get:NN #1#2 { T , F , TF }
1665 {
1666     \ior_if_eof:NTF #1
1667     { \prg_return_false: }
1668     {
1669         \__ior_get:NN #1 #2
1670         \prg_return_true:
1671     }
1672 }
1673 \cs_new_protected:Npn \ior_str_get:NN #1#2
1674 { \ior_str_get:NMF #1 #2 { \tl_set:Nn #2 { \q_no_value } } }
1675 \cs_new_protected:Npn \__ior_str_get:NN #1#2
1676 {
1677     \exp_args:Nno \use:n
1678     {
1679         \int_set:Nn \tex_endlinechar:D { -1 }
1680         \tex_readline:D #1 to #2
1681         \int_set:Nn \tex_endlinechar:D
1682     } { \int_use:N \tex_endlinechar:D }
1683 }
1684 \prg_new_protected_conditional:Npnn \ior_str_get:NN #1#2 { T , F , TF }
```

```

1685 {
1686     \ior_if_eof:NTF #1
1687     { \prg_return_false: }
1688     {
1689         \__ior_str_get:NN #1 #2
1690         \prg_return_true:
1691     }
1692 }
1693 \int_const:Nn \c__ior_term_noprompt_ior { -1 }
1694 \cs_new_protected:Npn \ior_get_term:nN #1#2
1695 { \__ior_get_term:NnN \__ior_get:NN {#1} #2 }
1696 \cs_new_protected:Npn \ior_str_get_term:nN #1#2
1697 { \__ior_get_term:NnN \__ior_str_get:NN {#1} #2 }
1698 \cs_new_protected:Npn \__ior_get_term:NnN #1#2#3
1699 {
1700     \group_begin:
1701     \tex_escapechar:D = -1 \scan_stop:
1702     \tl_if_blank:nTF {#2}
1703     { \exp_args:NNc #1 \c__ior_term_noprompt_ior }
1704     { \exp_args:NNc #1 \c__ior_term_ior }
1705     {#2}
1706     \exp_args:NNNv \group_end:
1707     \tl_set:Nn #3 {#2}
1708 }
1709 \cs_new:Npn \ior_map_break:
1710 { \prg_map_break:Nn \ior_map_break: { } }
1711 \cs_new:Npn \ior_map_break:n
1712 { \prg_map_break:Nn \ior_map_break: }
1713 \cs_new_protected:Npn \ior_map_inline:Nn
1714 { \__ior_map_inline:NNn \__ior_get:NN }
1715 \cs_new_protected:Npn \ior_str_map_inline:Nn
1716 { \__ior_map_inline:NNn \__ior_str_get:NN }
1717 \cs_new_protected:Npn \__ior_map_inline:NNn
1718 {
1719     \int_gincr:N \g__kernel_prg_map_int
1720     \exp_args:Nc \__ior_map_inline:NNNn
1721     { __ior_map_ \int_use:N \g__kernel_prg_map_int :n }
1722 }
1723 \cs_new_protected:Npn \__ior_map_inline:NNNn #1#2#3#4
1724 {
1725     \cs_gset_protected:Npn #1 ##1 {#4}
1726     \ior_if_eof:NF #3 { \__ior_map_inline_loop:NNN #1#2#3 }
1727     \prg_break_point:Nn \ior_map_break:
1728     { \int_gdecr:N \g__kernel_prg_map_int }
1729 }
1730 \cs_new_protected:Npn \__ior_map_inline_loop:NNN #1#2#3

```

```

1731 {
1732     #2 #3 \l__ior_internal_tl
1733     \if_eof:w #3
1734         \exp_after:wN \ior_map_break:
1735     \fi:
1736     \exp_args:No #1 \l__ior_internal_tl
1737     \__ior_map_inline_loop:NNN #1#2#3
1738 }
1739 \cs_new_protected:Npn \ior_map_variable:NNn
1740 { \__ior_map_variable:NNNn \ior_get:NN }
1741 \cs_new_protected:Npn \ior_str_map_variable:NNn
1742 { \__ior_map_variable:NNNn \ior_str_get:NN }
1743 \cs_new_protected:Npn \__ior_map_variable:NNNn #1#2#3#4
1744 {
1745     \ior_if_eof:NF #2 { \__ior_map_variable_loop:NNNn #1#2#3 {#4} }
1746     \prg_break_point:Nn \ior_map_break: { }
1747 }
1748 \cs_new_protected:Npn \__ior_map_variable_loop:NNNn #1#2#3#4
1749 {
1750     #1 #2 #3
1751     \if_eof:w #2
1752         \exp_after:wN \ior_map_break:
1753     \fi:
1754     #4
1755     \__ior_map_variable_loop:NNNn #1#2#3 {#4}
1756 }
1757 \tl_new:N \l__iow_internal_tl
1758 \int_const:Nn \c_log_iow { -1 }
1759 \int_const:Nn \c_term_iow
1760 {
1761     \bool_lazy_and:nnTF
1762         { \sys_if_engine luatex_p: }
1763         { \int_compare_p:nNn \tex luatexversion:D > { 80 } }
1764         { 128 }
1765         { 16 }
1766 }
1767 \seq_new:N \g__iow_streams_seq
1768 \tl_new:N \l__iow_stream_tl
1769 \prop_new:N \g__iow_streams_prop
1770 \int_step_inline:nnn
1771 { 0 }
1772 {
1773     \cs_if_exist:NTF \contextversion
1774         { \tex_count:D 39 ~ }
1775         {
1776             \tex_count:D 17 ~

```

```
1777 \cs_if_exist:NT \loccount { - 1 }
1778 }
1779 }
1780 {
1781 \prop_gput:Nnn \g__iow_streams_prop {#1} { Reserved~by~format }
1782 }
1783 \scan_new:N \s__iow_mark
1784 \scan_new:N \s__iow_stop
1785 \cs_new:Npn \__iow_use_i_delimit_by_s_stop:nw #1 #2 \s__iow_stop {#1}
1786 \quark_new:N \q__iow_nil
1787 \cs_new_protected:Npn \iow_new:N #1 { \cs_new_eq:NN #1 \c_term_iow }
1788 \cs_generate_variant:Nn \iow_new:N { c }
1789 \iow_new:N \g_tmpa_iow
1790 \iow_new:N \g_tmpb_iow
1791 \exp_args:NNf \cs_new_protected:Npn \__iow_new:N
1792 { \exp_args:NNc \exp_after:wN \exp_stop_f: { newwrite } }
1793 \cs_if_exist:NT \contextversion
1794 {
1795 \cs_new_eq:NN \__iow_new_aux:N \__iow_new:N
1796 \cs_gset_protected:Npn \__iow_new:N #1
1797 {
1798 \cs_undefine:N #1
1799 \__iow_new_aux:N #1
1800 }
1801 }
1802 \tl_new:N \l__iow_file_name_tl
1803 \cs_new_protected:Npn \iow_open:Nn #1#2
1804 {
1805 \__kernel_tl_set:Nx \l__iow_file_name_tl
1806 { \__kernel_file_name_sanitiz:n {#2} }
1807 \__kernel_iow_open:No #1 \l__iow_file_name_tl
1808 }
1809 \cs_generate_variant:Nn \iow_open:Nn { NV , c , cV }
1810 \cs_new_protected:Npn \__kernel_iow_open:Nn #1#2
1811 {
1812 \iow_close:N #1
1813 \seq_gpop:NNTF \g__iow_streams_seq \l__iow_stream_tl
1814 { \__iow_open_stream:Nn #1 {#2} }
1815 {
1816 \__iow_new:N #1
1817 \__kernel_tl_set:Nx \l__iow_stream_tl { \int_eval:n {#1} }
1818 \__iow_open_stream:Nn #1 {#2}
1819 }
1820 }
1821 \cs_generate_variant:Nn \__kernel_iow_open:Nn { No }
1822 \cs_new_protected:Npn \__iow_open_stream:Nn #1#2
```

```
1823 {
1824     \tex_global:D \tex_chardef:D #1 = \l__iow_stream_tl \scan_stop:
1825     \prop_gput:NVn \g__iow_streams_prop #1 {#2}
1826     \tex_immediate:D \tex_openout:D
1827         #1 \__kernel_file_name_quote:n {#2} \scan_stop:
1828 }
1829 \cs_generate_variant:Nn \__iow_open_stream:Nn { NV }
1830 \cs_new_protected:Npn \iow_shell_open:Nn #1#2
1831 {
1832     \sys_if_shell:TF
1833         { \__iow_shell_open:oN { \tl_to_str:n {#2} } #1 }
1834         { \msg_error:nn { kernel } { pipe-failed } }
1835     }
1836 \cs_new_protected:Npn \__iow_shell_open:nN #1#2
1837 {
1838     \tl_if_in:nnTF {#1} { " }
1839     {
1840         \msg_error:nne
1841             { kernel } { quote-in-shell } {#1}
1842     }
1843     { \__kernel_iow_open:Nn #2 { |#1 } }
1844 }
1845 \cs_generate_variant:Nn \__iow_shell_open:nN { o }
1846 \cs_new_protected:Npn \iow_close:N #1
1847 {
1848     \int_compare:nT { \c_log_iow < #1 < \c_term_iow }
1849     {
1850         \tex_immediate:D \tex_closeout:D #1
1851         \prop_gremove:NV \g__iow_streams_prop #1
1852         \seq_if_in:NVF \g__iow_streams_seq #1
1853             { \seq_gpush:NV \g__iow_streams_seq #1 }
1854         \cs_gset_eq:NN #1 \c_term_iow
1855     }
1856 }
1857 \cs_generate_variant:Nn \iow_close:N { c }
1858 \cs_new_protected:Npn \iow_show:N { \__iow_show:NN \tl_show:n }
1859 \cs_generate_variant:Nn \iow_show:N { c }
1860 \cs_new_protected:Npn \iow_log:N { \__iow_show:NN \tl_log:n }
1861 \cs_generate_variant:Nn \iow_log:N { c }
1862 \cs_new_protected:Npn \__iow_show:NN #1#2
1863 {
1864     \__kernel_chk_defined:NT #2
1865     {
1866         \prop_get:NVNTF \g__iow_streams_prop #2 \l__iow_internal_tl
1867         {
1868             \exp_args:Ne #1
```



```

1869         { \token_to_str:N #2 ~ open: ~ \l__iow_internal_tl }
1870     }
1871     { \exp_args:Nc #1 { \token_to_str:N #2 ~ closed } }
1872 }
1873 }
1874 \cs_new_protected:Npn \iow_show_list: { \__iow_list:N \msg_show:nneeee }
1875 \cs_new_protected:Npn \iow_log_list: { \__iow_list:N \msg_log:nneeee }
1876 \cs_new_protected:Npn \__iow_list:N #1
1877 {
1878     #1 { kernel } { show-streams }
1879     { iow }
1880     {
1881         \prop_map_function:NN \g__iow_streams_prop
1882         \msg_show_item_unbraced:nn
1883     }
1884     { } { }
1885 }
1886 \cs_new_protected:Npn \iow_shipout_e:Nn #1#2
1887 { \tex_write:D #1 {#2} }
1888 \cs_generate_variant:Nn \iow_shipout_e:Nn { Ne , c, ce }
1889 \cs_new_protected:Npn \iow_shipout:Nn #1#2
1890 { \tex_write:D #1 { \exp_not:n {#2} } }
1891 \cs_generate_variant:Nn \iow_shipout:Nn { Ne , c, ce }
1892 \cs_generate_variant:Nn \iow_shipout:Nn { Nx , cx }
1893 \cs_new_protected:Npn \__kernel_iow_with:Nnn #1#2
1894 {
1895     \int_compare:nNnTF {#1} = {#2}
1896     { \use:n }
1897     { \__iow_with:oNnn { \int_use:N #1 } #1 {#2} }
1898 }
1899 \cs_new_protected:Npn \__iow_with:nNnn #1#2#3#4
1900 {
1901     \int_set:Nn #2 {#3}
1902     #4
1903     \int_set:Nn #2 {#1}
1904 }
1905 \cs_generate_variant:Nn \__iow_with:nNnn { o }
1906 \cs_new_protected:Npn \iow_now:Nn #1#2
1907 {
1908     \__kernel_iow_with:Nnn \tex_newlinechar:D { \^^J }
1909     { \tex_immediate:D \tex_write:D #1 { \exp_not:n {#2} } }
1910 }
1911 \cs_generate_variant:Nn \iow_now:Nn { NV , Ne , c , cV , ce }
1912 \cs_generate_variant:Nn \iow_now:Nn { Nx , cx }
1913 \cs_new_protected:Npn \iow_log:n { \iow_now:Nn \c_log_iow }
1914 \cs_set_protected:Npn \iow_log:e { \iow_now:Ne \c_log_iow }

```

```
1915 \cs_generate_variant:Nn \iow_log:n { x } 1191
1916 \cs_new_protected:Npn \iow_term:n { \iow_now:Nn \c_term_iow } 1191
1917 \cs_set_protected:Npn \iow_term:e { \iow_now:Ne \c_term_iow } 1191
1918 \cs_generate_variant:Nn \iow_term:n { x } 1191
1919 \cs_new:Npn \iow_newline: { ^^J } 1191
1920 \cs_new_eq:NN \iow_char:N \cs_to_str:N 1192
1921 \int_new:N \l_iow_line_count_int 1192
1922 \int_set:Nn \l_iow_line_count_int { 78 } 1192
1923 \tl_new:N \l__iow_newline_tl 1192
1924 \int_new:N \l__iow_line_target_int 1192
1925 \tl_new:N \l__iow_one_indent_tl 1192
1926 \int_new:N \l__iow_one_indent_int 1192
1927 \cs_new:Npn \__iow_unindent:w { } 1192
1928 \cs_new_protected:Npn \__iow_set_indent:n #1 1192
1929 { 1192
1930     \__kernel_tl_set:Nx \l__iow_one_indent_tl 1193
1931     { \exp_args:No \__kernel_str_to_other_fast:n { \tl_to_str:n {#1} } } 1193
1932     \int_set:Nn \l__iow_one_indent_int 1193
1933     { \str_count:N \l__iow_one_indent_tl } 1193
1934     \exp_last_unbraced:NNo 1193
1935     \cs_set:Npn \__iow_unindent:w \l__iow_one_indent_tl { } 1193
1936 } 1193
1937 \exp_args:Ne \__iow_set_indent:n { \prg_replicate:nn { 4 } { ~ } } 1193
1938 \tl_new:N \l__iow_indent_tl 1193
1939 \int_new:N \l__iow_indent_int 1193
1940 \tl_new:N \l__iow_line_tl 1194
1941 \tl_new:N \l__iow_line_part_tl 1194
1942 \bool_new:N \l__iow_line_break_bool 1194
1943 \tl_new:N \l__iow_wrap_tl 1194
1944 \group_begin: 1194
1945     \int_set:Nn \tex_escapechar:D { -1 } 1194
1946     \tl_const:Ne \c__iow_wrap_marker_tl 1194
1947     { \tl_to_str:n { \^^I \^^O \^^W \^^_ \^^W \^^R \^^A \^^P } } 1194
1948 \group_end: 1194
1949 \tl_map_inline:nn 1194
1950 { { end } { newline } { allow_break } { indent } { unindent } } 1195
1951 { 1195
1952     \tl_const:ce { c__iow_wrap_ #1 _marker_tl } 1195
1953     { 1195
1954         \c__iow_wrap_marker_tl 1195
1955         #1 1195
1956         \c_catcode_other_space_tl 1195
1957     } 1195
1958 } 1195
1959 \cs_new_protected:Npn \iow_wrap_allow_break: 1195
1960 { 1196
```

```
1961 \msg_error:nnnn { kernel } { iow-indent }
1962 { \iow_wrap:nnnN } { \iow_wrap_allow_break: }
1963 }
1964 \cs_new:Npe \__iow_wrap_allow_break: { \c__iow_wrap_allow_break_marker_tl }
1965 \cs_new:Npn \__iow_wrap_allow_break_error:
1966 {
1967 \msg_expandable_error:nnnn { kernel } { iow-indent }
1968 { \iow_wrap:nnnN } { \iow_wrap_allow_break: }
1969 }
1970 \cs_new_protected:Npn \iow_indent:n #1
1971 {
1972 \msg_error:nnnnn { kernel } { iow-indent }
1973 { \iow_wrap:nnnN } { \iow_indent:n } {#1}
1974 #1
1975 }
1976 \cs_new:Npe \__iow_indent:n #1
1977 {
1978 \c__iow_wrap_indent_marker_tl
1979 #1
1980 \c__iow_wrap_unindent_marker_tl
1981 }
1982 \cs_new:Npn \__iow_indent_error:n #1
1983 {
1984 \msg_expandable_error:nnnnn { kernel } { iow-indent }
1985 { \iow_wrap:nnnN } { \iow_indent:n } {#1}
1986 #1
1987 }
1988 \cs_new_protected:Npn \iow_wrap:nnnN #1#2#3#4
1989 {
1990 \group_begin:
1991 \cs_if_exist_use:N \conditionally@traceoff
1992 \int_set:Nn \tex_escapechar:D { -1 }
1993 \cs_set:Npe \{ { \token_to_str:N \{ }
1994 \cs_set:Npe \# { \token_to_str:N \# }
1995 \cs_set:Npe \} { \token_to_str:N \} }
1996 \cs_set:Npe \% { \token_to_str:N \% }
1997 \cs_set:Npe \~ { \token_to_str:N \~ }
1998 \int_set:Nn \tex_escapechar:D { 92 }
1999 \cs_set_eq:NN \ \ \iow_newline:
2000 \cs_set_eq:NN \_ \c_catcode_other_space_tl
2001 \cs_set_eq:NN \iow_wrap_allow_break: \__iow_wrap_allow_break:
2002 \cs_set_eq:NN \iow_indent:n \__iow_indent:n
2003 #3
2004 \cs_set_eq:NN \protect \token_to_str:N
2005 \__kernel_tl_set:Nx \l__iow_wrap_tl {#1}
2006 \cs_set_eq:NN \iow_wrap_allow_break: \__iow_wrap_allow_break_error:
```

```
2007 \cs_set_eq:NN \iow_indent:n \__iow_indent_error:n 1200
2008 \__kernel_tl_set:Nx \l__iow_newline_tl { \iow_newline: #2 } 1200
2009 \__kernel_tl_set:Nx \l__iow_newline_tl { \tl_to_str:N \l__iow_newline_tl } 1200
2010 \int_set:Nn \l__iow_line_target_int 1201
2011 { \l__iow_line_count_int - \str_count:N \l__iow_newline_tl + 1 } 1201
2012 \int_compare:nNnT { \l__iow_line_target_int } < 0 1201
2013 { 1201
2014 \tl_set:Nn \l__iow_newline_tl { \iow_newline: } 1201
2015 \int_set:Nn \l__iow_line_target_int 1201
2016 { \l__iow_line_count_int + 1 } 1201
2017 } 1201
2018 \__iow_wrap_do: 1201
2019 \exp_args:NNf \group_end: 1201
2020 #4 { \tl_to_str:N \l__iow_wrap_tl } 1202
2021 } 1202
2022 \cs_generate_variant:Nn \iow_wrap:nnnN { ne } 1202
2023 \cs_new_protected:Npn \__iow_wrap_do: 1202
2024 { 1202
2025 \__kernel_tl_set:Nx \l__iow_wrap_tl 1202
2026 { 1202
2027 \exp_args:No \__kernel_str_to_other_fast:n \l__iow_wrap_tl 1202
2028 \c__iow_wrap_end_marker_tl 1202
2029 } 1202
2030 \__kernel_tl_set:Nx \l__iow_wrap_tl 1203
2031 { 1203
2032 \exp_after:wN \__iow_wrap_fix_newline:w \l__iow_wrap_tl 1203
2033 ^^J \q__iow_nil ^^J \s__iow_stop 1203
2034 } 1203
2035 \exp_after:wN \__iow_wrap_start:w \l__iow_wrap_tl 1203
2036 } 1203
2037 \cs_new:Npn \__iow_wrap_fix_newline:w #1 ^^J #2 ^^J 1203
2038 { 1203
2039 #1 1203
2040 \if_meaning:w \q__iow_nil #2 1204
2041 \__iow_use_i_delimit_by_s_stop:nw 1204
2042 \fi: 1204
2043 \c__iow_wrap_newline_marker_tl 1204
2044 \__iow_wrap_fix_newline:w #2 ^^J 1204
2045 } 1204
2046 \cs_new_protected:Npn \__iow_wrap_start:w 1204
2047 { 1204
2048 \bool_set_false:N \l__iow_line_break_bool 1204
2049 \tl_clear:N \l__iow_line_tl 1204
2050 \tl_clear:N \l__iow_line_part_tl 1205
2051 \tl_set:Nn \l__iow_wrap_tl { ~ \use_none:n } 1205
2052 \int_zero:N \l__iow_indent_int 1205
```

```
2053 \tl_clear:N \l__iow_indent_tl
2054 \__iow_wrap_chunk:nw { \l__iow_line_count_int }
2055 }
2056 \cs_new_eq:NN \__iow_sep: \__kernel_int_sep:
2057 \cs_set_protected:Npn \__iow_tmp:w #1#2
2058 {
2059 \cs_new_protected:Npn \__iow_wrap_chunk:nw ##1##2 #2
2060 {
2061 \tl_if_empty:nTF {##2}
2062 {
2063 \tl_clear:N \l__iow_line_part_tl
2064 \__iow_wrap_next:nw {##1}
2065 }
2066 {
2067 \tl_if_empty:NTF \l__iow_line_tl
2068 {
2069 \__iow_wrap_line:nw
2070 { \l__iow_indent_tl }
2071 ##1 - \l__iow_indent_int \__iow_sep:
2072 }
2073 { \__iow_wrap_line:nw { } ##1 \__iow_sep: }
2074 ##2 #1
2075 \__iow_wrap_end_chunk:w 7 6 5 4 3 2 1 0 \s__iow_stop
2076 }
2077 }
2078 \cs_new_protected:Npn \__iow_wrap_next:nw ##1##2 #1
2079 { \use:c { __iow_wrap_##2:n } {##1} }
2080 }
2081 \exp_args:NVV \__iow_tmp:w \c_catcode_other_space_tl \c__iow_wrap_marker_tl
2082 \cs_new_protected:Npn \__iow_wrap_line:nw #1
2083 {
2084 \tex_edef:D \l__iow_line_part_tl { \if_false: } \fi:
2085 #1
2086 \exp_after:wN \__iow_wrap_line_loop:w
2087 \int_value:w \int_eval:w
2088 }
2089 \cs_new:Npn \__iow_wrap_line_loop:w #1 \__iow_sep: #2#3#4#5#6#7#8#9
2090 {
2091 \if_int_compare:w #1 < 8 \exp_stop_f:
2092 \__iow_wrap_line_aux:Nw #1
2093 \fi:
2094 #2 #3 #4 #5 #6 #7 #8 #9
2095 \exp_after:wN \__iow_wrap_line_loop:w
2096 \int_value:w \int_eval:w #1 - 8 \__iow_sep:
2097 }
2098 \cs_new:Npn \__iow_wrap_line_aux:Nw #1#2#3 \exp_after:wN #4 \__iow_sep:
```

```
2099 {
2100 #2
2101 \exp_after:wN \__iow_wrap_line_end:NnnnnnnnnN
2102 \exp_after:wN #1
2103 \exp:w \exp_end_continue_f:w
2104 \exp_after:wN \exp_after:wN
2105 \if_case:w #1 \exp_stop_f:
2106 \prg_do_nothing:
2107 \or: \use_none:n
2108 \or: \use_none:nn
2109 \or: \use_none:nnn
2110 \or: \use_none:nnnn
2111 \or: \use_none:nnnnn
2112 \or: \use_none:nnnnnn
2113 \or: \__iow_wrap_line_seven:nnnnnnnn
2114 \fi:
2115 { } { } { } { } { } { } { } { } #3
2116 }
2117 \cs_new:Npn \__iow_wrap_line_seven:nnnnnnnn #1#2#3#4#5#6#7 { \exp_stop_f: }
2118 \cs_new:Npn \__iow_wrap_line_end:NnnnnnnnnN #1#2#3#4#5#6#7#8#9
2119 {
2120 #2 #3 #4 #5 #6 #7 #8
2121 \use_none:nnnnnn \int_eval:w 8 - \__iow_sep: #9
2122 \token_if_eq_charcode:NNTF \c_space_token #9
2123 { \__iow_wrap_line_end:nw { } }
2124 { \if_false: { \fi: } \__iow_wrap_break:w #9 }
2125 }
2126 \cs_new:Npn \__iow_wrap_line_end:nw #1
2127 {
2128 \if_false: { \fi: }
2129 \__iow_wrap_store_do:n {#1}
2130 \__iow_wrap_next_line:w
2131 }
2132 \cs_new:Npn \__iow_wrap_end_chunk:w
2133 #1 \int_eval:w #2 - #3 \__iow_sep: #4#5 \s__iow_stop
2134 {
2135 \if_false: { \fi: }
2136 \exp_args:Nf \__iow_wrap_next:nw { \int_eval:n { #2 - #4 } }
2137 }
2138 \cs_set_protected:Npn \__iow_tmp:w #1
2139 {
2140 \cs_new:Npn \__iow_wrap_break:w
2141 {
2142 \tex_edef:D \l__iow_line_part_tl
2143 { \if_false: } \fi:
2144 \exp_after:wN \__iow_wrap_break_first:w
```

```
2145         \l__iow_line_part_tl
2146         #1
2147         { ? \__iow_wrap_break_end:w }
2148         \s__iow_mark
2149     }
2150 \cs_new:Npn \__iow_wrap_break_first:w ##1 #1 ##2
2151 {
2152     \use_none:nn ##2 \__iow_wrap_break_none:w
2153     \__iow_wrap_break_loop:w ##1 #1 ##2
2154 }
2155 \cs_new:Npn \__iow_wrap_break_none:w ##1##2 #1 ##3 \s__iow_mark ##4 #1
2156 {
2157     \tl_if_empty:NTF \l__iow_line_tl
2158     { ##2 ##4 \__iow_wrap_line_end:nw { } }
2159     { \__iow_wrap_line_end:nw { \__iow_wrap_trim:N } ##2 ##4 #1 }
2160 }
2161 \cs_new:Npn \__iow_wrap_break_loop:w ##1 #1 ##2 #1 ##3
2162 {
2163     \use_none:n ##3
2164     ##1 #1
2165     \__iow_wrap_break_loop:w ##2 #1 ##3
2166 }
2167 \cs_new:Npn \__iow_wrap_break_end:w ##1 #1 ##2 ##3 #1 ##4 \s__iow_mark
2168 { ##1 \__iow_wrap_line_end:nw { } ##3 }
2169 }
2170 \exp_args:NV \__iow_tmp:w \c_catcode_other_space_tl
2171 \cs_new_protected:Npn \__iow_wrap_next_line:w #1#2 \s__iow_stop
2172 {
2173     \tl_clear:N \l__iow_line_tl
2174     \token_if_eq_meaning:NNTF #1 \__iow_wrap_end_chunk:w
2175     {
2176         \tl_clear:N \l__iow_line_part_tl
2177         \bool_set_true:N \l__iow_line_break_bool
2178         \__iow_wrap_next:nw { \l__iow_line_target_int }
2179     }
2180     {
2181         \__iow_wrap_line:nw
2182         { \l__iow_indent_tl }
2183         \l__iow_line_target_int - \l__iow_indent_int \__iow_sep:
2184         #1 #2 \s__iow_stop
2185     }
2186 }
2187 \cs_new_protected:Npn \__iow_wrap_allow_break:n #1
2188 {
2189     \__kernel_tl_set:Nx \l__iow_line_tl
2190     { \l__iow_line_tl \__iow_wrap_trim:N \l__iow_line_part_tl }
```



```
2191 \bool_set_false:N \l__iow_line_break_bool
2192 \tl_if_empty:NTF \l__iow_line_part_tl
2193 { \__iow_wrap_chunk:nw {#1} }
2194 { \exp_args:Nf \__iow_wrap_chunk:nw { \int_eval:n { #1 + 1 } } }
2195 }
2196 \cs_new_protected:Npn \__iow_wrap_indent:n #1
2197 {
2198   \tl_put_right:Ne \l__iow_line_tl { \l__iow_line_part_tl }
2199   \bool_set_false:N \l__iow_line_break_bool
2200   \int_add:Nn \l__iow_indent_int { \l__iow_one_indent_int }
2201   \tl_put_right:No \l__iow_indent_tl { \l__iow_one_indent_tl }
2202   \__iow_wrap_chunk:nw {#1}
2203 }
2204 \cs_new_protected:Npn \__iow_wrap_unindent:n #1
2205 {
2206   \tl_put_right:Ne \l__iow_line_tl { \l__iow_line_part_tl }
2207   \bool_set_false:N \l__iow_line_break_bool
2208   \int_sub:Nn \l__iow_indent_int { \l__iow_one_indent_int }
2209   \__kernel_tl_set:Nx \l__iow_indent_tl
2210   { \exp_after:wN \__iow_unindent:w \l__iow_indent_tl }
2211   \__iow_wrap_chunk:nw {#1}
2212 }
2213 \cs_new_protected:Npn \__iow_wrap_newline:n #1
2214 {
2215   \bool_if:NF \l__iow_line_break_bool
2216   { \__iow_wrap_store_do:n { \__iow_wrap_trim:N } }
2217   \bool_set_false:N \l__iow_line_break_bool
2218   \__iow_wrap_chunk:nw { \l__iow_line_target_int }
2219 }
2220 \cs_new_protected:Npn \__iow_wrap_end:n #1
2221 {
2222   \bool_if:NF \l__iow_line_break_bool
2223   { \__iow_wrap_store_do:n { \__iow_wrap_trim:N } }
2224   \bool_set_false:N \l__iow_line_break_bool
2225 }
2226 \cs_new_protected:Npn \__iow_wrap_store_do:n #1
2227 {
2228   \__kernel_tl_set:Nx \l__iow_line_tl
2229   { \l__iow_line_tl \l__iow_line_part_tl }
2230   \__kernel_tl_set:Nx \l__iow_wrap_tl
2231   {
2232     \l__iow_wrap_tl
2233     \l__iow_newline_tl
2234     #1 \l__iow_line_tl
2235   }
2236   \tl_clear:N \l__iow_line_tl
```

```
2237 }
2238 \cs_set_protected:Npn \__iow_tmp:w #1
2239 {
2240   \cs_new:Npn \__iow_wrap_trim:N ##1
2241     { \exp_after:wN \__iow_wrap_trim:w ##1 \s__iow_mark #1 \s__iow_mark \s__iow_stop }
2242   \cs_new:Npn \__iow_wrap_trim:w ##1 #1 \s__iow_mark
2243     { \__iow_wrap_trim_aux:w ##1 \s__iow_mark }
2244   \cs_new:Npn \__iow_wrap_trim_aux:w ##1 \s__iow_mark ##2 \s__iow_stop {##1}
2245 }
2246 \exp_args:NV \__iow_tmp:w \c_catcode_other_space_tl
2247 \tl_new:N \l__file_internal_tl
2248 \str_new:N \g_file_curr_dir_str
2249 \str_new:N \g_file_curr_ext_str
2250 \str_new:N \g_file_curr_name_str
2251 \seq_new:N \g__file_stack_seq
2252 \group_begin:
2253   \cs_set_protected:Npn \__file_tmp:w #1#2#3
2254   {
2255     \tl_if_blank:nTF {#1}
2256     {
2257       \cs_set:Npn \__file_tmp:w ##1 " ##2 " ##3 \s__file_stop
2258       { { } {##2} { } }
2259       \seq_gput_right:Ne \g__file_stack_seq
2260       {
2261         \exp_after:wN \__file_tmp:w \tex_jobname:D
2262         " \tex_jobname:D " \s__file_stop
2263       }
2264     }
2265     {
2266       \seq_gput_right:Nn \g__file_stack_seq { { } {#1} {#2} }
2267       \__file_tmp:w
2268     }
2269   }
2270 \cs_if_exist:NT \@currnamestack
2271 {
2272   \tl_if_empty:NF \@currnamestack
2273   { \exp_after:wN \__file_tmp:w \@currnamestack }
2274 }
2275 \group_end:
2276 \seq_new:N \g__file_record_seq
2277 \tl_new:N \l__file_base_name_tl
2278 \tl_new:N \l__file_full_name_tl
2279 \str_new:N \l__file_dir_str
2280 \str_new:N \l__file_ext_str
2281 \str_new:N \l__file_name_str
2282 \seq_new:N \l_file_search_path_seq
```

```
2283 \seq_new:N \l__file_tmp_seq
2284 \scan_new:N \s__file_stop
2285 \quark_new:N \q__file_nil
2286 \__kernel_quark_new_conditional:Nn \__file_quark_if_nil:n { TF }
2287 \quark_new:N \q__file_recursion_tail
2288 \quark_new:N \q__file_recursion_stop
2289 \__kernel_quark_new_test:N \__file_if_recursion_tail_stop:N
2290 \__kernel_quark_new_test:N \__file_if_recursion_tail_stop_do:nn
2291 \cs_new:Npn \__kernel_file_name_sanitiz:n #1
2292 {
2293   \exp_args:Ne \__file_name_trim_spaces:n
2294   {
2295     \exp_args:Ne \__file_name_strip_quotes:n
2296     { \__file_name_expand:n {#1} }
2297   }
2298 }
2299 \cs_new:Npn \__file_name_expand:n #1
2300 {
2301   \exp_after:wN \__file_name_expand_cleanup:Nw
2302   \cs:w __file_name = #1 \cs_end:
2303   \__file_name_expand_end:
2304 }
2305 \cs_new:Npn \__file_name_expand_cleanup:Nw #1 #2 \__file_name_expand_end:
2306 {
2307   \tl_if_empty:nF {#2}
2308   { \__file_name_expand_error:Nw #2 \__file_name_expand_end: }
2309   \exp_after:wN \__file_name_expand_cleanup:w \token_to_str:N #1
2310 }
2311 \exp_last_unbraced:NNNN
2312 \cs_new:Npn \__file_name_expand_cleanup:w #1 \tl_to_str:n { __file_name = } { }
2313 \cs_new:Npn \__file_name_expand_end:
2314 {
2315   \msg_expandable_error:nn
2316   { kernel } { filename-missing-endcsname }
2317   \cs_end: \__file_name_expand_end:
2318 }
2319 \cs_new:Npn \__file_name_expand_error:Nw #1 #2 \__file_name_expand_end:
2320 { \__file_name_expand_error_aux:Nw #1 #2 \cs_end: \__file_name_expand_end: }
2321 \cs_new:Npn \__file_name_expand_error_aux:Nw #1 #2 \cs_end: #3
2322   \__file_name_expand_end:
2323 {
2324   \msg_expandable_error:nnff
2325   { kernel } { filename-chars-lost }
2326   { \token_to_str:N #1 } { \exp_stop_f: #2 }
2327 }
2328 \cs_new:Npn \__file_name_strip_quotes:n #1
```

```
2329 {
2330     \__file_name_strip_quotes:nw { 0 }
2331     #1 " \q__file_recursion_tail " \q__file_recursion_stop {#1}
2332 }
2333 \cs_new:Npn \__file_name_strip_quotes:nw #1#2 "
2334 {
2335     \if_meaning:w \q__file_recursion_tail #2
2336     \__file_name_strip_quotes_end:wnwn
2337     \fi:
2338     #2
2339     \__file_name_strip_quotes:nw { #1 + 1 }
2340 }
2341 \cs_new:Npn \__file_name_strip_quotes_end:wnwn \fi: #1
2342     \__file_name_strip_quotes:nw #2 \q__file_recursion_stop #3
2343 {
2344     \fi:
2345     \int_if_odd:nT {#2}
2346     {
2347         \msg_expandable_error:nnn
2348         { kernel } { unbalanced-quote-in-filename } {#3}
2349     }
2350 }
2351 \cs_new:Npn \__file_name_trim_spaces:n #1
2352 { \__file_name_trim_spaces:nw {#1} #1 . \q__file_nil . \s__file_stop }
2353 \cs_new:Npn \__file_name_trim_spaces:nw #1#2 . #3 . #4 \s__file_stop
2354 {
2355     \__file_quark_if_nil:nTF {#3}
2356     {
2357         \tl_trim_spaces_apply:nN { #1 \s__file_stop }
2358         \__file_name_trim_spaces_aux:n
2359     }
2360     { \tl_trim_spaces:n {#1} }
2361 }
2362 \cs_new:Npn \__file_name_trim_spaces_aux:n #1
2363 { \__file_name_trim_spaces_aux:w #1 }
2364 \cs_new:Npn \__file_name_trim_spaces_aux:w #1 \s__file_stop {#1}
2365 \cs_new:Npn \__kernel_file_name_quote:n #1
2366 { \__file_name_quote:nw {#1} #1 ~ \q__file_nil \s__file_stop }
2367 \cs_new:Npn \__file_name_quote:nw #1 #2 ~ #3 \s__file_stop
2368 {
2369     \__file_quark_if_nil:nTF {#3}
2370     { #1 }
2371     { "#1" }
2372 }
2373 \tl_const:Ne \c__file_marker_tl { : \token_to_str:N : }
2374 \cs_new_protected:Npn \file_get:nnN #1#2#3
```

```
2375 {
2376     \file_get:nnNF {#1} {#2} #3
2377     { \tl_set:Nn #3 { \q_no_value } }
2378 }
2379 \cs_generate_variant:Nn \file_get:nnN { V }
2380 \prg_new_protected_conditional:Npnn \file_get:nnN #1#2#3 { T , F , TF }
2381 {
2382     \file_get_full_name:nNTF {#1} \l__file_full_name_tl
2383     {
2384         \exp_args:NV \__file_get_aux:nnN
2385             \l__file_full_name_tl
2386             {#2} #3
2387         \prg_return_true:
2388     }
2389     { \prg_return_false: }
2390 }
2391 \prg_generate_conditional_variant:Nnn \file_get:nnN { V } { T , F , TF }
2392 \cs_new_protected:Npe \__file_get_aux:nnN #1#2#3
2393 {
2394     \exp_not:N \if_false: { \exp_not:N \fi:
2395     \group_begin:
2396         \int_set_eq:NN \tex_tracingnesting:D \c_zero_int
2397         \exp_not:N \exp_args:No \tex_everyeof:D
2398             { \exp_not:N \c__file_marker_tl }
2399         #2 \scan_stop:
2400         \exp_not:N \exp_after:wN \exp_not:N \__file_get_do:Nw
2401         \exp_not:N \exp_after:wN #3
2402         \exp_not:N \exp_after:wN \exp_not:N \prg_do_nothing:
2403         \exp_not:N \tex_input:D
2404         \sys_if_engine luatex:TF
2405             { {#1} }
2406             { \exp_not:N \__kernel_file_name_quote:n {#1} \scan_stop: }
2407     \exp_not:N \if_false: } \exp_not:N \fi:
2408 }
2409 \exp_args:Nno \use:nn
2410 { \cs_new_protected:Npn \__file_get_do:Nw #1#2 }
2411 { \c__file_marker_tl }
2412 {
2413     \group_end:
2414     \tl_set:No #1 {#2}
2415 }
2416 \cs_new_eq:NN \__file_size:n \tex_filesize:D
2417 \cs_new:Npn \file_full_name:n #1
2418 {
2419     \exp_args:Ne \__file_full_name:n
2420     { \__kernel_file_name_sanitiz:n {#1} }
```

```
2421 }
2422 \cs_generate_variant:Nn \file_full_name:n { V }
2423 \cs_new:Npn \__file_full_name:n #1
2424 {
2425     \tl_if_blank:nF {#1}
2426     { \exp_args:Nne \__file_full_name_auxii:nn {#1} { \__file_full_name_aux:n {#1} } }
2427 }
2428 \cs_new:Npn \__file_full_name_aux:n #1
2429 {
2430     \if_cs_exist:w __file_seen_ \tl_to_str:n {#1} : \cs_end:
2431     -1
2432     \else:
2433         \exp_args:Ne \__file_full_name_auxi:nn { \__file_size:n {#1} } {#1}
2434         \fi:
2435     }
2436 \cs_new:Npn \__file_full_name_auxi:nn #1#2
2437 {
2438     \if:w \scan_stop: #1 \scan_stop:
2439     \else:
2440         \exp_after:wN \use_none:n
2441         \cs:w __file_seen_ \tl_to_str:n {#2} : \cs_end:
2442         #1
2443         \fi:
2444     }
2445 \cs_new:Npn \__file_full_name_auxii:nn #1 #2
2446 {
2447     \tl_if_blank:nTF {#2}
2448     {
2449         \seq_map_tokens:Nn \l_file_search_path_seq
2450         { \__file_full_name_aux:Nnn \seq_map_break:n {#1} }
2451         \cs_if_exist:NT \input@path
2452         {
2453             \tl_map_tokens:Nn \input@path
2454             { \__file_full_name_aux:Nnn \tl_map_break:n {#1} }
2455         }
2456         \__file_name_end:
2457     }
2458     { \__file_ext_check:nn {#1} {#2} }
2459 }
2460 \cs_new:Npn \__file_full_name_aux:Nnn #1#2#3
2461 {
2462     \exp_args:Ne \__file_full_name_aux:nN
2463     { \__file_full_name_slash:n {#3} #2 }
2464     #1
2465 }
2466 \cs_new:Npn \__file_full_name_slash:n #1
```

```
2467 {
2468     \__file_full_name_slash:nw {#1} #1 \q_nil / \q_nil / \q_nil \q_stop
2469 }
2470 \cs_new:Npn \__file_full_name_slash:nw #1#2 / \q_nil / #3 \q_stop
2471 {
2472     \quark_if_nil:nTF {#3}
2473     { #1 / }
2474     { #2 / }
2475 }
2476 \cs_new:Npn \__file_full_name_aux:nN #1
2477 { \exp_args:Nne \__file_full_name_aux:nnN {#1} { \__file_full_name_aux:n {#1} } }
2478 \cs_new:Npn \__file_full_name_aux:nnN #1 #2 #3
2479 {
2480     \tl_if_blank:nF {#2}
2481     {
2482         #3
2483         {
2484             \__file_ext_check:nn {#1} {#2}
2485             \__file_name_cleanup:w
2486         }
2487     }
2488 }
2489 \cs_new:Npn \__file_name_cleanup:w #1 \__file_name_end: { }
2490 \cs_new:Npn \__file_name_end: { }
2491 \cs_new:Npn \__file_ext_check:nn #1 #2
2492 { \__file_ext_check:nnw {#2} { / } #1 / \q_file_nil / \s_file_stop }
2493 \cs_new:Npn \__file_ext_check:nnw #1 #2 #3 / #4 / #5 \s_file_stop
2494 {
2495     \__file_quark_if_nil:nTF {#4}
2496     {
2497         \exp_args:No \__file_ext_check:nnnw
2498         { \use_none:n #2 } {#1} {#3} #3 . \q_file_nil . \s_file_stop
2499     }
2500     { \__file_ext_check:nnw {#1} { #2 #3 / } #4 / #5 \s_file_stop }
2501 }
2502 \cs_new:Npe \__file_ext_check:nnnw #1#2#3#4 . #5 . #6 \s_file_stop
2503 {
2504     \exp_not:N \__file_quark_if_nil:nTF {#5}
2505     {
2506         \exp_not:N \__file_ext_check:nnn
2507         { #1 #3 \tl_to_str:n { .tex } } { #1 #3 } {#2}
2508     }
2509     { #1 #3 }
2510 }
2511 \cs_new:Npn \__file_ext_check:nnn #1
2512 { \exp_args:Nne \__file_ext_check:nnnn {#1} { \__file_full_name_aux:n {#1} } }
```



```
2513 \cs_new:Npn \__file_ext_check:nnnn #1#2#3#4
2514 {
2515     \tl_if_blank:nTF {#2}
2516     {#3}
2517     {
2518         \bool_lazy_or:nnTF
2519         { \int_compare_p:nNn {#4} = {#2} }
2520         { \int_compare_p:nNn {#2} = { -1 } }
2521         {#1}
2522         {#3}
2523     }
2524 }
2525 \cs_new_protected:Npn \file_forget:n #1
2526 { \cs_undefine:c { __file_seen_ \file_full_name:n {#1} : } }
2527 \cs_new_protected:Npn \file_get_full_name:nN #1#2
2528 {
2529     \file_get_full_name:nNF {#1} #2
2530     { \tl_set:Nn #2 { \q_no_value } }
2531 }
2532 \cs_generate_variant:Nn \file_get_full_name:nN { V }
2533 \prg_new_protected_conditional:Npnn \file_get_full_name:nN #1#2 { T , F , TF }
2534 {
2535     \__kernel_tl_set:Nx #2
2536     { \file_full_name:n {#1} }
2537     \tl_if_empty:NTF #2
2538     { \prg_return_false: }
2539     { \prg_return_true: }
2540 }
2541 \prg_generate_conditional_variant:Nnn \file_get_full_name:nN
2542 { V } { T , F , TF }
2543 \ior_new:N \g__file_internal_ior
2544 \cs_new:Npn \file_size:n #1
2545 { \__file_details:nn {#1} { size } }
2546 \cs_generate_variant:Nn \file_size:n { V }
2547 \cs_new:Npn \file_timestamp:n #1
2548 { \__file_details:nn {#1} { moddate } }
2549 \cs_generate_variant:Nn \file_timestamp:n { V }
2550 \cs_new:Npn \__file_details:nn #1#2
2551 {
2552     \exp_args:Ne \__file_details_aux:nn
2553     { \file_full_name:n {#1} } {#2}
2554 }
2555 \cs_new:Npn \__file_details_aux:nn #1#2
2556 {
2557     \tl_if_blank:nF {#1}
2558     { \use:c { tex_file #2 :D } {#1} }
```

2559	}	12559
2560	\cs_new:Npn \file_mdfive_hash:n #1	12560
2561	{ \exp_args:Ne __file_mdfive_hash:n { \file_full_name:n {#1} } }	12561
2562	\cs_generate_variant:Nn \file_mdfive_hash:n { V }	12562
2563	\cs_new:Npn __file_mdfive_hash:n #1	12563
2564	{ \tex_mdffivesum:D file {#1} }	12564
2565	\cs_new:Npn \file_hex_dump:nnn #1#2#3	12565
2566	{	12566
2567	\exp_args:Neee __file_hex_dump_auxi:nnn	12567
2568	{ \file_full_name:n {#1} }	12568
2569	{ \int_eval:n {#2} }	12569
2570	{ \int_eval:n {#3} }	12570
2571	}	12571
2572	\cs_generate_variant:Nn \file_hex_dump:nnn { V }	12572
2573	\cs_new:Npn __file_hex_dump_auxi:nnn #1#2#3	12573
2574	{	12574
2575	\bool_lazy_any:nF	12575
2576	{	12576
2577	{ \tl_if_blank_p:n {#1} }	12577
2578	{ \int_compare_p:nNn {#2} = 0 }	12578
2579	{ \int_compare_p:nNn {#3} = 0 }	12579
2580	}	12580
2581	{	12581
2582	\exp_args:Ne __file_hex_dump_auxii:nnnn	12582
2583	{ __file_details_aux:nn {#1} { size } }	12583
2584	{#1} {#2} {#3}	12584
2585	}	12585
2586	}	12586
2587	\cs_new:Npn __file_hex_dump_auxii:nnnn #1#2#3#4	12587
2588	{	12588
2589	\int_compare:nNnTF {#3} > 0	12589
2590	{ __file_hex_dump_auxiii:nnnn {#3} }	12590
2591	{	12591
2592	\exp_args:Ne __file_hex_dump_auxiii:nnnn	12592
2593	{ \int_eval:n { #1 + #3 } }	12593
2594	}	12594
2595	{#1} {#2} {#4}	12595
2596	}	12596
2597	\cs_new:Npn __file_hex_dump_auxiii:nnnn #1#2#3#4	12597
2598	{	12598
2599	\int_compare:nNnTF {#4} > 0	12599
2600	{ __file_hex_dump_auxiv:nnn {#4} }	12600
2601	{	12601
2602	\exp_args:Ne __file_hex_dump_auxiv:nnn	12602
2603	{ \int_eval:n { #2 + #4 } }	12603
2604	}	12604

```
2605     {#1} {#3}
2606   }
2607   \cs_new:Npn \__file_hex_dump_auxiv:nnn #1#2#3
2608   {
2609     \tex_dump:D
2610     offset ~ \int_eval:n { #2 - 1 } ~
2611     length ~ \int_eval:n { #1 - #2 + 1 }
2612     {#3}
2613   }
2614   \cs_new:Npn \file_hex_dump:n #1
2615   { \exp_args:Ne \__file_hex_dump:n { \file_full_name:n {#1} } }
2616   \cs_generate_variant:Nn \file_hex_dump:n { V }
2617   \sys_if_engine luatex:TF
2618   {
2619     \cs_new:Npn \__file_hex_dump:n #1
2620     {
2621       \tl_if_blank:nF {#1}
2622       { \tex_dump:D whole {#1} {#1} }
2623     }
2624   }
2625   {
2626     \cs_new:Npn \__file_hex_dump:n #1
2627     {
2628       \tl_if_blank:nF {#1}
2629       { \tex_dump:D length \tex_filesize:D {#1} {#1} }
2630     }
2631   }
2632   \cs_new_protected:Npn \file_get_hex_dump:nN #1#2
2633   { \file_get_hex_dump:nNF {#1} #2 { \tl_set:Nn #2 { \q_no_value } } }
2634   \cs_generate_variant:Nn \file_get_hex_dump:nN { V }
2635   \cs_new_protected:Npn \file_get_md5hash:nN #1#2
2636   { \file_get_md5hash:nNF {#1} #2 { \tl_set:Nn #2 { \q_no_value } } }
2637   \cs_generate_variant:Nn \file_get_md5hash:nN { V }
2638   \cs_new_protected:Npn \file_get_size:nN #1#2
2639   { \file_get_size:nNF {#1} #2 { \tl_set:Nn #2 { \q_no_value } } }
2640   \cs_generate_variant:Nn \file_get_size:nN { V }
2641   \cs_new_protected:Npn \file_get_timestamp:nN #1#2
2642   { \file_get_timestamp:nNF {#1} #2 { \tl_set:Nn #2 { \q_no_value } } }
2643   \cs_generate_variant:Nn \file_get_timestamp:nN { V }
2644   \prg_new_protected_conditional:Npnn \file_get_hex_dump:nN #1#2 { T , F , TF }
2645   { \__file_get_details:nnN {#1} { hex_dump } #2 }
2646   \prg_generate_conditional_variant:Nnn \file_get_hex_dump:nN
2647   { V } { T , F , TF }
2648   \prg_new_protected_conditional:Npnn \file_get_md5hash:nN #1#2 { T , F , TF }
2649   { \__file_get_details:nnN {#1} { md5hash } #2 }
2650   \prg_generate_conditional_variant:Nnn \file_get_md5hash:nN
```

2651	{ V } { T , F , TF }	12651
2652	\prg_new_protected_conditional:Npnn \file_get_size:nN #1#2 { T , F , TF }	12652
2653	{ __file_get_details:nnN {#1} { size } #2 }	12653
2654	\prg_generate_conditional_variant:Nnn \file_get_size:nN	12654
2655	{ V } { T , F , TF }	12655
2656	\prg_new_protected_conditional:Npnn \file_get_timestamp:nN #1#2 { T , F , TF }	12656
2657	{ __file_get_details:nnN {#1} { timestamp } #2 }	12657
2658	\prg_generate_conditional_variant:Nnn \file_get_timestamp:nN	12658
2659	{ V } { T , F , TF }	12659
2660	\cs_new_protected:Npn __file_get_details:nnN #1#2#3	12660
2661	{	12661
2662	__kernel_tl_set:Nx #3	12662
2663	{ \use:c { file_ #2 :n } {#1} }	12663
2664	\tl_if_empty:NTF #3	12664
2665	{ \prg_return_false: }	12665
2666	{ \prg_return_true: }	12666
2667	}	12667
2668	\cs_new_protected:Npn \file_get_hex_dump:nnnN #1#2#3#4	12668
2669	{	12669
2670	\file_get_hex_dump:nnnNF {#1} {#2} {#3} #4	12670
2671	{ \tl_set:Nn #4 { \q_no_value } }	12671
2672	}	12672
2673	\cs_generate_variant:Nn \file_get_hex_dump:nnnN { V }	12673
2674	\prg_new_protected_conditional:Npnn \file_get_hex_dump:nnnN #1#2#3#4	12674
2675	{ T , F , TF }	12675
2676	{	12676
2677	__kernel_tl_set:Nx #4	12677
2678	{ \file_hex_dump:nnn {#1} {#2} {#3} }	12678
2679	\tl_if_empty:NTF #4	12679
2680	{ \prg_return_false: }	12680
2681	{ \prg_return_true: }	12681
2682	}	12682
2683	\prg_generate_conditional_variant:Nnn \file_get_hex_dump:nnnN	12683
2684	{ V } { T , F , TF }	12684
2685	\cs_new_eq:NN __file_str_cmp:nn \tex_strcmp:D	12685
2686	\prg_new_conditional:Npnn \file_compare_timestamp:nNn #1#2#3	12686
2687	{ p , T , F , TF }	12687
2688	{	12688
2689	\exp_args:Nee __file_compare_timestamp:nnN	12689
2690	{ \file_full_name:n {#1} }	12690
2691	{ \file_full_name:n {#3} }	12691
2692	#2	12692
2693	}	12693
2694	\prg_generate_conditional_variant:Nnn \file_compare_timestamp:nNn	12694
2695	{ nNV , V , VNV } { p , T , F , TF }	12695
2696	\cs_new:Npn __file_compare_timestamp:nnN #1#2#3	12696

```

2697 {
2698     \tl_if_blank:nTF {#1}
2699     {
2700         \if_charcode:w #3 <
2701         \prg_return_true:
2702     \else:
2703         \prg_return_false:
2704     \fi:
2705 }
2706 {
2707     \tl_if_blank:nTF {#2}
2708     {
2709         \if_charcode:w #3 >
2710         \prg_return_true:
2711     \else:
2712         \prg_return_false:
2713     \fi:
2714 }
2715 {
2716     \if_int_compare:w
2717         \__file_str_cmp:nn
2718         { \__file_timestamp:n {#1} }
2719         { \__file_timestamp:n {#2} }
2720         #3 \c_zero_int
2721         \prg_return_true:
2722     \else:
2723         \prg_return_false:
2724     \fi:
2725 }
2726 }
2727 }
2728 \cs_new_eq:NN \__file_timestamp:n \tex_filemoddate:D
2729 \prg_new_conditional:Npnn \file_if_exist:n #1 { p , T , F , TF }
2730 {
2731     \tl_if_blank:eTF { \file_full_name:n {#1} }
2732     { \prg_return_false: }
2733     { \prg_return_true: }
2734 }
2735 \prg_generate_conditional_variant:Nnn \file_if_exist:n { V } { p , T , F , TF }
2736 \cs_new_protected:Npn \file_if_exist_input:n #1
2737 {
2738     \file_get_full_name:nNT {#1} \l__file_full_name_tl
2739     { \__file_input:V \l__file_full_name_tl }
2740 }
2741 \cs_generate_variant:Nn \file_if_exist_input:n { V }
2742 \cs_new_protected:Npn \file_if_exist_input:nF #1#2

```

```
2743 {
2744   \file_get_full_name:nNTF {#1} \l__file_full_name_tl
2745   { \__file_input:V \l__file_full_name_tl }
2746   {#2}
2747 }
2748 \cs_generate_variant:Nn \file_if_exist_input:nF { V }
2749 \cs_new_protected:Npn \file_input_stop: { \tex_endinput:D }
2750 \cs_new_protected:Npn \__kernel_file_missing:n #1
2751 {
2752   \msg_error:nne { kernel } { file-not-found }
2753   { \__kernel_file_name_sanitiz:n {#1} }
2754 }
2755 \cs_new_protected:Npn \file_input:n #1
2756 {
2757   \file_get_full_name:nNTF {#1} \l__file_full_name_tl
2758   { \__file_input:V \l__file_full_name_tl }
2759   { \__kernel_file_missing:n {#1} }
2760 }
2761 \cs_generate_variant:Nn \file_input:n { V }
2762 \cs_new_protected:Npe \__file_input:n #1
2763 {
2764   \exp_not:N \clist_if_exist:NTF \exp_not:N \@filelist
2765   { \exp_not:N \@addtofilelist {#1} }
2766   { \seq_gput_right:Nn \exp_not:N \g__file_record_seq {#1} }
2767   \exp_not:N \__file_input_push:n {#1}
2768   \exp_not:N \tex_input:D
2769   \sys_if_engine luatex:TF
2770   { {#1} }
2771   { \exp_not:N \__kernel_file_name_quote:n {#1} \scan_stop: }
2772   \exp_not:N \__file_input_pop:
2773 }
2774 \cs_generate_variant:Nn \__file_input:n { V }
2775 \cs_new_protected:Npn \__file_input_push:n #1
2776 {
2777   \seq_gpush:Ne \g__file_stack_seq
2778   {
2779     { \g_file_curr_dir_str }
2780     { \g_file_curr_name_str }
2781     { \g_file_curr_ext_str }
2782   }
2783   \file_parse_full_name:nNNN {#1}
2784   \l__file_dir_str \l__file_name_str \l__file_ext_str
2785   \str_gset_eq:NN \g_file_curr_dir_str \l__file_dir_str
2786   \str_gset_eq:NN \g_file_curr_name_str \l__file_name_str
2787   \str_gset_eq:NN \g_file_curr_ext_str \l__file_ext_str
2788 }
```

```
2789 \cs_new_eq:NN \__kernel_file_input_push:n \__file_input_push:n
2790 \cs_new_protected:Npn \__file_input_pop:
2791 {
2792   \seq_gpop:NN \g__file_stack_seq \l__file_internal_tl
2793   \exp_after:wN \__file_input_pop:nnn \l__file_internal_tl
2794 }
2795 \cs_new_eq:NN \__kernel_file_input_pop: \__file_input_pop:
2796 \cs_new_protected:Npn \__file_input_pop:nnn #1#2#3
2797 {
2798   \str_gset:Nn \g_file_curr_dir_str {#1}
2799   \str_gset:Nn \g_file_curr_name_str {#2}
2800   \str_gset:Nn \g_file_curr_ext_str {#3}
2801 }
2802 \cs_new:Npn \file_input_raw:n #1
2803 { \exp_args:Ne \__file_input_raw:nn { \file_full_name:n {#1} } {#1} }
2804 \cs_generate_variant:Nn \file_input_raw:n { V }
2805 \cs_new:Npe \__file_input_raw:nn #1#2
2806 {
2807   \exp_not:N \tl_if_blank:nTF {#1}
2808   {
2809     \exp_not:N \exp_args:Nnne \exp_not:N \msg_expandable_error:nnn
2810       { kernel } { file-not-found }
2811       { \exp_not:N \__kernel_file_name_sanitiz:n {#2} }
2812   }
2813   {
2814     \exp_not:N \tex_input:D
2815     \sys_if_engine luatex:TF
2816       { {#1} }
2817       { \exp_not:N \__kernel_file_name_quote:n {#1} \scan_stop: }
2818   }
2819 }
2820 \exp_args_generate:n { nne }
2821 \cs_new:Npn \file_parse_full_name:n #1
2822 {
2823   \file_parse_full_name_apply:nN {#1}
2824   \prg_do_nothing:
2825 }
2826 \cs_generate_variant:Nn \file_parse_full_name:n { V }
2827 \cs_new:Npn \file_parse_full_name_apply:nN #1
2828 {
2829   \exp_args:Ne \__file_parse_full_name_auxi:nN
2830     { \__kernel_file_name_sanitiz:n {#1} }
2831 }
2832 \cs_generate_variant:Nn \file_parse_full_name_apply:nN { V }
2833 \cs_new:Npn \__file_parse_full_name_auxi:nN #1
2834 {
```



```
2835     \__file_parse_full_name_area:nw { } #1
2836     / \s__file_stop
2837 }
2838 \cs_new:Npn \__file_parse_full_name_area:nw #1 #2 / #3 \s__file_stop
2839 {
2840     \tl_if_empty:nTF {#3}
2841     { \__file_parse_full_name_base:nw { } #2 . \s__file_stop {#1} }
2842     { \__file_parse_full_name_area:nw { #1 / #2 } #3 \s__file_stop }
2843 }
2844 \cs_new:Npn \__file_parse_full_name_base:nw #1 #2 . #3 \s__file_stop
2845 {
2846     \tl_if_empty:nTF {#3}
2847     {
2848         \tl_if_empty:nTF {#1}
2849         {
2850             \tl_if_empty:nTF {#2}
2851             { \__file_parse_full_name_tidy:nnnN { } { } { } }
2852             { \__file_parse_full_name_tidy:nnnN { .#2 } { } { } }
2853         }
2854         { \__file_parse_full_name_tidy:nnnN {#1} { .#2 } { } }
2855     }
2856     { \__file_parse_full_name_base:nw { #1 . #2 } #3 \s__file_stop }
2857 }
2858 \cs_new:Npn \__file_parse_full_name_tidy:nnnN #1 #2 #3 #4
2859 {
2860     \exp_args:Nee #4
2861     {
2862         \str_if_eq:nnF {#3} { / } { \use_none:n }
2863         #3 \prg_do_nothing:
2864     }
2865     { \use_none:n #1 \prg_do_nothing: }
2866     {#2}
2867 }
2868 \cs_new_protected:Npn \file_parse_full_name:nNNN #1 #2 #3 #4
2869 {
2870     \file_parse_full_name_apply:nN {#1}
2871     \__file_full_name_assign:nnnNNN #2 #3 #4
2872 }
2873 \cs_new_protected:Npn \__file_full_name_assign:nnnNNN #1 #2 #3 #4 #5 #6
2874 {
2875     \str_set:Nn #4 {#1}
2876     \str_set:Nn #5 {#2}
2877     \str_set:Nn #6 {#3}
2878 }
2879 \cs_generate_variant:Nn \file_parse_full_name:nNNN { V }
2880 \cs_new_protected:Npn \file_show_list: { \__file_list:N \msg_show:nneeee }
```

```
2881 \cs_new_protected:Npn \file_log_list: { \__file_list:N \msg_log:nneeee } 1288
2882 \cs_new_protected:Npn \__file_list:N #1 1288
2883 { 1288
2884   \seq_clear:N \l__file_tmp_seq 1288
2885   \clist_if_exist:NT \@filelist 1288
2886   { 1288
2887     \exp_args:NNe \seq_set_from_clist:Nn \l__file_tmp_seq 1288
2888     { \tl_to_str:N \@filelist } 1288
2889   } 1288
2890   \seq_concat:NNN \l__file_tmp_seq \l__file_tmp_seq \g__file_record_seq 1289
2891   \seq_remove_duplicates:N \l__file_tmp_seq 1289
2892   #1 { kernel } { file-list } 1289
2893   { \seq_map_function:NN \l__file_tmp_seq \__file_list_aux:n } 1289
2894   { } { } { } 1289
2895 } 1289
2896 \cs_new:Npn \__file_list_aux:n #1 { \iow_newline: #1 } 1289
2897 \cs_if_exist:NT \@filelist 1289
2898 { 1289
2899   \AtBeginDocument 1289
2900   { 1290
2901     \exp_args:NNe \seq_set_from_clist:Nn \l__file_tmp_seq 1290
2902     { \tl_to_str:N \@filelist } 1290
2903     \seq_gconcat:NNN 1290
2904     \g__file_record_seq 1290
2905     \g__file_record_seq 1290
2906     \l__file_tmp_seq 1290
2907   } 1290
2908 } 1290
2909 \cs_new_protected:Npn \GetIdInfo 1290
2910 { 1291
2911   \tl_clear_new:N \ExplFileDescription 1291
2912   \tl_clear_new:N \ExplFileDate 1291
2913   \tl_clear_new:N \ExplFileName 1291
2914   \tl_clear_new:N \ExplFileExtension 1291
2915   \tl_clear_new:N \ExplFileVersion 1291
2916   \group_begin: 1291
2917   \char_set_catcode_space:n { 32 } 1291
2918   \exp_after:wN 1291
2919   \group_end: 1291
2920   \__file_id_info_auxi:w 1292
2921 } 1292
2922 \cs_new_protected:Npn \__file_id_info_auxi:w $ #1 $ #2 1292
2923 { 1292
2924   \tl_set:Nn \ExplFileDescription {#2} 1292
2925   \str_if_eq:nnTF {#1} { Id } 1292
2926   { 1292
```

```
2927         \tl_set:Nn \ExplFileDate { 0000/00/00 }
2928         \tl_set:Nn \ExplFileName { [unknown] }
2929         \tl_set:Nn \ExplFileExtension { [unknown~extension] }
2930         \tl_set:Nn \ExplFileVersion {-1}
2931     }
2932     { \__file_id_info_auxii:w #1 ~ \s__file_stop }
2933 }
2934 \cs_new_protected:Npn \__file_id_info_auxii:w
2935     #1 ~ #2.#3 ~ #4 ~ #5 ~ #6 \s__file_stop
2936 {
2937     \tl_set:Nn \ExplFileName {#2}
2938     \tl_set:Nn \ExplFileExtension {#3}
2939     \tl_set:Nn \ExplFileVersion {#4}
2940     \str_if_eq:nnTF {#4} {-1}
2941     { \tl_set:Nn \ExplFileDate { 0000/00/00 } }
2942     { \__file_id_info_auxiii:w #5 - 0 - 0 - \s__file_stop }
2943 }
2944 \cs_new_protected:Npn \__file_id_info_auxiii:w #1 - #2 - #3 - #4 \s__file_stop
2945     { \tl_set:Nn \ExplFileDate { #1/#2/#3 } }
2946 \cs_new_protected:Npn \__kernel_dependency_version_check:Nn #1
2947     { \exp_args:NV \__kernel_dependency_version_check:nn #1 }
2948 \cs_new_protected:Npn \__kernel_dependency_version_check:nn #1
2949 {
2950     \cs_if_exist:NTF \c__kernel_expl_date_tl
2951     {
2952         \exp_args:NV \__file_kernel_dependency_compare:nnn
2953             \c__kernel_expl_date_tl {#1}
2954     }
2955     { \__file_kernel_dependency_compare:nnn { 0000-00-00 } {#1} }
2956 }
2957 \cs_new_protected:Npn \__file_kernel_dependency_compare:nnn #1 #2 #3
2958 {
2959     \int_compare:nNnT
2960         { \__file_parse_version:w #1 \s__file_stop } <
2961         { \__file_parse_version:w #2 \s__file_stop }
2962     { \__file_mismatched_dependency_error:nn {#2} {#3} }
2963 }
2964 \cs_new:Npn \__file_parse_version:w #1 - #2 - #3 \s__file_stop {#1#2#3}
2965 \cs_new_protected:Npn \__file_mismatched_dependency_error:nn #1 #2
2966 {
2967     \exp_args:NNe \ior_shell_open:Nn \g__file_internal_ior
2968     {
2969         kpsewhich ~ --all ~
2970         --engine = \c_sys_engine_exec_str
2971         \c_space_tl \c_sys_engine_format_str
2972         \bool_lazy_and:nnT
```

```
2973         { \tl_if_exist_p:N \development@branch@name }
2974         { ! \tl_if_empty_p:N \development@branch@name }
2975         { -dev } .fmt
2976     }
2977 \seq_clear:N \l__file_tmp_seq
2978 \ior_map_inline:Nn \g__file_internal_ior
2979     { \seq_put_right:Nn \l__file_tmp_seq {##1} }
2980 \ior_close:N \g__file_internal_ior
2981 \msg_error:nnnn { kernel } { mismatched-support-file }
2982     {#1} {#2}
2983 \tex_endinput:D
2984 }
2985 \msg_new:nnnn { kernel } { mismatched-support-file }
2986 {
2987     Mismatched~LaTeX~support~files~detected. \\
2988     Loading~'#2'~aborted!
2989     \tl_if_exist:NT \c__kernel_expl_date_tl
2990     {
2991         \\ \\
2992         The~L3~programming~layer~in~the~LaTeX~format \\
2993         is~dated~\c__kernel_expl_date_tl,~but~in~your~TeX~
2994         tree~the~files~require \\ at~least~#1.
2995     }
2996 }
2997 {
2998     \int_compare:nNnTF { \seq_count:N \l__file_tmp_seq } > 1
2999     {
3000         The~cause~seems~to~be~an~old~format~file~in~the~user~tree. \\
3001         LaTeX~found~these~files:
3002         \seq_map_tokens:Nn \l__file_tmp_seq { \\~~~\use:n } \\
3003         Try~deleting~the~file~in~the~user~tree~then~run~LaTeX~again.
3004     }
3005     {
3006         The~most~likely~causes~are:
3007         \\~~~A~recent~format~generation~failed;
3008         \\~~~A~stray~format~file~in~the~user~tree~which~needs~
3009             to~be~removed~or~rebuilt;
3010         \\~~~You~are~running~a~manually~installed~version~of~#2 \\
3011         \underline{~~~} which~is~incompatible~with~the~version~in~LaTeX. \\
3012     }
3013     \\
3014     LaTeX~will~abort~loading~the~incompatible~support~files~
3015     but~this~may~lead~to \\ later~errors.~Please~ensure~that~
3016     your~LaTeX~format~is~correctly~regenerated.
3017 }
3018 \msg_new:nnnn { kernel } { file-not-found }
```

```

3019 { File~'#1'~not~found. }
3020 {
3021     The~requested~file~could~not~be~found~in~the~current~directory,~
3022     in~the~TeX~search~path~or~in~the~LaTeX~search~path.
3023 }
3024 \msg_new:nnn { kernel } { file-list }
3025 {
3026     >~File~List~<
3027     #1 \\\
3028     .....
3029 }
3030 \msg_new:nnnn { kernel } { filename-chars-lost }
3031 { #1~invalid~in~file~name.~Lost:~#2. }
3032 {
3033     There~was~an~invalid~token~in~the~file~name~that~caused~
3034     the~characters~following~it~to~be~lost.
3035 }
3036 \msg_new:nnnn { kernel } { filename-missing-endcsname }
3037 { Missing~\iow_char:N\\endcsname~inserted~in~filename. }
3038 {
3039     The~file~name~had~more~\iow_char:N\\csname~commands~than~
3040     \iow_char:N\\endcsname~ones.~LaTeX~will~add~the~missing~
3041     \iow_char:N\\endcsname~and~try~to~continue~as~best~as~it~can.
3042 }
3043 \msg_new:nnnn { kernel } { unbalanced-quote-in-filename }
3044 { Unbalanced~quotes~in~file~name~'#1'. }
3045 {
3046     File~names~must~contain~balanced~numbers~of~quotes~(").
3047 }
3048 \msg_new:nnnn { kernel } { iow-indent }
3049 { Only~#1~allows~#2 }
3050 {
3051     The~command~#2~can~only~be~used~in~messages~
3052     which~will~be~wrapped~using~#1.
3053     \tl_if_empty:nF {#3} { ~ It~was~called~with~argument~'#3'. }
3054 }
3055 \sys_if_engine luatex:TF
3056 {
3057     \str_const:Ne \c_sys_platform_str
3058     { \tex_directlua:D { tex.print(os.type) } }
3059 }
3060 {
3061     \file_if_exist:nTF { nul: }
3062     {
3063         \file_if_exist:nF { /dev/null }
3064         { \str_const:Nn \c_sys_platform_str { windows } }

```

```
3065     }
3066     {
3067         \file_if_exist:nT { /dev/null }
3068         { \str_const:Nn \c_sys_platform_str { unix } }
3069     }
3070 }
3071 \cs_if_exist:NF \c_sys_platform_str
3072 { \str_const:Nn \c_sys_platform_str { unknown } }
3073 \clist_map_inline:nn { unix , windows }
3074 {
3075     \__sys_const:nn { sys_if_platform_ #1 }
3076     { \str_if_eq_p:Vn \c_sys_platform_str { #1 } }
3077 }
3078 %% File: l3skip.dtx
3079 \cs_new_eq:NN \if_dim:w \tex_ifdim:D
3080 \cs_new_eq:NN \__dim_eval:w \tex_dimexpr:D
3081 \cs_new_eq:NN \__dim_eval_end: \tex_relax:D
3082 \scan_new:N \s__dim_mark
3083 \scan_new:N \s__dim_stop
3084 \cs_new:Npn \__dim_use_none_delimit_by_s_stop:w #1 \s__dim_stop { }
3085 \cs_new_protected:Npn \dim_new:N #1
3086 {
3087     \__kernel_chk_if_free_cs:N #1
3088     \cs:w newdimen \cs_end: #1
3089 }
3090 \cs_generate_variant:Nn \dim_new:N { c }
3091 \cs_new_protected:Npn \dim_const:Nn #1#2
3092 {
3093     \dim_new:N #1
3094     \tex_global:D #1 = \dim_eval:n {#2} \scan_stop:
3095 }
3096 \cs_generate_variant:Nn \dim_const:Nn { c }
3097 \cs_new_protected:Npn \dim_zero:N #1 { #1 = \c_zero_skip }
3098 \cs_new_protected:Npn \dim_gzero:N #1
3099 { \tex_global:D #1 = \c_zero_skip }
3100 \cs_generate_variant:Nn \dim_zero:N { c }
3101 \cs_generate_variant:Nn \dim_gzero:N { c }
3102 \cs_new_protected:Npn \dim_zero_new:N #1
3103 { \dim_if_exist:NTF #1 { \dim_zero:N #1 } { \dim_new:N #1 } }
3104 \cs_new_protected:Npn \dim_gzero_new:N #1
3105 { \dim_if_exist:NTF #1 { \dim_gzero:N #1 } { \dim_new:N #1 } }
3106 \cs_generate_variant:Nn \dim_zero_new:N { c }
3107 \cs_generate_variant:Nn \dim_gzero_new:N { c }
3108 \prg_new_eq_conditional:NNn \dim_if_exist:N \cs_if_exist:N
3109 { TF , T , F , p }
3110 \prg_new_eq_conditional:NNn \dim_if_exist:c \cs_if_exist:c
```

```

3111 { TF , T , F , p }
3112 \cs_new_protected:Npn \dim_set:Nn #1#2
3113 { #1 = \__dim_eval:w #2 \__dim_eval_end: \scan_stop: }
3114 \cs_new_protected:Npn \dim_gset:Nn #1#2
3115 { \tex_global:D #1 = \__dim_eval:w #2 \__dim_eval_end: \scan_stop: }
3116 \cs_generate_variant:Nn \dim_set:Nn { NV , c , cV }
3117 \cs_generate_variant:Nn \dim_gset:Nn { NV , c , cV }
3118 \cs_new_protected:Npn \dim_set_eq:NN #1#2
3119 { #1 = #2 \scan_stop: }
3120 \cs_generate_variant:Nn \dim_set_eq:NN { c , Nc , cc }
3121 \cs_new_protected:Npn \dim_gset_eq:NN #1#2
3122 { \tex_global:D #1 = #2 \scan_stop: }
3123 \cs_generate_variant:Nn \dim_gset_eq:NN { c , Nc , cc }
3124 \cs_new_protected:Npn \dim_add:Nn #1#2
3125 { \tex_advance:D #1 \__dim_eval:w #2 \__dim_eval_end: \scan_stop: }
3126 \cs_new_protected:Npn \dim_gadd:Nn #1#2
3127 {
3128     \tex_global:D \tex_advance:D #1
3129     \__dim_eval:w #2 \__dim_eval_end: \scan_stop:
3130 }
3131 \cs_generate_variant:Nn \dim_add:Nn { c }
3132 \cs_generate_variant:Nn \dim_gadd:Nn { c }
3133 \cs_new_protected:Npn \dim_sub:Nn #1#2
3134 { \tex_advance:D #1 - \__dim_eval:w #2 \__dim_eval_end: \scan_stop: }
3135 \cs_new_protected:Npn \dim_gsub:Nn #1#2
3136 {
3137     \tex_global:D \tex_advance:D #1
3138     -\__dim_eval:w #2 \__dim_eval_end: \scan_stop:
3139 }
3140 \cs_generate_variant:Nn \dim_sub:Nn { c }
3141 \cs_generate_variant:Nn \dim_gsub:Nn { c }
3142 \cs_new_eq:NN \__dim_sep: \__kernel_int_sep:
3143 \cs_new:Npn \dim_abs:n #1
3144 {
3145     \exp_after:wN \__dim_abs:N
3146     \dim_use:N \__dim_eval:w #1 \__dim_eval_end:
3147 }
3148 \cs_new:Npn \__dim_abs:N #1
3149 { \if_meaning:w - #1 \else: \exp_after:wN #1 \fi: }
3150 \cs_new:Npn \dim_max:nn #1#2
3151 {
3152     \dim_use:N \__dim_eval:w \exp_after:wN \__dim_maxmin:wwN
3153     \dim_use:N \__dim_eval:w #1 \exp_after:wN \__dim_sep:
3154     \dim_use:N \__dim_eval:w #2 \__dim_sep:
3155     >
3156     \__dim_eval_end:

```


3157	}	1315
3158	\cs_new:Npn \dim_min:nn #1#2	1315
3159	{	1315
3160	\dim_use:N __dim_eval:w \exp_after:wN __dim_maxmin:wwN	1316
3161	\dim_use:N __dim_eval:w #1 \exp_after:wN __dim_sep:	1316
3162	\dim_use:N __dim_eval:w #2 __dim_sep:	1316
3163	<	1316
3164	__dim_eval_end:	1316
3165	}	1316
3166	\cs_new:Npn __dim_maxmin:wwN #1 __dim_sep: #2 __dim_sep: #3	1316
3167	{	1316
3168	\if_dim:w #1 #3 #2 ~	1316
3169	#1	1316
3170	\else:	1317
3171	#2	1317
3172	\fi:	1317
3173	}	1317
3174	\cs_new:Npn \dim_ratio:nn #1#2	1317
3175	{ __dim_ratio:n {#1} / __dim_ratio:n {#2} }	1317
3176	\cs_new:Npn __dim_ratio:n #1	1317
3177	{ \int_value:w __dim_eval:w (#1) __dim_eval_end: }	1317
3178	\prg_new_conditional:Npnn \dim_compare:nNn #1#2#3 { p , T , F , TF }	1317
3179	{	1317
3180	\if_dim:w __dim_eval:w #1 #2 __dim_eval:w #3 __dim_eval_end:	1318
3181	\prg_return_true: \else: \prg_return_false: \fi:	1318
3182	}	1318
3183	\prg_new_conditional:Npnn \dim_compare:n #1 { p , T , F , TF }	1318
3184	{	1318
3185	\exp_after:wN __dim_compare:w	1318
3186	\dim_use:N __dim_eval:w #1 __dim_compare_error:	1318
3187	}	1318
3188	\cs_new:Npn __dim_compare:w #1 __dim_compare_error:	1318
3189	{	1318
3190	\exp_after:wN \if_false: \exp:w \exp_end_continue_f:w	1319
3191	__dim_compare:wNN #1 ? { = __dim_compare_end:w \else: } \s__dim_stop	1319
3192	}	1319
3193	\exp_args:Nno \use:nn	1319
3194	{ \cs_new:Npn __dim_compare:wNN #1 } { \tl_to_str:n {pt} #2#3 }	1319
3195	{	1319
3196	\if_meaning:w = #3	1319
3197	\use:c { __dim_compare_#2:w }	1319
3198	\fi:	1319
3199	#1 pt \exp_stop_f:	1319
3200	\prg_return_false:	1320
3201	\exp_after:wN __dim_use_none_delimit_by_s_stop:w	1320
3202	\fi:	1320

3203	\reverse_if:N \if_dim:w #1 pt #2	13203
3204	\exp_after:wN __dim_compare:wNN	13204
3205	\dim_use:N __dim_eval:w #3	13205
3206	}	13206
3207	\cs_new:cpn { __dim_compare_ ! :w }	13207
3208	#1 \reverse_if:N #2 ! #3 = { #1 #2 = #3 }	13208
3209	\cs_new:cpn { __dim_compare_ = :w }	13209
3210	#1 __dim_eval:w = { #1 __dim_eval:w }	13210
3211	\cs_new:cpn { __dim_compare_ < :w }	13211
3212	#1 \reverse_if:N #2 < #3 = { #1 #2 > #3 }	13212
3213	\cs_new:cpn { __dim_compare_ > :w }	13213
3214	#1 \reverse_if:N #2 > #3 = { #1 #2 < #3 }	13214
3215	\cs_new:Npn __dim_compare_end:w #1 \prg_return_false: #2 \s__dim_stop	13215
3216	{ #1 \prg_return_false: \else: \prg_return_true: \fi: }	13216
3217	\cs_new_protected:Npn __dim_compare_error:	13217
3218	{	13218
3219	\if_int_compare:w \c_zero_int \c_zero_int \fi:	13219
3220	=	13220
3221	__dim_compare_error:	13221
3222	}	13222
3223	\cs_new:Npn \dim_case:nnTF #1	13223
3224	{	13224
3225	\exp:w	13225
3226	\exp_args:Nf __dim_case:nnTF { \dim_eval:n {#1} }	13226
3227	}	13227
3228	\cs_new:Npn \dim_case:nnT #1#2#3	13228
3229	{	13229
3230	\exp:w	13230
3231	\exp_args:Nf __dim_case:nnTF { \dim_eval:n {#1} } {#2} {#3} { }	13231
3232	}	13232
3233	\cs_new:Npn \dim_case:nnF #1#2	13233
3234	{	13234
3235	\exp:w	13235
3236	\exp_args:Nf __dim_case:nnTF { \dim_eval:n {#1} } {#2} { }	13236
3237	}	13237
3238	\cs_new:Npn \dim_case:nn #1#2	13238
3239	{	13239
3240	\exp:w	13240
3241	\exp_args:Nf __dim_case:nnTF { \dim_eval:n {#1} } {#2} { } { }	13241
3242	}	13242
3243	\cs_new:Npn __dim_case:nnTF #1#2#3#4	13243
3244	{ __dim_case:nw {#1} #2 {#1} { } \s__dim_mark {#3} \s__dim_mark {#4} \s__dim_stop }	13244
3245	\cs_new:Npn __dim_case:nw #1#2#3	13245
3246	{	13246
3247	\dim_compare:nNnTF {#1} = {#2}	13247
3248	{ __dim_case_end:nw {#3} }	13248

```
3249 { \__dim_case:nw {#1} }
3250 }
3251 \cs_new:Npn \__dim_case_end:nw #1#2#3 \s__dim_mark #4#5 \s__dim_stop
3252 { \exp_end: #1 #4 }
3253 \cs_new:Npn \dim_while_do:nn #1#2
3254 {
3255   \dim_compare:nT {#1}
3256   {
3257     #2
3258     \dim_while_do:nn {#1} {#2}
3259   }
3260 }
3261 \cs_new:Npn \dim_until_do:nn #1#2
3262 {
3263   \dim_compare:nF {#1}
3264   {
3265     #2
3266     \dim_until_do:nn {#1} {#2}
3267   }
3268 }
3269 \cs_new:Npn \dim_do_while:nn #1#2
3270 {
3271   #2
3272   \dim_compare:nT {#1}
3273   { \dim_do_while:nn {#1} {#2} }
3274 }
3275 \cs_new:Npn \dim_do_until:nn #1#2
3276 {
3277   #2
3278   \dim_compare:nF {#1}
3279   { \dim_do_until:nn {#1} {#2} }
3280 }
3281 \cs_new:Npn \dim_while_do:nNnn #1#2#3#4
3282 {
3283   \dim_compare:nNnT {#1} #2 {#3}
3284   {
3285     #4
3286     \dim_while_do:nNnn {#1} #2 {#3} {#4}
3287   }
3288 }
3289 \cs_new:Npn \dim_until_do:nNnn #1#2#3#4
3290 {
3291   \dim_compare:nNnF {#1} #2 {#3}
3292   {
3293     #4
3294     \dim_until_do:nNnn {#1} #2 {#3} {#4}
```

```

3295 }
3296 }
3297 \cs_new:Npn \dim_do_while:nNnn #1#2#3#4
3298 {
3299     #4
3300     \dim_compare:nNnT {#1} #2 {#3}
3301     { \dim_do_while:nNnn {#1} #2 {#3} {#4} }
3302 }
3303 \cs_new:Npn \dim_do_until:nNnn #1#2#3#4
3304 {
3305     #4
3306     \dim_compare:nNnF {#1} #2 {#3}
3307     { \dim_do_until:nNnn {#1} #2 {#3} {#4} }
3308 }
3309 \cs_new:Npn \dim_step_function:nnnN #1#2#3
3310 {
3311     \exp_after:wN \__dim_step:wwwN
3312     \tex_the:D \__dim_eval:w #1 \exp_after:wN \__dim_sep:
3313     \tex_the:D \__dim_eval:w #2 \exp_after:wN \__dim_sep:
3314     \tex_the:D \__dim_eval:w #3 \__dim_sep:
3315 }
3316 \cs_new:Npn \__dim_step:wwwN #1 \__dim_sep: #2 \__dim_sep: #3 \__dim_sep: #4
3317 {
3318     \dim_compare:nNnTF {#2} > \c_zero_dim
3319     { \__dim_step:NnnnN > }
3320     {
3321         \dim_compare:nNnTF {#2} = \c_zero_dim
3322         {
3323             \msg_expandable_error:nnn { kernel } { zero-step } {#4}
3324             \use_none:nnnn
3325         }
3326         { \__dim_step:NnnnN < }
3327     }
3328     {#1} {#2} {#3} #4
3329 }
3330 \cs_new:Npn \__dim_step:NnnnN #1#2#3#4#5
3331 {
3332     \dim_compare:nNnF {#2} #1 {#4}
3333     {
3334         #5 {#2}
3335         \exp_args:NNf \__dim_step:NnnnN
3336         #1 { \dim_eval:n { #2 + #3 } } {#3} {#4} #5
3337     }
3338 }
3339 \cs_new_protected:Npn \dim_step_inline:nnnn
3340 {

```

```

3341 \int_gincr:N \g__kernel_prg_map_int
3342 \exp_args:NNc \__dim_step:NNnnnn
3343 \cs_gset_protected:Npn
3344 { \__dim_map_ \int_use:N \g__kernel_prg_map_int :w }
3345 }
3346 \cs_new_protected:Npn \dim_step_variable:nnnNn #1#2#3#4#5
3347 {
3348 \int_gincr:N \g__kernel_prg_map_int
3349 \exp_args:NNc \__dim_step:NNnnnn
3350 \cs_gset_protected:Npe
3351 { \__dim_map_ \int_use:N \g__kernel_prg_map_int :w }
3352 {#1}{#2}{#3}
3353 {
3354 \tl_set:Nn \exp_not:N #4 {##1}
3355 \exp_not:n {#5}
3356 }
3357 }
3358 \cs_new_protected:Npn \__dim_step:NNnnnn #1#2#3#4#5#6
3359 {
3360 #1 #2 ##1 {#6}
3361 \dim_step_function:nnnN {#3} {#4} {#5} #2
3362 \prg_break_point:Nn \scan_stop: { \int_gdecr:N \g__kernel_prg_map_int }
3363 }
3364 \cs_new:Npn \dim_eval:n #1
3365 { \dim_use:N \__dim_eval:w #1 \__dim_eval_end: }
3366 \cs_new:Npn \dim_sign:n #1
3367 {
3368 \int_value:w \exp_after:wN \__dim_sign:Nw
3369 \dim_use:N \__dim_eval:w #1 \__dim_eval_end: \__dim_sep:
3370 \exp_stop_f:
3371 }
3372 \cs_new:Npn \__dim_sign:Nw #1#2 \__dim_sep:
3373 {
3374 \if_dim:w #1#2 > \c_zero_dim
3375 1
3376 \else:
3377 \if_meaning:w - #1
3378 -1
3379 \else:
3380 0
3381 \fi:
3382 \fi:
3383 }
3384 \cs_new_eq:NN \dim_use:N \tex_the:D
3385 \cs_new:Npn \dim_use:c #1 { \tex_the:D \cs:w #1 \cs_end: }
3386 \cs_new:Npn \dim_to_decimal:n #1

```

```

3387 {
3388     \exp_after:wN
3389     \__dim_to_decimal:w \dim_use:N \__dim_eval:w #1 \__dim_eval_end:
3390 }
3391 \use:e
3392 {
3393     \cs_new:Npn \exp_not:N \__dim_to_decimal:w
3394     #1 . #2 \tl_to_str:n { pt }
3395 }
3396 {
3397     \int_compare:nNnTF {#2} > \c_zero_int
3398     { #1 . #2 }
3399     { #1 }
3400 }
3401 \cs_new:Npn \dim_to_decimal_in_sp:n #1
3402 { \int_value:w \__dim_eval:w #1 \__dim_eval_end: }
3403 \group_begin:
3404     \cs_set_protected:Npn \__dim_tmp:w #1#2
3405     {
3406         \cs_new:cpn { dim_to_decimal_in_ #1 :n } ##1
3407         {
3408             \exp_after:wN \__dim_to_decimal_aux:w
3409             \int_value:w \__dim_eval:w ##1 \__dim_eval_end: \__dim_sep: #2 \__dim_sep:
3410         }
3411     }
3412     \__dim_tmp:w { in } { 50 / 7227 } % delta = 7227/100
3413     \__dim_tmp:w { pc } { 1 / 24 } % delta = 12/1
3414     \__dim_tmp:w { cm } { 127 / 7227 } % delta = 7227/254
3415     \__dim_tmp:w { mm } { 1270 / 7227 } % delta = 7227/2540
3416     \__dim_tmp:w { bp } { 400 / 803 } % delta = 7227/7200
3417     \__dim_tmp:w { dd } { 1157 / 2476 } % delta = 1238/1157
3418     \__dim_tmp:w { cc } { 1157 / 29712 } % delta = 14856/1157
3419 \group_end:
3420 \cs_new:Npn \__dim_to_decimal_aux:w #1#2 \__dim_sep: #3 \__dim_sep:
3421 {
3422     \dim_to_decimal:n
3423     {
3424         \int_eval:n
3425         { ( 2 * #1#2 \if:w #1 - - \else: + \fi: 1 ) * #3 }
3426         sp
3427     }
3428 }
3429 \cs_new:Npn \dim_to_decimal_in_unit:nn #1#2
3430 {
3431     \exp_after:wN \__dim_chk_unit:w
3432     \int_value:w \__dim_eval:w #2 \__dim_eval_end: \__dim_sep: {#1}

```

```
3433 }
3434 \cs_new:Npn \__dim_chk_unit:w #1#2 \__dim_sep: #3
3435 {
3436   \token_if_eq_charcode:NNTF #1 0
3437   { \msg_expandable_error:nn { dim } { zero-unit } }
3438   {
3439     \exp_after:wN \__dim_branch_unit:w
3440     \int_value:w \if:w #1 - - \fi: \__dim_eval:w #3 \exp_after:wN \__dim_sep:
3441     \int_value:w \if:w #1 - - \fi: #1#2 \__dim_sep:
3442   }
3443 }
3444 \cs_new:Npn \__dim_branch_unit:w #1 \__dim_sep: #2 \__dim_sep:
3445 {
3446   \int_compare:nNnTF {#2} > { 65536 }
3447   { \__dim_to_decimal_aux:w #1 \__dim_sep: 32768 / #2 \__dim_sep: }
3448   {
3449     \int_compare:nNnTF {#2} = { 65536 }
3450     { \dim_to_decimal:n { #1sp } }
3451     { \__dim_get_quotient:w #1 \__dim_sep: #2 \__dim_sep: }
3452   }
3453 }
3454 \cs_new:Npn \__dim_get_quotient:w #1#2 \__dim_sep: #3 \__dim_sep:
3455 {
3456   \token_if_eq_charcode:NNTF #1 0
3457   { 0 }
3458   {
3459     \token_if_eq_charcode:NNTF #1 -
3460     {
3461       \exp_after:wN \exp_after:wN \exp_after:wN \__dim_get_remainder:w
3462       \int_eval:n { ( 2 * #2 - #3 ) / ( 2 * #3 ) } \__dim_sep:
3463       #2 \__dim_sep: #3 \__dim_sep: - \__dim_sep:
3464     }
3465     {
3466       \exp_after:wN \exp_after:wN \exp_after:wN \__dim_get_remainder:w
3467       \int_eval:n { ( 2 * #1#2 - #3 ) / ( 2 * #3 ) } \__dim_sep:
3468       #1#2 \__dim_sep: #3 \__dim_sep: \__dim_sep:
3469     }
3470   }
3471 }
3472 \cs_new:Npn \__dim_get_remainder:w #1 \__dim_sep: #2 \__dim_sep: #3 \__dim_sep:
3473 {
3474   \exp_after:wN \exp_after:wN \exp_after:wN \__dim_convert_remainder:w
3475   \int_eval:n { #2 - #1 * #3 } \__dim_sep:
3476   #3 \__dim_sep: #1 \__dim_sep:
3477 }
3478 \cs_new:Npn \__dim_convert_remainder:w #1 \__dim_sep: #2 \__dim_sep:
```



```
3479 {
3480 \exp_after:wN \exp_after:wN \exp_after:wN \_dim_test_candidate:w
3481 \int_eval:n { #1 * 65536 / #2 } \_dim_sep:
3482 #1 \_dim_sep: #2 \_dim_sep:
3483 }
3484 \cs_new:Npn \_dim_test_candidate:w #1 \_dim_sep: #2 \_dim_sep: #3 \_dim_sep:
3485 {
3486 \dim_compare:nNnTF { #2sp } =
3487 { \dim_to_decimal:n { #1sp } \_dim_eval:w #3sp \_dim_eval_end: }
3488 { \_dim_parse_decimal:w #1 \_dim_sep: }
3489 {
3490 \_dim_parse_decimal:w \int_eval:n { #1 + 1 } \_dim_sep:
3491 }
3492 }
3493 \cs_new:Npn \_dim_parse_decimal:w #1 \_dim_sep: #2 \_dim_sep: #3 \_dim_sep:
3494 {
3495 \exp_after:wN \_dim_parse_decimal_aux:w
3496 \int_value:w #3 \int_eval:w #2 + \dim_to_decimal:n { #1sp } \_dim_sep:
3497 }
3498 \cs_new:Npn \_dim_parse_decimal_aux:w #1 \_dim_sep: {#1}
3499 \cs_new_eq:NN \dim_show:N \__kernel_register_show:N
3500 \cs_generate_variant:Nn \dim_show:N { c }
3501 \cs_new_protected:Npn \dim_show:n
3502 { \__kernel_msg_show_eval:Nn \dim_eval:n }
3503 \cs_new_eq:NN \dim_log:N \__kernel_register_log:N
3504 \cs_new_eq:NN \dim_log:c \__kernel_register_log:c
3505 \cs_new_protected:Npn \dim_log:n
3506 { \__kernel_msg_log_eval:Nn \dim_eval:n }
3507 \dim_const:Nn \c_zero_dim { 0 pt }
3508 \dim_const:Nn \c_max_dim { 16383.99999 pt }
3509 \dim_new:N \l_tmpa_dim
3510 \dim_new:N \l_tmpb_dim
3511 \dim_new:N \g_tmpa_dim
3512 \dim_new:N \g_tmpb_dim
3513 \scan_new:N \s__skip_stop
3514 \cs_new_protected:Npn \skip_new:N #1
3515 {
3516 \__kernel_chk_if_free_cs:N #1
3517 \cs:w newskip \cs_end: #1
3518 }
3519 \cs_generate_variant:Nn \skip_new:N { c }
3520 \cs_new_protected:Npn \skip_const:Nn #1#2
3521 {
3522 \skip_new:N #1
3523 \tex_global:D #1 = \skip_eval:n {#2} \scan_stop:
3524 }
```

```
3525 \cs_generate_variant:Nn \skip_const:Nn { c }
3526 \cs_new_eq:NN \skip_zero:N \dim_zero:N
3527 \cs_new_eq:NN \skip_gzero:N \dim_gzero:N
3528 \cs_generate_variant:Nn \skip_zero:N { c }
3529 \cs_generate_variant:Nn \skip_gzero:N { c }
3530 \cs_new_protected:Npn \skip_zero_new:N #1
3531 { \skip_if_exist:NTF #1 { \skip_zero:N #1 } { \skip_new:N #1 } }
3532 \cs_new_protected:Npn \skip_gzero_new:N #1
3533 { \skip_if_exist:NTF #1 { \skip_gzero:N #1 } { \skip_new:N #1 } }
3534 \cs_generate_variant:Nn \skip_zero_new:N { c }
3535 \cs_generate_variant:Nn \skip_gzero_new:N { c }
3536 \prg_new_eq_conditional:NNn \skip_if_exist:N \cs_if_exist:N
3537 { TF , T , F , p }
3538 \prg_new_eq_conditional:NNn \skip_if_exist:c \cs_if_exist:c
3539 { TF , T , F , p }
3540 \cs_new_protected:Npn \skip_set:Nn #1#2
3541 { #1 = \tex_glueexpr:D #2 \scan_stop: }
3542 \cs_new_protected:Npn \skip_gset:Nn #1#2
3543 { \tex_global:D #1 = \tex_glueexpr:D #2 \scan_stop: }
3544 \cs_generate_variant:Nn \skip_set:Nn { NV , c , cV }
3545 \cs_generate_variant:Nn \skip_gset:Nn { NV , c , cV }
3546 \cs_new_protected:Npn \skip_set_eq:NN #1#2 { #1 = #2 }
3547 \cs_generate_variant:Nn \skip_set_eq:NN { c , Nc , cc }
3548 \cs_new_protected:Npn \skip_gset_eq:NN #1#2 { \tex_global:D #1 = #2 }
3549 \cs_generate_variant:Nn \skip_gset_eq:NN { c , Nc , cc }
3550 \cs_new_protected:Npn \skip_add:Nn #1#2
3551 { \tex_advance:D #1 \tex_glueexpr:D #2 \scan_stop: }
3552 \cs_new_protected:Npn \skip_gadd:Nn #1#2
3553 { \tex_global:D \tex_advance:D #1 \tex_glueexpr:D #2 \scan_stop: }
3554 \cs_generate_variant:Nn \skip_add:Nn { c }
3555 \cs_generate_variant:Nn \skip_gadd:Nn { c }
3556 \cs_new_protected:Npn \skip_sub:Nn #1#2
3557 { \tex_advance:D #1 - \tex_glueexpr:D #2 \scan_stop: }
3558 \cs_new_protected:Npn \skip_gsub:Nn #1#2
3559 { \tex_global:D \tex_advance:D #1 - \tex_glueexpr:D #2 \scan_stop: }
3560 \cs_generate_variant:Nn \skip_sub:Nn { c }
3561 \cs_generate_variant:Nn \skip_gsub:Nn { c }
3562 \prg_new_conditional:Npnn \skip_if_eq:nn #1#2 { p , T , F , TF }
3563 {
3564   \str_if_eq:eeTF { \skip_eval:n {#1} } { \skip_eval:n {#2} }
3565   { \prg_return_true: }
3566   { \prg_return_false: }
3567 }
3568 \cs_new_eq:NN \__skip_sep: \__kernel_int_sep:
3569 \cs_set_protected:Npn \__skip_tmp:w #1
3570 {
```

```
3571 \prg_new_conditional:Npnn \skip_if_finite:n ##1 { p , T , F , TF } 1357
3572 { 1357
3573 \exp_after:wN \__skip_if_finite:wwNw 1357
3574 \skip_use:N \tex_glueexpr:D ##1 \__skip_sep: \prg_return_false: 1357
3575 #1 \__skip_sep: \prg_return_true: \s__skip_stop 1357
3576 } 1357
3577 \cs_new:Npn \__skip_if_finite:wwNw ##1 #1 ##2 \__skip_sep: ##3 ##4 \s__skip_stop ✓
3578 {##3} 1357
3579 } 1357
3579 \exp_args:No \__skip_tmp:w { \tl_to_str:n { fil } } 1357
3580 \cs_new:Npn \skip_eval:n #1 1358
3581 { \skip_use:N \tex_glueexpr:D #1 \scan_stop: } 1358
3582 \cs_new_eq:NN \skip_use:N \dim_use:N 1358
3583 \cs_new_eq:NN \skip_use:c \dim_use:c 1358
3584 \cs_new_eq:NN \skip_horizontal:N \tex_hskip:D 1358
3585 \cs_new:Npn \skip_horizontal:n #1 1358
3586 { \skip_horizontal:N \tex_glueexpr:D #1 \scan_stop: } 1358
3587 \cs_new_eq:NN \skip_vertical:N \tex_vskip:D 1358
3588 \cs_new:Npn \skip_vertical:n #1 1358
3589 { \skip_vertical:N \tex_glueexpr:D #1 \scan_stop: } 1358
3590 \cs_generate_variant:Nn \skip_horizontal:N { c } 1359
3591 \cs_generate_variant:Nn \skip_vertical:N { c } 1359
3592 \cs_new_eq:NN \skip_show:N \__kernel_register_show:N 1359
3593 \cs_generate_variant:Nn \skip_show:N { c } 1359
3594 \cs_new_protected:Npn \skip_show:n 1359
3595 { \__kernel_msg_show_eval:Nn \skip_eval:n } 1359
3596 \cs_new_eq:NN \skip_log:N \__kernel_register_log:N 1359
3597 \cs_new_eq:NN \skip_log:c \__kernel_register_log:c 1359
3598 \cs_new_protected:Npn \skip_log:n 1359
3599 { \__kernel_msg_log_eval:Nn \skip_eval:n } 1359
3600 \skip_const:Nn \c_zero_skip { \c_zero_dim } 1360
3601 \skip_const:Nn \c_max_skip { \c_max_dim } 1360
3602 \skip_new:N \l_tmpa_skip 1360
3603 \skip_new:N \l_tmpb_skip 1360
3604 \skip_new:N \g_tmpa_skip 1360
3605 \skip_new:N \g_tmpb_skip 1360
3606 \cs_new_protected:Npn \muskip_new:N #1 1360
3607 { 1360
3608 \__kernel_chk_if_free_cs:N #1 1360
3609 \cs:w newmuskip \cs_end: #1 1360
3610 } 1361
3611 \cs_generate_variant:Nn \muskip_new:N { c } 1361
3612 \cs_new_protected:Npn \muskip_const:Nn #1#2 1361
3613 { 1361
3614 \muskip_new:N #1 1361
3615 \tex_global:D #1 = \muskip_eval:n {#2} \scan_stop: 1361
```

```
3616 }
3617 \cs_generate_variant:Nn \muskip_const:Nn { c }
3618 \cs_new_protected:Npn \muskip_zero:N #1
3619 { #1 = \c_zero_muskip }
3620 \cs_new_protected:Npn \muskip_gzero:N #1
3621 { \tex_global:D #1 = \c_zero_muskip }
3622 \cs_generate_variant:Nn \muskip_zero:N { c }
3623 \cs_generate_variant:Nn \muskip_gzero:N { c }
3624 \cs_new_protected:Npn \muskip_zero_new:N #1
3625 { \muskip_if_exist:NTF #1 { \muskip_zero:N #1 } { \muskip_new:N #1 } }
3626 \cs_new_protected:Npn \muskip_gzero_new:N #1
3627 { \muskip_if_exist:NTF #1 { \muskip_gzero:N #1 } { \muskip_new:N #1 } }
3628 \cs_generate_variant:Nn \muskip_zero_new:N { c }
3629 \cs_generate_variant:Nn \muskip_gzero_new:N { c }
3630 \prg_new_eq_conditional:NNn \muskip_if_exist:N \cs_if_exist:N
3631 { TF , T , F , p }
3632 \prg_new_eq_conditional:NNn \muskip_if_exist:c \cs_if_exist:c
3633 { TF , T , F , p }
3634 \cs_new_protected:Npn \muskip_set:Nn #1#2
3635 { #1 = \tex_muexpr:D #2 \scan_stop: }
3636 \cs_new_protected:Npn \muskip_gset:Nn #1#2
3637 { \tex_global:D #1 = \tex_muexpr:D #2 \scan_stop: }
3638 \cs_generate_variant:Nn \muskip_set:Nn { NV , c , cV }
3639 \cs_generate_variant:Nn \muskip_gset:Nn { NV , c , cV }
3640 \cs_new_protected:Npn \muskip_set_eq:NN #1#2 { #1 = #2 }
3641 \cs_generate_variant:Nn \muskip_set_eq:NN { c , Nc , cc }
3642 \cs_new_protected:Npn \muskip_gset_eq:NN #1#2 { \tex_global:D #1 = #2 }
3643 \cs_generate_variant:Nn \muskip_gset_eq:NN { c , Nc , cc }
3644 \cs_new_protected:Npn \muskip_add:Nn #1#2
3645 { \tex_advance:D #1 \tex_muexpr:D #2 \scan_stop: }
3646 \cs_new_protected:Npn \muskip_gadd:Nn #1#2
3647 { \tex_global:D \tex_advance:D #1 \tex_muexpr:D #2 \scan_stop: }
3648 \cs_generate_variant:Nn \muskip_add:Nn { c }
3649 \cs_generate_variant:Nn \muskip_gadd:Nn { c }
3650 \cs_new_protected:Npn \muskip_sub:Nn #1#2
3651 { \tex_advance:D #1 - \tex_muexpr:D #2 \scan_stop: }
3652 \cs_new_protected:Npn \muskip_gsub:Nn #1#2
3653 { \tex_global:D \tex_advance:D #1 - \tex_muexpr:D #2 \scan_stop: }
3654 \cs_generate_variant:Nn \muskip_sub:Nn { c }
3655 \cs_generate_variant:Nn \muskip_gsub:Nn { c }
3656 \cs_new:Npn \muskip_eval:n #1
3657 { \muskip_use:N \tex_muexpr:D #1 \scan_stop: }
3658 \cs_new_eq:NN \muskip_use:N \dim_use:N
3659 \cs_new_eq:NN \muskip_use:c \dim_use:c
3660 \cs_new_eq:NN \muskip_show:N \__kernel_register_show:N
3661 \cs_generate_variant:Nn \muskip_show:N { c }
```

```
3662 \cs_new_protected:Npn \muskip_show:n
3663 { \__kernel_msg_show_eval:Nn \muskip_eval:n }
3664 \cs_new_eq:NN \muskip_log:N \__kernel_register_log:N
3665 \cs_new_eq:NN \muskip_log:c \__kernel_register_log:c
3666 \cs_new_protected:Npn \muskip_log:n
3667 { \__kernel_msg_log_eval:Nn \muskip_eval:n }
3668 \muskip_const:Nn \c_zero_muskip { 0 mu }
3669 \muskip_const:Nn \c_max_muskip { 16383.99999 mu }
3670 \muskip_new:N \l_tmpa_muskip
3671 \muskip_new:N \l_tmpb_muskip
3672 \muskip_new:N \g_tmpa_muskip
3673 \muskip_new:N \g_tmpb_muskip
3674 %% File: l3keys.dtx
3675 \scan_new:N \s__keyval_nil
3676 \scan_new:N \s__keyval_mark
3677 \scan_new:N \s__keyval_stop
3678 \scan_new:N \s__keyval_tail
3679 \bool_new:N \l__kernel_keyval_allow_blank_keys_bool
3680 \group_begin:
3681 \cs_set_protected:Npn \__keyval_tmp:w #1#2
3682 {
3683   \cs_new:Npn \keyval_parse:nnn ##1 ##2 ##3
3684   {
3685     \__kernel_exp_not:w \tex_expanded:D
3686     {
3687       {
3688         \__keyval_loop_active:nnw {##1} {##2}
3689         \s__keyval_mark ##3 #1 \s__keyval_tail #1
3690       }
3691     }
3692   }
3693   \cs_new_eq:NN \keyval_parse:NNn \keyval_parse:nnn
3694   \cs_new:Npn \__keyval_loop_active:nnw ##1 ##2 ##3 #1
3695   {
3696     \__keyval_if_recursion_tail:w ##3
3697     \__keyval_end_loop_active:w \s__keyval_tail
3698     \__keyval_loop_other:nnw {##1} {##2} ##3 , \s__keyval_tail ,
3699   }
3700   \cs_new:Npn \__keyval_split_other:w ##1 = ##2 \s__keyval_mark ##3
3701   { ##3 ##1 \s__keyval_stop \s__keyval_mark ##2 }
3702   \cs_new:Npn \__keyval_split_active:w ##1 #2 ##2 \s__keyval_mark ##3
3703   { ##3 ##1 \s__keyval_stop \s__keyval_mark ##2 }
3704   \cs_new:Npn \__keyval_loop_other:nnw ##1 ##2 ##3 ,
3705   {
3706     \__keyval_if_recursion_tail:w ##3
3707     \__keyval_end_loop_other:w \s__keyval_tail
```



```
3708     \__keyval_split_active:w ##3 \s__keyval_nil 13708
3709     \s__keyval_mark \__keyval_split_active_auxi:w 13709
3710     #2 \s__keyval_mark \__keyval_clean_up_active:w 13710
3711     {##1} {##2} 13711
3712     \s__keyval_mark 13712
3713 } 13713
3714 \cs_new:Npn \__keyval_split_active_auxi:w ##1 \s__keyval_stop 13714
3715 { 13715
3716     \__keyval_split_other:w ##1 \s__keyval_nil 13716
3717     \s__keyval_mark \__keyval_misplaced_equal_after_active_error:w 13717
3718     = \s__keyval_mark \__keyval_split_active_auxii:w 13718
3719 } 13719
3720 \cs_new:Npn \__keyval_split_active_auxii:w 13720
3721     ##1 \s__keyval_nil \s__keyval_mark 13721
3722     \__keyval_misplaced_equal_after_active_error:w 13722
3723     \s__keyval_stop \s__keyval_mark 13722
3724     ##2 \s__keyval_nil #2 \s__keyval_mark \__keyval_clean_up_active:w 13723
3725     { \__keyval_trim:nN {##1} \__keyval_split_active_auxiii:w ##2 \s__keyval_nil } 13724
3726 \cs_new:Npn \__keyval_split_active_auxiii:w ##1 ##2 \s__keyval_nil 13725
3727 { 13726
3728     \__keyval_split_active:w ##2 \s__keyval_nil 13727
3729     \s__keyval_mark \__keyval_misplaced_equal_in_split_error:w 13728
3730     #2 \s__keyval_mark \__keyval_split_active_auxiv:w 13729
3731     {##1} 13730
3732 } 13731
3733 \cs_new:Npn \__keyval_split_active_auxiv:w 13732
3734     ##1 \s__keyval_nil \s__keyval_mark \__keyval_misplaced_equal_in_split_error:w 13733
3735     \s__keyval_stop \s__keyval_mark 13734
3736 { 13735
3737     \__keyval_split_other:w ##1 \s__keyval_nil 13736
3738     \s__keyval_mark \__keyval_misplaced_equal_in_split_error:w 13737
3739     = \s__keyval_mark \__keyval_split_active_auxv:w 13738
3740 } 13739
3741 \cs_new:Npn \__keyval_split_active_auxv:w 13740
3742     ##1 \s__keyval_nil \s__keyval_mark \__keyval_misplaced_equal_in_split_error:w 13741
3743     \s__keyval_stop \s__keyval_mark 13742
3744     { \__keyval_trim:nN { ##1 } \__keyval_pair:nnnn } 13743
3745 \cs_new:Npn \__keyval_clean_up_active:w 13744
3746     ##1 \s__keyval_nil \s__keyval_mark \__keyval_split_active_auxi:w 13745
3747     \s__keyval_stop \s__keyval_mark 13746
3748 { 13747
3749     \__keyval_split_other:w ##1 \s__keyval_nil 13748
3750     \s__keyval_mark \__keyval_split_other_auxi:w 13749
3751     = \s__keyval_mark \__keyval_clean_up_other:w 13750
3752 } 13751
3753 \cs_new:Npn \__keyval_split_other_auxi:w ##1 \s__keyval_stop 13752
```

```
3752 { \__keyval_trim:nN { ##1 } \__keyval_split_other_auxii:w } 13752
3753 \cs_new:Npn \__keyval_split_other_auxii:w 13753
3754   ##1 ##2 \s__keyval_nil = \s__keyval_mark \__keyval_clean_up_other:w 13754
3755 { 13755
3756   \__keyval_split_other:w ##2 \s__keyval_nil 13756
3757   \s__keyval_mark \__keyval_misplaced_equal_in_split_error:w 13757
3758   = \s__keyval_mark \__keyval_split_other_auxiii:w 13758
3759   { ##1 } 13759
3760 } 13760
3761 \cs_new:Npn \__keyval_split_other_auxiii:w 13761
3762   ##1 \s__keyval_nil \s__keyval_mark \__keyval_misplaced_equal_in_split_error:w 13762
3763   \s__keyval_stop \s__keyval_mark 13763
3764   { \__keyval_trim:nN { ##1 } \__keyval_pair:nnnn } 13764
3765 \cs_new:Npn \__keyval_clean_up_other:w 13765
3766   ##1 \s__keyval_nil \s__keyval_mark \__keyval_split_other_auxi:w ✓ 13766
\s__keyval_stop \s__keyval_mark 13766
3767 { 13767
3768   \__keyval_if_blank:w ##1 \s__keyval_nil \s__keyval_stop \__keyval_blank_true:w 13768
3769   \s__keyval_mark \s__keyval_stop 13769
3770   \__keyval_trim:nN { ##1 } \__keyval_key:nn 13770
3771 } 13771
3772 \cs_new:Npn \__keyval_misplaced_equal_after_active_error:w 13772
3773   \s__keyval_mark ##1 \s__keyval_stop \s__keyval_mark ##2 \s__keyval_nil 13773
3774   = \s__keyval_mark \__keyval_split_active_auxii:w 13774
3775   \s__keyval_mark ##3 \s__keyval_nil 13775
3776   #2 \s__keyval_mark \__keyval_clean_up_active:w 13776
3777 { 13777
3778   \msg_expandable_error:nn 13778
3779     { keyval } { misplaced-equals-sign } 13779
3780   \__keyval_loop_other:nnw 13780
3781 } 13781
3782 \cs_new:Npn \__keyval_misplaced_equal_in_split_error:w 13782
3783   \s__keyval_mark ##1 \s__keyval_stop \s__keyval_mark ##2 \s__keyval_nil 13783
3784   ##3 \s__keyval_mark ##4 ##5 13784
3785 { 13785
3786   \msg_expandable_error:nn 13786
3787     { keyval } { misplaced-equals-sign } 13787
3788   \__keyval_loop_other:nnw 13788
3789 } 13789
3790 \cs_new:Npn \__keyval_end_loop_other:w 13790
3791   \s__keyval_tail 13791
3792   \__keyval_split_active:w 13792
3793   \s__keyval_mark \s__keyval_tail 13793
3794   \s__keyval_nil \s__keyval_mark 13794
3795   \__keyval_split_active_auxi:w 13795
3796   #2 \s__keyval_mark \__keyval_clean_up_active:w 13796
```



```
3797 { \__keyval_loop_active:nnw } 1379
3798 \cs_new:Npn \__keyval_end_loop_active:w 1379
3799 \s_keyval_tail 1379
3800 \__keyval_loop_other:nnw ##1 \s__keyval_mark \s__keyval_tail , \s__keyval_tail ✓ 1380
, 1380
3801 { } 1380
3802 } 1380
3803 \char_set_catcode_active:n { `\", } 1380
3804 \char_set_catcode_active:n { `|= } 1380
3805 \__keyval_tmp:w , = 1380
3806 \group_end: 1380
3807 \cs_generate_variant:Nn \keyval_parse:NNn { NNv , NNv } 1380
3808 \cs_generate_variant:Nn \keyval_parse:nnn { nnV , nnv } 1380
3809 \group_begin: 1380
3810 \cs_set_protected:Npn \__keyval_tmp:w #1#2 1381
3811 { 1381
3812 \cs_new:Npn \__keyval_pair:nnnn ##1 ##2 ##3 ##4 1381
3813 { 1381
3814 \__keyval_if_blank:w \s__keyval_mark ##2 \s__keyval_nil \s__keyval_stop ✓ 1381
\__keyval_blank_key_error:w 1381
3815 \s__keyval_mark \s__keyval_stop 1381
3816 #1 1381
3817 \exp_not:n { ##4 {##2} {##1} } 1381
3818 #2 1381
3819 \__keyval_loop_other:nnw {##3} {##4} 1381
3820 } 1382
3821 \cs_new:Npn \__keyval_key:nn ##1 ##2 1382
3822 { 1382
3823 \__keyval_if_blank:w \s__keyval_mark ##1 \s__keyval_nil \s__keyval_stop ✓ 1382
\__keyval_blank_key_error:w 1382
3824 \s__keyval_mark \s__keyval_stop 1382
3825 #1 1382
3826 \exp_not:n { ##2 {##1} } 1382
3827 #2 1382
3828 \__keyval_loop_other:nnw {##2} 1382
3829 } 1382
3830 } 1383
3831 \__keyval_tmp:w { } { } 1383
3832 \group_end: 1383
3833 \cs_new:Npn \__keyval_if_empty:w #1 \s__keyval_mark \s__keyval_stop { } 1383
3834 \cs_new:Npn \__keyval_if_blank:w \s__keyval_mark #1 { \__keyval_if_empty:w ✓ 1383
\s__keyval_mark } 1383
3835 \cs_new:Npn \__keyval_if_recursion_tail:w \s__keyval_mark #1 \s__keyval_tail { } 1383
3836 \cs_new:Npn \__keyval_blank_true:w \s__keyval_mark \s__keyval_stop \__keyval_trim:nN #1 ✓ 1383
\__keyval_key:nn 1383
3837 { \__keyval_loop_other:nnw } 1383
```

```

3838 \cs_new:Npn \__keyval_blank_key_error:w \s__keyval_mark \s__keyval_stop #1 ↙
      \__keyval_loop_other:nnw
13838
3839 {
13839
3840   \bool_if:NTF \l__kernel_keyval_allow_blank_keys_bool
13840
3841     { #1 }
13841
3842     { \msg_expandable_error:nn { keyval } { blank-key-name } }
13842
3843     \__keyval_loop_other:nnw
13843
3844   }
13844
3845 \msg_new:nnn { keyval } { misplaced-equals-sign }
13845
3846   { Misplaced~'='~in~key-value~input~\msg_line_context: }
13846
3847 \msg_new:nnn { keyval } { blank-key-name }
13847
3848   { Blank~key~name~in~key-value~input~\msg_line_context: }
13848
3849 \prop_gput:Nnn \g_msg_module_name_prop { keyval } { LaTeX }
13849
3850 \prop_gput:Nnn \g_msg_module_type_prop { keyval } { }
13850
3851 \group_begin:
13851
3852   \cs_set_protected:Npn \__keyval_tmp:w #1
13852
3853   {
13853
3854     \cs_new:Npn \__keyval_trim:nN ##1
13854
3855     {
13855
3856       \__keyval_trim_auxi:w
13856
3857       ##1
13857
3858       \s__keyval_nil
13858
3859       \s__keyval_mark #1 { }
13859
3860       \s__keyval_mark \__keyval_trim_auxii:w
13860
3861       \__keyval_trim_auxiii:w
13861
3862       #1 \s__keyval_nil
13862
3863       \__keyval_trim_auxiv:w
13863
3864     }
13864
3865 \cs_new:Npn \__keyval_trim_auxi:w ##1 \s__keyval_mark #1 ##2 \s__keyval_mark ##3
13865
3866 {
13866
3867   ##3
13867
3868   \__keyval_trim_auxi:w
13868
3869   \s__keyval_mark
13869
3870   ##2
13870
3871   \s__keyval_mark #1 {##1}
13871
3872 }
13872
3873 \cs_new:Npn \__keyval_trim_auxii:w \__keyval_trim_auxi:w \s__keyval_mark ↙
      \s__keyval_mark ##1
13873
3874 {
13874
3875   \__keyval_trim_auxiii:w
13875
3876   ##1
13876
3877 }
13877
3878 \cs_new:Npn \__keyval_trim_auxiii:w ##1 #1 \s__keyval_nil ##2
13878
3879 {
13879
3880   ##2
13880
3881   ##1 \s__keyval_nil
13881

```

```
3882     \__keyval_trim_auxiii:w
3883 }
3884 \cs_new:Npn \__keyval_trim_auxiv:w
3885     \s__keyval_mark ##1 \s__keyval_nil
3886     \__keyval_trim_auxiii:w \s__keyval_nil \__keyval_trim_auxiii:w
3887     ##2
3888     { ##2 { ##1 } }
3889 }
3890 \__keyval_tmp:w { ~ }
3891 \group_end:
3892 \str_const:Nn \c__keys_code_root_str { key~code~>~ }
3893 \str_const:Nn \c__keys_check_root_str { key~check~>~ }
3894 \str_const:Nn \c__keys_default_root_str { key~default~>~ }
3895 \str_const:Nn \c__keys_groups_root_str { key~groups~>~ }
3896 \str_const:Nn \c__keys_inherit_root_str { key~inherit~>~ }
3897 \str_const:Nn \c__keys_type_root_str { key~type~>~ }
3898 \str_const:Nn \c__keys_props_root_str { key~prop~>~ }
3899 \int_new:N \l_keys_choice_int
3900 \tl_new:N \l_keys_choice_tl
3901 \clist_new:N \l__keys_groups_clist
3902 \bool_new:N \l__keys_inherit_bool
3903 \clist_new:N \l__keys_inherit_clist
3904 \str_new:N \l_keys_key_str
3905 \tl_new:N \l_keys_key_tl
3906 \str_new:N \l__keys_module_str
3907 \bool_new:N \l__keys_no_value_bool
3908 \bool_new:N \l__keys_only_known_bool
3909 \str_new:N \l_keys_path_str
3910 \tl_new:N \l_keys_path_tl
3911 \str_new:N \l__keys_inherit_str
3912 \tl_new:N \l__keys_relative_tl
3913 \tl_set:Nn \l__keys_relative_tl { \q__keys_no_value }
3914 \str_new:N \l__keys_property_str
3915 \bool_new:N \l__keys_selective_bool
3916 \bool_new:N \l__keys_exclude_bool
3917 \clist_new:N \l__keys_selective_clist
3918 \clist_new:N \l__keys_tmp_clist
3919 \clist_new:N \l__keys_unused_clist
3920 \tl_new:N \l_keys_value_tl
3921 \bool_new:N \l__keys_tmp_bool
3922 \tl_new:N \l__keys_tmpa_tl
3923 \tl_new:N \l__keys_tmpb_tl
3924 \bool_new:N \l__keys_precompile_bool
3925 \tl_new:N \l__keys_precompile_tl
3926 \prop_new:N \l_keys_usage_load_prop
3927 \prop_new:N \l_keys_usage_preamble_prop
```

```
3928 \scan_new:N \s__keys_nil
3929 \scan_new:N \s__keys_mark
3930 \scan_new:N \s__keys_stop
3931 \quark_new:N \q__keys_no_value
3932 \__kernel_quark_new_conditional:Nn \__keys_quark_if_no_value:N { TF }
3933 \cs_new_protected:Npn \__keys_precompile:n #1
3934 {
3935     \bool_if:NTF \l__keys_precompile_bool
3936     { \tl_put_right:Nn \l__keys_precompile_tl }
3937     { \use:n }
3938     {#1}
3939 }
3940 \cs_new_protected:Npn \__keys_cs_undefine:c #1
3941 {
3942     \if_cs_exist:w #1 \cs_end:
3943     \else:
3944         \use_i:nnnn
3945     \fi:
3946     \cs_set_eq:cN {#1} \tex_undefined:D
3947 }
3948 \cs_new_protected:Npn \keys_define:nn #1#2
3949 {
3950     \use:e
3951     {
3952         \exp_not:n
3953         {
3954             \str_set:Ne \l__keys_module_str { \__keys_trim_spaces:n {#1} }
3955             \keyval_parse:NNn \__keys_define:n \__keys_define:nn {#2}
3956         }
3957         \__keys_reset_var:N \l__keys_module_str
3958         \__keys_reset_var:N \l__keys_inherit_str
3959         \__keys_reset_var:N \l_keys_choice_tl
3960         \__keys_reset_var:N \l_keys_key_tl
3961         \__keys_reset_var:N \l_keys_key_str
3962         \__keys_reset_var:N \l_keys_path_tl
3963         \__keys_reset_var:N \l_keys_path_str
3964         \__keys_reset_var:N \l_keys_value_tl
3965         \int_set:Nn \l_keys_choice_int { \int_use:N \l_keys_choice_int }
3966     }
3967 }
3968 \cs_generate_variant:Nn \keys_define:nn { ne , nx }
3969 \cs_new_protected:Npn \__keys_define:n #1
3970 {
3971     \bool_set_true:N \l__keys_no_value_bool
3972     \__keys_define_aux:nn {#1} { }
3973 }
```

```
3974 \cs_new_protected:Npn \__keys_define:nn #1#2
3975 {
3976     \bool_set_false:N \l__keys_no_value_bool
3977     \__keys_define_aux:nn {#1} {#2}
3978 }
3979 \cs_new_protected:Npn \__keys_define_aux:nn #1#2
3980 {
3981     \__keys_property_find:n {#1}
3982     \cs_if_exist:cTF { \c__keys_props_root_str \l__keys_property_str }
3983     { \__keys_define_code:n {#2} }
3984     {
3985         \str_if_empty:NF \l__keys_property_str
3986         {
3987             \msg_error:nnee { keys } { property-unknown }
3988             \l__keys_property_str \l_keys_path_str
3989         }
3990     }
3991 }
3992 \cs_new_protected:Npn \__keys_property_find:n #1
3993 {
3994     \exp_after:wN \__keys_property_find_auxi:w \tl_to_str:n {#1}
3995     \s__keys_nil \__keys_property_find_auxii:w
3996     . \s__keys_nil \__keys_property_find_err:w
3997 }
3998 \cs_new:Npn \__keys_property_find_auxi:w #1 . #2 \s__keys_nil #3
3999 {
4000     #3 #1 \s__keys_mark #2 \s__keys_nil #3
4001 }
4002 \cs_new_protected:Npn \__keys_property_find_auxii:w
4003     #1 \s__keys_mark #2 \s__keys_nil \__keys_property_find_auxii:w . \s__keys_nil
4004     \__keys_property_find_err:w
4005 {
4006     \cs_set_nopar:Npe \l_keys_path_str
4007     {
4008         \str_if_empty:NF \l__keys_module_str { \l__keys_module_str / }
4009         \exp_after:wN \__keys_trim_spaces:n \tex_expanded:D {{
4010             #1
4011             \if_false: } } } \fi:
4012         \__keys_property_find_auxi:w #2 \s__keys_nil \__keys_property_find_auxiii:w
4013         . \s__keys_nil \__keys_property_find_auxiv:w
4014     }
4015 \cs_new:Npn \__keys_property_find_auxiii:w #1 \s__keys_mark #2 . #3 \s__keys_nil #4
4016 {
4017     . #1 #4 #2 \s__keys_mark #3 \s__keys_nil #4
4018 }
4019 \cs_new:Npn \__keys_property_find_auxiv:w
```

```
4020     #1 \s__keys_nil \__keys_property_find_auxiii:w 14020
4021     \s__keys_mark \s__keys_nil \__keys_property_find_auxiv:w 14021
4022 { 14022
4023     \if_false: {{{ \fi: }}} 14023
4024     \cs_set_nopar:Npe \l__keys_property_str { . #1 } 14024
4025     \tl_set_eq:NN \l_keys_path_tl \l_keys_path_str 14025
4026 } 14026
4027 \cs_new_protected:Npn \__keys_property_find_err:w 14027
4028     #1 \s__keys_nil #2 \__keys_property_find_err:w 14028
4029 { 14029
4030     \str_clear:N \l__keys_property_str 14030
4031     \msg_error:nnn { keys } { no-property } {#1} 14031
4032 } 14032
4033 \cs_new_protected:Npn \__keys_define_code:n #1 14033
4034 { 14034
4035     \bool_if:NTF \l__keys_no_value_bool 14035
4036     { 14036
4037         \__keys_define_code:nnn 14037
4038         { \use:c { \c__keys_props_root_str \l__keys_property_str } {#1} } 14038
4039         { \use:c { \c__keys_props_root_str \l__keys_property_str } } 14039
4040         { 14040
4041             \msg_error:nnee { keys } { property-requires-value } 14041
4042             \l__keys_property_str \l_keys_path_str 14042
4043         } 14043
4044     } 14044
4045     { \use:c { \c__keys_props_root_str \l__keys_property_str } {#1} } 14045
4046 } 14046
4047 \cs_new:Npe \__keys_define_code:nnn 14047
4048 { 14048
4049     \exp_not:N \exp_after:wN \exp_not:N \__keys_define_code:w 14049
4050     \exp_not:N \l__keys_property_str 14050
4051     \c_colon_str \c_colon_str 14051
4052     \exp_not:N \s__keys_stop 14052
4053 } 14053
4054 \use:e 14054
4055 { 14055
4056     \cs_new:Npn \exp_not:N \__keys_define_code:w 14056
4057     #1 \c_colon_str #2 \c_colon_str #3 \exp_not:N \s__keys_stop 14057
4058 } 14058
4059 { 14059
4060     \tl_if_empty:nTF {#3} 14060
4061     { \use_i:nnn } 14061
4062     { 14062
4063         \tl_if_empty:nTF {#2} 14063
4064         { \use_ii:nnn } 14064
4065         { \use_iii:nnn } 14065
```

```
4066     }
4067 }
4068 \cs_new_protected:Npn \__keys_bool_set:Nn #1#2
4069 { \__keys_bool_set:Nnnn #1 {#2} { true } { false } }
4070 \cs_generate_variant:Nn \__keys_bool_set:Nn { c }
4071 \cs_new_protected:Npn \__keys_bool_set_inverse:Nn #1#2
4072 { \__keys_bool_set:Nnnn #1 {#2} { false } { true } }
4073 \cs_generate_variant:Nn \__keys_bool_set_inverse:Nn { c }
4074 \cs_new_protected:Npn \__keys_bool_set:Nnnn #1#2#3#4
4075 {
4076   \bool_if_exist:NF #1 { \bool_new:N #1 }
4077   \__keys_choice_make:
4078   \__keys_cmd_set:ne { \l_keys_path_str / true }
4079   { \exp_not:c { bool_ #2 set_ #3 :N } \exp_not:N #1 }
4080   \__keys_cmd_set:ne { \l_keys_path_str / false }
4081   { \exp_not:c { bool_ #2 set_ #4 :N } \exp_not:N #1 }
4082   \__keys_cmd_set_direct:nn { \l_keys_path_str / unknown }
4083   {
4084     \msg_error:nne { keys } { boolean-values-only }
4085     \l_keys_path_str
4086   }
4087   \__keys_default_set:n { true }
4088 }
4089 \cs_generate_variant:Nn \__keys_bool_set:Nn { c }
4090 \cs_new_protected:Npn \__keys_choice_make:
4091 { \__keys_choice_make:N \__keys_choice_find:n }
4092 \cs_new_protected:Npn \__keys_multichoice_make:
4093 { \__keys_choice_make:N \__keys_multichoice_find:n }
4094 \cs_new_protected:Npn \__keys_choice_make:N #1
4095 {
4096   \cs_if_exist:cTF
4097   { \c__keys_type_root_str \__keys_parent:o \l_keys_path_str }
4098   {
4099     \str_if_eq:vnTF
4100     { \c__keys_type_root_str \__keys_parent:o \l_keys_path_str }
4101     { choice }
4102     {
4103       \msg_error:nnee { keys } { nested-choice-key }
4104       \l_keys_path_tl { \__keys_parent:o \l_keys_path_str }
4105     }
4106     { \__keys_choice_make_aux:N #1 }
4107   }
4108   { \__keys_choice_make_aux:N #1 }
4109 }
4110 \cs_new_protected:Npn \__keys_choice_make_aux:N #1
4111 {
```



```
4112 \cs_set_nopar:cpn { \c__keys_type_root_str \l_keys_path_str }
4113 { choice }
4114 \__keys_cmd_set_direct:nn \l_keys_path_str { #1 {##1} }
4115 \__keys_cmd_set_direct:nn { \l_keys_path_str / unknown }
4116 {
4117   \msg_error:nnee { keys } { choice-unknown }
4118   \l_keys_path_str {##1}
4119 }
4120 }
4121 \cs_new_protected:Npn \__keys_choices_make:nn
4122 { \__keys_choices_make:Nnn \__keys_choice_make: }
4123 \cs_new_protected:Npn \__keys_multichoice_make:nn
4124 { \__keys_choices_make:Nnn \__keys_multichoice_make: }
4125 \cs_new_protected:Npn \__keys_choices_make:Nnn #1#2#3
4126 {
4127   #1
4128   \int_zero:N \l_keys_choice_int
4129   \clist_map_inline:nn {#2}
4130   {
4131     \int_incr:N \l_keys_choice_int
4132     \__keys_cmd_set:ne
4133     { \l_keys_path_str / \__keys_trim_spaces:n {##1} }
4134     {
4135       \tl_set:Nn \exp_not:N \l_keys_choice_tl {##1}
4136       \int_set:Nn \exp_not:N \l_keys_choice_int
4137       { \int_use:N \l_keys_choice_int }
4138       \exp_not:n {#3}
4139     }
4140   }
4141 }
4142 \cs_new_protected:Npn \__keys_cmd_set:nn #1#2
4143 { \__keys_cmd_set_direct:nn {#1} { \__keys_precompile:n {#2} } }
4144 \cs_generate_variant:Nn \__keys_cmd_set:nn { ne , Vn , Vo }
4145 \cs_new_protected:Npn \__keys_cmd_set_direct:nn #1#2
4146 { \cs_set_protected:cpn { \c__keys_code_root_str #1 } ##1 {#2} }
4147 \cs_new_protected:Npn \__keys_cs_set:NNpn #1#2#3#
4148 {
4149   \cs_set_protected:cpe { \c__keys_code_root_str \l_keys_path_str } ##1
4150   {
4151     \__keys_precompile:n
4152     { #1 \exp_not:N #2 \exp_not:n {#3} {##1} }
4153   }
4154   \use_none:n
4155 }
4156 \cs_generate_variant:Nn \__keys_cs_set:NNpn { Nc }
4157 \cs_new_protected:Npn \__keys_default_set:n #1
```

```

4158 {
4159     \tl_if_empty:nTF {#1}
4160     {
4161         \__keys_cs_undefine:c
4162         { \c__keys_default_root_str \l_keys_path_str }
4163     }
4164     {
4165         \cs_set_nopar:cpe
4166         { \c__keys_default_root_str \l_keys_path_str }
4167         { \exp_not:n {#1} }
4168         \__keys_value_requirement:nn { required } { false }
4169     }
4170 }
4171 \cs_new_protected:Npn \__keys_groups_set:n #1
4172 {
4173     \clist_set:Nc \l__keys_groups_clist { \tl_to_str:n {#1} }
4174     \clist_if_empty:NTF \l__keys_groups_clist
4175     {
4176         \__keys_cs_undefine:c
4177         { \c__keys_groups_root_str \l_keys_path_str }
4178     }
4179     {
4180         \cs_set_eq:cN { \c__keys_groups_root_str \l_keys_path_str }
4181         \l__keys_groups_clist
4182     }
4183 }
4184 \cs_new_protected:Npn \__keys_inherit:n #1
4185 {
4186     \__keys_undefine:
4187     \clist_set:Nn \l__keys_inherit_clist {#1}
4188     \cs_set_eq:cN { \c__keys_inherit_root_str \l_keys_path_str }
4189     \l__keys_inherit_clist
4190 }
4191 \cs_new_protected:Npn \__keys_initialise:n #1
4192 {
4193     \typeout{----->}
4194     \exp_after:wN \__keys_find_key_module:wNN
4195     \l_keys_path_str \s__keys_stop
4196     \l_keys_key_tl \l_keys_key_str
4197     \tl_set_eq:NN \l_keys_key_tl \l_keys_key_str
4198     \tl_set:Nn \l_keys_value_tl {#1}
4199     \cs_if_exist:cTF { \c__keys_code_root_str \l_keys_path_str }
4200     {
4201         \str_clear:N \l__keys_inherit_str
4202         \__keys_execute:nn \l_keys_path_str {#1}
4203     }

```

```
4204     {
4205         \cs_if_exist:cT
4206             { \c_keys_inherit_root_str \__keys_parent:o \l_keys_path_str }
4207             { \__keys_execute_inherit: }
4208     }
4209 }
4210 \cs_new_protected:Npn \__keys_legacy_if_set:nn #1#2
4211     { \__keys_legacy_if_set:nnnn {#1} {#2} { true } { false } }
4212 \cs_new_protected:Npn \__keys_legacy_if_set_inverse:nn #1#2
4213     { \__keys_legacy_if_set:nnnn {#1} {#2} { false } { true } }
4214 \cs_new_protected:Npn \__keys_legacy_if_set:nnnn #1#2#3#4
4215     {
4216         \__keys_choice_make:
4217         \__keys_cmd_set:ne { \l_keys_path_str / true }
4218             { \exp_not:c { legacy_if_#2 set_ #3 :n } { \exp_not:n {#1} } }
4219         \__keys_cmd_set:ne { \l_keys_path_str / false }
4220             { \exp_not:c { legacy_if_#2 set_ #4 :n } { \exp_not:n {#1} } }
4221         \__keys_cmd_set:nn { \l_keys_path_str / unknown }
4222     {
4223         \msg_error:nne { keys } { boolean-values-only }
4224         \l_keys_path_str
4225     }
4226     \__keys_default_set:n { true }
4227     \cs_if_exist:cF { if#1 }
4228     {
4229         \cs:w newif \exp_after:wN \cs_end:
4230         \cs:w if#1 \cs_end:
4231     }
4232 }
4233 \cs_new_protected:Npn \__keys_meta_make:n
4234     { \exp_args:NV \__keys_meta_make:nn \l__keys_module_str }
4235 \cs_new_protected:Npn \__keys_meta_make:nn #1#2
4236     {
4237         \exp_args:NV \__keys_cmd_set_direct:nn
4238             \l_keys_path_str { \__keys_set:nn {#1} {#2} }
4239     }
4240 \cs_new_protected:Npn \__keys_prop_put:Nn #1#2
4241     {
4242         \prop_if_exist:NF #1 { \prop_new:N #1 }
4243         \exp_after:wN \__keys_find_key_module:wNN \l_keys_path_str \s__keys_stop
4244             \l__keys_tmpa_tl \l__keys_tmpb_tl
4245         \__keys_cmd_set:ne \l_keys_path_str
4246     {
4247         \exp_not:c { prop_ #2 put:Nnn }
4248         \exp_not:N #1
4249         { \l__keys_tmpb_tl }
```

```
4250 \exp_not:n { {##1} }
4251 }
4252 }
4253 \cs_generate_variant:Nn \__keys_prop_put:Nn { c }
4254 \cs_new_protected:Npn \__keys_undefine:
4255 {
4256   \clist_map_inline:nn
4257     { code , default , groups , inherit , type , check }
4258     {
4259       \__keys_cs_undefine:c
4260         { \tl_use:c { c__keys_ ##1 _root_str } \l_keys_path_str }
4261     }
4262 }
4263 \cs_new_protected:Npn \__keys_value_requirement:nn #1#2
4264 {
4265   \str_case:nnF {#2}
4266   {
4267     { true }
4268     {
4269       \cs_set_eq:cc
4270         { \c__keys_check_root_str \l_keys_path_str }
4271         { __keys_check_ #1 : }
4272     }
4273   { false }
4274   {
4275     \cs_if_eq:ccT
4276       { \c__keys_check_root_str \l_keys_path_str }
4277       { __keys_check_ #1 : }
4278     {
4279       \__keys_cs_undefine:c
4280         { \c__keys_check_root_str \l_keys_path_str }
4281     }
4282   }
4283 }
4284 {
4285   \msg_error:nne { keys }
4286   { boolean-values-only }
4287   { .value_ #1 :n }
4288 }
4289 }
4290 \cs_new_protected:Npn \__keys_check_forbidden:
4291 {
4292   \bool_if:NF \l__keys_no_value_bool
4293   {
4294     \msg_error:nnee { keys } { value-forbidden }
4295     \l_keys_path_str \l_keys_value_tl
```

4296	\use_none:nnn	14296
4297	}	14297
4298	}	14298
4299	\cs_new_protected:Npn __keys_check_required:	14299
4300	{	14300
4301	\bool_if:NT \l__keys_no_value_bool	14301
4302	{	14302
4303	\msg_error:nne { keys } { value-required }	14303
4304	\l_keys_path_str	14304
4305	\use_none:nnn	14305
4306	}	14306
4307	}	14307
4308	\cs_new_protected:Npn __keys_usage:n #1	14308
4309	{	14309
4310	\str_case:nnF {#1}	14310
4311	{	14311
4312	{ general }	14312
4313	{	14313
4314	__keys_usage:NN \l_keys_usage_load_prop	14314
4315	\c_false_bool	14315
4316	__keys_usage:NN \l_keys_usage_preamble_prop	14316
4317	\c_false_bool	14317
4318	}	14318
4319	{ load }	14319
4320	{	14320
4321	__keys_usage:NN \l_keys_usage_load_prop	14321
4322	\c_true_bool	14322
4323	__keys_usage:NN \l_keys_usage_preamble_prop	14323
4324	\c_false_bool	14324
4325	}	14325
4326	{ preamble }	14326
4327	{	14327
4328	__keys_usage:NN \l_keys_usage_load_prop	14328
4329	\c_false_bool	14329
4330	__keys_usage:NN \l_keys_usage_preamble_prop	14330
4331	\c_true_bool	14331
4332	}	14332
4333	}	14333
4334	{	14334
4335	\msg_error:nnnn { keys }	14335
4336	{ choice-unknown }	14336
4337	{ .usage:n }	14337
4338	{#1}	14338
4339	}	14339
4340	}	14340
4341	\cs_new_protected:Npn __keys_usage:NN #1#2	14341

```
4342 {
4343     \prop_get:NVNF #1 \l__keys_module_str \l__keys_tmpa_tl
4344     { \tl_clear:N \l__keys_tmpa_tl }
4345     \tl_set:Ne \l__keys_tmpb_tl
4346     {
4347         \exp_after:wN \exp_after:wN \exp_after:wN \__keys_usage:w \exp_after:wN
4348         \l_keys_path_str \exp_after:wN / \exp_after:wN \s__keys_stop
4349         \exp_after:wN { \l_keys_path_str }
4350     }
4351     \bool_if:NTF #2
4352     { \clist_put_right:NV \l__keys_tmpa_tl \l__keys_tmpb_tl }
4353     { \clist_remove_all:NV \l__keys_tmpa_tl \l__keys_tmpb_tl }
4354     \prop_put:NVV #1 \l__keys_module_str
4355     \l__keys_tmpa_tl
4356 }
4357 \cs_new:Npn \__keys_usage:w #1 / #2 \s__keys_stop #3
4358 {
4359     \tl_if_blank:nTF {#2}
4360     {#1}
4361     { \__keys_usage_aux:w #3 \s__keys_stop }
4362 }
4363 \cs_new:Npn \__keys_usage_aux:w #1 / #2 \s__keys_stop {#2}
4364 \cs_new_protected:Npn \__keys_variable_set:NnnN #1#2#3#4
4365 {
4366     \use:c { #2_if_exist:NF } #1 { \use:c { #2_new:N } #1 }
4367     \__keys_cmd_set:ne \l_keys_path_str
4368     {
4369         \exp_not:c { #2 _ #3 set:N #4 }
4370         \exp_not:N #1
4371         \exp_not:n { {##1} }
4372     }
4373 }
4374 \cs_generate_variant:Nn \__keys_variable_set:NnnN { c }
4375 \cs_new_protected:Npn \__keys_variable_set_required:NnnN #1#2#3#4
4376 {
4377     \__keys_variable_set:NnnN #1 {#2} {#3} #4
4378     \__keys_value_requirement:nn { required } { true }
4379 }
4380 \cs_generate_variant:Nn \__keys_variable_set_required:NnnN { c }
4381 \cs_new_protected:cpn { \c__keys_props_root_str .bool_set:N } #1
4382 { \__keys_bool_set:Nn #1 { } }
4383 \cs_new_protected:cpn { \c__keys_props_root_str .bool_set:c } #1
4384 { \__keys_bool_set:cn {#1} { } }
4385 \cs_new_protected:cpn { \c__keys_props_root_str .bool_gset:N } #1
4386 { \__keys_bool_set:Nn #1 { g } }
4387 \cs_new_protected:cpn { \c__keys_props_root_str .bool_gset:c } #1
```

```
4388 { \__keys_bool_set:cn {#1} { g } }
4389 \cs_new_protected:cpn { \c__keys_props_root_str .bool_set_inverse:N } #1
4390 { \__keys_bool_set_inverse:Nn #1 { } }
4391 \cs_new_protected:cpn { \c__keys_props_root_str .bool_set_inverse:c } #1
4392 { \__keys_bool_set_inverse:cn {#1} { } }
4393 \cs_new_protected:cpn { \c__keys_props_root_str .bool_gset_inverse:N } #1
4394 { \__keys_bool_set_inverse:Nn #1 { g } }
4395 \cs_new_protected:cpn { \c__keys_props_root_str .bool_gset_inverse:c } #1
4396 { \__keys_bool_set_inverse:cn {#1} { g } }
4397 \cs_new_protected:cpn { \c__keys_props_root_str .choice: }
4398 { \__keys_choice_make: }
4399 \cs_new_protected:cpn { \c__keys_props_root_str .choices:nn } #1
4400 { \__keys_choices_make:nn #1 }
4401 \cs_new_protected:cpn { \c__keys_props_root_str .choices:Vn } #1
4402 { \exp_args:NV \__keys_choices_make:nn #1 }
4403 \cs_new_protected:cpn { \c__keys_props_root_str .choices:en } #1
4404 { \exp_args:Ne \__keys_choices_make:nn #1 }
4405 \cs_new_protected:cpn { \c__keys_props_root_str .choices:on } #1
4406 { \exp_args:No \__keys_choices_make:nn #1 }
4407 \cs_new_protected:cpn { \c__keys_props_root_str .choices:xn } #1
4408 { \exp_args:Nx \__keys_choices_make:nn #1 }
4409 \cs_new_protected:cpn { \c__keys_props_root_str .code:n } #1
4410 { \__keys_cmd_set:nn \l_keys_path_str {#1} }
4411 \cs_new_protected:cpn { \c__keys_props_root_str .clist_set:N } #1
4412 { \__keys_variable_set:NnnN #1 { clist } { } n }
4413 \cs_new_protected:cpn { \c__keys_props_root_str .clist_set:c } #1
4414 { \__keys_variable_set:cnN {#1} { clist } { } n }
4415 \cs_new_protected:cpn { \c__keys_props_root_str .clist_gset:N } #1
4416 { \__keys_variable_set:NnnN #1 { clist } { g } n }
4417 \cs_new_protected:cpn { \c__keys_props_root_str .clist_gset:c } #1
4418 { \__keys_variable_set:cnN {#1} { clist } { g } n }
4419 \cs_new_protected:cpn { \c__keys_props_root_str .cs_set:Np } #1
4420 { \__keys_cs_set:NNpn \cs_set:Npn #1 { } }
4421 \cs_new_protected:cpn { \c__keys_props_root_str .cs_set:cp } #1
4422 { \__keys_cs_set:Ncpn \cs_set:Npn #1 { } }
4423 \cs_new_protected:cpn { \c__keys_props_root_str .cs_set_protected:Np } #1
4424 { \__keys_cs_set:NNpn \cs_set_protected:Npn #1 { } }
4425 \cs_new_protected:cpn { \c__keys_props_root_str .cs_set_protected:cp } #1
4426 { \__keys_cs_set:Ncpn \cs_set_protected:Npn #1 { } }
4427 \cs_new_protected:cpn { \c__keys_props_root_str .cs_gset:Np } #1
4428 { \__keys_cs_set:NNpn \cs_gset:Npn #1 { } }
4429 \cs_new_protected:cpn { \c__keys_props_root_str .cs_gset:cp } #1
4430 { \__keys_cs_set:Ncpn \cs_gset:Npn #1 { } }
4431 \cs_new_protected:cpn { \c__keys_props_root_str .cs_gset_protected:Np } #1
4432 { \__keys_cs_set:NNpn \cs_gset_protected:Npn #1 { } }
4433 \cs_new_protected:cpn { \c__keys_props_root_str .cs_gset_protected:cp } #1
```

```
14388
14389
14390
14391
14392
14393
14394
14395
14396
14397
14398
14399
14400
14401
14402
14403
14404
14405
14406
14407
14408
14409
14410
14411
14412
14413
14414
14415
14416
14417
14418
14419
14420
14421
14422
14423
14424
14425
14426
14427
14428
14429
14430
14431
14432
14433
```



```
4434 { \__keys_cs_set:Ncpn \cs_gset_protected:Npn #1 { } } 14434
4435 \cs_new_protected:cpn { \c__keys_props_root_str .default:n } #1 14435
4436 { \__keys_default_set:n {#1} } 14436
4437 \cs_new_protected:cpn { \c__keys_props_root_str .default:V } #1 14437
4438 { \exp_args:NV \__keys_default_set:n #1 } 14438
4439 \cs_new_protected:cpn { \c__keys_props_root_str .default:e } #1 14439
4440 { \exp_args:Ne \__keys_default_set:n {#1} } 14440
4441 \cs_new_protected:cpn { \c__keys_props_root_str .default:o } #1 14441
4442 { \exp_args:No \__keys_default_set:n {#1} } 14442
4443 \cs_new_protected:cpn { \c__keys_props_root_str .default:x } #1 14443
4444 { \exp_args:Nx \__keys_default_set:n {#1} } 14444
4445 \cs_new_protected:cpn { \c__keys_props_root_str .dim_set:N } #1 14445
4446 { \__keys_variable_set_required:NnnN #1 { dim } { } n } 14446
4447 \cs_new_protected:cpn { \c__keys_props_root_str .dim_set:c } #1 14447
4448 { \__keys_variable_set_required:cnnN {#1} { dim } { } n } 14448
4449 \cs_new_protected:cpn { \c__keys_props_root_str .dim_gset:N } #1 14449
4450 { \__keys_variable_set_required:NnnN #1 { dim } { g } n } 14450
4451 \cs_new_protected:cpn { \c__keys_props_root_str .dim_gset:c } #1 14451
4452 { \__keys_variable_set_required:cnnN {#1} { dim } { g } n } 14452
4453 \cs_new_protected:cpn { \c__keys_props_root_str .fp_set:N } #1 14453
4454 { \__keys_variable_set_required:NnnN #1 { fp } { } n } 14454
4455 \cs_new_protected:cpn { \c__keys_props_root_str .fp_set:c } #1 14455
4456 { \__keys_variable_set_required:cnnN {#1} { fp } { } n } 14456
4457 \cs_new_protected:cpn { \c__keys_props_root_str .fp_gset:N } #1 14457
4458 { \__keys_variable_set_required:NnnN #1 { fp } { g } n } 14458
4459 \cs_new_protected:cpn { \c__keys_props_root_str .fp_gset:c } #1 14459
4460 { \__keys_variable_set_required:cnnN {#1} { fp } { g } n } 14460
4461 \cs_new_protected:cpn { \c__keys_props_root_str .groups:n } #1 14461
4462 { \__keys_groups_set:n {#1} } 14462
4463 \cs_new_protected:cpn { \c__keys_props_root_str .inherit:n } #1 14463
4464 { \__keys_inherit:n {#1} } 14464
4465 \cs_new_protected:cpn { \c__keys_props_root_str .initial:n } #1 14465
4466 { \__keys_initialise:n {#1} } 14466
4467 \cs_new_protected:cpn { \c__keys_props_root_str .initial:V } #1 14467
4468 { \exp_args:NV \__keys_initialise:n #1 } 14468
4469 \cs_new_protected:cpn { \c__keys_props_root_str .initial:e } #1 14469
4470 { \exp_args:Ne \__keys_initialise:n {#1} } 14470
4471 \cs_new_protected:cpn { \c__keys_props_root_str .initial:o } #1 14471
4472 { \exp_args:No \__keys_initialise:n {#1} } 14472
4473 \cs_new_protected:cpn { \c__keys_props_root_str .initial:x } #1 14473
4474 { \exp_args:Nx \__keys_initialise:n {#1} } 14474
4475 \cs_new_protected:cpn { \c__keys_props_root_str .int_set:N } #1 14475
4476 { \__keys_variable_set_required:NnnN #1 { int } { } n } 14476
4477 \cs_new_protected:cpn { \c__keys_props_root_str .int_set:c } #1 14477
4478 { \__keys_variable_set_required:cnnN {#1} { int } { } n } 14478
4479 \cs_new_protected:cpn { \c__keys_props_root_str .int_gset:N } #1 14479
```

4480	{ __keys_variable_set_required:NnnN #1 { int } { g } n }	14480
4481	\cs_new_protected:cpn { \c__keys_props_root_str .int_gset:c } #1	14481
4482	{ __keys_variable_set_required:cnnN {#1} { int } { g } n }	14482
4483	\cs_new_protected:cpn { \c__keys_props_root_str .legacy_if_set:n } #1	14483
4484	{ __keys_legacy_if_set:nn {#1} { } }	14484
4485	\cs_new_protected:cpn { \c__keys_props_root_str .legacy_if_gset:n } #1	14485
4486	{ __keys_legacy_if_set:nn {#1} { g } }	14486
4487	\cs_new_protected:cpn { \c__keys_props_root_str .legacy_if_set_inverse:n } #1	14487
4488	{ __keys_legacy_if_set_inverse:nn {#1} { } }	14488
4489	\cs_new_protected:cpn { \c__keys_props_root_str .legacy_if_gset_inverse:n } #1	14489
4490	{ __keys_legacy_if_set_inverse:nn {#1} { g } }	14490
4491	\cs_new_protected:cpn { \c__keys_props_root_str .meta:n } #1	14491
4492	{ __keys_meta_make:n {#1} }	14492
4493	\cs_new_protected:cpn { \c__keys_props_root_str .meta:nn } #1	14493
4494	{ __keys_meta_make:nn #1 }	14494
4495	\cs_new_protected:cpn { \c__keys_props_root_str .multichoice: }	14495
4496	{ __keys_multichoice_make: }	14496
4497	\cs_new_protected:cpn { \c__keys_props_root_str .multichoices:nn } #1	14497
4498	{ __keys_multichoices_make:nn #1 }	14498
4499	\cs_new_protected:cpn { \c__keys_props_root_str .multichoices:Vn } #1	14499
4500	{ \exp_args:NV __keys_multichoices_make:nn #1 }	14500
4501	\cs_new_protected:cpn { \c__keys_props_root_str .multichoices:en } #1	14501
4502	{ \exp_args:Ne __keys_multichoices_make:nn #1 }	14502
4503	\cs_new_protected:cpn { \c__keys_props_root_str .multichoices:on } #1	14503
4504	{ \exp_args:No __keys_multichoices_make:nn #1 }	14504
4505	\cs_new_protected:cpn { \c__keys_props_root_str .multichoices:xn } #1	14505
4506	{ \exp_args:Nx __keys_multichoices_make:nn #1 }	14506
4507	\cs_new_protected:cpn { \c__keys_props_root_str .muskip_set:N } #1	14507
4508	{ __keys_variable_set_required:NnnN #1 { muskip } { } n }	14508
4509	\cs_new_protected:cpn { \c__keys_props_root_str .muskip_set:c } #1	14509
4510	{ __keys_variable_set_required:cnnN {#1} { muskip } { } n }	14510
4511	\cs_new_protected:cpn { \c__keys_props_root_str .muskip_gset:N } #1	14511
4512	{ __keys_variable_set_required:NnnN #1 { muskip } { g } n }	14512
4513	\cs_new_protected:cpn { \c__keys_props_root_str .muskip_gset:c } #1	14513
4514	{ __keys_variable_set_required:cnnN {#1} { muskip } { g } n }	14514
4515	\cs_new_protected:cpn { \c__keys_props_root_str .prop_put:N } #1	14515
4516	{ __keys_prop_put:Nn #1 { } }	14516
4517	\cs_new_protected:cpn { \c__keys_props_root_str .prop_put:c } #1	14517
4518	{ __keys_prop_put:cn {#1} { } }	14518
4519	\cs_new_protected:cpn { \c__keys_props_root_str .prop_gput:N } #1	14519
4520	{ __keys_prop_put:Nn #1 { g } }	14520
4521	\cs_new_protected:cpn { \c__keys_props_root_str .prop_gput:c } #1	14521
4522	{ __keys_prop_put:cn {#1} { g } }	14522
4523	\cs_new_protected:cpn { \c__keys_props_root_str .skip_set:N } #1	14523
4524	{ __keys_variable_set_required:NnnN #1 { skip } { } n }	14524
4525	\cs_new_protected:cpn { \c__keys_props_root_str .skip_set:c } #1	14525

4526	{ __keys_variable_set_required:cnN {#1} { skip } { } n }	1452
4527	\cs_new_protected:cpn { \c__keys_props_root_str .skip_gset:N } #1	1452
4528	{ __keys_variable_set_required:NnnN #1 { skip } { g } n }	1452
4529	\cs_new_protected:cpn { \c__keys_props_root_str .skip_gset:c } #1	1452
4530	{ __keys_variable_set_required:cnN {#1} { skip } { g } n }	1453
4531	\cs_new_protected:cpn { \c__keys_props_root_str .str_set:N } #1	1453
4532	{ __keys_variable_set:NnnN #1 { str } { } n }	1453
4533	\cs_new_protected:cpn { \c__keys_props_root_str .str_set:c } #1	1453
4534	{ __keys_variable_set:cnN {#1} { str } { } n }	1453
4535	\cs_new_protected:cpn { \c__keys_props_root_str .str_set_e:N } #1	1453
4536	{ __keys_variable_set:NnnN #1 { str } { } e }	1453
4537	\cs_new_protected:cpn { \c__keys_props_root_str .str_set_e:c } #1	1453
4538	{ __keys_variable_set:cnN {#1} { str } { } e }	1453
4539	\cs_new_protected:cpn { \c__keys_props_root_str .str_gset:N } #1	1453
4540	{ __keys_variable_set:NnnN #1 { str } { g } n }	1454
4541	\cs_new_protected:cpn { \c__keys_props_root_str .str_gset:c } #1	1454
4542	{ __keys_variable_set:cnN {#1} { str } { g } n }	1454
4543	\cs_new_protected:cpn { \c__keys_props_root_str .str_gset_e:N } #1	1454
4544	{ __keys_variable_set:NnnN #1 { str } { g } e }	1454
4545	\cs_new_protected:cpn { \c__keys_props_root_str .str_gset_e:c } #1	1454
4546	{ __keys_variable_set:cnN {#1} { str } { g } e }	1454
4547	\cs_new_protected:cpn { \c__keys_props_root_str .tl_set:N } #1	1454
4548	{ __keys_variable_set:NnnN #1 { tl } { } n }	1454
4549	\cs_new_protected:cpn { \c__keys_props_root_str .tl_set:c } #1	1454
4550	{ __keys_variable_set:cnN {#1} { tl } { } n }	1455
4551	\cs_new_protected:cpn { \c__keys_props_root_str .tl_set_e:N } #1	1455
4552	{ __keys_variable_set:NnnN #1 { tl } { } e }	1455
4553	\cs_new_protected:cpn { \c__keys_props_root_str .tl_set_e:c } #1	1455
4554	{ __keys_variable_set:cnN {#1} { tl } { } e }	1455
4555	\cs_new_protected:cpn { \c__keys_props_root_str .tl_gset:N } #1	1455
4556	{ __keys_variable_set:NnnN #1 { tl } { g } n }	1455
4557	\cs_new_protected:cpn { \c__keys_props_root_str .tl_gset:c } #1	1455
4558	{ __keys_variable_set:cnN {#1} { tl } { g } n }	1455
4559	\cs_new_protected:cpn { \c__keys_props_root_str .tl_gset_e:N } #1	1455
4560	{ __keys_variable_set:NnnN #1 { tl } { g } e }	1456
4561	\cs_new_protected:cpn { \c__keys_props_root_str .tl_gset_e:c } #1	1456
4562	{ __keys_variable_set:cnN {#1} { tl } { g } e }	1456
4563	\cs_new_protected:cpn { \c__keys_props_root_str .undefine: }	1456
4564	{ __keys_undefine: }	1456
4565	\cs_new_protected:cpn { \c__keys_props_root_str .usage:n } #1	1456
4566	{ __keys_usage:n {#1} }	1456
4567	\cs_new_protected:cpn { \c__keys_props_root_str .value_forbidden:n } #1	1456
4568	{ __keys_value_requirement:nn { forbidden } {#1} }	1456
4569	\cs_new_protected:cpn { \c__keys_props_root_str .value_required:n } #1	1456
4570	{ __keys_value_requirement:nn { required } {#1} }	1457
4571	\cs_new_protected:Npn __keys_set:nnnnNn #1#2#3#4#5#6	1457

```

4572 {
4573     \use:e
4574     {
4575         \exp_not:n
4576         {
4577             \clist_clear:N \l__keys_unused_clist
4578             \clist_set:Ne \l__keys_selective_clist { \tl_to_str:n {#2} }
4579             \tl_set:Nn \l__keys_relative_tl {#4}
4580             #6
4581             \__keys_set:nn {#1} {#3}
4582             \clist_set_eq:NN #5 \l__keys_unused_clist
4583         }
4584         \__keys_reset_bool:N \l__keys_only_known_bool
4585         \__keys_reset_bool:N \l__keys_exclude_bool
4586         \__keys_reset_bool:N \l__keys_selective_bool
4587         \__keys_reset_var:N \l__keys_unused_clist
4588         \__keys_reset_var:N \l__keys_selective_clist
4589         \__keys_reset_var:N \l__keys_relative_tl
4590         \__keys_reset_var:N \l__keys_inherit_str
4591         \__keys_reset_var:N \l_keys_choice_tl
4592         \__keys_reset_var:N \l_keys_key_tl
4593         \__keys_reset_var:N \l_keys_key_str
4594         \__keys_reset_var:N \l_keys_path_tl
4595         \__keys_reset_var:N \l_keys_path_str
4596         \__keys_reset_var:N \l_keys_value_tl
4597         \int_set:Nn \l_keys_choice_int { \int_use:N \l_keys_choice_int }
4598     }
4599 }
4600 \cs_new:Npn \__keys_reset_bool:N #1
4601 {
4602     \exp_not:c
4603     { bool_set_ \bool_if:NTF #1 { true } { false } :N }
4604     \exp_not:N #1
4605 }
4606 \cs_new:Npn \__keys_reset_var:N #1
4607 {
4608     \exp_not:n
4609     { \__kernel_tl_set:Nx #1 }
4610     { \exp_not:N \exp_not:n { \exp_not:o { #1 } } }
4611 }
4612 \cs_new_protected:Npn \__keys_set:nn #1#2
4613 { \exp_args:No \__keys_set:nnn \l__keys_module_str {#1} {#2} }
4614 \cs_new_protected:Npn \__keys_set:nnn #1#2#3
4615 {
4616     \str_set:Ne \l__keys_module_str { \__keys_trim_spaces:n {#2} }
4617     \keyval_parse:NNn \__keys_set_keyval:n \__keys_set_keyval:nn {#3}

```

```
4618 \str_set:Nn \l__keys_module_str {#1}
4619 }
4620 \cs_new_protected:Npn \keys_set:nn #1#2
4621 {
4622   \__keys_set:nnnnNn
4623     {#1} { } {#2} { \q_keys_no_value } \l__keys_tmp_clist
4624   {
4625     \bool_set_false:N \l__keys_only_known_bool
4626     \bool_set_false:N \l__keys_exclude_bool
4627     \bool_set_false:N \l__keys_selective_bool
4628   }
4629 }
4630 \cs_generate_variant:Nn \keys_set:nn { nV , nv , ne , no , nx }
4631 \cs_new_protected:Npn \keys_set_known:nnnN #1#2#3#4
4632 {
4633   \__keys_set:nnnnNn
4634     {#1} { } {#2} {#3} #4
4635   {
4636     \bool_set_true:N \l__keys_only_known_bool
4637     \bool_set_false:N \l__keys_exclude_bool
4638     \bool_set_false:N \l__keys_selective_bool
4639   }
4640 }
4641 \cs_generate_variant:Nn \keys_set_known:nnnN { nV , nv , ne , no }
4642 \cs_new_protected:Npn \keys_set_known:nnN #1#2#3
4643   { \keys_set_known:nnnN {#1} {#2} { \q_keys_no_value } #3 }
4644 \cs_generate_variant:Nn \keys_set_known:nnN { nV , nv , ne , no }
4645 \cs_new_protected:Npn \keys_set_known:nn #1#2
4646   { \keys_set_known:nnnN {#1} {#2} { \q_keys_no_value } \l__keys_tmp_clist }
4647 \cs_generate_variant:Nn \keys_set_known:nn { nV , nv , ne , no }
4648 \cs_new_protected:Npn \keys_set_exclude_groups:nnnnN #1#2#3#4#5
4649 {
4650   \__keys_set:nnnnNn
4651     {#1} {#2} {#3} {#4} #5
4652   {
4653     \bool_set_false:N \l__keys_only_known_bool
4654     \bool_set_true:N \l__keys_exclude_bool
4655     \bool_set_true:N \l__keys_selective_bool
4656   }
4657 }
4658 \cs_generate_variant:Nn \keys_set_exclude_groups:nnnnN { nnV , nnv , nno }
4659 \cs_new_protected:Npn \keys_set_exclude_groups:nnnN #1#2#3#4
4660   { \keys_set_exclude_groups:nnnnN {#1} {#2} {#3} { \q_keys_no_value } #4 }
4661 \cs_generate_variant:Nn \keys_set_exclude_groups:nnnN { nnV , nnv , nno }
4662 \cs_new_protected:Npn \keys_set_exclude_groups:nnn #1#2#3
4663 {
```



```
4664 \keys_set_exclude_groups:nnnnN {#1} {#2} {#3}
4665 { \q__keys_no_value } \l__keys_tmp_clist
4666 }
4667 \cs_generate_variant:Nn \keys_set_exclude_groups:nnn { nnV , nnv , nno }
4668 \cs_new_protected:Npn \keys_set_groups:nnnnN #1#2#3#4#5
4669 {
4670   \__keys_set:nnnnNn
4671     {#1} {#2} {#3} {#4} #5
4672   {
4673     \bool_set_false:N \l__keys_only_known_bool
4674     \bool_set_false:N \l__keys_exclude_bool
4675     \bool_set_true:N \l__keys_selective_bool
4676   }
4677 }
4678 \cs_generate_variant:Nn \keys_set_groups:nnnnN { nnV , nnv , nno }
4679 \cs_new_protected:Npn \keys_set_groups:nnnnN #1#2#3#4
4680 { \keys_set_groups:nnnnN {#1} {#2} {#3} { \q__keys_no_value } #4 }
4681 \cs_generate_variant:Nn \keys_set_groups:nnnnN { nnV , nnv , nno }
4682 \cs_new_protected:Npn \keys_set_groups:nnn #1#2#3
4683 {
4684   \keys_set_groups:nnnnN {#1} {#2} {#3}
4685   { \q__keys_no_value } \l__keys_tmp_clist
4686 }
4687 \cs_generate_variant:Nn \keys_set_groups:nnn { nnV , nnv , nno }
4688 \cs_new_protected:Npn \keys_precompile:nnN #1#2#3
4689 {
4690   \bool_set_true:N \l__keys_precompile_bool
4691   \tl_clear:N \l__keys_precompile_tl
4692   \keys_set:nn {#1} {#2}
4693   \bool_set_false:N \l__keys_precompile_bool
4694   \tl_set_eq:NN #3 \l__keys_precompile_tl
4695 }
4696 \cs_new_protected:Npn \__keys_set_keyval:n #1
4697 {
4698   \bool_set_true:N \l__keys_no_value_bool
4699   \__keys_set_keyval:onn \l__keys_module_str {#1} { }
4700 }
4701 \cs_new_protected:Npn \__keys_set_keyval:nn #1#2
4702 {
4703   \bool_set_false:N \l__keys_no_value_bool
4704   \__keys_set_keyval:onn \l__keys_module_str {#1} {#2}
4705 }
4706 \cs_new_protected:Npn \__keys_set_keyval:nnn #1#2#3
4707 {
4708   \__kernel_tl_set:Nx \l_keys_path_str
4709   {
```

```
4710     \tl_if_blank:nF {#1}
4711     { #1 / }
4712     \__keys_trim_spaces:n {#2}
4713 }
4714 \str_clear:N \l__keys_module_str
4715 \str_clear:N \l__keys_inherit_str
4716 \exp_after:wN \__keys_find_key_module:wNN \l_keys_path_str \s__keys_stop
4717     \l__keys_module_str \l_keys_key_str
4718 \tl_set_eq:NN \l_keys_key_tl \l_keys_key_str
4719 \__keys_value_or_default:n {#3}
4720 \bool_if:NTF \l__keys_selective_bool
4721     \__keys_set_selective:
4722     \__keys_execute:
4723 \str_set:Nn \l__keys_module_str {#1}
4724 }
4725 \cs_generate_variant:Nn \__keys_set_keyval:nnn { o }
4726 \cs_new_protected:Npn \__keys_find_key_module:wNN #1 \s__keys_stop #2 #3
4727 {
4728     \__keys_find_key_module_auxi:Nw #2 #1 \s__keys_nil \__keys_find_key_module_auxii:Nw
4729     / \s__keys_nil \__keys_find_key_module_auxiv:Nw #3
4730 }
4731 \cs_new_protected:Npn \__keys_find_key_module_auxi:Nw #1 #2 / #3 \s__keys_nil #4
4732 {
4733     #4 #1 #2 \s__keys_mark #3 \s__keys_nil #4
4734 }
4735 \cs_new_protected:Npn \__keys_find_key_module_auxii:Nw
4736     #1 #2 \s__keys_mark #3 \s__keys_nil \__keys_find_key_module_auxiii:Nw
4737 {
4738     \cs_set_nopar:Npe #1 { \tl_if_empty:NF #1 { #1 / } #2 }
4739     \__keys_find_key_module_auxi:Nw #1 #3 \s__keys_nil \__keys_find_key_module_auxiii:Nw
4740 }
4741 \cs_new_protected:Npn \__keys_find_key_module_auxiii:Nw #1 #2 \s__keys_mark
4742 {
4743     \cs_set_nopar:Npe #1 { \tl_if_empty:NF #1 { #1 / } #2 }
4744     \__keys_find_key_module_auxi:Nw #1
4745 }
4746 \cs_new_protected:Npn \__keys_find_key_module_auxiv:Nw
4747     #1 #2 \s__keys_nil #3 \s__keys_mark
4748     \s__keys_nil \__keys_find_key_module_auxiv:Nw #4
4749 {
4750     \cs_set_nopar:Npn #4 { #2 }
4751 }
4752 \cs_new_protected:Npn \__keys_set_selective:
4753 {
4754     \cs_if_exist:cTF { \c__keys_groups_root_str \l_keys_path_str }
4755     {
```



```
4756         \clist_set_eq:Nc \l__keys_groups_clist
4757         { \c__keys_groups_root_str \l_keys_path_str }
4758     \__keys_check_groups:
4759 }
4760 {
4761     \bool_if:NTF \l__keys_exclude_bool
4762     \__keys_execute:
4763     \__keys_store_unused:
4764 }
4765 }
4766 \cs_new_protected:Npn \__keys_check_groups:
4767 {
4768     \bool_set_false:N \l__keys_tmp_bool
4769     \clist_map_inline:Nn \l__keys_selective_clist
4770     {
4771         \clist_if_in:NnT \l__keys_groups_clist {##1}
4772         {
4773             \bool_set_true:N \l__keys_tmp_bool
4774             \clist_map_break:
4775         }
4776     }
4777     \bool_if:NTF \l__keys_tmp_bool
4778     {
4779         \bool_if:NTF \l__keys_exclude_bool
4780         \__keys_store_unused:
4781         \__keys_execute:
4782     }
4783     {
4784         \bool_if:NTF \l__keys_exclude_bool
4785         \__keys_execute:
4786         \__keys_store_unused:
4787     }
4788 }
4789 \cs_new_protected:Npn \__keys_value_or_default:n #1
4790 {
4791     \bool_if:NTF \l__keys_no_value_bool
4792     {
4793         \cs_if_exist:cTF { \c__keys_default_root_str \l_keys_path_str }
4794         {
4795             \tl_set_eq:Nc
4796             \l_keys_value_tl
4797             { \c__keys_default_root_str \l_keys_path_str }
4798         }
4799         {
4800             \tl_clear:N \l_keys_value_tl
4801             \cs_if_exist:cT
```

```
4802         { \c__keys_inherit_root_str \__keys_parent:o \l_keys_path_str }
4803         { \__keys_default_inherit: }
4804     }
4805 }
4806 { \tl_set:Nn \l_keys_value_tl {#1} }
4807 }
4808 \cs_new_protected:Npn \__keys_default_inherit:
4809 {
4810     \clist_map_inline:cn
4811     { \c__keys_inherit_root_str \__keys_parent:o \l_keys_path_str }
4812     {
4813         \cs_if_exist:cT
4814         { \c__keys_default_root_str ##1 / \l_keys_key_str }
4815         {
4816             \tl_set_eq:Nc
4817             \l_keys_value_tl
4818             { \c__keys_default_root_str ##1 / \l_keys_key_str }
4819             \clist_map_break:
4820         }
4821     }
4822 }
4823 \cs_new_protected:Npn \__keys_execute:
4824 {
4825     \cs_if_exist:cTF { \c__keys_code_root_str \l_keys_path_str }
4826     {
4827         \cs_if_exist_use:c { \c__keys_check_root_str \l_keys_path_str }
4828         \__keys_execute:no \l_keys_path_str \l_keys_value_tl
4829     }
4830     {
4831         \cs_if_exist:cTF
4832         { \c__keys_inherit_root_str \__keys_parent:o \l_keys_path_str }
4833         { \__keys_execute_inherit: }
4834         { \__keys_execute_unknown: }
4835     }
4836 }
4837 \cs_new_protected:Npn \__keys_execute_inherit:
4838 {
4839     \bool_set_false:N \l__keys_inherit_bool
4840     \clist_map_inline:cn
4841     { \c__keys_inherit_root_str \__keys_parent:o \l_keys_path_str }
4842     { \__keys_excute_inherit:n {##1} }
4843     \bool_if:NF \l__keys_inherit_bool
4844     { \__keys_execute_unknown: }
4845 }
4846 \cs_new_protected:Npn \__keys_excute_inherit:n #1
4847 {
```

```
4848 \cs_if_exist:cTF
4849 { \c__keys_inherit_root_str #1 }
4850 {
4851   \clist_map_inline:cn { \c__keys_inherit_root_str #1 }
4852     { \__keys_excute_inherit:n {##1} }
4853   \bool_if:NT \l__keys_inherit_bool
4854     { \clist_map_break: }
4855 }
4856 {
4857   \cs_if_exist:cT
4858     { \c__keys_code_root_str #1 / \l_keys_key_str }
4859     {
4860       \str_set:Nn \l__keys_inherit_str {#1}
4861       \cs_if_exist_use:c { \c__keys_check_root_str #1 / \l_keys_key_str }
4862       \__keys_execute:no { #1 / \l_keys_key_str } \l_keys_value_tl
4863       \clist_map_break:n
4864         { \bool_set_true:N \l__keys_inherit_bool }
4865     }
4866 }
4867 }
4868 \cs_new_protected:Npn \__keys_execute_unknown:
4869 {
4870   \bool_if:NTF \l__keys_only_known_bool
4871     { \__keys_store_unused: }
4872     {
4873       \cs_if_exist:cTF
4874         { \c__keys_code_root_str \l__keys_module_str / unknown }
4875         {
4876           \bool_if:NT \l__keys_no_value_bool
4877             {
4878               \cs_if_exist:cT
4879                 { \c__keys_default_root_str \l__keys_module_str / unknown }
4880                 {
4881                   \tl_set_eq:Nc
4882                     \l_keys_value_tl
4883                     { \c__keys_default_root_str \l__keys_module_str / unknown }
4884                 }
4885             }
4886           \__keys_execute:no { \l__keys_module_str / unknown } \l_keys_value_tl
4887         }
4888     }
4889     \msg_error:nnee { keys } { unknown }
4890     \l_keys_path_str \l__keys_module_str
4891 }
4892 }
4893 }
```

```
4894 \cs_new:Npn \__keys_execute:nn #1#2 14894
4895 { \__keys_execute:no {#1} { \prg_do_nothing: #2 } } 14895
4896 \cs_new:Npn \__keys_execute:no #1#2 14896
4897 { 14897
4898   \exp_args:NNo \exp_args:No \use:n 14898
4899   { 14899
4900     \cs:w \c__keys_code_root_str #1 \exp_after:wN \cs_end: 14900
4901     \exp_after:wN {#2} 14901
4902   } 14902
4903 } 14903
4904 \cs_new_protected:Npn \__keys_store_unused: 14904
4905 { 14905
4906   \__keys_quark_if_no_value:NTF \l__keys_relative_tl 14906
4907   { 14907
4908     \clist_put_right:Ne \l__keys_unused_clist 14908
4909     { 14909
4910       \l_keys_key_str 14910
4911       \bool_if:NF \l__keys_no_value_bool 14911
4912       { = { \exp_not:o \l_keys_value_tl } } 14912
4913     } 14913
4914   } 14914
4915   { 14915
4916     \tl_if_empty:NTF \l__keys_relative_tl 14916
4917     { 14917
4918       \clist_put_right:Ne \l__keys_unused_clist 14918
4919       { 14919
4920         \l_keys_path_str 14920
4921         \bool_if:NF \l__keys_no_value_bool 14921
4922         { = { \exp_not:o \l_keys_value_tl } } 14922
4923       } 14923
4924     } 14924
4925     { \__keys_store_unused_aux: } 14925
4926   } 14926
4927 } 14927
4928 \cs_new_protected:Npn \__keys_store_unused_aux: 14928
4929 { 14929
4930   \__kernel_tl_set:Nx \l__keys_relative_tl 14930
4931   { \exp_args:No \__keys_trim_spaces:n \l__keys_relative_tl } 14931
4932   \use:e 14932
4933   { 14933
4934     \cs_set_protected:Npn \__keys_store_unused:w 14934
4935     ##1 \l__keys_relative_tl / 14935
4936     ##2 \l__keys_relative_tl / 14936
4937     ##3 \s__keys_stop 14937
4938   } 14938
4939   { 14939
```

```
4940         \tl_if_blank:nF {##1}
4941     {
4942         \msg_error:nnee { keys } { bad-relative-key-path }
4943         \l_keys_path_str
4944         \l__keys_relative_tl
4945     }
4946     \clist_put_right:Ne \l__keys_unused_clist
4947     {
4948         \exp_not:n {##2}
4949         \bool_if:NF \l__keys_no_value_bool
4950         { = { \exp_not:o \l_keys_value_tl } }
4951     }
4952 }
4953 \use:e
4954 {
4955     \__keys_store_unused:w \l_keys_path_str
4956     \l_keys_relative_tl / \l_keys_relative_tl /
4957     \s_keys_stop
4958 }
4959 }
4960 \cs_new_protected:Npn \__keys_store_unused:w { }
4961 \cs_new:Npn \__keys_choice_find:n #1
4962 {
4963     \str_if_empty:NTF \l__keys_inherit_str
4964     { \__keys_choice_find:nn \l_keys_path_str {#1} }
4965     {
4966         \__keys_choice_find:nn
4967         { \l_keys_inherit_str / \l_keys_key_str } {#1}
4968     }
4969 }
4970 \cs_new:Npn \__keys_choice_find:nn #1#2
4971 {
4972     \cs_if_exist:cTF { \c__keys_code_root_str #1 / \__keys_trim_spaces:n {#2} }
4973     { \__keys_execute:nn { #1 / \__keys_trim_spaces:n {#2} } {#2} }
4974     { \__keys_execute:nn { #1 / unknown } {#2} }
4975 }
4976 \cs_new:Npn \__keys_multichoice_find:n #1
4977 { \clist_map_function:nN {#1} \__keys_choice_find:n }
4978 \cs_new:Npn \__keys_parent:o #1
4979 {
4980     \exp_after:wN \__keys_parent_auxi:w #1 \q_nil \__keys_parent_auxii:w
4981     / \q_nil \__keys_parent_auxiv:w
4982 }
4983 \cs_new:Npn \__keys_parent_auxi:w #1 / #2 \q_nil #3
4984 {
4985     #3 { #1 } #2 \q_nil #3
```

4986	}	14986
4987	\cs_new:Npn __keys_parent_auxii:w #1 #2 \q_nil __keys_parent_auxii:w	14987
4988	{	14988
4989	#1 __keys_parent_auxi:w #2 \q_nil __keys_parent_auxiii:n	14989
4990	}	14990
4991	\cs_new:Npn __keys_parent_auxiii:n #1	14991
4992	{	14992
4993	/ #1 __keys_parent_auxi:w	14993
4994	}	14994
4995	\cs_new:Npn __keys_parent_auxiv:w #1 \q_nil __keys_parent_auxiv:w	14995
4996	{	14996
4997	}	14997
4998	\group_begin:	14998
4999	\cs_set:Npn __keys_tmp:w #1	14999
5000	{	15000
5001	\cs_new:Npn __keys_trim_spaces:n ##1	15001
5002	{	15002
5003	\exp_after:wN __keys_trim_spaces_auxi:w \tl_to_str:n { / ##1 } /	15003
5004	\s_keys_nil __keys_trim_spaces_auxi:w	15004
5005	\s_keys_mark __keys_trim_spaces_auxii:w	15005
5006	#1 / #1	15006
5007	\s_keys_nil __keys_trim_spaces_auxii:w	15007
5008	\s_keys_mark __keys_trim_spaces_auxiii:w	15008
5009	}	15009
5010	}	15010
5011	__keys_tmp:w { ~ }	15011
5012	\group_end:	15012
5013	\cs_new:Npn __keys_trim_spaces_auxi:w #1 ~ / #2 \s_keys_nil #3	15013
5014	{	15014
5015	#3 #1 / #2 \s_keys_nil #3	15015
5016	}	15016
5017	\cs_new:Npn __keys_trim_spaces_auxii:w #1 / ~ #2 \s_keys_mark #3	15017
5018	{	15018
5019	#3 #1 / #2 \s_keys_mark #3	15019
5020	}	15020
5021	\cs_new:Npn __keys_trim_spaces_auxiii:w	15021
5022	/ #1 /	15022
5023	\s_keys_nil __keys_trim_spaces_auxi:w	15023
5024	\s_keys_mark __keys_trim_spaces_auxii:w	15024
5025	/	15025
5026	\s_keys_nil __keys_trim_spaces_auxii:w	15026
5027	\s_keys_mark __keys_trim_spaces_auxiii:w	15027
5028	{	15028
5029	#1	15029
5030	}	15030
5031	\prg_new_conditional:Npnn \keys_if_exist:nn #1#2 { p , T , F , TF }	15031

5032	{	15032
5033	_keys_if_exist:ee	15033
5034	{ _keys_trim_spaces:n {#1} }	15034
5035	{ _keys_trim_spaces:n {#2} }	15035
5036	}	15036
5037	\prg_generate_conditional_variant:Nnn \keys_if_exist:nn { ne } { p , T , F , TF }	15037
5038	\cs_new:Npn _keys_if_exist:nn #1#2	15038
5039	{	15039
5040	_cs_if_exist:cTF	15040
5041	{ \c__keys_code_root_str #1 \tl_if_blank:nF {#1} { / } #2 }	15041
5042	{ \prg_return_true: }	15042
5043	{ \prg_return_false: }	15043
5044	}	15044
5045	\cs_generate_variant:Nn _keys_if_exist:nn { ee }	15045
5046	\prg_new_conditional:Npnn \keys_if_choice_exist:nnn #1#2#3	15046
5047	{ p , T , F , TF }	15047
5048	{	15048
5049	_keys_if_exist:eee	15049
5050	{ _keys_trim_spaces:n {#1} }	15050
5051	{ _keys_trim_spaces:n {#2} }	15051
5052	{ _keys_trim_spaces:n {#3} }	15052
5053	}	15053
5054	\cs_new:Npn _keys_if_exist:nnn #1#2#3	15054
5055	{	15055
5056	_cs_if_exist:cTF	15056
5057	{	15057
5058	\c__keys_code_root_str	15058
5059	#1 \tl_if_blank:nF {#1} { / }	15059
5060	#2 \tl_if_blank:nF {#2} { / }	15060
5061	#3	15061
5062	}	15062
5063	{ \prg_return_true: }	15063
5064	{ \prg_return_false: }	15064
5065	}	15065
5066	\cs_generate_variant:Nn _keys_if_exist:nnn { eee }	15066
5067	\cs_new_protected:Npn \keys_show:nn	15067
5068	{ _keys_show:Nnn \msg_show:nneeee }	15068
5069	\cs_new_protected:Npn \keys_log:nn	15069
5070	{ _keys_show:Nnn \msg_log:nneeee }	15070
5071	\cs_new_protected:Npn _keys_show:Nnn #1#2#3	15071
5072	{	15072
5073	_keys_show_aux:Nee	15073
5074	#1	15074
5075	{ _keys_trim_spaces:n {#2} }	15075
5076	{ _keys_trim_spaces:n {#3} }	15076
5077	}	15077


```
5078 \cs_new_protected:Npn \__keys_show_aux:Nnn #1#2#3
5079 {
5080   #1 { keys } { show-key }
5081   { #2 \tl_if_blank:nF {#2} { / } #3 }
5082   {
5083     \keys_if_exist:nnT {#2} {#3}
5084     {
5085       \exp_args:Nnf \msg_show_item_unbraced:nn { code }
5086       {
5087         \exp_args:Ne \__keys_show:n
5088         {
5089           \exp_args:Nc \cs_replacement_spec:N
5090           {
5091             \c__keys_code_root_str
5092             #2 \tl_if_blank:nF {#2} { / } #3
5093           }
5094         }
5095       }
5096     }
5097   }
5098   { } { }
5099 }
5100 \cs_generate_variant:Nn \__keys_show_aux:Nnn { Nee }
5101 \cs_new:Npe \__keys_show:n #1
5102 {
5103   \exp_not:N \__keys_show:w
5104   #1
5105   \tl_to_str:n { \__keys_precompile:n }
5106   #1
5107   \tl_to_str:n { \__keys_precompile:n }
5108   \exp_not:N \s__keys_stop
5109 }
5110 \use:e
5111 {
5112   \cs_new:Npn \exp_not:N \__keys_show:w
5113     #1 \tl_to_str:n { \__keys_precompile:n }
5114     #2 \tl_to_str:n { \__keys_precompile:n }
5115     #3 \exp_not:N \s__keys_stop
5116 }
5117 {
5118   \tl_if_blank:nTF {#2}
5119     {#1}
5120     { \__keys_show:Nw #2 \s__keys_stop }
5121 }
5122 \use:e
5123 {
```

```
5124 \cs_new:Npn \exp_not:N \__keys_show:Nw #1#2 15124
5125 \c_right_brace_str \exp_not:N \s__keys_stop 15125
5126 } 15126
5127 {#2} 15127
5128 \msg_new:nnnn { keys } { bad-relative-key-path } 15128
5129 { The~key~'#1'~is~not~inside~the~'#2'~path. } 15129
5130 { The~key~'#1'~cannot~be~expressed~relative~to~path~'#2'. } 15130
5131 \msg_new:nnnn { keys } { boolean-values-only } 15131
5132 { Key~'#1'~accepts~boolean~values~only. } 15132
5133 { The~key~'#1'~only~accepts~the~values~'true'~and~'false'. } 15133
5134 \msg_new:nnnn { keys } { choice-unknown } 15134
5135 { Key~'#1'~accepts~only~a~fixed~set~of~choices. } 15135
5136 { 15136
5137 The~key~'#1'~only~accepts~predefined~values,~ 15137
5138 and~'#2'~is~not~one~of~these. 15138
5139 } 15139
5140 \msg_new:nnnn { keys } { unknown } 15140
5141 { The~key~'#1'~is~unknown~and~is~being~ignored. } 15141
5142 { 15142
5143 The~module~'#2'~does~not~have~a~key~called~'#1'.\\ 15143
5144 Check~that~you~have~spelled~the~key~name~correctly. 15144
5145 } 15145
5146 \msg_new:nnnn { keys } { nested-choice-key } 15146
5147 { Attempt~to~define~'#1'~as~a~nested~choice~key. } 15147
5148 { 15148
5149 The~key~'#1'~cannot~be~defined~as~a~choice~as~the~parent~key~'#2'~is~ 15149
5150 itself~a~choice. 15150
5151 } 15151
5152 \msg_new:nnnn { keys } { value-forbidden } 15152
5153 { The~key~'#1'~does~not~take~a~value. } 15153
5154 { 15154
5155 The~key~'#1'~should~be~given~without~a~value.\\ 15155
5156 The~value~'#2'~was~present:~the~key~will~be~ignored. 15156
5157 } 15157
5158 \msg_new:nnnn { keys } { value-required } 15158
5159 { The~key~'#1'~requires~a~value. } 15159
5160 { 15160
5161 The~key~'#1'~must~have~a~value.\\ 15161
5162 No~value~was~present:~the~key~will~be~ignored. 15162
5163 } 15163
5164 \msg_new:nnn { keys } { show-key } 15164
5165 { 15165
5166 The~key~#1~ 15166
5167 \tl_if_empty:nTF {#2} 15167
5168 { is~undefined. } 15168
5169 { has~the~properties: #2 . } 15169
```

```

5170 }
5171 \prop_gput:Nnn \g_msg_module_name_prop { keys } { LaTeX }
5172 \prop_gput:Nnn \g_msg_module_type_prop { keys } { }
5173 %% File: l3intarray.dtx
5174 \msg_new:nnn { kernel } { negative-array-size }
5175 { Size-of-array-may-not-be-negative:~#1 }
5176 \cs_new_eq:NN \__intarray_sep: \__kernel_int_sep:
5177 \int_new:N \l__intarray_loop_int
5178 \cs_if_exist:NTF \__intarray_gset_count:Nw
5179 {
5180 \int_new:N \g__intarray_table_int
5181 \int_new:N \l__intarray_bad_index_int
5182 \cs_new_protected:Npn \__intarray_new:N #1
5183 {
5184 \__kernel_chk_if_free_cs:N #1
5185 \int_gincr:N \g__intarray_table_int
5186 \cs_gset_nopar:Npe #1 { \__intarray:w \int_use:N \g__intarray_table_int }
5187 }
5188 \cs_new_protected:Npn \intarray_new:Nn #1#2
5189 {
5190 \__intarray_new:N #1
5191 \__intarray_gset_count:Nw #1 \int_eval:n {#2} \scan_stop:
5192 \int_compare:nNnT { \intarray_count:N #1 } < 0
5193 {
5194 \msg_error:nne { kernel } { negative-array-size }
5195 { \intarray_count:N #1 }
5196 }
5197 }
5198 \cs_generate_variant:Nn \intarray_new:Nn { c }
5199 \cs_generate_variant:Nn \intarray_count:N { c }
5200 \cs_new_protected:Npn \__kernel_intarray_gset:Nnn #1#2#3
5201 { \__intarray_gset:w #2 #1 #3 \scan_stop: }
5202 \cs_new_protected:Npn \intarray_gset:Nnn #1#2#3
5203 {
5204 \__intarray_gset:wF \int_eval:n {#2} #1 \int_eval:n{#3}
5205 {
5206 \msg_error:nneee { kernel } { out-of-bounds }
5207 { \token_to_str:N #1 } { \int_use:N \l__intarray_bad_index_int } {
5208 \intarray_count:N #1 }
5209 }
5210 \cs_generate_variant:Nn \intarray_gset:Nnn { c }
5211 \cs_generate_variant:Nn \intarray_gzero:N { c }
5212 \cs_new:Npn \__kernel_intarray_item:Nn #1#2
5213 { \__intarray_item:w #2 #1 }

```

```

5214 \cs_new:Npn \intarray_item:Nn #1#2
5215 {
5216     \__intarray_item:wF \int_eval:n {#2} #1
5217     {
5218         \msg_expandable_error:nnfff { kernel } { out-of-bounds }
5219         { \token_to_str:N #1 } { \int_use:N \l__intarray_bad_index_int } {
5220 \intarray_count:N #1 }
5221         0
5222     }
5223 }
5224 \cs_generate_variant:Nn \intarray_item:Nn { c }
5225 \cs_new:Npn \intarray_rand_item:N #1
5226 { \intarray_item:Nn #1 { \int_rand:n { \intarray_count:N #1 } } }
5227 \cs_generate_variant:Nn \intarray_rand_item:N { c }
5228 \cs_new_protected:Npn \intarray_const_from_clist:Nn #1#2
5229 {
5230     \__intarray_new:N #1
5231     \int_zero:N \l__intarray_loop_int
5232     \clist_map_inline:nn {#2}
5233     {
5234         \int_incr:N \l__intarray_loop_int
5235         \__kernel_intarray_gset:Nnn #1 \l__intarray_loop_int { \int_eval:n {##1} } }
5236 }
5237 \cs_generate_variant:Nn \intarray_const_from_clist:Nn { c }
5238 \cs_new:Npn \__kernel_intarray_range_to_clist:Nnn #1#2#3
5239 {
5240     \__intarray_range_to_clist:w #1
5241     \int_eval:n {#2} ~ \int_eval:n {#3} ~
5242 }
5243 \cs_new_protected:Npn \__kernel_intarray_gset_range_from_clist:Nnn #1#2#3
5244 {
5245     \__intarray_gset_range:w \int_eval:w #2 #1 #3 , , \scan_stop:
5246 }
5247 \cs_new_protected:Npn \__intarray_gset_overflow_test:nw #1
5248 {
5249 }
5250 {
5251     \cs_new_eq:NN \__intarray_entry:w \tex_fontdimen:D
5252     \cs_new_eq:NN \__intarray_count:w \tex_hyphenchar:D
5253     \dim_const:Nn \c__intarray_sp_dim { 1 sp }
5254     \int_new:N \g__intarray_font_int
5255     \cs_new_protected:Npn \__intarray_new:N #1
5256     {
5257         \__kernel_chk_if_free_cs:N #1
5258         \int_gincr:N \g__intarray_font_int

```

```

5259 \tex_global:D\tex_font:D #1
5260 = cmr10~at~ \g__intarray_font_int \c__intarray_sp_dim \scan_stop:
5261 \int_step_inline:nn { 8 }
5262 { \__kernel_intarray_gset:Nnn #1 {##1} \c_zero_int }
5263 }
5264 \cs_new_protected:Npn \intarray_new:Nn #1#2
5265 {
5266 \__intarray_new:N #1
5267 \__intarray_count:w #1 = \int_eval:n {#2} \scan_stop:
5268 \int_compare:nNnT { \intarray_count:N #1 } < 0
5269 {
5270 \msg_error:nne { kernel } { negative-array-size }
5271 { \intarray_count:N #1 }
5272 }
5273 \int_compare:nNnT { \intarray_count:N #1 } > 0
5274 { \__kernel_intarray_gset:Nnn #1 { \intarray_count:N #1 } { 0 } }
5275 }
5276 \cs_generate_variant:Nn \intarray_new:Nn { c }
5277 \cs_new:Npn \intarray_count:N #1 { \int_value:w \__intarray_count:w #1 }
5278 \cs_generate_variant:Nn \intarray_count:N { c }
5279 \cs_new:Npn \__intarray_signed_max_dim:n #1
5280 { \int_value:w \int_compare:nNnT {#1} < 0 { - } \c_max_dim }
5281 \cs_new:Npn \__intarray_bounds:NNnTF #1#2#3
5282 {
5283 \if_int_compare:w 1 > #3 \exp_stop_f:
5284 \__intarray_bounds_error:NNnw #1 #2 {#3}
5285 \else:
5286 \if_int_compare:w #3 > \intarray_count:N #2 \exp_stop_f:
5287 \__intarray_bounds_error:NNnw #1 #2 {#3}
5288 \fi:
5289 \fi:
5290 \use_i:nn
5291 }
5292 \cs_new:Npn \__intarray_bounds_error:NNnw #1#2#3#4 \use_i:nn #5#6
5293 {
5294 #4
5295 #1 { kernel } { out-of-bounds }
5296 { \token_to_str:N #2 } {#3} { \intarray_count:N #2 }
5297 #6
5298 }
5299 \cs_new_protected:Npn \__kernel_intarray_gset:Nnn #1#2#3
5300 { \__intarray_entry:w #2 #1 #3 \c__intarray_sp_dim }
5301 \cs_new_protected:Npn \intarray_gset:Nnn #1#2#3
5302 {
5303 \exp_after:wN \__intarray_gset:Nww
5304 \exp_after:wN #1

```

```
5305 \int_value:w \int_eval:n {#2} \exp_after:wN \__intarray_sep: 15305
5306 \int_value:w \int_eval:n {#3} \__intarray_sep: 15306
5307 } 15307
5308 \cs_generate_variant:Nn \intarray_gset:Nnn { c } 15308
5309 \cs_new_protected:Npn \__intarray_gset:Nww #1#2 \__intarray_sep: #3 \__intarray_sep: 15309
5310 { 15310
5311 \__intarray_bounds:NNnTF \msg_error:nneee #1 {#2} 15311
5312 { 15312
5313 \__intarray_gset_overflow_test:nw {#3} 15313
5314 \__kernel_intarray_gset:Nnn #1 {#2} {#3} 15314
5315 } 15315
5316 {} 15316
5317 } 15317
5318 \cs_if_exist:NTF \tex_ifabsnum:D 15318
5319 { 15319
5320 \cs_new_protected:Npn \__intarray_gset_overflow_test:nw #1 15320
5321 { 15321
5322 \tex_ifabsnum:D #1 > \c_max_dim 15322
5323 \exp_after:wN \__intarray_gset_overflow:NNnn 15323
5324 \fi: 15324
5325 } 15325
5326 } 15326
5327 { 15327
5328 \cs_new_protected:Npn \__intarray_gset_overflow_test:nw #1 15328
5329 { 15329
5330 \if_int_compare:w \int_abs:n {#1} > \c_max_dim 15330
5331 \exp_after:wN \__intarray_gset_overflow:NNnn 15331
5332 \fi: 15332
5333 } 15333
5334 } 15334
5335 \cs_new_protected:Npn \__intarray_gset_overflow:NNnn #1#2#3#4 15335
5336 { 15336
5337 \msg_error:nneeee { kernel } { overflow } 15337
5338 { \token_to_str:N #2 } {#3} {#4} { \__intarray_signed_max_dim:n {#4} } 15338
5339 #1 #2 {#3} { \__intarray_signed_max_dim:n {#4} } 15339
5340 } 15340
5341 \cs_new_protected:Npn \intarray_gzero:N #1 15341
5342 { 15342
5343 \int_zero:N \l__intarray_loop_int 15343
5344 \prg_replicate:nn { \intarray_count:N #1 } 15344
5345 { 15345
5346 \int_incr:N \l__intarray_loop_int 15346
5347 \__intarray_entry:w \l__intarray_loop_int #1 \c_zero_dim 15347
5348 } 15348
5349 } 15349
5350 \cs_generate_variant:Nn \intarray_gzero:N { c } 15350
```

```
5351 \cs_new:Npn \__kernel_intarray_item:Nn #1#2
5352 { \int_value:w \__intarray_entry:w #2 #1 }
5353 \cs_new:Npn \intarray_item:Nn #1#2
5354 {
5355   \exp_after:wN \__intarray_item:Nw
5356   \exp_after:wN #1
5357   \int_value:w \int_eval:n {#2} \__intarray_sep:
5358 }
5359 \cs_generate_variant:Nn \intarray_item:Nn { c }
5360 \cs_new:Npn \__intarray_item:Nw #1#2 \__intarray_sep:
5361 {
5362   \__intarray_bounds:NNnTF \msg_expandable_error:nnfff #1 {#2}
5363   { \__kernel_intarray_item:Nn #1 {#2} }
5364   { 0 }
5365 }
5366 \cs_new:Npn \intarray_rand_item:N #1
5367 { \intarray_item:Nn #1 { \int_rand:n { \intarray_count:N #1 } } }
5368 \cs_generate_variant:Nn \intarray_rand_item:N { c }
5369 \cs_new_protected:Npn \intarray_const_from_clist:Nn #1#2
5370 {
5371   \__intarray_new:N #1
5372   \int_zero:N \l__intarray_loop_int
5373   \clist_map_inline:nn {#2}
5374   { \exp_args:Nf \__intarray_const_from_clist:nN { \int_eval:n {##1} } #1 }
5375   \__intarray_count:w #1 \l__intarray_loop_int
5376 }
5377 \cs_generate_variant:Nn \intarray_const_from_clist:Nn { c }
5378 \cs_new_protected:Npn \__intarray_const_from_clist:nN #1#2
5379 {
5380   \int_incr:N \l__intarray_loop_int
5381   \__intarray_gset_overflow_test:nw {#1}
5382   \__kernel_intarray_gset:Nnn #2 \l__intarray_loop_int {#1}
5383 }
5384 \cs_new:Npn \__intarray_to_clist:Nn #1#2
5385 {
5386   \int_compare:nNnF { \intarray_count:N #1 } = \c_zero_int
5387   {
5388     \exp_last_unbraced:Nf \use_none:n
5389     { \__intarray_to_clist:w 1 \__intarray_sep: #1 {#2} \prg_break_point: }
5390   }
5391 }
5392 \cs_new:Npn \__intarray_to_clist:w #1 \__intarray_sep: #2#3
5393 {
5394   \if_int_compare:w #1 > \__intarray_count:w #2
5395     \prg_break:n
5396   \fi:
```



```
5397         #3 \__kernel_intarray_item:Nn #2 {#1}
5398         \exp_after:wN \__intarray_to_clist:w
5399         \int_value:w \int_eval:w #1 + \c_one_int \__intarray_sep: #2 {#3}
5400     }
5401 \cs_new:Npn \__kernel_intarray_range_to_clist:Nnn #1#2#3
5402 {
5403     \exp_last_unbraced:Nf \use_none:n
5404     {
5405         \exp_after:wN \__intarray_range_to_clist:ww
5406         \int_value:w \int_eval:w #2 \exp_after:wN \__intarray_sep:
5407         \int_value:w \int_eval:w #3 \__intarray_sep:
5408         #1 \prg_break_point:
5409     }
5410 }
5411 \cs_new:Npn \__intarray_range_to_clist:ww #1 \__intarray_sep: #2 \__intarray_sep: #3
5412 {
5413     \if_int_compare:w #1 > #2 \exp_stop_f:
5414     \prg_break:n
5415     \fi:
5416     , \__kernel_intarray_item:Nn #3 {#1}
5417     \exp_after:wN \__intarray_range_to_clist:ww
5418     \int_value:w \int_eval:w #1 + \c_one_int \__intarray_sep: #2 \__intarray_sep: #3
5419 }
5420 \cs_new_protected:Npn \__kernel_intarray_gset_range_from_clist:Nnn #1#2#3
5421 {
5422     \int_set:Nn \l__intarray_loop_int {#2}
5423     \__intarray_gset_range:Nw #1 #3 , , \prg_break_point:
5424 }
5425 \cs_new_protected:Npn \__intarray_gset_range:Nw #1 #2 ,
5426 {
5427     \if_catcode:w \scan_stop: \tl_to_str:n {#2} \scan_stop:
5428     \prg_break:n
5429     \fi:
5430     \__kernel_intarray_gset:Nnn #1 \l__intarray_loop_int {#2}
5431     \int_incr:N \l__intarray_loop_int
5432     \__intarray_gset_range:Nw #1
5433 }
5434 }
5435 \prg_new_eq_conditional:NNn \intarray_if_exist:N \cs_if_exist:N
5436 { TF , T , F , p }
5437 \prg_new_eq_conditional:NNn \intarray_if_exist:c \cs_if_exist:c
5438 { TF , T , F , p }
5439 \cs_new_protected:Npn \intarray_show:N { \__intarray_show:NN \msg_show:nneeee }
5440 \cs_generate_variant:Nn \intarray_show:N { c }
5441 \cs_new_protected:Npn \intarray_log:N { \__intarray_show:NN \msg_log:nneeee }
5442 \cs_generate_variant:Nn \intarray_log:N { c }
```

```
5443 \cs_new_protected:Npn \__intarray_show:NN #1#2
5444 {
5445   \__kernel_chk_defined:NT #2
5446   {
5447     #1 { intarray } { show }
5448     { \token_to_str:N #2 }
5449     { \intarray_count:N #2 }
5450     { >~ \__intarray_to_clist:Nn #2 { , ~ } }
5451     { }
5452   }
5453 }
5454 %% File: l3fp.dtx
5455 %% File: l3fp-aux.dtx
5456 \cs_new_eq:NN \__fp_int_eval:w \tex_numexpr:D
5457 \cs_new_eq:NN \__fp_int_eval_end: \scan_stop:
5458 \cs_new_eq:NN \__fp_int_to_roman:w \tex_romannumeral:D
5459 \cs_new_eq:NN \__fp_sep: \__kernel_int_sep:
5460 \cs_new:Npn \__fp_use_none_stop_f:n #1 { \exp_stop_f: }
5461 \cs_new:Npn \__fp_use_s:n #1 { #1\__fp_sep: }
5462 \cs_new:Npn \__fp_use_s:nn #1#2 { #1#2\__fp_sep: }
5463 \cs_new:Npn \__fp_use_none_until_s:w #1\__fp_sep: { }
5464 \cs_new:Npn \__fp_use_i_until_s:nw #1#2\__fp_sep: {#1}
5465 \cs_new:Npn \__fp_use_ii_until_s:nnw #1#2#3\__fp_sep: {#2}
5466 \cs_new:Npn \__fp_reverse_args:Nww #1 #2\__fp_sep: #3\__fp_sep:
5467   { #1 #3\__fp_sep: #2\__fp_sep: }
5468 \cs_new:Npn \__fp_rrot:www #1\__fp_sep: #2\__fp_sep: #3\__fp_sep:
5469   { #2\__fp_sep: #3\__fp_sep: #1\__fp_sep: }
5470 \cs_new:Npn \__fp_use_i:ww #1\__fp_sep: #2\__fp_sep: { #1\__fp_sep: }
5471 \cs_new:Npn \__fp_use_i:www #1\__fp_sep: #2\__fp_sep: #3\__fp_sep: { #1\__fp_sep: }
5472 \cs_new_protected:Npn \__fp_misused:n #1
5473   { \msg_error:nne { fp } { misused } { \fp_to_tl:n {#1} } }
5474 \scan_new:N \s__fp
5475 \cs_new_protected:Npn \__fp_chk:w #1 \__fp_sep:
5476   { \__fp_misused:n { \s__fp \__fp_chk:w #1 \__fp_sep: } }
5477 \scan_new:N \s__fp_expr_mark
5478 \scan_new:N \s__fp_expr_stop
5479 \scan_new:N \s__fp_mark
5480 \scan_new:N \s__fp_stop
5481 \cs_new:Npn \__fp_use_i_delimit_by_s_stop:nw #1 #2 \s__fp_stop {#1}
5482 \scan_new:N \s__fp_invalid
5483 \scan_new:N \s__fp_underflow
5484 \scan_new:N \s__fp_overflow
5485 \scan_new:N \s__fp_division
5486 \scan_new:N \s__fp_exact
5487 \tl_const:Nn \c_zero_fp { \s__fp \__fp_chk:w 0 0 \s__fp_exact \__fp_sep: }
5488 \tl_const:Nn \c_minus_zero_fp { \s__fp \__fp_chk:w 0 2 \s__fp_exact \__fp_sep: }
```

```
5489 \tl_const:Nn \c_inf_fp { \s__fp \__fp_chk:w 2 0 \s__fp_exact \__fp_sep: } 15489
5490 \tl_const:Nn \c_minus_inf_fp { \s__fp \__fp_chk:w 2 2 \s__fp_exact \__fp_sep: } 15490
5491 \tl_const:Nn \c_nan_fp { \s__fp \__fp_chk:w 3 1 \s__fp_exact \__fp_sep: } 15491
5492 \int_const:Nn \c__fp_prec_int { 16 } 15492
5493 \int_const:Nn \c__fp_half_prec_int { 8 } 15493
5494 \int_const:Nn \c__fp_block_int { 4 } 15494
5495 \int_const:Nn \c__fp_myriad_int { 10000 } 15495
5496 \int_const:Nn \c__fp_minus_min_exponent_int { 10000 } 15496
5497 \int_const:Nn \c__fp_max_exponent_int { 10000 } 15497
5498 \int_const:Nn \c__fp_max_exp_exponent_int { 5 } 15498
5499 \tl_const:Ne \c__fp_overflowing_fp 15499
5500 { 15500
5501 \s__fp \__fp_chk:w 1 0 15501
5502 { \int_eval:n { \c__fp_max_exponent_int + 1 } } 15502
5503 {1000} {0000} {0000} {0000} \__fp_sep: 15503
5504 } 15504
5505 \cs_new:Npn \__fp_zero_fp:N #1 15505
5506 { \s__fp \__fp_chk:w 0 #1 \s__fp_underflow \__fp_sep: } 15506
5507 \cs_new:Npn \__fp_inf_fp:N #1 15507
5508 { \s__fp \__fp_chk:w 2 #1 \s__fp_overflow \__fp_sep: } 15508
5509 \cs_new:Npn \__fp_exponent:w \s__fp \__fp_chk:w #1 15509
5510 { 15510
5511 \if_meaning:w 1 #1 15511
5512 \exp_after:wN \__fp_use_ii_until_s:nnw 15512
5513 \else: 15513
5514 \exp_after:wN \__fp_use_i_until_s:nw 15514
5515 \exp_after:wN 0 15515
5516 \fi: 15516
5517 } 15517
5518 \cs_new:Npn \__fp_neg_sign:N #1 15518
5519 { \__fp_int_eval:w 2 - #1 \__fp_int_eval_end: } 15519
5520 \cs_new:Npn \__fp_kind:w #1 15520
5521 { 15521
5522 \__fp_if_type_fp:NTwFw 15522
5523 #1 \__fp_use_ii_until_s:nnw 15523
5524 \s__fp { \__fp_use_i_until_s:nw 4 } 15524
5525 \s__fp_stop 15525
5526 } 15526
5527 \cs_new:Npn \__fp_sanitize:Nw #1 #2\__fp_sep: 15527
5528 { 15528
5529 \if_case:w 15529
5530 \if_int_compare:w #2 > \c__fp_max_exponent_int 1 ~ \else: 15530
5531 \if_int_compare:w #2 < - \c__fp_minus_min_exponent_int 2 ~ \else: 15531
5532 \if_meaning:w 1 #1 3 ~ \fi: \fi: \fi: 0 ~ 15532
5533 \or: \exp_after:wN \__fp_overflow:w 15533
5534 \or: \exp_after:wN \__fp_underflow:w 15534
```

```
5535 \or: \exp_after:wN \__fp_sanitizew 15535
5536 \fi: 15536
5537 \s__fp \__fp_chk:w 1 #1 {#2} 15537
5538 } 15538
5539 \cs_new:Npn \__fp_sanitizew #1\__fp_sep: #2 { \__fp_sanitizew #2 #1\__fp_sep: } 15539
5540 \cs_new:Npn \__fp_sanitizew #1 \s__fp \__fp_chk:w #1 #2 #3\__fp_sep: 15540
5541 { \c_zero_fp } 15541
5542 \cs_new:Npn \__fp_exp_after_o:w \s__fp \__fp_chk:w #1 15542
5543 { 15543
5544 \if_meaning:w 1 #1 15544
5545 \exp_after:wN \__fp_exp_after_normal:nNNw 15545
5546 \else: 15546
5547 \exp_after:wN \__fp_exp_after_special:nNNw 15547
5548 \fi: 15548
5549 { } 15549
5550 #1 15550
5551 } 15551
5552 \cs_new:Npn \__fp_exp_after_f:nw #1 \s__fp \__fp_chk:w #2 15552
5553 { 15553
5554 \if_meaning:w 1 #2 15554
5555 \exp_after:wN \__fp_exp_after_normal:nNNw 15555
5556 \else: 15556
5557 \exp_after:wN \__fp_exp_after_special:nNNw 15557
5558 \fi: 15558
5559 { \exp:w \exp_end_continue_f:w #1 } 15559
5560 #2 15560
5561 } 15561
5562 \cs_new:Npn \__fp_exp_after_special:nNNw #1#2#3#4\__fp_sep: 15562
5563 { 15563
5564 \exp_after:wN \s__fp 15564
5565 \exp_after:wN \__fp_chk:w 15565
5566 \exp_after:wN #2 15566
5567 \exp_after:wN #3 15567
5568 \exp_after:wN #4 15568
5569 \exp_after:wN \__fp_sep: 15569
5570 #1 15570
5571 } 15571
5572 \cs_new:Npn \__fp_exp_after_normal:nNNw #1 1 #2 #3 #4#5#6#7\__fp_sep: 15572
5573 { 15573
5574 \exp_after:wN \__fp_exp_after_normal:Nwwwww 15574
5575 \exp_after:wN #2 15575
5576 \int_value:w #3 \exp_after:wN \__fp_sep: 15576
5577 \int_value:w 1 #4 \exp_after:wN \__fp_sep: 15577
5578 \int_value:w 1 #5 \exp_after:wN \__fp_sep: 15578
5579 \int_value:w 1 #6 \exp_after:wN \__fp_sep: 15579
5580 \int_value:w 1 #7 \exp_after:wN \__fp_sep: #1 15580
```

```
5581 }
5582 \cs_new:Npn \__fp_exp_after_normal:Nwwwww
5583   #1 #2\__fp_sep: 1 #3 \__fp_sep: 1 #4 \__fp_sep: 1 #5 \__fp_sep: 1 #6 \__fp_sep:
5584   { \s__fp \__fp_chk:w 1 #1 {#2} {#3} {#4} {#5} {#6} \__fp_sep: }
5585 \scan_new:N \s__fp_tuple
5586 \cs_new_protected:Npn \__fp_tuple_chk:w #1 \__fp_sep:
5587   { \__fp_misused:n { \s__fp_tuple \__fp_tuple_chk:w #1 \__fp_sep: } }
5588 \tl_const:Nn \c__fp_empty_tuple_fp
5589   { \s__fp_tuple \__fp_tuple_chk:w { } \__fp_sep: }
5590 \cs_new:Npn \__fp_array_count:n #1
5591   { \__fp_tuple_count:w \s__fp_tuple \__fp_tuple_chk:w {#1} \__fp_sep: }
5592 \cs_new:Npn \__fp_tuple_count:w \s__fp_tuple \__fp_tuple_chk:w #1 \__fp_sep:
5593   {
5594     \int_value:w \__fp_int_eval:w 0
5595     \__fp_tuple_count_loop:Nw #1 { ? \prg_break: } \__fp_sep:
5596     \prg_break_point:
5597     \__fp_int_eval_end:
5598   }
5599 \cs_new:Npn \__fp_tuple_count_loop:Nw #1#2\__fp_sep:
5600   { \use_none:n #1 + 1 \__fp_tuple_count_loop:Nw }
5601 \cs_new:Npn \__fp_if_type_fp:NTwFw #1 \s__fp #2 #3 \s__fp_stop {#2}
5602 \cs_new:Npn \__fp_array_if_all_fp:nTF #1
5603   {
5604     \__fp_array_if_all_fp_loop:w #1 { \s__fp \prg_break: } \__fp_sep:
5605     \prg_break_point: \use_i:nn
5606   }
5607 \cs_new:Npn \__fp_array_if_all_fp_loop:w #1#2 \__fp_sep:
5608   {
5609     \__fp_if_type_fp:NTwFw
5610     #1 \__fp_array_if_all_fp_loop:w
5611     \s__fp { \prg_break:n \use_iii:nnn }
5612     \s__fp_stop
5613   }
5614 \cs_new:Npn \__fp_type_from_scan:N #1
5615   {
5616     \__fp_if_type_fp:NTwFw
5617     #1 { }
5618     \s__fp { \__fp_type_from_scan_other:N #1 }
5619     \s__fp_stop
5620   }
5621 \cs_new:Npe \__fp_type_from_scan_other:N #1
5622   {
5623     \exp_not:N \exp_after:wN \exp_not:N \__fp_type_from_scan:w
5624     \exp_not:N \token_to_str:N #1 \s__fp_mark
5625     \tl_to_str:n { s__fp_? } \s__fp_mark \s__fp_stop
5626   }
```

```
5627 \exp_last_unbraced:NNNNNo
5628 \cs_new:Npn \__fp_type_from_scan:w #1
5629 { \tl_to_str:n { s__fp } } #2 \s__fp_mark #3 \s__fp_stop {#2}
5630 \cs_new:Npn \__fp_change_func_type:NNN #1#2#3
5631 {
5632 \__fp_if_type_fp:NTwFw
5633 #1 #2
5634 \s__fp
5635 {
5636 \exp_after:wN \__fp_change_func_type_chk:NNN
5637 \cs:w
5638 __fp \__fp_type_from_scan_other:N #1
5639 \exp_after:wN \__fp_change_func_type_aux:w \token_to_str:N #2
5640 \cs_end:
5641 #2 #3
5642 }
5643 \s__fp_stop
5644 }
5645 \exp_last_unbraced:NNNNNo
5646 \cs_new:Npn \__fp_change_func_type_aux:w #1 { \tl_to_str:n { __fp } } { }
5647 \cs_new:Npn \__fp_change_func_type_chk:NNN #1#2#3
5648 {
5649 \if_meaning:w \scan_stop: #1
5650 \exp_after:wN #3 \exp_after:wN #2
5651 \else:
5652 \exp_after:wN #1
5653 \fi:
5654 }
5655 \cs_new:Npn \__fp_exp_after_any_f:Nnw #1
5656 { \cs:w __fp_exp_after \__fp_type_from_scan_other:N #1 _f:nw \cs_end: }
5657 \cs_new:Npn \__fp_exp_after_any_f:nw #1#2
5658 {
5659 \__fp_if_type_fp:NTwFw
5660 #2 \__fp_exp_after_f:nw
5661 \s__fp { \__fp_exp_after_any_f:Nnw #2 }
5662 \s__fp_stop
5663 {#1} #2
5664 }
5665 \cs_new_eq:NN \__fp_exp_after_expr_stop_f:nw \use_none:nn
5666 \cs_new:Npn \__fp_exp_after_tuple_o:w
5667 { \__fp_exp_after_tuple_f:nw { \exp_after:wN \exp_stop_f: } }
5668 \cs_new:Npn \__fp_exp_after_tuple_f:nw
5669 #1 \s__fp_tuple \__fp_tuple_chk:w #2 \__fp_sep:
5670 {
5671 \exp_after:wN \s__fp_tuple
5672 \exp_after:wN \__fp_tuple_chk:w
```



```
5673 \exp_after:wN {
5674 \exp:w \exp_end_continue_f:w
5675 \_fp_exp_after_array_f:w #2 \s\_fp_expr_stop
5676 \exp_after:wN }
5677 \exp_after:wN \_fp_sep:
5678 \exp:w \exp_end_continue_f:w #1
5679 }
5680 \cs_new:Npn \_fp_exp_after_array_f:w
5681 { \_fp_exp_after_any_f:nw { \_fp_exp_after_array_f:w } }
5682 \int_const:Nn \c__fp_leading_shift_int { - 5 0000 }
5683 \int_const:Nn \c__fp_middle_shift_int { 5 0000 * 9999 }
5684 \int_const:Nn \c__fp_trailing_shift_int { 5 0000 * 10000 }
5685 \cs_new:Npn \_fp_pack:NNNNNw #1 #2#3#4#5 #6\_fp_sep:
5686 { + #1#2#3#4#5 \_fp_sep: {#6} }
5687 \int_const:Nn \c__fp_big_leading_shift_int { - 15 2374 }
5688 \int_const:Nn \c__fp_big_middle_shift_int { 15 2374 * 9999 }
5689 \int_const:Nn \c__fp_big_trailing_shift_int { 15 2374 * 10000 }
5690 \cs_new:Npn \_fp_pack_big:NNNNNNw #1#2 #3#4#5#6 #7\_fp_sep:
5691 { + #1#2#3#4#5#6 \_fp_sep: {#7} }
5692 \int_const:Nn \c__fp_Bigg_leading_shift_int { - 20 0000 }
5693 \int_const:Nn \c__fp_Bigg_middle_shift_int { 20 0000 * 9999 }
5694 \int_const:Nn \c__fp_Bigg_trailing_shift_int { 20 0000 * 10000 }
5695 \cs_new:Npn \_fp_pack_Bigg:NNNNNNw #1#2 #3#4#5#6 #7\_fp_sep:
5696 { + #1#2#3#4#5#6 \_fp_sep: {#7} }
5697 \cs_new:Npn \_fp_pack_twice_four:wNNNNNNNN #1\_fp_sep: #2#3#4#5 #6#7#8#9
5698 { #1 {#2#3#4#5} {#6#7#8#9} \_fp_sep: }
5699 \cs_new:Npn \_fp_pack_eight:wNNNNNNNN #1\_fp_sep: #2#3#4#5 #6#7#8#9
5700 { #1 {#2#3#4#5#6#7#8#9} \_fp_sep: }
5701 \cs_new:Npn \_fp_basics_pack_low:NNNNNw #1 #2#3#4#5 #6\_fp_sep:
5702 { + #1 - 1 \_fp_sep: {#2#3#4#5} {#6} \_fp_sep: }
5703 \cs_new:Npn \_fp_basics_pack_high:NNNNNw #1 #2#3#4#5 #6\_fp_sep:
5704 {
5705 \if_meaning:w 2 #1
5706 \_fp_basics_pack_high_carry:w
5707 \fi:
5708 \_fp_sep: {#2#3#4#5} {#6}
5709 }
5710 \cs_new:Npn \_fp_basics_pack_high_carry:w \fi: \_fp_sep: #1
5711 { \fi: + 1 \_fp_sep: {1000} }
5712 \cs_new:Npn \_fp_basics_pack_weird_low:NNNNw #1 #2#3#4 #5\_fp_sep:
5713 {
5714 \if_meaning:w 2 #1
5715 + 1
5716 \fi:
5717 \_fp_int_eval_end:
5718 #2#3#4\_fp_sep: {#5} \_fp_sep:
```



```
5719 }
5720 \cs_new:Npn \__fp_basics_pack_weird_high:NNNNNNNNw
5721 1 #1#2#3#4 #5#6#7#8 #9\__fp_sep: { \__fp_sep: {#1#2#3#4} {#5#6#7#8} {#9} }
5722 \cs_new:Npn \__fp_decimate:nNnnnn #1
5723 {
5724   \cs:w
5725     __fp_decimate_
5726     \if_int_compare:w \__fp_int_eval:w #1 > \c__fp_prec_int
5727       tiny
5728     \else:
5729       \__fp_int_to_roman:w \__fp_int_eval:w #1
5730     \fi:
5731     :Nnnnn
5732   \cs_end:
5733 }
5734 \cs_new:Npn \__fp_decimate_:Nnnnn #1 #2#3#4#5
5735 { #1 0 {#2#3} {#4#5} \__fp_sep: }
5736 \cs_new:Npn \__fp_decimate_tiny:Nnnnn #1 #2#3#4#5
5737 { #1 1 { 0000 0000 } { 0000 0000 } 0 #2#3#4#5 \__fp_sep: }
5738 \cs_new:Npn \__fp_tmp:w #1 #2 #3
5739 {
5740   \cs_new:cpn { __fp_decimate_ #1 :Nnnnn } ##1 ##2##3##4##5
5741   {
5742     \exp_after:wN ##1
5743     \int_value:w
5744     \exp_after:wN \__fp_round_digit:Nw #2 \__fp_sep:
5745     \__fp_decimate_pack:nnnnnnnnnnw #3 \__fp_sep:
5746   }
5747 }
5748 \__fp_tmp:w {i} {\use_none:nnn #50}{ 0{#2}#3{#4}#5 }
5749 \__fp_tmp:w {ii} {\use_none:nn #5 }{ 00{#2}#3{#4}#5 }
5750 \__fp_tmp:w {iii} {\use_none:n #5 }{ 000{#2}#3{#4}#5 }
5751 \__fp_tmp:w {iv} { #5 }{ {0000}#2{#3}#4 #5 }
5752 \__fp_tmp:w {v} {\use_none:nnn #4#5 }{ 0{0000}#2{#3}#4 #5 }
5753 \__fp_tmp:w {vi} {\use_none:nn #4#5 }{ 00{0000}#2{#3}#4 #5 }
5754 \__fp_tmp:w {vii} {\use_none:n #4#5 }{ 000{0000}#2{#3}#4 #5 }
5755 \__fp_tmp:w {viii}{ #4#5 }{ {0000}0000{#2}#3 #4 #5 }
5756 \__fp_tmp:w {ix} {\use_none:nnn #3#4+#5}{ 0{0000}0000{#2}#3 #4 #5 }
5757 \__fp_tmp:w {x} {\use_none:nn #3#4+#5}{ 00{0000}0000{#2}#3 #4 #5 }
5758 \__fp_tmp:w {xi} {\use_none:n #3#4+#5}{ 000{0000}0000{#2}#3 #4 #5 }
5759 \__fp_tmp:w {xii} { #3#4+#5}{ {0000}0000{0000}#2 #3 #4 #5 }
5760 \__fp_tmp:w {xiii}{\use_none:nnn#2#3+#4#5}{ 0{0000}0000{0000}#2 #3 #4 #5 }
5761 \__fp_tmp:w {xiv} {\use_none:nn #2#3+#4#5}{ 00{0000}0000{0000}#2 #3 #4 #5 }
5762 \__fp_tmp:w {xv} {\use_none:n #2#3+#4#5}{ 000{0000}0000{0000}#2 #3 #4 #5 }
5763 \__fp_tmp:w {xvi} { #2#3+#4#5}{ {0000}0000{0000}0000 #2 #3 #4 #5 }
5764 \cs_new:Npn \__fp_decimate_pack:nnnnnnnnnnw #1#2#3#4#5
```

```
5765 { \__fp_decimate_pack:nnnnnnw { #1#2#3#4#5 } }
5766 \cs_new:Npn \__fp_decimate_pack:nnnnnnw #1 #2#3#4#5#6
5767 { {#1} {#2#3#4#5#6} }
5768 \cs_new:Npn \__fp_case_use:nw #1#2 \fi: #3 \s__fp { \fi: #1 \s__fp }
5769 \cs_new:Npn \__fp_case_return:nw #1#2 \fi: #3 \__fp_sep: { \fi: #1 }
5770 \cs_new:Npn \__fp_case_return_o:Nw #1#2 \fi: #3 \s__fp #4 \__fp_sep:
5771 { \fi: \exp_after:wN #1 }
5772 \cs_new:Npn \__fp_case_return_same_o:w #1 \fi: #2 \s__fp
5773 { \fi: \__fp_exp_after_o:w \s__fp }
5774 \cs_new:Npn \__fp_case_return_o:Nww #1#2 \fi: #3 \s__fp #4 \__fp_sep: #5 \__fp_sep:
5775 { \fi: \exp_after:wN #1 }
5776 \cs_new:Npn \__fp_case_return_i_o:ww
5777 #1 \fi: #2 \s__fp #3 \__fp_sep: \s__fp #4 \__fp_sep:
5778 { \fi: \__fp_exp_after_o:w \s__fp #3 \__fp_sep: }
5779 \cs_new:Npn \__fp_case_return_ii_o:ww #1 \fi: #2 \s__fp #3 \__fp_sep:
5780 { \fi: \__fp_exp_after_o:w }
5781 \prg_new_conditional:Npnn \__fp_int:w \s__fp \__fp_chk:w #1 #2 #3 #4 \__fp_sep:
5782 { TF , T , F , p }
5783 {
5784   \if_case:w #1 \exp_stop_f:
5785     \prg_return_true:
5786   \or:
5787     \if_charcode:w 0
5788       \__fp_decimate:nNnnnn { \c__fp_prec_int - #3 }
5789       \__fp_use_i_until_s:nw #4
5790       \prg_return_true:
5791     \else:
5792       \prg_return_false:
5793     \fi:
5794   \else: \prg_return_false:
5795   \fi:
5796 }
5797 \cs_new:Npn \__fp_small_int:wTF \s__fp \__fp_chk:w #1#2
5798 {
5799   \if_case:w #1 \exp_stop_f:
5800     \__fp_case_return:nw { \__fp_small_int_true:wTF 0 \__fp_sep: }
5801   \or: \exp_after:wN \__fp_small_int_normal:NnwTF
5802   \or:
5803     \__fp_case_return:nw
5804     {
5805       \exp_after:wN \__fp_small_int_true:wTF \int_value:w
5806       \if_meaning:w 2 #2 - \fi: 1 0000 0000 \__fp_sep:
5807     }
5808   \else: \__fp_case_return:nw \use_ii:nn
5809   \fi:
5810   #2
```

```

5811 }
5812 \cs_new:Npn \__fp_small_int_true:wTF #1\__fp_sep: #2#3 { #2 {#1} }
5813 \cs_new:Npn \__fp_small_int_normal:NnwTF #1#2#3\__fp_sep:
5814 {
5815     \__fp_decimate:nNnnnn { \c__fp_prec_int - #2 }
5816     \__fp_small_int_test:NnnwNw
5817     #3 #1
5818 }
5819 \cs_new:Npn \__fp_small_int_test:NnnwNw #1#2#3#4\__fp_sep: #5
5820 {
5821     \if_meaning:w 0 #1
5822     \exp_after:wN \__fp_small_int_true:wTF
5823     \int_value:w \if_meaning:w 2 #5 - \fi:
5824     \if_int_compare:w #2 > \c_zero_int
5825         1 0000 0000
5826     \else:
5827         #3
5828     \fi:
5829     \exp_after:wN \__fp_sep:
5830 \else:
5831     \exp_after:wN \use_ii:nn
5832 \fi:
5833 }
5834 \cs_new_eq:NN \__fp_str_if_eq:nn \tex_strcmp:D
5835 \cs_new:Npn \__fp_func_to_name:N #1
5836 {
5837     \exp_last_unbraced:Nf
5838     \__fp_func_to_name_aux:w { \cs_to_str:N #1 } X
5839 }
5840 \cs_set_protected:Npn \__fp_tmp:w #1 #2
5841 { \cs_new:Npn \__fp_func_to_name_aux:w ##1 #1 ##2 #2 ##3 X {##2} }
5842 \exp_args:Nff \__fp_tmp:w { \tl_to_str:n { __fp_ } }
5843 { \tl_to_str:n { _o: } }
5844 \msg_new:nnnn { fp } { { misused }
5845 { A~floating~point~with~value~'#1'~was~misused. }
5846 {
5847     To~obtain~the~value~of~a~floating~point~variable,~use~
5848     '\token_to_str:N \fp_to_decimal:N',~
5849     '\token_to_str:N \fp_to_tl:N',~or~other~
5850     conversion~functions.
5851 }
5852 \prop_gput:Nnn \g_msg_module_name_prop { fp } { LaTeX }
5853 \prop_gput:Nnn \g_msg_module_type_prop { fp } { }
5854 %% File: l3fp-traps.dtx
5855 \flag_new:N \l_fp_invalid_operation_flag
5856 \flag_new:N \l_fp_division_by_zero_flag

```

```
5857 \flag_new:N \l_fp_overflow_flag 1585
5858 \flag_new:N \l_fp_underflow_flag 1585
5859 \cs_new_protected:Npn \fp_trap:nn #1#2 1585
5860 { 1586
5861     \cs_if_exist_use:cF { __fp_trap_#1_set_#2: } 1586
5862     { 1586
5863         \clist_if_in:nnTF 1586
5864         { invalid_operation , division_by_zero , overflow , underflow } 1586
5865         {#1} 1586
5866         { 1586
5867             \msg_error:nnee { fp } 1586
5868             { unknown-fpu-trap-type } {#1} {#2} 1586
5869         } 1586
5870         { 1587
5871             \msg_error:nne 1587
5872             { fp } { unknown-fpu-exception } {#1} 1587
5873         } 1587
5874     } 1587
5875 } 1587
5876 \cs_new_protected:Npn \__fp_trap_invalid_operation_set_error: 1587
5877 { \__fp_trap_invalid_operation_set:N \prg_do_nothing: } 1587
5878 \cs_new_protected:Npn \__fp_trap_invalid_operation_set_flag: 1587
5879 { \__fp_trap_invalid_operation_set:N \use_none:nnnnn } 1587
5880 \cs_new_protected:Npn \__fp_trap_invalid_operation_set_none: 1588
5881 { \__fp_trap_invalid_operation_set:N \use_none:nnnnnnn } 1588
5882 \cs_new_protected:Npn \__fp_trap_invalid_operation_set:N #1 1588
5883 { 1588
5884     \exp_args:Nno \use:n 1588
5885     { \cs_set:Npn \__fp_invalid_operation:nnw ##1##2##3\__fp_sep: } 1588
5886     { 1588
5887         #1 1588
5888         \__fp_error:nnfn { invalid } {##2} { \fp_to_tl:n { ##3\__fp_sep: } } { } 1588
5889         \flag_ensure_raised:N \l_fp_invalid_operation_flag 1588
5890         ##1 1589
5891     } 1589
5892     \exp_args:Nno \use:n 1589
5893     { \cs_set:Npn \__fp_invalid_operation_o:Nww ##1##2\__fp_sep: ##3\__fp_sep: } 1589
5894     { 1589
5895         #1 1589
5896         \__fp_error:nffn { invalid-ii } 1589
5897         { \fp_to_tl:n { ##2\__fp_sep: } } 1589
5898         { \fp_to_tl:n { ##3\__fp_sep: } } 1589
5899         {##1} 1589
5900         \flag_ensure_raised:N \l_fp_invalid_operation_flag 1590
5901         \exp_after:wN \c_nan_fp 1590
5902     } 1590
```

```
5903 \exp_args:Nno \use:n
5904 { \cs_set:Npn \__fp_invalid_operation_tl_o:ff ##1##2 }
5905 {
5906 #1
5907 \__fp_error:nfn { invalid } {##1} {##2} { }
5908 \flag_ensure_raised:N \l_fp_invalid_operation_flag
5909 \exp_after:wN \c_nan_fp
5910 }
5911 }
5912 \cs_new_protected:Npn \__fp_trap_division_by_zero_set_error:
5913 { \__fp_trap_division_by_zero_set:N \prg_do_nothing: }
5914 \cs_new_protected:Npn \__fp_trap_division_by_zero_set_flag:
5915 { \__fp_trap_division_by_zero_set:N \use_none:nnnnn }
5916 \cs_new_protected:Npn \__fp_trap_division_by_zero_set_none:
5917 { \__fp_trap_division_by_zero_set:N \use_none:nnnnnnn }
5918 \cs_new_protected:Npn \__fp_trap_division_by_zero_set:N #1
5919 {
5920 \exp_args:Nno \use:n
5921 { \cs_set:Npn \__fp_division_by_zero_o:Nnw ##1##2##3\__fp_sep: }
5922 {
5923 #1
5924 \__fp_error:nnfn { zero-div } {##2} { \fp_to_tl:n { ##3\__fp_sep: } } { }
5925 \flag_ensure_raised:N \l_fp_division_by_zero_flag
5926 \exp_after:wN ##1
5927 }
5928 \exp_args:Nno \use:n
5929 {
5930 \cs_set:Npn \__fp_division_by_zero_o:NNww ##1##2##3\__fp_sep: ##4\__fp_sep:
5931 }
5932 {
5933 #1
5934 \__fp_error:nfn { zero-div-ii }
5935 { \fp_to_tl:n { ##3\__fp_sep: } }
5936 { \fp_to_tl:n { ##4\__fp_sep: } }
5937 {##2}
5938 \flag_ensure_raised:N \l_fp_division_by_zero_flag
5939 \exp_after:wN ##1
5940 }
5941 }
5942 \cs_new_protected:Npn \__fp_trap_overflow_set_error:
5943 { \__fp_trap_overflow_set:N \prg_do_nothing: }
5944 \cs_new_protected:Npn \__fp_trap_overflow_set_flag:
5945 { \__fp_trap_overflow_set:N \use_none:nnnnn }
5946 \cs_new_protected:Npn \__fp_trap_overflow_set_none:
5947 { \__fp_trap_overflow_set:N \use_none:nnnnnnn }
5948 \cs_new_protected:Npn \__fp_trap_overflow_set:N #1
```

```
5949 { \__fp_trap_overflow_set:NnNn #1 { overflow } \__fp_inf_fp:N { inf } }
5950 \cs_new_protected:Npn \__fp_trap_underflow_set_error:
5951 { \__fp_trap_underflow_set:N \prg_do_nothing: }
5952 \cs_new_protected:Npn \__fp_trap_underflow_set_flag:
5953 { \__fp_trap_underflow_set:N \use_none:nnnnn }
5954 \cs_new_protected:Npn \__fp_trap_underflow_set_none:
5955 { \__fp_trap_underflow_set:N \use_none:nnnnnnn }
5956 \cs_new_protected:Npn \__fp_trap_underflow_set:N #1
5957 { \__fp_trap_overflow_set:NnNn #1 { underflow } \__fp_zero_fp:N { 0 } }
5958 \cs_new_protected:Npn \__fp_trap_overflow_set:NnNn #1#2#3#4
5959 {
5960   \exp_args:Nno \use:n
5961     { \cs_set:cpn { __fp_ #2 :w } \s__fp \__fp_chk:w ##1##2##3\__fp_sep: }
5962     {
5963       #1
5964       \__fp_error:nffn
5965         { flow \if_meaning:w 1 ##1 -to \fi: }
5966         { \fp_to_tl:n { \s__fp \__fp_chk:w ##1##2##3\__fp_sep: } }
5967         { \token_if_eq_meaning:NNF 0 ##2 { - } #4 }
5968         {#2}
5969         \flag_ensure_raised:c { l_fp_#2_flag }
5970         #3 ##2
5971     }
5972 }
5973 \cs_new:Npn \__fp_invalid_operation:nnw #1#2#3\__fp_sep: { }
5974 \cs_new:Npn \__fp_invalid_operation_o:Nww #1#2\__fp_sep: #3\__fp_sep: { }
5975 \cs_new:Npn \__fp_invalid_operation_tl_o:ff #1 #2 { }
5976 \cs_new:Npn \__fp_division_by_zero_o:Nnw #1#2#3\__fp_sep: { }
5977 \cs_new:Npn \__fp_division_by_zero_o:NNww #1#2#3\__fp_sep: #4\__fp_sep: { }
5978 \cs_new:Npn \__fp_overflow:w { }
5979 \cs_new:Npn \__fp_underflow:w { }
5980 \fp_trap:nn { invalid_operation } { error }
5981 \fp_trap:nn { division_by_zero } { flag }
5982 \fp_trap:nn { overflow } { flag }
5983 \fp_trap:nn { underflow } { flag }
5984 \cs_new:Npn \__fp_invalid_operation_o:nw
5985 { \__fp_invalid_operation:nnw { \exp_after:wN \c_nan_fp } }
5986 \cs_generate_variant:Nn \__fp_invalid_operation_o:nw { f }
5987 \cs_new:Npn \__fp_error:nnnn
5988 { \msg_expandable_error:nnnnn { fp } }
5989 \cs_generate_variant:Nn \__fp_error:nnnn { nnf, nff , nfff }
5990 \cs_new:Npn \__fp_error_num_args:nnnn #1#2#3#4
5991 {
5992   \int_compare:nNnTF {#2} = {#3}
5993     { \msg_expandable_error:nnnnn { fp } { num-args-eq } {#1} {#2} {#4} }
5994     { \msg_expandable_error:nnnnnn { fp } { num-args } {#1} {#2} {#3} {#4} }
```



```
5995 }
5996 \cs_generate_variant:Nn \__fp_error_num_args:nnnn { ffff }
5997 \msg_new:nnn { fp } { num-args-eq }
5998 { #1()~needs~#2~arguments,~got~#3. }
5999 \msg_new:nnn { fp } { num-args }
6000 { #1()~needs~#2~to~#3~arguments,~got~#4. }
6001 \msg_new:nnnn { fp } { unknown-fpu-exception }
6002 {
6003   The~FPU~exception~'#1'~is~not~known:~
6004   that~trap~will~never~be~triggered.
6005 }
6006 {
6007   The~only~exceptions~to~which~traps~can~be~attached~are \\
6008   \iow_indent:n
6009   {
6010     * ~ invalid_operation \\
6011     * ~ division_by_zero \\
6012     * ~ overflow \\
6013     * ~ underflow
6014   }
6015 }
6016 \msg_new:nnnn { fp } { unknown-fpu-trap-type }
6017 { The~FPU~trap~type~'#2'~is~not~known. }
6018 {
6019   The~trap~type~must~be~one~of \\
6020   \iow_indent:n
6021   {
6022     * ~ error \\
6023     * ~ flag \\
6024     * ~ none
6025   }
6026 }
6027 \msg_new:nnn { fp } { flow }
6028 { An ~ #3 ~ occurred. }
6029 \msg_new:nnn { fp } { flow-to }
6030 { #1 ~ #3 ed ~ to ~ #2 . }
6031 \msg_new:nnn { fp } { zero-div }
6032 { Division~by~zero~in~ #1 (#2) }
6033 \msg_new:nnn { fp } { zero-div-ii }
6034 { Division~by~zero~in~ (#1) #3 (#2) }
6035 \msg_new:nnn { fp } { invalid }
6036 { Invalid~operation~ #1 (#2) }
6037 \msg_new:nnn { fp } { invalid-ii }
6038 { Invalid~operation~ (#1) #3 (#2) }
6039 \msg_new:nnn { fp } { unknown-type }
6040 { Unknown~type~for~'#1' }
```



```
6041 %% File: l3fp-round.dtx 1604
6042 \cs_new:Npn \__fp_parse_word_trunc:N 1604
6043 { \__fp_parse_function:NNN \__fp_round_o:Nw \__fp_round_to_zero:NNN } 1604
6044 \cs_new:Npn \__fp_parse_word_floor:N 1604
6045 { \__fp_parse_function:NNN \__fp_round_o:Nw \__fp_round_to_ninf:NNN } 1604
6046 \cs_new:Npn \__fp_parse_word_ceil:N 1604
6047 { \__fp_parse_function:NNN \__fp_round_o:Nw \__fp_round_to_pinf:NNN } 1604
6048 \cs_new:Npn \__fp_parse_word_round:N #1#2 1604
6049 { 1604
6050 \__fp_parse_function:NNN 1605
6051 \__fp_round_o:Nw \__fp_round_to_nearest:NNN #1 1605
6052 #2 1605
6053 } 1605
6054 \cs_new:Npn \__fp_parse_round:Nw #1 #2 \__fp_round_to_nearest:NNN #3#4 1605
6055 { #2 #1 #3 } 1605
6056 \int_const:Nn \c__fp_five_int { 5 } 1605
6057 \cs_new:Npn \__fp_round_return_one: 1605
6058 { \exp_after:wN 1 \exp_after:wN \exp_stop_f: \exp:w } 1605
6059 \cs_new:Npn \__fp_round_to_ninf:NNN #1 #2 #3 1605
6060 { 1606
6061 \if_meaning:w 2 #1 1606
6062 \if_int_compare:w #3 > \c_zero_int 1606
6063 \__fp_round_return_one: 1606
6064 \fi: 1606
6065 \fi: 1606
6066 \c_zero_int 1606
6067 } 1606
6068 \cs_new:Npn \__fp_round_to_zero:NNN #1 #2 #3 { \c_zero_int } 1606
6069 \cs_new:Npn \__fp_round_to_pinf:NNN #1 #2 #3 1606
6070 { 1607
6071 \if_meaning:w 0 #1 1607
6072 \if_int_compare:w #3 > \c_zero_int 1607
6073 \__fp_round_return_one: 1607
6074 \fi: 1607
6075 \fi: 1607
6076 \c_zero_int 1607
6077 } 1607
6078 \cs_new:Npn \__fp_round_to_nearest:NNN #1 #2 #3 1607
6079 { 1607
6080 \if_int_compare:w #3 > \c__fp_five_int 1608
6081 \__fp_round_return_one: 1608
6082 \else: 1608
6083 \if_meaning:w 5 #3 1608
6084 \if_int_odd:w #2 \exp_stop_f: 1608
6085 \__fp_round_return_one: 1608
6086 \fi: 1608
```

6087	\fi:	1608
6088	\fi:	1608
6089	\c_zero_int	1608
6090	}	1609
6091	\cs_new:Npn __fp_round_to_nearest_ninf:NNN #1 #2 #3	1609
6092	{	1609
6093	\if_int_compare:w #3 > \c__fp_five_int	1609
6094	__fp_round_return_one:	1609
6095	\else:	1609
6096	\if_meaning:w 5 #3	1609
6097	\if_meaning:w 2 #1	1609
6098	__fp_round_return_one:	1609
6099	\fi:	1609
6100	\fi:	1610
6101	\fi:	1610
6102	\c_zero_int	1610
6103	}	1610
6104	\cs_new:Npn __fp_round_to_nearest_zero:NNN #1 #2 #3	1610
6105	{	1610
6106	\if_int_compare:w #3 > \c__fp_five_int	1610
6107	__fp_round_return_one:	1610
6108	\fi:	1610
6109	\c_zero_int	1610
6110	}	1611
6111	\cs_new:Npn __fp_round_to_nearest_pinf:NNN #1 #2 #3	1611
6112	{	1611
6113	\if_int_compare:w #3 > \c__fp_five_int	1611
6114	__fp_round_return_one:	1611
6115	\else:	1611
6116	\if_meaning:w 5 #3	1611
6117	\if_meaning:w 0 #1	1611
6118	__fp_round_return_one:	1611
6119	\fi:	1611
6120	\fi:	1612
6121	\fi:	1612
6122	\c_zero_int	1612
6123	}	1612
6124	\cs_new_eq:NN __fp_round:NNN __fp_round_to_nearest:NNN	1612
6125	\cs_new:Npn __fp_round_s:NNNw #1 #2 #3 #4__fp_sep:	1612
6126	{	1612
6127	\exp_after:wN __fp_round:NNN	1612
6128	\exp_after:wN #1	1612
6129	\exp_after:wN #2	1612
6130	\int_value:w __fp_int_eval:w	1613
6131	\if_int_odd:w 0 \if_meaning:w 0 #3 1 \fi:	1613
6132	\if_meaning:w 5 #3 1 \fi:	1613

```
6133         \exp_stop_f:
6134         \if_int_compare:w \__fp_int_eval:w #4 > \c_zero_int
6135             1 +
6136         \fi:
6137         \fi:
6138         #3
6139     \__fp_sep:
6140 }
6141 \cs_new:Npn \__fp_round_digit:Nw #1 #2\__fp_sep:
6142 {
6143     \if_int_odd:w \if_meaning:w 0 #1 1 \else:
6144         \if_meaning:w 5 #1 1 \else:
6145             0 \fi: \fi: \exp_stop_f:
6146     \if_int_compare:w \__fp_int_eval:w #2 > \c_zero_int
6147         \__fp_int_eval:w 1 +
6148     \fi:
6149     \fi:
6150     #1
6151 }
6152 \cs_new_eq:NN \__fp_round_to_ninf_neg:NNN \__fp_round_to_pinf:NNN
6153 \cs_new:Npn \__fp_round_to_zero_neg:NNN #1 #2 #3
6154 {
6155     \if_int_compare:w #3 > \c_zero_int
6156         \__fp_round_return_one:
6157     \fi:
6158     \c_zero_int
6159 }
6160 \cs_new_eq:NN \__fp_round_to_pinf_neg:NNN \__fp_round_to_ninf:NNN
6161 \cs_new_eq:NN \__fp_round_to_nearest_neg:NNN \__fp_round_to_nearest:NNN
6162 \cs_new_eq:NN \__fp_round_to_nearest_ninf_neg:NNN
6163     \__fp_round_to_nearest_pinf:NNN
6164 \cs_new:Npn \__fp_round_to_nearest_zero_neg:NNN #1 #2 #3
6165 {
6166     \if_int_compare:w #3 < \c__fp_five_int \else:
6167         \__fp_round_return_one:
6168     \fi:
6169     \c_zero_int
6170 }
6171 \cs_new_eq:NN \__fp_round_to_nearest_pinf_neg:NNN
6172     \__fp_round_to_nearest_ninf:NNN
6173 \cs_new_eq:NN \__fp_round_neg:NNN \__fp_round_to_nearest_neg:NNN
6174 \cs_new:Npn \__fp_round_o:Nw #1
6175 {
6176     \__fp_parse_function_all_fp_o:fnw
6177     { \__fp_round_name_from_cs:N #1 }
6178     { \__fp_round_aux_o:Nw #1 }
```

```
6179 }
6180 \cs_new:Npn \__fp_round_aux_o:Nw #1#2 @
6181 {
6182   \if_case:w
6183     \__fp_int_eval:w \__fp_array_count:n {#2} \__fp_int_eval_end:
6184     \__fp_round_no_arg_o:Nw #1 \exp:w
6185     \or: \__fp_round:Nwn #1 #2 {0} \exp:w
6186     \or: \__fp_round:Nww #1 #2 \exp:w
6187     \else: \__fp_round:Nwww #1 #2 @ \exp:w
6188     \fi:
6189     \exp_after:wN \exp_end:
6190 }
6191 \cs_new:Npn \__fp_round_no_arg_o:Nw #1
6192 {
6193   \cs_if_eq:NNTF #1 \__fp_round_to_nearest:NNN
6194   { \__fp_error_num_args:nnnn { round } { 1 } { 3 } { 0 } }
6195   {
6196     \__fp_error_num_args:ffff
6197     { \__fp_round_name_from_cs:N #1 } { 1 } { 2 } { 0 }
6198   }
6199   \exp_after:wN \c_nan_fp
6200 }
6201 \cs_new:Npn \__fp_round:Nwww
6202   #1#2 \__fp_sep: #3 \__fp_sep: \s_fp \__fp_chk:w #4#5#6 \__fp_sep: #7 @
6203 {
6204   \cs_if_eq:NNTF #1 \__fp_round_to_nearest:NNN
6205   {
6206     \tl_if_empty:nTF {#7}
6207     {
6208       \exp_args:Nc \__fp_round:Nww
6209       {
6210         \__fp_round_to_nearest
6211         \if_meaning:w 0 #4 _zero \else:
6212         \if_case:w #5 \exp_stop_f: _pinf \or: \else: _ninf \fi: \fi:
6213         :NNN
6214       }
6215       #2 \__fp_sep: #3 \__fp_sep:
6216     }
6217     {
6218       \__fp_error_num_args:ffff { round } { 1 } { 3 }
6219       { \int_eval:n { 3 + \__fp_array_count:n {#7} } }
6220       \exp_after:wN \c_nan_fp
6221     }
6222   }
6223   {
6224     \__fp_error_num_args:ffff
```

```

6225         { \__fp_round_name_from_cs:N #1 } { 1 } { 2 }
6226         { \int_eval:n { 3 + \__fp_array_count:n {#7} } }
6227         \exp_after:wN \c_nan_fp
6228     }
6229 }
6230 \cs_new:Npn \__fp_round_name_from_cs:N #1
6231 {
6232     \cs_if_eq:NNTF #1 \__fp_round_to_zero:NNN { trunc }
6233     {
6234         \cs_if_eq:NNTF #1 \__fp_round_to_ninf:NNN { floor }
6235         {
6236             \cs_if_eq:NNTF #1 \__fp_round_to_pinf:NNN { ceil }
6237             { round }
6238         }
6239     }
6240 }
6241 \cs_new:Npn \__fp_round:Nww #1#2 \__fp_sep: #3 \__fp_sep:
6242 {
6243     \__fp_small_int:wTF #3\__fp_sep: { \__fp_round:Nwn #1#2\__fp_sep: }
6244     {
6245         \if:w 3 \__fp_kind:w #3 \__fp_sep:
6246             \exp_after:wN \use_i:nn
6247         \else:
6248             \exp_after:wN \use_ii:nn
6249         \fi:
6250         { \exp_after:wN \c_nan_fp }
6251     }
6252     \__fp_invalid_operation_tl_o:ff
6253     { \__fp_round_name_from_cs:N #1 }
6254     { \__fp_array_to_clist:n { #2\__fp_sep: #3\__fp_sep: } }
6255 }
6256 }
6257 }
6258 \cs_new:Npn \__fp_round:Nwn #1 \s__fp \__fp_chk:w #2#3#4\__fp_sep: #5
6259 {
6260     \if_meaning:w 1 #2
6261         \exp_after:wN \__fp_round_normal:NwNNnw
6262         \exp_after:wN #1
6263         \int_value:w #5
6264     \else:
6265         \exp_after:wN \__fp_exp_after_o:w
6266     \fi:
6267     \s__fp \__fp_chk:w #2#3#4\__fp_sep:
6268 }
6269 \cs_new:Npn \__fp_round_normal:NwNNnw #1#2 \s__fp \__fp_chk:w 1#3#4#5\__fp_sep:
6270 {

```

```
6271 \__fp_decimate:nNnnnn { \c__fp_prec_int - #4 - #2 } 1627
6272 \__fp_round_normal:NnnwNNnn #5 #1 #3 {#4} {#2} 1627
6273 } 1627
6274 \cs_new:Npn \__fp_round_normal:NnnwNNnn #1#2#3#4\__fp_sep: #5#6 1627
6275 { 1627
6276 \exp_after:wN \__fp_round_normal:NNwNnn 1627
6277 \int_value:w \__fp_int_eval:w 1627
6278 \if_int_compare:w #2 > \c_zero_int 1627
6279 1 \int_value:w #2 1627
6280 \exp_after:wN \__fp_round_pack:Nw 1628
6281 \int_value:w \__fp_int_eval:w 1#3 + 1628
6282 \else: 1628
6283 \if_int_compare:w #3 > \c_zero_int 1628
6284 1 \int_value:w #3 + 1628
6285 \fi: 1628
6286 \fi: 1628
6287 \exp_after:wN #5 1628
6288 \exp_after:wN #6 1628
6289 \use_none:nnnnnnn #3 1628
6290 #1 1629
6291 \__fp_int_eval_end: 1629
6292 0000 0000 0000 0000 \__fp_sep: #6 1629
6293 } 1629
6294 \cs_new:Npn \__fp_round_pack:Nw #1 1629
6295 { \if_meaning:w 2 #1 + 1 \fi: \__fp_int_eval_end: } 1629
6296 \cs_new:Npn \__fp_round_normal:NNwNnn #1 #2 1629
6297 { 1629
6298 \if_meaning:w 0 #2 1629
6299 \exp_after:wN \__fp_round_special:NwwNnn 1629
6300 \exp_after:wN #1 1630
6301 \fi: 1630
6302 \__fp_pack_twice_four:wNNNNNNNN 1630
6303 \__fp_pack_twice_four:wNNNNNNNN 1630
6304 \__fp_round_normal_end:wwNnn 1630
6305 \__fp_sep: #2 1630
6306 } 1630
6307 \cs_new:Npn \__fp_round_normal_end:wwNnn #1\__fp_sep:#2\__fp_sep:#3#4#5 1630
6308 { 1630
6309 \exp_after:wN \__fp_exp_after_o:w \exp:w \exp_end_continue_f:w 1630
6310 \__fp_sanitizew #3 #4 \__fp_sep: #1 \__fp_sep: 1631
6311 } 1631
6312 \cs_new:Npn \__fp_round_special:NwwNnn #1#2\__fp_sep:#3\__fp_sep:#4#5#6 1631
6313 { 1631
6314 \if_meaning:w 0 #1 1631
6315 \__fp_case_return:nw 1631
6316 { \exp_after:wN \__fp_zero_fp:N \exp_after:wN #4 } 1631
```

```

6317 \else:
6318 \exp_after:wN \_fp_round_special_aux:Nw
6319 \exp_after:wN #4
6320 \int_value:w \_fp_int_eval:w 1
6321 \if_meaning:w 1 #1 -#6 \else: +#5 \fi:
6322 \fi:
6323 \_fp_sep:
6324 }
6325 \cs_new:Npn \_fp_round_special_aux:Nw #1#2\_fp_sep:
6326 {
6327 \exp_after:wN \_fp_exp_after_o:w \exp:w \exp_end_continue_f:w
6328 \_fp_sanitize:Nw #1#2\_fp_sep: {1000}{0000}{0000}{0000}\_fp_sep:
6329 }
6330 %% File: l3fp-parse.dtx
6331 \int_const:Nn \c\_fp_prec_func_int { 16 }
6332 \int_const:Nn \c\_fp_prec_hatii_int { 14 }
6333 \int_const:Nn \c\_fp_prec_hat_int { 13 }
6334 \int_const:Nn \c\_fp_prec_not_int { 12 }
6335 \int_const:Nn \c\_fp_prec_juxt_int { 11 }
6336 \int_const:Nn \c\_fp_prec_times_int { 10 }
6337 \int_const:Nn \c\_fp_prec_plus_int { 9 }
6338 \int_const:Nn \c\_fp_prec_comp_int { 7 }
6339 \int_const:Nn \c\_fp_prec_and_int { 6 }
6340 \int_const:Nn \c\_fp_prec_or_int { 5 }
6341 \int_const:Nn \c\_fp_prec_quest_int { 4 }
6342 \int_const:Nn \c\_fp_prec_colon_int { 3 }
6343 \int_const:Nn \c\_fp_prec_comma_int { 2 }
6344 \int_const:Nn \c\_fp_prec_tuple_int { 1 }
6345 \int_const:Nn \c\_fp_prec_end_int { 0 }
6346 \cs_new:Npn \_fp_parse_expand:w #1 { \exp_end_continue_f:w #1 }
6347 \cs_new:Npn \_fp_parse_return_sep:w
6348 #1 \fi: \_fp_parse_expand:w { \fi: \_fp_sep: #1 }
6349 \cs_set_protected:Npn \_fp_tmp:w #1 #2 #3
6350 {
6351 \cs_new:cpn { \_fp_parse_digits_ #1 :N } ##1
6352 {
6353 \if_int_compare:w 9 < 1 \token_to_str:N ##1 \exp_stop_f:
6354 \token_to_str:N ##1 \exp_after:wN #2 \exp:w
6355 \else:
6356 \_fp_parse_return_sep:w #3 ##1
6357 \fi:
6358 \_fp_parse_expand:w
6359 }
6360 }
6361 \_fp_tmp:w {vii} \_fp_parse_digits_vi:N { 0000000 \_fp_sep: 7 }
6362 \_fp_tmp:w {vi} \_fp_parse_digits_v:N { 000000 \_fp_sep: 6 }

```



```
6363 \__fp_tmp:w {v} \__fp_parse_digits_iv:N { 00000 \__fp_sep: 5 }
6364 \__fp_tmp:w {iv} \__fp_parse_digits_iii:N { 0000 \__fp_sep: 4 }
6365 \__fp_tmp:w {iii} \__fp_parse_digits_ii:N { 000 \__fp_sep: 3 }
6366 \__fp_tmp:w {ii} \__fp_parse_digits_i:N { 00 \__fp_sep: 2 }
6367 \__fp_tmp:w {i} \__fp_parse_digits_:N { 0 \__fp_sep: 1 }
6368 \cs_new:Npn \__fp_parse_digits_:N { \__fp_sep: \__fp_sep: 0 }
6369 \cs_new:Npn \__fp_parse_one:Nw #1 #2
6370 {
6371 \if_catcode:w \scan_stop: \exp_not:N #2
6372 \exp_after:wN \if_meaning:w \exp_not:N #2 #2 \else:
6373 \exp_after:wN \reverse_if:N
6374 \fi:
6375 \if_meaning:w \scan_stop: #2
6376 \exp_after:wN \exp_after:wN
6377 \exp_after:wN \__fp_parse_one_fp:NN
6378 \else:
6379 \exp_after:wN \exp_after:wN
6380 \exp_after:wN \__fp_parse_one_register:NN
6381 \fi:
6382 \else:
6383 \if_int_compare:w 9 < 1 \token_to_str:N #2 \exp_stop_f:
6384 \exp_after:wN \exp_after:wN
6385 \exp_after:wN \__fp_parse_one_digit:NN
6386 \else:
6387 \exp_after:wN \exp_after:wN
6388 \exp_after:wN \__fp_parse_one_other:NN
6389 \fi:
6390 \fi:
6391 #1 #2
6392 }
6393 \cs_new:Npn \__fp_parse_one_fp:NN #1
6394 {
6395 \__fp_exp_after_any_f:nw
6396 {
6397 \exp_after:wN \__fp_parse_infix:NN
6398 \exp_after:wN #1 \exp:w \__fp_parse_expand:w
6399 }
6400 }
6401 \cs_new:Npn \__fp_exp_after_expr_mark_f:nw #1
6402 {
6403 \int_case:nnF { \exp_after:wN \use_i:nnn \use_none:nnn #1 }
6404 {
6405 \c__fp_prec_comma_int { }
6406 \c__fp_prec_tuple_int { }
6407 \c__fp_prec_end_int
6408 {
```

```
6409         \exp_after:wN \c__fp_empty_tuple_fp
6410         \exp:w \exp_end_continue_f:w
6411     }
6412 }
6413 {
6414     \msg_expandable_error:nn { fp } { early-end }
6415     \exp_after:wN \c_nan_fp \exp:w \exp_end_continue_f:w
6416 }
6417 #1
6418 }
6419 \cs_new:cpn { __fp_exp_after_?_f:nw } #1#2
6420 {
6421     \msg_expandable_error:nnn { kernel } { bad-variable }
6422     {#2}
6423     \exp_after:wN \c_nan_fp \exp:w \exp_end_continue_f:w #1
6424 }
6425 \cs_set_protected:Npn \__fp_tmp:w #1
6426 {
6427     \cs_if_exist:NT #1
6428     {
6429         \cs_gset:cpn { __fp_exp_after_?_f:nw } ##1##2
6430         {
6431             \exp_after:wN \c_nan_fp \exp:w \exp_end_continue_f:w ##1
6432             \str_if_eq:nnTF {##2} { \protect }
6433             {
6434                 \cs_if_eq:NNTF ##2 #1 { \use_i:nn } { \use:n }
6435                 {
6436                     \msg_expandable_error:nnn { fp }
6437                     { robust-cmd }
6438                 }
6439             }
6440             {
6441                 \msg_expandable_error:nnn { kernel }
6442                 { bad-variable } {##2}
6443             }
6444         }
6445     }
6446 }
6447 \exp_args:Nc \__fp_tmp:w { @unexpandable@protect }
6448 \cs_new:Npn \__fp_parse_one_register:NN #1#2
6449 {
6450     \exp_after:wN \__fp_parse_infix_after_operand:NwN
6451     \exp_after:wN #1
6452     \exp:w \exp_end_continue_f:w
6453     \__fp_parse_one_register_special:N #2
6454     \exp_after:wN \__fp_parse_one_register_aux:Nw
```

```

6455     \exp_after:wN #2
6456     \int_value:w
6457     \exp_after:wN \__fp_parse_exponent:N
6458     \exp:w \__fp_parse_expand:w
6459 }
6460 \cs_new:Npe \__fp_parse_one_register_aux:Nw #1
6461 {
6462     \exp_not:n
6463     {
6464         \exp_after:wN \use:nn
6465         \exp_after:wN \__fp_parse_one_register_auxii:wwwNw
6466     }
6467     \exp_not:N \exp_after:wN { \exp_not:N \tex_the:D #1 }
6468     \__fp_sep: \exp_not:N \__fp_parse_one_register_dim:ww
6469     \tl_to_str:n { pt } \__fp_sep: \exp_not:N \__fp_parse_one_register_mu:www
6470     . \tl_to_str:n { pt } \__fp_sep: \exp_not:N \__fp_parse_one_register_int:www
6471     \s__fp_stop
6472 }
6473 \exp_args:Nno \use:nn
6474 { \cs_new:Npn \__fp_parse_one_register_auxii:wwwNw #1 . #2 }
6475 { \tl_to_str:n { pt } #3 \__fp_sep: #4#5 \s__fp_stop }
6476 { #4 #1.#2\__fp_sep: }
6477 \exp_args:Nno \use:nn
6478 { \cs_new:Npn \__fp_parse_one_register_mu:www #1 }
6479 { \tl_to_str:n { mu } \__fp_sep: #2 \__fp_sep: }
6480 { \__fp_parse_one_register_dim:ww #1 \__fp_sep: }
6481 \cs_new:Npn \__fp_parse_one_register_int:www #1\__fp_sep: #2.\__fp_sep: #3\__fp_sep:
6482 { \__fp_parse:n { #1 e #3 } }
6483 \cs_new:Npn \__fp_parse_one_register_dim:ww #1\__fp_sep: #2\__fp_sep:
6484 {
6485     \exp_after:wN \__fp_from_dim_test:ww
6486     \int_value:w #2 \exp_after:wN ,
6487     \int_value:w \dim_to_decimal_in_sp:n { #1 pt } \__fp_sep:
6488 }
6489 \cs_new:Npn \__fp_parse_one_register_special:N #1
6490 {
6491     \if_meaning:w \box_wd:N #1 \__fp_parse_one_register_wd:w \fi:
6492     \if_meaning:w \box_ht:N #1 \__fp_parse_one_register_wd:w \fi:
6493     \if_meaning:w \box_dp:N #1 \__fp_parse_one_register_wd:w \fi:
6494     \if_meaning:w \infty #1
6495         \__fp_parse_one_register_math:NNw \infty #1
6496     \fi:
6497     \if_meaning:w \pi #1
6498         \__fp_parse_one_register_math:NNw \pi #1
6499     \fi:
6500 }

```

```
6501 \cs_new:Npn \__fp_parse_one_register_math:NNw 16501
6502   #1#2#3#4 \__fp_parse_expand:w 16502
6503   { 16503
6504   #3 16504
6505   \str_if_eq:nnTF {#1} {#2} 16505
6506   { 16506
6507     \msg_expandable_error:nnn 16507
6508     { fp } { infinity-pi } {#1} 16508
6509     \c_nan_fp 16509
6510   } 16510
6511   { #4 \__fp_parse_expand:w } 16511
6512 } 16512
6513 \cs_new:Npn \__fp_parse_one_register_wd:w 16513
6514   #1#2 \exp_after:wN #3#4 \__fp_parse_expand:w 16514
6515   { 16515
6516   #1 16516
6517   \exp_after:wN \__fp_parse_one_register_wd:Nw 16517
6518   #4 \__fp_parse_expand:w e 16518
6519   } 16519
6520 \cs_new:Npn \__fp_parse_one_register_wd:Nw #1#2 \__fp_sep: 16520
6521   { 16521
6522   \exp_after:wN \__fp_from_dim_test:ww 16522
6523   \exp_after:wN 0 \exp_after:wN , 16523
6524   \int_value:w \dim_to_decimal_in_sp:n { #1 #2 } \__fp_sep: 16524
6525   } 16525
6526 \cs_new:Npn \__fp_parse_one_digit:NN #1 16526
6527   { 16527
6528   \exp_after:wN \__fp_parse_infix_after_operand:NwN 16528
6529   \exp_after:wN #1 16529
6530   \exp:w \exp_end_continue_f:w 16530
6531   \exp_after:wN \__fp_sanitise:wN 16531
6532   \int_value:w \__fp_int_eval:w 0 \__fp_parse_trim_zeros:N 16532
6533   } 16533
6534 \cs_new:Npn \__fp_parse_one_other:NN #1 #2 16534
6535   { 16535
6536   \if_int_compare:w 16536
6537     \__fp_int_eval:w 16537
6538     ( `#2 \if_int_compare:w `#2 > `Z - 32 \fi: ) / 26 16538
6539     = 3 \exp_stop_f: 16539
6540     \exp_after:wN \__fp_parse_word:Nw 16540
6541     \exp_after:wN #1 16541
6542     \exp_after:wN #2 16542
6543     \exp:w \exp_after:wN \__fp_parse_letters:N 16543
6544     \exp:w 16544
6545   \else: 16545
6546     \exp_after:wN \__fp_parse_prefix:NNN 16546
```

```
6547 \exp_after:wN #1 1654
6548 \exp_after:wN #2 1654
6549 \cs:w 1654
6550 __fp_parse_prefix_ \token_to_str:N #2 :Nw 1655
6551 \exp_after:wN 1655
6552 \cs_end: 1655
6553 \exp:w 1655
6554 \fi: 1655
6555 \__fp_parse_expand:w 1655
6556 } 1655
6557 \cs_new:Npn \__fp_parse_word:Nw #1#2\__fp_sep: 1655
6558 { 1655
6559 \cs_if_exist_use:cF { __fp_parse_word_#2:N } 1655
6560 { 1656
6561 \cs_if_exist_use:cF 1656
6562 { __fp_parse_caseless_ \str_casefold:n {#2} :N } 1656
6563 { 1656
6564 \msg_expandable_error:nnn 1656
6565 { fp } { unknown-fp-word } {#2} 1656
6566 \exp_after:wN \c_nan_fp \exp:w \exp_end_continue_f:w 1656
6567 \__fp_parse_infix:NN 1656
6568 } 1656
6569 } 1656
6570 #1 1657
6571 } 1657
6572 \cs_new:Npn \__fp_parse_letters:N #1 1657
6573 { 1657
6574 \exp_end_continue_f:w 1657
6575 \if_int_compare:w 1657
6576 \if_catcode:w \scan_stop: \exp_not:N #1 1657
6577 0 1657
6578 \else: 1657
6579 \__fp_int_eval:w 1657
6580 ( `#1 \if_int_compare:w `#1 > `Z - 32 \fi: ) / 26 1658
6581 \fi: 1658
6582 = 3 \exp_stop_f: 1658
6583 \exp_after:wN #1 1658
6584 \exp:w \exp_after:wN \__fp_parse_letters:N 1658
6585 \exp:w 1658
6586 \else: 1658
6587 \__fp_parse_return_sep:w #1 1658
6588 \fi: 1658
6589 \__fp_parse_expand:w 1658
6590 } 1659
6591 \cs_new:Npn \__fp_parse_prefix:NNN #1#2#3 1659
6592 { 1659
```

```
6593 \if_meaning:w \scan_stop: #3 16593
6594 \exp_after:wN \__fp_parse_prefix_unknown:NNN 16594
6595 \exp_after:wN #2 16595
6596 \fi: 16596
6597 #3 #1 16597
6598 } 16598
6599 \cs_new:Npn \__fp_parse_prefix_unknown:NNN #1#2#3 16599
6600 { 16600
6601 \cs_if_exist:cTF { __fp_parse_infix_ \token_to_str:N #1 :N } 16601
6602 { 16602
6603 \msg_expandable_error:nnn 16603
6604 { fp } { missing-number } {#1} 16604
6605 \exp_after:wN \c_nan_fp \exp:w \exp_end_continue_f:w 16605
6606 \__fp_parse_infix:NN #3 #1 16606
6607 } 16607
6608 { 16608
6609 \msg_expandable_error:nnn 16609
6610 { fp } { unknown-symbol } {#1} 16610
6611 \__fp_parse_one:Nw #3 16611
6612 } 16612
6613 } 16613
6614 \cs_new:Npn \__fp_parse_trim_zeros:N #1 16614
6615 { 16615
6616 \if:w 0 \exp_not:N #1 16616
6617 \exp_after:wN \__fp_parse_trim_zeros:N 16617
6618 \exp:w 16618
6619 \else: 16619
6620 \if:w . \exp_not:N #1 16620
6621 \exp_after:wN \__fp_parse_strim_zeros:N 16621
6622 \exp:w 16622
6623 \else: 16623
6624 \__fp_parse_trim_end:w #1 16624
6625 \fi: 16625
6626 \fi: 16626
6627 \__fp_parse_expand:w 16627
6628 } 16628
6629 \cs_new:Npn \__fp_parse_trim_end:w #1 \fi: \fi: \__fp_parse_expand:w 16629
6630 { 16630
6631 \fi: 16631
6632 \fi: 16632
6633 \if_int_compare:w 9 < 1 \token_to_str:N #1 \exp_stop_f: 16633
6634 \exp_after:wN \__fp_parse_large:N 16634
6635 \else: 16635
6636 \exp_after:wN \__fp_parse_zero: 16636
6637 \fi: 16637
6638 #1 16638
```

```
6639 } 16639
6640 \cs_new:Npn \__fp_parse_strim_zeros:N #1 16640
6641 { 16641
6642 \if:w 0 \exp_not:N #1 16642
6643 - 1 16643
6644 \exp_after:wN \__fp_parse_strim_zeros:N \exp:w 16644
6645 \else: 16645
6646 \__fp_parse_strim_end:w #1 16646
6647 \fi: 16647
6648 \__fp_parse_expand:w 16648
6649 } 16649
6650 \cs_new:Npn \__fp_parse_strim_end:w #1 \fi: \__fp_parse_expand:w 16650
6651 { 16651
6652 \fi: 16652
6653 \if_int_compare:w 9 < 1 \token_to_str:N #1 \exp_stop_f: 16653
6654 \exp_after:wN \__fp_parse_small:N 16654
6655 \else: 16655
6656 \exp_after:wN \__fp_parse_zero: 16656
6657 \fi: 16657
6658 #1 16658
6659 } 16659
6660 \cs_new:Npn \__fp_parse_zero: 16660
6661 { 16661
6662 \exp_after:wN \__fp_sep: \exp_after:wN 1 16662
6663 \int_value:w \__fp_parse_exponent:N 16663
6664 } 16664
6665 \cs_new:Npn \__fp_parse_small:N #1 16665
6666 { 16666
6667 \exp_after:wN \__fp_parse_pack_leading:NNNNNww 16667
6668 \int_value:w \__fp_int_eval:w 1 \token_to_str:N #1 16668
6669 \exp_after:wN \__fp_parse_small_leading:wwNN 16669
6670 \int_value:w 1 16670
6671 \exp_after:wN \__fp_parse_digits_vii:N 16671
6672 \exp:w \__fp_parse_expand:w 16672
6673 } 16673
6674 \cs_new:Npn \__fp_parse_small_leading:wwNN 1 #1 \__fp_sep: #2\__fp_sep: #3 #4 16674
6675 { 16675
6676 #1 #2 16676
6677 \exp_after:wN \__fp_parse_pack_trailing:NNNNNww 16677
6678 \exp_after:wN 0 16678
6679 \int_value:w \__fp_int_eval:w 1 16679
6680 \if_int_compare:w 9 < 1 \token_to_str:N #4 \exp_stop_f: 16680
6681 \token_to_str:N #4 16681
6682 \exp_after:wN \__fp_parse_small_trailing:wwNN 16682
6683 \int_value:w 1 16683
6684 \exp_after:wN \__fp_parse_digits_vi:N 16684
```



```
6685         \exp:w                                     16685
6686     \else:                                           16686
6687         0000 0000 \__fp_parse_exponent:Nw #4       16687
6688     \fi:                                             16688
6689     \__fp_parse_expand:w                             16689
6690 }                                                    16690
6691 \cs_new:Npn \__fp_parse_small_trailing:wwNN 1 #1 \__fp_sep: #2\__fp_sep: #3 #4 16691
6692 {                                                    16692
6693     #1 #2                                           16693
6694     \if_int_compare:w 9 < 1 \token_to_str:N #4 \exp_stop_f: 16694
6695         \token_to_str:N #4                         16695
6696         \exp_after:wN \__fp_parse_small_round:NN 16696
6697         \exp_after:wN #4                           16697
6698         \exp:w                                       16698
6699     \else:                                           16699
6700         0 \__fp_parse_exponent:Nw #4               16700
6701     \fi:                                             16701
6702     \__fp_parse_expand:w                             16702
6703 }                                                    16703
6704 \cs_new:Npn \__fp_parse_pack_trailing:NNNNNNww      16704
6705     #1 #2 #3#4#5#6 #7\__fp_sep: #8 \__fp_sep:      16705
6706 {                                                    16706
6707     \if_meaning:w 2 #2 + 1 \fi:                    16707
6708     \__fp_sep: #8 + #1 \__fp_sep: {#3#4#5#6} {#7}\__fp_sep: 16708
6709 }                                                    16709
6710 \cs_new:Npn \__fp_parse_pack_leading:NNNNNww #1 #2#3#4#5 #6\__fp_sep: #7\__fp_sep: 16710
6711 {                                                    16711
6712     + #7                                             16712
6713     \if_meaning:w 2 #1 \__fp_parse_pack_carry:w \fi: 16713
6714     \__fp_sep: 0 {#2#3#4#5} {#6}                   16714
6715 }                                                    16715
6716 \cs_new:Npn \__fp_parse_pack_carry:w \fi: \__fp_sep: 0 #1 16716
6717 { \fi: + 1 \__fp_sep: 0 {1000} }                   16717
6718 \cs_new:Npn \__fp_parse_large:N #1                  16718
6719 {                                                    16719
6720     \exp_after:wN \__fp_parse_large_leading:wwNN 16720
6721     \int_value:w 1 \token_to_str:N #1              16721
6722     \exp_after:wN \__fp_parse_digits_vii:N         16722
6723     \exp:w \__fp_parse_expand:w                    16723
6724 }                                                    16724
6725 \cs_new:Npn \__fp_parse_large_leading:wwNN 1 #1 \__fp_sep: #2\__fp_sep: #3 #4 16725
6726 {                                                    16726
6727     + \c__fp_half_prec_int - #3                    16727
6728     \exp_after:wN \__fp_parse_pack_leading:NNNNNww 16728
6729     \int_value:w \__fp_int_eval:w 1 #1             16729
6730     \if_int_compare:w 9 < 1 \token_to_str:N #4 \exp_stop_f: 16730
```

```
6731 \exp_after:wN \__fp_parse_large_trailing:wwNN 1673
6732 \int_value:w 1 \token_to_str:N #4 1673
6733 \exp_after:wN \__fp_parse_digits_vi:N 1673
6734 \exp:w 1673
6735 \else: 1673
6736 \if:w . \exp_not:N #4 1673
6737 \exp_after:wN \__fp_parse_small_leading:wwNN 1673
6738 \int_value:w 1 1673
6739 \cs:w 1673
6740 __fp_parse_digits_ 1674
6741 \__fp_int_to_roman:w #3 1674
6742 :N \exp_after:wN 1674
6743 \cs_end: 1674
6744 \exp:w 1674
6745 \else: 1674
6746 #2 1674
6747 \exp_after:wN \__fp_parse_pack_trailing:NNNNNNww 1674
6748 \exp_after:wN 0 1674
6749 \int_value:w 1 0000 0000 1674
6750 \__fp_parse_exponent:Nw #4 1675
6751 \fi: 1675
6752 \fi: 1675
6753 \__fp_parse_expand:w 1675
6754 } 1675
6755 \cs_new:Npn \__fp_parse_large_trailing:wwNN 1 #1 \__fp_sep: #2\__fp_sep: #3 #4 1675
6756 { 1675
6757 \if_int_compare:w 9 < 1 \token_to_str:N #4 \exp_stop_f: 1675
6758 \exp_after:wN \__fp_parse_pack_trailing:NNNNNNww 1675
6759 \exp_after:wN \c__fp_half_prec_int 1675
6760 \int_value:w \__fp_int_eval:w 1 #1 \token_to_str:N #4 1676
6761 \exp_after:wN \__fp_parse_large_round:NN 1676
6762 \exp_after:wN #4 1676
6763 \exp:w 1676
6764 \else: 1676
6765 \exp_after:wN \__fp_parse_pack_trailing:NNNNNNww 1676
6766 \int_value:w \__fp_int_eval:w 7 - #3 \exp_stop_f: 1676
6767 \int_value:w \__fp_int_eval:w 1 #1 1676
6768 \if:w . \exp_not:N #4 1676
6769 \exp_after:wN \__fp_parse_small_trailing:wwNN 1676
6770 \int_value:w 1 1677
6771 \cs:w 1677
6772 __fp_parse_digits_ 1677
6773 \__fp_int_to_roman:w #3 1677
6774 :N \exp_after:wN 1677
6775 \cs_end: 1677
6776 \exp:w 1677
```

```
6777         \else:
6778             #2 0 \__fp_parse_exponent:Nw #4
6779         \fi:
6780     \fi:
6781     \__fp_parse_expand:w
6782 }
6783 \cs_new:Npn \__fp_parse_round_loop:N #1
6784 {
6785     \if_int_compare:w 9 < 1 \token_to_str:N #1 \exp_stop_f:
6786         + 1
6787         \if:w 0 \token_to_str:N #1
6788             \exp_after:wN \__fp_parse_round_loop:N
6789             \exp:w
6790         \else:
6791             \exp_after:wN \__fp_parse_round_up:N
6792             \exp:w
6793         \fi:
6794     \else:
6795         \__fp_parse_return_sep:w 0 #1
6796     \fi:
6797     \__fp_parse_expand:w
6798 }
6799 \cs_new:Npn \__fp_parse_round_up:N #1
6800 {
6801     \if_int_compare:w 9 < 1 \token_to_str:N #1 \exp_stop_f:
6802         + 1
6803         \exp_after:wN \__fp_parse_round_up:N
6804         \exp:w
6805     \else:
6806         \__fp_parse_return_sep:w 1 #1
6807     \fi:
6808     \__fp_parse_expand:w
6809 }
6810 \cs_new:Npn \__fp_parse_round_after:wN #1\__fp_sep: #2
6811 {
6812     + #2 \exp_after:wN \__fp_sep:
6813     \int_value:w \__fp_int_eval:w #1 + \__fp_parse_exponent:N
6814 }
6815 \cs_new:Npn \__fp_parse_small_round:NN #1#2
6816 {
6817     \if_int_compare:w 9 < 1 \token_to_str:N #2 \exp_stop_f:
6818         +
6819         \exp_after:wN \__fp_round_s:NNNw
6820         \exp_after:wN 0
6821         \exp_after:wN #1
6822         \exp_after:wN #2
```

```
6823         \int_value:w \__fp_int_eval:w                                     16823
6824         \exp_after:wN \__fp_parse_round_after:wN                       16824
6825         \int_value:w \__fp_int_eval:w 0 * \__fp_int_eval:w 0           16825
6826         \exp_after:wN \__fp_parse_round_loop:N                         16826
6827         \exp:w                                                            16827
6828     \else:                                                                16828
6829         \__fp_parse_exponent:Nw #2                                       16829
6830     \fi:                                                                  16830
6831     \__fp_parse_expand:w                                                  16831
6832 }                                                                           16832
6833 \cs_new:Npn \__fp_parse_large_round:NN #1#2                             16833
6834 {                                                                           16834
6835     \if_int_compare:w 9 < 1 \token_to_str:N #2 \exp_stop_f:           16835
6836     +                                                                      16836
6837     \exp_after:wN \__fp_round_s:NNNw                                     16837
6838     \exp_after:wN 0                                                       16838
6839     \exp_after:wN #1                                                       16839
6840     \exp_after:wN #2                                                       16840
6841     \int_value:w \__fp_int_eval:w                                         16841
6842     \exp_after:wN \__fp_parse_large_round_aux:wNN                       16842
6843     \int_value:w \__fp_int_eval:w 1                                       16843
6844     \exp_after:wN \__fp_parse_round_loop:N                               16844
6845 \else: %^^A could be dot, or e, or other                                16845
6846     \exp_after:wN \__fp_parse_large_round_test:NN                       16846
6847     \exp_after:wN #1                                                       16847
6848     \exp_after:wN #2                                                       16848
6849 \fi:                                                                      16849
6850 }                                                                           16850
6851 \cs_new:Npn \__fp_parse_large_round_test:NN #1#2                       16851
6852 {                                                                           16852
6853     \if:w . \exp_not:N #2                                                 16853
6854         \exp_after:wN \__fp_parse_small_round:NN                       16854
6855         \exp_after:wN #1                                                 16855
6856         \exp:w                                                            16856
6857     \else:                                                                16857
6858         \__fp_parse_exponent:Nw #2                                       16858
6859     \fi:                                                                  16859
6860     \__fp_parse_expand:w                                                  16860
6861 }                                                                           16861
6862 \cs_new:Npn \__fp_parse_large_round_aux:wNN #1 \__fp_sep: #2 #3       16862
6863 {                                                                           16863
6864     + #2                                                                    16864
6865     \exp_after:wN \__fp_parse_round_after:wN                             16865
6866     \int_value:w \__fp_int_eval:w #1                                     16866
6867     \if:w . \exp_not:N #3                                                 16867
6868         + 0 * \__fp_int_eval:w 0                                         16868
```

```
6869         \exp_after:wN \__fp_parse_round_loop:N
6870         \exp:w \exp_after:wN \__fp_parse_expand:w
6871     \else:
6872         \exp_after:wN \__fp_sep:
6873         \exp_after:wN 0
6874         \exp_after:wN #3
6875     \fi:
6876 }
6877 \cs_new:Npn \__fp_parse_exponent:Nw #1 #2 \__fp_parse_expand:w
6878 {
6879     \exp_after:wN \__fp_sep:
6880     \int_value:w #2 \__fp_parse_exponent:N #1
6881 }
6882 \cs_new:Npn \__fp_parse_exponent:N #1
6883 {
6884     \if:w e \if:w E \exp_not:N #1 e \else: \exp_not:N #1 \fi:
6885     \exp_after:wN \__fp_parse_exponent_aux:NN
6886     \exp_after:wN #1
6887     \exp:w
6888 \else:
6889     0 \__fp_parse_return_sep:w #1
6890 \fi:
6891 \__fp_parse_expand:w
6892 }
6893 \cs_new:Npn \__fp_parse_exponent_aux:NN #1#2
6894 {
6895     \if_int_compare:w \if_catcode:w \scan_stop: \exp_not:N #2
6896         0 \else: `#2 \fi: > `9 \exp_stop_f:
6897     0 \exp_after:wN \__fp_sep: \exp_after:wN #1
6898 \else:
6899     \exp_after:wN \__fp_parse_exponent_sign:N
6900 \fi:
6901 #2
6902 }
6903 \cs_new:Npn \__fp_parse_exponent_sign:N #1
6904 {
6905     \if:w + \if:w - \exp_not:N #1 + \fi: \token_to_str:N #1
6906     \exp_after:wN \__fp_parse_exponent_sign:N
6907     \exp:w \exp_after:wN \__fp_parse_expand:w
6908 \else:
6909     \exp_after:wN \__fp_parse_exponent_body:N
6910     \exp_after:wN #1
6911 \fi:
6912 }
6913 \cs_new:Npn \__fp_parse_exponent_body:N #1
6914 {
```

```
6915 \if_int_compare:w 9 < 1 \token_to_str:N #1 \exp_stop_f: 16915
6916 \token_to_str:N #1 16916
6917 \exp_after:wN \__fp_parse_exponent_digits:N 16917
6918 \exp:w 16918
6919 \else: 16919
6920 \__fp_parse_exponent_keep:NTF #1 16920
6921 { \__fp_parse_return_sep:w #1 } 16921
6922 { 16922
6923 \exp_after:wN \__fp_sep: 16923
6924 \exp:w 16924
6925 } 16925
6926 \fi: 16926
6927 \__fp_parse_expand:w 16927
6928 } 16928
6929 \cs_new:Npn \__fp_parse_exponent_digits:N #1 16929
6930 { 16930
6931 \if_int_compare:w 9 < 1 \token_to_str:N #1 \exp_stop_f: 16931
6932 \token_to_str:N #1 16932
6933 \exp_after:wN \__fp_parse_exponent_digits:N 16933
6934 \exp:w 16934
6935 \else: 16935
6936 \__fp_parse_return_sep:w #1 16936
6937 \fi: 16937
6938 \__fp_parse_expand:w 16938
6939 } 16939
6940 \prg_new_conditional:Npnn \__fp_parse_exponent_keep:N #1 { TF } 16940
6941 { 16941
6942 \if_catcode:w \scan_stop: \exp_not:N #1 16942
6943 \if_meaning:w \scan_stop: #1 16943
6944 \if:w 0 \__fp_str_if_eq:nn { \s__fp } { \exp_not:N #1 } 16944
6945 0 16945
6946 \msg_expandable_error:nnn 16946
6947 { fp } { after-e } { floating~point~ } 16947
6948 \prg_return_true: 16948
6949 \else: 16949
6950 0 16950
6951 \msg_expandable_error:nnn 16951
6952 { kernel } { bad-variable } { #1 } 16952
6953 \prg_return_false: 16953
6954 \fi: 16954
6955 \else: 16955
6956 \if:w 0 \__fp_str_if_eq:nn { \int_value:w #1 } { \tex_the:D #1 } 16956
6957 \int_value:w #1 16957
6958 \else: 16958
6959 0 16959
6960 \msg_expandable_error:nnn 16960
```

```
6961         { fp } { after-e } { dimension~#1 }
6962     \fi:
6963     \prg_return_false:
6964 \fi:
6965 \else:
6966     0
6967     \msg_expandable_error:nnn
6968     { fp } { missing } { exponent }
6969     \prg_return_true:
6970 \fi:
6971 }
6972 \cs_new_eq:cN { __fp_parse_prefix+:Nw } \__fp_parse_one:Nw
6973 \cs_new:Npn \__fp_parse_apply_function:NNNwN #1#2#3#4@#5
6974 {
6975     #3 #2 #4 @
6976     \exp:w \exp_end_continue_f:w #5 #1
6977 }
6978 \cs_new:Npn \__fp_parse_apply_unary:NNNwN #1#2#3#4@#5
6979 {
6980     \__fp_parse_apply_unary_chk:NwNw #4 @ \__fp_sep: . \s__fp_stop
6981     \__fp_parse_apply_unary_type:NNN
6982     #3 #2 #4 @
6983     \exp:w \exp_end_continue_f:w #5 #1
6984 }
6985 \cs_new:Npn \__fp_parse_apply_unary_chk:NwNw #1#2 \__fp_sep: #3#4 \s__fp_stop
6986 {
6987     \if_meaning:w @ #3 \else:
6988         \token_if_eq_meaning:NNTF . #3
6989         { \__fp_parse_apply_unary_chk:nNNNNw { no } }
6990         { \__fp_parse_apply_unary_chk:nNNNNw { multi } }
6991     \fi:
6992 }
6993 \cs_new:Npn \__fp_parse_apply_unary_chk:nNNNNw #1#2#3#4#5#6 @
6994 {
6995     #2
6996     \__fp_error:nffn { #1-arg } { \__fp_func_to_name:N #4 } { } { }
6997     \exp_after:wN #4 \exp_after:wN #5 \c_nan_fp @
6998 }
6999 \cs_new:Npn \__fp_parse_apply_unary_type:NNN #1#2#3
7000 {
7001     \__fp_change_func_type:NNN #3 #1 \__fp_parse_apply_unary_error:NNw
7002     #2 #3
7003 }
7004 \cs_new:Npn \__fp_parse_apply_unary_error:NNw #1#2#3 @
7005 { \__fp_invalid_operation_o:fw { \__fp_func_to_name:N #1 } #3 }
7006 \cs_set_protected:Npn \__fp_tmp:w #1#2#3#4
```



```
7007 {
7008     \cs_new:cpn { __fp_parse_prefix_ #1 :Nw } ##1
7009     {
7010         \exp_after:wN \__fp_parse_apply_unary:NNNwN
7011         \exp_after:wN ##1
7012         \exp_after:wN #4
7013         \exp_after:wN #3
7014         \exp:w
7015         \if_int_compare:w #2 < ##1
7016             \__fp_parse_operand:Nw ##1
7017         \else:
7018             \__fp_parse_operand:Nw #2
7019         \fi:
7020         \__fp_parse_expand:w
7021     }
7022 }
7023 \__fp_tmp:w - \c__fp_prec_not_int \__fp_set_sign_o:w 2
7024 \__fp_tmp:w ! \c__fp_prec_not_int \__fp_not_o:w ?
7025 \cs_new:cpn { __fp_parse_prefix_.:Nw } #1
7026 {
7027     \exp_after:wN \__fp_parse_infix_after_operand:NwN
7028     \exp_after:wN #1
7029     \exp:w \exp_end_continue_f:w
7030     \exp_after:wN \__fp_sanitize:wN
7031     \int_value:w \__fp_int_eval:w 0 \__fp_parse_strim_zeros:N
7032 }
7033 \cs_new:cpn { __fp_parse_prefix_( :Nw } #1
7034 {
7035     \exp_after:wN \__fp_parse_lparen_after:NwN
7036     \exp_after:wN #1
7037     \exp:w
7038     \if_int_compare:w #1 = \c__fp_prec_func_int
7039         \__fp_parse_operand:Nw \c__fp_prec_comma_int
7040     \else:
7041         \__fp_parse_operand:Nw \c__fp_prec_tuple_int
7042     \fi:
7043     \__fp_parse_expand:w
7044 }
7045 \cs_new:Npe \__fp_parse_lparen_after:NwN #1#2 @ #3
7046 {
7047     \exp_not:N \token_if_eq_meaning:NNTF #3
7048     \exp_not:c { __fp_parse_infix_) :N }
7049     {
7050         \exp_not:N \__fp_exp_after_array_f:w #2 \s__fp_expr_stop
7051         \exp_not:N \exp_after:wN
7052         \exp_not:N \__fp_parse_infix_after_paren:NN
```

```
7053         \exp_not:N \exp_after:wN #1
7054         \exp_not:N \exp:w
7055         \exp_not:N \__fp_parse_expand:w
7056     }
7057 {
7058     \exp_not:N \msg_expandable_error:nnn
7059     { fp } { missing } { ) }
7060     \exp_not:N \tl_if_empty:nT {#2} \exp_not:N \c__fp_empty_tuple_fp
7061     #2 @
7062     \exp_not:N \use_none:n #3
7063 }
7064 }
7065 \cs_new:cpn { __fp_parse_prefix_):Nw } #1
7066 {
7067     \if_int_compare:w #1 = \c__fp_prec_comma_int
7068     \else:
7069         \if_int_compare:w #1 = \c__fp_prec_tuple_int
7070         \exp_after:wN \c__fp_empty_tuple_fp \exp:w
7071     \else:
7072         \msg_expandable_error:nnn
7073         { fp } { missing-number } { ) }
7074         \exp_after:wN \c_nan_fp \exp:w
7075     \fi:
7076     \exp_end_continue_f:w
7077     \fi:
7078     \__fp_parse_infix_after_paren:NN #1 )
7079 }
7080 \cs_set_protected:Npn \__fp_tmp:w #1 #2
7081 {
7082     \cs_new:cpn { __fp_parse_word_#1:N }
7083     { \exp_after:wN #2 \exp:w \exp_end_continue_f:w \__fp_parse_infix:NN }
7084 }
7085 \__fp_tmp:w { inf } \c_inf_fp
7086 \__fp_tmp:w { nan } \c_nan_fp
7087 \__fp_tmp:w { pi } \c_pi_fp
7088 \__fp_tmp:w { deg } \c_one_degree_fp
7089 \__fp_tmp:w { true } \c_one_fp
7090 \__fp_tmp:w { false } \c_zero_fp
7091 \cs_new_eq:NN \__fp_parse_caseless_inf:N \__fp_parse_word_inf:N
7092 \cs_new_eq:NN \__fp_parse_caseless_infinity:N \__fp_parse_word_inf:N
7093 \cs_new_eq:NN \__fp_parse_caseless_nan:N \__fp_parse_word_nan:N
7094 \cs_set_protected:Npn \__fp_tmp:w #1 #2
7095 {
7096     \cs_new:cpn { __fp_parse_word_#1:N }
7097     {
7098         \__fp_exp_after_f:nw { \__fp_parse_infix:NN }
```

```
7099 \s__fp __fp_chk:w 10 #2 \__fp_sep: 17099
7100 } 17100
7101 } 17101
7102 \__fp_tmp:w {pt} { {1} {1000} {0000} {0000} {0000} } 17102
7103 \__fp_tmp:w {in} { {2} {7227} {0000} {0000} {0000} } 17103
7104 \__fp_tmp:w {pc} { {2} {1200} {0000} {0000} {0000} } 17104
7105 \__fp_tmp:w {cm} { {2} {2845} {2755} {9055} {1181} } 17105
7106 \__fp_tmp:w {mm} { {1} {2845} {2755} {9055} {1181} } 17106
7107 \__fp_tmp:w {dd} { {1} {1070} {0085} {6496} {0630} } 17107
7108 \__fp_tmp:w {cc} { {2} {1284} {0102} {7795} {2756} } 17108
7109 \__fp_tmp:w {nd} { {1} {1066} {9783} {4645} {6693} } 17109
7110 \__fp_tmp:w {nc} { {2} {1280} {3740} {1574} {8031} } 17110
7111 \__fp_tmp:w {bp} { {1} {1003} {7500} {0000} {0000} } 17111
7112 \__fp_tmp:w {sp} { {-4} {1525} {8789} {0625} {0000} } 17112
7113 \tl_map_inline:nn { {em} {ex} } 17113
7114 { 17114
7115 \cs_new:cpn { __fp_parse_word_#1:N } 17115
7116 { 17116
7117 \exp_after:wN \__fp_from_dim_test:ww 17117
7118 \exp_after:wN 0 \exp_after:wN , 17118
7119 \int_value:w \dim_to_decimal_in_sp:n { 1 #1 } \exp_after:wN \__fp_sep: 17119
7120 \exp:w \exp_end_continue_f:w \__fp_parse_infix:NN 17120
7121 } 17121
7122 } 17122
7123 \cs_new:Npn \__fp_parse_unary_function:NNN #1#2#3 17123
7124 { 17124
7125 \exp_after:wN \__fp_parse_apply_unary:NNNwN 17125
7126 \exp_after:wN #3 17126
7127 \exp_after:wN #2 17127
7128 \exp_after:wN #1 17128
7129 \exp:w 17129
7130 \__fp_parse_operand:Nw \c__fp_prec_func_int \__fp_parse_expand:w 17130
7131 } 17131
7132 \cs_new:Npn \__fp_parse_function:NNN #1#2#3 17132
7133 { 17133
7134 \exp_after:wN \__fp_parse_apply_function:NNNwN 17134
7135 \exp_after:wN #3 17135
7136 \exp_after:wN #2 17136
7137 \exp_after:wN #1 17137
7138 \exp:w 17138
7139 \__fp_parse_operand:Nw \c__fp_prec_func_int \__fp_parse_expand:w 17139
7140 } 17140
7141 \cs_new:Npn \__fp_parse:n #1 17141
7142 { 17142
7143 \exp:w 17143
7144 \exp_after:wN \__fp_parse_after:ww 17144
```

```
7145         \exp:w                                     17145
7146         \__fp_parse_operand:Nw \c__fp_prec_end_int 17146
7147         \__fp_parse_expand:w #1                     17147
7148         \s__fp_expr_mark \__fp_parse_infix_end:N     17148
7149         \s__fp_expr_stop                             17149
7150     \exp_end:                                         17150
7151 }                                                     17151
7152 \cs_new:Npn \__fp_parse_after:ww                    17152
7153     #1@ \__fp_parse_infix_end:N \s__fp_expr_stop #2 { #2 #1 } 17153
7154 \cs_new:Npn \__fp_parse_o:n #1                      17154
7155 {                                                     17155
7156     \exp:w                                           17156
7157     \exp_after:wN \__fp_parse_after:ww              17157
7158     \exp:w                                           17158
7159     \__fp_parse_operand:Nw \c__fp_prec_end_int      17159
7160     \__fp_parse_expand:w #1                          17160
7161     \s__fp_expr_mark \__fp_parse_infix_end:N        17161
7162     \s__fp_expr_stop                                17162
7163     {                                                 17163
7164         \exp_end_continue_f:w                        17164
7165         \__fp_exp_after_any_f:nw { \exp_after:wN \exp_stop_f: } 17165
7166     }                                                 17166
7167 }                                                     17167
7168 \cs_new:Npn \__fp_parse_operand:Nw #1               17168
7169 {                                                     17169
7170     \exp_end_continue_f:w                            17170
7171     \exp_after:wN \__fp_parse_continue:NwN          17171
7172     \exp_after:wN #1                                 17172
7173     \exp:w \exp_end_continue_f:w                    17173
7174     \exp_after:wN \__fp_parse_one:Nw                17174
7175     \exp_after:wN #1                                 17175
7176     \exp:w                                           17176
7177 }                                                     17177
7178 \cs_new:Npn \__fp_parse_continue:NwN #1 #2 @ #3 { #3 #1 #2 @ } 17178
7179 \cs_new:Npn \__fp_parse_apply_binary:NwNwN #1 #2#3@ #4 #5#6@ #7 17179
7180 {                                                     17180
7181     \exp_after:wN \__fp_parse_continue:NwN          17181
7182     \exp_after:wN #1                                 17182
7183     \exp:w \exp_end_continue_f:w                    17183
7184     \exp_after:wN \__fp_parse_apply_binary_chk:NN   17184
7185     \cs:w                                             17185
7186         __fp                                          17186
7187         \__fp_type_from_scan:N #2                    17187
7188         _#4                                           17188
7189         \__fp_type_from_scan:N #5                    17189
7190         _o:ww                                         17190
```

7191	\cs_end:	1719
7192	#4	1719
7193	#2#3 #5#6	1719
7194	\exp:w \exp_end_continue_f:w #7 #1	1719
7195	}	1719
7196	\cs_new:Npn __fp_parse_apply_binary_chk:NN #1#2	1719
7197	{	1719
7198	\if_meaning:w \scan_stop: #1	1719
7199	__fp_parse_apply_binary_error:NNN #2	1719
7200	\fi:	1720
7201	#1	1720
7202	}	1720
7203	\cs_new:Npn __fp_parse_apply_binary_error:NNN #1#2#3	1720
7204	{	1720
7205	#2	1720
7206	__fp_invalid_operation_o:Nww #1	1720
7207	}	1720
7208	\cs_new:Npn __fp_binary_type_o:Nww #1 #2#3 __fp_sep: #4	1720
7209	{	1720
7210	\exp_after:wN __fp_parse_apply_binary_chk:NN	1721
7211	\cs:w	1721
7212	__fp	1721
7213	__fp_type_from_scan:N #2	1721
7214	_ #1	1721
7215	__fp_type_from_scan:N #4	1721
7216	_o:ww	1721
7217	\cs_end:	1721
7218	#1	1721
7219	#2 #3 __fp_sep: #4	1721
7220	}	1722
7221	\cs_new:Npn __fp_binary_rev_type_o:Nww #1 #2#3 __fp_sep: #4#5 __fp_sep:	1722
7222	{	1722
7223	\exp_after:wN __fp_parse_apply_binary_chk:NN	1722
7224	\cs:w	1722
7225	__fp	1722
7226	__fp_type_from_scan:N #4	1722
7227	_ #1	1722
7228	__fp_type_from_scan:N #2	1722
7229	_o:ww	1722
7230	\cs_end:	1723
7231	#1	1723
7232	#4 #5 __fp_sep: #2 #3 __fp_sep:	1723
7233	}	1723
7234	\cs_new:Npn __fp_parse_infix_after_operand:NwN #1 #2__fp_sep:	1723
7235	{	1723
7236	__fp_exp_after_f:nw { __fp_parse_infix:NN #1 }	1723

```
7237 #2\__fp_sep: 1723
7238 } 1723
7239 \cs_new:Npn \__fp_parse_infix:NN #1 #2 1723
7240 { 1724
7241   \if_catcode:w \scan_stop: \exp_not:N #2 1724
7242   \if:w 0 \__fp_str_if_eq:nn { \s__fp_expr_mark } { \exp_not:N #2 } 1724
7243   \exp_after:wN \exp_after:wN 1724
7244   \exp_after:wN \__fp_parse_infix_mark:NNN 1724
7245   \else: 1724
7246   \exp_after:wN \exp_after:wN 1724
7247   \exp_after:wN \__fp_parse_infix_juxt:N 1724
7248   \fi: 1724
7249 \else: 1724
7250   \if_int_compare:w 1725
7251     \__fp_int_eval:w 1725
7252     ( `#2 \if_int_compare:w `#2 > `Z - 32 \fi: ) / 26 1725
7253     = 3 \exp_stop_f: 1725
7254     \exp_after:wN \exp_after:wN 1725
7255     \exp_after:wN \__fp_parse_infix_juxt:N 1725
7256   \else: 1725
7257     \exp_after:wN \__fp_parse_infix_check:NNN 1725
7258     \cs:w 1725
7259     __fp_parse_infix_ \token_to_str:N #2 :N 1725
7260     \exp_after:wN \exp_after:wN \exp_after:wN 1726
7261     \cs_end: 1726
7262     \fi: 1726
7263   \fi: 1726
7264   #1 1726
7265   #2 1726
7266 } 1726
7267 \cs_new:Npn \__fp_parse_infix_check:NNN #1#2#3 1726
7268 { 1726
7269   \if_meaning:w \scan_stop: #1 1726
7270   \msg_expandable_error:nnn 1727
7271     { fp } { missing } { * } 1727
7272   \exp_after:wN \__fp_parse_infix_mul:N 1727
7273   \exp_after:wN #2 1727
7274   \exp_after:wN #3 1727
7275   \else: 1727
7276   \exp_after:wN #1 1727
7277   \exp_after:wN #2 1727
7278   \exp:w \exp_after:wN \__fp_parse_expand:w 1727
7279   \fi: 1727
7280 } 1728
7281 \cs_new:Npn \__fp_parse_infix_after_paren:NN #1 #2 1728
7282 { 1728
```

```
7283 \if_catcode:w \scan_stop: \exp_not:N #2 17283
7284 \if:w 0 \__fp_str_if_eq:nn { \s__fp_expr_mark } { \exp_not:N #2 } 17284
7285 \exp_after:wN \exp_after:wN 17285
7286 \exp_after:wN \__fp_parse_infix_mark:NNN 17286
7287 \else: 17287
7288 \exp_after:wN \exp_after:wN 17288
7289 \exp_after:wN \__fp_parse_infix_mul:N 17289
7290 \fi: 17290
7291 \else: 17291
7292 \if_int_compare:w 17292
7293 \__fp_int_eval:w 17293
7294 ( `#2 \if_int_compare:w `#2 > `Z - 32 \fi: ) / 26 17294
7295 = 3 \exp_stop_f: 17295
7296 \exp_after:wN \exp_after:wN 17296
7297 \exp_after:wN \__fp_parse_infix_mul:N 17297
7298 \else: 17298
7299 \exp_after:wN \__fp_parse_infix_check:NNN 17299
7300 \cs:w 17300
7301 \__fp_parse_infix_ \token_to_str:N #2 :N 17301
7302 \exp_after:wN \exp_after:wN \exp_after:wN 17302
7303 \cs_end: 17303
7304 \fi: 17304
7305 \fi: 17305
7306 #1 17306
7307 #2 17307
7308 } 17308
7309 \cs_new:Npn \__fp_parse_infix_mark:NNN #1#2#3 { #3 #1 } 17309
7310 \cs_new:Npn \__fp_parse_infix_end:N #1 17310
7311 { @ \use_none:n \__fp_parse_infix_end:N } 17311
7312 \cs_set_protected:Npn \__fp_tmp:w #1 17312
7313 { 17313
7314 \cs_new:Npn #1 ##1 17314
7315 { 17315
7316 \if_int_compare:w ##1 > \c__fp_prec_end_int 17316
7317 \exp_after:wN @ 17317
7318 \exp_after:wN \use_none:n 17318
7319 \exp_after:wN #1 17319
7320 \else: 17320
7321 \msg_expandable_error:nnn { fp } { extra } { ) } 17321
7322 \exp_after:wN \__fp_parse_infix:NN 17322
7323 \exp_after:wN ##1 17323
7324 \exp:w \exp_after:wN \__fp_parse_expand:w 17324
7325 \fi: 17325
7326 } 17326
7327 } 17327
7328 \exp_args:Nc \__fp_tmp:w { __fp_parse_infix_):N } 17328
```



```
7329 \cs_set_protected:Npn \__fp_tmp:w #1
7330 {
7331   \cs_new:Npn #1 ##1
7332   {
7333     \if_int_compare:w ##1 > \c__fp_prec_comma_int
7334     \exp_after:wN @
7335     \exp_after:wN \use_none:n
7336     \exp_after:wN #1
7337   \else:
7338     \if_int_compare:w ##1 < \c__fp_prec_comma_int
7339     \exp_after:wN @
7340     \exp_after:wN \__fp_parse_apply_comma:NwNwN
7341     \exp_after:wN ,
7342     \exp:w
7343   \else:
7344     \exp_after:wN \__fp_parse_infix_comma:w
7345     \exp:w
7346   \fi:
7347   \__fp_parse_operand:Nw \c__fp_prec_comma_int
7348   \exp_after:wN \__fp_parse_expand:w
7349   \fi:
7350 }
7351 }
7352 \exp_args:Nc \__fp_tmp:w { \__fp_parse_infix_,:N }
7353 \cs_new:Npn \__fp_parse_infix_comma:w #1 @
7354 { #1 @ \use_none:n }
7355 \cs_new:Npn \__fp_parse_apply_comma:NwNwN #1 #2@ #3 #4@ #5
7356 {
7357   \exp_after:wN \__fp_parse_continue:NwN
7358   \exp_after:wN #1
7359   \exp:w \exp_end_continue_f:w
7360   \__fp_exp_after_tuple_f:nw { }
7361   \s__fp_tuple \__fp_tuple_chk:w { #2 #4 } \__fp_sep:
7362   #5 #1
7363 }
7364 \cs_set_protected:Npn \__fp_tmp:w #1#2#3#4
7365 {
7366   \cs_new:Npn #1 ##1
7367   {
7368     \if_int_compare:w ##1 < #3
7369     \exp_after:wN @
7370     \exp_after:wN \__fp_parse_apply_binary:NwNwN
7371     \exp_after:wN #2
7372     \exp:w
7373     \__fp_parse_operand:Nw #4
7374     \exp_after:wN \__fp_parse_expand:w
```

```
7375         \else:
7376             \exp_after:wN @
7377             \exp_after:wN \use_none:n
7378             \exp_after:wN #1
7379         \fi:
7380     }
7381 }
7382 \exp_args:Nc \__fp_tmp:w { __fp_parse_infix_^:N } ^
7383     \c__fp_prec_hatii_int \c__fp_prec_hat_int
7384 \exp_args:Nc \__fp_tmp:w { __fp_parse_infix_juxt:N } *
7385     \c__fp_prec_juxt_int \c__fp_prec_juxt_int
7386 \exp_args:Nc \__fp_tmp:w { __fp_parse_infix_/:N } /
7387     \c__fp_prec_times_int \c__fp_prec_times_int
7388 \exp_args:Nc \__fp_tmp:w { __fp_parse_infix_mul:N } *
7389     \c__fp_prec_times_int \c__fp_prec_times_int
7390 \exp_args:Nc \__fp_tmp:w { __fp_parse_infix -:N } -
7391     \c__fp_prec_plus_int \c__fp_prec_plus_int
7392 \exp_args:Nc \__fp_tmp:w { __fp_parse_infix_+:N } +
7393     \c__fp_prec_plus_int \c__fp_prec_plus_int
7394 \exp_args:Nc \__fp_tmp:w { __fp_parse_infix_and:N } &
7395     \c__fp_prec_and_int \c__fp_prec_and_int
7396 \exp_args:Nc \__fp_tmp:w { __fp_parse_infix_or:N } |
7397     \c__fp_prec_or_int \c__fp_prec_or_int
7398 \cs_new:cpn { __fp_parse_infix_(:N } #1
7399     { \__fp_parse_infix_mul:N #1 ( }
7400 \cs_set_protected:Npn \__fp_tmp:w #1
7401     {
7402         \cs_new:cpn { __fp_parse_infix_*:N } ##1##2
7403         {
7404             \if:w * \exp_not:N ##2
7405                 \exp_after:wN #1
7406                 \exp_after:wN ##1
7407             \else:
7408                 \exp_after:wN \__fp_parse_infix_mul:N
7409                 \exp_after:wN ##1
7410                 \exp_after:wN ##2
7411             \fi:
7412         }
7413     }
7414 \exp_args:Nc \__fp_tmp:w { __fp_parse_infix_^:N }
7415 \cs_set_protected:Npn \__fp_tmp:w #1#2#3
7416     {
7417         \cs_new:Npn #1 ##1##2
7418         {
7419             \if:w #2 \exp_not:N ##2
7420                 \exp_after:wN #1
```

```
7421         \exp_after:wN ##1
7422         \exp:w \exp_after:wN \__fp_parse_expand:w
7423     \else:
7424         \exp_after:wN #3
7425         \exp_after:wN ##1
7426         \exp_after:wN ##2
7427     \fi:
7428 }
7429 }
7430 \exp_args:Nc \__fp_tmp:w { __fp_parse_infix_|:N } | \__fp_parse_infix_or:N
7431 \exp_args:Nc \__fp_tmp:w { __fp_parse_infix_&:N } & \__fp_parse_infix_and:N
7432 \cs_set_protected:Npn \__fp_tmp:w #1#2#3#4
7433 {
7434     \cs_new:Npn #1 ##1
7435     {
7436         \if_int_compare:w ##1 < \c__fp_prec_quest_int
7437             #4
7438             \exp_after:wN @
7439             \exp_after:wN #2
7440             \exp:w
7441             \__fp_parse_operand:Nw #3
7442             \exp_after:wN \__fp_parse_expand:w
7443         \else:
7444             \exp_after:wN @
7445             \exp_after:wN \use_none:n
7446             \exp_after:wN #1
7447         \fi:
7448     }
7449 }
7450 \exp_args:Nc \__fp_tmp:w { __fp_parse_infix_?:N }
7451 \__fp_ternary:NwwN \c__fp_prec_quest_int { }
7452 \exp_args:Nc \__fp_tmp:w { __fp_parse_infix_:N }
7453 \__fp_ternary_auxii:NwwN \c__fp_prec_colon_int
7454 {
7455     \msg_expandable_error:nnnn
7456     { fp } { missing } { ? } { ~for~?: }
7457 }
7458 \cs_new:cpn { __fp_parse_infix_<:N } #1
7459 { \__fp_parse_compare:NNNNNNN #1 1 0 0 0 0 < }
7460 \cs_new:cpn { __fp_parse_infix_=:N } #1
7461 { \__fp_parse_compare:NNNNNNN #1 1 0 0 0 0 = }
7462 \cs_new:cpn { __fp_parse_infix_>:N } #1
7463 { \__fp_parse_compare:NNNNNNN #1 1 0 0 0 0 > }
7464 \cs_new:cpn { __fp_parse_infix_!:N } #1
7465 {
7466     \exp_after:wN \__fp_parse_compare:NNNNNNN
```

```
7467 \exp_after:wN #1
7468 \exp_after:wN 0
7469 \exp_after:wN 1
7470 \exp_after:wN 1
7471 \exp_after:wN 1
7472 \exp_after:wN 1
7473 }
7474 \cs_new:Npn \__fp_parse_excl_error:
7475 {
7476 \msg_expandable_error:nnnn
7477 { fp } { missing } { = } { ~after~!. }
7478 }
7479 \cs_new:Npn \__fp_parse_compare:NNNNNNN #1
7480 {
7481 \if_int_compare:w #1 < \c__fp_prec_comp_int
7482 \exp_after:wN \__fp_parse_compare_auxi:NNNNNNN
7483 \exp_after:wN \__fp_parse_excl_error:
7484 \else:
7485 \exp_after:wN @
7486 \exp_after:wN \use_none:n
7487 \exp_after:wN \__fp_parse_compare:NNNNNNN
7488 \fi:
7489 }
7490 \cs_new:Npn \__fp_parse_compare_auxi:NNNNNNN #1#2#3#4#5#6#7
7491 {
7492 \if_case:w
7493 \__fp_int_eval:w \exp_after:wN ` \token_to_str:N #7 - `<
7494 \__fp_int_eval_end:
7495 \__fp_parse_compare_auxii:NNNNN #2#2#4#5#6
7496 \or: \__fp_parse_compare_auxii:NNNNN #2#3#2#5#6
7497 \or: \__fp_parse_compare_auxii:NNNNN #2#3#4#2#6
7498 \or: \__fp_parse_compare_auxii:NNNNN #2#3#4#5#2
7499 \else: #1 \__fp_parse_compare_end:NNNNw #3#4#5#6#7
7500 \fi:
7501 }
7502 \cs_new:Npn \__fp_parse_compare_auxii:NNNNN #1#2#3#4#5
7503 {
7504 \exp_after:wN \__fp_parse_compare_auxi:NNNNNNN
7505 \exp_after:wN \prg_do_nothing:
7506 \exp_after:wN #1
7507 \exp_after:wN #2
7508 \exp_after:wN #3
7509 \exp_after:wN #4
7510 \exp_after:wN #5
7511 \exp:w \exp_after:wN \__fp_parse_expand:w
7512 }
```

7513	\cs_new:Npn __fp_parse_compare_end:NNNNw #1#2#3#4#5 \fi:	17513
7514	{	17514
7515	\fi:	17515
7516	\exp_after:wN @	17516
7517	\exp_after:wN __fp_parse_apply_compare:NwNNNNNwN	17517
7518	\exp_after:wN \c_one_fp	17518
7519	\exp_after:wN #1	17519
7520	\exp_after:wN #2	17520
7521	\exp_after:wN #3	17521
7522	\exp_after:wN #4	17522
7523	\exp:w	17523
7524	__fp_parse_operand:Nw \c__fp_prec_comp_int __fp_parse_expand:w #5	17524
7525	}	17525
7526	\cs_new:Npn __fp_parse_apply_compare:NwNNNNNwN	17526
7527	#1 #2@ #3 #4#5#6#7 #8@ #9	17527
7528	{	17528
7529	\if_int_odd:w	17529
7530	\if_meaning:w \c_zero_fp #3	17530
7531	0	17531
7532	\else:	17532
7533	\if_case:w __fp_compare_back_any:ww #8 #2 \exp_stop_f:	17533
7534	#5 \or: #6 \or: #7 \else: #4	17534
7535	\fi:	17535
7536	\fi:	17536
7537	\exp_stop_f:	17537
7538	\exp_after:wN __fp_parse_apply_compare_aux:NNwN	17538
7539	\exp_after:wN \c_one_fp	17539
7540	\else:	17540
7541	\exp_after:wN __fp_parse_apply_compare_aux:NNwN	17541
7542	\exp_after:wN \c_zero_fp	17542
7543	\fi:	17543
7544	#1 #8 #9	17544
7545	}	17545
7546	\cs_new:Npn __fp_parse_apply_compare_aux:NNwN #1 #2 #3__fp_sep: #4	17546
7547	{	17547
7548	\if_meaning:w __fp_parse_compare:NNNNNNN #4	17548
7549	\exp_after:wN __fp_parse_continue_compare:NNwNN	17549
7550	\exp_after:wN #1	17550
7551	\exp_after:wN #2	17551
7552	\exp:w \exp_end_continue_f:w	17552
7553	__fp_exp_after_o:w #3__fp_sep:	17553
7554	\exp:w \exp_end_continue_f:w	17554
7555	\else:	17555
7556	\exp_after:wN __fp_parse_continue:NwN	17556
7557	\exp_after:wN #2	17557
7558	\exp:w \exp_end_continue_f:w	17558

```
7559     \exp_after:wN #1
7560     \exp:w \exp_end_continue_f:w
7561     \fi:
7562     #4 #2
7563 }
7564 \cs_new:Npn \__fp_parse_continue_compare:NNwNN #1#2 #3@ #4#5
7565 { #4 #2 #3@ #1 }
7566 \cs_new:Npn \__fp_parse_function_all_fp_o:fnw #1#2#3 @
7567 {
7568     \__fp_array_if_all_fp:nTF {#3}
7569     { #2 #3 @ }
7570     {
7571         \__fp_error:nffn { bad-args }
7572         {#1}
7573         { \fp_to_tl:n { \s__fp_tuple \__fp_tuple_chk:w {#3} \__fp_sep: } }
7574         { }
7575         \exp_after:wN \c_nan_fp
7576     }
7577 }
7578 \cs_new:Npn \__fp_parse_function_one_two:nnw #1#2#3
7579 {
7580     \__fp_if_type_fp:NTwFw
7581     #3 { } \s__fp \__fp_parse_function_one_two_error_o:w \s__fp_stop
7582     \__fp_parse_function_one_two_aux:nnw {#1} {#2} #3
7583 }
7584 \cs_new:Npn \__fp_parse_function_one_two_error_o:w #1#2#3#4 @
7585 {
7586     \__fp_error:nffn { bad-args }
7587     {#2}
7588     { \fp_to_tl:n { \s__fp_tuple \__fp_tuple_chk:w {#4} \__fp_sep: } }
7589     { }
7590     \exp_after:wN \c_nan_fp
7591 }
7592 \cs_new:Npn \__fp_parse_function_one_two_aux:nnw #1#2 #3\__fp_sep: #4
7593 {
7594     \__fp_if_type_fp:NTwFw
7595     #4 { }
7596     \s__fp
7597     {
7598         \if_meaning:w @ #4
7599         \exp_after:wN \use_iv:nnnn
7600         \fi:
7601         \__fp_parse_function_one_two_error_o:w
7602     }
7603     \s__fp_stop
7604     \__fp_parse_function_one_two_auxii:nnw {#1} {#2} #3\__fp_sep: #4
```

```
7605 }
7606 \cs_new:Npn \__fp_parse_function_one_two_auxii:nw #1#2#3\__fp_sep: #4\__fp_sep: #5
7607 {
7608   \if_meaning:w @ #5 \else:
7609     \exp_after:wN \__fp_parse_function_one_two_error_o:w
7610   \fi:
7611   \use_ii:nn {#1} { \use_none:n #2 } #3\__fp_sep: #4\__fp_sep: #5
7612 }
7613 \cs_new:Npn \__fp_tuple_map_o:nw #1 \s__fp_tuple \__fp_tuple_chk:w #2 \__fp_sep:
7614 {
7615   \exp_after:wN \s__fp_tuple
7616   \exp_after:wN \__fp_tuple_chk:w
7617   \exp_after:wN {
7618     \exp:w \exp_end_continue_f:w
7619     \__fp_tuple_map_loop_o:nw {#1} #2
7620     { \s__fp \prg_break: } \__fp_sep:
7621     \prg_break_point:
7622     \exp_after:wN } \exp_after:wN \__fp_sep:
7623   }
7624 \cs_new:Npn \__fp_tuple_map_loop_o:nw #1#2#3 \__fp_sep:
7625 {
7626   \use_none:n #2
7627   #1 #2 #3 \__fp_sep:
7628   \exp:w \exp_end_continue_f:w
7629   \__fp_tuple_map_loop_o:nw {#1}
7630 }
7631 \cs_new:Npn \__fp_tuple_mapthread_o:nww #1
7632   \s__fp_tuple \__fp_tuple_chk:w #2 \__fp_sep:
7633   \s__fp_tuple \__fp_tuple_chk:w #3 \__fp_sep:
7634 {
7635   \exp_after:wN \s__fp_tuple
7636   \exp_after:wN \__fp_tuple_chk:w
7637   \exp_after:wN {
7638     \exp:w \exp_end_continue_f:w
7639     \__fp_tuple_mapthread_loop_o:nw {#1}
7640     #2 { \s__fp \prg_break: } \__fp_sep: @
7641     #3 { \s__fp \prg_break: } \__fp_sep:
7642     \prg_break_point:
7643     \exp_after:wN } \exp_after:wN \__fp_sep:
7644   }
7645 \cs_new:Npn \__fp_tuple_mapthread_loop_o:nw #1#2#3 \__fp_sep: #4 @ #5#6 \__fp_sep:
7646 {
7647   \use_none:n #2
7648   \use_none:n #5
7649   #1 #2 #3 \__fp_sep: #5 #6 \__fp_sep:
7650   \exp:w \exp_end_continue_f:w
```



```

7651 \__fp_tuple_maphread_loop_o:nw {#1} #4 @
7652 }
7653 \msg_new:nnn { fp } { deprecated }
7654 { '#1'~deprecated;~use~'#2' }
7655 \msg_new:nnn { fp } { unknown-fp-word }
7656 { Unknown~fp~word~#1. }
7657 \msg_new:nnn { fp } { missing }
7658 { Missing~#1~inserted #2. }
7659 \msg_new:nnn { fp } { extra }
7660 { Extra~#1~ignored. }
7661 \msg_new:nnn { fp } { early-end }
7662 { Premature~end~in~fp~expression. }
7663 \msg_new:nnn { fp } { after-e }
7664 { Cannot~use~#1 after~'e'. }
7665 \msg_new:nnn { fp } { missing-number }
7666 { Missing~number~before~'#1'. }
7667 \msg_new:nnn { fp } { unknown-symbol }
7668 { Unknown~symbol~#1~ignored. }
7669 \msg_new:nnn { fp } { extra-comma }
7670 { Unexpected~comma~turned~to~nan~result. }
7671 \msg_new:nnn { fp } { no-arg }
7672 { #1~got~no~argument;~used~nan. }
7673 \msg_new:nnn { fp } { multi-arg }
7674 { #1~got~more~than~one~argument;~used~nan. }
7675 \msg_new:nnn { fp } { bad-args }
7676 { Arguments~in~#1#2~are~invalid. }
7677 \msg_new:nnn { fp } { infty-pi }
7678 { Math~command~#1 is~not~an~fp }
7679 \cs_if_exist:cT { @unexpandable@protect }
7680 {
7681 \msg_new:nnn { fp } { robust-cmd }
7682 { Robust~command~#1 invalid~in~fp~expression! }
7683 }
7684 %% File: l3fp-assign.dtx
7685 \cs_new_protected:Npn \fp_new:N #1
7686 { \cs_new_eq:NN #1 \c_zero_fp }
7687 \cs_generate_variant:Nn \fp_new:N {c}
7688 \cs_new_protected:Npn \fp_set:Nn #1#2
7689 { \__kernel_tl_set:Nx #1 { \exp_not:f { \__fp_parse:n {#2} } } }
7690 \cs_new_protected:Npn \fp_gset:Nn #1#2
7691 { \__kernel_tl_gset:Nx #1 { \exp_not:f { \__fp_parse:n {#2} } } }
7692 \cs_new_protected:Npn \fp_const:Nn #1#2
7693 { \tl_const:Ne #1 { \exp_not:f { \__fp_parse:n {#2} } } }
7694 \cs_generate_variant:Nn \fp_set:Nn { NV , c , cV }
7695 \cs_generate_variant:Nn \fp_gset:Nn { NV , c , cV }
7696 \cs_generate_variant:Nn \fp_const:Nn {c}

```

```

7697 \cs_new_eq:NN \fp_set_eq:NN \tl_set_eq:NN
7698 \cs_new_eq:NN \fp_gset_eq:NN \tl_gset_eq:NN
7699 \cs_generate_variant:Nn \fp_set_eq:NN { c , Nc , cc }
7700 \cs_generate_variant:Nn \fp_gset_eq:NN { c , Nc , cc }
7701 \cs_new_protected:Npn \fp_zero:N #1 { \fp_set_eq:NN #1 \c_zero_fp }
7702 \cs_new_protected:Npn \fp_gzero:N #1 { \fp_gset_eq:NN #1 \c_zero_fp }
7703 \cs_generate_variant:Nn \fp_zero:N { c }
7704 \cs_generate_variant:Nn \fp_gzero:N { c }
7705 \cs_new_protected:Npn \fp_zero_new:N #1
7706 { \fp_if_exist:NTF #1 { \fp_zero:N #1 } { \fp_new:N #1 } }
7707 \cs_new_protected:Npn \fp_gzero_new:N #1
7708 { \fp_if_exist:NTF #1 { \fp_gzero:N #1 } { \fp_new:N #1 } }
7709 \cs_generate_variant:Nn \fp_zero_new:N { c }
7710 \cs_generate_variant:Nn \fp_gzero_new:N { c }
7711 \cs_new_protected:Npn \fp_add:Nn { \__fp_add:NNNn \fp_set:Nn + }
7712 \cs_new_protected:Npn \fp_gadd:Nn { \__fp_add:NNNn \fp_gset:Nn + }
7713 \cs_new_protected:Npn \fp_sub:Nn { \__fp_add:NNNn \fp_set:Nn - }
7714 \cs_new_protected:Npn \fp_gsub:Nn { \__fp_add:NNNn \fp_gset:Nn - }
7715 \cs_new_protected:Npn \__fp_add:NNNn #1#2#3#4
7716 { #1 #3 { #3 #2 \__fp_parse:n {#4} } }
7717 \cs_generate_variant:Nn \fp_add:Nn { c }
7718 \cs_generate_variant:Nn \fp_gadd:Nn { c }
7719 \cs_generate_variant:Nn \fp_sub:Nn { c }
7720 \cs_generate_variant:Nn \fp_gsub:Nn { c }
7721 \cs_new_protected:Npn \fp_show:N { \__fp_show:NN \tl_show:n }
7722 \cs_generate_variant:Nn \fp_show:N { c }
7723 \cs_new_protected:Npn \fp_log:N { \__fp_show:NN \tl_log:n }
7724 \cs_generate_variant:Nn \fp_log:N { c }
7725 \cs_new_protected:Npn \__fp_show:NN #1#2
7726 {
7727 \__kernel_chk_tl_type:NnnT #2 { fp }
7728 { \exp_args:No \__fp_show_validate:n #2 }
7729 { \exp_args:Ne #1 { \token_to_str:N #2 = \fp_to_tl:N #2 } }
7730 }
7731 \cs_new:Npn \__fp_show_validate:n #1
7732 {
7733 \__fp_show_validate:nn { #1 }
7734 {
7735 \s__fp \__fp_chk:w ??? \__fp_sep: or \iow_newline:
7736 \s__fp_tuple \__fp_tuple_chk:w ? \__fp_sep: or \iow_newline:
7737 \s__fp_symbolic \__fp_symbolic_chk:w ? , ? \__fp_sep:
7738 }
7739 }
7740 \cs_new:Npn \__fp_show_validate_aux:n #1
7741 {
7742 \__fp_show_validate:nn { #1 } { }

```

```
7743 }
7744 \cs_new:Npn \__fp_show_validate:nn #1#2
7745 {
7746   \tl_if_empty:nF { #1 }
7747   {
7748     \str_case:enF { \tl_head:n { #1 } }
7749     {
7750       { \s__fp }
7751       {
7752         \__fp_show_validate:w #1 \s__fp
7753         \__fp_chk:w ??? \__fp_sep: \s__fp_stop
7754       }
7755       { \s__fp_tuple }
7756       {
7757         \__fp_tuple_show_validate:w #1
7758         \s__fp_tuple \__fp_tuple_chk:w ?? \__fp_sep: \s__fp_stop
7759       }
7760       { \s__fp_symbolic }
7761       {
7762         \__fp_symbolic_show_validate:w #1
7763         \s__fp_symbolic \__fp_symbolic_chk:w ? , ?? \__fp_sep: \s__fp_stop
7764       }
7765     }
7766     { #2 }
7767   }
7768 }
7769 \cs_new:Npn \__fp_show_validate:w
7770 #1 \s__fp \__fp_chk:w #2#3#4#5 \__fp_sep: #6 \s__fp_stop
7771 {
7772   \str_if_eq:nnF { #2 } {?}
7773   {
7774     \token_if_eq_meaning:NNTF #2 1
7775     { \s__fp \__fp_chk:w #2 #3 { #4 } #5 \__fp_sep: }
7776     { \s__fp \__fp_chk:w #2 #3 #4 #5 \__fp_sep: }
7777     \__fp_show_validate_aux:n { #6 }
7778   }
7779 }
7780 \cs_new:Npn \__fp_tuple_show_validate:w
7781 #1 \s__fp_tuple \__fp_tuple_chk:w #2#3 \__fp_sep: #4 \s__fp_stop
7782 {
7783   \str_if_eq:nnF { #2 } {?}
7784   {
7785     \s__fp_tuple \__fp_tuple_chk:w { \__fp_show_validate_aux:n { #2 } }
7786     \__fp_sep:
7787   }
7788 }
```

```
7789 \cs_new:Npn \__fp_symbolic_show_validate:w
7790     #1 \s__fp_symbolic \__fp_symbolic_chk:w #2 , #3#4 \__fp_sep: #5 \s__fp_stop
7791 {
7792     \str_if_eq:nnF { #2 } {?}
7793     {
7794         \s__fp_symbolic \__fp_symbolic_chk:w \exp_not:n { #2 } ,
7795         { \__fp_show_validate_aux:n { #3 } }\__fp_sep:
7796         \__fp_show_validate_aux:n { #5 }
7797     }
7798 }
7799 \cs_new_protected:Npn \fp_show:n
7800 { \__kernel_msg_show_eval:Nn \fp_to_tl:n }
7801 \cs_new_protected:Npn \fp_log:n
7802 { \__kernel_msg_log_eval:Nn \fp_to_tl:n }
7803 \fp_const:Nn \c_e_fp { 2.718 2818 2845 9045 }
7804 \fp_const:Nn \c_one_fp { 1 }
7805 \fp_const:Nn \c_pi_fp { 3.141 5926 5358 9793 }
7806 \fp_const:Nn \c_one_degree_fp { 0.0 1745 3292 5199 4330 }
7807 \fp_new:N \l_tmpa_fp
7808 \fp_new:N \l_tmpb_fp
7809 \fp_new:N \g_tmpa_fp
7810 \fp_new:N \g_tmpb_fp
7811 %% File: l3fp-logic.dtx
7812 \cs_new:Npn \__fp_parse_word_max:N
7813 { \__fp_parse_function:NNN \__fp_minmax_o:Nw 2 }
7814 \cs_new:Npn \__fp_parse_word_min:N
7815 { \__fp_parse_function:NNN \__fp_minmax_o:Nw 0 }
7816 \prg_new_eq_conditional:NNn \fp_if_exist:N \cs_if_exist:N { TF , T , F , p }
7817 \prg_new_eq_conditional:NNn \fp_if_exist:c \cs_if_exist:c { TF , T , F , p }
7818 \prg_new_conditional:Npnn \fp_if_nan:n #1 { TF , T , F , p }
7819 {
7820     \if:w 3 \exp_last_unbraced:Nf \__fp_kind:w { \__fp_parse:n {#1} }
7821     \prg_return_true:
7822     \else:
7823     \prg_return_false:
7824     \fi:
7825 }
7826 \prg_new_conditional:Npnn \fp_compare:n #1 { p , T , F , TF }
7827 {
7828     \exp_after:wN \__fp_compare_return:w
7829     \exp:w \exp_end_continue_f:w \__fp_parse:n {#1}
7830 }
7831 \cs_new:Npn \__fp_compare_return:w #1#2#3\__fp_sep:
7832 {
7833     \if_charcode:w 0
7834     \__fp_if_type_fp:NTwFw
```

```
7835         #1 { \__fp_use_i_delimit_by_s_stop:nw #3 \s__fp_stop } 17835
7836         \s__fp 1 \s__fp_stop 17836
7837         \prg_return_false: 17837
7838     \else: 17838
7839         \prg_return_true: 17839
7840     \fi: 17840
7841 } 17841
7842 \prg_new_conditional:Npnn \fp_compare:nNn #1#2#3 { p , T , F , TF } 17842
7843 { 17843
7844     \if_int_compare:w 17844
7845         \exp_after:wN \__fp_compare_aux:wn 17845
7846         \exp:w \exp_end_continue_f:w \__fp_parse:n {#1} {#3} 17846
7847         = \__fp_int_eval:w `#2 - `= \__fp_int_eval_end: 17847
7848         \prg_return_true: 17848
7849     \else: 17849
7850         \prg_return_false: 17850
7851     \fi: 17851
7852 } 17852
7853 \cs_new:Npn \__fp_compare_aux:wn #1\__fp_sep: #2 17853
7854 { 17854
7855     \exp_after:wN \__fp_compare_back_any:ww 17855
7856     \exp:w \exp_end_continue_f:w \__fp_parse:n {#2} #1\__fp_sep: 17856
7857 } 17857
7858 \cs_new:Npn \__fp_compare_back:ww #1#2\__fp_sep: #3#4\__fp_sep: 17858
7859 { 17859
7860     \cs:w 17860
7861         __fp 17861
7862         \__fp_type_from_scan:N #1 17862
7863         _bcmp 17863
7864         \__fp_type_from_scan:N #3 17864
7865         :ww 17865
7866     \cs_end: 17866
7867     #1#2\__fp_sep: #3#4\__fp_sep: 17867
7868 } 17868
7869 \cs_new:Npn \__fp_compare_back_any:ww #1#2\__fp_sep: #3 17869
7870 { 17870
7871     \__fp_if_type_fp:NTwFw 17871
7872     #1 { \__fp_if_type_fp:NTwFw #3 \use_i:nn \s__fp \use_ii:nn \s__fp_stop } 17872
7873     \s__fp \use_ii:nn \s__fp_stop 17873
7874     \__fp_compare_back:ww 17874
7875     { 17875
7876         \cs:w 17876
7877             __fp 17877
7878             \__fp_type_from_scan:N #1 17878
7879             _compare_back 17879
7880             \__fp_type_from_scan:N #3 17880
```

```
7881         :ww
7882     \cs_end:
7883 }
7884 #1#2 \__fp_sep: #3
7885 }
7886 \cs_new:Npn \__fp_bcmp:ww
7887   \s__fp \__fp_chk:w #1 #2 #3\__fp_sep:
7888   \s__fp \__fp_chk:w #4 #5 #6\__fp_sep:
7889 {
7890   \int_value:w
7891     \if_meaning:w 3 #1 \exp_after:wN \__fp_compare_nan:w \fi:
7892     \if_meaning:w 3 #4 \exp_after:wN \__fp_compare_nan:w \fi:
7893     \if_meaning:w 2 #5 - \fi:
7894     \if_meaning:w #2 #5
7895       \if_meaning:w #1 #4
7896         \if_meaning:w 1 #1
7897           \__fp_compare_npos:nwnw #6\__fp_sep: #3\__fp_sep:
7898         \else:
7899           0
7900         \fi:
7901       \else:
7902         \if_int_compare:w #4 < #1 - \fi: 1
7903       \fi:
7904     \else:
7905       \if_int_compare:w #1#4 = \c_zero_int
7906       0
7907     \else:
7908       1
7909     \fi:
7910   \fi:
7911   \exp_stop_f:
7912 }
7913 \cs_new:Npn \__fp_compare_nan:w #1 \fi: \exp_stop_f: { 2 \exp_stop_f: }
7914 \cs_new:Npn \__fp_compare_back_tuple:ww #1\__fp_sep: #2\__fp_sep: { 2 }
7915 \cs_new:Npn \__fp_tuple_compare_back:ww #1\__fp_sep: #2\__fp_sep: { 2 }
7916 \cs_new:Npn \__fp_tuple_compare_back_tuple:ww
7917   \s__fp_tuple \__fp_tuple_chk:w #1\__fp_sep:
7918   \s__fp_tuple \__fp_tuple_chk:w #2\__fp_sep:
7919 {
7920   \int_compare:nNnTF { \__fp_array_count:n {#1} } =
7921     { \__fp_array_count:n {#2} }
7922   {
7923     \int_value:w 0
7924     \__fp_tuple_compare_back_loop:w
7925       #1 { \s__fp \prg_break: } \__fp_sep: @
7926       #2 { \s__fp \prg_break: } \__fp_sep:
```

```
7927         \prg_break_point:
7928         \exp_stop_f:
7929     }
7930     { 2 }
7931 }
7932 \cs_new:Npn \__fp_tuple_compare_back_loop:w #1#2 \__fp_sep: #3 @ #4#5 \__fp_sep:
7933 {
7934     \use_none:n #1
7935     \use_none:n #4
7936     \if_int_compare:w
7937         \__fp_compare_back_any:ww #1 #2 \__fp_sep: #4 #5 \__fp_sep: = \c_zero_int
7938     \else:
7939         2 \exp_after:wN \prg_break:
7940     \fi:
7941     \__fp_tuple_compare_back_loop:w #3 @
7942 }
7943 \cs_new:Npn \__fp_compare_npos:nwnw #1#2\__fp_sep: #3#4\__fp_sep:
7944 {
7945     \if_int_compare:w #1 = #3 \exp_stop_f:
7946         \__fp_compare_significand:nnnnnnnn #2 #4
7947     \else:
7948         \if_int_compare:w #1 < #3 - \fi: 1
7949     \fi:
7950 }
7951 \cs_new:Npn \__fp_compare_significand:nnnnnnnn #1#2#3#4#5#6#7#8
7952 {
7953     \if_int_compare:w #1#2 = #5#6 \exp_stop_f:
7954     \if_int_compare:w #3#4 = #7#8 \exp_stop_f:
7955         0
7956     \else:
7957         \if_int_compare:w #3#4 < #7#8 - \fi: 1
7958     \fi:
7959     \else:
7960         \if_int_compare:w #1#2 < #5#6 - \fi: 1
7961     \fi:
7962 }
7963 \cs_new:Npn \fp_do_until:nn #1#2
7964 {
7965     #2
7966     \fp_compare:nF {#1}
7967     { \fp_do_until:nn {#1} {#2} }
7968 }
7969 \cs_new:Npn \fp_do_while:nn #1#2
7970 {
7971     #2
7972     \fp_compare:nT {#1}
```


7973	{ \fp_do_while:nn {#1} {#2} }	17973
7974	}	17974
7975	\cs_new:Npn \fp_until_do:nn #1#2	17975
7976	{	17976
7977	\fp_compare:nF {#1}	17977
7978	{	17978
7979	#2	17979
7980	\fp_until_do:nn {#1} {#2}	17980
7981	}	17981
7982	}	17982
7983	\cs_new:Npn \fp_while_do:nn #1#2	17983
7984	{	17984
7985	\fp_compare:nT {#1}	17985
7986	{	17986
7987	#2	17987
7988	\fp_while_do:nn {#1} {#2}	17988
7989	}	17989
7990	}	17990
7991	\cs_new:Npn \fp_do_until:nNnn #1#2#3#4	17991
7992	{	17992
7993	#4	17993
7994	\fp_compare:nNnF {#1} #2 {#3}	17994
7995	{ \fp_do_until:nNnn {#1} #2 {#3} {#4} }	17995
7996	}	17996
7997	\cs_new:Npn \fp_do_while:nNnn #1#2#3#4	17997
7998	{	17998
7999	#4	17999
8000	\fp_compare:nNnT {#1} #2 {#3}	18000
8001	{ \fp_do_while:nNnn {#1} #2 {#3} {#4} }	18001
8002	}	18002
8003	\cs_new:Npn \fp_until_do:nNnn #1#2#3#4	18003
8004	{	18004
8005	\fp_compare:nNnF {#1} #2 {#3}	18005
8006	{	18006
8007	#4	18007
8008	\fp_until_do:nNnn {#1} #2 {#3} {#4}	18008
8009	}	18009
8010	}	18010
8011	\cs_new:Npn \fp_while_do:nNnn #1#2#3#4	18011
8012	{	18012
8013	\fp_compare:nNnT {#1} #2 {#3}	18013
8014	{	18014
8015	#4	18015
8016	\fp_while_do:nNnn {#1} #2 {#3} {#4}	18016
8017	}	18017
8018	}	18018

```
8019 \cs_new:Npn \fp_step_function:nnnN #1#2#3
8020 {
8021   \exp_after:wN \__fp_step:wwwN
8022   \exp:w \exp_end_continue_f:w \__fp_parse_o:n {#1}
8023   \exp:w \exp_end_continue_f:w \__fp_parse_o:n {#2}
8024   \exp:w \exp_end_continue_f:w \__fp_parse:n {#3}
8025 }
8026 \cs_generate_variant:Nn \fp_step_function:nnnN { nnnnc }
8027 \cs_new:Npn \__fp_step:wwwN #1#2\__fp_sep: #3#4\__fp_sep: #5#6\__fp_sep: #7
8028 {
8029   \__fp_if_type_fp:NTwFw #1 { } \s__fp \prg_break: \s__fp_stop
8030   \__fp_if_type_fp:NTwFw #3 { } \s__fp \prg_break: \s__fp_stop
8031   \__fp_if_type_fp:NTwFw #5 { } \s__fp \prg_break: \s__fp_stop
8032   \use_i:nnnn
8033   { \__fp_step_fp:wwwN #1#2\__fp_sep: #3#4\__fp_sep: #5#6\__fp_sep: #7 }
8034   \prg_break_point:
8035   \use:n
8036   {
8037     \__fp_error:nfff { step-tuple } { \fp_to_tl:n { #1#2 \__fp_sep: } }
8038     { \fp_to_tl:n { #3#4 \__fp_sep: } } { \fp_to_tl:n { #5#6 \__fp_sep: } }
8039   }
8040 }
8041 \cs_new:Npn \__fp_step_fp:wwwN
8042 #1 \__fp_sep: \s__fp \__fp_chk:w #2#3#4 \__fp_sep: #5\__fp_sep: #6
8043 {
8044   \token_if_eq_meaning:NNTF #2 1
8045   {
8046     \token_if_eq_meaning:NNTF #3 0
8047     { \__fp_step:NnnnnN > }
8048     { \__fp_step:NnnnnN < }
8049   }
8050   {
8051     \token_if_eq_meaning:NNTF #2 0
8052     {
8053       \msg_expandable_error:nnn { kernel }
8054       { zero-step } {#6}
8055     }
8056     {
8057       \__fp_error:nnfn { bad-step } { }
8058       { \fp_to_tl:n { \s__fp \__fp_chk:w #2#3#4 \__fp_sep: } } {#6}
8059     }
8060     \use_none:nnnnn
8061   }
8062   { #1 \__fp_sep: }
8063   { \c_nan_fp }
8064   { \s__fp \__fp_chk:w #2#3#4 \__fp_sep: }
```

```
8065         { #5 \__fp_sep: }
8066         #6
8067     }
8068 \cs_new:Npn \__fp_step:NnnnnN #1#2#3#4#5#6
8069 {
8070     \fp_compare:nNnTF {#2} = {#3}
8071     {
8072         \__fp_error:nffn { tiny-step }
8073         { \fp_to_tl:n {#3} } { \fp_to_tl:n {#4} } {#6}
8074     }
8075     {
8076         \fp_compare:nNnF {#2} #1 {#5}
8077         {
8078             \exp_args:Nf #6 { \__fp_to_decimal_dispatch:w #2 }
8079             \__fp_step:NfnnnN
8080                 #1 { \__fp_parse:n { #2 + #4 } } {#2} {#4} {#5} #6
8081         }
8082     }
8083 }
8084 \cs_generate_variant:Nn \__fp_step:NnnnnN { Nf }
8085 \cs_new_protected:Npn \fp_step_inline:nnnn
8086 {
8087     \int_gincr:N \g__kernel_prg_map_int
8088     \exp_args:NNc \__fp_step:NNnnnn
8089     \cs_gset_protected:Npn
8090         { __fp_map_ \int_use:N \g__kernel_prg_map_int :w }
8091 }
8092 \cs_new_protected:Npn \fp_step_variable:nnnNn #1#2#3#4#5
8093 {
8094     \int_gincr:N \g__kernel_prg_map_int
8095     \exp_args:NNc \__fp_step:NNnnnn
8096     \cs_gset_protected:Npe
8097         { __fp_map_ \int_use:N \g__kernel_prg_map_int :w }
8098         {#1} {#2} {#3}
8099     {
8100         \tl_set:Nn \exp_not:N #4 {##1}
8101         \exp_not:n {#5}
8102     }
8103 }
8104 \cs_new_protected:Npn \__fp_step:NNnnnn #1#2#3#4#5#6
8105 {
8106     #1 #2 ##1 {#6}
8107     \fp_step_function:nnnN {#3} {#4} {#5} #2
8108     \prg_break_point:Nn \scan_stop: { \int_gdecr:N \g__kernel_prg_map_int }
8109 }
8110 \msg_new:nnn { fp } { step-tuple }
```

```
8111 { Tuple~argument~in~fp_step...~{#1}{#2}{#3}. }
8112 \msg_new:nnn { fp } { bad-step }
8113 { Invalid~step~size~#2~for~function~#3. }
8114 \msg_new:nnn { fp } { tiny-step }
8115 { Tiny~step~size~(#1+#2=#1)~for~function~#3. }
8116 \cs_new:Npn \__fp_minmax_o:Nw #1
8117 {
8118   \__fp_parse_function_all_fp_o:fnw
8119   { \token_if_eq_meaning:NNTF 0 #1 { min } { max } }
8120   { \__fp_minmax_aux_o:Nw #1 }
8121 }
8122 \cs_new:Npn \__fp_minmax_aux_o:Nw #1#2 @
8123 {
8124   \if_meaning:w 0 #1
8125     \exp_after:wN \__fp_minmax_loop:Nww \exp_after:wN +
8126   \else:
8127     \exp_after:wN \__fp_minmax_loop:Nww \exp_after:wN -
8128   \fi:
8129   #2
8130   \s__fp \__fp_chk:w 2 #1 \s__fp_exact \__fp_sep:
8131   \s__fp \__fp_chk:w { 3 \__fp_minmax_break_o:w } \__fp_sep:
8132 }
8133 \cs_new:Npn \__fp_minmax_loop:Nww
8134   #1 \s__fp \__fp_chk:w #2#3\__fp_sep: \s__fp \__fp_chk:w #4#5\__fp_sep:
8135 {
8136   \if_meaning:w 3 #4
8137     \if_meaning:w 3 #2
8138       \__fp_minmax_auxi:ww
8139     \else:
8140       \__fp_minmax_auxii:ww
8141     \fi:
8142   \else:
8143     \if_int_compare:w
8144       \__fp_compare_back:ww
8145       \s__fp \__fp_chk:w #4#5\__fp_sep:
8146       \s__fp \__fp_chk:w #2#3\__fp_sep:
8147       = #1 1 \exp_stop_f:
8148       \__fp_minmax_auxii:ww
8149     \else:
8150       \__fp_minmax_auxi:ww
8151     \fi:
8152   \fi:
8153   \__fp_minmax_loop:Nww #1
8154   \s__fp \__fp_chk:w #2#3\__fp_sep:
8155   \s__fp \__fp_chk:w #4#5\__fp_sep:
8156 }
```

```
8157 \cs_new:Npn \__fp_minmax_auxi:ww 1815
8158     #1 \fi: \fi: #2 \s__fp #3 \__fp_sep: \s__fp #4\__fp_sep: 1815
8159     { \fi: \fi: #2 \s__fp #3 \__fp_sep: } 1815
8160 \cs_new:Npn \__fp_minmax_auxii:ww #1 \fi: \fi: #2 \s__fp #3 \__fp_sep: 1816
8161     { \fi: \fi: #2 } 1816
8162 \cs_new:Npn \__fp_minmax_break_o:w #1 \fi: \fi: #2 \s__fp #3\__fp_sep: #4\__fp_sep: 1816
8163     { \fi: \__fp_exp_after_o:w \s__fp #3\__fp_sep: } 1816
8164 \cs_new:Npn \__fp_not_o:w #1 \s__fp \__fp_chk:w #2#3\__fp_sep: @ 1816
8165     { 1816
8166         \if_meaning:w 0 #2 1816
8167         \exp_after:wN \exp_after:wN \exp_after:wN \c_one_fp 1816
8168         \else: 1816
8169         \exp_after:wN \exp_after:wN \exp_after:wN \c_zero_fp 1816
8170         \fi: 1817
8171     } 1817
8172 \cs_new:Npn \__fp_tuple_not_o:w #1 @ { \exp_after:wN \c_zero_fp } 1817
8173 \group_begin: 1817
8174     \char_set_catcode_letter:N & 1817
8175     \char_set_catcode_letter:N | 1817
8176     \cs_new:Npn \__fp_&_o:ww #1 \s__fp \__fp_chk:w #2#3\__fp_sep: 1817
8177     { 1817
8178         \if_meaning:w 0 #2 #1 1817
8179         \__fp_and_return:wNw \s__fp \__fp_chk:w #2#3\__fp_sep: 1817
8180         \fi: 1818
8181         \__fp_exp_after_o:w 1818
8182     } 1818
8183 \cs_new:Npn \__fp_&_tuple_o:ww #1 \s__fp \__fp_chk:w #2#3\__fp_sep: 1818
8184     { 1818
8185         \if_meaning:w 0 #2 #1 1818
8186         \__fp_and_return:wNw \s__fp \__fp_chk:w #2#3\__fp_sep: 1818
8187         \fi: 1818
8188         \__fp_exp_after_tuple_o:w 1818
8189     } 1818
8190 \cs_new:Npn \__fp_tuple_&_o:ww #1\__fp_sep: { \__fp_exp_after_o:w } 1819
8191 \cs_new:Npn \__fp_tuple_&_tuple_o:ww #1\__fp_sep: { \__fp_exp_after_tuple_o:w } 1819
8192 \cs_new:Npn \__fp_|_o:ww { \__fp_&_o:ww \else: } 1819
8193 \cs_new:Npn \__fp_|_tuple_o:ww { \__fp_&_tuple_o:ww \else: } 1819
8194 \cs_new:Npn \__fp_tuple_|_o:ww #1\__fp_sep: #2\__fp_sep: 1819
8195     { \__fp_exp_after_tuple_o:w #1\__fp_sep: } 1819
8196 \cs_new:Npn \__fp_tuple_|_tuple_o:ww #1\__fp_sep: #2\__fp_sep: 1819
8197     { \__fp_exp_after_tuple_o:w #1\__fp_sep: } 1819
8198 \group_end: 1819
8199 \cs_new:Npn \__fp_and_return:wNw #1\__fp_sep: \fi: #2\__fp_sep: 1819
8200     { \fi: \__fp_exp_after_o:w #1\__fp_sep: } 1820
8201 \cs_new:Npn \__fp_ternary:NwwN #1 #2#3@ #4@ #5 1820
8202     { 1820
```

8203	\if_meaning:w __fp_parse_infix_:N #5	18203
8204	\if_charcode:w 0	18204
8205	__fp_if_type_fp:NTwFw	18205
8206	#2 { \use_i:nn __fp_use_i_delimit_by_s_stop:nw #3 \s__fp_stop }	18206
8207	\s__fp 1 \s__fp_stop	18207
8208	\exp_after:wN \exp_after:wN \exp_after:wN __fp_ternary_auxii:NwwN	18208
8209	\else:	18209
8210	\exp_after:wN \exp_after:wN \exp_after:wN __fp_ternary_auxi:NwwN	18210
8211	\fi:	18211
8212	\exp_after:wN #1	18212
8213	\exp:w \exp_end_continue_f:w	18213
8214	__fp_exp_after_array_f:w #4 \s__fp_expr_stop	18214
8215	\exp_after:wN @	18215
8216	\exp:w	18216
8217	__fp_parse_operand:Nw \c__fp_prec_colon_int	18217
8218	__fp_parse_expand:w	18218
8219	\else:	18219
8220	\msg_expandable_error:nnnn	18220
8221	{ fp } { missing } { : } { ~for~?: }	18221
8222	\exp_after:wN __fp_parse_continue:NwN	18222
8223	\exp_after:wN #1	18223
8224	\exp:w \exp_end_continue_f:w	18224
8225	__fp_exp_after_array_f:w #4 \s__fp_expr_stop	18225
8226	\exp_after:wN #5	18226
8227	\exp_after:wN #1	18227
8228	\fi:	18228
8229	}	18229
8230	\cs_new:Npn __fp_ternary_auxi:NwwN #1#2@#3@#4	18230
8231	{	18231
8232	\exp_after:wN __fp_parse_continue:NwN	18232
8233	\exp_after:wN #1	18233
8234	\exp:w \exp_end_continue_f:w	18234
8235	__fp_exp_after_array_f:w #2 \s__fp_expr_stop	18235
8236	#4 #1	18236
8237	}	18237
8238	\cs_new:Npn __fp_ternary_auxii:NwwN #1#2@#3@#4	18238
8239	{	18239
8240	\exp_after:wN __fp_parse_continue:NwN	18240
8241	\exp_after:wN #1	18241
8242	\exp:w \exp_end_continue_f:w	18242
8243	__fp_exp_after_array_f:w #3 \s__fp_expr_stop	18243
8244	#4 #1	18244
8245	}	18245
8246	%% File: l3fp-basics.dtx	18246
8247	\cs_new:Npn __fp_parse_word_abs:N	18247
8248	{ __fp_parse_unary_function:NNN __fp_set_sign_o:w 0 }	18248

```
8249 \cs_new:Npn \__fp_parse_word_logb:N
8250 { \__fp_parse_unary_function:NNN \__fp_logb_o:w ? }
8251 \cs_new:Npn \__fp_parse_word_sign:N
8252 { \__fp_parse_unary_function:NNN \__fp_sign_o:w ? }
8253 \cs_new:Npn \__fp_parse_word_sqrt:N
8254 { \__fp_parse_unary_function:NNN \__fp_sqrt_o:w ? }
8255 \cs_new:cpe { __fp_-_o:ww } \s__fp
8256 {
8257   \exp_not:c { __fp+_o:ww }
8258   \exp_not:n { \s__fp \__fp_neg_sign:N }
8259 }
8260 \cs_new:cpn { __fp+_o:ww }
8261 \s__fp #1 \__fp_chk:w #2 #3 \__fp_sep: \s__fp \__fp_chk:w #4 #5
8262 {
8263   \if_case:w
8264     \if_meaning:w #2 #4
8265       #2
8266     \else:
8267       \if_int_compare:w #2 > #4 \exp_stop_f:
8268         3
8269       \else:
8270         4
8271       \fi:
8272     \fi:
8273     \exp_stop_f:
8274     \exp_after:wN \__fp_add_zeros_o:Nww \int_value:w
8275   \or: \exp_after:wN \__fp_add_normal_o:Nww \int_value:w
8276   \or: \exp_after:wN \__fp_add_inf_o:Nww \int_value:w
8277   \or: \__fp_case_return_i_o:ww
8278   \else: \exp_after:wN \__fp_add_return_ii_o:Nww \int_value:w
8279   \fi:
8280   #1 #5
8281   \s__fp \__fp_chk:w #2 #3 \__fp_sep:
8282   \s__fp \__fp_chk:w #4 #5
8283 }
8284 \cs_new:Npn \__fp_add_return_ii_o:Nww #1 #2 \__fp_sep: \s__fp \__fp_chk:w #3 #4
8285 { \__fp_exp_after_o:w \s__fp \__fp_chk:w #3 #1 }
8286 \cs_new:Npn \__fp_add_zeros_o:Nww #1 \s__fp \__fp_chk:w 0 #2
8287 {
8288   \if_int_compare:w #2 #1 = 20 \exp_stop_f:
8289     \exp_after:wN \__fp_add_return_ii_o:Nww
8290   \else:
8291     \__fp_case_return_i_o:ww
8292   \fi:
8293   #1
8294   \s__fp \__fp_chk:w 0 #2
```



```
8295 }
8296 \cs_new:Npn \__fp_add_inf_o:Nww
8297   #1 \s__fp \__fp_chk:w 2 #2 #3\__fp_sep: \s__fp \__fp_chk:w 2 #4
8298 {
8299   \if_meaning:w #1 #2
8300     \__fp_case_return_i_o:ww
8301   \else:
8302     \__fp_case_use:nw
8303     {
8304       \exp_last_unbraced:Nf \__fp_invalid_operation_o:Nww
8305       { \token_if_eq_meaning:NNTF #1 #4 + - }
8306     }
8307   \fi:
8308   \s__fp \__fp_chk:w 2 #2 #3\__fp_sep:
8309   \s__fp \__fp_chk:w 2 #4
8310 }
8311 \cs_new:Npn \__fp_add_normal_o:Nww #1 \s__fp \__fp_chk:w 1 #2
8312 {
8313   \if_meaning:w #1#2
8314     \exp_after:wN \__fp_add_npos_o:NnwNnw
8315   \else:
8316     \exp_after:wN \__fp_sub_npos_o:NnwNnw
8317   \fi:
8318   #2
8319 }
8320 \cs_new:Npn \__fp_add_npos_o:NnwNnw #1#2#3 \__fp_sep: \s__fp \__fp_chk:w 1 #4 #5
8321 {
8322   \exp_after:wN \__fp_sanitize:Nw
8323   \exp_after:wN #1
8324   \int_value:w \__fp_int_eval:w
8325   \if_int_compare:w #2 > #5 \exp_stop_f:
8326     #2
8327     \exp_after:wN \__fp_add_big_i_o:wNww \int_value:w -
8328   \else:
8329     #5
8330     \exp_after:wN \__fp_add_big_ii_o:wNww \int_value:w
8331   \fi:
8332   \__fp_int_eval:w #5 - #2 \__fp_sep: #1 #3\__fp_sep:
8333 }
8334 \cs_new:Npn \__fp_add_big_i_o:wNww #1\__fp_sep: #2 #3\__fp_sep: #4\__fp_sep:
8335 {
8336   \__fp_decimate:nNnnnn {#1}
8337   \__fp_add_significand_o:NnnwnnnnN
8338   #4
8339   #3
8340   #2
```

```
8341 } 1834
8342 \cs_new:Npn \__fp_add_big_ii_o:wNww #1\__fp_sep: #2 #3\__fp_sep: #4\__fp_sep: 1834
8343 { 1834
8344 \__fp_decimate:nNnnnn {#1} 1834
8345 \__fp_add_significand_o:NnnwnnnnN 1834
8346 #3 1834
8347 #4 1834
8348 #2 1834
8349 } 1834
8350 \cs_new:Npn \__fp_add_significand_o:NnnwnnnnN #1 #2#3 #4\__fp_sep: #5#6#7#8 1835
8351 { 1835
8352 \exp_after:wN \__fp_add_significand_test_o:N 1835
8353 \int_value:w \__fp_int_eval:w 1#5#6 + #2 1835
8354 \exp_after:wN \__fp_add_significand_pack:NNNNNNN 1835
8355 \int_value:w \__fp_int_eval:w 1#7#8 + #3 \__fp_sep: #1 1835
8356 } 1835
8357 \cs_new:Npn \__fp_add_significand_pack:NNNNNNN #1 #2#3#4#5#6#7 1835
8358 { 1835
8359 \if_meaning:w 2 #1 1835
8360 + 1 1836
8361 \fi: 1836
8362 \__fp_sep: #2 #3 #4 #5 #6 #7 \__fp_sep: 1836
8363 } 1836
8364 \cs_new:Npn \__fp_add_significand_test_o:N #1 1836
8365 { 1836
8366 \if_meaning:w 2 #1 1836
8367 \exp_after:wN \__fp_add_significand_carry_o:wwwNN 1836
8368 \else: 1836
8369 \exp_after:wN \__fp_add_significand_no_carry_o:wwwNN 1836
8370 \fi: 1837
8371 } 1837
8372 \cs_new:Npn \__fp_add_significand_no_carry_o:wwwNN 1837
8373 #1\__fp_sep: #2\__fp_sep: #3#4 \__fp_sep: #5#6 1837
8374 { 1837
8375 \exp_after:wN \__fp_basics_pack_high:NNNNNw 1837
8376 \int_value:w \__fp_int_eval:w 1 #1 1837
8377 \exp_after:wN \__fp_basics_pack_low:NNNNNw 1837
8378 \int_value:w \__fp_int_eval:w 1 #2 #3#4 1837
8379 + \__fp_round:NNN #6 #4 #5 1837
8380 \exp_after:wN \__fp_sep: 1838
8381 } 1838
8382 \cs_new:Npn \__fp_add_significand_carry_o:wwwNN 1838
8383 #1\__fp_sep: #2\__fp_sep: #3#4\__fp_sep: #5#6 1838
8384 { 1838
8385 + 1 1838
8386 \exp_after:wN \__fp_basics_pack_weird_high:NNNNNNNNw 1838
```

```
8387 \int_value:w \__fp_int_eval:w 1 1 #1
8388 \exp_after:wN \__fp_basics_pack_weird_low:NNNNw
8389 \int_value:w \__fp_int_eval:w 1 #2#3 +
8390 \exp_after:wN \__fp_round:NNN
8391 \exp_after:wN #6
8392 \exp_after:wN #3
8393 \int_value:w \__fp_round_digit:Nw #4 #5 \__fp_sep:
8394 \exp_after:wN \__fp_sep:
8395 }
8396 \cs_new:Npn \__fp_sub_npos_o:NnwNnw
8397 #1#2#3\__fp_sep: \s__fp \__fp_chk:w 1 #4#5#6\__fp_sep:
8398 {
8399 \if_case:w
8400 \__fp_compare_npos:nwnw {#2} #3\__fp_sep: {#5} #6\__fp_sep: \exp_stop_f:
8401 \exp_after:wN \__fp_sub_eq_o:Nnwnw
8402 \or:
8403 \exp_after:wN \__fp_sub_npos_i_o:Nnwnw
8404 \else:
8405 \exp_after:wN \__fp_sub_npos_ii_o:Nnwnw
8406 \fi:
8407 #1 {#2} #3\__fp_sep: {#5} #6\__fp_sep:
8408 }
8409 \cs_new:Npn \__fp_sub_eq_o:Nnwnw #1#2\__fp_sep: #3\__fp_sep:
8410 { \exp_after:wN \c_zero_fp }
8411 \cs_new:Npn \__fp_sub_npos_ii_o:Nnwnw #1 #2\__fp_sep: #3\__fp_sep:
8412 {
8413 \exp_after:wN \__fp_sub_npos_i_o:Nnwnw
8414 \int_value:w \__fp_neg_sign:N #1
8415 #3\__fp_sep: #2\__fp_sep:
8416 }
8417 \cs_new:Npn \__fp_sub_npos_i_o:Nnwnw #1 #2#3\__fp_sep: #4#5\__fp_sep:
8418 {
8419 \exp_after:wN \__fp_sanitize:Nw
8420 \exp_after:wN #1
8421 \int_value:w \__fp_int_eval:w
8422 #2
8423 \if_int_compare:w #2 = #4 \exp_stop_f:
8424 \exp_after:wN \__fp_sub_back_near_o:nnnnnnnnN
8425 \else:
8426 \exp_after:wN \__fp_decimate:nNnnnn \exp_after:wN
8427 { \int_value:w \__fp_int_eval:w #2 - #4 - 1 \exp_after:wN }
8428 \exp_after:wN \__fp_sub_back_far_o:NnnwnnnnN
8429 \fi:
8430 #5
8431 #3
8432 #1
```

8433 } 18433
8434 \cs_new:Npn __fp_sub_back_near_o:nnnnnnnnN #1#2#3#4 #5#6#7#8 #9 18434
8435 { 18435
8436 \exp_after:wN __fp_sub_back_near_after:wNNNNw 18436
8437 \int_value:w __fp_int_eval:w 10#5#6 - #1#2 - 11 18437
8438 \exp_after:wN __fp_sub_back_near_pack:NNNNNNw 18438
8439 \int_value:w __fp_int_eval:w 11#7#8 - #3#4 \exp_after:wN __fp_sep: 18439
8440 } 18440
8441 \cs_new:Npn __fp_sub_back_near_pack:NNNNNNw #1#2#3#4#5#6#7 __fp_sep: 18441
8442 { + #1#2 __fp_sep: {#3#4#5#6} {#7} __fp_sep: } 18442
8443 \cs_new:Npn __fp_sub_back_near_after:wNNNNw 10 #1#2#3#4 #5 __fp_sep: 18443
8444 { 18444
8445 \if_meaning:w 0 #1 18445
8446 \exp_after:wN __fp_sub_back_shift:wnnnn 18446
8447 \fi: 18447
8448 __fp_sep: {#1#2#3#4} {#5} 18448
8449 } 18449
8450 \cs_new:Npn __fp_sub_back_shift:wnnnn __fp_sep: #1#2 18450
8451 { 18451
8452 \exp_after:wN __fp_sub_back_shift_ii:ww 18452
8453 \int_value:w #1 #2 0 __fp_sep: 18453
8454 } 18454
8455 \cs_new:Npn __fp_sub_back_shift_ii:ww #1 0 __fp_sep: #2#3 __fp_sep: 18455
8456 { 18456
8457 \if_meaning:w @ #1 @ 18457
8458 - 7 18458
8459 - \exp_after:wN \use_i:nnn 18459
8460 \exp_after:wN __fp_sub_back_shift_iii:NNNNNNNNw 18460
8461 \int_value:w #2#3 0 ~ 123456789__fp_sep: 18461
8462 \else: 18462
8463 - __fp_sub_back_shift_iii:NNNNNNNNw #1 123456789__fp_sep: 18463
8464 \fi: 18464
8465 \exp_after:wN __fp_pack_twice_four:wNNNNNNNN 18465
8466 \exp_after:wN __fp_pack_twice_four:wNNNNNNNN 18466
8467 \exp_after:wN __fp_sub_back_shift_iv:nnnnw 18467
8468 \exp_after:wN __fp_sep: 18468
8469 \int_value:w 18469
8470 #1 ~ #2#3 0 ~ 0000 0000 0000 000 __fp_sep: 18470
8471 } 18471
8472 \cs_new:Npn __fp_sub_back_shift_iii:NNNNNNNNw #1#2#3#4#5#6#7#8#9__fp_sep: {#8} 18472
8473 \cs_new:Npn __fp_sub_back_shift_iv:nnnnw #1 __fp_sep: #2 __fp_sep: 18473
8474 { __fp_sep: #1 __fp_sep: } 18474
8475 \cs_new:Npn __fp_sub_back_far_o:NnnwnnnnN #1 #2#3 #4__fp_sep: #5#6#7#8 18475
8476 { 18476
8477 \if_case:w 18477
8478 \if_int_compare:w 1 #2 = #5#6 \use_i:nnnn #7 \exp_stop_f: 18478

```
8479         \if_int_compare:w #3 = \use_none:n #7#8 0 \exp_stop_f:
8480             0
8481         \else:
8482             \if_int_compare:w #3 > \use_none:n #7#8 0 - \fi: 1
8483         \fi:
8484     \else:
8485         \if_int_compare:w 1 #2 > #5#6 \use_i:nnnn #7 - \fi: 1
8486     \fi:
8487     \exp_stop_f:
8488         \exp_after:wN \__fp_sub_back_quite_far_o:wwNN
8489 \or: \exp_after:wN \__fp_sub_back_very_far_o:wwwNN
8490 \else: \exp_after:wN \__fp_sub_back_not_far_o:wwwNN
8491 \fi:
8492 #2 ~ #3 \__fp_sep: #5 #6 ~ #7 #8 \__fp_sep: #1
8493 }
8494 \cs_new:Npn \__fp_sub_back_quite_far_o:wwNN #1\__fp_sep: #2\__fp_sep: #3#4
8495 {
8496     \exp_after:wN \__fp_sub_back_quite_far_ii:NN
8497     \exp_after:wN #3
8498     \exp_after:wN #4
8499 }
8500 \cs_new:Npn \__fp_sub_back_quite_far_ii:NN #1#2
8501 {
8502     \if_case:w \__fp_round_neg:NNN #2 0 #1
8503         \exp_after:wN \use_i:nn
8504     \else:
8505         \exp_after:wN \use_ii:nn
8506     \fi:
8507     { \__fp_sep: {1000} {0000} {0000} {0000} \__fp_sep: }
8508     { - 1 \__fp_sep: {9999} {9999} {9999} {9999} \__fp_sep: }
8509 }
8510 \cs_new:Npn \__fp_sub_back_not_far_o:wwwNN #1 ~ #2\__fp_sep: #3 ~ #4\__fp_sep: #5#6
8511 {
8512     - 1
8513     \exp_after:wN \__fp_sub_back_near_after:wNNNNw
8514     \int_value:w \__fp_int_eval:w 1#30 - #1 - 11
8515     \exp_after:wN \__fp_sub_back_near_pack:NNNNNNw
8516     \int_value:w \__fp_int_eval:w 11 0000 0000 + #40 - #2
8517     - \exp_after:wN \__fp_round_neg:NNN
8518     \exp_after:wN #6
8519     \use_none:nnnnnnn #2 #5
8520     \exp_after:wN \__fp_sep:
8521 }
8522 \cs_new:Npn \__fp_sub_back_very_far_o:wwwNN #1#2#3#4#5#6#7
8523 {
8524     \__fp_pack_eight:wNNNNNNNNN
```

```
8525 \__fp_sub_back_very_far_ii_o:nnNwwNN 1852
8526 { 0 #1#2#3 #4#5#6#7 } 1852
8527 \__fp_sep: 1852
8528 } 1852
8529 \cs_new:Npn \__fp_sub_back_very_far_ii_o:nnNwwNN 1852
8530 #1#2 \__fp_sep: #3 \__fp_sep: #4 ~ #5\__fp_sep: #6#7 1853
8531 { 1853
8532 \exp_after:wN \__fp_basics_pack_high:NNNNNw 1853
8533 \int_value:w \__fp_int_eval:w 1#4 - #1 - 1 1853
8534 \exp_after:wN \__fp_basics_pack_low:NNNNNw 1853
8535 \int_value:w \__fp_int_eval:w 2#5 - #2 1853
8536 - \exp_after:wN \__fp_round_neg:NNN 1853
8537 \exp_after:wN #7 1853
8538 \int_value:w 1853
8539 \if_int_odd:w \__fp_int_eval:w #5 - #2 \__fp_int_eval_end: 1853
8540 1 \else: 2 \fi: 1854
8541 \int_value:w \__fp_round_digit:Nw #3 #6 \__fp_sep: 1854
8542 \exp_after:wN \__fp_sep: 1854
8543 } 1854
8544 \cs_new:cpn { __fp*_o:ww } 1854
8545 { 1854
8546 \__fp_mul_cases_o:NnNww 1854
8547 * 1854
8548 { - 2 + } 1854
8549 \__fp_mul_npos_o:Nww 1854
8550 { } 1855
8551 } 1855
8552 \cs_new:Npn \__fp_mul_cases_o:NnNww 1855
8553 #1#2#3#4 \s__fp \__fp_chk:w #5#6#7\__fp_sep: \s__fp \__fp_chk:w #8#9 1855
8554 { 1855
8555 \if_case:w \__fp_int_eval:w 1855
8556 \if_int_compare:w #5 #8 = 11 ~ 1855
8557 1 1855
8558 \else: 1855
8559 \if_meaning:w 3 #8 1855
8560 3 1856
8561 \else: 1856
8562 \if_meaning:w 3 #5 1856
8563 2 1856
8564 \else: 1856
8565 \if_int_compare:w #5 #8 = 10 ~ 1856
8566 9 #2 - 2 1856
8567 \else: 1856
8568 (#5 #2 #8) / 2 * 2 + 7 1856
8569 \fi: 1856
8570 \fi: 1857
```

```
8571         \fi:
8572     \fi:
8573     \if_meaning:w #6 #9 - 1 \fi:
8574     \__fp_int_eval_end:
8575     \__fp_case_use:nw { #3 0 }
8576 \or: \__fp_case_use:nw { #3 2 }
8577 \or: \__fp_case_return_i_o:ww
8578 \or: \__fp_case_return_ii_o:ww
8579 \or: \__fp_case_return_o:Nww \c_zero_fp
8580 \or: \__fp_case_return_o:Nww \c_minus_zero_fp
8581 \or: \__fp_case_use:nw { \__fp_invalid_operation_o:Nww #1 }
8582 \or: \__fp_case_use:nw { \__fp_invalid_operation_o:Nww #1 }
8583 \or: \__fp_case_return_o:Nww \c_inf_fp
8584 \or: \__fp_case_return_o:Nww \c_minus_inf_fp
8585 #4
8586 \fi:
8587 \s__fp \__fp_chk:w #5 #6 #7 \__fp_sep:
8588 \s__fp \__fp_chk:w #8 #9
8589 }
8590 \cs_new:Npn \__fp_mul_npos_o:Nww
8591 #1 \s__fp \__fp_chk:w #2 #3 #4 #5 \__fp_sep:
8592 \s__fp \__fp_chk:w #6 #7 #8 #9 \__fp_sep:
8593 {
8594 \exp_after:wN \__fp_sanitize:Nw
8595 \exp_after:wN #1
8596 \int_value:w \__fp_int_eval:w
8597 #4 + #8
8598 \__fp_mul_significand_o:nnnnNnnnn #5 #1 #9
8599 }
8600 \cs_new:Npn \__fp_mul_significand_o:nnnnNnnnn #1#2#3#4 #5 #6#7#8#9
8601 {
8602 \exp_after:wN \__fp_mul_significand_test_f:NNN
8603 \exp_after:wN #5
8604 \int_value:w \__fp_int_eval:w 99990000 + #1*#6 +
8605 \exp_after:wN \__fp_mul_significand_keep:NNNNNw
8606 \int_value:w \__fp_int_eval:w 99990000 + #1*#7 + #2*#6 +
8607 \exp_after:wN \__fp_mul_significand_keep:NNNNNw
8608 \int_value:w \__fp_int_eval:w 99990000 + #1*#8 + #2*#7 + #3*#6 +
8609 \exp_after:wN \__fp_mul_significand_drop:NNNNNw
8610 \int_value:w \__fp_int_eval:w 99990000 + #1*#9 + #2*#8 +
8611 #3*#7 + #4*#6 +
8612 \exp_after:wN \__fp_mul_significand_drop:NNNNNw
8613 \int_value:w \__fp_int_eval:w 99990000 + #2*#9 + #3*#8 +
8614 #4*#7 +
8615 \exp_after:wN \__fp_mul_significand_drop:NNNNNw
8616 \int_value:w \__fp_int_eval:w 99990000 + #3*#9 + #4*#8 +
```


8617	\exp_after:wN __fp_mul_significand_drop:NNNNNw	1861
8618	\int_value:w __fp_int_eval:w 100000000 + #4*#9 __fp_sep:	1861
8619	__fp_sep: \exp_after:wN __fp_sep:	1861
8620	}	1862
8621	\cs_new:Npn __fp_mul_significand_drop:NNNNNw #1#2#3#4#5 #6__fp_sep:	1862
8622	{ #1#2#3#4#5 __fp_sep: + #6 }	1862
8623	\cs_new:Npn __fp_mul_significand_keep:NNNNNw #1#2#3#4#5 #6__fp_sep:	1862
8624	{ #1#2#3#4#5 __fp_sep: #6 __fp_sep: }	1862
8625	\cs_new:Npn __fp_mul_significand_test_f:NNN #1 #2 #3	1862
8626	{	1862
8627	\if_meaning:w 0 #3	1862
8628	\exp_after:wN __fp_mul_significand_small_f:NNwwwN	1862
8629	\else:	1862
8630	\exp_after:wN __fp_mul_significand_large_f:NwwNNNN	1863
8631	\fi:	1863
8632	#1 #3	1863
8633	}	1863
8634	\cs_new:Npn __fp_mul_significand_large_f:NwwNNNN	1863
8635	#1 #2__fp_sep: #3__fp_sep: #4#5#6#7__fp_sep: +	1863
8636	{	1863
8637	\exp_after:wN __fp_basics_pack_high:NNNNNw	1863
8638	\int_value:w __fp_int_eval:w 1#2	1863
8639	\exp_after:wN __fp_basics_pack_low:NNNNNw	1863
8640	\int_value:w __fp_int_eval:w 1#3#4#5#6#7	1864
8641	+ \exp_after:wN __fp_round:NNN	1864
8642	\exp_after:wN #1	1864
8643	\exp_after:wN #7	1864
8644	\int_value:w __fp_round_digit:Nw	1864
8645	}	1864
8646	\cs_new:Npn __fp_mul_significand_small_f:NNwwwN	1864
8647	#1 #2#3__fp_sep: #4#5__fp_sep: #6__fp_sep: + #7	1864
8648	{	1864
8649	- 1	1864
8650	\exp_after:wN __fp_basics_pack_high:NNNNNw	1865
8651	\int_value:w __fp_int_eval:w 1#3#4	1865
8652	\exp_after:wN __fp_basics_pack_low:NNNNNw	1865
8653	\int_value:w __fp_int_eval:w 1#5#6#7	1865
8654	+ \exp_after:wN __fp_round:NNN	1865
8655	\exp_after:wN #1	1865
8656	\exp_after:wN #7	1865
8657	\int_value:w __fp_round_digit:Nw	1865
8658	}	1865
8659	\cs_new:cpn { __fp/_o:ww }	1865
8660	{	1866
8661	__fp_mul_cases_o:NnNnw	1866
8662	/	1866

```
8663 { - }
8664 \__fp_div_npos_o:Nww
8665 {
8666   \or:
8667     \__fp_case_use:nw
8668     { \__fp_division_by_zero_o:NNww \c_inf_fp / }
8669   \or:
8670     \__fp_case_use:nw
8671     { \__fp_division_by_zero_o:NNww \c_minus_inf_fp / }
8672 }
8673 }
8674 \cs_new:Npn \__fp_div_npos_o:Nww
8675   #1 \s__fp \__fp_chk:w 1 #2 #3 #4 \__fp_sep:
8676   \s__fp \__fp_chk:w 1 #5 #6 #7#8#9\__fp_sep:
8677 {
8678   \exp_after:wN \__fp_sanitize:Nw
8679   \exp_after:wN #1
8680   \int_value:w \__fp_int_eval:w
8681   #3 - #6
8682   \exp_after:wN \__fp_div_significand_i_o:wnnw
8683   \int_value:w \__fp_int_eval:w #7 \use_i:nnnn #8 + 1 \__fp_sep:
8684   #4
8685   {#7}{#8}#9 \__fp_sep:
8686   #1
8687 }
8688 \cs_new:Npn \__fp_div_significand_i_o:wnnw #1 \__fp_sep: #2#3 #4 \__fp_sep:
8689 {
8690   \exp_after:wN \__fp_div_significand_test_o:w
8691   \int_value:w \__fp_int_eval:w
8692   \exp_after:wN \__fp_div_significand_calc:wnnnnnnnn
8693   \int_value:w \__fp_int_eval:w 999999 + #2 #3 0 / #1 \__fp_sep:
8694   #2 #3 \__fp_sep:
8695   #4
8696   { \exp_after:wN \__fp_div_significand_ii:wwn \int_value:w #1 }
8697   { \exp_after:wN \__fp_div_significand_ii:wwn \int_value:w #1 }
8698   { \exp_after:wN \__fp_div_significand_ii:wwn \int_value:w #1 }
8699   { \exp_after:wN \__fp_div_significand_iii:wnnnnnn \int_value:w #1 }
8700 }
8701 \cs_new:Npn \__fp_div_significand_calc:wnnnnnnnn 1#1
8702 {
8703   \if_meaning:w 1 #1
8704     \exp_after:wN \__fp_div_significand_calc_i:wnnnnnnnn
8705   \else:
8706     \exp_after:wN \__fp_div_significand_calc_ii:wnnnnnnnn
8707   \fi:
8708 }
```

```
8709 \cs_new:Npn \__fp_div_significand_calc_i:wwnnnnnnnn 18709
8710 #1\__fp_sep: #2\__fp_sep:#3#4 #5#6#7#8 #9 18710
8711 { 18711
8712 1 1 #1 18712
8713 #9 \exp_after:wN \__fp_sep: 18713
8714 \int_value:w \__fp_int_eval:w \c__fp_Bigg_leading_shift_int 18714
8715 + #2 - #1 * #5 - #5#60 18715
8716 \exp_after:wN \__fp_pack_Bigg:NNNNNNw 18716
8717 \int_value:w \__fp_int_eval:w \c__fp_Bigg_middle_shift_int 18717
8718 + #3 - #1 * #6 - #70 18718
8719 \exp_after:wN \__fp_pack_Bigg:NNNNNNw 18719
8720 \int_value:w \__fp_int_eval:w \c__fp_Bigg_middle_shift_int 18720
8721 + #4 - #1 * #7 - #80 18721
8722 \exp_after:wN \__fp_pack_Bigg:NNNNNNw 18722
8723 \int_value:w \__fp_int_eval:w \c__fp_Bigg_trailing_shift_int 18723
8724 - #1 * #8 \__fp_sep: 18724
8725 {#5}{#6}{#7}{#8} 18725
8726 } 18726
8727 \cs_new:Npn \__fp_div_significand_calc_ii:wwnnnnnnnn 18727
8728 #1\__fp_sep: #2\__fp_sep:#3#4 #5#6#7#8 #9 18728
8729 { 18729
8730 1 0 #1 18730
8731 #9 \exp_after:wN \__fp_sep: 18731
8732 \int_value:w \__fp_int_eval:w \c__fp_Bigg_leading_shift_int 18732
8733 + #2 - #1 * #5 18733
8734 \exp_after:wN \__fp_pack_Bigg:NNNNNNw 18734
8735 \int_value:w \__fp_int_eval:w \c__fp_Bigg_middle_shift_int 18735
8736 + #3 - #1 * #6 18736
8737 \exp_after:wN \__fp_pack_Bigg:NNNNNNw 18737
8738 \int_value:w \__fp_int_eval:w \c__fp_Bigg_middle_shift_int 18738
8739 + #4 - #1 * #7 18739
8740 \exp_after:wN \__fp_pack_Bigg:NNNNNNw 18740
8741 \int_value:w \__fp_int_eval:w \c__fp_Bigg_trailing_shift_int 18741
8742 - #1 * #8 \__fp_sep: 18742
8743 {#5}{#6}{#7}{#8} 18743
8744 } 18744
8745 \cs_new:Npn \__fp_div_significand_ii:wnn #1\__fp_sep: #2\__fp_sep:#3 18745
8746 { 18746
8747 \exp_after:wN \__fp_div_significand_pack:NNN 18747
8748 \int_value:w \__fp_int_eval:w 18748
8749 \exp_after:wN \__fp_div_significand_calc:wwnnnnnnnn 18749
8750 \int_value:w \__fp_int_eval:w 18750
8751 999999 + #2 #3 0 / #1 \__fp_sep: #2 #3 \__fp_sep: 18751
8752 } 18752
8753 \cs_new:Npn \__fp_div_significand_iii:wwnnnnnn #1\__fp_sep: #2\__fp_sep:#3#4#5 #6#7 18753
8754 { 18754
```

```

8755 0
8756 \exp_after:wN \__fp_div_significand_iv:wwnnnnnnnn
8757 \int_value:w \__fp_int_eval:w ( 2 * #2 #3) / #6 #7 \__fp_sep: % <- P
8758 #2 \__fp_sep: {#3} {#4} {#5}
8759 {#6} {#7}
8760 }
8761 \cs_new:Npn \__fp_div_significand_iv:wwnnnnnnnn
8762 #1\__fp_sep: #2\__fp_sep:#3#4#5 #6#7#8#9
8763 {
8764 + 5 * #1
8765 \exp_after:wN \__fp_div_significand_vi:Nw
8766 \int_value:w \__fp_int_eval:w -50 + 2*#2#3 - #1*#6#7 +
8767 \exp_after:wN \__fp_div_significand_v:NN
8768 \int_value:w \__fp_int_eval:w 499950 + 2*#4 - #1*#8 +
8769 \exp_after:wN \__fp_div_significand_v:NN
8770 \int_value:w \__fp_int_eval:w 500000 + 2*#5 - #1*#9 \__fp_sep:
8771 }
8772 \cs_new:Npn \__fp_div_significand_v:NN #1#2 { #1#2 \__fp_int_eval_end: + }
8773 \cs_new:Npn \__fp_div_significand_vi:Nw #1#2\__fp_sep:
8774 {
8775 \if_meaning:w 0 #1
8776 \if_int_compare:w \__fp_int_eval:w #2 > 0 + 1 \fi:
8777 \else:
8778 \if_meaning:w - #1 - \else: + \fi: 1
8779 \fi:
8780 \__fp_sep:
8781 }
8782 \cs_new:Npn \__fp_div_significand_pack:NNN 1 #1 #2 { + #1 #2 \__fp_sep: }
8783 \cs_new:Npn \__fp_div_significand_test_o:w 10 #1
8784 {
8785 \if_meaning:w 0 #1
8786 \exp_after:wN \__fp_div_significand_small_o:wwwNNNNwN
8787 \else:
8788 \exp_after:wN \__fp_div_significand_large_o:wwwNNNNwN
8789 \fi:
8790 #1
8791 }
8792 \cs_new:Npn \__fp_div_significand_small_o:wwwNNNNwN
8793 0 #1\__fp_sep: #2\__fp_sep: #3\__fp_sep: #4#5#6#7#8\__fp_sep: #9
8794 {
8795 \exp_after:wN \__fp_basics_pack_high:NNNNw
8796 \int_value:w \__fp_int_eval:w 1 #1#2
8797 \exp_after:wN \__fp_basics_pack_low:NNNNw
8798 \int_value:w \__fp_int_eval:w 1 #3#4#5#6#7
8799 + \__fp_round:NNN #9 #7 #8
8800 \exp_after:wN \__fp_sep:

```

```
8801 }
8802 \cs_new:Npn \__fp_div_significand_large_o:wwwNNNNwN
8803 #1\__fp_sep: #2\__fp_sep: #3\__fp_sep: #4#5#6#7#8\__fp_sep: #9
8804 {
8805 + 1
8806 \exp_after:wN \__fp_basics_pack_weird_high:NNNNNNNNw
8807 \int_value:w \__fp_int_eval:w 1 #1 #2
8808 \exp_after:wN \__fp_basics_pack_weird_low:NNNNw
8809 \int_value:w \__fp_int_eval:w 1 #3 #4 #5 #6 +
8810 \exp_after:wN \__fp_round:NNN
8811 \exp_after:wN #9
8812 \exp_after:wN #6
8813 \int_value:w \__fp_round_digit:Nw #7 #8 \__fp_sep:
8814 \exp_after:wN \__fp_sep:
8815 }
8816 \cs_new:Npn \__fp_sqrt_o:w #1 \s__fp \__fp_chk:w #2#3#4\__fp_sep: @
8817 {
8818 \if_meaning:w 0 #2 \__fp_case_return_same_o:w \fi:
8819 \if_meaning:w 2 #3
8820 \__fp_case_use:nw { \__fp_invalid_operation_o:nw { sqrt } }
8821 \fi:
8822 \if_meaning:w 1 #2 \else: \__fp_case_return_same_o:w \fi:
8823 \__fp_sqrt_npos_o:w
8824 \s__fp \__fp_chk:w #2 #3 #4\__fp_sep:
8825 }
8826 \cs_new:Npn \__fp_sqrt_npos_o:w \s__fp \__fp_chk:w 1 0 #1#2#3#4#5\__fp_sep:
8827 {
8828 \exp_after:wN \__fp_sanitize:Nw
8829 \exp_after:wN 0
8830 \int_value:w \__fp_int_eval:w
8831 \if_int_odd:w #1 \exp_stop_f:
8832 \exp_after:wN \__fp_sqrt_npos_auxi_o:wwnnN
8833 \fi:
8834 #1 / 2
8835 \__fp_sqrt_Newton_o:wwn 56234133\__fp_sep: 0\__fp_sep: {#2#3} {#4#5} 0
8836 }
8837 \cs_new:Npn \__fp_sqrt_npos_auxi_o:wwnnN #1 / 2 #2\__fp_sep: 0\__fp_sep: #3#4#5
8838 {
8839 ( #1 + 1 ) / 2
8840 \__fp_pack_eight:wNNNNNNNNN
8841 \__fp_sqrt_npos_auxii_o:wwnnNNNNN
8842 \__fp_sep:
8843 0 #3 #4
8844 }
8845 \cs_new:Npn \__fp_sqrt_npos_auxii_o:wwnnNNNNN #1\__fp_sep: #2#3#4#5#6#7#8#9
8846 { \__fp_sqrt_Newton_o:wwn 17782794\__fp_sep: 0\__fp_sep: {#1}{#2#3#4#5#6#7#8#9} }
```

```
8847 \cs_new:Npn \__fp_sqrt_Newton_o:wnn #1\__fp_sep: #2\__fp_sep: #3
8848 {
8849   \if_int_compare:w #1 = #2 \exp_stop_f:
8850   \exp_after:wN \__fp_sqrt_auxi_o:NNNNwnnnN
8851   \int_value:w \__fp_int_eval:w 9999 9999 +
8852   \exp_after:wN \__fp_use_none_until_s:w
8853   \fi:
8854   \exp_after:wN \__fp_sqrt_Newton_o:wnn
8855   \int_value:w \__fp_int_eval:w (#1 + #3 * 1 0000 0000 / #1) / 2 \__fp_sep:
8856   #1\__fp_sep: {#3}
8857 }
8858 \cs_new:Npn \__fp_sqrt_auxi_o:NNNNwnnnN 1 #1#2#3#4#5\__fp_sep:
8859 {
8860   \__fp_sqrt_auxii_o:NnnnnnnnnN
8861   \__fp_sqrt_auxiii_o:wnnnnnnnnn
8862   {#1#2#3#4} {#5} {2499} {9988} {7500}
8863 }
8864 \cs_new:Npn \__fp_sqrt_auxii_o:NnnnnnnnnN #1 #2#3#4#5#6 #7#8#9
8865 {
8866   \exp_after:wN #1
8867   \int_value:w \__fp_int_eval:w \c__fp_big_leading_shift_int
8868   + #7 - #2 * #2
8869   \exp_after:wN \__fp_pack_big:NNNNNNw
8870   \int_value:w \__fp_int_eval:w \c__fp_big_middle_shift_int
8871   - 2 * #2 * #3
8872   \exp_after:wN \__fp_pack_big:NNNNNNw
8873   \int_value:w \__fp_int_eval:w \c__fp_big_middle_shift_int
8874   + #8 - #3 * #3 - 2 * #2 * #4
8875   \exp_after:wN \__fp_pack_big:NNNNNNw
8876   \int_value:w \__fp_int_eval:w \c__fp_big_middle_shift_int
8877   - 2 * #3 * #4 - 2 * #2 * #5
8878   \exp_after:wN \__fp_pack_big:NNNNNNw
8879   \int_value:w \__fp_int_eval:w \c__fp_big_middle_shift_int
8880   + #9 000 0000 - #4 * #4 - 2 * #3 * #5 - 2 * #2 * #6
8881   \exp_after:wN \__fp_pack_big:NNNNNNw
8882   \int_value:w \__fp_int_eval:w \c__fp_big_middle_shift_int
8883   - 2 * #4 * #5 - 2 * #3 * #6
8884   \exp_after:wN \__fp_pack_big:NNNNNNw
8885   \int_value:w \__fp_int_eval:w \c__fp_big_middle_shift_int
8886   - #5 * #5 - 2 * #4 * #6
8887   \exp_after:wN \__fp_pack_big:NNNNNNw
8888   \int_value:w \__fp_int_eval:w
8889   \c__fp_big_middle_shift_int
8890   - 2 * #5 * #6
8891   \exp_after:wN \__fp_pack_big:NNNNNNw
8892   \int_value:w \__fp_int_eval:w
```



```
8893         \c__fp_big_trailing_shift_int 18893
8894         - #6 * #6 \__fp_sep: 18894
8895 % ( 18895
8896 - 257 ) * 5000 0000 / (#2#3 + 1) + 10 0000 0000 \__fp_sep: 18896
8897 {#2}{#3}{#4}{#5}{#6} {#7}{#8}#9 18897
8898 } 18898
8899 \cs_new:Npn \__fp_sqrt_auxiii_o:wnnnnnnnn 18899
8900 #1\__fp_sep: #2#3#4#5#6#7#8#9 18900
8901 { 18901
8902 \if_int_compare:w #1 > \c_one_int 18902
8903 \exp_after:wN \__fp_sqrt_auxiv_o:NNNNNw 18903
8904 \int_value:w \__fp_int_eval:w (#1#2 %) 18904
8905 \else: 18905
8906 \if_int_compare:w #1#2 > \c_one_int 18906
8907 \exp_after:wN \__fp_sqrt_auxv_o:NNNNNw 18907
8908 \int_value:w \__fp_int_eval:w (#1#2#3 %) 18908
8909 \else: 18909
8910 \if_int_compare:w #1#2#3 > \c_one_int 18910
8911 \exp_after:wN \__fp_sqrt_auxvi_o:NNNNNw 18911
8912 \int_value:w \__fp_int_eval:w (#1#2#3#4 %) 18912
8913 \else: 18913
8914 \exp_after:wN \__fp_sqrt_auxvii_o:NNNNNw 18914
8915 \int_value:w \__fp_int_eval:w (#1#2#3#4#5 %) 18915
8916 \fi: 18916
8917 \fi: 18917
8918 \fi: 18918
8919 } 18919
8920 \cs_new:Npn \__fp_sqrt_auxiv_o:NNNNNw 1#1#2#3#4#5#6\__fp_sep: 18920
8921 { \__fp_sqrt_auxviii_o:nnnnnnn {#1#2#3#4#5#6} {00000000} } 18921
8922 \cs_new:Npn \__fp_sqrt_auxv_o:NNNNNw 1#1#2#3#4#5#6\__fp_sep: 18922
8923 { \__fp_sqrt_auxviii_o:nnnnnnn {000#1#2#3#4#5} {#60000} } 18923
8924 \cs_new:Npn \__fp_sqrt_auxvi_o:NNNNNw 1#1#2#3#4#5#6\__fp_sep: 18924
8925 { \__fp_sqrt_auxviii_o:nnnnnnn {0000000#1} {#2#3#4#5#6} } 18925
8926 \cs_new:Npn \__fp_sqrt_auxvii_o:NNNNNw 1#1#2#3#4#5#6\__fp_sep: 18926
8927 { 18927
8928 \if_int_compare:w #1#2 = \c_zero_int 18928
8929 \exp_after:wN \__fp_sqrt_auxx_o:Nnnnnnnn 18929
8930 \fi: 18930
8931 \__fp_sqrt_auxviii_o:nnnnnnn {00000000} {000#1#2#3#4#5} 18931
8932 } 18932
8933 \cs_new:Npn \__fp_sqrt_auxviii_o:nnnnnnn #1#2 #3#4#5#6#7 18933
8934 { 18934
8935 \exp_after:wN \__fp_sqrt_auxix_o:wnnwnw 18935
8936 \int_value:w \__fp_int_eval:w #3 18936
8937 \exp_after:wN \__fp_basics_pack_low:NNNNNw 18937
8938 \int_value:w \__fp_int_eval:w #1 + 1#4#5 18938
```



```
8939         \exp_after:wN \__fp_basics_pack_low:NNNNNw
8940         \int_value:w \__fp_int_eval:w #2 + 1#6#7 \__fp_sep:
8941     }
8942 \cs_new:Npn \__fp_sqrt_auxix_o:wnwnw #1\__fp_sep: #2#3\__fp_sep: #4#5\__fp_sep:
8943 {
8944     \__fp_sqrt_auxii_o:NnnnnnnnnN
8945     \__fp_sqrt_auxiii_o:wnnnnnnnn {#1}{#2}{#3}{#4}{#5}
8946 }
8947 \cs_new:Npn \__fp_sqrt_auxx_o:Nnnnnnnn #1#2#3 #4#5#6#7#8
8948 {
8949     \exp_after:wN \__fp_sqrt_auxxi_o:wwnnN
8950     \int_value:w \__fp_int_eval:w
8951     (#8 + 2499) / 5000 * 5000 \__fp_sep:
8952     {#4} {#5} {#6} {#7} \__fp_sep:
8953 }
8954 \cs_new:Npn \__fp_sqrt_auxxi_o:wwnnN #1\__fp_sep: #2\__fp_sep: #3#4#5
8955 {
8956     \__fp_sqrt_auxii_o:NnnnnnnnnN
8957     \__fp_sqrt_auxxii_o:nnnnnnnnnw
8958     #2 {#1}
8959     {#3} { #4 + 1 } #5
8960 }
8961 \cs_new:Npn \__fp_sqrt_auxxii_o:nnnnnnnnw 0\__fp_sep: #1#2#3#4#5#6#7#8 #9\__fp_sep:
8962 {
8963     \if_int_compare:w #1#2 > \c_zero_int
8964     \if_int_compare:w #1#2 = \c_one_int
8965     \if_int_compare:w #3#4 = \c_zero_int
8966     \if_int_compare:w #5#6 = \c_zero_int
8967     \if_int_compare:w #7#8 = \c_zero_int
8968     \__fp_sqrt_auxxiii_o:w
8969     \fi:
8970     \fi:
8971     \fi:
8972     \fi:
8973     \exp_after:wN \__fp_sqrt_auxxiv_o:wnnnnnnnnN
8974     \int_value:w 9998
8975 \else:
8976     \exp_after:wN \__fp_sqrt_auxxiv_o:wnnnnnnnnN
8977     \int_value:w 10000
8978     \fi:
8979     \__fp_sep:
8980 }
8981 \cs_new:Npn \__fp_sqrt_auxxiii_o:w \fi: \fi: \fi: \fi: #1 \fi: \__fp_sep:
8982 {
8983     \fi: \fi: \fi: \fi: \fi:
8984     \__fp_sqrt_auxxiv_o:wnnnnnnnnN 9999 \__fp_sep:
```

```
8985 }
8986 \cs_new:Npn \__fp_sqrt_auxxiv_o:w nnnnnnnnN #1\__fp_sep: #2#3#4#5#6 #7#8#9
8987 {
8988   \exp_after:wN \__fp_basics_pack_high:NNNNNw
8989   \int_value:w \__fp_int_eval:w 1 0000 0000 + #2#3
8990   \exp_after:wN \__fp_basics_pack_low:NNNNNw
8991   \int_value:w \__fp_int_eval:w 1 0000 0000
8992   + #4#5
8993   \if_int_compare:w #6 > #1 \exp_stop_f: + 1 \fi:
8994   + \exp_after:wN \__fp_round:NNN
8995   \exp_after:wN 0
8996   \exp_after:wN 0
8997   \int_value:w
8998   \exp_after:wN \use_i:nn
8999   \exp_after:wN \__fp_round_digit:Nw
9000   \int_value:w \__fp_int_eval:w #6 + 19999 - #1 \__fp_sep:
9001   \exp_after:wN \__fp_sep:
9002 }
9003 \cs_new:Npn \__fp_logb_o:w ? \s__fp \__fp_chk:w #1#2\__fp_sep: @
9004 {
9005   \if_case:w #1 \exp_stop_f:
9006     \__fp_case_use:nw
9007     { \__fp_division_by_zero_o:Nnw \c_minus_inf_fp { logb } }
9008   \or: \exp_after:wN \__fp_logb_aux_o:w
9009   \or: \__fp_case_return_o:Nw \c_inf_fp
9010   \else: \__fp_case_return_same_o:w
9011   \fi:
9012   \s__fp \__fp_chk:w #1 #2\__fp_sep:
9013 }
9014 \cs_new:Npn \__fp_logb_aux_o:w \s__fp \__fp_chk:w #1 #2 #3 #4 \__fp_sep:
9015 {
9016   \exp_after:wN \__fp_parse:n \exp_after:wN
9017   { \int_value:w \int_eval:w #3 - 1 \exp_after:wN }
9018 }
9019 \cs_new:Npn \__fp_sign_o:w ? \s__fp \__fp_chk:w #1#2\__fp_sep: @
9020 {
9021   \if_case:w #1 \exp_stop_f:
9022     \__fp_case_return_same_o:w
9023   \or: \exp_after:wN \__fp_sign_aux_o:w
9024   \or: \exp_after:wN \__fp_sign_aux_o:w
9025   \else: \__fp_case_return_same_o:w
9026   \fi:
9027   \s__fp \__fp_chk:w #1 #2\__fp_sep:
9028 }
9029 \cs_new:Npn \__fp_sign_aux_o:w \s__fp \__fp_chk:w #1 #2 #3 \__fp_sep:
9030 { \exp_after:wN \__fp_set_sign_o:w \exp_after:wN #2 \c_one_fp@ }
```

```

9031 \cs_new:Npn \__fp_set_sign_o:w #1 \s__fp \__fp_chk:w #2#3#4\__fp_sep: @
9032 {
9033   \exp_after:wN \__fp_exp_after_o:w
9034   \exp_after:wN \s__fp
9035   \exp_after:wN \__fp_chk:w
9036   \exp_after:wN #2
9037   \int_value:w
9038   \if_case:w #3 \exp_stop_f: #1 \or: 1 \or: 0 \fi: \exp_stop_f:
9039   #4\__fp_sep:
9040 }
9041 \cs_new:Npn \__fp_tuple_set_sign_o:w #1#2 @
9042 {
9043   \if_meaning:w 2 #1
9044     \exp_after:wN \__fp_tuple_set_sign_aux_o:Nnw
9045     \fi:
9046     \__fp_invalid_operation_o:nw { abs }
9047     #2
9048 }
9049 \cs_new:Npn \__fp_tuple_set_sign_aux_o:Nnw #1#2
9050 { \__fp_tuple_map_o:nw \__fp_tuple_set_sign_aux_o:w }
9051 \cs_new:Npn \__fp_tuple_set_sign_aux_o:w #1#2 \__fp_sep:
9052 {
9053   \__fp_change_func_type:NNN #1 \__fp_set_sign_o:w
9054   \__fp_parse_apply_unary_error:NNw
9055   2 #1 #2 \__fp_sep: @
9056 }
9057 \cs_new:cpn { \__fp_*_tuple_o:ww } #1 \__fp_sep:
9058 { \__fp_tuple_map_o:nw { \__fp_binary_type_o:Nww * #1 \__fp_sep: } }
9059 \cs_new:cpn { \__fp_tuple_*_o:ww } #1 \__fp_sep: #2 \__fp_sep:
9060 {
9061   \__fp_tuple_map_o:nw { \__fp_binary_rev_type_o:Nww * #2 \__fp_sep: }
9062   #1 \__fp_sep:
9063 }
9064 \cs_new:cpn { \__fp_tuple/_o:ww } #1 \__fp_sep: #2 \__fp_sep:
9065 {
9066   \__fp_tuple_map_o:nw { \__fp_binary_rev_type_o:Nww / #2 \__fp_sep: }
9067   #1 \__fp_sep:
9068 }
9069 \cs_set_protected:Npn \__fp_tmp:w #1
9070 {
9071   \cs_new:cpn { \__fp_tuple_#1_tuple_o:ww }
9072     \s__fp_tuple \__fp_tuple_chk:w ##1 \__fp_sep:
9073     \s__fp_tuple \__fp_tuple_chk:w ##2 \__fp_sep:
9074     {
9075       \int_compare:nNnTF
9076         { \__fp_array_count:n {##1} } = { \__fp_array_count:n {##2} }

```

```

9077         { \__fp_tuple_mapthread_o:nww { \__fp_binary_type_o:Nww #1 } }
9078         { \__fp_invalid_operation_o:nww #1 }
9079         \s__fp_tuple \__fp_tuple_chk:w {##1} \__fp_sep:
9080         \s__fp_tuple \__fp_tuple_chk:w {##2} \__fp_sep:
9081     }
9082 }
9083 \__fp_tmp:w +
9084 \__fp_tmp:w -
9085 %% File: l3fp-extended.dtx
9086 \tl_const:Nn \c__fp_one_fixed_tl
9087   { {10000} {0000} {0000} {0000} {0000} {0000} \__fp_sep: }
9088 \cs_new:Npn \__fp_fixed_continue:wn #1\__fp_sep: #2 { #2 #1\__fp_sep: }
9089 \cs_new:Npn \__fp_fixed_add_one:wN #1#2\__fp_sep: #3
9090   {
9091     \exp_after:wN #3 \exp_after:wN
9092     { \int_value:w \__fp_int_eval:w \c__fp_myriad_int + #1 } #2 \__fp_sep:
9093   }
9094 \cs_new:Npn \__fp_fixed_div_myriad:wn #1#2#3#4#5#6\__fp_sep:
9095   {
9096     \exp_after:wN \__fp_fixed_mul_after:wn
9097     \int_value:w \__fp_int_eval:w \c__fp_leading_shift_int
9098     \exp_after:wN \__fp_pack:NNNNNw
9099     \int_value:w \__fp_int_eval:w \c__fp_trailing_shift_int
9100     + #1 \__fp_sep: {#2}{#3}{#4}{#5}\__fp_sep:
9101   }
9102 \cs_new:Npn \__fp_fixed_mul_after:wn #1\__fp_sep: #2\__fp_sep: #3
9103   { #3 {#1} #2\__fp_sep: }
9104 \cs_new:Npn \__fp_fixed_mul_short:wn #1#2#3#4#5#6\__fp_sep: #7#8#9\__fp_sep:
9105   {
9106     \exp_after:wN \__fp_fixed_mul_after:wn
9107     \int_value:w \__fp_int_eval:w \c__fp_leading_shift_int
9108     + #1*#7
9109     \exp_after:wN \__fp_pack:NNNNNw
9110     \int_value:w \__fp_int_eval:w \c__fp_middle_shift_int
9111     + #1*#8 + #2*#7
9112     \exp_after:wN \__fp_pack:NNNNNw
9113     \int_value:w \__fp_int_eval:w \c__fp_middle_shift_int
9114     + #1*#9 + #2*#8 + #3*#7
9115     \exp_after:wN \__fp_pack:NNNNNw
9116     \int_value:w \__fp_int_eval:w \c__fp_middle_shift_int
9117     + #2*#9 + #3*#8 + #4*#7
9118     \exp_after:wN \__fp_pack:NNNNNw
9119     \int_value:w \__fp_int_eval:w \c__fp_middle_shift_int
9120     + #3*#9 + #4*#8 + #5*#7
9121     \exp_after:wN \__fp_pack:NNNNNw
9122     \int_value:w \__fp_int_eval:w \c__fp_trailing_shift_int

```

```
9123         + #4*#9 + #5*#8 + #6*#7
9124         + ( #5*#9 + #6*#8 + #6*#9 / \c__fp_myriad_int )
9125         / \c__fp_myriad_int \__fp_sep: \__fp_sep:
9126     }
9127 \cs_new:Npn \__fp_fixed_div_int:wnN #1#2#3#4#5#6 \__fp_sep: #7 \__fp_sep: #8
9128 {
9129     \exp_after:wN \__fp_fixed_div_int_after:Nw
9130     \exp_after:wN #8
9131     \int_value:w \__fp_int_eval:w - 1
9132     \__fp_fixed_div_int:wnN
9133     #1\__fp_sep: {#7} \__fp_fixed_div_int_auxi:wnn
9134     #2\__fp_sep: {#7} \__fp_fixed_div_int_auxi:wnn
9135     #3\__fp_sep: {#7} \__fp_fixed_div_int_auxi:wnn
9136     #4\__fp_sep: {#7} \__fp_fixed_div_int_auxi:wnn
9137     #5\__fp_sep: {#7} \__fp_fixed_div_int_auxi:wnn
9138     #6\__fp_sep: {#7} \__fp_fixed_div_int_auxii:wnn \__fp_sep:
9139 }
9140 \cs_new:Npn \__fp_fixed_div_int:wnN #1\__fp_sep: #2 #3
9141 {
9142     \exp_after:wN #3
9143     \int_value:w \__fp_int_eval:w #1 / #2 - 1 \__fp_sep:
9144     {#2}
9145     {#1}
9146 }
9147 \cs_new:Npn \__fp_fixed_div_int_auxi:wnn #1\__fp_sep: #2 #3
9148 {
9149     + #1
9150     \exp_after:wN \__fp_fixed_div_int_pack:Nw
9151     \int_value:w \__fp_int_eval:w 9999
9152     \exp_after:wN \__fp_fixed_div_int:wnN
9153     \int_value:w \__fp_int_eval:w #3 - #1*#2 \__fp_int_eval_end:
9154 }
9155 \cs_new:Npn \__fp_fixed_div_int_auxii:wnn #1\__fp_sep: #2 #3 { + #1 + 2 \__fp_sep: }
9156 \cs_new:Npn \__fp_fixed_div_int_pack:Nw #1 #2\__fp_sep: { + #1\__fp_sep: {#2} }
9157 \cs_new:Npn \__fp_fixed_div_int_after:Nw #1 #2\__fp_sep: { #1 {#2} }
9158 \cs_new:Npn \__fp_fixed_add:wnn { \__fp_fixed_add:Nnnnnwnn + }
9159 \cs_new:Npn \__fp_fixed_sub:wnn { \__fp_fixed_add:Nnnnnwnn - }
9160 \cs_new:Npn \__fp_fixed_add:Nnnnnwnn #1 #2#3#4#5 #6\__fp_sep: #7#8
9161 {
9162     \exp_after:wN \__fp_fixed_add_after:NNNNNwn
9163     \int_value:w \__fp_int_eval:w 9 9999 9998 + #2#3 #1 #7#8
9164     \exp_after:wN \__fp_fixed_add_pack:NNNNNwn
9165     \int_value:w \__fp_int_eval:w 1 9999 9998 + #4#5
9166     \__fp_fixed_add:nnNnnwn #6 #1
9167 }
9168 \cs_new:Npn \__fp_fixed_add:nnNnnwn #1#2 #3 #4#5 #6#7 \__fp_sep: #8
```

```

9169 {
9170     #3 #4#5
9171     \exp_after:wN \__fp_fixed_add_pack:NNNNNwn
9172     \int_value:w \__fp_int_eval:w
9173     2 0000 0000 #3 #6#7 + #1#2 \__fp_sep: {#8} \__fp_sep:
9174 }
9175 \cs_new:Npn \__fp_fixed_add_pack:NNNNNwn #1 #2#3#4#5 #6\__fp_sep: #7
9176 { + #1 \__fp_sep: {#7} {#2#3#4#5} {#6} }
9177 \cs_new:Npn \__fp_fixed_add_after:NNNNNwn 1 #1 #2#3#4#5 #6\__fp_sep: #7
9178 { #7 {#1#2#3#4#5} {#6} }
9179 \cs_new:Npn \__fp_fixed_mul:wwn #1#2#3#4 #5\__fp_sep: #6#7#8#9
9180 {
9181     \exp_after:wN \__fp_fixed_mul_after:wwn
9182     \int_value:w \__fp_int_eval:w \c__fp_leading_shift_int
9183     \exp_after:wN \__fp_pack:NNNNNw
9184     \int_value:w \__fp_int_eval:w \c__fp_middle_shift_int
9185     + #1*#6
9186     \exp_after:wN \__fp_pack:NNNNNw
9187     \int_value:w \__fp_int_eval:w \c__fp_middle_shift_int
9188     + #1*#7 + #2*#6
9189     \exp_after:wN \__fp_pack:NNNNNw
9190     \int_value:w \__fp_int_eval:w \c__fp_middle_shift_int
9191     + #1*#8 + #2*#7 + #3*#6
9192     \exp_after:wN \__fp_pack:NNNNNw
9193     \int_value:w \__fp_int_eval:w \c__fp_middle_shift_int
9194     + #1*#9 + #2*#8 + #3*#7 + #4*#6
9195     \exp_after:wN \__fp_pack:NNNNNw
9196     \int_value:w \__fp_int_eval:w \c__fp_trailing_shift_int
9197     + #2*#9 + #3*#8 + #4*#7
9198     + ( #3*#9 + #4*#8
9199         + \__fp_fixed_mul:nnnnnnnw #5 {#6}{#7} {#1}{#2}
9200 )
9201 \cs_new:Npn \__fp_fixed_mul:nnnnnnnw #1#2 #3#4 #5#6 #7#8 \__fp_sep:
9202 {
9203     #1*#4 + #2*#3 + #5*#8 + #6*#7 ) / \c__fp_myriad_int
9204     + #1*#3 + #5*#7 \__fp_sep: \__fp_sep:
9205 }
9206 \cs_new:Npn \__fp_fixed_mul_add:wwwn #1\__fp_sep: #2\__fp_sep: #3#4#5#6#7#8\__fp_sep:
9207 {
9208     \exp_after:wN \__fp_fixed_mul_after:wwn
9209     \int_value:w \__fp_int_eval:w \c__fp_big_leading_shift_int
9210     \exp_after:wN \__fp_pack_big:NNNNNNw
9211     \int_value:w \__fp_int_eval:w \c__fp_big_middle_shift_int + #3 #4
9212     \__fp_fixed_mul_add:Nwnnnwnnn +
9213     + #5 #6 \__fp_sep: #2 \__fp_sep: #1 \__fp_sep: #2 \__fp_sep: +
9214     + #7 #8 \__fp_sep: \__fp_sep:

```



```
9215 }
9216 \cs_new:Npn \__fp_fixed_mul_sub_back:wwn
9217 #1\__fp_sep: #2\__fp_sep: #3#4#5#6#7#8\__fp_sep:
9218 {
9219 \exp_after:wN \__fp_fixed_mul_after:wwn
9220 \int_value:w \__fp_int_eval:w \c__fp_big_leading_shift_int
9221 \exp_after:wN \__fp_pack_big:NNNNNNw
9222 \int_value:w \__fp_int_eval:w \c__fp_big_middle_shift_int + #3 #4
9223 \__fp_fixed_mul_add:Nwnnnwnnn -
9224 + #5 #6 \__fp_sep: #2 \__fp_sep: #1 \__fp_sep: #2 \__fp_sep: -
9225 + #7 #8 \__fp_sep: \__fp_sep:
9226 }
9227 \cs_new:Npn \__fp_fixed_one_minus_mul:wwn #1\__fp_sep: #2\__fp_sep:
9228 {
9229 \exp_after:wN \__fp_fixed_mul_after:wwn
9230 \int_value:w \__fp_int_eval:w \c__fp_big_leading_shift_int
9231 \exp_after:wN \__fp_pack_big:NNNNNNw
9232 \int_value:w \__fp_int_eval:w \c__fp_big_middle_shift_int +
9233 1 0000 0000
9234 \__fp_fixed_mul_add:Nwnnnwnnn -
9235 \__fp_sep: #2 \__fp_sep: #1 \__fp_sep: #2 \__fp_sep: -
9236 \__fp_sep: \__fp_sep:
9237 }
9238 \cs_new:Npn \__fp_fixed_mul_add:Nwnnnwnnn #1 #2\__fp_sep: #3#4#5#6\__fp_sep: #7#8#9
9239 {
9240 #1 #7*#3
9241 \exp_after:wN \__fp_pack_big:NNNNNNw
9242 \int_value:w \__fp_int_eval:w \c__fp_big_middle_shift_int
9243 #1 #7*#4 #1 #8*#3
9244 \exp_after:wN \__fp_pack_big:NNNNNNw
9245 \int_value:w \__fp_int_eval:w \c__fp_big_middle_shift_int
9246 #1 #7*#5 #1 #8*#4 #1 #9*#3 #2
9247 \exp_after:wN \__fp_pack_big:NNNNNNw
9248 \int_value:w \__fp_int_eval:w \c__fp_big_middle_shift_int
9249 #1 \__fp_fixed_mul_add:nnnnwnnnn {#7}{#8}{#9}
9250 }
9251 \cs_new:Npn \__fp_fixed_mul_add:nnnnwnnnn #1#2#3#4#5\__fp_sep: #6#7#8#9
9252 {
9253 ( #1*#9 + #2*#8 + #3*#7 + #4*#6 )
9254 \exp_after:wN \__fp_pack_big:NNNNNNw
9255 \int_value:w \__fp_int_eval:w \c__fp_big_trailing_shift_int
9256 \__fp_fixed_mul_add:nnnnwnnnwN
9257 { #6 + #4*#7 + #3*#8 + #2*#9 + #1 }
9258 { #7 + #4*#8 + #3*#9 + #2 }
9259 {#1} #5\__fp_sep:
9260 {#6}
```



```
9261 }
9262 \cs_new:Npn \__fp_fixed_mul_add:nnnnwnnwN #1#2 #3#4#5\__fp_sep: #6#7#8\__fp_sep: #9
9263 {
9264     #9 (#4* #1 *#7)
9265     #9 (#5*#6+#4* #2 *#7+#3*#8) / \c__fp_myriad_int
9266 }
9267 \cs_new:Npn \__fp_ep_to_fixed:wwn #1,#2
9268 {
9269     \exp_after:wN \__fp_ep_to_fixed_auxi:www
9270     \int_value:w \__fp_int_eval:w 1 0000 0000 + #2 \exp_after:wN \__fp_sep:
9271     \exp:w \exp_end_continue_f:w
9272     \prg_replicate:nn { 4 - \int_max:nn {#1} { -32 } } { 0 } \__fp_sep:
9273 }
9274 \cs_new:Npn \__fp_ep_to_fixed_auxi:www
9275     1#1\__fp_sep: #2\__fp_sep: #3#4#5#6#7\__fp_sep:
9276 {
9277     \__fp_pack_eight:wNNNNNNNNN
9278     \__fp_pack_twice_four:wNNNNNNNNN
9279     \__fp_pack_twice_four:wNNNNNNNNN
9280     \__fp_pack_twice_four:wNNNNNNNNN
9281     \__fp_ep_to_fixed_auxii:nnnnnnnwN \__fp_sep:
9282     #2 #1#3#4#5#6#7 0000 !
9283 }
9284 \cs_new:Npn \__fp_ep_to_fixed_auxii:nnnnnnnwN #1#2#3#4#5#6#7\__fp_sep: #8! #9
9285 { #9 {#1#2}{#3}{#4}{#5}{#6}{#7}\__fp_sep: }
9286 \cs_new:Npn \__fp_ep_to_ep:wwN #1,#2#3#4#5#6#7\__fp_sep: #8
9287 {
9288     \exp_after:wN #8
9289     \int_value:w \__fp_int_eval:w #1 + 4
9290     \exp_after:wN \use_i:nn
9291     \exp_after:wN \__fp_ep_to_ep_loop:N
9292     \int_value:w \__fp_int_eval:w 1 0000 0000 + #2 \__fp_int_eval_end:
9293     #3#4#5#6#7 \__fp_sep: \__fp_sep: !
9294 }
9295 \cs_new:Npn \__fp_ep_to_ep_loop:N #1
9296 {
9297     \if_meaning:w 0 #1
9298         - 1
9299     \else:
9300         \__fp_ep_to_ep_end:www #1
9301     \fi:
9302     \__fp_ep_to_ep_loop:N
9303 }
9304 \cs_new:Npn \__fp_ep_to_ep_end:www
9305     #1 \fi: \__fp_ep_to_ep_loop:N #2\__fp_sep: #3!
9306 {
```

```
9307 \fi:
9308 \if_meaning:w \__fp_sep: #1
9309 - 2 * \c__fp_max_exponent_int
9310 \__fp_ep_to_ep_zero:ww
9311 \fi:
9312 \__fp_pack_twice_four:wNNNNNNNN
9313 \__fp_pack_twice_four:wNNNNNNNN
9314 \__fp_pack_twice_four:wNNNNNNNN
9315 \__fp_use_i:ww , \__fp_sep:
9316 #1 #2 0000 0000 0000 0000 0000 0000 \__fp_sep:
9317 }
9318 \cs_new:Npn \__fp_ep_to_ep_zero:ww \fi: #1\__fp_sep: #2\__fp_sep: #3\__fp_sep:
9319 { \fi: , {1000}{0000}{0000}{0000}{0000}{0000} \__fp_sep: }
9320 \cs_new:Npn \__fp_ep_compare:www #1,#2#3#4#5#6#7\__fp_sep:
9321 { \__fp_ep_compare_aux:www {#1}{#2}{#3}{#4}{#5}\__fp_sep: #6#7\__fp_sep: }
9322 \cs_new:Npn \__fp_ep_compare_aux:www
9323 #1\__fp_sep: #2\__fp_sep: #3,#4#5#6#7#8#9\__fp_sep:
9324 {
9325 \if_case:w
9326 \__fp_compare_npos:nwnw
9327 #1\__fp_sep: {#3}{#4}{#5}{#6}{#7}\__fp_sep: \exp_stop_f:
9328 \if_int_compare:w #2 = #8#9 \exp_stop_f:
9329 0
9330 \else:
9331 \if_int_compare:w #2 < #8#9 - \fi: 1
9332 \fi:
9333 \or: 1
9334 \else: -1
9335 \fi:
9336 }
9337 \cs_new:Npn \__fp_ep_mul:wwwn #1,#2\__fp_sep: #3,#4\__fp_sep:
9338 {
9339 \__fp_ep_to_ep:wwN #3,#4\__fp_sep:
9340 \__fp_fixed_continue:wn
9341 {
9342 \__fp_ep_to_ep:wwN #1,#2\__fp_sep:
9343 \__fp_ep_mul_raw:wwwN
9344 }
9345 \__fp_fixed_continue:wn
9346 }
9347 \cs_new:Npn \__fp_ep_mul_raw:wwwN #1,#2\__fp_sep: #3,#4\__fp_sep: #5
9348 {
9349 \__fp_fixed_mul:wn #2\__fp_sep: #4\__fp_sep:
9350 { \exp_after:wN #5 \int_value:w \__fp_int_eval:w #1 + #3 , }
9351 }
9352 \cs_new:Npn \__fp_ep_div:wwwn #1,#2\__fp_sep: #3,#4\__fp_sep:
```

```
9353 {
9354     \__fp_ep_to_ep:wwN #1,#2\__fp_sep:
9355     \__fp_fixed_continue:wn
9356 {
9357     \__fp_ep_to_ep:wwN #3,#4\__fp_sep:
9358     \__fp_ep_div_esti:wwwN
9359 }
9360 }
9361 \cs_new:Npn \__fp_ep_div_esti:wwwN #1,#2#3\__fp_sep: #4,
9362 {
9363     \exp_after:wN \__fp_ep_div_estii:wwnnwwN
9364     \int_value:w \__fp_int_eval:w 10 0000 0000 / ( #2 + 1 )
9365     \exp_after:wN \__fp_sep:
9366     \int_value:w \__fp_int_eval:w #4 - #1 + 1 ,
9367     {#2} #3\__fp_sep:
9368 }
9369 \cs_new:Npn \__fp_ep_div_estii:wwnnwwN
9370     #1\__fp_sep: #2,#3#4#5\__fp_sep: #6\__fp_sep: #7
9371 {
9372     \exp_after:wN \__fp_ep_div_estiii:NNNNNwwN
9373     \int_value:w \__fp_int_eval:w 10 0000 0000 - 1750
9374     + #1 000 + (10 0000 0000 / #3 - #1) * (1000 - #4 / 10) \__fp_sep:
9375     {#3}{#4}#5\__fp_sep: #6\__fp_sep: { #7 #2, }
9376 }
9377 \cs_new:Npn \__fp_ep_div_estiii:NNNNNwwN 1#1#2#3#4#5#6\__fp_sep: #7\__fp_sep:
9378 {
9379     \__fp_fixed_mul_short:wn #7\__fp_sep: {#1}{#2#3#4#5}{#6}\__fp_sep:
9380     \__fp_ep_div_epsilon:wnNNNNNn {#1#2#3#4}#5#6
9381     \__fp_fixed_mul:wn
9382 }
9383 \cs_new:Npn \__fp_ep_div_epsilon:wnNNNNNn #1#2#3#4#5#6\__fp_sep:
9384 {
9385     \exp_after:wN \__fp_ep_div_epsiloni:wnNNNNNn
9386     \int_value:w \__fp_int_eval:w 1 9998 - #2
9387     \exp_after:wN \__fp_ep_div_epsilon_pack:NNNNNw
9388     \int_value:w \__fp_int_eval:w 1 9999 9998 - #3#4
9389     \exp_after:wN \__fp_ep_div_epsilon_pack:NNNNNw
9390     \int_value:w \__fp_int_eval:w 2 0000 0000 - #5#6 \__fp_sep: \__fp_sep:
9391 }
9392 \cs_new:Npn \__fp_ep_div_epsilon_pack:NNNNNw #1#2#3#4#5#6\__fp_sep:
9393 { + #1 \__fp_sep: {#2#3#4#5} {#6} }
9394 \cs_new:Npn \__fp_ep_div_epsiloni:wnNNNNNn 1#1\__fp_sep: #2\__fp_sep: #3#4#5#6#7#8
9395 {
9396     \__fp_fixed_mul:wn {0000}{#1}#2\__fp_sep: {0000}{#1}#2\__fp_sep:
9397     \__fp_fixed_add_one:wN
9398     \__fp_fixed_mul:wn {10000} {#1} #2 \__fp_sep:
```

```
9399 {
9400     \__fp_fixed_mul_short:wwn
9401         {0000}{#1}#2\__fp_sep: {#3}{#4#5#6#7}{#8000}\__fp_sep:
9402     \__fp_fixed_div_myriad:wn
9403     \__fp_fixed_mul:wwn
9404 }
9405 \__fp_fixed_add:wwn {#3}{#4#5#6#7}{#8000}{0000}{0000}{0000}\__fp_sep:
9406 }
9407 \cs_new:Npn \__fp_ep_isqrt:wwn #1,#2\__fp_sep:
9408 {
9409     \__fp_ep_to_ep:wwN #1,#2\__fp_sep:
9410     \__fp_ep_isqrt_auxi:wwn
9411 }
9412 \cs_new:Npn \__fp_ep_isqrt_auxi:wwn #1,
9413 {
9414     \exp_after:wN \__fp_ep_isqrt_auxii:wwnnwn
9415     \int_value:w \__fp_int_eval:w
9416     \int_if_odd:nTF {#1}
9417         { (1 - #1) / 2 , 535 , { 0 } { } }
9418         { 1 - #1 / 2 , 168 , { } { 0 } }
9419 }
9420 \cs_new:Npn \__fp_ep_isqrt_auxii:wwnnwn #1, #2, #3#4 #5#6\__fp_sep: #7
9421 {
9422     \__fp_ep_isqrt_esti:wwnnwn #2, 0, #5, {#3} {#4}
9423     {#5} #6 \__fp_sep: { #7 #1 , }
9424 }
9425 \cs_new:Npn \__fp_ep_isqrt_esti:wwnnwn #1, #2, #3, #4
9426 {
9427     \if_int_compare:w #1 = #2 \exp_stop_f:
9428     \exp_after:wN \__fp_ep_isqrt_estii:wwnnwn
9429     \fi:
9430     \exp_after:wN \__fp_ep_isqrt_esti:wwnnwn
9431     \int_value:w \__fp_int_eval:w
9432     (#1 + 1 0050 0000 #4 / (#1 * #3)) / 2 ,
9433     #1, #3, {#4}
9434 }
9435 \cs_new:Npn \__fp_ep_isqrt_estii:wwnnwn #1, #2, #3, #4#5
9436 {
9437     \exp_after:wN \__fp_ep_isqrt_estiii:NNNNNwwwn
9438     \int_value:w \__fp_int_eval:w 1000 0000 + #2 * #2 #5 * 5
9439     \exp_after:wN , \int_value:w \__fp_int_eval:w 10000 + #2 \__fp_sep:
9440 }
9441 \cs_new:Npn \__fp_ep_isqrt_estiii:NNNNNwwwn
9442     1#1#2#3#4#5#6, 1#7#8\__fp_sep: #9\__fp_sep:
9443 {
9444     \__fp_fixed_mul_short:wwn #9\__fp_sep: {#1} {#2#3#4#5} {#600} \__fp_sep:
```

```
9445 \__fp_ep_isqrt_epsilon:wN
9446 \__fp_fixed_mul_short:wwn {#7} {#80} {0000} \__fp_sep:
9447 }
9448 \cs_new:Npn \__fp_ep_isqrt_epsilon:wN #1\__fp_sep:
9449 {
9450 \__fp_fixed_sub:wwn {15000}{0000}{0000}{0000}{0000}{0000}\__fp_sep: #1\__fp_sep:
9451 \__fp_ep_isqrt_epsilon:wwN #1\__fp_sep:
9452 \__fp_ep_isqrt_epsilon:wwN #1\__fp_sep:
9453 \__fp_ep_isqrt_epsilon:wwN #1\__fp_sep:
9454 }
9455 \cs_new:Npn \__fp_ep_isqrt_epsilon:wwN #1\__fp_sep: #2\__fp_sep:
9456 {
9457 \__fp_fixed_mul:wwn #1\__fp_sep: #1\__fp_sep:
9458 \__fp_fixed_mul_sub_back:wwn #2\__fp_sep:
9459 {15000}{0000}{0000}{0000}{0000}{0000}\__fp_sep:
9460 \__fp_fixed_mul:wwn #1\__fp_sep:
9461 }
9462 \cs_new:Npn \__fp_ep_to_float_o:wwN #1,
9463 { + \__fp_int_eval:w #1 \__fp_fixed_to_float_o:wN }
9464 \cs_new:Npn \__fp_ep_inv_to_float_o:wwN #1,#2\__fp_sep:
9465 {
9466 \__fp_ep_div:wwwwn
9467 1,{1000}{0000}{0000}{0000}{0000}{0000}\__fp_sep: #1,#2\__fp_sep:
9468 \__fp_ep_to_float_o:wwN
9469 }
9470 \cs_new:Npn \__fp_fixed_inv_to_float_o:wN
9471 { \__fp_ep_inv_to_float_o:wwN 0, }
9472 \cs_new:Npn \__fp_fixed_to_float_rad_o:wN #1\__fp_sep:
9473 {
9474 \__fp_fixed_mul:wwn #1\__fp_sep: {5729}{5779}{5130}{8232}{0876}{7981}\__fp_sep:
9475 { \__fp_ep_to_float_o:wwN 2, }
9476 }
9477 \cs_new:Npn \__fp_fixed_to_float_o:Nw #1#2\__fp_sep:
9478 { \__fp_fixed_to_float_o:wN #2\__fp_sep: #1 }
9479 \cs_new:Npn \__fp_fixed_to_float_o:wN #1#2#3#4#5#6\__fp_sep: #7
9480 { % for the 8-digit-at-the-start thing
9481 + \__fp_int_eval:w \c__fp_block_int
9482 \exp_after:wN \exp_after:wN
9483 \exp_after:wN \__fp_fixed_to_loop:N
9484 \exp_after:wN \use_none:n
9485 \int_value:w \__fp_int_eval:w
9486 1 0000 0000 + #1 \exp_after:wN \__fp_use_none_stop_f:n
9487 \int_value:w 1#2 \exp_after:wN \__fp_use_none_stop_f:n
9488 \int_value:w 1#3#4 \exp_after:wN \__fp_use_none_stop_f:n
9489 \int_value:w 1#5#6
9490 \exp_after:wN \__fp_sep:
```

```
9491 \exp_after:wN \__fp_sep:
9492 }
9493 \cs_new:Npn \__fp_fixed_to_loop:N #1
9494 {
9495   \if_meaning:w 0 #1
9496     - 1
9497     \exp_after:wN \__fp_fixed_to_loop:N
9498   \else:
9499     \exp_after:wN \__fp_fixed_to_loop_end:w
9500     \exp_after:wN #1
9501   \fi:
9502 }
9503 \cs_new:Npn \__fp_fixed_to_loop_end:w #1 #2 \__fp_sep:
9504 {
9505   \if_meaning:w \__fp_sep: #1
9506     \exp_after:wN \__fp_fixed_to_float_zero:w
9507   \else:
9508     \exp_after:wN \__fp_pack_twice_four:wNNNNNNNN
9509     \exp_after:wN \__fp_pack_twice_four:wNNNNNNNN
9510     \exp_after:wN \__fp_fixed_to_float_pack:ww
9511     \exp_after:wN \__fp_sep:
9512   \fi:
9513   #1 #2 0000 0000 0000 0000 \__fp_sep:
9514 }
9515 \cs_new:Npn \__fp_fixed_to_float_zero:w \__fp_sep: 0000 0000 0000 0000 \__fp_sep:
9516 {
9517   - 2 * \c__fp_max_exponent_int \__fp_sep:
9518   {0000} {0000} {0000} {0000} \__fp_sep:
9519 }
9520 \cs_new:Npn \__fp_fixed_to_float_pack:ww #1 \__fp_sep: #2#3 \__fp_sep: \__fp_sep:
9521 {
9522   \if_int_compare:w #2 > 4 \exp_stop_f:
9523     \exp_after:wN \__fp_fixed_to_float_round_up:wnnnnw
9524   \fi:
9525   \__fp_sep: #1 \__fp_sep:
9526 }
9527 \cs_new:Npn \__fp_fixed_to_float_round_up:wnnnnw \__fp_sep: #1#2#3#4 \__fp_sep:
9528 {
9529   \exp_after:wN \__fp_basics_pack_high:NNNNNw
9530   \int_value:w \__fp_int_eval:w 1 #1#2
9531   \exp_after:wN \__fp_basics_pack_low:NNNNNw
9532   \int_value:w \__fp_int_eval:w 1 #3#4 + 1 \__fp_sep:
9533 }
9534 %% File: l3fp-expo.dtx
9535 \cs_new:Npn \__fp_parse_word_exp:N
9536 { \__fp_parse_unary_function:NNN \__fp_exp_o:w ? }
```

```
9537 \cs_new:Npn \__fp_parse_word_ln:N 1953
9538 { \__fp_parse_unary_function:NNN \__fp_ln_o:w ? } 1953
9539 \cs_new:Npn \__fp_parse_word_fact:N 1953
9540 { \__fp_parse_unary_function:NNN \__fp_fact_o:w ? } 1954
9541 \tl_const:Nn \c__fp_ln_i_fixed_tl 1954
9542 { {0000}{0000}{0000}{0000}{0000}{0000}\__fp_sep:} 1954
9543 \tl_const:Nn \c__fp_ln_ii_fixed_tl 1954
9544 { {6931}{4718}{0559}{9453}{0941}{7232}\__fp_sep:} 1954
9545 \tl_const:Nn \c__fp_ln_iii_fixed_tl 1954
9546 { {10986}{1228}{8668}{1096}{9139}{5245}\__fp_sep:} 1954
9547 \tl_const:Nn \c__fp_ln_iv_fixed_tl 1954
9548 { {13862}{9436}{1119}{8906}{1883}{4464}\__fp_sep:} 1954
9549 \tl_const:Nn \c__fp_ln_vi_fixed_tl 1954
9550 { {17917}{5946}{9228}{0550}{0081}{2477}\__fp_sep:} 1955
9551 \tl_const:Nn \c__fp_ln_vii_fixed_tl 1955
9552 { {19459}{1014}{9055}{3133}{0510}{5353}\__fp_sep:} 1955
9553 \tl_const:Nn \c__fp_ln_viii_fixed_tl 1955
9554 { {20794}{4154}{1679}{8359}{2825}{1696}\__fp_sep:} 1955
9555 \tl_const:Nn \c__fp_ln_ix_fixed_tl 1955
9556 { {21972}{2457}{7336}{2193}{8279}{0490}\__fp_sep:} 1955
9557 \tl_const:Nn \c__fp_ln_x_fixed_tl 1955
9558 { {23025}{8509}{2994}{0456}{8401}{7991}\__fp_sep:} 1955
9559 \cs_new:Npn \__fp_ln_o:w #1 \s__fp \__fp_chk:w #2#3#4\__fp_sep: @ 1955
9560 { 1956
9561 \if_meaning:w 2 #3 1956
9562 \__fp_case_use:nw { \__fp_invalid_operation_o:nw { ln } } 1956
9563 \fi: 1956
9564 \if_case:w #2 \exp_stop_f: 1956
9565 \__fp_case_use:nw 1956
9566 { \__fp_division_by_zero_o:Nnw \c_minus_inf_fp { ln } } 1956
9567 \or: 1956
9568 \else: 1956
9569 \__fp_case_return_same_o:w 1956
9570 \fi: 1957
9571 \__fp_ln_npos_o:w \s__fp \__fp_chk:w #2#3#4\__fp_sep: 1957
9572 } 1957
9573 \cs_new:Npn \__fp_ln_npos_o:w \s__fp \__fp_chk:w 10#1#2#3\__fp_sep: 1957
9574 { %^^A todo: ln(1) should be "exact zero", not "underflow" 1957
9575 \exp_after:wN \__fp_sanitize:Nw 1957
9576 \int_value:w % for the overall sign 1957
9577 \if_int_compare:w #1 < \c_one_int 1957
9578 2 1957
9579 \else: 1957
9580 0 1958
9581 \fi: 1958
9582 \exp_after:wN \exp_stop_f: 1958
```



```
9583 \int_value:w \__fp_int_eval:w % for the exponent
9584 \__fp_ln_significand:NNNNnnnnN #2#3
9585 \__fp_ln_exponent:wn {#1}
9586 }
9587 \cs_new:Npn \__fp_ln_significand:NNNNnnnnN #1#2#3#4
9588 {
9589 \exp_after:wN \__fp_ln_x_ii:wnnnn
9590 \int_value:w
9591 \if_case:w #1 \exp_stop_f:
9592 \or:
9593 \if_int_compare:w #2 < 4 \exp_stop_f:
9594 \__fp_int_eval:w 10 - #2
9595 \else:
9596 6
9597 \fi:
9598 \or: 4
9599 \or: 3
9600 \or: 2
9601 \or: 2
9602 \or: 2
9603 \else: 1
9604 \fi:
9605 \__fp_sep: { #1 #2 #3 #4 }
9606 }
9607 \cs_new:Npn \__fp_ln_x_ii:wnnnn #1\__fp_sep: #2#3#4#5
9608 {
9609 \exp_after:wN \__fp_ln_div_after:Nw
9610 \cs:w c__fp_ln_ \__fp_int_to_roman:w #1 _fixed_tl \exp_after:wN \cs_end:
9611 \int_value:w
9612 \exp_after:wN \__fp_ln_x_iv:wnnnnnnnn
9613 \int_value:w \__fp_int_eval:w
9614 \exp_after:wN \__fp_ln_x_iii_var:NNNNNw
9615 \int_value:w \__fp_int_eval:w 9999 9990 + #1*#2#3 +
9616 \exp_after:wN \__fp_ln_x_iii:NNNNNNw
9617 \int_value:w \__fp_int_eval:w 10 0000 0000 + #1*#4#5 \__fp_sep:
9618 {20000} {0000} {0000} {0000}
9619 } %^^A todo: reoptimize (a generalization attempt failed).
9620 \cs_new:Npn \__fp_ln_x_iii:NNNNNNw #1#2 #3#4#5#6 #7\__fp_sep:
9621 { #1#2\__fp_sep: {#3#4#5#6} {#7} }
9622 \cs_new:Npn \__fp_ln_x_iii_var:NNNNNw #1 #2#3#4#5 #6\__fp_sep:
9623 {
9624 #1#2#3#4#5 + 1 \__fp_sep:
9625 {#1#2#3#4#5} {#6}
9626 }
9627 \cs_new:Npn \__fp_ln_x_iv:wnnnnnnnn #1\__fp_sep: #2#3#4#5 #6#7#8#9
9628 {
```

```
9629 \exp_after:wN \__fp_div_significand_pack:NNN 19629
9630 \int_value:w \__fp_int_eval:w 19630
9631 \__fp_ln_div_i:w #1 \__fp_sep: 19631
9632 #6 #7 \__fp_sep: {#8} {#9} 19632
9633 {#2} {#3} {#4} {#5} 19633
9634 { \exp_after:wN \__fp_ln_div_ii:wwn \int_value:w #1 } 19634
9635 { \exp_after:wN \__fp_ln_div_ii:wwn \int_value:w #1 } 19635
9636 { \exp_after:wN \__fp_ln_div_ii:wwn \int_value:w #1 } 19636
9637 { \exp_after:wN \__fp_ln_div_ii:wwn \int_value:w #1 } 19637
9638 { \exp_after:wN \__fp_ln_div_vi:wwn \int_value:w #1 } 19638
9639 } 19639
9640 \cs_new:Npn \__fp_ln_div_i:w #1\__fp_sep: 19640
9641 { 19641
9642 \exp_after:wN \__fp_div_significand_calc:wwnnnnnnnn 19642
9643 \int_value:w \__fp_int_eval:w 999999 + 2 0000 0000 / #1 \__fp_sep: % Q1 19643
9644 } 19644
9645 \cs_new:Npn \__fp_ln_div_ii:wwn 19645
9646 #1\__fp_sep: #2\__fp_sep:#3 % y\__fp_sep: B1\__fp_sep:B2 <- for k=1 19646
9647 { 19647
9648 \exp_after:wN \__fp_div_significand_pack:NNN 19648
9649 \int_value:w \__fp_int_eval:w 19649
9650 \exp_after:wN \__fp_div_significand_calc:wwnnnnnnnn 19650
9651 \int_value:w \__fp_int_eval:w 999999 + #2 #3 / #1 \__fp_sep: % Q2 19651
9652 #2 #3 \__fp_sep: 19652
9653 } 19653
9654 \cs_new:Npn \__fp_ln_div_vi:wwn 19654
9655 #1\__fp_sep: #2\__fp_sep:#3#4#5 #6#7#8#9 %y\__fp_sep:F1\__fp_sep:F2F3F4x1x2x3x4 19655
9656 { 19656
9657 \exp_after:wN \__fp_div_significand_pack:NNN 19657
9658 \int_value:w \__fp_int_eval:w 1000000 + #2 #3 / #1 \__fp_sep: % Q6 19658
9659 } 19659
9660 \cs_new:Npn \__fp_ln_div_after:Nw #1#2\__fp_sep: 19660
9661 { 19661
9662 \if_meaning:w 0 #2 19662
9663 \exp_after:wN \__fp_ln_t_small:Nw 19663
9664 \else: 19664
9665 \exp_after:wN \__fp_ln_t_large:NNw 19665
9666 \exp_after:wN - 19666
9667 \fi: 19667
9668 #1 19668
9669 } 19669
9670 \cs_new:Npn \__fp_ln_t_small:Nw 19670
9671 #1 #2\__fp_sep: #3\__fp_sep: #4\__fp_sep: #5\__fp_sep: #6\__fp_sep: #7\__fp_sep: 19671
9672 { 19672
9673 \exp_after:wN \__fp_ln_t_large:NNw 19673
9674 \exp_after:wN + % <sign> 19674
```

```
9675 \exp_after:wN #1 19675
9676 \int_value:w \__fp_int_eval:w 9999 - #2 \exp_after:wN \__fp_sep: 19676
9677 \int_value:w \__fp_int_eval:w 9999 - #3 \exp_after:wN \__fp_sep: 19677
9678 \int_value:w \__fp_int_eval:w 9999 - #4 \exp_after:wN \__fp_sep: 19678
9679 \int_value:w \__fp_int_eval:w 9999 - #5 \exp_after:wN \__fp_sep: 19679
9680 \int_value:w \__fp_int_eval:w 9999 - #6 \exp_after:wN \__fp_sep: 19680
9681 \int_value:w \__fp_int_eval:w 1 0000 - #7 \__fp_sep: 19681
9682 } 19682
9683 \cs_new:Npn \__fp_ln_t_large:NNw 19683
9684 #1 #2 19684
9685 #3\__fp_sep: #4\__fp_sep: #5\__fp_sep: #6\__fp_sep: #7\__fp_sep: #8\__fp_sep: 19685
9686 { 19686
9687 \exp_after:wN \__fp_ln_square_t_after:w 19687
9688 \int_value:w \__fp_int_eval:w 9999 0000 + #3*#3 19688
9689 \exp_after:wN \__fp_ln_square_t_pack:NNNNNw 19689
9690 \int_value:w \__fp_int_eval:w 9999 0000 + 2*#3*#4 19690
9691 \exp_after:wN \__fp_ln_square_t_pack:NNNNNw 19691
9692 \int_value:w \__fp_int_eval:w 9999 0000 + 2*#3*#5 + #4*#4 19692
9693 \exp_after:wN \__fp_ln_square_t_pack:NNNNNw 19693
9694 \int_value:w \__fp_int_eval:w 9999 0000 + 2*#3*#6 + 2*#4*#5 19694
9695 \exp_after:wN \__fp_ln_square_t_pack:NNNNNw 19695
9696 \int_value:w \__fp_int_eval:w 19696
9697 1 0000 0000 + 2*#3*#7 + 2*#4*#6 + #5*#5 19697
9698 + (2*#3*#8 + 2*#4*#7 + 2*#5*#6) / 1 0000 19698
9699 % \__fp_sep: \__fp_sep: \__fp_sep: 19699
9700 \exp_after:wN \__fp_ln_twice_t_after:w 19700
9701 \int_value:w \__fp_int_eval:w -1 + 2*#3 19701
9702 \exp_after:wN \__fp_ln_twice_t_pack:Nw 19702
9703 \int_value:w \__fp_int_eval:w 9999 + 2*#4 19703
9704 \exp_after:wN \__fp_ln_twice_t_pack:Nw 19704
9705 \int_value:w \__fp_int_eval:w 9999 + 2*#5 19705
9706 \exp_after:wN \__fp_ln_twice_t_pack:Nw 19706
9707 \int_value:w \__fp_int_eval:w 9999 + 2*#6 19707
9708 \exp_after:wN \__fp_ln_twice_t_pack:Nw 19708
9709 \int_value:w \__fp_int_eval:w 9999 + 2*#7 19709
9710 \exp_after:wN \__fp_ln_twice_t_pack:Nw 19710
9711 \int_value:w \__fp_int_eval:w 10000 + 2*#8 \__fp_sep: \__fp_sep: 19711
9712 { \__fp_ln_c:NwNw #1 } 19712
9713 #2 19713
9714 } 19714
9715 \cs_new:Npn \__fp_ln_twice_t_pack:Nw #1 #2\__fp_sep: { + #1 \__fp_sep: {#2} } 19715
9716 \cs_new:Npn \__fp_ln_twice_t_after:w #1\__fp_sep: 19716
9717 { \__fp_sep:\__fp_sep:\__fp_sep: {#1} } 19717
9718 \cs_new:Npn \__fp_ln_square_t_pack:NNNNNw #1 #2#3#4#5 #6\__fp_sep: 19718
9719 { + #1#2#3#4#5 \__fp_sep: {#6} } 19719
9720 \cs_new:Npn \__fp_ln_square_t_after:w 1 0 #1#2#3 #4\__fp_sep: 19720
```

```
9721 { \__fp_ln_Taylor:wwNw {0#1#2#3} {#4} }
9722 \cs_new:Npn \__fp_ln_Taylor:wwNw
9723 {
9724   \__fp_ln_Taylor_loop:www
9725   21 \__fp_sep: {0000}{0000}{0000}{0000}{0000}{0000} \__fp_sep:
9726 }
9727 \cs_new:Npn \__fp_ln_Taylor_loop:www #1\__fp_sep: #2\__fp_sep: #3\__fp_sep:
9728 {
9729   \if_int_compare:w #1 = \c_one_int
9730     \__fp_ln_Taylor_break:w
9731   \fi:
9732   \exp_after:wN \__fp_fixed_div_int:wwN \c__fp_one_fixed_tl #1\__fp_sep:
9733   \__fp_fixed_add:wwn #2\__fp_sep:
9734   \__fp_fixed_mul:wwn #3\__fp_sep:
9735   {
9736     \exp_after:wN \__fp_ln_Taylor_loop:www
9737     \int_value:w \__fp_int_eval:w #1 - 2 \__fp_sep:
9738   }
9739   #3\__fp_sep:
9740 }
9741 \cs_new:Npn \__fp_ln_Taylor_break:w
9742   \fi: #1 \__fp_fixed_add:wwn #2#3\__fp_sep: #4 \__fp_sep:\__fp_sep:
9743 {
9744   \fi:
9745   \exp_after:wN \__fp_fixed_mul:wwn
9746   \exp_after:wN { \int_value:w \__fp_int_eval:w 10000 + #2 } #3\__fp_sep:
9747 }
9748 \cs_new:Npn \__fp_ln_c:NwNw #1 #2\__fp_sep: #3
9749 {
9750   \if_meaning:w + #1
9751     \exp_after:wN \exp_after:wN \exp_after:wN \__fp_fixed_sub:wwn
9752   \else:
9753     \exp_after:wN \exp_after:wN \exp_after:wN \__fp_fixed_add:wwn
9754   \fi:
9755   #3 #2 \__fp_sep:
9756 }
9757 \cs_new:Npn \__fp_ln_exponent:wn #1\__fp_sep: #2
9758 {
9759   \if_case:w #2 \exp_stop_f:
9760     0 \__fp_case_return:nw { \__fp_fixed_to_float_o:Nw 2 }
9761   \or:
9762     \exp_after:wN \__fp_ln_exponent_one:ww \int_value:w
9763   \else:
9764     \if_int_compare:w #2 > \c_zero_int
9765       \exp_after:wN \__fp_ln_exponent_small:NNww
9766     \exp_after:wN 0
```

```
9767 \exp_after:wN \__fp_fixed_sub:wwn \int_value:w 19767
9768 \else: 19768
9769 \exp_after:wN \__fp_ln_exponent_small:NNww 19769
9770 \exp_after:wN 2 19770
9771 \exp_after:wN \__fp_fixed_add:wwn \int_value:w - 19771
9772 \fi: 19772
9773 \fi: 19773
9774 #2\__fp_sep: #1\__fp_sep: 19774
9775 } 19775
9776 \cs_new:Npn \__fp_ln_exponent_one:ww 1\__fp_sep: #1\__fp_sep: 19776
9777 { 19777
9778 0 19778
9779 \exp_after:wN \__fp_fixed_sub:wwn \c__fp_ln_x_fixed_tl #1\__fp_sep: 19779
9780 \__fp_fixed_to_float_o:wN 0 19780
9781 } 19781
9782 \cs_new:Npn \__fp_ln_exponent_small:NNww #1#2#3\__fp_sep: #4#5#6#7#8#9\__fp_sep: 19782
9783 { 19783
9784 4 19784
9785 \exp_after:wN \__fp_fixed_mul:wwn 19785
9786 \c__fp_ln_x_fixed_tl 19786
9787 {#3}{0000}{0000}{0000}{0000}{0000} \__fp_sep: 19787
9788 #2 19788
9789 {0000}{#4}{#5}{#6}{#7}{#8}\__fp_sep: 19789
9790 \__fp_fixed_to_float_o:wN #1 19790
9791 } 19791
9792 \cs_new:Npn \__fp_exp_o:w #1 \s__fp \__fp_chk:w #2#3#4\__fp_sep: @ 19792
9793 { 19793
9794 \if_case:w #2 \exp_stop_f: 19794
9795 \__fp_case_return_o:Nw \c_one_fp 19795
9796 \or: 19796
9797 \exp_after:wN \__fp_exp_normal_o:w 19797
9798 \or: 19798
9799 \if_meaning:w 0 #3 19799
9800 \exp_after:wN \__fp_case_return_o:Nw 19800
9801 \exp_after:wN \c_inf_fp 19801
9802 \else: 19802
9803 \exp_after:wN \__fp_case_return_o:Nw 19803
9804 \exp_after:wN \c_zero_fp 19804
9805 \fi: 19805
9806 \or: 19806
9807 \__fp_case_return_same_o:w 19807
9808 \fi: 19808
9809 \s__fp \__fp_chk:w #2#3#4\__fp_sep: 19809
9810 } 19810
9811 \cs_new:Npn \__fp_exp_normal_o:w \s__fp \__fp_chk:w 1#1 19811
9812 { 19812
```

```
9813 \if_meaning:w 0 #1
9814 \__fp_exp_pos_o:NNwnw + \__fp_fixed_to_float_o:wN
9815 \else:
9816 \__fp_exp_pos_o:NNwnw - \__fp_fixed_inv_to_float_o:wN
9817 \fi:
9818 }
9819 \cs_new:Npn \__fp_exp_pos_o:NNwnw #1#2#3 \fi: #4#5\__fp_sep:
9820 {
9821 \fi:
9822 \if_int_compare:w #4 > \c__fp_max_exp_exponent_int
9823 \token_if_eq_charcode:NNTF + #1
9824 { \__fp_exp_overflow:NN \__fp_overflow:w \c_inf_fp }
9825 { \__fp_exp_overflow:NN \__fp_underflow:w \c_zero_fp }
9826 \exp:w
9827 \else:
9828 \exp_after:wN \__fp_sanitize:Nw
9829 \exp_after:wN 0
9830 \int_value:w #1 \__fp_int_eval:w
9831 \if_int_compare:w #4 < \c_zero_int
9832 \exp_after:wN \use_i:nn
9833 \else:
9834 \exp_after:wN \use_ii:nn
9835 \fi:
9836 {
9837 0
9838 \__fp_decimate:nNnnnn { - #4 }
9839 \__fp_exp_Taylor:Nnnwn
9840 }
9841 {
9842 \__fp_decimate:nNnnnn { \c__fp_prec_int - #4 }
9843 \__fp_exp_pos_large:NnnNwn
9844 }
9845 #5
9846 {#4}
9847 #1 #2 0
9848 \exp:w
9849 \fi:
9850 \exp_after:wN \exp_end:
9851 }
9852 \cs_new:Npn \__fp_exp_overflow:NN #1#2
9853 {
9854 \exp_after:wN \exp_after:wN
9855 \exp_after:wN #1
9856 \exp_after:wN #2
9857 }
9858 \cs_new:Npn \__fp_exp_Taylor:Nnnwn #1#2#3 #4\__fp_sep: #5 #6
```

```
9859 {
9860 #6
9861 \__fp_pack_twice_four:wNNNNNNNN
9862 \__fp_pack_twice_four:wNNNNNNNN
9863 \__fp_pack_twice_four:wNNNNNNNN
9864 \__fp_exp_Taylor_ii:ww
9865 \__fp_sep: #2#3#4 0000 0000 \__fp_sep:
9866 }
9867 \cs_new:Npn \__fp_exp_Taylor_ii:ww #1\__fp_sep: #2\__fp_sep:
9868 { \__fp_exp_Taylor_loop:www 10 \__fp_sep: #1 \__fp_sep: #1 \__fp_sep: \s__fp_stop }
9869 \cs_new:Npn \__fp_exp_Taylor_loop:www #1\__fp_sep: #2\__fp_sep: #3\__fp_sep:
9870 {
9871 \if_int_compare:w #1 = \c_one_int
9872 \exp_after:wN \__fp_exp_Taylor_break:Nww
9873 \fi:
9874 \__fp_fixed_div_int:wwN #3 \__fp_sep: #1 \__fp_sep:
9875 \__fp_fixed_add_one:wN
9876 \__fp_fixed_mul:wwN #2 \__fp_sep:
9877 {
9878 \exp_after:wN \__fp_exp_Taylor_loop:www
9879 \int_value:w \__fp_int_eval:w #1 - 1 \__fp_sep:
9880 #2 \__fp_sep:
9881 }
9882 }
9883 \cs_new:Npn \__fp_exp_Taylor_break:Nww #1 #2\__fp_sep: #3 \s__fp_stop
9884 { \__fp_fixed_add_one:wN #2 \__fp_sep: }
9885 \intarray_const_from_clist:Nn \c__fp_exp_intarray
9886 {
9887 1 , 1 1105 1709 , 1 1807 5647 , 1 6248 1171 ,
9888 1 , 1 1221 4027 , 1 5816 0169 , 1 8339 2107 ,
9889 1 , 1 1349 8588 , 1 0757 6003 , 1 1039 8374 ,
9890 1 , 1 1491 8246 , 1 9764 1270 , 1 3178 2485 ,
9891 1 , 1 1648 7212 , 1 7070 0128 , 1 1468 4865 ,
9892 1 , 1 1822 1188 , 1 0039 0508 , 1 9748 7537 ,
9893 1 , 1 2013 7527 , 1 0747 0476 , 1 5216 2455 ,
9894 1 , 1 2225 5409 , 1 2849 2467 , 1 6045 7954 ,
9895 1 , 1 2459 6031 , 1 1115 6949 , 1 6638 0013 ,
9896 1 , 1 2718 2818 , 1 2845 9045 , 1 2353 6029 ,
9897 1 , 1 7389 0560 , 1 9893 0650 , 1 2272 3043 ,
9898 2 , 1 2008 5536 , 1 9231 8766 , 1 7740 9285 ,
9899 2 , 1 5459 8150 , 1 0331 4423 , 1 9078 1103 ,
9900 3 , 1 1484 1315 , 1 9102 5766 , 1 0342 1116 ,
9901 3 , 1 4034 2879 , 1 3492 7351 , 1 2260 8387 ,
9902 4 , 1 1096 6331 , 1 5842 8458 , 1 5992 6372 ,
9903 4 , 1 2980 9579 , 1 8704 1728 , 1 2747 4359 ,
9904 4 , 1 8103 0839 , 1 2757 5384 , 1 0077 1000 ,
```


9905 5 , 1 2202 6465 , 1 7948 0671 , 1 6516 9579 , 19905
9906 9 , 1 4851 6519 , 1 5409 7902 , 1 7796 9107 , 19906
9907 14 , 1 1068 6474 , 1 5815 2446 , 1 2146 9905 , 19907
9908 18 , 1 2353 8526 , 1 6837 0199 , 1 8540 7900 , 19908
9909 22 , 1 5184 7055 , 1 2858 7072 , 1 4640 8745 , 19909
9910 27 , 1 1142 0073 , 1 8981 5684 , 1 2836 6296 , 19910
9911 31 , 1 2515 4386 , 1 7091 9167 , 1 0062 6578 , 19911
9912 35 , 1 5540 6223 , 1 8439 3510 , 1 0525 7117 , 19912
9913 40 , 1 1220 4032 , 1 9431 7840 , 1 8020 0271 , 19913
9914 44 , 1 2688 1171 , 1 4181 6135 , 1 4484 1263 , 19914
9915 87 , 1 7225 9737 , 1 6812 5749 , 1 2581 7748 , 19915
9916 131 , 1 1942 4263 , 1 9524 1255 , 1 9365 8421 , 19916
9917 174 , 1 5221 4696 , 1 8976 4143 , 1 9505 8876 , 19917
9918 218 , 1 1403 5922 , 1 1785 2837 , 1 4107 3977 , 19918
9919 261 , 1 3773 0203 , 1 0092 9939 , 1 8234 0143 , 19919
9920 305 , 1 1014 2320 , 1 5473 5004 , 1 5094 5533 , 19920
9921 348 , 1 2726 3745 , 1 7211 2566 , 1 5673 6478 , 19921
9922 391 , 1 7328 8142 , 1 2230 7421 , 1 7051 8866 , 19922
9923 435 , 1 1970 0711 , 1 1401 7046 , 1 9938 8888 , 19923
9924 869 , 1 3881 1801 , 1 9428 4368 , 1 5764 8232 , 19924
9925 1303 , 1 7646 2009 , 1 8905 4704 , 1 8893 1073 , 19925
9926 1738 , 1 1506 3559 , 1 7005 0524 , 1 9009 7592 , 19926
9927 2172 , 1 2967 6283 , 1 8402 3667 , 1 0689 6630 , 19927
9928 2606 , 1 5846 4389 , 1 5650 2114 , 1 7278 5046 , 19928
9929 3041 , 1 1151 7900 , 1 5080 6878 , 1 2914 4154 , 19929
9930 3475 , 1 2269 1083 , 1 0850 6857 , 1 8724 4002 , 19930
9931 3909 , 1 4470 3047 , 1 3316 5442 , 1 6408 6591 , 19931
9932 4343 , 1 8806 8182 , 1 2566 2921 , 1 5872 6150 , 19932
9933 8686 , 1 7756 0047 , 1 2598 6861 , 1 0458 3204 , 19933
9934 13029 , 1 6830 5723 , 1 7791 4884 , 1 1932 7351 , 19934
9935 17372 , 1 6015 5609 , 1 3095 3052 , 1 3494 7574 , 19935
9936 21715 , 1 5297 7951 , 1 6443 0315 , 1 3251 3576 , 19936
9937 26058 , 1 4665 6719 , 1 0099 3379 , 1 5527 2929 , 19937
9938 30401 , 1 4108 9724 , 1 3326 3186 , 1 5271 5665 , 19938
9939 34744 , 1 3618 6973 , 1 3140 0875 , 1 3856 4102 , 19939
9940 39087 , 1 3186 9209 , 1 6113 3900 , 1 6705 9685 , 19940
9941 } 19941
9942 \cs_new:Npn __fp_exp_pos_large:NnnNwn #1#2#3 #4#5__fp_sep: #6 19942
9943 { 19943
9944 \exp_after:wN \exp_after:wN \exp_after:wN __fp_exp_large:NwN 19944
9945 \exp_after:wN \exp_after:wN \exp_after:wN #6 19945
9946 \exp_after:wN \c__fp_one_fixed_tl 19946
9947 \int_value:w #3 #4 \exp_stop_f: 19947
9948 #5 00000 __fp_sep: 19948
9949 } 19949
9950 \cs_new:Npn __fp_exp_large:NwN #1#2__fp_sep: #3 19950

```
9951 {
9952     \if_case:w #3 ~
9953         \exp_after:wN \__fp_fixed_continue:wn
9954     \else:
9955         \exp_after:wN \__fp_exp_intarray:w
9956         \int_value:w \__fp_int_eval:w 36 * #1 + 4 * #3 \exp_after:wN \__fp_sep:
9957     \fi:
9958     #2\__fp_sep:
9959     {
9960         \if_meaning:w 0 #1
9961             \exp_after:wN \__fp_exp_large_after:wn
9962         \else:
9963             \exp_after:wN \__fp_exp_large:NwN
9964             \int_value:w \__fp_int_eval:w #1 - 1 \exp_after:wN \scan_stop:
9965         \fi:
9966     }
9967 }
9968 \cs_new:Npn \__fp_exp_intarray:w #1 \__fp_sep:
9969 {
9970     +
9971     \__kernel_intarray_item:Nn \c__fp_exp_intarray
9972     { \__fp_int_eval:w #1 - 3 \scan_stop: }
9973     \exp_after:wN \use_i:nnn
9974     \exp_after:wN \__fp_fixed_mul:wn
9975     \int_value:w 0
9976     \exp_after:wN \__fp_exp_intarray_aux:w
9977     \int_value:w \__kernel_intarray_item:Nn
9978         \c__fp_exp_intarray { \__fp_int_eval:w #1 - 2 }
9979     \exp_after:wN \__fp_exp_intarray_aux:w
9980     \int_value:w \__kernel_intarray_item:Nn
9981         \c__fp_exp_intarray { \__fp_int_eval:w #1 - 1 }
9982     \exp_after:wN \__fp_exp_intarray_aux:w
9983     \int_value:w \__kernel_intarray_item:Nn
9984         \c__fp_exp_intarray {#1} \__fp_sep: \__fp_sep:
9985 }
9986 \cs_new:Npn \__fp_exp_intarray_aux:w 1 #1#2#3#4#5 \__fp_sep:
9987 { \__fp_sep: {#1#2#3#4} {#5} }
9988 \cs_new:Npn \__fp_exp_large_after:wn #1\__fp_sep: #2\__fp_sep: #3
9989 {
9990     \__fp_exp_Taylor:Nnnwn ? { } { } 0 #2\__fp_sep: { } #3
9991     \__fp_fixed_mul:wn #1\__fp_sep:
9992 }
9993 \cs_new:cpn { __fp_ \iow_char:N \^_o:ww }
9994     \s_fp \__fp_chk:w #1#2#3\__fp_sep: \s_fp \__fp_chk:w #4#5#6\__fp_sep:
9995 {
9996     \if_meaning:w 0 #4
```

9997	__fp_case_return_o:Nw \c_one_fp	1999
9998	\fi:	1999
9999	\if_case:w #2 \exp_stop_f:	1999
0000	\exp_after:wN \use_i:nn	2000
0001	\or:	2000
0002	__fp_case_return_o:Nw \c_nan_fp	2000
0003	\else:	2000
0004	\exp_after:wN __fp_pow_neg:ww	2000
0005	\exp:w \exp_end_continue_f:w \exp_after:wN \use:nn	2000
0006	\fi:	2000
0007	{	2000
0008	\if_meaning:w 1 #1	2000
0009	\exp_after:wN __fp_pow_normal_o:ww	2000
0010	\else:	2001
0011	\exp_after:wN __fp_pow_zero_or_inf:ww	2001
0012	\fi:	2001
0013	\s__fp __fp_chk:w #1#2#3__fp_sep:	2001
0014	}	2001
0015	{ \s__fp __fp_chk:w #4#5#6__fp_sep: \s__fp __fp_chk:w #1#2#3__fp_sep: }	2001
0016	\s__fp __fp_chk:w #4#5#6__fp_sep:	2001
0017	}	2001
0018	\cs_new:Npn __fp_pow_zero_or_inf:ww	2001
0019	\s__fp __fp_chk:w #1#2__fp_sep: \s__fp __fp_chk:w #3#4	2001
0020	{	2002
0021	\if_meaning:w 1 #4	2002
0022	__fp_case_return_same_o:w	2002
0023	\fi:	2002
0024	\if_meaning:w #1 #4	2002
0025	__fp_case_return_o:Nw \c_zero_fp	2002
0026	\fi:	2002
0027	\if_meaning:w 2 #1	2002
0028	__fp_case_return_o:Nw \c_inf_fp	2002
0029	\fi:	2002
0030	\if_meaning:w 2 #3	2003
0031	__fp_case_return_o:Nw \c_inf_fp	2003
0032	\else:	2003
0033	__fp_case_use:nw	2003
0034	{	2003
0035	__fp_division_by_zero_o:NNww \c_inf_fp ^	2003
0036	\s__fp __fp_chk:w #1 #2 __fp_sep:	2003
0037	}	2003
0038	\fi:	2003
0039	\s__fp __fp_chk:w #3#4	2003
0040	}	2004
0041	\cs_new:Npn __fp_pow_normal_o:ww	2004
0042	\s__fp __fp_chk:w 1 #1#2#3__fp_sep: \s__fp __fp_chk:w #4#5	2004

```
0043 {
0044     \if:w 0 \__fp_str_if_eq:nn { #2 #3 } { 1 {1000} {0000} {0000} {0000} }
0045     \if_int_compare:w #4 #1 = 32 \exp_stop_f:
0046     \exp_after:wN \__fp_case_return_ii_o:ww
0047     \fi:
0048     \__fp_case_return_o:Nww \c_one_fp
0049     \fi:
0050     \if_case:w #4 \exp_stop_f:
0051     \or:
0052     \exp_after:wN \__fp_pow_npos_o:Nww
0053     \exp_after:wN #5
0054     \or:
0055     \if_meaning:w 2 #5 \exp_after:wN \reverse_if:N \fi:
0056     \if_int_compare:w #2 > \c_zero_int
0057     \exp_after:wN \__fp_case_return_o:Nww
0058     \exp_after:wN \c_inf_fp
0059     \else:
0060     \exp_after:wN \__fp_case_return_o:Nww
0061     \exp_after:wN \c_zero_fp
0062     \fi:
0063     \or:
0064     \__fp_case_return_ii_o:ww
0065     \fi:
0066     \s__fp \__fp_chk:w 1 #1 {#2} #3 \__fp_sep:
0067     \s__fp \__fp_chk:w #4 #5
0068 }
0069 \cs_new:Npn \__fp_pow_npos_o:Nww #1 \s__fp \__fp_chk:w 1#2#3
0070 {
0071     \exp_after:wN \__fp_sanitimize:Nw
0072     \exp_after:wN 0
0073     \int_value:w
0074     \if:w #1 \if_int_compare:w #3 > \c_zero_int 0 \else: 2 \fi:
0075     \exp_after:wN \__fp_pow_npos_aux:NNnww
0076     \exp_after:wN +
0077     \exp_after:wN \__fp_fixed_to_float_o:wN
0078     \else:
0079     \exp_after:wN \__fp_pow_npos_aux:NNnww
0080     \exp_after:wN -
0081     \exp_after:wN \__fp_fixed_inv_to_float_o:wN
0082     \fi:
0083     {#3}
0084 }
0085 \cs_new:Npn \__fp_pow_npos_aux:NNnww
0086     #1#2#3#4#5\__fp_sep: \s__fp \__fp_chk:w 1#6#7#8\__fp_sep:
0087 {
0088     #1
```

```

0089 \__fp_int_eval:w
0090 \__fp_ln_significand:NNNNnnnnN #4#5
0091 \__fp_pow_exponent:wnN {#3}
0092 \__fp_fixed_mul:wwn #8 {0000}{0000} \__fp_sep:
0093 \__fp_pow_B:wwN #7\__fp_sep:
0094 #1 #2 0 % fixed_to_float_o:wn
0095 }
0096 \cs_new:Npn \__fp_pow_exponent:wnN #1\__fp_sep: #2
0097 {
0098 \if_int_compare:w #2 > \c_zero_int
0099 \exp_after:wN \__fp_pow_exponent:Nwnnnnnw % n\ln(10) - (-\ln(x))
0100 \exp_after:wN +
0101 \else:
0102 \exp_after:wN \__fp_pow_exponent:Nwnnnnnw % -(|n|\ln(10) + (-\ln(x)))
0103 \exp_after:wN -
0104 \fi:
0105 #2\__fp_sep: #1\__fp_sep:
0106 }
0107 \cs_new:Npn \__fp_pow_exponent:Nwnnnnnw #1#2\__fp_sep: #3#4#5#6#7#8\__fp_sep:
0108 { %^A todo: use that in ln.
0109 \exp_after:wN \__fp_fixed_mul_after:wwn
0110 \int_value:w \__fp_int_eval:w \c_fp_leading_shift_int
0111 \exp_after:wN \__fp_pack:NNNNNw
0112 \int_value:w \__fp_int_eval:w \c_fp_middle_shift_int
0113 #1#2*23025 - #1 #3
0114 \exp_after:wN \__fp_pack:NNNNNw
0115 \int_value:w \__fp_int_eval:w \c_fp_middle_shift_int
0116 #1 #2*8509 - #1 #4
0117 \exp_after:wN \__fp_pack:NNNNNw
0118 \int_value:w \__fp_int_eval:w \c_fp_middle_shift_int
0119 #1 #2*2994 - #1 #5
0120 \exp_after:wN \__fp_pack:NNNNNw
0121 \int_value:w \__fp_int_eval:w \c_fp_middle_shift_int
0122 #1 #2*0456 - #1 #6
0123 \exp_after:wN \__fp_pack:NNNNNw
0124 \int_value:w \__fp_int_eval:w \c_fp_trailing_shift_int
0125 #1 #2*8401 - #1 #7
0126 #1 ( #2*7991 - #8 ) / 1 0000 \__fp_sep: \__fp_sep:
0127 }
0128 \cs_new:Npn \__fp_pow_B:wwN #1#2#3#4#5#6\__fp_sep: #7\__fp_sep:
0129 {
0130 \if_int_compare:w #7 < \c_zero_int
0131 \exp_after:wN \__fp_pow_C_neg:w \int_value:w -
0132 \else:
0133 \if_int_compare:w #7 < 22 \exp_stop_f:
0134 \exp_after:wN \__fp_pow_C_pos:w \int_value:w

```

```
0135     \else:
0136         \exp_after:wN \__fp_pow_C_overflow:w \int_value:w
0137     \fi:
0138 \fi:
0139 #7 \exp_after:wN \__fp_sep:
0140 \int_value:w \__fp_int_eval:w 10 0000 + #1 \__fp_int_eval_end:
0141 #2#3#4#5#6 0000 0000 0000 0000 0000 0000 \__fp_sep: %^^A todo: how many 0?
0142 }
0143 \cs_new:Npn \__fp_pow_C_overflow:w #1\__fp_sep: #2\__fp_sep: #3
0144 {
0145     + 2 * \c__fp_max_exponent_int
0146     \exp_after:wN \__fp_fixed_continue:wn \c__fp_one_fixed_tl
0147 }
0148 \cs_new:Npn \__fp_pow_C_neg:w #1 \__fp_sep: 1
0149 {
0150     \exp_after:wN \exp_after:wN \exp_after:wN \__fp_pow_C_pack:w
0151     \prg_replicate:nn {#1} {0}
0152 }
0153 \cs_new:Npn \__fp_pow_C_pos:w #1\__fp_sep: 1
0154 { \__fp_pow_C_pos_loop:wN #1\__fp_sep: }
0155 \cs_new:Npn \__fp_pow_C_pos_loop:wN #1\__fp_sep: #2
0156 {
0157     \if_meaning:w 0 #1
0158         \exp_after:wN \__fp_pow_C_pack:w
0159         \exp_after:wN #2
0160     \else:
0161         \if_meaning:w 0 #2
0162             \exp_after:wN \__fp_pow_C_pos_loop:wN \int_value:w
0163         \else:
0164             \exp_after:wN \__fp_pow_C_overflow:w \int_value:w
0165             \fi:
0166             \__fp_int_eval:w #1 - 1 \exp_after:wN \__fp_sep:
0167         \fi:
0168 }
0169 \cs_new:Npn \__fp_pow_C_pack:w
0170 {
0171     \exp_after:wN \__fp_exp_large:NwN
0172     \exp_after:wN 5
0173     \c__fp_one_fixed_tl
0174 }
0175 \cs_new:Npn \__fp_pow_neg:www
0176 \s__fp \__fp_chk:w #1#2\__fp_sep: #3\__fp_sep: #4\__fp_sep:
0177 {
0178     \if_case:w \__fp_pow_neg_case:w #4 \__fp_sep:
0179         \exp_after:wN \__fp_pow_neg_aux:wNN
0180     \or:
```

```
0181 \if_int_compare:w \__fp_int_eval:w #1 / 2 = \c_one_int 2018
0182 \__fp_invalid_operation_o:Nww ^ #3\__fp_sep: #4\__fp_sep: 2018
0183 \exp:w \exp_end_continue_f:w 2018
0184 \exp_after:wN \exp_after:wN 2018
0185 \exp_after:wN \__fp_use_none_until_s:w 2018
0186 \fi: 2018
0187 \fi: 2018
0188 \__fp_exp_after_o:w 2018
0189 \s__fp \__fp_chk:w #1#2\__fp_sep: 2018
0190 } 2019
0191 \cs_new:Npn \__fp_pow_neg_aux:wNN #1 \s__fp \__fp_chk:w #2#3 2019
0192 { 2019
0193 \exp_after:wN \__fp_exp_after_o:w 2019
0194 \exp_after:wN \s__fp 2019
0195 \exp_after:wN \__fp_chk:w 2019
0196 \exp_after:wN #2 2019
0197 \int_value:w \__fp_int_eval:w 2 - #3 \__fp_int_eval_end: 2019
0198 } 2019
0199 \cs_new:Npn \__fp_pow_neg_case:w \s__fp \__fp_chk:w #1#2#3\__fp_sep: 2019
0200 { 2020
0201 \if_case:w #1 \exp_stop_f: 2020
0202 -1 2020
0203 \or: \__fp_pow_neg_case_aux:nnnnn #3 2020
0204 \or: -1 2020
0205 \else: 1 2020
0206 \fi: 2020
0207 \exp_stop_f: 2020
0208 } 2020
0209 \cs_new:Npn \__fp_pow_neg_case_aux:nnnnn #1#2#3#4#5 2020
0210 { 2021
0211 \if_int_compare:w #1 > \c__fp_prec_int 2021
0212 -1 2021
0213 \else: 2021
0214 \__fp_decimate:nNnnnn { \c__fp_prec_int - #1 } 2021
0215 \__fp_pow_neg_case_aux:Nnnw 2021
0216 {#2} {#3} {#4} {#5} 2021
0217 \fi: 2021
0218 } 2021
0219 \cs_new:Npn \__fp_pow_neg_case_aux:Nnnw #1#2#3#4 \__fp_sep: 2021
0220 { 2022
0221 \if_meaning:w 0 #1 2022
0222 \if_int_odd:w #3 \exp_stop_f: 2022
0223 0 2022
0224 \else: 2022
0225 -1 2022
0226 \fi: 2022
```



```
0227     \else:
0228         1
0229     \fi:
0230 }
0231 \int_const:Nn \c__fp_fact_max_arg_int { 3248 }
0232 \cs_new:Npn \__fp_fact_o:w #1 \s__fp \__fp_chk:w #2#3#4\__fp_sep: @
0233 {
0234     \if_case:w #2 \exp_stop_f:
0235         \__fp_case_return_o:Nw \c_one_fp
0236     \or:
0237     \or:
0238         \if_meaning:w 0 #3
0239             \exp_after:wN \__fp_case_return_same_o:w
0240         \fi:
0241     \or:
0242         \__fp_case_return_same_o:w
0243     \fi:
0244     \if_meaning:w 2 #3
0245         \__fp_case_use:nw { \__fp_invalid_operation_o:fw { fact } }
0246     \fi:
0247     \__fp_fact_pos_o:w
0248     \s__fp \__fp_chk:w #2 #3 #4 \__fp_sep:
0249 }
0250 \cs_new:Npn \__fp_fact_pos_o:w #1\__fp_sep:
0251 {
0252     \__fp_small_int:wTF #1\__fp_sep:
0253     { \__fp_fact_int_o:n }
0254     { \__fp_invalid_operation_o:fw { fact } #1\__fp_sep: }
0255 }
0256 \cs_new:Npn \__fp_fact_int_o:n #1
0257 {
0258     \if_int_compare:w #1 > \c__fp_fact_max_arg_int
0259         \__fp_case_return:nw
0260         {
0261             \exp_after:wN \exp_after:wN \exp_after:wN \__fp_overflow:w
0262             \exp_after:wN \c_inf_fp
0263         }
0264     \fi:
0265     \exp_after:wN \__fp_sanitise:Nw
0266     \exp_after:wN 0
0267     \int_value:w \__fp_int_eval:w
0268     \__fp_fact_loop_o:w #1 . 4 , { 1 } { } { } { } { } { } { } \__fp_sep:
0269 }
0270 \cs_new:Npn \__fp_fact_loop_o:w #1 . #2 \__fp_sep:
0271 {
0272     \if_int_compare:w #1 < 12 \exp_stop_f:
```

```

0273     \__fp_fact_small_o:w #1
0274     \fi:
0275     \exp_after:wN \__fp_ep_mul:wwwwn
0276     \exp_after:wN 4 \exp_after:wN ,
0277     \exp_after:wN { \int_value:w \__fp_int_eval:w #1 * (#1 - 1) }
0278     { } { } { } { } { } { } \__fp_sep:
0279     #2 \__fp_sep:
0280     {
0281         \exp_after:wN \__fp_fact_loop_o:w
0282         \int_value:w \__fp_int_eval:w #1 - 2 .
0283     }
0284 }
0285 \cs_new:Npn \__fp_fact_small_o:w #1 \fi: #2 \__fp_sep: #3 \__fp_sep: #4
0286 {
0287     \fi:
0288     \exp_after:wN \__fp_ep_mul:wwwwn
0289     \exp_after:wN 4 \exp_after:wN ,
0290     \exp_after:wN
0291     {
0292         \int_value:w
0293         \if_case:w #1 \exp_stop_f:
0294         1 \or: 1 \or: 2 \or: 6 \or: 24 \or: 120 \or: 720 \or: 5040
0295         \or: 40320 \or: 362880 \or: 3628800 \or: 39916800
0296         \fi:
0297         { } { } { } { } { } { } { } \__fp_sep:
0298         #3 \__fp_sep:
0299         \__fp_ep_to_float_o:wwN 0
0300     }
0301 %% File: l3fp-trig.dtx
0302 \tl_map_inline:nn
0303 {
0304     {acos} {acsc} {asec} {asin}
0305     {cos} {cot} {csc} {sec} {sin} {tan}
0306 }
0307 {
0308     \cs_new:cpe { __fp_parse_word_#1:N }
0309     {
0310         \exp_not:N \__fp_parse_unary_function:NNN
0311         \exp_not:c { __fp_#1_o:w }
0312         \exp_not:N \use_i:nn
0313     }
0314     \cs_new:cpe { __fp_parse_word_#1d:N }
0315     {
0316         \exp_not:N \__fp_parse_unary_function:NNN
0317         \exp_not:c { __fp_#1_o:w }
0318         \exp_not:N \use_ii:nn

```

```
0319     }
0320 }
0321 \cs_new:Npn \__fp_parse_word_acot:N
0322 { \__fp_parse_function:NNN \__fp_acot_o:Nw \use_i:nn }
0323 \cs_new:Npn \__fp_parse_word_acotd:N
0324 { \__fp_parse_function:NNN \__fp_acot_o:Nw \use_ii:nn }
0325 \cs_new:Npn \__fp_parse_word_atan:N
0326 { \__fp_parse_function:NNN \__fp_atan_o:Nw \use_i:nn }
0327 \cs_new:Npn \__fp_parse_word_atand:N
0328 { \__fp_parse_function:NNN \__fp_atan_o:Nw \use_ii:nn }
0329 \cs_new:Npn \__fp_sin_o:w #1 \s__fp \__fp_chk:w #2#3#4\__fp_sep: @
0330 {
0331     \if_case:w #2 \exp_stop_f:
0332         \__fp_case_return_same_o:w
0333     \or: \__fp_case_use:nw
0334         {
0335             \__fp_trig:NNNNNwn #1 \__fp_sin_series_o:NNwww
0336             \__fp_ep_to_float_o:wwN #3 0
0337         }
0338     \or: \__fp_case_use:nw
0339         { \__fp_invalid_operation_o:fw { #1 { sin } { sind } } }
0340     \else: \__fp_case_return_same_o:w
0341     \fi:
0342     \s__fp \__fp_chk:w #2 #3 #4\__fp_sep:
0343 }
0344 \cs_new:Npn \__fp_cos_o:w #1 \s__fp \__fp_chk:w #2#3\__fp_sep: @
0345 {
0346     \if_case:w #2 \exp_stop_f:
0347         \__fp_case_return_o:Nw \c_one_fp
0348     \or: \__fp_case_use:nw
0349         {
0350             \__fp_trig:NNNNNwn #1 \__fp_sin_series_o:NNwww
0351             \__fp_ep_to_float_o:wwN 0 2
0352         }
0353     \or: \__fp_case_use:nw
0354         { \__fp_invalid_operation_o:fw { #1 { cos } { cosd } } }
0355     \else: \__fp_case_return_same_o:w
0356     \fi:
0357     \s__fp \__fp_chk:w #2 #3\__fp_sep:
0358 }
0359 \cs_new:Npn \__fp_csc_o:w #1 \s__fp \__fp_chk:w #2#3#4\__fp_sep: @
0360 {
0361     \if_case:w #2 \exp_stop_f:
0362         \__fp_cot_zero_o:Nfw #3 { #1 { csc } { cscd } }
0363     \or: \__fp_case_use:nw
0364         {
```

```
0365         \__fp_trig:NNNNNwn #1 \__fp_sin_series_o:NNwww
0366         \__fp_ep_inv_to_float_o:wwN #3 0
0367     }
0368 \or: \__fp_case_use:nw
0369     { \__fp_invalid_operation_o:fw { #1 { csc } { cscd } } }
0370 \else: \__fp_case_return_same_o:w
0371 \fi:
0372 \s__fp \__fp_chk:w #2 #3 #4\__fp_sep:
0373 }
0374 \cs_new:Npn \__fp_sec_o:w #1 \s__fp \__fp_chk:w #2#3\__fp_sep: @
0375 {
0376     \if_case:w #2 \exp_stop_f:
0377         \__fp_case_return_o:Nw \c_one_fp
0378 \or: \__fp_case_use:nw
0379     {
0380         \__fp_trig:NNNNNwn #1 \__fp_sin_series_o:NNwww
0381         \__fp_ep_inv_to_float_o:wwN 0 2
0382     }
0383 \or: \__fp_case_use:nw
0384     { \__fp_invalid_operation_o:fw { #1 { sec } { secd } } }
0385 \else: \__fp_case_return_same_o:w
0386 \fi:
0387 \s__fp \__fp_chk:w #2 #3\__fp_sep:
0388 }
0389 \cs_new:Npn \__fp_tan_o:w #1 \s__fp \__fp_chk:w #2#3#4\__fp_sep: @
0390 {
0391     \if_case:w #2 \exp_stop_f:
0392         \__fp_case_return_same_o:w
0393 \or: \__fp_case_use:nw
0394     {
0395         \__fp_trig:NNNNNwn #1
0396         \__fp_tan_series_o:NNwww 0 #3 1
0397     }
0398 \or: \__fp_case_use:nw
0399     { \__fp_invalid_operation_o:fw { #1 { tan } { tand } } }
0400 \else: \__fp_case_return_same_o:w
0401 \fi:
0402 \s__fp \__fp_chk:w #2 #3 #4\__fp_sep:
0403 }
0404 \cs_new:Npn \__fp_cot_o:w #1 \s__fp \__fp_chk:w #2#3#4\__fp_sep: @
0405 {
0406     \if_case:w #2 \exp_stop_f:
0407         \__fp_cot_zero_o:Nfw #3 { #1 { cot } { cotd } }
0408 \or: \__fp_case_use:nw
0409     {
0410         \__fp_trig:NNNNNwn #1
```

```
0411         \__fp_tan_series_o:NNwww 2 #3 3
0412     }
0413     \or: \__fp_case_use:nw
0414         { \__fp_invalid_operation_o:fw { #1 { cot } { cotd } } }
0415     \else: \__fp_case_return_same_o:w
0416     \fi:
0417     \s__fp \__fp_chk:w #2 #3 #4\__fp_sep:
0418 }
0419 \cs_new:Npn \__fp_cot_zero_o:Nfw #1#2#3 \fi:
0420 {
0421     \fi:
0422     \token_if_eq_meaning:NNTF 0 #1
0423     { \exp_args:NNf \__fp_division_by_zero_o:Nnw \c_inf_fp }
0424     { \exp_args:NNf \__fp_division_by_zero_o:Nnw \c_minus_inf_fp }
0425     {#2}
0426 }
0427 \cs_new:Npn \__fp_trig:NNNNNwn #1#2#3#4#5 \s__fp \__fp_chk:w 1#6#7#8\__fp_sep:
0428 {
0429     \exp_after:wN #2
0430     \exp_after:wN #3
0431     \exp_after:wN #4
0432     \int_value:w \__fp_int_eval:w #5
0433     \exp_after:wN \exp_after:wN \exp_after:wN \exp_after:wN
0434     \if_int_compare:w #7 > #1 0 1 \exp_stop_f:
0435         #1 \__fp_trig_large:ww \__fp_trigd_large:ww
0436     \else:
0437         #1 \__fp_trig_small:ww \__fp_trigd_small:ww
0438     \fi:
0439     #7,#8{0000}{0000}\__fp_sep:
0440 }
0441 \cs_new:Npn \__fp_trig_small:ww #1,#2\__fp_sep:
0442     { \__fp_ep_to_fixed:wwn #1,#2\__fp_sep: . #1,#2\__fp_sep: }
0443 \cs_new:Npn \__fp_trigd_small:ww #1,#2\__fp_sep:
0444     {
0445         \__fp_ep_mul_raw:wwwN
0446         -1,{1745}{3292}{5199}{4329}{5769}{2369}\__fp_sep: #1,#2\__fp_sep:
0447         \__fp_trig_small:ww
0448     }
0449 \cs_new:Npn \__fp_trigd_large:ww #1, #2#3#4#5#6#7\__fp_sep:
0450     {
0451         \exp_after:wN \__fp_pack_eight:wNNNNNNNN
0452         \exp_after:wN \__fp_pack_eight:wNNNNNNNN
0453         \exp_after:wN \__fp_pack_twice_four:wNNNNNNNN
0454         \exp_after:wN \__fp_pack_twice_four:wNNNNNNNN
0455         \exp_after:wN \__fp_trigd_large_auxi:nnnnwNNNN
0456         \exp_after:wN \__fp_sep:
```

```
0457 \exp:w \exp_end_continue_f:w 2045
0458 \prg_replicate:nn { \int_max:nn { 22 - #1 } { 0 } } { 0 } 2045
0459 #2#3#4#5#6#7 0000 0000 0000 ! 2045
0460 } 2046
0461 \cs_new:Npn \__fp_trigd_large_auxi:nnnnwNNNN #1#2#3#4#5\__fp_sep: #6#7#8#9 2046
0462 { 2046
0463 \exp_after:wN \__fp_trigd_large_auxii:wNw 2046
0464 \int_value:w \__fp_int_eval:w #1 + #2 2046
0465 - (#1 + #2 - 4) / 9 * 9 \__fp_int_eval_end: 2046
0466 #3\__fp_sep: 2046
0467 #4\__fp_sep: #5{#6#7#8#9}\__fp_sep: 2046
0468 } 2046
0469 \cs_new:Npn \__fp_trigd_large_auxii:wNw #1\__fp_sep: #2#3\__fp_sep: 2046
0470 { 2047
0471 + (#1#2 - 4) / 9 * 2 2047
0472 \exp_after:wN \__fp_trigd_large_auxiii:www 2047
0473 \int_value:w \__fp_int_eval:w #1#2 2047
0474 - (#1#2 - 4) / 9 * 9 \__fp_int_eval_end: #3 \__fp_sep: 2047
0475 } 2047
0476 \cs_new:Npn \__fp_trigd_large_auxiii:www #1\__fp_sep: #2\__fp_sep: #3! 2047
0477 { 2047
0478 \if_int_compare:w #1 < 4500 \exp_stop_f: 2047
0479 \exp_after:wN \__fp_use_i_until_s:nw 2047
0480 \exp_after:wN \__fp_fixed_continue:wn 2048
0481 \else: 2048
0482 + 1 2048
0483 \fi: 2048
0484 \__fp_fixed_sub:wwn {9000}{0000}{0000}{0000}{0000}{0000}\__fp_sep: 2048
0485 {#1}#2{0000}{0000}\__fp_sep: 2048
0486 { \__fp_trigd_small:ww 2, } 2048
0487 } 2048
0488 \intarray_const_from_clist:Nn \c__fp_trig_intarray 2048
0489 { 2048
0490 100000000, 100000000, 115915494, 130918953, 135768883, 176337251, 2049
0491 143620344, 159645740, 145644874, 176673440, 158896797, 163422653, 2049
0492 150901138, 102766253, 108595607, 128427267, 157958036, 189291184, 2049
0493 161145786, 152877967, 141073169, 198392292, 139966937, 140907757, 2049
0494 130777463, 196925307, 168871739, 128962173, 197661693, 136239024, 2049
0495 117236290, 111832380, 111422269, 197557159, 140461890, 108690267, 2049
0496 139561204, 189410936, 193784408, 155287230, 199946443, 140024867, 2049
0497 123477394, 159610898, 132309678, 130749061, 166986462, 180469944, 2049
0498 186521878, 181574786, 156696424, 110389958, 174139348, 160998386, 2049
0499 180991999, 162442875, 158517117, 188584311, 117518767, 116054654, 2049
0500 175369880, 109739460, 136475933, 137680593, 102494496, 163530532, 2050
0501 171567755, 103220324, 177781639, 171660229, 146748119, 159816584, 2050
0502 106060168, 103035998, 113391198, 174988327, 186654435, 127975507, 2050
```

0503	100162406,	177564388,	184957131,	108801221,	199376147,	168137776,	20503
0504	147378906,	133068046,	145797848,	117613124,	127314069,	196077502,	20504
0505	145002977,	159857089,	105690279,	167851315,	125210016,	131774602,	20505
0506	109248116,	106240561,	145620314,	164840892,	148459191,	143521157,	20506
0507	154075562,	100871526,	160680221,	171591407,	157474582,	172259774,	20507
0508	162853998,	175155329,	139081398,	117724093,	158254797,	107332871,	20508
0509	190406999,	175907657,	170784934,	170393589,	182808717,	134256403,	20509
0510	166895116,	162545705,	194332763,	112686500,	126122717,	197115321,	20510
0511	112599504,	138667945,	103762556,	108363171,	116952597,	158128224,	20511
0512	194162333,	143145106,	112353687,	185631136,	136692167,	114206974,	20512
0513	169601292,	150578336,	105311960,	185945098,	139556718,	170995474,	20513
0514	165104316,	123815517,	158083944,	129799709,	199505254,	138756612,	20514
0515	194458833,	106846050,	178529151,	151410404,	189298850,	163881607,	20515
0516	176196993,	107341038,	199957869,	118905980,	193737772,	106187543,	20516
0517	122271893,	101366255,	126123878,	103875388,	181106814,	106765434,	20517
0518	108282785,	126933426,	179955607,	107903860,	160352738,	199624512,	20518
0519	159957492,	176297023,	159409558,	143011648,	129641185,	157771240,	20519
0520	157544494,	157021789,	176979240,	194903272,	194770216,	164960356,	20520
0521	153181535,	144003840,	168987471,	176915887,	163190966,	150696440,	20521
0522	147769706,	187683656,	177810477,	197954503,	153395758,	130188183,	20522
0523	186879377,	166124814,	195305996,	155802190,	183598751,	103512712,	20523
0524	190432315,	180498719,	168687775,	194656634,	162210342,	104440855,	20524
0525	149785037,	192738694,	129353661,	193778292,	187359378,	143470323,	20525
0526	102371458,	137923557,	111863634,	119294601,	183182291,	196416500,	20526
0527	187830793,	131353497,	179099745,	186492902,	167450609,	189368909,	20527
0528	145883050,	133703053,	180547312,	132158094,	131976760,	132283131,	20528
0529	141898097,	149822438,	133517435,	169898475,	101039500,	168388003,	20529
0530	197867235,	199608024,	100273901,	108749548,	154787923,	156826113,	20530
0531	199489032,	168997427,	108349611,	149208289,	103776784,	174303550,	20531
0532	145684560,	183671479,	130845672,	133270354,	185392556,	120208683,	20532
0533	193240995,	162211753,	131839402,	109707935,	170774965,	149880868,	20533
0534	160663609,	168661967,	103747454,	121028312,	119251846,	122483499,	20534
0535	111611495,	166556037,	196967613,	199312829,	196077608,	127799010,	20535
0536	107830360,	102338272,	198790854,	102387615,	157445430,	192601191,	20536
0537	100543379,	198389046,	154921248,	129516070,	172853005,	122721023,	20537
0538	160175233,	113173179,	175931105,	103281551,	109373913,	163964530,	20538
0539	157926071,	180083617,	195487672,	146459804,	173977292,	144810920,	20539
0540	109371257,	186918332,	189588628,	139904358,	168666639,	175673445,	20540
0541	114095036,	137327191,	174311388,	106638307,	125923027,	159734506,	20541
0542	105482127,	178037065,	133778303,	121709877,	134966568,	149080032,	20542
0543	169885067,	141791464,	168350828,	116168533,	114336160,	173099514,	20543
0544	198531198,	119733758,	144420984,	116559541,	152250643,	139431286,	20544
0545	144403838,	183561508,	179771645,	101706470,	167518774,	156059160,	20545
0546	187168578,	157939226,	123475633,	117111329,	198655941,	159689071,	20546
0547	198506887,	144230057,	151919770,	156900382,	118392562,	120338742,	20547
0548	135362568,	108354156,	151729710,	188117217,	195936832,	156488518,	20548

0549	174997487,	108553116,	159830610,	113921445,	144601614,	188452770,	20549
0550	125114110,	170248521,	173974510,	138667364,	103872860,	109967489,	20550
0551	131735618,	112071174,	104788993,	168886556,	192307848,	150230570,	20551
0552	157144063,	163863202,	136852010,	174100574,	185922811,	115721968,	20552
0553	100397824,	175953001,	166958522,	112303464,	118773650,	143546764,	20553
0554	164565659,	171901123,	108476709,	193097085,	191283646,	166919177,	20554
0555	169387914,	133315566,	150669813,	121641521,	100895711,	172862384,	20555
0556	126070678,	145176011,	113450800,	169947684,	122356989,	162488051,	20556
0557	157759809,	153397080,	185475059,	175362656,	149034394,	145420581,	20557
0558	178864356,	183042000,	131509559,	147434392,	152544850,	167491429,	20558
0559	108647514,	142303321,	133245695,	111634945,	167753939,	142403609,	20559
0560	105438335,	152829243,	142203494,	184366151,	146632286,	102477666,	20560
0561	166049531,	140657343,	157553014,	109082798,	180914786,	169343492,	20561
0562	127376026,	134997829,	195701816,	119643212,	133140475,	176289748,	20562
0563	140828911,	174097478,	126378991,	181699939,	148749771,	151989818,	20563
0564	172666294,	160183053,	195832752,	109236350,	168538892,	128468247,	20564
0565	125997252,	183007668,	156937583,	165972291,	198244297,	147406163,	20565
0566	181831139,	158306744,	134851692,	185973832,	137392662,	140243450,	20566
0567	119978099,	140402189,	161348342,	173613676,	144991382,	171541660,	20567
0568	163424829,	136374185,	106122610,	186132119,	198633462,	184709941,	20568
0569	183994274,	129559156,	128333990,	148038211,	175011612,	111667205,	20569
0570	119125793,	103552929,	124113440,	131161341,	112495318,	138592695,	20570
0571	184904438,	146807849,	109739828,	108855297,	104515305,	139914009,	20571
0572	188698840,	188365483,	166522246,	168624087,	125401404,	100911787,	20572
0573	142122045,	123075334,	173972538,	114940388,	141905868,	142311594,	20573
0574	163227443,	139066125,	116239310,	162831953,	123883392,	113153455,	20574
0575	163815117,	152035108,	174595582,	101123754,	135976815,	153401874,	20575
0576	107394340,	136339780,	138817210,	104531691,	182951948,	179591767,	20576
0577	139541778,	179243527,	161740724,	160593916,	102732282,	187946819,	20577
0578	136491289,	149714953,	143255272,	135916592,	198072479,	198580612,	20578
0579	169007332,	118844526,	179433504,	155801952,	149256630,	162048766,	20579
0580	116134365,	133992028,	175452085,	155344144,	109905129,	182727454,	20580
0581	165911813,	122232840,	151166615,	165070983,	175574337,	129548631,	20581
0582	120411217,	116380915,	160616116,	157320000,	183306114,	160618128,	20582
0583	103262586,	195951602,	146321661,	138576614,	180471993,	127077713,	20583
0584	116441201,	159496011,	106328305,	120759583,	148503050,	179095584,	20584
0585	198298218,	167402898,	138551383,	123957020,	180763975,	150429225,	20585
0586	198476470,	171016426,	197438450,	143091658,	164528360,	132493360,	20586
0587	143546572,	137557916,	113663241,	120457809,	196971566,	134022158,	20587
0588	180545794,	131328278,	100552461,	132088901,	187421210,	192448910,	20588
0589	141005215,	149680971,	113720754,	100571096,	134066431,	135745439,	20589
0590	191597694,	135788920,	179342561,	177830222,	137011486,	142492523,	20590
0591	192487287,	113132021,	176673607,	156645598,	127260957,	141566023,	20591
0592	143787436,	129132109,	174858971,	150713073,	191040726,	143541417,	20592
0593	197057222,	165479803,	181512759,	157912400,	125344680,	148220261,	20593
0594	173422990,	101020483,	106246303,	137964746,	178190501,	181183037,	20594

0595	151538028,	179523433,	141955021,	135689770,	191290561,	143178787,	2059
0596	192086205,	174499925,	178975690,	118492103,	124206471,	138519113,	2059
0597	188147564,	102097605,	154895793,	178514140,	141453051,	151583964,	2059
0598	128232654,	106020603,	131189158,	165702720,	186250269,	191639375,	2059
0599	115278873,	160608114,	155694842,	110322407,	177272742,	116513642,	2059
0600	134366992,	171634030,	194053074,	180652685,	109301658,	192136921,	2060
0601	141431293,	171341061,	157153714,	106203978,	147618426,	150297807,	2060
0602	186062669,	169960809,	118422347,	163350477,	146719017,	145045144,	2060
0603	161663828,	146208240,	186735951,	102371302,	190444377,	194085350,	2060
0604	134454426,	133413062,	163074595,	113830310,	122931469,	134466832,	2060
0605	185176632,	182415152,	110179422,	164439571,	181217170,	121756492,	2060
0606	119644493,	196532222,	118765848,	182445119,	109401340,	150443213,	2060
0607	198586286,	121083179,	139396084,	143898019,	114787389,	177233102,	2060
0608	186310131,	148695521,	126205182,	178063494,	157118662,	177825659,	2060
0609	188310053,	151552316,	165984394,	109022180,	163144545,	121212978,	2060
0610	197344714,	188741258,	126822386,	102360271,	109981191,	152056882,	2061
0611	134723983,	158013366,	106837863,	128867928,	161973236,	172536066,	2061
0612	185216856,	132011948,	197807339,	158419190,	166595838,	167852941,	2061
0613	124187182,	117279875,	106103946,	106481958,	157456200,	160892122,	2061
0614	184163943,	173846549,	158993202,	184812364,	133466119,	170732430,	2061
0615	195458590,	173361878,	162906318,	150165106,	126757685,	112163575,	2061
0616	188696307,	145199922,	100107766,	176830946,	198149756,	122682434,	2061
0617	179367131,	108412102,	119520899,	148191244,	140487511,	171059184,	2061
0618	141399078,	189455775,	118462161,	190415309,	134543802,	180893862,	2061
0619	180732375,	178615267,	179711433,	123241969,	185780563,	176301808,	2061
0620	184386640,	160717536,	183213626,	129671224,	126094285,	140110963,	2062
0621	121826276,	151201170,	122552929,	128965559,	146082049,	138409069,	2062
0622	107606920,	103954646,	119164002,	115673360,	117909631,	187289199,	2062
0623	186343410,	186903200,	157966371,	103128612,	135698881,	176403642,	2062
0624	152540837,	109810814,	183519031,	121318624,	172281810,	150845123,	2062
0625	169019064,	166322359,	138872454,	163073727,	128087898,	130041018,	2062
0626	194859136,	173742589,	141812405,	167291912,	138003306,	134499821,	2062
0627	196315803,	186381054,	124578934,	150084553,	128031351,	118843410,	2062
0628	107373060,	159565443,	173624887,	171292628,	198074235,	139074061,	2062
0629	178690578,	144431052,	174262641,	176783005,	182214864,	162289361,	2062
0630	192966929,	192033046,	169332843,	181580535,	164864073,	118444059,	2063
0631	195496893,	153773183,	167266131,	130108623,	158802128,	180432893,	2063
0632	144562140,	147978945,	142337360,	158506327,	104399819,	132635916,	2063
0633	168734194,	136567839,	101281912,	120281622,	195003330,	112236091,	2063
0634	185875592,	101959081,	122415367,	194990954,	148881099,	175891989,	2063
0635	108115811,	163538891,	163394029,	123722049,	184837522,	142362091,	2063
0636	100834097,	156679171,	100841679,	157022331,	178971071,	102928884,	2063
0637	189701309,	195339954,	124415335,	106062584,	139214524,	133864640,	2063
0638	134324406,	157317477,	155340540,	144810061,	177612569,	108474646,	2063
0639	114329765,	143900008,	138265211,	145210162,	136643111,	197987319,	2063
0640	102751191,	144121361,	169620456,	193602633,	161023559,	162140467,	2064

0641	102901215,	167964187,	135746835,	187317233,	110047459,	163339773,	2064
0642	124770449,	118885134,	141536376,	100915375,	164267438,	145016622,	2064
0643	113937193,	106748706,	128815954,	164819775,	119220771,	102367432,	2064
0644	189062690,	170911791,	194127762,	112245117,	123546771,	115640433,	2064
0645	135772061,	166615646,	174474627,	130562291,	133320309,	153340551,	2064
0646	138417181,	194605321,	150142632,	180008795,	151813296,	175497284,	2064
0647	167018836,	157425342,	150169942,	131069156,	134310662,	160434122,	2064
0648	105213831,	158797111,	150754540,	163290657,	102484886,	148697402,	2064
0649	187203725,	198692811,	149360627,	140384233,	128749423,	132178578,	2064
0650	177507355,	171857043,	178737969,	134023369,	102911446,	196144864,	2065
0651	197697194,	134527467,	144296030,	189437192,	154052665,	188907106,	2065
0652	162062575,	150993037,	199766583,	167936112,	181374511,	104971506,	2065
0653	115378374,	135795558,	167972129,	135876446,	130937572,	103221320,	2065
0654	124605656,	161129971,	131027586,	191128460,	143251843,	143269155,	2065
0655	129284585,	173495971,	150425653,	199302112,	118494723,	121323805,	2065
0656	116549802,	190991967,	168151180,	122483192,	151273721,	199792134,	2065
0657	133106764,	121874844,	126215985,	112167639,	167793529,	182985195,	2065
0658	185453921,	106957880,	158685312,	132775454,	133229161,	198905318,	2065
0659	190537253,	191582222,	192325972,	178133427,	181825606,	148823337,	2065
0660	160719681,	101448145,	131983362,	137910767,	112550175,	128826351,	2066
0661	183649210,	135725874,	110356573,	189469487,	154446940,	118175923,	2066
0662	106093708,	128146501,	185742532,	149692127,	164624247,	183221076,	2066
0663	154737505,	168198834,	156410354,	158027261,	125228550,	131543250,	2066
0664	139591848,	191898263,	104987591,	115406321,	103542638,	190012837,	2066
0665	142615518,	178773183,	175862355,	117537850,	169565995,	170028011,	2066
0666	158412588,	170150030,	117025916,	174630208,	142412449,	112839238,	2066
0667	105257725,	114737141,	123102301,	172563968,	130555358,	132628403,	2066
0668	183638157,	168682846,	143304568,	105994018,	170010719,	152092970,	2066
0669	117799058,	132164175,	179868116,	158654714,	177489647,	116547948,	2066
0670	183121404,	131836079,	184431405,	157311793,	149677763,	173989893,	2067
0671	102277656,	107058530,	140837477,	152640947,	143507039,	152145247,	2067
0672	101683884,	107090870,	161471944,	137225650,	128231458,	172995869,	2067
0673	173831689,	171268519,	139042297,	111072135,	107569780,	137262545,	2067
0674	181410950,	138270388,	198736451,	162848201,	180468288,	120582913,	2067
0675	153390138,	135649144,	130040157,	106509887,	192671541,	174507066,	2067
0676	186888783,	143805558,	135011967,	145862340,	180595327,	124727843,	2067
0677	182925939,	157715840,	136885940,	198993925,	152416883,	178793572,	2067
0678	179679516,	154076673,	192703125,	164187609,	162190243,	104699348,	2067
0679	159891990,	160012977,	174692145,	132970421,	167781726,	115178506,	2067
0680	153008552,	155999794,	102099694,	155431545,	127458567,	104403686,	2068
0681	168042864,	184045128,	181182309,	179349696,	127218364,	192935516,	2068
0682	120298724,	169583299,	148193297,	183358034,	159023227,	105261254,	2068
0683	121144370,	184359584,	194433836,	138388317,	175184116,	108817112,	2068
0684	151279233,	137457721,	193398208,	119005406,	132929377,	175306906,	2068
0685	160741530,	149976826,	147124407,	176881724,	186734216,	185881509,	2068
0686	191334220,	175930947,	117385515,	193408089,	157124410,	163472089,	2068

```
0687 131949128, 180783576, 131158294, 100549708, 191802336, 165960770, 2068
0688 170927599, 101052702, 181508688, 197828549, 143403726, 142729262, 2068
0689 110348701, 139928688, 153550062, 106151434, 130786653, 196085995, 2068
0690 100587149, 139141652, 106530207, 100852656, 124074703, 166073660, 2069
0691 153338052, 163766757, 120188394, 197277047, 122215363, 138511354, 2069
0692 183463624, 161985542, 159938719, 133367482, 104220974, 149956672, 2069
0693 170250544, 164232439, 157506869, 159133019, 137469191, 142980999, 2069
0694 134242305, 150172665, 121209241, 145596259, 160554427, 159095199, 2069
0695 168243130, 184279693, 171132070, 121049823, 123819574, 171759855, 2069
0696 119501864, 163094029, 175943631, 194450091, 191506160, 149228764, 2069
0697 132319212, 197034460, 193584259, 126727638, 168143633, 109856853, 2069
0698 127860243, 132141052, 133076065, 188414958, 158718197, 107124299, 2069
0699 159592267, 181172796, 144388537, 196763139, 127431422, 179531145, 2069
0700 100064922, 112650013, 132686230, 121550837, 2070
0701 } 2070
0702 \cs_new:Npn \__fp_trig_large:ww #1, #2#3#4#5#6\__fp_sep: 2070
0703 { 2070
0704 \exp_after:wN \__fp_trig_large_auxi:w 2070
0705 \int_value:w \__fp_int_eval:w (#1 - 4) / 8 \exp_after:wN , 2070
0706 \int_value:w #1 , \__fp_sep: 2070
0707 {#2}{#3}{#4}{#5} \__fp_sep: 2070
0708 } 2070
0709 \cs_new:Npn \__fp_trig_large_auxi:w #1, #2, 2070
0710 { 2071
0711 \exp_after:wN \exp_after:wN 2071
0712 \exp_after:wN \__fp_trig_large_auxiii:w 2071
0713 \cs:w 2071
0714 use_none:n \prg_replicate:nn { #2 - #1 * 8 } { n } 2071
0715 \exp_after:wN 2071
0716 \cs_end: 2071
0717 \int_value:w 2071
0718 \__kernel_intarray_item:Nn \c__fp_trig_intarray 2071
0719 { \__fp_int_eval:w #1 + 1 \scan_stop: } 2071
0720 \exp_after:wN \__fp_trig_large_auxiii:w \int_value:w 2072
0721 \__kernel_intarray_item:Nn \c__fp_trig_intarray 2072
0722 { \__fp_int_eval:w #1 + 2 \scan_stop: } 2072
0723 \exp_after:wN \__fp_trig_large_auxiii:w \int_value:w 2072
0724 \__kernel_intarray_item:Nn \c__fp_trig_intarray 2072
0725 { \__fp_int_eval:w #1 + 3 \scan_stop: } 2072
0726 \exp_after:wN \__fp_trig_large_auxiii:w \int_value:w 2072
0727 \__kernel_intarray_item:Nn \c__fp_trig_intarray 2072
0728 { \__fp_int_eval:w #1 + 4 \scan_stop: } 2072
0729 \exp_after:wN \__fp_trig_large_auxiii:w \int_value:w 2072
0730 \__kernel_intarray_item:Nn \c__fp_trig_intarray 2073
0731 { \__fp_int_eval:w #1 + 5 \scan_stop: } 2073
0732 \exp_after:wN \__fp_trig_large_auxiii:w \int_value:w 2073
```

```

0733 \__kernel_intarray_item:Nn \c__fp_trig_intarray
0734 { \__fp_int_eval:w #1 + 6 \scan_stop: }
0735 \exp_after:wN \__fp_trig_large_auxiii:w \int_value:w
0736 \__kernel_intarray_item:Nn \c__fp_trig_intarray
0737 { \__fp_int_eval:w #1 + 7 \scan_stop: }
0738 \exp_after:wN \__fp_trig_large_auxiii:w \int_value:w
0739 \__kernel_intarray_item:Nn \c__fp_trig_intarray
0740 { \__fp_int_eval:w #1 + 8 \scan_stop: }
0741 \exp_after:wN \__fp_trig_large_auxiii:w \int_value:w
0742 \__kernel_intarray_item:Nn \c__fp_trig_intarray
0743 { \__fp_int_eval:w #1 + 9 \scan_stop: }
0744 \exp_stop_f:
0745 }
0746 \cs_new:Npn \__fp_trig_large_auxii:w
0747 {
0748 \__fp_pack_twice_four:wNNNNNNNN \__fp_pack_twice_four:wNNNNNNNN
0749 \__fp_pack_twice_four:wNNNNNNNN \__fp_pack_twice_four:wNNNNNNNN
0750 \__fp_pack_twice_four:wNNNNNNNN \__fp_pack_twice_four:wNNNNNNNN
0751 \__fp_pack_twice_four:wNNNNNNNN \__fp_pack_twice_four:wNNNNNNNN
0752 \__fp_trig_large_auxv:www \__fp_sep:
0753 }
0754 \cs_new:Npn \__fp_trig_large_auxiii:w 1 { \exp_stop_f: }
0755 \cs_new:Npn \__fp_trig_large_auxv:www #1\__fp_sep: #2\__fp_sep: #3\__fp_sep:
0756 {
0757 \exp_after:wN \__fp_use_i_until_s:nw
0758 \exp_after:wN \__fp_trig_large_auxvii:w
0759 \int_value:w \__fp_int_eval:w \c__fp_leading_shift_int
0760 \prg_replicate:nn { 13 }
0761 { \__fp_trig_large_auxvi:wnnnnnnnn }
0762 + \c__fp_trailing_shift_int - \c__fp_middle_shift_int
0763 \__fp_use_i_until_s:nw
0764 \__fp_sep: #3 #1 \__fp_sep: \__fp_sep:
0765 }
0766 \cs_new:Npn \__fp_trig_large_auxvi:wnnnnnnnn #1\__fp_sep: #2#3#4#5#6#7#8#9
0767 {
0768 \exp_after:wN \__fp_trig_large_pack:NNNNNw
0769 \int_value:w \__fp_int_eval:w \c__fp_middle_shift_int
0770 + #2*#9 + #3*#8 + #4*#7 + #5*#6
0771 #1\__fp_sep: {#2}{#3}{#4}{#5} {#7}{#8}{#9}
0772 }
0773 \cs_new:Npn \__fp_trig_large_pack:NNNNNw #1#2#3#4#5#6\__fp_sep:
0774 { + #1#2#3#4#5 \__fp_sep: #6 }
0775 \cs_new:Npn \__fp_trig_large_auxvii:w #1#2#3
0776 {
0777 \exp_after:wN \__fp_trig_large_auxviii:ww
0778 \int_value:w \__fp_int_eval:w (#1#2#3 - 62) / 125 \__fp_sep:

```

```

0779 #1#2#3
0780 }
0781 \cs_new:Npn \__fp_trig_large_auxviii:ww #1\__fp_sep:
0782 {
0783   + #1
0784   \if_int_odd:w #1 \exp_stop_f:
0785     \exp_after:wN \__fp_trig_large_auxix:Nw
0786     \exp_after:wN -
0787   \else:
0788     \exp_after:wN \__fp_trig_large_auxix:Nw
0789     \exp_after:wN +
0790   \fi:
0791 }
0792 \cs_new:Npn \__fp_trig_large_auxix:Nw
0793 {
0794   \exp_after:wN \__fp_use_i_until_s:nw
0795   \exp_after:wN \__fp_trig_large_auxxi:w
0796   \int_value:w \__fp_int_eval:w \c__fp_leading_shift_int
0797   \prg_replicate:nn { 13 }
0798   { \__fp_trig_large_auxx:wNNNNN }
0799   + \c__fp_trailing_shift_int - \c__fp_middle_shift_int
0800   \__fp_sep:
0801 }
0802 \cs_new:Npn \__fp_trig_large_auxx:wNNNNN #1\__fp_sep: #2 #3#4#5#6
0803 {
0804   \exp_after:wN \__fp_trig_large_pack:NNNNNw
0805   \int_value:w \__fp_int_eval:w \c__fp_middle_shift_int
0806   #2 8 * #3#4#5#6
0807   #1\__fp_sep: #2
0808 }
0809 \cs_new:Npn \__fp_trig_large_auxxi:w #1\__fp_sep:
0810 {
0811   \exp_after:wN \__fp_ep_mul_raw:wwwN
0812   \int_value:w \__fp_int_eval:w 0 \__fp_ep_to_ep_loop:N #1 \__fp_sep: \__fp_sep: !
0813   0,{7853}{9816}{3397}{4483}{0961}{5661}\__fp_sep:
0814   \__fp_trig_small:ww
0815 }
0816 \cs_new:Npn \__fp_sin_series_o:NNwww #1#2#3. #4\__fp_sep:
0817 {
0818   \__fp_fixed_mul:wwn #4\__fp_sep: #4\__fp_sep:
0819   {
0820     \exp_after:wN \__fp_sin_series_aux_o:NNwww
0821     \exp_after:wN #1
0822     \int_value:w
0823     \if_int_odd:w \__fp_int_eval:w (#3 + 2) / 4 \__fp_int_eval_end:
0824     #2

```

```
0825         \else:
0826             \if_meaning:w #2 0 2 \else: 0 \fi:
0827         \fi:
0828     {#3}
0829 }
0830 }
0831 \cs_new:Npn \__fp_sin_series_aux_o:NNnwww #1#2#3 #4\__fp_sep: #5,#6\__fp_sep:
0832 {
0833     \if_int_odd:w \__fp_int_eval:w #3 / 2 \__fp_int_eval_end:
0834         \exp_after:wN \use_i:nn
0835     \else:
0836         \exp_after:wN \use_ii:nn
0837     \fi:
0838     { % 1/18!
0839         \__fp_fixed_mul_sub_back:wwwn {0000}{0000}{0000}{0001}{5619}{2070}\__fp_sep:
0840             #4\__fp_sep:
0841             {0000}{0000}{0000}{0477}{9477}{3324}\__fp_sep:
0842         \__fp_fixed_mul_sub_back:wwwn #4\__fp_sep:
0843             {0000}{0000}{0011}{4707}{4559}{7730}\__fp_sep:
0844         \__fp_fixed_mul_sub_back:wwwn #4\__fp_sep:
0845             {0000}{0000}{2087}{6756}{9878}{6810}\__fp_sep:
0846         \__fp_fixed_mul_sub_back:wwwn #4\__fp_sep:
0847             {0000}{0027}{5573}{1922}{3985}{8907}\__fp_sep:
0848         \__fp_fixed_mul_sub_back:wwwn #4\__fp_sep:
0849             {0000}{2480}{1587}{3015}{8730}{1587}\__fp_sep:
0850         \__fp_fixed_mul_sub_back:wwwn #4\__fp_sep:
0851             {0013}{8888}{8888}{8888}{8888}{8889}\__fp_sep:
0852         \__fp_fixed_mul_sub_back:wwwn #4\__fp_sep:
0853             {0416}{6666}{6666}{6666}{6666}{6667}\__fp_sep:
0854         \__fp_fixed_mul_sub_back:wwwn #4\__fp_sep:
0855             {5000}{0000}{0000}{0000}{0000}{0000}\__fp_sep:
0856         \__fp_fixed_mul_sub_back:wwwn #4\__fp_sep:
0857             {10000}{0000}{0000}{0000}{0000}{0000}\__fp_sep:
0858         { \__fp_fixed_continue:wn 0, }
0859     }
0860     { % 1/17!
0861         \__fp_fixed_mul_sub_back:wwwn {0000}{0000}{0000}{0028}{1145}{7254}\__fp_sep:
0862             #4\__fp_sep:
0863             {0000}{0000}{0000}{7647}{1637}{3182}\__fp_sep:
0864         \__fp_fixed_mul_sub_back:wwwn #4\__fp_sep:
0865             {0000}{0000}{0160}{5904}{3836}{8216}\__fp_sep:
0866         \__fp_fixed_mul_sub_back:wwwn #4\__fp_sep:
0867             {0000}{0002}{5052}{1083}{8544}{1719}\__fp_sep:
0868         \__fp_fixed_mul_sub_back:wwwn #4\__fp_sep:
0869             {0000}{0275}{5731}{9223}{9858}{9065}\__fp_sep:
0870         \__fp_fixed_mul_sub_back:wwwn #4\__fp_sep:
```



```
0871             {0001}{9841}{2698}{4126}{9841}{2698}\__fp_sep: 2087
0872 \__fp_fixed_mul_sub_back:wwwn #4\__fp_sep: 2087
0873             {0083}{3333}{3333}{3333}{3333}{3333}\__fp_sep: 2087
0874 \__fp_fixed_mul_sub_back:wwwn #4\__fp_sep: 2087
0875             {1666}{6666}{6666}{6666}{6666}{6667}\__fp_sep: 2087
0876 \__fp_fixed_mul_sub_back:wwwn#4\__fp_sep: 2087
0877             {10000}{0000}{0000}{0000}{0000}{0000}\__fp_sep: 2087
0878 { \__fp_ep_mul:wwwn 0, } #5,#6\__fp_sep: 2087
0879 } 2087
0880 { 2088
0881 \exp_after:wN \__fp_sanitizew 2088
0882 \exp_after:wN #2 2088
0883 \int_value:w \__fp_int_eval:w #1 2088
0884 } 2088
0885 #2 2088
0886 } 2088
0887 \cs_new:Npn \__fp_tan_series_o:NNwww #1#2#3. #4\__fp_sep: 2088
0888 { 2088
0889 \__fp_fixed_mul:wwn #4\__fp_sep: #4\__fp_sep: 2088
0890 { 2089
0891 \exp_after:wN \__fp_tan_series_aux_o:Nnwww 2089
0892 \int_value:w 2089
0893 \if_int_odd:w \__fp_int_eval:w #3 / 2 \__fp_int_eval_end: 2089
0894 \exp_after:wN \reverse_if:N 2089
0895 \fi: 2089
0896 \if_meaning:w #1#2 2 \else: 0 \fi: 2089
0897 {#3} 2089
0898 } 2089
0899 } 2089
0900 \cs_new:Npn \__fp_tan_series_aux_o:Nnwww #1 #2 #3\__fp_sep: #4,#5\__fp_sep: 2090
0901 { 2090
0902 \__fp_fixed_mul_sub_back:wwwn {0000}{0000}{1527}{3493}{0856}{7059}\__fp_sep: 2090
0903 #3\__fp_sep: 2090
0904 {0000}{0159}{6080}{0274}{5257}{6472}\__fp_sep: 2090
0905 \__fp_fixed_mul_sub_back:wwwn #3\__fp_sep: 2090
0906 {0002}{4571}{2320}{0157}{2558}{8481}\__fp_sep: 2090
0907 \__fp_fixed_mul_sub_back:wwwn #3\__fp_sep: 2090
0908 {0115}{5830}{7533}{5397}{3168}{2147}\__fp_sep: 2090
0909 \__fp_fixed_mul_sub_back:wwwn #3\__fp_sep: 2090
0910 {1929}{8245}{6140}{3508}{7719}{2982}\__fp_sep: 2091
0911 \__fp_fixed_mul_sub_back:wwwn #3\__fp_sep: 2091
0912 {10000}{0000}{0000}{0000}{0000}{0000}\__fp_sep: 2091
0913 { \__fp_ep_mul:wwwn 0, } #4,#5\__fp_sep: 2091
0914 { 2091
0915 \__fp_fixed_mul_sub_back:wwwn {0000}{0007}{0258}{0681}{9408}{4706}\__fp_sep: 2091
0916 #3\__fp_sep: 2091
```

```
0917             {0000}{2343}{7175}{1399}{6151}{7670}\__fp_sep: 2091
0918 \__fp_fixed_mul_sub_back:wwwn #3\__fp_sep: 2091
0919             {0019}{2638}{4588}{9232}{8861}{3691}\__fp_sep: 2091
0920 \__fp_fixed_mul_sub_back:wwwn #3\__fp_sep: 2092
0921             {0536}{6357}{0691}{4344}{6852}{4252}\__fp_sep: 2092
0922 \__fp_fixed_mul_sub_back:wwwn #3\__fp_sep: 2092
0923             {5263}{1578}{9473}{6842}{1052}{6315}\__fp_sep: 2092
0924 \__fp_fixed_mul_sub_back:wwwn#3\__fp_sep: 2092
0925             {10000}{0000}{0000}{0000}{0000}{0000}\__fp_sep: 2092
0926 { 2092
0927     \reverse_if:N \if_int_odd:w 2092
0928         \__fp_int_eval:w (#2 - 1) / 2 \__fp_int_eval_end: 2092
0929     \exp_after:wN \__fp_reverse_args:Nww 2092
0930     \fi: 2093
0931     \__fp_ep_div:wwwn 0, 2093
0932 } 2093
0933 } 2093
0934 { 2093
0935     \exp_after:wN \__fp_sanitize:Nw 2093
0936     \exp_after:wN #1 2093
0937     \int_value:w \__fp_int_eval:w \__fp_ep_to_float_o:wwN 2093
0938 } 2093
0939 #1 2093
0940 } 2094
0941 \cs_new:Npn \__fp_atan_o:Nw #1 2094
0942 { 2094
0943     \__fp_parse_function_one_two:nnw 2094
0944     { #1 { atan } { atand } } 2094
0945     { \__fp_atan_default:w \__fp_atanii_o:Nww #1 } 2094
0946 } 2094
0947 \cs_new:Npn \__fp_acot_o:Nw #1 2094
0948 { 2094
0949     \__fp_parse_function_one_two:nnw 2094
0950     { #1 { acot } { acotd } } 2095
0951     { \__fp_atan_default:w \__fp_acotii_o:Nww #1 } 2095
0952 } 2095
0953 \cs_new:Npe \__fp_atan_default:w #1#2#3 @ { #1 #2 #3 \c_one_fp @ } 2095
0954 \cs_new:Npn \__fp_atanii_o:Nww 2095
0955     #1 \s__fp \__fp_chk:w #2#3#4\__fp_sep: \s__fp \__fp_chk:w #5 #6 @ 2095
0956 { 2095
0957     \if_meaning:w 3 #2 \__fp_case_return_i_o:ww \fi: 2095
0958     \if_meaning:w 3 #5 \__fp_case_return_ii_o:ww \fi: 2095
0959     \if_case:w 2095
0960         \if_meaning:w #2 #5 2096
0961             \if_meaning:w 1 #2 10 \else: 0 \fi: 2096
0962     \else: 2096
```

```
0963 \if_int_compare:w #2 > #5 \exp_stop_f: 1 \else: 2 \fi:
0964 \fi:
0965 \exp_stop_f:
0966 \__fp_case_return:nw { \__fp_atan_inf_o:NNNw #1 #3 2 }
0967 \or: \__fp_case_return:nw { \__fp_atan_inf_o:NNNw #1 #3 4 }
0968 \or: \__fp_case_return:nw { \__fp_atan_inf_o:NNNw #1 #3 0 }
0969 \fi:
0970 \__fp_atan_normal_o:NNnwNnw #1
0971 \s__fp \__fp_chk:w #2#3#4\__fp_sep:
0972 \s__fp \__fp_chk:w #5 #6
0973 }
0974 \cs_new:Npn \__fp_acotii_o:Nww #1#2\__fp_sep: #3\__fp_sep:
0975 { \__fp_atanii_o:Nww #1#3\__fp_sep: #2\__fp_sep: }
0976 \cs_new:Npn \__fp_atan_inf_o:NNNw #1#2#3 \s__fp \__fp_chk:w #4#5#6\__fp_sep:
0977 {
0978 \exp_after:wN \__fp_atan_combine_o:NwwwwwN
0979 \exp_after:wN #2
0980 \int_value:w \__fp_int_eval:w
0981 \if_meaning:w 2 #5 7 - \fi: #3 \exp_after:wN \__fp_sep:
0982 \c__fp_one_fixed_tl
0983 {0000}{0000}{0000}{0000}{0000}{0000}\__fp_sep:
0984 0,{0000}{0000}{0000}{0000}{0000}{0000}\__fp_sep: #1
0985 }
0986 \cs_new_protected:Npn \__fp_atan_normal_o:NNnwNnw
0987 #1 \s__fp \__fp_chk:w 1#2#3#4\__fp_sep: \s__fp \__fp_chk:w 1#5#6#7\__fp_sep:
0988 {
0989 \__fp_atan_test_o:NwwNwwN
0990 #2 #3, #4{0000}{0000}\__fp_sep:
0991 #5 #6, #7{0000}{0000}\__fp_sep: #1
0992 }
0993 \cs_new:Npn \__fp_atan_test_o:NwwNwwN #1#2,#3\__fp_sep: #4#5,#6\__fp_sep:
0994 {
0995 \exp_after:wN \__fp_atan_combine_o:NwwwwwN
0996 \exp_after:wN #1
0997 \int_value:w \__fp_int_eval:w
0998 \if_meaning:w 2 #4
0999 7 - \__fp_int_eval:w
1000 \fi:
1001 \if_int_compare:w
1002 \__fp_ep_compare:www #2,#3\__fp_sep: #5,#6\__fp_sep: > \c_zero_int
1003 3 -
1004 \exp_after:wN \__fp_reverse_args:Nww
1005 \fi:
1006 \__fp_atan_div:wnwnw #2,#3\__fp_sep: #5,#6\__fp_sep:
1007 }
1008 \cs_new:Npn \__fp_atan_div:wnwnw #1,#2#3\__fp_sep: #4,#5#6\__fp_sep:
```

```
1009 {
1010     \if_int_compare:w
1011         \__fp_int_eval:w 41421 * #5 < #2 000
1012         \if_case:w \__fp_int_eval:w #4 - #1 \__fp_int_eval_end:
1013             00 \or: 0 \fi:
1014         \exp_stop_f:
1015         \exp_after:wN \__fp_atan_near:wwwn
1016     \fi:
1017     0
1018     \__fp_ep_div:wwwn #1,{#2}#3\__fp_sep: #4,{#5}#6\__fp_sep:
1019     \__fp_atan_auxi:ww
1020 }
1021 \cs_new:Npn \__fp_atan_near:wwwn
1022     0 \__fp_ep_div:wwwn #1,#2\__fp_sep: #3,
1023 {
1024     1
1025     \__fp_ep_to_fixed:wwn #1 - #3, #2\__fp_sep:
1026     \__fp_atan_near_aux:wwn
1027 }
1028 \cs_new:Npn \__fp_atan_near_aux:wwn #1\__fp_sep: #2\__fp_sep:
1029 {
1030     \__fp_fixed_add:wwn #1\__fp_sep: #2\__fp_sep:
1031     { \__fp_fixed_sub:wwn #2\__fp_sep: #1\__fp_sep: { \__fp_ep_div:wwwn 0, } 0, }
1032 }
1033 \cs_new:Npn \__fp_atan_auxi:ww #1,#2\__fp_sep:
1034 { \__fp_ep_to_fixed:wwn #1,#2\__fp_sep: \__fp_atan_auxii:w #1,#2\__fp_sep: }
1035 \cs_new:Npn \__fp_atan_auxii:w #1\__fp_sep:
1036 {
1037     \__fp_fixed_mul:wwn #1\__fp_sep: #1\__fp_sep:
1038     {
1039         \__fp_atan_Taylor_loop:www 39 \__fp_sep:
1040         {0000}{0000}{0000}{0000}{0000}{0000} \__fp_sep:
1041     }
1042     ! #1\__fp_sep:
1043 }
1044 \cs_new:Npn \__fp_atan_Taylor_loop:www #1\__fp_sep: #2\__fp_sep: #3\__fp_sep:
1045 {
1046     \if_int_compare:w #1 = - \c_one_int
1047         \__fp_atan_Taylor_break:w
1048     \fi:
1049     \exp_after:wN \__fp_fixed_div_int:wwN \c__fp_one_fixed_tl #1\__fp_sep:
1050     \__fp_rrot:www \__fp_fixed_mul_sub_back:wwwn #2\__fp_sep: #3\__fp_sep:
1051     {
1052         \exp_after:wN \__fp_atan_Taylor_loop:www
1053         \int_value:w \__fp_int_eval:w #1 - 2 \__fp_sep:
1054     }
```

```
1055 #3\__fp_sep:
1056 }
1057 \cs_new:Npn \__fp_atan_Taylor_break:w
1058   \fi: #1 \__fp_fixed_mul_sub_back:wwn #2\__fp_sep: #3 !
1059   { \fi: \__fp_sep: #2 \__fp_sep: }
1060 \cs_new:Npn \__fp_atan_combine_o:NwwwN
1061   #1 #2\__fp_sep: #3\__fp_sep: #4\__fp_sep: #5,#6\__fp_sep: #7
1062   {
1063     \exp_after:wN \__fp_sanitize:Nw
1064     \exp_after:wN #1
1065     \int_value:w \__fp_int_eval:w
1066     \if_meaning:w 0 #2
1067       \exp_after:wN \use_i:nn
1068     \else:
1069       \exp_after:wN \use_ii:nn
1070     \fi:
1071     { #5 \__fp_fixed_mul:wwn #3\__fp_sep: #6\__fp_sep: }
1072     {
1073       \__fp_fixed_mul:wwn #3\__fp_sep: #4\__fp_sep:
1074       {
1075         \exp_after:wN \__fp_atan_combine_aux:ww
1076         \int_value:w \__fp_int_eval:w #2 / 2 \__fp_sep: #2\__fp_sep:
1077       }
1078     }
1079     { #7 \__fp_fixed_to_float_o:wN \__fp_fixed_to_float_rad_o:wN }
1080     #1
1081   }
1082 \cs_new:Npn \__fp_atan_combine_aux:ww #1\__fp_sep: #2\__fp_sep:
1083   {
1084     \__fp_fixed_mul_short:wwn
1085     {7853}{9816}{3397}{4483}{0961}{5661}\__fp_sep:
1086     {#1}{0000}{0000}\__fp_sep:
1087     {
1088       \if_int_odd:w #2 \exp_stop_f:
1089       \exp_after:wN \__fp_fixed_sub:wwn
1090     \else:
1091       \exp_after:wN \__fp_fixed_add:wwn
1092     \fi:
1093   }
1094 }
1095 \cs_new:Npn \__fp_asin_o:w #1 \s__fp \__fp_chk:w #2#3\__fp_sep: @
1096 {
1097   \if_case:w #2 \exp_stop_f:
1098   \__fp_case_return_same_o:w
1099 \or:
1100   \__fp_case_use:nw
```

```
1101         { \__fp_asin_normal_o:NfwNnnnnnw #1 { #1 { asin } { asind } } }
1102     \or:
1103         \__fp_case_use:nw
1104         { \__fp_invalid_operation_o:fw { #1 { asin } { asind } } }
1105     \else:
1106         \__fp_case_return_same_o:w
1107     \fi:
1108     \s__fp \__fp_chk:w #2 #3\__fp_sep:
1109 }
1110 \cs_new:Npn \__fp_acos_o:w #1 \s__fp \__fp_chk:w #2#3\__fp_sep: @
1111 {
1112     \if_case:w #2 \exp_stop_f:
1113         \__fp_case_use:nw { \__fp_atan_inf_o:NNWw #1 0 4 }
1114     \or:
1115         \__fp_case_use:nw
1116         {
1117             \__fp_asin_normal_o:NfwNnnnnnw #1 { #1 { acos } { acosd } }
1118             \__fp_reverse_args:Nww
1119         }
1120     \or:
1121         \__fp_case_use:nw
1122         { \__fp_invalid_operation_o:fw { #1 { acos } { acosd } } }
1123     \else:
1124         \__fp_case_return_same_o:w
1125     \fi:
1126     \s__fp \__fp_chk:w #2 #3\__fp_sep:
1127 }
1128 \cs_new:Npn \__fp_asin_normal_o:NfwNnnnnnw
1129 #1#2#3 \s__fp \__fp_chk:w 1#4#5#6#7#8#9\__fp_sep:
1130 {
1131     \if_int_compare:w #5 < \c_one_int
1132         \exp_after:wN \__fp_use_none_until_s:w
1133     \fi:
1134     \if_int_compare:w \__fp_int_eval:w #5 + #6#7 + #8#9 = 1000 0001 ~
1135         \exp_after:wN \__fp_use_none_until_s:w
1136     \fi:
1137     \__fp_use_i:ww
1138     \__fp_invalid_operation_o:fw {#2}
1139     \s__fp \__fp_chk:w 1#4#{#5}{#6}{#7}{#8}{#9}\__fp_sep:
1140     \__fp_asin_auxi_o:NnNww
1141     #1 {#3} #4 #5,{#6}{#7}{#8}{#9}{0000}{0000}\__fp_sep:
1142 }
1143 \cs_new:Npn \__fp_asin_auxi_o:NnNww #1#2#3#4,#5\__fp_sep:
1144 {
1145     \__fp_ep_to_fixed:wwn #4,#5\__fp_sep:
1146     \__fp_asin_isqrt:wn
```

```
1147 \__fp_ep_mul:wwwn #4,#5\__fp_sep: 2114
1148 \__fp_ep_to_ep:wwN 2114
1149 \__fp_fixed_continue:wn 2114
1150 { #2 \__fp_atan_test_o:NwwNwwN #3 } 2115
1151 0 1,{1000}{0000}{0000}{0000}{0000}{0000}\__fp_sep: #1 2115
1152 } 2115
1153 \cs_new:Npn \__fp_asin_isqrt:wn #1\__fp_sep: 2115
1154 { 2115
1155 \exp_after:wN \__fp_fixed_sub:wwn \c__fp_one_fixed_tl #1\__fp_sep: 2115
1156 { 2115
1157 \__fp_fixed_add_one:wN #1\__fp_sep: 2115
1158 \__fp_fixed_continue:wn { \__fp_ep_mul:wwwn 0, } 0, 2115
1159 } 2115
1160 \__fp_ep_isqrt:wwn 2116
1161 } 2116
1162 \cs_new:Npn \__fp_acsc_o:w #1 \s__fp \__fp_chk:w #2#3#4\__fp_sep: @ 2116
1163 { 2116
1164 \if_case:w \if_meaning:w 2 #2 #3 \fi: #2 \exp_stop_f: 2116
1165 \__fp_case_use:nw 2116
1166 { \__fp_invalid_operation_o:fw { #1 { acsc } { acscd } } } 2116
1167 \or: \__fp_case_use:nw 2116
1168 { \__fp_acsc_normal_o:NfwNnw #1 { #1 { acsc } { acscd } } } 2116
1169 \or: \__fp_case_return_o:Nw \c_zero_fp 2116
1170 \or: \__fp_case_return_same_o:w 2117
1171 \else: \__fp_case_return_o:Nw \c_minus_zero_fp 2117
1172 \fi: 2117
1173 \s__fp \__fp_chk:w #2 #3 #4\__fp_sep: 2117
1174 } 2117
1175 \cs_new:Npn \__fp_asec_o:w #1 \s__fp \__fp_chk:w #2#3\__fp_sep: @ 2117
1176 { 2117
1177 \if_case:w #2 \exp_stop_f: 2117
1178 \__fp_case_use:nw 2117
1179 { \__fp_invalid_operation_o:fw { #1 { asec } { asecd } } } 2117
1180 \or: 2118
1181 \__fp_case_use:nw 2118
1182 { 2118
1183 \__fp_acsc_normal_o:NfwNnw #1 { #1 { asec } { asecd } } 2118
1184 \__fp_reverse_args:Nww 2118
1185 } 2118
1186 \or: \__fp_case_use:nw { \__fp_atan_inf_o:NNnw #1 0 4 } 2118
1187 \else: \__fp_case_return_same_o:w 2118
1188 \fi: 2118
1189 \s__fp \__fp_chk:w #2 #3\__fp_sep: 2118
1190 } 2119
1191 \cs_new:Npn \__fp_acsc_normal_o:NfwNnw #1#2#3 \s__fp \__fp_chk:w 1#4#5#6\__fp_sep: 2119
1192 { 2119
```



```
1193 \int_compare:nNnTF {#5} < 1
1194 {
1195     \__fp_invalid_operation_o:fw {#2}
1196     \s__fp \__fp_chk:w 1#4{#5}#6\__fp_sep:
1197 }
1198 {
1199     \__fp_ep_div:wwwn
1200     1,{1000}{0000}{0000}{0000}{0000}{0000}\__fp_sep:
1201     #5,#6{0000}{0000}\__fp_sep:
1202     { \__fp_asin_auxi_o:NnNww #1 {#3} #4 }
1203 }
1204 }
1205 %% File: l3fp-convert.dtx
1206 \cs_new:Npn \__fp_tuple_convert:Nw #1 \s__fp_tuple \__fp_tuple_chk:w #2 \__fp_sep:
1207 {
1208     \int_case:nnF { \__fp_array_count:n {#2} }
1209     {
1210         { 0 } { ( ) }
1211         { 1 } { \__fp_tuple_convert_end:w @ { #1 #2 , } }
1212     }
1213     {
1214         \__fp_tuple_convert_loop:nNw { } #1
1215         #2 { ? \__fp_tuple_convert_end:w } \__fp_sep:
1216         @ { \use_none:nn }
1217     }
1218 }
1219 \cs_new:Npn \__fp_tuple_convert_loop:nNw #1#2#3#4\__fp_sep: #5 @ #6
1220 {
1221     \use_none:n #3
1222     \exp_args:Nf \__fp_tuple_convert_loop:nNw { #2 #3#4 \__fp_sep: } #2 #5
1223     @ { #6 , ~ #1 }
1224 }
1225 \cs_new:Npn \__fp_tuple_convert_end:w #1 @ #2
1226 { \exp_after:wN ( \exp:w \exp_end_continue_f:w #2 ) }
1227 \cs_new:Npn \__fp_trim_zeros:w #1 \__fp_sep:
1228 {
1229     \__fp_trim_zeros_loop:w #1 \__fp_sep:
1230     \__fp_trim_zeros_loop:w 0\__fp_sep:
1231     \__fp_trim_zeros_dot:w .\__fp_sep:
1232     \s__fp_stop
1233 }
1234 \cs_new:Npn \__fp_trim_zeros_loop:w #1 0\__fp_sep: #2 { #2 #1 \__fp_sep: #2 }
1235 \cs_new:Npn \__fp_trim_zeros_dot:w #1 .\__fp_sep:
1236 { \__fp_trim_zeros_end:w #1 \__fp_sep: }
1237 \cs_new:Npn \__fp_trim_zeros_end:w #1 \__fp_sep: #2 \s__fp_stop { #1 }
1238 \cs_new:Npn \fp_to_scientific:N #1
```

```
1239 { \exp_after:wN \__fp_to_scientific_dispatch:w #1 } 21239
1240 \cs_generate_variant:Nn \fp_to_scientific:N { c } 21240
1241 \cs_new:Npn \fp_to_scientific:n 21241
1242 { 21242
1243     \exp_after:wN \__fp_to_scientific_dispatch:w 21243
1244     \exp:w \exp_end_continue_f:w \__fp_parse:n 21244
1245 } 21245
1246 \cs_new:Npn \__fp_to_scientific_dispatch:w #1 21246
1247 { 21247
1248     \__fp_change_func_type:NNN 21248
1249     #1 \__fp_to_scientific:w \__fp_to_scientific_recover:w 21249
1250     #1 21250
1251 } 21251
1252 \cs_new:Npn \__fp_to_scientific_recover:w #1 #2 \__fp_sep: 21252
1253 { 21253
1254     \__fp_error:nffn { unknown-type } { \tl_to_str:n { #2 \__fp_sep: } } { } { } 21254
1255     nan 21255
1256 } 21256
1257 \cs_new:Npn \__fp_tuple_to_scientific:w 21257
1258 { \__fp_tuple_convert:Nw \__fp_to_scientific_dispatch:w } 21258
1259 \cs_new:Npn \__fp_to_scientific:w \s__fp \__fp_chk:w #1#2 21259
1260 { 21260
1261     \if_meaning:w 2 #2 \exp_after:wN - \exp:w \exp_end_continue_f:w \fi: 21261
1262     \if_case:w #1 \exp_stop_f: 21262
1263         \__fp_case_return:nw { 0.0000000000000000e0 } 21263
1264     \or: \exp_after:wN \__fp_to_scientific_normal:wnnnnn 21264
1265     \or: 21265
1266         \__fp_case_use:nw 21266
1267         { 21267
1268             \__fp_invalid_operation:nnw 21268
1269             { \fp_to_scientific:N \c__fp_overflowing_fp } 21269
1270             { fp_to_scientific } 21270
1271         } 21271
1272     \or: 21272
1273         \__fp_case_use:nw 21273
1274         { 21274
1275             \__fp_invalid_operation:nnw 21275
1276             { \fp_to_scientific:N \c_zero_fp } 21276
1277             { fp_to_scientific } 21277
1278         } 21278
1279     \fi: 21279
1280     \s__fp \__fp_chk:w #1 #2 21280
1281 } 21281
1282 \cs_new:Npn \__fp_to_scientific_normal:wnnnnn 21282
1283 \s__fp \__fp_chk:w 1 #1 #2 #3#4#5#6 \__fp_sep: 21283
1284 { 21284
```

```
1285 \exp_after:wN \__fp_to_scientific_normal:wNw
1286 \exp_after:wN e
1287 \int_value:w \__fp_int_eval:w #2 - 1
1288 \__fp_sep: #3 #4 #5 #6 \__fp_sep:
1289 }
1290 \cs_new:Npn \__fp_to_scientific_normal:wNw #1 \__fp_sep: #2#3\__fp_sep:
1291 { #2.#3 #1 }
1292 \cs_new:Npn \fp_to_decimal:N #1
1293 { \exp_after:wN \__fp_to_decimal_dispatch:w #1 }
1294 \cs_generate_variant:Nn \fp_to_decimal:N { c }
1295 \cs_new:Npn \fp_to_decimal:n
1296 {
1297   \exp_after:wN \__fp_to_decimal_dispatch:w
1298   \exp:w \exp_end_continue_f:w \__fp_parse:n
1299 }
1300 \cs_new:Npn \__fp_to_decimal_dispatch:w #1
1301 {
1302   \__fp_change_func_type:NNN
1303   #1 \__fp_to_decimal:w \__fp_to_decimal_recover:w
1304   #1
1305 }
1306 \cs_new:Npn \__fp_to_decimal_recover:w #1 #2 \__fp_sep:
1307 {
1308   \__fp_error:nffn { unknown-type } { \tl_to_str:n { #2 \__fp_sep: } } { } { }
1309   nan
1310 }
1311 \cs_new:Npn \__fp_tuple_to_decimal:w
1312 { \__fp_tuple_convert:Nw \__fp_to_decimal_dispatch:w }
1313 \cs_new:Npn \__fp_to_decimal:w \s__fp \__fp_chk:w #1#2
1314 {
1315   \if_meaning:w 2 #2 \exp_after:wN - \exp:w \exp_end_continue_f:w \fi:
1316   \if_case:w #1 \exp_stop_f:
1317     \__fp_case_return:nw { 0 }
1318   \or: \exp_after:wN \__fp_to_decimal_normal:wnnnnn
1319   \or:
1320     \__fp_case_use:nw
1321     {
1322       \__fp_invalid_operation:nnw
1323       { \fp_to_decimal:N \c__fp_overflowing_fp }
1324       { fp_to_decimal }
1325     }
1326   \or:
1327     \__fp_case_use:nw
1328     {
1329       \__fp_invalid_operation:nnw
1330       { 0 }
```

```
1331         { fp_to_decimal }
1332     }
1333     \fi:
1334     \s__fp \__fp_chk:w #1 #2
1335 }
1336 \cs_new:Npn \__fp_to_decimal_normal:wnnnnn
1337   \s__fp \__fp_chk:w 1 #1 #2 #3#4#5#6 \__fp_sep:
1338 {
1339   \int_compare:nNnTF {#2} > 0
1340   {
1341     \int_compare:nNnTF {#2} < \c__fp_prec_int
1342     {
1343       \__fp_decimate:nNnnnn { \c__fp_prec_int - #2 }
1344       \__fp_to_decimal_large:Nnnw
1345     }
1346     {
1347       \exp_after:wN \exp_after:wN
1348       \exp_after:wN \__fp_to_decimal_huge:wnnnn
1349       \prg_replicate:nn { #2 - \c__fp_prec_int } { 0 } \__fp_sep:
1350     }
1351     {#3} {#4} {#5} {#6}
1352   }
1353   {
1354     \exp_after:wN \__fp_trim_zeros:w
1355     \exp_after:wN 0
1356     \exp_after:wN .
1357     \exp:w \exp_end_continue_f:w \prg_replicate:nn { - #2 } { 0 }
1358     #3#4#5#6 \__fp_sep:
1359   }
1360 }
1361 \cs_new:Npn \__fp_to_decimal_large:Nnnw #1#2#3#4\__fp_sep:
1362 {
1363   \exp_after:wN \__fp_trim_zeros:w \int_value:w
1364   \if_int_compare:w #2 > \c_zero_int
1365     #2
1366   \fi:
1367   \exp_stop_f:
1368   #3.#4 \__fp_sep:
1369 }
1370 \cs_new:Npn \__fp_to_decimal_huge:wnnnn #1\__fp_sep: #2#3#4#5 { #2#3#4#5 #1 }
1371 \cs_new:Npn \fp_to_tl:N #1 { \exp_after:wN \__fp_to_tl_dispatch:w #1 }
1372 \cs_generate_variant:Nn \fp_to_tl:N { c }
1373 \cs_new:Npn \fp_to_tl:n
1374 {
1375   \exp_after:wN \__fp_to_tl_dispatch:w
1376   \exp:w \exp_end_continue_f:w \__fp_parse:n
```

```
1377 }
1378 \cs_new:Npn \__fp_to_tl_dispatch:w #1
1379 { \__fp_change_func_type:NNN #1 \__fp_to_tl:w \__fp_to_tl_recover:w #1 }
1380 \cs_new:Npn \__fp_to_tl_recover:w #1 #2 \__fp_sep:
1381 {
1382   \__fp_error:nffn { unknown-type } { \tl_to_str:n { #2 \__fp_sep: } } { } { }
1383   nan
1384 }
1385 \cs_new:Npn \__fp_tuple_to_tl:w
1386 { \__fp_tuple_convert:Nw \__fp_to_tl_dispatch:w }
1387 \cs_new:Npn \__fp_to_tl:w \s__fp \__fp_chk:w #1#2
1388 {
1389   \if_meaning:w 2 #2 \exp_after:wN - \exp:w \exp_end_continue_f:w \fi:
1390   \if_case:w #1 \exp_stop_f:
1391     \__fp_case_return:nw { 0 }
1392   \or: \exp_after:wN \__fp_to_tl_normal:nnnnn
1393   \or: \__fp_case_return:nw { inf }
1394   \else: \__fp_case_return:nw { nan }
1395   \fi:
1396 }
1397 \cs_new:Npn \__fp_to_tl_normal:nnnnn #1
1398 {
1399   \int_compare:nTF
1400     { -2 <= #1 <= \c__fp_prec_int }
1401     { \__fp_to_decimal_normal:wnnnnn }
1402     { \__fp_to_tl_scientific:wnnnnn }
1403   \s__fp \__fp_chk:w 1 0 {#1}
1404 }
1405 \cs_new:Npn \__fp_to_tl_scientific:wnnnnn
1406   \s__fp \__fp_chk:w 1 #1 #2 #3#4#5#6 \__fp_sep:
1407 {
1408   \exp_after:wN \__fp_to_tl_scientific:wNw
1409   \exp_after:wN e
1410   \int_value:w \__fp_int_eval:w #2 - 1
1411   \__fp_sep: #3 #4 #5 #6 \__fp_sep:
1412 }
1413 \cs_new:Npn \__fp_to_tl_scientific:wNw #1 \__fp_sep: #2#3\__fp_sep:
1414 { \__fp_trim_zeros:w #2.#3 \__fp_sep: #1 }
1415 \cs_new:Npn \fp_to_dim:N #1
1416 { \exp_after:wN \__fp_to_dim_dispatch:w #1 }
1417 \cs_generate_variant:Nn \fp_to_dim:N { c }
1418 \cs_new:Npn \fp_to_dim:n
1419 {
1420   \exp_after:wN \__fp_to_dim_dispatch:w
1421   \exp:w \exp_end_continue_f:w \__fp_parse:n
1422 }
```

```

1423 \cs_new:Npn \__fp_to_dim_dispatch:w #1#2 \__fp_sep:
1424 {
1425     \__fp_change_func_type:NNN #1 \__fp_to_dim:w \__fp_to_dim_recover:w
1426     #1 #2 \__fp_sep:
1427 }
1428 \cs_new:Npn \__fp_to_dim_recover:w #1
1429 { \__fp_invalid_operation:nnw { Opt } { fp_to_dim } }
1430 \cs_new:Npn \__fp_to_dim:w #1 \__fp_sep: { \__fp_to_decimal:w #1 \__fp_sep: pt }
1431 \cs_new:Npn \fp_to_int:N #1 { \exp_after:wN \__fp_to_int_dispatch:w #1 }
1432 \cs_generate_variant:Nn \fp_to_int:N { c }
1433 \cs_new:Npn \fp_to_int:n
1434 {
1435     \exp_after:wN \__fp_to_int_dispatch:w
1436     \exp:w \exp_end_continue_f:w \__fp_parse:n
1437 }
1438 \cs_new:Npn \__fp_to_int_dispatch:w #1#2 \__fp_sep:
1439 {
1440     \__fp_change_func_type:NNN #1 \__fp_to_int:w \__fp_to_int_recover:w
1441     #1 #2 \__fp_sep:
1442 }
1443 \cs_new:Npn \__fp_to_int_recover:w #1
1444 { \__fp_invalid_operation:nnw { 0 } { fp_to_int } }
1445 \cs_new:Npn \__fp_to_int:w #1\__fp_sep:
1446 {
1447     \exp_after:wN \__fp_to_decimal:w \exp:w \exp_end_continue_f:w
1448     \__fp_round:Nwn \__fp_round_to_nearest:NNN #1\__fp_sep: { 0 }
1449 }
1450 \cs_new:Npn \dim_to_fp:n #1
1451 {
1452     \exp_after:wN \__fp_from_dim_test:ww
1453     \exp_after:wN 0
1454     \exp_after:wN ,
1455     \int_value:w \tex_glueexpr:D #1 \__fp_sep:
1456 }
1457 \cs_new:Npn \__fp_from_dim_test:ww #1, #2
1458 {
1459     \if_meaning:w 0 #2
1460     \__fp_case_return:nw { \exp_after:wN \c_zero_fp }
1461     \else:
1462     \exp_after:wN \__fp_from_dim:wNw
1463     \int_value:w \__fp_int_eval:w #1 - 4
1464     \if_meaning:w - #2
1465     \exp_after:wN , \exp_after:wN 2 \int_value:w
1466     \else:
1467     \exp_after:wN , \exp_after:wN 0 \int_value:w #2
1468     \fi:

```

```
1469 \fi:
1470 }
1471 \cs_new:Npn \__fp_from_dim:wNw #1,#2#3\__fp_sep:
1472 {
1473   \__fp_pack_twice_four:wNNNNNNNN \__fp_from_dim:wNNnnnnnn \__fp_sep:
1474   #3 000 0000 00 {10}987654321\__fp_sep: #2 {#1}
1475 }
1476 \cs_new:Npn \__fp_from_dim:wNNnnnnnn #1\__fp_sep: #2#3#4#5#6#7#8#9
1477 { \__fp_from_dim:wnnnnwNn #1 {#2#300} {0000} \__fp_sep: }
1478 \cs_new:Npn \__fp_from_dim:wnnnnwNn #1\__fp_sep: #2#3#4#5#6\__fp_sep: #7#8
1479 {
1480   \__fp_mul_npos_o:Nww #7
1481   \s__fp \__fp_chk:w 1 #7 {#5} #1 \__fp_sep:
1482   \s__fp \__fp_chk:w 1 0 {#8} {1525} {8789} {0625} {0000} \__fp_sep:
1483   \prg_do_nothing:
1484 }
1485 \cs_new_eq:NN \fp_use:N \fp_to_decimal:N
1486 \cs_generate_variant:Nn \fp_use:N { c }
1487 \cs_new_eq:NN \fp_eval:n \fp_to_decimal:n
1488 \cs_new:Npn \fp_sign:n #1
1489 { \fp_to_decimal:n { sign \__fp_parse:n {#1} } }
1490 \cs_new:Npn \fp_abs:n #1
1491 { \fp_to_decimal:n { abs \__fp_parse:n {#1} } }
1492 \cs_new:Npn \fp_max:nn #1#2
1493 { \fp_to_decimal:n { max ( \__fp_parse:n {#1} , \__fp_parse:n {#2} ) } }
1494 \cs_new:Npn \fp_min:nn #1#2
1495 { \fp_to_decimal:n { min ( \__fp_parse:n {#1} , \__fp_parse:n {#2} ) } }
1496 \cs_new:Npn \__fp_array_to_clist:n #1
1497 {
1498   \tl_if_empty:nF {#1}
1499   {
1500     \exp_last_unbraced:Ne \use_ii:nn
1501     {
1502       \__fp_array_to_clist_loop:Nw #1 { ? \prg_break: } \__fp_sep:
1503       \prg_break_point:
1504     }
1505   }
1506 }
1507 \cs_new:Npn \__fp_array_to_clist_loop:Nw #1#2\__fp_sep:
1508 {
1509   \use_none:n #1
1510   , ~
1511   \exp_not:f { \__fp_to_tl_dispatch:w #1 #2 \__fp_sep: }
1512   \__fp_array_to_clist_loop:Nw
1513 }
1514 %% File: l3fp-random.dtx
```



```

1515 \cs_new:Npn \__fp_parse_word_rand:N
1516 { \__fp_parse_function:NNN \__fp_rand_o:Nw ? }
1517 \cs_new:Npn \__fp_parse_word_randint:N
1518 { \__fp_parse_function:NNN \__fp_randint_o:Nw ? }
1519 \int_const:Nn \c__kernel_randint_max_int { 131071 }
1520 \cs_new:Npn \__kernel_randint:n #1
1521 {
1522   (#1 * \tex_uniformdeviate:D 16384
1523     + \tex_uniformdeviate:D #1 + 8192 ) / 16384
1524 }
1525 \cs_new:Npn \__fp_rand_myriads:n #1
1526 { \__fp_rand_myriads_loop:w #1 \prg_break: X \prg_break_point: \__fp_sep: }
1527 \cs_new:Npn \__fp_rand_myriads_loop:w #1 X
1528 {
1529   #1
1530   \exp_after:wN \__fp_rand_myriads_get:w
1531   \int_value:w \__fp_int_eval:w 9999 +
1532     \__kernel_randint:n { 10000 }
1533   \__fp_rand_myriads_loop:w
1534 }
1535 \cs_new:Npn \__fp_rand_myriads_get:w 1 #1 \__fp_sep: { \__fp_sep: {#1} }
1536 \cs_new:Npn \__fp_rand_o:Nw ? #1 @
1537 {
1538   \tl_if_empty:nTF {#1}
1539   {
1540     \exp_after:wN \__fp_rand_o:w
1541     \exp:w \exp_end_continue_f:w
1542     \__fp_rand_myriads:n { XXXX } { 0000 } { 0000 } \__fp_sep: 0
1543   }
1544   {
1545     \__fp_error_num_args:ffff { rand } { 0 } { 0 }
1546     { \__fp_array_count:n {#1} }
1547     \exp_after:wN \c_nan_fp
1548   }
1549 }
1550 \cs_new:Npn \__fp_rand_o:w \__fp_sep:
1551 {
1552   \exp_after:wN \__fp_sanitize:Nw
1553   \exp_after:wN 0
1554   \int_value:w \__fp_int_eval:w \c_zero_int
1555   \__fp_fixed_to_float_o:wN
1556 }
1557 \cs_new:Npn \__fp_randint_o:Nw ?
1558 {
1559   \__fp_parse_function_one_two:nnw
1560   { randint }

```

```
1561 { \__fp_randint_default:w \__fp_randint_o:w }
1562 }
1563 \cs_new:Npn \__fp_randint_default:w #1 { \exp_after:wN #1 \c_one_fp }
1564 \cs_new:Npn \__fp_randint_badarg:w \s__fp \__fp_chk:w #1#2#3\__fp_sep:
1565 {
1566   \__fp_int:wTF \s__fp \__fp_chk:w #1#2#3\__fp_sep:
1567   {
1568     \if_meaning:w 1 #1
1569     \if_int_compare:w
1570       \__fp_use_i_until_s:nw #3 \__fp_sep: > \c__fp_prec_int
1571       \c_one_int
1572       \fi:
1573       \fi:
1574     }
1575     { \c_one_int }
1576   }
1577 \cs_new:Npn \__fp_randint_o:w #1\__fp_sep: #2\__fp_sep: @
1578 {
1579   \if_case:w
1580     \__fp_randint_badarg:w #1\__fp_sep:
1581     \__fp_randint_badarg:w #2\__fp_sep:
1582     \if:w 1 \__fp_compare_back:ww #2\__fp_sep: #1\__fp_sep: \c_one_int \fi:
1583     \c_zero_int
1584     \__fp_randint_auxi_o:ww #1\__fp_sep: #2\__fp_sep:
1585   \or:
1586     \__fp_invalid_operation_tl_o:ff
1587     { randint } { \__fp_array_to_clist:n { #1\__fp_sep: #2\__fp_sep: } }
1588     \exp:w
1589     \fi:
1590     \exp_after:wN \exp_end:
1591   }
1592 \cs_new:Npn \__fp_randint_auxi_o:ww #1 \__fp_sep: #2 \__fp_sep: #3 \exp_end:
1593 {
1594   \fi:
1595   \__fp_randint_auxii:wn #2 \__fp_sep:
1596   { \__fp_randint_auxii:wn #1 \__fp_sep: \__fp_randint_auxiii_o:ww }
1597 }
1598 \cs_new:Npn \__fp_randint_auxii:wn \s__fp \__fp_chk:w #1#2#3#4 \__fp_sep:
1599 {
1600   \if_meaning:w 0 #1
1601     \exp_after:wN \use_i:nn
1602   \else:
1603     \exp_after:wN \use_ii:nn
1604   \fi:
1605   { \exp_after:wN \__fp_fixed_continue:wn \c__fp_one_fixed_tl }
1606   {
```

1607	\exp_after:wN __fp_ep_to_fixed:wwn	2160
1608	\int_value:w __fp_int_eval:w	2160
1609	#3 - \c__fp_prec_int , #4 {0000} {0000} __fp_sep:	2160
1610	{	2161
1611	\if_meaning:w 0 #2	2161
1612	\exp_after:wN \use_i:nnnn	2161
1613	\exp_after:wN __fp_fixed_add_one:wN	2161
1614	\fi:	2161
1615	\exp_after:wN __fp_fixed_sub:wwn \c__fp_one_fixed_tl	2161
1616	}	2161
1617	__fp_fixed_continue:wn	2161
1618	}	2161
1619	}	2161
1620	\cs_new:Npn __fp_randint_auxiii_o:ww #1 __fp_sep: #2 __fp_sep:	2162
1621	{	2162
1622	__fp_fixed_add:wwn #2 __fp_sep:	2162
1623	{0000} {0000} {0000} {0001} {0000} {0000} __fp_sep:	2162
1624	__fp_fixed_sub:wwn #1 __fp_sep:	2162
1625	{	2162
1626	\exp_after:wN \use_i:nn	2162
1627	\exp_after:wN __fp_fixed_mul_add:wwn	2162
1628	\exp:w \exp_end_continue_f:w __fp_rand_myriads:n { XXXXXX } __fp_sep:	2162
1629	}	2162
1630	#1 __fp_sep:	2163
1631	__fp_randint_auxiv_o:ww	2163
1632	#2 __fp_sep:	2163
1633	__fp_randint_auxv_o:w #1 __fp_sep: @	2163
1634	}	2163
1635	\cs_new:Npn __fp_randint_auxiv_o:ww #1#2#3#4#5 __fp_sep: #6#7#8#9	2163
1636	{	2163
1637	\if_int_compare:w	2163
1638	\if_int_compare:w #1#2 > #6#7 \exp_stop_f: 1 \else:	2163
1639	\if_int_compare:w #1#2 < #6#7 \exp_stop_f: - \fi: \fi:	2163
1640	#3#4 > #8#9 \exp_stop_f:	2164
1641	__fp_use_i_until_s:nw	2164
1642	\fi:	2164
1643	__fp_randint_auxv_o:w {#1}{#2}{#3}{#4}#5	2164
1644	}	2164
1645	\cs_new:Npn __fp_randint_auxv_o:w #1#2#3#4#5 __fp_sep: #6 @	2164
1646	{	2164
1647	\exp_after:wN __fp_sanitize:Nw	2164
1648	\int_value:w	2164
1649	\if_int_compare:w #1 < 10000 \exp_stop_f:	2164
1650	2	2165
1651	\else:	2165
1652	0	2165

```
1653     \exp_after:wN \exp_after:wN
1654     \exp_after:wN \__fp_reverse_args:Nww
1655 \fi:
1656 \exp_after:wN \__fp_fixed_sub:wwn \c__fp_one_fixed_tl
1657 {#1} {#2} {#3} {#4} {0000} {0000} \__fp_sep:
1658 {
1659     \exp_after:wN \exp_stop_f:
1660     \int_value:w \__fp_int_eval:w \c__fp_prec_int
1661     \__fp_fixed_to_float_o:wN
1662 }
1663 0
1664 \exp:w \exp_after:wN \exp_end:
1665 }
1666 \cs_new:Npn \int_rand:nn #1#2
1667 {
1668     \int_eval:n
1669     {
1670         \exp_after:wN \__fp_randint:ww
1671         \int_value:w \int_eval:n {#1} \exp_after:wN \__fp_sep:
1672         \int_value:w \int_eval:n {#2} \__fp_sep:
1673     }
1674 }
1675 \cs_new:Npn \__fp_randint:ww #1\__fp_sep: #2\__fp_sep:
1676 {
1677     \if_int_compare:w #1 > #2 \exp_stop_f:
1678     \msg_expandable_error:nnnn
1679     { kernel } { randint-backward-range } {#1} {#2}
1680     \__fp_randint:ww #2\__fp_sep: #1\__fp_sep:
1681 \else:
1682     \if_int_compare:w \__fp_int_eval:w #2
1683     \if_int_compare:w #1 > \c_zero_int
1684     - #1 < \__fp_int_eval:w
1685     \else:
1686     < \__fp_int_eval:w #1 +
1687     \fi:
1688     \c__kernel_randint_max_int
1689     \__fp_int_eval_end:
1690     \__kernel_randint:n
1691     { \__fp_int_eval:w #2 - #1 + 1 \__fp_int_eval_end: }
1692     - 1 + #1
1693 \else:
1694     \__kernel_randint:nn {#1} {#2}
1695 \fi:
1696 \fi:
1697 }
1698 \cs_new:Npn \__kernel_randint:nn #1#2
```

```

1699 {
1700 #1
1701 \exp_after:wN \__fp_randint_wide_aux:w
1702 \int_value:w
1703 \exp_after:wN \__fp_randint_split_o:Nw
1704 \tex_uniformdeviate:D 268435456 \__fp_sep:
1705 \int_value:w
1706 \exp_after:wN \__fp_randint_split_o:Nw
1707 \tex_uniformdeviate:D 268435456 \__fp_sep:
1708 \int_value:w
1709 \exp_after:wN \__fp_randint_split_o:Nw
1710 \int_value:w \__fp_int_eval:w 131072 +
1711 \exp_after:wN \__fp_randint_split_o:Nw
1712 \int_value:w
1713 \__kernel_int_add:nnn {#2} { -#1 } { -\c_max_int } \__fp_sep:
1714 .
1715 }
1716 \cs_new:Npn \__fp_randint_split_o:Nw #1#2 \__fp_sep:
1717 {
1718 \if_meaning:w 0 #1
1719 0 \exp_after:wN \__fp_sep: \int_value:w 0
1720 \else:
1721 \exp_after:wN \__fp_randint_split_aux:w
1722 \int_value:w \__fp_int_eval:w (#1#2 - 8192) / 16384 \__fp_sep:
1723 + #1#2
1724 \fi:
1725 \exp_after:wN \__fp_sep:
1726 }
1727 \cs_new:Npn \__fp_randint_split_aux:w #1 \__fp_sep:
1728 {
1729 #1 \exp_after:wN \__fp_sep:
1730 \int_value:w \__fp_int_eval:w - #1 * 16384
1731 }
1732 \cs_new:Npn \__fp_randint_wide_aux:w
1733 #1\__fp_sep:#2\__fp_sep: #3\__fp_sep:#4\__fp_sep:
1734 #5\__fp_sep:#6\__fp_sep:#7\__fp_sep: .
1735 {
1736 \exp_after:wN \__fp_randint_wide_auxii:w
1737 \int_value:w \__fp_int_eval:w #5 * #3 + #6 * #1 +
1738 (#5 * #4 + #6 * #3 + #7 * #1 +
1739 (#5 * #2 + #7 * #3 +
1740 (16384 * #6 + #7) * (16384 * #4 + #2) / 268435456) / 16384
1741 ) / 16384 \exp_after:wN \__fp_sep:
1742 \int_value:w \__fp_int_eval:w (#5 + #6) * 16384 + #7 \__fp_sep:
1743 #1 \__fp_sep: #5 \__fp_sep:
1744 }

```

```

1745 \cs_new:Npn \__fp_randint_wide_auxii:w
1746 #1\__fp_sep: #2\__fp_sep: #3\__fp_sep: #4\__fp_sep:
1747 {
1748 \if_int_odd:w 0
1749 \if_int_compare:w #1 = #2 \else: \exp_stop_f: \fi:
1750 \if_int_compare:w #4 = \c_zero_int 1 \fi:
1751 \if_int_compare:w #3 = 16383 ~ 1 \fi:
1752 \exp_stop_f:
1753 \exp_after:wN \prg_break:
1754 \fi:
1755 \if_int_compare:w #4 < 8 \exp_stop_f:
1756 + #4 * #3 * 16384
1757 \else:
1758 + 8 * #3 * 16384 + (#4 - 8) * #3 * 16384
1759 \fi:
1760 + #1
1761 \prg_break_point:
1762 }
1763 \cs_new:Npn \int_rand:n #1
1764 {
1765 \int_eval:n
1766 { \exp_args:Nf \__fp_randint:n { \int_eval:n {#1} } }
1767 }
1768 \cs_new:Npn \__fp_randint:n #1
1769 {
1770 \if_int_compare:w #1 < \c_one_int
1771 \msg_expandable_error:nnnn
1772 { kernel } { randint-backward-range } { 1 } {#1}
1773 \__fp_randint:ww #1\__fp_sep: 1\__fp_sep:
1774 \else:
1775 \if_int_compare:w #1 > \c__kernel_randint_max_int
1776 \__kernel_randint:nn { 1 } {#1}
1777 \else:
1778 \__kernel_randint:n {#1}
1779 \fi:
1780 \fi:
1781 }
1782 %% File l3fp-types.dtx (C) Copyright 2012-2015,2017,2018,2020,2021,2023 The LaTeX Project
1783 \cs_new:Npe \__fp_types_cs_to_op:N #1
1784 {
1785 \exp_not:N \exp_after:wN \exp_not:N \__fp_types_cs_to_op_auxi:wwwn
1786 \exp_not:N \token_to_str:N #1 \s__fp_mark
1787 \exp_not:N \__fp_use_i_delimit_by_s_stop:nw
1788 \tl_to_str:n { __fp_o:w } \s__fp_mark
1789 { \exp_not:N \__fp_use_i_delimit_by_s_stop:nw ? }
1790 \s__fp_stop

```

```
1791 }
1792 \use:e
1793 {
1794   \cs_new:Npn \exp_not:N \__fp_types_cs_to_op_auxi:wwwn
1795     #1 \tl_to_str:n { __fp_ } #2
1796     \tl_to_str:n { _o:w } #3 \s__fp_mark #4 { #4 {#2} }
1797 }
1798 \cs_new:Npn \__fp_types_unary:NNw #1
1799 {
1800   \exp_args:Nf \__fp_types_unary_auxi:nNw
1801     { \__fp_types_cs_to_op:N #1 }
1802 }
1803 \cs_new:Npn \__fp_types_unary_auxi:nNw #1#2#3
1804 {
1805   \exp_after:wN \__fp_types_unary_auxii:NnNw
1806   \cs:w __fp_#1 \__fp_type_from_scan:N #3 _o:w \cs_end:
1807   {#1}
1808   #2#3
1809 }
1810 \cs_new:Npn \__fp_types_unary_auxii:NnNw #1#2#3
1811 {
1812   \token_if_eq_meaning:NNTF \scan_stop: #1
1813     { \__fp_invalid_operation_o:nw {#2} }
1814     { #1 #3 }
1815 }
1816 \cs_new:Npn \__fp_types_binary:Nww #1
1817 {
1818   \exp_last_unbraced:Nf \__fp_types_binary_auxi:Nww
1819     { \__fp_types_cs_to_op:N #1 }
1820 }
1821 \cs_new:Npn \__fp_types_binary_auxi:Nww #1#2#3\__fp_sep: #4#5\__fp_sep: @
1822 {
1823   \exp_after:wN \__fp_types_binary_auxii:NNww
1824   \cs:w
1825     __fp
1826     \__fp_type_from_scan:N #2
1827     _#1
1828     \__fp_type_from_scan:N #4
1829     _o:ww
1830   \cs_end:
1831   #1 #2#3\__fp_sep: #4#5\__fp_sep:
1832 }
1833 \cs_new:Npn \__fp_types_binary_auxii:NNww #1#2
1834 {
1835   \token_if_eq_meaning:NNTF \scan_stop: #1
1836     { \__fp_invalid_operation_o:Nww #2 }
```



```
1837     {#1}
1838   }
1839   %% File l3fp-symbolic.dtx (C) Copyright 2012-2025 The LaTeX Project
1840   \fp_new:N \l__fp_symbolic_fp
1841   \scan_new:N \s__fp_symbolic
1842   \cs_new_protected:Npn \__fp_symbolic_chk:w #1,#2#3\__fp_sep:
1843   {
1844     \msg_error:nne { fp } { misused-fp }
1845     {
1846       \__fp_to_tl_dispatch:w
1847       \s__fp_symbolic \__fp_symbolic_chk:w #1,{#2}\__fp_sep:
1848     }
1849   }
1850   \cs_new:Npn \__fp_if_has_symbolic:nTF #1
1851   {
1852     \__fp_if_has_symbolic_aux:w
1853     #1 \s__fp_mark \use_i:nn
1854     \s__fp_symbolic \s__fp_mark \use_ii:nn
1855     \s__fp_stop
1856   }
1857   \cs_new:Npn \__fp_if_has_symbolic_aux:w
1858   #1 \s__fp_symbolic #2 \s__fp_mark #3#4 \s__fp_stop { #3 }
1859   \cs_new:Npn \__fp_exp_after_symbolic_f:nw
1860   #1 \s__fp_symbolic \__fp_symbolic_chk:w #2, #3#4\__fp_sep:
1861   {
1862     \exp_after:wN \__fp_exp_after_symbolic_aux:w
1863     \exp:w
1864     \__fp_exp_after_symbolic_loop:N #2
1865     { , \exp:w \use_none:nn }
1866     \exp_after:wN \exp_end: \exp_after:wN
1867     {
1868       \exp:w \exp_end_continue_f:w
1869       \__fp_exp_after_array_f:w #3 \s__fp_expr_stop
1870       \exp_after:wN
1871     }
1872     \exp_after:wN \__fp_sep:
1873     \exp:w \exp_end_continue_f:w #1
1874   }
1875   \cs_new:Npn \__fp_exp_after_symbolic_aux:w #1, #2\__fp_sep:
1876   {
1877     \__fp_if_has_symbolic:nTF {#2}
1878     { \s__fp_symbolic \__fp_symbolic_chk:w #1, {#2} \__fp_sep: }
1879     { #1 #2 @ \prg_do_nothing: }
1880   }
1881   \cs_new:Npn \__fp_exp_after_symbolic_loop:N #1
1882   {
```

```
1883 \exp_after:wN \exp_end: 21883
1884 \exp_after:wN #1 21884
1885 \exp:w 21885
1886 \__fp_exp_after_symbolic_loop:N 21886
1887 } 21887
1888 \cs_new:Npn \__fp_symbolic_binary_o:Nww #1 #2\__fp_sep: #3\__fp_sep: 21888
1889 { 21889
1890 \__fp_exp_after_symbolic_f:nw { \exp_after:wN \exp_stop_f: } 21890
1891 \s__fp_symbolic \__fp_symbolic_chk:w 21891
1892 \__fp_types_binary:Nww #1 , { #2\__fp_sep: #3\__fp_sep: } \__fp_sep: 21892
1893 } 21893
1894 \cs_set_protected:Npn \__fp_tmp:w #1#2 21894
1895 { 21895
1896 \cs_new:cpn 21896
1897 { __fp_symbolic_#2_symbolic_o:ww } 21897
1898 { \__fp_symbolic_binary_o:Nww #1 } 21898
1899 \cs_new_eq:cc 21899
1900 { __fp_symbolic_#2 _o:ww } 21900
1901 { __fp_symbolic_#2_symbolic_o:ww } 21901
1902 \cs_new_eq:cc 21902
1903 { __fp _#2_symbolic_o:ww } 21903
1904 { __fp_symbolic_#2_symbolic_o:ww } 21904
1905 } 21905
1906 \tl_map_inline:nn { + - * / ^ & | } % | 21906
1907 { \exp_args:Nc \__fp_tmp:w { __fp_#1_o:ww } {#1} } 21907
1908 \cs_new:Npn \__fp_symbolic_unary_o:NNw #1#2#3\__fp_sep: @ 21908
1909 { 21909
1910 \__fp_exp_after_symbolic_f:nw { \exp_after:wN \exp_stop_f: } 21910
1911 \s__fp_symbolic \__fp_symbolic_chk:w 21911
1912 \__fp_types_unary:NNw #1#2 , { #3\__fp_sep: } \__fp_sep: 21912
1913 } 21913
1914 \tl_map_inline:nn 21914
1915 { 21915
1916 {acos} {acsc} {asec} {asin} {cos} {cot} {csc} {exp} {fact} {ln} 21916
1917 {not} {sec} {set_sign} {sin} {sign} {sqrt} {tan} 21917
1918 } 21918
1919 { 21919
1920 \cs_new:cpe { __fp_symbolic_#1_o:w } 21920
1921 { 21921
1922 \exp_not:N \__fp_symbolic_unary_o:NNw 21922
1923 \exp_not:c { __fp_#1_o:w } 21923
1924 } 21924
1925 } 21925
1926 \cs_set_protected:Npn \__fp_tmp:w #1#2#3 21926
1927 { 21927
1928 \cs_new:cpn { __fp_symbolic_to_#1:w } 21928
```

```

1929     {
1930         \exp_after:wN \__fp_symbolic_convert:wnnN
1931         \exp:w \exp_end_continue_f:w
1932         \__fp_exp_after_symbolic_f:nw { { #2 } { fp_to_#1 } #3 }
1933     }
1934 }
1935 \__fp_tmp:w { decimal } { 0 } \__fp_to_decimal_dispatch:w
1936 \__fp_tmp:w { int } { 0 } \__fp_to_int_dispatch:w
1937 \__fp_tmp:w { scientific } { nan } \__fp_to_scientific_dispatch:w
1938 \cs_new:Npn \__fp_symbolic_convert:wnnN #1#2\__fp_sep: #3#4#5
1939 {
1940     \str_if_eq:nnTF {#1} { \s__fp_symbolic }
1941     { \__fp_invalid_operation:nnw {#3} {#4} #1#2\__fp_sep: }
1942     { #5 #1#2\__fp_sep: }
1943 }
1944 \cs_new:Npn \__fp_symbolic_cs_arg_to_fn:NN #1
1945 {
1946     \exp_args:Nf \__fp_symbolic_op_arg_to_fn:nN
1947     { \__fp_types_cs_to_op:N #1 }
1948 }
1949 \cs_new:Npn \__fp_symbolic_op_arg_to_fn:nN #1#2
1950 {
1951     \str_case:nnF { #1 #2 }
1952     {
1953         { not ? } { ! }
1954         { set_sign 0 } { abs }
1955         { set_sign 2 } { - }
1956     }
1957     {
1958         \token_if_eq_meaning:NNTF #2 \use_ii:nn
1959         { #1 d } {#1}
1960     }
1961 }
1962 \cs_new:Npn \__fp_symbolic_to_tl:w
1963     \s__fp_symbolic \__fp_symbolic_chk:w #1#2, #3#4\__fp_sep:
1964 {
1965     \str_case:nnTF {#1}
1966     {
1967         { \__fp_types_unary:NNw } { \__fp_symbolic_unary_to_tl:NNw }
1968         { \__fp_types_binary:Nww } { \__fp_symbolic_binary_to_tl:Nww }
1969         { \__fp_function_o:w } { \__fp_symbolic_function_to_tl:Nw }
1970     }
1971     { #2, #3 @ }
1972     { \tl_to_str:n {#2} }
1973 }
1974 \cs_new:Npn \__fp_symbolic_unary_to_tl:NNw #1#2 , #3 @

```

```
1975 {
1976     \use:e
1977     {
1978         \__fp_symbolic_cs_arg_to_fn:NN #1#2
1979         ( \__fp_to_tl_dispatch:w #3 )
1980     }
1981 }
1982 \cs_new:Npn \__fp_symbolic_binary_to_tl:Nww #1, #2\__fp_sep: #3\__fp_sep: @
1983 {
1984     \use:e
1985     {
1986         ( \__fp_to_tl_dispatch:w #2\__fp_sep: )
1987         \__fp_types_cs_to_op:N #1
1988         ( \__fp_to_tl_dispatch:w #3\__fp_sep: )
1989     }
1990 }
1991 \cs_new:Npn \__fp_symbolic_function_to_tl:Nw #1, #2@
1992 {
1993     \use:e
1994     {
1995         \__fp_types_cs_to_op:N #1
1996         ( \__fp_array_to_clist:n {#2} )
1997     }
1998 }
1999 \prg_new_protected_conditional:Npnn
2000     \__fp_id_if_invalid:n #1 { T , F , TF }
2001 {
2002     \tl_if_empty:nTF {#1}
2003     { \prg_return_true: }
2004     {
2005         \tl_if_in:nnTF { #1 } { ~ }
2006         { \prg_return_true: }
2007         {
2008             \__fp_id_if_invalid_aux:N #1
2009             { ? \prg_break:n \prg_return_false: }
2010             \prg_break_point:
2011         }
2012     }
2013 }
2014 \cs_new:Npn \__fp_id_if_invalid_aux:N #1
2015 {
2016     \use_none:n #1
2017     \int_compare:nF { `a <= `#1 <= `z }
2018     {
2019         \int_compare:nF { `A <= `#1 <= `Z }
2020         { \prg_break:n \prg_return_true: }
```

```
2021     }
2022     \__fp_id_if_invalid_aux:N
2023 }
2024 \cs_new:Npn \__fp_variable_o:w #1 @ #2
2025 {
2026     \fp_if_exist:cTF { l__fp_variable_#1_fp }
2027     {
2028         \exp_last_unbraced:Nf \__fp_exp_after_array_f:w
2029         { \use:c { l__fp_variable_#1_fp } } \s__fp_expr_stop
2030         \exp_after:wN \exp_stop_f: #2
2031     }
2032     {
2033         \token_if_eq_meaning:NNTF #2 \prg_do_nothing:
2034         {
2035             \s__fp_symbolic \__fp_symbolic_chk:w
2036             \__fp_variable_o:w #1 , { } \__fp_sep:
2037         }
2038         {
2039             \exp_after:wN \s__fp_symbolic
2040             \exp_after:wN \__fp_symbolic_chk:w
2041             \exp_after:wN \__fp_variable_o:w
2042             \exp:w
2043             \__fp_exp_after_symbolic_loop:N #1
2044             { , \exp:w \use_none:nn }
2045             \exp_after:wN \exp_end:
2046             \exp_after:wN { \exp_after:wN } \exp_after:wN \__fp_sep:
2047             #2
2048         }
2049     }
2050 }
2051 \cs_new_protected:Npn \__fp_variable_set_parsing:Nn #1#2
2052 {
2053     \cs_set:Npn \__fp_tmp:w
2054     {
2055         \__fp_exp_after_symbolic_f:nw { \__fp_parse_infix:NN }
2056         \s__fp_symbolic \__fp_symbolic_chk:w
2057         \__fp_variable_o:w #2 , { } \__fp_sep:
2058     }
2059     \exp_args:NNc \__fp_variable_set_parsing_aux:NNn #1
2060     { __fp_parse_word_#2:N } {#2}
2061 }
2062 \cs_new_protected:Npn \__fp_variable_set_parsing_aux:NNn #1#2#3
2063 {
2064     \cs_if_eq:NNF #2 \__fp_tmp:w
2065     {
2066         \cs_if_exist:NTF #2
```

```
2067 {
2068     \msg_warning:nnnn
2069     { fp } { id-used-elsewhere } {#3} { variable }
2070     #1 #2 \__fp_tmp:w
2071 }
2072 {
2073     \cs_new_eq:NN #2 \scan_stop: % to declare the function
2074     #1 #2 \__fp_tmp:w
2075 }
2076 }
2077 }
2078 \cs_new_protected:Npn \fp_clear_variable:n #1
2079 {
2080     \exp_args:No \__fp_clear_variable:n { \tl_to_str:n {#1} }
2081 }
2082 \cs_new_protected:Npn \__fp_clear_variable:n #1
2083 {
2084     \__fp_id_if_invalid:nTF {#1}
2085     { \msg_error:nnn { fp } { id-invalid } {#1} }
2086     { \__fp_clear_variable_aux:n {#1} }
2087 }
2088 \cs_new_protected:Npn \__fp_clear_variable_aux:n #1
2089 {
2090     \cs_set_eq:cN { l__fp_variable_#1_fp } \tex_undefined:D
2091     \__fp_variable_set_parsing:Nn \cs_set_eq:NN {#1}
2092 }
2093 \cs_new_protected:Npn \fp_new_variable:n #1
2094 {
2095     \exp_args:No \__fp_new_variable:n { \tl_to_str:n {#1} }
2096 }
2097 \cs_new_protected:Npn \__fp_new_variable:n #1
2098 {
2099     \__fp_id_if_invalid:nTF {#1}
2100     { \msg_error:nnn { fp } { id-invalid } {#1} }
2101     {
2102         \cs_if_exist:cT { __fp_parse_word_#1:N }
2103         {
2104             \msg_error:nnn
2105             { fp } { id-already-defined } {#1}
2106             \cs_undefine:c { __fp_parse_word_#1:N }
2107             \cs_set_eq:cN { l__fp_variable_#1_fp } \tex_undefined:D
2108         }
2109         \__fp_variable_set_parsing:Nn \cs_gset_eq:NN {#1}
2110     }
2111 }
2112 \flag_new:N \l__fp_symbolic_flag
```

```

2113 \cs_new_protected:Npn \fp_set_variable:nn #1
2114 {
2115     \exp_args:No \__fp_set_variable:nn { \tl_to_str:n {#1} }
2116 }
2117 \cs_new_protected:Npn \__fp_set_variable:nn #1#2
2118 {
2119     \__fp_id_if_invalid:nTF {#1}
2120     { \msg_error:nnn { fp } { id-invalid } {#1} }
2121     {
2122         \cs_if_exist:cF { __fp_parse_word_#1:N }
2123         {
2124             \msg_error:nnn {fp} { id-undefined } {#1}
2125             \__fp_variable_set_parsing:Nn \cs_set_eq:NN {#1}
2126         }
2127         \fp_set:Nn \l__fp_symbolic_fp {#2}
2128         \cs_set_nopar:cpn { l__fp_variable_#1_fp }
2129         { \flag_ensure_raised:N \l__fp_symbolic_flag \c_nan_fp }
2130         \flag_clear:N \l__fp_symbolic_flag
2131         \fp_set:cn { l__fp_variable_#1_fp } { \l__fp_symbolic_fp }
2132         \flag_if_raised:NT \l__fp_symbolic_flag
2133         {
2134             \msg_error:nneee { fp } { id-loop }
2135             { #1 }
2136             { \tl_to_str:n {#2} }
2137             { \fp_to_tl:N \l__fp_symbolic_fp }
2138         }
2139     }
2140 }
2141 \msg_new:nnnn { fp } { id-invalid }
2142 { Floating-point~identifier~'#1'~invalid. }
2143 {
2144     LaTeX~has~been~asked~to~create~a~new~floating-point~identifier~'#1'~
2145     but~this~may~only~contain~ASCII~letters.
2146 }
2147 \msg_new:nnnn { fp } { id-already-defined }
2148 { Floating-point~identifier~'#1'~already~defined. }
2149 {
2150     LaTeX~has~been~asked~to~create~a~new~floating-point~identifier~'#1'~
2151     but~this~name~has~already~been~used~elsewhere.
2152 }
2153 \msg_new:nnnn { fp } { id-undefined }
2154 { Floating~point~identifier~'#1'~is~undefined. }
2155 {
2156     LaTeX~has~been~asked~to~set~a~floating~point~identifier~'#1'~
2157     but~this~name~has~not~been~declared.
2158 }

```



```
2159 \msg_new:nnnn { fp } { id-used-elsewhere }
2160 { Floating-point~identifier~'#1'~already~used~for~something~else. }
2161 {
2162   LaTeX~has~been~asked~to~create~a~new~floating~point~identifier~'#1'~
2163   but~this~name~is~used,~and~is~not~a~user~defined~#2.
2164 }
2165 \msg_new:nnnn { fp } { id-loop }
2166 { Variable~'#1'~used~in~the~definition~of~'#1'. }
2167 {
2168   LaTeX~has~been~asked~to~set~the~floating~point~identifier~'#1'~
2169   to~the~expression~'#2'.~Evaluating~this~expression~yields~'#3',~
2170   which~contains~'#1'~itself.
2171 }
2172 %% File l3fp-functions.dtx (C) Copyright 2012-2018,2020,2021,2023 The LaTeX Project
2173 \cs_new_protected:Npn \fp_new_function:n #1
2174 { \exp_args:No \__fp_new_function:n { \tl_to_str:n {#1} } }
2175 \cs_new_protected:Npn \__fp_new_function:n #1
2176 {
2177   \__fp_id_if_invalid:nTF {#1}
2178   { \msg_error:nnn { fp } { id-invalid } {#1} }
2179   {
2180     \cs_if_exist:cT { __fp_parse_word_#1:N }
2181     {
2182       \msg_error:nnn
2183       { fp } { id-already-defined } {#1}
2184       \cs_undefine:c { __fp_parse_word_#1:N }
2185       \cs_undefine:c { __fp_#1_o:w }
2186     }
2187     \__fp_function_set_parsing:Nn \cs_gset_eq:NN {#1}
2188   }
2189 }
2190 \cs_new_protected:Npn \__fp_function_set_parsing:Nn #1#2
2191 {
2192   \exp_args:NNc \__fp_function_set_parsing_aux:NNn #1
2193   { __fp_parse_word_#2:N } {#2}
2194 }
2195 \cs_new_protected:Npn \__fp_function_set_parsing_aux:NNn #1#2#3
2196 {
2197   \cs_set:Npe \__fp_tmp:w
2198   {
2199     \exp_not:N \__fp_parse_function:NNN
2200     \exp_not:N \__fp_function_o:w
2201     \exp_not:c { __fp_#3_o:w }
2202   }
2203   \cs_if_eq:NNF #2 \__fp_tmp:w
2204   {
```

```
2205     \cs_if_exist:NTF #2
2206     {
2207         \msg_warning:nnnn
2208         { fp } { id-used-elsewhere } {#3} { function }
2209         #1 #2 \__fp_tmp:w
2210     }
2211     {
2212         \cs_new_eq:NN #2 \scan_stop: % to declare the function
2213         #1 #2 \__fp_tmp:w
2214     }
2215 }
2216 }
2217 \cs_new:Npn \__fp_function_o:w #1#2 @
2218 {
2219     \cs_if_exist:NTF #1
2220     { #1 #2 @ }
2221     {
2222         \exp_after:wN \s__fp_symbolic
2223         \exp_after:wN \__fp_symbolic_chk:w
2224         \exp_after:wN \__fp_function_o:w
2225         \exp_after:wN #1
2226         \exp_after:wN ,
2227         \exp_after:wN {
2228             \exp:w \exp_end_continue_f:w
2229             \__fp_exp_after_array_f:w #2 \s__fp_expr_stop
2230             \exp_after:wN
2231         }
2232         \exp_after:wN \__fp_sep:
2233     }
2234 }
2235 \int_new:N \l__fp_function_arg_int
2236 \cs_new_protected:Npn \fp_set_function:nnn #1
2237 {
2238     \exp_args:NNo \__fp_set_function:Nnnn \cs_set_eq:cN
2239     { \tl_to_str:n {#1} }
2240 }
2241 \cs_new_protected:Npn \__fp_set_function:Nnnn #1#2#3#4
2242 {
2243     \__fp_id_if_invalid:nTF {#2}
2244     { \msg_error:nnn { fp } { id-invalid } {#2} }
2245     {
2246         \cs_if_exist:cF { __fp_parse_word_#2:N }
2247         {
2248             \msg_error:nnn {fp} { id-undefined } {#2}
2249             \__fp_function_set_parsing:Nn \cs_set_eq:NN {#2}
2250         }
2251     }
```

```

2251 \group_begin:
2252 \int_zero:N \l__fp_function_arg_int
2253 \exp_args:No \clist_map_inline:nn { \tl_to_str:n {#3} }
2254 {
2255 \int_incr:N \l__fp_function_arg_int
2256 \exp_args:Ne \__fp_clear_variable_aux:n
2257 {
2258 \c_underscore_str \tex_romannumeral:D \l__fp_function_arg_int
2259 }
2260 \fp_clear_variable:n {##1}
2261 \cs_set_nopar:cpe { l__fp_variable_##1_fp }
2262 {
2263 \exp_not:N \s__fp_symbolic
2264 \exp_not:N \__fp_symbolic_chk:w
2265 \exp_not:N \__fp_function_arg_o:w
2266 \int_use:N \l__fp_function_arg_int
2267 #####1 , { } \__fp_sep:
2268 }
2269 }
2270 \cs_set:Npn \__fp_function_arg_o:w ##1 @
2271 {
2272 \exp_after:wN \s__fp_symbolic
2273 \exp_after:wN \__fp_symbolic_chk:w
2274 \exp_after:wN \__fp_function_arg_o:w
2275 \tex_romannumeral:D
2276 \__fp_exp_after_symbolic_loop:N ##1
2277 { , \tex_romannumeral:D \use_none:nn }
2278 \exp_after:wN \c_zero_int
2279 \exp_after:wN { \exp_after:wN } \exp_after:wN \__fp_sep:
2280 }
2281 \fp_set:Nn \l__fp_symbolic_fp {#4}
2282 \use:e
2283 {
2284 \exp_not:n { \cs_gset:Npn \__fp_tmp:w ##1 }
2285 { \exp_not:o { \l__fp_symbolic_fp } }
2286 }
2287 \use:e
2288 {
2289 \exp_not:n { \cs_gset:Npn \__fp_tmp:w ##1 @ }
2290 {
2291 \exp_not:N \__fp_exp_after_symbolic_f:nw
2292 \exp_not:n { { \exp_after:wN \exp_stop_f: } }
2293 \exp_not:o { \__fp_tmp:w { . , {##1} } }
2294 }
2295 }
2296 \group_end:

```

```

2297     #1 { __fp_#2_o:w } \__fp_tmp:w
2298 }
2299 }
2300 \cs_new:Npn \__fp_function_arg_o:w #1. #2
2301 {
2302     \if_meaning:w @ #2
2303     \exp_after:wN \__fp_function_arg_few:w
2304     \fi:
2305     \if_int_compare:w #1 = \c_one_int
2306     \exp_after:wN \__fp_function_arg_get:w
2307     \fi:
2308     \__fp_use_i_until_s:nw
2309     {
2310         \exp_after:wN \__fp_function_arg_o:w
2311         \int_value:w \int_eval:n { #1 - 1 } .
2312     }
2313     #2
2314 }
2315 \cs_new:Npn \__fp_function_arg_few:w #1 @ { \exp_after:wN \c_nan_fp }
2316 \cs_new:Npn \__fp_function_arg_get:w #1#2#3\__fp_sep: #4 @
2317 {
2318     \__fp_exp_after_array_f:w #3\__fp_sep: \s__fp_expr_stop
2319     \exp_after:wN \exp_stop_f:
2320 }
2321 \cs_new_protected:Npn \fp_clear_function:n #1
2322 { \exp_args:No \__fp_clear_function:n { \tl_to_str:n {#1} } }
2323 \cs_new_protected:Npn \__fp_clear_function:n #1
2324 {
2325     \__fp_id_if_invalid:nTF {#1}
2326     { \msg_error:nnn { fp } { id-invalid } {#1} }
2327     {
2328         \cs_set_eq:cN { __fp_#1_o:w } \tex_undefine:D
2329         \__fp_function_set_parsing:Nn \cs_set_eq:NN {#1}
2330     }
2331 }
2332 %% File: l3fparray.dtx
2333 \int_new:N \g__fp_array_int
2334 \int_new:N \l__fp_array_loop_int
2335 \cs_new_protected:Npn \fparray_new:Nn #1#2
2336 {
2337     \tl_new:N #1
2338     \prg_replicate:nn { 3 }
2339     {
2340         \int_gincr:N \g__fp_array_int
2341         \exp_args:NNc \tl_gput_right:Nn #1
2342         { g__fp_array_ \__fp_int_to_roman:w \g__fp_array_int _intarray }

```

2343	}	2234
2344	\exp_last_unbraced:Nfo __fp_array_new:nNNNN	2234
2345	{ \int_eval:n {#2} } #1 #1	2234
2346	}	2234
2347	\cs_generate_variant:Nn \fparray_new:Nn { c }	2234
2348	\cs_new_protected:Npn __fp_array_new:nNNNN #1#2#3#4#5	2234
2349	{	2234
2350	\int_compare:nNnTF {#1} < 0	2235
2351	{	2235
2352	\msg_error:nnn { kernel } { negative-array-size } {#1}	2235
2353	\cs_undefine:N #1	2235
2354	\int_gsub:Nn \g__fp_array_int { 3 }	2235
2355	}	2235
2356	{	2235
2357	\intarray_new:Nn #2 {#1}	2235
2358	\intarray_new:Nn #3 {#1}	2235
2359	\intarray_new:Nn #4 {#1}	2235
2360	}	2236
2361	}	2236
2362	\cs_new:Npn \fparray_count:N #1	2236
2363	{	2236
2364	\exp_after:wN \use_i:nnn	2236
2365	\exp_after:wN \intarray_count:N #1	2236
2366	}	2236
2367	\cs_generate_variant:Nn \fparray_count:N { c }	2236
2368	\cs_new:Npn __fp_array_bounds:NNnTF #1#2#3#4#5	2236
2369	{	2236
2370	\if_int_compare:w 1 > #3 \exp_stop_f:	2237
2371	__fp_array_bounds_error:NNn #1 #2 {#3}	2237
2372	#5	2237
2373	\else:	2237
2374	\if_int_compare:w #3 > \fparray_count:N #2 \exp_stop_f:	2237
2375	__fp_array_bounds_error:NNn #1 #2 {#3}	2237
2376	#5	2237
2377	\else:	2237
2378	#4	2237
2379	\fi:	2237
2380	\fi:	2238
2381	}	2238
2382	\cs_new:Npn __fp_array_bounds_error:NNn #1#2#3	2238
2383	{	2238
2384	#1 { kernel } { out-of-bounds }	2238
2385	{ \token_to_str:N #2 } {#3} { \fparray_count:N #2 }	2238
2386	}	2238
2387	\cs_new_protected:Npn \fparray_gset:Nnn #1#2#3	2238
2388	{	2238

```
2389 \exp_after:wN \exp_after:wN
2390 \exp_after:wN \__fp_array_gset:NNNNww
2391 \exp_after:wN #1
2392 \exp_after:wN #1
2393 \int_value:w \int_eval:n {#2} \exp_after:wN \__fp_sep:
2394 \exp:w \exp_end_continue_f:w \__fp_parse:n {#3}
2395 }
2396 \cs_generate_variant:Nn \fparray_gset:Nnn { c }
2397 \cs_new_protected:Npn \__fp_array_gset:NNNNww #1#2#3#4#5 \__fp_sep: #6 \__fp_sep:
2398 {
2399 \__fp_array_bounds:NNnTF \msg_error:nneee #4 {#5}
2400 {
2401 \exp_after:wN \__fp_change_func_type:NNN
2402 \__fp_use_i_until_s:nw #6 \__fp_sep:
2403 \__fp_array_gset:w
2404 \__fp_array_gset_recover:Nw
2405 #6 \__fp_sep: {#5} #1 #2 #3
2406 }
2407 { }
2408 }
2409 \cs_new_protected:Npn \__fp_array_gset_recover:Nw #1#2 \__fp_sep:
2410 {
2411 \__fp_error:nffn { unknown-type } { \tl_to_str:n { #2 \__fp_sep: } } { } { } { }
2412 \exp_after:wN #1 \c_nan_fp
2413 }
2414 \cs_new_protected:Npn \__fp_array_gset:w \s__fp \__fp_chk:w #1#2
2415 {
2416 \if_case:w #1 \exp_stop_f:
2417 \__fp_case_return:nw { \__fp_array_gset_special:nnNNN {#2} }
2418 \or: \exp_after:wN \__fp_array_gset_normal:w
2419 \or: \__fp_case_return:nw { \__fp_array_gset_special:nnNNN { #2 3 } }
2420 \or: \__fp_case_return:nw { \__fp_array_gset_special:nnNNN { 1 } }
2421 \fi:
2422 \s__fp \__fp_chk:w #1 #2
2423 }
2424 \cs_new_protected:Npn \__fp_array_gset_normal:w
2425 \s__fp \__fp_chk:w 1 #1 #2 #3#4#5 \__fp_sep: #6#7#8#9
2426 {
2427 \__kernel_intarray_gset:Nnn #7 {#6} {#2}
2428 \__kernel_intarray_gset:Nnn #8 {#6}
2429 { \if_meaning:w 2 #1 3 \else: 1 \fi: #3#4 }
2430 \__kernel_intarray_gset:Nnn #9 {#6} { 1 \use:nn #5 }
2431 }
2432 \cs_new_protected:Npn \__fp_array_gset_special:nnNNN #1#2#3#4#5
2433 {
2434 \__kernel_intarray_gset:Nnn #3 {#2} {#1}
```

```
2435 \__kernel_intarray_gset:Nnn #4 {#2} {0} 22435
2436 \__kernel_intarray_gset:Nnn #5 {#2} {0} 22436
2437 } 22437
2438 \cs_new_protected:Npn \fparray_gzero:N #1 22438
2439 { 22439
2440 \int_zero:N \l__fp_array_loop_int 22440
2441 \prg_replicate:nn { \fparray_count:N #1 } 22441
2442 { 22442
2443 \int_incr:N \l__fp_array_loop_int 22443
2444 \exp_after:wN \__fp_array_gset_special:nnNNN 22444
2445 \exp_after:wN 0 22445
2446 \exp_after:wN \l__fp_array_loop_int 22446
2447 #1 22447
2448 } 22448
2449 } 22449
2450 \cs_generate_variant:Nn \fparray_gzero:N { c } 22450
2451 \cs_new:Npn \fparray_item:Nn #1#2 22451
2452 { 22452
2453 \exp_after:wN \__fp_array_item:NwN 22453
2454 \exp_after:wN #1 22454
2455 \int_value:w \int_eval:n {#2} \__fp_sep: 22455
2456 \__fp_to_decimal:w 22456
2457 } 22457
2458 \cs_generate_variant:Nn \fparray_item:Nn { c } 22458
2459 \cs_new:Npn \fparray_item_to_tl:Nn #1#2 22459
2460 { 22460
2461 \exp_after:wN \__fp_array_item:NwN 22461
2462 \exp_after:wN #1 22462
2463 \int_value:w \int_eval:n {#2} \__fp_sep: 22463
2464 \__fp_to_tl:w 22464
2465 } 22465
2466 \cs_generate_variant:Nn \fparray_item_to_tl:Nn { c } 22466
2467 \cs_new:Npn \__fp_array_item:NwN #1#2 \__fp_sep: #3 22467
2468 { 22468
2469 \__fp_array_bounds:NNnTF \msg_expandable_error:nnfff #1 {#2} 22469
2470 { \exp_after:wN \__fp_array_item:NNNnN #1 {#2} #3 } 22470
2471 { \exp_after:wN #3 \c_nan_fp } 22471
2472 } 22472
2473 \cs_new:Npn \__fp_array_item:NNNnN #1#2#3#4 22473
2474 { 22474
2475 \exp_after:wN \__fp_array_item:N 22475
2476 \int_value:w \__kernel_intarray_item:Nn #2 {#4} \exp_after:wN \__fp_sep: 22476
2477 \int_value:w \__kernel_intarray_item:Nn #3 {#4} \exp_after:wN \__fp_sep: 22477
2478 \int_value:w \__kernel_intarray_item:Nn #1 {#4} \__fp_sep: 22478
2479 } 22479
2480 \cs_new:Npn \__fp_array_item:N #1 22480
```



```
2481 {
2482     \if_meaning:w 0 #1 \exp_after:wN \__fp_array_item_special:w \fi:
2483     \__fp_array_item:w #1
2484 }
2485 \cs_new:Npn \__fp_array_item:w #1 #2#3#4#5 #6 \__fp_sep: 1 #7 \__fp_sep:
2486 {
2487     \exp_after:wN \__fp_array_item_normal:w
2488     \int_value:w \if_meaning:w #1 1 0 \else: 2 \fi: \exp_stop_f:
2489     #7 \__fp_sep: {#2#3#4#5} {#6} \__fp_sep:
2490 }
2491 \cs_new:Npn \__fp_array_item_special:w #1 \__fp_sep: #2 \__fp_sep: #3 \__fp_sep: #4
2492 {
2493     \exp_after:wN #4
2494     \exp:w \exp_end_continue_f:w
2495     \if_case:w #3 \exp_stop_f:
2496         \exp_after:wN \c_zero_fp
2497     \or: \exp_after:wN \c_nan_fp
2498     \or: \exp_after:wN \c_minus_zero_fp
2499     \or: \exp_after:wN \c_inf_fp
2500     \else: \exp_after:wN \c_minus_inf_fp
2501     \fi:
2502 }
2503 \cs_new:Npn \__fp_array_item_normal:w
2504     #1 #2#3#4#5 #6 \__fp_sep: #7 \__fp_sep: #8 \__fp_sep: #9
2505     { #9 \s__fp \__fp_chk:w 1 #1 {#8} #7 {#2#3#4#5} {#6} \__fp_sep: }
2506 \prg_new_eq_conditional:NNn \fpararray_if_exist:N \cs_if_exist:N
2507 { TF , T , F , p }
2508 \prg_new_eq_conditional:NNn \fpararray_if_exist:c \cs_if_exist:c
2509 { TF , T , F , p }
2510 %% File: l3bitset.dtx
2511 \cs_if_exist:NT \@expl@finalise@setup@@
2512 {
2513     \tl_gput_right:Nn \@expl@finalise@setup@@
2514     { \declare@file@substitution { l3bitset.sty } { null.tex } }
2515 }
2516 \cs_new_protected:Npn \bitset_new:N #1
2517 {
2518     \__kernel_chk_if_free_cs:N #1
2519     \cs_gset_eq:NN #1 \c_zero_str
2520     \prop_new:c { g__bitset_ \cs_to_str:N #1 _name_prop }
2521 }
2522 \cs_new_protected:Npn \bitset_new:Nn #1 #2
2523 {
2524     \__kernel_chk_if_free_cs:N #1
2525     \cs_gset_eq:NN #1 \c_zero_str
2526     \prop_new:c { g__bitset_ \cs_to_str:N #1 _name_prop }
```

```

2527     \prop_gset_from_keyval:cn
2528     { g__bitset_ \cs_to_str:N #1 _name_prop }
2529     {#2}
2530 }
2531 \cs_generate_variant:Nn \bitset_new:N { c }
2532 \cs_generate_variant:Nn \bitset_new:Nn { c }
2533 \cs_new_protected:Npn \bitset_addto_named_index:Nn #1#2
2534 {
2535     \prop_gput_from_keyval:cn
2536     { g__bitset_ \cs_to_str:N #1 _name_prop } { #2 }
2537 }
2538 \prg_new_eq_conditional:NNn
2539     \bitset_if_exist:N \str_if_exist:N { p , T , F , TF }
2540 \prg_new_eq_conditional:NNn
2541     \bitset_if_exist:c \str_if_exist:c { p , T , F , TF }
2542 \cs_new_protected:Npn \__bitset_set_true:Nn #1#2
2543 { \__bitset_set:NNnN \str_set:Ne #1 {#2} 1 }
2544 \cs_new_protected:Npn \__bitset_gset_true:Nn #1#2
2545 { \__bitset_set:NNnN \str_gset:Ne #1 {#2} 1 }
2546 \cs_new_protected:Npn \__bitset_set_false:Nn #1#2
2547 { \__bitset_set:NNnN \str_set:Ne #1 {#2} 0 }
2548 \cs_new_protected:Npn \__bitset_gset_false:Nn #1#2
2549 { \__bitset_set:NNnN \str_gset:Ne #1 {#2} 0 }
2550 \cs_new_protected:Npn \__bitset_set:NNnN #1#2#3#4
2551 {
2552     \int_compare:nNnT {#3} > { 0 }
2553     {
2554         \int_compare:nNnTF { \str_count:N #2 } < {#3}
2555         {
2556             #1 #2
2557             {
2558                 #4
2559                 \prg_replicate:nn { #3 - \str_count:N #2 - 1 } { 0 }
2560                 #2
2561             }
2562         }
2563         {
2564             #1 #2
2565             {
2566                 \str_range:Nnn #2 { 1 } { -1 - (#3) }
2567                 #4
2568                 \str_range:Nnn #2 { 1 - (#3) } { -1 }
2569             }
2570         }
2571     }
2572 }

```

```
2573 \int_new:N \l__bitset_internal_int
2574 \prg_new_protected_conditional:Npnn \__bitset_test_digits:n #1 { TF }
2575 {
2576   \tex_afterassignment:D \__bitset_test_digits:w
2577   \l__bitset_internal_int = 0 \tl_trim_spaces_apply:nN {#1} \tl_to_str:n
2578   \__bitset_test_digits_end:
2579   \use_i:nnn \if_false:
2580   \__bitset_test_digits_end:
2581   \if_int_compare:w \c_zero_int < \l__bitset_internal_int
2582     \prg_return_true:
2583   \else:
2584     \prg_return_false:
2585   \fi:
2586 }
2587 \cs_new_eq:NN \__bitset_test_digits_end: \exp_stop_f:
2588 \cs_new_protected:Npn \__bitset_test_digits:w #1 \__bitset_test_digits_end: { }
2589 \cs_new_protected:Npn \bitset_set_true:Nn #1#2
2590   { \__bitset_set:NNn \__bitset_set_true:Nn #1 {#2} }
2591 \cs_new_protected:Npn \bitset_gset_true:Nn #1#2
2592   { \__bitset_set:NNn \__bitset_gset_true:Nn #1 {#2} }
2593 \cs_new_protected:Npn \bitset_set_false:Nn #1#2
2594   { \__bitset_set:NNn \__bitset_set_false:Nn #1 {#2} }
2595 \cs_new_protected:Npn \bitset_gset_false:Nn #1#2
2596   { \__bitset_set:NNn \__bitset_gset_false:Nn #1 {#2} }
2597 \cs_new_protected:Npn \__bitset_set:NNn #1#2#3
2598   {
2599     \prop_if_in:cnTF { g__bitset_ \cs_to_str:N #2 _name_prop } {#3}
2600     {
2601       #1 #2
2602       {
2603         \prop_item:cn { g__bitset_ \cs_to_str:N #2 _name_prop } {#3}
2604       }
2605     }
2606     {
2607       \__bitset_test_digits:nTF {#3}
2608       {
2609         #1 #2 {#3}
2610         \prop_gput:cnn { g__bitset_ \cs_to_str:N #2 _name_prop } {#3} {#3}
2611       }
2612       {
2613         \msg_warning:nnee { bitset } { unknown-name }
2614         { \token_to_str:N #2 }
2615         { \tl_to_str:n {#3} }
2616       }
2617     }
2618   }
```

```
2619 \cs_generate_variant:Nn \bitset_set_true:Nn { c } 22619
2620 \cs_generate_variant:Nn \bitset_gset_true:Nn { c } 22620
2621 \cs_generate_variant:Nn \bitset_set_false:Nn { c } 22621
2622 \cs_generate_variant:Nn \bitset_gset_false:Nn { c } 22622
2623 \cs_new_protected:Npn \bitset_clear:N #1 22623
2624 { 22624
2625     \str_set_eq:NN #1 \c_zero_str 22625
2626 } 22626
2627 \cs_new_protected:Npn \bitset_gclear:N #1 22627
2628 { 22628
2629     \str_gset_eq:NN #1 \c_zero_str 22629
2630 } 22630
2631 \cs_generate_variant:Nn \bitset_clear:N { c } 22631
2632 \cs_generate_variant:Nn \bitset_gclear:N { c } 22632
2633 \cs_new:Npn \bitset_to_arabic:N #1 22633
2634 { 22634
2635     \int_compare:nNnTF { \str_count:N #1 } < { 32 } 22635
2636     { \exp_args:No \int_from_bin:n {#1} } 22636
2637     { 22637
2638         \exp_after:wN \__bitset_to_int:nN \exp_after:wN 0 22638
2639         #1 \q_recursion_tail \q_recursion_stop 22639
2640     } 22640
2641 } 22641
2642 \cs_new:Npn \__bitset_to_int:nN #1#2 22642
2643 { 22643
2644     \quark_if_recursion_tail_stop_do:Nn #2 {#1} 22644
2645     \exp_args:Nf \__bitset_to_int:nN { \fp_eval:n { #1 * 2 + #2 } } 22645
2646 } 22646
2647 \cs_new:Npn \bitset_to_bin:N #1 22647
2648 { 22648
2649     #1 22649
2650 } 22650
2651 \cs_generate_variant:Nn \bitset_to_arabic:N { c } 22651
2652 \cs_generate_variant:Nn \bitset_to_bin:N { c } 22652
2653 \cs_new_eq:NN \bitset_use:N \tl_use:N 22653
2654 \cs_generate_variant:Nn \bitset_use:N { c } 22654
2655 \cs_new:Npn \bitset_item:Nn #1#2 22655
2656 { 22656
2657     \prop_if_in:cnTF { g__bitset_ \cs_to_str:N #1 _name_prop } {#2} 22657
2658     { 22658
2659         \int_eval:n 22659
2660         { 22660
2661             \str_item:Nn #1 22661
2662             { 0 - ( \prop_item:cn { g__bitset_ \cs_to_str:N #1 _name_prop } {#2} ) } 22662
2663             +0 22663
2664         } 22664
```

```
2665     }
2666     {
2667         0
2668     }
2669 }
2670 \cs_generate_variant:Nn \bitset_item:Nn { c }
2671 \cs_new_protected:Npn \bitset_show:N { \__bitset_show:NN \msg_show:nneeee }
2672 \cs_generate_variant:Nn \bitset_show:N { c }
2673 \cs_new_protected:Npn \bitset_log:N { \__bitset_show:NN \msg_log:nneeee }
2674 \cs_generate_variant:Nn \bitset_log:N { c }
2675 \cs_new_protected:Npn \__bitset_show:NN #1#2
2676 {
2677     \__kernel_chk_defined:NT #2
2678     {
2679         #1 { \bitset } { \show }
2680         { \token_to_str:N #2 }
2681         { \bitset_to_bin:N #2 }
2682         { \bitset_to_arabic:N #2 }
2683         { }
2684     }
2685 }
2686 \cs_new_protected:Npn \bitset_show_named_index:N
2687 { \__bitset_show_named_index:NN \msg_show:nneeee }
2688 \cs_generate_variant:Nn \bitset_show_named_index:N { c }
2689 \cs_new_protected:Npn \bitset_log_named_index:N
2690 { \__bitset_show_named_index:NN \msg_log:nneeee }
2691 \cs_generate_variant:Nn \bitset_log_named_index:N { c }
2692 \cs_new_protected:Npn \__bitset_show_named_index:NN #1#2
2693 {
2694     \__kernel_chk_defined:NT #2
2695     {
2696         #1 { \bitset } { \show-names }
2697         { \token_to_str:N #2 }
2698         { \prop_map_function:cN { g__bitset_ \cs_to_str:N #2 _name_prop }
2699 \msg_show_item:nn }
2700         { } { }
2701     }
2702 }
2703 \msg_new:nnn { \bitset } { \show }
2704 {
2705     The~\bitset~#1~has~the~representation: \\
2706     >~binary:~#2 \\
2707     >~arabic:~#3 .
2708 }
2709 \msg_new:nnn { \bitset } { \show-names }
```

```

2710 The~bitset~#1~
2711 \tl_if_empty:nTF {#2}
2712 { knows~no~names~yet \>~ . }
2713 { knows~the~name/index~pairs~(without~outer~braces): #2 . }
2714 }
2715 \msg_new:nnn { bitset } { unknown-name }
2716 { The~name~'#2'~is~unknown~for~bitset~\tl_to_str:n {#1} }
2717 \prop_gput:Nnn \g_msg_module_name_prop { bitset } { LaTeX }
2718 \prop_gput:Nnn \g_msg_module_type_prop { bitset } { }
2719 %% File: l3cctab.dtx
2720 \seq_new:N \g__cctab_stack_seq
2721 \seq_new:N \g__cctab_unused_seq
2722 \seq_new:N \g__cctab_group_seq
2723 \int_new:N \g__cctab_allocate_int
2724 \tl_new:N \l__cctab_internal_a_tl
2725 \tl_new:N \l__cctab_internal_b_tl
2726 \prop_new:N \g__cctab_endlinechar_prop
2727 \sys_if_engine luatex:TF
2728 {
2729 \cs_new_protected:Npn \cctab_new:N #1
2730 {
2731 \__kernel_chk_if_free_cs:N #1
2732 \__cctab_new:N #1
2733 }
2734 \cs_new_protected:Npn \__cctab_new:N #1
2735 {
2736 \newcatcodetable #1
2737 \tex_initcatcodetable:D #1
2738 }
2739 }
2740 {
2741 \cs_new_protected:Npn \__cctab_new:N #1
2742 {
2743 \debug_suspend:
2744 \intarray_new:Nn #1 { 257 }
2745 \debug_resume:
2746 }
2747 \cs_new_protected:Npn \__cctab_gstore:Nnn #1#2#3
2748 { \intarray_gset:Nnn #1 { #2 + 1 } {#3} }
2749 \cs_new_protected:Npn \cctab_new:N #1
2750 {
2751 \__kernel_chk_if_free_cs:N #1
2752 \__cctab_new:N #1
2753 \int_step_inline:nn { 256 }
2754 { \__kernel_intarray_gset:Nnn #1 {##1} { 12 } }
2755 \__kernel_intarray_gset:Nnn #1 { 257 } { 13 }

```

```

2756     \__cctab_gstore:Nnn #1 { 0 } { 9 }
2757     \__cctab_gstore:Nnn #1 { 13 } { 5 }
2758     \__cctab_gstore:Nnn #1 { 32 } { 10 }
2759     \__cctab_gstore:Nnn #1 { 37 } { 14 }
2760     \int_step_inline:nnn { 65 } { 90 }
2761     { \__cctab_gstore:Nnn #1 {##1} { 11 } }
2762     \__cctab_gstore:Nnn #1 { 92 } { 0 }
2763     \int_step_inline:nnn { 97 } { 122 }
2764     { \__cctab_gstore:Nnn #1 {##1} { 11 } }
2765     \__cctab_gstore:Nnn #1 { 127 } { 15 }
2766 }
2767 }
2768 \cs_generate_variant:Nn \cctab_new:N { c }
2769 \sys_if_engine luatex:TF
2770 {
2771     \cs_new_protected:Npn \__cctab_gset:n #1
2772     { \exp_args:Nf \__cctab_gset_aux:n { \int_eval:n {#1} } }
2773     \cs_new_protected:Npn \__cctab_gset_aux:n #1
2774     {
2775         \tex_savecatcodetable:D #1 \scan_stop:
2776         \int_compare:nNnTF { \tex_endlinechar:D } = { 13 }
2777         { \prop_gremove:Nn \g__cctab_endlinechar_prop {#1} }
2778         {
2779             \prop_gput:NnV \g__cctab_endlinechar_prop {#1}
2780             \tex_endlinechar:D
2781         }
2782     }
2783 }
2784 {
2785     \cs_new_protected:Npn \__cctab_gset:n #1
2786     {
2787         \int_step_inline:nn { 256 }
2788         {
2789             \__kernel_intarray_gset:Nnn #1 {##1}
2790             { \char_value_catcode:n { ##1 - 1 } }
2791         }
2792         \__kernel_intarray_gset:Nnn #1 { 257 }
2793         { \tex_endlinechar:D }
2794     }
2795 }
2796 \cs_new_protected:Npn \cctab_gset:Nn #1#2
2797 {
2798     \__cctab_chk_if_valid:NT #1
2799     {
2800         \group_begin:
2801         \cctab_select:N \c_initex_cctab

```



```

2802         #2 \scan_stop:
2803         \__cctab_gset:n {#1}
2804     \group_end:
2805 }
2806 }
2807 \cs_generate_variant:Nn \cctab_gset:Nn { c }
2808 \cs_new_protected:Npn \cctab_gsave_current:N #1
2809 {
2810     \__cctab_chk_if_valid:NT #1
2811     { \__cctab_gset:n {#1} }
2812 }
2813 \cs_generate_variant:Nn \cctab_gsave_current:N { c }
2814 \sys_if_engine luatex:T
2815 {
2816     \__cctab_new:N \g__cctab_internal_cctab
2817     \cs_new:Npn \__cctab_internal_cctab_name:
2818     {
2819         g__cctab_internal
2820         \tex_romannumeral:D \tex_currentgrouplevel:D
2821         _cctab
2822     }
2823 }
2824 \cs_new_protected:Npn \cctab_select:N #1
2825 { \__cctab_chk_if_valid:NT #1 { \__cctab_select:N #1 } }
2826 \cs_generate_variant:Nn \cctab_select:N { c }
2827 \sys_if_engine luatex:TF
2828 {
2829     \cs_new_protected:Npn \__cctab_select:N #1
2830     {
2831         \tex_catcodetable:D #1
2832         \prop_get:NVNTF \g__cctab_endlinechar_prop #1 \l__cctab_internal_a_tl
2833         { \int_set:Nn \tex_endlinechar:D { \l__cctab_internal_a_tl } }
2834         { \int_set:Nn \tex_endlinechar:D { 13 } }
2835         \cs_if_exist:cF { \__cctab_internal_cctab_name: }
2836         { \exp_args:Nc \__cctab_new:N { \__cctab_internal_cctab_name: } }
2837         \exp_args:Nc \tex_savecatcodetable:D { \__cctab_internal_cctab_name: }
2838         \exp_args:Nc \tex_catcodetable:D { \__cctab_internal_cctab_name: }
2839     }
2840 }
2841 {
2842     \cs_new_protected:Npn \__cctab_select:N #1
2843     {
2844         \int_step_inline:nn { 256 }
2845         {
2846             \char_set_catcode:nn { ##1 - 1 }
2847             { \__kernel_intarray_item:Nn #1 {##1} }

```

```

2848     }
2849     \int_set:Nn \tex_endlinechar:D
2850     { \__kernel_intarray_item:Nn #1 { 257 } }
2851 }
2852 }
2853 \sys_if_engine_luatex:TF
2854 {
2855     \cs_new_protected:Npn \__cctab_begin_aux:
2856     {
2857         \__cctab_new:N \g__cctab_next_cctab
2858         \tl_set:NV \l__cctab_internal_a_tl \g__cctab_next_cctab
2859         \cs_undefine:N \g__cctab_next_cctab
2860     }
2861 }
2862 {
2863     \cs_new_protected:Npn \__cctab_begin_aux:
2864     {
2865         \int_gincr:N \g__cctab_allocate_int
2866         \exp_args:Nc \__cctab_new:N
2867         { g__cctab_ \int_use:N \g__cctab_allocate_int _cctab }
2868         \exp_args:NNc \tl_set:Nn \l__cctab_internal_a_tl
2869         { g__cctab_ \int_use:N \g__cctab_allocate_int _cctab }
2870     }
2871 }
2872 \cs_new_protected:Npn \cctab_begin:N #1
2873 {
2874     \__cctab_chk_if_valid:NT #1
2875     {
2876         \seq_gpop:NNF \g__cctab_unused_seq \l__cctab_internal_a_tl
2877         { \__cctab_begin_aux: }
2878         \__cctab_chk_group_begin:e
2879         { \__cctab_nesting_number:N \l__cctab_internal_a_tl }
2880         \seq_gpush:NV \g__cctab_stack_seq \l__cctab_internal_a_tl
2881         \exp_args:NV \__cctab_gset:n \l__cctab_internal_a_tl
2882         \__cctab_select:N #1
2883     }
2884 }
2885 \cs_generate_variant:Nn \cctab_begin:N { c }
2886 \cs_new_protected:Npn \cctab_end:
2887 {
2888     \seq_gpop:NNTF \g__cctab_stack_seq \l__cctab_internal_a_tl
2889     {
2890         \seq_gpush:NV \g__cctab_unused_seq \l__cctab_internal_a_tl
2891         \exp_args:Ne \__cctab_chk_group_end:n
2892         { \__cctab_nesting_number:N \l__cctab_internal_a_tl }
2893         \__cctab_select:N \l__cctab_internal_a_tl

```

```

2894     }
2895     { \msg_error:nn { cctab } { extra-end } }
2896 }
2897 \cs_new_protected:Npn \__cctab_chk_group_begin:n #1
2898 {
2899     \seq_gpush:Ne \g__cctab_group_seq
2900     { \int_use:N \tex_currentgrouplevel:D }
2901     \cs_set_eq:cN { __cctab_group_ #1 _chk: } \prg_do_nothing:
2902 }
2903 \cs_generate_variant:Nn \__cctab_chk_group_begin:n { e }
2904 \cs_new_protected:Npn \__cctab_chk_group_end:n #1
2905 {
2906     \seq_gpop:NN \g__cctab_group_seq \l__cctab_internal_b_tl
2907     \bool_lazy_and:nnF
2908     {
2909         \int_compare_p:nNn
2910         { \tex_currentgrouplevel:D } = { \l__cctab_internal_b_tl }
2911     }
2912     { \cs_if_exist_p:c { __cctab_group_ #1 _chk: } }
2913     {
2914         \msg_error:nne { cctab } { group-mismatch }
2915         {
2916             \int_sign:n
2917             { \tex_currentgrouplevel:D - \l__cctab_internal_b_tl }
2918         }
2919     }
2920     \cs_undefine:c { __cctab_group_ #1 _chk: }
2921 }
2922 \sys_if_engine luatex:TF
2923 { \cs_new:Npn \__cctab_nesting_number:N #1 {#1} }
2924 {
2925     \cs_new:Npn \__cctab_nesting_number:N #1
2926     {
2927         \exp_after:wN \exp_after:wN \exp_after:wN \__cctab_nesting_number:w
2928         \exp_after:wN \token_to_str:N #1
2929     }
2930     \use:e
2931     {
2932         \cs_new:Npn \exp_not:N \__cctab_nesting_number:w
2933         #1 \tl_to_str:n { g__cctab_ } #2 \tl_to_str:n { _cctab } {#2}
2934     }
2935 }
2936 \cs_if_exist:NT \hook_gput_code:nnn
2937 {
2938     \hook_gput_code:nnn { enddocument/end } { cctab }
2939     {

```

```
2940 \seq_if_empty:NF \g__cctab_stack_seq
2941 { \msg_error:nn { cctab } { missing-end } }
2942 }
2943 }
2944 \cs_new:Npn \cctab_item:Nn #1#2
2945 { \exp_args:Nf \__cctab_item:nN { \int_eval:n {#2} } #1 }
2946 \sys_if_engine luatex:TF
2947 {
2948 \cs_new:Npn \__cctab_item:nN #1#2
2949 { \lua_now:e { tex.print(-2, tex.getcatcode(\int_use:N #2, #1)) } }
2950 }
2951 {
2952 \cs_new:Npn \__cctab_item:nN #1#2
2953 {
2954 \int_compare:nNnTF {#1} < { 256 }
2955 { \intarray_item:Nn #2 { #1 + 1 } }
2956 { \char_value_catcode:n {#1} }
2957 }
2958 }
2959 \cs_generate_variant:Nn \cctab_item:Nn { c }
2960 \prg_new_eq_conditional:NNn \cctab_if_exist:N \cs_if_exist:N
2961 { TF , T , F , p }
2962 \prg_new_eq_conditional:NNn \cctab_if_exist:c \cs_if_exist:c
2963 { TF , T , F , p }
2964 \prg_new_protected_conditional:Npnn \__cctab_chk_if_valid:N #1
2965 { TF , T , F }
2966 {
2967 \cctab_if_exist:NTF #1
2968 {
2969 \__cctab_chk_if_valid_aux:NTF #1
2970 { \prg_return_true: }
2971 {
2972 \msg_error:nne { cctab } { invalid-cctab }
2973 { \token_to_str:N #1 }
2974 \prg_return_false:
2975 }
2976 }
2977 {
2978 \msg_error:nne { kernel } { command-not-defined }
2979 { \token_to_str:N #1 }
2980 \prg_return_false:
2981 }
2982 }
2983 \sys_if_engine luatex:TF
2984 {
2985 \cs_new_protected:Npn \__cctab_chk_if_valid_aux:NTF #1
```

```

2986     {
2987         \int_compare:nNnTF {#1-1} < { \e@alloc@ccodetable@count }
2988     }
2989 \cs_if_exist:NT \c_syst_catcodes_n
2990 {
2991     \cs_gset_protected:Npn \__cctab_chk_if_valid_aux:NTF #1
2992     {
2993         \int_compare:nTF { #1 <= \c_syst_catcodes_n }
2994     }
2995 }
2996 }
2997 {
2998     \cs_new_protected:Npn \__cctab_chk_if_valid_aux:NTF #1
2999     {
3000         \exp_args:Nf \str_if_in:nnTF
3001         { \cs_meaning:N #1 }
3002         { select~font~cmr10~at~ }
3003     }
3004 }
3005 \cs_new_protected:Npn \cctab_const:Nn #1#2
3006 {
3007     \__kernel_chk_if_free_cs:N #1
3008     \__cctab_new:N #1
3009     \group_begin:
3010         \cctab_select:N \c_initex_cctab
3011         #2 \scan_stop:
3012         \__cctab_gset:n {#1}
3013     \group_end:
3014 }
3015 \cs_generate_variant:Nn \cctab_const:Nn { c }
3016 \cctab_new:N \c_initex_cctab
3017 \cctab_const:Nn \c_other_cctab
3018 {
3019     \cctab_select:N \c_initex_cctab
3020     \int_set:Nn \tex_endlinechar:D { -1 }
3021     \int_step_inline:nnn { 0 } { 127 }
3022     { \char_set_catcode_other:n {#1} }
3023 }
3024 \cctab_const:Nn \c_str_cctab
3025 {
3026     \cctab_select:N \c_other_cctab
3027     \char_set_catcode_space:n { 32 }
3028 }
3029 \cs_if_exist:NTF \@expl@finalise@setup@@
3030 { \tl_gput_right:Nn \@expl@finalise@setup@@ }
3031 { \use:n }

```

```

3032 {
3033   \__cctab_new:N \c_code_cctab
3034   \group_begin:
3035     \int_set:Nn \tex_endlinechar:D { 32 }
3036     \char_set_catcode_invalid:n { 0 }
3037     \sys_if_engine_opentype:TF
3038       { \int_step_function:nnN { 31 } { 31 } \char_set_catcode_invalid:n }
3039       { \int_step_function:nnN { 31 } { 31 } \char_set_catcode_active:n }
3040     \int_step_function:nnN { 33 } { 64 } \char_set_catcode_other:n
3041     \int_step_function:nnN { 65 } { 90 } \char_set_catcode_letter:n
3042     \int_step_function:nnN { 91 } { 96 } \char_set_catcode_other:n
3043     \int_step_function:nnN { 97 } { 122 } \char_set_catcode_letter:n
3044     \char_set_catcode_ignore:n { 9 } % tab
3045     \char_set_catcode_other:n { 10 } % lf
3046     \char_set_catcode_active:n { 12 } % ff
3047     \char_set_catcode_end_line:n { 13 } % cr
3048     \char_set_catcode_ignore:n { 32 } % space
3049     \char_set_catcode_parameter:n { 35 } % hash
3050     \char_set_catcode_math_toggle:n { 36 } % dollar
3051     \char_set_catcode_comment:n { 37 } % percent
3052     \char_set_catcode_alignment:n { 38 } % ampersand
3053     \char_set_catcode_letter:n { 58 } % colon
3054     \char_set_catcode_escape:n { 92 } % backslash
3055     \char_set_catcode_math_superscript:n { 94 } % circumflex
3056     \char_set_catcode_letter:n { 95 } % underscore
3057     \char_set_catcode_group_begin:n { 123 } % left brace
3058     \char_set_catcode_other:n { 124 } % pipe
3059     \char_set_catcode_group_end:n { 125 } % right brace
3060     \char_set_catcode_space:n { 126 } % tilde
3061     \char_set_catcode_invalid:n { 127 } % ^^?
3062     \sys_if_engine_opentype:F
3063       { \int_step_function:nnN { 128 } { 255 } \char_set_catcode_active:n }
3064     \__cctab_gset:n { \c_code_cctab }
3065   \group_end:
3066   \cctab_const:Nn \c_document_cctab
3067   {
3068     \cctab_select:N \c_code_cctab
3069     \int_set:Nn \tex_endlinechar:D { 13 }
3070     \char_set_catcode_space:n { 9 }
3071     \char_set_catcode_space:n { 32 }
3072     \char_set_catcode_other:n { 58 }
3073     \char_set_catcode_math_subscript:n { 95 }
3074     \char_set_catcode_active:n { 126 }
3075   }
3076 }
3077 \cctab_new:N \g_tmpa_cctab

```

```
3078 \cctab_new:N \g_tmpb_cctab
3079 \msg_new:nnnn { cctab } { stack-full }
3080 { The~category~code~table~stack~is~exhausted. }
3081 {
3082   LaTeX~has~been~asked~to~switch~to~a~new~category~code~table,~
3083   but~there~is~no~more~space~to~do~this!
3084 }
3085 \msg_new:nnnn { cctab } { extra-end }
3086 { Extra~\iow_char:N\\cctab_end:~ignored~\msg_line_context:. }
3087 {
3088   LaTeX~came~across~a~\iow_char:N\\cctab_end:~without~a~matching~
3089   \iow_char:N\\cctab_begin:N.~This~command~will~be~ignored.
3090 }
3091 \msg_new:nnnn { cctab } { missing-end }
3092 { Missing~\iow_char:N\\cctab_end:~before~end~of~TeX~run. }
3093 {
3094   LaTeX~came~across~more~\iow_char:N\\cctab_begin:N~than~
3095   \iow_char:N\\cctab_end:.
3096 }
3097 \msg_new:nnnn { cctab } { invalid-cctab }
3098 { Invalid~\iow_char:N\\catcode~table. }
3099 {
3100   You~can~only~switch~to~a~\iow_char:N\\catcode~table~that~is~
3101   initialized~using~\iow_char:N\\cctab_new:N~or~
3102   \iow_char:N\\cctab_const:Nn.
3103 }
3104 \msg_new:nnnn { cctab } { group-mismatch }
3105 {
3106   \iow_char:N\\cctab_end:~occurred~in~a~
3107   \int_case:nn {#1}
3108   {
3109     { 0 } { different~group }
3110     { 1 } { higher~group~level }
3111     { -1 } { lower~group~level }
3112   } ~than~
3113   the~matching~\iow_char:N\\cctab_begin:N.
3114 }
3115 {
3116   Catcode~tables~and~groups~must~be~properly~nested,~but~
3117   you~tried~to~interleave~them.~LaTeX~will~try~to~proceed,~
3118   but~results~may~be~unexpected.
3119 }
3120 \prop_gput:Nnn \g_msg_module_name_prop { cctab } { LaTeX }
3121 \prop_gput:Nnn \g_msg_module_type_prop { cctab } { }
3122 %% File l3sort.dtx
3123 \cs_new_eq:NN \__sort_sep: \__kernel_int_sep:
```



```

3124 \seq_new:N \g__sort_internal_seq
3125 \tl_new:N \g__sort_internal_tl
3126 \int_new:N \l__sort_length_int
3127 \int_new:N \l__sort_min_int
3128 \int_new:N \l__sort_top_int
3129 \int_new:N \l__sort_max_int
3130 \int_new:N \l__sort_true_max_int
3131 \int_new:N \l__sort_block_int
3132 \int_new:N \l__sort_begin_int
3133 \int_new:N \l__sort_end_int
3134 \int_new:N \l__sort_A_int
3135 \int_new:N \l__sort_B_int
3136 \int_new:N \l__sort_C_int
3137 \scan_new:N \s__sort_mark
3138 \scan_new:N \s__sort_stop
3139 \cs_new_protected:Npn \__sort_shrink_range:
3140 {
3141   \int_set:Nn \l__sort_A_int
3142     { \l__sort_true_max_int - \l__sort_min_int + 1 }
3143   \int_set:Nn \l__sort_block_int { \c_max_register_int / 2 }
3144   \__sort_shrink_range_loop:
3145   \int_set:Nn \l__sort_max_int
3146     {
3147     \int_compare:nNnTF
3148       { \l__sort_block_int * 3 / 2 } > \l__sort_A_int
3149       {
3150         \l__sort_min_int
3151         + ( \l__sort_A_int - 1 ) / 2
3152         + \l__sort_block_int / 4
3153         - 1
3154       }
3155       { \l__sort_true_max_int - \l__sort_block_int / 2 }
3156     }
3157 }
3158 \cs_new_protected:Npn \__sort_shrink_range_loop:
3159 {
3160   \if_int_compare:w \l__sort_A_int < \l__sort_block_int
3161     \tex_divide:D \l__sort_block_int 2 \exp_stop_f:
3162     \exp_after:wN \__sort_shrink_range_loop:
3163   \fi:
3164 }
3165 \cs_new_protected:Npn \__sort_compute_range:
3166 {
3167   \int_set:Nn \l__sort_min_int { \tex_count:D 15 + 1 }
3168   \int_set:Nn \l__sort_true_max_int { \c_max_register_int + 1 }
3169   \__sort_shrink_range:

```

```
3170 \if_meaning:w \loctoks \tex_undefined:D \else: 23170
3171 \if_meaning:w \loctoks \scan_stop: \else: 23171
3172 \__sort_redefine_compute_range: 23172
3173 \__sort_compute_range: 23173
3174 \fi: 23174
3175 \fi: 23175
3176 } 23176
3177 \cs_new_protected:Npn \__sort_redefine_compute_range: 23177
3178 { 23178
3179 \cs_if_exist:cTF { ver@elocalloc.sty } 23179
3180 { 23180
3181 \cs_gset_protected:Npn \__sort_compute_range: 23181
3182 { 23182
3183 \int_set:Nn \l__sort_min_int { \tex_count:D 265 } 23183
3184 \int_set_eq:NN \l__sort_true_max_int \e@alloc@top 23184
3185 \__sort_shrink_range: 23185
3186 } 23186
3187 } 23187
3188 { 23188
3189 \cs_gset_protected:Npn \__sort_compute_range: 23189
3190 { 23190
3191 \int_set:Nn \l__sort_min_int { \tex_count:D 265 } 23191
3192 \int_set:Nn \l__sort_true_max_int { \tex_count:D 275 } 23192
3193 \__sort_shrink_range: 23193
3194 } 23194
3195 } 23195
3196 } 23196
3197 \cs_if_exist:NT \loctoks { \__sort_redefine_compute_range: } 23197
3198 \tl_map_inline:nn { \lastallocatedtoks \c_syst_last_allocated_toks } 23198
3199 { 23199
3200 \cs_if_exist:NT #1 23200
3201 { 23201
3202 \cs_gset_protected:Npn \__sort_compute_range: 23202
3203 { 23203
3204 \int_set:Nn \l__sort_min_int { #1 + 1 } 23204
3205 \int_set:Nn \l__sort_true_max_int { \c_max_register_int + 1 } 23205
3206 \__sort_shrink_range: 23206
3207 } 23207
3208 } 23208
3209 } 23209
3210 \cs_new_protected:Npn \__sort_main:NNNn #1#2#3#4 23210
3211 { 23211
3212 \__sort_disable_toksdef: 23212
3213 \__sort_compute_range: 23213
3214 \int_set_eq:NN \l__sort_top_int \l__sort_min_int 23214
3215 #1 #3 23215
```

```
3216 {
3217     \if_int_compare:w \l__sort_top_int = \l__sort_max_int
3218         \__sort_too_long_error:NNw #2 #3
3219     \fi:
3220     \tex_toks:D \l__sort_top_int {##1}
3221     \int_incr:N \l__sort_top_int
3222 }
3223 \int_set:Nn \l__sort_length_int
3224 { \l__sort_top_int - \l__sort_min_int }
3225 \cs_set:Npn \__sort_compare:nn ##1 ##2 {#4}
3226 \int_set:Nn \l__sort_block_int { 1 }
3227 \__sort_level:
3228 }
3229 \cs_new_protected:Npn \tl_sort:Nn { \__sort_tl:NNn \tl_set_eq:NN }
3230 \cs_generate_variant:Nn \tl_sort:Nn { c }
3231 \cs_new_protected:Npn \tl_gsort:Nn { \__sort_tl:NNn \tl_gset_eq:NN }
3232 \cs_generate_variant:Nn \tl_gsort:Nn { c }
3233 \cs_new_protected:Npn \__sort_tl:NNn #1#2#3
3234 {
3235     \group_begin:
3236         \__sort_main:NNNn \tl_map_inline:Nn \tl_map_break:n #2 {#3}
3237         \__kernel_tl_gset:Nx \g__sort_internal_tl
3238         { \__sort_tl_toks:w \l__sort_min_int \__sort_sep: }
3239     \group_end:
3240     #1 #2 \g__sort_internal_tl
3241     \tl_gclear:N \g__sort_internal_tl
3242     \prg_break_point:
3243 }
3244 \cs_new:Npn \__sort_tl_toks:w #1 \__sort_sep:
3245 {
3246     \if_int_compare:w #1 < \l__sort_top_int
3247         { \tex_the:D \tex_toks:D #1 }
3248         \exp_after:wN \__sort_tl_toks:w
3249         \int_value:w \int_eval:n { #1 + 1 } \exp_after:wN \__sort_sep:
3250     \fi:
3251 }
3252 \cs_new_protected:Npn \seq_sort:Nn
3253 { \__sort_seq:NNNNn \seq_map_inline:Nn \seq_map_break:n \seq_set_eq:NN }
3254 \cs_generate_variant:Nn \seq_sort:Nn { c }
3255 \cs_new_protected:Npn \seq_gsort:Nn
3256 { \__sort_seq:NNNNn \seq_map_inline:Nn \seq_map_break:n \seq_gset_eq:NN }
3257 \cs_generate_variant:Nn \seq_gsort:Nn { c }
3258 \cs_new_protected:Npn \clist_sort:Nn
3259 {
3260     \__sort_seq:NNNNn \clist_map_inline:Nn \clist_map_break:n
3261     \clist_set_from_seq:NN
```

```

3262 }
3263 \cs_generate_variant:Nn \clist_sort:Nn { c }
3264 \cs_new_protected:Npn \clist_gsort:Nn
3265 {
3266     \__sort_seq:NNNNn \clist_map_inline:Nn \clist_map_break:n
3267     \clist_gset_from_seq:NN
3268 }
3269 \cs_generate_variant:Nn \clist_gsort:Nn { c }
3270 \cs_new_protected:Npn \__sort_seq:NNNNn #1#2#3#4#5
3271 {
3272     \group_begin:
3273     \__sort_main:NNNn #1 #2 #4 {#5}
3274     \seq_gclear:N \g__sort_internal_seq
3275     \int_step_inline:nnn
3276     \l__sort_min_int { \l__sort_top_int - 1 }
3277     {
3278         \seq_gput_right:Ne \g__sort_internal_seq
3279         { \tex_the:D \tex_toks:D ##1 }
3280     }
3281     \group_end:
3282     #3 #4 \g__sort_internal_seq
3283     \seq_gclear:N \g__sort_internal_seq
3284     \prg_break_point:
3285 }
3286 \cs_new_protected:Npn \__sort_level:
3287 {
3288     \if_int_compare:w \l__sort_block_int < \l__sort_length_int
3289     \l__sort_end_int \l__sort_min_int
3290     \__sort_merge_blocks:
3291     \tex_advance:D \l__sort_block_int \l__sort_block_int
3292     \exp_after:wN \__sort_level:
3293     \fi:
3294 }
3295 \cs_new_protected:Npn \__sort_merge_blocks:
3296 {
3297     \l__sort_begin_int \l__sort_end_int
3298     \tex_advance:D \l__sort_end_int \l__sort_block_int
3299     \if_int_compare:w \l__sort_end_int < \l__sort_top_int
3300     \l__sort_A_int \l__sort_end_int
3301     \tex_advance:D \l__sort_end_int \l__sort_block_int
3302     \if_int_compare:w \l__sort_end_int > \l__sort_top_int
3303     \l__sort_end_int \l__sort_top_int
3304     \fi:
3305     \l__sort_B_int \l__sort_A_int
3306     \l__sort_C_int \l__sort_top_int
3307     \__sort_copy_block:

```

```

3308     \int_decr:N \l__sort_A_int
3309     \int_decr:N \l__sort_B_int
3310     \int_decr:N \l__sort_C_int
3311     \exp_after:wN \__sort_merge_blocks_aux:
3312     \exp_after:wN \__sort_merge_blocks:
3313 \fi:
3314 }
3315 \cs_new_protected:Npn \__sort_copy_block:
3316 {
3317     \tex_toks:D \l__sort_C_int \tex_toks:D \l__sort_B_int
3318     \int_incr:N \l__sort_C_int
3319     \int_incr:N \l__sort_B_int
3320     \if_int_compare:w \l__sort_B_int = \l__sort_end_int
3321         \use_i:nn
3322     \fi:
3323     \__sort_copy_block:
3324 }
3325 \cs_new_protected:Npn \__sort_merge_blocks_aux:
3326 {
3327     \exp_after:wN \__sort_compare:nn \exp_after:wN
3328     { \tex_the:D \tex_toks:D \exp_after:wN \l__sort_A_int \exp_after:wN }
3329     \exp_after:wN { \tex_the:D \tex_toks:D \l__sort_C_int }
3330 \prg_do_nothing:
3331 \__sort_return_mark:w
3332 \__sort_return_mark:w
3333 \s__sort_mark
3334 \__sort_return_none_error:
3335 }
3336 \cs_new_protected:Npn \sort_return_same:
3337 #1 \__sort_return_mark:w #2 \s__sort_mark
3338 {
3339     #1
3340     #2
3341     \__sort_return_two_error:
3342     \__sort_return_mark:w
3343     \s__sort_mark
3344     \__sort_return_same:w
3345 }
3346 \cs_new_protected:Npn \sort_return_swapped:
3347 #1 \__sort_return_mark:w #2 \s__sort_mark
3348 {
3349     #1
3350     #2
3351     \__sort_return_two_error:
3352     \__sort_return_mark:w
3353     \s__sort_mark

```

```
3354 \__sort_return_swapped:w 2335
3355 } 2335
3356 \cs_new_protected:Npn \__sort_return_mark:w #1 \s__sort_mark { } 2335
3357 \cs_new_protected:Npn \__sort_return_none_error: 2335
3358 { 2335
3359 \msg_error:nnee { sort } { return-none } 2335
3360 { \tex_the:D \tex_toks:D \l__sort_A_int } 2336
3361 { \tex_the:D \tex_toks:D \l__sort_C_int } 2336
3362 \__sort_return_same:w \__sort_return_none_error: 2336
3363 } 2336
3364 \cs_new_protected:Npn \__sort_return_two_error: 2336
3365 { 2336
3366 \msg_error:nnee { sort } { return-two } 2336
3367 { \tex_the:D \tex_toks:D \l__sort_A_int } 2336
3368 { \tex_the:D \tex_toks:D \l__sort_C_int } 2336
3369 } 2336
3370 \cs_new_protected:Npn \__sort_return_same:w #1 \__sort_return_none_error: 2337
3371 { 2337
3372 \tex_toks:D \l__sort_B_int \tex_toks:D \l__sort_C_int 2337
3373 \int_decr:N \l__sort_B_int 2337
3374 \int_decr:N \l__sort_C_int 2337
3375 \if_int_compare:w \l__sort_C_int < \l__sort_top_int 2337
3376 \use_i:nn 2337
3377 \fi: 2337
3378 \__sort_merge_blocks_aux: 2337
3379 } 2337
3380 \cs_new_protected:Npn \__sort_return_swapped:w #1 \__sort_return_none_error: 2338
3381 { 2338
3382 \tex_toks:D \l__sort_B_int \tex_toks:D \l__sort_A_int 2338
3383 \int_decr:N \l__sort_B_int 2338
3384 \int_decr:N \l__sort_A_int 2338
3385 \if_int_compare:w \l__sort_A_int < \l__sort_begin_int 2338
3386 \__sort_merge_blocks_end: \use_i:nn 2338
3387 \fi: 2338
3388 \__sort_merge_blocks_aux: 2338
3389 } 2338
3390 \cs_new_protected:Npn \__sort_merge_blocks_end: 2339
3391 { 2339
3392 \tex_toks:D \l__sort_B_int \tex_toks:D \l__sort_C_int 2339
3393 \int_decr:N \l__sort_B_int 2339
3394 \int_decr:N \l__sort_C_int 2339
3395 \if_int_compare:w \l__sort_B_int < \l__sort_begin_int 2339
3396 \use_i:nn 2339
3397 \fi: 2339
3398 \__sort_merge_blocks_end: 2339
3399 } 2339
```

```
3400 \cs_new:Npn \tl_sort:nN #1#2
3401 {
3402   \exp_not:f
3403   {
3404     \tl_if_blank:nF {#1}
3405     {
3406       \__sort_quick_prepare:Nnnn #2 { } { }
3407       #1
3408       { \prg_break_point: \__sort_quick_prepare_end:NNNnw }
3409       \s__sort_stop
3410     }
3411   }
3412 }
3413 \cs_new:Npn \__sort_quick_prepare:Nnnn #1#2#3#4
3414 {
3415   \prg_break: #4 \prg_break_point:
3416   \__sort_quick_prepare:Nnnn #1 { #2 #3 } { #1 {#4} }
3417 }
3418 \cs_new:Npn \__sort_quick_prepare_end:NNNnw #1#2#3#4#5 \s__sort_stop
3419 {
3420   \__sort_quick_split:NnNn #4 \__sort_quick_end:nnTFNn { }
3421   \s__sort_mark { \__sort_quick_cleanup:w \exp_stop_f: }
3422   \s__sort_mark \s__sort_stop
3423 }
3424 \cs_new:Npn \__sort_quick_cleanup:w #1 \s__sort_mark \s__sort_stop {#1}
3425 \cs_new:Npn \__sort_quick_split:NnNn #1#2#3#4
3426 {
3427   #3 {#2} {#4} \__sort_quick_only_ii:NnnnnNn
3428   \__sort_quick_only_i:NnnnnNn
3429   \__sort_quick_single_end:nnwnw
3430   { #3 {#4} } { } { } {#2}
3431 }
3432 \cs_new:Npn \__sort_quick_only_i:NnnnnNn #1#2#3#4#5#6#7
3433 {
3434   #6 {#5} {#7} \__sort_quick_split_ii:NnnnnNn
3435   \__sort_quick_only_i:NnnnnNn
3436   \__sort_quick_only_i_end:nnwnw
3437   { #6 {#7} } { #3 #2 } { } {#5}
3438 }
3439 \cs_new:Npn \__sort_quick_only_ii:NnnnnNn #1#2#3#4#5#6#7
3440 {
3441   #6 {#5} {#7} \__sort_quick_only_ii:NnnnnNn
3442   \__sort_quick_split_i:NnnnnNn
3443   \__sort_quick_only_ii_end:nnwnw
3444   { #6 {#7} } { } { #4 #2 } {#5}
3445 }
```



```
3446 \cs_new:Npn \__sort_quick_split_i:NnnnnNn #1#2#3#4#5#6#7
3447 {
3448     #6 {#5} {#7} \__sort_quick_split_ii:NnnnnNn
3449     \__sort_quick_split_i:NnnnnNn
3450     \__sort_quick_split_end:nnnwnw
3451     { #6 {#7} } { #3 #2 } {#4} {#5}
3452 }
3453 \cs_new:Npn \__sort_quick_split_ii:NnnnnNn #1#2#3#4#5#6#7
3454 {
3455     #6 {#5} {#7} \__sort_quick_split_ii:NnnnnNn
3456     \__sort_quick_split_i:NnnnnNn
3457     \__sort_quick_split_end:nnnwnw
3458     { #6 {#7} } {#3} { #4 #2 } {#5}
3459 }
3460 \cs_new:Npn \__sort_quick_end:nnTFNn #1#2#3#4#5#6 {#5}
3461 \cs_new:Npn \__sort_quick_single_end:nnnwnw #1#2#3#4 \s__sort_mark #5#6 \s__sort_stop
3462 { #5 {#3} #6 \s__sort_stop }
3463 \cs_new:Npn \__sort_quick_only_i_end:nnnwnw #1#2#3#4 \s__sort_mark #5#6 \s__sort_stop
3464 {
3465     \__sort_quick_split:NnNn #1
3466     \__sort_quick_end:nnTFNn { } \s__sort_mark {#5}
3467     {#3}
3468     #6 \s__sort_stop
3469 }
3470 \cs_new:Npn \__sort_quick_only_ii_end:nnnwnw #1#2#3#4 \s__sort_mark #5#6 \s__sort_stop
3471 {
3472     \__sort_quick_split:NnNn #2
3473     \__sort_quick_end:nnTFNn { } \s__sort_mark { #5 {#3} }
3474     #6 \s__sort_stop
3475 }
3476 \cs_new:Npn \__sort_quick_split_end:nnnwnw #1#2#3#4 \s__sort_mark #5#6 \s__sort_stop
3477 {
3478     \__sort_quick_split:NnNn #2 \__sort_quick_end:nnTFNn { } \s__sort_mark
3479     {
3480         \__sort_quick_split:NnNn #1
3481         \__sort_quick_end:nnTFNn { } \s__sort_mark {#5}
3482         {#3}
3483     }
3484     #6 \s__sort_stop
3485 }
3486 \cs_new_protected:Npn \__sort_error:
3487 {
3488     \cs_set_eq:NN \__sort_merge_blocks_aux: \prg_do_nothing:
3489     \cs_set_eq:NN \__sort_merge_blocks: \prg_do_nothing:
3490     \cs_set_protected:Npn \__sort_level: { \group_end: \prg_break: }
3491 }
```

```

3492 \cs_new_protected:Npn \__sort_disable_toksdef:
3493 { \cs_set_eq:NN \toksdef \__sort_disabled_toksdef:n }
3494 \cs_new_protected:Npn \__sort_disabled_toksdef:n #1
3495 {
3496   \msg_error:nne { sort } { toksdef }
3497   { \token_to_str:N #1 }
3498   \__sort_error:
3499   \tex_toksdef:D #1
3500 }
3501 \msg_new:nnnn { sort } { toksdef }
3502 { Allocation~of~\iow_char:N\\toks~registers~impossible~while~sorting. }
3503 {
3504   The~comparison~code~used~for~sorting~a~list~has~attempted~to~
3505   define~#1~as~a~new~\iow_char:N\\toks~register~using~
3506   \iow_char:N\\newtoks~
3507   or~a~similar~command.~The~list~will~not~be~sorted.
3508 }
3509 \cs_new_protected:Npn \__sort_too_long_error:NNw #1#2 \fi:
3510 {
3511   \fi:
3512   \msg_error:nneee { sort } { too-large }
3513   { \token_to_str:N #2 }
3514   { \int_eval:n { \l__sort_true_max_int - \l__sort_min_int } }
3515   { \int_eval:n { \l__sort_top_int - \l__sort_min_int } }
3516   #1 \__sort_error:
3517 }
3518 \msg_new:nnnn { sort } { too-large }
3519 { The~list~#1~is~too~long~to~be~sorted~by~TeX. }
3520 {
3521   TeX~has~#2~toks~registers~still~available:~
3522   this~only~allows~to~sort~with~up~to~#3~
3523   items.~The~list~will~not~be~sorted.
3524 }
3525 \msg_new:nnnn { sort } { return-none }
3526 { The~comparison~code~did~not~return. }
3527 {
3528   When~sorting~a~list,~the~code~to~compare~items~#1~and~#2~
3529   did~not~call~
3530   \iow_char:N\\sort_return_same: ~nor~
3531   \iow_char:N\\sort_return_swapped: .~
3532   Exactly~one~of~these~should~be~called.
3533 }
3534 \msg_new:nnnn { sort } { return-two }
3535 { The~comparison~code~returned~multiple~times. }
3536 {
3537   When~sorting~a~list,~the~code~to~compare~items~#1~and~#2~called~

```

```
3538 \iow_char:N\sort_return_same: ~or~ 2353
3539 \iow_char:N\sort_return_swapped: ~multiple~times.~ 2353
3540 Exactly~one~of~these~should~be~called. 2354
3541 } 2354
3542 \prop_gput:Nnn \g_msg_module_name_prop { sort } { LaTeX } 2354
3543 \prop_gput:Nnn \g_msg_module_type_prop { sort } { } 2354
3544 %% File: l3str-convert.dtx 2354
3545 \cs_new_protected:Npn \__str_tmp:w { } 2354
3546 \tl_new:N \l__str_internal_tl 2354
3547 \tl_new:N \g__str_result_tl 2354
3548 \int_const:Nn \c__str_replacement_char_int { "FFFD } 2354
3549 \int_const:Nn \c__str_max_byte_int { 255 } 2354
3550 \scan_new:N \s__str 2355
3551 \quark_new:N \q__str_nil 2355
3552 \prop_new:N \g__str_alias_prop 2355
3553 \prop_gput:Nnn \g__str_alias_prop { latin1 } { iso88591 } 2355
3554 \prop_gput:Nnn \g__str_alias_prop { latin2 } { iso88592 } 2355
3555 \prop_gput:Nnn \g__str_alias_prop { latin3 } { iso88593 } 2355
3556 \prop_gput:Nnn \g__str_alias_prop { latin4 } { iso88594 } 2355
3557 \prop_gput:Nnn \g__str_alias_prop { latin5 } { iso88599 } 2355
3558 \prop_gput:Nnn \g__str_alias_prop { latin6 } { iso885910 } 2355
3559 \prop_gput:Nnn \g__str_alias_prop { latin7 } { iso885913 } 2355
3560 \prop_gput:Nnn \g__str_alias_prop { latin8 } { iso885914 } 2356
3561 \prop_gput:Nnn \g__str_alias_prop { latin9 } { iso885915 } 2356
3562 \prop_gput:Nnn \g__str_alias_prop { latin10 } { iso885916 } 2356
3563 \prop_gput:Nnn \g__str_alias_prop { utf16le } { utf16 } 2356
3564 \prop_gput:Nnn \g__str_alias_prop { utf16be } { utf16 } 2356
3565 \prop_gput:Nnn \g__str_alias_prop { utf32le } { utf32 } 2356
3566 \prop_gput:Nnn \g__str_alias_prop { utf32be } { utf32 } 2356
3567 \prop_gput:Nnn \g__str_alias_prop { hexadecimal } { hex } 2356
3568 \bool_lazy_any:nTF 2356
3569 { 2356
3570 \sys_if_engine luatex_p: 2357
3571 \sys_if_engine xetex_p: 2357
3572 } 2357
3573 { 2357
3574 \prop_gput:Nnn \g__str_alias_prop { default } { } 2357
3575 } 2357
3576 { 2357
3577 \prop_gput:Nnn \g__str_alias_prop { default } { utf8 } 2357
3578 } 2357
3579 \bool_new:N \g__str_error_bool 2357
3580 \flag_new:N \l__str_byte_flag 2358
3581 \flag_new:N \l__str_error_flag 2358
3582 \prg_new_conditional:Npnn \__str_if_contains_char:Nn #1#2 { T , TF } 2358
3583 { 2358
```

```
3584 \exp_after:wN \__str_if_contains_char_aux:nn \exp_after:wN {#1} {#2} 2358
3585 { \prg_break:n { ? \fi: } } 2358
3586 \prg_break_point: 2358
3587 \prg_return_false: 2358
3588 } 2358
3589 \cs_new:Npn \__str_if_contains_char_aux:nn #1#2 2358
3590 { \__str_if_contains_char_auxi:nN {#2} #1 } 2359
3591 \prg_new_conditional:Npnn \__str_if_contains_char:nn #1#2 { TF } 2359
3592 { 2359
3593 \__str_if_contains_char_auxi:nN {#2} #1 { \prg_break:n { ? \fi: } } 2359
3594 \prg_break_point: 2359
3595 \prg_return_false: 2359
3596 } 2359
3597 \cs_new:Npn \__str_if_contains_char_auxi:nN #1#2 2359
3598 { 2359
3599 \if_charcode:w #1 #2 2359
3600 \exp_after:wN \__str_if_contains_char_true: 2360
3601 \fi: 2360
3602 \__str_if_contains_char_auxi:nN {#1} 2360
3603 } 2360
3604 \cs_new:Npn \__str_if_contains_char_true: 2360
3605 { \prg_break:n { \prg_return_true: \use_none:n } } 2360
3606 \prg_new_conditional:Npnn \__str_octal_use:N #1 { TF } 2360
3607 { 2360
3608 \if_int_compare:w 1 < '1 \token_to_str:N #1 \exp_stop_f: 2360
3609 #1 \prg_return_true: 2360
3610 \else: 2361
3611 \prg_return_false: 2361
3612 \fi: 2361
3613 } 2361
3614 \prg_new_conditional:Npnn \__str_hexadecimal_use:N #1 { TF } 2361
3615 { 2361
3616 \if_int_compare:w 1 < "1 \token_to_str:N #1 \exp_stop_f: 2361
3617 #1 \prg_return_true: 2361
3618 \else: 2361
3619 \if_case:w \int_eval:n { \exp_after:wN ` \token_to_str:N #1 - `a } 2361
3620 A 2362
3621 \or: B 2362
3622 \or: C 2362
3623 \or: D 2362
3624 \or: E 2362
3625 \or: F 2362
3626 \else: 2362
3627 \prg_return_false: 2362
3628 \exp_after:wN \use_none:n 2362
3629 \fi: 2362
```

```
3630 \prg_return_true:
3631 \fi:
3632 }
3633 \group_begin:
3634 \__kernel_tl_set:Nx \l__str_internal_tl { \tl_to_str:n { 0123456789ABCDEF } }
3635 \tl_map_inline:Nn \l__str_internal_tl
3636 {
3637   \tl_map_inline:Nn \l__str_internal_tl
3638   {
3639     \tl_const:ce { c__str_byte_ \int_eval:n {"#1##1} _tl }
3640     { \char_generate:nn { "#1##1 } { 12 } #1 ##1 }
3641   }
3642 }
3643 \group_end:
3644 \tl_const:cn { c__str_byte_-1_tl } { { } \use_none:n { } }
3645 \cs_new:Npn \__str_output_byte:n #1
3646 { \__str_output_byte:w #1 \__str_output_end: }
3647 \cs_new:Npn \__str_output_byte:w
3648 {
3649   \exp_after:wN \exp_after:wN
3650   \exp_after:wN \use_i:nnn
3651   \cs:w c__str_byte_ \int_eval:w
3652 }
3653 \cs_new:Npn \__str_output_hexadecimal:n #1
3654 {
3655   \exp_after:wN \exp_after:wN
3656   \exp_after:wN \use_none:n
3657   \cs:w c__str_byte_ \int_eval:n {#1} _tl \cs_end:
3658 }
3659 \cs_new:Npn \__str_output_end:
3660 { \scan_stop: _tl \cs_end: }
3661 \cs_new:Npn \__str_output_byte_pair_be:n #1
3662 {
3663   \exp_args:Nf \__str_output_byte_pair:nnN
3664   { \int_div_truncate:nn { #1 } { "100 } } {#1} \use:nn
3665 }
3666 \cs_new:Npn \__str_output_byte_pair_le:n #1
3667 {
3668   \exp_args:Nf \__str_output_byte_pair:nnN
3669   { \int_div_truncate:nn { #1 } { "100 } } {#1} \use_ii_i:nn
3670 }
3671 \cs_new:Npn \__str_output_byte_pair:nnN #1#2#3
3672 {
3673   #3
3674   { \__str_output_byte:n { #1 } }
3675   { \__str_output_byte:n { #2 - #1 * "100 } }
```

```
3676 }
3677 \cs_new_protected:Npn \__str_convert_gmap:N #1
3678 {
3679     \__kernel_tl_gset:Nx \g__str_result_tl
3680     {
3681         \exp_after:wN \__str_convert_gmap_loop:NN
3682         \exp_after:wN #1
3683         \g__str_result_tl { ? \prg_break: }
3684         \prg_break_point:
3685     }
3686 }
3687 \cs_new:Npn \__str_convert_gmap_loop:NN #1#2
3688 {
3689     \use_none:n #2
3690     #1#2
3691     \__str_convert_gmap_loop:NN #1
3692 }
3693 \cs_new_protected:Npn \__str_convert_gmap_internal:N #1
3694 {
3695     \__kernel_tl_gset:Nx \g__str_result_tl
3696     {
3697         \exp_after:wN \__str_convert_gmap_internal_loop:Nww
3698         \exp_after:wN #1
3699         \g__str_result_tl \s__str \s__str_stop \prg_break: \s__str
3700         \prg_break_point:
3701     }
3702 }
3703 \cs_new:Npn \__str_convert_gmap_internal_loop:Nww #1 #2 \s__str #3 \s__str
3704 {
3705     \__str_use_none_delimit_by_s_stop:w #3 \s__str_stop
3706     #1 {#3}
3707     \__str_convert_gmap_internal_loop:Nww #1
3708 }
3709 \cs_new_protected:Npn \__str_if_flag_error:Nne #1
3710 {
3711     \flag_if_raised:NTF #1
3712     { \msg_error:nne { str } }
3713     { \use_none:nn }
3714 }
3715 \cs_new_protected:Npn \__str_if_flag_no_error:Nne #1#2#3
3716 { \flag_if_raised:NT #1 { \bool_gset_true:N \g__str_error_bool } }
3717 \cs_new:Npn \__str_if_flag_times:NT #1#2
3718 { \flag_if_raised:NT #1 { #2~(x \flag_height:N #1 ) } }
3719 \cs_new_protected:Npn \str_set_convert:Nnnn
3720 { \__str_convert:nNNnnn { } \tl_set_eq:NN }
3721 \cs_new_protected:Npn \str_gset_convert:Nnnn
```



```

3722 { \__str_convert:nNNnnn { } \tl_gset_eq:NN }
3723 \prg_new_protected_conditional:Npnn
3724   \str_set_convert:Nnnn #1#2#3#4 { T , F , TF }
3725 {
3726   \bool_gset_false:N \g__str_error_bool
3727   \__str_convert:nNNnnn
3728     { \cs_set_eq:NN \__str_if_flag_error:Nne \__str_if_flag_no_error:Nne }
3729     \tl_set_eq:NN #1 {#2} {#3} {#4}
3730   \bool_if:NTF \g__str_error_bool \prg_return_false: \prg_return_true:
3731 }
3732 \prg_new_protected_conditional:Npnn
3733   \str_gset_convert:Nnnn #1#2#3#4 { T , F , TF }
3734 {
3735   \bool_gset_false:N \g__str_error_bool
3736   \__str_convert:nNNnnn
3737     { \cs_set_eq:NN \__str_if_flag_error:Nne \__str_if_flag_no_error:Nne }
3738     \tl_gset_eq:NN #1 {#2} {#3} {#4}
3739   \bool_if:NTF \g__str_error_bool \prg_return_false: \prg_return_true:
3740 }
3741 \cs_new_protected:Npn \__str_convert:nNNnnn #1#2#3#4#5#6
3742 {
3743   \group_begin:
3744     #1
3745     \__kernel_tl_gset:Nx \g__str_result_tl { \__kernel_str_to_other_fast:n {#4} }
3746     \exp_after:wN \__str_convert:wwwnn
3747       \tl_to_str:n {#5} /// \s__str_stop
3748       { decode } { unescape }
3749     \prg_do_nothing:
3750     \__str_convert_decode_:
3751     \exp_after:wN \__str_convert:wwwnn
3752       \tl_to_str:n {#6} /// \s__str_stop
3753       { encode } { escape }
3754     \use_ii_i:nn
3755     \__str_convert_encode_:
3756     \__kernel_tl_gset:Nx \g__str_result_tl
3757       { \tl_to_str:V \g__str_result_tl }
3758   \group_end:
3759   #2 #3 \g__str_result_tl
3760 }
3761 \cs_new_protected:Npn \__str_convert:wwwnn
3762   #1 / #2 // #3 \s__str_stop #4#5
3763 {
3764   \__str_convert:nnn {enc} {#4} {#1}
3765   \__str_convert:nnn {esc} {#5} {#2}
3766   \exp_args:Ncc \__str_convert:NNnNN
3767     { __str_convert_#4_#1: } { __str_convert_#5_#2: } {#2}

```



```
3768 }
3769 \cs_new_protected:Npn \__str_convert:NNnNN #1#2#3#4#5
3770 {
3771   \if_meaning:w #1 #5
3772     \tl_if_empty:nF {#3}
3773     { \msg_error:nne { str } { native-escaping } {#3} }
3774     #1
3775   \else:
3776     #4 #2 #1
3777   \fi:
3778 }
3779 \cs_new_protected:Npn \__str_convert:nnn #1#2#3
3780 {
3781   \cs_if_exist:cF { __str_convert_#2_#3: }
3782   {
3783     \exp_args:Ne \__str_convert:nnnn
3784     { \__str_convert_lowercase_alphanum:n {#3} }
3785     {#1} {#2} {#3}
3786   }
3787 }
3788 \cs_new_protected:Npn \__str_convert:nnnn #1#2#3#4
3789 {
3790   \cs_if_exist:cF { __str_convert_#3_#1: }
3791   {
3792     \prop_get:NnNF \g__str_alias_prop {#1} \l__str_internal_tl
3793     { \tl_set:Nn \l__str_internal_tl {#1} }
3794     \cs_if_exist:cF { __str_convert_#3_ \l__str_internal_tl : }
3795     {
3796       \file_if_exist:nTF { l3str-#2- \l__str_internal_tl .def }
3797       {
3798         \group_begin:
3799         \cctab_select:N \c_code_cctab
3800         \file_input:n { l3str-#2- \l__str_internal_tl .def }
3801         \group_end:
3802       }
3803       {
3804         \tl_clear:N \l__str_internal_tl
3805         \msg_error:nnee { str } { unknown-#2 } {#4} {#1}
3806       }
3807     }
3808     \cs_if_exist:cF { __str_convert_#3_#1: }
3809     {
3810       \cs_gset_eq:cc { __str_convert_#3_#1: }
3811       { __str_convert_#3_ \l__str_internal_tl : }
3812     }
3813   }
```

```
3814 \cs_gset_eq:cc {__str_convert_#3_#4: } {__str_convert_#3_#1: }
3815 }
3816 \cs_new:Npn \__str_convert_lowercase_alphanum:n #1
3817 {
3818   \exp_after:wN \__str_convert_lowercase_alphanum_loop:N
3819   \tl_to_str:n {#1} { ? \prg_break: }
3820   \prg_break_point:
3821 }
3822 \cs_new:Npn \__str_convert_lowercase_alphanum_loop:N #1
3823 {
3824   \use_none:n #1
3825   \if_int_compare:w `#1 > `Z \exp_stop_f:
3826   \if_int_compare:w `#1 > `z \exp_stop_f: \else:
3827     \if_int_compare:w `#1 < `a \exp_stop_f: \else:
3828       #1
3829       \fi:
3830   \fi:
3831   \else:
3832     \if_int_compare:w `#1 < `A \exp_stop_f:
3833     \if_int_compare:w 1 < 1#1 \exp_stop_f:
3834       #1
3835       \fi:
3836     \else:
3837       \__str_output_byte:n { `#1 + `a - `A }
3838       \fi:
3839     \fi:
3840     \__str_convert_lowercase_alphanum_loop:N
3841 }
3842 \bool_lazy_any:nTF
3843 {
3844   \sys_if_engine luatex_p:
3845   \sys_if_engine xetex_p:
3846 }
3847 {
3848   \cs_new:Npn \__str_filter_bytes:n #1
3849   {
3850     \__str_filter_bytes_aux:N #1
3851     { ? \prg_break: }
3852     \prg_break_point:
3853   }
3854   \cs_new:Npn \__str_filter_bytes_aux:N #1
3855   {
3856     \use_none:n #1
3857     \if_int_compare:w `#1 < 256 \exp_stop_f:
3858       #1
3859     \else:
```

```
3860         \flag_raise:N \l__str_byte_flag
3861         \fi:
3862         \__str_filter_bytes_aux:N
3863     }
3864 }
3865 { \cs_new_eq:NN \__str_filter_bytes:n \use:n }
3866 \bool_lazy_any:nTF
3867 {
3868     \sys_if_engine luatex_p:
3869     \sys_if_engine xetex_p:
3870 }
3871 {
3872     \cs_new_protected:Npn \__str_convert_unescape_:
3873     {
3874         \flag_clear:N \l__str_byte_flag
3875         \__kernel_tl_gset:Nx \g__str_result_tl
3876         { \exp_args:No \__str_filter_bytes:n \g__str_result_tl }
3877         \__str_if_flag_error:Nne \l__str_byte_flag { non-byte } { bytes }
3878     }
3879 }
3880 { \cs_new_protected:Npn \__str_convert_unescape_: { } }
3881 \cs_new_eq:NN \__str_convert_unescape_bytes: \__str_convert_unescape_:
3882 \cs_new_protected:Npn \__str_convert_escape_: { }
3883 \cs_new_eq:NN \__str_convert_escape_bytes: \__str_convert_escape_:
3884 \cs_new_protected:Npn \__str_convert_decode_:
3885 { \__str_convert_gmap:N \__str_decode_native_char:N }
3886 \cs_new:Npn \__str_decode_native_char:N #1
3887 { #1 \s_str \int_value:w `#1 \s_str }
3888 \bool_lazy_any:nTF
3889 {
3890     \sys_if_engine luatex_p:
3891     \sys_if_engine xetex_p:
3892 }
3893 {
3894     \cs_new_protected:Npn \__str_convert_encode_:
3895     { \__str_convert_gmap_internal:N \__str_encode_native_char:n }
3896     \cs_new:Npn \__str_encode_native_char:n #1
3897     { \char_generate:nn {#1} {12} }
3898 }
3899 {
3900     \cs_new_protected:Npn \__str_convert_encode_:
3901     {
3902         \flag_clear:N \l__str_error_flag
3903         \__str_convert_gmap_internal:N \__str_encode_native_char:n
3904         \__str_if_flag_error:Nne \l__str_error_flag
3905         { native-overflow } { }
```

```
3906 }
3907 \cs_new:Npn \__str_encode_native_char:n #1
3908 {
3909   \if_int_compare:w #1 > \c__str_max_byte_int
3910     \flag_raise:N \l__str_error_flag
3911     ?
3912   \else:
3913     \char_generate:nn {#1} {12}
3914   \fi:
3915 }
3916 \msg_new:nnnn { str } { native-overflow }
3917 { Character~code~too~large~for~this~engine. }
3918 {
3919   This~engine~only~support~8-bit~characters:~
3920   valid~character~codes~are~in~the~range~[0,255].~
3921   To~manipulate~arbitrary~Unicode,~use~LuaTeX~or~XeTeX.
3922 }
3923 }
3924 \cs_new_protected:Npn \__str_convert_decode_clist:
3925 {
3926   \clist_gset:No \g__str_result_tl \g__str_result_tl
3927   \__kernel_tl_gset:Nx \g__str_result_tl
3928   {
3929     \exp_args:No \clist_map_function:nN
3930     \g__str_result_tl \__str_decode_clist_char:n
3931   }
3932 }
3933 \cs_new:Npn \__str_decode_clist_char:n #1
3934 { #1 \s__str \int_eval:n {#1} \s__str }
3935 \cs_new_protected:Npn \__str_convert_encode_clist:
3936 {
3937   \__str_convert_gmap_internal:N \__str_encode_clist_char:n
3938   \__kernel_tl_gset:Nx \g__str_result_tl { \tl_tail:N \g__str_result_tl }
3939 }
3940 \cs_new:Npn \__str_encode_clist_char:n #1 { , #1 }
3941 \cs_new_protected:Npn \__str_declare_eight_bit_encoding:nnnn #1
3942 {
3943   \tl_set:Nn \l__str_internal_tl {#1}
3944   \cs_new_protected:cpn { __str_convert_decode_#1: }
3945   { \__str_convert_decode_eight_bit:n {#1} }
3946   \cs_new_protected:cpn { __str_convert_encode_#1: }
3947   { \__str_convert_encode_eight_bit:n {#1} }
3948   \exp_args:Ncc \__str_declare_eight_bit_aux:NNnnn
3949   { g__str_decode_#1_intarray } { g__str_encode_#1_intarray }
3950 }
3951 \cs_new_protected:Npn \__str_declare_eight_bit_aux:NNnnn #1#2#3#4#5
```

```

3952 {
3953     \intarray_new:Nn #1 { 256 }
3954     \int_step_inline:nnn { 0 } { 255 }
3955         { \intarray_gset:Nnn #1 { 1 + ##1 } {##1} }
3956     \__str_declare_eight_bit_loop:Nnn #1
3957         #4 { \s__str_stop \prg_break: } { }
3958     \prg_break_point:
3959     \__str_declare_eight_bit_loop:Nn #1
3960         #5 { \s__str_stop \prg_break: }
3961     \prg_break_point:
3962     \intarray_new:Nn #2 {#3}
3963     \int_step_inline:nnn { 0 } { 255 }
3964     {
3965         \int_compare:nNnF { \intarray_item:Nn #1 { 1 + ##1 } } = { -1 }
3966         {
3967             \intarray_gset:Nnn #2
3968                 {
3969                     1 +
3970                     \int_mod:nn { \intarray_item:Nn #1 { 1 + ##1 } }
3971                     { \intarray_count:N #2 }
3972                 }
3973                 {##1}
3974             }
3975         }
3976     }
3977 \cs_new_protected:Npn \__str_declare_eight_bit_loop:Nnn #1#2#3
3978 {
3979     \__str_use_none_delimit_by_s_stop:w #2 \s__str_stop
3980     \intarray_gset:Nnn #1 { 1 + "#2 } { "#3 }
3981     \__str_declare_eight_bit_loop:Nnn #1
3982 }
3983 \cs_new_protected:Npn \__str_declare_eight_bit_loop:Nn #1#2
3984 {
3985     \__str_use_none_delimit_by_s_stop:w #2 \s__str_stop
3986     \intarray_gset:Nnn #1 { 1 + "#2 } { -1 }
3987     \__str_declare_eight_bit_loop:Nn #1
3988 }
3989 \cs_new_protected:Npn \__str_convert_decode_eight_bit:n #1
3990 {
3991     \cs_set:Npe \__str_tmp:w
3992     {
3993         \exp_not:N \__str_decode_eight_bit_aux:Nn
3994         \exp_not:c { g__str_decode_#1_intarray }
3995     }
3996     \flag_clear:N \l__str_error_flag
3997     \__str_convert_gmap:N \__str_tmp:w

```

```
3998 \__str_if_flag_error:Nne \l__str_error_flag { decode-8-bit } {#1}
3999 }
4000 \cs_new:Npn \__str_decode_eight_bit_aux:Nn #1#2
4001 {
4002     #2 \s__str
4003     \exp_args:Nf \__str_decode_eight_bit_aux:n
4004     { \intarray_item:Nn #1 { 1 + `#2 } }
4005     \s__str
4006 }
4007 \cs_new:Npn \__str_decode_eight_bit_aux:n #1
4008 {
4009     \if_int_compare:w #1 < \c_zero_int
4010         \flag_raise:N \l__str_error_flag
4011         \int_value:w \c__str_replacement_char_int
4012     \else:
4013         #1
4014     \fi:
4015 }
4016 \int_new:N \l__str_modulo_int
4017 \cs_new_protected:Npn \__str_convert_encode_eight_bit:n #1
4018 {
4019     \cs_set:Npe \__str_tmp:w
4020     {
4021         \exp_not:N \__str_encode_eight_bit_aux:NNn
4022         \exp_not:c { g__str_encode_#1_intarray }
4023         \exp_not:c { g__str_decode_#1_intarray }
4024     }
4025     \flag_clear:N \l__str_error_flag
4026     \__str_convert_gmap_internal:N \__str_tmp:w
4027     \__str_if_flag_error:Nne \l__str_error_flag { encode-8-bit } {#1}
4028 }
4029 \cs_new:Npn \__str_encode_eight_bit_aux:NNn #1#2#3
4030 {
4031     \exp_args:Nf \__str_encode_eight_bit_aux:nnN
4032     {
4033         \intarray_item:Nn #1
4034         { 1 + \int_mod:nn {#3} { \intarray_count:N #1 } }
4035     }
4036     {#3}
4037     #2
4038 }
4039 \cs_new:Npn \__str_encode_eight_bit_aux:nnN #1#2#3
4040 {
4041     \int_compare:nNnTF { \intarray_item:Nn #3 { 1 + #1 } } = {#2}
4042     { \__str_output_byte:n {#1} }
4043     { \flag_raise:N \l__str_error_flag }
```

```

4044 }
4045 \msg_new:nnn { str } { unknown-esc }
4046 { Escaping~scheme~'#1'~(filtered:~'#2')~unknown. }
4047 \msg_new:nnn { str } { unknown-enc }
4048 { Encoding~scheme~'#1'~(filtered:~'#2')~unknown. }
4049 \msg_new:nnnn { str } { native-escaping }
4050 { The~'native'~encoding~scheme~does~not~support~any~escaping. }
4051 {
4052     Since~native~strings~do~not~consist~in~bytes,~
4053     none~of~the~escaping~methods~make~sense.~
4054     The~specified~escaping,~'#1',~will~be~ignored.
4055 }
4056 \msg_new:nnn { str } { file-not-found }
4057 { File~'l3str-#1.def'~not~found. }
4058 \bool_lazy_any:nT
4059 {
4060     \sys_if_engine_luatex_p:
4061     \sys_if_engine_xetex_p:
4062 }
4063 {
4064     \msg_new:nnnn { str } { non-byte }
4065     { String~invalid~in~escaping~'#1':~it~may~only~contain~bytes. }
4066     {
4067         Some~characters~in~the~string~you~asked~to~convert~are~not~
4068         8-bit~characters.~Perhaps~the~string~is~a~'native'~Unicode~string?~
4069         If~it~is,~try~using\\
4070         \\
4071         \iow_indent:n
4072         {
4073             \iow_char:N\\str_set_convert:Nnnn \\
4074             \_\_<str~var>~\{~<string>~\}~\{~native~\}~\{~<target~encoding>~\}
4075         }
4076     }
4077 }
4078 \msg_new:nnnn { str } { decode-8-bit }
4079 { Invalid~string~in~encoding~'#1'. }
4080 {
4081     LaTeX~came~across~a~byte~which~is~not~defined~to~represent~
4082     any~character~in~the~encoding~'#1'.
4083 }
4084 \msg_new:nnnn { str } { encode-8-bit }
4085 { Unicode~string~cannot~be~converted~to~encoding~'#1'. }
4086 {
4087     The~encoding~'#1'~only~contains~a~subset~of~all~Unicode~characters.~
4088     LaTeX~was~asked~to~convert~a~string~to~that~encoding,~but~that~
4089     string~contains~a~character~that~'#1'~does~not~support.

```



```
4090 }
4091 \cs_new_protected:Npn \__str_convert_unescape_hex:
4092 {
4093   \group_begin:
4094     \flag_clear:N \l__str_error_flag
4095     \int_set:Nn \tex_escapechar:D { 92 }
4096     \__kernel_tl_gset:Nx \g__str_result_tl
4097     {
4098       \__str_output_byte:w "
4099       \exp_last_unbraced:Nf \__str_unescape_hex_auxi:N
4100         { \tl_to_str:N \g__str_result_tl }
4101       0 { ? 0 - 1 \prg_break: }
4102       \prg_break_point:
4103       \__str_output_end:
4104     }
4105     \__str_if_flag_error:Nne \l__str_error_flag { unescape-hex } { }
4106   \group_end:
4107 }
4108 \cs_new:Npn \__str_unescape_hex_auxi:N #1
4109 {
4110   \use_none:n #1
4111   \__str_hexadecimal_use:NTF #1
4112   { \__str_unescape_hex_auxii:N }
4113   {
4114     \flag_raise:N \l__str_error_flag
4115     \__str_unescape_hex_auxi:N
4116   }
4117 }
4118 \cs_new:Npn \__str_unescape_hex_auxii:N #1
4119 {
4120   \use_none:n #1
4121   \__str_hexadecimal_use:NTF #1
4122   {
4123     \__str_output_end:
4124     \__str_output_byte:w " \__str_unescape_hex_auxi:N
4125   }
4126   {
4127     \flag_raise:N \l__str_error_flag
4128     \__str_unescape_hex_auxii:N
4129   }
4130 }
4131 \msg_new:nnnn { str } { unescape-hex }
4132 { String~invalid~in~escaping~'hex':~only~hexadecimal~digits~allowed. }
4133 {
4134   Some~characters~in~the~string~you~asked~to~convert~are~not~
4135   hexadecimal~digits~(0-9,~A-F,~a-f)~nor~spaces.
```

```
4136 }
4137 \cs_set_protected:Npn \__str_tmp:w #1#2#3
4138 {
4139   \cs_new_protected:cpn { __str_convert_unescape_#2: }
4140   {
4141     \group_begin:
4142     \flag_clear:N \l__str_byte_flag
4143     \flag_clear:N \l__str_error_flag
4144     \int_set:Nn \tex_escapechar:D { 92 }
4145     \__kernel_tl_gset:Nx \g__str_result_tl
4146     {
4147       \exp_after:wN #3 \g__str_result_tl
4148       #1 ? { ? \prg_break: }
4149       \prg_break_point:
4150     }
4151     \__str_if_flag_error:Nne \l__str_byte_flag { non-byte } { #2 }
4152     \__str_if_flag_error:Nne \l__str_error_flag { unescape-#2 } { }
4153   \group_end:
4154 }
4155 \cs_new:Npn #3 ##1#1##2##3
4156 {
4157   \__str_filter_bytes:n {##1}
4158   \use_none:n ##3
4159   \__str_output_byte:w "
4160   \__str_hexadecimal_use:NTF ##2
4161   {
4162     \__str_hexadecimal_use:NTF ##3
4163     { }
4164     {
4165       \flag_raise:N \l__str_error_flag
4166       * 0 + `#1 \use_i:nn
4167     }
4168   }
4169   {
4170     \flag_raise:N \l__str_error_flag
4171     0 + `#1 \use_i:nn
4172   }
4173   \__str_output_end:
4174   \use_i:nnn #3 ##2##3
4175 }
4176 \msg_new:nnnn { str } { unescape-#2 }
4177 { String~invalid~in~escaping~'#2'. }
4178 {
4179   LaTeX~came~across~the~escape~character~'#1'~not~followed~by~
4180   two~hexadecimal~digits.~This~is~invalid~in~the~escaping~'#2'.
4181 }
```

```
4182 }
4183 \exp_after:wN \__str_tmp:w \c_hash_str { name }
4184 \__str_unescape_name_loop:wNN
4185 \exp_after:wN \__str_tmp:w \c_percent_str { url }
4186 \__str_unescape_url_loop:wNN
4187 \group_begin:
4188 \char_set_catcode_other:N ^^J
4189 \char_set_catcode_other:N ^^M
4190 \cs_set_protected:Npn \__str_tmp:w #1
4191 {
4192 \cs_new_protected:Npn \__str_convert_unescape_string:
4193 {
4194 \group_begin:
4195 \flag_clear:N \l__str_byte_flag
4196 \flag_clear:N \l__str_error_flag
4197 \int_set:Nn \tex_escapechar:D { 92 }
4198 \__kernel_tl_gset:Nx \g__str_result_tl
4199 {
4200 \exp_after:wN \__str_unescape_string_newlines:wN
4201 \g__str_result_tl \prg_break: ^^M ?
4202 \prg_break_point:
4203 }
4204 \__kernel_tl_gset:Nx \g__str_result_tl
4205 {
4206 \exp_after:wN \__str_unescape_string_loop:wNNN
4207 \g__str_result_tl #1 ?? { ? \prg_break: }
4208 \prg_break_point:
4209 }
4210 \__str_if_flag_error:Nne \l__str_byte_flag { non-byte } { string }
4211 \__str_if_flag_error:Nne \l__str_error_flag { unescape-string } { }
4212 \group_end:
4213 }
4214 }
4215 \exp_args:No \__str_tmp:w { \c_backslash_str }
4216 \exp_last_unbraced:NNNNNo
4217 \cs_new:Npn \__str_unescape_string_loop:wNNN #1 \c_backslash_str #2#3#4
4218 {
4219 \__str_filter_bytes:n {#1}
4220 \use_none:n #4
4221 \__str_output_byte:w '
4222 \__str_octal_use:NTF #2
4223 {
4224 \__str_octal_use:NTF #3
4225 }
4226 \__str_octal_use:NTF #4
4227 }
```

```
4228         \if_int_compare:w #2 > 3 \exp_stop_f:
4229         - 256
4230         \fi:
4231         \__str_unescape_string_repeat:NNNNNN
4232     }
4233     { \__str_unescape_string_repeat:NNNNNN ? }
4234 }
4235 { \__str_unescape_string_repeat:NNNNNN ?? }
4236 }
4237 {
4238     \str_case_e:nnF {#2}
4239     {
4240         { \c_backslash_str } { 134 }
4241         { ( } { 50 }
4242         { ) } { 51 }
4243         { r } { 15 }
4244         { f } { 14 }
4245         { n } { 12 }
4246         { t } { 11 }
4247         { b } { 10 }
4248         { ^^J } { 0 - 1 }
4249     }
4250     {
4251         \flag_raise:N \l__str_error_flag
4252         0 - 1 \use_i:nn
4253     }
4254 }
4255 \__str_output_end:
4256 \use_i:nn \__str_unescape_string_loop:wNNN #2#3#4
4257 }
4258 \cs_new:Npn \__str_unescape_string_repeat:NNNNNN #1#2#3#4#5#6
4259 { \__str_output_end: \__str_unescape_string_loop:wNNN }
4260 \cs_new:Npn \__str_unescape_string_newlines:wN #1 ^^M #2
4261 {
4262     #1
4263     \if_charcode:w ^^J #2 \else: ^^J \fi:
4264     \__str_unescape_string_newlines:wN #2
4265 }
4266 \msg_new:nnnn { str } { unescape-string }
4267 { String~invalid~in~escaping~'string'. }
4268 {
4269     LaTeX~came~across~an~escape~character~'\c_backslash_str'~
4270     not~followed~by~any~of:~'n',~'r',~'t',~'b',~'f',~'(',~')',~
4271     '\c_backslash_str',~one~to~three~octal~digits,~or~the~end~
4272     of~a~line.
4273 }
```

```

4274 \group_end:
4275 \cs_new_protected:Npn \__str_convert_escape_hex:
4276 { \__str_convert_gmap:N \__str_escape_hex_char:N }
4277 \cs_new:Npn \__str_escape_hex_char:N #1
4278 { \__str_output_hexadecimal:n {`#1} }
4279 \str_const:Nn \c__str_escape_name_not_str { ! " $ & ' } %$
4280 \str_const:Nn \c__str_escape_name_str { {}/<>[] }
4281 \cs_new_protected:Npn \__str_convert_escape_name:
4282 { \__str_convert_gmap:N \__str_escape_name_char:n }
4283 \cs_new:Npn \__str_escape_name_char:n #1
4284 {
4285   \__str_if_escape_name:nTF {#1} {#1}
4286   { \c_hash_str \__str_output_hexadecimal:n {`#1} }
4287 }
4288 \prg_new_conditional:Npnn \__str_if_escape_name:n #1 { TF }
4289 {
4290   \if_int_compare:w `#1 < "2A \exp_stop_f:
4291     \__str_if_contains_char:NnTF \c__str_escape_name_not_str {#1}
4292     \prg_return_true: \prg_return_false:
4293   \else:
4294     \if_int_compare:w `#1 > "7E \exp_stop_f:
4295     \prg_return_false:
4296   \else:
4297     \__str_if_contains_char:NnTF \c__str_escape_name_str {#1}
4298     \prg_return_false: \prg_return_true:
4299   \fi:
4300   \fi:
4301 }
4302 \str_const:Ne \c__str_escape_string_str
4303 { \c_backslash_str ( ) }
4304 \cs_new_protected:Npn \__str_convert_escape_string:
4305 { \__str_convert_gmap:N \__str_escape_string_char:N }
4306 \cs_new:Npn \__str_escape_string_char:N #1
4307 {
4308   \__str_if_escape_string:NnTF #1
4309   {
4310     \__str_if_contains_char:NnT
4311     \c__str_escape_string_str {#1}
4312     { \c_backslash_str }
4313     #1
4314   }
4315   {
4316     \c_backslash_str
4317     \int_div_truncate:nn {`#1} {64}
4318     \int_mod:nn { \int_div_truncate:nn {`#1} { 8 } } { 8 }
4319     \int_mod:nn {`#1} { 8 }

```

```
4320     }
4321 }
4322 \prg_new_conditional:Npnn \__str_if_escape_string:N #1 { TF }
4323 {
4324     \if_int_compare:w `#1 < "27 \exp_stop_f:
4325     \prg_return_false:
4326 \else:
4327     \if_int_compare:w `#1 > "7A \exp_stop_f:
4328     \prg_return_false:
4329 \else:
4330     \prg_return_true:
4331 \fi:
4332 \fi:
4333 }
4334 \cs_new_protected:Npn \__str_convert_escape_url:
4335 { \__str_convert_gmap:N \__str_escape_url_char:n }
4336 \cs_new:Npn \__str_escape_url_char:n #1
4337 {
4338     \__str_if_escape_url:nTF {#1} {#1}
4339     { \c_percent_str \__str_output_hexadecimal:n { `#1 } }
4340 }
4341 \prg_new_conditional:Npnn \__str_if_escape_url:n #1 { TF }
4342 {
4343     \if_int_compare:w `#1 < "30 \exp_stop_f:
4344     \__str_if_contains_char:nnTF { "-. } {#1}
4345     \prg_return_true: \prg_return_false:
4346 \else:
4347     \if_int_compare:w `#1 > "7E \exp_stop_f:
4348     \prg_return_false:
4349 \else:
4350     \__str_if_contains_char:nnTF { : ; = ? @ [ ] } {#1}
4351     \prg_return_false: \prg_return_true:
4352 \fi:
4353 \fi:
4354 }
4355 \cs_new_protected:cpn { __str_convert_encode_utf8: }
4356 { \__str_convert_gmap_internal:N \__str_encode_utf_viii_char:n }
4357 \cs_new:Npn \__str_encode_utf_viii_char:n #1
4358 {
4359     \__str_encode_utf_viii_loop:wnnw #1 \__str_sep: - 1 + 0 * \__str_sep:
4360     { 128 } { 0 }
4361     { 32 } { 192 }
4362     { 16 } { 224 }
4363     { 8 } { 240 }
4364 \s__str_stop
4365 }
```

```
4366 \cs_new:Npn \__str_encode_utf_viii_loop:wwnnw
4367 #1 \__str_sep: #2 \__str_sep: #3#4 #5 \s__str_stop
4368 {
4369   \if_int_compare:w #1 < #3 \exp_stop_f:
4370     \__str_output_byte:n { #1 + #4 }
4371     \exp_after:wN \__str_use_none_delimit_by_s_stop:w
4372   \fi:
4373   \exp_after:wN \__str_encode_utf_viii_loop:wwnnw
4374   \int_value:w \int_div_truncate:nn {#1} {64} \__str_sep: #1 \__str_sep:
4375   #5 \s__str_stop
4376   \__str_output_byte:n { #2 - 64 * ( #1 - 2 ) }
4377 }
4378 \flag_clear_new:N \l__str_missing_flag
4379 \flag_clear_new:N \l__str_extra_flag
4380 \flag_clear_new:N \l__str_overlong_flag
4381 \flag_clear_new:N \l__str_overflow_flag
4382 \msg_new:nnnn { str } { utf8-decode }
4383 {
4384   Invalid~UTF-8~string:
4385   \exp_last_unbraced:Nf \use_none:n
4386   {
4387     \__str_if_flag_times:NT \l__str_missing_flag { ,~missing~continuation~byte }
4388     \__str_if_flag_times:NT \l__str_extra_flag { ,~extra~continuation~byte }
4389     \__str_if_flag_times:NT \l__str_overlong_flag { ,~overlong~form }
4390     \__str_if_flag_times:NT \l__str_overflow_flag { ,~code~point~too~large }
4391   }
4392   .
4393 }
4394 {
4395   In~the~UTF-8~encoding,~each~Unicode~character~consists~in~
4396   1~to~4~bytes,~with~the~following~bit~pattern: \\
4397   \iow_indent:n
4398   {
4399     Code~point~\\ \\ \\ <~128:~0xxxxxxx \\
4400     Code~point~\\ \\ \\ <~2048:~110xxxxx~10xxxxxx \\
4401     Code~point~\\ \\ <~65536:~1110xxxx~10xxxxxx~10xxxxxx \\
4402     Code~point~ <~1114112:~11110xxx~10xxxxxx~10xxxxxx~10xxxxxx \\
4403   }
4404   Bytes~of~the~form~10xxxxxx~are~called~continuation~bytes.
4405   \flag_if_raised:NT \l__str_missing_flag
4406   {
4407     \\\
4408     A~leading~byte~(in~the~range~[192,255])~was~not~followed~by~
4409     the~appropriate~number~of~continuation~bytes.
4410   }
4411   \flag_if_raised:NT \l__str_extra_flag
```



```
4412 {
4413     \\\
4414     LaTeX~came~across~a~continuation~byte~when~it~was~not~expected.
4415 }
4416 \flag_if_raised:NT \l__str_overlong_flag
4417 {
4418     \\\
4419     Every~Unicode~code~point~must~be~expressed~in~the~shortest~
4420     possible~form.~For~instance,~'0xC0'~'0x83'~is~not~a~valid~
4421     representation~for~the~code~point~3.
4422 }
4423 \flag_if_raised:NT \l__str_overflow_flag
4424 {
4425     \\\
4426     Unicode~limits~code~points~to~the~range~[0,1114111].
4427 }
4428 }
4429 \prop_gput:Nnn \g_msg_module_name_prop { str } { LaTeX }
4430 \prop_gput:Nnn \g_msg_module_type_prop { str } { }
4431 \cs_new_protected:cpn { __str_convert_decode_utf8: }
4432 {
4433     \flag_clear:N \l__str_error_flag
4434     \flag_clear:N \l__str_missing_flag
4435     \flag_clear:N \l__str_extra_flag
4436     \flag_clear:N \l__str_overlong_flag
4437     \flag_clear:N \l__str_overflow_flag
4438     \__kernel_tl_gset:Nx \g__str_result_tl
4439     {
4440         \exp_after:wN \__str_decode_utf_viii_start:N \g__str_result_tl
4441         { \prg_break: \__str_decode_utf_viii_end: }
4442         \prg_break_point:
4443     }
4444     \__str_if_flag_error:Nne \l__str_error_flag { utf8-decode } { }
4445 }
4446 \cs_new:Npn \__str_decode_utf_viii_start:N #1
4447 {
4448     #1
4449     \if_int_compare:w `#1 < "C0 \exp_stop_f:
4450         \s__str
4451         \if_int_compare:w `#1 < "80 \exp_stop_f:
4452             \int_value:w `#1
4453         \else:
4454             \flag_raise:N \l__str_extra_flag
4455             \flag_raise:N \l__str_error_flag
4456             \int_use:N \c__str_replacement_char_int
4457             \fi:
```

```

4458 \else:
4459 \exp_after:wN \__str_decode_utf_viii_continuation:wwN
4460 \int_value:w \int_eval:n { `#1 - "C0 } \exp_after:wN
4461 \fi:
4462 \s__str
4463 \__str_use_none_delimit_by_s_stop:w {"80} {"800} {"10000} {"110000} \s__str_stop
4464 \__str_decode_utf_viii_start:N
4465 }
4466 \cs_new:Npn \__str_decode_utf_viii_continuation:wwN
4467 #1 \s__str #2 \__str_decode_utf_viii_start:N #3
4468 {
4469 \use_none:n #3
4470 \if_int_compare:w `#3 <
4471 \if_int_compare:w `#3 < "80 \exp_stop_f: - \fi:
4472 "C0 \exp_stop_f:
4473 #3
4474 \exp_after:wN \__str_decode_utf_viii_aux:wNnnwN
4475 \int_value:w \int_eval:n { #1 * "40 + `#3 - "80 } \exp_after:wN
4476 \else:
4477 \s__str
4478 \flag_raise:N \l__str_missing_flag
4479 \flag_raise:N \l__str_error_flag
4480 \int_use:N \c__str_replacement_char_int
4481 \fi:
4482 \s__str
4483 #2
4484 \__str_decode_utf_viii_start:N #3
4485 }
4486 \cs_new:Npn \__str_decode_utf_viii_aux:wNnnwN
4487 #1 \s__str #2#3#4 #5 \__str_decode_utf_viii_start:N #6
4488 {
4489 \if_int_compare:w #1 < #4 \exp_stop_f:
4490 \s__str
4491 \if_int_compare:w #1 < #3 \exp_stop_f:
4492 \flag_raise:N \l__str_overlong_flag
4493 \flag_raise:N \l__str_error_flag
4494 \int_use:N \c__str_replacement_char_int
4495 \else:
4496 #1
4497 \fi:
4498 \else:
4499 \if_meaning:w \s__str_stop #5
4500 \__str_decode_utf_viii_overflow:w #1
4501 \fi:
4502 \exp_after:wN \__str_decode_utf_viii_continuation:wwN
4503 \int_value:w \int_eval:n { #1 - #4 } \exp_after:wN

```

```
4504     \fi:
4505     \s__str
4506     #2 {#4} #5
4507     \__str_decode_utf_viii_start:N
4508 }
4509 \cs_new:Npn \__str_decode_utf_viii_overflow:w #1 \fi: #2 \fi:
4510 {
4511     \fi: \fi:
4512     \flag_raise:N \l__str_overflow_flag
4513     \flag_raise:N \l__str_error_flag
4514     \int_use:N \c__str_replacement_char_int
4515 }
4516 \cs_new:Npn \__str_decode_utf_viii_end:
4517 {
4518     \s__str
4519     \flag_raise:N \l__str_missing_flag
4520     \flag_raise:N \l__str_error_flag
4521     \int_use:N \c__str_replacement_char_int \s__str
4522     \prg_break:
4523 }
4524 \group_begin:
4525     \char_set_catcode_other:N ^^fe
4526     \char_set_catcode_other:N ^^ff
4527     \cs_new_protected:cpn { __str_convert_encode_utf16: }
4528     {
4529         \__str_encode_utf_xvi_aux:N \__str_output_byte_pair_be:n
4530         \tl_gput_left:Ne \g__str_result_tl { ^^fe ^^ff }
4531     }
4532     \cs_new_protected:cpn { __str_convert_encode_utf16be: }
4533     { \__str_encode_utf_xvi_aux:N \__str_output_byte_pair_be:n }
4534     \cs_new_protected:cpn { __str_convert_encode_utf16le: }
4535     { \__str_encode_utf_xvi_aux:N \__str_output_byte_pair_le:n }
4536     \cs_new_protected:Npn \__str_encode_utf_xvi_aux:N #1
4537     {
4538         \flag_clear:N \l__str_error_flag
4539         \cs_set_eq:NN \__str_tmp:w #1
4540         \__str_convert_gmap_internal:N \__str_encode_utf_xvi_char:n
4541         \__str_if_flag_error:Nne \l__str_error_flag { utf16-encode } { }
4542     }
4543     \cs_new:Npn \__str_encode_utf_xvi_char:n #1
4544     {
4545         \if_int_compare:w #1 < "D800 \exp_stop_f:
4546             \__str_tmp:w {#1}
4547         \else:
4548             \if_int_compare:w #1 < "10000 \exp_stop_f:
4549                 \if_int_compare:w #1 < "E000 \exp_stop_f:
```

```
4550         \flag_raise:N \l__str_error_flag
4551         \__str_tmp:w { \c__str_replacement_char_int }
4552     \else:
4553         \__str_tmp:w {#1}
4554     \fi:
4555 \else:
4556     \exp_args:Nf \__str_tmp:w { \int_div_truncate:nn {#1} {"400} + "D7C0 }
4557     \exp_args:Nf \__str_tmp:w { \int_mod:nn {#1} {"400} + "DC00 }
4558 \fi:
4559 \fi:
4560 }
4561 \flag_clear_new:N \l__str_missing_flag
4562 \flag_clear_new:N \l__str_extra_flag
4563 \flag_clear_new:N \l__str_end_flag
4564 \msg_new:nnnn { str } { utf16-encode }
4565 { Unicode~string~cannot~be~expressed~in~UTF-16:~surrogate. }
4566 {
4567     Surrogate~code~points~(in~the~range~[U+D800,~U+DFFF])~
4568     can~be~expressed~in~the~UTF-8~and~UTF-32~encodings,~
4569     but~not~in~the~UTF-16~encoding.
4570 }
4571 \msg_new:nnnn { str } { utf16-decode }
4572 {
4573     Invalid~UTF-16~string:
4574     \exp_last_unbraced:Nf \use_none:n
4575     {
4576         \__str_if_flag_times:NT \l__str_missing_flag { ,~missing~trail~surrogate }
4577         \__str_if_flag_times:NT \l__str_extra_flag { ,~extra~trail~surrogate }
4578         \__str_if_flag_times:NT \l__str_end_flag { ,~odd~number~of~bytes }
4579     }
4580     .
4581 }
4582 {
4583     In~the~UTF-16~encoding,~each~Unicode~character~is~encoded~as~
4584     2~or~4~bytes: \\
4585     \iow_indent:n
4586     {
4587         Code~point~in~[U+0000,~U+D7FF]:~two~bytes \\
4588         Code~point~in~[U+D800,~U+DFFF]:~illegal \\
4589         Code~point~in~[U+E000,~U+FFFF]:~two~bytes \\
4590         Code~point~in~[U+10000,~U+10FFFF]:~
4591         a~lead~surrogate~and~a~trail~surrogate \\
4592     }
4593     Lead~surrogates~are~pairs~of~bytes~in~the~range~[0xD800,~0xDBFF]~,~
4594     and~trail~surrogates~are~in~the~range~[0xDC00,~0xDFFF]~.
4595     \flag_if_raised:NT \l__str_missing_flag
```

```
4596 {
4597     \\\
4598     A~lead~surrogate~was~not~followed~by~a~trail~surrogate.
4599 }
4600 \flag_if_raised:NT \l__str_extra_flag
4601 {
4602     \\\
4603     LaTeX~came~across~a~trail~surrogate~when~it~was~not~expected.
4604 }
4605 \flag_if_raised:NT \l__str_end_flag
4606 {
4607     \\\
4608     The~string~contained~an~odd~number~of~bytes.~This~is~invalid:~
4609     the~basic~code~unit~for~UTF-16~is~16~bits~(2~bytes).
4610 }
4611 }
4612 \cs_new_protected:cpn { __str_convert_decode_utf16be: }
4613 { \__str_decode_utf_xvi:Nw 1 \g__str_result_tl \s__str_stop }
4614 \cs_new_protected:cpn { __str_convert_decode_utf16le: }
4615 { \__str_decode_utf_xvi:Nw 2 \g__str_result_tl \s__str_stop }
4616 \cs_new_protected:cpn { __str_convert_decode_utf16: }
4617 {
4618     \exp_after:wN \__str_decode_utf_xvi_bom:NN
4619     \g__str_result_tl \s__str_stop \s__str_stop \s__str_stop
4620 }
4621 \cs_new_protected:Npn \__str_decode_utf_xvi_bom:NN #1#2
4622 {
4623     \str_if_eq:nnTF { #1#2 } { ^^ff ^^fe }
4624     { \__str_decode_utf_xvi:Nw 2 }
4625     {
4626         \str_if_eq:nnTF { #1#2 } { ^^fe ^^ff }
4627         { \__str_decode_utf_xvi:Nw 1 }
4628         { \__str_decode_utf_xvi:Nw 1 #1#2 }
4629     }
4630 }
4631 \cs_new_protected:Npn \__str_decode_utf_xvi:Nw #1#2 \s__str_stop
4632 {
4633     \flag_clear:N \l__str_error_flag
4634     \flag_clear:N \l__str_missing_flag
4635     \flag_clear:N \l__str_extra_flag
4636     \flag_clear:N \l__str_end_flag
4637     \cs_set:Npn \__str_tmp:w ##1 ##2 { ` ## #1 }
4638     \__kernel_tl_gset:Nx \g__str_result_tl
4639     {
4640         \exp_after:wN \__str_decode_utf_xvi_pair:NN
4641         #2 \q__str_nil \q__str_nil
```

```
4642 \prg_break_point:
4643 }
4644 \__str_if_flag_error:Nne \l__str_error_flag { utf16-decode } { }
4645 }
4646 \cs_new:Npn \__str_decode_utf_xvi_pair:NN #1#2
4647 {
4648   \if_meaning:w \q__str_nil #2
4649     \__str_decode_utf_xvi_pair_end:Nw #1
4650   \fi:
4651   \if_case:w
4652     \int_eval:n { ( \__str_tmp:w #1#2 - "D6 ) / 4 } \scan_stop:
4653   \or: \exp_after:wN \__str_decode_utf_xvi_quad:NNwNN
4654   \or: \exp_after:wN \__str_decode_utf_xvi_extra:NNw
4655   \fi:
4656   #1#2 \s__str
4657   \int_eval:n { "100 * \__str_tmp:w #1#2 + \__str_tmp:w #2#1 } \s__str
4658   \__str_decode_utf_xvi_pair:NN
4659 }
4660 \cs_new:Npn \__str_decode_utf_xvi_quad:NNwNN
4661   #1#2 #3 \__str_decode_utf_xvi_pair:NN #4#5
4662 {
4663   \if_meaning:w \q__str_nil #5
4664     \__str_decode_utf_xvi_error:nNN { missing } #1#2
4665     \__str_decode_utf_xvi_pair_end:Nw #4
4666   \fi:
4667   \if_int_compare:w
4668     \if_int_compare:w \__str_tmp:w #4#5 < "DC \exp_stop_f:
4669       0 = 1
4670     \else:
4671       \__str_tmp:w #4#5 < "E0
4672     \fi:
4673     \exp_stop_f:
4674     #1 #2 #4 #5 \s__str
4675     \int_eval:n
4676       {
4677         ( "100 * \__str_tmp:w #1#2 + \__str_tmp:w #2#1 - "D7F7 ) * "400
4678         + "100 * \__str_tmp:w #4#5 + \__str_tmp:w #5#4
4679       }
4680     \s__str
4681     \exp_after:wN \use_i:nnn
4682   \else:
4683     \__str_decode_utf_xvi_error:nNN { missing } #1#2
4684   \fi:
4685   \__str_decode_utf_xvi_pair:NN #4#5
4686 }
4687 \cs_new:Npn \__str_decode_utf_xvi_pair_end:Nw #1 \fi:
```

```

4688 {
4689     \fi:
4690     \if_meaning:w \q__str_nil #1
4691     \else:
4692         \__str_decode_utf_xvi_error:nNN { end } #1 \prg_do_nothing:
4693     \fi:
4694     \prg_break:
4695 }
4696 \cs_new:Npn \__str_decode_utf_xvi_extra:NNw #1#2 \s__str #3 \s__str
4697 { \__str_decode_utf_xvi_error:nNN { extra } #1#2 }
4698 \cs_new:Npn \__str_decode_utf_xvi_error:nNN #1#2#3
4699 {
4700     \flag_raise:N \l__str_error_flag
4701     \flag_raise:c { l__str_#1_flag }
4702     #2 #3 \s__str
4703     \int_use:N \c__str_replacement_char_int \s__str
4704 }
4705 \group_end:
4706 \group_begin:
4707     \char_set_catcode_other:N ^^00
4708     \char_set_catcode_other:N ^^fe
4709     \char_set_catcode_other:N ^^ff
4710     \cs_new_protected:cpn { __str_convert_encode_utf32: }
4711     {
4712         \__str_convert_gmap_internal:N \__str_encode_utf_xxxii_be:n
4713         \tl_gput_left:Ne \g__str_result_tl { ^^00 ^^00 ^^fe ^^ff }
4714     }
4715     \cs_new_protected:cpn { __str_convert_encode_utf32be: }
4716     { \__str_convert_gmap_internal:N \__str_encode_utf_xxxii_be:n }
4717     \cs_new_protected:cpn { __str_convert_encode_utf32le: }
4718     { \__str_convert_gmap_internal:N \__str_encode_utf_xxxii_le:n }
4719     \cs_new:Npn \__str_encode_utf_xxxii_be:n #1
4720     {
4721         \exp_args:Nf \__str_encode_utf_xxxii_be_aux:nn
4722         { \int_div_truncate:nn {#1} { "100 } } {#1}
4723     }
4724     \cs_new:Npn \__str_encode_utf_xxxii_be_aux:nn #1#2
4725     {
4726         ^^00
4727         \__str_output_byte_pair_be:n {#1}
4728         \__str_output_byte:n { #2 - #1 * "100 }
4729     }
4730     \cs_new:Npn \__str_encode_utf_xxxii_le:n #1
4731     {
4732         \exp_args:Nf \__str_encode_utf_xxxii_le_aux:nn
4733         { \int_div_truncate:nn {#1} { "100 } } {#1}

```



```

4734 }
4735 \cs_new:Npn \__str_encode_utf_xxxii_le_aux:nn #1#2
4736 {
4737   \__str_output_byte:n { #2 - #1 * "100 }
4738   \__str_output_byte_pair_le:n {#1}
4739   ^^00
4740 }
4741 \flag_clear_new:N \l__str_overflow_flag
4742 \flag_clear_new:N \l__str_end_flag
4743 \msg_new:nnnn { str } { utf32-decode }
4744 {
4745   Invalid~UTF-32~string:
4746   \exp_last_unbraced:Nf \use_none:n
4747   {
4748     \__str_if_flag_times:NT \l__str_overflow_flag { ,~code~point~too~large }
4749     \__str_if_flag_times:NT \l__str_end_flag { ,~truncated~string }
4750   }
4751   .
4752 }
4753 {
4754   In~the~UTF-32~encoding,~every~Unicode~character~
4755   (in~the~range~[U+0000,~U+10FFFF])~is~encoded~as~4~bytes.
4756   \flag_if_raised:NT \l__str_overflow_flag
4757   {
4758     \\\
4759     LaTeX~came~across~a~code~point~larger~than~1114111,~
4760     the~maximum~code~point~defined~by~Unicode.~
4761     Perhaps~the~string~was~not~encoded~in~the~UTF-32~encoding?
4762   }
4763   \flag_if_raised:NT \l__str_end_flag
4764   {
4765     \\\
4766     The~length~of~the~string~is~not~a~multiple~of~4.~
4767     Perhaps~the~string~was~truncated?
4768   }
4769 }
4770 \cs_new_protected:cpn { __str_convert_decode_utf32be: }
4771 { \__str_decode_utf_xxxii:Nw 1 \g__str_result_tl \s__str_stop }
4772 \cs_new_protected:cpn { __str_convert_decode_utf32le: }
4773 { \__str_decode_utf_xxxii:Nw 2 \g__str_result_tl \s__str_stop }
4774 \cs_new_protected:cpn { __str_convert_decode_utf32: }
4775 {
4776   \exp_after:wN \__str_decode_utf_xxxii_bom:NNNN \g__str_result_tl
4777   \s__str_stop \s__str_stop \s__str_stop \s__str_stop \s__str_stop
4778 }
4779 \cs_new_protected:Npn \__str_decode_utf_xxxii_bom:NNNN #1#2#3#4

```

```
4780 {
4781     \str_if_eq:nnTF { #1#2#3#4 } { ^^ff ^^fe ^^00 ^^00 }
4782     { \__str_decode_utf_xxxii:Nw 2 }
4783     {
4784         \str_if_eq:nnTF { #1#2#3#4 } { ^^00 ^^00 ^^fe ^^ff }
4785         { \__str_decode_utf_xxxii:Nw 1 }
4786         { \__str_decode_utf_xxxii:Nw 1 #1#2#3#4 }
4787     }
4788 }
4789 \cs_new_protected:Npn \__str_decode_utf_xxxii:Nw #1#2 \s__str_stop
4790 {
4791     \flag_clear:N \l__str_overflow_flag
4792     \flag_clear:N \l__str_end_flag
4793     \flag_clear:N \l__str_error_flag
4794     \cs_set:Npn \__str_tmp:w ##1 ##2 { ` ## #1 }
4795     \__kernel_tl_gset:Nx \g__str_result_tl
4796     {
4797         \exp_after:wN \__str_decode_utf_xxxii_loop:NNNN
4798         #2 \s__str_stop \s__str_stop \s__str_stop \s__str_stop
4799         \prg_break_point:
4800     }
4801     \__str_if_flag_error:Nne \l__str_error_flag { utf32-decode } { }
4802 }
4803 \cs_new:Npn \__str_decode_utf_xxxii_loop:NNNN #1#2#3#4
4804 {
4805     \if_meaning:w \s__str_stop #4
4806     \exp_after:wN \__str_decode_utf_xxxii_end:w
4807     \fi:
4808     #1#2#3#4 \s__str
4809     \if_int_compare:w \__str_tmp:w #1#4 > \c_zero_int
4810         \flag_raise:N \l__str_overflow_flag
4811         \flag_raise:N \l__str_error_flag
4812         \int_use:N \c__str_replacement_char_int
4813     \else:
4814         \if_int_compare:w \__str_tmp:w #2#3 > 16 \exp_stop_f:
4815             \flag_raise:N \l__str_overflow_flag
4816             \flag_raise:N \l__str_error_flag
4817             \int_use:N \c__str_replacement_char_int
4818         \else:
4819             \int_eval:n
4820             { \__str_tmp:w #2#3*"10000 + \__str_tmp:w #3#2*"100 + \__str_tmp:w #4#1 }
4821             \fi:
4822         \fi:
4823         \s__str
4824         \__str_decode_utf_xxxii_loop:NNNN
4825     }
```

```
4826 \cs_new:Npn \__str_decode_utf_xxxii_end:w #1 \s__str_stop 2482
4827 { 2482
4828 \tl_if_empty:nF {#1} 2482
4829 { 2482
4830 \flag_raise:N \l__str_end_flag 2483
4831 \flag_raise:N \l__str_error_flag 2483
4832 #1 \s__str 2483
4833 \int_use:N \c__str_replacement_char_int \s__str 2483
4834 } 2483
4835 \prg_break: 2483
4836 } 2483
4837 \group_end: 2483
4838 \cs_new:Npn \str_convert_pdfname:n #1 2483
4839 { 2483
4840 \exp_args:Ne \tl_to_str:n 2484
4841 { \str_map_function:nN {#1} \__str_convert_pdfname:n } 2484
4842 } 2484
4843 \sys_if_engine_opentype:TF 2484
4844 { 2484
4845 \cs_new:Npn \__str_convert_pdfname:n #1 2484
4846 { 2484
4847 \int_compare:nNnTF { `#1 } > { "7F } 2484
4848 { \__str_convert_pdfname_bytes:n {#1} } 2484
4849 { \__str_escape_name_char:n {#1} } 2484
4850 } 2485
4851 \cs_new:Npn \__str_convert_pdfname_bytes:n #1 2485
4852 { 2485
4853 \exp_args:Ne \__str_convert_pdfname_bytes_aux:n 2485
4854 { \__kernel_codepoint_to_bytes:n {`#1} } 2485
4855 } 2485
4856 \cs_new:Npn \__str_convert_pdfname_bytes_aux:n #1 2485
4857 { \__str_convert_pdfname_bytes_aux:nnnn #1 } 2485
4858 \cs_new:Npe \__str_convert_pdfname_bytes_aux:nnnn #1#2#3#4 2485
4859 { 2485
4860 \c_hash_str \exp_not:N \__str_output_hexadecimal:n {#1} 2486
4861 \c_hash_str \exp_not:N \__str_output_hexadecimal:n {#2} 2486
4862 \exp_not:N \tl_if_blank:nF {#3} 2486
4863 { 2486
4864 \c_hash_str \exp_not:N \__str_output_hexadecimal:n {#3} 2486
4865 \exp_not:N \tl_if_blank:nF {#4} 2486
4866 { 2486
4867 \c_hash_str \exp_not:N \__str_output_hexadecimal:n {#4} 2486
4868 } 2486
4869 } 2486
4870 } 2487
4871 } 2487
```

```
4872 { \cs_new_eq:NN \__str_convert_pdfname:n \__str_escape_name_char:n } 24872
4873 %% File: l3tl-analysis.dtx 24873
4874 \scan_new:N \s__tl 24874
4875 \cs_new_eq:NN \l__tl_analysis_token ? 24875
4876 \cs_new_eq:NN \l__tl_analysis_char_token ? 24876
4877 \tl_new:N \l__tl_peek_code_tl 24877
4878 \group_begin: 24878
4879 \char_set_active_eq:NN \_ \scan_stop: 24879
4880 \tl_const:Ne \c__tl_peek_catcodes_tl 24880
4881 { 24881
4882   \char_generate:nn { 32 } { 3 } 3 24882
4883   \char_generate:nn { 32 } { 4 } 4 24883
4884   \char_generate:nn { 32 } { 7 } 7 24884
4885   \char_generate:nn { 32 } { 8 } 8 24885
4886   \c_space_tl \token_to_str:N A 24886
4887   \char_generate:nn { 32 } { 11 } \token_to_str:N B 24887
4888   \char_generate:nn { 32 } { 12 } \token_to_str:N C 24888
4889   \char_generate:nn { 32 } { 13 } \token_to_str:N D 24889
4890 } 24890
4891 \group_end: 24891
4892 \int_new:N \l__tl_analysis_normal_int 24892
4893 \int_new:N \l__tl_analysis_index_int 24893
4894 \int_new:N \l__tl_analysis_nesting_int 24894
4895 \int_new:N \l__tl_analysis_type_int 24895
4896 \tl_new:N \g__tl_analysis_result_tl 24896
4897 \cs_new:Npn \__tl_analysis_extract_charcode: 24897
4898 { 24898
4899   \exp_after:wN \__tl_analysis_extract_charcode_aux:w 24899
4900   \token_to_meaning:N \l__tl_analysis_token 24900
4901 } 24901
4902 \cs_new:Npn \__tl_analysis_extract_charcode_aux:w #1 ~ #2 ~ { ` } 24902
4903 \bool_lazy_and:nnT 24903
4904 { \cs_if_exist_p:N \tex_luatexversion:D } 24904
4905 { \int_compare_p:nNn { \int_div_truncate:nn { \tex_luatexversion:D } { 100 } } > 1 } 24905
4906 { 24906
4907   \cs_gset:Npn \__tl_analysis_extract_charcode_aux:w #1 + #2 ~ ' #3 ' {"#2} 24907
4908 } 24908
4909 \cs_new:Npe \__tl_analysis_cs_space_count:NN #1 #2 24909
4910 { 24910
4911   \exp_not:N \exp_after:wN #1 24911
4912   \exp_not:N \int_value:w \exp_not:N \int_eval:w 0 24912
4913   \exp_not:N \exp_after:wN \exp_not:N \__tl_analysis_cs_space_count:w 24913
4914   \exp_not:N \token_to_str:N #2 24914
4915   \exp_not:N \fi: \exp_not:N \__tl_analysis_cs_space_count_end:w 24915
4916   \exp_not:N \__tl_sep: \c_space_tl ! 24916
4917 } 24917
```

```
4918 \cs_new:Npn \__tl_analysis_cs_space_count:w #1 ~
4919 {
4920   \if_false: #1 #1 \fi:
4921   + 1
4922   \__tl_analysis_cs_space_count:w
4923 }
4924 \cs_new:Npn \__tl_analysis_cs_space_count_end:w \__tl_sep: #1 \fi: #2 !
4925 {
4926   \exp_after:wN \__tl_sep: \int_value:w
4927   \str_count_ignore_spaces:n {#1} \__tl_sep:
4928 }
4929 \cs_new_protected:Npn \__tl_analysis:n #1
4930 {
4931   \group_begin:
4932   \group_align_safe_begin:
4933     \__tl_analysis_a:n {#1}
4934     \__tl_analysis_b:n {#1}
4935   \group_align_safe_end:
4936   \group_end:
4937 }
4938 \group_begin:
4939   \char_set_catcode_active:N \^^@
4940   \cs_new_protected:Npn \__tl_analysis_disable:n #1
4941   {
4942     \tex_lccode:D 0 = #1 \exp_stop_f:
4943     \tex_lowercase:D { \tex_let:D \^^@ } \tex_undefined:D
4944   }
4945   \bool_lazy_or:nnT
4946   { \sys_if_engine_ptex_p: }
4947   { \sys_if_engine_uptex_p: }
4948   {
4949     \cs_gset_protected:Npn \__tl_analysis_disable:n #1
4950     {
4951       \if_int_compare:w 256 > #1 \exp_stop_f:
4952       \tex_lccode:D 0 = #1 \exp_stop_f:
4953       \tex_lowercase:D { \tex_let:D \^^@ } \tex_undefined:D
4954       \fi:
4955     }
4956   }
4957 \group_end:
4958 \group_begin:
4959   \char_set_catcode_active:N \^^@
4960   \cs_new_protected:Npn \__tl_analysis_disable_char:N #1
4961   {
4962     \tex_lccode:D `#1 = 32 \exp_stop_f:
4963     \tex_lowercase:D { \if_meaning:w #1 } \tex_undefined:D
```

```
4964 \tex_let:D #1 \tex_undefined:D
4965 \fi:
4966 }
4967 \bool_lazy_or:nnT
4968 { \sys_if_engine_ptex_p: }
4969 { \sys_if_engine_uptex_p: }
4970 {
4971 \cs_gset_protected:Npn \__tl_analysis_disable_char:N #1
4972 {
4973 \if_int_compare:w 256 > `#1 \exp_stop_f:
4974 \tex_lccode:D `#1 = 32 \exp_stop_f:
4975 \tex_lowercase:D { \if_meaning:w #1 } \tex_undefined:D
4976 \tex_let:D #1 \tex_undefined:D
4977 \fi:
4978 \fi:
4979 }
4980 }
4981 \group_end:
4982 \cs_new_protected:Npn \__tl_analysis_a:n #1
4983 {
4984 \__tl_analysis_disable:n { 32 }
4985 \int_set:Nn \tex_escapechar:D { 92 }
4986 \int_zero:N \l__tl_analysis_normal_int
4987 \int_zero:N \l__tl_analysis_index_int
4988 \int_zero:N \l__tl_analysis_nesting_int
4989 \if_false: { \fi: \__tl_analysis_a_loop:w #1 }
4990 \int_decr:N \l__tl_analysis_index_int
4991 }
4992 \cs_new_protected:Npn \__tl_analysis_a_loop:w
4993 { \tex_futurelet:D \l__tl_analysis_token \__tl_analysis_a_type:w }
4994 \cs_new_protected:Npn \__tl_analysis_a_type:w
4995 {
4996 \l__tl_analysis_type_int =
4997 \if_meaning:w \l__tl_analysis_token \c_space_token
4998 0
4999 \else:
5000 \if_catcode:w \exp_not:N \l__tl_analysis_token \c_group_begin_token
5001 1
5002 \else:
5003 \if_catcode:w \exp_not:N \l__tl_analysis_token \c_group_end_token
5004 - 1
5005 \else:
5006 2
5007 \fi:
5008 \fi:
5009 \fi:
```

```
5010 \exp_stop_f:
5011 \if_case:w \l__tl_analysis_type_int
5012 \exp_after:wN \__tl_analysis_a_space:w
5013 \or: \exp_after:wN \__tl_analysis_a_bgroup:w
5014 \or: \exp_after:wN \__tl_analysis_a_safe:N
5015 \else: \exp_after:wN \__tl_analysis_a_egroup:w
5016 \fi:
5017 }
5018 \cs_new_protected:Npn \__tl_analysis_a_space:w
5019 {
5020 \tex_afterassignment:D \__tl_analysis_a_space_test:w
5021 \exp_after:wN \cs_set_eq:NN
5022 \exp_after:wN \l__tl_analysis_char_token
5023 \token_to_str:N
5024 }
5025 \cs_new_protected:Npn \__tl_analysis_a_space_test:w
5026 {
5027 \if_meaning:w \l__tl_analysis_char_token \c_space_token
5028 \tex_toks:D \l__tl_analysis_index_int { \exp_not:n { ~ } }
5029 \__tl_analysis_a_store:
5030 \else:
5031 \int_incr:N \l__tl_analysis_normal_int
5032 \fi:
5033 \__tl_analysis_a_loop:w
5034 }
5035 \group_begin:
5036 \char_set_catcode_group_begin:N ^^@ % {
5037 \cs_new_protected:Npn \__tl_analysis_a_bgroup:w
5038 { \__tl_analysis_a_group:nw { \exp_after:wN ^^@ \if_false: } \fi: } }
5039 \char_set_catcode_group_end:N ^^@
5040 \cs_new_protected:Npn \__tl_analysis_a_egroup:w
5041 { \__tl_analysis_a_group:nw { \if_false: { \fi: ^^@ } } % }
5042 \group_end:
5043 \cs_new_protected:Npn \__tl_analysis_a_group:nw #1
5044 {
5045 \tex_lccode:D 0 = \__tl_analysis_extract_charcode: \scan_stop:
5046 \tex_lowercase:D { \tex_toks:D \l__tl_analysis_index_int {#1} }
5047 \if_int_compare:w \tex_lccode:D 0 = \tex_escapechar:D
5048 \int_set:Nn \tex_escapechar:D { 139 - \tex_escapechar:D }
5049 \fi:
5050 \__tl_analysis_disable:n { \tex_lccode:D 0 }
5051 \tex_futurelet:D \l__tl_analysis_token \__tl_analysis_a_group_aux:w
5052 }
5053 \cs_new_protected:Npn \__tl_analysis_a_group_aux:w
5054 {
5055 \if_meaning:w \l__tl_analysis_token \tex_undefined:D
```



```

5056 \exp_after:wN \__tl_analysis_a_safe:N
5057 \else:
5058 \exp_after:wN \__tl_analysis_a_group_auxii:w
5059 \fi:
5060 }
5061 \cs_new_protected:Npn \__tl_analysis_a_group_auxii:w
5062 {
5063 \tex_afterassignment:D \__tl_analysis_a_group_test:w
5064 \exp_after:wN \cs_set_eq:NN
5065 \exp_after:wN \l__tl_analysis_char_token
5066 \token_to_str:N
5067 }
5068 \cs_new_protected:Npn \__tl_analysis_a_group_test:w
5069 {
5070 \if_charcode:w \l__tl_analysis_token \l__tl_analysis_char_token
5071 \__tl_analysis_a_store:
5072 \else:
5073 \int_incr:N \l__tl_analysis_normal_int
5074 \fi:
5075 \__tl_analysis_a_loop:w
5076 }
5077 \cs_new_protected:Npn \__tl_analysis_a_store:
5078 {
5079 \tex_advance:D \l__tl_analysis_nesting_int \l__tl_analysis_type_int
5080 \if_int_compare:w \tex_lccode:D 0 = `\_ \exp_stop_f:
5081 \tex_advance:D \l__tl_analysis_type_int \l__tl_analysis_type_int
5082 \fi:
5083 \tex_skip:D \l__tl_analysis_index_int
5084 = \l__tl_analysis_normal_int sp
5085 plus \l__tl_analysis_type_int sp \scan_stop:
5086 \int_incr:N \l__tl_analysis_index_int
5087 \int_zero:N \l__tl_analysis_normal_int
5088 \if_int_compare:w \l__tl_analysis_nesting_int = - \c_one_int
5089 \cs_set_eq:NN \__tl_analysis_a_loop:w \scan_stop:
5090 \fi:
5091 }
5092 \cs_new_protected:Npn \__tl_analysis_a_safe:N #1
5093 {
5094 \if_charcode:w
5095 \scan_stop:
5096 \exp_after:wN \use_none:n \token_to_str:N #1 \prg_do_nothing:
5097 \scan_stop:
5098 \exp_after:wN \use_i:nn
5099 \else:
5100 \exp_after:wN \use_ii:nn
5101 \fi:

```

```

5102     {
5103         \_tl_analysis_disable_char:N #1
5104         \int_incr:N \l__tl_analysis_normal_int
5105     }
5106     { \_tl_analysis_cs_space_count:NN \_tl_analysis_a_cs:ww #1 }
5107     \_tl_analysis_a_loop:w
5108 }
5109 \cs_new_protected:Npn \_tl_analysis_a_cs:ww #1 \_tl_sep: #2 \_tl_sep:
5110 {
5111     \if_int_compare:w #1 > \c_zero_int
5112         \tex_skip:D \l__tl_analysis_index_int
5113         = \int_eval:n { \l__tl_analysis_normal_int + 1 } sp \exp_stop_f:
5114         \tex_advance:D \l__tl_analysis_index_int #1 \exp_stop_f:
5115     \else:
5116         \tex_advance:D
5117     \fi:
5118     \l__tl_analysis_normal_int #2 \exp_stop_f:
5119 }
5120 \cs_new_protected:Npn \_tl_analysis_b:n #1
5121 {
5122     \__kernel_tl_gset:Nx \g__tl_analysis_result_tl
5123     {
5124         \_tl_analysis_b_loop:w 0 \_tl_sep: #1
5125         \prg_break_point:
5126     }
5127 }
5128 \cs_new:Npn \_tl_analysis_b_loop:w #1 \_tl_sep:
5129 {
5130     \exp_after:wN \_tl_analysis_b_normals:ww
5131     \int_value:w \tex_skip:D #1 \_tl_sep: #1 \_tl_sep:
5132 }
5133 \cs_new:Npn \_tl_analysis_b_normals:ww #1 \_tl_sep:
5134 {
5135     \if_int_compare:w #1 = \c_zero_int
5136         \_tl_analysis_b_special:w
5137     \fi:
5138     \_tl_analysis_b_normal:wwN #1 \_tl_sep:
5139 }
5140 \cs_new:Npn \_tl_analysis_b_normal:wwN #1 \_tl_sep: #2 \_tl_sep: #3
5141 {
5142     \exp_not:n { \exp_not:n { #3 } } \s__tl
5143     \if_charcode:w
5144         \scan_stop:
5145         \exp_after:wN \use_none:n \token_to_str:N #3 \prg_do_nothing:
5146         \scan_stop:
5147         \exp_after:wN \_tl_analysis_b_char:Nn

```

```

5148 \exp_after:wN \_tl_analysis_b_char_aux:nww
5149 \else:
5150 \exp_after:wN \_tl_analysis_b_cs:Nww
5151 \fi:
5152 #3 #1 \_tl_sep: #2 \_tl_sep:
5153 }
5154 \cs_new:Npe \_tl_analysis_b_char:Nn #1#2
5155 {
5156 \exp_not:N \if_meaning:w #2 \exp_not:N \tex_undefined:D
5157 \token_to_str:N D \exp_not:N \else:
5158 \exp_not:N \if_catcode:w #2 \c_catcode_other_token
5159 \token_to_str:N C \exp_not:N \else:
5160 \exp_not:N \if_catcode:w #2 \c_catcode_letter_token
5161 \token_to_str:N B \exp_not:N \else:
5162 \exp_not:N \if_catcode:w #2 \c_math_toggle_token 3
5163 \exp_not:N \else:
5164 \exp_not:N \if_catcode:w #2 \c_alignment_token 4
5165 \exp_not:N \else:
5166 \exp_not:N \if_catcode:w #2 \c_math_superscript_token 7
5167 \exp_not:N \else:
5168 \exp_not:N \if_catcode:w #2 \c_math_subscript_token 8
5169 \exp_not:N \else:
5170 \exp_not:N \if_catcode:w #2 \c_space_token
5171 \token_to_str:N A \exp_not:N \else:
5172 6
5173 \exp_not:n { \fi: \fi: \fi: \fi: \fi: \fi: \fi: \fi: }
5174 #1 {#2}
5175 }
5176 \cs_new:Npn \_tl_analysis_b_char_aux:nww #1
5177 {
5178 \int_value:w `#1 \s_tl
5179 \exp_after:wN \_tl_analysis_b_normals:ww
5180 \int_value:w \int_eval:w - 1 +
5181 }
5182 \cs_new:Npn \_tl_analysis_b_cs:Nww #1
5183 {
5184 0 -1 \s_tl
5185 \_tl_analysis_cs_space_count:NN \_tl_analysis_b_cs_test:ww #1
5186 }
5187 \cs_new:Npn \_tl_analysis_b_cs_test:ww
5188 #1 \_tl_sep: #2 \_tl_sep: #3 \_tl_sep: #4 \_tl_sep:
5189 {
5190 \exp_after:wN \_tl_analysis_b_normals:ww
5191 \int_value:w \int_eval:w
5192 \if_int_compare:w #1 = \c_zero_int
5193 #3

```

```

5194 \else:
5195 \tex_skip:D \int_eval:n { #4 + #1 } \exp_stop_f:
5196 \fi:
5197 - #2
5198 \exp_after:wN \__tl_sep:
5199 \int_value:w \int_eval:n { #4 + #1 } \__tl_sep:
5200 }
5201 \group_begin:
5202 \char_set_catcode_other:N A
5203 \cs_new:Npn \__tl_analysis_b_special:w
5204 \fi: \__tl_analysis_b_normal:wwN 0 \__tl_sep: #1 \__tl_sep:
5205 {
5206 \fi:
5207 \if_int_compare:w #1 = \l__tl_analysis_index_int
5208 \exp_after:wN \prg_break:
5209 \fi:
5210 \tex_the:D \tex_toks:D #1 \s__tl
5211 \if_case:w \tex_gluestretch:D \tex_skip:D #1 \exp_stop_f:
5212 \token_to_str:N A
5213 \or: 1
5214 \or: 1
5215 \else: 2
5216 \fi:
5217 \if_int_odd:w \tex_gluestretch:D \tex_skip:D #1 \exp_stop_f:
5218 \exp_after:wN \__tl_analysis_b_special_char:wN \int_value:w
5219 \else:
5220 \exp_after:wN \__tl_analysis_b_special_space:w \int_value:w
5221 \fi:
5222 \int_eval:n { 1 + #1 } \exp_after:wN \__tl_sep:
5223 \token_to_str:N
5224 }
5225 \group_end:
5226 \cs_new:Npn \__tl_analysis_b_special_char:wN #1 \__tl_sep: #2
5227 {
5228 \int_value:w `#2 \s__tl
5229 \__tl_analysis_b_loop:w #1 \__tl_sep:
5230 }
5231 \use:e
5232 {
5233 \cs_new:Npn \exp_not:N \__tl_analysis_b_special_space:w
5234 #1 \exp_not:N \__tl_sep: \c_space_tl
5235 }
5236 {
5237 32 \s__tl
5238 \__tl_analysis_b_loop:w #1 \__tl_sep:
5239 }

```

```
5240 \cs_new_protected:Npn \tl_analysis_map_inline:Nn #1
5241 { \exp_args:No \tl_analysis_map_inline:nn #1 }
5242 \cs_new_protected:Npn \tl_analysis_map_inline:nn #1
5243 {
5244   \__tl_analysis:n {#1}
5245   \int_gincr:N \g__kernel_prg_map_int
5246   \exp_args:Nc \__tl_analysis_map:Nn
5247     { \__tl_analysis_map_inline_ \int_use:N \g__kernel_prg_map_int :wNw }
5248 }
5249 \cs_new_protected:Npn \__tl_analysis_map:Nn #1#2
5250 {
5251   \cs_gset_protected:Npn #1 ##1##2##3 {#2}
5252   \exp_after:wN \__tl_analysis_map:NwNw \exp_after:wN #1
5253   \g__tl_analysis_result_tl
5254   \s__tl { ? \tl_map_break: } \s__tl
5255   \prg_break_point:Nn \tl_map_break:
5256   { \int_gdecr:N \g__kernel_prg_map_int }
5257 }
5258 \cs_new_protected:Npn \__tl_analysis_map:NwNw #1 #2 \s__tl #3 #4 \s__tl
5259 {
5260   \use_none:n #3
5261   #1 {#2} {#4} {#3}
5262   \__tl_analysis_map:NwNw #1
5263 }
5264 \cs_new_protected:Npn \tl_analysis_show:N
5265 { \__tl_analysis_show:NNnnN \msg_show:nneeee \tl_show:N {} {} }
5266 \cs_new_protected:Npn \tl_analysis_log:N
5267 {
5268   \__tl_analysis_show:NNnnN \msg_log:nneeee \tl_log:N
5269   { \iow_newline: >~ . } { . }
5270 }
5271 \cs_new_protected:Npn \__tl_analysis_show:NNnnN #1#2#3#4#5
5272 {
5273   \tl_if_exist:NTF #5
5274   {
5275     \exp_args:No \__tl_analysis:n {#5}
5276     #1 { tl } { show-analysis }
5277     { \token_to_str:N #5 } { \__tl_analysis_show: } {#3} {#4}
5278   }
5279   { #2 #3 }
5280 }
5281 \cs_new_protected:Npn \tl_analysis_show:n
5282 { \__tl_analysis_show:Nnnn \msg_show:nneeee {} {} }
5283 \cs_new_protected:Npn \tl_analysis_log:n
5284 { \__tl_analysis_show:Nnnn \msg_log:nneeee { \iow_newline: >~ . } { . } }
5285 \cs_new_protected:Npn \__tl_analysis_show:Nnnn #1#2#3#4
```

```

5286 {
5287     \__tl_analysis:n {#4}
5288     #1 { tl } { show-analysis } { } { \__tl_analysis_show: } {#2} {#3}
5289 }
5290 \cs_new:Npn \__tl_analysis_show:
5291 {
5292     \exp_after:wN \__tl_analysis_show_loop:wNw \g__tl_analysis_result_tl
5293     \s__tl { ? \prg_break: } \s__tl
5294     \prg_break_point:
5295 }
5296 \cs_new:Npn \__tl_analysis_show_loop:wNw #1 \s__tl #2 #3 \s__tl
5297 {
5298     \use_none:n #2
5299     \iow_newline: > \use:nn { ~ } { ~ }
5300     \if_int_compare:w "#2 = \c_zero_int
5301         \exp_after:wN \__tl_analysis_show_cs:n
5302     \else:
5303         \if_int_compare:w "#2 = 13 \exp_stop_f:
5304             \exp_after:wN \exp_after:wN
5305             \exp_after:wN \__tl_analysis_show_active:n
5306         \else:
5307             \exp_after:wN \exp_after:wN
5308             \exp_after:wN \__tl_analysis_show_normal:n
5309         \fi:
5310     \fi:
5311     {#1}
5312     \__tl_analysis_show_loop:wNw
5313 }
5314 \cs_new:Npn \__tl_analysis_show_normal:n #1
5315 {
5316     \exp_after:wN \token_to_str:N #1 ~
5317     ( \exp_after:wN \token_to_meaning:N #1 )
5318 }
5319 \cs_new:Npn \__tl_analysis_show_value:N #1
5320 {
5321     \token_if_expandable:NF #1
5322     {
5323         \token_if_chardef:NTF #1 \prg_break: { }
5324         \token_if_mathchardef:NTF #1 \prg_break: { }
5325         \token_if_dim_register:NTF #1 \prg_break: { }
5326         \token_if_int_register:NTF #1 \prg_break: { }
5327         \token_if_skip_register:NTF #1 \prg_break: { }
5328         \token_if_toks_register:NTF #1 \prg_break: { }
5329         \use_none:nnn
5330         \prg_break_point:
5331         \use:n { \exp_after:wN = \tex_the:D #1 }

```

```

5332     }
5333 }
5334 \cs_new:Npn \__tl_analysis_show_cs:n #1
5335 { \exp_args:No \__tl_analysis_show_long:nn {#1} { control~sequence= } }
5336 \cs_new:Npn \__tl_analysis_show_active:n #1
5337 { \exp_args:No \__tl_analysis_show_long:nn {#1} { active~character= } }
5338 \cs_new:Npn \__tl_analysis_show_long:nn #1
5339 {
5340   \__tl_analysis_show_long_aux:oofn
5341   { \token_to_str:N #1 }
5342   { \token_to_meaning:N #1 }
5343   { \__tl_analysis_show_value:N #1 }
5344 }
5345 \cs_new:Npn \__tl_analysis_show_long_aux:nnnn #1#2#3#4
5346 {
5347   \int_compare:nNnTF
5348     { \str_count:n { #1 ~ ( #4 #2 #3 ) } }
5349     > { \l_iow_line_count_int - 3 }
5350     {
5351       \str_range:nnn { #1 ~ ( #4 #2 #3 ) } { 1 }
5352       {
5353         \l_iow_line_count_int - 3
5354         - \str_count:N \c__tl_analysis_show_etc_str
5355       }
5356       \c__tl_analysis_show_etc_str
5357     }
5358     { #1 ~ ( #4 #2 #3 ) }
5359 }
5360 \cs_generate_variant:Nn \__tl_analysis_show_long_aux:nnnn { oof }
5361 \cs_new:Npn \peek_analysis_map_break:
5362 { \prg_map_break:Nn \peek_analysis_map_break: { } }
5363 \cs_new:Npn \peek_analysis_map_break:n
5364 { \prg_map_break:Nn \peek_analysis_map_break: }
5365 \int_new:N \l__tl_peek_charcode_int
5366 \cs_new:Npn \__tl_analysis_char_arg:Nw
5367 {
5368   \if_meaning:w \l__tl_analysis_token \c_space_token
5369     \exp_after:wN \__tl_analysis_char_arg_aux:Nw
5370   \fi:
5371 }
5372 \cs_new:Npn \__tl_analysis_char_arg_aux:Nw #1 ~ { #1 { ~ } }
5373 \cs_new_protected:Npn \peek_analysis_map_inline:n #1
5374 {
5375   \group_align_safe_begin:
5376   \int_gincr:N \g__kernel_prg_map_int
5377   \cs_set_protected:cpn

```



```
5378 { __tl_analysis_map_ \int_use:N \g__kernel_prg_map_int :nnN } 25378
5379 ##1##2##3 25379
5380 { 25380
5381 \group_end: 25381
5382 #1 25382
5383 \__tl_peek_analysis_loop:NNn 25383
5384 \prg_break_point:Nn \peek_analysis_map_break: 25384
5385 { 25385
5386 \int_gdecr:N \g__kernel_prg_map_int 25386
5387 \group_align_safe_end: 25387
5388 } 25388
5389 } 25389
5390 \__tl_peek_analysis_loop:NNn ? ? ? 25390
5391 } 25391
5392 \cs_new_protected:Npn \__tl_peek_analysis_loop:NNn #1#2#3 25392
5393 { 25393
5394 \group_begin: 25394
5395 \tl_set:Nc \l__tl_peek_code_tl 25395
5396 { 25396
5397 \exp_not:c 25397
5398 { __tl_analysis_map_ \int_use:N \g__kernel_prg_map_int :nnN } 25398
5399 } 25399
5400 \int_set:Nn \tex_escapechar:D { ``\ } 25400
5401 \peek_after:Nw \__tl_peek_analysis_test: 25401
5402 } 25402
5403 \cs_new_protected:Npn \__tl_peek_analysis_test: 25403
5404 { 25404
5405 \if_case:w 25405
5406 \if_catcode:w \exp_not:N \l_peek_token { \c_max_int \fi: 25406
5407 \if_catcode:w \exp_not:N \l_peek_token } \c_max_int \fi: 25407
5408 \if_meaning:w \l_peek_token \c_space_token \c_max_int \fi: 25408
5409 \exp_after:wN \if_meaning:w \exp_not:N \l_peek_token \l_peek_token 25409
5410 \c_one_int 25410
5411 \fi: 25411
5412 \c_zero_int 25412
5413 \exp_after:wN \exp_after:wN 25413
5414 \exp_after:wN \__tl_peek_analysis_exp:N 25414
5415 \exp_after:wN \exp_not:N 25415
5416 \or: 25416
5417 \exp_after:wN \__tl_peek_analysis_nonexp:N 25417
5418 \else: 25418
5419 \exp_after:wN \__tl_peek_analysis_special: 25419
5420 \fi: 25420
5421 } 25421
5422 \cs_new_protected:Npn \__tl_peek_analysis_exp:N #1 25422
5423 { 25423
```

```
5424 \cs_set_nopar:Npe \l__tl_peek_code_tl 25424
5425 { 25425
5426 \tex_let:D \exp_not:N #1 \l__tl_peek_code_tl 25426
5427 \l__tl_peek_code_tl 25427
5428 { 25428
5429 \exp_not:n { \__kernel_exp_not:w \exp_after:wN } 25429
5430 { \exp_not:N \exp_not:N \exp_not:N #1 } 25430
5431 } 25431
5432 \exp_after:wN \__tl_peek_analysis_exp_aux:Nw 25432
5433 \token_to_str:N #1 { } \s__tl 25433
5434 } 25434
5435 \l__tl_peek_code_tl 25435
5436 } 25436
5437 \cs_new:Npe \__tl_peek_analysis_exp_aux:Nw #1#2 \s__tl 25437
5438 { 25438
5439 \exp_not:N \if_meaning:w \scan_stop: #2 \scan_stop: 25439
5440 { \exp_not:N \int_value:w `#1 ~ } \token_to_str:N D 25440
5441 \exp_not:N \else: 25441
5442 { -1 } 0 25442
5443 \exp_not:N \fi: 25443
5444 } 25444
5445 \cs_new_protected:Npn \__tl_peek_analysis_nonexp:N #1 25445
5446 { 25446
5447 \if_charcode:w 25447
5448 \scan_stop: 25448
5449 \exp_after:wN \use_none:n \token_to_str:N #1 \prg_do_nothing: 25449
5450 \scan_stop: 25450
5451 \exp_after:wN \__tl_peek_analysis_char:N 25451
5452 \else: 25452
5453 \exp_after:wN \__tl_peek_analysis_cs:N 25453
5454 \fi: 25454
5455 #1 25455
5456 } 25456
5457 \cs_new_protected:Npn \__tl_peek_analysis_cs:N #1 25457
5458 { \l__tl_peek_code_tl { \exp_not:n {#1} } { -1 } 0 } 25458
5459 \group_begin: 25459
5460 \char_set_active_eq:NN \__scan_stop: 25460
5461 \cs_new_protected:Npe \__tl_peek_analysis_char:N #1 25461
5462 { 25462
5463 \cs_set_eq:NN 25463
5464 \char_generate:nn { 32 } { 13 } 25464
5465 \exp_not:N \tex_undefined:D 25465
5466 \tex_lccode:D `#1 = 32 \exp_stop_f: 25466
5467 \tex_lowercase:D 25467
5468 { 25468
5469 \tl_put_right:Ne \exp_not:N \l__tl_peek_code_tl 25469
```

```

5470         { \exp_not:n { \__tl_analysis_b_char:Nn \use_none:n } {#1} }
5471     }
5472 \exp_not:n
5473 {
5474     \exp_after:wN \__tl_peek_analysis_char:w
5475     \int_value:w
5476 }
5477 `#1
5478 \exp_not:n { \exp_after:wN \s__tl \l__tl_peek_code_tl }
5479 #1
5480 }
5481 \group_end:
5482 \cs_new_protected:Npn \__tl_peek_analysis_char:w #1 \s__tl #2#3#4
5483 {
5484     \if_charcode:w 6 #3
5485     \else:
5486         \if_charcode:w D #3
5487         \else:
5488             \exp_args:NNNo
5489             \fi:
5490         \fi:
5491         #2 { \exp_not:n {#4} } {#1} #3
5492     }
5493 \cs_new_protected:Npn \__tl_peek_analysis_special:
5494 {
5495     \tex_let:D \l__tl_analysis_token = ~ \l_peek_token
5496     \int_set:Nn \l__tl_peek_charcode_int
5497     { \__tl_analysis_extract_charcode: }
5498     \if_int_compare:w \l__tl_peek_charcode_int = \tex_escapechar:D
5499     \int_set:Nn \tex_escapechar:D { `\_ }
5500     \fi:
5501     \char_set_active_eq:nN { \l__tl_peek_charcode_int } \scan_stop:
5502     \char_set_active_eq:nN { \tex_escapechar:D } \scan_stop:
5503     \cs_set_eq:cN { } \scan_stop:
5504     \tex_futurelet:D \l__tl_analysis_token
5505     \__tl_peek_analysis_retest:
5506 }
5507 \cs_new_protected:Npn \__tl_peek_analysis_retest:
5508 {
5509     \if_meaning:w \l__tl_analysis_token \scan_stop:
5510         \exp_after:wN \__tl_peek_analysis_nonexp:N
5511     \else:
5512         \exp_after:wN \__tl_peek_analysis_str:
5513     \fi:
5514 }
5515 \cs_new_protected:Npn \__tl_peek_analysis_str:

```

```

5516 {
5517     \exp_after:wN \tex_futurelet:D
5518     \exp_after:wN \l__tl_analysis_token
5519     \exp_after:wN \__tl_peek_analysis_str:w
5520     \token_to_str:N
5521 }
5522 \cs_new_protected:Npn \__tl_peek_analysis_str:w
5523 { \__tl_analysis_char_arg:Nw \__tl_peek_analysis_str:n }
5524 \cs_new_protected:Npn \__tl_peek_analysis_str:n #1
5525 {
5526     \int_case:nnF { `#1 }
5527     {
5528         { \l__tl_peek_charcode_int }
5529         { \__tl_peek_analysis_explicit:n {#1} }
5530         { \tex_escapechar:D } { \__tl_peek_analysis_escape: }
5531     }
5532     { \__tl_peek_analysis_active_str:n {#1} }
5533 }
5534 \cs_new_protected:Npn \__tl_peek_analysis_active_str:n #1
5535 {
5536     \tl_put_right:Ne \l__tl_peek_code_tl
5537     {
5538         { \char_generate:nn { `#1 } { 13 } }
5539         { \int_value:w `#1 }
5540         \token_to_str:N D
5541     }
5542     \l__tl_peek_code_tl
5543 }
5544 \cs_new_protected:Npn \__tl_peek_analysis_explicit:n #1
5545 {
5546     \tl_put_right:Ne \l__tl_peek_code_tl
5547     {
5548         \if_meaning:w \l_peek_token \c_space_token
5549             { ~ } { 32 } \token_to_str:N A
5550         \else:
5551             \if_catcode:w \l_peek_token \c_group_begin_token
5552             {
5553                 \exp_not:N \exp_after:wN
5554                 \char_generate:nn { `#1 } { 1 }
5555                 \exp_not:N \if_false:
5556                 \if_false: { \fi: }
5557                 \exp_not:N \fi:
5558             }
5559             { \int_value:w `#1 }
5560             1
5561         \else:

```

```

5562         {
5563             \exp_not:N \if_false:
5564             { \if_false: } \fi:
5565             \exp_not:N \fi:
5566             \char_generate:nn { `#1 } { 2 }
5567         }
5568         { \int_value:w `#1 }
5569         2
5570         \fi:
5571         \fi:
5572     }
5573     \l__tl_peek_code_tl
5574 }
5575 \cs_new_protected:Npn \__tl_peek_analysis_escape:
5576 {
5577     \tl_clear:N \l__tl_internal_a_tl
5578     \tex_futurelet:D \l__tl_analysis_token
5579     \__tl_peek_analysis_collect:w
5580 }
5581 \cs_new_protected:Npn \__tl_peek_analysis_collect:w
5582 { \__tl_analysis_char_arg:Nw \__tl_peek_analysis_collect:n }
5583 \cs_new_protected:Npn \__tl_peek_analysis_collect:n #1
5584 {
5585     \tl_put_right:Nn \l__tl_internal_a_tl {#1}
5586     \__tl_peek_analysis_collect_loop:
5587 }
5588 \cs_new_protected:Npn \__tl_peek_analysis_collect_loop:
5589 {
5590     \exp_after:wN \if_meaning:w
5591     \cs:w
5592     \if_cs_exist:w \l__tl_internal_a_tl \cs_end:
5593     \l__tl_internal_a_tl
5594     \else:
5595         c_one % anything short
5596         \fi:
5597         \cs_end:
5598         \l_peek_token
5599         \__tl_peek_analysis_collect_end:NNNN
5600     \fi:
5601     \tex_futurelet:D \l__tl_analysis_token
5602     \__tl_peek_analysis_collect:w
5603 }
5604 \cs_new_protected:Npn \__tl_peek_analysis_collect_end:NNNN #1#2#3#4
5605 {
5606     #1
5607     \tl_put_right:Ne \l__tl_peek_code_tl

```

```
5608     {
5609         { \exp_not:N \exp_not:n { \exp_not:c { \l__tl_internal_a_tl } } }
5610         { -1 }
5611         0
5612     }
5613     \l__tl_peek_code_tl
5614 }
5615 \tl_const:Nc \c__tl_analysis_show_etc_str % (
5616     { \token_to_str:N \ETC.) }
5617 \msg_new:nnn { tl } { show-analysis }
5618 {
5619     The~token~list~ \tl_if_empty:nF {#1} { #1 ~ }
5620     \tl_if_empty:nTF {#2}
5621     { is~empty #3 }
5622     { contains~the~tokens: #2 #4 }
5623 }
5624 %% File: l3benchmark.dtx
5625 \fp_new:N \g_benchmark_duration_target_fp
5626 \fp_gset:Nn \g_benchmark_duration_target_fp { 1 }
5627 \int_new:N \g__benchmark_nesting_int
5628 \cs_new_protected:Npn \__benchmark_raw:nN #1
5629 {
5630     \int_gincr:N \g__benchmark_nesting_int
5631     \exp_args:Nc \__benchmark_raw_aux:N
5632     { g__benchmark_ \int_use:N \g__benchmark_nesting_int _int }
5633     \__benchmark_raw_aux:
5634     #1
5635     \__benchmark_raw_end:N
5636 }
5637 \cs_new_protected:Npn \__benchmark_raw_aux:N #1
5638 {
5639     \int_gzero_new:N #1
5640     \cs_gset_protected:Npn \__benchmark_raw_aux: { \int_gset:Nn #1 { \sys_timer: } }
5641 }
5642 \cs_new_protected:Npn \__benchmark_raw_end:N #1
5643 {
5644     \int_gset:Nn #1
5645     {
5646         \sys_timer: -
5647         \int_use:c { g__benchmark_ \int_use:N \g__benchmark_nesting_int _int }
5648     }
5649     \int_gdecr:N \g__benchmark_nesting_int
5650 }
5651 \cs_new_eq:NN \__benchmark_tmp:w ?
5652 \cs_new_protected:Npn \__benchmark_raw_replicate:nnN #1
5653 {
```

```
5654 \int_compare:nNnTF {#1} > { 500000 }
5655 { \__benchmark_raw_replicate_large:nnN {#1} }
5656 { \__benchmark_raw_replicate_small:nnN {#1} }
5657 }
5658 \cs_new_protected:Npn \__benchmark_raw_replicate_large:nnN #1#2
5659 {
5660   \cs_set:Npe \__benchmark_tmp:w ##1 { \prg_replicate:nn { 5000 } {##1} }
5661   \exp_args:Nno \__benchmark_raw_replicate:nnN { #1 / 5000 }
5662   { \__benchmark_tmp:w {#2} }
5663 }
5664 \cs_new_protected:Npn \__benchmark_raw_replicate_small:nnN #1#2
5665 {
5666   \cs_set:Npe \__benchmark_tmp:w ##1##2 { \prg_replicate:nn {#1} {##1} }
5667   \__benchmark_raw:nN { \__benchmark_tmp:w {#2} { } } \g__benchmark_time_int
5668   \exp_args:No \__benchmark_raw_replicate_aux:nnN
5669   { \int_use:N \g__benchmark_time_int } {#2}
5670 }
5671 \cs_new_protected:Npn \__benchmark_raw_replicate_aux:nnN #1#2#3
5672 {
5673   \__benchmark_raw:nN { \__benchmark_tmp:w { } {#2} } \g__benchmark_time_int
5674   \int_gset:Nn #3 { #1 - \g__benchmark_time_int }
5675   \cs_set_eq:NN \__benchmark_tmp:w \prg_do_nothing:
5676 }
5677 \fp_new:N \g__benchmark_time_fp
5678 \fp_new:N \g__benchmark_ops_fp
5679 \int_new:N \g__benchmark_duration_int
5680 \int_new:N \g__benchmark_time_int
5681 \int_new:N \g__benchmark_time_a_int
5682 \int_new:N \g__benchmark_time_b_int
5683 \int_new:N \g__benchmark_time_c_int
5684 \int_new:N \g__benchmark_time_d_int
5685 \int_new:N \g__benchmark_repeat_int
5686 \tl_new:N \g__benchmark_code_tl
5687 \cs_new_protected:Npn \benchmark_once:n #1
5688 {
5689   \benchmark_once_silent:n {#1}
5690   \__benchmark_display:
5691 }
5692 \cs_new_protected:Npn \benchmark_once_silent:n #1
5693 {
5694   \__benchmark_measure_op:
5695   \__benchmark_raw:nN {#1} \g__benchmark_time_int
5696   \fp_gset:Nn \g__benchmark_time_fp { \g__benchmark_time_int / 65536 }
5697   \fp_gset:Nn \g__benchmark_ops_fp { \g__benchmark_time_fp / \g__benchmark_one_op_fp }
5698 }
5699 \cs_new_protected:Npn \benchmark:n #1
```



```

5700 {
5701     \benchmark_silent:n {#1}
5702     \__benchmark_display:
5703 }
5704 \cs_new_protected:Npn \benchmark_silent:n #1
5705 {
5706     \__benchmark_measure_op:
5707     \tl_gset:Nn \g__benchmark_code_tl {#1}
5708     \__benchmark_aux:
5709     \fp_gset:Nn \g_benchmark_ops_fp { \g_benchmark_time_fp / \g__benchmark_one_op_fp }
5710 }
5711 \cs_new_protected:Npn \__benchmark_aux:
5712 {
5713     \int_gset:Nn \g__benchmark_repeat_int { 1 }
5714     \__benchmark_raw:nN { \g__benchmark_code_tl } \g__benchmark_time_int
5715     \int_compare:nNf \g__benchmark_time_int < { \g__benchmark_duration_int / 2 }
5716         { \prg_break: }
5717     \bool_until_do:nn
5718     {
5719         \int_compare_p:nNn \g__benchmark_time_int > { \g__benchmark_duration_int / 32 }
5720         || \int_compare_p:nNn \g__benchmark_repeat_int > { \c_max_int / 4 }
5721     }
5722     {
5723         \int_gset:Nn \g__benchmark_repeat_int { 4 * \g__benchmark_repeat_int }
5724         \__benchmark_run:N \g__benchmark_time_int
5725     }
5726     \int_gset:Nn \g__benchmark_repeat_int
5727     {
5728         \fp_to_int:n
5729         {
5730             max ( 1 , min ( \c_max_int ,
5731                 \g__benchmark_duration_int * \g__benchmark_repeat_int /
5732                 \int_eval:n { 4 * \g__benchmark_time_int } ) )
5733         }
5734     }
5735     \int_compare:nNfTF \g__benchmark_repeat_int = 1
5736         { \int_gset_eq:NN \g__benchmark_time_a_int \g__benchmark_time_int }
5737         { \__benchmark_run:N \g__benchmark_time_a_int }
5738     \__benchmark_run:N \g__benchmark_time_b_int
5739     \__benchmark_run:N \g__benchmark_time_c_int
5740     \__benchmark_run:N \g__benchmark_time_d_int
5741     \int_gset:Nn \g__benchmark_time_int
5742     {
5743         \int_min:nn
5744         { \int_min:nn \g__benchmark_time_a_int \g__benchmark_time_b_int }
5745         { \int_min:nn \g__benchmark_time_c_int \g__benchmark_time_d_int }

```

```

5746 }
5747 \prg_break_point:
5748 \int_compare:nNnT \g__benchmark_time_int < 3 { \int_gzero:N \g__benchmark_time_int }
5749 \fp_gset:Nn \g_benchmark_time_fp
5750 { \g__benchmark_time_int / \g__benchmark_repeat_int / 65536 }
5751 }
5752 \cs_new_protected:Npn \__benchmark_run:N
5753 { \exp_args:NNo \__benchmark_raw_replicate:nnN \g__benchmark_repeat_int {
5754 \g__benchmark_code_tl } }
5755 \fp_new:N \g__benchmark_one_op_fp
5756 \cs_new_protected:Npn \__benchmark_measure_op:
5757 {
5758 \int_gset:Nn \g__benchmark_duration_int
5759 { \fp_to_int:n { 65536 * \g_benchmark_duration_target_fp } / 4 }
5760 \tl_gset:Nn \g__benchmark_code_tl
5761 { \int_gadd:Nn \g__benchmark_duration_int { 0 } }
5762 \__benchmark_aux:
5763 \fp_gset:Nn \g__benchmark_one_op_fp { max(\g_benchmark_time_fp, 1e-8) }
5764 \int_gset:Nn \g__benchmark_duration_int
5765 { \fp_to_int:n { 65536 * \g_benchmark_duration_target_fp } }
5766 }
5767 \cs_new:Npn \__benchmark_fp_to_tl:N #1
5768 {
5769 \fp_compare:nTF { abs(#1) < 1000 }
5770 { \fp_to_tl:n { round(#1, 2 - logb(#1)) } }
5771 {
5772 \exp_args:Nf \__benchmark_fp_to_tl_aux:nN
5773 { \fp_to_int:n { logb(#1) } } #1
5774 }
5775 }
5776 \cs_new:Npn \__benchmark_fp_to_tl_aux:nN #1#2
5777 { \fp_to_tl:n { round(#2 * 1e-#1, 2) } e#1 }
5778 \cs_new_protected:Npn \__benchmark_display:
5779 {
5780 \iow_term:e
5781 {
5782 \__benchmark_fp_to_tl:N \g_benchmark_time_fp \c_space_tl seconds \c_space_tl
5783 ( \__benchmark_fp_to_tl:N \g_benchmark_ops_fp \c_space_tl ops)
5784 }
5785 }
5786 \int_new:N \g__benchmark_tictoc_int
5787 \seq_new:N \g__benchmark_tictoc_seq
5788 \tl_new:N \l__benchmark_tictoc_pop_tl
5789 \cs_new:Npn \__benchmark_tictoc_prefix:
5790 {
5791 (l3benchmark) \c_space_tl

```

```

5791 + \prg_replicate:nn { \g__benchmark_tictoc_int } { -- } \c_space_tl
5792 }
5793 \cs_new_protected:Npn \benchmark_tic:
5794 {
5795   \iow_term:e { \__benchmark_tictoc_prefix: TIC }
5796   \exp_args:NNf \seq_gput_right:Nn \g__benchmark_tictoc_seq { \sys_timer: }
5797   \int_gincr:N \g__benchmark_tictoc_int
5798 }
5799 \cs_new:Npn \benchmark_toc:
5800 {
5801   \seq_gpop_right:NNTF \g__benchmark_tictoc_seq \l__benchmark_tictoc_pop_tl
5802   { \__benchmark_toc: }
5803   { \msg_error:nn { benchmark } { toc-first } }
5804 }
5805 \cs_new_protected:Npn \__benchmark_toc:
5806 {
5807   \int_gdecr:N \g__benchmark_tictoc_int
5808   \fp_gset:Nn \g_benchmark_time_fp
5809   { ( \sys_timer: - \l__benchmark_tictoc_pop_tl ) / 65536 }
5810   \iow_term:e
5811   {
5812     \__benchmark_tictoc_prefix:
5813     TOC: \c_space_tl
5814     \__benchmark_fp_to_tl:N \g_benchmark_time_fp \c_space_tl s
5815   }
5816 }
5817 \msg_new:nnn { benchmark } { toc-first }
5818 {
5819   \token_to_str:N \benchmark_toc: \c_space_tl without~
5820   \token_to_str:N \benchmark_tic: \c_space_tl !
5821 }
5822 %% File: l3regex.dtx
5823 \cs_new_eq:NN \__regex_int_eval:w \tex_numexpr:D
5824 \cs_new_eq:NN \__regex_sep: \__kernel_int_sep:
5825 \cs_new_protected:Npn \__regex_standard_escapechar:
5826 { \int_set:Nn \tex_escapechar:D { `` } }
5827 \cs_new:Npn \__regex_toks_use:w { \tex_the:D \tex_toks:D }
5828 \cs_new_protected:Npn \__regex_toks_clear:N #1
5829 { \tex_toks:D #1 = { } }
5830 \cs_new_eq:NN \__regex_toks_set:Nn \tex_toks:D
5831 \cs_new_protected:Npn \__regex_toks_set:No #1
5832 { \tex_toks:D #1 = \exp_after:wN }
5833 \cs_new_protected:Npn \__regex_toks_memcpy:NNn #1#2#3
5834 {
5835   \prg_replicate:nn {#3}
5836   {

```

```
5837         \tex_toks:D #1 = \tex_toks:D #2
5838         \int_incr:N #1
5839         \int_incr:N #2
5840     }
5841 }
5842 \cs_if_exist:NTF \tex_etokspre:D
5843 { \cs_new_eq:NN \__regex_toks_put_left:Ne \tex_etokspre:D }
5844 {
5845     \cs_new_protected:Npn \__regex_toks_put_left:Ne #1#2
5846     { \tex_toks:D #1 = \tex_expanded:D {{ #2 \tex_the:D \tex_toks:D #1 }} }
5847 }
5848 \cs_if_exist:NTF \tex_etoksapp:D
5849 { \cs_new_eq:NN \__regex_toks_put_right:Ne \tex_etoksapp:D }
5850 {
5851     \cs_new_protected:Npn \__regex_toks_put_right:Ne #1#2
5852     { \tex_toks:D #1 = \tex_expanded:D {{ \tex_the:D \tex_toks:D #1 #2 }} }
5853 }
5854 \cs_if_exist:NTF \tex_toksapp:D
5855 { \cs_new_eq:NN \__regex_toks_put_right:Nn \tex_toksapp:D }
5856 {
5857     \cs_new_protected:Npn \__regex_toks_put_right:Nn #1#2
5858     { \tex_toks:D #1 = \exp_after:wN { \tex_the:D \tex_toks:D #1 #2 } }
5859 }
5860 \cs_new:Npn \__regex_curr_cs_to_str:
5861 {
5862     \exp_after:wN \exp_after:wN \exp_after:wN \cs_to_str:N
5863     \l__regex_curr_token_tl
5864 }
5865 \cs_new:Npn \__regex_intarray_item:NnF #1#2
5866 { \exp_args:No \__regex_intarray_item_aux:nNF { \tex_the:D \__regex_int_eval:w #2 }
5867 #1 }
5868 \cs_new:Npn \__regex_intarray_item_aux:nNF #1#2
5869 {
5870     \if_int_compare:w #1 > \c_zero_int
5871     \exp_after:wN \use_ii:nnn
5872     \fi:
5873     \use_ii:nn { \__kernel_intarray_item:Nn #2 {#1} }
5874 }
5875 \cs_new:Npn \__regex_maplike_break:
5876 { \prg_map_break:Nn \__regex_maplike_break: { } }
5877 \cs_new:Npn \__regex_tl_odd_items:n #1 { \__regex_tl_even_items:n { ? #1 } }
5878 \cs_new:Npn \__regex_tl_even_items:n #1
5879 {
5880     \__regex_tl_even_items_loop:nn #1 \q__regex_nil \q__regex_nil
5881     \prg_break_point:
5882 }
```

```
5882 \cs_new:Npn \__regex_tl_even_items_loop:nn #1#2
5883 {
5884     \__regex_use_none_delimit_by_q_nil:w #2 \prg_break: \q__regex_nil
5885     { \exp_not:n {#2} }
5886     \__regex_tl_even_items_loop:nn
5887 }
5888 \cs_new:Npn \__regex_tmp:w { }
5889 \tl_new:N \l__regex_internal_a_tl
5890 \tl_new:N \l__regex_internal_b_tl
5891 \int_new:N \l__regex_internal_a_int
5892 \int_new:N \l__regex_internal_b_int
5893 \int_new:N \l__regex_internal_c_int
5894 \bool_new:N \l__regex_internal_bool
5895 \seq_new:N \l__regex_internal_seq
5896 \tl_new:N \g__regex_internal_tl
5897 \tl_new:N \l__regex_build_tl
5898 \tl_const:Nn \c__regex_no_match_regex
5899 {
5900     \__regex_branch:n
5901     { \__regex_class:NnnnN \c_true_bool { } { 1 } { 0 } \c_true_bool }
5902 }
5903 \int_new:N \l__regex_balance_int
5904 \int_const:Nn \c__regex_ascii_min_int { 0 }
5905 \int_const:Nn \c__regex_ascii_max_control_int { 31 }
5906 \int_const:Nn \c__regex_ascii_max_int { 127 }
5907 \int_const:Nn \c__regex_ascii_lower_int { `a - `A }
5908 \quark_new:N \q__regex_recursion_stop
5909 \quark_new:N \q__regex_nil
5910 \cs_new:Npn \__regex_use_none_delimit_by_q_recursion_stop:w
5911     #1 \q__regex_recursion_stop { }
5912 \cs_new:Npn \__regex_use_i_delimit_by_q_recursion_stop:nw
5913     #1 #2 \q__regex_recursion_stop {#1}
5914 \cs_new:Npn \__regex_use_none_delimit_by_q_nil:w #1 \q__regex_nil { }
5915 \__kernel_quark_new_conditional:Nn \__regex_quark_if_nil:N { F }
5916 \cs_new_protected:Npn \__regex_break_true:w
5917     #1 \__regex_break_point:TF #2 #3 {#2}
5918 \cs_new_protected:Npn \__regex_break_point:TF #1 #2 { #2 }
5919 \cs_new_protected:Npn \__regex_item_reverse:n #1
5920 {
5921     #1
5922     \__regex_break_point:TF { } \__regex_break_true:w
5923 }
5924 \cs_new_protected:Npn \__regex_item_caseful_equal:n #1
5925 {
5926     \if_int_compare:w #1 = \l__regex_curr_char_int
5927     \exp_after:wN \__regex_break_true:w
```

```
5928 \fi:
5929 }
5930 \cs_new_protected:Npn \__regex_item_caseful_range:nn #1 #2
5931 {
5932   \reverse_if:N \if_int_compare:w #1 > \l__regex_curr_char_int
5933   \reverse_if:N \if_int_compare:w #2 < \l__regex_curr_char_int
5934   \exp_after:wN \exp_after:wN \exp_after:wN \__regex_break_true:w
5935   \fi:
5936   \fi:
5937 }
5938 \cs_new_protected:Npn \__regex_item_caseless_equal:n #1
5939 {
5940   \if_int_compare:w #1 = \l__regex_curr_char_int
5941   \exp_after:wN \__regex_break_true:w
5942   \fi:
5943   \__regex_maybe_compute_ccc:
5944   \if_int_compare:w #1 = \l__regex_case_changed_char_int
5945   \exp_after:wN \__regex_break_true:w
5946   \fi:
5947 }
5948 \cs_new_protected:Npn \__regex_item_caseless_range:nn #1 #2
5949 {
5950   \reverse_if:N \if_int_compare:w #1 > \l__regex_curr_char_int
5951   \reverse_if:N \if_int_compare:w #2 < \l__regex_curr_char_int
5952   \exp_after:wN \exp_after:wN \exp_after:wN \__regex_break_true:w
5953   \fi:
5954   \fi:
5955   \__regex_maybe_compute_ccc:
5956   \reverse_if:N \if_int_compare:w #1 > \l__regex_case_changed_char_int
5957   \reverse_if:N \if_int_compare:w #2 < \l__regex_case_changed_char_int
5958   \exp_after:wN \exp_after:wN \exp_after:wN \__regex_break_true:w
5959   \fi:
5960   \fi:
5961 }
5962 \cs_new_protected:Npn \__regex_compute_case_changed_char:
5963 {
5964   \int_set_eq:NN \l__regex_case_changed_char_int \l__regex_curr_char_int
5965   \if_int_compare:w \l__regex_curr_char_int > `Z \exp_stop_f:
5966   \if_int_compare:w \l__regex_curr_char_int > `z \exp_stop_f: \else:
5967   \if_int_compare:w \l__regex_curr_char_int < `a \exp_stop_f: \else:
5968   \int_sub:Nn \l__regex_case_changed_char_int \c__regex_ascii_lower_int
5969   \fi:
5970   \fi:
5971   \else:
5972   \if_int_compare:w \l__regex_curr_char_int < `A \exp_stop_f: \else:
5973   \int_add:Nn \l__regex_case_changed_char_int \c__regex_ascii_lower_int
```



```

5974 \fi:
5975 \fi:
5976 \cs_set_eq:NN \__regex_maybe_compute_ccc: \prg_do_nothing:
5977 }
5978 \cs_new_eq:NN \__regex_maybe_compute_ccc: \__regex_compute_case_changed_char:
5979 \cs_new_eq:NN \__regex_item_equal:n ?
5980 \cs_new_eq:NN \__regex_item_range:nn ?
5981 \cs_new_protected:Npn \__regex_item_catcode:
5982 {
5983 "
5984 \if_case:w \l__regex_curr_catcode_int
5985 1 \or: 4 \or: 10 \or: 40
5986 \or: 100 \or: \or: 1000 \or: 4000
5987 \or: 10000 \or: \or: 100000 \or: 400000
5988 \or: 1000000 \or: 4000000 \else: 1*0
5989 \fi:
5990 }
5991 \prg_new_protected_conditional:Npnn \__regex_item_catcode:n #1 { T }
5992 {
5993 \if_int_odd:w \__regex_int_eval:w #1 / \__regex_item_catcode: \scan_stop:
5994 \prg_return_true:
5995 \else:
5996 \prg_return_false:
5997 \fi:
5998 }
5999 \cs_new_protected:Npn \__regex_item_catcode_reverse:nT #1#2
6000 { \__regex_item_catcode:nT {#1} { \__regex_item_reverse:n {#2} } }
6001 \cs_new_protected:Npn \__regex_item_exact:nn #1#2
6002 {
6003 \if_int_compare:w #1 = \l__regex_curr_catcode_int
6004 \if_int_compare:w #2 = \l__regex_curr_char_int
6005 \exp_after:wN \exp_after:wN \exp_after:wN \__regex_break_true:w
6006 \fi:
6007 \fi:
6008 }
6009 \cs_new_protected:Npn \__regex_item_exact_cs:n #1
6010 {
6011 \int_compare:nNnTF \l__regex_curr_catcode_int = \c_zero_int
6012 {
6013 \__kernel_tl_set:Nx \l__regex_internal_a_tl
6014 { \scan_stop: \__regex_curr_cs_to_str: \scan_stop: }
6015 \tl_if_in:noTF { \scan_stop: #1 \scan_stop: }
6016 \l__regex_internal_a_tl
6017 { \__regex_break_true:w } { }
6018 }
6019 { }

```



```
6020 }
6021 \cs_new_protected:Npn \__regex_item_cs:n #1
6022 {
6023   \int_compare:nNnT \l__regex_curr_catcode_int = \c_zero_int
6024   {
6025     \group_begin:
6026       \__regex_single_match:
6027       \__regex_disable_submatches:
6028       \__regex_build_for_cs:n {#1}
6029       \bool_set_eq:NN \l__regex_saved_success_bool
6030       \g__regex_success_bool
6031       \exp_args:Ne \__regex_match_cs:n { \__regex_curr_cs_to_str: }
6032       \if_meaning:w \c_true_bool \g__regex_success_bool
6033       \group_insert_after:N \__regex_break_true:w
6034       \fi:
6035       \bool_gset_eq:NN \g__regex_success_bool
6036       \l__regex_saved_success_bool
6037     \group_end:
6038   }
6039 }
6040 \cs_new_protected:Npn \__regex_prop_d:
6041 { \__regex_item_caseful_range:nn { `0 } { `9 } }
6042 \cs_new_protected:Npn \__regex_prop_h:
6043 {
6044   \__regex_item_caseful_equal:n { `\_ }
6045   \__regex_item_caseful_equal:n { `^~I }
6046 }
6047 \cs_new_protected:Npn \__regex_prop_s:
6048 {
6049   \__regex_item_caseful_equal:n { `\_ }
6050   \__regex_item_caseful_equal:n { `^~I }
6051   \__regex_item_caseful_equal:n { `^~J }
6052   \__regex_item_caseful_equal:n { `^~L }
6053   \__regex_item_caseful_equal:n { `^~M }
6054 }
6055 \cs_new_protected:Npn \__regex_prop_v:
6056 { \__regex_item_caseful_range:nn { `^~J } { `^~M } } % lf, vtab, ff, cr
6057 \cs_new_protected:Npn \__regex_prop_w:
6058 {
6059   \__regex_item_caseful_range:nn { `a } { `z }
6060   \__regex_item_caseful_range:nn { `A } { `Z }
6061   \__regex_item_caseful_range:nn { `0 } { `9 }
6062   \__regex_item_caseful_equal:n { `_ }
6063 }
6064 \cs_new_protected:Npn \__regex_prop_N:
6065 {
```

```
6066     \__regex_item_reverse:n
6067     { \__regex_item_caseful_equal:n { `\\~J } }
6068 }
6069 \cs_new_protected:Npn \__regex_posix_alnum:
6070 { \__regex_posix_alpha: \__regex_posix_digit: }
6071 \cs_new_protected:Npn \__regex_posix_alpha:
6072 { \__regex_posix_lower: \__regex_posix_upper: }
6073 \cs_new_protected:Npn \__regex_posix_ascii:
6074 {
6075     \__regex_item_caseful_range:nn
6076     \c__regex_ascii_min_int
6077     \c__regex_ascii_max_int
6078 }
6079 \cs_new_eq:NN \__regex_posix_blank: \__regex_prop_h:
6080 \cs_new_protected:Npn \__regex_posix_cntrl:
6081 {
6082     \__regex_item_caseful_range:nn
6083     \c__regex_ascii_min_int
6084     \c__regex_ascii_max_control_int
6085     \__regex_item_caseful_equal:n \c__regex_ascii_max_int
6086 }
6087 \cs_new_eq:NN \__regex_posix_digit: \__regex_prop_d:
6088 \cs_new_protected:Npn \__regex_posix_graph:
6089 { \__regex_item_caseful_range:nn { `! } { `~ } }
6090 \cs_new_protected:Npn \__regex_posix_lower:
6091 { \__regex_item_caseful_range:nn { `a } { `z } }
6092 \cs_new_protected:Npn \__regex_posix_print:
6093 { \__regex_item_caseful_range:nn { `\_ } { `~ } }
6094 \cs_new_protected:Npn \__regex_posix_punct:
6095 {
6096     \__regex_item_caseful_range:nn { `! } { `/ }
6097     \__regex_item_caseful_range:nn { `: } { `@ }
6098     \__regex_item_caseful_range:nn { `[ } { `` }
6099     \__regex_item_caseful_range:nn { `{ } { `~ }
6100 }
6101 \cs_new_protected:Npn \__regex_posix_space:
6102 {
6103     \__regex_item_caseful_equal:n { `\_ }
6104     \__regex_item_caseful_range:nn { `~I } { `~M }
6105 }
6106 \cs_new_protected:Npn \__regex_posix_upper:
6107 { \__regex_item_caseful_range:nn { `A } { `Z } }
6108 \cs_new_eq:NN \__regex_posix_word: \__regex_prop_w:
6109 \cs_new_protected:Npn \__regex_posix_xdigit:
6110 {
6111     \__regex_posix_digit:
```

```
6112 \__regex_item_caseful_range:nn { `A } { `F } 26112
6113 \__regex_item_caseful_range:nn { `a } { `f } 26113
6114 } 26114
6115 \cs_new_protected:Npn \__regex_escape_use:nnnn #1#2#3#4 26115
6116 { 26116
6117 \group_begin: 26117
6118 \tl_clear:N \l__regex_internal_a_tl 26118
6119 \cs_set:Npn \__regex_escape_unescaped:N ##1 { #1 } 26119
6120 \cs_set:Npn \__regex_escape_escaped:N ##1 { #2 } 26120
6121 \cs_set:Npn \__regex_escape_raw:N ##1 { #3 } 26121
6122 \__regex_standard_escapechar: 26122
6123 \__kernel_tl_gset:Nx \g__regex_internal_tl 26123
6124 { \__kernel_str_to_other_fast:n {#4} } 26124
6125 \tl_put_right:Ne \l__regex_internal_a_tl 26125
6126 { 26126
6127 \exp_after:wN \__regex_escape_loop:N \g__regex_internal_tl 26127
6128 \scan_stop: \prg_break_point: 26128
6129 } 26129
6130 \exp_after:wN 26130
6131 \group_end: 26131
6132 \l__regex_internal_a_tl 26132
6133 } 26133
6134 \cs_new:Npn \__regex_escape_loop:N #1 26134
6135 { 26135
6136 \cs_if_exist_use:cF { __regex_escape_\token_to_str:N #1:w } 26136
6137 { \__regex_escape_unescaped:N #1 } 26137
6138 \__regex_escape_loop:N 26138
6139 } 26139
6140 \cs_new:cpn { __regex_escape_ \c_backslash_str :w } 26140
6141 \__regex_escape_loop:N #1 26141
6142 { 26142
6143 \cs_if_exist_use:cF { __regex_escape_/\token_to_str:N #1:w } 26143
6144 { \__regex_escape_escaped:N #1 } 26144
6145 \__regex_escape_loop:N 26145
6146 } 26146
6147 \cs_new_eq:NN \__regex_escape_unescaped:N ? 26147
6148 \cs_new_eq:NN \__regex_escape_escaped:N ? 26148
6149 \cs_new_eq:NN \__regex_escape_raw:N ? 26149
6150 \cs_new_eq:cN { __regex_escape_ \iow_char:N\scan_stop: :w } \prg_break: 26150
6151 \cs_new:cpn { __regex_escape_/ \iow_char:N\scan_stop: :w } 26151
6152 { 26152
6153 \msg_expandable_error:nn { regex } { trailing-backslash } 26153
6154 \prg_break: 26154
6155 } 26155
6156 \cs_new:cpn { __regex_escape_~:w } { } 26156
6157 \cs_new:cpe { __regex_escape_/a:w } 26157
```

```
6158 { \exp_not:N \__regex_escape_raw:N \iow_char:N \^^G } 26158
6159 \cs_new:cpe { __regex_escape_/t:w } 26159
6160 { \exp_not:N \__regex_escape_raw:N \iow_char:N \^^I } 26160
6161 \cs_new:cpe { __regex_escape_/n:w } 26161
6162 { \exp_not:N \__regex_escape_raw:N \iow_char:N \^^J } 26162
6163 \cs_new:cpe { __regex_escape_/f:w } 26163
6164 { \exp_not:N \__regex_escape_raw:N \iow_char:N \^^L } 26164
6165 \cs_new:cpe { __regex_escape_/r:w } 26165
6166 { \exp_not:N \__regex_escape_raw:N \iow_char:N \^^M } 26166
6167 \cs_new:cpe { __regex_escape_/e:w } 26167
6168 { \exp_not:N \__regex_escape_raw:N \iow_char:N \^^[ } 26168
6169 \cs_new:cpn { __regex_escape_/x:w } \__regex_escape_loop:N 26169
6170 { 26170
6171 \exp_after:wN \__regex_escape_x_end:w 26171
6172 \int_value:w "0 \__regex_escape_x_test:N 26172
6173 } 26173
6174 \cs_new:Npn \__regex_escape_x_end:w #1 \__regex_sep: 26174
6175 { 26175
6176 \int_compare:nNnTF {#1} > \c_max_char_int 26176
6177 { 26177
6178 \msg_expandable_error:nnff { regex } { x-overflow } 26178
6179 {#1} { \int_to_Hex:n {#1} } 26179
6180 } 26180
6181 { 26181
6182 \exp_last_unbraced:Nf \__regex_escape_raw:N 26182
6183 { \char_generate:nn {#1} { 12 } } 26183
6184 } 26184
6185 } 26185
6186 \cs_new:Npn \__regex_escape_x_test:N #1 26186
6187 { 26187
6188 \if_meaning:w \scan_stop: #1 26188
6189 \exp_after:wN \use_i:nnn \exp_after:wN \__regex_sep: 26189
6190 \fi: 26190
6191 \use:n 26191
6192 { 26192
6193 \if_charcode:w \c_space_token #1 26193
6194 \exp_after:wN \__regex_escape_x_test:N 26194
6195 \else: 26195
6196 \exp_after:wN \__regex_escape_x_testii:N 26196
6197 \exp_after:wN #1 26197
6198 \fi: 26198
6199 } 26199
6200 } 26200
6201 \cs_new:Npn \__regex_escape_x_testii:N #1 26201
6202 { 26202
6203 \if_charcode:w \c_left_brace_str #1 26203
```

```
6204 \exp_after:wN \__regex_escape_x_loop:N
6205 \else:
6206 \__regex_hexadecimal_use:NTF #1
6207 { \exp_after:wN \__regex_escape_x:N }
6208 { \__regex_sep: \exp_after:wN \__regex_escape_loop:N \exp_after:wN #1 }
6209 \fi:
6210 }
6211 \cs_new:Npn \__regex_escape_x:N #1
6212 {
6213 \if_meaning:w \scan_stop: #1
6214 \exp_after:wN \use_i:nnn \exp_after:wN \__regex_sep:
6215 \fi:
6216 \use:n
6217 {
6218 \__regex_hexadecimal_use:NTF #1
6219 { \__regex_sep: \__regex_escape_loop:N }
6220 { \__regex_sep: \__regex_escape_loop:N #1 }
6221 }
6222 }
6223 \cs_new:Npn \__regex_escape_x_loop:N #1
6224 {
6225 \if_meaning:w \scan_stop: #1
6226 \exp_after:wN \use_ii:nnn
6227 \fi:
6228 \use_ii:nn
6229 { \__regex_sep: \__regex_escape_x_loop_error:n { } {#1} }
6230 {
6231 \__regex_hexadecimal_use:NTF #1
6232 { \__regex_escape_x_loop:N }
6233 {
6234 \token_if_eq_charcode:NNTF \c_space_token #1
6235 { \__regex_escape_x_loop:N }
6236 {
6237 \__regex_sep:
6238 \exp_after:wN
6239 \token_if_eq_charcode:NNTF \c_right_brace_str #1
6240 { \__regex_escape_loop:N }
6241 { \__regex_escape_x_loop_error:n {#1} }
6242 }
6243 }
6244 }
6245 }
6246 \cs_new:Npn \__regex_escape_x_loop_error:n #1
6247 {
6248 \msg_expandable_error:nnn { regex } { x-missing-rbrace } {#1}
6249 \__regex_escape_loop:N #1
```

```
6250 }
6251 \cs_new:Npn \__regex_hexadecimal_use:NTF #1
6252 {
6253   \if_int_compare:w \c_one_int < "1 \token_to_str:N #1 \exp_stop_f:
6254     #1
6255   \else:
6256     \if_case:w
6257       \__regex_int_eval:w \exp_after:wN ` \token_to_str:N #1 - `a \scan_stop:
6258       A
6259       \or: B
6260       \or: C
6261       \or: D
6262       \or: E
6263       \or: F
6264       \else:
6265         \exp_after:wN \exp_after:wN \exp_after:wN \use_iii:nnn
6266       \fi:
6267     \fi:
6268     \use_i:nn
6269   }
6270 \prg_new_conditional:Npnn \__regex_char_if_special:N #1 { TF }
6271 {
6272   \if:w
6273     T
6274     \if_int_compare:w `#1 > `Z \exp_stop_f:
6275     \if_int_compare:w `#1 > `z \exp_stop_f:
6276     \if_int_compare:w `#1 < \c__regex_ascii_max_int
6277       \else: F \fi:
6278     \else:
6279       \if_int_compare:w `#1 < `a \exp_stop_f:
6280       \else: F \fi:
6281     \fi:
6282   \else:
6283     \if_int_compare:w `#1 > `9 \exp_stop_f:
6284     \if_int_compare:w `#1 < `A \exp_stop_f:
6285     \else: F \fi:
6286   \else:
6287     \if_int_compare:w `#1 < `0 \exp_stop_f:
6288     \if_int_compare:w `#1 < `_\exp_stop_f:
6289     F \fi:
6290   \else: F \fi:
6291   \fi:
6292   \fi:
6293   T
6294   \prg_return_true:
6295 \else:
```

6296	\prg_return_false:	26290
6297	\fi:	26291
6298	}	26292
6299	\prg_new_conditional:Npnn __regex_char_if_alphanumeric:N #1 { TF }	26293
6300	{	26300
6301	\if:w	26301
6302	T	26302
6303	\if_int_compare:w `#1 > `Z \exp_stop_f:	26303
6304	\if_int_compare:w `#1 > `z \exp_stop_f:	26304
6305	F	26305
6306	\else:	26306
6307	\if_int_compare:w `#1 < `a \exp_stop_f:	26307
6308	F \fi:	26308
6309	\fi:	26309
6310	\else:	26310
6311	\if_int_compare:w `#1 > `9 \exp_stop_f:	26311
6312	\if_int_compare:w `#1 < `A \exp_stop_f:	26312
6313	F \fi:	26313
6314	\else:	26314
6315	\if_int_compare:w `#1 < `0 \exp_stop_f:	26315
6316	F \fi:	26316
6317	\fi:	26317
6318	\fi:	26318
6319	T	26319
6320	\prg_return_true:	26320
6321	\else:	26321
6322	\prg_return_false:	26322
6323	\fi:	26323
6324	}	26324
6325	\int_new:N \l__regex_group_level_int	26325
6326	\int_new:N \l__regex_mode_int	26326
6327	\int_const:Nn \c__regex_cs_in_class_mode_int { -6 }	26327
6328	\int_const:Nn \c__regex_cs_mode_int { -2 }	26328
6329	\int_const:Nn \c__regex_outer_mode_int { 0 }	26329
6330	\int_const:Nn \c__regex_catcode_mode_int { 2 }	26330
6331	\int_const:Nn \c__regex_class_mode_int { 3 }	26331
6332	\int_const:Nn \c__regex_catcode_in_class_mode_int { 6 }	26332
6333	\int_new:N \l__regex_catcodes_int	26333
6334	\int_new:N \l__regex_default_catcodes_int	26334
6335	\bool_new:N \l__regex_catcodes_bool	26335
6336	\int_const:Nn \c__regex_catcode_C_int { "1 }	26336
6337	\int_const:Nn \c__regex_catcode_B_int { "4 }	26337
6338	\int_const:Nn \c__regex_catcode_E_int { "10 }	26338
6339	\int_const:Nn \c__regex_catcode_M_int { "40 }	26339
6340	\int_const:Nn \c__regex_catcode_T_int { "100 }	26340
6341	\int_const:Nn \c__regex_catcode_P_int { "1000 }	26341


```
6342 \int_const:Nn \c__regex_catcode_U_int { "4000 } 2634
6343 \int_const:Nn \c__regex_catcode_D_int { "10000 } 2634
6344 \int_const:Nn \c__regex_catcode_S_int { "100000 } 2634
6345 \int_const:Nn \c__regex_catcode_L_int { "400000 } 2634
6346 \int_const:Nn \c__regex_catcode_O_int { "1000000 } 2634
6347 \int_const:Nn \c__regex_catcode_A_int { "4000000 } 2634
6348 \int_const:Nn \c__regex_all_catcodes_int { "5515155 } 2634
6349 \cs_new_eq:NN \l__regex_internal_regex \c__regex_no_match_regex 2634
6350 \seq_new:N \l__regex_show_prefix_seq 2635
6351 \int_new:N \l__regex_show_lines_int 2635
6352 \cs_new:Npn \__regex_two_if_eq:NNNTF #1#2#3#4 2635
6353 { 2635
6354     \if_meaning:w #1 #3 2635
6355     \if:w #2 #4 2635
6356         \exp_after:wN \exp_after:wN \exp_after:wN \use_ii:nnn 2635
6357         \fi: 2635
6358         \fi: 2635
6359         \use_ii:nn 2635
6360     } 2636
6361 \cs_new_protected:Npn \__regex_get_digits:NTFw #1#2#3#4#5 2636
6362 { 2636
6363     \__regex_if_raw_digit:NNTF #4 #5 2636
6364     { #1 = #5 \__regex_get_digits_loop:nw {#2} } 2636
6365     { #3 #4 #5 } 2636
6366 } 2636
6367 \cs_new:Npn \__regex_get_digits_loop:nw #1#2#3 2636
6368 { 2636
6369     \__regex_if_raw_digit:NNTF #2 #3 2636
6370     { #3 \__regex_get_digits_loop:nw {#1} } 2637
6371     { \scan_stop: #1 #2 #3 } 2637
6372 } 2637
6373 \cs_new:Npn \__regex_if_raw_digit:NNTF #1#2 2637
6374 { 2637
6375     \if_meaning:w \__regex_compile_raw:N #1 2637
6376     \if_int_compare:w \c_one_int < 1 #2 \exp_stop_f: 2637
6377     \exp_after:wN \exp_after:wN \exp_after:wN \use_ii:nnn 2637
6378     \fi: 2637
6379     \fi: 2637
6380     \use_ii:nn 2638
6381 } 2638
6382 \prg_new_conditional:Npnn \__regex_if_in_class: { TF } 2638
6383 { 2638
6384     \if_int_odd:w \l__regex_mode_int 2638
6385     \prg_return_true: 2638
6386     \else: 2638
6387     \prg_return_false: 2638
```

```
6388 \fi: 26388
6389 } 26389
6390 \cs_new:Npn \__regex_if_in_cs:TF 26390
6391 { 26391
6392 \if_int_odd:w \l__regex_mode_int 26392
6393 \else: 26393
6394 \if_int_compare:w \l__regex_mode_int < \c__regex_outer_mode_int 26394
6395 \exp_after:wN \exp_after:wN \exp_after:wN \use_ii:nnn 26395
6396 \fi: 26396
6397 \fi: 26397
6398 \use_ii:nn 26398
6399 } 26399
6400 \cs_new:Npn \__regex_if_in_class_or_catcode:TF 26400
6401 { 26401
6402 \if_int_odd:w \l__regex_mode_int 26402
6403 \else: 26403
6404 \if_int_compare:w \l__regex_mode_int > \c__regex_outer_mode_int 26404
6405 \else: 26405
6406 \exp_after:wN \exp_after:wN \exp_after:wN \use_iii:nnn 26406
6407 \fi: 26407
6408 \fi: 26408
6409 \use_i:nn 26409
6410 } 26410
6411 \prg_new_conditional:Npnn \__regex_if_within_catcode: { TF } 26411
6412 { 26412
6413 \if_int_compare:w \l__regex_mode_int > \c__regex_outer_mode_int 26413
6414 \prg_return_true: 26414
6415 \else: 26415
6416 \prg_return_false: 26416
6417 \fi: 26417
6418 } 26418
6419 \cs_new_protected:Npn \__regex_chk_c_allowed:T 26419
6420 { 26420
6421 \if_int_compare:w \l__regex_mode_int = \c__regex_outer_mode_int 26421
6422 \else: 26422
6423 \if_int_compare:w \l__regex_mode_int = \c__regex_class_mode_int 26423
6424 \else: 26424
6425 \msg_error:nn { regex } { c-bad-mode } 26425
6426 \exp_after:wN \use_i:nnn 26426
6427 \fi: 26427
6428 \fi: 26428
6429 \use:n 26429
6430 } 26430
6431 \cs_new_protected:Npn \__regex_mode_quit_c: 26431
6432 { 26432
6433 \if_int_compare:w \l__regex_mode_int = \c__regex_catcode_mode_int 26433
```

```

6434 \int_set_eq:NN \l__regex_mode_int \c__regex_outer_mode_int 26434
6435 \else: 26435
6436 \if_int_compare:w \l__regex_mode_int = 26436
6437 \c__regex_catcode_in_class_mode_int 26437
6438 \int_set_eq:NN \l__regex_mode_int \c__regex_class_mode_int 26438
6439 \fi: 26439
6440 \fi: 26440
6441 } 26441
6442 \cs_new_protected:Npn \__regex_compile:w 26442
6443 { 26443
6444 \group_begin: 26444
6445 \tl_build_begin:N \l__regex_build_tl 26445
6446 \int_zero:N \l__regex_group_level_int 26446
6447 \int_set_eq:NN \l__regex_default_catcodes_int 26447
6448 \c__regex_all_catcodes_int 26448
6449 \int_set_eq:NN \l__regex_catcodes_int \l__regex_default_catcodes_int 26449
6450 \cs_set:Npn \__regex_item_equal:n { \__regex_item_caseful_equal:n } 26450
6451 \cs_set:Npn \__regex_item_range:nn { \__regex_item_caseful_range:nn } 26451
6452 \tl_build_put_right:Nn \l__regex_build_tl 26452
6453 { \__regex_branch:n { \if_false: } \fi: } 26453
6454 } 26454
6455 \cs_new_protected:Npn \__regex_compile_end: 26455
6456 { 26456
6457 \__regex_if_in_class:TF 26457
6458 { 26458
6459 \msg_error:nn { regex } { missing-rbrack } 26459
6460 \use:c { __regex_compile_]: } 26460
6461 \prg_do_nothing: \prg_do_nothing: 26461
6462 } 26462
6463 { } 26463
6464 \if_int_compare:w \l__regex_group_level_int > \c_zero_int 26464
6465 \msg_error:nne { regex } { missing-rparen } 26465
6466 { \int_use:N \l__regex_group_level_int } 26466
6467 \prg_replicate:nn 26467
6468 \l__regex_group_level_int 26468
6469 { 26469
6470 \tl_build_put_right:Nn \l__regex_build_tl 26470
6471 { 26471
6472 \if_false: { \fi: } 26472
6473 \if_false: { \fi: } { 1 } { 0 } \c_true_bool 26473
6474 } 26474
6475 \tl_build_end:N \l__regex_build_tl 26475
6476 \exp_args:NNNo 26476
6477 \group_end: 26477
6478 \tl_build_put_right:Nn \l__regex_build_tl 26478
6479 { \l__regex_build_tl } 26479

```

```
6480     }
6481     \fi:
6482     \tl_build_put_right:Nn \l__regex_build_tl { \if_false: { \fi: } }
6483     \tl_build_end:N \l__regex_build_tl
6484     \exp_args:NNNe
6485     \group_end:
6486     \tl_set:Nn \l__regex_internal_regex { \l__regex_build_tl }
6487 }
6488 \cs_new_protected:Npn \__regex_compile:n #1
6489 {
6490     \__regex_compile:w
6491     \__regex_standard_escapechar:
6492     \int_set_eq:NN \l__regex_mode_int \c__regex_outer_mode_int
6493     \__regex_escape_use:nnnn
6494     {
6495         \__regex_char_if_special:NTF ##1
6496         \__regex_compile_special:N \__regex_compile_raw:N ##1
6497     }
6498     {
6499         \__regex_char_if_alphanumeric:NTF ##1
6500         \__regex_compile_escaped:N \__regex_compile_raw:N ##1
6501     }
6502     { \__regex_compile_raw:N ##1 }
6503     { #1 }
6504     \prg_do_nothing: \prg_do_nothing:
6505     \prg_do_nothing: \prg_do_nothing:
6506     \int_compare:nNnT \l__regex_mode_int = \c__regex_catcode_mode_int
6507     { \msg_error:nn { regex } { c-trailing } }
6508     \int_compare:nNnT \l__regex_mode_int < \c__regex_outer_mode_int
6509     {
6510         \msg_error:nn { regex } { c-missing-rbrace }
6511         \__regex_compile_end_cs:
6512         \prg_do_nothing: \prg_do_nothing:
6513         \prg_do_nothing: \prg_do_nothing:
6514     }
6515     \__regex_compile_end:
6516 }
6517 \cs_new_protected:Npn \__regex_compile_use:n #1
6518 {
6519     \tl_if_single_token:nT {#1}
6520     {
6521         \exp_after:wN \__regex_compile_use_aux:w
6522         \token_to_meaning:N #1 ~ \q__regex_nil
6523     }
6524     \__regex_compile:n {#1} \l__regex_internal_regex
6525 }
```

```
6526 \cs_new_protected:Npn \__regex_compile_use_aux:w #1 ~ #2 \q__regex_nil 2652
6527 { 2652
6528   \str_if_eq:nnT { #1 ~ } { macro:->\__regex_branch:n } 2652
6529   { \use_ii:nnn } 2652
6530 } 2653
6531 \cs_new_protected:Npn \__regex_compile_special:N #1 2653
6532 { 2653
6533   \cs_if_exist_use:cF { __regex_compile_#1: } 2653
6534   { \__regex_compile_raw:N #1 } 2653
6535 } 2653
6536 \cs_new_protected:Npn \__regex_compile_escaped:N #1 2653
6537 { 2653
6538   \cs_if_exist_use:cF { __regex_compile_/#1: } 2653
6539   { \__regex_compile_raw:N #1 } 2653
6540 } 2654
6541 \cs_new_protected:Npn \__regex_compile_one:n #1 2654
6542 { 2654
6543   \__regex_mode_quit_c: 2654
6544   \__regex_if_in_class:TF { } 2654
6545   { 2654
6546     \tl_build_put_right:Nn \l__regex_build_tl 2654
6547     { \__regex_class:NnnnN \c_true_bool { \if_false: } \fi: } 2654
6548   } 2654
6549   \tl_build_put_right:Ne \l__regex_build_tl 2654
6550   { 2655
6551     \if_int_compare:w \l__regex_catcodes_int < 2655
6552     \c__regex_all_catcodes_int 2655
6553     \__regex_item_catcode:nT { \int_use:N \l__regex_catcodes_int } 2655
6554     { \exp_not:N \exp_not:n {#1} } 2655
6555     \else: 2655
6556     \exp_not:N \exp_not:n {#1} 2655
6557     \fi: 2655
6558   } 2655
6559   \int_set_eq:NN \l__regex_catcodes_int \l__regex_default_catcodes_int 2655
6560   \__regex_if_in_class:TF { } { \__regex_compile_quantifier:w } 2656
6561 } 2656
6562 \cs_new_protected:Npn \__regex_compile_abort_tokens:n #1 2656
6563 { 2656
6564   \use:e 2656
6565   { 2656
6566     \exp_args:No \tl_map_function:nN { \tl_to_str:n {#1} } 2656
6567     \__regex_compile_raw:N 2656
6568   } 2656
6569 } 2656
6570 \cs_generate_variant:Nn \__regex_compile_abort_tokens:n { e } 2657
6571 \cs_new_protected:Npn \__regex_compile_if_quantifier:TFw #1#2#3#4 2657
```

```
6572 {
6573     \token_if_eq_meaning:NNTF #3 \__regex_compile_special:N
6574     { \cs_if_exist:cTF { __regex_compile_quantifier_#4:w } }
6575     { \use_ii:nn }
6576     {#1} {#2} #3 #4
6577 }
6578 \cs_new_protected:Npn \__regex_compile_quantifier:w #1#2
6579 {
6580     \token_if_eq_meaning:NNTF #1 \__regex_compile_special:N
6581     {
6582         \cs_if_exist_use:cF { __regex_compile_quantifier_#2:w }
6583         { \__regex_compile_quantifier_none: #1 #2 }
6584     }
6585     { \__regex_compile_quantifier_none: #1 #2 }
6586 }
6587 \cs_new_protected:Npn \__regex_compile_quantifier_none:
6588 {
6589     \tl_build_put_right:Nn \l__regex_build_tl
6590     { \if_false: { \fi: } { 1 } { 0 } \c_false_bool }
6591 }
6592 \cs_new_protected:Npn \__regex_compile_quantifier_abort:eNN #1#2#3
6593 {
6594     \__regex_compile_quantifier_none:
6595     \msg_warning:nnee { regex } { invalid-quantifier } {#1} {#3}
6596     \__regex_compile_abort_tokens:e {#1}
6597     #2 #3
6598 }
6599 \cs_new_protected:Npn \__regex_compile_quantifier_laziness:nnNN #1#2#3#4
6600 {
6601     \__regex_two_if_eq:NNNTF #3 #4 \__regex_compile_special:N ?
6602     {
6603         \tl_build_put_right:Nn \l__regex_build_tl
6604         { \if_false: { \fi: } { #1 } { #2 } \c_true_bool }
6605     }
6606     {
6607         \tl_build_put_right:Nn \l__regex_build_tl
6608         { \if_false: { \fi: } { #1 } { #2 } \c_false_bool }
6609         #3 #4
6610     }
6611 }
6612 \cs_new_protected:cpn { __regex_compile_quantifier_?:w }
6613 { \__regex_compile_quantifier_laziness:nnNN { 0 } { 1 } }
6614 \cs_new_protected:cpn { __regex_compile_quantifier_*:w }
6615 { \__regex_compile_quantifier_laziness:nnNN { 0 } { -1 } }
6616 \cs_new_protected:cpn { __regex_compile_quantifier_+:w }
6617 { \__regex_compile_quantifier_laziness:nnNN { 1 } { -1 } }
```

```
6618 \cs_new_protected:cpn { __regex_compile_quantifier_ \c_left_brace_str :w } 2661
6619 { 2661
6620 \__regex_get_digits:NTFw \l__regex_internal_a_int 2662
6621 { \__regex_compile_quantifier_braced_auxi:w } 2662
6622 { \__regex_compile_quantifier_abort:eNN { \c_left_brace_str } } 2662
6623 } 2662
6624 \cs_new_protected:Npn \__regex_compile_quantifier_braced_auxi:w #1#2 2662
6625 { 2662
6626 \str_case:e:nnF { #1 #2 } 2662
6627 { 2662
6628 { \__regex_compile_special:N \c_right_brace_str } 2662
6629 { 2662
6630 \exp_args:No \__regex_compile_quantifier_laziness:nnNN 2663
6631 { \int_use:N \l__regex_internal_a_int } 0 2663
6632 } 2663
6633 { \__regex_compile_special:N , } 2663
6634 { 2663
6635 \__regex_get_digits:NTFw \l__regex_internal_b_int 2663
6636 { \__regex_compile_quantifier_braced_auxiii:w } 2663
6637 { \__regex_compile_quantifier_braced_auxii:w } 2663
6638 } 2663
6639 } 2663
6640 { 2664
6641 \__regex_compile_quantifier_abort:eNN 2664
6642 { \c_left_brace_str \int_use:N \l__regex_internal_a_int } 2664
6643 #1 #2 2664
6644 } 2664
6645 } 2664
6646 \cs_new_protected:Npn \__regex_compile_quantifier_braced_auxii:w #1#2 2664
6647 { 2664
6648 \__regex_two_if_eq:NNNTF #1 #2 \__regex_compile_special:N \c_right_brace_str 2664
6649 { 2664
6650 \exp_args:No \__regex_compile_quantifier_laziness:nnNN 2665
6651 { \int_use:N \l__regex_internal_a_int } { -1 } 2665
6652 } 2665
6653 { 2665
6654 \__regex_compile_quantifier_abort:eNN 2665
6655 { \c_left_brace_str \int_use:N \l__regex_internal_a_int , } 2665
6656 #1 #2 2665
6657 } 2665
6658 } 2665
6659 \cs_new_protected:Npn \__regex_compile_quantifier_braced_auxiii:w #1#2 2665
6660 { 2666
6661 \__regex_two_if_eq:NNNTF #1 #2 \__regex_compile_special:N \c_right_brace_str 2666
6662 { 2666
6663 \if_int_compare:w \l__regex_internal_a_int > 2666
```



```
6664         \l__regex_internal_b_int
6665         \msg_error:nnee { regex } { backwards-quantifier }
6666         { \int_use:N \l__regex_internal_a_int }
6667         { \int_use:N \l__regex_internal_b_int }
6668         \int_zero:N \l__regex_internal_b_int
6669     \else:
6670         \int_sub:Nn \l__regex_internal_b_int \l__regex_internal_a_int
6671     \fi:
6672     \exp_args:Noo \__regex_compile_quantifier_laziness:nnNN
6673         { \int_use:N \l__regex_internal_a_int }
6674         { \int_use:N \l__regex_internal_b_int }
6675 }
6676 {
6677     \__regex_compile_quantifier_abort:eNN
6678     {
6679         \c_left_brace_str
6680         \int_use:N \l__regex_internal_a_int ,
6681         \int_use:N \l__regex_internal_b_int
6682     }
6683     #1 #2
6684 }
6685 }
6686 \cs_new_protected:Npn \__regex_compile_raw_error:N #1
6687 {
6688     \msg_error:nne { regex } { bad-escape } {#1}
6689     \__regex_compile_raw:N #1
6690 }
6691 \cs_new_protected:Npn \__regex_compile_raw:N #1#2#3
6692 {
6693     \__regex_if_in_class:TF
6694     {
6695         \__regex_two_if_eq:NNNNTF #2 #3 \__regex_compile_special:N -
6696         { \__regex_compile_range:Nw #1 }
6697         {
6698             \__regex_compile_one:n
6699             { \__regex_item_equal:n { \int_value:w `#1 } }
6700             #2 #3
6701         }
6702     }
6703     {
6704         \__regex_compile_one:n
6705         { \__regex_item_equal:n { \int_value:w `#1 } }
6706         #2 #3
6707     }
6708 }
6709 \cs_new_protected:Npn \__regex_if_end_range:NNTF #1#2
```

```
6710 {
6711     \if_meaning:w \__regex_compile_raw:N #1
6712 \else:
6713     \if_meaning:w \__regex_compile_special:N #1
6714     \if_charcode:w ] #2
6715     \use_i:nn
6716     \fi:
6717 \else:
6718     \exp_after:wN \exp_after:wN \exp_after:wN \use_iii:nnn
6719     \fi:
6720 \fi:
6721 \use_i:nn
6722 }
6723 \cs_new_protected:Npn \__regex_compile_range:Nw #1#2#3
6724 {
6725     \__regex_if_end_range:NNTF #2 #3
6726     {
6727         \if_int_compare:w `#1 > `#3 \exp_stop_f:
6728         \msg_error:nnee { regex } { range-backwards } {#1} {#3}
6729     \else:
6730         \tl_build_put_right:Ne \l__regex_build_tl
6731         {
6732             \if_int_compare:w `#1 = `#3 \exp_stop_f:
6733             \__regex_item_equal:n
6734             \else:
6735                 \__regex_item_range:nn { \int_value:w `#1 }
6736             \fi:
6737             { \int_value:w `#3 }
6738         }
6739     \fi:
6740 }
6741 {
6742     \msg_warning:nnee { regex } { range-missing-end }
6743     {#1} { \c_backslash_str #3 }
6744     \tl_build_put_right:Ne \l__regex_build_tl
6745     {
6746         \__regex_item_equal:n { \int_value:w `#1 \exp_stop_f: }
6747         \__regex_item_equal:n { \int_value:w `~ \exp_stop_f: }
6748     }
6749     #2#3
6750 }
6751 }
6752 \cs_new_protected:cpe { __regex_compile_.: }
6753 {
6754     \exp_not:N \__regex_if_in_class:TF
6755     { \__regex_compile_raw:N . }
```

```
6756 { \__regex_compile_one:n \exp_not:c { __regex_prop_.: } }
6757 }
6758 \cs_new_protected:cpn { __regex_prop_.: }
6759 {
6760   \if_int_compare:w \l__regex_curr_char_int > - 2 \exp_stop_f:
6761   \exp_after:wN \__regex_break_true:w
6762   \fi:
6763 }
6764 \cs_set_protected:Npn \__regex_tmp:w #1#2
6765 {
6766   \cs_new_protected:cpe { __regex_compile_/#1: }
6767   { \__regex_compile_one:n \exp_not:c { __regex_prop_#1: } }
6768   \cs_new_protected:cpe { __regex_compile_/#2: }
6769   {
6770     \__regex_compile_one:n
6771     { \__regex_item_reverse:n { \exp_not:c { __regex_prop_#1: } } }
6772   }
6773 }
6774 \__regex_tmp:w d D
6775 \__regex_tmp:w h H
6776 \__regex_tmp:w s S
6777 \__regex_tmp:w v V
6778 \__regex_tmp:w w W
6779 \cs_new_protected:cpn { __regex_compile_/N: }
6780 { \__regex_compile_one:n \__regex_prop_N: }
6781 \cs_new_protected:Npn \__regex_compile_anchor_letter:NNN #1#2#3
6782 {
6783   \__regex_if_in_class_or_catcode:TF { \__regex_compile_raw_error:N #1 }
6784   {
6785     \tl_build_put_right:Nn \l__regex_build_tl
6786     { \__regex_assertion:Nn #2 {#3} }
6787   }
6788 }
6789 \cs_new_protected:cpn { __regex_compile_/A: }
6790 { \__regex_compile_anchor_letter:NNN A \c_true_bool \__regex_A_test: }
6791 \cs_new_protected:cpn { __regex_compile_/G: }
6792 { \__regex_compile_anchor_letter:NNN G \c_true_bool \__regex_G_test: }
6793 \cs_new_protected:cpn { __regex_compile_/Z: }
6794 { \__regex_compile_anchor_letter:NNN Z \c_true_bool \__regex_Z_test: }
6795 \cs_new_protected:cpn { __regex_compile_/z: }
6796 { \__regex_compile_anchor_letter:NNN z \c_true_bool \__regex_Z_test: }
6797 \cs_new_protected:cpn { __regex_compile_/b: }
6798 { \__regex_compile_anchor_letter:NNN b \c_true_bool \__regex_b_test: }
6799 \cs_new_protected:cpn { __regex_compile_/B: }
6800 { \__regex_compile_anchor_letter:NNN B \c_false_bool \__regex_b_test: }
6801 \cs_set_protected:Npn \__regex_tmp:w #1#2
```

```

6802 {
6803     \cs_new_protected:cpn { __regex_compile_#1: }
6804     {
6805         \__regex_if_in_class_or_catcode:TF { \__regex_compile_raw:N #1 }
6806         {
6807             \tl_build_put_right:Nn \l__regex_build_tl
6808             { \__regex_assertion:Nn \c_true_bool {#2} }
6809         }
6810     }
6811 }
6812 \exp_args:Ne \__regex_tmp:w { \iow_char:N \^ } { \__regex_A_test: }
6813 \exp_args:Ne \__regex_tmp:w { \iow_char:N \$ } { \__regex_Z_test: }
6814 \cs_new_protected:cpn { __regex_compile_]: }
6815 {
6816     \__regex_if_in_class:TF
6817     {
6818         \if_int_compare:w \l__regex_mode_int >
6819         \c__regex_catcode_in_class_mode_int
6820         \tl_build_put_right:Nn \l__regex_build_tl { \if_false: { \fi: } }
6821         \fi:
6822         \tex_advance:D \l__regex_mode_int - 15 \exp_stop_f:
6823         \tex_divide:D \l__regex_mode_int 13 \exp_stop_f:
6824         \if_int_odd:w \l__regex_mode_int \else:
6825             \exp_after:wN \__regex_compile_quantifier:w
6826         \fi:
6827     }
6828     { \__regex_compile_raw:N ] }
6829 }
6830 \cs_new_protected:cpn { __regex_compile_[: }
6831 {
6832     \__regex_if_in_class:TF
6833     { \__regex_compile_class_posix_test:w }
6834     {
6835         \__regex_if_within_catcode:TF
6836         {
6837             \exp_after:wN \__regex_compile_class_catcode:w
6838             \int_use:N \l__regex_catcodes_int \__regex_sep:
6839         }
6840         { \__regex_compile_class_normal:w }
6841     }
6842 }
6843 \cs_new_protected:Npn \__regex_compile_class_normal:w
6844 {
6845     \__regex_compile_class:TFNN
6846     { \__regex_class:NnnnN \c_true_bool }
6847     { \__regex_class:NnnnN \c_false_bool }

```

```
6848 }
6849 \cs_new_protected:Npn \__regex_compile_class_catcode:w #1 \__regex_sep:
6850 {
6851   \if_int_compare:w \l__regex_mode_int = \c__regex_catcode_mode_int
6852     \tl_build_put_right:Nn \l__regex_build_tl
6853       { \__regex_class:NnnnN \c_true_bool { \if_false: } \fi: }
6854   \fi:
6855   \int_set_eq:NN \l__regex_catcodes_int \l__regex_default_catcodes_int
6856   \__regex_compile_class:TFNN
6857     { \__regex_item_catcode:nT {#1} }
6858     { \__regex_item_catcode_reverse:nT {#1} }
6859 }
6860 \cs_new_protected:Npn \__regex_compile_class:TFNN #1#2#3#4
6861 {
6862   \l__regex_mode_int = \int_value:w \l__regex_mode_int 3 \exp_stop_f:
6863   \__regex_two_if_eq:NNNTF #3 #4 \__regex_compile_special:N ^
6864   {
6865     \tl_build_put_right:Nn \l__regex_build_tl { #2 { \if_false: } \fi: }
6866     \__regex_compile_class:NN
6867   }
6868   {
6869     \tl_build_put_right:Nn \l__regex_build_tl { #1 { \if_false: } \fi: }
6870     \__regex_compile_class:NN #3 #4
6871   }
6872 }
6873 \cs_new_protected:Npn \__regex_compile_class:NN #1#2
6874 {
6875   \token_if_eq_charcode:NNTF #2 ]
6876     { \__regex_compile_raw:N #2 }
6877     { #1 #2 }
6878 }
6879 \cs_new_protected:Npn \__regex_compile_class_posix_test:w #1#2
6880 {
6881   \token_if_eq_meaning:NNT \__regex_compile_special:N #1
6882   {
6883     \str_case:nn { #2 }
6884     {
6885       : { \__regex_compile_class_posix:NNNNw }
6886       = {
6887         \msg_warning:nne { regex }
6888         { posix-unsupported } { = }
6889       }
6890       . {
6891         \msg_warning:nne { regex }
6892         { posix-unsupported } { . }
6893       }
6894     }
```

```

6894     }
6895 }
6896 \__regex_compile_raw:N [ #1 #2
6897 }
6898 \cs_new_protected:Npn \__regex_compile_class_posix:NNNNw #1#2#3#4#5#6
6899 {
6900     \__regex_two_if_eq:NNNNTF #5 #6 \__regex_compile_special:N ^
6901     {
6902         \bool_set_false:N \l__regex_internal_bool
6903         \__kernel_tl_set:Nx \l__regex_internal_a_tl { \if_false: } \fi:
6904         \__regex_compile_class_posix_loop:w
6905     }
6906     {
6907         \bool_set_true:N \l__regex_internal_bool
6908         \__kernel_tl_set:Nx \l__regex_internal_a_tl { \if_false: } \fi:
6909         \__regex_compile_class_posix_loop:w #5 #6
6910     }
6911 }
6912 \cs_new:Npn \__regex_compile_class_posix_loop:w #1#2
6913 {
6914     \token_if_eq_meaning:NNTF \__regex_compile_raw:N #1
6915     { #2 \__regex_compile_class_posix_loop:w }
6916     { \if_false: { \fi: } \__regex_compile_class_posix_end:w #1 #2 }
6917 }
6918 \cs_new_protected:Npn \__regex_compile_class_posix_end:w #1#2#3#4
6919 {
6920     \__regex_two_if_eq:NNNNTF #1 #2 \__regex_compile_special:N :
6921     { \__regex_two_if_eq:NNNNTF #3 #4 \__regex_compile_special:N ] }
6922     { \use_ii:nn }
6923     {
6924         \cs_if_exist:cTF { __regex_posix_ \l__regex_internal_a_tl : }
6925         {
6926             \__regex_compile_one:n
6927             {
6928                 \bool_if:NTF \l__regex_internal_bool \use:n \__regex_item_reverse:n
6929                 { \exp_not:c { __regex_posix_ \l__regex_internal_a_tl : } }
6930             }
6931         }
6932         {
6933             \msg_warning:nne { regex } { posix-unknown }
6934             { \l__regex_internal_a_tl }
6935             \__regex_compile_abort_tokens:e
6936             {
6937                 [: \bool_if:NF \l__regex_internal_bool { ^ }
6938                 \l__regex_internal_a_tl :]
6939             }

```

```

6940     }
6941 }
6942 {
6943     \msg_error:nnee { regex } { posix-missing-close }
6944     { [: \l__regex_internal_a_tl ] { #2 #4 }
6945     \__regex_compile_abort_tokens:e { [: \l__regex_internal_a_tl ]
6946     #1 #2 #3 #4
6947 }
6948 }
6949 \cs_new_protected:Npn \__regex_compile_group_begin:N #1
6950 {
6951     \tl_build_put_right:Nn \l__regex_build_tl { #1 { \if_false: } \fi: }
6952     \__regex_mode_quit_c:
6953     \group_begin:
6954         \tl_build_begin:N \l__regex_build_tl
6955         \int_set_eq:NN \l__regex_default_catcodes_int \l__regex_catcodes_int
6956         \int_incr:N \l__regex_group_level_int
6957         \tl_build_put_right:Nn \l__regex_build_tl
6958         { \__regex_branch:n { \if_false: } \fi: }
6959     }
6960 \cs_new_protected:Npn \__regex_compile_group_end:
6961 {
6962     \if_int_compare:w \l__regex_group_level_int > \c_zero_int
6963         \tl_build_put_right:Nn \l__regex_build_tl { \if_false: { \fi: } }
6964         \tl_build_end:N \l__regex_build_tl
6965         \exp_args:NNNe
6966         \group_end:
6967         \tl_build_put_right:Nn \l__regex_build_tl { \l__regex_build_tl }
6968         \int_set_eq:NN \l__regex_catcodes_int \l__regex_default_catcodes_int
6969         \exp_after:wN \__regex_compile_quantifier:w
6970     \else:
6971         \msg_warning:nn { regex } { extra-rparen }
6972         \exp_after:wN \__regex_compile_raw:N \exp_after:wN )
6973     \fi:
6974 }
6975 \cs_new_protected:cpn { __regex_compile(: }
6976 {
6977     \__regex_if_in_class:TF { \__regex_compile_raw:N ( }
6978     {
6979         \if_int_compare:w \l__regex_mode_int =
6980         \c__regex_catcode_in_class_mode_int
6981         \msg_error:nn { regex } { c-lparen-in-class }
6982         \exp_after:wN \__regex_compile_raw:N \exp_after:wN (
6983     \else:
6984         \exp_after:wN \__regex_compile_lparen:w
6985     \fi:

```



```
6986     }
6987 }
6988 \cs_new_protected:Npn \__regex_compile_lparen:w #1#2#3#4
6989 {
6990     \__regex_two_if_eq:NNNTF #1 #2 \__regex_compile_special:N ?
6991     {
6992         \cs_if_exist_use:cF
6993         { \__regex_compile_special_group\_token_to_str:N #4 :w }
6994         {
6995             \msg_warning:nne { regex } { special-group-unknown }
6996             { (? #4 )
6997             \__regex_compile_group_begin:N \__regex_group:nnnN
6998             \__regex_compile_raw:N ? #3 #4
6999             }
7000     }
7001     {
7002         \__regex_compile_group_begin:N \__regex_group:nnnN
7003         #1 #2 #3 #4
7004     }
7005 }
7006 \cs_new_protected:cpn { \__regex_compile_|: }
7007 {
7008     \__regex_if_in_class:TF { \__regex_compile_raw:N | }
7009     {
7010         \tl_build_put_right:Nn \l__regex_build_tl
7011         { \if_false: { \fi: } \__regex_branch:n { \if_false: } \fi: }
7012     }
7013 }
7014 \cs_new_protected:cpn { \__regex_compile_): }
7015 {
7016     \__regex_if_in_class:TF { \__regex_compile_raw:N ) }
7017     { \__regex_compile_group_end: }
7018 }
7019 \cs_new_protected:cpn { \__regex_compile_special_group_:w }
7020 { \__regex_compile_group_begin:N \__regex_group_no_capture:nnnN }
7021 \cs_new_protected:cpn { \__regex_compile_special_group_|:w }
7022 { \__regex_compile_group_begin:N \__regex_group_resetting:nnnN }
7023 \cs_new_protected:Npn \__regex_compile_special_group_i:w #1#2
7024 {
7025     \__regex_two_if_eq:NNNTF #1 #2 \__regex_compile_special:N )
7026     {
7027         \cs_set:Npn \__regex_item_equal:n
7028         { \__regex_item_caseless_equal:n }
7029         \cs_set:Npn \__regex_item_range:nn
7030         { \__regex_item_caseless_range:nn }
7031     }
```

```
7032 {
7033     \msg_warning:nne { regex } { unknown-option } { (?i #2 }
7034     \__regex_compile_raw:N (
7035     \__regex_compile_raw:N ?
7036     \__regex_compile_raw:N i
7037     #1 #2
7038 }
7039 }
7040 \cs_new_protected:cpn { __regex_compile_special_group_~:w } #1#2#3#4
7041 {
7042     \__regex_two_if_eq:NNNTF #1 #2 \__regex_compile_raw:N i
7043     { \__regex_two_if_eq:NNNTF #3 #4 \__regex_compile_special:N ) }
7044     { \use_ii:nn }
7045     {
7046         \cs_set:Npn \__regex_item_equal:n
7047             { \__regex_item_caseful_equal:n }
7048         \cs_set:Npn \__regex_item_range:nn
7049             { \__regex_item_caseful_range:nn }
7050     }
7051     {
7052         \msg_warning:nne { regex } { unknown-option } { (?-#2#4 }
7053         \__regex_compile_raw:N (
7054         \__regex_compile_raw:N ?
7055         \__regex_compile_raw:N -
7056         #1 #2 #3 #4
7057     }
7058 }
7059 \cs_new_protected:cpn { __regex_compile_/c: }
7060 { \__regex_chk_c_allowed:T { \__regex_compile_c_test:NN } }
7061 \cs_new_protected:Npn \__regex_compile_c_test:NN #1#2
7062 {
7063     \token_if_eq_meaning:NNTF #1 \__regex_compile_raw:N
7064     {
7065         \int_if_exist:cTF { c__regex_catcode_#2_int }
7066         {
7067             \int_set_eq:Nc \l__regex_catcodes_int
7068             { c__regex_catcode_#2_int }
7069             \l__regex_mode_int
7070             = \if_case:w \l__regex_mode_int
7071                 \c__regex_catcode_mode_int
7072             \else:
7073                 \c__regex_catcode_in_class_mode_int
7074             \fi:
7075             \token_if_eq_charcode:NNT C #2 { \__regex_compile_c_C:NN }
7076         }
7077     }
```

```
7078 { \cs_if_exist_use:cF { __regex_compile_c_#2:w } } 27078
7079 { 27079
7080     \msg_error:nne { regex } { c-missing-category } {#2} 27080
7081     #1 #2 27081
7082 } 27082
7083 } 27083
7084 \cs_new_protected:Npn \__regex_compile_c_C:NN #1#2 27084
7085 { 27085
7086     \token_if_eq_meaning:NNTF #1 \__regex_compile_special:N 27086
7087     { 27087
7088         \token_if_eq_charcode:NNTF #2 . 27088
7089         { \use_none:n } 27089
7090         { \token_if_eq_charcode:NNF #2 ( } % ) 27090
7091     } 27091
7092     { \use:n } 27092
7093     { \msg_error:nnn { regex } { c-C-invalid } {#2} } 27093
7094     #1 #2 27094
7095 } 27095
7096 \cs_new_protected:cpn { __regex_compile_c[:w ] #1#2 27096
7097 { 27097
7098     \l__regex_mode_int 27098
7099     = \if_case:w \l__regex_mode_int 27099
7100         \c__regex_catcode_mode_int 27100
7101         \else: 27101
7102             \c__regex_catcode_in_class_mode_int 27102
7103             \fi: 27103
7104     \int_zero:N \l__regex_catcodes_int 27104
7105     \__regex_two_if_eq:NNNTF #1 #2 \__regex_compile_special:N ^ 27105
7106     { 27106
7107         \bool_set_false:N \l__regex_catcodes_bool 27107
7108         \__regex_compile_c_lbrack_loop:NN 27108
7109     } 27109
7110     { 27110
7111         \bool_set_true:N \l__regex_catcodes_bool 27111
7112         \__regex_compile_c_lbrack_loop:NN 27112
7113         #1 #2 27113
7114     } 27114
7115 } 27115
7116 \cs_new_protected:Npn \__regex_compile_c_lbrack_loop:NN #1#2 27116
7117 { 27117
7118     \token_if_eq_meaning:NNTF #1 \__regex_compile_raw:N 27118
7119     { 27119
7120         \int_if_exist:cTF { c__regex_catcode_#2_int } 27120
7121         { 27121
7122             \exp_args:Nc \__regex_compile_c_lbrack_add:N 27122
7123             { c__regex_catcode_#2_int } 27123
```

```

7124 \__regex_compile_c_lbrack_loop:NN
7125 }
7126 }
7127 {
7128 \token_if_eq_charcode:NNTF #2 ]
7129 { \__regex_compile_c_lbrack_end: }
7130 }
7131 {
7132 \msg_error:nne { regex } { c-missing-rbrack } {#2}
7133 \__regex_compile_c_lbrack_end:
7134 #1 #2
7135 }
7136 }
7137 \cs_new_protected:Npn \__regex_compile_c_lbrack_add:N #1
7138 {
7139 \if_int_odd:w \__regex_int_eval:w \l__regex_catcodes_int / #1 \scan_stop:
7140 \else:
7141 \int_add:Nn \l__regex_catcodes_int {#1}
7142 \fi:
7143 }
7144 \cs_new_protected:Npn \__regex_compile_c_lbrack_end:
7145 {
7146 \if_meaning:w \c_false_bool \l__regex_catcodes_bool
7147 \int_set:Nn \l__regex_catcodes_int
7148 { \c__regex_all_catcodes_int - \l__regex_catcodes_int }
7149 \fi:
7150 }
7151 \cs_new_protected:cpn { __regex_compile_c_ \c_left_brace_str :w }
7152 {
7153 \__regex_compile:w
7154 \__regex_disable_submatches:
7155 \l__regex_mode_int
7156 = \if_case:w \l__regex_mode_int
7157 \c__regex_cs_mode_int
7158 \else:
7159 \c__regex_cs_in_class_mode_int
7160 \fi:
7161 }
7162 \cs_new_protected:cpn { __regex_compile_ \c_left_brace_str : }
7163 {
7164 \__regex_if_in_cs:TF
7165 { \msg_error:nnn { regex } { cu-lbrace } { c } }
7166 { \exp_after:wN \__regex_compile_raw:N \c_left_brace_str }
7167 }
7168 \flag_new:N \l__regex_cs_flag
7169 \cs_new_protected:cpn { __regex_compile_ \c_right_brace_str : }

```

```

7170 {
7171     \__regex_if_in_cs:TF
7172     { \__regex_compile_end_cs: }
7173     { \exp_after:wN \__regex_compile_raw:N \c_right_brace_str }
7174 }
7175 \cs_new_protected:Npn \__regex_compile_end_cs:
7176 {
7177     \__regex_compile_end:
7178     \flag_clear:N \l__regex_cs_flag
7179     \__kernel_tl_set:Nx \l__regex_internal_a_tl
7180     {
7181         \exp_after:wN \__regex_compile_cs_aux:Nn \l__regex_internal_regex
7182         \q__regex_nil \q__regex_nil \q__regex_recursion_stop
7183     }
7184     \exp_args:Ne \__regex_compile_one:n
7185     {
7186         \flag_if_raised:NTF \l__regex_cs_flag
7187         { \__regex_item_cs:n { \exp_not:o \l__regex_internal_regex } }
7188         {
7189             \__regex_item_exact_cs:n
7190             { \tl_tail:N \l__regex_internal_a_tl }
7191         }
7192     }
7193 }
7194 \cs_new:Npn \__regex_compile_cs_aux:Nn #1#2
7195 {
7196     \cs_if_eq:NNTF #1 \__regex_branch:n
7197     {
7198         \scan_stop:
7199         \__regex_compile_cs_aux:NNnnnN #2
7200         \q__regex_nil \q__regex_nil \q__regex_nil
7201         \q__regex_nil \q__regex_nil \q__regex_nil \q__regex_recursion_stop
7202         \__regex_compile_cs_aux:Nn
7203     }
7204     {
7205         \__regex_quark_if_nil:NF #1 { \flag_ensure_raised:N \l__regex_cs_flag }
7206         \__regex_use_none_delimit_by_q_recursion_stop:w
7207     }
7208 }
7209 \cs_new:Npn \__regex_compile_cs_aux:NNnnnN #1#2#3#4#5#6
7210 {
7211     \bool_lazy_all:nTF
7212     {
7213         { \cs_if_eq_p:NN #1 \__regex_class:NnnnN }
7214         {#2}
7215         { \tl_if_head_eq_meaning_p:nN {#3} \__regex_item_caseful_equal:n }

```

```
7216 { \int_compare_p:nNn { \tl_count:n {#3} } = { 2 } }
7217 { \int_compare_p:nNn {#5} = \c_zero_int }
7218 }
7219 {
7220   \prg_replicate:nn {#4}
7221     { \char_generate:nn { \use_ii:nn #3 } {12} }
7222   \__regex_compile_cs_aux:NNnnnN
7223 }
7224 {
7225   \__regex_quark_if_nil:NF #1
7226   {
7227     \flag_ensure_raised:N \l__regex_cs_flag
7228     \__regex_use_i_delimit_by_q_recursion_stop:nw
7229   }
7230   \__regex_use_none_delimit_by_q_recursion_stop:w
7231 }
7232 }
7233 \cs_new_protected:cpn { __regex_compile_/u: } #1#2
7234 {
7235   \__regex_if_in_class_or_catcode:TF
7236   { \__regex_compile_raw_error:N u #1 #2 }
7237   {
7238     \__regex_two_if_eq:NNNNTF #1 #2 \__regex_compile_raw:N r
7239     { \__regex_compile_u_brace:NNN \__regex_compile_ur_end: }
7240     { \__regex_compile_u_brace:NNN \__regex_compile_u_end: #1 #2 }
7241   }
7242 }
7243 \cs_new:Npn \__regex_compile_u_brace:NNN #1#2#3
7244 {
7245   \__regex_two_if_eq:NNNNTF #2 #3 \__regex_compile_special:N \c_left_brace_str
7246   {
7247     \tl_set:Nn \l__regex_internal_b_tl {#1}
7248     \__kernel_tl_set:Nx \l__regex_internal_a_tl { \if_false: } \fi:
7249     \__regex_compile_u_loop:NN
7250   }
7251   {
7252     \msg_error:nn { regex } { u-missing-lbrace }
7253     \token_if_eq_meaning:NNTF #1 \__regex_compile_ur_end:
7254     { \__regex_compile_raw:N u \__regex_compile_raw:N r }
7255     { \__regex_compile_raw:N u }
7256     #2 #3
7257   }
7258 }
7259 \cs_new:Npn \__regex_compile_u_loop:NN #1#2
7260 {
7261   \token_if_eq_meaning:NNTF #1 \__regex_compile_raw:N
```

```
7262 { #2 \__regex_compile_u_loop:NN }
7263 {
7264     \token_if_eq_meaning:NNTF #1 \__regex_compile_special:N
7265     {
7266         \exp_after:wN \token_if_eq_charcode:NNTF \c_right_brace_str #2
7267         { \if_false: { \fi: } \l__regex_internal_b_tl }
7268         {
7269             \if_charcode:w \c_left_brace_str #2
7270             \msg_expandable_error:nnn { regex } { cu-lbrace } { u }
7271             \else:
7272                 #2
7273             \fi:
7274             \__regex_compile_u_loop:NN
7275         }
7276     }
7277     {
7278         \if_false: { \fi: }
7279         \msg_error:nne { regex } { u-missing-rbrace } {#2}
7280         \l__regex_internal_b_tl
7281         #1 #2
7282     }
7283 }
7284 }
7285 \cs_new_protected:Npn \__regex_compile_ur_end:
7286 {
7287     \group_begin:
7288     \cs_set:Npn \__regex_group:nnnN { \__regex_group_no_capture:nnnN }
7289     \cs_set:Npn \__regex_group_resetting:nnnN { \__regex_group_no_capture:nnnN }
7290     \exp_args:NNe
7291     \group_end:
7292     \__regex_compile_ur:n { \use:c { \l__regex_internal_a_tl } }
7293 }
7294 \cs_new_protected:Npn \__regex_compile_ur:n #1
7295 {
7296     \tl_if_empty:oTF { \__regex_compile_ur_aux:w #1 {} ? ? \q__regex_nil }
7297     { \__regex_compile_if_quantifier:TFw }
7298     { \use_i:nn }
7299     {
7300         \tl_build_put_right:Nn \l__regex_build_tl
7301         { \__regex_group_no_capture:nnnN { \if_false: } \fi: #1 }
7302         \__regex_compile_quantifier:w
7303     }
7304     { \tl_build_put_right:Nn \l__regex_build_tl { \use_ii:nn #1 } }
7305 }
7306 \cs_new:Npn \__regex_compile_ur_aux:w \__regex_branch:n #1#2#3 \q__regex_nil {#2}
7307 \cs_new_protected:Npn \__regex_compile_u_end:
```



```
7308 {
7309     \__regex_compile_if_quantifier:TFw
7310     {
7311         \tl_build_put_right:Nn \l__regex_build_tl
7312         {
7313             \__regex_group_no_capture:nnnN { \if_false: } \fi:
7314             \__regex_branch:n { \if_false: } \fi:
7315         }
7316         \__regex_compile_u_payload:
7317         \tl_build_put_right:Nn \l__regex_build_tl { \if_false: { \fi: } }
7318         \__regex_compile_quantifier:w
7319     }
7320     { \__regex_compile_u_payload: }
7321 }
7322 \cs_new_protected:Npn \__regex_compile_u_payload:
7323 {
7324     \tl_set:Nv \l__regex_internal_a_tl { \l__regex_internal_a_tl }
7325     \if_int_compare:w \l__regex_mode_int = \c__regex_outer_mode_int
7326         \__regex_compile_u_not_cs:
7327     \else:
7328         \__regex_compile_u_in_cs:
7329     \fi:
7330 }
7331 \cs_new_protected:Npn \__regex_compile_u_in_cs:
7332 {
7333     \__kernel_tl_gset:Nx \g__regex_internal_tl
7334     {
7335         \exp_args:No \__kernel_str_to_other_fast:n
7336         { \l__regex_internal_a_tl }
7337     }
7338     \tl_build_put_right:Ne \l__regex_build_tl
7339     {
7340         \tl_map_function:NN \g__regex_internal_tl
7341         \__regex_compile_u_in_cs_aux:n
7342     }
7343 }
7344 \cs_new:Npn \__regex_compile_u_in_cs_aux:n #1
7345 {
7346     \__regex_class:NnnnN \c_true_bool
7347     { \__regex_item_caseful_equal:n { \int_value:w `#1 } }
7348     { 1 } { 0 } \c_false_bool
7349 }
7350 \cs_new_protected:Npn \__regex_compile_u_not_cs:
7351 {
7352     \tl_analysis_map_inline:Nn \l__regex_internal_a_tl
7353     {
```

```
7354         \tl_build_put_right:Ne \l__regex_build_tl
7355     {
7356         \__regex_class:NnnnN \c_true_bool
7357     {
7358         \if_int_compare:w "##3 = \c_zero_int
7359             \__regex_item_exact_cs:n
7360             { \exp_after:wN \cs_to_str:N ##1 }
7361         \else:
7362             \__regex_item_exact:nn { \int_value:w "##3 } { ##2 }
7363         \fi:
7364     }
7365     { 1 } { 0 } \c_false_bool
7366 }
7367 }
7368 }
7369 \cs_new_protected:cpn { __regex_compile_/K: }
7370 {
7371     \int_compare:nNnTF \l__regex_mode_int = \c__regex_outer_mode_int
7372     { \tl_build_put_right:Nn \l__regex_build_tl { \__regex_command_K: } }
7373     { \__regex_compile_raw_error:N K }
7374 }
7375 \cs_new:Npn \__regex_clean_bool:n #1
7376 {
7377     \tl_if_single:nTF {#1}
7378     { \bool_if:NTF #1 \c_true_bool \c_false_bool }
7379     { \c_true_bool }
7380 }
7381 \cs_new:Npn \__regex_clean_int:n #1
7382 {
7383     \tl_if_head_eq_meaning:nNTF {#1} -
7384     { - \exp_args:No \__regex_clean_int:n { \use_none:n #1 } }
7385     { \int_eval:n { 0 \str_map_function:nN {#1} \__regex_clean_int_aux:N } }
7386 }
7387 \cs_new:Npn \__regex_clean_int_aux:N #1
7388 {
7389     \if_int_compare:w \c_one_int < 1 #1 ~
7390     #1
7391     \else:
7392         \str_map_break:n
7393     \fi:
7394 }
7395 \cs_new:Npn \__regex_clean_regex:n #1
7396 {
7397     \__regex_clean_regex_loop:w #1
7398     \__regex_branch:n { \q_recursion_tail } \q_recursion_stop
7399 }
```

```
7400 \cs_new:Npn \__regex_clean_regex_loop:w #1 \__regex_branch:n #2
7401 {
7402     \quark_if_recursion_tail_stop:n {#2}
7403     \__regex_branch:n { \__regex_clean_branch:n {#2} }
7404     \__regex_clean_regex_loop:w
7405 }
7406 \cs_new:Npn \__regex_clean_branch:n #1
7407 {
7408     \__regex_clean_branch_loop:n #1
7409     ? ? ? ? ? ? \prg_break_point:
7410 }
7411 \cs_new:Npn \__regex_clean_branch_loop:n #1
7412 {
7413     \tl_if_single:nF {#1} \prg_break:
7414     \token_case_meaning:NnF #1
7415     {
7416         \__regex_command_K: { #1 \__regex_clean_branch_loop:n }
7417         \__regex_assertion:Nn { #1 \__regex_clean_assertion:Nn }
7418         \__regex_class:NnnnN { #1 \__regex_clean_class:NnnnN }
7419         \__regex_group:nnnN { #1 \__regex_clean_group:nnnN }
7420         \__regex_group_no_capture:nnnN { #1 \__regex_clean_group:nnnN }
7421         \__regex_group_resetting:nnnN { #1 \__regex_clean_group:nnnN }
7422     }
7423     \prg_break:
7424 }
7425 \cs_new:Npn \__regex_clean_assertion:Nn #1#2
7426 {
7427     \__regex_clean_bool:n {#1}
7428     \tl_if_single:nF {#2} { { \__regex_A_test: } \prg_break: }
7429     \token_case_meaning:NnTF #2
7430     {
7431         \__regex_A_test: { }
7432         \__regex_G_test: { }
7433         \__regex_Z_test: { }
7434         \__regex_b_test: { }
7435     }
7436     { {#2} }
7437     { { \__regex_A_test: } \prg_break: }
7438     \__regex_clean_branch_loop:n
7439 }
7440 \cs_new:Npn \__regex_clean_class:NnnnN #1#2#3#4#5
7441 {
7442     \__regex_clean_bool:n {#1}
7443     { \__regex_clean_class:n {#2} }
7444     { \int_max:nn \c_zero_int { \__regex_clean_int:n {#3} } }
7445     { \int_max:nn { -\c_one_int } { \__regex_clean_int:n {#4} } }
```

```
7446     \__regex_clean_bool:n {#5}
7447     \__regex_clean_branch_loop:n
7448 }
7449 \cs_new:Npn \__regex_clean_group:nnnN #1#2#3#4
7450 {
7451     { \__regex_clean_regex:n {#1} }
7452     { \int_max:nn \c_zero_int { \__regex_clean_int:n {#2} } }
7453     { \int_max:nn { -\c_one_int } { \__regex_clean_int:n {#3} } }
7454     \__regex_clean_bool:n {#4}
7455     \__regex_clean_branch_loop:n
7456 }
7457 \cs_new:Npn \__regex_clean_class:n #1
7458 { \__regex_clean_class_loop:nnn #1 ????? \prg_break_point: }
7459 \cs_new:Npn \__regex_clean_class_loop:nnn #1#2#3
7460 {
7461     \tl_if_single:nF {#1} \prg_break:
7462     \token_case_meaning:NnTF #1
7463     {
7464         \__regex_item_cs:n { #1 { \__regex_clean_regex:n {#2} } }
7465         \__regex_item_exact_cs:n { #1 { \__regex_clean_exact_cs:n {#2} } }
7466         \__regex_item_caseful_equal:n { #1 { \__regex_clean_int:n {#2} } }
7467         \__regex_item_caseless_equal:n { #1 { \__regex_clean_int:n {#2} } }
7468         \__regex_item_reverse:n { #1 { \__regex_clean_class:n {#2} } }
7469     }
7470     { \__regex_clean_class_loop:nnn {#3} }
7471     {
7472         \token_case_meaning:NnTF #1
7473         {
7474             \__regex_item_caseful_range:nn { }
7475             \__regex_item_caseless_range:nn { }
7476             \__regex_item_exact:nn { }
7477         }
7478         {
7479             #1 { \__regex_clean_int:n {#2} } { \__regex_clean_int:n {#3} }
7480             \__regex_clean_class_loop:nnn
7481         }
7482         {
7483             \token_case_meaning:NnTF #1
7484             {
7485                 \__regex_item_catcode:nT { }
7486                 \__regex_item_catcode_reverse:nT { }
7487             }
7488             {
7489                 #1 { \__regex_clean_int:n {#2} } { \__regex_clean_class:n {#3} }
7490                 \__regex_clean_class_loop:nnn
7491             }
7492         }
7493     }
7494 }
```

```
7492         {
7493             \exp_args:Ne \str_case:nnTF
7494             {
7495                 \exp_args:Ne \str_range:nnn
7496                 { \cs_to_str:N #1 } \c_one_int { 13 }
7497             }
7498             {
7499                 { __regex_prop_ } { }
7500                 { __regex_posix } { }
7501             }
7502             {
7503                 #1
7504                 \__regex_clean_class_loop:nnn {#2} {#3}
7505             }
7506             \prg_break:
7507         }
7508     }
7509 }
7510 }
7511 \cs_new:Npn \__regex_clean_exact_cs:n #1
7512 {
7513     \exp_last_unbraced:Nf \use_none:n
7514     {
7515         \__regex_clean_exact_cs:w #1
7516         \scan_stop: \q_recursion_tail \scan_stop:
7517         \q_recursion_stop
7518     }
7519 }
7520 \cs_new:Npn \__regex_clean_exact_cs:w #1 \scan_stop:
7521 {
7522     \quark_if_recursion_tail_stop:n {#1}
7523     \scan_stop: \tl_to_str:n {#1}
7524     \__regex_clean_exact_cs:w
7525 }
7526 \cs_new_protected:Npn \__regex_show:N #1
7527 {
7528     \group_begin:
7529     \tl_build_begin:N \l__regex_build_tl
7530     \cs_set_protected:Npn \__regex_branch:n
7531     {
7532         \seq_pop_right:NN \l__regex_show_prefix_seq
7533         \l__regex_internal_a_tl
7534         \__regex_show_one:n { +-branch }
7535         \seq_put_right:No \l__regex_show_prefix_seq
7536         \l__regex_internal_a_tl
7537         \use:n
```

```
7538 }
7539 \cs_set_protected:Npn \__regex_group:nnnN
7540 { \__regex_show_group_aux:nnnnN { } }
7541 \cs_set_protected:Npn \__regex_group_no_capture:nnnN
7542 { \__regex_show_group_aux:nnnnN { ~(no~capture) } }
7543 \cs_set_protected:Npn \__regex_group_resetting:nnnN
7544 { \__regex_show_group_aux:nnnnN { ~(resetting) } }
7545 \cs_set_eq:NN \__regex_class:NnnnN \__regex_show_class:NnnnN
7546 \cs_set_protected:Npn \__regex_command_K:
7547 { \__regex_show_one:n { reset~match~start~(\iow_char:N\\K) } }
7548 \cs_set_protected:Npn \__regex_assertion:Nn ##1##2
7549 {
7550   \__regex_show_one:n
7551     { \bool_if:NF ##1 { negative~ } assertion:~##2 }
7552 }
7553 \cs_set:Npn \__regex_b_test: { word~boundary }
7554 \cs_set:Npn \__regex_Z_test: { anchor~at~end~(\iow_char:N\\Z) }
7555 \cs_set:Npn \__regex_A_test: { anchor~at~start~(\iow_char:N\\A) }
7556 \cs_set:Npn \__regex_G_test: { anchor~at~start~of~match~(\iow_char:N\\G) }
7557 \cs_set_protected:Npn \__regex_item_caseful_equal:n ##1
7558 { \__regex_show_one:n { char~code~\__regex_show_char:n{##1} } }
7559 \cs_set_protected:Npn \__regex_item_caseful_range:nn ##1##2
7560 {
7561   \__regex_show_one:n
7562     { range~[\__regex_show_char:n{##1}, \__regex_show_char:n{##2}] }
7563 }
7564 \cs_set_protected:Npn \__regex_item_caseless_equal:n ##1
7565 { \__regex_show_one:n { char~code~\__regex_show_char:n{##1}~(caseless) } }
7566 \cs_set_protected:Npn \__regex_item_caseless_range:nn ##1##2
7567 {
7568   \__regex_show_one:n
7569     { Range~[\__regex_show_char:n{##1}, \__regex_show_char:n{##2}]~(caseless) }
7570 }
7571 \cs_set_protected:Npn \__regex_item_catcode:nT
7572 { \__regex_show_item_catcode:NnT \c_true_bool }
7573 \cs_set_protected:Npn \__regex_item_catcode_reverse:nT
7574 { \__regex_show_item_catcode:NnT \c_false_bool }
7575 \cs_set_protected:Npn \__regex_item_reverse:n
7576 { \__regex_show_scope:nn { Reversed~match } }
7577 \cs_set_protected:Npn \__regex_item_exact:nn ##1##2
7578 { \__regex_show_one:n { char~\__regex_show_char:n{##2},~catcode~##1 } }
7579 \cs_set_eq:NN \__regex_item_exact_cs:n \__regex_show_item_exact_cs:n
7580 \cs_set_protected:Npn \__regex_item_cs:n
7581 { \__regex_show_scope:nn { control~sequence } }
7582 \cs_set:cpn { __regex_prop_.: } { \__regex_show_one:n { any~token } }
7583 \seq_clear:N \l__regex_show_prefix_seq
```

```
7584     \__regex_show_push:n { ~ }
7585     \cs_if_exist_use:N #1
7586     \tl_build_end:N \l__regex_build_tl
7587     \exp_args:NNNo
7588     \group_end:
7589     \tl_set:Nn \l__regex_internal_a_tl { \l__regex_build_tl }
7590 }
7591 \cs_new:Npn \__regex_show_char:n #1
7592 {
7593     \int_eval:n {#1}
7594     \int_compare:nT { 32 <= #1 <= 126 }
7595     { ~ ( \char_generate:nn {#1} {12} ) }
7596 }
7597 \cs_new_protected:Npn \__regex_show_one:n #1
7598 {
7599     \int_incr:N \l__regex_show_lines_int
7600     \tl_build_put_right:Ne \l__regex_build_tl
7601     {
7602         \exp_not:N \iow_newline:
7603         \seq_map_function:NN \l__regex_show_prefix_seq \use:n
7604         #1
7605     }
7606 }
7607 \cs_new_protected:Npn \__regex_show_push:n #1
7608 { \seq_put_right:Ne \l__regex_show_prefix_seq { #1 ~ } }
7609 \cs_new_protected:Npn \__regex_show_pop:
7610 { \seq_pop_right:NN \l__regex_show_prefix_seq \l__regex_internal_a_tl }
7611 \cs_new_protected:Npn \__regex_show_scope:nn #1#2
7612 {
7613     \__regex_show_one:n {#1}
7614     \__regex_show_push:n { ~ }
7615     #2
7616     \__regex_show_pop:
7617 }
7618 \cs_new_protected:Npn \__regex_show_group_aux:nnnnN #1#2#3#4#5
7619 {
7620     \__regex_show_one:n { , -group~begin #1 }
7621     \__regex_show_push:n { | }
7622     \use_ii:nn #2
7623     \__regex_show_pop:
7624     \__regex_show_one:n
7625     { ` -group~end \__regex_msg_repeated:nnN {#3} {#4} #5 }
7626 }
7627 \cs_new:Npn \__regex_show_class:NnnnN #1#2#3#4#5
7628 {
7629     \group_begin:
```



```
7630 \tl_build_begin:N \l__regex_build_tl
7631 \int_zero:N \l__regex_show_lines_int
7632 \__regex_show_push:n {~}
7633 #2
7634 \int_compare:nTF { \l__regex_show_lines_int = \c_zero_int }
7635 {
7636   \group_end:
7637   \__regex_show_one:n { \bool_if:NTF #1 { Fail } { Pass } }
7638 }
7639 {
7640   \bool_if:NTF
7641     { #1 && \int_compare_p:n { \l__regex_show_lines_int = \c_one_int } }
7642     {
7643       \group_end:
7644       #2
7645       \tl_build_put_right:Nn \l__regex_build_tl
7646         { \__regex_msg_repeated:nnN {#3} {#4} #5 }
7647     }
7648     {
7649       \tl_build_end:N \l__regex_build_tl
7650       \exp_args:NNNo
7651       \group_end:
7652       \tl_set:Nn \l__regex_internal_a_tl \l__regex_build_tl
7653       \__regex_show_one:n
7654         {
7655           \bool_if:NTF #1 { Match } { Don't~match }
7656           \__regex_msg_repeated:nnN {#3} {#4} #5
7657         }
7658       \tl_build_put_right:Ne \l__regex_build_tl
7659         { \exp_not:o \l__regex_internal_a_tl }
7660     }
7661 }
7662 }
7663 \cs_new_protected:Npn \__regex_show_item_catcode:NnT #1#2
7664 {
7665   \seq_set_split:Nnn \l__regex_internal_seq { } { CBEMTPUDSLOA }
7666   \seq_set_filter:NNn \l__regex_internal_seq \l__regex_internal_seq
7667     { \int_if_odd_p:n { #2 / \int_use:c { c__regex_catcode_##1_int } } }
7668   \__regex_show_scope:nn
7669     {
7670       categories~
7671       \seq_map_function:NN \l__regex_internal_seq \use:n
7672       , ~
7673       \bool_if:NF #1 { negative~ } class
7674     }
7675 }
```

```
7676 \cs_new_protected:Npn \__regex_show_item_exact_cs:n #1
7677 {
7678   \seq_set_split:Nnn \l__regex_internal_seq { \scan_stop: } {#1}
7679   \seq_set_map_e:NNn \l__regex_internal_seq
7680     \l__regex_internal_seq { \iow_char:N\|#1 }
7681   \__regex_show_one:n
7682   { control~sequence~ \seq_use:Nn \l__regex_internal_seq { ~or~ } }
7683 }
7684 \int_new:N \l__regex_min_state_int
7685 \int_set:Nn \l__regex_min_state_int { 1 }
7686 \int_new:N \l__regex_max_state_int
7687 \int_new:N \l__regex_left_state_int
7688 \int_new:N \l__regex_right_state_int
7689 \seq_new:N \l__regex_left_state_seq
7690 \seq_new:N \l__regex_right_state_seq
7691 \int_new:N \l__regex_capturing_group_int
7692 \cs_new_protected:Npn \__regex_build:n
7693 { \__regex_build_aux:Nn \c_true_bool }
7694 \cs_new_protected:Npn \__regex_build:N
7695 { \__regex_build_aux:NN \c_true_bool }
7696 \cs_new_protected:Npn \__regex_build_aux:Nn #1#2
7697 {
7698   \__regex_compile:n {#2}
7699   \__regex_build_aux:NN #1 \l__regex_internal_regex
7700 }
7701 \cs_new_protected:Npn \__regex_build_aux:NN #1#2
7702 {
7703   \__regex_standard_escapechar:
7704   \int_zero:N \l__regex_capturing_group_int
7705   \int_set_eq:NN \l__regex_max_state_int \l__regex_min_state_int
7706   \__regex_build_new_state:
7707   \__regex_build_new_state:
7708   \__regex_toks_put_right:Nn \l__regex_left_state_int
7709     { \__regex_action_start_wildcard:N #1 }
7710   \__regex_group:nnnN {#2} { 1 } { 0 } \c_false_bool
7711   \__regex_toks_put_right:Nn \l__regex_right_state_int
7712     { \__regex_action_success: }
7713 }
7714 \int_new:N \g__regex_case_int
7715 \int_new:N \l__regex_case_max_group_int
7716 \cs_new_protected:Npn \__regex_case_build:n #1
7717 {
7718   \__regex_case_build_aux:Nn \c_true_bool {#1}
7719   \int_gzero:N \g__regex_case_int
7720 }
7721 \cs_generate_variant:Nn \__regex_case_build:n { e }
```

```
7722 \cs_new_protected:Npn \__regex_case_build_aux:Nn #1#2
7723 {
7724   \__regex_standard_escapechar:
7725   \int_set_eq:NN \l__regex_max_state_int \l__regex_min_state_int
7726   \__regex_build_new_state:
7727   \__regex_build_new_state:
7728   \__regex_toks_put_right:Nn \l__regex_left_state_int
7729   { \__regex_action_start_wildcard:N #1 }
7730   %
7731   \__regex_build_new_state:
7732   \__regex_toks_put_left:Ne \l__regex_left_state_int
7733   { \__regex_action_submatch:nN \c_zero_int < }
7734   \__regex_push_lr_states:
7735   \int_zero:N \l__regex_case_max_group_int
7736   \int_gzero:N \g__regex_case_int
7737   \tl_map_inline:nn {#2}
7738   {
7739     \int_gincr:N \g__regex_case_int
7740     \__regex_case_build_loop:n {##1}
7741   }
7742   \int_set_eq:NN \l__regex_capturing_group_int \l__regex_case_max_group_int
7743   \__regex_pop_lr_states:
7744   }
7745 \cs_new_protected:Npn \__regex_case_build_loop:n #1
7746 {
7747   \int_set_eq:NN \l__regex_capturing_group_int \c_one_int
7748   \__regex_compile_use:n {#1}
7749   \int_set:Nn \l__regex_case_max_group_int
7750   { \int_max:nn \l__regex_case_max_group_int \l__regex_capturing_group_int }
7751   \seq_pop:NN \l__regex_right_state_seq \l__regex_internal_a_tl
7752   \int_set:Nn \l__regex_right_state_int \l__regex_internal_a_tl
7753   \__regex_toks_put_left:Ne \l__regex_right_state_int
7754   {
7755     \__regex_action_submatch:nN \c_zero_int >
7756     \int_gset:Nn \g__regex_case_int
7757     { \int_use:N \g__regex_case_int }
7758     \__regex_action_success:
7759   }
7760   \__regex_toks_clear:N \l__regex_max_state_int
7761   \seq_push:No \l__regex_right_state_seq
7762   { \int_use:N \l__regex_max_state_int }
7763   \int_incr:N \l__regex_max_state_int
7764   }
7765 \cs_new_protected:Npn \__regex_build_for_cs:n #1
7766 {
7767   \int_set_eq:NN \l__regex_min_state_int \l__regex_max_state_int
```

```

7768 \__regex_build_new_state:
7769 \__regex_build_new_state:
7770 \__regex_push_lr_states:
7771 #1
7772 \__regex_pop_lr_states:
7773 \__regex_toks_put_right:Nn \l__regex_right_state_int
7774 {
7775     \if_int_compare:w -2 = \l__regex_curr_char_int
7776     \exp_after:wN \__regex_action_success:
7777     \fi:
7778 }
7779 }
7780 \cs_new_protected:Npn \__regex_push_lr_states:
7781 {
7782     \seq_push:No \l__regex_left_state_seq
7783     { \int_use:N \l__regex_left_state_int }
7784     \seq_push:No \l__regex_right_state_seq
7785     { \int_use:N \l__regex_right_state_int }
7786 }
7787 \cs_new_protected:Npn \__regex_pop_lr_states:
7788 {
7789     \seq_pop:NN \l__regex_left_state_seq \l__regex_internal_a_tl
7790     \int_set:Nn \l__regex_left_state_int \l__regex_internal_a_tl
7791     \seq_pop:NN \l__regex_right_state_seq \l__regex_internal_a_tl
7792     \int_set:Nn \l__regex_right_state_int \l__regex_internal_a_tl
7793 }
7794 \cs_new_protected:Npn \__regex_build_transition_left:NNN #1#2#3
7795 { \__regex_toks_put_left:Ne #2 { #1 { \tex_the:D \__regex_int_eval:w #3 - #2 } } }
7796 \cs_new_protected:Npn \__regex_build_transition_right:nNn #1#2#3
7797 { \__regex_toks_put_right:Ne #2 { #1 { \tex_the:D \__regex_int_eval:w #3 - #2 } } }
7798 \cs_new_protected:Npn \__regex_build_new_state:
7799 {
7800     \__regex_toks_clear:N \l__regex_max_state_int
7801     \int_set_eq:NN \l__regex_left_state_int \l__regex_right_state_int
7802     \int_set_eq:NN \l__regex_right_state_int \l__regex_max_state_int
7803     \int_incr:N \l__regex_max_state_int
7804 }
7805 \cs_new_protected:Npn \__regex_build_transitions_laziness:NNNNN #1#2#3#4#5
7806 {
7807     \__regex_build_new_state:
7808     \__regex_toks_put_right:Ne \l__regex_left_state_int
7809     {
7810         \if_meaning:w \c_true_bool #1
7811             #2 { \tex_the:D \__regex_int_eval:w #3 - \l__regex_left_state_int }
7812             #4 { \tex_the:D \__regex_int_eval:w #5 - \l__regex_left_state_int }
7813         \else:

```

```
7814         #4 { \tex_the:D \_regex_int_eval:w #5 - \l__regex_left_state_int } 27814
7815         #2 { \tex_the:D \_regex_int_eval:w #3 - \l__regex_left_state_int } 27815
7816     \fi: 27816
7817 } 27817
7818 } 27818
7819 \cs_new_protected:Npn \_regex_class:NnnnN #1#2#3#4#5 27819
7820 { 27820
7821     \cs_set:Npe \_regex_tests_action_cost:n ##1 27821
7822     { 27822
7823         \exp_not:n { \exp_not:n {#2} } 27823
7824         \bool_if:NTF #1 27824
7825         { \_regex_break_point:TF { \_regex_action_cost:n {##1} } { } } 27825
7826         { \_regex_break_point:TF { } { \_regex_action_cost:n {##1} } } 27826
7827     } 27827
7828     \if_case:w - #4 \exp_stop_f: 27828
7829         \_regex_class_repeat:n {#3} 27829
7830     \or: \_regex_class_repeat:nN {#3} #5 27830
7831     \else: \_regex_class_repeat:nnN {#3} {#4} #5 27831
7832     \fi: 27832
7833 } 27833
7834 \cs_new:Npn \_regex_tests_action_cost:n { \_regex_action_cost:n } 27834
7835 \cs_new_protected:Npn \_regex_class_repeat:n #1 27835
7836 { 27836
7837     \prg_replicate:nn {#1} 27837
7838     { 27838
7839         \_regex_build_new_state: 27839
7840         \_regex_build_transition_right:nNn \_regex_tests_action_cost:n 27840
7841         \l__regex_left_state_int \l__regex_right_state_int 27841
7842     } 27842
7843 } 27843
7844 \cs_new_protected:Npn \_regex_class_repeat:nN #1#2 27844
7845 { 27845
7846     \if_int_compare:w #1 = \c_zero_int 27846
7847         \_regex_build_transitions_laziness:NNNNN #2 27847
7848         \_regex_action_free:n \l__regex_right_state_int 27848
7849         \_regex_tests_action_cost:n \l__regex_left_state_int 27849
7850     \else: 27850
7851         \_regex_class_repeat:n {#1} 27851
7852         \int_set_eq:NN \l__regex_internal_a_int \l__regex_left_state_int 27852
7853         \_regex_build_transitions_laziness:NNNNN #2 27853
7854         \_regex_action_free:n \l__regex_right_state_int 27854
7855         \_regex_action_free:n \l__regex_internal_a_int 27855
7856     \fi: 27856
7857 } 27857
7858 \cs_new_protected:Npn \_regex_class_repeat:nnN #1#2#3 27858
7859 { 27859
```

```
7860 \__regex_class_repeat:n {#1}
7861 \int_set:Nn \l__regex_internal_a_int
7862 { \l__regex_max_state_int + #2 - \c_one_int }
7863 \prg_replicate:nn { #2 }
7864 {
7865   \__regex_build_transitions_laziness:NNNN #3
7866   \__regex_action_free:n \l__regex_internal_a_int
7867   \__regex_tests_action_cost:n \l__regex_right_state_int
7868 }
7869 }
7870 \cs_new_protected:Npn \__regex_group_aux:nnnnN #1#2#3#4#5
7871 {
7872   \if_int_compare:w #3 = \c_zero_int
7873     \__regex_build_new_state:
7874     \__regex_build_transition_right:nNn \__regex_action_free_group:n
7875     \l__regex_left_state_int \l__regex_right_state_int
7876   \fi:
7877   \__regex_build_new_state:
7878   \__regex_push_lr_states:
7879   #2
7880   \__regex_pop_lr_states:
7881   \if_case:w - #4 \exp_stop_f:
7882     \__regex_group_repeat:nn {#1} {#3}
7883   \or: \__regex_group_repeat:nnN {#1} {#3} #5
7884   \else: \__regex_group_repeat:nnnN {#1} {#3} {#4} #5
7885   \fi:
7886 }
7887 \cs_new_protected:Npn \__regex_group:nnnN #1
7888 {
7889   \exp_args:No \__regex_group_aux:nnnnN
7890   { \int_use:N \l__regex_capturing_group_int }
7891   {
7892     \int_incr:N \l__regex_capturing_group_int
7893     #1
7894   }
7895 }
7896 \cs_new_protected:Npn \__regex_group_no_capture:nnnN
7897 { \__regex_group_aux:nnnnN { -1 } }
7898 \cs_new_protected:Npn \__regex_group_resetting:nnnN #1
7899 {
7900   \__regex_group_aux:nnnnN { -1 }
7901   {
7902     \exp_args:Noo \__regex_group_resetting_loop:nnNn
7903     { \int_use:N \l__regex_capturing_group_int }
7904     { \int_use:N \l__regex_capturing_group_int }
7905     #1
```

```
7906 { ?? \prg_break:n } { }
7907 \prg_break_point:
7908 }
7909 }
7910 \cs_new_protected:Npn \__regex_group_resetting_loop:nnNn #1#2#3#4
7911 {
7912   \use_none:nn #3 { \int_set:Nn \l__regex_capturing_group_int {#1} }
7913   \int_set:Nn \l__regex_capturing_group_int {#2}
7914   #3 {#4}
7915   \exp_args:Ne \__regex_group_resetting_loop:nnNn
7916     { \int_max:nn {#1} \l__regex_capturing_group_int }
7917     {#2}
7918 }
7919 \cs_new_protected:Npn \__regex_branch:n #1
7920 {
7921   \__regex_build_new_state:
7922   \seq_get:NN \l__regex_left_state_seq \l__regex_internal_a_tl
7923   \int_set:Nn \l__regex_left_state_int \l__regex_internal_a_tl
7924   \__regex_build_transition_right:nNn \__regex_action_free:n
7925     \l__regex_left_state_int \l__regex_right_state_int
7926   #1
7927   \seq_get:NN \l__regex_right_state_seq \l__regex_internal_a_tl
7928   \__regex_build_transition_right:nNn \__regex_action_free:n
7929     \l__regex_right_state_int \l__regex_internal_a_tl
7930 }
7931 \cs_new_protected:Npn \__regex_group_repeat:nn #1#2
7932 {
7933   \if_int_compare:w #2 = \c_zero_int
7934     \int_set:Nn \l__regex_max_state_int
7935       { \l__regex_left_state_int - \c_one_int }
7936     \__regex_build_new_state:
7937   \else:
7938     \__regex_group_repeat_aux:n {#2}
7939     \__regex_group_submatches:nNN {#1}
7940     \l__regex_internal_a_int \l__regex_right_state_int
7941     \__regex_build_new_state:
7942   \fi:
7943 }
7944 \cs_new_protected:Npn \__regex_group_submatches:nNN #1#2#3
7945 {
7946   \if_int_compare:w #1 > - \c_one_int
7947     \__regex_toks_put_left:Ne #2 { \__regex_action_submatch:nN {#1} < }
7948     \__regex_toks_put_left:Ne #3 { \__regex_action_submatch:nN {#1} > }
7949   \fi:
7950 }
7951 \cs_new_protected:Npn \__regex_group_repeat_aux:n #1
```



```
7952 {
7953     \__regex_build_transition_right:nNn \__regex_action_free:n
7954     \l__regex_right_state_int \l__regex_max_state_int
7955     \int_set_eq:NN \l__regex_internal_a_int \l__regex_left_state_int
7956     \int_set_eq:NN \l__regex_internal_b_int \l__regex_max_state_int
7957     \if_int_compare:w \__regex_int_eval:w #1 > \c_one_int
7958         \int_set:Nn \l__regex_internal_c_int
7959         {
7960             ( #1 - \c_one_int )
7961             * ( \l__regex_internal_b_int - \l__regex_internal_a_int )
7962         }
7963     \int_add:Nn \l__regex_right_state_int \l__regex_internal_c_int
7964     \int_add:Nn \l__regex_max_state_int \l__regex_internal_c_int
7965     \__regex_toks_memcpy:NNn
7966         \l__regex_internal_b_int
7967         \l__regex_internal_a_int
7968         \l__regex_internal_c_int
7969     \fi:
7970 }
7971 \cs_new_protected:Npn \__regex_group_repeat:nnN #1#2#3
7972 {
7973     \if_int_compare:w #2 = \c_zero_int
7974         \__regex_group_submatches:nNN {#1}
7975         \l__regex_left_state_int \l__regex_right_state_int
7976         \int_set:Nn \l__regex_internal_a_int
7977             { \l__regex_left_state_int - \c_one_int }
7978         \__regex_build_transition_right:nNn \__regex_action_free:n
7979             \l__regex_right_state_int \l__regex_internal_a_int
7980         \__regex_build_new_state:
7981         \if_meaning:w \c_true_bool #3
7982             \__regex_build_transition_left:NNN \__regex_action_free:n
7983                 \l__regex_internal_a_int \l__regex_right_state_int
7984         \else:
7985             \__regex_build_transition_right:nNn \__regex_action_free:n
7986                 \l__regex_internal_a_int \l__regex_right_state_int
7987         \fi:
7988     \else:
7989         \__regex_group_repeat_aux:n {#2}
7990         \__regex_group_submatches:nNN {#1}
7991         \l__regex_internal_a_int \l__regex_right_state_int
7992         \if_meaning:w \c_true_bool #3
7993             \__regex_build_transition_right:nNn \__regex_action_free_group:n
7994                 \l__regex_right_state_int \l__regex_internal_a_int
7995         \else:
7996             \__regex_build_transition_left:NNN \__regex_action_free_group:n
7997                 \l__regex_right_state_int \l__regex_internal_a_int
```

```
7998     \fi:
7999     \__regex_build_new_state:
8000 \fi:
8001 }
8002 \cs_new_protected:Npn \__regex_group_repeat:nnnN #1#2#3#4
8003 {
8004     \__regex_group_submatches:nNN {#1}
8005     \l__regex_left_state_int \l__regex_right_state_int
8006     \__regex_group_repeat_aux:n { #2 + #3 }
8007     \if_meaning:w \c_true_bool #4
8008         \int_set_eq:NN \l__regex_left_state_int \l__regex_max_state_int
8009         \prg_replicate:nn { #3 }
8010         {
8011             \int_sub:Nn \l__regex_left_state_int
8012             { \l__regex_internal_b_int - \l__regex_internal_a_int }
8013             \__regex_build_transition_left:NNN \__regex_action_free:n
8014             \l__regex_left_state_int \l__regex_max_state_int
8015         }
8016     \else:
8017         \prg_replicate:nn { #3 - \c_one_int }
8018         {
8019             \int_sub:Nn \l__regex_right_state_int
8020             { \l__regex_internal_b_int - \l__regex_internal_a_int }
8021             \__regex_build_transition_right:nNn \__regex_action_free:n
8022             \l__regex_right_state_int \l__regex_max_state_int
8023         }
8024     \if_int_compare:w #2 = \c_zero_int
8025         \int_set:Nn \l__regex_right_state_int
8026         { \l__regex_left_state_int - \c_one_int }
8027     \else:
8028         \int_sub:Nn \l__regex_right_state_int
8029         { \l__regex_internal_b_int - \l__regex_internal_a_int }
8030     \fi:
8031     \__regex_build_transition_right:nNn \__regex_action_free:n
8032     \l__regex_right_state_int \l__regex_max_state_int
8033 \fi:
8034 \__regex_build_new_state:
8035 }
8036 \cs_new_protected:Npn \__regex_assertion:Nn #1#2
8037 {
8038     \__regex_build_new_state:
8039     \__regex_toks_put_right:Ne \l__regex_left_state_int
8040     {
8041         \exp_not:n {#2}
8042         \__regex_break_point:TF
8043         \bool_if:NF #1 { { } }
```

```
8044         {
8045             \_regex_action_free:n
8046         {
8047             \tex_the:D \_regex_int_eval:w
8048             \l__regex_right_state_int - \l__regex_left_state_int
8049         }
8050     }
8051     \bool_if:NT #1 { { } }
8052 }
8053 }
8054 \cs_new_protected:Npn \_regex_b_test:
8055 {
8056     \group_begin:
8057     \int_set_eq:NN \l__regex_curr_char_int \l__regex_last_char_int
8058     \_regex_prop_w:
8059     \_regex_break_point:TF
8060     { \group_end: \_regex_item_reverse:n { \_regex_prop_w: } }
8061     { \group_end: \_regex_prop_w: }
8062 }
8063 \cs_new_protected:Npn \_regex_Z_test:
8064 {
8065     \if_int_compare:w -2 = \l__regex_curr_char_int
8066     \exp_after:wN \_regex_break_true:w
8067     \fi:
8068 }
8069 \cs_new_protected:Npn \_regex_A_test:
8070 {
8071     \if_int_compare:w -2 = \l__regex_last_char_int
8072     \exp_after:wN \_regex_break_true:w
8073     \fi:
8074 }
8075 \cs_new_protected:Npn \_regex_G_test:
8076 {
8077     \if_int_compare:w \l__regex_curr_pos_int = \l__regex_start_pos_int
8078     \exp_after:wN \_regex_break_true:w
8079     \fi:
8080 }
8081 \cs_new_protected:Npn \_regex_command_K:
8082 {
8083     \_regex_build_new_state:
8084     \_regex_toks_put_right:Ne \l__regex_left_state_int
8085     {
8086         \_regex_action_submatch:nN \c_zero_int <
8087         \bool_set_true:N \l__regex_fresh_thread_bool
8088         \_regex_action_free:n
8089     }
```

```

8090         \tex_the:D \__regex_int_eval:w
8091         \l__regex_right_state_int - \l__regex_left_state_int
8092     }
8093     \bool_set_false:N \l__regex_fresh_thread_bool
8094 }
8095 }
8096 \int_new:N \l__regex_min_pos_int
8097 \int_new:N \l__regex_max_pos_int
8098 \int_new:N \l__regex_curr_pos_int
8099 \int_new:N \l__regex_start_pos_int
8100 \int_new:N \l__regex_success_pos_int
8101 \int_new:N \l__regex_curr_char_int
8102 \int_new:N \l__regex_curr_catcode_int
8103 \tl_new:N \l__regex_curr_token_tl
8104 \int_new:N \l__regex_last_char_int
8105 \int_new:N \l__regex_last_char_success_int
8106 \int_new:N \l__regex_case_changed_char_int
8107 \int_new:N \l__regex_curr_state_int
8108 \tl_new:N \l__regex_curr_submatches_tl
8109 \tl_new:N \l__regex_success_submatches_tl
8110 \int_new:N \l__regex_step_int
8111 \int_new:N \l__regex_min_thread_int
8112 \int_new:N \l__regex_max_thread_int
8113 \intarray_new:Nn \g__regex_state_active_intarray { 65536 }
8114 \intarray_new:Nn \g__regex_thread_info_intarray { 65536 }
8115 \tl_new:N \l__regex_matched_analysis_tl
8116 \tl_new:N \l__regex_curr_analysis_tl
8117 \tl_new:N \l__regex_every_match_tl
8118 \bool_new:N \l__regex_fresh_thread_bool
8119 \bool_new:N \l__regex_empty_success_bool
8120 \cs_new_eq:NN \__regex_if_two_empty_matches:F \use:n
8121 \bool_new:N \g__regex_success_bool
8122 \bool_new:N \l__regex_saved_success_bool
8123 \bool_new:N \l__regex_match_success_bool
8124 \cs_new_protected:Npn \__regex_match:n #1
8125 {
8126     \__regex_match_init:
8127     \__regex_match_once_init:
8128     \tl_analysis_map_inline:nn {#1}
8129         { \__regex_match_one_token:nnN {##1} {##2} ##3 }
8130     \__regex_match_one_token:nnN { } { -2 } F
8131     \prg_break_point:Nn \__regex_maplike_break: { }
8132 }
8133 \cs_new_protected:Npn \__regex_match_cs:n #1
8134 {
8135     \int_set_eq:NN \l__regex_min_thread_int \l__regex_max_thread_int

```

```

8136 \__regex_match_init:
8137 \__regex_match_once_init:
8138 \str_map_inline:nn {#1}
8139 {
8140     \tl_if_blank:nTF {##1}
8141         { \__regex_match_one_token:nnN {##1} {`##1} A }
8142         { \__regex_match_one_token:nnN {##1} {`##1} C }
8143     }
8144     \__regex_match_one_token:nnN { } { -2 } F
8145     \prg_break_point:Nn \__regex_maplike_break: { }
8146 }
8147 \cs_new_protected:Npn \__regex_match_init:
8148 {
8149     \bool_gset_false:N \g__regex_success_bool
8150     \int_step_inline:nnn
8151         \l__regex_min_state_int { \l__regex_max_state_int - \c_one_int }
8152     {
8153         \__kernel_intarray_gset:Nnn
8154         \g__regex_state_active_intarray {##1} \c_one_int
8155     }
8156     \int_zero:N \l__regex_step_int
8157     \int_set:Nn \l__regex_min_pos_int { 2 }
8158     \int_set_eq:NN \l__regex_success_pos_int \l__regex_min_pos_int
8159     \int_set:Nn \l__regex_last_char_success_int { -2 }
8160     \tl_build_begin:N \l__regex_matched_analysis_tl
8161     \tl_clear:N \l__regex_curr_analysis_tl
8162     \int_set_eq:NN \l__regex_min_submatch_int \c_one_int
8163     \int_set_eq:NN \l__regex_submatch_int \l__regex_min_submatch_int
8164     \bool_set_false:N \l__regex_empty_success_bool
8165 }
8166 \cs_new_protected:Npn \__regex_match_once_init:
8167 {
8168     \if_meaning:w \c_true_bool \l__regex_empty_success_bool
8169         \cs_set:Npn \__regex_if_two_empty_matches:F
8170         {
8171             \int_compare:nNnF
8172                 \l__regex_start_pos_int = \l__regex_curr_pos_int
8173             }
8174     \else:
8175         \cs_set_eq:NN \__regex_if_two_empty_matches:F \use:n
8176     \fi:
8177     \int_set_eq:NN \l__regex_start_pos_int \l__regex_success_pos_int
8178     \bool_set_false:N \l__regex_match_success_bool
8179     \tl_set:Ne \l__regex_curr_submatches_tl
8180         { \prg_replicate:nn { 2 * \l__regex_capturing_group_int } { 0 , } }
8181     \int_set_eq:NN \l__regex_max_thread_int \l__regex_min_thread_int

```

```
8182 \__regex_store_state:n { \l__regex_min_state_int } 28182
8183 \int_set:Nn \l__regex_curr_pos_int { \l__regex_start_pos_int - \c_one_int } 28183
8184 \int_set_eq:NN \l__regex_curr_char_int \l__regex_last_char_success_int 28184
8185 \tl_build_get_intermediate:NN \l__regex_matched_analysis_tl \l__regex_internal_a_tl 28185
8186 \exp_args:NNf \__regex_match_once_init_aux: 28186
8187 \tl_map_inline:nn 28187
8188 { \exp_after:wN \l__regex_internal_a_tl \l__regex_curr_analysis_tl } 28188
8189 { \__regex_match_one_token:nnN ##1 } 28189
8190 \prg_break_point:Nn \__regex_maplike_break: { } 28190
8191 } 28191
8192 \cs_new_protected:Npn \__regex_match_once_init_aux: 28192
8193 { 28193
8194 \tl_build_begin:N \l__regex_matched_analysis_tl 28194
8195 \tl_clear:N \l__regex_curr_analysis_tl 28195
8196 } 28196
8197 \cs_new_protected:Npn \__regex_single_match: 28197
8198 { 28198
8199 \tl_set:Nn \l__regex_every_match_tl 28199
8200 { 28200
8201 \bool_gset_eq:NN 28201
8202 \g__regex_success_bool 28202
8203 \l__regex_match_success_bool 28203
8204 \__regex_maplike_break: 28204
8205 } 28205
8206 } 28206
8207 \cs_new_protected:Npn \__regex_multi_match:n #1 28207
8208 { 28208
8209 \tl_set:Nn \l__regex_every_match_tl 28209
8210 { 28210
8211 \if_meaning:w \c_false_bool \l__regex_match_success_bool 28211
8212 \exp_after:wN \__regex_maplike_break: 28212
8213 \fi: 28213
8214 \bool_gset_true:N \g__regex_success_bool 28214
8215 #1 28215
8216 \__regex_match_once_init: 28216
8217 } 28217
8218 } 28218
8219 \cs_new_protected:Npn \__regex_match_one_token:nnN #1#2#3 28219
8220 { 28220
8221 \int_add:Nn \l__regex_step_int { 2 } 28221
8222 \int_incr:N \l__regex_curr_pos_int 28222
8223 \int_set_eq:NN \l__regex_last_char_int \l__regex_curr_char_int 28223
8224 \cs_set_eq:NN \__regex_maybe_compute_ccc: \__regex_compute_case_changed_char: 28224
8225 \tl_set:Nn \l__regex_curr_token_tl {#1} 28225
8226 \int_set:Nn \l__regex_curr_char_int {#2} 28226
8227 \int_set:Nn \l__regex_curr_catcode_int { "#3 } 28227
```



```
8228 \tl_build_put_right:Ne \l__regex_matched_analysis_tl
8229 { \exp_not:o \l__regex_curr_analysis_tl }
8230 \tl_set:Nn \l__regex_curr_analysis_tl { { {#1} {#2} #3 } }
8231 \use:e
8232 {
8233   \int_set_eq:NN \l__regex_max_thread_int \l__regex_min_thread_int
8234   \int_step_function:nnN
8235     \l__regex_min_thread_int
8236     { \l__regex_max_thread_int - \c_one_int }
8237     \__regex_match_one_active:n
8238 }
8239 \prg_break_point:
8240 \bool_set_false:N \l__regex_fresh_thread_bool
8241 \if_int_compare:w \l__regex_max_thread_int > \l__regex_min_thread_int
8242   \if_int_compare:w -2 < \l__regex_curr_char_int
8243     \exp_after:wN \use_i:nn
8244     \fi:
8245   \fi:
8246   \l__regex_every_match_tl
8247 }
8248 \cs_new:Npn \__regex_match_one_active:n #1
8249 {
8250   \__regex_use_state_and_submatches:w
8251   \__kernel_intarray_range_to_clist:Nnn
8252     \g__regex_thread_info_intarray
8253     { \c_one_int + #1 * (\l__regex_capturing_group_int * 2 + \c_one_int) }
8254     { (\c_one_int + #1) * (\l__regex_capturing_group_int * 2 + \c_one_int) }
8255   \__regex_sep:
8256 }
8257 \cs_new_protected:Npn \__regex_use_state:
8258 {
8259   \__kernel_intarray_gset:Nnn \g__regex_state_active_intarray
8260     \l__regex_curr_state_int \l__regex_step_int
8261   \__regex_toks_use:w \l__regex_curr_state_int
8262   \__kernel_intarray_gset:Nnn \g__regex_state_active_intarray
8263     \l__regex_curr_state_int
8264     { \__regex_int_eval:w \l__regex_step_int + \c_one_int \scan_stop: }
8265 }
8266 \cs_new_protected:Npn \__regex_use_state_and_submatches:w #1 , #2 \__regex_sep:
8267 {
8268   \int_set:Nn \l__regex_curr_state_int {#1}
8269   \if_int_compare:w
8270     \__kernel_intarray_item:Nn \g__regex_state_active_intarray
8271     \l__regex_curr_state_int
8272     < \l__regex_step_int
8273     \tl_set:Nn \l__regex_curr_submatches_tl { #2 , }
```



```
8274 \exp_after:wN \__regex_use_state:
8275 \fi:
8276 \scan_stop:
8277 }
8278 \cs_new_protected:Npn \__regex_action_start_wildcard:N #1
8279 {
8280 \bool_set_true:N \l__regex_fresh_thread_bool
8281 \__regex_action_free:n {1}
8282 \bool_set_false:N \l__regex_fresh_thread_bool
8283 \bool_if:NT #1 { \__regex_action_cost:n {0} }
8284 }
8285 \cs_new_protected:Npn \__regex_action_free:n
8286 { \__regex_action_free_aux:nn { > \l__regex_step_int \else: } }
8287 \cs_new_protected:Npn \__regex_action_free_group:n
8288 { \__regex_action_free_aux:nn { < \l__regex_step_int } }
8289 \cs_new_protected:Npn \__regex_action_free_aux:nn #1#2
8290 {
8291 \use:e
8292 {
8293 \int_add:Nn \l__regex_curr_state_int {#2}
8294 \exp_not:n
8295 {
8296 \if_int_compare:w
8297 \__kernel_intarray_item:Nn \g__regex_state_active_intarray
8298 \l__regex_curr_state_int
8299 #1
8300 \exp_after:wN \__regex_use_state:
8301 \fi:
8302 }
8303 \int_set:Nn \l__regex_curr_state_int
8304 { \int_use:N \l__regex_curr_state_int }
8305 \tl_set:Nn \exp_not:N \l__regex_curr_submatches_tl
8306 { \exp_not:o \l__regex_curr_submatches_tl }
8307 }
8308 }
8309 \cs_new_protected:Npn \__regex_action_cost:n #1
8310 {
8311 \exp_args:No \__regex_store_state:n
8312 { \tex_the:D \__regex_int_eval:w \l__regex_curr_state_int + #1 }
8313 }
8314 \cs_new_protected:Npn \__regex_store_state:n #1
8315 {
8316 \exp_args:No \__regex_store_submatches:nn
8317 \l__regex_curr_submatches_tl {#1}
8318 \int_incr:N \l__regex_max_thread_int
8319 }
```

```
8320 \cs_new_protected:Npn \__regex_store_submatches:nn #1#2
8321 {
8322   \__kernel_intarray_gset_range_from_clist:Nnn
8323   \g__regex_thread_info_intarray
8324   {
8325     \__regex_int_eval:w
8326     \c_one_int + \l__regex_max_thread_int *
8327     (\l__regex_capturing_group_int * 2 + \c_one_int)
8328   }
8329   { #2 , #1 }
8330 }
8331 \cs_new_protected:Npn \__regex_disable_submatches:
8332 {
8333   \cs_set_protected:Npn \__regex_store_submatches:n ##1 { }
8334   \cs_set_protected:Npn \__regex_action_submatch:nN ##1##2 { }
8335 }
8336 \cs_new_protected:Npn \__regex_action_submatch:nN #1#2
8337 {
8338   \exp_after:wN \__regex_action_submatch_aux:w
8339   \l__regex_curr_submatches_tl \__regex_sep: {#1} #2
8340 }
8341 \cs_new_protected:Npn \__regex_action_submatch_aux:w #1 \__regex_sep: #2#3
8342 {
8343   \tl_set:Nc \l__regex_curr_submatches_tl
8344   {
8345     \prg_replicate:nn
8346     { #2 \if_meaning:w > #3 + \l__regex_capturing_group_int \fi: }
8347     { \__regex_action_submatch_auxii:w }
8348     \__regex_action_submatch_auxiii:w
8349     #1
8350   }
8351 }
8352 \cs_new:Npn \__regex_action_submatch_auxii:w
8353 #1 \__regex_action_submatch_auxiii:w #2 ,
8354 { #2 , #1 \__regex_action_submatch_auxiii:w }
8355 \cs_new:Npn \__regex_action_submatch_auxiii:w #1 ,
8356 { \int_use:N \l__regex_curr_pos_int , }
8357 \cs_new_protected:Npn \__regex_action_success:
8358 {
8359   \__regex_if_two_empty_matches:F
8360   {
8361     \bool_set_true:N \l__regex_match_success_bool
8362     \bool_set_eq:NN \l__regex_empty_success_bool
8363     \l__regex_fresh_thread_bool
8364     \int_set_eq:NN \l__regex_success_pos_int \l__regex_curr_pos_int
8365     \int_set_eq:NN \l__regex_last_char_success_int \l__regex_last_char_int
```

```
8366         \tl_build_begin:N \l__regex_matched_analysis_tl 28366
8367         \tl_set_eq:NN \l__regex_success_submatches_tl 28367
8368         \l__regex_curr_submatches_tl 28368
8369     \prg_break: 28369
8370 } 28370
8371 } 28371
8372 \int_new:N \l__regex_replacement_csnames_int 28372
8373 \tl_new:N \l__regex_replacement_category_tl 28373
8374 \seq_new:N \l__regex_replacement_category_seq 28374
8375 \tl_new:N \g__regex_balance_tl 28375
8376 \cs_new:Npn \__regex_replacement_balance_one_match:n #1 28376
8377 { - \__regex_submatch_balance:n {#1} } 28377
8378 \cs_new:Npn \__regex_replacement_do_one_match:n #1 28378
8379 { 28379
8380     \__regex_query_range:nn 28380
8381     { \__kernel_intarray_item:Nn \g__regex_submatch_prev_intarray {#1} } 28381
8382     { \__kernel_intarray_item:Nn \g__regex_submatch_begin_intarray {#1} } 28382
8383 } 28383
8384 \cs_new:Npn \__regex_replacement_exp_not:N #1 { \exp_not:n {#1} } 28384
8385 \cs_new_eq:NN \__regex_replacement_exp_not:V \exp_not:V 28385
8386 \cs_new:Npn \__regex_query_range:nn #1#2 28386
8387 { 28387
8388     \exp_after:wN \__regex_query_range_loop:ww 28388
8389     \int_value:w \__regex_int_eval:w #1 \exp_after:wN \__regex_sep: 28389
8390     \int_value:w \__regex_int_eval:w #2 \__regex_sep: 28390
8391     \prg_break_point: 28391
8392 } 28392
8393 \cs_new:Npn \__regex_query_range_loop:ww #1 \__regex_sep: #2 \__regex_sep: 28393
8394 { 28394
8395     \if_int_compare:w #1 < #2 \exp_stop_f: 28395
8396     \else: 28396
8397         \prg_break:n 28397
8398     \fi: 28398
8399     \__regex_toks_use:w #1 \exp_stop_f: 28399
8400     \exp_after:wN \__regex_query_range_loop:ww 28400
8401     \int_value:w \__regex_int_eval:w #1 + \c_one_int \__regex_sep: #2 \__regex_sep: 28401
8402 } 28402
8403 \cs_new:Npn \__regex_query_submatch:n #1 28403
8404 { 28404
8405     \__regex_query_range:nn 28405
8406     { \__kernel_intarray_item:Nn \g__regex_submatch_begin_intarray {#1} } 28406
8407     { \__kernel_intarray_item:Nn \g__regex_submatch_end_intarray {#1} } 28407
8408 } 28408
8409 \cs_new_protected:Npn \__regex_submatch_balance:n #1 28409
8410 { 28410
8411     \tex_the:D \__regex_int_eval:w 28411
```

```
8412 \__regex_intarray_item:NnF \g__regex_balance_intarray 28412
8413 { 28413
8414 \__kernel_intarray_item:Nn 28414
8415 \g__regex_submatch_end_intarray {#1} 28415
8416 } 28416
8417 \c_zero_int 28417
8418 - 28418
8419 \__regex_intarray_item:NnF \g__regex_balance_intarray 28419
8420 { 28420
8421 \__kernel_intarray_item:Nn 28421
8422 \g__regex_submatch_begin_intarray {#1} 28422
8423 } 28423
8424 \c_zero_int 28424
8425 \scan_stop: 28425
8426 } 28426
8427 \cs_new_protected:Npn \__regex_replacement:n 28427
8428 { \__regex_replacement_apply:Nn \__regex_replacement_set:n } 28428
8429 \cs_new_protected:Npn \__regex_replacement_apply:Nn #1#2 28429
8430 { 28430
8431 \group_begin: 28431
8432 \tl_build_begin:N \l__regex_build_tl 28432
8433 \int_zero:N \l__regex_balance_int 28433
8434 \tl_gclear:N \g__regex_balance_tl 28434
8435 \__regex_escape_use:nnnn 28435
8436 { 28436
8437 \if_charcode:w \c_right_brace_str ##1 28437
8438 \__regex_replacement_rbrace:N 28438
8439 \else: 28439
8440 \if_charcode:w \c_left_brace_str ##1 28440
8441 \__regex_replacement_lbrace:N 28441
8442 \else: 28442
8443 \__regex_replacement_normal:n 28443
8444 \fi: 28444
8445 \fi: 28445
8446 ##1 28446
8447 } 28447
8448 { \__regex_replacement_escaped:N ##1 } 28448
8449 { \__regex_replacement_normal:n ##1 } 28449
8450 {#2} 28450
8451 \prg_do_nothing: \prg_do_nothing: 28451
8452 \if_int_compare:w \l__regex_replacement_csnames_int > \c_zero_int 28452
8453 \msg_error:nne { regex } { replacement-missing-rbrace } 28453
8454 { \int_use:N \l__regex_replacement_csnames_int } 28454
8455 \tl_build_put_right:Ne \l__regex_build_tl 28455
8456 { \prg_replicate:nn \l__regex_replacement_csnames_int \cs_end: } 28456
8457 \fi: 28457
```

```
8458 \seq_if_empty:NF \l__regex_replacement_category_seq
8459 {
8460     \msg_error:nne { regex } { replacement-missing-rparen }
8461     { \seq_count:N \l__regex_replacement_category_seq }
8462     \seq_clear:N \l__regex_replacement_category_seq
8463 }
8464 \tl_gput_right:Ne \g__regex_balance_tl
8465 { + \int_use:N \l__regex_balance_int }
8466 \tl_build_end:N \l__regex_build_tl
8467 \exp_args:NNo
8468 \group_end:
8469 #1 \l__regex_build_tl
8470 }
8471 \cs_generate_variant:Nn \__regex_replacement:n { e }
8472 \cs_new_protected:Npn \__regex_replacement_set:n #1
8473 {
8474     \cs_set:Npn \__regex_replacement_do_one_match:n ##1
8475     {
8476         \__regex_query_range:nn
8477         {
8478             \__kernel_intarray_item:Nn
8479             \g__regex_submatch_prev_intarray {##1}
8480         }
8481         {
8482             \__kernel_intarray_item:Nn
8483             \g__regex_submatch_begin_intarray {##1}
8484         }
8485         #1
8486     }
8487     \exp_args:Nno \use:n
8488     { \cs_gset:Npn \__regex_replacement_balance_one_match:n ##1 }
8489     {
8490         \g__regex_balance_tl
8491         - \__regex_submatch_balance:n {##1}
8492     }
8493 }
8494 \tl_new:N \g__regex_case_replacement_tl
8495 \tl_new:N \g__regex_case_balance_tl
8496 \cs_new_protected:Npn \__regex_case_replacement:n #1
8497 {
8498     \tl_gset:Nn \g__regex_case_balance_tl
8499     {
8500         \if_case:w
8501             \__kernel_intarray_item:Nn
8502             \g__regex_submatch_case_intarray {##1}
8503         }
```

```
8504 \tl_gset_eq:NN \g__regex_case_replacement_tl \g__regex_case_balance_tl 28504
8505 \tl_map_tokens:nn {#1} 28505
8506 { \__regex_replacement_apply:Nn \__regex_case_replacement_aux:n } 28506
8507 \tl_gset:No \g__regex_balance_tl 28507
8508 { \g__regex_case_balance_tl \fi: } 28508
8509 \exp_args:No \__regex_replacement_set:n 28509
8510 { \g__regex_case_replacement_tl \fi: } 28510
8511 } 28511
8512 \cs_generate_variant:Nn \__regex_case_replacement:n { e } 28512
8513 \cs_new_protected:Npn \__regex_case_replacement_aux:n #1 28513
8514 { 28514
8515 \tl_gput_right:Nn \g__regex_case_replacement_tl { \or: #1 } 28515
8516 \tl_gput_right:No \g__regex_case_balance_tl 28516
8517 { \exp_after:wN \or: \g__regex_balance_tl } 28517
8518 } 28518
8519 \cs_new_protected:Npn \__regex_replacement_put:n 28519
8520 { \tl_build_put_right:Nn \l__regex_build_tl } 28520
8521 \cs_new_protected:Npn \__regex_replacement_normal:n #1 28521
8522 { 28522
8523 \int_compare:nNnTF \l__regex_replacement_csnames_int > \c_zero_int 28523
8524 { \exp_args:No \__regex_replacement_put:n { \token_to_str:N #1 } } 28524
8525 { 28525
8526 \tl_if_empty:NTF \l__regex_replacement_category_tl 28526
8527 { \__regex_replacement_normal_aux:N #1 } 28527
8528 { % ( 28528
8529 \token_if_eq_charcode:NNTF #1 ) 28529
8530 { 28530
8531 \seq_pop:NN \l__regex_replacement_category_seq 28531
8532 \l__regex_replacement_category_tl 28532
8533 } 28533
8534 { 28534
8535 \use:c { __regex_replacement_c_ \l__regex_replacement_category_tl :w } 28535
8536 ? #1 28536
8537 } 28537
8538 } 28538
8539 } 28539
8540 } 28540
8541 \cs_new_protected:Npn \__regex_replacement_normal_aux:N #1 28541
8542 { 28542
8543 \token_if_eq_charcode:NNTF #1 \c_space_token 28543
8544 { \__regex_replacement_c_S:w } 28544
8545 { 28545
8546 \exp_after:wN \exp_after:wN 28546
8547 \if_case:w \tex_catcode:D `#1 \exp_stop_f: 28547
8548 \__regex_replacement_c_O:w 28548
8549 \or: \__regex_replacement_c_B:w 28549
```


8550	\or: __regex_replacement_c_E:w	28550
8551	\or: __regex_replacement_c_M:w	28551
8552	\or: __regex_replacement_c_T:w	28552
8553	\or: __regex_replacement_c_O:w	28553
8554	\or: __regex_replacement_c_P:w	28554
8555	\or: __regex_replacement_c_U:w	28555
8556	\or: __regex_replacement_c_D:w	28556
8557	\or: __regex_replacement_c_0:w	28557
8558	\or: __regex_replacement_c_S:w	28558
8559	\or: __regex_replacement_c_L:w	28559
8560	\or: __regex_replacement_c_0:w	28560
8561	\or: __regex_replacement_c_A:w	28561
8562	\else: __regex_replacement_c_0:w	28562
8563	\fi:	28563
8564	}	28564
8565	? #1	28565
8566	}	28566
8567	\cs_new_protected:Npn __regex_replacement_escaped:N #1	28567
8568	{	28568
8569	\cs_if_exist_use:cF { __regex_replacement_#1:w }	28569
8570	{	28570
8571	\if_int_compare:w \c_one_int < 1#1 \exp_stop_f:	28571
8572	__regex_replacement_put_submatch:n {#1}	28572
8573	\else:	28573
8574	__regex_replacement_normal:n {#1}	28574
8575	\fi:	28575
8576	}	28576
8577	}	28577
8578	\cs_new_protected:Npn __regex_replacement_put_submatch:n #1	28578
8579	{	28579
8580	\if_int_compare:w #1 < \l__regex_capturing_group_int	28580
8581	__regex_replacement_put_submatch_aux:n {#1}	28581
8582	\else:	28582
8583	\msg_expandable_error:nnff { regex } { submatch-too-big }	28583
8584	{#1} { \int_eval:n { \l__regex_capturing_group_int - \c_one_int } }	28584
8585	\fi:	28585
8586	}	28586
8587	\cs_new_protected:Npn __regex_replacement_put_submatch_aux:n #1	28587
8588	{	28588
8589	\tl_build_put_right:Nn \l__regex_build_tl	28589
8590	{ __regex_query_submatch:n { __regex_int_eval:w #1 + ##1 \scan_stop: } }	28590
8591	\if_int_compare:w \l__regex_replacement_csnames_int = \c_zero_int	28591
8592	\tl_gput_right:Nn \g__regex_balance_tl	28592
8593	{ + __regex_submatch_balance:n { __regex_int_eval:w #1 + ##1 \scan_stop: } }	28593
8594	\fi:	28594
8595	}	28595


```
8596 \cs_new_protected:Npn \__regex_replacement_g:w #1#2
8597 {
8598   \token_if_eq_meaning:NNTF #1 \__regex_replacement_lbrace:N
8599   { \l__regex_internal_a_int = \__regex_replacement_g_digits:NN }
8600   { \__regex_replacement_error:NNN g #1 #2 }
8601 }
8602 \cs_new:Npn \__regex_replacement_g_digits:NN #1#2
8603 {
8604   \token_if_eq_meaning:NNTF #1 \__regex_replacement_normal:n
8605   {
8606     \if_int_compare:w \c_one_int < 1#2 \exp_stop_f:
8607     #2
8608     \exp_after:wN \use_i:nnn
8609     \exp_after:wN \__regex_replacement_g_digits:NN
8610   \else:
8611     \exp_stop_f:
8612     \exp_after:wN \__regex_replacement_error:NNN
8613     \exp_after:wN g
8614   \fi:
8615 }
8616 {
8617   \exp_stop_f:
8618   \if_meaning:w \__regex_replacement_rbrace:N #1
8619   \exp_args:No \__regex_replacement_put_submatch:n
8620   { \int_use:N \l__regex_internal_a_int }
8621   \exp_after:wN \use_none:nn
8622   \else:
8623     \exp_after:wN \__regex_replacement_error:NNN
8624     \exp_after:wN g
8625   \fi:
8626 }
8627 #1 #2
8628 }
8629 \cs_new_protected:Npn \__regex_replacement_c:w #1#2
8630 {
8631   \token_if_eq_meaning:NNTF #1 \__regex_replacement_normal:n
8632   {
8633     \cs_if_exist:CTF { __regex_replacement_c_#2:w }
8634     { \__regex_replacement_cat:NNN #2 }
8635     { \__regex_replacement_error:NNN c #1#2 }
8636   }
8637   {
8638     \token_if_eq_meaning:NNTF #1 \__regex_replacement_lbrace:N
8639     { \__regex_replacement_cu_aux:Nw \__regex_replacement_exp_not:N }
8640     { \__regex_replacement_error:NNN c #1#2 }
8641   }
```

8642	}	28642
8643	\cs_new_protected:Npn __regex_replacement_cu_aux:Nw #1	28643
8644	{	28644
8645	\if_case:w \l__regex_replacement_csnames_int	28645
8646	\tl_build_put_right:Nn \l__regex_build_tl	28646
8647	{ \exp_not:n { \exp_after:wN #1 \cs:w } }	28647
8648	\else:	28648
8649	\tl_build_put_right:Nn \l__regex_build_tl	28649
8650	{ \exp_not:n { \exp_after:wN \tl_to_str:V \cs:w } }	28650
8651	\fi:	28651
8652	\int_incr:N \l__regex_replacement_csnames_int	28652
8653	}	28653
8654	\cs_new_protected:Npn __regex_replacement_u:w #1#2	28654
8655	{	28655
8656	\token_if_eq_meaning:NNTF #1 __regex_replacement_lbrace:N	28656
8657	{ __regex_replacement_cu_aux:Nw __regex_replacement_exp_not:V }	28657
8658	{ __regex_replacement_error:NNN u #1#2 }	28658
8659	}	28659
8660	\cs_new_protected:Npn __regex_replacement_rbrace:N #1	28660
8661	{	28661
8662	\if_int_compare:w \l__regex_replacement_csnames_int > \c_zero_int	28662
8663	\tl_build_put_right:Nn \l__regex_build_tl { \cs_end: }	28663
8664	\int_decr:N \l__regex_replacement_csnames_int	28664
8665	\else:	28665
8666	__regex_replacement_normal:n {#1}	28666
8667	\fi:	28667
8668	}	28668
8669	\cs_new_protected:Npn __regex_replacement_lbrace:N #1	28669
8670	{	28670
8671	\if_int_compare:w \l__regex_replacement_csnames_int > \c_zero_int	28671
8672	\msg_error:nnn { regex } { cu-lbrace } { u }	28672
8673	\else:	28673
8674	__regex_replacement_normal:n {#1}	28674
8675	\fi:	28675
8676	}	28676
8677	\cs_new_protected:Npn __regex_replacement_cat:NNN #1#2#3	28677
8678	{	28678
8679	\token_if_eq_meaning:NNTF \prg_do_nothing: #3	28679
8680	{ \msg_error:nn { regex } { replacement-catcode-end } }	28680
8681	{	28681
8682	\int_compare:nNnTF \l__regex_replacement_csnames_int > \c_zero_int	28682
8683	{	28683
8684	\msg_error:nnnn	28684
8685	{ regex } { replacement-catcode-in-cs } {#1} {#3}	28685
8686	#2 #3	28686
8687	}	28687

```

8688 {
8689 \_regex_two_if_eq:NNNTF #2 #3 \_regex_replacement_normal:n (
8690 {
8691 \seq_push:NV \l__regex_replacement_category_seq
8692 \l__regex_replacement_category_tl
8693 \tl_set:Nn \l__regex_replacement_category_tl {#1}
8694 }
8695 {
8696 \token_if_eq_meaning:NNT #2 \_regex_replacement_escaped:N
8697 {
8698 \_regex_char_if_alphanumeric:NTF #3
8699 {
8700 \msg_error:nnnn
8701 { regex } { replacement-catcode-escaped }
8702 {#1} {#3}
8703 }
8704 { }
8705 }
8706 \use:c { __regex_replacement_c_#1:w } #2 #3
8707 }
8708 }
8709 }
8710 }
8711 \group_begin:
8712 \cs_new_protected:Npn \_regex_replacement_char:nNN #1#2#3
8713 {
8714 \tex_lccode:D \c_zero_int = `#3 \scan_stop:
8715 \tex_lowercase:D { \_regex_replacement_put:n {#1} }
8716 }
8717 \char_set_catcode_active:N ^^@
8718 \cs_new_protected:Npn \_regex_replacement_c_A:w
8719 { \_regex_replacement_char:nNN { \exp_not:n { \exp_not:N ^^@ } } }
8720 \char_set_catcode_group_begin:N ^^@
8721 \cs_new_protected:Npn \_regex_replacement_c_B:w
8722 {
8723 \if_int_compare:w \l__regex_replacement_csnames_int = \c_zero_int
8724 \int_incr:N \l__regex_balance_int
8725 \fi:
8726 \_regex_replacement_char:nNN
8727 { \exp_not:n { \exp_after:wN ^^@ \if_false: } \fi: } }
8728 }
8729 \cs_new_protected:Npn \_regex_replacement_c_C:w #1#2
8730 {
8731 \tl_build_put_right:Nn \l__regex_build_tl
8732 { \exp_not:N \_regex_replacement_exp_not:N \exp_not:c {#2} }
8733 }

```

```

8734 \char_set_catcode_math_subscript:N \^^@
8735 \cs_new_protected:Npn \__regex_replacement_c_D:w
8736 { \__regex_replacement_char:nNN { ^^@ } }
8737 \char_set_catcode_group_end:N \^^@
8738 \cs_new_protected:Npn \__regex_replacement_c_E:w
8739 {
8740 \if_int_compare:w \l__regex_replacement_csnames_int = \c_zero_int
8741 \int_decr:N \l__regex_balance_int
8742 \fi:
8743 \__regex_replacement_char:nNN
8744 { \exp_not:n { \if_false: { \fi: ^^@ } } }
8745 }
8746 \char_set_catcode_letter:N \^^@
8747 \cs_new_protected:Npn \__regex_replacement_c_L:w
8748 { \__regex_replacement_char:nNN { ^^@ } }
8749 \char_set_catcode_math_toggle:N \^^@
8750 \cs_new_protected:Npn \__regex_replacement_c_M:w
8751 { \__regex_replacement_char:nNN { ^^@ } }
8752 \char_set_catcode_other:N \^^@
8753 \cs_new_protected:Npn \__regex_replacement_c_O:w
8754 { \__regex_replacement_char:nNN { ^^@ } }
8755 \char_set_catcode_parameter:N \^^@
8756 \cs_new_protected:Npn \__regex_replacement_c_P:w
8757 {
8758 \__regex_replacement_char:nNN
8759 { \exp_not:n { \exp_not:n { ^^@^^@^^@^^@ } } } }
8760 }
8761 \cs_new_protected:Npn \__regex_replacement_c_S:w #1#2
8762 {
8763 \if_int_compare:w `#2 = \c_zero_int
8764 \msg_error:nn { regex } { replacement-null-space }
8765 \fi:
8766 \tex_lccode:D `\_ = `#2 \scan_stop:
8767 \tex_lowercase:D { \__regex_replacement_put:n {~} }
8768 }
8769 \char_set_catcode_alignment:N \^^@
8770 \cs_new_protected:Npn \__regex_replacement_c_T:w
8771 { \__regex_replacement_char:nNN { ^^@ } }
8772 \char_set_catcode_math_superscript:N \^^@
8773 \cs_new_protected:Npn \__regex_replacement_c_U:w
8774 { \__regex_replacement_char:nNN { ^^@ } }
8775 \group_end:
8776 \cs_new_protected:Npn \__regex_replacement_error:NNN #1#2#3
8777 {
8778 \msg_error:nne { regex } { replacement-#1 } {#3}
8779 #2 #3

```

```
8780 }
8781 \cs_new_protected:Npn \regex_new:N #1
8782 { \cs_new_eq:NN #1 \c__regex_no_match_regex }
8783 \regex_new:N \l_tmpa_regex
8784 \regex_new:N \l_tmpb_regex
8785 \regex_new:N \g_tmpa_regex
8786 \regex_new:N \g_tmpb_regex
8787 \cs_new_protected:Npn \regex_set:Nn #1#2
8788 {
8789   \__regex_compile:n {#2}
8790   \tl_set_eq:NN #1 \l__regex_internal_regex
8791 }
8792 \cs_new_protected:Npn \regex_gset:Nn #1#2
8793 {
8794   \__regex_compile:n {#2}
8795   \tl_gset_eq:NN #1 \l__regex_internal_regex
8796 }
8797 \cs_new_protected:Npn \regex_const:Nn #1#2
8798 {
8799   \__regex_compile:n {#2}
8800   \tl_const:Ne #1 { \exp_not:o \l__regex_internal_regex }
8801 }
8802 \cs_new_protected:Npn \regex_show:n { \__regex_show:Nn \msg_show:nneeee }
8803 \cs_new_protected:Npn \regex_log:n { \__regex_show:Nn \msg_log:nneeee }
8804 \cs_new_protected:Npn \__regex_show:Nn #1#2
8805 {
8806   \__regex_compile:n {#2}
8807   \__regex_show:N \l__regex_internal_regex
8808   #1 { regex } { show }
8809   { \tl_to_str:n {#2} } { }
8810   { \l__regex_internal_a_tl } { }
8811 }
8812 \cs_new_protected:Npn \regex_show:N { \__regex_show:NN \msg_show:nneeee }
8813 \cs_new_protected:Npn \regex_log:N { \__regex_show:NN \msg_log:nneeee }
8814 \cs_new_protected:Npn \__regex_show:NN #1#2
8815 {
8816   \__kernel_chk_tl_type:NnnT #2 { regex }
8817   { \exp_args:No \__regex_clean_regex:n {#2} }
8818   {
8819     \__regex_show:N #2
8820     #1 { regex } { show }
8821     { } { \token_to_str:N #2 }
8822     { \l__regex_internal_a_tl } { }
8823   }
8824 }
8825 \prg_new_protected_conditional:Npnn \regex_match:nn #1#2 { T , F , TF }
```

```
8826 {
8827     \__regex_if_match:nn { \__regex_build:n {#1} } {#2}
8828     \__regex_return:
8829 }
8830 \prg_generate_conditional_variant:Nnn \regex_match:nn { nV } { T , F , TF }
8831 \prg_new_protected_conditional:Npnn \regex_match:Nn #1#2 { T , F , TF }
8832 {
8833     \__regex_if_match:nn { \__regex_build:N #1 } {#2}
8834     \__regex_return:
8835 }
8836 \prg_generate_conditional_variant:Nnn \regex_match:Nn { NV } { T , F , TF }
8837 \cs_new_protected:Npn \regex_count:nnN #1
8838 { \__regex_count:nnN { \__regex_build:n {#1} } }
8839 \cs_new_protected:Npn \regex_count:NnN #1
8840 { \__regex_count:nnN { \__regex_build:N #1 } }
8841 \cs_generate_variant:Nn \regex_count:nnN { nV }
8842 \cs_generate_variant:Nn \regex_count:NnN { NV }
8843 \cs_new_protected:Npn \regex_match_case:nnTF #1#2#3
8844 {
8845     \__regex_match_case:nnTF {#1} {#2}
8846     {
8847         \tl_item:nn {#1} { 2 * \g__regex_case_int }
8848         #3
8849     }
8850 }
8851 \cs_new_protected:Npn \regex_match_case:nn #1#2
8852 { \regex_match_case:nnTF {#1} {#2} { } { } }
8853 \cs_new_protected:Npn \regex_match_case:nnT #1#2#3
8854 { \regex_match_case:nnTF {#1} {#2} {#3} { } }
8855 \cs_new_protected:Npn \regex_match_case:nnF #1#2
8856 { \regex_match_case:nnTF {#1} {#2} { } }
8857 \cs_set_protected:Npn \__regex_tmp:w #1#2#3
8858 {
8859     \cs_new_protected:Npn #2 ##1 { #1 { \__regex_build:n {##1} } }
8860     \cs_new_protected:Npn #3 ##1 { #1 { \__regex_build:N ##1 } }
8861     \prg_new_protected_conditional:Npnn #2 ##1##2##3 { T , F , TF }
8862     { #1 { \__regex_build:n {##1} } {##2} ##3 \__regex_return: }
8863     \prg_new_protected_conditional:Npnn #3 ##1##2##3 { T , F , TF }
8864     { #1 { \__regex_build:N ##1 } {##2} ##3 \__regex_return: }
8865     \cs_generate_variant:Nn #2 { nV }
8866     \prg_generate_conditional_variant:Nnn #2 { nV } { T , F , TF }
8867     \cs_generate_variant:Nn #3 { NV }
8868     \prg_generate_conditional_variant:Nnn #3 { NV } { T , F , TF }
8869 }
8870 \__regex_tmp:w \__regex_extract_once:nnN
8871 \regex_extract_once:nnN \regex_extract_once:NnN
```



```
8872 \__regex_tmp:w \__regex_extract_all:nnN
8873 \regex_extract_all:nnN \regex_extract_all:NnN
8874 \__regex_tmp:w \__regex_replace_once:nnN
8875 \regex_replace_once:nnN \regex_replace_once:NnN
8876 \__regex_tmp:w \__regex_replace_all:nnN
8877 \regex_replace_all:nnN \regex_replace_all:NnN
8878 \__regex_tmp:w \__regex_split:nnN \regex_split:nnN \regex_split:NnN
8879 \cs_new_protected:Npn \regex_replace_case_once:nNTF #1#2
8880 {
8881 \int_if_odd:nTF { \tl_count:n {#1} }
8882 {
8883 \msg_error:nneeee { regex } { case-odd }
8884 { \token_to_str:N \regex_replace_case_once:nN(TF) } { code }
8885 { \tl_count:n {#1} } { \tl_to_str:n {#1} }
8886 \use_ii:nn
8887 }
8888 {
8889 \__regex_replace_once_aux:nnN
8890 { \__regex_case_build:e { \__regex_tl_odd_items:n {#1} } }
8891 { \__regex_replacement:e { \tl_item:nn {#1} { 2 * \g__regex_case_int } } }
8892 #2
8893 \bool_if:NTF \g__regex_success_bool
8894 }
8895 }
8896 \cs_new_protected:Npn \regex_replace_case_once:nN #1#2
8897 { \regex_replace_case_once:nNTF {#1} {#2} { } { } }
8898 \cs_new_protected:Npn \regex_replace_case_once:nNT #1#2#3
8899 { \regex_replace_case_once:nNTF {#1} {#2} {#3} { } }
8900 \cs_new_protected:Npn \regex_replace_case_once:nNF #1#2
8901 { \regex_replace_case_once:nNTF {#1} {#2} { } }
8902 \cs_new_protected:Npn \regex_replace_case_all:nNTF #1#2
8903 {
8904 \int_if_odd:nTF { \tl_count:n {#1} }
8905 {
8906 \msg_error:nneeee { regex } { case-odd }
8907 { \token_to_str:N \regex_replace_case_all:nN(TF) } { code }
8908 { \tl_count:n {#1} } { \tl_to_str:n {#1} }
8909 \use_ii:nn
8910 }
8911 {
8912 \__regex_replace_all_aux:nnN
8913 { \__regex_case_build:e { \__regex_tl_odd_items:n {#1} } }
8914 { \__regex_case_replacement:e { \__regex_tl_even_items:n {#1} } }
8915 #2
8916 \bool_if:NTF \g__regex_success_bool
8917 }
```



```
8918 }
8919 \cs_new_protected:Npn \regex_replace_case_all:nN #1#2
8920 { \regex_replace_case_all:nNTF {#1} {#2} { } { } }
8921 \cs_new_protected:Npn \regex_replace_case_all:nNT #1#2#3
8922 { \regex_replace_case_all:nNTF {#1} {#2} {#3} { } }
8923 \cs_new_protected:Npn \regex_replace_case_all:nNF #1#2
8924 { \regex_replace_case_all:nNTF {#1} {#2} { } { } }
8925 \int_new:N \l__regex_match_count_int
8926 \flag_new:N \l__regex_begin_flag
8927 \flag_new:N \l__regex_end_flag
8928 \int_new:N \l__regex_min_submatch_int
8929 \int_new:N \l__regex_submatch_int
8930 \int_new:N \l__regex_zeroth_submatch_int
8931 \intarray_new:Nn \g__regex_submatch_prev_intarray { 65536 }
8932 \intarray_new:Nn \g__regex_submatch_begin_intarray { 65536 }
8933 \intarray_new:Nn \g__regex_submatch_end_intarray { 65536 }
8934 \intarray_new:Nn \g__regex_submatch_case_intarray { 65536 }
8935 \intarray_new:Nn \g__regex_balance_intarray { 65536 }
8936 \int_new:N \l__regex_added_begin_int
8937 \int_new:N \l__regex_added_end_int
8938 \cs_new_protected:Npn \__regex_return:
8939 {
8940   \if_meaning:w \c_true_bool \g__regex_success_bool
8941     \prg_return_true:
8942   \else:
8943     \prg_return_false:
8944   \fi:
8945 }
8946 \cs_new_protected:Npn \__regex_query_set:n #1
8947 {
8948   \int_zero:N \l__regex_balance_int
8949   \int_zero:N \l__regex_curr_pos_int
8950   \__regex_query_set_aux:nN { } F
8951   \tl_analysis_map_inline:nn {#1}
8952     { \__regex_query_set_aux:nN {##1} ##3 }
8953   \__regex_query_set_aux:nN { } F
8954   \int_set_eq:NN \l__regex_max_pos_int \l__regex_curr_pos_int
8955 }
8956 \cs_new_protected:Npn \__regex_query_set_aux:nN #1#2
8957 {
8958   \int_incr:N \l__regex_curr_pos_int
8959   \__regex_toks_set:Nn \l__regex_curr_pos_int {#1}
8960   \__kernel_intarray_gset:Nnn \g__regex_balance_intarray
8961     \l__regex_curr_pos_int \l__regex_balance_int
8962   \if_case:w "#2 \exp_stop_f:
8963   \or: \int_incr:N \l__regex_balance_int
```

```
8964 \or: \int_decr:N \l__regex_balance_int 2896
8965 \fi: 2896
8966 } 2896
8967 \cs_new_protected:Npn \__regex_if_match:nn #1#2 2896
8968 { 2896
8969 \group_begin: 2896
8970 \__regex_disable_submatches: 2897
8971 \__regex_single_match: 2897
8972 #1 2897
8973 \__regex_match:n {#2} 2897
8974 \group_end: 2897
8975 } 2897
8976 \cs_new_protected:Npn \__regex_match_case:nnTF #1#2 2897
8977 { 2897
8978 \int_if_odd:nTF { \tl_count:n {#1} } 2897
8979 { 2897
8980 \msg_error:nneeee { regex } { case-odd } 2898
8981 { \token_to_str:N \regex_match_case:nn(TF) } { code } 2898
8982 { \tl_count:n {#1} } { \tl_to_str:n {#1} } 2898
8983 \use_ii:nn 2898
8984 } 2898
8985 { 2898
8986 \__regex_if_match:nn 2898
8987 { \__regex_case_build:e { \__regex_tl_odd_items:n {#1} } } 2898
8988 {#2} 2898
8989 \bool_if:NTF \g__regex_success_bool 2898
8990 } 2899
8991 } 2899
8992 \cs_new:Npn \__regex_match_case_aux:nn #1#2 { \exp_not:n { {#1} } } 2899
8993 \cs_new_protected:Npn \__regex_count:nnN #1#2#3 2899
8994 { 2899
8995 \group_begin: 2899
8996 \__regex_disable_submatches: 2899
8997 \int_zero:N \l__regex_match_count_int 2899
8998 \__regex_multi_match:n { \int_incr:N \l__regex_match_count_int } 2899
8999 #1 2899
9000 \__regex_match:n {#2} 2900
9001 \exp_args:NNNo 2900
9002 \group_end: 2900
9003 \int_set:Nn #3 { \int_use:N \l__regex_match_count_int } 2900
9004 } 2900
9005 \cs_new_protected:Npn \__regex_extract_once:nnN #1#2#3 2900
9006 { 2900
9007 \group_begin: 2900
9008 \__regex_single_match: 2900
9009 #1 2900
```

```
9010     \__regex_match:n {#2}
9011     \__regex_extract:
9012     \__regex_query_set:n {#2}
9013     \__regex_group_end_extract_seq:N #3
9014 }
9015 \cs_new_protected:Npn \__regex_extract_all:nnN #1#2#3
9016 {
9017     \group_begin:
9018         \__regex_multi_match:n { \__regex_extract: }
9019         #1
9020         \__regex_match:n {#2}
9021         \__regex_query_set:n {#2}
9022         \__regex_group_end_extract_seq:N #3
9023     }
9024 \cs_new_protected:Npn \__regex_split:nnN #1#2#3
9025 {
9026     \group_begin:
9027         \__regex_multi_match:n
9028         {
9029             \if_int_compare:w
9030                 \l__regex_start_pos_int < \l__regex_success_pos_int
9031                 \__regex_extract:
9032                 \__kernel_intarray_gset:Nnn \g__regex_submatch_prev_intarray
9033                 \l__regex_zeroth_submatch_int \c_zero_int
9034                 \__kernel_intarray_gset:Nnn \g__regex_submatch_end_intarray
9035                 \l__regex_zeroth_submatch_int
9036                 {
9037                     \__kernel_intarray_item:Nn \g__regex_submatch_begin_intarray
9038                     \l__regex_zeroth_submatch_int
9039                 }
9040                 \__kernel_intarray_gset:Nnn \g__regex_submatch_begin_intarray
9041                 \l__regex_zeroth_submatch_int
9042                 \l__regex_start_pos_int
9043             \fi:
9044         }
9045         #1
9046         \__regex_match:n {#2}
9047         \__regex_query_set:n {#2}
9048         \__kernel_intarray_gset:Nnn \g__regex_submatch_prev_intarray
9049         \l__regex_submatch_int \c_zero_int
9050         \__kernel_intarray_gset:Nnn \g__regex_submatch_end_intarray
9051         \l__regex_submatch_int
9052         \l__regex_max_pos_int
9053         \__kernel_intarray_gset:Nnn \g__regex_submatch_begin_intarray
9054         \l__regex_submatch_int
9055         \l__regex_start_pos_int
```

```
9056 \int_incr:N \l__regex_submatch_int 2905
9057 \if_meaning:w \c_true_bool \l__regex_empty_success_bool 2905
9058 \if_int_compare:w \l__regex_start_pos_int = \l__regex_max_pos_int 2905
9059 \int_decr:N \l__regex_submatch_int 2905
9060 \fi: 2906
9061 \fi: 2906
9062 \__regex_group_end_extract_seq:N #3 2906
9063 } 2906
9064 \cs_new_protected:Npn \__regex_group_end_extract_seq:N #1 2906
9065 { 2906
9066 \flag_clear:N \l__regex_begin_flag 2906
9067 \flag_clear:N \l__regex_end_flag 2906
9068 \cs_set_eq:NN \__regex_tmp:w \scan_stop: 2906
9069 \__kernel_tl_gset:Nx \g__regex_internal_tl 2906
9070 { 2907
9071 \int_step_function:nnN \l__regex_min_submatch_int 2907
9072 { \l__regex_submatch_int - \c_one_int } \__regex_extract_seq_aux:n 2907
9073 \__regex_tmp:w 2907
9074 } 2907
9075 \int_set:Nn \l__regex_added_begin_int 2907
9076 { \flag_height:N \l__regex_begin_flag } 2907
9077 \int_set:Nn \l__regex_added_end_int 2907
9078 { \flag_height:N \l__regex_end_flag } 2907
9079 \tex_afterassignment:D \__regex_extract_check:w 2907
9080 \__kernel_tl_gset:Nx \g__regex_internal_tl 2908
9081 { \g__regex_internal_tl \if_false: { \fi: } } 2908
9082 \int_compare:nNnT 2908
9083 { \l__regex_added_begin_int + \l__regex_added_end_int } > \c_zero_int 2908
9084 { 2908
9085 \msg_error:nneee { regex } { result-unbalanced } 2908
9086 { splitting~or~extracting~submatches } 2908
9087 { \int_use:N \l__regex_added_begin_int } 2908
9088 { \int_use:N \l__regex_added_end_int } 2908
9089 } 2908
9090 \group_end: 2909
9091 \__regex_extract_seq:N #1 2909
9092 } 2909
9093 \cs_gset_protected:Npn \__regex_extract_seq:N #1 2909
9094 { 2909
9095 \seq_clear:N #1 2909
9096 \cs_set_eq:NN \__regex_tmp:w \__regex_extract_seq_loop:Nw 2909
9097 \exp_after:wN \__regex_extract_seq:NNn 2909
9098 \exp_after:wN #1 2909
9099 \g__regex_internal_tl \use_none:nnn 2909
9100 } 2910
9101 \cs_new_protected:Npn \__regex_extract_seq:NNn #1#2#3 2910
```

```
9102 { #3 #2 #1 \prg_do_nothing: } 29102
9103 \cs_new_protected:Npn \__regex_extract_seq_loop:Nw #1#2 \__regex_tmp:w #3 29103
9104 { 29104
9105     \seq_put_right:No #1 {#2} 29105
9106     #3 \__regex_extract_seq_loop:Nw #1 \prg_do_nothing: 29106
9107 } 29107
9108 \cs_new:Npn \__regex_extract_seq_aux:n #1 29108
9109 { 29109
9110     \__regex_tmp:w { } 29110
9111     \exp_after:wN \__regex_extract_seq_aux:ww 29111
9112     \int_value:w \__regex_submatch_balance:n {#1} \__regex_sep: #1 \__regex_sep: 29112
9113 } 29113
9114 \cs_new:Npn \__regex_extract_seq_aux:ww #1 \__regex_sep: #2 \__regex_sep: 29114
9115 { 29115
9116     \if_int_compare:w #1 < \c_zero_int 29116
9117     \prg_replicate:nn {-#1} 29117
9118     { 29118
9119         \flag_raise:N \l__regex_begin_flag 29119
9120         \exp_not:n { { \if_false: } \fi: } 29120
9121     } 29121
9122     \fi: 29122
9123     \__regex_query_submatch:n {#2} 29123
9124     \if_int_compare:w #1 > \c_zero_int 29124
9125     \prg_replicate:nn {#1} 29125
9126     { 29126
9127         \flag_raise:N \l__regex_end_flag 29127
9128         \exp_not:n { \if_false: { \fi: } } 29128
9129     } 29129
9130     \fi: 29130
9131 } 29131
9132 \cs_new_protected:Npn \__regex_extract_check:w 29132
9133 { 29133
9134     \exp_after:wN \__regex_extract_check:n 29134
9135     \exp_after:wN { \if_false: } \fi: 29135
9136 } 29136
9137 \cs_new_protected:Npn \__regex_extract_check:n #1 29137
9138 { 29138
9139     \tl_if_empty:nF {#1} 29139
9140     { 29140
9141         \int_incr:N \l__regex_added_begin_int 29141
9142         \int_incr:N \l__regex_added_end_int 29142
9143         \tex_afterassignment:D \__regex_extract_check:w 29143
9144         \__kernel_tl_gset:Nx \g__regex_internal_tl 29144
9145         { 29145
9146             \exp_after:wN \__regex_extract_check_loop:w 29146
9147             \g__regex_internal_tl 29147
```

```
9148         \__regex_tmp:w \__regex_extract_check_end:w
9149         #1
9150     }
9151 }
9152 }
9153 \cs_new:Npn \__regex_extract_check_loop:w #1 \__regex_tmp:w #2
9154 {
9155     #2
9156     \exp_not:o {#1}
9157     \__regex_tmp:w { }
9158     \__regex_extract_check_loop:w \prg_do_nothing:
9159 }
9160 \cs_new:Npn \__regex_extract_check_end:w
9161     \exp_not:o #1#2 \__regex_extract_check_loop:w #3 \__regex_tmp:w
9162 {
9163     { \exp_not:o {#1} }
9164     #3
9165     \if_false: { \fi: }
9166     \__regex_tmp:w
9167 }
9168 \cs_new_protected:Npn \__regex_extract:
9169 {
9170     \if_meaning:w \c_true_bool \g__regex_success_bool
9171     \int_set_eq:NN \l__regex_zeroth_submatch_int \l__regex_submatch_int
9172     \prg_replicate:nn \l__regex_capturing_group_int
9173     {
9174         \__kernel_intarray_gset:Nnn \g__regex_submatch_prev_intarray
9175         \l__regex_submatch_int \c_zero_int
9176         \__kernel_intarray_gset:Nnn \g__regex_submatch_case_intarray
9177         \l__regex_submatch_int \c_zero_int
9178         \int_incr:N \l__regex_submatch_int
9179     }
9180     \__kernel_intarray_gset:Nnn \g__regex_submatch_prev_intarray
9181     \l__regex_zeroth_submatch_int \l__regex_start_pos_int
9182     \__kernel_intarray_gset:Nnn \g__regex_submatch_case_intarray
9183     \l__regex_zeroth_submatch_int \g__regex_case_int
9184     \int_zero:N \l__regex_internal_a_int
9185     \exp_after:wN \__regex_extract_aux:w \l__regex_success_submatches_tl
9186     \prg_break_point: \__regex_use_none_delimit_by_q_recursion_stop:w ,
9187     \q__regex_recursion_stop
9188     \fi:
9189 }
9190 \cs_new_protected:Npn \__regex_extract_aux:w #1 ,
9191 {
9192     \prg_break: #1 \prg_break_point:
9193     \if_int_compare:w \l__regex_internal_a_int < \l__regex_capturing_group_int
```



```
9194     \__kernel_intarray_gset:Nnn \g__regex_submatch_begin_intarray 29194
9195     { \__regex_int_eval:w \l__regex_zeroth_submatch_int + \l__regex_internal_a_int } 29195 ✓
9196 {#1} 29196
9197 \else: 29197
9198     \__kernel_intarray_gset:Nnn \g__regex_submatch_end_intarray 29198
9199     { 29199
9200         \__regex_int_eval:w 29199
9201         \l__regex_zeroth_submatch_int + \l__regex_internal_a_int 29200
9202         - \l__regex_capturing_group_int 29201
9203     } 29202
9204     {#1} 29203
9205 \fi: 29204
9206 \int_incr:N \l__regex_internal_a_int 29205
9207 \__regex_extract_aux:w 29206
9208 } 29207
9209 \cs_new_protected:Npn \__regex_replace_once:nnN #1#2 29208
9210 { \__regex_replace_once_aux:nnN {#1} { \__regex_replacement:n {#2} } } 29209
9211 \cs_new_protected:Npn \__regex_replace_once_aux:nnN #1#2#3 29210
9212 { 29211
9213     \group_begin: 29212
9214     \__regex_single_match: 29213
9215     #1 29214
9216     \exp_args:No \__regex_match:n {#3} 29215
9217 \bool_if:NTF \g__regex_success_bool 29216
9218 { 29217
9219     \__regex_extract: 29218
9220     \exp_args:No \__regex_query_set:n {#3} 29219
9221     #2 29220
9222     \int_set:Nn \l__regex_balance_int 29221
9223     { \__regex_replacement_balance_one_match:n \l__regex_zeroth_submatch_int } 29222
9224     \__kernel_tl_set:Nx \l__regex_internal_a_tl 29223
9225     { 29224
9226         \__regex_replacement_do_one_match:n \l__regex_zeroth_submatch_int 29225
9227         \__regex_query_range:nn 29226
9228         { 29227
9229             \__kernel_intarray_item:Nn \g__regex_submatch_end_intarray 29228
9230             \l__regex_zeroth_submatch_int 29229
9231         } 29230
9232         \l__regex_max_pos_int 29231
9233     } 29232
9234     \__regex_group_end_replace:N #3 29233
9235 } 29234
9236 { \group_end: } 29235
9237 } 29236
9238 \cs_new_protected:Npn \__regex_replace_all:nnN #1#2 29237
9239 { \__regex_replace_all_aux:nnN {#1} { \__regex_replacement:n {#2} } } 29238
```



```
9239 \cs_new_protected:Npn \__regex_replace_all_aux:nnN #1#2#3
9240 {
9241   \group_begin:
9242     \__regex_multi_match:n { \__regex_extract: }
9243     #1
9244     \exp_args:No \__regex_match:n {#3}
9245     \exp_args:No \__regex_query_set:n {#3}
9246     #2
9247     \int_set:Nn \l__regex_balance_int
9248     {
9249       \c_zero_int
9250       \int_step_function:nnnN
9251         \l__regex_min_submatch_int
9252         \l__regex_capturing_group_int
9253         { \l__regex_submatch_int - \c_one_int }
9254         \__regex_replacement_balance_one_match:n
9255     }
9256     \__kernel_tl_set:Nx \l__regex_internal_a_tl
9257     {
9258       \int_step_function:nnnN
9259         \l__regex_min_submatch_int
9260         \l__regex_capturing_group_int
9261         { \l__regex_submatch_int - \c_one_int }
9262         \__regex_replacement_do_one_match:n
9263         \__regex_query_range:nn
9264         \l__regex_start_pos_int \l__regex_max_pos_int
9265     }
9266     \__regex_group_end_replace:N #3
9267   }
9268 \cs_new_protected:Npn \__regex_group_end_replace:N #1
9269 {
9270   \int_set:Nn \l__regex_added_begin_int
9271   { \int_max:nn { - \l__regex_balance_int } \c_zero_int }
9272   \int_set:Nn \l__regex_added_end_int
9273   { \int_max:nn \l__regex_balance_int \c_zero_int }
9274   \__regex_group_end_replace_try:
9275   \int_compare:nNnT { \l__regex_added_begin_int + \l__regex_added_end_int }
9276   > \c_zero_int
9277   {
9278     \msg_error:nneee { regex } { result-unbalanced }
9279     { replacing } { \int_use:N \l__regex_added_begin_int }
9280     { \int_use:N \l__regex_added_end_int }
9281   }
9282   \group_end:
9283   \tl_set_eq:NN #1 \g__regex_internal_tl
9284 }
```

```
9285 \cs_new_protected:Npn \__regex_group_end_replace_try: 2928
9286 { 2928
9287   \tex_afterassignment:D \__regex_group_end_replace_check:w 2928
9288   \__kernel_tl_gset:Nx \g__regex_internal_tl 2928
9289   { 2928
9290     \prg_replicate:nn \l__regex_added_begin_int { { \if_false: } \fi: } 2929
9291     \l__regex_internal_a_tl 2929
9292     \prg_replicate:nn \l__regex_added_end_int { \if_false: { \fi: } } 2929
9293     \if_false: { \fi: } 2929
9294   } 2929
9295 } 2929
9296 \cs_new_protected:Npn \__regex_group_end_replace_check:w 2929
9297 { 2929
9298   \exp_after:wN \__regex_group_end_replace_check:n 2929
9299   \exp_after:wN { \if_false: } \fi: 2929
9300 } 2930
9301 \cs_new_protected:Npn \__regex_group_end_replace_check:n #1 2930
9302 { 2930
9303   \tl_if_empty:nF {#1} 2930
9304   { 2930
9305     \int_incr:N \l__regex_added_begin_int 2930
9306     \int_incr:N \l__regex_added_end_int 2930
9307     \__regex_group_end_replace_try: 2930
9308   } 2930
9309 } 2930
9310 \tl_new:N \l__regex_peek_true_tl 2931
9311 \tl_new:N \l__regex_peek_false_tl 2931
9312 \tl_new:N \l__regex_replacement_tl 2931
9313 \tl_new:N \l__regex_input_tl 2931
9314 \cs_new_eq:NN \__regex_input_item:n ? 2931
9315 \cs_new_protected:Npn \peek_regex:nTF #1 2931
9316 { 2931
9317   \__regex_peek:nnTF 2931
9318   { \__regex_build_aux:Nn \c_false_bool {#1} } 2931
9319   { \__regex_peek_end: } 2931
9320 } 2932
9321 \cs_new_protected:Npn \peek_regex:nT #1#2 2932
9322 { \peek_regex:nTF {#1} {#2} { } } 2932
9323 \cs_new_protected:Npn \peek_regex:nF #1 { \peek_regex:nTF {#1} { } } 2932
9324 \cs_new_protected:Npn \peek_regex:NTF #1 2932
9325 { 2932
9326   \__regex_peek:nnTF 2932
9327   { \__regex_build_aux:NN \c_false_bool #1 } 2932
9328   { \__regex_peek_end: } 2932
9329 } 2932
9330 \cs_new_protected:Npn \peek_regex:NT #1#2 2933
```

```
9331 { \peek_regex:NTF #1 {#2} { } }
9332 \cs_new_protected:Npn \peek_regex:NF #1 { \peek_regex:NTF {#1} { } }
9333 \cs_new_protected:Npn \peek_regex_remove_once:nTF #1
9334 {
9335   \__regex_peek:nnTF
9336     { \__regex_build_aux:Nn \c_false_bool {#1} }
9337     { \__regex_peek_remove_end:n {##1} }
9338 }
9339 \cs_new_protected:Npn \peek_regex_remove_once:nT #1#2
9340 { \peek_regex_remove_once:nTF {#1} {#2} { } }
9341 \cs_new_protected:Npn \peek_regex_remove_once:nF #1
9342 { \peek_regex_remove_once:nTF {#1} { } }
9343 \cs_new_protected:Npn \peek_regex_remove_once:NTF #1
9344 {
9345   \__regex_peek:nnTF
9346     { \__regex_build_aux:NN \c_false_bool #1 }
9347     { \__regex_peek_remove_end:n {##1} }
9348 }
9349 \cs_new_protected:Npn \peek_regex_remove_once:NT #1#2
9350 { \peek_regex_remove_once:NTF #1 {#2} { } }
9351 \cs_new_protected:Npn \peek_regex_remove_once:NF #1
9352 { \peek_regex_remove_once:NTF #1 { } }
9353 \cs_new_protected:Npn \__regex_peek:nnTF #1
9354 {
9355   \__regex_peek_aux:nnTF
9356   {
9357     \__regex_disable_submatches:
9358     #1
9359   }
9360 }
9361 \cs_new_protected:Npn \__regex_peek_aux:nnTF #1#2#3#4
9362 {
9363   \group_begin:
9364     \tl_set:Nn \l__regex_peek_true_tl { \group_end: #3 }
9365     \tl_set:Nn \l__regex_peek_false_tl { \group_end: #4 }
9366     \__regex_single_match:
9367     #1
9368     \__regex_match_init:
9369     \tl_build_begin:N \l__regex_input_tl
9370     \__regex_match_once_init:
9371     \peek_analysis_map_inline:n
9372     {
9373       \tl_build_put_right:Nn \l__regex_input_tl
9374         { \__regex_input_item:n {##1} }
9375       \__regex_match_one_token:nnN {##1} {##2} ##3
9376       \use_none:nnn
```

```
9377         \prg_break_point:Nn \__regex_maplike_break: 29377
9378         { \peek_analysis_map_break:n {#2} } 29378
9379     } 29379
9380 } 29380
9381 \cs_new_protected:Npn \__regex_peek_end: 29381
9382 { 29382
9383     \bool_if:NTF \g__regex_success_bool 29383
9384     { \__regex_peek_reinsert:N \l__regex_peek_true_tl } 29384
9385     { \__regex_peek_reinsert:N \l__regex_peek_false_tl } 29385
9386 } 29386
9387 \cs_new_protected:Npn \__regex_peek_remove_end:n #1 29387
9388 { 29388
9389     \bool_if:NTF \g__regex_success_bool 29389
9390     { \exp_args:NNo \use:nn \l__regex_peek_true_tl {#1} } 29390
9391     { \__regex_peek_reinsert:N \l__regex_peek_false_tl } 29391
9392 } 29392
9393 \cs_new_protected:Npn \__regex_peek_reinsert:N #1 29393
9394 { 29394
9395     \tl_build_end:N \l__regex_input_tl 29395
9396     \cs_set_eq:NN \__regex_input_item:n \__regex_reinsert_item:n 29396
9397     \exp_after:wN #1 \exp:w \l__regex_input_tl \exp_end: 29397
9398 } 29398
9399 \cs_new_protected:Npn \__regex_reinsert_item:n #1 29399
9400 { 29400
9401     \exp_after:wN \exp_after:wN 29401
9402     \exp_after:wN \exp_end: 29402
9403     \exp_after:wN \exp_after:wN 29403
9404     #1 29404
9405     \exp:w 29405
9406 } 29406
9407 \cs_new_protected:Npn \peek_regex_replace_once:nnTF #1 29407
9408 { \__regex_peek_replace:nnTF { \__regex_build_aux:Nn \c_false_bool {#1} } } 29408
9409 \cs_new_protected:Npn \peek_regex_replace_once:nnT #1#2#3 29409
9410 { \peek_regex_replace_once:nnTF {#1} {#2} {#3} { } } 29410
9411 \cs_new_protected:Npn \peek_regex_replace_once:nnF #1#2 29411
9412 { \peek_regex_replace_once:nnTF {#1} {#2} { } } 29412
9413 \cs_new_protected:Npn \peek_regex_replace_once:nn #1#2 29413
9414 { \peek_regex_replace_once:nnTF {#1} {#2} { } { } } 29414
9415 \cs_new_protected:Npn \peek_regex_replace_once:NnTF #1 29415
9416 { \__regex_peek_replace:nnTF { \__regex_build_aux:NN \c_false_bool #1 } } 29416
9417 \cs_new_protected:Npn \peek_regex_replace_once:NnT #1#2#3 29417
9418 { \peek_regex_replace_once:NnTF #1 {#2} {#3} { } } 29418
9419 \cs_new_protected:Npn \peek_regex_replace_once:NnF #1#2 29419
9420 { \peek_regex_replace_once:NnTF #1 {#2} { } } 29420
9421 \cs_new_protected:Npn \peek_regex_replace_once:Nn #1#2 29421
9422 { \peek_regex_replace_once:NnTF #1 {#2} { } { } } 29422
```

```
9423 \cs_new_protected:Npn \__regex_peek_replace:nnTF #1#2
9424 {
9425     \tl_set:Nn \l__regex_replacement_tl {#2}
9426     \__regex_peek_aux:nnTF {#1} { \__regex_peek_replace_end: }
9427 }
9428 \cs_new_protected:Npn \__regex_peek_replace_end:
9429 {
9430     \bool_if:NTF \g__regex_success_bool
9431     {
9432         \__regex_extract:
9433         \__regex_query_set_from_input_tl:
9434         \cs_set_eq:NN \__regex_replacement_put:n \__regex_peek_replacement_put:n
9435         \cs_set_eq:NN \__regex_replacement_put_submatch_aux:n
9436         \__regex_peek_replacement_put_submatch_aux:n
9437         \cs_set_eq:NN \__regex_input_item:n \__regex_reinsert_item:n
9438         \cs_set_eq:NN \__regex_replacement_exp_not:N \__regex_peek_replacement_token:n
9439         \cs_set_eq:NN \__regex_replacement_exp_not:V \__regex_peek_replacement_var:N
9440         \exp_args:No \__regex_replacement:n { \l__regex_replacement_tl }
9441         \use:e
9442         {
9443             \exp_not:n { \exp_after:wN \l__regex_peek_true_tl \exp:w }
9444             \__regex_replacement_do_one_match:n \l__regex_zeroth_submatch_int
9445             \__regex_query_range:nn
9446             {
9447                 \__kernel_intarray_item:Nn \g__regex_submatch_end_intarray
9448                 \l__regex_zeroth_submatch_int
9449             }
9450             \l__regex_max_pos_int
9451             \exp_end:
9452         }
9453     }
9454     { \__regex_peek_reinsert:N \l__regex_peek_false_tl }
9455 }
9456 \cs_new_protected:Npn \__regex_query_set_from_input_tl:
9457 {
9458     \tl_build_end:N \l__regex_input_tl
9459     \int_zero:N \l__regex_curr_pos_int
9460     \cs_set_eq:NN \__regex_input_item:n \__regex_query_set_item:n
9461     \__regex_query_set_item:n { }
9462     \l__regex_input_tl
9463     \__regex_query_set_item:n { }
9464     \int_set_eq:NN \l__regex_max_pos_int \l__regex_curr_pos_int
9465 }
9466 \cs_new_protected:Npn \__regex_query_set_item:n #1
9467 {
9468     \int_incr:N \l__regex_curr_pos_int
```

```
9469 \__regex_toks_set:Nn \l__regex_curr_pos_int { \__regex_input_item:n {#1} }
9470 }
9471 \cs_new_protected:Npn \__regex_peek_replacement_put:n #1
9472 {
9473   \if_case:w \l__regex_replacement_csnames_int
9474     \tl_build_put_right:Nn \l__regex_build_tl
9475       { \exp_not:N \__regex_reinsert_item:n {#1} }
9476   \else:
9477     \tl_build_put_right:Nn \l__regex_build_tl {#1}
9478   \fi:
9479 }
9480 \cs_new_protected:Npn \__regex_peek_replacement_token:n #1
9481 { \exp_after:wN \exp_end: \exp_after:wN #1 \exp:w }
9482 \cs_new_protected:Npn \__regex_peek_replacement_put_submatch_aux:n #1
9483 {
9484   \if_case:w \l__regex_replacement_csnames_int
9485     \tl_build_put_right:Nn \l__regex_build_tl
9486       { \__regex_query_submatch:n { \__regex_int_eval:w #1 + ##1 \scan_stop: } }
9487   \else:
9488     \tl_build_put_right:Nn \l__regex_build_tl
9489     {
9490       \exp:w
9491         \__regex_query_submatch:n { \__regex_int_eval:w #1 + ##1 \scan_stop: }
9492       \exp_end:
9493     }
9494   \fi:
9495 }
9496 \cs_new_protected:Npn \__regex_peek_replacement_var:N #1
9497 {
9498   \exp_after:wN \exp_last_unbraced:NV
9499   \exp_after:wN \exp_end:
9500   \exp_after:wN #1
9501   \exp:w
9502 }
9503 \use:e
9504 {
9505   \msg_new:nnn { regex } { trailing-backslash }
9506     { Trailing~'\iow_char:N\\'\in~regex~or~replacement. }
9507   \msg_new:nnn { regex } { x-missing-rbrace }
9508     {
9509       Missing~brace~'\iow_char:N\}'in~regex~
9510       '...\iow_char:N\\x\iow_char:N{...##1'.
9511     }
9512   \msg_new:nnn { regex } { x-overflow }
9513     {
9514       Character~code~##1~too~large~in~
```

```

9515 \iow_char:N\{x\iow_char:N\{##2\iow_char:N\}~regex.
9516 }
9517 }
9518 \msg_new:nnnn { regex } { invalid-quantifier }
9519 { Braced~quantifier~'#1'~may~not~be~followed~by~'#2'. }
9520 {
9521   The~character~'#2'~is~invalid~in~the~braced~quantifier~'#1'.~
9522   The~only~valid~quantifiers~are~'~*~','~?~','~+~','~{<int>}',~
9523   '{<min>,<max>}'~and~'{<min>,<max>}',~optionally~followed~by~'?'~.
9524 }
9525 \msg_new:nnnn { regex } { missing-rbrack }
9526 { Missing~right~bracket~inserted~in~regular~expression. }
9527 {
9528   LaTeX~was~given~a~regular~expression~where~a~character~class~
9529   was~started~with~'[',~but~the~matching~']'~is~missing.
9530 }
9531 \msg_new:nnnn { regex } { missing-rparen }
9532 {
9533   Missing~right~
9534   \int_compare:nTF { #1 = 1 } { parenthesis } { parentheses } ~
9535   inserted~in~regular~expression.
9536 }
9537 {
9538   LaTeX~was~given~a~regular~expression~with~\int_eval:n {#1} ~
9539   more~left~parentheses~than~right~parentheses.
9540 }
9541 \msg_new:nnnn { regex } { extra-rparen }
9542 { Extra~right~parenthesis~ignored~in~regular~expression. }
9543 {
9544   LaTeX~came~across~a~closing~parenthesis~when~no~submatch~group~
9545   was~open.~The~parenthesis~will~be~ignored.
9546 }
9547 \msg_new:nnnn { regex } { bad-escape }
9548 {
9549   Invalid~escape~'\iow_char:N\{#1'~
9550   \__regex_if_in_cs:TF { within~a~control~sequence. }
9551   {
9552     \__regex_if_in_class:TF
9553     { in~a~character~class. }
9554     { following~a~category~test. }
9555   }
9556 }
9557 {
9558   The~escape~sequence~'\iow_char:N\{#1'~may~not~appear~
9559   \__regex_if_in_cs:TF
9560   {

```



```
9561         within~a~control~sequence~test~introduced~by~
9562         '\iow_char:N\\c\iow_char:N\{' .
9563     }
9564     {
9565         \__regex_if_in_class:TF
9566         { within~a~character~class~ }
9567         { following~a~category~test~such~as~'\iow_char:N\\cL'~ }
9568         because~it~does~not~match~exactly~one~character.
9569     }
9570 }
9571 \msg_new:nnnn { regex } { range-missing-end }
9572 { Invalid~end~point~for~range~'#1-#2'~in~character~class. }
9573 {
9574     The~end~point~'#2'~of~the~range~'#1-#2'~may~not~serve~as~an~
9575     end~point~for~a~range:~alphanumeric~characters~should~not~be~
9576     escaped,~and~non~alphanumeric~characters~should~be~escaped.
9577 }
9578 \msg_new:nnnn { regex } { range-backwards }
9579 { Range~'[#1-#2]'~out~of~order~in~character~class. }
9580 {
9581     In~ranges~of~characters~'[x-y]'~appearing~in~character~classes,~
9582     the~first~character~code~must~not~be~larger~than~the~second.~
9583     Here,~'#1'~has~character~code~\int_eval:n {`#1},~while~
9584     '#2'~has~character~code~\int_eval:n {`#2}.
9585 }
9586 \msg_new:nnnn { regex } { c-bad-mode }
9587 { Invalid~nested~'\iow_char:N\\c'~escape~in~regular~expression. }
9588 {
9589     The~'\iow_char:N\\c'~escape~cannot~be~used~within~
9590     a~control~sequence~test~'\iow_char:N\\c{...}'~
9591     nor~another~category~test.~
9592     To~combine~several~category~tests,~use~'\iow_char:N\\c[...]'.
9593 }
9594 \msg_new:nnnn { regex } { c-C-invalid }
9595 { '\iow_char:N\\cC'~should~be~followed~by~'.'~or~'(',~not~'#1'. }
9596 {
9597     The~'\iow_char:N\\cC'~construction~restricts~the~next~item~to~be~a~
9598     control~sequence~or~the~next~group~to~be~made~of~control~sequences.~
9599     It~only~makes~sense~to~follow~it~by~'.'~or~by~a~group.
9600 }
9601 \msg_new:nnnn { regex } { cu-lbrace }
9602 { Left~braces~must~be~escaped~in~'\iow_char:N\\#1{...}'. }
9603 {
9604     Constructions~such~as~'\iow_char:N\\#1{... \iow_char:N\{...}'~are~
9605     not~allowed~and~should~be~replaced~by~
9606     '\iow_char:N\\#1{... \token_to_str:N\{...}'.
```

```

9607 }
9608 \msg_new:nnnn { regex } { c-lparen-in-class }
9609 { Catcode~test~cannot~apply~to~group~in~character~class }
9610 {
9611   Construction~such~as~'\iow_char:N\\cL(abc)'\~are~not~allowed~inside~a~
9612   class~'[...]\~because~classes~do~not~match~multiple~characters~at~once.
9613 }
9614 \msg_new:nnnn { regex } { c-missing-rbrace }
9615 { Missing~right~brace~inserted~for~'\iow_char:N\\c'\~escape. }
9616 {
9617   LaTeX~was~given~a~regular~expression~where~a~
9618   '\iow_char:N\\c\iow_char:N{\...'\~construction~was~not~ended~
9619   with~a~closing~brace~'\iow_char:N\}'.
9620 }
9621 \msg_new:nnnn { regex } { c-missing-rbrack }
9622 { Missing~right~bracket~inserted~for~'\iow_char:N\\c'\~escape. }
9623 {
9624   A~construction~'\iow_char:N\\c[...'\~appears~in~a~
9625   regular~expression,~but~the~closing~']'\~is~not~present.
9626 }
9627 \msg_new:nnnn { regex } { c-missing-category }
9628 { Invalid~character~'#1'\~following~'\iow_char:N\\c'\~escape. }
9629 {
9630   In~regular~expressions,~the~'\iow_char:N\\c'\~escape~sequence~
9631   may~only~be~followed~by~a~left~brace,~a~left~bracket,~or~a~
9632   capital~letter~representing~a~character~category,~namely~
9633   one~of~'ABCDELMOPTU'.
9634 }
9635 \msg_new:nnnn { regex } { c-trailing }
9636 { Trailing~category~code~escape~'\iow_char:N\\c'... }
9637 {
9638   A~regular~expression~ends~with~'\iow_char:N\\c'\~followed~
9639   by~a~letter.~It~will~be~ignored.
9640 }
9641 \msg_new:nnnn { regex } { u-missing-lbrace }
9642 { Missing~left~brace~following~'\iow_char:N\\u'\~escape. }
9643 {
9644   The~'\iow_char:N\\u'\~escape~sequence~must~be~followed~by~
9645   a~brace~group~with~the~name~of~the~variable~to~use.
9646 }
9647 \msg_new:nnnn { regex } { u-missing-rbrace }
9648 { Missing~right~brace~inserted~for~'\iow_char:N\\u'\~escape. }
9649 {
9650   LaTeX~
9651   \str_if_eq:eeTF { } {#2}
9652   { reached~the~end~of~the~string~ }

```

```
9653 { encountered~an~escaped~alphanumeric~character '\iow_char:N\|#2'~ }
9654 when~parsing~the~argument~of~an~
9655 '\iow_char:N\\u\iow_char:N\{...\}'~escape.
9656 }
9657 \msg_new:nnnn { regex } { posix-unsupported }
9658 { POSIX~collating~element~'#1 ~ #1'~not~supported. }
9659 {
9660 The~'[.foo.]'~and~'[=bar=]'~syntaxes~have~a~special~meaning~
9661 in~POSIX~regular~expressions.~This~is~not~supported~by~LaTeX.~
9662 Maybe~you~forgot~to~escape~a~left~bracket~in~a~character~class?
9663 }
9664 \msg_new:nnnn { regex } { posix-unknown }
9665 { POSIX~class~'[:#1:]'~unknown. }
9666 {
9667 '[:#1:]'~is~not~among~the~known~POSIX~classes~
9668 '[:alnum:]',~'[:alpha:]',~'[:ascii:]',~'[:blank:]',~
9669 '[:cntrl:]',~'[:digit:]',~'[:graph:]',~'[:lower:]',~
9670 '[:print:]',~'[:punct:]',~'[:space:]',~'[:upper:]',~
9671 '[:word:]',~and~'[:xdigit:]'.
9672 }
9673 \msg_new:nnnn { regex } { posix-missing-close }
9674 { Missing~closing~':'~for~POSIX~class. }
9675 { The~POSIX~syntax~'#1'~must~be~followed~by~':',~not~'#2'. }
9676 \msg_new:nnnn { regex } { result-unbalanced }
9677 { Missing~brace~inserted~when~#1. }
9678 {
9679 LaTeX~was~asked~to~do~some~regular~expression~operation,~
9680 and~the~resulting~token~list~would~not~have~the~same~number~
9681 of~begin-group~and~end-group~tokens.~Braces~were~inserted:~
9682 #2~left,~#3~right.
9683 }
9684 \msg_new:nnnn { regex } { unknown-option }
9685 { Unknown~option~'#1'~for~regular~expressions. }
9686 {
9687 The~only~available~option~is~'case-insensitive',~toggled~by~
9688 '(?i)'~and~'(?-i)'.
9689 }
9690 \msg_new:nnnn { regex } { special-group-unknown }
9691 { Unknown~special~group~'#1~...~'in~a~regular~expression. }
9692 {
9693 The~only~valid~constructions~starting~with~'(?~are~
9694 '(:~...~)',~'(|~...~)',~'(?i)',~and~'(?-i)'.
9695 }
9696 \msg_new:nnnn { regex } { replacement-c }
9697 { Misused~'\iow_char:N\c'~command~in~a~replacement~text. }
9698 {
```

```
9699 In~a~replacement~text,~the~'\iow_char:N\\c'~escape~sequence~
9700 can~be~followed~by~one~of~the~letters~'ABCDELMOPSTU'~
9701 or~a~brace~group,~not~by~'#1'.
9702 }
9703 \msg_new:nnnn { regex } { replacement-u }
9704 { Misused~'\iow_char:N\\u'~command~in~a~replacement~text. }
9705 {
9706 In~a~replacement~text,~the~'\iow_char:N\\u'~escape~sequence~
9707 must~be~followed~by~a~brace~group~holding~the~name~of~the~
9708 variable~to~use.
9709 }
9710 \msg_new:nnnn { regex } { replacement-g }
9711 {
9712 Missing~brace~for~the~'\iow_char:N\\g'~construction~
9713 in~a~replacement~text.
9714 }
9715 {
9716 In~the~replacement~text~for~a~regular~expression~search,~
9717 submatches~are~represented~either~as~'\iow_char:N \\g{dd..d}',~
9718 or~'\\d',~where~'d'~are~single~digits.~Here,~a~brace~is~missing.
9719 }
9720 \msg_new:nnnn { regex } { replacement-catcode-end }
9721 {
9722 Missing~character~for~the~'\iow_char:N\\c<category><character>'~
9723 construction~in~a~replacement~text.
9724 }
9725 {
9726 In~a~replacement~text,~the~'\iow_char:N\\c'~escape~sequence~
9727 can~be~followed~by~one~of~the~letters~'ABCDELMOPSTU'~representing~
9728 the~character~category.~Then,~a~character~must~follow.~LaTeX~
9729 reached~the~end~of~the~replacement~when~looking~for~that.
9730 }
9731 \msg_new:nnnn { regex } { replacement-catcode-escaped }
9732 {
9733 Escaped~letter~or~digit~after~category~code~in~replacement~text.
9734 }
9735 {
9736 In~a~replacement~text,~the~'\iow_char:N\\c'~escape~sequence~
9737 can~be~followed~by~one~of~the~letters~'ABCDELMOPSTU'~representing~
9738 the~character~category.~Then,~a~character~must~follow,~not~
9739 '\iow_char:N\\#2'.
9740 }
9741 \msg_new:nnnn { regex } { replacement-catcode-in-cs }
9742 {
9743 Category~code~'\iow_char:N\\c#1#3'~ignored~inside~
9744 '\iow_char:N\\c{\...}'~in~a~replacement~text.
```

```

9745 }
9746 {
9747     In a replacement text, the category codes of the argument of
9748     '\iow_char:N\c\{...\}' are ignored when building the control
9749     sequence name.
9750 }
9751 \msg_new:nnnn { regex } { replacement-null-space }
9752 { TeX cannot build a space token with character code 0. }
9753 {
9754     You asked for a character token with category space,
9755     and character code 0, for instance through
9756     '\iow_char:N\cS\iow_char:N\x00'.
9757     This specific case is impossible and will be replaced
9758     by a normal space.
9759 }
9760 \msg_new:nnnn { regex } { replacement-missing-rbrace }
9761 { Missing right brace inserted in replacement text. }
9762 {
9763     There \int_compare:nTF { #1 = 1 } { was } { were } ~ #1~
9764     missing right \int_compare:nTF { #1 = 1 } { brace } { braces } .
9765 }
9766 \msg_new:nnnn { regex } { replacement-missing-rparen }
9767 { Missing right parenthesis inserted in replacement text. }
9768 {
9769     There \int_compare:nTF { #1 = 1 } { was } { were } ~ #1~
9770     missing right
9771     \int_compare:nTF { #1 = 1 } { parenthesis } { parentheses } .
9772 }
9773 \msg_new:nnn { regex } { submatch-too-big }
9774 { Submatch #1 used but regex only has #2 group(s) }
9775 \msg_new:nnnn { regex } { backwards-quantifier }
9776 { Quantifier "{#1,#2}" is backwards. }
9777 { The values given in a quantifier must be in order. }
9778 \msg_new:nnnn { regex } { case-odd }
9779 { #1 with odd number of items }
9780 {
9781     There must be a #2 part for each regex:
9782     found odd number of items (#3) in \
9783     \iow_indent:n {#4}
9784 }
9785 \msg_new:nnn { regex } { show }
9786 {
9787     >Compiled regex~
9788     \tl_if_empty:nTF {#1} { variable #2 } { {#1} } :
9789     #3
9790 }

```

```

9791 \prop_gput:Nnn \g_msg_module_name_prop { regex } { LaTeX }
9792 \prop_gput:Nnn \g_msg_module_type_prop { regex } { }
9793 \cs_new:Npn \__regex_msg_repeated:nnN #1#2#3
9794 {
9795   \str_if_eq:eeF { #1 #2 } { 1 0 }
9796   {
9797     , ~ repeated ~
9798     \int_case:nnF {#2}
9799     {
9800       { -1 } { #1~or~more~times,~\bool_if:NTF #3 { lazy } { greedy } }
9801       { 0 } { #1~times }
9802     }
9803     {
9804       between~#1~and~\int_eval:n {#1+#2}~times,~
9805       \bool_if:NTF #3 { lazy } { greedy }
9806     }
9807   }
9808 }
9809 \cs_new_protected:Npn \__regex_trace_push:nnN #1#2#3
9810 { \__regex_trace:nne {#1} {#2} { entering~ \token_to_str:N #3 } }
9811 \cs_new_protected:Npn \__regex_trace_pop:nnN #1#2#3
9812 { \__regex_trace:nne {#1} {#2} { leaving~ \token_to_str:N #3 } }
9813 \cs_new_protected:Npn \__regex_trace:nne #1#2#3
9814 {
9815   \int_compare:nNnF
9816   { \int_use:c { g__regex_trace_#1_int } } < {#2}
9817   { \iow_term:e { Trace:~#3 } }
9818 }
9819 \int_new:N \g__regex_trace_regex_int
9820 \cs_new_protected:Npn \__regex_trace_states:n #1
9821 {
9822   \int_step_inline:nnn
9823   \l__regex_min_state_int
9824   { \l__regex_max_state_int - \c_one_int }
9825   {
9826     \__regex_trace:nne { regex } {#1}
9827     { \iow_char:N \toks ##1 = { \__regex_toks_use:w ##1 } }
9828   }
9829 }
9830 %% File: l3box.dtx
9831 \cs_new_eq:NN \__box_dim_eval:w \tex_dimexpr:D
9832 \cs_new:Npn \__box_dim_eval:n #1
9833 { \__box_dim_eval:w #1 \scan_stop: }
9834 \cs_new_protected:Npn \__kernel_kern:n #1
9835 { \tex_kern:D \__box_dim_eval:n {#1} }
9836 \cs_new_protected:Npn \box_new:N #1

```


9837	{	2983
9838	_kernel_chk_if_free_cs:N #1	2983
9839	\cs:w newbox \cs_end: #1	2983
9840	}	2984
9841	\cs_generate_variant:Nn \box_new:N { c }	2984
9842	\cs_new_protected:Npn \box_clear:N #1	2984
9843	{ \box_set_eq:NN #1 \c_empty_box }	2984
9844	\cs_new_protected:Npn \box_gclear:N #1	2984
9845	{ \box_gset_eq:NN #1 \c_empty_box }	2984
9846	\cs_generate_variant:Nn \box_clear:N { c }	2984
9847	\cs_generate_variant:Nn \box_gclear:N { c }	2984
9848	\cs_new_protected:Npn \box_clear_new:N #1	2984
9849	{ \box_if_exist:NTF #1 { \box_clear:N #1 } { \box_new:N #1 } }	2984
9850	\cs_new_protected:Npn \box_gclear_new:N #1	2985
9851	{ \box_if_exist:NTF #1 { \box_gclear:N #1 } { \box_new:N #1 } }	2985
9852	\cs_generate_variant:Nn \box_clear_new:N { c }	2985
9853	\cs_generate_variant:Nn \box_gclear_new:N { c }	2985
9854	\cs_new_protected:Npn \box_set_eq:NN #1#2	2985
9855	{ \tex_setbox:D #1 \tex_copy:D #2 }	2985
9856	\cs_new_protected:Npn \box_gset_eq:NN #1#2	2985
9857	{ \tex_global:D \tex_setbox:D #1 \tex_copy:D #2 }	2985
9858	\cs_generate_variant:Nn \box_set_eq:NN { c , Nc , cc }	2985
9859	\cs_generate_variant:Nn \box_gset_eq:NN { c , Nc , cc }	2985
9860	\cs_new_protected:Npn \box_set_eq_drop:NN #1#2	2986
9861	{ \tex_setbox:D #1 \tex_box:D #2 }	2986
9862	\cs_new_protected:Npn \box_gset_eq_drop:NN #1#2	2986
9863	{ \tex_global:D \tex_setbox:D #1 \tex_box:D #2 }	2986
9864	\cs_generate_variant:Nn \box_set_eq_drop:NN { c , Nc , cc }	2986
9865	\cs_generate_variant:Nn \box_gset_eq_drop:NN { c , Nc , cc }	2986
9866	\prg_new_eq_conditional:NNn \box_if_exist:N \cs_if_exist:N	2986
9867	{ TF , T , F , p }	2986
9868	\prg_new_eq_conditional:NNn \box_if_exist:c \cs_if_exist:c	2986
9869	{ TF , T , F , p }	2986
9870	\cs_new_eq:NN \box_ht:N \tex_ht:D	2987
9871	\cs_new_eq:NN \box_dp:N \tex_dp:D	2987
9872	\cs_new_eq:NN \box_wd:N \tex_wd:D	2987
9873	\cs_generate_variant:Nn \box_ht:N { c }	2987
9874	\cs_generate_variant:Nn \box_dp:N { c }	2987
9875	\cs_generate_variant:Nn \box_wd:N { c }	2987
9876	\cs_new_protected:Npn \box_ht_plus_dp:N #1	2987
9877	{ _box_dim_eval:n { \box_ht:N #1 + \box_dp:N #1 } }	2987
9878	\cs_generate_variant:Nn \box_ht_plus_dp:N { c }	2987
9879	\cs_new_protected:Npn \box_set_dp:Nn #1#2	2987
9880	{	2988
9881	\tex_setbox:D #1 = \tex_copy:D #1	2988
9882	\box_dp:N #1 _box_dim_eval:n {#2}	2988


```

9883 }
9884 \cs_generate_variant:Nn \box_set_dp:Nn { c }
9885 \cs_new_protected:Npn \box_gset_dp:Nn #1#2
9886 { \box_dp:N #1 \__box_dim_eval:n {#2} }
9887 \cs_generate_variant:Nn \box_gset_dp:Nn { c }
9888 \cs_new_protected:Npn \box_set_ht:Nn #1#2
9889 {
9890     \tex_setbox:D #1 = \tex_copy:D #1
9891     \box_ht:N #1 \__box_dim_eval:n {#2}
9892 }
9893 \cs_generate_variant:Nn \box_set_ht:Nn { c }
9894 \cs_new_protected:Npn \box_gset_ht:Nn #1#2
9895 { \box_ht:N #1 \__box_dim_eval:n {#2} }
9896 \cs_generate_variant:Nn \box_gset_ht:Nn { c }
9897 \cs_new_protected:Npn \box_set_wd:Nn #1#2
9898 {
9899     \tex_setbox:D #1 = \tex_copy:D #1
9900     \box_wd:N #1 \__box_dim_eval:n {#2}
9901 }
9902 \cs_generate_variant:Nn \box_set_wd:Nn { c }
9903 \cs_new_protected:Npn \box_gset_wd:Nn #1#2
9904 { \box_wd:N #1 \__box_dim_eval:n {#2} }
9905 \cs_generate_variant:Nn \box_gset_wd:Nn { c }
9906 \cs_new_eq:NN \box_use_drop:N \tex_box:D
9907 \cs_new_eq:NN \box_use:N \tex_copy:D
9908 \cs_generate_variant:Nn \box_use_drop:N { c }
9909 \cs_generate_variant:Nn \box_use:N { c }
9910 \cs_new_protected:Npn \box_move_left:nn #1#2
9911 { \tex_movelleft:D \__box_dim_eval:n {#1} #2 }
9912 \cs_new_protected:Npn \box_move_right:nn #1#2
9913 { \tex_moveright:D \__box_dim_eval:n {#1} #2 }
9914 \cs_new_protected:Npn \box_move_up:nn #1#2
9915 { \tex_raise:D \__box_dim_eval:n {#1} #2 }
9916 \cs_new_protected:Npn \box_move_down:nn #1#2
9917 { \tex_lower:D \__box_dim_eval:n {#1} #2 }
9918 \cs_new_eq:NN \if_hbox:N \tex_ifhbox:D
9919 \cs_new_eq:NN \if_vbox:N \tex_ifvbox:D
9920 \cs_new_eq:NN \if_box_empty:N \tex_ifvoid:D
9921 \prg_new_conditional:Npnn \box_if_horizontal:N #1 { p , T , F , TF }
9922 { \if_hbox:N #1 \prg_return_true: \else: \prg_return_false: \fi: }
9923 \prg_new_conditional:Npnn \box_if_vertical:N #1 { p , T , F , TF }
9924 { \if_vbox:N #1 \prg_return_true: \else: \prg_return_false: \fi: }
9925 \prg_generate_conditional_variant:Nnn \box_if_horizontal:N
9926 { c } { p , T , F , TF }
9927 \prg_generate_conditional_variant:Nnn \box_if_vertical:N
9928 { c } { p , T , F , TF }

```

```
9929 \prg_new_conditional:Npnn \box_if_empty:N #1 { p , T , F , TF }
9930 { \if_box_empty:N #1 \prg_return_true: \else: \prg_return_false: \fi: }
9931 \prg_generate_conditional_variant:Nnn \box_if_empty:N
9932 { c } { p , T , F , TF }
9933 \cs_new_protected:Npn \box_set_to_last:N #1
9934 { \tex_setbox:D #1 \tex_lastbox:D }
9935 \cs_new_protected:Npn \box_gset_to_last:N #1
9936 { \tex_global:D \tex_setbox:D #1 \tex_lastbox:D }
9937 \cs_generate_variant:Nn \box_set_to_last:N { c }
9938 \cs_generate_variant:Nn \box_gset_to_last:N { c }
9939 \box_new:N \c_empty_box
9940 \box_new:N \l_tmpa_box
9941 \box_new:N \l_tmpb_box
9942 \box_new:N \g_tmpa_box
9943 \box_new:N \g_tmpb_box
9944 \cs_new_protected:Npn \box_show:N #1
9945 { \box_show:Nnn #1 \c_max_int \c_max_int }
9946 \cs_generate_variant:Nn \box_show:N { c }
9947 \cs_new_protected:Npn \box_show:Nnn #1#2#3
9948 { \__box_show:NNff 1 #1 { \int_eval:n {#2} } { \int_eval:n {#3} } }
9949 \cs_generate_variant:Nn \box_show:Nnn { c }
9950 \cs_new_protected:Npn \box_log:N #1
9951 { \box_log:Nnn #1 \c_max_int \c_max_int }
9952 \cs_generate_variant:Nn \box_log:N { c }
9953 \cs_new_protected:Npn \box_log:Nnn
9954 { \exp_args:No \__box_log:nNnn { \tex_the:D \tex_interactionmode:D } }
9955 \cs_new_protected:Npn \__box_log:nNnn #1#2#3#4
9956 {
9957   \int_gset:Nn \tex_interactionmode:D { 0 }
9958   \__box_show:NNff 0 #2 { \int_eval:n {#3} } { \int_eval:n {#4} }
9959   \int_gset:Nn \tex_interactionmode:D {#1}
9960 }
9961 \cs_generate_variant:Nn \box_log:Nnn { c }
9962 \cs_new_protected:Npn \__box_show:NNnn #1#2#3#4
9963 {
9964   \box_if_exist:NTF #2
9965   {
9966     \group_begin:
9967       \int_set:Nn \tex_showboxbreadth:D {#3}
9968       \int_set:Nn \tex_showboxdepth:D {#4}
9969       \int_set:Nn \tex_tracingonline:D {#1}
9970       \int_set:Nn \tex_errorcontextlines:D { -1 }
9971       \tex_showbox:D \use:n {#2}
9972     \group_end:
9973   }
9974 }
```

```
9975 \msg_error:nne { kernel } { variable-not-defined } 29975
9976 { \token_to_str:N #2 } 29976
9977 } 29977
9978 } 29978
9979 \cs_generate_variant:Nn \__box_show:NNnn { NNff } 29979
9980 \cs_new_protected:Npn \hbox:n #1 29980
9981 { \tex_hbox:D \scan_stop: { \color_group_begin: #1 \color_group_end: } } 29981
9982 \cs_new_protected:Npn \hbox_set:Nn #1#2 29982
9983 { 29983
9984 \tex_setbox:D #1 \tex_hbox:D 29984
9985 { \color_group_begin: #2 \color_group_end: } 29985
9986 } 29986
9987 \cs_new_protected:Npn \hbox_gset:Nn #1#2 29987
9988 { 29988
9989 \tex_global:D \tex_setbox:D #1 \tex_hbox:D 29989
9990 { \color_group_begin: #2 \color_group_end: } 29990
9991 } 29991
9992 \cs_generate_variant:Nn \hbox_set:Nn { c } 29992
9993 \cs_generate_variant:Nn \hbox_gset:Nn { c } 29993
9994 \cs_new_protected:Npn \hbox_set_to_wd:Nnn #1#2#3 29994
9995 { 29995
9996 \tex_setbox:D #1 \tex_hbox:D to \__box_dim_eval:n {#2} 29996
9997 { \color_group_begin: #3 \color_group_end: } 29997
9998 } 29998
9999 \cs_new_protected:Npn \hbox_gset_to_wd:Nnn #1#2#3 29999
0000 { 30000
0001 \tex_global:D \tex_setbox:D #1 \tex_hbox:D to \__box_dim_eval:n {#2} 30001
0002 { \color_group_begin: #3 \color_group_end: } 30002
0003 } 30003
0004 \cs_generate_variant:Nn \hbox_set_to_wd:Nnn { c } 30004
0005 \cs_generate_variant:Nn \hbox_gset_to_wd:Nnn { c } 30005
0006 \cs_new_protected:Npn \hbox_set:Nw #1 30006
0007 { 30007
0008 \tex_setbox:D #1 \tex_hbox:D 30008
0009 \c_group_begin_token 30009
0010 \color_group_begin: 30010
0011 } 30011
0012 \cs_new_protected:Npn \hbox_gset:Nw #1 30012
0013 { 30013
0014 \tex_global:D \tex_setbox:D #1 \tex_hbox:D 30014
0015 \c_group_begin_token 30015
0016 \color_group_begin: 30016
0017 } 30017
0018 \cs_generate_variant:Nn \hbox_set:Nw { c } 30018
0019 \cs_generate_variant:Nn \hbox_gset:Nw { c } 30019
0020 \cs_new_protected:Npn \hbox_set_end: 30020
```

```

0021 {
0022     \color_group_end:
0023     \c_group_end_token
0024 }
0025 \cs_new_eq:NN \hbox_gset_end: \hbox_set_end:
0026 \cs_new_protected:Npn \hbox_set_to_wd:Nnw #1#2
0027 {
0028     \tex_setbox:D #1 \tex_hbox:D to \__box_dim_eval:n {#2}
0029     \c_group_begin_token
0030     \color_group_begin:
0031 }
0032 \cs_new_protected:Npn \hbox_gset_to_wd:Nnw #1#2
0033 {
0034     \tex_global:D \tex_setbox:D #1 \tex_hbox:D to \__box_dim_eval:n {#2}
0035     \c_group_begin_token
0036     \color_group_begin:
0037 }
0038 \cs_generate_variant:Nn \hbox_set_to_wd:Nnw { c }
0039 \cs_generate_variant:Nn \hbox_gset_to_wd:Nnw { c }
0040 \cs_new_protected:Npn \hbox_to_wd:nn #1#2
0041 {
0042     \tex_hbox:D to \__box_dim_eval:n {#1}
0043     { \color_group_begin: #2 \color_group_end: }
0044 }
0045 \cs_new_protected:Npn \hbox_to_zero:n #1
0046 {
0047     \tex_hbox:D to \c_zero_dim
0048     { \color_group_begin: #1 \color_group_end: }
0049 }
0050 \cs_new_protected:Npn \hbox_overlap_center:n #1
0051 { \hbox_to_zero:n { \tex_hss:D #1 \tex_hss:D } }
0052 \cs_new_protected:Npn \hbox_overlap_left:n #1
0053 { \hbox_to_zero:n { \tex_hss:D #1 } }
0054 \cs_new_protected:Npn \hbox_overlap_right:n #1
0055 { \hbox_to_zero:n { #1 \tex_hss:D } }
0056 \cs_new_eq:NN \hbox_unpack:N \tex_unhcopy:D
0057 \cs_new_eq:NN \hbox_unpack_drop:N \tex_unhbox:D
0058 \cs_generate_variant:Nn \hbox_unpack:N { c }
0059 \cs_generate_variant:Nn \hbox_unpack_drop:N { c }
0060 \cs_new_protected:Npn \vbox:n #1
0061 { \tex_vbox:D { \color_group_begin: #1 \par \color_group_end: } }
0062 \cs_new_protected:Npn \vbox_top:n #1
0063 { \tex_vtop:D { \color_group_begin: #1 \par \color_group_end: } }
0064 \cs_new_protected:Npn \vbox_to_ht:nn #1#2
0065 {
0066     \tex_vbox:D to \__box_dim_eval:n {#1}

```

```
0067 { \color_group_begin: #2 \par \color_group_end: } 3006
0068 } 3006
0069 \cs_new_protected:Npn \vbox_to_zero:n #1 3006
0070 { 3007
0071 \tex_vbox:D to \c_zero_dim 3007
0072 { \color_group_begin: #1 \par \color_group_end: } 3007
0073 } 3007
0074 \cs_new_protected:Npn \vbox_set:Nn #1#2 3007
0075 { 3007
0076 \tex_setbox:D #1 \tex_vbox:D 3007
0077 { \color_group_begin: #2 \par \color_group_end: } 3007
0078 } 3007
0079 \cs_new_protected:Npn \vbox_gset:Nn #1#2 3007
0080 { 3008
0081 \tex_global:D \tex_setbox:D #1 \tex_vbox:D 3008
0082 { \color_group_begin: #2 \par \color_group_end: } 3008
0083 } 3008
0084 \cs_generate_variant:Nn \vbox_set:Nn { c } 3008
0085 \cs_generate_variant:Nn \vbox_gset:Nn { c } 3008
0086 \cs_new_protected:Npn \vbox_set_top:Nn #1#2 3008
0087 { 3008
0088 \tex_setbox:D #1 \tex_vtop:D 3008
0089 { \color_group_begin: #2 \par \color_group_end: } 3008
0090 } 3009
0091 \cs_new_protected:Npn \vbox_gset_top:Nn #1#2 3009
0092 { 3009
0093 \tex_global:D \tex_setbox:D #1 \tex_vtop:D 3009
0094 { \color_group_begin: #2 \par \color_group_end: } 3009
0095 } 3009
0096 \cs_generate_variant:Nn \vbox_set_top:Nn { c } 3009
0097 \cs_generate_variant:Nn \vbox_gset_top:Nn { c } 3009
0098 \cs_new_protected:Npn \vbox_set_to_ht:Nnn #1#2#3 3009
0099 { 3009
0100 \tex_setbox:D #1 \tex_vbox:D to \__box_dim_eval:n {#2} 3010
0101 { \color_group_begin: #3 \par \color_group_end: } 3010
0102 } 3010
0103 \cs_new_protected:Npn \vbox_gset_to_ht:Nnn #1#2#3 3010
0104 { 3010
0105 \tex_global:D \tex_setbox:D #1 \tex_vbox:D to \__box_dim_eval:n {#2} 3010
0106 { \color_group_begin: #3 \par \color_group_end: } 3010
0107 } 3010
0108 \cs_generate_variant:Nn \vbox_set_to_ht:Nnn { c } 3010
0109 \cs_generate_variant:Nn \vbox_gset_to_ht:Nnn { c } 3010
0110 \cs_new_protected:Npn \vbox_set:Nw #1 3011
0111 { 3011
0112 \tex_setbox:D #1 \tex_vbox:D 3011
```

```
0113 \c_group_begin_token 3011
0114 \color_group_begin: 3011
0115 } 3011
0116 \cs_new_protected:Npn \vbox_gset:Nw #1 3011
0117 { 3011
0118 \tex_global:D \tex_setbox:D #1 \tex_vbox:D 3011
0119 \c_group_begin_token 3011
0120 \color_group_begin: 3012
0121 } 3012
0122 \cs_generate_variant:Nn \vbox_set:Nw { c } 3012
0123 \cs_generate_variant:Nn \vbox_gset:Nw { c } 3012
0124 \cs_new_protected:Npn \vbox_set_end: 3012
0125 { 3012
0126 \par 3012
0127 \color_group_end: 3012
0128 \c_group_end_token 3012
0129 } 3012
0130 \cs_new_eq:NN \vbox_gset_end: \vbox_set_end: 3013
0131 \cs_new_protected:Npn \vbox_set_to_ht:Nnw #1#2 3013
0132 { 3013
0133 \tex_setbox:D #1 \tex_vbox:D to \__box_dim_eval:n {#2} 3013
0134 \c_group_begin_token 3013
0135 \color_group_begin: 3013
0136 } 3013
0137 \cs_new_protected:Npn \vbox_gset_to_ht:Nnw #1#2 3013
0138 { 3013
0139 \tex_global:D \tex_setbox:D #1 \tex_vbox:D to \__box_dim_eval:n {#2} 3013
0140 \c_group_begin_token 3014
0141 \color_group_begin: 3014
0142 } 3014
0143 \cs_generate_variant:Nn \vbox_set_to_ht:Nnw { c } 3014
0144 \cs_generate_variant:Nn \vbox_gset_to_ht:Nnw { c } 3014
0145 \cs_new_eq:NN \vbox_unpack:N \tex_unvcopy:D 3014
0146 \cs_new_eq:NN \vbox_unpack_drop:N \tex_unvbox:D 3014
0147 \cs_generate_variant:Nn \vbox_unpack:N { c } 3014
0148 \cs_generate_variant:Nn \vbox_unpack_drop:N { c } 3014
0149 \cs_new_protected:Npn \vbox_set_split_to_ht:NNn #1#2#3 3014
0150 { \tex_setbox:D #1 \tex_vsplit:D #2 to \__box_dim_eval:n {#3} } 3015
0151 \cs_generate_variant:Nn \vbox_set_split_to_ht:NNn { c , Nc , cc } 3015
0152 \cs_new_protected:Npn \vbox_gset_split_to_ht:NNn #1#2#3 3015
0153 { 3015
0154 \tex_global:D \tex_setbox:D #1 3015
0155 \tex_vsplit:D #2 to \__box_dim_eval:n {#3} 3015
0156 } 3015
0157 \cs_generate_variant:Nn \vbox_gset_split_to_ht:NNn { c , Nc , cc } 3015
0158 \fp_new:N \l__box_angle_fp 3015
```



```

0159 \fp_new:N \l__box_cos_fp
0160 \fp_new:N \l__box_sin_fp
0161 \dim_new:N \l__box_top_dim
0162 \dim_new:N \l__box_bottom_dim
0163 \dim_new:N \l__box_left_dim
0164 \dim_new:N \l__box_right_dim
0165 \dim_new:N \l__box_top_new_dim
0166 \dim_new:N \l__box_bottom_new_dim
0167 \dim_new:N \l__box_left_new_dim
0168 \dim_new:N \l__box_right_new_dim
0169 \box_new:N \l__box_internal_box
0170 \cs_new_protected:Npn \box_rotate:Nn #1#2
0171 { \__box_rotate:NnN #1 {#2} \hbox_set:Nn }
0172 \cs_generate_variant:Nn \box_rotate:Nn { c }
0173 \cs_new_protected:Npn \box_grotate:Nn #1#2
0174 { \__box_rotate:NnN #1 {#2} \hbox_gset:Nn }
0175 \cs_generate_variant:Nn \box_grotate:Nn { c }
0176 \cs_new_protected:Npn \__box_rotate:NnN #1#2#3
0177 {
0178     #3 #1
0179     {
0180         \fp_set:Nn \l__box_angle_fp {#2}
0181         \fp_set:Nn \l__box_sin_fp { sind ( \l__box_angle_fp ) }
0182         \fp_set:Nn \l__box_cos_fp { cosd ( \l__box_angle_fp ) }
0183         \__box_rotate:N #1
0184     }
0185 }
0186 \cs_new_protected:Npn \__box_rotate:N #1
0187 {
0188     \dim_set:Nn \l__box_top_dim { \box_ht:N #1 }
0189     \dim_set:Nn \l__box_bottom_dim { -\box_dp:N #1 }
0190     \dim_set:Nn \l__box_right_dim { \box_wd:N #1 }
0191     \dim_zero:N \l__box_left_dim
0192     \fp_compare:nNnTF \l__box_sin_fp > \c_zero_fp
0193     {
0194         \fp_compare:nNnTF \l__box_cos_fp > \c_zero_fp
0195         { \__box_rotate_quadrant_one: }
0196         { \__box_rotate_quadrant_two: }
0197     }
0198     {
0199         \fp_compare:nNnTF \l__box_cos_fp < \c_zero_fp
0200         { \__box_rotate_quadrant_three: }
0201         { \__box_rotate_quadrant_four: }
0202     }
0203     \hbox_set:Nn \l__box_internal_box { \box_use:N #1 }
0204     \hbox_set:Nn \l__box_internal_box

```



```

0205 {
0206     \__kernel_kern:n { -\l__box_left_new_dim }
0207     \hbox:n
0208     {
0209         \__box_backend_rotate:Nn
0210         \l__box_internal_box
0211         \l__box_angle_fp
0212     }
0213 }
0214 \box_set_ht:Nn \l__box_internal_box { \l__box_top_new_dim }
0215 \box_set_dp:Nn \l__box_internal_box { -\l__box_bottom_new_dim }
0216 \box_set_wd:Nn \l__box_internal_box
0217 { \l__box_right_new_dim - \l__box_left_new_dim }
0218 \box_use_drop:N \l__box_internal_box
0219 }
0220 \cs_new_protected:Npn \__box_rotate_xdir:nnN #1#2#3
0221 {
0222     \dim_set:Nn #3
0223     {
0224         \fp_to_dim:n
0225         {
0226             \l__box_cos_fp * \dim_to_fp:n {#1}
0227             - \l__box_sin_fp * \dim_to_fp:n {#2}
0228         }
0229     }
0230 }
0231 \cs_new_protected:Npn \__box_rotate_ydir:nnN #1#2#3
0232 {
0233     \dim_set:Nn #3
0234     {
0235         \fp_to_dim:n
0236         {
0237             \l__box_sin_fp * \dim_to_fp:n {#1}
0238             + \l__box_cos_fp * \dim_to_fp:n {#2}
0239         }
0240     }
0241 }
0242 \cs_new_protected:Npn \__box_rotate_quadrant_one:
0243 {
0244     \__box_rotate_ydir:nnN \l__box_right_dim \l__box_top_dim
0245     \l__box_top_new_dim
0246     \__box_rotate_ydir:nnN \l__box_left_dim \l__box_bottom_dim
0247     \l__box_bottom_new_dim
0248     \__box_rotate_xdir:nnN \l__box_left_dim \l__box_top_dim
0249     \l__box_left_new_dim
0250     \__box_rotate_xdir:nnN \l__box_right_dim \l__box_bottom_dim

```

```
0251     \l__box_right_new_dim
0252 }
0253 \cs_new_protected:Npn \__box_rotate_quadrant_two:
0254 {
0255     \__box_rotate_ydir:nnN \l__box_right_dim \l__box_bottom_dim
0256     \l__box_top_new_dim
0257     \__box_rotate_ydir:nnN \l__box_left_dim \l__box_top_dim
0258     \l__box_bottom_new_dim
0259     \__box_rotate_xdir:nnN \l__box_right_dim \l__box_top_dim
0260     \l__box_left_new_dim
0261     \__box_rotate_xdir:nnN \l__box_left_dim \l__box_bottom_dim
0262     \l__box_right_new_dim
0263 }
0264 \cs_new_protected:Npn \__box_rotate_quadrant_three:
0265 {
0266     \__box_rotate_ydir:nnN \l__box_left_dim \l__box_bottom_dim
0267     \l__box_top_new_dim
0268     \__box_rotate_ydir:nnN \l__box_right_dim \l__box_top_dim
0269     \l__box_bottom_new_dim
0270     \__box_rotate_xdir:nnN \l__box_right_dim \l__box_bottom_dim
0271     \l__box_left_new_dim
0272     \__box_rotate_xdir:nnN \l__box_left_dim \l__box_top_dim
0273     \l__box_right_new_dim
0274 }
0275 \cs_new_protected:Npn \__box_rotate_quadrant_four:
0276 {
0277     \__box_rotate_ydir:nnN \l__box_left_dim \l__box_top_dim
0278     \l__box_top_new_dim
0279     \__box_rotate_ydir:nnN \l__box_right_dim \l__box_bottom_dim
0280     \l__box_bottom_new_dim
0281     \__box_rotate_xdir:nnN \l__box_left_dim \l__box_bottom_dim
0282     \l__box_left_new_dim
0283     \__box_rotate_xdir:nnN \l__box_right_dim \l__box_top_dim
0284     \l__box_right_new_dim
0285 }
0286 \fp_new:N \l__box_scale_x_fp
0287 \fp_new:N \l__box_scale_y_fp
0288 \cs_new_protected:Npn \box_resize_to_wd_and_ht_plus_dp:Nnn #1#2#3
0289 {
0290     \__box_resize_to_wd_and_ht_plus_dp:NnnN #1 {#2} {#3}
0291     \hbox_set:Nn
0292 }
0293 \cs_generate_variant:Nn \box_resize_to_wd_and_ht_plus_dp:Nnn { c }
0294 \cs_new_protected:Npn \box_gresize_to_wd_and_ht_plus_dp:Nnn #1#2#3
0295 {
0296     \__box_resize_to_wd_and_ht_plus_dp:NnnN #1 {#2} {#3}
```

```

0297 \hbox_gset:Nn
0298 }
0299 \cs_generate_variant:Nn \box_gresize_to_wd_and_ht_plus_dp:Nnn { c }
0300 \cs_new_protected:Npn \__box_resize_to_wd_and_ht_plus_dp:NnnN #1#2#3#4
0301 {
0302   #4 #1
0303   {
0304     \__box_resize_set_corners:N #1
0305     \fp_set:Nn \l__box_scale_x_fp
0306       { \dim_to_fp:n {#2} / \dim_to_fp:n { \l__box_right_dim } }
0307     \fp_set:Nn \l__box_scale_y_fp
0308       {
0309         \dim_to_fp:n {#3}
0310         / \dim_to_fp:n { \l__box_top_dim - \l__box_bottom_dim }
0311       }
0312     \__box_resize:N #1
0313   }
0314 }
0315 \cs_new_protected:Npn \__box_resize_set_corners:N #1
0316 {
0317   \dim_set:Nn \l__box_top_dim { \box_ht:N #1 }
0318   \dim_set:Nn \l__box_bottom_dim { -\box_dp:N #1 }
0319   \dim_set:Nn \l__box_right_dim { \box_wd:N #1 }
0320   \dim_zero:N \l__box_left_dim
0321 }
0322 \cs_new_protected:Npn \__box_resize:N #1
0323 {
0324   \__box_resize:NNN \l__box_right_new_dim
0325     \l__box_scale_x_fp \l__box_right_dim
0326   \__box_resize:NNN \l__box_bottom_new_dim
0327     \l__box_scale_y_fp \l__box_bottom_dim
0328   \__box_resize:NNN \l__box_top_new_dim
0329     \l__box_scale_y_fp \l__box_top_dim
0330   \__box_resize_common:N #1
0331 }
0332 \cs_new_protected:Npn \__box_resize:NNN #1#2#3
0333 {
0334   \dim_set:Nn #1
0335     { \fp_to_dim:n { \fp_abs:n { #2 } * \dim_to_fp:n { #3 } } }
0336 }
0337 \cs_new_protected:Npn \box_resize_to_ht:Nn #1#2
0338 { \__box_resize_to_ht:NnN #1 {#2} \hbox_set:Nn }
0339 \cs_generate_variant:Nn \box_resize_to_ht:Nn { c }
0340 \cs_new_protected:Npn \box_gresize_to_ht:Nn #1#2
0341 { \__box_resize_to_ht:NnN #1 {#2} \hbox_gset:Nn }
0342 \cs_generate_variant:Nn \box_gresize_to_ht:Nn { c }

```

```
0343 \cs_new_protected:Npn \__box_resize_to_ht:NnN #1#2#3
0344 {
0345     #3 #1
0346     {
0347         \__box_resize_set_corners:N #1
0348         \fp_set:Nn \l__box_scale_y_fp
0349         {
0350             \dim_to_fp:n {#2}
0351             / \dim_to_fp:n { \l__box_top_dim }
0352         }
0353         \fp_set_eq:NN \l__box_scale_x_fp \l__box_scale_y_fp
0354         \__box_resize:N #1
0355     }
0356 }
0357 \cs_new_protected:Npn \box_resize_to_ht_plus_dp:Nn #1#2
0358 { \__box_resize_to_ht_plus_dp:NnN #1 {#2} \hbox_set:Nn }
0359 \cs_generate_variant:Nn \box_resize_to_ht_plus_dp:Nn { c }
0360 \cs_new_protected:Npn \box_gresize_to_ht_plus_dp:Nn #1#2
0361 { \__box_resize_to_ht_plus_dp:NnN #1 {#2} \hbox_gset:Nn }
0362 \cs_generate_variant:Nn \box_gresize_to_ht_plus_dp:Nn { c }
0363 \cs_new_protected:Npn \__box_resize_to_ht_plus_dp:NnN #1#2#3
0364 {
0365     #3 #1
0366     {
0367         \__box_resize_set_corners:N #1
0368         \fp_set:Nn \l__box_scale_y_fp
0369         {
0370             \dim_to_fp:n {#2}
0371             / \dim_to_fp:n { \l__box_top_dim - \l__box_bottom_dim }
0372         }
0373         \fp_set_eq:NN \l__box_scale_x_fp \l__box_scale_y_fp
0374         \__box_resize:N #1
0375     }
0376 }
0377 \cs_new_protected:Npn \box_resize_to_wd:Nn #1#2
0378 { \__box_resize_to_wd:NnN #1 {#2} \hbox_set:Nn }
0379 \cs_generate_variant:Nn \box_resize_to_wd:Nn { c }
0380 \cs_new_protected:Npn \box_gresize_to_wd:Nn #1#2
0381 { \__box_resize_to_wd:NnN #1 {#2} \hbox_gset:Nn }
0382 \cs_generate_variant:Nn \box_gresize_to_wd:Nn { c }
0383 \cs_new_protected:Npn \__box_resize_to_wd:NnN #1#2#3
0384 {
0385     #3 #1
0386     {
0387         \__box_resize_set_corners:N #1
0388         \fp_set:Nn \l__box_scale_x_fp
```

```
0389 { \dim_to_fp:n {#2} / \dim_to_fp:n { \l__box_right_dim } } 30389
0390 \fp_set_eq:NN \l__box_scale_y_fp \l__box_scale_x_fp 30390
0391 \__box_resize:N #1 30391
0392 } 30392
0393 } 30393
0394 \cs_new_protected:Npn \box_resize_to_wd_and_ht:Nnn #1#2#3 30394
0395 { \__box_resize_to_wd_and_ht:NnnN #1 {#2} {#3} \hbox_set:Nn } 30395
0396 \cs_generate_variant:Nn \box_resize_to_wd_and_ht:Nnn { c } 30396
0397 \cs_new_protected:Npn \box_gresize_to_wd_and_ht:Nnn #1#2#3 30397
0398 { \__box_resize_to_wd_and_ht:NnnN #1 {#2} {#3} \hbox_gset:Nn } 30398
0399 \cs_generate_variant:Nn \box_gresize_to_wd_and_ht:Nnn { c } 30399
0400 \cs_new_protected:Npn \__box_resize_to_wd_and_ht:NnnN #1#2#3#4 30400
0401 { 30401
0402 #4 #1 30402
0403 { 30403
0404 \__box_resize_set_corners:N #1 30404
0405 \fp_set:Nn \l__box_scale_x_fp 30405
0406 { \dim_to_fp:n {#2} / \dim_to_fp:n { \l__box_right_dim } } 30406
0407 \fp_set:Nn \l__box_scale_y_fp 30407
0408 { 30408
0409 \dim_to_fp:n {#3} 30409
0410 / \dim_to_fp:n { \l__box_top_dim } 30410
0411 } 30411
0412 \__box_resize:N #1 30412
0413 } 30413
0414 } 30414
0415 \cs_new_protected:Npn \box_scale:Nnn #1#2#3 30415
0416 { \__box_scale:NnnN #1 {#2} {#3} \hbox_set:Nn } 30416
0417 \cs_generate_variant:Nn \box_scale:Nnn { c } 30417
0418 \cs_new_protected:Npn \box_gscale:Nnn #1#2#3 30418
0419 { \__box_scale:NnnN #1 {#2} {#3} \hbox_gset:Nn } 30419
0420 \cs_generate_variant:Nn \box_gscale:Nnn { c } 30420
0421 \cs_new_protected:Npn \__box_scale:NnnN #1#2#3#4 30421
0422 { 30422
0423 #4 #1 30423
0424 { 30424
0425 \fp_set:Nn \l__box_scale_x_fp {#2} 30425
0426 \fp_set:Nn \l__box_scale_y_fp {#3} 30426
0427 \__box_scale:N #1 30427
0428 } 30428
0429 } 30429
0430 \cs_new_protected:Npn \__box_scale:N #1 30430
0431 { 30431
0432 \dim_set:Nn \l__box_top_dim { \box_ht:N #1 } 30432
0433 \dim_set:Nn \l__box_bottom_dim { -\box_dp:N #1 } 30433
0434 \dim_set:Nn \l__box_right_dim { \box_wd:N #1 } 30434
```

```
0435 \dim_zero:N \l__box_left_dim
0436 \dim_set:Nn \l__box_top_new_dim
0437 { \fp_abs:n { \l__box_scale_y_fp } \l__box_top_dim }
0438 \dim_set:Nn \l__box_bottom_new_dim
0439 { \fp_abs:n { \l__box_scale_y_fp } \l__box_bottom_dim }
0440 \dim_set:Nn \l__box_right_new_dim
0441 { \fp_abs:n { \l__box_scale_x_fp } \l__box_right_dim }
0442 \__box_resize_common:N #1
0443 }
0444 \cs_new_protected:Npn \box_autosize_to_wd_and_ht:Nnn #1#2#3
0445 { \__box_autosize:NnnnN #1 {#2} {#3} { \box_ht:N #1 } \hbox_set:Nn }
0446 \cs_generate_variant:Nn \box_autosize_to_wd_and_ht:Nnn { c }
0447 \cs_new_protected:Npn \box_gautosize_to_wd_and_ht:Nnn #1#2#3
0448 { \__box_autosize:NnnnN #1 {#2} {#3} { \box_ht:N #1 } \hbox_gset:Nn }
0449 \cs_generate_variant:Nn \box_gautosize_to_wd_and_ht:Nnn { c }
0450 \cs_new_protected:Npn \box_autosize_to_wd_and_ht_plus_dp:Nnn #1#2#3
0451 {
0452   \__box_autosize:NnnnN #1 {#2} {#3} { \box_ht:N #1 + \box_dp:N #1 }
0453   \hbox_set:Nn
0454 }
0455 \cs_generate_variant:Nn \box_autosize_to_wd_and_ht_plus_dp:Nnn { c }
0456 \cs_new_protected:Npn \box_gautosize_to_wd_and_ht_plus_dp:Nnn #1#2#3
0457 {
0458   \__box_autosize:NnnnN #1 {#2} {#3} { \box_ht:N #1 + \box_dp:N #1 }
0459   \hbox_gset:Nn
0460 }
0461 \cs_generate_variant:Nn \box_gautosize_to_wd_and_ht_plus_dp:Nnn { c }
0462 \cs_new_protected:Npn \__box_autosize:NnnnN #1#2#3#4#5
0463 {
0464   #5 #1
0465   {
0466     \fp_set:Nn \l__box_scale_x_fp { ( \dim_to_fp:n {#2} ) / \box_wd:N #1 }
0467     \fp_set:Nn \l__box_scale_y_fp
0468       { ( \dim_to_fp:n {#3} ) / ( \dim_to_fp:n {#4} ) }
0469     \fp_compare:nNnTF \l__box_scale_x_fp > \l__box_scale_y_fp
0470       { \fp_set_eq:NN \l__box_scale_x_fp \l__box_scale_y_fp }
0471       { \fp_set_eq:NN \l__box_scale_y_fp \l__box_scale_x_fp }
0472     \__box_scale:N #1
0473   }
0474 }
0475 \cs_new_protected:Npn \__box_resize_common:N #1
0476 {
0477   \hbox_set:Nn \l__box_internal_box
0478   {
0479     \__box_backend_scale:Nnn
0480     #1
```

```
0481         \l__box_scale_x_fp
0482         \l__box_scale_y_fp
0483     }
0484 \fp_compare:nNnTF \l__box_scale_y_fp > \c_zero_fp
0485 {
0486     \box_set_ht:Nn \l__box_internal_box { \l__box_top_new_dim }
0487     \box_set_dp:Nn \l__box_internal_box { -\l__box_bottom_new_dim }
0488 }
0489 {
0490     \box_set_dp:Nn \l__box_internal_box { \l__box_top_new_dim }
0491     \box_set_ht:Nn \l__box_internal_box { -\l__box_bottom_new_dim }
0492 }
0493 \fp_compare:nNnTF \l__box_scale_x_fp < \c_zero_fp
0494 {
0495     \hbox_to_wd:nn { \l__box_right_new_dim }
0496     {
0497         \__kernel_kern:n { \l__box_right_new_dim }
0498         \box_use_drop:N \l__box_internal_box
0499         \tex_hss:D
0500     }
0501 }
0502 {
0503     \box_set_wd:Nn \l__box_internal_box { \l__box_right_new_dim }
0504     \hbox:n
0505     {
0506         \__kernel_kern:n { Opt }
0507         \box_use_drop:N \l__box_internal_box
0508         \tex_hss:D
0509     }
0510 }
0511 }
0512 \cs_new_protected:Npn \box_set_clipped:N #1
0513 { \hbox_set:Nn #1 { \__box_backend_clip:N #1 } }
0514 \cs_generate_variant:Nn \box_set_clipped:N { c }
0515 \cs_new_protected:Npn \box_gset_clipped:N #1
0516 { \hbox_gset:Nn #1 { \__box_backend_clip:N #1 } }
0517 \cs_generate_variant:Nn \box_gset_clipped:N { c }
0518 \cs_new_protected:Npn \box_set_trim:Nnnnn #1#2#3#4#5
0519 { \__box_set_trim:NnnnnN #1 {#2} {#3} {#4} {#5} \box_set_eq:NN }
0520 \cs_generate_variant:Nn \box_set_trim:Nnnnn { c }
0521 \cs_new_protected:Npn \box_gset_trim:Nnnnn #1#2#3#4#5
0522 { \__box_set_trim:NnnnnN #1 {#2} {#3} {#4} {#5} \box_gset_eq:NN }
0523 \cs_generate_variant:Nn \box_gset_trim:Nnnnn { c }
0524 \cs_new_protected:Npn \__box_set_trim:NnnnnN #1#2#3#4#5#6
0525 {
0526     \hbox_set:Nn \l__box_internal_box
```



```
0527 {
0528     \__kernel_kern:n { -#2 }
0529     \box_use:N #1
0530     \__kernel_kern:n { -#4 }
0531 }
0532 \dim_compare:nNnTF { \box_dp:N #1 } > {#3}
0533 {
0534     \hbox_set:Nn \l__box_internal_box
0535     {
0536         \box_move_down:nn \c_zero_dim
0537         { \box_use_drop:N \l__box_internal_box }
0538     }
0539     \box_set_dp:Nn \l__box_internal_box { \box_dp:N #1 - (#3) }
0540 }
0541 {
0542     \hbox_set:Nn \l__box_internal_box
0543     {
0544         \box_move_down:nn { (#3) - \box_dp:N #1 }
0545         { \box_use_drop:N \l__box_internal_box }
0546     }
0547     \box_set_dp:Nn \l__box_internal_box \c_zero_dim
0548 }
0549 \dim_compare:nNnTF { \box_ht:N \l__box_internal_box } > {#5}
0550 {
0551     \hbox_set:Nn \l__box_internal_box
0552     {
0553         \box_move_up:nn \c_zero_dim
0554         { \box_use_drop:N \l__box_internal_box }
0555     }
0556     \box_set_ht:Nn \l__box_internal_box
0557     { \box_ht:N \l__box_internal_box - (#5) }
0558 }
0559 {
0560     \hbox_set:Nn \l__box_internal_box
0561     {
0562         \box_move_up:nn { (#5) - \box_ht:N \l__box_internal_box }
0563         { \box_use_drop:N \l__box_internal_box }
0564     }
0565     \box_set_ht:Nn \l__box_internal_box \c_zero_dim
0566 }
0567 #6 #1 \l__box_internal_box
0568 }
0569 \cs_new_protected:Npn \box_set_viewport:Nnnnn #1#2#3#4#5
0570 { \__box_set_viewport:NnnnnN #1 {#2} {#3} {#4} {#5} \box_set_eq:NN }
0571 \cs_generate_variant:Nn \box_set_viewport:Nnnnn { c }
0572 \cs_new_protected:Npn \box_gset_viewport:Nnnnn #1#2#3#4#5
```

```
0573 { \_box_set_viewport:NnnnnN #1 {#2} {#3} {#4} {#5} \box_gset_eq:NN } 30573
0574 \cs_generate_variant:Nn \box_gset_viewport:Nnnnnn { c } 30574
0575 \cs_new_protected:Npn \_box_set_viewport:NnnnnN #1#2#3#4#5#6 30575
0576 { 30576
0577   \hbox_set:Nn \l__box_internal_box 30577
0578   { 30578
0579     \__kernel_kern:n { -#2 } 30579
0580     \box_use:N #1 30580
0581     \__kernel_kern:n { #4 - \box_wd:N #1 } 30581
0582   } 30582
0583 \dim_compare:nNnTF {#3} < \c_zero_dim 30583
0584 { 30584
0585   \hbox_set:Nn \l__box_internal_box 30585
0586   { 30586
0587     \box_move_down:nn \c_zero_dim 30587
0588     { \box_use_drop:N \l__box_internal_box } 30588
0589   } 30589
0590   \box_set_dp:Nn \l__box_internal_box { - \__box_dim_eval:n {#3} } 30590
0591 } 30591
0592 { 30592
0593   \hbox_set:Nn \l__box_internal_box 30593
0594   { \box_move_down:nn {#3} { \box_use_drop:N \l__box_internal_box } } 30594
0595   \box_set_dp:Nn \l__box_internal_box \c_zero_dim 30595
0596 } 30596
0597 \dim_compare:nNnTF {#5} > \c_zero_dim 30597
0598 { 30598
0599   \hbox_set:Nn \l__box_internal_box 30599
0600   { 30600
0601     \box_move_up:nn \c_zero_dim 30601
0602     { \box_use_drop:N \l__box_internal_box } 30602
0603   } 30603
0604   \box_set_ht:Nn \l__box_internal_box 30604
0605   { 30605
0606     (#5) 30606
0607     \dim_compare:nNnT {#3} > \c_zero_dim 30607
0608     { - (#3) } 30608
0609   } 30609
0610 } 30610
0611 { 30611
0612   \hbox_set:Nn \l__box_internal_box 30612
0613   { 30613
0614     \box_move_up:nn { - \__box_dim_eval:n {#5} } 30614
0615     { \box_use_drop:N \l__box_internal_box } 30615
0616   } 30616
0617   \box_set_ht:Nn \l__box_internal_box \c_zero_dim 30617
0618 } 30618
```

```
0619 #6 #1 \l__box_internal_box 30619
0620 } 30620
0621 %% File: l3color.dtx 30621
0622 \cs_new_eq:NN \color_group_begin: \group_begin: 30622
0623 \cs_new_eq:NN \color_group_end: \group_end: 30623
0624 \cs_new_protected:Npn \color_ensure_current: 30624
0625 { \__color_select:N \l__color_current_tl } 30625
0626 \scan_new:N \s__color_stop 30626
0627 \cs_new_protected:Npn \__color_select:N #1 30627
0628 { 30628
0629 \exp_after:wN \__color_select:nn #1 30629
0630 \group_insert_after:N \__color_backend_reset: 30630
0631 } 30631
0632 \cs_new_protected:Npn \__color_select_math:N #1 30632
0633 { \exp_after:wN \__color_select:nn #1 } 30633
0634 \cs_new_protected:Npn \__color_select:nn #1#2 30634
0635 { \use:c { __color_backend_select_ #1 :n } {#2} } 30635
0636 \tl_new:N \l__color_current_tl 30636
0637 \tl_set:Nn \l__color_current_tl { { gray } { 0 } } 30637
0638 \int_new:N \l__color_internal_int 30638
0639 \tl_new:N \l__color_internal_tl 30639
0640 \scan_new:N \s__color_mark 30640
0641 \bool_new:N \l__color_ignore_error_bool 30641
0642 \prg_new_conditional:Npnn \color_if_exist:n #1 { p , T, F, TF } 30642
0643 { 30643
0644 \prop_if_exist:cTF { l__color_named_ #1 _prop } 30644
0645 { 30645
0646 \prop_if_empty:cTF { l__color_named_ #1 _prop } 30646
0647 \prg_return_false: 30647
0648 \prg_return_true: 30648
0649 } 30649
0650 \prg_return_false: 30650
0651 } 30651
0652 \cs_new:Npn \__color_model:N #1 { \exp_after:wN \use_i:nn #1 } 30652
0653 \cs_new:Npn \__color_values:N #1 { \exp_after:wN \use_ii:nn #1 } 30653
0654 \cs_new_protected:Npn \__color_extract:nNN #1#2#3 30654
0655 { 30655
0656 \tl_set_eq:Nc #2 { l__color_named_ #1 _tl } 30656
0657 \prop_get:cVN { l__color_named_ #1 _prop } #2 #3 30657
0658 } 30658
0659 \cs_generate_variant:Nn \__color_extract:nNN { V } 30659
0660 \cs_new_protected:Npn \__color_convert:nnN #1#2#3 30660
0661 { \__color_convert:nnVN {#1} {#2} #3 #3 } 30661
0662 \cs_generate_variant:Nn \__color_convert:nnN { VV } 30662
0663 \cs_generate_variant:Nn \exp_last_unbraced:Nf { c } 30663
0664 \cs_new_protected:Npn \__color_convert:nnnN #1#2#3#4 30664
```

```

0665 {
0666     \tl_set:Nx #4
0667     {
0668         \cs_if_exist_use:cTF { __color_convert_ #1 _ #2 :w }
0669         { #3 \s__color_stop }
0670         {
0671             \cs_if_exist:cTF { __color_convert_ \use:c { c__color_fallback_ #1 _tl } _
0672             #2 :w }
0673             {
0674                 \exp_last_unbraced:cf
0675                 { __color_convert_ \use:c { c__color_fallback_ #1 _tl } _ #2 :w }
0676                 { \use:c { __color_convert_ #1 _ \use:c { c__color_fallback_ #1 _tl }
0677                 :w } #3 \s__color_stop }
0678                 \s__color_stop
0679             }
0680             {
0681                 \exp_last_unbraced:cf
0682                 { __color_convert_ \use:c { c__color_fallback_ #2 _tl } _ #2 :w }
0683                 {
0684                     \cs_if_exist_use:cTF { __color_convert_ #1 _ \use:c {
0685                     c__color_fallback_ #2 _tl } :w }
0686                     { #3 \s__color_stop }
0687                     {
0688                         \exp_last_unbraced:cf
0689                         { __color_convert_ \use:c { c__color_fallback_ #1 _tl } _
0690                         \use:c { c__color_fallback_ #2 _tl } :w }
0691                         { \use:c { __color_convert_ #1 _ \use:c { c__color_fallback_
0692                         #1 _tl } :w } #3 \s__color_stop }
0693                         \s__color_stop
0694                     }
0695                 }
0696             }
0697         }
0698     }
0699 }
0700 \cs_generate_variant:Nn \__color_convert:nnnN { nV , nnV }
0701 \cs_new:Npn \__color_convert_gray_gray:w #1 \s__color_stop
0702 { #1 }
0703 \cs_new:Npn \__color_convert_gray_rgb:w #1 \s__color_stop
0704 { #1 ~ #1 ~ #1 }
0705 \cs_new:Npn \__color_convert_gray_cmyk:w #1 \s__color_stop
0706 { 0 ~ 0 ~ 0 ~ \fp_eval:n { 1 - #1 } }
0707 \cs_new:Npn \__color_convert_rgb_gray:w #1 ~ #2 ~ #3 \s__color_stop
0708 { \fp_eval:n { 0.3 * #1 + 0.59 * #2 + 0.11 * #3 } }
0709 \cs_new:Npn \__color_convert_rgb_rgb:w #1 \s__color_stop

```

```
0706 { #1 }
0707 \cs_new:Npn \__color_convert_rgb_cmyk:w #1 ~ #2 ~ #3 \s__color_stop
0708 {
0709   \exp_args:Neee \__color_convert_rgb_cmyk:nnn
0710   { \fp_eval:n { 1 - #1 } }
0711   { \fp_eval:n { 1 - #2 } }
0712   { \fp_eval:n { 1 - #3 } }
0713 }
0714 \cs_new:Npn \__color_convert_rgb_cmyk:nnn #1#2#3
0715 {
0716   \exp_args:Ne \__color_convert_rgb_cmyk:nnnn
0717   { \fp_eval:n { min( #1, #2 , #3 ) } } {#1} {#2} {#3}
0718 }
0719 \cs_new:Npn \__color_convert_rgb_cmyk:nnnn #1#2#3#4
0720 {
0721   \fp_eval:n { min ( 1 , max ( 0 , #2 - #1 ) ) } \c_space_tl
0722   \fp_eval:n { min ( 1 , max ( 0 , #3 - #1 ) ) } \c_space_tl
0723   \fp_eval:n { min ( 1 , max ( 0 , #4 - #1 ) ) } \c_space_tl
0724   #1
0725 }
0726 \cs_new:Npn \__color_convert_cmyk_gray:w #1 ~ #2 ~ #3 ~ #4 \s__color_stop
0727 { \fp_eval:n { 1 - min ( 1 , 0.3 * #1 + 0.59 * #2 + 0.11 * #3 + #4 ) } }
0728 \cs_new:Npn \__color_convert_cmyk_rgb:w #1 ~ #2 ~ #3 ~ #4 \s__color_stop
0729 {
0730   \fp_eval:n { 1 - min ( 1 , #1 + #4 ) } \c_space_tl
0731   \fp_eval:n { 1 - min ( 1 , #2 + #4 ) } \c_space_tl
0732   \fp_eval:n { 1 - min ( 1 , #3 + #4 ) }
0733 }
0734 \cs_new:Npn \__color_convert_cmyk_cmyk:w #1 \s__color_stop
0735 { #1 }
0736 \tl_new:N \l__color_model_tl
0737 \tl_new:N \l__color_value_tl
0738 \tl_new:N \l__color_next_model_tl
0739 \tl_new:N \l__color_next_value_tl
0740 \cs_new_protected:Npe \__color_parse:nN #1#2
0741 {
0742   \tl_set:Ne \exp_not:c { l__color_named_ . _tl }
0743   { \exp_not:N \__color_model:N \exp_not:N \l__color_current_tl }
0744   \prop_put:NVe \exp_not:c { l__color_named_ . _prop }
0745   \exp_not:c { l__color_named_ . _tl }
0746   { \exp_not:N \__color_values:N \exp_not:N \l__color_current_tl }
0747   \exp_not:N \exp_args:Ne \exp_not:N \__color_parse_aux:nN
0748   { \exp_not:N \tl_to_str:n {#1} } #2
0749 }
0750 \cs_new_protected:Npn \__color_parse_aux:nN #1#2
0751 {
```

```
0752     \color_if_exist:nTF {#1}
0753         { \__color_parse_set_eq:Nn #2 {#1} }
0754         { \__color_parse:Nw #2#1 ! \s__color_stop }
0755     \__color_check_model:N #2
0756 }
0757 \cs_new_protected:Npn \__color_parse_set_eq:Nn #1#2
0758 {
0759     \tl_if_empty:NTF \l_color_fixed_model_tl
0760         { \exp_args:Nv \__color_parse_set_eq:nNn { l__color_named_ #2 _tl } }
0761         { \exp_args:NV \__color_parse_set_eq:nNn \l_color_fixed_model_tl }
0762         #1 {#2}
0763 }
0764 \cs_new_protected:Npn \__color_parse_set_eq:nNn #1#2#3
0765 {
0766     \prop_get:cnNTF
0767         { l__color_named_ #3 _prop } {#1}
0768         \l__color_value_tl
0769         { \tl_set:Ne #2 { {#1} { \l__color_value_tl } } }
0770         {
0771             \tl_set_eq:Nc \l__color_model_tl { l__color_named_ #3 _tl }
0772             \prop_get:cVN { l__color_named_ #3 _prop } \l__color_model_tl
0773             \l__color_value_tl
0774             \__color_convert:nnN
0775             \l__color_model_tl {#1} \l__color_value_tl
0776             \tl_set:Ne #2
0777                 {
0778                     {#1}
0779                     { \l__color_value_tl }
0780                 }
0781         }
0782 }
0783 \cs_new_protected:Npn \__color_parse:Nw #1#2 ! #3 \s__color_stop
0784 {
0785     \color_if_exist:nTF {#2}
0786     {
0787         \tl_if_blank:nTF {#3}
0788             { \__color_parse_set_eq:Nn #1 {#2} }
0789             { \__color_parse_loop_init:Nnn #1 {#2} {#3} }
0790     }
0791     {
0792         \msg_error:nnn { color } { unknown-color } {#2}
0793         \tl_set:Nn \l__color_current_tl { { gray } { 0 } }
0794     }
0795 }
0796 \cs_new_protected:Npn \__color_parse_loop_init:Nnn #1#2#3
0797 {
```

```
0798 \group_begin:
0799 \__color_extract:nNN {#2} \l__color_model_tl \l__color_value_tl
0800 \__color_parse_loop:w #3 ! ! ! ! \s__color_stop
0801 \tl_set:Nc \l__color_internal_tl
0802 { { \l__color_model_tl } { \l__color_value_tl } }
0803 \exp_args:NNNV \group_end:
0804 \tl_set:Nn #1 \l__color_internal_tl
0805 }
0806 \cs_new_protected:Npn \__color_parse_loop:w #1 ! #2 ! #3 ! #4 ! #5 \s__color_stop
0807 {
0808 \tl_if_blank:nF {#1}
0809 {
0810 \bool_lazy_and:nnTF
0811 { \fp_compare_p:nNn {#1} > { 0 } }
0812 { \fp_compare_p:nNn {#1} < { 100 } }
0813 {
0814 \use:e
0815 {
0816 \__color_parse_loop:nn {#1}
0817 { \tl_if_blank:nTF {#2} { white } {#2} }
0818 }
0819 }
0820 { \__color_parse_loop_check:nn {#1} {#2} }
0821 }
0822 \tl_if_blank:nF {#3}
0823 { \__color_parse_loop:w #3 ! #4 ! #5 \s__color_stop }
0824 \__color_parse_end:
0825 }
0826 \cs_new_protected:Npn \__color_parse_loop_check:nn #1#2
0827 {
0828 \bool_if:NF \l__color_ignore_error_bool
0829 {
0830 \bool_lazy_or:nnT
0831 { \fp_compare_p:nNn {#1} < { 0 } }
0832 { \fp_compare_p:nNn {#1} > { 100 } }
0833 { \msg_error:nnnnn { color } { out-of-range } {#1} { 0 } { 100 } }
0834 }
0835 \fp_compare:nNnF {#1} > \c_zero_fp
0836 {
0837 \tl_if_blank:nTF {#2}
0838 { \__color_extract:nNN { white } }
0839 { \__color_extract:nNN {#2} }
0840 \l__color_model_tl \l__color_value_tl
0841 }
0842 }
0843 \cs_new_protected:Npn \__color_parse_loop:nn #1#2
```



```
0844 {
0845     \color_if_exist:nTF {#2}
0846     {
0847         \__color_extract:nNN {#2} \l__color_next_model_tl \l__color_next_value_tl
0848         \tl_if_eq:NNF \l__color_model_tl \l__color_next_model_tl
0849         {
0850             \str_if_eq:VnTF \l__color_model_tl { gray }
0851             { \__color_parse_gray:n {#2} }
0852             { \__color_parse_std:n {#2} }
0853         }
0854         \tl_set:Ne \l__color_value_tl
0855         {
0856             \__color_parse_mix:NVVn
0857             \l__color_model_tl \l__color_value_tl \l__color_next_value_tl {#1}
0858         }
0859     }
0860     {
0861         \msg_error:nnn { color } { unknown-color } {#2}
0862         \__color_extract:nNN { black } \l__color_model_tl \l__color_value_tl
0863         \__color_parse_break:w
0864     }
0865 }
0866 \cs_new_protected:Npn \__color_parse_gray:n #1
0867 {
0868     \tl_set_eq:NN \l__color_model_tl \l__color_next_model_tl
0869     \tl_set:Nn \l__color_next_model_tl { gray }
0870     \exp_args:NnV \__color_convert:nnN { gray } \l__color_model_tl
0871     \l__color_value_tl
0872     \prop_get:cVN { l__color_named_ #1 _prop } \l__color_model_tl
0873     \l__color_next_value_tl
0874 }
0875 \cs_new_protected:Npn \__color_parse_std:n #1
0876 {
0877     \prop_get:cVNF { l__color_named_ #1 _prop }
0878     \l__color_model_tl
0879     \l__color_next_value_tl
0880     {
0881         \__color_convert:VVN
0882         \l__color_next_model_tl
0883         \l__color_model_tl
0884         \l__color_next_value_tl
0885     }
0886 }
0887 \cs_new_protected:Npn \__color_parse_break:w #1 \__color_parse_end: { }
0888 \cs_new_protected:Npn \__color_parse_end: { }
0889 \cs_new:Npn \__color_parse_mix:Nnnn #1#2#3#4
```

```
0890 {
0891     \exp_args:Nf \__color_parse_mix:nNnn
0892     { \fp_eval:n { #4 / 100 } }
0893     #1 {#2} {#3}
0894 }
0895 \cs_generate_variant:Nn \__color_parse_mix:Nnnn { NVV }
0896 \cs_new:Npn \__color_parse_mix:nNnn #1#2#3#4
0897 {
0898     \use:c { __color_parse_mix_ #2 :nw } {#1}
0899     #3 \s__color_mark #4 \s__color_stop
0900 }
0901 \cs_new:Npn \__color_parse_mix_gray:nw #1#2 \s__color_mark #3 \s__color_stop
0902 { \fp_eval:n { #2 * #1 + #3 * ( 1 - #1 ) } }
0903 \cs_new:Npn \__color_parse_mix_rgb:nw
0904 #1#2 ~ #3 ~ #4 \s__color_mark #5 ~ #6 ~ #7 \s__color_stop
0905 {
0906     \fp_eval:n { #2 * #1 + #5 * ( 1 - #1 ) } \c_space_tl
0907     \fp_eval:n { #3 * #1 + #6 * ( 1 - #1 ) } \c_space_tl
0908     \fp_eval:n { #4 * #1 + #7 * ( 1 - #1 ) }
0909 }
0910 \cs_new:Npn \__color_parse_mix_cmyk:nw
0911 #1#2 ~ #3 ~ #4 ~ #5 \s__color_mark #6 ~ #7 ~ #8 ~ #9 \s__color_stop
0912 {
0913     \fp_eval:n { #2 * #1 + #6 * ( 1 - #1 ) } \c_space_tl
0914     \fp_eval:n { #3 * #1 + #7 * ( 1 - #1 ) } \c_space_tl
0915     \fp_eval:n { #4 * #1 + #8 * ( 1 - #1 ) } \c_space_tl
0916     \fp_eval:n { #5 * #1 + #9 * ( 1 - #1 ) }
0917 }
0918 \cs_new:Npn \__color_parse_model_gray:w #1 , #2 \s__color_stop
0919 { { gray } { \__color_parse_number:n {#1} } }
0920 \cs_new:Npn \__color_parse_model_rgb:w #1 , #2 , #3 , #4 \s__color_stop
0921 {
0922     { rgb }
0923     {
0924         \__color_parse_number:n {#1} ~
0925         \__color_parse_number:n {#2} ~
0926         \__color_parse_number:n {#3}
0927     }
0928 }
0929 \cs_new:Npn \__color_parse_model_cmyk:w #1 , #2 , #3 , #4 , #5 \s__color_stop
0930 {
0931     { cmyk }
0932     {
0933         \__color_parse_number:n {#1} ~
0934         \__color_parse_number:n {#2} ~
0935         \__color_parse_number:n {#3} ~
```

```
0936 \__color_parse_number:n {#4}
0937 }
0938 }
0939 \cs_new:Npn \__color_parse_number:n #1
0940 { \__color_parse_number:w #1 . 0 . \s__color_stop }
0941 \cs_new:Npn \__color_parse_number:w #1 . #2 . #3 \s__color_stop
0942 { \tl_if_blank:nTF {#1} { 0 } {#1} . #2 }
0943 \cs_new:Npn \__color_parse_model_Gray:w #1 , #2 \s__color_stop
0944 { { gray } { \fp_eval:n { #1 / 15 } } }
0945 \cs_new:Npn \__color_parse_model_hsb:w #1 , #2 , #3 , #4 \s__color_stop
0946 { \__color_parse_model_hsb:nnn {#1} {#2} {#3} }
0947 \cs_new:Npn \__color_parse_model_Hsb:w #1 , #2 , #3 , #4 \s__color_stop
0948 {
0949 \exp_args:Ne \__color_parse_model_hsb:nnn { \fp_eval:n { #1 / 360 } }
0950 {#2} {#3}
0951 }
0952 \cs_new:Npn \__color_parse_model_hsb:nnn #1#2#3
0953 {
0954 { rgb }
0955 {
0956 \exp_args:Ne \__color_parse_model_hsb_aux:nnn
0957 { \fp_eval:n { 6 * (#1) } } {#2} {#3}
0958 }
0959 }
0960 \cs_new:Npn \__color_parse_model_hsb_aux:nnn #1#2#3
0961 {
0962 \exp_args:Nee \__color_parse_model_hsb_aux:nnnn
0963 { \fp_eval:n { floor(#1) } } { \fp_eval:n { #1 - floor(#1) } }
0964 {#2} {#3}
0965 }
0966 \cs_new:Npn \__color_parse_model_hsb_aux:nnnn #1#2#3#4
0967 {
0968 \use:e
0969 {
0970 \exp_not:N \__color_parse_model_hsb_aux:nnnnn
0971 { \__color_parse_number:n {#4} }
0972 { \fp_eval:n { round(#4 * (1 - #3) ,5) } }
0973 { \fp_eval:n { round(#4 * (1 - #3 * #2) ,5) } }
0974 { \fp_eval:n { round(#4 * (1 - #3 * (1 - #2) ) ,5) } }
0975 {#1}
0976 }
0977 }
0978 \cs_new:Npn \__color_parse_model_hsb_aux:nnnnn #1#2#3#4#5
0979 { \use:c { __color_parse_model_hsb_ #5 :nnnn } {#1} {#2} {#3} {#4} }
0980 \cs_new:cpn { __color_parse_model_hsb_0:nnnn } #1#2#3#4 { #1 ~ #4 ~ #2 }
0981 \cs_new:cpn { __color_parse_model_hsb_1:nnnn } #1#2#3#4 { #3 ~ #1 ~ #2 }
```

```
0982 \cs_new:cpn { __color_parse_model_hsb_2:nnnn } #1#2#3#4 { #2 ~ #1 ~ #4 }
0983 \cs_new:cpn { __color_parse_model_hsb_3:nnnn } #1#2#3#4 { #2 ~ #3 ~ #1 }
0984 \cs_new:cpn { __color_parse_model_hsb_4:nnnn } #1#2#3#4 { #4 ~ #2 ~ #1 }
0985 \cs_new:cpn { __color_parse_model_hsb_5:nnnn } #1#2#3#4 { #1 ~ #2 ~ #3 }
0986 \cs_new:cpn { __color_parse_model_hsb_6:nnnn } #1#2#3#4 { #1 ~ #2 ~ #2 }
0987 \cs_new:Npn \__color_parse_model_HSB:w #1 , #2 , #3 , #4 \s__color_stop
0988 {
0989     \exp_args:Neee \__color_parse_model_hsb:nnn
0990     { \fp_eval:n { round((#1) / 240,5) } }
0991     { \fp_eval:n { round((#2) / 240,5) } }
0992     { \fp_eval:n { round((#3) / 240,5) } }
0993 }
0994 \cs_new:Npn \__color_parse_model_HTML:w #1 , #2 \s__color_stop
0995 { \__color_parse_model_HTML_aux:w #1 0 0 0 0 0 0 \s__color_stop }
0996 \cs_new:Npn \__color_parse_model_HTML_aux:w #1#2#3#4#5#6#7 \s__color_stop
0997 {
0998     { rgb }
0999     {
1000         \fp_eval:n { round(\int_from_hex:n {#1#2} / 255,5) } ~
1001         \fp_eval:n { round(\int_from_hex:n {#3#4} / 255,5) } ~
1002         \fp_eval:n { round(\int_from_hex:n {#5#6} / 255,5) }
1003     }
1004 }
1005 \cs_new:Npn \__color_parse_model_RGB:w #1 , #2 , #3 , #4 \s__color_stop
1006 {
1007     { rgb }
1008     {
1009         \fp_eval:n { round((#1) / 255,5) } ~
1010         \fp_eval:n { round((#2) / 255,5) } ~
1011         \fp_eval:n { round((#3) / 255,5) }
1012     }
1013 }
1014 \cs_new:Npn \__color_parse_model_wave:w #1 , #2 \s__color_stop
1015 {
1016     { rgb }
1017     {
1018         \fp_compare:nNnTF {#1} < { 420 }
1019         { \__color_parse_model_wave_aux:nn {#1} { 0.3 + 0.7 * (#1 - 380) / 40 } }
1020         {
1021             \fp_compare:nNnTF {#1} > { 700 }
1022             { \__color_parse_model_wave_aux:nn {#1} { 0.3 + 0.7 * (#1 - 780) / -80 } }
1023             { \__color_parse_model_wave_aux:nn {#1} { 1 } }
1024         }
1025     }
1026 }
1027 }
```

```
1028 \cs_new:Npn \__color_parse_model_wave_auxi:nn #1#2
1029 {
1030   \fp_compare:nNnTF {#1} < { 440 }
1031   {
1032     \__color_parse_model_wave_auxii:nn
1033     { 4 + \__color_parse_model_wave_rho:n { (#1 - 440) / -60 } }
1034     {#2}
1035   }
1036   {
1037     \fp_compare:nNnTF {#1} < { 490 }
1038     {
1039       \__color_parse_model_wave_auxii:nn
1040       { 4 - \__color_parse_model_wave_rho:n { (#1 - 440) / 50 } }
1041       {#2}
1042     }
1043     {
1044       \fp_compare:nNnTF {#1} < { 510 }
1045       {
1046         \__color_parse_model_wave_auxii:nn
1047         { 2 + \__color_parse_model_wave_rho:n { (#1 - 510) / -20 } }
1048         {#2}
1049       }
1050       {
1051         \fp_compare:nNnTF {#1} < { 580 }
1052         {
1053           \__color_parse_model_wave_auxii:nn
1054           { 2 - \__color_parse_model_wave_rho:n { (#1 - 510) / 70 } }
1055           {#2}
1056         }
1057         {
1058           \fp_compare:nNnTF {#1} < { 645 }
1059           {
1060             \__color_parse_model_wave_auxii:nn
1061             { \__color_parse_model_wave_rho:n { (#1 - 645) / -65 } }
1062             {#2}
1063           }
1064           { \__color_parse_model_wave_auxii:nn { 0 } {#2} }
1065         }
1066       }
1067     }
1068   }
1069 }
1070 \cs_new:Npn \__color_parse_model_wave_auxii:nn #1#2
1071 {
1072   \exp_args:Neee \__color_parse_model_hsb_aux:nnn
1073   { \fp_eval:n {#1} }
```

```

1074 { 1 }
1075 { \_color_parse_model_wave_rho:n {#2} }
1076 }
1077 \cs_new:Npn \_color_parse_model_wave_rho:n #1
1078 { \fp_eval:n { min(1, max(0,#1) ) } }
1079 \cs_new:Npn \_color_parse_model_cmy:w #1 , #2 , #3 , #4 \s_color_stop
1080 {
1081 { cmyk }
1082 { \_color_convert_rgb_cmyk:nnn {#1} {#2} {#3} }
1083 }
1084 \cs_new:Npn \_color_parse_model_tHsb:w #1 , #2 , #3 , #4 \s_color_stop
1085 {
1086 \exp_args:Ne \_color_parse_model_hsb:nnn
1087 { \_color_parse_model_tHsb:n {#1} } {#2} {#3}
1088 }
1089 \cs_new:Npn \_color_parse_model_tHsb:n #1
1090 {
1091 \_color_parse_model_tHsb:nw {#1}
1092 0 , 0 ;
1093 60 , 30 ;
1094 120 , 60 ;
1095 180 , 120 ;
1096 210 , 180 ;
1097 240 , 240 ;
1098 360 , 360 ;
1099 \q_recursion_tail , ;
1100 \q_recursion_stop
1101 }
1102 \cs_new:Npn \_color_parse_model_tHsb:nw #1 #2 , #3 ; #4 , #5 ;
1103 {
1104 \quark_if_recursion_tail_stop_do:nn {#4} { 0 }
1105 \fp_compare:nNnTF {#1} > {#4}
1106 { \_color_parse_model_tHsb:nw {#1} #4 , #5 ; }
1107 {
1108 \use_i_delimit_by_q_recursion_stop:nw
1109 { \fp_eval:n { ((#1 - #2) / (#4 - #2) * (#5 - #3) + #3) / 360 } }
1110 }
1111 }
1112 \cs_new:cpn { \_color_parse_model_&spot:w } #1 , #2 \s_color_stop
1113 { { gray } { #1 } }
1114 \tl_new:N \l_color_fixed_model_tl
1115 \cs_new_protected:Npn \_color_check_model:N #1
1116 {
1117 \tl_if_empty:NF \l_color_fixed_model_tl
1118 {
1119 \exp_after:wN \_color_check_model:nn #1

```

```
1120     \tl_if_eq:NNF \l__color_model_tl \l_color_fixed_model_tl 31120
1121     { 31121
1122         \__color_convert:VVN \l__color_model_tl \l_color_fixed_model_tl 31122
1123         \l__color_value_tl 31123
1124     } 31124
1125     \tl_set:Ne #1 31125
1126     { { \l_color_fixed_model_tl } { \l__color_value_tl } } 31126
1127 } 31127
1128 } 31128
1129 \cs_new_protected:Npn \__color_check_model:nn #1#2 31129
1130 { 31130
1131     \tl_set:Nn \l__color_model_tl {#1} 31131
1132     \tl_set:Nn \l__color_value_tl {#2} 31132
1133 } 31133
1134 \cs_new_protected:Npe \__color_finalise_current: 31134
1135 { 31135
1136     \tl_set:Ne \exp_not:c { l__color_named_ . _tl } 31136
1137     { \exp_not:N \__color_model:N \exp_not:N \l__color_current_tl } 31137
1138     \prop_clear:N \exp_not:c { l__color_named_ . _prop } 31138
1139     \prop_put:NVe \exp_not:c { l__color_named_ . _prop } 31139
1140     \exp_not:c { l__color_named_ . _tl } 31140
1141     { \exp_not:N \__color_values:N \exp_not:N \l__color_current_tl } 31141
1142 } 31142
1143 \cs_new_protected:Npn \color_select:n #1 31143
1144 { 31144
1145     \__color_parse:nN {#1} \l__color_current_tl 31145
1146     \__color_finalise_current: 31146
1147     \__color_select:N \l__color_current_tl 31147
1148 } 31148
1149 \cs_generate_variant:Nn \color_select:n { V } 31149
1150 \cs_new_protected:Npn \color_select:nn #1#2 31150
1151 { 31151
1152     \__color_select_main:Nnn \l__color_current_tl {#1} {#2} 31152
1153     \__color_finalise_current: 31153
1154     \__color_select:N \l__color_current_tl 31154
1155 } 31155
1156 \cs_generate_variant:Nn \color_select:nn { nV , V , VV } 31156
1157 \cs_new_protected:Npn \__color_select_main:Nnn #1#2#3 31157
1158 { 31158
1159     \use:e 31159
1160     { 31160
1161         \exp_not:N \__color_select_main:Nw \exp_not:N #1 31161
1162         \exp_not:n {#2} / / \exp_not:N \s__color_mark 31162
1163         #3 / / \exp_not:N \s__color_stop 31163
1164     } 31164
1165 } 31165
```



```
1166 \cs_new_protected:Npn \__color_select_main:Nw 31166
1167 #1 #2 / #3 / #4 \s__color_mark #5 / #6 / #7 \s__color_stop 31167
1168 { 31168
1169 \__color_select:nnN {#2} {#5} #1 31169
1170 \bool_lazy_or:nnF 31170
1171 { \tl_if_empty_p:N \l_color_fixed_model_tl } 31171
1172 { \str_if_eq_p:nV {#2} \l_color_fixed_model_tl } 31172
1173 { \__color_select_loop:Nw #1 #3 / #4 \s__color_mark #6 / #7 \s__color_stop } 31173
1174 } 31174
1175 \cs_new_protected:Npn \__color_select_loop:Nw 31175
1176 #1 #2 / #3 \s__color_mark #4 / #5 \s__color_stop 31176
1177 { 31177
1178 \str_if_eq:nVTF {#2} \l_color_fixed_model_tl 31178
1179 { \__color_select:nnN {#2} {#4} #1 } 31179
1180 { 31180
1181 \tl_if_blank:nTF {#2} 31181
1182 { \exp_after:wN \__color_select_swap:Nnn \exp_after:wN #1 #1 } 31182
1183 { \__color_select_loop:Nw #1 #3 \s__color_mark #5 \s__color_stop } 31183
1184 } 31184
1185 } 31185
1186 \cs_new_protected:Npn \__color_select:nnN #1#2#3 31186
1187 { 31187
1188 \cs_if_exist:cTF { __color_parse_model_ #1 :w } 31188
1189 { 31189
1190 \tl_set:Ne #3 31190
1191 { \use:c { __color_parse_model_ #1 :w } #2 , 0 , 0 , 0 , 0 \s__color_stop } 31191
1192 } 31192
1193 { \msg_error:nnn { color } { unknown-model } {#1} } 31193
1194 } 31194
1195 \cs_new_protected:Npn \__color_select_swap:Nnn #1#2#3 31195
1196 { 31196
1197 \__color_convert:nVnN {#2} \l_color_fixed_model_tl {#3} \l__color_value_tl 31197
1198 \tl_set:Ne #1 31198
1199 { { \l_color_fixed_model_tl } { \l__color_value_tl } } 31199
1200 } 31200
1201 \tl_new:N \l_color_math_active_tl 31201
1202 \tl_set:Nn \l_color_math_active_tl { ' } 31202
1203 \seq_new:N \g__color_math_seq 31203
1204 \cs_new_protected:Npn \color_math:nn #1#2 31204
1205 { 31205
1206 \__color_math:nn {#2} 31206
1207 { \__color_parse:nN {#1} \l__color_current_tl } 31207
1208 } 31208
1209 \cs_new_protected:Npn \color_math:nnn #1#2#3 31209
1210 { 31210
1211 \__color_math:nn {#3} 31211
```

```

1212 { \_color_select_main:Nnn \l\_color_current_tl {#1} {#2} }
1213 }
1214 \cs_new_protected:Npn \_color_math:nn #1#2
1215 {
1216   \seq_gpush:NV \g\_color_math_seq \l\_color_current_tl
1217   #2
1218   \_color_select_math:N \l\_color_current_tl
1219   #1
1220   \_color_math_scan:w
1221 }
1222 \cs_new_protected:Npn \_color_math_scan:w
1223 {
1224   \peek_remove_filler:n
1225   {
1226     \group_align_safe_begin:
1227     \peek_catcode:NTF \c_alignment_token
1228     {
1229       \group_align_safe_end:
1230       \_color_math_scan_end:
1231     }
1232     {
1233       \group_align_safe_end:
1234       \_color_math_scan_auxi:
1235     }
1236   }
1237 }
1238 \cs_new_protected:Npn \_color_math_scan_auxi:
1239 {
1240   \token_case_catcode:NnTF \l_peek_token
1241   {
1242     \c_math_subscript_token { }
1243     \c_math_superscript_token { }
1244   }
1245   { \_color_math_scripts:Nw }
1246   {
1247     \token_case_meaning:NnTF \l_peek_token
1248     {
1249       \tex_limits:D { \tex_limits:D }
1250       \tex_nolimits:D { \tex_nolimits:D }
1251       \tex_displaylimits:D { \tex_displaylimits:D }
1252     }
1253     { \_color_math_scan:w \use_none:n }
1254     { \_color_math_scan_auxii: }
1255   }
1256 }
1257 \cs_new_protected:Npn \_color_math_scan_auxii:

```

```

1258 {
1259     \tl_map_inline:Nn \l_color_math_active_tl
1260     {
1261         \token_if_eq_meaning:NNT \l_peek_token ##1
1262         {
1263             \tl_map_break:n
1264             {
1265                 \use_i:nn
1266                 { \_color_math_scan_auxiii:N ##1 }
1267             }
1268         }
1269         \_color_math_scan_end:
1270     }
1271 }
1272 \cs_new_protected:Npn \_color_math_scan_auxiii:N #1
1273 {
1274     \exp_after:wN \exp_after:wN \exp_after:wN \_color_math_scan:w
1275     \char_generate:nn { `#1 } { 13 }
1276 }
1277 \cs_new_protected:Npn \_color_math_scan_end:
1278 {
1279     \_color_backend_reset:
1280     \seq_gpop:NN \g_color_math_seq \l_color_current_tl
1281 }
1282 \cs_new_protected:Npn \_color_math_scripts:Nw #1
1283 {
1284     #1
1285     \c_group_begin_token
1286     \c_group_begin_token
1287     \seq_get:NN \g_color_math_seq \l_color_current_tl
1288     \_color_select:N \l_color_current_tl
1289     \group_insert_after:N \c_group_end_token
1290     \group_insert_after:N \_color_math_scan:w
1291     \peek_remove_filler:n
1292     {
1293         \peek_catcode_remove:NF \c_group_begin_token
1294         { \_color_math_script_aux:N }
1295     }
1296 }
1297 \cs_new_protected:Npn \_color_math_script_aux:N #1 { #1 \c_group_end_token }
1298 \cs_new_protected:Npn \color_fill:n #1
1299 {
1300     \_color_parse:nN {#1} \l_color_current_tl
1301     \exp_after:wN \_color_draw:nnn \l_color_current_tl { fill }
1302 }
1303 \cs_new_protected:Npn \color_stroke:n #1

```

```

1304 {
1305     \__color_parse:nN {#1} \l__color_current_tl
1306     \exp_after:wN \__color_draw:nnn \l__color_current_tl { stroke }
1307 }
1308 \cs_new_protected:Npn \color_fill:nn #1#2
1309 {
1310     \__color_select_main:Nnn \l__color_current_tl {#1} {#2}
1311     \exp_after:wN \__color_draw:nnn \l__color_current_tl { fill }
1312 }
1313 \cs_new_protected:Npn \color_stroke:nn #1#2
1314 {
1315     \__color_select_main:Nnn \l__color_current_tl {#1} {#2}
1316     \exp_after:wN \__color_draw:nnn \l__color_current_tl { stroke }
1317 }
1318 \cs_new_protected:Npn \__color_draw:nnn #1#2#3
1319 {
1320     \use:c { __color_backend_ #3 _ #1 :n } {#2}
1321     \exp_args:Nc \group_insert_after:N { __color_backend_ #3 _ reset: }
1322 }
1323 \tl_new:N \l__color_named_tl
1324 \cs_new_protected:Npn \color_set:nn #1#2
1325 {
1326     \exp_args:NV \__color_set:nnn
1327     \l_color_fixed_model_tl {#1} {#2}
1328 }
1329 \cs_new_protected:Npn \__color_set:nnn #1#2#3
1330 {
1331     \tl_clear:N \l_color_fixed_model_tl
1332     \__color_set:nn {#2} {#3}
1333     \tl_set:Nn \l_color_fixed_model_tl {#1}
1334 }
1335 \cs_new_protected:Npn \__color_set:nn #1#2
1336 {
1337     \str_if_eq:nnF {#1} { . }
1338     {
1339         \__color_parse:nN {#2} \l__color_named_tl
1340         \tl_clear_new:c { l__color_named_ #1 _tl }
1341         \tl_set:ce { l__color_named_ #1 _tl }
1342         { \__color_model:N \l__color_named_tl }
1343         \prop_clear_new:c { l__color_named_ #1 _prop }
1344         \prop_put:cve { l__color_named_ #1 _prop } { l__color_named_ #1 _tl }
1345         { \__color_values:N \l__color_named_tl }
1346         \__color_set:nnw {#1} {#2} #2 ! \s__color_stop
1347     }
1348 }
1349 \cs_new_protected:Npn \__color_set:nnw #1#2#3 ! #4 \s__color_stop

```

```

1350 {
1351     \clist_map_inline:nn { cmyk , gray , rgb }
1352     {
1353         \prop_get:cnNT { l__color_named_ #3 _prop } {##1} \l__color_internal_tl
1354         {
1355             \prop_if_in:cnF { l__color_named_ #1 _prop } {##1}
1356             {
1357                 \group_begin:
1358                     \bool_set_true:N \l__color_ignore_error_bool
1359                     \tl_set:cn { l__color_named_ #3 _tl } {##1}
1360                     \__color_parse:nN {#2} \l__color_internal_tl
1361                     \exp_args:NNNV \group_end:
1362                     \tl_set:Nn \l__color_internal_tl \l__color_internal_tl
1363                     \prop_put:cee { l__color_named_ #1 _prop }
1364                     { \__color_model:N \l__color_internal_tl }
1365                     { \__color_values:N \l__color_internal_tl }
1366                 }
1367             }
1368         }
1369     }
1370     \cs_new_protected:Npn \color_set:nnn #1#2#3
1371     {
1372         \str_if_eq:nnF {#1} { . }
1373         {
1374             \tl_clear_new:c { l__color_named_ #1 _tl }
1375             \prop_clear_new:c { l__color_named_ #1 _prop }
1376             \exp_args:Ne \__color_set_aux:nnn { \tl_to_str:n {#2} }
1377             {#1} {#3}
1378         }
1379     }
1380     \cs_new_protected:Npe \__color_set_aux:nnn #1#2#3
1381     {
1382         \exp_not:N \__color_set_colon:nnw {#2} {#3}
1383         #1 \c_colon_str \c_colon_str \exp_not:N \s__color_stop
1384     }
1385     \use:e
1386     {
1387         \cs_new_protected:Npn \exp_not:N \__color_set_colon:nnw
1388         #1#2 #3 \c_colon_str #4 \c_colon_str
1389         #5 \exp_not:N \s__color_stop
1390     }
1391     {
1392         \tl_if_blank:nTF {#4}
1393         { \__color_set_loop:nw {#1} #3 }
1394         { \__color_set_loop:nw {#1} #4 }
1395         // \s__color_mark #2 // \s__color_stop

```

```

1396 }
1397 \cs_new_protected:Npn \__color_set_loop:nw
1398   #1#2 / #3 \s__color_mark #4 / #5 \s__color_stop
1399   {
1400     \tl_if_blank:nF {#2}
1401     {
1402       \__color_select:nnN {#2} {#4} \l__color_named_tl
1403       \tl_set:Ne \l__color_internal_tl { \__color_model:N \l__color_named_tl }
1404       \tl_if_empty:cT { l__color_named_ #1 _tl }
1405       { \tl_set_eq:cN { l__color_named_ #1 _tl } \l__color_internal_tl }
1406       \prop_put:cVe { l__color_named_ #1 _prop } \l__color_internal_tl
1407       { \__color_values:N \l__color_named_tl }
1408       \__color_set_loop:nw {#1} #3 \s__color_mark #5 \s__color_stop
1409     }
1410   }
1411 \cs_new_protected:Npn \color_set_eq:nn #1#2
1412   {
1413     \color_if_exist:nTF {#2}
1414     {
1415       \tl_clear_new:c { l__color_named_ #1 _tl }
1416       \prop_clear_new:c { l__color_named_ #1 _prop }
1417       \str_if_eq:nnTF {#2} { . }
1418       {
1419         \tl_set:ce { l__color_named_ #1 _tl }
1420         { \__color_model:N \l__color_current_tl }
1421         \prop_put:cve { l__color_named_ #1 _prop } { l__color_named_ #1 _tl }
1422         { \__color_values:N \l__color_current_tl }
1423       }
1424       {
1425         \tl_set_eq:cc { l__color_named_ #1 _tl } { l__color_named_ #2 _tl }
1426         \prop_set_eq:cc { l__color_named_ #1 _prop } { l__color_named_ #2 _prop }
1427       }
1428     }
1429   }
1430   \msg_error:nnn { color } { unknown-color } {#2}
1431 }
1432 }
1433 \color_set:nnn { black } { gray } { 0 }
1434 \color_set:nnn { white } { gray } { 1 }
1435 \color_set:nnn { cyan } { cmyk } { 1 , 0 , 0 , 0 }
1436 \color_set:nnn { magenta } { cmyk } { 0 , 1 , 0 , 0 }
1437 \color_set:nnn { yellow } { cmyk } { 0 , 0 , 1 , 0 }
1438 \color_set:nnn { red } { rgb } { 1 , 0 , 0 }
1439 \color_set:nnn { green } { rgb } { 0 , 1 , 0 }
1440 \color_set:nnn { blue } { rgb } { 0 , 0 , 1 }
1441 \prop_new:c { l__color_named_._prop }

```

```
1442 \tl_new:c { l__color_named_.tl }
1443 \tl_set:ce { l__color_named_.tl } { \__color_model:N \l__color_current_tl }
1444 \prop_put:cve { l__color_named_.prop } { l__color_named_.tl }
1445 { \__color_values:N \l__color_current_tl }
1446 \cs_new_protected:Npn \color_export:nnN #1#2#3
1447 {
1448   \group_begin:
1449     \tl_if_exist:cT { c__color_export_ #2_tl }
1450     { \tl_set_eq:Nc \l__color_fixed_model_tl { c__color_export_ #2_tl } }
1451     \__color_parse:nN {#1} #3
1452     \__color_export:nN {#2} #3
1453     \exp_args:NNNV \group_end:
1454     \tl_set:Nn #3 #3
1455   }
1456 \cs_new_protected:Npn \color_export:nnnN #1#2#3#4
1457 {
1458   \__color_select_main:Nnn #4 {#1} {#2}
1459   \__color_export:nN {#3} #4
1460 }
1461 \cs_new_protected:Npn \__color_export:nN #1#2
1462 { \exp_after:wN \__color_export:nnnN #2 {#1} #2 }
1463 \cs_new:Npn \__color_export:nnnN #1#2#3#4
1464 {
1465   \cs_if_exist_use:cF { __color_export_format_ #3 :nnN }
1466   {
1467     \msg_error:nnn { color } { unknown-export-format } {#3}
1468     \use_none:nnn
1469   }
1470   {#1} {#2} #4
1471 }
1472 \cs_new_protected:Npn \__color_export_format_backend:nnN #1#2#3
1473 { \tl_set:Nn #3 { {#1} {#2} } }
1474 \cs_new_protected:Npn \__color_export:nnnNN #1#2#3#4#5
1475 {
1476   \str_if_eq:nnTF {#2} {#1}
1477   { #5 #4 #3 \s__color_stop }
1478   {
1479     \__color_convert:nnnN {#2} {#1} {#3} #4
1480     \exp_after:wN #5 \exp_after:wN #4
1481     #4 \s__color_stop
1482   }
1483 }
1484 \tl_const:cn { c__color_export_comma-sep-cmyk_tl } { cmyk }
1485 \tl_const:cn { c__color_export_comma-sep-rgb_tl } { rgb }
1486 \tl_const:Nn \c__color_export_HTML_tl { rgb }
1487 \tl_const:cn { c__color_export_space-sep-cmyk_tl } { cmyk }
```



```
1488 \tl_const:cn { c__color_export_space-sep-rgb_tl } { rgb }
1489 \group_begin:
1490 \cs_set_protected:Npn \__color_tmp:w #1#2
1491 {
1492   \cs_new_protected:cpe { __color_export_format_ #1 :nnN } ##1##2##3
1493   {
1494     \exp_not:N \__color_export:nnnNN {#2} {##1} {##2} ##3
1495     \exp_not:c { __color_export_ #1 :Nw }
1496   }
1497 }
1498 \__color_tmp:w { comma-sep-cmyk } { cmyk }
1499 \__color_tmp:w { comma-sep-rgb } { rgb }
1500 \__color_tmp:w { HTML } { rgb }
1501 \__color_tmp:w { space-sep-cmyk } { cmyk }
1502 \__color_tmp:w { space-sep-rgb } { rgb }
1503 \group_end:
1504 \cs_new_protected:cpn { __color_export_comma-sep-cmyk:Nw }
1505   #1#2 ~ #3 ~ #4 ~ #5 \s__color_stop
1506   { \tl_set:Nn #1 { #2 , #3 , #4 , #5 } }
1507 \cs_new_protected:cpn { __color_export_space-sep-cmyk:Nw } #1#2 \s__color_stop
1508   { \tl_set:Nn #1 {#2} }
1509 \cs_new_protected:cpn { __color_export_comma-sep-rgb:Nw } #1#2 ~ #3 ~ #4 \s__color_stop
1510   { \tl_set:Nn #1 { #2 , #3 , #4 } }
1511 \cs_new_protected:Npn \__color_export_HTML:Nw #1#2 ~ #3 ~ #4 \s__color_stop
1512 {
1513   \tl_set:Nn #1
1514   {
1515     \__color_export_HTML:n {#2}
1516     \__color_export_HTML:n {#3}
1517     \__color_export_HTML:n {#4}
1518   }
1519 }
1520 \cs_new:Npn \__color_export_HTML:n #1
1521 {
1522   \fp_compare:nNnTF {#1} = { 0 }
1523   { 00 }
1524   {
1525     \fp_compare:nNnT { #1 * 255 } < { 16 } { 0 }
1526     \int_to_Hex:n { \fp_to_int:n { #1 * 255 } }
1527   }
1528 }
1529 \cs_new_protected:cpn { __color_export_space-sep-rgb:Nw } #1#2 \s__color_stop
1530   { \tl_set:Nn #1 {#2} }
1531 \prop_new:N \l__color_internal_prop
1532 \int_new:N \g__color_model_int
1533 \tl_const:Nn \c__color_fallback_cmyk_tl { cmyk }
```

```
1534 \tl_const:Nn \c__color_fallback_gray_tl { gray } 31534
1535 \tl_const:Nn \c__color_fallback_rgb_tl { rgb } 31535
1536 \prop_new:N \g__color_colorants_prop 31536
1537 \prop_gput:Nnn \g__color_colorants_prop { black } { Black } 31537
1538 \prop_gput:Nnn \g__color_colorants_prop { blue } { Blue } 31538
1539 \prop_gput:Nnn \g__color_colorants_prop { cyan } { Cyan } 31539
1540 \prop_gput:Nnn \g__color_colorants_prop { green } { Green } 31540
1541 \prop_gput:Nnn \g__color_colorants_prop { magenta } { Magenta } 31541
1542 \prop_gput:Nnn \g__color_colorants_prop { none } { None } 31542
1543 \prop_gput:Nnn \g__color_colorants_prop { red } { Red } 31543
1544 \prop_gput:Nnn \g__color_colorants_prop { yellow } { Yellow } 31544
1545 \tl_const:Nn \c__color_model_whitepoint_CIELAB_a_tl { 1.0985 ~ 1 ~ 0.3558 } 31545
1546 \tl_const:Nn \c__color_model_whitepoint_CIELAB_b_tl { 0.9807 ~ 1 ~ 1.1822 } 31546
1547 \tl_const:Nn \c__color_model_whitepoint_CIELAB_e_tl { 1 ~ 1 ~ 1 } 31547
1548 \tl_const:cn { c__color_model_whitepoint_CIELAB_d50_tl } { 0.9642 ~ 1 ~ 0.8251 } 31548
1549 \tl_const:cn { c__color_model_whitepoint_CIELAB_d55_tl } { 0.9568 ~ 1 ~ 0.9214 } 31549
1550 \tl_const:cn { c__color_model_whitepoint_CIELAB_d65_tl } { 0.9504 ~ 1 ~ 1.0888 } 31550
1551 \tl_const:cn { c__color_model_whitepoint_CIELAB_d75_tl } { 0.9497 ~ 1 ~ 1.2261 } 31551
1552 \tl_const:Nn \c__color_model_range_CIELAB_tl { 0 ~ 100 ~ -128 ~ 127 ~ -128 ~ 127 } 31552
1553 \prop_new:N \g__color_alternative_model_prop 31553
1554 \clist_map_inline:nn { cyan , magenta , yellow , black } 31554
1555   { \prop_gput:Nnn \g__color_alternative_model_prop {#1} { cmyk } } 31555
1556 \clist_map_inline:nn { red , green , blue } 31556
1557   { \prop_gput:Nnn \g__color_alternative_model_prop {#1} { rgb } } 31557
1558 \prop_new:N \g__color_alternative_values_prop 31558
1559 \prop_gput:Nnn \g__color_alternative_values_prop { cyan } { 1 , 0 , 0 , 0 } 31559
1560 \prop_gput:Nnn \g__color_alternative_values_prop { magenta } { 0 , 1 , 0 , 0 } 31560
1561 \prop_gput:Nnn \g__color_alternative_values_prop { yellow } { 0 , 0 , 1 , 0 } 31561
1562 \prop_gput:Nnn \g__color_alternative_values_prop { black } { 0 , 0 , 0 , 1 } 31562
1563 \prop_gput:Nnn \g__color_alternative_values_prop { red } { 1 , 0 , 0 } 31563
1564 \prop_gput:Nnn \g__color_alternative_values_prop { green } { 0 , 1 , 0 } 31564
1565 \prop_gput:Nnn \g__color_alternative_values_prop { blue } { 0 , 0 , 1 } 31565
1566 \cs_new_protected:Npn \color_model_new:nnn #1#2#3 31566
1567 { 31567
1568   \exp_args:Nee \__color_model_new:nnn 31568
1569     { \tl_to_str:n {#1} } 31569
1570     { \str_casefold:n {#2} } {#3} 31570
1571   } 31571
1572 \cs_new_protected:Npn \__color_model_new:nnn #1#2#3 31572
1573 { 31573
1574   \cs_if_exist:cTF { __color_parse_model_ #1 :w } 31574
1575   { 31575
1576     \msg_error:nnn { color } { model-already-defined } {#1} 31576
1577   } 31577
1578   { 31578
1579     \cs_if_exist:cTF { __color_model_ #2 :n } 31579
```

```
1580     {
1581         \prop_set_from_keyval:Nn \l__color_internal_prop {#3}
1582         \use:c { __color_model_ #2 :n } {#1}
1583     }
1584     {
1585         \msg_error:nnn { color } { unknown-model-type } {#2}
1586     }
1587 }
1588 }
1589 \cs_new_protected:Npn \__color_model_init:nnn #1#2#3
1590 {
1591     \int_gincr:N \g__color_model_int
1592     \clist_map_inline:nn { fill , stroke , select }
1593     {
1594         \cs_new_protected:cpe { __color_backend_ ##1 _ #1 :n } ####1
1595         {
1596             \exp_not:c { __color_backend_ ##1 _ #2 :nn }
1597             { color \int_use:N \g__color_model_int } {####1}
1598         }
1599     }
1600     \cs_new_protected:cpe { __color_model_ #1 _white: }
1601     {
1602         \prop_put:Nnn \exp_not:N \l__color_named_white_prop {#1}
1603         { \exp_not:n {#3} }
1604         \exp_not:N \int_compare:nNnF { \tex_currentgrouplevel:D } = 0
1605         { \group_insert_after:N \exp_not:c { __color_model_ #1 _ white: } }
1606     }
1607     \use:c { __color_model_ #1 _white: }
1608 }
1609 \cs_generate_variant:Nn \__color_model_init:nnn { nne }
1610 \cs_new_protected:Npn \__color_model_separation:n #1
1611 {
1612     \prop_get:NnNTF \l__color_internal_prop { name }
1613     \l__color_internal_tl
1614     {
1615         \exp_args:NV \__color_model_separation:nn
1616         \l__color_internal_tl {#1}
1617     }
1618     {
1619         \msg_error:nnn { color }
1620         { separation-requires-name } {#1}
1621     }
1622 }
1623 \cs_new_protected:Npn \__color_model_separation:nn #1#2
1624 {
1625     \prop_get:NnNTF \l__color_internal_prop { alternative-model }
```

```
1626 \l__color_internal_tl
1627 {
1628     \exp_args:NV \__color_model_separation:nnn
1629     \l__color_internal_tl {#2} {#1}
1630 }
1631 {
1632     \msg_error:nnn { color }
1633     { separation-alternative-model } {#2}
1634 }
1635 }
1636 \cs_new_protected:Npn \__color_model_separation:nnn #1#2#3
1637 {
1638     \cs_if_exist:cTF { __color_model_separation_ #1 :nnnnnn }
1639     {
1640         \prop_get:NnNTF \l__color_internal_prop { alternative-values }
1641         \l__color_internal_tl
1642         {
1643             \exp_after:wN \__color_model_separation:w \l__color_internal_tl
1644             , 0 , 0 , 0 , 0 \s__color_stop {#2} {#3} {#1}
1645         }
1646         {
1647             \msg_error:nnn { color }
1648             { separation-alternative-values } {#2}
1649         }
1650     }
1651     {
1652         \msg_error:nnn { color }
1653         { unknown-alternative-model } {#1}
1654     }
1655 }
1656 \cs_new_protected:Npn \__color_model_separation:w
1657 #1 , #2 , #3 , #4 , #5 \s__color_stop #6#7#8
1658 {
1659     \__color_model_init:nnn {#6} { separation } { 0 }
1660     \cs_new_eq:cN { __color_parse_mix_ #6 :nw } \__color_parse_mix_gray:nw
1661     \cs_new:cpn { __color_parse_model_ #6 :w } ##1 , ##2 \s__color_stop
1662     { {#6} { \__color_parse_number:n {##1} } }
1663     \use:c { __color_model_separation_ #8 :nnnnnn }
1664     {#6} {#7} {#1} {#2} {#3} {#4}
1665     \prop_gput:Nnn \g__color_alternative_model_prop {#6} {#8}
1666     \prop_gput:Nne \g__color_colorants_prop {#6}
1667     { \str_convert_pdfname:n {#7} }
1668 }
1669 \cs_new_protected:Npn \__color_model_separation_cmyk:nnnnnn #1#2#3#4#5#6
1670 {
1671     \tl_const:cn { c__color_fallback_ #1 _tl } { cmyk }
```

```
1672 \cs_new:cpn { __color_convert_ #1 _cmyk:w } ##1 \s__color_stop
1673 {
1674     \fp_eval:n {##1 * #3} ~
1675     \fp_eval:n {##1 * #4} ~
1676     \fp_eval:n {##1 * #5} ~
1677     \fp_eval:n {##1 * #6}
1678 }
1679 \cs_new:cpn { __color_convert_cmyk_ #1 :w } ##1 \s__color_stop { 1 }
1680 \prop_gput:Nnn \g__color_alternative_values_prop {#1} { #3 , #4 , #5 , #6 }
1681 \__color_backend_separation_init:nnnnn {#2} { /DeviceCMYK } { }
1682 { 0 ~ 0 ~ 0 ~ 0 } { #3 ~ #4 ~ #5 ~ #6 }
1683 }
1684 \cs_new_protected:Npn \__color_model_separation_rgb:nnnnnn #1#2#3#4#5#6
1685 {
1686     \tl_const:cn { c__color_fallback_ #1 _tl } { rgb }
1687     \cs_new:cpn { __color_convert_ #1 _rgb:w } ##1 \s__color_stop
1688     {
1689         \fp_eval:n {##1 * #3} ~
1690         \fp_eval:n {##1 * #4} ~
1691         \fp_eval:n {##1 * #5}
1692     }
1693     \cs_new:cpn { __color_convert_rgb_ #1 :w } ##1 \s__color_stop { 1 }
1694     \prop_gput:Nnn \g__color_alternative_values_prop {#1} { #3 , #4 , #5 }
1695     \__color_backend_separation_init:nnnnn {#2} { /DeviceRGB } { }
1696     { 0 ~ 0 ~ 0 } { #3 ~ #4 ~ #5 }
1697 }
1698 \cs_new_protected:Npn \__color_model_separation_gray:nnnnnn #1#2#3#4#5#6
1699 {
1700     \tl_const:cn { c__color_fallback_ #1 _tl } { gray }
1701     \cs_new:cpn { __color_convert_ #1 _gray:w } ##1 \s__color_stop
1702     { \fp_eval:n {##1 * #3} }
1703     \cs_new:cpn { __color_convert_gray_ #1 :w } ##1 \s__color_stop { 1 }
1704     \prop_gput:Nnn \g__color_alternative_values_prop {#1} {#3}
1705     \__color_backend_separation_init:nnnnn {#2} { /DeviceGray } { } { 0 } {#3}
1706 }
1707 \cs_new_protected:Npn \__color_model_convert:nnn #1#2#3
1708 {
1709     \cs_new:cpe { __color_convert_ #1 _ #3 :w } ##1 \s__color_stop
1710     {
1711         \exp_not:N \exp_args:NNe \exp_not:N \use:nn
1712         \exp_not:c { __color_convert_ #2 _ #3 :w }
1713         { \exp_not:c { __color_convert_ #1 _ #2 :w } ##1 \s__color_stop }
1714         \c_space_tl \exp_not:N \s__color_stop
1715     }
1716 }
1717 \cs_new_protected:Npn \__color_model_separation_CIELAB:nnnnnn #1#2#3#4#5#6
```

```
1718 {
1719     \prop_get:NnNF \l__color_internal_prop { illuminant }
1720     \l__color_internal_tl
1721     {
1722         \msg_error:nnn { color }
1723         { CIELAB-requires-illuminant } {#1}
1724         \tl_set:Nn \l__color_internal_tl { d50 }
1725     }
1726     \exp_args:NV \__color_model_separation_CIELAB:nnnnnnn
1727     \l__color_internal_tl {#1} {#2} {#3} {#4} {#5} {#6}
1728 }
1729 \cs_new_protected:Npn \__color_model_separation_CIELAB:nnnnnnn #1#2#3#4#5#6#7
1730 {
1731     \tl_if_exist:cTF { c__color_model_whitepoint_CIELAB_ #1 _tl }
1732     {
1733         \__color_backend_separation_init_CIELAB:nnn {#1} {#3} { #4 ~ #5 ~ #6 }
1734         \tl_const:cn { c__color_fallback_ #2 _tl } { gray }
1735         \cs_new:cpn { __color_convert_ #2 _gray:w } ##1 \s__color_stop
1736         { 0 }
1737         \cs_new:cpn { __color_convert_gray_ #2 :w } ##1 \s__color_stop
1738         { 1 }
1739     }
1740     {
1741         \msg_error:nnn { color }
1742         { unknown-CIELAB-illuminant } {#1}
1743     }
1744 }
1745 \cs_new_protected:Npn \__color_model_devicen:n #1
1746 {
1747     \prop_get:NnNTF \l__color_internal_prop { names }
1748     \l__color_internal_tl
1749     {
1750         \exp_args:NV \__color_model_devicen:nn
1751         \l__color_internal_tl {#1}
1752     }
1753     {
1754         \msg_error:nnn { color }
1755         { DeviceN-requires-names } {#1}
1756     }
1757 }
1758 \cs_new_protected:Npn \__color_model_devicen:nn #1#2
1759 {
1760     \tl_clear:N \l__color_model_tl
1761     \clist_map_inline:nn {#1}
1762     {
1763         \prop_get:NnNTF \g__color_alternative_model_prop {##1}
```

```

1764         \l__color_internal_tl
1765     {
1766         \tl_if_empty:NTF \l__color_model_tl
1767         { \tl_set_eq:NN \l__color_model_tl \l__color_internal_tl }
1768         {
1769             \str_if_eq:VVF \l__color_model_tl \l__color_internal_tl
1770             {
1771                 \msg_error:nnn { color }
1772                 { DeviceN-inconsistent-alternative }
1773                 {#2}
1774                 \clist_map_break:n { \use_none:nnnn }
1775             }
1776         }
1777     }
1778     {
1779         \str_if_eq:nnF {##1} { none }
1780         {
1781             \msg_error:nnn { color }
1782             { DeviceN-no-alternative }
1783             {#2}
1784         }
1785     }
1786 }
1787 \tl_if_empty:NTF \l__color_model_tl
1788 {
1789     \msg_error:nnn { color }
1790     { DeviceN-no-alternative } {#2}
1791 }
1792 { \exp_args:NV \__color_model_devicen:nnn \l__color_model_tl {#1} {#2} }
1793 }
1794 \cs_new_protected:Npn \__color_model_devicen:nnn #1#2#3
1795 {
1796     \exp_args:Ne \__color_model_devicen:nnnn
1797     { \clist_count:n {#2} } {#1} {#2} {#3}
1798 }
1799 \cs_new_protected:Npn \__color_model_devicen:nnnn #1#2#3#4
1800 {
1801     \__color_model_init:nne {#4} { devicen }
1802     {
1803         0 \prg_replicate:nn { #1 - 1 } { ~ 0 }
1804     }
1805     \cs_if_exist_use:cF { __color_model_devicen_parse_ #1 :nn }
1806     { \__color_model_devicen_parse_generic:nn }
1807     {#4} {#1}
1808     \__color_model_devicen_init:nnn {#1} {#2} {#3}
1809     \__color_model_devicen_convert:nnne {#4} {#2} {#3}

```



```
1810 {
1811     1 \prg_replicate:nn { #1 - 1 } { ~ 1 }
1812 }
1813 }
1814 \cs_new_protected:cpn { __color_model_devicen_parse_1:nn } #1#2
1815 {
1816     \cs_new:cpn { __color_parse_model_ #1 :w } ##1 , ##2 \s__color_stop
1817     { {#1} { \__color_parse_number:n {##1} } }
1818     \cs_new_eq:cN { __color_parse_mix_ #1 :nw } \__color_parse_mix_gray:nw
1819 }
1820 \cs_new_protected:cpn { __color_model_devicen_parse_2:nn } #1#2
1821 {
1822     \cs_new:cpn { __color_parse_model_ #1 :w } ##1 , ##2 , ##3 \s__color_stop
1823     { {#1} { \__color_parse_number:n {##1} ~ \__color_parse_number:n {##2} } }
1824     \cs_new:cpn { __color_parse_mix_ #1 :nw }
1825     ##1##2 ~ ##3 \s__color_mark ##4 ~ ##5 \s__color_stop
1826     {
1827         \fp_eval:n { ##2 * ##1 + ##4 * ( 1 - ##1 ) } \c_space_tl
1828         \fp_eval:n { ##3 * ##1 + ##5 * ( 1 - ##1 ) }
1829     }
1830 }
1831 \cs_new_protected:cpn { __color_model_devicen_parse_3:nn } #1#2
1832 {
1833     \cs_new:cpn { __color_parse_model_ #1 :w } ##1 , ##2 , ##3 , ##4 \s__color_stop
1834     {
1835         {#1}
1836         {
1837             \__color_parse_number:n {##1} ~
1838             \__color_parse_number:n {##2} ~
1839             \__color_parse_number:n {##3}
1840         }
1841     }
1842     \cs_new_eq:cN { __color_parse_mix_ #1 :nw } \__color_parse_mix_rgb:nw
1843 }
1844 \cs_new_protected:cpn { __color_model_devicen_parse_4:nn } #1#2
1845 {
1846     \cs_new:cpn { __color_parse_model_ #1 :w }
1847     ##1 , ##2 , ##3 , ##4 , ##5 \s__color_stop
1848     {
1849         {#1}
1850         {
1851             \__color_parse_number:n {##1} ~
1852             \__color_parse_number:n {##2} ~
1853             \__color_parse_number:n {##3} ~
1854             \__color_parse_number:n {##4}
1855         }
```

```

1856     }
1857     \cs_new_eq:cN { __color_parse_mix_ #1 :nw } \__color_parse_mix_cmyk:nw
1858 }
1859 \cs_new_protected:Npn \__color_model_devicen_parse_generic:nn #1#2
1860 {
1861     \cs_new:cpn { __color_parse_model_ #1 :w } ##1 , ##2 \s__color_stop
1862     {
1863         {#1}
1864         { \__color_model_devicen_parse:nw {#2} ##1 , ##2 , \q_nil , \s__color_stop }
1865     }
1866     \cs_new:cpe { __color_parse_mix_ #1 :nw }
1867     ##1 ##2 \s__color_mark ##3 \s__color_stop
1868     {
1869         \exp_not:N \__color_model_devicen_mix:nw {##1}
1870         ##2 \c_space_tl \exp_not:N \q_nil \c_space_tl \exp_not:N \s__color_mark
1871         ##3 \c_space_tl \exp_not:N \q_nil \c_space_tl \exp_not:N \s__color_stop
1872     }
1873 }
1874 \cs_new:Npn \__color_model_devicen_parse:nw #1#2 , #3 \s__color_stop
1875 {
1876     \int_compare:nNnT {#1} > 0
1877     {
1878         \quark_if_nil:nTF {#2}
1879         { \prg_replicate:nn {#1} { 0 ~ } }
1880         {
1881             \__color_parse_number:n {#2}
1882             \int_compare:nNnT {#1} > 1 { ~ }
1883             \exp_args:Nf \__color_model_devicen_parse:nw
1884             { \int_eval:n { #1 - 1 } } #3 \s__color_stop
1885         }
1886     }
1887 }
1888 \cs_new:Npn \__color_model_devicen_mix:nw #1#2 ~ #3 \s__color_mark #4 ~ #5 \s__color_stop
1889 {
1890     \fp_eval:n { #2 * #1 + #4 * ( 1 - #1 ) }
1891     \quark_if_nil:oF { \tl_head:w #3 \q_stop }
1892     {
1893         \c_space_tl
1894         \__color_model_devicen_mix:nw {#1} #3 \s__color_mark #5 \s__color_stop
1895     }
1896 }
1897 \cs_new_protected:Npn \__color_model_devicen_init:nnn #1#2#3
1898 {
1899     \exp_args:Ne \__color_model_devicen_init:nnnn
1900     {
1901         \str_case:nn {#2}

```

```

1902         {
1903             { cmyk } { 4 }
1904             { gray } { 1 }
1905             { rgb } { 3 }
1906         }
1907     }
1908     {#1} {#2} {#3}
1909 }
1910 \cs_new_protected:Npn \__color_model_devicen_init:nnnn #1#2#3#4
1911 {
1912     \tl_set:Nc \l__color_internal_tl
1913     { \prg_replicate:nn {#1} { 1.0 ~ } }
1914     \int_zero:N \l__color_internal_int
1915     \clist_map_inline:nn {#4}
1916     {
1917         \int_incr:N \l__color_internal_int
1918         \prop_get:NnN \g__color_alternative_values_prop {##1}
1919         \l__color_value_tl
1920         \exp_after:wN \__color_model_devicen_transform:w
1921         \l__color_value_tl , 0 , 0 , 0 , \s__color_stop {#1} {#2}
1922     }
1923     \tl_put_right:Nc \l__color_internal_tl
1924     {
1925         \prg_replicate:nn {#1}
1926         { neg ~ 1.0 ~ add ~ #1 ~ -1 ~ roll ~ }
1927         \int_eval:n { #2 + #1 } ~ #1 ~ roll
1928         \prg_replicate:nn {#2} { ~ pop } ~
1929         #1 ~ 1 ~ roll
1930     }
1931     \use:e
1932     {
1933         \__color_backend_devicen_init:nnn
1934         {
1935             \clist_map_function:nN {#4}
1936             \__color_model_devicen_colorant:n
1937         }
1938         {
1939             \str_case:nn {#3}
1940             {
1941                 { cmyk } { /DeviceCMYK }
1942                 { gray } { /DeviceGray }
1943                 { rgb } { /DeviceRGB }
1944             }
1945         }
1946         { \exp_not:V \l__color_internal_tl }
1947     }

```

```

1948 }
1949 \cs_new_protected:Npn \__color_model_devicen_transform:w
1950 #1 , #2 , #3 , #4 , #5 \s__color_stop #6#7
1951 {
1952   \use:c { __color_model_devicen_transform_ #6 :nnnnn }
1953   {#1} {#2} {#3} {#4} {#7}
1954 }
1955 \cs_new_protected:cpn { __color_model_devicen_transform_1:nnnnn } #1#2#3#4#5
1956 { \__color_model_devicen_transform:nnn {#5} { 1 } {#1} }
1957 \cs_new_protected:cpn { __color_model_devicen_transform_3:nnnnn } #1#2#3#4#5
1958 {
1959   \clist_map_inline:nn { #1 , #2 , #3 }
1960   { \__color_model_devicen_transform:nnn {#5} { 3 } {##1} }
1961 }
1962 \cs_new_protected:cpn { __color_model_devicen_transform_4:nnnnn } #1#2#3#4#5
1963 {
1964   \clist_map_inline:nn { #1 , #2 , #3 , #4 }
1965   { \__color_model_devicen_transform:nnn {#5} { 4 } {##1} }
1966 }
1967 \cs_new_protected:Npn \__color_model_devicen_transform:nnn #1#2#3
1968 {
1969   \tl_put_right:Ne \l__color_internal_tl
1970   {
1971     \fp_compare:nNnF {#3} = \c_zero_fp
1972     {
1973       \int_eval:n { #1 - \l__color_internal_int + #2 } ~ index ~
1974       -#3 ~ mul ~ 1.0 ~ add ~ mul ~
1975     }
1976     #2 ~ -1 ~ roll ~
1977   }
1978 }
1979 \cs_new:Npn \__color_model_devicen_colorant:n #1
1980 {
1981   / \prop_item:Nn \g__color_colorants_prop {#1} ~
1982 }
1983 \cs_new_protected:Npn \__color_model_devicen_convert:nnnn #1#2#3
1984 {
1985   \use:c { __color_model_devicen_convert_ #2 :nnn } {#1} {#3}
1986 }
1987 \cs_generate_variant:Nn \__color_model_devicen_convert:nnnn { nnne }
1988 \cs_new_protected:Npn \__color_model_devicen_convert_cmyk:nnn #1#2
1989 {
1990   \tl_const:cn { c__color_fallback_ #1 _tl } { cmyk }
1991   \__color_model_devicen_convert:nnnnn {#1} { cmyk } { 4 } {#2}
1992 }
1993 \cs_new_protected:Npn \__color_model_devicen_convert_gray:nnn #1#2

```

```
1994 {
1995     \tl_const:cn { c__color_fallback_ #1 _tl } { gray }
1996     \__color_model_devicen_convert:nnnnn {#1} { gray } { 1 } {#2}
1997 }
1998 \cs_new_protected:Npn \__color_model_devicen_convert_rgb:nnn #1#2
1999 {
2000     \tl_const:cn { c__color_fallback_ #1 _tl } { rgb }
2001     \__color_model_devicen_convert:nnnnn {#1} { rgb } { 3 } {#2}
2002 }
2003 \cs_new_protected:Npn \__color_model_devicen_convert:nnnnn #1#2#3#4#5
2004 {
2005     \cs_new:cpn { __color_convert_ #2 _ #1 :w } ##1 \s__color_stop {#5}
2006     \cs_new:cpe { __color_convert_ #1 _ #2 :w } ##1 \s__color_stop
2007     {
2008         \exp_not:c { __color_convert_devicen_ #2 : \prg_replicate:nn {#3} { n } w }
2009         \prg_replicate:nn {#3} { { 1 } }
2010         ##1 ~ \exp_not:N \s__color_mark
2011         \clist_map_function:nN {#4} \__color_model_devicen_convert:n
2012         {}
2013         \exp_not:N \s__color_stop
2014     }
2015 }
2016 \cs_new:Npn \__color_model_devicen_convert:n #1
2017 {
2018     {
2019         \exp_args:Ne \__color_model_devicen_convert_aux:n
2020         { \prop_item:Nn \g__color_alternative_values_prop {#1} }
2021     }
2022 }
2023 \cs_new:Npn \__color_model_devicen_convert_aux:n #1
2024 { \__color_model_devicen_convert_aux:w #1 , , , , \s__color_stop }
2025 \cs_new:Npn \__color_model_devicen_convert_aux:w #1 , #2 , #3 , #4 , #5 \s__color_stop
2026 {
2027     {#1}
2028     \tl_if_blank:nF {#2}
2029     {
2030         {#2}
2031         \tl_if_blank:nF {#3}
2032         {
2033             {#3}
2034             \tl_if_blank:nF {#4} { {#4} }
2035         }
2036     }
2037 }
2038 \cs_new:Npn \__color_convert_devicen_cmyk:nnnnw
2039 #1#2#3#4#5 ~ #6 \s__color_mark #7#8 \s__color_stop
```

```
2040 { 3204
2041 \_color_convert_devicen_cmyk:nnnnnnnn {#5} {#1} {#2} {#3} {#4} #7 3204
2042 #6 \s_color_mark #8 \s_color_stop 3204
2043 } 3204
2044 \cs_new:Npn \_color_convert_devicen_cmyk:nnnnnnnn #1#2#3#4#5#6#7#8#9 3204
2045 { 3204
2046 \use:e 3204
2047 { 3204
2048 \exp_not:N \_color_convert_devicen_cmyk_aux:nnnnw 3204
2049 { \fp_eval:n { #2 * (1 - (#1 * #6)) } } 3204
2050 { \fp_eval:n { #3 * (1 - (#1 * #7)) } } 3205
2051 { \fp_eval:n { #4 * (1 - (#1 * #8)) } } 3205
2052 { \fp_eval:n { #5 * (1 - (#1 * #9)) } } 3205
2053 } 3205
2054 } 3205
2055 \cs_new:Npn \_color_convert_devicen_cmyk_aux:nnnnw 3205
2056 #1#2#3#4 #5 \s_color_mark #6 \s_color_stop 3205
2057 { 3205
2058 \tl_if_blank:nTF {#5} 3205
2059 { 3205
2060 \fp_eval:n { 1 - #1 } ~ 3206
2061 \fp_eval:n { 1 - #2 } ~ 3206
2062 \fp_eval:n { 1 - #3 } ~ 3206
2063 \fp_eval:n { 1 - #4 } 3206
2064 } 3206
2065 { 3206
2066 \_color_convert_devicen_cmyk:nnnnw {#1} {#2} {#3} {#4} 3206
2067 #5 \s_color_mark #6 \s_color_stop 3206
2068 } 3206
2069 } 3206
2070 \cs_new:Npn \_color_convert_devicen_gray:nw 3207
2071 #1#2 ~ #3 \s_color_mark #4#5 \s_color_stop 3207
2072 { 3207
2073 \_color_convert_devicen_gray:nnn {#2} {#1} #4 3207
2074 #3 \s_color_mark #5 \s_color_stop 3207
2075 } 3207
2076 \cs_new:Npn \_color_convert_devicen_gray:nnn #1#2#3 3207
2077 { 3207
2078 \exp_args:Ne \_color_convert_devicen_gray_aux:nw 3207
2079 { \fp_eval:n { #2 * (1 - (#1 * #3)) } } 3207
2080 } 3208
2081 \cs_new:Npn \_color_convert_devicen_gray_aux:nw 3208
2082 #1 #2 \s_color_mark #3 \s_color_stop 3208
2083 { 3208
2084 \tl_if_blank:nTF {#2} 3208
2085 { \fp_eval:n { 1 - #1 } } 3208
```

```
2086 {
2087     \__color_convert_devicen_gray:nw {#1}
2088     #2 \s__color_mark #3 \s__color_stop
2089 }
2090 }
2091 \cs_new:Npn \__color_convert_devicen_rgb:nnnw
2092 #1#2#3#4 ~ #5 \s__color_mark #6#7 \s__color_stop
2093 {
2094     \__color_convert_devicen_rgb:nnnnnnn {#4} {#1} {#2} {#3} #6
2095     #5 \s__color_mark #7 \s__color_stop
2096 }
2097 \cs_new:Npn \__color_convert_devicen_rgb:nnnnnnn #1#2#3#4#5#6#7
2098 {
2099     \use:e
2100     {
2101         \exp_not:N \__color_convert_devicen_rgb_aux:nnnw
2102         { \fp_eval:n { #2 * (1 - (#1 * #5)) } }
2103         { \fp_eval:n { #3 * (1 - (#1 * #6)) } }
2104         { \fp_eval:n { #4 * (1 - (#1 * #7)) } }
2105     }
2106 }
2107 \cs_new:Npn \__color_convert_devicen_rgb_aux:nnnw
2108 #1#2#3 #4 \s__color_mark #5 \s__color_stop
2109 {
2110     \tl_if_blank:nTF {#4}
2111     {
2112         \fp_eval:n { 1 - #1 } ~
2113         \fp_eval:n { 1 - #2 } ~
2114         \fp_eval:n { 1 - #3 }
2115     }
2116     {
2117         \__color_convert_devicen_rgb:nnnw {#1} {#2} {#3}
2118         #4 \s__color_mark #5 \s__color_stop
2119     }
2120 }
2121 \prop_const_from_keyval:Nn \c__color_icc_colorspace_signatures_prop
2122 {
2123     47524159 = {1} {1} {0} {},
2124     52474220 = {3} {0~0~0} {1~1~1} {},
2125     434D594B = {4} {0~0~0~1} {0~0~0~0} {},
2126     4C616220 = {3} {0~0~0} {100~0~0} {0~100~-128~127~-128~127}
2127 }
2128 \cs_new_protected:Npn \__color_model_iccbased:n #1
2129 {
2130     \prop_get:NnNTF \l__color_internal_prop { file }
2131     \l__color_internal_tl
```



```
2132 {
2133     \exp_args:NV \__color_model_iccbased:nn
2134     \l__color_internal_tl {#1}
2135 }
2136 {
2137     \msg_error:nnn { color }
2138     { ICCBased-requires-file } {#1}
2139 }
2140 }
2141 \cs_new_protected:Npn \__color_model_iccbased:nn #1#2
2142 {
2143     \prop_get:NeNTF \c__color_icc_colorspace_signatures_prop
2144     { \file_hex_dump:nnn { #1 } { 17 } { 20 } } \l__color_internal_tl
2145     {
2146         \exp_last_unbraced:NV \__color_model_iccbased_aux:nnnnnn
2147         \l__color_internal_tl { #2 } { #1 }
2148     }
2149     {
2150         \msg_error:nnn { color }
2151         { ICCBased-unsupported-colorspace } {#2}
2152     }
2153 }
2154 \cs_new_protected:Npn \__color_model_iccbased_aux:nnnnnn #1#2#3#4#5#6
2155 {
2156     \__color_model_init:nnn {#5} { iccbased } {#3}
2157     \tl_const:cn { c__color_fallback_ #5 _tl } { gray }
2158     \cs_new:cpn { __color_convert_ #5 _gray:w } ##1 \s__color_stop { 0 }
2159     \cs_new:cpn { __color_convert_gray_ #5 :w } ##1 \s__color_stop { #2 }
2160     \use:c { __color_model_devicen_parse_ #1 :nn } {#5} {#1}
2161     \exp_args:Ne \__color_backend_iccbased_init:nnn
2162     { \file_full_name:n {#6} } {#1} {#4}
2163 }
2164 \cs_new_protected:Npn \color_profile_apply:nn #1#2
2165 {
2166     \exp_args:Ne \__color_profile_apply:nn
2167     { \file_full_name:n {#1} } {#2}
2168 }
2169 \cs_new_protected:Npn \__color_profile_apply:nn #1#2
2170 {
2171     \cs_if_exist_use:cF { __color_profile_apply_ \tl_to_str:n {#2} :n }
2172     {
2173         \msg_error:nnn { color } { ICC-Device-unknown } {#2}
2174         \use_none:n
2175     }
2176     {#1}
2177 }
```

```
2178 \cs_new_protected:Npn \__color_profile_apply_gray:n #1
2179 {
2180   \int_gincr:N \g__color_model_int
2181   \__color_backend_iccbased_device:nnn {#1} { Gray } { 1 }
2182 }
2183 \cs_new_protected:Npn \__color_profile_apply_rgb:n #1
2184 {
2185   \int_gincr:N \g__color_model_int
2186   \__color_backend_iccbased_device:nnn {#1} { RGB } { 3 }
2187 }
2188 \cs_new_protected:Npn \__color_profile_apply_cmyk:n #1
2189 {
2190   \int_gincr:N \g__color_model_int
2191   \__color_backend_iccbased_device:nnn {#1} { CMYK } { 4 }
2192 }
2193 \cs_new_protected:Npn \color_show:n
2194 { \__color_show:Nn \msg_show:nneeee }
2195 \cs_new_protected:Npn \color_log:n
2196 { \__color_show:Nn \msg_log:nneeee }
2197 \cs_new_protected:Npn \__color_show:Nn #1#2
2198 {
2199   #1 { color } { show }
2200   {#2}
2201   {
2202     \color_if_exist:nT {#2}
2203     {
2204       \exp_args:Nv \__color_show:n { l__color_named_ #2 _tl }
2205       \prop_map_function:cN
2206         { l__color_named_ #2 _prop }
2207         \msg_show_item_unbraced:nn
2208     }
2209   }
2210   { }
2211   { }
2212 }
2213 \cs_new:Npn \__color_show:n #1
2214 {
2215   \msg_show_item_unbraced:nn { model } {#1}
2216 }
2217 \msg_new:nnnn { color } { CIELAB-requires-illuminant }
2218 { CIELAB~color~space~'#1'~require~an~illuminant. }
2219 {
2220   LaTeX~has~been~asked~to~create~a~separation~color~space~using~
2221   CIELAB~specifications,~but~no~\\ \\
2222   \iow_indent:n { illuminant~::~<basis> }
2223   \\ \\
```

```
2224 key~was~given~with~the~correct~information.~LaTeX~will~use~illuminant~ 32224
2225 'd50'~for~recovery. 32225
2226 } 32226
2227 \msg_new:nnnn { color } { conversion-not-available } 32227
2228 { No~model~conversion~available~from~'#1'~to~'#2'. } 32228
2229 { 32229
2230 LaTeX~has~been~asked~to~convert~a~color~from~model~'#1'~ 32230
2231 to~model~'#2',~but~there~is~no~method~available~to~do~that. 32231
2232 } 32232
2233 \msg_new:nnnn { color } { DeviceN-inconsistent-alternative } 32233
2234 { DeviceN~color~spaces~require~a~single~alternative~space. } 32234
2235 { 32235
2236 LaTeX~has~been~asked~to~create~a~DeviceN~color~space~'#1',~ 32236
2237 but~the~constituent~colors~do~not~have~a~common~alternative~ 32237
2238 color. 32238
2239 } 32239
2240 \msg_new:nnnn { color } { DeviceN-no-alternative } 32240
2241 { DeviceN~color~spaces~require~an~alternative~space. } 32241
2242 { 32242
2243 LaTeX~has~been~asked~to~create~a~DeviceN~color~space~'#1',~ 32243
2244 but~the~constituent~colors~do~not~all~have~a~device-based~alternative. 32244
2245 } 32245
2246 \msg_new:nnnn { color } { DeviceN-requires-names } 32246
2247 { DeviceN~color~space~'#1'~require~a~list~of~names. } 32247
2248 { 32248
2249 LaTeX~has~been~asked~to~create~a~DeviceN~color~space,~ 32249
2250 but~no~\\ \\ 32250
2251 \iow_indent:n { names~==<names> } 32251
2252 \\ \\ 32252
2253 key~was~given~with~the~correct~information. 32253
2254 } 32254
2255 \msg_new:nnnn { color } { ICC-Device-unknown } 32255
2256 { Unknown~device~color~space~'#1'. } 32256
2257 { 32257
2258 LaTeX~has~been~asked~to~apply~an~ICC~profile~but~the~device~color~space~ 32258
2259 '#1'~is~unknown. 32259
2260 } 32260
2261 \msg_new:nnnn { color } { ICCBased-unsupported-colorspace } 32261
2262 { ICCBased~color~space~'#1'~uses~an~unsupported~data~color~space. } 32262
2263 { 32263
2264 LaTeX~has~been~asked~to~create~a~ICCBased~colorspace,~but~the~ 32264
2265 used~data~colorspace~is~not~supported.~ICC~profiles~used~for~ 32265
2266 defining~a~ICCBased~colorspace~should~use~a~Lab,~RGB,~or~ 32266
2267 CMYK~data~colorspace.~LaTeX~will~ignore~this~request. 32267
2268 } 32268
2269 \msg_new:nnnn { color } { ICCBased-requires-file } 32269
```

```
2270 { ICCBased~color~space~'#1'~require~an~file. } 32270
2271 { 32271
2272 LaTeX~has~been~asked~to~create~an~ICCBased~color~space,~but~no~\\ \\ 32272
2273 \iow_indent:n { file~==<name> } 32273
2274 \\ \\ 32274
2275 key~was~given~with~the~correct~information.~LaTeX~will~ignore~this~ 32275
2276 request. 32276
2277 } 32277
2278 \msg_new:nnnn { color } { model-already-defined } 32278
2279 { Color~model~'#1'~already~defined. } 32279
2280 { 32280
2281 LaTeX~was~asked~to~define~a~new~color~model~called~'#1',~but~ 32281
2282 this~color~model~already~exists. 32282
2283 } 32283
2284 \msg_new:nnnn { color } { out-of-range } 32284
2285 { Input~value~#1~out~of~range~[#2,~#3]. } 32285
2286 { 32286
2287 LaTeX~was~expecting~a~value~in~the~range~[#2,~#3]~as~part~of~a~color,~ 32287
2288 but~you~gave~#1.~LaTeX~will~assume~you~meant~the~limit~of~the~range~ 32288
2289 and~continue. 32289
2290 } 32290
2291 \msg_new:nnnn { color } { separation-alternative-model } 32291
2292 { Separation~color~space~'#1'~require~an~alternative~model. } 32292
2293 { 32293
2294 LaTeX~has~been~asked~to~create~a~separation~color~space,~ 32294
2295 but~no~\\ \\ 32295
2296 \iow_indent:n { alternative-model~==<model> } 32296
2297 \\ \\ 32297
2298 key~was~given~with~the~correct~information. 32298
2299 } 32299
2300 \msg_new:nnnn { color } { separation-alternative-values } 32300
2301 { Separation~color~space~'#1'~require~values~for~the~alternative~space. } 32301
2302 { 32302
2303 LaTeX~has~been~asked~to~create~a~separation~color~space,~ 32303
2304 but~no~\\ \\ 32304
2305 \iow_indent:n { alternative-values~==<model> } 32305
2306 \\ \\ 32306
2307 key~was~given~with~the~correct~information. 32307
2308 } 32308
2309 \msg_new:nnnn { color } { separation-requires-name } 32309
2310 { Separation~color~space~'#1'~require~a~formal~name. } 32310
2311 { 32311
2312 LaTeX~has~been~asked~to~create~a~separation~color~space,~ 32312
2313 but~no~\\ \\ 32313
2314 \iow_indent:n { name~==<formal~name> } 32314
2315 \\ \\ 32315
```

```
2316 key~was~given~with~the~correct~information.
2317 }
2318 \msg_new:nnn { color } { unhandled-model }
2319 {
2320   Unhandled~color~model~in~LaTeX2e~value~"#1":
2321   \\ \\
2322   falling~back~on~grayscale.
2323 }
2324 \msg_new:nnnn { color } { unknown-color }
2325 { Unknown~color~'#1'. }
2326 {
2327   LaTeX~has~been~asked~to~use~a~color~named~'#1',~
2328   but~this~has~never~been~defined.
2329 }
2330 \msg_new:nnnn { color } { unknown-alternative-model }
2331 { Separation~color~space~'#1'~require~an~valid~alternative~space. }
2332 {
2333   LaTeX~has~been~asked~to~create~a~separation~color~space,~
2334   but~the~model~given~as\\ \\
2335   \iow_indent:n { alternative-model~==<model> }
2336   \\ \\
2337   is~unknown.
2338 }
2339 \msg_new:nnnn { color } { unknown-export-format }
2340 { Unknown~export~format~'#1'. }
2341 {
2342   LaTeX~has~been~asked~to~export~a~color~in~format~'#1',~
2343   but~this~has~never~been~defined.
2344 }
2345 \msg_new:nnnn { color } { unknown-CIELAB-illuminant }
2346 { Unknown~illuminant~model~'#1'. }
2347 {
2348   LaTeX~has~been~asked~to~use~create~a~color~space~using~CIELAB~
2349   illuminant~'#1',~but~this~does~not~exist.
2350 }
2351 \msg_new:nnnn { color } { unknown-model }
2352 { Unknown~color~model~'#1'. }
2353 {
2354   LaTeX~has~been~asked~to~use~a~color~model~called~'#1',~
2355   but~this~model~is~not~set~up.
2356 }
2357 \msg_new:nnnn { color } { unknown-model-type }
2358 { Unknown~color~model~type~'#1'. }
2359 {
2360   LaTeX~has~been~asked~to~create~a~new~color~model~called~'#1',~
2361   but~this~type~of~model~was~never~set~up.
```

```

2362 }
2363 \prop_gput:Nnn \g_msg_module_name_prop { color } { LaTeX }
2364 \prop_gput:Nnn \g_msg_module_type_prop { color } { }
2365 \msg_new:nnn { color } { show }
2366 {
2367   The~color~#1~
2368   \tl_if_empty:nTF {#2}
2369     { is~undefined. }
2370     { has~the~properties: #2 }
2371 }
2372 %% File: l3graphics.dtx
2373 \cs_if_exist:NT \@expl@finalise@setup@@
2374 {
2375   \tl_gput_right:Nn \@expl@finalise@setup@@
2376     { \declare@file@substitution { l3graphics.sty } { null.tex } }
2377 }
2378 \dim_new:N \l__graphics_internal_dim
2379 \ior_new:N \l__graphics_internal_ior
2380 \tl_new:N \l__graphics_internal_tl
2381 \scan_new:N \s__graphics_stop
2382 \tl_new:N \l__graphics_pagebox_tl
2383 \keys_define:nn { graphics }
2384 {
2385   decodearray .str_set:N =
2386     \l__graphics_decodearray_str ,
2387   draft .bool_set:N =
2388     \l__graphics_draft_bool ,
2389   interpolate .bool_set:N =
2390     \l__graphics_interpolate_bool ,
2391   pagebox .choices:nn =
2392     { art , bleed , crop , media , trim }
2393     {
2394       \tl_set:Ne \l__graphics_pagebox_tl
2395         { \l_keys_choice_tl box }
2396     } ,
2397   pagebox .initial:n =
2398     crop ,
2399   page .int_set:N =
2400     \l__graphics_page_int ,
2401   pdf-attr .str_set:N =
2402     \l__graphics_pdf_str ,
2403   type . str_set:N =
2404     \l__graphics_type_str
2405 }
2406 \dim_new:N \l__graphics_llx_dim
2407 \dim_new:N \l__graphics_lly_dim

```

```
2408 \dim_new:N \l__graphics_urx_dim 32408
2409 \dim_new:N \l__graphics_ury_dim 32409
2410 \cs_new_protected:Npn \__graphics_bb_save:n #1 32410
2411 { 32411
2412     \dim_if_exist:cTF { c__graphics_ #1 _urx_dim } 32412
2413     { \msg_error:nnn { graphic } { bb-already-cached } {#1} } 32413
2414     { 32414
2415         \dim_compare:nNnF \l__graphics_llx_dim = { Opt } 32415
2416         { \dim_const:cn { c__graphics_ #1 _llx_dim } { \l__graphics_llx_dim } } 32416
2417         \dim_compare:nNnF \l__graphics_lly_dim = { Opt } 32417
2418         { \dim_const:cn { c__graphics_ #1 _lly_dim } { \l__graphics_lly_dim } } 32418
2419         \dim_const:cn { c__graphics_ #1 _urx_dim } { \l__graphics_urx_dim } 32419
2420         \dim_const:cn { c__graphics_ #1 _ury_dim } { \l__graphics_ury_dim } 32420
2421     } 32421
2422 } 32422
2423 \cs_generate_variant:Nn \__graphics_bb_save:n { e } 32423
2424 \cs_new_protected:Npn \__graphics_bb_restore:nF #1#2 32424
2425 { 32425
2426     \dim_if_exist:cTF { c__graphics_ #1 _urx_dim } 32426
2427     { 32427
2428         \dim_set_eq:Nc \l__graphics_urx_dim { c__graphics_ #1 _urx_dim } 32428
2429         \dim_set_eq:Nc \l__graphics_ury_dim { c__graphics_ #1 _ury_dim } 32429
2430         \dim_if_exist:cTF { c__graphics_ #1 _llx_dim } 32430
2431         { \dim_set_eq:Nc \l__graphics_llx_dim { c__graphics_ #1 _llx_dim } } 32431
2432         { \dim_zero:N \l__graphics_llx_dim } 32432
2433         \dim_if_exist:cTF { c__graphics_ #1 _lly_dim } 32433
2434         { \dim_set_eq:Nc \l__graphics_lly_dim { c__graphics_ #1 _lly_dim } } 32434
2435         { \dim_zero:N \l__graphics_lly_dim } 32435
2436     } 32436
2437     {#2} 32437
2438 } 32438
2439 \cs_generate_variant:Nn \__graphics_bb_restore:nF { e } 32439
2440 \cs_new_protected:Npn \__graphics_extract_bb:n #1 32440
2441 { 32441
2442     \int_compare:nNnTF \l__graphics_page_int > 0 32442
2443     { \__graphics_extract_bb_auxi:Vn \l__graphics_page_int {#1} } 32443
2444     { \__graphics_extract_bb_auxii:nnn {#1} { } { } } 32444
2445 } 32445
2446 \cs_new_protected:Npn \__graphics_extract_bb_auxi:nn #1#2 32446
2447 { \__graphics_extract_bb_auxii:nnn {#2} { :P #1 } { -p~#1~ } } 32447
2448 \cs_generate_variant:Nn \__graphics_extract_bb_auxi:nn { Vn } 32448
2449 \cs_new_protected:Npn \__graphics_extract_bb_auxii:nnn #1#2#3 32449
2450 { 32450
2451     \tl_if_empty:NTF \l__graphics_pagebox_tl 32451
2452     { \__graphics_extract_bb_auxiv:nnn } 32452
2453     { \__graphics_extract_bb_auxiii:Vnnn \l__graphics_pagebox_tl } 32453
```



```
2454     {#1} {#2} {#3}
2455 }
2456 \cs_new_protected:Npn \__graphics_extract_bb_auxiii:nnnn #1#2#3#4
2457 { \__graphics_extract_bb_auxiv:nnn {#2} { : #1 #3 } { #4 -B~#1~ } }
2458 \cs_generate_variant:Nn \__graphics_extract_bb_auxiii:nnnn { V }
2459 \cs_new_protected:Npn \__graphics_extract_bb_auxiv:nnn #1#2#3
2460 {
2461     \__graphics_read_bb_auxi:nnnn {#1} {#2}
2462     { \ior_shell_open:Nn \l__graphics_internal_ior { extractbb~#3-0~#1 } }
2463     { pipe-failed }
2464 }
2465 \cs_new_protected:Npn \__graphics_read_bb:n #1
2466 {
2467     \__graphics_read_bb_auxi:nnnn {#1} { }
2468     { \ior_open:Nn \l__graphics_internal_ior {#1} }
2469     { graphic-not-found }
2470 }
2471 \cs_new_protected:Npn \__graphics_read_bb_auxi:nnnn #1#2#3#4
2472 {
2473     \__graphics_bb_restore:nF {#1#2}
2474     { \__graphics_read_bb_auxii:nnnn {#3} {#4} {#1} {#2} }
2475 }
2476 \cs_new_protected:Npe \__graphics_read_bb_auxii:nnnn #1#2#3#4
2477 {
2478     #1
2479     \exp_not:N \ior_if_eof:NTF \exp_not:N \l__graphics_internal_ior
2480     { \msg_error:nnn { graphics } {#2} {#3} }
2481     {
2482         \ior_str_map_inline:Nn \exp_not:N \l__graphics_internal_ior
2483         {
2484             \exp_not:N \__graphics_read_bb_auxiii:w
2485             ##1 ~ \c_colon_str \s__graphics_stop
2486         }
2487         \__graphics_bb_save:n {#3#4}
2488     }
2489     \ior_close:N \exp_not:N \l__graphics_internal_ior
2490 }
2491 \use:e
2492 {
2493     \cs_new_protected:Npn \exp_not:N \__graphics_read_bb_auxiii:w
2494     #1 \c_colon_str #2 \s__graphics_stop
2495     {
2496         \exp_not:N \str_if_eq:nnT
2497         { \c_percent_str \c_percent_str BoundingBox }
2498         {#1}
2499         { \exp_not:N \__graphics_read_bb_auxiv:w #2 ( ) \s__graphics_stop }
```

```

2500     }
2501 }
2502 \cs_new_protected:Npn \__graphics_read_bb_auxiv:w #1 ( #2 ) #3 \s__graphics_stop
2503 {
2504     \str_if_eq:nnF {#2} { atend }
2505     {
2506         \tl_set_rescan:Nne \l__graphics_internal_tl
2507         {
2508             \char_set_catcode_space:n { 9 }
2509             \char_set_catcode_space:n { 32 }
2510         }
2511         { \use:n #1 }
2512         \exp_after:wN \__graphics_read_bb_auxv:w \l__graphics_internal_tl
2513     }
2514 }
2515 \cs_new_protected:Npn \__graphics_read_bb_auxv:w #1~#2~#3~#4~#5 \s__graphics_stop
2516 {
2517     \dim_set:Nn \l__graphics_llx_dim { #1 bp }
2518     \dim_set:Nn \l__graphics_lly_dim { #2 bp }
2519     \dim_set:Nn \l__graphics_urx_dim { #3 bp }
2520     \dim_set:Nn \l__graphics_ury_dim { #4 bp }
2521     \ior_map_break:
2522 }
2523 \str_new:N \l__graphics_final_name_str
2524 \str_new:N \l__graphics_full_name_str
2525 \box_new:N \l__graphics_internal_box
2526 \str_new:N \l__graphics_dir_str
2527 \str_new:N \l__graphics_name_str
2528 \str_new:N \l__graphics_ext_str
2529 \seq_new:N \l__graphics_search_path_seq
2530 \seq_new:N \l__graphics_search_ext_seq
2531 \prop_new:N \l__graphics_ext_type_prop
2532 \prop_put:Nnn \l__graphics_ext_type_prop { .ps } { eps }
2533 \seq_new:N \g__graphics_record_seq
2534 \cs_new_protected:Npn \graphics_include:nn #1#2
2535 {
2536     \group_begin:
2537     \keys_set:nn { graphics } {#1}
2538     \seq_set_eq:NN \l_file_search_path_seq \l__graphics_search_path_seq
2539     \file_get_full_name:nNTF {#2} \l__graphics_full_name_str
2540     {
2541         \str_if_eq:eeTF { \l__graphics_full_name_str } { #2 .tex }
2542         { \msg_error:nnn { graphics } { graphic-not-found } {#2} }
2543         { \__graphics_include: }
2544     }

```

```
2545 { \msg_error:nnn { graphics } { graphic-not-found } {#2} } 3254
2546 \group_end: 3254
2547 } 3254
2548 \cs_generate_variant:Nn \graphics_include:nn { nV } 3254
2549 \cs_new_protected:Npn \__graphics_include: 3254
2550 { 3255
2551 \str_if_empty:NTF \l__graphics_type_str 3255
2552 { 3255
2553 \file_parse_full_name:VNNN \l__graphics_full_name_str 3255
2554 \l__graphics_dir_str \l__graphics_name_str \l__graphics_ext_str 3255
2555 \__graphics_include_auxi:e 3255
2556 { 3255
2557 \exp_args:Ne \str_tail:n 3255
2558 { \str_casefold:V \l__graphics_ext_str } 3255
2559 } 3255
2560 } 3256
2561 { \__graphics_include_auxi:e { \l__graphics_type_str } } 3256
2562 } 3256
2563 \cs_new_protected:Npn \__graphics_include_auxi:n #1 3256
2564 { 3256
2565 \prop_get:NnNF \l_graphics_ext_type_prop { .#1 } \l__graphics_internal_tl 3256
2566 { \tl_set:Nn \l__graphics_internal_tl {#1} } 3256
2567 \exp_args:NV \__graphics_include_auxii:n \l__graphics_internal_tl 3256
2568 } 3256
2569 \cs_generate_variant:Nn \__graphics_include_auxi:n { e } 3256
2570 \cs_new_protected:Npn \__graphics_include_auxii:n #1 3257
2571 { 3257
2572 \mode_leave_vertical: 3257
2573 \cs_if_exist:cTF { __graphics_backend_include_ #1 :n } 3257
2574 { 3257
2575 \tl_set_eq:NN \l__graphics_final_name_str \l__graphics_full_name_str 3257
2576 \str_set:Ne \l__graphics_full_name_str 3257
2577 { \exp_args:NV \__kernel_file_name_quote:n \l__graphics_full_name_str } 3257
2578 \exp_args:NnV \use:c { __graphics_backend_getbb_ #1 :n } 3257
2579 \l__graphics_full_name_str 3257
2580 \seq_gput_right:NV \g__graphics_record_seq \l__graphics_final_name_str 3258
2581 \clist_if_exist:NT \@filelist 3258
2582 { \exp_args:NV \@addtofilelist \l__graphics_final_name_str } 3258
2583 \bool_if:NTF \l__graphics_draft_bool 3258
2584 { \__graphics_include_auxiii:n } 3258
2585 { \__graphics_include_auxiv:n } 3258
2586 {#1} 3258
2587 } 3258
2588 { \msg_error:nnn { graphics } { unsupported-graphic-type } {#1} } 3258
2589 } 3258
2590 \cs_new_protected:Npn \__graphics_include_auxiii:n #1 3259
```

```

2591 {
2592   \hbox_to_wd:nn { \l__graphics_urx_dim - \l__graphics_llx_dim }
2593   {
2594     \tex_vrule:D
2595     \tex_hss:D
2596     \vbox_to_ht:nn
2597       { \l__graphics_ury_dim - \l__graphics_lly_dim }
2598       {
2599         \tex_hrulerule:D width
2600           \dim_eval:n { \l__graphics_urx_dim - \l__graphics_llx_dim }
2601         \tex_vss:D
2602         \hbox_to_wd:nn
2603           { \l__graphics_urx_dim - \l__graphics_llx_dim }
2604           {
2605             \ttfamily
2606             \tex_hss:D \l__graphics_full_name_str \tex_hss:D
2607           }
2608         \tex_vss:D
2609         \tex_hrulerule:D
2610       }
2611     \tex_hss:D
2612     \tex_vrule:D
2613   }
2614 }
2615 \cs_new_protected:Npn \__graphics_include_auxiv:n #1
2616 {
2617   \hbox_set:Nn \l__graphics_internal_box
2618   {
2619     \exp_args:NnV \use:c { __graphics_backend_include_ #1 :n }
2620     \l__graphics_full_name_str
2621   }
2622   \box_set_dp:Nn \l__graphics_internal_box { Opt }
2623   \box_set_ht:Nn \l__graphics_internal_box
2624     { \l__graphics_ury_dim - \l__graphics_lly_dim }
2625   \box_set_wd:Nn \l__graphics_internal_box
2626     { \l__graphics_urx_dim - \l__graphics_llx_dim }
2627   \box_use_drop:N \l__graphics_internal_box
2628 }
2629 \cs_new_protected:Npn \graphics_show_list: { \__graphics_list:N \msg_show:nneeee }
2630 \cs_new_protected:Npn \graphics_log_list: { \__graphics_list:N \msg_log:nneeee }
2631 \cs_new_protected:Npn \__graphics_list:N #1
2632 {
2633   \seq_remove_duplicates:N \g__graphics_record_seq
2634   #1 { kernel } { file-list }
2635   { \seq_map_function:NN \g__graphics_record_seq \__graphics_list_aux:n }
2636   { } { } { } { }

```

```

2637 }
2638 \cs_new:Npn \__graphics_list_aux:n #1 { \iow_newline: #1 }
2639 \cs_new_protected:Npn \graphics_get_full_name:nN #1#2
2640 {
2641   \graphics_get_full_name:nNF {#1} #2
2642   { \tl_set:Nn #2 { \q_no_value } }
2643 }
2644 \prg_new_protected_conditional:Npnn \graphics_get_full_name:nN #1#2
2645 { T , F , TF }
2646 {
2647   \group_begin:
2648     \seq_set_eq:NN \l_file_search_path_seq \l_graphics_search_path_seq
2649     \file_get_full_name:nNTF {#1} \l__graphics_full_name_str
2650     {
2651       \str_if_eq:eeTF { \l__graphics_full_name_str } { #1 .tex }
2652       { \__graphics_get_full_name:n {#1} }
2653       {
2654         \file_parse_full_name:VNNN \l__graphics_full_name_str
2655         \l__graphics_dir_str \l__graphics_name_str \l__graphics_ext_str
2656         \seq_map_inline:Nn \l_graphics_search_ext_seq
2657         {
2658           \str_if_eq:nVT {##1} \l__graphics_ext_str
2659           { \seq_map_break:n { \use_none:nn } }
2660         }
2661         \__graphics_get_full_name:n {#1}
2662       }
2663     }
2664     { \__graphics_get_full_name:n {#1} }
2665   \exp_args:NNNV \group_end:
2666   \tl_set:Nn #2 \l__graphics_full_name_str
2667   \tl_if_empty:NTF #2
2668   { \prg_return_false: }
2669   { \prg_return_true: }
2670 }
2671 \cs_new_protected:Npn \__graphics_get_full_name:n #1
2672 {
2673   \str_clear:N \l__graphics_full_name_str
2674   \seq_map_inline:Nn \l_graphics_search_ext_seq
2675   {
2676     \file_get_full_name:nNT { #1 ##1 } \l__graphics_full_name_str
2677     { \seq_map_break:n { \use_none:nn } }
2678   }
2679   \use:n
2680   { \str_clear:N \l__graphics_full_name_str }
2681 }
2682 \cs_new_protected:Npn \graphics_get_pagecount:nN #1#2

```

```

2683 {
2684 \group_begin:
2685 \seq_set_eq:NN \l_file_search_path_seq \l_graphics_search_path_seq
2686 \file_get_full_name:nNTF {#1} \l__graphics_full_name_str
2687 {
2688 \int_if_exist:cF { c__graphics_ \l__graphics_full_name_str _pages_int }
2689 {
2690 \exp_args:NV \__graphics_backend_get_pagecount:n
2691 \l__graphics_full_name_str
2692 }
2693 \tl_set:Nv #2 { c__graphics_ \l__graphics_full_name_str _pages_int }
2694 }
2695 {
2696 \tl_set:Nn #2 { 0 }
2697 \msg_error:nnn { graphics } { graphic-not-found } {#1}
2698 }
2699 \exp_args:NNNV \group_end:
2700 \tl_set:Nn #2 #2
2701 }
2702 \cs_new_protected:Npe \__graphics_get_pagecount:n #1
2703 {
2704 \exp_not:N \ior_shell_open:Nn \exp_not:N \l__graphics_internal_ior
2705 { extractbb~-0~#1 }
2706 \exp_not:N \ior_if_eof:NTF \exp_not:N \l__graphics_internal_ior
2707 { \msg_error:nnn { graphics } { pipe-failed } }
2708 {
2709 \ior_str_map_inline:Nn \exp_not:N \l__graphics_internal_ior
2710 {
2711 \exp_not:N \__graphics_get_pagecount:nw {#1}
2712 ##1 ~ \c_colon_str \c_colon_str \s__graphics_stop
2713 }
2714 \exp_not:N \int_if_exist:cF { c__graphics_ #1 _pages_int }
2715 { \int_const:cn { c__graphics_ #1 _pages_int } { 1 } }
2716 }
2717 \ior_close:N \exp_not:N \l__graphics_internal_ior
2718 }
2719 \use:e
2720 {
2721 \cs_new_protected:Npn \exp_not:N \__graphics_get_pagecount:nw
2722 #1#2 \c_colon_str #3 \c_colon_str #4 \s__graphics_stop
2723 {
2724 \exp_not:N \str_if_eq:nnT
2725 { \c_percent_str \c_percent_str Pages }
2726 {#2}
2727 {
2728 \int_const:cn { c__graphics_ #1 _pages_int } {#3}

```



```

2729 \exp_not:N \ior_map_break:
2730 }
2731 }
2732 }
2733 \msg_new:nnnn { graphics } { graphic-not-found }
2734 { Image~file~'#1'~not~found. }
2735 {
2736 LaTeX~tried~to~open~graphic~file~'#1',~
2737 but~the~file~could~not~be~read.
2738 }
2739 \msg_new:nnnn { graphics } { pipe-failed }
2740 { Cannot~run~piped~system~commands. }
2741 {
2742 LaTeX~tried~to~call~a~system~process~but~this~was~not~possible.\\
2743 Try~the~"--shell-escape"~(or~"--enable-pipes")~option.
2744 }
2745 \msg_new:nnnn { graphics } { unsupported-graphic-type }
2746 { Image~type~'#1'~not~supported~by~current~driver. }
2747 {
2748 LaTeX~was~asked~to~include~an~graphic~of~type~'#1',~
2749 but~this~is~not~supported~by~the~current~driver~(production~route).
2750 }
2751 %% File: l3opacity.dtx
2752 \cs_if_exist:NT \@expl@finalise@setup@@
2753 {
2754 \tl_gput_right:Nn \@expl@finalise@setup@@
2755 { \declare@file@substitution { l3opacity.sty } { null.tex } }
2756 }
2757 \fp_new:N \l__opacity_tmp_fp
2758 \cs_new_protected:Npn \opacity_select:n #1
2759 { \__opacity_select:nN {#1} \__opacity_backend_select:n }
2760 \cs_new_protected:Npn \opacity_fill:n #1
2761 { \__opacity_select:nN {#1} \__opacity_backend_fill:n }
2762 \cs_new_protected:Npn \opacity_stroke:n #1
2763 { \__opacity_select:nN {#1} \__opacity_backend_stroke:n }
2764 \cs_new_protected:Npn \__opacity_select:nN #1#2
2765 {
2766 \fp_set:Nn \l__opacity_tmp_fp { #1 }
2767 \bool_lazy_or:nnTF
2768 { \fp_compare_p:nNn \l__opacity_tmp_fp < \c_zero_fp }
2769 { \fp_compare_p:nNn \l__opacity_tmp_fp > \c_one_fp }
2770 { \msg_error:nnn { opacity } { out-of-range } {#1} }
2771 { \exp_args:Ne #2 { \fp_use:N \l__opacity_tmp_fp } }
2772 }
2773 \msg_new:nnnn { opacity } { out-of-range }
2774 { Opacity~value~out~of~range. }

```



```
2775 {
2776 LaTeX~was~asked~to~set~opacity~of~#1,~but~only~values~in~the~range~
2777 0~to~1~are~supported.
2778 }
2779 %% File: l3pdf.dtx
2780 \scan_new:N \s__pdf_stop
2781 \bool_new:N \g__pdf_init_bool
2782 \bool_lazy_and:nnT
2783 { \str_if_eq_p:Vn \fmtname { LaTeX2e } }
2784 { \tl_if_exist_p:N \@expl@finalise@setup@@ }
2785 {
2786 \tl_gput_right:Nn \@expl@finalise@setup@@
2787 {
2788 \tl_gput_right:Nn \@kernel@after@begindocument
2789 { \bool_gset_true:N \g__pdf_init_bool }
2790 }
2791 }
2792 \cs_new_protected:Npn \pdf_uncompress:
2793 {
2794 \bool_if:NF \g__pdf_init_bool
2795 {
2796 \__pdf_backend_compresslevel:n { 0 }
2797 \__pdf_backend_compress_objects:n { \c_false_bool }
2798 }
2799 }
2800 \int_new:N \g__pdf_backend_object_int
2801 \cs_new_protected:Npn \pdf_object_new:n #1
2802 {
2803 \__pdf_backend_object_new:
2804 \__pdf_object_record:nN {#1} \g__pdf_backend_object_int
2805 }
2806 \cs_new_protected:Npn \pdf_object_write:nnn #1#2#3
2807 {
2808 \exp_args:Ne \__pdf_backend_object_write:nnn
2809 { \__pdf_object_retrieve:n {#1} } {#2} {#3}
2810 \bool_gset_true:N \g__pdf_init_bool
2811 }
2812 \cs_generate_variant:Nn \pdf_object_write:nnn { nne , nnx }
2813 \cs_new:Npn \pdf_object_ref:n #1
2814 {
2815 \exp_args:Ne \__pdf_backend_object_ref:n
2816 { \__pdf_object_retrieve:n {#1} }
2817 }
2818 \cs_new:Npn \__kernel_pdf_object_id:n #1
2819 {
2820 \exp_args:Ne \__pdf_backend_object_id:n
```

```
2821 { \__pdf_object_retrieve:n {#1} }
2822 }
2823 \sys_if_engine luatex:F
2824 {
2825   \cs_new_protected:Npn \__pdf_object_record:nN #1#2
2826   {
2827     \int_const:cn
2828     { c__pdf_object_ #1 _int } {#2}
2829   }
2830   \cs_new:Npn \__pdf_object_retrieve:n #1
2831   {
2832     \int_if_exist:cTF { c__pdf_object_ #1 _int }
2833     {
2834       \int_use:c
2835       { c__pdf_object_ #1 _int }
2836     }
2837     { 0 }
2838   }
2839 }
2840 \prg_new_conditional:Npnn \pdf_object_if_exist:n #1 { p , T , F , TF }
2841 {
2842   \int_compare:nNnTF { \__pdf_object_retrieve:n {#1} } = 0
2843   \prg_return_false:
2844   \prg_return_true:
2845 }
2846 \cs_new_protected:Npn \pdf_object_new_indexed:nn #1#2
2847 {
2848   \__pdf_backend_object_new:
2849   \__pdf_object_record:neN {#1}
2850   { \int_eval:n {#2} } \g__pdf_backend_object_int
2851 }
2852 \cs_new_protected:Npn \pdf_object_write_indexed:nnnn #1#2#3#4
2853 {
2854   \exp_args:Ne \__pdf_backend_object_write:nnn
2855   { \__pdf_object_retrieve:ne {#1} { \int_eval:n {#2} } } {#3} {#4}
2856   \bool_gset_true:N \g__pdf_init_bool
2857 }
2858 \cs_generate_variant:Nn \pdf_object_write_indexed:nnnn { nnne }
2859 \cs_new:Npn \pdf_object_ref_indexed:nn #1#2
2860 {
2861   \exp_args:Ne \__pdf_backend_object_ref:n
2862   { \__pdf_object_retrieve:ne {#1} { \int_eval:n {#2} } }
2863 }
2864 \cs_new:Npn \__kernel_pdf_object_id_indexed:nn #1#2
2865 {
2866   \exp_args:Ne \__pdf_backend_object_id:n
```

```
2867 { \_pdf_object_retrieve:ne {#1} { \int_eval:n {#2} } } 3286
2868 } 3286
2869 \sys_if_engine luatex:F 3286
2870 { 3287
2871   \cs_new_protected:Npn \_pdf_object_record:nnN #1#2#3 3287
2872   { 3287
2873     \use:e 3287
2874     { 3287
2875       \_pdf_object_record:NnN 3287
2876       \_pdf_object_index_split:nn {#1} {#2} 3287
2877       \exp_not:N #3 3287
2878     } 3287
2879   } 3287
2880 \cs_new_protected:Npn \_pdf_object_record:NnN #1#2#3 3288
2881 { 3288
2882   \intarray_if_exist:NF #1 3288
2883   { \intarray_new:Nn #1 \c_pdf_object_block_size_int } 3288
2884   \intarray_gset:Nnn #1 {#2} #3 3288
2885 } 3288
2886 \cs_new:Npn \_pdf_object_retrieve:nn #1#2 3288
2887 { 3288
2888   \use:e 3288
2889   { 3288
2890     \exp_not:N \_pdf_object_retrieve:Nn 3289
2891     \_pdf_object_index_split:nn {#1} {#2} 3289
2892   } 3289
2893 } 3289
2894 \cs_new:Npn \_pdf_object_retrieve:Nn #1#2 3289
2895 { \intarray_item:Nn #1 {#2} } 3289
2896 \cs_new:Npn \_pdf_object_index_split:nn #1#2 3289
2897 { 3289
2898   \exp_not:c 3289
2899   { 3289
2900     g__pdf_object_ #1 _ 3290
2901     \int_eval:n 3290
2902     { 3290
2903       \int_div_truncate:nn { #2 - 1 } 3290
2904       \c_pdf_object_block_size_int + 1 3290
2905     } 3290
2906     _intarray 3290
2907   } 3290
2908   { 3290
2909     \int_eval:n 3290
2910     { \int_mod:nn { #2 - 1 } \c_pdf_object_block_size_int + 1 } 3291
2911   } 3291
2912 } 3291
```

```
2913 \int_const:Nn \c__pdf_object_block_size_int { 10000 }
2914 }
2915 \cs_generate_variant:Nn \__pdf_object_record:nnN { ne }
2916 \cs_generate_variant:Nn \__pdf_object_retrieve:nn { ne }
2917 \cs_new_protected:Npn \pdf_object_unnamed_write:nn #1#2
2918 {
2919   \exp_args:Ne \__pdf_backend_object_now:nn {#1} {#2}
2920   \bool_gset_true:N \g__pdf_init_bool
2921 }
2922 \cs_generate_variant:Nn \pdf_object_unnamed_write:nn { ne , nx }
2923 \cs_new:Npn \pdf_object_ref_last: { \__pdf_backend_object_last: }
2924 \cs_new:Npn \pdf_pageobject_ref:n #1
2925 { \exp_args:Ne \__pdf_backend_pageobject_ref:n {#1} }
2926 \prg_new_conditional:Npnn \pdf_version_compare:Nn #1#2 { p , T , F , TF }
2927 { \use:c { __pdf_version_compare_ #1 :w } #2 . . \s__pdf_stop }
2928 \cs_new:cpn { __pdf_version_compare_=w } #1 . #2 . #3 \s__pdf_stop
2929 {
2930   \bool_lazy_and:nnTF
2931     { \int_compare_p:nNn \__pdf_backend_version_major: = {#1} }
2932     { \int_compare_p:nNn \__pdf_backend_version_minor: = {#2} }
2933     { \prg_return_true: }
2934     { \prg_return_false: }
2935 }
2936 \cs_new:cpn { __pdf_version_compare_<:w } #1 . #2 . #3 \s__pdf_stop
2937 {
2938   \bool_lazy_or:nnTF
2939     { \int_compare_p:nNn \__pdf_backend_version_major: < {#1} }
2940     {
2941       \bool_lazy_and_p:nn
2942         { \int_compare_p:nNn \__pdf_backend_version_major: = {#1} }
2943         { \int_compare_p:nNn \__pdf_backend_version_minor: < {#2} }
2944     }
2945     { \prg_return_true: }
2946     { \prg_return_false: }
2947 }
2948 \cs_new:cpn { __pdf_version_compare_>:w } #1 . #2 . #3 \s__pdf_stop
2949 {
2950   \bool_lazy_or:nnTF
2951     { \int_compare_p:nNn \__pdf_backend_version_major: > {#1} }
2952     {
2953       \bool_lazy_and_p:nn
2954         { \int_compare_p:nNn \__pdf_backend_version_major: = {#1} }
2955         { \int_compare_p:nNn \__pdf_backend_version_minor: > {#2} }
2956     }
2957     { \prg_return_true: }
2958     { \prg_return_false: }
```

```

2959 }
2960 \cs_new_protected:Npn \pdf_version_gset:n #1
2961 { \__pdf_version_gset:w #1 . . \s__pdf_stop }
2962 \cs_new_protected:Npn \pdf_version_min_gset:n #1
2963 {
2964   \pdf_version_compare:NnT < {#1}
2965   { \__pdf_version_gset:w #1 . . \s__pdf_stop }
2966 }
2967 \cs_new_protected:Npn \__pdf_version_gset:w #1 . #2 . #3\s__pdf_stop
2968 {
2969   \bool_if:NF \g__pdf_init_bool
2970   {
2971     \__pdf_backend_version_major_gset:n {#1}
2972     \__pdf_backend_version_minor_gset:n {#2}
2973   }
2974 }
2975 \cs_new:Npn \pdf_version:
2976 { \__pdf_backend_version_major: . \__pdf_backend_version_minor: }
2977 \cs_new:Npn \pdf_version_major: { \__pdf_backend_version_major: }
2978 \cs_new:Npn \pdf_version_minor: { \__pdf_backend_version_minor: }
2979 \cs_new_protected:Npn \pdf_pagesize_gset:nn #1#2
2980 { \__pdf_backend_pagesize_gset:nn {#1} {#2} }
2981 \cs_new_protected:Npn \pdf_destination:nn #1#2
2982 { \__pdf_backend_destination:nn {#1} {#2} }
2983 \cs_new_protected:Npn \pdf_destination:nnnn #1#2#3#4
2984 {
2985   \hbox_to_zero:n
2986   { \__pdf_backend_destination:nnnn {#1} {#2} {#3} {#4} }
2987 }
2988 \cs_if_exist:NT \@kernel@before@begindocument
2989 {
2990   \tl_gput_right:Nn \@kernel@before@begindocument
2991   {
2992     \bool_lazy_all:nT
2993     {
2994       { \cs_if_exist_p:N \stockheight }
2995       { \cs_if_exist_p:N \stockwidth }
2996       { \cs_if_exist_p:N \IfDocumentMetadataTF }
2997       { \IfDocumentMetadataTF { \c_true_bool } { \c_false_bool } }
2998       { \int_compare_p:nNn \tex_mag:D = { 1000 } }
2999     }
3000   {
3001     \bool_lazy_and:nnTF
3002     { \dim_compare_p:nNn \stockheight > { 0pt } }
3003     { \dim_compare_p:nNn \stockwidth > { 0pt } }
3004     {

```

```
3005         \__pdf_backend_pagesize_gset:nn
3006         \stockwidth \stockheight
3007     }
3008     {
3009         \bool_lazy_or:nnF
3010         { \dim_compare_p:nNn \stockheight < { 0pt } }
3011         { \dim_compare_p:nNn \stockwidth < { 0pt } }
3012         {
3013             \bool_lazy_and:nnT
3014             { \dim_compare_p:nNn \paperheight > { 0pt } }
3015             { \dim_compare_p:nNn \paperwidth > { 0pt } }
3016             {
3017                 \__pdf_backend_pagesize_gset:nn
3018                 \paperwidth \paperheight
3019             }
3020         }
3021     }
3022 }
3023 }
3024 }
3025 %% File: l3coffins.dtx
3026 \box_new:N \l__coffin_internal_box
3027 \dim_new:N \l__coffin_internal_dim
3028 \tl_new:N \l__coffin_internal_tl
3029 \prop_const_from_keyval:Nn \c__coffin_corners_prop
3030 {
3031     tl = { 0pt } { 0pt } ,
3032     tr = { 0pt } { 0pt } ,
3033     bl = { 0pt } { 0pt } ,
3034     br = { 0pt } { 0pt } ,
3035 }
3036 \prop_const_from_keyval:Nn \c__coffin_poles_prop
3037 {
3038     l  = { 0pt } { 0pt } { 0pt } { 1000pt } ,
3039     hc = { 0pt } { 0pt } { 0pt } { 1000pt } ,
3040     r  = { 0pt } { 0pt } { 0pt } { 1000pt } ,
3041     b  = { 0pt } { 0pt } { 1000pt } { 0pt } ,
3042     vc = { 0pt } { 0pt } { 1000pt } { 0pt } ,
3043     t  = { 0pt } { 0pt } { 1000pt } { 0pt } ,
3044     B  = { 0pt } { 0pt } { 1000pt } { 0pt } ,
3045     H  = { 0pt } { 0pt } { 1000pt } { 0pt } ,
3046     T  = { 0pt } { 0pt } { 1000pt } { 0pt } ,
3047 }
3048 \fp_new:N \l__coffin_slope_A_fp
3049 \fp_new:N \l__coffin_slope_B_fp
3050 \bool_new:N \l__coffin_error_bool
```

```
3051 \dim_new:N \l__coffin_offset_x_dim
3052 \dim_new:N \l__coffin_offset_y_dim
3053 \tl_new:N \l__coffin_pole_a_tl
3054 \tl_new:N \l__coffin_pole_b_tl
3055 \dim_new:N \l__coffin_x_dim
3056 \dim_new:N \l__coffin_y_dim
3057 \dim_new:N \l__coffin_x_prime_dim
3058 \dim_new:N \l__coffin_y_prime_dim
3059 \cs_new_eq:NN \__coffin_to_value:N \tex_number:D
3060 \prg_new_conditional:Npnn \coffin_if_exist:N #1 { p , T , F , TF }
3061 {
3062   \cs_if_exist:NTF #1
3063   {
3064     \cs_if_exist:cTF { coffin ~ \__coffin_to_value:N #1 ~ poles }
3065     { \prg_return_true: }
3066     { \prg_return_false: }
3067   }
3068   { \prg_return_false: }
3069 }
3070 \prg_generate_conditional_variant:Nnn \coffin_if_exist:N
3071 { c } { p , T , F , TF }
3072 \cs_new_protected:Npn \__coffin_if_exist:NT #1#2
3073 {
3074   \coffin_if_exist:NTF #1
3075   { #2 }
3076   {
3077     \msg_error:nne { coffin } { unknown }
3078     { \token_to_str:N #1 }
3079   }
3080 }
3081 \cs_new_protected:Npn \coffin_clear:N #1
3082 {
3083   \__coffin_if_exist:NT #1
3084   {
3085     \box_clear:N #1
3086     \__coffin_reset_structure:N #1
3087   }
3088 }
3089 \cs_generate_variant:Nn \coffin_clear:N { c }
3090 \cs_new_protected:Npn \coffin_gclear:N #1
3091 {
3092   \__coffin_if_exist:NT #1
3093   {
3094     \box_gclear:N #1
3095     \__coffin_greset_structure:N #1
3096   }
```



```
3097 }
3098 \cs_generate_variant:Nn \coffin_gclear:N { c }
3099 \cs_new_protected:Npn \coffin_new:N #1
3100 {
3101   \box_new:N #1
3102   \debug_suspend:
3103   \prop_gclear_new:c { coffin ~ \__coffin_to_value:N #1 ~ corners }
3104   \prop_gclear_new:c { coffin ~ \__coffin_to_value:N #1 ~ poles }
3105   \prop_gset_eq:cN { coffin ~ \__coffin_to_value:N #1 ~ corners }
3106     \c__coffin_corners_prop
3107   \prop_gset_eq:cN { coffin ~ \__coffin_to_value:N #1 ~ poles }
3108     \c__coffin_poles_prop
3109   \debug_resume:
3110 }
3111 \cs_generate_variant:Nn \coffin_new:N { c }
3112 \cs_new_protected:Npn \hcoffin_set:Nn #1#2
3113 {
3114   \__coffin_if_exist:NT #1
3115   {
3116     \hbox_set:Nn #1
3117     {
3118       \color_ensure_current:
3119       #2
3120     }
3121     \coffin_reset_poles:N #1
3122   }
3123 }
3124 \cs_generate_variant:Nn \hcoffin_set:Nn { c }
3125 \cs_new_protected:Npn \hcoffin_gset:Nn #1#2
3126 {
3127   \__coffin_if_exist:NT #1
3128   {
3129     \hbox_gset:Nn #1
3130     {
3131       \color_ensure_current:
3132       #2
3133     }
3134     \coffin_greset_poles:N #1
3135   }
3136 }
3137 \cs_generate_variant:Nn \hcoffin_gset:Nn { c }
3138 \cs_new_protected:Npn \vcoffin_set:Nnn #1#2#3
3139 {
3140   \__coffin_set_vertical:NnnNNN #1 {#2} {#3}
3141   \vbox_set:Nn \coffin_reset_poles:N \__coffin_set_pole:Nnn
3142 }
```

```

3143 \cs_generate_variant:Nn \vcoffin_set:Nnn { c }
3144 \cs_new_protected:Npn \vcoffin_gset:Nnn #1#2#3
3145 {
3146   \__coffin_set_vertical:NnnNNN #1 {#2} {#3}
3147   \vbox_gset:Nn \coffin_greset_poles:N \__coffin_gset_pole:Nnn
3148 }
3149 \cs_generate_variant:Nn \vcoffin_gset:Nnn { c }
3150 \cs_new_protected:Npn \__coffin_set_vertical:NnnNNN #1#2#3#4#5#6
3151 {
3152   \__coffin_if_exist:NT #1
3153   {
3154     #4 #1
3155     {
3156       \dim_set:Nn \tex_hsize:D {#2}
3157       \__coffin_set_vertical_aux:
3158       #3
3159     }
3160     #5 #1
3161     \vbox_set_top:Nn \l__coffin_internal_box { \vbox_unpack:N #1 }
3162     #6 #1 { T }
3163     {
3164       { 0pt }
3165       {
3166         \dim_eval:n
3167         { \box_ht:N #1 - \box_ht:N \l__coffin_internal_box }
3168       }
3169       { 1000pt }
3170       { 0pt }
3171     }
3172     \box_clear:N \l__coffin_internal_box
3173   }
3174 }
3175 \cs_new_protected:Npe \__coffin_set_vertical_aux:
3176 {
3177   \bool_lazy_and:nnT
3178   { \cs_if_exist_p:N \fmtname }
3179   { \str_if_eq_p:Vn \fmtname { LaTeX2e } }
3180   {
3181     \dim_set_eq:NN \exp_not:N \linewidth \tex_hsize:D
3182     \dim_set_eq:NN \exp_not:N \columnwidth \tex_hsize:D
3183   }
3184 }
3185 \cs_new_protected:Npn \hcoffin_set:Nw #1
3186 {
3187   \__coffin_if_exist:NT #1
3188   {

```

3189	\hbox_set:Nw #1 \color_ensure_current:	33189
3190	\cs_set_protected:Npn \hcoffin_set_end:	33190
3191	{	33191
3192	\hbox_set_end:	33192
3193	\coffin_reset_poles:N #1	33193
3194	}	33194
3195	}	33195
3196	}	33196
3197	\cs_generate_variant:Nn \hcoffin_set:Nw { c }	33197
3198	\cs_new_protected:Npn \hcoffin_gset:Nw #1	33198
3199	{	33199
3200	__coffin_if_exist:NT #1	33200
3201	{	33201
3202	\hbox_gset:Nw #1 \color_ensure_current:	33202
3203	\cs_set_protected:Npn \hcoffin_gset_end:	33203
3204	{	33204
3205	\hbox_gset_end:	33205
3206	\coffin_greset_poles:N #1	33206
3207	}	33207
3208	}	33208
3209	}	33209
3210	\cs_generate_variant:Nn \hcoffin_gset:Nw { c }	33210
3211	\cs_new_protected:Npn \hcoffin_set_end: { }	33211
3212	\cs_new_protected:Npn \hcoffin_gset_end: { }	33212
3213	\cs_new_protected:Npn \vcoffin_set:Nnw #1#2	33213
3214	{	33214
3215	__coffin_set_vertical:NnNNNNNw #1 {#2} \vbox_set:Nw	33215
3216	\vcoffin_set_end:	33216
3217	\vbox_set_end: \coffin_reset_poles:N __coffin_set_pole:Nnn	33217
3218	}	33218
3219	\cs_generate_variant:Nn \vcoffin_set:Nnw { c }	33219
3220	\cs_new_protected:Npn \vcoffin_gset:Nnw #1#2	33220
3221	{	33221
3222	__coffin_set_vertical:NnNNNNNw #1 {#2} \vbox_gset:Nw	33222
3223	\vcoffin_gset_end:	33223
3224	\vbox_gset_end: \coffin_greset_poles:N __coffin_gset_pole:Nnn	33224
3225	}	33225
3226	\cs_generate_variant:Nn \vcoffin_gset:Nnw { c }	33226
3227	\cs_new_protected:Npn __coffin_set_vertical:NnNNNNNw #1#2#3#4#5#6#7	33227
3228	{	33228
3229	__coffin_if_exist:NT #1	33229
3230	{	33230
3231	#3 #1	33231
3232	\dim_set:Nn \tex_hsize:D {#2}	33232
3233	__coffin_set_vertical_aux:	33233
3234	\cs_set_protected:Npn #4	33234

```

3235         {
3236             #5
3237             #6 #1
3238             \vbox_set_top:Nn \l__coffin_internal_box { \vbox_unpack:N #1 }
3239             #7 #1 { T }
3240             {
3241                 { Opt }
3242                 {
3243                     \dim_eval:n
3244                     { \box_ht:N #1 - \box_ht:N \l__coffin_internal_box }
3245                 }
3246                 { 1000pt }
3247                 { Opt }
3248             }
3249             \box_clear:N \l__coffin_internal_box
3250         }
3251     }
3252 }
3253 \cs_new_protected:Npn \vcoffin_set_end: { }
3254 \cs_new_protected:Npn \vcoffin_gset_end: { }
3255 \cs_new_protected:Npn \coffin_set_eq:NN #1#2
3256 {
3257     \__coffin_if_exist:NT #2
3258     {
3259         \box_set_eq:NN #1 #2
3260         \prop_set_eq:cc { coffin ~ \__coffin_to_value:N #1 ~ corners }
3261         { coffin ~ \__coffin_to_value:N #2 ~ corners }
3262         \prop_set_eq:cc { coffin ~ \__coffin_to_value:N #1 ~ poles }
3263         { coffin ~ \__coffin_to_value:N #2 ~ poles }
3264     }
3265 }
3266 \cs_generate_variant:Nn \coffin_set_eq:NN { c , Nc , cc }
3267 \cs_new_protected:Npn \coffin_gset_eq:NN #1#2
3268 {
3269     \__coffin_if_exist:NT #2
3270     {
3271         \box_gset_eq:NN #1 #2
3272         \prop_gset_eq:cc { coffin ~ \__coffin_to_value:N #1 ~ corners }
3273         { coffin ~ \__coffin_to_value:N #2 ~ corners }
3274         \prop_gset_eq:cc { coffin ~ \__coffin_to_value:N #1 ~ poles }
3275         { coffin ~ \__coffin_to_value:N #2 ~ poles }
3276     }
3277 }
3278 \cs_generate_variant:Nn \coffin_gset_eq:NN { c , Nc , cc }
3279 \coffin_new:N \c_empty_coffin
3280 \coffin_new:N \l__coffin_aligned_coffin

```

```
3281 \coffin_new:N \l__coffin_aligned_internal_coffin
3282 \coffin_new:N \l_tmpa_coffin
3283 \coffin_new:N \l_tmpb_coffin
3284 \coffin_new:N \g_tmpa_coffin
3285 \coffin_new:N \g_tmpb_coffin
3286 \cs_new_eq:NN \coffin_dp:N \box_dp:N
3287 \cs_new_eq:NN \coffin_dp:c \box_dp:c
3288 \cs_new_eq:NN \coffin_ht:N \box_ht:N
3289 \cs_new_eq:NN \coffin_ht:c \box_ht:c
3290 \cs_new_eq:NN \coffin_ht_plus_dp:N \box_ht_plus_dp:N
3291 \cs_new_eq:NN \coffin_ht_plus_dp:c \box_ht_plus_dp:c
3292 \cs_new_eq:NN \coffin_wd:N \box_wd:N
3293 \cs_new_eq:NN \coffin_wd:c \box_wd:c
3294 \cs_new_protected:Npn \__coffin_get_pole:NnN #1#2#3
3295 {
3296   \prop_get:cnNF
3297     { coffin ~ \__coffin_to_value:N #1 ~ poles } {#2} #3
3298   {
3299     \msg_error:nnee { coffin } { unknown-pole }
3300     { \exp_not:n {#2} } { \token_to_str:N #1 }
3301     \tl_set:Nn #3 { { Opt } { Opt } { Opt } { Opt } }
3302   }
3303 }
3304 \cs_new_protected:Npn \__coffin_reset_structure:N #1
3305 {
3306   \prop_set_eq:cN { coffin ~ \__coffin_to_value:N #1 ~ corners }
3307   \c__coffin_corners_prop
3308   \prop_set_eq:cN { coffin ~ \__coffin_to_value:N #1 ~ poles }
3309   \c__coffin_poles_prop
3310 }
3311 \cs_new_protected:Npn \__coffin_greset_structure:N #1
3312 {
3313   \prop_gset_eq:cN { coffin ~ \__coffin_to_value:N #1 ~ corners }
3314   \c__coffin_corners_prop
3315   \prop_gset_eq:cN { coffin ~ \__coffin_to_value:N #1 ~ poles }
3316   \c__coffin_poles_prop
3317 }
3318 \cs_new_protected:Npn \coffin_set_horizontal_pole:Nnn #1#2#3
3319 { \__coffin_set_horizontal_pole:NnnN #1 {#2} {#3} \prop_put:cne }
3320 \cs_generate_variant:Nn \coffin_set_horizontal_pole:Nnn { c }
3321 \cs_new_protected:Npn \coffin_gset_horizontal_pole:Nnn #1#2#3
3322 { \__coffin_set_horizontal_pole:NnnN #1 {#2} {#3} \prop_gput:cne }
3323 \cs_generate_variant:Nn \coffin_gset_horizontal_pole:Nnn { c }
3324 \cs_new_protected:Npn \__coffin_set_horizontal_pole:NnnN #1#2#3#4
3325 {
3326   \__coffin_if_exist:NT #1
```

```

3327 {
3328     #4 { coffin ~ \__coffin_to_value:N #1 ~ poles }
3329     {#2}
3330     {
3331         { Opt } { \dim_eval:n {#3} }
3332         { 1000pt } { Opt }
3333     }
3334 }
3335 }
3336 \cs_new_protected:Npn \coffin_set_vertical_pole:Nnn #1#2#3
3337 { \__coffin_set_vertical_pole:NnnN #1 {#2} {#3} \prop_put:cne }
3338 \cs_generate_variant:Nn \coffin_set_vertical_pole:Nnn { c }
3339 \cs_new_protected:Npn \coffin_gset_vertical_pole:Nnn #1#2#3
3340 { \__coffin_set_vertical_pole:NnnN #1 {#2} {#3} \prop_gput:cne }
3341 \cs_generate_variant:Nn \coffin_gset_vertical_pole:Nnn { c }
3342 \cs_new_protected:Npn \__coffin_set_vertical_pole:NnnN #1#2#3#4
3343 {
3344     \__coffin_if_exist:NT #1
3345     {
3346         #4 { coffin ~ \__coffin_to_value:N #1 ~ poles }
3347         {#2}
3348         {
3349             { \dim_eval:n {#3} } { Opt }
3350             { Opt } { 1000pt }
3351         }
3352     }
3353 }
3354 \cs_new_protected:Npn \__coffin_set_pole:Nnn #1#2#3
3355 {
3356     \prop_put:cne { coffin ~ \__coffin_to_value:N #1 ~ poles }
3357     {#2} {#3}
3358 }
3359 \cs_new_protected:Npn \__coffin_gset_pole:Nnn #1#2#3
3360 {
3361     \prop_gput:cne { coffin ~ \__coffin_to_value:N #1 ~ poles }
3362     {#2} {#3}
3363 }
3364 \cs_new_protected:Npn \coffin_reset_poles:N #1
3365 {
3366     \__coffin_reset_structure:N #1
3367     \__coffin_update_corners:N #1
3368     \__coffin_update_poles:N #1
3369 }
3370 \cs_new_protected:Npn \coffin_greset_poles:N #1
3371 {
3372     \__coffin_greset_structure:N #1

```

```
3373 \__coffin_gupdate_corners:N #1
3374 \__coffin_gupdate_poles:N #1
3375 }
3376 \cs_new_protected:Npn \__coffin_update_corners:N #1
3377 { \__coffin_update_corners:NN #1 \prop_put:Nne }
3378 \cs_new_protected:Npn \__coffin_gupdate_corners:N #1
3379 { \__coffin_update_corners:NN #1 \prop_gput:Nne }
3380 \cs_new_protected:Npn \__coffin_update_corners:NN #1#2
3381 {
3382   \exp_args:Nc \__coffin_update_corners:NNN
3383     { coffin ~ \__coffin_to_value:N #1 ~ corners }
3384     #1 #2
3385 }
3386 \cs_new_protected:Npn \__coffin_update_corners:NNN #1#2#3
3387 {
3388   #3 #1
3389   { tl }
3390   { { Opt } { \dim_eval:n { \box_ht:N #2 } } } }
3391   #3 #1
3392   { tr }
3393   {
3394     { \dim_eval:n { \box_wd:N #2 } }
3395     { \dim_eval:n { \box_ht:N #2 } }
3396   }
3397   #3 #1
3398   { bl }
3399   { { Opt } { \dim_eval:n { -\box_dp:N #2 } } } }
3400   #3 #1
3401   { br }
3402   {
3403     { \dim_eval:n { \box_wd:N #2 } }
3404     { \dim_eval:n { -\box_dp:N #2 } }
3405   }
3406 }
3407 \cs_new_protected:Npn \__coffin_update_poles:N #1
3408 { \__coffin_update_poles:NN #1 \prop_put:Nne }
3409 \cs_new_protected:Npn \__coffin_gupdate_poles:N #1
3410 { \__coffin_update_poles:NN #1 \prop_gput:Nne }
3411 \cs_new_protected:Npn \__coffin_update_poles:NN #1#2
3412 {
3413   \exp_args:Nc \__coffin_update_poles:NNN
3414     { coffin ~ \__coffin_to_value:N #1 ~ poles }
3415     #1 #2
3416 }
3417 \cs_new_protected:Npn \__coffin_update_poles:NNN #1#2#3
3418 {
```



```
3419 #3 #1 { hc }
3420 {
3421     { \dim_eval:n { 0.5 \box_wd:N #2 } }
3422     { Opt } { Opt } { 1000pt }
3423 }
3424 #3 #1 { r }
3425 {
3426     { \dim_eval:n { \box_wd:N #2 } }
3427     { Opt } { Opt } { 1000pt }
3428 }
3429 #3 #1 { vc }
3430 {
3431     { Opt }
3432     { \dim_eval:n { ( \box_ht:N #2 - \box_dp:N #2 ) / 2 } }
3433     { 1000pt }
3434     { Opt }
3435 }
3436 #3 #1 { t }
3437 {
3438     { Opt }
3439     { \dim_eval:n { \box_ht:N #2 } }
3440     { 1000pt }
3441     { Opt }
3442 }
3443 #3 #1 { b }
3444 {
3445     { Opt }
3446     { \dim_eval:n { -\box_dp:N #2 } }
3447     { 1000pt }
3448     { Opt }
3449 }
3450 }
3451 \cs_new_protected:Npn \__coffin_calculate_intersection:Nnn #1#2#3
3452 {
3453     \__coffin_get_pole:NnN #1 {#2} \l__coffin_pole_a_tl
3454     \__coffin_get_pole:NnN #1 {#3} \l__coffin_pole_b_tl
3455     \bool_set_false:N \l__coffin_error_bool
3456     \exp_last_two_unbraced:Noo
3457         \__coffin_calculate_intersection:nnnnnnnn
3458         \l__coffin_pole_a_tl \l__coffin_pole_b_tl
3459     \bool_if:NT \l__coffin_error_bool
3460     {
3461         \msg_error:nn { coffin } { no-pole-intersection }
3462         \dim_zero:N \l__coffin_x_dim
3463         \dim_zero:N \l__coffin_y_dim
3464     }
```

```
3465 }
3466 \cs_new_protected:Npn \__coffin_calculate_intersection:nnnnnnnn
3467 #1#2#3#4#5#6#7#8
3468 {
3469   \dim_compare:nNnTF {#3} = \c_zero_dim
3470   {
3471     \dim_set:Nn \l__coffin_x_dim {#1}
3472     \dim_compare:nNnTF {#7} = \c_zero_dim
3473     { \bool_set_true:N \l__coffin_error_bool }
3474     {
3475       \dim_set:Nn \l__coffin_y_dim
3476       {
3477         \dim_compare:nNnTF {#8} = \c_zero_dim
3478         {#6}
3479         {
3480           \fp_to_dim:n
3481           {
3482             ( \dim_to_fp:n {#8} / \dim_to_fp:n {#7} )
3483             * ( \dim_to_fp:n {#1} - \dim_to_fp:n {#5} )
3484             + \dim_to_fp:n {#6}
3485           }
3486         }
3487       }
3488     }
3489   }
3490   {
3491     \dim_compare:nNnTF {#4} = \c_zero_dim
3492     {
3493       \dim_set:Nn \l__coffin_y_dim {#2}
3494       \dim_compare:nNnTF {#8} = { \c_zero_dim }
3495       { \bool_set_true:N \l__coffin_error_bool }
3496       {
3497         \dim_set:Nn \l__coffin_x_dim
3498         {
3499           \dim_compare:nNnTF {#7} = \c_zero_dim
3500           {#5}
3501           {
3502             \fp_to_dim:n
3503             {
3504               ( \dim_to_fp:n {#7} / \dim_to_fp:n {#8} )
3505               * ( \dim_to_fp:n {#4} - \dim_to_fp:n {#6} )
3506               + \dim_to_fp:n {#5}
3507             }
3508           }
3509         }
3510       }
3511     }
3512   }
}
```

```

3511     }
3512     {
3513         \use:e
3514         {
3515             \__coffin_calculate_intersection:nnnnnn
3516             { \dim_to_fp:n {#4} / \dim_to_fp:n {#3} }
3517             { \dim_to_fp:n {#8} / \dim_to_fp:n {#7} }
3518         }
3519         {#1} {#2} {#5} {#6}
3520     }
3521 }
3522 }
3523 \cs_new_protected:Npn \__coffin_calculate_intersection:nnnnnn #1#2#3#4#5#6
3524 {
3525     \fp_compare:nNnTF {#1} = {#2}
3526     { \bool_set_true:N \l__coffin_error_bool }
3527     {
3528         \dim_set:Nn \l__coffin_x_dim
3529         {
3530             \fp_to_dim:n
3531             {
3532                 (
3533                     #1 * \dim_to_fp:n {#3}
3534                     - #2 * \dim_to_fp:n {#5}
3535                     - \dim_to_fp:n {#4}
3536                     + \dim_to_fp:n {#6}
3537                 )
3538                 /
3539                 ( #1 - #2 )
3540             }
3541         }
3542         \dim_set:Nn \l__coffin_y_dim
3543         {
3544             \fp_to_dim:n
3545             {
3546                 #1 * ( \l__coffin_x_dim - \dim_to_fp:n {#3} )
3547                 + \dim_to_fp:n {#4}
3548             }
3549         }
3550     }
3551 }
3552 \fp_new:N \l__coffin_sin_fp
3553 \fp_new:N \l__coffin_cos_fp
3554 \prop_new:N \l__coffin_bounding_prop
3555 \prop_new:N \l__coffin_corners_prop
3556 \prop_new:N \l__coffin_poles_prop

```

```
3557 \dim_new:N \l__coffin_bounding_shift_dim
3558 \dim_new:N \l__coffin_left_corner_dim
3559 \dim_new:N \l__coffin_right_corner_dim
3560 \dim_new:N \l__coffin_bottom_corner_dim
3561 \dim_new:N \l__coffin_top_corner_dim
3562 \cs_new_protected:Npn \coffin_rotate:Nn #1#2
3563 { \__coffin_rotate:Nnnnn #1 {#2} \box_rotate:Nn \prop_set_eq:cN \hbox_set:Nn }
3564 \cs_generate_variant:Nn \coffin_rotate:Nn { c }
3565 \cs_new_protected:Npn \coffin_grotate:Nn #1#2
3566 { \__coffin_rotate:Nnnnn #1 {#2} \box_grotate:Nn \prop_gset_eq:cN \hbox_gset:Nn }
3567 \cs_generate_variant:Nn \coffin_grotate:Nn { c }
3568 \cs_new_protected:Npn \__coffin_rotate:Nnnnn #1#2#3#4#5
3569 {
3570   \fp_set:Nn \l__coffin_sin_fp { sind ( #2 ) }
3571   \fp_set:Nn \l__coffin_cos_fp { cosd ( #2 ) }
3572   \prop_set_eq:Nc \l__coffin_corners_prop
3573   { coffin ~ \__coffin_to_value:N #1 ~ corners }
3574   \prop_set_eq:Nc \l__coffin_poles_prop
3575   { coffin ~ \__coffin_to_value:N #1 ~ poles }
3576   \prop_map_inline:Nn \l__coffin_corners_prop
3577   { \__coffin_rotate_corner:Nnnn #1 {##1} ##2 }
3578   \prop_map_inline:Nn \l__coffin_poles_prop
3579   { \__coffin_rotate_pole:Nnnnnn #1 {##1} ##2 }
3580   \__coffin_set_bounding:N #1
3581   \prop_map_inline:Nn \l__coffin_bounding_prop
3582   { \__coffin_rotate_bounding:nnn {##1} ##2 }
3583   \__coffin_find_corner_maxima:N #1
3584   \__coffin_find_bounding_shift:
3585   #3 #1 {#2}
3586   \hbox_set:Nn \l__coffin_internal_box
3587   {
3588     \__kernel_kern:n
3589     { \l__coffin_bounding_shift_dim - \l__coffin_left_corner_dim }
3590     \box_move_down:nn { \l__coffin_bottom_corner_dim }
3591     { \box_use:N #1 }
3592   }
3593   \box_set_ht:Nn \l__coffin_internal_box
3594   { \l__coffin_top_corner_dim - \l__coffin_bottom_corner_dim }
3595   \box_set_dp:Nn \l__coffin_internal_box { Opt }
3596   \box_set_wd:Nn \l__coffin_internal_box
3597   { \l__coffin_right_corner_dim - \l__coffin_left_corner_dim }
3598   #5 #1 { \box_use_drop:N \l__coffin_internal_box }
3599   \prop_map_inline:Nn \l__coffin_corners_prop
3600   { \__coffin_shift_corner:Nnnn #1 {##1} ##2 }
3601   \prop_map_inline:Nn \l__coffin_poles_prop
3602   { \__coffin_shift_pole:Nnnnnn #1 {##1} ##2 }
```

```
3603 #4 { coffin ~ \__coffin_to_value:N #1 ~ corners } 33603
3604 \l__coffin_corners_prop 33604
3605 #4 { coffin ~ \__coffin_to_value:N #1 ~ poles } 33605
3606 \l__coffin_poles_prop 33606
3607 } 33607
3608 \cs_new_protected:Npn \__coffin_set_bounding:N #1 33608
3609 { 33609
3610 \prop_put:Nne \l__coffin_bounding_prop { tl } 33610
3611 { { Opt } { \dim_eval:n { \box_ht:N #1 } } } 33611
3612 \prop_put:Nne \l__coffin_bounding_prop { tr } 33612
3613 { 33613
3614 { \dim_eval:n { \box_wd:N #1 } } 33614
3615 { \dim_eval:n { \box_ht:N #1 } } 33615
3616 } 33616
3617 \dim_set:Nn \l__coffin_internal_dim { -\box_dp:N #1 } 33617
3618 \prop_put:Nne \l__coffin_bounding_prop { bl } 33618
3619 { { Opt } { \dim_use:N \l__coffin_internal_dim } } 33619
3620 \prop_put:Nne \l__coffin_bounding_prop { br } 33620
3621 { 33621
3622 { \dim_eval:n { \box_wd:N #1 } } 33622
3623 { \dim_use:N \l__coffin_internal_dim } 33623
3624 } 33624
3625 } 33625
3626 \cs_new_protected:Npn \__coffin_rotate_bounding:nnn #1#2#3 33626
3627 { 33627
3628 \__coffin_rotate_vector:nnNN {#2} {#3} \l__coffin_x_dim \l__coffin_y_dim 33628
3629 \prop_put:Nne \l__coffin_bounding_prop {#1} 33629
3630 { { \dim_use:N \l__coffin_x_dim } { \dim_use:N \l__coffin_y_dim } } 33630
3631 } 33631
3632 \cs_new_protected:Npn \__coffin_rotate_corner:Nnnn #1#2#3#4 33632
3633 { 33633
3634 \__coffin_rotate_vector:nnNN {#3} {#4} \l__coffin_x_dim \l__coffin_y_dim 33634
3635 \prop_put:Nne \l__coffin_corners_prop {#2} 33635
3636 { { \dim_use:N \l__coffin_x_dim } { \dim_use:N \l__coffin_y_dim } } 33636
3637 } 33637
3638 \cs_new_protected:Npn \__coffin_rotate_pole:Nnnnnn #1#2#3#4#5#6 33638
3639 { 33639
3640 \__coffin_rotate_vector:nnNN {#3} {#4} \l__coffin_x_dim \l__coffin_y_dim 33640
3641 \__coffin_rotate_vector:nnNN {#5} {#6} 33641
3642 \l__coffin_x_prime_dim \l__coffin_y_prime_dim 33642
3643 \prop_put:Nne \l__coffin_poles_prop {#2} 33643
3644 { 33644
3645 { \dim_use:N \l__coffin_x_dim } { \dim_use:N \l__coffin_y_dim } 33645
3646 { \dim_use:N \l__coffin_x_prime_dim } 33646
3647 { \dim_use:N \l__coffin_y_prime_dim } 33647
3648 } 33648
```

```

3649 }
3650 \cs_new_protected:Npn \__coffin_rotate_vector:nnNN #1#2#3#4
3651 {
3652   \dim_set:Nn #3
3653   {
3654     \fp_to_dim:n
3655     {
3656       \dim_to_fp:n {#1} * \l__coffin_cos_fp
3657       - \dim_to_fp:n {#2} * \l__coffin_sin_fp
3658     }
3659   }
3660   \dim_set:Nn #4
3661   {
3662     \fp_to_dim:n
3663     {
3664       \dim_to_fp:n {#1} * \l__coffin_sin_fp
3665       + \dim_to_fp:n {#2} * \l__coffin_cos_fp
3666     }
3667   }
3668 }
3669 \cs_new_protected:Npn \__coffin_find_corner_maxima:N #1
3670 {
3671   \dim_set:Nn \l__coffin_top_corner_dim { -\c_max_dim }
3672   \dim_set:Nn \l__coffin_right_corner_dim { -\c_max_dim }
3673   \dim_set:Nn \l__coffin_bottom_corner_dim { \c_max_dim }
3674   \dim_set:Nn \l__coffin_left_corner_dim { \c_max_dim }
3675   \prop_map_inline:Nn \l__coffin_corners_prop
3676   { \__coffin_find_corner_maxima_aux:nn ##2 }
3677 }
3678 \cs_new_protected:Npn \__coffin_find_corner_maxima_aux:nn #1#2
3679 {
3680   \dim_set:Nn \l__coffin_left_corner_dim
3681   { \dim_min:nn { \l__coffin_left_corner_dim } {#1} }
3682   \dim_set:Nn \l__coffin_right_corner_dim
3683   { \dim_max:nn { \l__coffin_right_corner_dim } {#1} }
3684   \dim_set:Nn \l__coffin_bottom_corner_dim
3685   { \dim_min:nn { \l__coffin_bottom_corner_dim } {#2} }
3686   \dim_set:Nn \l__coffin_top_corner_dim
3687   { \dim_max:nn { \l__coffin_top_corner_dim } {#2} }
3688 }
3689 \cs_new_protected:Npn \__coffin_find_bounding_shift:
3690 {
3691   \dim_set:Nn \l__coffin_bounding_shift_dim { \c_max_dim }
3692   \prop_map_inline:Nn \l__coffin_bounding_prop
3693   { \__coffin_find_bounding_shift_aux:nn ##2 }
3694 }

```

```
3695 \cs_new_protected:Npn \__coffin_find_bounding_shift_aux:nn #1#2
3696 {
3697     \dim_set:Nn \l__coffin_bounding_shift_dim
3698     { \dim_min:nn { \l__coffin_bounding_shift_dim } {#1} }
3699 }
3700 \cs_new_protected:Npn \__coffin_shift_corner:Nnnn #1#2#3#4
3701 {
3702     \prop_put:Nne \l__coffin_corners_prop {#2}
3703     {
3704         { \dim_eval:n { #3 - \l__coffin_left_corner_dim } }
3705         { \dim_eval:n { #4 - \l__coffin_bottom_corner_dim } }
3706     }
3707 }
3708 \cs_new_protected:Npn \__coffin_shift_pole:Nnnnnn #1#2#3#4#5#6
3709 {
3710     \prop_put:Nne \l__coffin_poles_prop {#2}
3711     {
3712         { \dim_eval:n { #3 - \l__coffin_left_corner_dim } }
3713         { \dim_eval:n { #4 - \l__coffin_bottom_corner_dim } }
3714         {#5} {#6}
3715     }
3716 }
3717 \fp_new:N \l__coffin_scale_x_fp
3718 \fp_new:N \l__coffin_scale_y_fp
3719 \dim_new:N \l__coffin_scaled_total_height_dim
3720 \dim_new:N \l__coffin_scaled_width_dim
3721 \cs_new_protected:Npn \coffin_resize:Nnn #1#2#3
3722 {
3723     \__coffin_resize:NnnNN #1 {#2} {#3}
3724     \box_resize_to_wd_and_ht_plus_dp:Nnn
3725     \prop_set_eq:cN
3726 }
3727 \cs_generate_variant:Nn \coffin_resize:Nnn { c }
3728 \cs_new_protected:Npn \coffin_gresize:Nnn #1#2#3
3729 {
3730     \__coffin_resize:NnnNN #1 {#2} {#3}
3731     \box_gresize_to_wd_and_ht_plus_dp:Nnn
3732     \prop_gset_eq:cN
3733 }
3734 \cs_generate_variant:Nn \coffin_gresize:Nnn { c }
3735 \cs_new_protected:Npn \__coffin_resize:NnnNN #1#2#3#4#5
3736 {
3737     \fp_set:Nn \l__coffin_scale_x_fp
3738     { \dim_to_fp:n {#2} / \dim_to_fp:n { \coffin_wd:N #1 } }
3739     \fp_set:Nn \l__coffin_scale_y_fp
3740     {
```



```

3741         \dim_to_fp:n {#3}
3742         / \dim_to_fp:n { \coffin_ht:N #1 + \coffin_dp:N #1 }
3743     }
3744     #4 #1 {#2} {#3}
3745     \__coffin_resize_common:NnnN #1 {#2} {#3} #5
3746 }
3747 \cs_new_protected:Npn \__coffin_resize_common:NnnN #1#2#3#4
3748 {
3749     \prop_set_eq:Nc \l__coffin_corners_prop
3750     { coffin ~ \__coffin_to_value:N #1 ~ corners }
3751     \prop_set_eq:Nc \l__coffin_poles_prop
3752     { coffin ~ \__coffin_to_value:N #1 ~ poles }
3753     \prop_map_inline:Nn \l__coffin_corners_prop
3754     { \__coffin_scale_corner:Nnnn #1 {##1} ##2 }
3755     \prop_map_inline:Nn \l__coffin_poles_prop
3756     { \__coffin_scale_pole:Nnnnnn #1 {##1} ##2 }
3757     \fp_compare:nNnT \l__coffin_scale_x_fp < \c_zero_fp
3758     {
3759         \prop_map_inline:Nn \l__coffin_corners_prop
3760         { \__coffin_x_shift_corner:Nnnn #1 {##1} ##2 }
3761         \prop_map_inline:Nn \l__coffin_poles_prop
3762         { \__coffin_x_shift_pole:Nnnnnn #1 {##1} ##2 }
3763     }
3764     #4 { coffin ~ \__coffin_to_value:N #1 ~ corners }
3765     \l__coffin_corners_prop
3766     #4 { coffin ~ \__coffin_to_value:N #1 ~ poles }
3767     \l__coffin_poles_prop
3768 }
3769 \cs_new_protected:Npn \coffin_scale:Nnn #1#2#3
3770 { \__coffin_scale:NnnNN #1 {#2} {#3} \box_scale:Nnn \prop_set_eq:cN }
3771 \cs_generate_variant:Nn \coffin_scale:Nnn { c }
3772 \cs_new_protected:Npn \coffin_gscale:Nnn #1#2#3
3773 { \__coffin_scale:NnnNN #1 {#2} {#3} \box_gscale:Nnn \prop_gset_eq:cN }
3774 \cs_generate_variant:Nn \coffin_gscale:Nnn { c }
3775 \cs_new_protected:Npn \__coffin_scale:NnnNN #1#2#3#4#5
3776 {
3777     \fp_set:Nn \l__coffin_scale_x_fp {#2}
3778     \fp_set:Nn \l__coffin_scale_y_fp {#3}
3779     #4 #1 { \l__coffin_scale_x_fp } { \l__coffin_scale_y_fp }
3780     \dim_set:Nn \l__coffin_internal_dim
3781     { \coffin_ht:N #1 + \coffin_dp:N #1 }
3782     \dim_set:Nn \l__coffin_scaled_total_height_dim
3783     { \fp_abs:n { \l__coffin_scale_y_fp } \l__coffin_internal_dim }
3784     \dim_set:Nn \l__coffin_scaled_width_dim
3785     { -\fp_abs:n { \l__coffin_scale_x_fp } \coffin_wd:N #1 }
3786     \__coffin_resize_common:NnnN #1

```

```
3787 { \l__coffin_scaled_width_dim } { \l__coffin_scaled_total_height_dim } 3378
3788 #5 3378
3789 } 3378
3790 \cs_new_protected:Npn \__coffin_scale_vector:nnNN #1#2#3#4 3379
3791 { 3379
3792   \dim_set:Nn #3 3379
3793   { \fp_to_dim:n { \dim_to_fp:n {#1} * \l__coffin_scale_x_fp } } 3379
3794   \dim_set:Nn #4 3379
3795   { \fp_to_dim:n { \dim_to_fp:n {#2} * \l__coffin_scale_y_fp } } 3379
3796 } 3379
3797 \cs_new_protected:Npn \__coffin_scale_corner:Nnnn #1#2#3#4 3379
3798 { 3379
3799   \__coffin_scale_vector:nnNN {#3} {#4} \l__coffin_x_dim \l__coffin_y_dim 3379
3800   \prop_put:Nne \l__coffin_corners_prop {#2} 3380
3801   { { \dim_use:N \l__coffin_x_dim } { \dim_use:N \l__coffin_y_dim } } 3380
3802 } 3380
3803 \cs_new_protected:Npn \__coffin_scale_pole:Nnnnnn #1#2#3#4#5#6 3380
3804 { 3380
3805   \__coffin_scale_vector:nnNN {#3} {#4} \l__coffin_x_dim \l__coffin_y_dim 3380
3806   \prop_put:Nne \l__coffin_poles_prop {#2} 3380
3807   { 3380
3808     { \dim_use:N \l__coffin_x_dim } { \dim_use:N \l__coffin_y_dim } 3380
3809     {#5} {#6} 3380
3810   } 3381
3811 } 3381
3812 \cs_new_protected:Npn \__coffin_x_shift_corner:Nnnn #1#2#3#4 3381
3813 { 3381
3814   \prop_put:Nne \l__coffin_corners_prop {#2} 3381
3815   { 3381
3816     { \dim_eval:n { #3 + \box_wd:N #1 } } {#4} 3381
3817   } 3381
3818 } 3381
3819 \cs_new_protected:Npn \__coffin_x_shift_pole:Nnnnnn #1#2#3#4#5#6 3381
3820 { 3382
3821   \prop_put:Nne \l__coffin_poles_prop {#2} 3382
3822   { 3382
3823     { \dim_eval:n { #3 + \box_wd:N #1 } } {#4} 3382
3824     {#5} {#6} 3382
3825   } 3382
3826 } 3382
3827 \cs_new_protected:Npn \coffin_join:NnnNnnnn #1#2#3#4#5#6#7#8 3382
3828 { 3382
3829   \__coffin_join:NnnNnnnnN #1 {#2} {#3} #4 {#5} {#6} {#7} {#8} 3382
3830   \coffin_set_eq:NN 3383
3831 } 3383
3832 \cs_generate_variant:Nn \coffin_join:NnnNnnnn { c , Nnnc , cnnc } 3383
```

```
3833 \cs_new_protected:Npn \coffin_gjoin:NnnNnnnn #1#2#3#4#5#6#7#8 3383
3834 { 3383
3835 \__coffin_join:NnnNnnnnN #1 {#2} {#3} #4 {#5} {#6} {#7} {#8} 3383
3836 \coffin_gset_eq:NN 3383
3837 } 3383
3838 \cs_generate_variant:Nn \coffin_gjoin:NnnNnnnn { c , Nnnc , cnnc } 3383
3839 \cs_new_protected:Npn \__coffin_join:NnnNnnnnN #1#2#3#4#5#6#7#8#9 3383
3840 { 3384
3841 \__coffin_align:NnnNnnnnN 3384
3842 #1 {#2} {#3} #4 {#5} {#6} {#7} {#8} \l__coffin_aligned_coffin 3384
3843 \hbox_set:Nn \l__coffin_aligned_coffin 3384
3844 { 3384
3845 \dim_compare:nNnT { \l__coffin_offset_x_dim } < \c_zero_dim 3384
3846 { \__kernel_kern:n { -\l__coffin_offset_x_dim } } 3384
3847 \hbox_unpack:N \l__coffin_aligned_coffin 3384
3848 \dim_set:Nn \l__coffin_internal_dim 3384
3849 { \l__coffin_offset_x_dim - \box_wd:N #1 + \box_wd:N #4 } 3384
3850 \dim_compare:nNnT \l__coffin_internal_dim < \c_zero_dim 3385
3851 { \__kernel_kern:n { -\l__coffin_internal_dim } } 3385
3852 } 3385
3853 \__coffin_reset_structure:N \l__coffin_aligned_coffin 3385
3854 \prop_clear:c 3385
3855 { 3385
3856 coffin ~ \__coffin_to_value:N \l__coffin_aligned_coffin 3385
3857 \c_space_tl corners 3385
3858 } 3385
3859 \__coffin_update_poles:N \l__coffin_aligned_coffin 3385
3860 \dim_compare:nNnTF \l__coffin_offset_x_dim < \c_zero_dim 3386
3861 { 3386
3862 \__coffin_offset_poles:Nnn #1 { -\l__coffin_offset_x_dim } { Opt } 3386
3863 \__coffin_offset_poles:Nnn #4 { Opt } { \l__coffin_offset_y_dim } 3386
3864 \__coffin_offset_corners:Nnn #1 { -\l__coffin_offset_x_dim } { Opt } 3386
3865 \__coffin_offset_corners:Nnn #4 { Opt } { \l__coffin_offset_y_dim } 3386
3866 } 3386
3867 { 3386
3868 \__coffin_offset_poles:Nnn #1 { Opt } { Opt } 3386
3869 \__coffin_offset_poles:Nnn #4 3386
3870 { \l__coffin_offset_x_dim } { \l__coffin_offset_y_dim } 3387
3871 \__coffin_offset_corners:Nnn #1 { Opt } { Opt } 3387
3872 \__coffin_offset_corners:Nnn #4 3387
3873 { \l__coffin_offset_x_dim } { \l__coffin_offset_y_dim } 3387
3874 } 3387
3875 \__coffin_update_vertical_poles:NNN #1 #4 \l__coffin_aligned_coffin 3387
3876 #9 #1 \l__coffin_aligned_coffin 3387
3877 } 3387
3878 \cs_new_protected:Npn \coffin_attach:NnnNnnnn #1#2#3#4#5#6#7#8 3387
```

```
3879 {
3880     \__coffin_attach:NnnNnnnnN #1 {#2} {#3} #4 {#5} {#6} {#7} {#8}
3881     \coffin_set_eq:NN
3882 }
3883 \cs_generate_variant:Nn \coffin_attach:NnnNnnnn { c , Nnnc , cnnc }
3884 \cs_new_protected:Npn \coffin_gattach:NnnNnnnn #1#2#3#4#5#6#7#8
3885 {
3886     \__coffin_attach:NnnNnnnnN #1 {#2} {#3} #4 {#5} {#6} {#7} {#8}
3887     \coffin_gset_eq:NN
3888 }
3889 \cs_generate_variant:Nn \coffin_gattach:NnnNnnnn { c , Nnnc , cnnc }
3890 \cs_new_protected:Npn \__coffin_attach:NnnNnnnnN #1#2#3#4#5#6#7#8#9
3891 {
3892     \__coffin_align:NnnNnnnnN
3893     #1 {#2} {#3} #4 {#5} {#6} {#7} {#8} \l__coffin_aligned_coffin
3894     \box_set_ht:Nn \l__coffin_aligned_coffin { \box_ht:N #1 }
3895     \box_set_dp:Nn \l__coffin_aligned_coffin { \box_dp:N #1 }
3896     \box_set_wd:Nn \l__coffin_aligned_coffin { \box_wd:N #1 }
3897     \__coffin_reset_structure:N \l__coffin_aligned_coffin
3898     \prop_set_eq:cc
3899     {
3900         coffin ~ \__coffin_to_value:N \l__coffin_aligned_coffin
3901         \c_space_tl corners
3902     }
3903     { coffin ~ \__coffin_to_value:N #1 ~ corners }
3904     \__coffin_update_poles:N \l__coffin_aligned_coffin
3905     \__coffin_offset_poles:Nnn #1 { Opt } { Opt }
3906     \__coffin_offset_poles:Nnn #4
3907     { \l__coffin_offset_x_dim } { \l__coffin_offset_y_dim }
3908     \__coffin_update_vertical_poles:NNN #1 #4 \l__coffin_aligned_coffin
3909     #9 #1 \l__coffin_aligned_coffin
3910 }
3911 \cs_new_protected:Npn \__coffin_attach_mark:NnnNnnnn #1#2#3#4#5#6#7#8
3912 {
3913     \__coffin_align:NnnNnnnnN
3914     #1 {#2} {#3} #4 {#5} {#6} {#7} {#8} \l__coffin_aligned_coffin
3915     \box_set_ht:Nn \l__coffin_aligned_coffin { \box_ht:N #1 }
3916     \box_set_dp:Nn \l__coffin_aligned_coffin { \box_dp:N #1 }
3917     \box_set_wd:Nn \l__coffin_aligned_coffin { \box_wd:N #1 }
3918     \box_set_eq:NN #1 \l__coffin_aligned_coffin
3919 }
3920 \cs_new_protected:Npn \__coffin_align:NnnNnnnnN #1#2#3#4#5#6#7#8#9
3921 {
3922     \__coffin_calculate_intersection:Nnn #4 {#5} {#6}
3923     \dim_set:Nn \l__coffin_x_prime_dim { \l__coffin_x_dim }
3924     \dim_set:Nn \l__coffin_y_prime_dim { \l__coffin_y_dim }
```

```

3925 \__coffin_calculate_intersection:Nnn #1 {#2} {#3}
3926 \dim_set:Nn \l__coffin_offset_x_dim
3927 { \l__coffin_x_dim - \l__coffin_x_prime_dim + #7 }
3928 \dim_set:Nn \l__coffin_offset_y_dim
3929 { \l__coffin_y_dim - \l__coffin_y_prime_dim + #8 }
3930 \hbox_set:Nn \l__coffin_aligned_internal_coffin
3931 {
3932   \box_use:N #1
3933   \__kernel_kern:n { -\box_wd:N #1 }
3934   \__kernel_kern:n { \l__coffin_offset_x_dim }
3935   \box_move_up:nn { \l__coffin_offset_y_dim } { \box_use:N #4 }
3936 }
3937 \coffin_set_eq:NN #9 \l__coffin_aligned_internal_coffin
3938 }
3939 \cs_new_protected:Npn \__coffin_offset_poles:Nnn #1#2#3
3940 {
3941   \prop_map_inline:cn { coffin ~ \__coffin_to_value:N #1 ~ poles }
3942   { \__coffin_offset_pole:Nnnnnnn #1 {##1} ##2 {#2} {#3} }
3943 }
3944 \cs_new_protected:Npn \__coffin_offset_pole:Nnnnnnn #1#2#3#4#5#6#7#8
3945 {
3946   \dim_set:Nn \l__coffin_x_dim { #3 + #7 }
3947   \dim_set:Nn \l__coffin_y_dim { #4 + #8 }
3948   \tl_if_in:nnTF {#2} { - }
3949   { \tl_set:Nn \l__coffin_internal_tl { {#2} } }
3950   { \tl_set:Nn \l__coffin_internal_tl { { #1 - #2 } } }
3951   \exp_last_unbraced:NNo \__coffin_set_pole:Nnn \l__coffin_aligned_coffin
3952   { \l__coffin_internal_tl }
3953   {
3954     { \dim_use:N \l__coffin_x_dim } { \dim_use:N \l__coffin_y_dim }
3955     {#5} {#6}
3956   }
3957 }
3958 \cs_new_protected:Npn \__coffin_offset_corners:Nnn #1#2#3
3959 {
3960   \prop_map_inline:cn { coffin ~ \__coffin_to_value:N #1 ~ corners }
3961   { \__coffin_offset_corner:Nnnnn #1 {##1} ##2 {#2} {#3} }
3962 }
3963 \cs_new_protected:Npn \__coffin_offset_corner:Nnnnn #1#2#3#4#5#6
3964 {
3965   \prop_put:cne
3966   {
3967     coffin ~ \__coffin_to_value:N \l__coffin_aligned_coffin
3968     \c_space_tl corners
3969   }
3970   { #1 - #2 }

```

```

3971 {
3972     { \dim_eval:n { #3 + #5 } }
3973     { \dim_eval:n { #4 + #6 } }
3974 }
3975 }
3976 \cs_new_protected:Npn \__coffin_update_vertical_poles:NNN #1#2#3
3977 {
3978     \__coffin_get_pole:NnN #3 { #1 -T } \l__coffin_pole_a_tl
3979     \__coffin_get_pole:NnN #3 { #2 -T } \l__coffin_pole_b_tl
3980     \exp_last_two_unbraced:Noo \__coffin_update_T:nnnnnnnnN
3981     \l__coffin_pole_a_tl \l__coffin_pole_b_tl #3
3982     \__coffin_get_pole:NnN #3 { #1 -B } \l__coffin_pole_a_tl
3983     \__coffin_get_pole:NnN #3 { #2 -B } \l__coffin_pole_b_tl
3984     \exp_last_two_unbraced:Noo \__coffin_update_B:nnnnnnnnN
3985     \l__coffin_pole_a_tl \l__coffin_pole_b_tl #3
3986 }
3987 \cs_new_protected:Npn \__coffin_update_T:nnnnnnnnN #1#2#3#4#5#6#7#8#9
3988 {
3989     \dim_compare:nNnTF {#2} < {#6}
3990     {
3991         \__coffin_set_pole:Nnn #9 { T }
3992         { { Opt } {#6} { 1000pt } { Opt } }
3993     }
3994     {
3995         \__coffin_set_pole:Nnn #9 { T }
3996         { { Opt } {#2} { 1000pt } { Opt } }
3997     }
3998 }
3999 \cs_new_protected:Npn \__coffin_update_B:nnnnnnnnN #1#2#3#4#5#6#7#8#9
4000 {
4001     \dim_compare:nNnTF {#2} < {#6}
4002     {
4003         \__coffin_set_pole:Nnn #9 { B }
4004         { { Opt } {#2} { 1000pt } { Opt } }
4005     }
4006     {
4007         \__coffin_set_pole:Nnn #9 { B }
4008         { { Opt } {#6} { 1000pt } { Opt } }
4009     }
4010 }
4011 \coffin_new:N \c__coffin_empty_coffin
4012 \tex_setbox:D \c__coffin_empty_coffin = \tex_hbox:D { }
4013 \cs_new_protected:Npn \coffin_typeset:Nnnnn #1#2#3#4#5
4014 {
4015     \mode_leave_vertical:
4016     \__coffin_align:NnnNnnnnN \c__coffin_empty_coffin { H } { l }

```



```
4017 #1 {#2} {#3} {#4} {#5} \l__coffin_aligned_coffin 3401
4018 \box_use_drop:N \l__coffin_aligned_coffin 3401
4019 } 3401
4020 \cs_generate_variant:Nn \coffin_typeset:Nnnnn { c } 3402
4021 \coffin_new:N \l__coffin_display_coffin 3402
4022 \coffin_new:N \l__coffin_display_coord_coffin 3402
4023 \coffin_new:N \l__coffin_display_pole_coffin 3402
4024 \prop_new:N \l__coffin_display_handles_prop 3402
4025 \prop_put:Nnn \l__coffin_display_handles_prop { tl } 3402
4026 { { b } { r } { -1 } { 1 } } 3402
4027 \prop_put:Nnn \l__coffin_display_handles_prop { thc } 3402
4028 { { b } { hc } { 0 } { 1 } } 3402
4029 \prop_put:Nnn \l__coffin_display_handles_prop { tr } 3402
4030 { { b } { l } { 1 } { 1 } } 3403
4031 \prop_put:Nnn \l__coffin_display_handles_prop { vcl } 3403
4032 { { vc } { r } { -1 } { 0 } } 3403
4033 \prop_put:Nnn \l__coffin_display_handles_prop { vhc } 3403
4034 { { vc } { hc } { 0 } { 0 } } 3403
4035 \prop_put:Nnn \l__coffin_display_handles_prop { vcr } 3403
4036 { { vc } { l } { 1 } { 0 } } 3403
4037 \prop_put:Nnn \l__coffin_display_handles_prop { bl } 3403
4038 { { t } { r } { -1 } { -1 } } 3403
4039 \prop_put:Nnn \l__coffin_display_handles_prop { bhc } 3403
4040 { { t } { hc } { 0 } { -1 } } 3404
4041 \prop_put:Nnn \l__coffin_display_handles_prop { br } 3404
4042 { { t } { l } { 1 } { -1 } } 3404
4043 \prop_put:Nnn \l__coffin_display_handles_prop { Tl } 3404
4044 { { t } { r } { -1 } { -1 } } 3404
4045 \prop_put:Nnn \l__coffin_display_handles_prop { Thc } 3404
4046 { { t } { hc } { 0 } { -1 } } 3404
4047 \prop_put:Nnn \l__coffin_display_handles_prop { Tr } 3404
4048 { { t } { l } { 1 } { -1 } } 3404
4049 \prop_put:Nnn \l__coffin_display_handles_prop { Hl } 3404
4050 { { vc } { r } { -1 } { 1 } } 3405
4051 \prop_put:Nnn \l__coffin_display_handles_prop { Hhc } 3405
4052 { { vc } { hc } { 0 } { 1 } } 3405
4053 \prop_put:Nnn \l__coffin_display_handles_prop { Hr } 3405
4054 { { vc } { l } { 1 } { 1 } } 3405
4055 \prop_put:Nnn \l__coffin_display_handles_prop { Bl } 3405
4056 { { b } { r } { -1 } { -1 } } 3405
4057 \prop_put:Nnn \l__coffin_display_handles_prop { Bhc } 3405
4058 { { b } { hc } { 0 } { -1 } } 3405
4059 \prop_put:Nnn \l__coffin_display_handles_prop { Br } 3405
4060 { { b } { l } { 1 } { -1 } } 3406
4061 \dim_new:N \l__coffin_display_offset_dim 3406
4062 \dim_set:Nn \l__coffin_display_offset_dim { 2pt } 3406
```



```
4063 \dim_new:N \l__coffin_display_x_dim 34063
4064 \dim_new:N \l__coffin_display_y_dim 34064
4065 \prop_new:N \l__coffin_display_poles_prop 34065
4066 \tl_new:N \l__coffin_display_font_tl 34066
4067 \bool_lazy_and:nnT 34067
4068 { \cs_if_exist_p:N \fmtname } 34068
4069 { \str_if_eq_p:Vn \fmtname { LaTeX2e } } 34069
4070 { 34070
4071 \tl_set:Nn \l__coffin_display_font_tl 34071
4072 { \sffamily \tiny } 34072
4073 } 34073
4074 \cs_new_protected:Npn \__coffin_rule:nn #1#2 34074
4075 { 34075
4076 \mode_leave_vertical: 34076
4077 \hbox:n { \tex_vrule:D width #1 height #2 \scan_stop: } 34077
4078 } 34078
4079 \cs_new_protected:Npn \coffin_mark_handle:Nnnn #1#2#3#4 34079
4080 { 34080
4081 \hcoffin_set:Nn \l__coffin_display_pole_coffin 34081
4082 { 34082
4083 \color_select:n {#4} 34083
4084 \__coffin_rule:nn { 1pt } { 1pt } 34084
4085 } 34085
4086 \__coffin_attach_mark:NnnNnnnn #1 {#2} {#3} 34086
4087 \l__coffin_display_pole_coffin { hc } { vc } { Opt } { Opt } 34087
4088 \hcoffin_set:Nn \l__coffin_display_coord_coffin 34088
4089 { 34089
4090 \color_select:n {#4} 34090
4091 \l__coffin_display_font_tl 34091
4092 ( \tl_to_str:n { #2 , #3 } ) 34092
4093 } 34093
4094 \prop_get:NnN \l__coffin_display_handles_prop 34094
4095 { #2 #3 } \l__coffin_internal_tl 34095
4096 \quark_if_no_value:NTF \l__coffin_internal_tl 34096
4097 { 34097
4098 \prop_get:NnN \l__coffin_display_handles_prop 34098
4099 { #3 #2 } \l__coffin_internal_tl 34099
4100 \quark_if_no_value:NTF \l__coffin_internal_tl 34100
4101 { 34101
4102 \__coffin_attach_mark:NnnNnnnn #1 {#2} {#3} 34102
4103 \l__coffin_display_coord_coffin { l } { vc } 34103
4104 { 1pt } { Opt } 34104
4105 } 34105
4106 { 34106
4107 \exp_last_unbraced:No \__coffin_mark_handle_aux:nnnnNnn 34107
4108 \l__coffin_internal_tl #1 {#2} {#3} 34108
```

```
4109     }
4110 }
4111 {
4112     \exp_last_unbraced:No \__coffin_mark_handle_aux:nnnnNnn
4113     \l__coffin_internal_tl #1 {#2} {#3}
4114 }
4115 }
4116 \cs_new_protected:Npn \__coffin_mark_handle_aux:nnnnNnn #1#2#3#4#5#6#7
4117 {
4118     \__coffin_attach_mark:NnnNnnnn #5 {#6} {#7}
4119     \l__coffin_display_coord_coffin {#1} {#2}
4120     { #3 \l__coffin_display_offset_dim }
4121     { #4 \l__coffin_display_offset_dim }
4122 }
4123 \cs_generate_variant:Nn \coffin_mark_handle:Nnnn { c }
4124 \cs_new_protected:Npn \coffin_display_handles:Nn #1#2
4125 {
4126     \hcoffin_set:Nn \l__coffin_display_pole_coffin
4127     {
4128         \color_select:n {#2}
4129         \__coffin_rule:nn { 1pt } { 1pt }
4130     }
4131     \prop_set_eq:Nc \l__coffin_display_poles_prop
4132     { coffin ~ \__coffin_to_value:N #1 ~ poles }
4133     \__coffin_get_pole:NnN #1 { H } \l__coffin_pole_a_tl
4134     \__coffin_get_pole:NnN #1 { T } \l__coffin_pole_b_tl
4135     \tl_if_eq:NNT \l__coffin_pole_a_tl \l__coffin_pole_b_tl
4136     { \prop_remove:Nn \l__coffin_display_poles_prop { T } }
4137     \__coffin_get_pole:NnN #1 { B } \l__coffin_pole_b_tl
4138     \tl_if_eq:NNT \l__coffin_pole_a_tl \l__coffin_pole_b_tl
4139     { \prop_remove:Nn \l__coffin_display_poles_prop { B } }
4140     \coffin_set_eq:NN \l__coffin_display_coffin #1
4141     \prop_map_inline:Nn \l__coffin_display_poles_prop
4142     {
4143         \prop_remove:Nn \l__coffin_display_poles_prop {##1}
4144         \__coffin_display_handles_aux:nnnnnn {##1} ##2 {#2}
4145     }
4146     \box_use_drop:N \l__coffin_display_coffin
4147 }
4148 \cs_new_protected:Npn \__coffin_display_handles_aux:nnnnnn #1#2#3#4#5#6
4149 {
4150     \prop_map_inline:Nn \l__coffin_display_poles_prop
4151     {
4152         \bool_set_false:N \l__coffin_error_bool
4153         \__coffin_calculate_intersection:nnnnnnnn {#2} {#3} {#4} {#5} ##2
4154         \bool_if:NF \l__coffin_error_bool
```

```

4155 {
4156     \dim_set:Nn \l__coffin_display_x_dim { \l__coffin_x_dim }
4157     \dim_set:Nn \l__coffin_display_y_dim { \l__coffin_y_dim }
4158     \__coffin_display_attach:Nnnnn
4159     \l__coffin_display_pole_coffin { hc } { vc }
4160     { Opt } { Opt }
4161     \hcoffin_set:Nn \l__coffin_display_coord_coffin
4162     {
4163         \color_select:n {#6}
4164         \l__coffin_display_font_tl
4165         ( \tl_to_str:n { #1 , ##1 } )
4166     }
4167     \prop_get:NnN \l__coffin_display_handles_prop
4168     { #1 ##1 } \l__coffin_internal_tl
4169     \quark_if_no_value:NTF \l__coffin_internal_tl
4170     {
4171         \prop_get:NnN \l__coffin_display_handles_prop
4172         { ##1 #1 } \l__coffin_internal_tl
4173         \quark_if_no_value:NTF \l__coffin_internal_tl
4174         {
4175             \__coffin_display_attach:Nnnnn
4176             \l__coffin_display_coord_coffin { l } { vc }
4177             { 1pt } { Opt }
4178         }
4179         {
4180             \exp_last_unbraced:No
4181             \__coffin_display_handles_aux:nnnn
4182             \l__coffin_internal_tl
4183         }
4184     }
4185     {
4186         \exp_last_unbraced:No \__coffin_display_handles_aux:nnnn
4187         \l__coffin_internal_tl
4188     }
4189 }
4190 }
4191 }
4192 \cs_new_protected:Npn \__coffin_display_handles_aux:nnnn #1#2#3#4
4193 {
4194     \__coffin_display_attach:Nnnnn
4195     \l__coffin_display_coord_coffin {#1} {#2}
4196     { #3 \l__coffin_display_offset_dim }
4197     { #4 \l__coffin_display_offset_dim }
4198 }
4199 \cs_generate_variant:Nn \coffin_display_handles:Nn { c }
4200 \cs_new_protected:Npn \__coffin_display_attach:Nnnnn #1#2#3#4#5

```

```

4201 {
4202   \__coffin_calculate_intersection:Nnn #1 {#2} {#3}
4203   \dim_set:Nn \l__coffin_x_prime_dim { \l__coffin_x_dim }
4204   \dim_set:Nn \l__coffin_y_prime_dim { \l__coffin_y_dim }
4205   \dim_set:Nn \l__coffin_offset_x_dim
4206     { \l__coffin_display_x_dim - \l__coffin_x_prime_dim + #4 }
4207   \dim_set:Nn \l__coffin_offset_y_dim
4208     { \l__coffin_display_y_dim - \l__coffin_y_prime_dim + #5 }
4209   \hbox_set:Nn \l__coffin_aligned_coffin
4210     {
4211       \box_use:N \l__coffin_display_coffin
4212       \__kernel_kern:n { -\box_wd:N \l__coffin_display_coffin }
4213       \__kernel_kern:n { \l__coffin_offset_x_dim }
4214       \box_move_up:nn { \l__coffin_offset_y_dim } { \box_use:N #1 }
4215     }
4216   \box_set_ht:Nn \l__coffin_aligned_coffin
4217     { \box_ht:N \l__coffin_display_coffin }
4218   \box_set_dp:Nn \l__coffin_aligned_coffin
4219     { \box_dp:N \l__coffin_display_coffin }
4220   \box_set_wd:Nn \l__coffin_aligned_coffin
4221     { \box_wd:N \l__coffin_display_coffin }
4222   \box_set_eq:NN \l__coffin_display_coffin \l__coffin_aligned_coffin
4223 }
4224 \cs_new_protected:Npn \coffin_show_structure:N
4225 { \__coffin_show_structure:NN \msg_show:nneeee }
4226 \cs_generate_variant:Nn \coffin_show_structure:N { c }
4227 \cs_new_protected:Npn \coffin_log_structure:N
4228 { \__coffin_show_structure:NN \msg_log:nneeee }
4229 \cs_generate_variant:Nn \coffin_log_structure:N { c }
4230 \cs_new_protected:Npn \__coffin_show_structure:NN #1#2
4231 {
4232   \__coffin_if_exist:NT #2
4233   {
4234     #1 { coffin } { show }
4235     { \token_to_str:N #2 }
4236     {
4237       \iow_newline: >~ ht ~~~ \dim_eval:n { \coffin_ht:N #2 }
4238       \iow_newline: >~ dp ~~~ \dim_eval:n { \coffin_dp:N #2 }
4239       \iow_newline: >~ wd ~~~ \dim_eval:n { \coffin_wd:N #2 }
4240     }
4241     {
4242       \prop_map_function:cN
4243         { coffin ~ \__coffin_to_value:N #2 ~ poles }
4244         \msg_show_item_unbraced:nn
4245       }
4246     { }

```

```

4247 }
4248 }
4249 \cs_new_protected:Npn \coffin_show:N #1
4250 { \coffin_show:Nnn #1 \c_max_int \c_max_int }
4251 \cs_generate_variant:Nn \coffin_show:N { c }
4252 \cs_new_protected:Npn \coffin_log:N #1
4253 { \coffin_log:Nnn #1 \c_max_int \c_max_int }
4254 \cs_generate_variant:Nn \coffin_log:N { c }
4255 \cs_new_protected:Npn \coffin_show:Nnn
4256 { \__coffin_show:NNNnn \msg_term:nneeee \box_show:Nnn }
4257 \cs_generate_variant:Nn \coffin_show:Nnn { c }
4258 \cs_new_protected:Npn \coffin_log:Nnn
4259 { \__coffin_show:NNNnn \msg_log:nneeee \box_show:Nnn }
4260 \cs_generate_variant:Nn \coffin_log:Nnn { c }
4261 \cs_new_protected:Npn \__coffin_show:NNNnn #1#2#3#4#5
4262 {
4263   \__coffin_if_exist:NT #3
4264   {
4265     \__coffin_show_structure:NN #1 #3
4266     #2 #3 {#4} {#5}
4267   }
4268 }
4269 \msg_new:nnnn { coffin } { no-pole-intersection }
4270 { No~intersection~between~coffin~poles. }
4271 {
4272   LaTeX~was~asked~to~find~the~intersection~between~two~poles,~
4273   but~they~do~not~have~a~unique~meeting~point:~
4274   the~value~(0pt,~0pt)~will~be~used.
4275 }
4276 \msg_new:nnnn { coffin } { unknown }
4277 { Unknown~coffin~'#1'. }
4278 { The~coffin~'#1'~was~never~defined. }
4279 \msg_new:nnnn { coffin } { unknown-pole }
4280 { Pole~'#1'~unknown~for~coffin~'#2'. }
4281 {
4282   LaTeX~was~asked~to~find~a~typesetting~pole~for~a~coffin,~
4283   but~either~the~coffin~does~not~exist~or~the~pole~name~is~wrong.
4284 }
4285 \msg_new:nnn { coffin } { show }
4286 {
4287   Size~of~coffin~#1 : #2 \\
4288   Poles~of~coffin~#1 : #3 .
4289 }
4290 %% File: l3luatex.dtx
4291 \cs_new_eq:NN \__lua_escape:n \tex_luaescapestring:D
4292 \cs_new_eq:NN \__lua_now:n \tex_directlua:D

```

4293	\cs_new_eq:NN __lua_shipout:n \tex_latelua:D	3429
4294	\cs_undefine:N \lua_escape:e	3429
4295	\cs_undefine:N \lua_now:e	3429
4296	\cs_new:Npn \lua_now:e #1 { __lua_now:n {#1} }	3429
4297	\cs_new:Npn \lua_now:n #1 { \lua_now:e { \exp_not:n {#1} } }	3429
4298	\cs_new_protected:Npn \lua_shipout_e:n #1 { __lua_shipout:n {#1} }	3429
4299	\cs_new_protected:Npn \lua_shipout:n #1	3429
4300	{ \lua_shipout_e:n { \exp_not:n {#1} } }	3430
4301	\cs_new:Npn \lua_escape:e #1 { __lua_escape:n {#1} }	3430
4302	\cs_new:Npn \lua_escape:n #1 { \lua_escape:e { \exp_not:n {#1} } }	3430
4303	\str_new:N \l__lua_err_msg_str	3430
4304	\cs_new_protected:Npn \lua_load_module:n #1	3430
4305	{	3430
4306	\bool_if:nF { __lua_load_module_p:n { #1 } }	3430
4307	{	3430
4308	\msg_error:nnnV	3430
4309	{ luatex } { module-not-found } { #1 } \l__lua_err_msg_str	3430
4310	}	3431
4311	}	3431
4312	\sys_if_engine luatex:F	3431
4313	{	3431
4314	\clist_map_inline:nn	3431
4315	{	3431
4316	\lua_escape:n , \lua_escape:e ,	3431
4317	\lua_now:n , \lua_now:e	3431
4318	}	3431
4319	{	3431
4320	\cs_gset:Npn #1 ##1	3432
4321	{	3432
4322	\msg_expandable_error:nnn	3432
4323	{ luatex } { luatex-required } { #1 }	3432
4324	}	3432
4325	}	3432
4326	\clist_map_inline:nn	3432
4327	{ \lua_shipout_e:n , \lua_shipout:n, \lua_load_module:n }	3432
4328	{	3432
4329	\cs_gset_protected:Npn #1 ##1	3432
4330	{	3433
4331	\msg_error:nnn	3433
4332	{ luatex } { luatex-required } { #1 }	3433
4333	}	3433
4334	}	3433
4335	}	3433
4336	\msg_new:nnnn { luatex } { luatex-required }	3433
4337	{ LuaTeX-engine~not~in~use!~Ignoring~#1. }	3433
4338	{	3433

```
4339     The~feature~you~are~using~is~only~available~
4340     with~the~LuaTeX~engine.~LaTeX3~ignored~'#1'.
4341 }
4342 \msg_new:nnnn { luatex } { module-not-found }
4343 { Lua~module~'#1'~not~found. }
4344 {
4345     The~file~'#1.lua'~could~not~be~found.~Please~ensure~
4346     that~the~file~was~properly~installed~and~that~the~
4347     filename~database~is~current.  \ \ \
4348     The~Lua~loader~provided~this~additional~information:  \ \
4349     #2
4350 }
4351 \prop_gput:Nnn \g_msg_module_name_prop { luatex } { LaTeX }
4352 \prop_gput:Nnn \g_msg_module_type_prop { luatex } { }
4353 %% File: l3unicode.dtx
4354 \sys_if_engine_opentype:TF
4355 {
4356     \cs_new:Npn \codepoint_str_generate:n #1
4357     {
4358         \int_compare:nNnTF {#1} = { `\_ }
4359         { ~ }
4360         { \char_generate:nn {#1} { 12 } }
4361     }
4362     \cs_new:Npn \codepoint_generate:nn #1#2
4363     {
4364         \int_compare:nNnTF {#1} = { `\_ }
4365         { ~ }
4366         {
4367             \__kernel_exp_not:w \exp_after:wN \exp_after:wN \exp_after:wN
4368             { \char_generate:nn {#1} {#2} }
4369         }
4370     }
4371 }
4372 {
4373     \cs_new:Npn \codepoint_str_generate:n #1
4374     {
4375         \int_compare:nNnTF {#1} = { `\_ }
4376         { ~ }
4377         {
4378             \use:e
4379             {
4380                 \exp_not:N \__codepoint_str_generate:nnnn
4381                 \__kernel_codepoint_to_bytes:n {#1}
4382             }
4383         }
4384     }
```



```
4385 \cs_new:Npn \__codepoint_str_generate:nnnn #1#2#3#4
4386 {
4387   \char_generate:nn {#1} { 12 }
4388   \tl_if_blank:nF {#2}
4389   {
4390     \char_generate:nn {#2} { 12 }
4391     \tl_if_blank:nF {#3}
4392     {
4393       \char_generate:nn {#3} { 12 }
4394       \tl_if_blank:nF {#4}
4395       { \char_generate:nn {#4} { 12 } }
4396     }
4397   }
4398 }
4399 \cs_new:Npn \codepoint_generate:nn #1#2
4400 {
4401   \int_compare:nNnTF {#1} = { `\_ }
4402   { ~ }
4403   {
4404     \int_compare:nNnTF {#1} < { "80 }
4405     {
4406       \__kernel_exp_not:w \exp_after:wN \exp_after:wN \exp_after:wN
4407       { \char_generate:nn {#1} {#2} }
4408     }
4409     {
4410       \use:e
4411       {
4412         \exp_not:N \__codepoint_generate:nnnn
4413         \__kernel_codepoint_to_bytes:n {#1}
4414       }
4415     }
4416   }
4417 }
4418 \cs_new:Npn \__codepoint_generate:nnnn #1#2#3#4
4419 {
4420   \__kernel_exp_not:w \exp_after:wN
4421   {
4422     \tex_expanded:D
4423     {
4424       \__codepoint_generate:n {#1}
4425       \__codepoint_generate:n {#2}
4426       \tl_if_blank:nF {#3}
4427       {
4428         \__codepoint_generate:n {#3}
4429         \tl_if_blank:nF {#4}
4430         { \__codepoint_generate:n {#4} }
```

```
4431     }
4432 }
4433 }
4434 }
4435 \cs_new:Npn \__codepoint_generate:n #1
4436 {
4437     \__kernel_exp_not:w \exp_after:wN \exp_after:wN \exp_after:wN
4438     { \char_generate:nn {#1} { 13 } }
4439 }
4440 }
4441 \cs_new:Npn \__kernel_codepoint_to_bytes:n #1
4442 {
4443     \exp_args:Nf \__codepoint_to_bytes_auxi:n
4444     { \int_eval:n {#1} }
4445 }
4446 \cs_new:Npn \__codepoint_to_bytes_auxi:n #1
4447 {
4448     \if_int_compare:w #1 > "80 \exp_stop_f:
4449     \if_int_compare:w #1 < "800 \exp_stop_f:
4450         \__codepoint_to_bytes_outputi:nw
4451         { \__codepoint_to_bytes_auxii:Nnn C {#1} { 64 } }
4452         \__codepoint_to_bytes_outputii:nw
4453         { \__codepoint_to_bytes_auxiii:n {#1} }
4454     \else:
4455         \if_int_compare:w #1 < "10000 \exp_stop_f:
4456             \__codepoint_to_bytes_outputi:nw
4457             { \__codepoint_to_bytes_auxii:Nnn E {#1} { 64 * 64 } }
4458             \__codepoint_to_bytes_outputii:nw
4459             {
4460                 \__codepoint_to_bytes_auxiii:n
4461                 { \int_div_truncate:nn {#1} { 64 } }
4462             }
4463             \__codepoint_to_bytes_outputiii:nw
4464             { \__codepoint_to_bytes_auxiii:n {#1} }
4465         \else:
4466             \__codepoint_to_bytes_outputi:nw
4467             {
4468                 \__codepoint_to_bytes_auxii:Nnn F
4469                 {#1} { 64 * 64 * 64 }
4470             }
4471             \__codepoint_to_bytes_outputii:nw
4472             {
4473                 \__codepoint_to_bytes_auxiii:n
4474                 { \int_div_truncate:nn {#1} { 64 * 64 } }
4475             }
4476             \__codepoint_to_bytes_outputiii:nw
```

```

4477         {
4478             \__codepoint_to_bytes_auxiii:n
4479             { \int_div_truncate:nn {#1} { 64 } }
4480         }
4481         \__codepoint_to_bytes_outputiv:nw
4482         { \__codepoint_to_bytes_auxiii:n {#1} }
4483         \fi:
4484         \fi:
4485         \else:
4486             \__codepoint_to_bytes_outputi:nw {#1}
4487             \fi:
4488             \__codepoint_to_bytes_end: { } { } { } { }
4489         }
4490         \cs_new:Npn \__codepoint_to_bytes_auxii:Nnn #1#2#3
4491         { "#10 + \int_div_truncate:nn {#2} {#3} }
4492         \cs_new:Npn \__codepoint_to_bytes_auxiii:n #1
4493         { \int_mod:nn {#1} { 64 } + 128 }
4494         \cs_new:Npn \__codepoint_to_bytes_outputi:nw
4495         #1 #2 \__codepoint_to_bytes_end: #3
4496         { \__codepoint_to_bytes_output:fnn { \int_eval:n {#1} } { } {#2} }
4497         \cs_new:Npn \__codepoint_to_bytes_outputii:nw
4498         #1 #2 \__codepoint_to_bytes_end: #3#4
4499         { \__codepoint_to_bytes_output:fnn { \int_eval:n {#1} } { {#3} } {#2} }
4500         \cs_new:Npn \__codepoint_to_bytes_outputiii:nw
4501         #1 #2 \__codepoint_to_bytes_end: #3#4#5
4502         {
4503             \__codepoint_to_bytes_output:fnn
4504             { \int_eval:n {#1} } { {#3} {#4} } {#2}
4505         }
4506         \cs_new:Npn \__codepoint_to_bytes_outputiv:nw
4507         #1 #2 \__codepoint_to_bytes_end: #3#4#5#6
4508         {
4509             \__codepoint_to_bytes_output:fnn
4510             { \int_eval:n {#1} } { {#3} {#4} {#5} } {#2}
4511         }
4512         \cs_new:Npn \__codepoint_to_bytes_output:nnn #1#2#3
4513         {
4514             #3
4515             \__codepoint_to_bytes_end: #2 {#1}
4516         }
4517         \cs_generate_variant:Nn \__codepoint_to_bytes_output:nnn { f }
4518         \cs_new:Npn \__codepoint_to_bytes_end: { }
4519         \cs_new:Npn \codepoint_to_category:n #1
4520         {
4521             \cs:w
4522             c__codepoint_category_

```

```

4523 \tex_romannumeral:D
4524 \__kernel_codepoint_data:nn { category } {#1}
4525 _str
4526 \cs_end:
4527 }
4528 \cs_new:Npn \codepoint_to_nfd:n #1
4529 { \exp_args:Ne \__codepoint_to_nfd:n { \int_eval:n {#1} } }
4530 \cs_new:Npn \__codepoint_to_nfd:n #1
4531 { \__codepoint_to_nfd:nn {#1} { \char_value_catcode:n {#1} } }
4532 \sys_if_engine_opentype:F
4533 {
4534 \cs_gset:Npn \__codepoint_to_nfd:n #1
4535 {
4536 \int_compare:nNnTF {#1} > { "80 }
4537 { \__codepoint_to_nfd:nn {#1} { 12 } }
4538 { \__codepoint_to_nfd:nn {#1} { \char_value_catcode:n {#1} } }
4539 }
4540 }
4541 \cs_new:Npn \__codepoint_to_nfd:nn #1#2
4542 {
4543 \exp_args:Ne \__codepoint_to_nfd:nnn
4544 { \__codepoint_nfd:n {#1} } {#1} {#2}
4545 }
4546 \cs_new:Npn \__codepoint_to_nfd:nnn #1#2#3 { \__codepoint_to_nfd:nnnn #1 {#2} {#3} }
4547 \cs_new:Npn \__codepoint_to_nfd:nnnn #1#2#3#4
4548 {
4549 \int_compare:nNnTF {#1} = {#3}
4550 { \codepoint_generate:nn {#1} {#4} }
4551 {
4552 \__codepoint_to_nfd:nn {#1} {#4}
4553 \tl_if_blank:nF {#2}
4554 { \__codepoint_to_nfd:nn {#2} {#4} }
4555 }
4556 }
4557 \int_const:Nn \c__codepoint_block_size_int { 64 }
4558 \ior_new:N \g__codepoint_data_ior
4559 \group_begin:
4560 \clist_map_inline:nn
4561 { category , grapheme , lowercase , uppercase , wordbreak }
4562 {
4563 \cs_set_nopar:cpn { l__codepoint_ #1 _block_clist } { }
4564 \cs_set_nopar:cpn { l__codepoint_ #1 _block_tl } { 1 }
4565 \cs_set_nopar:cpn { l__codepoint_ #1 _pos_tl } { 0 }
4566 \cs_set_nopar:cpn { l__codepoint_ #1 _next_tl } { 0 }
4567 \intarray_new:cn { g__codepoint_ #1 _index_intarray }
4568 { \int_div_truncate:nn { "110000 } \c__codepoint_block_size_int }

```

```

4569 }
4570 \cs_set_nopar:Npn \l__codepoint_matched_block_tl { 0 }
4571 \cs_set_nopar:Npn \l__codepoint_uppercase_default_tl { 0 }
4572 \cs_set_nopar:Npn \l__codepoint_lowercase_default_tl { 0 }
4573 \cs_set_protected:Npn \__codepoint_data_auxi:w #1#2#3
4574 {
4575   \quark_if_recursion_tail_stop:n {#3}
4576   \cs_set_nopar:cpn { l__codepoint_ #2 _ #3 _tl } {#1}
4577   \str_const:cn { c__codepoint_ #2 _ \tex_romannumeral:D #1 _str } {#3}
4578   \exp_args:Ne \__codepoint_data_auxi:w { \int_eval:n { #1 + 1 } } {#2}
4579 }
4580 \__codepoint_data_auxi:w { 1 } { category }
4581 { Lu } { Ll } { Lt } { Lm } { Lo }
4582 { Mn } { Me } { Mc }
4583 { Nd } { Nl } { No }
4584 { Zs } { Zl } { Zp }
4585 { Cc } { Cf } { Co } { Cs } { Cn }
4586 { Pd } { Ps } { Pe } { Pc } { Po } { Pi } { Pf }
4587 { Sm } { Sc } { Sk } { So }
4588 \q_recursion_tail
4589 \q_recursion_stop
4590 \cs_set_eq:NN \l__codepoint_category_default_tl \l__codepoint_category_Cn_tl
4591 \__codepoint_data_auxi:w { 1 } { grapheme }
4592 { Control }
4593 { CR } { LF } { ZWJ }
4594 { Extend }
4595 { L } { LV } { LVT } { T } { V }
4596 { Prepend }
4597 { Regional_Indicator }
4598 { SpacingMark }
4599 { Other }
4600 \q_recursion_tail
4601 \q_recursion_stop
4602 \cs_set_eq:NN \l__codepoint_grapheme_default_tl \l__codepoint_grapheme_Other_tl
4603 \__codepoint_data_auxi:w { 1 } { wordbreak }
4604 { Double_Quote } { Single_Quote }
4605 { CR } { LF } { Newline }
4606 { WSegSpace } { ZWJ }
4607 { Extend } { ExtendNumLet }
4608 { Regional_Indicator }
4609 { Format }
4610 { Katakana }
4611 { ALetter } { MidLetter } { Hebrew_Letter }
4612 { Numeric } { MidNum } { MidNumLet }
4613 { Other }
4614 \q_recursion_tail

```

```

4615 \q_recursion_stop
4616 \cs_set_eq:NN \l__codepoint_wordbreak_default_tl \l__codepoint_wordbreak_Other_tl
4617 \cs_set_protected:Npn \__codepoint_data_auxi:w #1 ;~ #2 ~ #3 \q_stop
4618 { \__codepoint_data_auxii:w #1 .. \q_stop {#2} }
4619 \cs_set_protected:Npn \__codepoint_data_auxii:w #1 .. #2 \q_stop
4620 { \__codepoint_data_auxiii:w #1 ~ .. #2 ~ \q_stop }
4621 \cs_set_protected:Npn \__codepoint_data_auxiii:w #1 ~ #2 .. #3 ~ #4 \q_stop #5#6
4622 {
4623   \cs_set_nopar:cpe { l__codepoint_ #6 _ \tex_romannumeral:D "#1 _tl }
4624   {
4625     {#3}
4626     { \use:c { l__codepoint_ #6 _ #5 _tl } }
4627   }
4628 }
4629 \cs_set_protected:Npn \__codepoint_data_auxvi:w #1#2
4630 {
4631   \ior_open:Nn \g__codepoint_data_ior {#1}
4632   \ior_str_map_inline:Nn \g__codepoint_data_ior
4633   {
4634     \str_if_eq:eeF { \tl_head:w ##1 \c_hash_str \q_stop } { \c_hash_str }
4635     {
4636       \tl_if_blank:nF {##1}
4637       { \__codepoint_data_auxi:w ##1 \q_stop {#2} }
4638     }
4639   }
4640   \ior_close:N \g__codepoint_data_ior
4641 }
4642 \__codepoint_data_auxvi:w { GraphemeBreakProperty.txt } { grapheme }
4643 \__codepoint_data_auxvi:w { WordBreakProperty.txt } { wordbreak }
4644 \cs_set_protected:Npn \__codepoint_data_property:nnnn #1#2#3#4
4645 {
4646   \int_compare:nNnT {#3} > { \use:c { l__codepoint_ #4 _next_tl } }
4647   {
4648     \__codepoint_range:nnv {#3} {#4}
4649     { l__codepoint_ #4 _default_tl }
4650   }
4651   \__codepoint_add:nn {#4} {#2}
4652   \tl_set:ce { l__codepoint_ #4 _next_tl } { \int_eval:n { #3 + 1 } }
4653   \tl_if_blank:nF {#1}
4654   {
4655     \__codepoint_range:nnn {"#1} {#4} {#2}
4656     \__codepoint_add:nn {#4} {#2}
4657     \tl_set:ce { l__codepoint_ #4 _next_tl } { \int_eval:n { "#1 + 1 } }
4658   }
4659 }
4660 \cs_set_protected:Npn \__codepoint_data_auxi:w

```

```

4661 #1 ; #2 ; #3 ; #4 ; #5 ; #6 ; #7 ; #8 ; #9 ;
4662 {
4663     \tl_if_blank:nF {#6}
4664     {
4665         \tl_if_head_eq_charcode:nNF {#6} < % >
4666         { \__codepoint_data_auxii:w #1 ; #6 ~ \q_stop }
4667     }
4668     \__codepoint_data_auxiii:w #1 ; #2 ; #3 ;
4669 }
4670 \cs_set_protected:Npn \__codepoint_data_auxii:w #1 ; #2 ~ #3 \q_stop
4671 {
4672     \tl_const:ce
4673     { c__codepoint_nfd_ \codepoint_str_generate:n {"#1} _tl }
4674     {
4675         {"#2}
4676         { \tl_if_blank:nF {#3} {"#3} }
4677     }
4678 }
4679 \cs_set_protected:Npn \__codepoint_data_auxiii:w
4680 #1 ; #2 ; #3 ; #4 ; #5 ; #6 ; #7 ; #8 ; #9 ~ \q_stop
4681 {
4682     \use:e
4683     {
4684         \__codepoint_data_auxiv:w
4685         #1 ; #2 ;
4686         \__codepoint_data_category:n {#3} ;
4687         \__codepoint_data_offset:nn {#1} {#7} ;
4688         \__codepoint_data_offset:nn {#1} {#8} ;
4689         #9;
4690     }
4691 }
4692 \cs_set:Npn \__codepoint_data_category:n #1
4693 { \use:c { l__codepoint_category_ #1 _tl } }
4694 \cs_set:Npn \__codepoint_data_offset:nn #1#2
4695 {
4696     \tl_if_blank:nTF {#2}
4697     { 0 }
4698     { \int_eval:n { "#2 - "#1 } }
4699 }
4700 \cs_set_protected:Npn \__codepoint_data_auxiv:w #1 ; #2 ; #3 ; #4 ; #5 ; #6 ;
4701 {
4702     \int_compare:nNnT {"#1} > \l__codepoint_category_next_tl
4703     {
4704         \__codepoint_data_auxv:nnnnw {#1} {#3} {#4} {#5}
4705         #2 Last> \q_stop
4706     }

```



```
4707 \__codepoint_add:nn { category } {#3} 3470
4708 \__codepoint_add:nn { uppercase } {#4} 3470
4709 \__codepoint_add:nn { lowercase } {#5} 3470
4710 \int_compare:nNnF {#4} = { \__codepoint_data_offset:nn {#1} {#6} } 3471
4711 { 3471
4712 \tl_const:ce 3471
4713 { c__codepoint_titlecase_ \codepoint_str_generate:n {"#1} _tl } 3471
4714 { {"#6} { } { } } 3471
4715 } 3471
4716 \__codepoint_data_auxvi:nn { grapheme } {"#1} 3471
4717 \__codepoint_data_auxvi:nn { wordbreak } {"#1} 3471
4718 \int_compare:nNnT {"#1} = { "AC00 } 3471
4719 { 3471
4720 \int_step_inline:nnn { "AC01 } { "D7A2 } 3472
4721 { \__codepoint_data_auxvi:nn { grapheme } {#1} } 3472
4722 } 3472
4723 \tl_set:Ne \l__codepoint_category_next_tl 3472
4724 { \int_eval:n { "#1 + 1 } } 3472
4725 \tl_set_eq:NN \l__codepoint_lowercase_next_tl \l__codepoint_category_next_tl 3472
4726 \tl_set_eq:NN \l__codepoint_uppercase_next_tl \l__codepoint_category_next_tl 3472
4727 } 3472
4728 \cs_set_protected:Npn \__codepoint_add:nn #1#2 3472
4729 { 3472
4730 \clist_put_right:cn { l__codepoint_ #1 _block_clist } {#2} 3473
4731 \int_compare:nNnT { \clist_count:c { l__codepoint_ #1 _block_clist } } 3473
4732 = \c__codepoint_block_size_int 3473
4733 { \__codepoint_save_blocks:nn {#1} { 1 } } 3473
4734 } 3473
4735 \cs_set_protected:Npn \__codepoint_data_auxv:nnnnw #1#2#3#4#5 Last> #6 \q_stop 3473
4736 { 3473
4737 \tl_if_blank:nTF {#6} 3473
4738 { 3473
4739 \__codepoint_range:nno {"#1} { category } 3473
4740 \l__codepoint_category_default_tl 3474
4741 \__codepoint_range:nnn {"#1} { uppercase } { 0 } 3474
4742 \__codepoint_range:nnn {"#1} { lowercase } { 0 } 3474
4743 } 3474
4744 { 3474
4745 \__codepoint_range:nnn {"#1} { category } {#2} 3474
4746 \__codepoint_range:nnn {"#1} { uppercase } {#3} 3474
4747 \__codepoint_range:nnn {"#1} { lowercase } {#4} 3474
4748 } 3474
4749 } 3474
4750 \cs_set_protected:Npn \__codepoint_data_auxvi:nn #1#2 3475
4751 { 3475
4752 \cs_if_exist:cT 3475
```

```

4753 { l__codepoint_ #1 _ \tex_romannumeral:D #2 _tl } 3475
4754 { 3475
4755 \exp_after:wN \exp_after:wN \exp_after:wN \__codepoint_data_property:nnnn 3475
4756 \cs:w 3475
4757 l__codepoint_ #1 _ \tex_romannumeral:D #2 _tl 3475
4758 \cs_end: {#2} {#1} 3475
4759 } 3475
4760 } 3476
4761 \cs_set_protected:Npn \__codepoint_range:nnn #1#2 3476
4762 { 3476
4763 \exp_args:Nf \__codepoint_range_aux:nnn 3476
4764 { \int_eval:n { #1 - \use:c { l__codepoint_ #2 _next_tl } } } 3476
4765 {#2} 3476
4766 } 3476
4767 \cs_set_protected:Npn \__codepoint_range:nno { \exp_args:Nnno \__codepoint_range:nnn } 3476
4768 \cs_set_protected:Npn \__codepoint_range:nnv { \exp_args:Nnnv \__codepoint_range:nnn } 3476
4769 \cs_set_protected:Npn \__codepoint_range_aux:nnn #1#2 3476
4770 { 3477
4771 \exp_args:Nf \__codepoint_range:nnnn 3477
4772 { 3477
4773 \int_min:nn 3477
4774 {#1} 3477
4775 { 3477
4776 \c__codepoint_block_size_int 3477
4777 - \clist_count:c { l__codepoint_ #2 _block_clist } 3477
4778 } 3477
4779 } 3477
4780 {#1} {#2} 3478
4781 } 3478
4782 \cs_set_protected:Npn \__codepoint_range:nnnn #1#2#3#4 3478
4783 { 3478
4784 \prg_replicate:nn {#1} 3478
4785 { \clist_put_right:cn { l__codepoint_ #3 _block_clist } {#4} } 3478
4786 \int_compare:nNnT { \clist_count:c { l__codepoint_ #3 _block_clist } } 3478
4787 = \c__codepoint_block_size_int 3478
4788 { \__codepoint_save_blocks:nn {#3} { 1 } } 3478
4789 \int_compare:nNnF 3478
4790 { \int_div_truncate:nn { #2 - #1 } \c__codepoint_block_size_int } = 0 3479
4791 { 3479
4792 \tl_set:ce { l__codepoint_ #3 _block_clist } 3479
4793 { 3479
4794 \exp_args:NNe \use:nn \use_none:n 3479
4795 { \prg_replicate:nn { \c__codepoint_block_size_int } { , #4 } } 3479
4796 } 3479
4797 \__codepoint_save_blocks:nn {#3} 3479
4798 { \int_div_truncate:nn { (#2 - #1) } \c__codepoint_block_size_int } 3479

```

```

4799 }
4800 \prg_replicate:nn
4801 { \int_mod:nn { #2 - #1 } \c__codepoint_block_size_int }
4802 { \clist_put_right:ce { l__codepoint_ #3 _block_clist } {#4} }
4803 }
4804 \cs_set_protected:Npn \__codepoint_save_blocks:nn #1#2
4805 {
4806   \tl_set_eq:Nc \l__codepoint_matched_block_tl { l__codepoint_ #1 _block_tl }
4807   \int_step_inline:nn { \tl_use:c { l__codepoint_ #1 _block_tl } - 1 }
4808   {
4809     \tl_if_eq:ccT { l__codepoint_ #1 _block_clist }
4810     { l__codepoint_ #1 _block_ ##1 _clist }
4811     { \tl_set:Nn \l__codepoint_matched_block_tl {##1} }
4812   }
4813   \int_compare:nNnT
4814   { \tl_use:c { l__codepoint_ #1 _block_tl } } = \l__codepoint_matched_block_tl
4815   {
4816     \clist_set_eq:cc
4817     {
4818       l__codepoint_ #1 _block_
4819       \tl_use:c { l__codepoint_ #1 _block_tl } _clist
4820     }
4821     { l__codepoint_ #1 _block_clist }
4822     \tl_set:ce { l__codepoint_ #1 _block_tl }
4823     { \int_eval:n { \tl_use:c { l__codepoint_ #1 _block_tl } + 1 } }
4824   }
4825   \int_step_inline:nnn
4826   { \tl_use:c { l__codepoint_ #1 _pos_tl } + 1 }
4827   { \tl_use:c { l__codepoint_ #1 _pos_tl } + #2 }
4828   {
4829     \exp_args:Nc \__kernel_intarray_gset:Nnn
4830     { g__codepoint_ #1 _index_intarray }
4831     {##1}
4832     \l__codepoint_matched_block_tl
4833   }
4834   \tl_set:ce { l__codepoint_ #1 _pos_tl }
4835   { \int_eval:n { \tl_use:c { l__codepoint_ #1 _pos_tl } + #2 } }
4836   \clist_clear:c { l__codepoint_ #1 _block_clist }
4837 }
4838 \cs_set_protected:Npn \__codepoint_finalise_blocks:n #1
4839 {
4840   \clist_map_inline:nn {#1}
4841   {
4842     \exp_args:Nnnv \__codepoint_range:nnn { "110000 } {##1}
4843     { l__codepoint_ ##1 _default_tl }
4844     \__codepoint_finalise_blocks_aux:n {##1}

```

```

4845     }
4846 }
4847 \cs_set_protected:Npn \__codepoint_finalise_blocks_aux:n #1
4848 {
4849     \cs_gset_eq:cc { c__codepoint_ #1 _index_intarray } { g__codepoint_ #1
4850 _index_intarray }
4851     \cs_undefine:c { g__codepoint_ #1 _index_intarray }
4852     \intarray_new:cn { g__codepoint_ #1 _blocks_intarray }
4853     { ( \tl_use:c { l__codepoint_ #1 _block_tl } - 1 ) *
4854 \c__codepoint_block_size_int }
4855     \int_step_inline:nn { \tl_use:c { l__codepoint_ #1 _block_tl } - 1 }
4856     {
4857         \exp_args:Nv \__codepoint_finalise_blocks:nnn
4858         { l__codepoint_ #1 _block_ ##1 _clist }
4859         {##1} {#1}
4860     }
4861     \cs_gset_eq:cc { c__codepoint_ #1 _blocks_intarray }
4862     { g__codepoint_ #1 _blocks_intarray }
4863     \cs_undefine:c { g__codepoint_ #1 _blocks_intarray }
4864 }
4865 \cs_set_protected:Npn \__codepoint_finalise_blocks:nnn #1#2#3
4866 {
4867     \exp_args:Nnf \__codepoint_finalise_blocks:nnnw { 1 }
4868     { \int_eval:n { ( #2 - 1 ) * \c__codepoint_block_size_int } }
4869     {#3}
4870     #1 , \q_recursion_tail , \q_recursion_stop
4871 }
4872 \cs_set_protected:Npn \__codepoint_finalise_blocks:nnnw #1#2#3#4 ,
4873 {
4874     \quark_if_recursion_tail_stop:n {#4}
4875     \intarray_gset:cn { g__codepoint_ #3 _blocks_intarray }
4876     { #1 + #2 }
4877     {#4}
4878     \exp_args:Nf \__codepoint_finalise_blocks:nnnw
4879     { \int_eval:n { #1 + 1 } } {#2} {#3}
4880 }
4881 \ior_open:Nn \g__codepoint_data_ior { UnicodeData.txt }
4882 \char_set_catcode_space:n { \_ }%
4883 \ior_map_variable:NNn \g__codepoint_data_ior \l__codepoint_tmpa_tl
4884 {%
4885     \if_meaning:w \l__codepoint_tmpa_tl \c_space_tl
4886     \exp_after:wN \ior_map_break:
4887     \fi:
4888     \exp_after:wN \__codepoint_data_auxi:w \l__codepoint_tmpa_tl \q_stop
4889 }%
4890 \char_set_catcode_ignore:n { \_ }%

```

```

4889 \__codepoint_finalise_blocks:n
4890 { category , grapheme , lowercase , uppercase , wordbreak }
4891 \group_end:
4892 \cs_new:Npn \__kernel_codepoint_data:nn #1#2
4893 {
4894 \exp_args:Nf \__codepoint_data:nnn
4895 {
4896 \int_eval:n
4897 {
4898 \c__codepoint_block_size_int *
4899 (
4900 \intarray_item:cn { c__codepoint_ #1 _index_intarray }
4901 {
4902 \int_div_truncate:nn {#2}
4903 \c__codepoint_block_size_int
4904 + 1
4905 }
4906 - 1
4907 )
4908 }
4909 }
4910 {#2} {#1}
4911 }
4912 \cs_new:Npn \__codepoint_data:nnn #1#2#3
4913 {
4914 \intarray_item:cn { c__codepoint_ #3 _blocks_intarray }
4915 { #1 + \int_mod:nn {#2} \c__codepoint_block_size_int + 1 }
4916 }
4917 \group_begin:
4918 \ior_open:Nn \g__codepoint_data_ior { CaseFolding.txt }
4919 \cs_set_protected:Npn \__codepoint_data_auxi:w #1 ;~ #2 ;~ #3 ; #4 \q_stop
4920 {
4921 \if:w \tl_head:n { #2 ? } C
4922 \reverse_if:N \if_int_compare:w
4923 \int_eval:n { \__kernel_codepoint_data:nn { lowercase } {"#1} + "#1 }
4924 = "#3 ~
4925 \tl_const:ce
4926 { c__codepoint_casefold_ \codepoint_str_generate:n {"#1} _tl }
4927 { {"#3} { } { } }
4928 \fi:
4929 \else:
4930 \if:w \tl_head:n { #2 ? } F
4931 \__codepoint_data_auxii:w #1 ~ #3 ~ \q_stop
4932 \fi:
4933 \fi:
4934 }

```

```
4935 \cs_set_protected:Npn \__codepoint_data_auxii:w #1 ~ #2 ~ #3 ~ #4 \q_stop 3493
4936 { 3493
4937     \tl_const:ce { c__codepoint_casefold_ \codepoint_str_generate:n {"#1} _tl } 3493
4938     { 3493
4939         {"#2} 3493
4940         {"#3} 3494
4941         { \tl_if_blank:nF {#4} { " \int_to_Hex:n {"#4} } } 3494
4942     } 3494
4943 } 3494
4944 \ior_str_map_inline:Nn \g__codepoint_data_ior 3494
4945 { 3494
4946     \reverse_if:N \if:w \c_hash_str \tl_head:w #1 \c_hash_str \q_stop 3494
4947     \__codepoint_data_auxi:w #1 \q_stop 3494
4948     \fi: 3494
4949 } 3494
4950 \ior_close:N \g__codepoint_data_ior 3495
4951 \ior_open:Nn \g__codepoint_data_ior { SpecialCasing.txt } 3495
4952 \cs_set_protected:Npn \__codepoint_data_auxi:w 3495
4953     #1 ;~ #2 ;~ #3 ;~ #4 ; #5 \q_stop 3495
4954     { 3495
4955         \use:n { \__codepoint_data_auxii:w #1 ~ lower ~ #2 ~ } ~ \q_stop 3495
4956         \use:n { \__codepoint_data_auxii:w #1 ~ upper ~ #4 ~ } ~ \q_stop 3495
4957         \str_if_eq:nnF {#3} {#4} 3495
4958         { \use:n { \__codepoint_data_auxii:w #1 ~ title ~ #3 ~ } ~ \q_stop } 3495
4959     } 3495
4960 \cs_set_protected:Npn \__codepoint_data_auxii:w 3496
4961     #1 ~ #2 ~ #3 ~ #4 ~ #5 \q_stop 3496
4962     { 3496
4963         \tl_if_empty:nF {#4} 3496
4964         { 3496
4965             \tl_const:ce { c__codepoint_ #2 case_ \codepoint_str_generate:n {"#1} _tl } 3496
4966             { 3496
4967                 {"#3} 3496
4968                 {"#4} 3496
4969                 { \tl_if_blank:nF {#5} {"#5} } 3496
4970             } 3497
4971         } 3497
4972     } 3497
4973 \ior_str_map_inline:Nn \g__codepoint_data_ior 3497
4974 { 3497
4975     \str_if_eq:eeTF { \tl_head:w #1 \c_hash_str \q_stop } { \c_hash_str } 3497
4976     { 3497
4977         \str_if_eq:eeT 3497
4978         {#1} 3497
4979         { \c_hash_str \c_space_tl Conditional~Mappings } 3497
4980         { \ior_map_break: } 3498
```

```

4981     }
4982     { \__codepoint_data_auxi:w #1 \q_stop }
4983 }
4984 \ior_close:N \g__codepoint_data_ior
4985 \group_end:
4986 \cs_new:Npn \__kernel_codepoint_case:nn #1#2
4987 {
4988     \exp_args:Ne \__codepoint_case:nnn
4989     { \codepoint_str_generate:n {#2} } {#1} {#2}
4990 }
4991 \cs_new:Npn \__codepoint_case:nnn #1#2#3
4992 {
4993     \cs_if_exist:cTF { c__codepoint_ #2 _ #1 _tl }
4994     {
4995         \tl_use:c
4996         { c__codepoint_ #2 _ #1 _tl }
4997     }
4998     { \use:c { __codepoint_ #2 :n } {#3} }
4999 }
5000 \cs_new:Npn \__codepoint_uppercase:n { \__codepoint_case:nn { uppercase } }
5001 \cs_new:Npn \__codepoint_lowercase:n { \__codepoint_case:nn { lowercase } }
5002 \cs_new:Npn \__codepoint_titlecase:n { \__codepoint_case:nn { uppercase } }
5003 \cs_new:Npn \__codepoint_casefold:n { \__codepoint_case:nn { lowercase } }
5004 \cs_new:Npn \__codepoint_case:nn #1#2
5005 {
5006     { \int_eval:n { \__kernel_codepoint_data:nn {#1} {#2} + #2 } }
5007     { }
5008     { }
5009 }
5010 \cs_new:Npn \__kernel_codepoint_to_grapheme_class:n #1
5011 {
5012     \cs:w
5013     c__codepoint_grapheme_
5014     \tex_romannumeral:D
5015     \__kernel_codepoint_data:nn { grapheme } {#1}
5016     _str
5017     \cs_end:
5018 }
5019 \cs_new:Npn \__kernel_codepoint_to_wordbreak_class:n #1
5020 {
5021     \cs:w
5022     c__codepoint_wordbreak_
5023     \tex_romannumeral:D
5024     \__kernel_codepoint_data:nn { wordbreak } {#1}
5025     _str
5026     \cs_end:

```



```

5027 }
5028 \cs_new:Npn \__codepoint_nfd:n #1
5029 { \exp_args:Ne \__codepoint_nfd:nn { \codepoint_str_generate:n {#1} } {#1} }
5030 \cs_new:Npn \__codepoint_nfd:nn #1#2
5031 {
5032   \tl_if_exist:cTF { c__codepoint_nfd_ #1 _tl }
5033   { \tl_use:c { c__codepoint_nfd_ #1 _tl } }
5034   { {#2} { } }
5035 }
5036 %% File: l3text.dtx
5037 \cs_generate_variant:Nn \tl_if_head_eq_meaning_p:nN { o }
5038 \scan_new:N \s__text_stop
5039 \quark_new:N \q__text_nil
5040 \__kernel_quark_new_conditional:Nn \__text_quark_if_nil:n { TF }
5041 \quark_new:N \q__text_recursion_tail
5042 \quark_new:N \q__text_recursion_stop
5043 \cs_new:Npn \__text_use_i_delimit_by_q_recursion_stop:nw
5044 #1 #2 \q__text_recursion_stop {#1}
5045 \__kernel_quark_new_test:N \__text_if_q_recursion_tail_stop_do:Nn
5046 \__kernel_quark_new_test:N \__text_if_q_recursion_tail_stop_do:nn
5047 \scan_new:N \s__text_recursion_tail
5048 \scan_new:N \s__text_recursion_stop
5049 \cs_new:Npn \__text_use_i_delimit_by_s_recursion_stop:nw
5050 #1 #2 \s__text_recursion_stop {#1}
5051 \cs_new:Npn \__text_if_s_recursion_tail_stop_do:Nn #1
5052 {
5053   \bool_lazy_and:nnTF
5054   { \cs_if_eq_p:NN \s__text_recursion_tail #1 }
5055   { \str_if_eq_p:nn { \s__text_recursion_tail } {#1} }
5056   { \__text_use_i_delimit_by_s_recursion_stop:nw }
5057   { \use_none:n }
5058 }
5059 \cs_new_eq:NN \__text_sep: \__kernel_int_sep:
5060 \group_begin:
5061   \char_set_catcode_active:n { 0 }
5062   \cs_new:Npn \__text_token_to_explicit:N #1
5063   {
5064     \if_catcode:w \exp_not:N #1
5065     \if_catcode:w \scan_stop: \exp_not:N #1
5066     \scan_stop:
5067     \else:
5068     \exp_not:N ^^@
5069     \fi:
5070     \exp_after:wN \__text_token_to_explicit_cs:N
5071   \else:
5072     \exp_after:wN \__text_token_to_explicit_char:N

```

```
5073     \fi:
5074     #1
5075 }
5076 \group_end:
5077 \cs_new:Npn \__text_token_to_explicit_cs:N #1
5078 {
5079     \exp_after:wN \if_meaning:w \exp_not:N #1 #1
5080     \exp_after:wN \use:nn \exp_after:wN
5081         \__text_token_to_explicit_cs_aux:N
5082     \else:
5083         \exp_after:wN \exp_not:n
5084     \fi:
5085     {#1}
5086 }
5087 \cs_new:Npn \__text_token_to_explicit_cs_aux:N #1
5088 {
5089     \bool_lazy_or:nnTF
5090     { \token_if_chardef_p:N #1 }
5091     { \token_if_mathchardef_p:N #1 }
5092     {
5093         \char_generate:nn {#1}
5094         {
5095             \if_int_compare:w \char_value_catcode:n {#1} = 10 \exp_stop_f:
5096                 10
5097             \else:
5098                 12
5099             \fi:
5100         }
5101     }
5102     {#1}
5103 }
5104 \cs_new:Npn \__text_token_to_explicit_char:N #1
5105 {
5106     \if:w
5107         \if_catcode:w ^ \exp_args:No \str_tail:n { \token_to_str:N #1 } ^
5108         \token_to_str:N #1 #1
5109     \else:
5110         AB
5111     \fi:
5112     \exp_after:wN \exp_not:n
5113 \else:
5114     \exp_after:wN \__text_token_to_explicit:n
5115 \fi:
5116     {#1}
5117 }
5118 \cs_new:Npn \__text_token_to_explicit:n #1
```

```

5119 {
5120 \exp_after:wN \__text_token_to_explicit_auxi:w
5121 \int_value:w
5122 \if_catcode:w \c_group_begin_token #1 1 \else:
5123 \if_catcode:w \c_group_end_token #1 2 \else:
5124 \if_catcode:w \c_math_toggle_token #1 3 \else:
5125 \if_catcode:w ## #1 6 \else:
5126 \if_catcode:w ^ #1 7 \else:
5127 \if_catcode:w \c_math_subscript_token #1 8 \else:
5128 \if_catcode:w \c_space_token #1 10 \else:
5129 \if_catcode:w A #1 11 \else:
5130 \if_catcode:w + #1 12 \else:
5131 4 \fi: \fi: \fi: \fi: \fi: \fi: \fi: \fi:
5132 \exp_after:wN \__text_sep:
5133 \token_to_meaning:N #1 \s__text_stop
5134 }
5135 \cs_new:Npn \__text_token_to_explicit_auxi:w #1 \__text_sep: #2 \s__text_stop
5136 {
5137 \char_generate:nn
5138 {
5139 \if_int_compare:w #1 < 9 \exp_stop_f:
5140 \exp_after:wN \__text_token_to_explicit_auxii:w
5141 \else:
5142 \exp_after:wN \__text_token_to_explicit_auxiii:w
5143 \fi:
5144 #2
5145 }
5146 {#1}
5147 }
5148 \exp_last_unbraced:NNNN \cs_new:Npn \__text_token_to_explicit_auxii:w
5149 #1 { \tl_to_str:n { character ~ } } { ` }
5150 \cs_new:Npn \__text_token_to_explicit_auxiii:w #1 ~ #2 ~ { ` }
5151 \cs_new:Npn \__text_char_catcode:N #1
5152 {
5153 \if_catcode:w \exp_not:N #1 \c_math_toggle_token
5154 3
5155 \else:
5156 \if_catcode:w \exp_not:N #1 \c_alignment_token
5157 4
5158 \else:
5159 \if_catcode:w \exp_not:N #1 \c_math_superscript_token
5160 7
5161 \else:
5162 \if_catcode:w \exp_not:N #1 \c_math_subscript_token
5163 8
5164 \else:

```

```
5165         \if_catcode:w \exp_not:N #1 \c_space_token
5166         10
5167     \else:
5168         \if_catcode:w \exp_not:N #1 \c_catcode_letter_token
5169         11
5170     \else:
5171         \if_catcode:w \exp_not:N #1 \c_catcode_other_token
5172         12
5173     \else:
5174         13
5175     \fi:
5176     \fi:
5177     \fi:
5178     \fi:
5179     \fi:
5180     \fi:
5181     \fi:
5182 }
5183 \prg_new_conditional:Npnn \__text_if_expandable:N #1 { T , F , TF }
5184 {
5185     \token_if_expandable:NTF #1
5186     {
5187         \bool_lazy_any:nTF
5188         {
5189             { \token_if_protected_macro_p:N #1 }
5190             { \token_if_protected_long_macro_p:N #1 }
5191             { \token_if_eq_meaning_p:NN \q__text_recursion_tail #1 }
5192         }
5193         { \prg_return_false: }
5194         { \prg_return_true: }
5195     }
5196     { \prg_return_false: }
5197 }
5198 \sys_if_engine_opentype:TF
5199 {
5200     \cs_new:Npn \__text_codepoint_process:nN #1#2 { #1 {#2} }
5201 }
5202 {
5203     \cs_new:Npe \__text_codepoint_process:nN #1#2
5204     {
5205         \exp_not:N \int_compare:nNnTF {`#2} > { "80 }
5206         {
5207             \sys_if_engine_pdftex:TF
5208             { \exp_not:N \__text_codepoint_process_aux:nN }
5209             {
5210                 \exp_not:N \int_compare:nNnTF {`#2} > { "FF }
```

```
5211         { \exp_not:N \use:n }
5212         { \exp_not:N \__text_codepoint_process_aux:nN }
5213     }
5214 }
5215 { \exp_not:N \use:n }
5216 {#1} #2
5217 }
5218 \cs_new:Npn \__text_codepoint_process_aux:nN #1#2
5219 {
5220     \int_compare:nNnTF { `#2 } < { "E0 }
5221     { \__text_codepoint_process:nNN }
5222     {
5223         \int_compare:nNnTF { `#2 } < { "F0 }
5224         { \__text_codepoint_process:nNNN }
5225         { \__text_codepoint_process:nNNNN }
5226     }
5227     {#1} #2
5228 }
5229 \cs_new:Npn \__text_codepoint_process:nNN #1#2#3
5230 { #1 {#2#3} }
5231 \cs_new:Npn \__text_codepoint_process:nNNN #1#2#3#4
5232 { #1 {#2#3#4} }
5233 \cs_new:Npn \__text_codepoint_process:nNNNN #1#2#3#4#5
5234 { #1 {#2#3#4#5} }
5235 }
5236 \sys_if_engine_opentype:TF
5237 {
5238     \prg_new_conditional:Npnn
5239     \__text_codepoint_compare:nNn #1#2#3 { TF , p }
5240     {
5241         \int_compare:nNnTF { `#1 } #2 {#3}
5242         \prg_return_true: \prg_return_false:
5243     }
5244     \cs_new:Npn \__text_codepoint_from_chars:Nw #1 {`#1}
5245 }
5246 {
5247     \prg_new_conditional:Npnn
5248     \__text_codepoint_compare:nNn #1#2#3 { TF , p }
5249     {
5250         \int_compare:nNnTF { \__text_codepoint_from_chars:Nw #1 }
5251         #2 {#3}
5252         \prg_return_true: \prg_return_false:
5253     }
5254     \cs_new:Npe \__text_codepoint_from_chars:Nw #1
5255     {
5256         \exp_not:N \if_int_compare:w `#1 > "80 \exp_not:N \exp_stop_f:
```

```

5257 \sys_if_engine_pdftex:TF
5258 {
5259     \exp_not:N \exp_after:wN
5260     \exp_not:N \__text_codepoint_from_chars_aux:Nw
5261 }
5262 {
5263     \exp_not:N \if_int_compare:w `#1 > "FF \exp_not:N \exp_stop_f:
5264     \exp_not:N \exp_after:wN \exp_not:N \exp_after:wN
5265     \exp_not:N \exp_after:wN
5266     \exp_not:N \__text_codepoint_from_chars:N
5267 \exp_not:N \else:
5268     \exp_not:N \exp_after:wN \exp_not:N \exp_after:wN
5269     \exp_not:N \exp_after:wN
5270     \exp_not:N \__text_codepoint_from_chars_aux:Nw
5271 \exp_not:N \fi:
5272 }
5273 \exp_not:N \else:
5274     \exp_not:N \exp_after:wN \exp_not:N \__text_codepoint_from_chars:N
5275 \exp_not:N \fi:
5276 #1
5277 }
5278 \cs_new:Npn \__text_codepoint_from_chars_aux:Nw #1
5279 {
5280     \if_int_compare:w `#1 < "E0 \exp_stop_f:
5281     \exp_after:wN \__text_codepoint_from_chars:NN
5282 \else:
5283     \if_int_compare:w `#1 < "F0 \exp_stop_f:
5284     \exp_after:wN \exp_after:wN \exp_after:wN
5285     \__text_codepoint_from_chars:NNN
5286 \else:
5287     \exp_after:wN \exp_after:wN \exp_after:wN
5288     \__text_codepoint_from_chars:NNNN
5289 \fi:
5290 \fi:
5291 #1
5292 }
5293 \cs_new:Npn \__text_codepoint_from_chars:N #1 {`#1}
5294 \cs_new:Npn \__text_codepoint_from_chars:NN #1#2
5295 { (`#1 - "C0) * "40 + `#2 - "80 }
5296 \cs_new:Npn \__text_codepoint_from_chars:NNN #1#2#3
5297 { (`#1 - "E0) * "1000 + (`#2 - "80) * "40 + `#3 - "80 }
5298 \cs_new:Npn \__text_codepoint_from_chars:NNNN #1#2#3#4
5299 {
5300     (`#1 - "F0) * "40000
5301     + (`#2 - "80) * "1000
5302     + (`#3 - "80) * "40

```

```

5303 + `#4 - "80
5304 }
5305 }
5306 \tl_new:N \l_text_accents_tl
5307 \tl_new:N \l_text_letterlike_tl
5308 \tl_new:N \l_text_case_exclude_arg_tl
5309 \tl_set:Nc \l_text_case_exclude_arg_tl
5310 {
5311   \exp_not:n { \begin \cite \end \label \ref }
5312   \exp_not:c { cite ~ }
5313   \exp_not:n { \babelshorthand }
5314 }
5315 \tl_new:N \l_text_math_arg_tl
5316 \tl_set:Nn \l_text_math_arg_tl { \ensuremath }
5317 \tl_new:N \l_text_math_delims_tl
5318 \tl_set:Nn \l_text_math_delims_tl { $ $ \ ( \ ) }
5319 \tl_new:N \l_text_expand_exclude_tl
5320 \tl_set:Nn \l_text_expand_exclude_tl
5321 { \begin \cite \end \label \ref }
5322 \bool_lazy_and:nnT
5323 { \str_if_eq_p:Vn \fmtname { LaTeX2e } }
5324 { \tl_if_exist_p:N \@expl@finalise@setup@@ }
5325 {
5326   \tl_gput_right:Nn \@expl@finalise@setup@@
5327   {
5328     \tl_gput_right:Nn \@kernel@after@begindocument
5329     {
5330       \group_begin:
5331       \cs_set_protected:Npn \__text_tmp:w #1
5332       {
5333         \tl_clear:N \l_text_expand_exclude_tl
5334         \tl_map_inline:nn {#1}
5335         {
5336           \bool_lazy_any:nF
5337           {
5338             { \token_if_protected_macro_p:N ##1 }
5339             { \token_if_protected_long_macro_p:N ##1 }
5340             {
5341               \str_if_eq_p:ee
5342               { \cs_replacement_spec:N ##1 }
5343               { \exp_not:n { \protect ##1 } \c_space_tl }
5344             }
5345           }
5346           { \tl_put_right:Nn \l_text_expand_exclude_tl {##1} }
5347         }
5348       }

```



```
5349 \exp_args:NV \__text_tmp:w \l_text_expand_exclude_tl 3534
5350 \exp_args:NNNV \group_end: 3535
5351 \tl_set:Nn \l_text_expand_exclude_tl \l_text_expand_exclude_tl 3536
5352 } 3537
5353 } 3538
5354 } 3539
5355 \tl_new:N \l__text_math_mode_tl 3540
5356 \tex_global:D \tex_chardef:D \c__text_chardef_space_token = `\_ % 3541
5357 \tex_global:D \tex_mathchardef:D \c__text_mathchardef_space_token = `\_ % 3542
5358 \tex_global:D \tex_chardef:D \c__text_chardef_group_begin_token = `{ % \} 3543
5359 \tex_global:D \tex_mathchardef:D \c__text_mathchardef_group_begin_token = `{ % \} \{ 3544
5360 \tex_global:D \tex_chardef:D \c__text_chardef_group_end_token = `}% % \{ 3545
5361 \tex_global:D \tex_mathchardef:D \c__text_mathchardef_group_end_token = `}% % 3546
5362 \cs_new:Npn \text_expand:n #1 3547
5363 { 3548
5364 \__kernel_exp_not:w \exp_after:wN 3549
5365 { 3550
5366 \exp:w 3551
5367 \__text_expand:n {#1} 3552
5368 } 3553
5369 } 3554
5370 \cs_new:Npn \__text_expand:n #1 3555
5371 { 3556
5372 \group_align_safe_begin: 3557
5373 \__text_expand_loop:w #1 3558
5374 \s__text_recursion_tail \s__text_recursion_stop 3559
5375 \__text_expand_result:n { } 3560
5376 } 3561
5377 \cs_new:Npn \__text_expand_store:n #1 3562
5378 { \__text_expand_store:nw {#1} } 3563
5379 \cs_generate_variant:Nn \__text_expand_store:n { o } 3564
5380 \cs_new:Npn \__text_expand_store:nw #1#2 \__text_expand_result:n #3 3565
5381 { #2 \__text_expand_result:n { #3 #1 } } 3566
5382 \cs_new:Npn \__text_expand_end:w #1 \__text_expand_result:n #2 3567
5383 { 3568
5384 \group_align_safe_end: 3569
5385 \exp_end: 3570
5386 #2 3571
5387 } 3572
5388 \cs_new:Npn \__text_expand_loop:w #1 \s__text_recursion_stop 3573
5389 { 3574
5390 \tl_if_head_is_N_type:nTF {#1} 3575
5391 { \__text_expand_N_type:N } 3576
5392 { 3577
5393 \tl_if_head_is_group:nTF {#1} 3578
5394 { \__text_expand_group:n } 3579
```

```
5395         { \_text_expand_space:w }
5396     }
5397     #1 \s__text_recursion_stop
5398 }
5399 \cs_new:Npn \_text_expand_group:n #1
5400 {
5401     \_text_expand_store:o
5402     {
5403         \exp_after:wN
5404         {
5405             \exp:w
5406             \_text_expand:n {#1}
5407         }
5408     }
5409     \_text_expand_loop:w
5410 }
5411 \exp_last_unbraced:NNo \cs_new:Npn \_text_expand_space:w \c_space_tl
5412 {
5413     \_text_expand_store:n { ~ }
5414     \_text_expand_loop:w
5415 }
5416 \cs_new:Npn \_text_expand_N_type:N #1
5417 {
5418     \_text_if_s_recursion_tail_stop_do:Nn #1
5419     { \_text_expand_end:w }
5420     \exp_after:wN \_text_expand_math_search:NNN
5421     \exp_after:wN #1 \l_text_math_delims_tl
5422     \q_text_recursion_tail \q_text_recursion_tail
5423     \q_text_recursion_stop
5424 }
5425 \cs_new:Npn \_text_expand_math_search:NNN #1#2#3
5426 {
5427     \_text_if_q_recursion_tail_stop_do:Nn #2
5428     { \_text_expand_explicit:N #1 }
5429     \token_if_eq_meaning:NNTF #1 #2
5430     {
5431         \_text_use_i_delimit_by_q_recursion_stop:nw
5432         {
5433             \_text_expand_store:n {#1}
5434             \_text_expand_math_loop:Nw #3
5435         }
5436     }
5437     { \_text_expand_math_search:NNN #1 }
5438 }
5439 \cs_new:Npn \_text_expand_math_loop:Nw #1#2 \s__text_recursion_stop
5440 {
```

```

5441 \tl_if_head_is_N_type:nTF {#2}
5442 { \__text_expand_math_N_type:NN }
5443 {
5444 \tl_if_head_is_group:nTF {#2}
5445 { \__text_expand_math_group:Nn }
5446 { \__text_expand_math_space:Nw }
5447 }
5448 #1#2 \s__text_recursion_stop
5449 }
5450 \cs_new:Npn \__text_expand_math_N_type:NN #1#2
5451 {
5452 \__text_if_s_recursion_tail_stop_do:Nn #2
5453 { \__text_expand_end:w }
5454 \token_if_eq_meaning:NNTF #2 \exp_not:N
5455 { \__text_expand_store:n {#2} }
5456 \token_if_eq_meaning:NNTF #2 #1
5457 { \__text_expand_loop:w }
5458 { \__text_expand_math_loop:Nw #1 }
5459 }
5460 \cs_new:Npn \__text_expand_math_group:Nn #1#2
5461 {
5462 \__text_expand_store:n { {#2} }
5463 \__text_expand_math_loop:Nw #1
5464 }
5465 \exp_after:wN \cs_new:Npn \exp_after:wN \__text_expand_math_space:Nw
5466 \exp_after:wN # \exp_after:wN 1 \c_space_tl
5467 {
5468 \__text_expand_store:n { ~ }
5469 \__text_expand_math_loop:Nw #1
5470 }
5471 \cs_new:Npn \__text_expand_explicit:N #1
5472 {
5473 \token_if_cs:NNTF #1
5474 { \__text_expand_exclude:N #1 }
5475 {
5476 \bool_lazy_and:nnTF
5477 { \token_if_active_p:N #1 }
5478 {
5479 ! \bool_lazy_any_p:n
5480 {
5481 { \token_if_protected_macro_p:N #1 }
5482 { \token_if_protected_long_macro_p:N #1 }
5483 { \tl_if_head_eq_meaning_p:oN {#1} \UTFviii@two@octets }
5484 { \tl_if_head_eq_meaning_p:oN {#1} \UTFviii@three@octets }
5485 { \tl_if_head_eq_meaning_p:oN {#1} \UTFviii@four@octets }
5486 { \tl_if_head_eq_meaning_p:oN {#1} \active@prefix }

```

```
5487     }
5488 }
5489 { \exp_after:wN \__text_expand_loop:w #1 }
5490 {
5491     \__text_expand_store:n {#1}
5492     \__text_expand_loop:w
5493 }
5494 }
5495 }
5496 \cs_new:Npn \__text_expand_exclude:N #1
5497 {
5498     \cs_if_eq:NNTF #1 \text_case_switch:nnnn
5499     { \__text_expand_exclude_switch:Nnnnn #1 }
5500     {
5501         \exp_args:Ne \__text_expand_exclude:nN
5502         {
5503             \exp_not:V \l_text_math_arg_tl
5504             \exp_not:V \l_text_expand_exclude_tl
5505             \exp_not:V \l_text_case_exclude_arg_tl
5506         }
5507         #1
5508     }
5509 }
5510 \cs_new:Npn \__text_expand_exclude_switch:Nnnnn #1#2#3#4#5
5511 {
5512     \__text_expand_store:n { #1 {#2} {#3} {#4} {#5} }
5513     \__text_expand_loop:w
5514 }
5515 \cs_new:Npn \__text_expand_exclude:nN #1#2
5516 {
5517     \__text_expand_exclude:NN #2 #1
5518     \q__text_recursion_tail \q__text_recursion_stop
5519 }
5520 \cs_new:Npn \__text_expand_exclude:NN #1#2
5521 {
5522     \__text_if_q_recursion_tail_stop_do:Nn #2
5523     { \__text_expand_accent:N #1 }
5524     \str_if_eq:nnTF {#1} {#2}
5525     {
5526         \__text_use_i_delimit_by_q_recursion_stop:nw
5527         { \__text_expand_exclude:Nw #1 }
5528     }
5529     { \__text_expand_exclude:NN #1 }
5530 }
5531 \cs_new:Npn \__text_expand_exclude:Nw #1#2#
5532 { \__text_expand_exclude:Nnn #1 {#2} }
```

```
5533 \cs_new:Npn \__text_expand_exclude:Nnn #1#2#3
5534 {
5535     \__text_expand_store:n { #1#2 {#3} }
5536     \__text_expand_loop:w
5537 }
5538 \cs_new:Npn \__text_expand_accent:N #1
5539 {
5540     \exp_after:wN \__text_expand_accent:NN \exp_after:wN
5541     #1 \l_text_accents_tl
5542     \q__text_recursion_tail \q__text_recursion_stop
5543 }
5544 \cs_new:Npn \__text_expand_accent:NN #1#2
5545 {
5546     \__text_if_q_recursion_tail_stop_do:Nn #2
5547     { \__text_expand_letterlike:N #1 }
5548     \cs_if_eq:NNTF #2 #1
5549     {
5550         \__text_use_i_delimit_by_q_recursion_stop:nw
5551         {
5552             \__text_expand_store:n {#1}
5553             \__text_expand_loop:w
5554         }
5555     }
5556     { \__text_expand_accent:NN #1 }
5557 }
5558 \cs_new:Npn \__text_expand_letterlike:N #1
5559 {
5560     \exp_after:wN \__text_expand_letterlike:NN \exp_after:wN
5561     #1 \l_text_letterlike_tl
5562     \q__text_recursion_tail \q__text_recursion_stop
5563 }
5564 \cs_new:Npn \__text_expand_letterlike:NN #1#2
5565 {
5566     \__text_if_q_recursion_tail_stop_do:Nn #2
5567     { \__text_expand_cs:N #1 }
5568     \cs_if_eq:NNTF #2 #1
5569     {
5570         \__text_use_i_delimit_by_q_recursion_stop:nw
5571         {
5572             \__text_expand_store:n {#1}
5573             \__text_expand_loop:w
5574         }
5575     }
5576     { \__text_expand_letterlike:NN #1 }
5577 }
5578 \cs_new:Npe \__text_expand_cs:N #1
```

```
5579 { 35579
5580 \exp_not:N \str_if_eq:nnTF {#1} { \exp_not:N \protect } 35580
5581 { \exp_not:N \__text_expand_protect:w } 35581
5582 { 35582
5583 \bool_lazy_and:nnTF 35583
5584 { \cs_if_exist_p:N \fmtname } 35584
5585 { \str_if_eq_p:Vn \fmtname { LaTeX2e } } 35585
5586 { \exp_not:N \__text_expand_testopt:N #1 } 35586
5587 { \exp_not:N \__text_expand_replace:N #1 } 35587
5588 } 35588
5589 } 35589
5590 \cs_new:Npn \__text_expand_protect:w #1 \s__text_recursion_stop 35590
5591 { 35591
5592 \tl_if_head_is_N_type:nTF {#1} 35592
5593 { \__text_expand_protect:N } 35593
5594 { 35594
5595 \__text_expand_store:n { \protect } 35595
5596 \__text_expand_loop:w 35596
5597 } 35597
5598 #1 \s__text_recursion_stop 35598
5599 } 35599
5600 \cs_new:Npn \__text_expand_protect:N #1 35600
5601 { 35601
5602 \__text_if_s_recursion_tail_stop_do:Nn #1 35602
5603 { 35603
5604 \__text_expand_store:n { \protect } 35604
5605 \__text_expand_end:w 35605
5606 } 35606
5607 \exp_args:Ne \__text_expand_protect:nN 35607
5608 { \cs_to_str:N #1 } #1 35608
5609 } 35609
5610 \cs_new:Npn \__text_expand_protect:nN #1#2 35610
5611 { \__text_expand_protect:Nw #2 #1 \q__text_nil #1 ~ \q__text_nil \q__text_nil ✓ 35611
5612 \s__text_stop } 35612
5613 \cs_new:Npn \__text_expand_protect:Nw #1 #2 ~ \q__text_nil #3 \q__text_nil #4 ✓ 35613
5614 \s__text_stop 35614
5615 { 35615
5616 \__text_quark_if_nil:nTF {#4} 35616
5617 { 35617
5618 \cs_if_exist:cTF {#2} 35618
5619 { \exp_args:Ne \__text_expand_store:n { \exp_not:c {#2} } } 35619
5620 { \__text_expand_store:n { \protect #1 } } 35620
5621 } 35621
5622 \__text_expand_loop:w 35622
5623 } 35623
```

```
5623 \cs_new:Npn \__text_expand_testopt:N #1
5624 {
5625     \token_if_eq_meaning:NNTF #1 \@protected@testopt
5626     { \__text_expand_testopt:NNn }
5627     { \__text_expand_encoding:N #1 }
5628 }
5629 \cs_new:Npn \__text_expand_testopt:NNn #1#2#3
5630 {
5631     \__text_expand_store:n {#1}
5632     \__text_expand_loop:w
5633 }
5634 \cs_new:Npn \__text_expand_encoding:N #1
5635 {
5636     \bool_lazy_or:nnTF
5637     { \cs_if_eq_p:NN #1 \@current@cmd }
5638     { \cs_if_eq_p:NN #1 \@changed@cmd }
5639     { \exp_after:wN \__text_expand_loop:w \__text_expand_encoding_escape:NN }
5640     { \__text_expand_replace:N #1 }
5641 }
5642 \cs_new:Npn \__text_expand_encoding_escape:NN #1#2 { \exp_not:n {#1} }
5643 \cs_new:Npn \__text_expand_replace:N #1
5644 {
5645     \bool_lazy_and:nnTF
5646     { \cs_if_exist_p:c { l__text_expand_ \token_to_str:N #1 _tl } }
5647     {
5648         \bool_lazy_or_p:nn
5649         { \token_if_cs_p:N #1 }
5650         { \token_if_active_p:N #1 }
5651     }
5652     {
5653         \exp_args:Nv \__text_expand_replace:n
5654         { l__text_expand_ \token_to_str:N #1 _tl }
5655     }
5656     { \__text_expand_cs_expand:N #1 }
5657 }
5658 \cs_new:Npn \__text_expand_replace:n #1 { \__text_expand_loop:w #1 }
5659 \cs_new:Npn \__text_expand_cs_expand:N #1
5660 {
5661     \__text_if_expandable:NNTF #1
5662     {
5663         \token_if_eq_meaning:NNTF #1 \exp_not:n
5664         { \__text_expand_unexpanded:w }
5665         { \exp_after:wN \__text_expand_loop:w #1 }
5666     }
5667     {
5668         \__text_expand_store:n {#1}
```



```
5669     \_text_expand_loop:w
5670 }
5671 }
5672 \cs_new:Npn \_text_expand_unexpanded:w
5673 {
5674     \exp_after:wN \_text_expand_unexpanded_test:w
5675     \exp:w \exp_end_continue_f:w
5676 }
5677 \cs_new:Npn \_text_expand_unexpanded_test:w #1 \s__text_recursion_stop
5678 {
5679     \tl_if_head_is_group:nTF {#1}
5680     { \_text_expand_unexpanded:n }
5681     {
5682         \_text_expand_unexpanded:w
5683         \tl_if_head_is_N_type:nT {#1} { \_text_expand_unexpanded:N }
5684     }
5685     #1 \s__text_recursion_stop
5686 }
5687 \cs_new:Npn \_text_expand_unexpanded:N #1
5688 {
5689     \exp_after:wN \if_meaning:w \exp_not:N #1 #1
5690     \else:
5691         \exp_after:wN #1
5692     \fi:
5693 }
5694 \cs_new:Npn \_text_expand_unexpanded:n #1
5695 {
5696     \_text_expand_store:n {#1}
5697     \_text_expand_loop:w
5698 }
5699 \cs_new_protected:Npn \text_declare_expand_equivalent:Nn #1#2
5700 {
5701     \tl_clear_new:c { l__text_expand_ \token_to_str:N #1 _tl }
5702     \tl_set:cn { l__text_expand_ \token_to_str:N #1 _tl } {#2}
5703 }
5704 \cs_generate_variant:Nn \text_declare_expand_equivalent:Nn { c }
5705 \tl_map_inline:nn
5706 { \_ \_ \_ \^ \~ \= \u \. \" \r \H \v \d \c \k \b \t }
5707 { \text_declare_expand_equivalent:Nn #1 { \exp_not:n {#1} } }
5708 \tl_map_inline:nn
5709 {
5710     \AA \aa
5711     \AE \ae
5712     \DH \dh
5713     \DJ \dj
5714     \IJ \ij
```

```
5715 \L \l 35715
5716 \NG \ng 35716
5717 \O \o 35717
5718 \OE \oe 35718
5719 \SS \ss 35719
5720 \TH \th 35720
5721 } 35721
5722 { \text_declare_expand_equivalent:Nn #1 { \exp_not:n {#1} } } 35722
5723 %% File: l3text-case.dtx 35723
5724 \bool_new:N \l_text_titlecase_check_letter_bool 35724
5725 \bool_set_true:N \l_text_titlecase_check_letter_bool 35725
5726 \cs_new:Npn \text_lowercase:n #1 35726
5727 { \__text_change_case:nnn { lower } { } {#1} } 35727
5728 \cs_new:Npn \text_uppercase:n #1 35728
5729 { \__text_change_case:nnn { upper } { } {#1} } 35729
5730 \cs_new:Npn \text_titlecase_all:n #1 35730
5731 { \__text_change_case:nnn { title } { } {#1} } 35731
5732 \cs_new:Npn \text_titlecase_first:n #1 35732
5733 { \__text_change_case:nnnn { title } { break } { } {#1} } 35733
5734 \cs_new:Npn \text_lowercase:nn #1#2 35734
5735 { \__text_change_case:nnn { lower } {#1} {#2} } 35735
5736 \cs_new:Npn \text_uppercase:nn #1#2 35736
5737 { \__text_change_case:nnn { upper } {#1} {#2} } 35737
5738 \cs_new:Npn \text_titlecase_all:nn #1#2 35738
5739 { \__text_change_case:nnn { title } {#1} {#2} } 35739
5740 \cs_new:Npn \text_titlecase_first:nn #1#2 35740
5741 { \__text_change_case:nnnn { title } { break } {#1} {#2} } 35741
5742 \cs_new:Npn \__text_change_case:nnn #1#2#3 35742
5743 { \__text_change_case:nnnn {#1} {#1} {#2} {#3} } 35743
5744 \cs_new:Npn \__text_change_case:nnnn #1#2#3#4 35744
5745 { 35745
5746 \__kernel_exp_not:w \exp_after:wN 35746
5747 { 35747
5748 \exp:w 35748
5749 \exp_args:Ne \__text_change_case_auxi:nnnn 35749
5750 { \text_expand:n {#4} } 35750
5751 {#1} {#2} {#3} 35751
5752 } 35752
5753 } 35753
5754 \cs_new:Npn \__text_change_case_auxi:nnnn #1#2#3#4 35754
5755 { 35755
5756 \exp_args:No \__text_change_case_BCP:nnnn 35756
5757 { \tl_to_str:n {#4} } {#1} {#2} {#3} 35757
5758 } 35758
5759 \cs_new:Npe \__text_change_case_BCP:nnnn #1#2#3#4 35759
5760 { 35760
```

```
5761 \exp_not:N \__text_change_case_BCP:nnnw 3576
5762 {#2} {#3} {#4} #1 \tl_to_str:n { -x- -x- } \exp_not:N \q__text_stop 3576
5763 } 3576
5764 \use:e 3576
5765 { 3576
5766 \cs_new:Npn \exp_not:N \__text_change_case_BCP:nnnw 3576
5767 #1#2#3#4 \tl_to_str:n { -x- } #5 \tl_to_str:n { -x- } #6 3576
5768 \exp_not:N \q__text_stop 3576
5769 } 3576
5770 { \__text_change_case_BCP:nnnnnw {#1} {#2} {#3} {#5} {#4} #4 - \q__text_stop } 3577
5771 \cs_new:Npn \__text_change_case_BCP:nnnnnw #1#2#3#4#5#6 - #7 \q__text_stop 3577
5772 { 3577
5773 \bool_lazy_or:nnTF 3577
5774 { \cs_if_exist_p:c { __text_change_case_ #2 _ #6 -x- #4 :nnnnn } } 3577
5775 { \tl_if_exist_p:c { l__text_ #2 case_special_ #6 -x- #4 _tl } } 3577
5776 { \__text_change_case_auxii:nnnn {#1} {#2} {#3} { #6 -x- #4 } } 3577
5777 { 3577
5778 \cs_if_exist:cTF { __text_change_case_ #2 _ #6 :nnnnn } 3577
5779 { \__text_change_case_auxii:nnnn {#1} {#2} {#3} {#6} } 3577
5780 { \__text_change_case_auxii:nnnn {#1} {#2} {#3} {#5} } 3578
5781 } 3578
5782 } 3578
5783 \cs_new:Npn \__text_change_case_auxii:nnnn #1#2#3#4 3578
5784 { 3578
5785 \group_align_safe_begin: 3578
5786 \cs_if_exist_use:c { __text_change_case_boundary_ #2 _ #4 :Nnnnw } 3578
5787 \__text_change_case_loop:nnnw {#2} {#3} {#4} #1 3578
5788 \q__text_recursion_tail \q__text_recursion_stop 3578
5789 \__text_change_case_result:n { } 3578
5790 } 3579
5791 \cs_new:Npn \__text_change_case_store:n #1 3579
5792 { \__text_change_case_store:nw {#1} } 3579
5793 \cs_generate_variant:Nn \__text_change_case_store:n { o , e , V , v } 3579
5794 \cs_new:Npn \__text_change_case_store:nw #1#2 \__text_change_case_result:n #3 3579
5795 { #2 \__text_change_case_result:n { #3 #1 } } 3579
5796 \cs_new:Npn \__text_change_case_end:w #1 \__text_change_case_result:n #2 3579
5797 { 3579
5798 \group_align_safe_end: 3579
5799 \exp_end: 3579
5800 #2 3580
5801 } 3580
5802 \cs_new:Npn \__text_change_case_loop:nnnw #1#2#3#4 \q__text_recursion_stop 3580
5803 { 3580
5804 \tl_if_head_is_N_type:nTF {#4} 3580
5805 { \__text_change_case_N_type:nnnN } 3580
5806 { 3580
```

```
5807         \tl_if_head_is_group:nTF {#4}
5808         { \use:c { __text_change_case_group_ #1 :nnnn } }
5809         { \__text_change_case_space:nnnw }
5810     }
5811     {#1} {#2} {#3} #4 \q__text_recursion_stop
5812 }
5813 \cs_new:Npn \__text_change_case_break:w
5814 { \__text_change_case_break_aux:w \prg_do_nothing: }
5815 \cs_new:Npn \__text_change_case_break_aux:w
5816 #1 \q__text_recursion_tail \q__text_recursion_stop
5817 {
5818     \__text_change_case_store:o {#1}
5819     \__text_change_case_end:w
5820 }
5821 \cs_new:Npn \__text_change_case_group_lower:nnnn #1#2#3#4
5822 {
5823     \__text_change_case_store:o
5824     {
5825         \exp_after:wN
5826         {
5827             \exp:w
5828             \__text_change_case_auxii:nnnn {#4} {#1} {#2} {#3}
5829         }
5830     }
5831     \__text_change_case_loop:nnnw {#1} {#2} {#3}
5832 }
5833 \cs_new_eq:NN \__text_change_case_group_upper:nnnn
5834 \__text_change_case_group_lower:nnnn
5835 \cs_new:Npn \__text_change_case_group_title:nnnn #1#2#3#4
5836 {
5837     \__text_change_case_store:o
5838     {
5839         \exp_after:wN
5840         {
5841             \exp:w
5842             \__text_change_case_auxii:nnnn {#4} {#1} {#2} {#3}
5843         }
5844     }
5845     \__text_change_case_skip:nnw {#2} {#3}
5846 }
5847 \use:e
5848 {
5849     \cs_new:Npn \exp_not:N \__text_change_case_space:nnnw #1#2#3 \c_space_tl
5850 }
5851 {
5852     \__text_change_case_store:n { ~ }
```

```
5853 \cs_if_exist_use:cF { __text_change_case_space_ #2 :nnn } 3585
5854 { 3585
5855 \cs_if_exist_use:c { __text_change_case_boundary_ #1 _ #3 :Nnnnw } 3585
5856 \__text_change_case_loop:nnnw 3585
5857 } 3585
5858 {#2} {#2} {#3} 3585
5859 } 3585
5860 \cs_new:Npn \__text_change_case_space_break:nnn #1#2#3 3586
5861 { \__text_change_case_break:w } 3586
5862 \cs_new:Npn \__text_change_case_N_type:nnnN #1#2#3#4 3586
5863 { 3586
5864 \__text_if_q_recursion_tail_stop_do:Nn #4 3586
5865 { \__text_change_case_end:w } 3586
5866 \__text_change_case_N_type_aux:nnnN {#1} {#2} {#3} #4 3586
5867 } 3586
5868 \cs_new:Npn \__text_change_case_N_type_aux:nnnN #1#2#3#4 3586
5869 { 3586
5870 \exp_args:NV \__text_change_case_N_type:nnnnN 3587
5871 \l_text_math_delims_tl {#1} {#2} {#3} #4 3587
5872 } 3587
5873 \cs_new:Npn \__text_change_case_N_type:nnnnN #1#2#3#4#5 3587
5874 { 3587
5875 \__text_change_case_math_search:nnnNNN {#2} {#3} {#4} #5 #1 3587
5876 \q__text_recursion_tail \q__text_recursion_tail 3587
5877 \q__text_recursion_stop 3587
5878 } 3587
5879 \cs_new:Npn \__text_change_case_math_search:nnnNNN #1#2#3#4#5#6 3587
5880 { 3588
5881 \__text_if_q_recursion_tail_stop_do:Nn #5 3588
5882 { \__text_change_case_cs_check:nnnN {#1} {#2} {#3} #4 } 3588
5883 \token_if_eq_meaning:NNTF #4 #5 3588
5884 { 3588
5885 \__text_use_i_delimit_by_q_recursion_stop:nw 3588
5886 { 3588
5887 \__text_change_case_store:n {#4} 3588
5888 \__text_change_case_math_loop:nnnNw {#1} {#2} {#3} #6 3588
5889 } 3588
5890 } 3589
5891 { \__text_change_case_math_search:nnnNNN {#1} {#2} {#3} #4 } 3589
5892 } 3589
5893 \cs_new:Npn \__text_change_case_math_loop:nnnNw #1#2#3#4#5 \q__text_recursion_stop 3589
5894 { 3589
5895 \tl_if_head_is_N_type:nTF {#5} 3589
5896 { \__text_change_case_math_N_type:nnnNN } 3589
5897 { 3589
5898 \tl_if_head_is_group:nTF {#5} 3589
```

```
5899         { \_text_change_case_math_group:nnnNn } 3589
5900         { \_text_change_case_math_space:nnnNw } 3590
5901     } 3590
5902     {#1} {#2} {#3} #4 #5 \q__text_recursion_stop 3590
5903 } 3590
5904 \cs_new:Npn \_text_change_case_math_N_type:nnnNN #1#2#3#4#5 3590
5905 { 3590
5906     \_text_if_q_recursion_tail_stop_do:Nn #5 3590
5907     { \_text_change_case_end:w } 3590
5908     \_text_change_case_store:n {#5} 3590
5909     \token_if_eq_meaning:NNTF #5 #4 3590
5910     { \_text_change_case_loop:nnnw {#1} {#2} {#3} } 3591
5911     { \_text_change_case_math_loop:nnnNw {#1} {#2} {#3} #4 } 3591
5912 } 3591
5913 \cs_new:Npn \_text_change_case_math_group:nnnNn #1#2#3#4#5 3591
5914 { 3591
5915     \_text_change_case_store:n { {#5} } 3591
5916     \_text_change_case_math_loop:nnnNw {#1} {#2} {#3} #4 3591
5917 } 3591
5918 \use:e 3591
5919 { 3591
5920     \cs_new:Npn \exp_not:N \_text_change_case_math_space:nnnNw #1#2#3#4 3592
5921     \c_space_tl 3592
5922 } 3592
5923 { 3592
5924     \_text_change_case_store:n { ~ } 3592
5925     \_text_change_case_math_loop:nnnNw {#1} {#2} {#3} #4 3592
5926 } 3592
5927 \cs_new:Npn \_text_change_case_cs_check:nnnN #1#2#3#4 3592
5928 { 3592
5929     \token_if_cs:NTF #4 3592
5930     { \_text_change_case_exclude:nnnN {#1} {#2} {#3} } 3593
5931     { 3593
5932         \_text_codepoint_process:nN 3593
5933         { \use:c { \_text_change_case_custom_ #1 :nnnn } {#1} {#2} {#3} } 3593
5934     } 3593
5935     #4 3593
5936 } 3593
5937 \cs_new:Npn \_text_change_case_exclude:nnnN #1#2#3#4 3593
5938 { 3593
5939     \exp_args:Ne \_text_change_case_exclude:nnnnN 3593
5940     { 3594
5941         \exp_not:V \l_text_math_arg_tl 3594
5942         \exp_not:V \l_text_case_exclude_arg_tl 3594
5943     } 3594
5944     {#1} {#2} {#3} #4 3594
```

```
5945 } 35945
5946 \cs_new:Npn \__text_change_case_exclude:nnnnN #1#2#3#4#5 35946
5947 { 35947
5948 \__text_change_case_exclude:nnnNN {#2} {#3} {#4} #5 #1 35948
5949 \q__text_recursion_tail \q__text_recursion_stop 35949
5950 } 35950
5951 \cs_new:Npn \__text_change_case_exclude:nnnNN #1#2#3#4#5 35951
5952 { 35952
5953 \__text_if_q_recursion_tail_stop_do:Nn #5 35953
5954 { \__text_change_case_replace:nnnN {#1} {#2} {#3} #4 } 35954
5955 \str_if_eq:nnTF {#4} {#5} 35955
5956 { 35956
5957 \__text_use_i_delimit_by_q_recursion_stop:nw 35957
5958 { \__text_change_case_exclude:nnnNw {#1} {#2} {#3} #4 } 35958
5959 } 35959
5960 { \__text_change_case_exclude:nnnNN {#1} {#2} {#3} #4 } 35960
5961 } 35961
5962 \cs_new:Npn \__text_change_case_exclude:nnnNw #1#2#3#4#5# 35962
5963 { \__text_change_case_exclude:nnnNnn {#1} {#2} {#3} {#4} {#5} } 35963
5964 \cs_new:Npn \__text_change_case_exclude:nnnNnn #1#2#3#4#5#6 35964
5965 { 35965
5966 \tl_if_blank:nTF {#5} 35966
5967 { \__text_change_case_store:n { #4 {#6} } } 35967
5968 { 35968
5969 \__text_change_case_store:o 35969
5970 { 35970
5971 \exp_after:wN #4 35971
5972 \exp:w \__text_change_case_auxii:nnnn {#5} {#1} {#2} {#3} 35972
5973 {#6} 35973
5974 } 35974
5975 } 35975
5976 \__text_change_case_loop:nnnw {#1} {#2} {#3} 35976
5977 } 35977
5978 \cs_new:Npn \__text_change_case_replace:nnnN #1#2#3#4 35978
5979 { 35979
5980 \cs_if_exist:cTF { l__text_case_ \token_to_str:N #4 _tl } 35980
5981 { 35981
5982 \__text_change_case_replace:vn nn 35982
5983 { l__text_case_ \token_to_str:N #4 _tl } {#1} {#2} {#3} 35983
5984 } 35984
5985 { \__text_change_case_switch:nnnN {#1} {#2} {#3} #4 } 35985
5986 } 35986
5987 \cs_new:Npn \__text_change_case_replace:nnnn #1#2#3#4 35987
5988 { \__text_change_case_loop:nnnw {#2} {#3} {#4} #1 } 35988
5989 \cs_generate_variant:Nn \__text_change_case_replace:nnnn { v } 35989
5990 \cs_new:Npn \__text_change_case_switch:nnnN #1#2#3#4 35990
```



```
5991 {
5992     \cs_if_eq:NNTF #4 \text_case_switch:nnnn
5993     { \use:c { __text_change_case_switch_ #1 :nnnNnnnn } }
5994     { \use:c { __text_change_case_letterlike_ #1 :nnnN } }
5995     {#1} {#2} {#3} #4
5996 }
5997 \cs_new:Npn \__text_change_case_switch_lower:nnnNnnnn #1#2#3#4#5#6#7#8
5998 {
5999     \__text_change_case_store:n {#7}
6000     \__text_change_case_loop:nnnw {#1} {#2} {#3}
6001 }
6002 \cs_new:Npn \__text_change_case_switch_upper:nnnNnnnn #1#2#3#4#5#6#7#8
6003 {
6004     \__text_change_case_store:n {#6}
6005     \__text_change_case_loop:nnnw {#1} {#2} {#3}
6006 }
6007 \cs_new:Npn \__text_change_case_switch_title:nnnNnnnn #1#2#3#4#5#6#7#8
6008 {
6009     \__text_change_case_store:n {#8}
6010     \__text_change_case_skip:nnw {#2} {#3}
6011 }
6012 \cs_new:Npn \__text_change_case_skip:nnw #1#2#3 \q__text_recursion_stop
6013 {
6014     \tl_if_head_is_N_type:nTF {#3}
6015     { \__text_change_case_skip_N_type:nnN }
6016     {
6017         \tl_if_head_is_group:nTF {#3}
6018         { \__text_change_case_skip_group:nnn }
6019         { \__text_change_case_skip_space:nnw }
6020     }
6021     {#1} {#2} #3 \q__text_recursion_stop
6022 }
6023 \cs_new:Npn \__text_change_case_skip_N_type:nnN #1#2#3
6024 {
6025     \__text_if_q_recursion_tail_stop_do:Nn #3
6026     { \__text_change_case_end:w }
6027     \__text_change_case_store:n {#3}
6028     \__text_change_case_skip:nnw {#1} {#2}
6029 }
6030 \cs_new:Npn \__text_change_case_skip_group:nnn #1#2#3
6031 {
6032     \__text_change_case_store:n { {#3} }
6033     \__text_change_case_skip:nnw {#1} {#2}
6034 }
6035 \cs_new:Npn \__text_change_case_skip_space:nnw #1#2
6036 { \__text_change_case_space:nnnw {#1} {#1} {#2} }
```

```
6037 \cs_new:Npn \__text_change_case_letterlike_lower:nnnN #1#2#3#4 3603
6038 { \__text_change_case_letterlike:nnnnN {#1} {#1} {#1} {#2} {#3} #4 } 3603
6039 \cs_new_eq:NN \__text_change_case_letterlike_upper:nnnN 3603
6040 \__text_change_case_letterlike_lower:nnnN 3604
6041 \cs_new:Npn \__text_change_case_letterlike_title:nnnN #1#2#3#4 3604
6042 { \__text_change_case_letterlike:nnnnN { upper } { end } {#1} {#2} {#3} #4 } 3604
6043 \cs_new:Npn \__text_change_case_letterlike:nnnnN #1#2#3#4#5#6 3604
6044 { 3604
6045 \cs_if_exist:cTF { c__text_ #1 case_ \token_to_str:N #6 _tl } 3604
6046 { 3604
6047 \__text_change_case_store:v 3604
6048 { c__text_ #1 case_ \token_to_str:N #6 _tl } 3604
6049 \use:c { __text_change_case_next_ #2 :nnn } {#2} {#4} {#5} 3604
6050 } 3605
6051 { 3605
6052 \__text_change_case_store:n {#6} 3605
6053 \cs_if_exist:cTF 3605
6054 { 3605
6055 c__text_ 3605
6056 \str_if_eq:nnTF {#1} { lower } { upper } { lower } 3605
6057 case_ \token_to_str:N #6 _tl 3605
6058 } 3605
6059 { \use:c { __text_change_case_next_ #2 :nnn } {#2} {#4} {#5} } 3605
6060 { \__text_change_case_loop:nnnw {#3} {#4} {#5} } 3606
6061 } 3606
6062 } 3606
6063 \cs_new:Npn \__text_change_case_custom_lower:nnnn #1#2#3#4 3606
6064 { 3606
6065 \__text_change_case_custom:nnnnnn {#1} {#1} {#2} {#3} {#4} 3606
6066 { \use:c { __text_change_case_codepoint_ #1 :nnnn } {#1} {#2} {#3} {#4} } 3606
6067 } 3606
6068 \cs_new_eq:NN \__text_change_case_custom_upper:nnnn 3606
6069 \__text_change_case_custom_lower:nnnn 3606
6070 \cs_new:Npn \__text_change_case_custom_title:nnnn #1#2#3#4 3607
6071 { 3607
6072 \__text_change_case_custom:nnnnnn { title } {#1} {#2} {#3} {#4} 3607
6073 { 3607
6074 \__text_change_case_custom:nnnnnn { upper } {#1} {#2} {#3} {#4} 3607
6075 { \use:c { __text_change_case_codepoint_ #1 :nnnn } {#1} {#2} {#3} {#4} } 3607
6076 } 3607
6077 } 3607
6078 \cs_new:Npn \__text_change_case_custom:nnnnnn #1#2#3#4#5#6 3607
6079 { 3607
6080 \tl_if_exist:cTF { l__text_ #1 case _ \tl_to_str:n {#5} _ #4 _tl } 3608
6081 { 3608
6082 \__text_change_case_replace:vnnn 3608
```

```
6083 { l__text_ #1 case _ \tl_to_str:n {#5} _ #4 _tl } {#2} {#3} {#4} 36083
6084 } 36084
6085 { 36085
6086 \tl_if_exist:cTF { l__text_ #1 case _ \tl_to_str:n {#5} _tl } 36086
6087 { 36087
6088 \__text_change_case_replace:vn 36088
6089 { l__text_ #1 case _ \tl_to_str:n {#5} _tl } {#2} {#3} {#4} 36089
6090 } 36090
6091 {#6} 36091
6092 } 36092
6093 } 36093
6094 \cs_new:Npn \__text_change_case_codepoint_lower:nnnn #1#2#3#4 36094
6095 { 36095
6096 \cs_if_exist_use:cF { __text_change_case_lower_ #3 :nnnnn } 36096
6097 { \__text_change_case_lower_sigma:nnnnn } 36097
6098 {#1} {#1} {#2} {#3} {#4} 36098
6099 } 36099
6100 \cs_new:Npn \__text_change_case_codepoint_upper:nnnn #1#2#3#4 36100
6101 { 36101
6102 \cs_if_exist_use:cF { __text_change_case_upper_ #3 :nnnnn } 36102
6103 { \__text_change_case_codepoint:nnnnn } 36103
6104 {#1} {#1} {#2} {#3} {#4} 36104
6105 } 36105
6106 \cs_new:Npn \__text_change_case_lower_sigma:nnnnn #1#2#3#4#5 36106
6107 { 36107
6108 \__text_codepoint_compare:nNnTF {#5} = { "03A3 } 36108
6109 { \__text_change_case_lower_sigma:nnnnw {#2} } 36109
6110 { \__text_change_case_codepoint:nnnnn {#1} {#2} } 36110
6111 {#3} {#4} {#5} 36111
6112 } 36112
6113 \cs_new:Npn \__text_change_case_lower_sigma:nnnnw #1#2#3#4#5 \q_text_recursion_stop 36113
6114 { 36114
6115 \tl_if_head_is_N_type:nTF {#5} 36115
6116 { \__text_change_case_lower_sigma:nnnnN {#4} } 36116
6117 { 36117
6118 \__text_change_case_store:e 36118
6119 { \codepoint_generate:nn { "03C2 } { \__text_char_catcode:N #4 } } 36119
6120 \__text_change_case_loop:nnnw 36120
6121 } 36121
6122 {#1} {#2} {#3} #5 \q_text_recursion_stop 36122
6123 } 36123
6124 \cs_new:Npn \__text_change_case_lower_sigma:nnnnN #1#2#3#4#5 36124
6125 { 36125
6126 \__text_change_case_store:e 36126
6127 { 36127
6128 \bool_lazy_or:nnTF 36128
```

```
6129         { \token_if_letter_p:N #5 }
6130     {
6131         \bool_lazy_and_p:nn
6132             { \token_if_active_p:N #5 }
6133             { \int_compare_p:nNn {\`#5} > { "80 } }
6134     }
6135     { \codepoint_generate:nn { "03C3 } { \__text_char_catcode:N #1 } }
6136     { \codepoint_generate:nn { "03C2 } { \__text_char_catcode:N #1 } }
6137 }
6138 \__text_change_case_loop:nnnw {#2} {#3} {#4} #5
6139 }
6140 \cs_new:Npn \__text_change_case_codepoint_title:nnnn #1#2#3#4
6141 {
6142     \bool_if:NTF \l_text_titlecase_check_letter_bool
6143     {
6144         \exp_args:Ne \__text_change_case_codepoint_title_auxi:nnnn
6145         {
6146             \codepoint_to_category:n
6147                 { \__text_codepoint_from_chars:Nw #4 }
6148         }
6149     }
6150     { \__text_change_case_codepoint_title:nnn }
6151     {#2} {#3} {#4}
6152 }
6153 \cs_new:Npn \__text_change_case_codepoint_title_auxi:nnnn #1#2#3#4
6154 {
6155     \tl_if_head_eq_charcode:nNTF {#1} { L }
6156     { \__text_change_case_codepoint_title:nnn }
6157     { \__text_change_case_codepoint_title_auxii:nnnn { title } }
6158     {#2} {#3} {#4}
6159 }
6160 \cs_new:Npn \__text_change_case_codepoint_title:nnn #1#2#3
6161 { \__text_change_case_codepoint_title_auxii:nnnn { end } {#1} {#2} {#3} }
6162 \cs_new:Npn \__text_change_case_codepoint_title_auxii:nnnn #1#2#3#4
6163 {
6164     \cs_if_exist_use:cF { __text_change_case_title_ #3 :nnnnn }
6165     {
6166         \cs_if_exist_use:cF { __text_change_case_upper_ #3 :nnnnn }
6167         { \__text_change_case_codepoint:nnnn }
6168     }
6169     { title } {#1} {#2} {#3} {#4}
6170 }
6171 \cs_new:Npn \__text_change_case_codepoint:nnnnn #1#2#3#4#5
6172 {
6173     \bool_lazy_and:nnTF
6174         { \tl_if_single_p:n {#5} }
```

```

6175 { \token_if_active_p:N #5 }
6176 { \__text_change_case_store:n {#5} }
6177 {
6178   \__text_change_case_store:e
6179   { \__text_change_case_codepoint:nn {#1} {#5} }
6180 }
6181 \use:c { \__text_change_case_next_ #2 :nnn } {#2} {#3} {#4}
6182 }
6183 \cs_new:Npn \__text_change_case_codepoint:nn #1#2
6184 {
6185   \__text_change_case_codepoint:fnn
6186   { \int_eval:n { \__text_codepoint_from_chars:Nw #2 } } {#1} {#2}
6187 }
6188 \cs_new:Npn \__text_change_case_codepoint:nnn #1#2#3
6189 {
6190   \exp_args:Ne \__text_change_case_codepoint_aux:nn
6191   { \__kernel_codepoint_case:nn { #2 case } {#1} } {#3}
6192 }
6193 \cs_generate_variant:Nn \__text_change_case_codepoint:nnn { f }
6194 \sys_if_engine_ptex:T
6195 {
6196   \cs_new_eq:NN \__text_change_case_codepoint_aux:nnn
6197   \__text_change_case_codepoint:nnn
6198   \cs_gset:Npn \__text_change_case_codepoint:nnn #1#2#3
6199   {
6200     \int_compare:nNnTF {#1} = { -1 }
6201     { \exp_not:n {#3} }
6202     { \__text_change_case_codepoint_aux:nnn {#1} {#2} {#3} }
6203   }
6204 }
6205 \cs_new:Npn \__text_change_case_codepoint_aux:nn #1#2
6206 {
6207   \use:e { \__text_change_case_codepoint_aux:nnnn #1 {#2} }
6208 }
6209 \cs_new:Npn \__text_change_case_codepoint_aux:nnnn #1#2#3#4
6210 {
6211   \__text_codepoint_compare:nNnTF {#4} = {#1}
6212   { \exp_not:n {#4} }
6213   {
6214     \codepoint_generate:nn {#1}
6215     { \__text_change_case_catcode:nn {#4} {#1} }
6216     \tl_if_blank:nF {#2}
6217     {
6218       \codepoint_generate:nn {#2}
6219       { \char_value_catcode:n {#2} }
6220       \tl_if_blank:nF {#3}

```

```

6221         {
6222             \codepoint_generate:nn {#3}
6223             { \char_value_catcode:n {#3} }
6224         }
6225     }
6226 }
6227 }
6228 \sys_if_engine_opentype:TF
6229 {
6230     \cs_new:Npn \__text_change_case_catcode:nn #1#2
6231     { \__text_char_catcode:N #1 }
6232 }
6233 {
6234     \cs_new:Npn \__text_change_case_catcode:nn #1#2
6235     {
6236         \__text_codepoint_compare:nNnTF {#1} < { "80 }
6237         { \__text_char_catcode:N #1 }
6238         {
6239             \int_compare:nNnTF {#2} < { "80 }
6240             { \char_value_catcode:n {#2} }
6241             { 13 }
6242         }
6243     }
6244 }
6245 \cs_new:Npn \__text_change_case_next_lower:nnn #1#2#3
6246 { \__text_change_case_loop:nnnw {#1} {#2} {#3} }
6247 \cs_new_eq:NN \__text_change_case_next_upper:nnn
6248 \__text_change_case_next_lower:nnn
6249 \cs_new_eq:NN \__text_change_case_next_title:nnn
6250 \__text_change_case_next_lower:nnn
6251 \cs_new:Npn \__text_change_case_next_end:nnn #1#2#3
6252 { \__text_change_case_skip:nnw {#2} {#3} }
6253 \cs_new_protected:Npn \text_declare_case_equivalent:Nn #1#2
6254 {
6255     \tl_clear_new:c { l__text_case_ \token_to_str:N #1 _tl }
6256     \tl_set:cn { l__text_case_ \token_to_str:N #1 _tl } {#2}
6257 }
6258 \cs_new_protected:Npn \text_declare_lowercase_mapping:nn #1#2
6259 { \__text_declare_case_mapping:nnn { lower } {#1} {#2} }
6260 \cs_new_protected:Npn \text_declare_titlecase_mapping:nn #1#2
6261 { \__text_declare_case_mapping:nnn { title } {#1} {#2} }
6262 \cs_new_protected:Npn \text_declare_uppercase_mapping:nn #1#2
6263 { \__text_declare_case_mapping:nnn { upper } {#1} {#2} }
6264 \cs_new_protected:Npn \__text_declare_case_mapping:nnn #1#2#3
6265 {
6266     \exp_args:Ne \__text_declare_case_mapping_aux:nnn

```


6267	{ \codepoint_str_generate:n {#2} } {#1} {#3}	36267
6268	}	36268
6269	\cs_new_protected:Npn __text_declare_case_mapping_aux:nnn #1#2#3	36269
6270	{	36270
6271	\tl_clear_new:c { l__text_ #2 case _ #1 _tl }	36271
6272	\tl_set:cn { l__text_ #2 case _ #1 _tl } {#3}	36272
6273	}	36273
6274	\cs_new_protected:Npn \text_declare_lowercase_mapping:nnn #1#2#3	36274
6275	{ __text_declare_case_mapping:nnnn { lower } {#1} {#2} {#3} }	36275
6276	\cs_new_protected:Npn \text_declare_titlecase_mapping:nnn #1#2#3	36276
6277	{ __text_declare_case_mapping:nnnn { title } {#1} {#2} {#3} }	36277
6278	\cs_new_protected:Npn \text_declare_uppercase_mapping:nnn #1#2#3	36278
6279	{ __text_declare_case_mapping:nnnn { upper } {#1} {#2} {#3} }	36279
6280	\cs_new_protected:Npn __text_declare_case_mapping:nnnn #1#2#3#4	36280
6281	{	36281
6282	\exp_args:Ne __text_declare_case_mapping_aux:nnnn	36282
6283	{ \codepoint_str_generate:n {#3} } {#1} {#2} {#4}	36283
6284	}	36284
6285	\cs_new_protected:Npn __text_declare_case_mapping_aux:nnnn #1#2#3#4	36285
6286	{	36286
6287	\tl_clear_new:c { l__text_ #2 case _ #1 _ #3 _tl }	36287
6288	\tl_set:cn { l__text_ #2 case _ #1 _ #3 _tl } {#4}	36288
6289	\tl_clear_new:c { l__text_ #2 case_special_ #3 _tl }	36289
6290	}	36290
6291	\cs_new:Npn \text_case_switch:nnnn #1#2#3#4	36291
6292	{	36292
6293	__text_case_switch_marker:	36293
6294	#1	36294
6295	}	36295
6296	\cs_new:Npn __text_case_switch_marker: { }	36296
6297	\cs_new:Npn __text_change_case_generate:n #1	36297
6298	{ \codepoint_generate:nn {#1} { \char_value_catcode:n {#1} } }	36298
6299	\cs_new:cpn { __text_change_case_upper_de-x-eszett:nnnnn } #1#2#3#4#5	36299
6300	{	36300
6301	__text_codepoint_compare:nNnTF {#5} = { "00DF }	36301
6302	{	36302
6303	__text_change_case_store:e	36303
6304	{	36304
6305	\codepoint_generate:nn { "1E9E }	36305
6306	{ __text_change_case_catcode:nn {#5} { "1E9E } }	36306
6307	}	36307
6308	\use:c { __text_change_case_next_ #2 :nnn }	36308
6309	{#2} {#3} {#4}	36309
6310	}	36310
6311	{ __text_change_case_codepoint:nnnnn {#1} {#2} {#3} {#4} {#5} }	36311
6312	}	36312


```
6313 \cs_new_eq:cc { __text_change_case_upper_de-alt:nnnnn } 3631
6314 { __text_change_case_upper_de-x-eszett:nnnnn } 3631
6315 \cs_new:Npn \__text_change_case_upper_el:nnnnn #1#2#3#4#5 3631
6316 { 3631
6317 \bool_lazy_and:nnTF 3631
6318 { \__text_change_case_if_greek_p:n {#5} } 3631
6319 { 3631
6320 ! \bool_lazy_or_p:nn 3632
6321 { \__text_codepoint_compare_p:nNn {#5} = { "0374 } } 3632
6322 { \__text_codepoint_compare_p:nNn {#5} = { "037E } } 3632
6323 } 3632
6324 { 3632
6325 \__text_change_case_if_greek_spacing_diacritic:nTF {#5} 3632
6326 { 3632
6327 \__text_change_case_store:n {#5} 3632
6328 \__text_change_case_loop:nnnw 3632
6329 } 3632
6330 { 3633
6331 \exp_args:Ne \__text_change_case_upper_el:nnnn 3633
6332 { 3633
6333 \codepoint_to_nfd:n 3633
6334 { \__text_codepoint_from_chars:Nw #5 } 3633
6335 } 3633
6336 } 3633
6337 {#2} {#3} {#4} 3633
6338 } 3633
6339 { 3633
6340 \__text_codepoint_compare:nNnTF {#5} = { "0345 } 3634
6341 { 3634
6342 \__text_change_case_store:e 3634
6343 { 3634
6344 \codepoint_generate:nn { "0399 } 3634
6345 { \char_value_catcode:n { "0399 } } 3634
6346 } 3634
6347 \__text_change_case_loop:nnnw {#2} {#3} {#4} 3634
6348 } 3634
6349 { \__text_change_case_codepoint:nnnnn {#1} {#2} {#3} {#4} {#5} } 3634
6350 } 3635
6351 } 3635
6352 \cs_new_eq:cN { __text_change_case_upper_el-x-iota:nnnnn } 3635
6353 \__text_change_case_upper_el:nnnnn 3635
6354 \cs_new:Npn \__text_change_case_upper_el:nnnn #1#2#3#4 3635
6355 { 3635
6356 \__text_codepoint_process:nN 3635
6357 { \__text_change_case_upper_el:nnnnw {#2} {#3} {#4} } #1 3635
6358 } 3635
```

```
6359 \cs_new:Npn \__text_change_case_upper_el:nnnnw #1#2#3#4#5 \q__text_recursion_stop 3635
6360 { 3636
6361 \tl_if_head_is_N_type:nTF {#5} 3636
6362 { \__text_change_case_upper_el:nnnnN {#4} } 3636
6363 { 3636
6364 \__text_change_case_store:e 3636
6365 { \__text_change_case_codepoint:nn { upper } {#4} } 3636
6366 \__text_change_case_loop:nnnw 3636
6367 } 3636
6368 {#1} {#2} {#3} #5 \q__text_recursion_stop 3636
6369 } 3636
6370 \cs_new:Npn \__text_change_case_upper_el:nnnnN #1#2#3#4#5 3637
6371 { 3637
6372 \token_if_cs:NTF #5 3637
6373 { 3637
6374 \__text_change_case_store:e 3637
6375 { \__text_change_case_codepoint:nn { upper } {#1} } 3637
6376 \__text_change_case_loop:nnnw {#2} {#3} {#4} #5 3637
6377 } 3637
6378 { 3637
6379 \__text_change_case_if_takes_ypogegrammeni:nTF {#1} 3637
6380 { 3638
6381 \__text_change_case_upper_el_ypogegrammeni:nnnnnnw 3638
6382 {#1} {#2} {#3} {#4} { } { } #5 3638
6383 } 3638
6384 { \__text_change_case_upper_el_aux:nnnnN {#1} {#2} {#3} {#4} #5 } 3638
6385 } 3638
6386 } 3638
6387 \cs_new:Npn \__text_change_case_upper_el_ypogegrammeni:nnnnnnw 3638
6388 #1#2#3#4#5#6#7 \q__text_recursion_stop 3638
6389 { 3638
6390 \tl_if_head_is_N_type:nTF {#7} 3639
6391 { 3639
6392 \__text_change_case_upper_el_ypogegrammeni:nnnnnnN 3639
6393 {#1} {#2} {#3} {#4} {#5} {#6} 3639
6394 } 3639
6395 { \__text_change_case_upper_el_aux:nnnnN {#1} {#2} {#3} {#4} #5#6 } 3639
6396 #7 \q__text_recursion_stop 3639
6397 } 3639
6398 \cs_new:Npn \__text_change_case_upper_el_ypogegrammeni:nnnnnnN #1#2#3#4#5#6#7 3639
6399 { 3639
6400 \token_if_cs:NTF #7 3640
6401 { \__text_change_case_upper_el_aux:nnnnN {#1} {#2} {#3} {#4} #5#6 } 3640
6402 { 3640
6403 \__text_codepoint_process:nN 3640
6404 { 3640
```

```
6405         \__text_change_case_upper_el_ypogegrammeni:nnnnnnn 36405
6406         {#1} {#2} {#3} {#4} {#5} {#6} 36406
6407     } 36407
6408 } 36408
6409 #7 36409
6410 } 36410
6411 \cs_new:Npn \__text_change_case_upper_el_ypogegrammeni:nnnnnnn #1#2#3#4#5#6#7 36411
6412 { 36412
6413     \__text_codepoint_compare:nNnTF {#7} = { "0345 } 36413
6414     { 36414
6415         \__text_change_case_upper_el_ypogegrammeni:nnnnnnnw 36415
6416         {#1} {#2} {#3} {#4} {#5} {#7} 36416
6417     } 36417
6418     { 36418
6419         \bool_lazy_or:nnTF 36419
6420         { \__text_change_case_if_greek_accent_p:n {#7} } 36420
6421         { \__text_change_case_if_greek_breathing_p:n {#7} } 36421
6422         { 36422
6423             \__text_change_case_upper_el_ypogegrammeni:nnnnnnw 36423
6424             {#1} {#2} {#3} {#4} {#5#7} {#6} 36424
6425         } 36425
6426         { \__text_change_case_upper_el_aux:nnnnN {#1} {#2} {#3} {#4} #5#6 #7 } 36426
6427     } 36427
6428 } 36428
6429 \cs_new:Npn \__text_change_case_upper_el_aux:nnnnN #1#2#3#4#5 36429
6430 { 36430
6431     \__text_codepoint_process:nN 36431
6432     { \__text_change_case_upper_el_aux:nnnnn {#1} {#2} {#3} {#4} } #5 36432
6433 } 36433
6434 \cs_new:Npn \__text_change_case_upper_el_aux:nnnnn #1#2#3#4#5 36434
6435 { 36435
6436     \__text_codepoint_compare:nNnTF {#5} = { "0308 } 36436
6437     { \__text_change_case_upper_el_dialytika:nnnn {#2} {#3} {#4} {#1} } 36437
6438     { 36438
6439         \__text_change_case_if_greek_accent:nTF {#5} 36439
6440         { \__text_change_case_upper_el_hiatus:nnnnw {#2} {#3} {#4} {#1} } 36440
6441         { 36441
6442             \__text_change_case_if_greek_breathing:nTF {#5} 36442
6443             { \__text_change_case_upper_el:nnnn {#1} {#2} {#3} {#4} } 36443
6444             { 36444
6445                 \__text_codepoint_compare:nNnTF {#5} = { "0345 } 36445
6446                 { 36446
6447                     \__text_change_case_store:e 36447
6448                     { \use:c { \__text_change_case_upper_ #4 _ypogegrammeni:n } {#1} } 36448
6449                     \__text_change_case_loop:nnnw {#2} {#3} {#4} 36449
6450                 } 36450
```

```
6451         {
6452             \__text_change_case_if_greek_stress:nTF {#5}
6453             {
6454                 \__text_change_case_store:e
6455                 { \__text_change_case_upper_el_stress:nn {#1} {#5} }
6456                 \__text_change_case_loop:nnnw {#2} {#3} {#4}
6457             }
6458             {
6459                 \__text_change_case_store:e
6460                 { \__text_change_case_codepoint:nn { upper } {#1} }
6461                 \__text_change_case_loop:nnnw {#2} {#3} {#4} #5
6462             }
6463         }
6464     }
6465 }
6466 }
6467 }
6468 \cs_new:Npn \__text_change_case_upper_el_dialytika:nnnn #1#2#3#4
6469 {
6470     \__text_change_case_if_takes_dialytika:nTF {#4}
6471     { \__text_change_case_upper_el_dialytika:n {#4} }
6472     {
6473         \__text_change_case_store:e
6474         { \__text_change_case_codepoint:nn { upper } {#4} }
6475     }
6476     \__text_change_case_upper_el_gobble:nnnw {#1} {#2} {#3}
6477 }
6478 \cs_new:Npn \__text_change_case_upper_el_dialytika:n #1
6479 {
6480     \__text_change_case_store:e
6481     {
6482         \bool_lazy_or:nnTF
6483         { \__text_codepoint_compare_p:nNn {#1} = { "0399 } }
6484         { \__text_codepoint_compare_p:nNn {#1} = { "03B9 } }
6485         {
6486             \codepoint_generate:nn { "03AA }
6487             { \__text_change_case_catcode:nn {#1} { "03AA } }
6488         }
6489         {
6490             \codepoint_generate:nn { "03AB }
6491             { \__text_change_case_catcode:nn {#1} { "03AB } }
6492         }
6493     }
6494 }
6495 \cs_new:Npn \__text_change_case_upper_el_hiatus:nnnnw
6496 #1#2#3#4#5 \q__text_recursion_stop
```

```
6497 {
6498     \tl_if_head_is_N_type:nTF {#5}
6499     { \__text_change_case_upper_el_hiatus:nnnnN {#4} }
6500     {
6501         \__text_change_case_store:e
6502         { \__text_change_case_codepoint:nn { upper } {#4} }
6503         \__text_change_case_loop:nnnw
6504     }
6505     {#1} {#2} {#3} #5 \q__text_recursion_stop
6506 }
6507 \cs_new:Npn \__text_change_case_upper_el_hiatus:nnnnN #1#2#3#4#5
6508 {
6509     \token_if_cs:NTF #5
6510     {
6511         \__text_change_case_store:e
6512         { \__text_change_case_codepoint:nn { upper } {#1} }
6513         \__text_change_case_loop:nnnw {#2} {#3} {#4} #5
6514     }
6515     {
6516         \__text_codepoint_process:nN
6517         { \__text_change_case_upper_el_hiatus:nnnnn {#1} {#2} {#3} {#4} } #5
6518     }
6519 }
6520 \cs_new:Npn \__text_change_case_upper_el_hiatus:nnnnn #1#2#3#4#5
6521 {
6522     \__text_change_case_if_takes_dialytika:nTF {#5}
6523     {
6524         \__text_change_case_store:e
6525         { \__text_change_case_codepoint:nn { upper } {#1} }
6526         \__text_change_case_upper_el_dialytika:n {#5}
6527         \__text_change_case_upper_el_gobble:nnnw {#2} {#3} {#4}
6528     }
6529     { \__text_change_case_upper_el:nnnn {#1} {#2} {#3} {#4} #5 }
6530 }
6531 \cs_new:Npn \__text_change_case_upper_el_ypogegrammeni:n #1
6532 {
6533     \exp_args:Ne \__text_change_case_generate:n
6534     {
6535         \int_case:nn
6536         { \__text_codepoint_from_chars:Nw #1 }
6537         {
6538             { "0391 } { "1FBC }
6539             { "03B1 } { "1FBC }
6540             { "0397 } { "1FCC }
6541             { "03B7 } { "1FCC }
6542             { "03A9 } { "1FFC }
```

6543	{ "03C9 } { "1FFC }	3654
6544	}	3654
6545	}	3654
6546	}	3654
6547	\cs_new:cpn { __text_change_case_upper_el-x-iota_ypogegrammeni:n } #1	3654
6548	{	3654
6549	__text_change_case_codepoint:nn { upper } {#1}	3654
6550	\codepoint_generate:nn { "0399 }	3655
6551	{ \char_value_catcode:n { "0399 } }	3655
6552	}	3655
6553	\cs_new:Npn __text_change_case_upper_el_stress:nn #1#2	3655
6554	{	3655
6555	\exp_args:Ne __text_change_case_generate:n	3655
6556	{	3655
6557	\int_case:nn	3655
6558	{ __text_codepoint_from_chars:Nw #2 }	3655
6559	{	3655
6560	{ "0304 }	3656
6561	{	3656
6562	\int_case:nn { __text_codepoint_from_chars:Nw #1 }	3656
6563	{	3656
6564	{ "0391 } { "1FB9 }	3656
6565	{ "03B1 } { "1FB9 }	3656
6566	{ "0399 } { "1FD9 }	3656
6567	{ "03B9 } { "1FD9 }	3656
6568	{ "03A5 } { "1FE9 }	3656
6569	{ "03C5 } { "1FE9 }	3656
6570	}	3657
6571	}	3657
6572	{ "0306 }	3657
6573	{	3657
6574	\int_case:nn { __text_codepoint_from_chars:Nw #1 }	3657
6575	{	3657
6576	{ "0391 } { "1FB8 }	3657
6577	{ "03B1 } { "1FB8 }	3657
6578	{ "0399 } { "1FD8 }	3657
6579	{ "03B9 } { "1FD8 }	3657
6580	{ "03A5 } { "1FE8 }	3658
6581	{ "03C5 } { "1FE8 }	3658
6582	}	3658
6583	}	3658
6584	}	3658
6585	}	3658
6586	}	3658
6587	\cs_new:Npn __text_change_case_upper_el_gobble:nnnw	3658
6588	#1#2#3#4 \q__text_recursion_stop	3658

```
6589 {
6590     \tl_if_head_is_N_type:nTF {#4}
6591     { \__text_change_case_upper_el_gobble:nnnN }
6592     { \__text_change_case_loop:nnnw }
6593     {#1} {#2} {#3} #4 \q__text_recursion_stop
6594 }
6595 \cs_new:Npn \__text_change_case_upper_el_gobble:nnnN #1#2#3#4
6596 {
6597     \token_if_cs:NTF #4
6598     { \__text_change_case_loop:nnnw {#1} {#2} {#3} }
6599     {
6600         \__text_codepoint_process:nN
6601         { \__text_change_case_upper_el_gobble:nnnn {#1} {#2} {#3} }
6602     }
6603     #4
6604 }
6605 \cs_new:Npn \__text_change_case_upper_el_gobble:nnnn #1#2#3#4
6606 {
6607     \bool_lazy_or:nnTF
6608     { \__text_change_case_if_greek_accent_p:n {#4} }
6609     { \__text_change_case_if_greek_breathing_p:n {#4} }
6610     { \__text_change_case_upper_el_gobble:nnnw {#1} {#2} {#3} }
6611     { \__text_change_case_loop:nnnw {#1} {#2} {#3} #4 }
6612 }
6613 \prg_new_conditional:Npnn \__text_change_case_if_greek:n #1 { p , TF }
6614 {
6615     \exp_args:Nf \__text_change_case_if_greek:n
6616     { \int_eval:n { \__text_codepoint_from_chars:Nw #1 } }
6617 }
6618 \cs_new:Npn \__text_change_case_if_greek:n #1
6619 {
6620     \if_int_compare:w #1 < "0370 \exp_stop_f:
6621     \prg_return_false:
6622 \else:
6623     \if_int_compare:w #1 > "03FF \exp_stop_f:
6624     \if_int_compare:w #1 < "1F00 \exp_stop_f:
6625     \prg_return_false:
6626 \else:
6627     \if_int_compare:w #1 > "1FFF \exp_stop_f:
6628     \if_int_compare:w #1 = "2126 \exp_stop_f:
6629     \prg_return_true:
6630 \else:
6631     \prg_return_false:
6632 \fi:
6633 \else:
6634     \prg_return_true:
```


6635	\fi:	36635
6636	\fi:	36636
6637	\else:	36637
6638	\prg_return_true:	36638
6639	\fi:	36639
6640	\fi:	36640
6641	}	36641
6642	\prg_new_conditional:Npnn __text_change_case_if_greek_accent:n #1 { TF , p }	36642
6643	{	36643
6644	\exp_args:Nf __text_change_case_if_greek_accent:n	36644
6645	{ \int_eval:n { __text_codepoint_from_chars:Nw #1 } }	36645
6646	}	36646
6647	\cs_new:Npn __text_change_case_if_greek_accent:n #1	36647
6648	{	36648
6649	\if_int_compare:w #1 = "0300 \exp_stop_f:	36649
6650	\prg_return_true:	36650
6651	\else:	36651
6652	\if_int_compare:w #1 = "0301 \exp_stop_f:	36652
6653	\prg_return_true:	36653
6654	\else:	36654
6655	\if_int_compare:w #1 = "0342 \exp_stop_f:	36655
6656	\prg_return_true:	36656
6657	\else:	36657
6658	\if_int_compare:w #1 = "0302 \exp_stop_f:	36658
6659	\prg_return_true:	36659
6660	\else:	36660
6661	\if_int_compare:w #1 = "0303 \exp_stop_f:	36661
6662	\prg_return_true:	36662
6663	\else:	36663
6664	\if_int_compare:w #1 = "0311 \exp_stop_f:	36664
6665	\prg_return_true:	36665
6666	\else:	36666
6667	\prg_return_false:	36667
6668	\fi:	36668
6669	\fi:	36669
6670	\fi:	36670
6671	\fi:	36671
6672	\fi:	36672
6673	\fi:	36673
6674	}	36674
6675	\prg_new_conditional:Npnn __text_change_case_if_greek_spacing_diacritic:n	36675
6676	#1 { TF }	36676
6677	{	36677
6678	\exp_args:Nf __text_change_case_if_greek_spacing_diacritic:n	36678
6679	{ \int_eval:n { __text_codepoint_from_chars:Nw #1 } }	36679
6680	}	36680

6681	\cs_new:Npn __text_change_case_if_greek_spacing_diacritic:n #1	3668
6682	{	3668
6683	\if_int_compare:w #1 < "1FBD \exp_stop_f:	3668
6684	\if_int_compare:w #1 = "037A \exp_stop_f:	3668
6685	\prg_return_true:	3668
6686	\else:	3668
6687	\prg_return_false:	3668
6688	\fi:	3668
6689	\else:	3668
6690	\if_int_compare:w #1 = "1FBD \exp_stop_f:	3669
6691	\prg_return_true:	3669
6692	\else:	3669
6693	\if_int_compare:w #1 = "1FBF \exp_stop_f:	3669
6694	\prg_return_true:	3669
6695	\else:	3669
6696	\if_int_compare:w #1 = "1FC0 \exp_stop_f:	3669
6697	\prg_return_true:	3669
6698	\else:	3669
6699	\if_int_compare:w #1 = "1FC1 \exp_stop_f:	3669
6700	\prg_return_true:	3670
6701	\else:	3670
6702	\if_int_compare:w #1 = "1FCD \exp_stop_f:	3670
6703	\prg_return_true:	3670
6704	\else:	3670
6705	\if_int_compare:w #1 = "1FCE \exp_stop_f:	3670
6706	\prg_return_true:	3670
6707	\else:	3670
6708	\if_int_compare:w #1 = "1FCF \exp_stop_f:	3670
6709	\prg_return_true:	3670
6710	\else:	3671
6711	\if_int_compare:w #1 = "1FDD \exp_stop_f:	3671
6712	\prg_return_true:	3671
6713	\else:	3671
6714	\if_int_compare:w #1 = "1FDE \exp_stop_f:	3671
6715	\prg_return_true:	3671
6716	\else:	3671
6717	\if_int_compare:w #1 = "1FDF \exp_stop_f:	3671
6718	\prg_return_true:	3671
6719	\else:	3671
6720	\if_int_compare:w #1 = "1FED \exp_stop_f:	3672
6721	\prg_return_true:	3672
6722	\else:	3672
6723	\if_int_compare:w #1 = "1FEE \exp_stop_f:	3672
6724	\prg_return_true:	3672
6725	\else:	3672
6726	\if_int_compare:w #1 = "1FEF \exp_stop_f:	3672

```
6727         \prg_return_true: 3672
6728     \else: 3672
6729         \if_int_compare:w #1 = "1FFD \exp_stop_f: 3672
6730         \prg_return_true: 3673
6731     \else: 3673
6732         \if_int_compare:w #1 = "1FFE \exp_stop_f: 3673
6733         \prg_return_true: 3673
6734     \else: 3673
6735         \prg_return_false: 3673
6736     \fi: 3673
6737 \fi: 3673
6738 \fi: 3673
6739 \fi: 3673
6740 \fi: 3674
6741 \fi: 3674
6742 \fi: 3674
6743 \fi: 3674
6744 \fi: 3674
6745 \fi: 3674
6746 \fi: 3674
6747 \fi: 3674
6748 \fi: 3674
6749 \fi: 3674
6750 \fi: 3675
6751 \fi: 3675
6752 } 3675
6753 \prg_new_conditional:Npnn \__text_change_case_if_greek_breathing:n 3675
6754 #1 { TF , p } 3675
6755 { 3675
6756     \exp_args:Nf \__text_change_case_if_greek_breathing:n 3675
6757     { \int_eval:n { \__text_codepoint_from_chars:Nw #1 } } 3675
6758 } 3675
6759 \cs_new:Npn \__text_change_case_if_greek_breathing:n #1 3675
6760 { 3676
6761     \if_int_compare:w #1 = "0313 \exp_stop_f: 3676
6762     \prg_return_true: 3676
6763 \else: 3676
6764     \if_int_compare:w #1 = "0314 \exp_stop_f: 3676
6765     \prg_return_true: 3676
6766     \else: 3676
6767         \prg_return_false: 3676
6768     \fi: 3676
6769 \fi: 3676
6770 } 3677
6771 \prg_new_conditional:Npnn \__text_change_case_if_greek_stress:n 3677
6772 #1 { TF , p } 3677
```

6773	{	36773
6774	\exp_args:Nf _\text_change_case_if_greek_stress:n	36774
6775	{ \int_eval:n { _\text_codepoint_from_chars:Nw #1 } }	36775
6776	}	36776
6777	\cs_new:Npn _\text_change_case_if_greek_stress:n #1	36777
6778	{	36778
6779	\if_int_compare:w #1 = "0304 \exp_stop_f:	36779
6780	\prg_return_true:	36780
6781	\else:	36781
6782	\if_int_compare:w #1 = "0306 \exp_stop_f:	36782
6783	\prg_return_true:	36783
6784	\else:	36784
6785	\prg_return_false:	36785
6786	\fi:	36786
6787	\fi:	36787
6788	}	36788
6789	\prg_new_conditional:Npnn _\text_change_case_if_takes_dialytika:n #1 { TF }	36789
6790	{	36790
6791	\exp_args:Nf _\text_change_case_if_takes_dialytika:n	36791
6792	{ \int_eval:n { _\text_codepoint_from_chars:Nw #1 } }	36792
6793	}	36793
6794	\cs_new:Npn _\text_change_case_if_takes_dialytika:n #1	36794
6795	{	36795
6796	\if_int_compare:w #1 = "0399 \exp_stop_f:	36796
6797	\prg_return_true:	36797
6798	\else:	36798
6799	\if_int_compare:w #1 = "03B9 \exp_stop_f:	36799
6800	\prg_return_true:	36800
6801	\else:	36801
6802	\if_int_compare:w #1 = "03A5 \exp_stop_f:	36802
6803	\prg_return_true:	36803
6804	\else:	36804
6805	\if_int_compare:w #1 = "03C5 \exp_stop_f:	36805
6806	\prg_return_true:	36806
6807	\else:	36807
6808	\prg_return_false:	36808
6809	\fi:	36809
6810	\fi:	36810
6811	\fi:	36811
6812	\fi:	36812
6813	}	36813
6814	\prg_new_conditional:Npnn _\text_change_case_if_takes_ypogegrammeni:n #1 { TF }	36814
6815	{	36815
6816	\exp_args:Nf _\text_change_case_if_takes_ypogegrammeni:n	36816
6817	{ \int_eval:n { _\text_codepoint_from_chars:Nw #1 } }	36817
6818	}	36818

```
6819 \cs_new:Npn \__text_change_case_if_takes_ypogegrammeni:n #1
6820 {
6821     \if_int_compare:w #1 = "03B1 \exp_stop_f:
6822     \prg_return_true:
6823 \else:
6824     \if_int_compare:w #1 = "03B7 \exp_stop_f:
6825     \prg_return_true:
6826 \else:
6827     \if_int_compare:w #1 = "03C9 \exp_stop_f:
6828     \prg_return_true:
6829 \else:
6830     \prg_return_false:
6831 \fi:
6832 \fi:
6833 \fi:
6834 }
6835 \cs_new:Npn \__text_change_case_boundary_upper_el:Nnnnw
6836 #1#2#3#4#5 \q__text_recursion_stop
6837 {
6838     \tl_if_head_is_N_type:nTF {#5}
6839     { \__text_change_case_boundary_upper_el:nnnN }
6840     { \__text_change_case_loop:nnnw }
6841     {#2} {#3} {#4} #5 \q__text_recursion_stop
6842 }
6843 \cs_new_eq:cN { __text_change_case_boundary_upper_el-x-iota:Nnnnw }
6844 \__text_change_case_boundary_upper_el:Nnnnw
6845 \cs_new:Npn \__text_change_case_boundary_upper_el:nnnN #1#2#3#4
6846 {
6847     \token_if_cs:NTF #4
6848     { \__text_change_case_loop:nnnw {#1} {#2} {#3} }
6849     {
6850         \__text_codepoint_process:nN
6851         { \__text_change_case_boundary_upper_el:nnnn {#1} {#2} {#3} }
6852     }
6853     #4
6854 }
6855 \cs_new:Npn \__text_change_case_boundary_upper_el:nnnn #1#2#3#4
6856 {
6857     \bool_lazy_any:nTF
6858     {
6859         { \__text_codepoint_compare_p:nNn {#4} = { "0389 } }
6860         { \__text_codepoint_compare_p:nNn {#4} = { "03AE } }
6861         { \__text_codepoint_compare_p:nNn {#4} = { "1F22 } }
6862         { \__text_codepoint_compare_p:nNn {#4} = { "1F2A } }
6863     }
6864     { \__text_change_case_boundary_upper_el:nnnnw {#1} {#2} {#3} {#4} }
```

```
6865 { \_text_change_case_breathing:nnnn {#1} {#2} {#3} {#4} } 36865
6866 } 36866
6867 \cs_new:Npn \_text_change_case_boundary_upper_el:nnnnw 36867
6868 #1#2#3#4#5 \q__text_recursion_stop 36868
6869 { 36869
6870 \tl_if_head_is_N_type:nTF {#5} 36870
6871 { \_text_change_case_loop:nnnw {#1} {#2} {#3} #4 } 36871
6872 { 36872
6873 \_text_change_case_store:e 36873
6874 { 36874
6875 \codepoint_generate:nn { "0389 } 36875
6876 { \_text_change_case_catcode:nn {#4} { "0389 } } 36876
6877 } 36877
6878 \_text_change_case_loop:nnnw {#1} {#2} {#3} 36878
6879 } 36879
6880 #5 \q__text_recursion_stop 36880
6881 } 36881
6882 \cs_new:Npn \_text_change_case_breathing:nnnn #1#2#3#4 36882
6883 { 36883
6884 \_text_change_case_if_greek:nTF {#4} 36884
6885 { 36885
6886 \exp_args:Ne \_text_change_case_breathing:nnnnn 36886
6887 { 36887
6888 \codepoint_to_nfd:n 36888
6889 { \_text_codepoint_from_chars:Nw #4 } 36889
6890 } 36890
6891 {#1} {#2} {#3} {#4} 36891
6892 } 36892
6893 { \_text_change_case_loop:nnnw {#1} {#2} {#3} #4 } 36893
6894 } 36894
6895 \cs_new:Npn \_text_change_case_breathing:nnnnn #1#2#3#4#5 36895
6896 { 36896
6897 \_text_codepoint_process:nN 36897
6898 { \_text_change_case_breathing:nnnnnw {#2} {#3} {#4} {#5} } 36898
6899 #1 \q__mark 36899
6900 } 36900
6901 \cs_new:Npn \_text_change_case_breathing:nnnnnw #1#2#3#4#5#6 \q__mark 36901
6902 { 36902
6903 \tl_if_blank:nTF {#6} 36903
6904 { \_text_change_case_loop:nnnw {#1} {#2} {#3} #4 } 36904
6905 { 36905
6906 \_text_codepoint_process:nN 36906
6907 { \_text_change_case_breathing:nnnnnnw {#1} {#2} {#3} {#4} {#5} } 36907
6908 #6 \q__mark 36908
6909 } 36909
6910 } 36910
```

```
6911 \cs_new:Npn \__text_change_case_breathing:nnnnnnw #1#2#3#4#5#6#7 \q_mark 3691
6912 { 3691
6913   \tl_if_blank:nTF {#7} 3691
6914   { 3691
6915     \__text_change_case_breathing_aux:nnnnnn 3691
6916     {#1} {#2} {#3} {#4} {#5} {#6} 3691
6917   } 3691
6918   { 3691
6919     \__text_codepoint_process:nN 3691
6920     { \__text_change_case_breathing:nnnnnnw {#1} {#2} {#3} {#4} {#5} } 3692
6921     #7 \q_mark 3692
6922   } 3692
6923 } 3692
6924 \cs_new:Npn \__text_change_case_breathing_aux:nnnnnn #1#2#3#4#5#6 3692
6925 { 3692
6926   \bool_lazy_or:nnTF 3692
6927   { \__text_codepoint_compare_p:nNn {#6} = { "0313 } } 3692
6928   { \__text_codepoint_compare_p:nNn {#6} = { "0314 } } 3692
6929   { \__text_change_case_breathing_aux:nnnnw {#1} {#2} {#3} {#5} } 3692
6930   { \__text_change_case_loop:nnnw {#1} {#2} {#3} #4 } 3693
6931 } 3693
6932 \cs_new:Npn \__text_change_case_breathing_aux:nnnnw #1#2#3#4#5 3693
6933 \q_text_recursion_stop 3693
6934 { 3693
6935   \__text_change_case_store:e 3693
6936   { \__text_change_case_codepoint:nn { upper } {#4} } 3693
6937   \tl_if_head_is_N_type:nTF {#5} 3693
6938   { \__text_change_case_breathing_aux:nnnN } 3693
6939   { \__text_change_case_loop:nnnw } 3693
6940   {#1} {#2} {#3} #5 \q_text_recursion_stop 3694
6941 } 3694
6942 \cs_new:Npn \__text_change_case_breathing_aux:nnnN #1#2#3#4 3694
6943 { 3694
6944   \__text_codepoint_process:nN 3694
6945   { \__text_change_case_breathing_dialytika:nnnn {#1} {#2} {#3} } #4 3694
6946 } 3694
6947 \cs_new:Npn \__text_change_case_breathing_dialytika:nnnn #1#2#3#4 3694
6948 { 3694
6949   \__text_change_case_if_takes_dialytika:nTF {#4} 3694
6950   { 3695
6951     \__text_change_case_upper_el_dialytika:n {#4} 3695
6952     \__text_change_case_loop:nnnw {#1} {#2} {#3} 3695
6953   } 3695
6954   { \__text_change_case_loop:nnnw {#1} {#2} {#3} #4 } 3695
6955 } 3695
6956 \cs_new:Npn \__text_change_case_title_el:nnnnn #1#2#3#4#5 3695
```



```
6957 { \__text_change_case_codepoint:nnnnn {#1} {#2} {#3} {#4} {#5} } 3695
6958 \cs_new:Npn \__text_change_case_upper_hy:nnnnn #1#2#3#4#5 3695
6959 { 3695
6960 \__text_codepoint_compare:nNnTF {#5} = { "0587 } 3696
6961 { 3696
6962 \__text_change_case_store:e 3696
6963 { 3696
6964 \codepoint_generate:nn { "0535 } 3696
6965 { \__text_change_case_catcode:nn {#5} { "0535 } } 3696
6966 \codepoint_generate:nn { "054E } 3696
6967 { \__text_change_case_catcode:nn {#5} { "054E } } 3696
6968 } 3696
6969 \use:c { __text_change_case_next_ #2 :nnn } 3696
6970 {#2} {#3} {#4} 3697
6971 } 3697
6972 { \__text_change_case_codepoint:nnnnn {#1} {#2} {#3} {#4} {#5} } 3697
6973 } 3697
6974 \cs_new:Npn \__text_change_case_title_hy:nnnnn #1#2#3#4#5 3697
6975 { 3697
6976 \__text_codepoint_compare:nNnTF {#5} = { "0587 } 3697
6977 { 3697
6978 \__text_change_case_store:e 3697
6979 { 3697
6980 \codepoint_generate:nn { "0535 } 3698
6981 { \__text_change_case_catcode:nn {#5} { "0535 } } 3698
6982 \codepoint_generate:nn { "057E } 3698
6983 { \__text_change_case_catcode:nn {#5} { "057E } } 3698
6984 } 3698
6985 \use:c { __text_change_case_next_ #2 :nnn } 3698
6986 {#2} {#3} {#4} 3698
6987 } 3698
6988 { \__text_change_case_codepoint:nnnnn {#1} {#2} {#3} {#4} {#5} } 3698
6989 } 3698
6990 \cs_new:cpn { __text_change_case_upper_hy-x-yiwn:nnnnn } #1#2#3#4#5 3699
6991 { \__text_change_case_codepoint:nnnnn {#1} {#2} {#3} {#4} {#5} } 3699
6992 \cs_new_eq:cc { __text_change_case_title_hy-x-yiwn:nnnnn } 3699
6993 { __text_change_case_upper_hy-x-yiwn:nnnnn } 3699
6994 \cs_new:cpn { __text_change_case_lower_la-x-medieval:nnnnn } #1#2#3#4#5 3699
6995 { 3699
6996 \__text_codepoint_compare:nNnTF {#5} = { "0056 } 3699
6997 { 3699
6998 \__text_change_case_store:e 3699
6999 { \char_generate:nn { "0075 } { \__text_char_catcode:N #5 } } 3699
7000 \use:c { __text_change_case_next_ #2 :nnn } 3700
7001 {#2} {#3} {#4} 3700
7002 } 3700
```

```
7003 { \__text_change_case_codepoint:nnnnn {#1} {#2} {#3} {#4} {#5} }
7004 }
7005 \cs_new:cpn { \__text_change_case_upper_la-x-medieval:nnnnn } #1#2#3#4#5
7006 {
7007   \__text_codepoint_compare:nNnTF {#5} = { "0075 }
7008   {
7009     \__text_change_case_store:e
7010     { \char_generate:nn { "0056 } { \__text_char_catcode:N #5 } }
7011     \use:c { \__text_change_case_next_ #2 :nnn }
7012     {#2} {#3} {#4}
7013   }
7014   { \__text_change_case_codepoint:nnnnn {#1} {#2} {#3} {#4} {#5} }
7015 }
7016 \cs_new:Npn \__text_change_case_lower_lt:nnnnn #1#2#3#4#5
7017 {
7018   \exp_args:Ne \__text_change_case_lower_lt_auxi:nnnnn
7019   {
7020     \int_case:nn { \__text_codepoint_from_chars:Nw #5 }
7021     {
7022       { "00CC } { "0300 }
7023       { "00CD } { "0301 }
7024       { "0128 } { "0303 }
7025     }
7026   }
7027   {#2} {#3} {#4} {#5}
7028 }
7029 \cs_new:Npn \__text_change_case_lower_lt_auxi:nnnnn #1#2#3#4#5
7030 {
7031   \tl_if_blank:nTF {#1}
7032   {
7033     \exp_args:Ne \__text_change_case_lower_lt_auxii:nnnnn
7034     {
7035       \int_case:nn { \__text_codepoint_from_chars:Nw #5 }
7036       {
7037         { "0049 } { "0069 }
7038         { "004A } { "006A }
7039         { "012E } { "012F }
7040       }
7041     }
7042     {#2} {#3} {#4} {#5}
7043   }
7044   {
7045     \__text_change_case_store:e
7046     {
7047       \codepoint_generate:nn { "0069 }
7048       { \__text_change_case_catcode:nn {#5} { "0069 } }
```

```
7049         \codepoint_generate:nn { "0307 }
7050         { \__text_change_case_catcode:nn {#5} { "0307 } }
7051     \codepoint_generate:nn {#1}
7052     { \__text_change_case_catcode:nn {#5} {#1} }
7053 }
7054 \__text_change_case_loop:nnnw {#2} {#3} {#4}
7055 }
7056 }
7057 \cs_new:Npn \__text_change_case_lower_lt_auxii:nnnnn #1#2#3#4#5
7058 {
7059     \tl_if_blank:nTF {#1}
7060     { \__text_change_case_codepoint:nnnnn {#2} {#2} {#3} {#4} {#5} }
7061     {
7062         \__text_change_case_store:e
7063         {
7064             \codepoint_generate:nn {#1}
7065             { \__text_change_case_catcode:nn {#5} {#1} }
7066         }
7067         \__text_change_case_lower_lt:nnnw {#2} {#3} {#4}
7068     }
7069 }
7070 \cs_new:Npn \__text_change_case_lower_lt:nnnw #1#2#3#4 \q__text_recursion_stop
7071 {
7072     \tl_if_head_is_N_type:nTF {#4}
7073     { \__text_change_case_lower_lt:nnnN }
7074     { \__text_change_case_loop:nnnw }
7075     {#1} {#2} {#3} #4 \q__text_recursion_stop
7076 }
7077 \cs_new:Npn \__text_change_case_lower_lt:nnnN #1#2#3#4
7078 {
7079     \__text_codepoint_process:nN
7080     { \__text_change_case_lower_lt:nnnn {#1} {#2} {#3} } #4
7081 }
7082 \cs_new:Npn \__text_change_case_lower_lt:nnnn #1#2#3#4
7083 {
7084     \bool_lazy_and:nnT
7085     {
7086         \bool_lazy_or_p:nn
7087         { ! \tl_if_single_p:n {#4} }
7088         { ! \token_if_cs_p:N #4 }
7089     }
7090     {
7091         \bool_lazy_any_p:n
7092         {
7093             { \__text_codepoint_compare_p:nNn {#4} = { "0300 } }
7094             { \__text_codepoint_compare_p:nNn {#4} = { "0301 } }
```

```
7095 { \__text_codepoint_compare_p:nNn {#4} = { "0303 } } 37095
7096 } 37096
7097 } 37097
7098 { 37098
7099 \__text_change_case_store:e 37099
7100 { 37100
7101 \codepoint_generate:nn { "0307 } 37101
7102 { \__text_change_case_catcode:nn {#4} { "0307 } } 37102
7103 } 37103
7104 } 37104
7105 \__text_change_case_loop:nnnw {#1} {#2} {#3} #4 37105
7106 } 37106
7107 \cs_new:Npn \__text_change_case_upper_lt:nnnnn #1#2#3#4#5 37107
7108 { 37108
7109 \exp_args:Ne \__text_change_case_upper_lt_aux:nnnnn 37109
7110 { 37110
7111 \int_case:nn { \__text_codepoint_from_chars:Nw #5 } 37111
7112 { 37112
7113 { "0069 } { "0049 } 37113
7114 { "006A } { "004A } 37114
7115 { "012F } { "012E } 37115
7116 } 37116
7117 } 37117
7118 {#2} {#3} {#4} {#5} 37118
7119 } 37119
7120 \cs_new:Npn \__text_change_case_upper_lt_aux:nnnnn #1#2#3#4#5 37120
7121 { 37121
7122 \tl_if_blank:nTF {#1} 37122
7123 { \__text_change_case_codepoint:nnnnn { upper } {#2} {#3} {#4} {#5} } 37123
7124 { 37124
7125 \__text_change_case_store:e 37125
7126 { 37126
7127 \codepoint_generate:nn {#1} 37127
7128 { \__text_change_case_catcode:nn {#5} {#1} } 37128
7129 } 37129
7130 \__text_change_case_upper_lt:nnnw {#2} {#3} {#4} 37130
7131 } 37131
7132 } 37132
7133 \cs_new:Npn \__text_change_case_upper_lt:nnnw #1#2#3#4 \q__text_recursion_stop 37133
7134 { 37134
7135 \tl_if_head_is_N_type:nTF {#4} 37135
7136 { \__text_change_case_upper_lt:nnnN } 37136
7137 { \use:c { __text_change_case_next_ #1 :nnn } } 37137
7138 {#1} {#2} {#3} #4 \q__text_recursion_stop 37138
7139 } 37139
7140 \cs_new:Npn \__text_change_case_upper_lt:nnnN #1#2#3#4 37140
```

```
7141 { 3714
7142 \__text_codepoint_process:nN 3714
7143 { \__text_change_case_upper_lt:nnnn {#1} {#2} {#3} } #4 3714
7144 } 3714
7145 \cs_new:Npn \__text_change_case_upper_lt:nnnn #1#2#3#4 3714
7146 { 3714
7147 \bool_lazy_and:nnTF 3714
7148 { 3714
7149 \bool_lazy_or_p:nn 3714
7150 { ! \tl_if_single_p:n {#4} } 3715
7151 { ! \token_if_cs_p:N #4 } 3715
7152 } 3715
7153 { \__text_codepoint_compare_p:nNn {#4} = { "0307 } } 3715
7154 { \use:c { __text_change_case_next_ #1 :nnn } {#1} {#2} {#3} } 3715
7155 { \use:c { __text_change_case_next_ #1 :nnn } {#1} {#2} {#3} #4 } 3715
7156 } 3715
7157 \cs_new:Npn \__text_change_case_title_nl:nnnnn #1#2#3#4#5 3715
7158 { 3715
7159 \tl_if_single:nTF {#5} 3715
7160 { \__text_change_case_title_nl_aux:nnnnn } 3716
7161 { \__text_change_case_codepoint:nnnnn } 3716
7162 {#1} {#2} {#3} {#4} {#5} 3716
7163 } 3716
7164 \cs_new:Npn \__text_change_case_title_nl_aux:nnnnn #1#2#3#4#5 3716
7165 { 3716
7166 \bool_lazy_or:nnTF 3716
7167 { \int_compare_p:nNn {`#5} = { "0049 } } 3716
7168 { \int_compare_p:nNn {`#5} = { "0069 } } 3716
7169 { 3716
7170 \__text_change_case_store:e 3717
7171 { \char_generate:nn { "0049 } { \__text_char_catcode:N #5 } } 3717
7172 \__text_change_case_title_nl:nnnw {#2} {#3} {#4} 3717
7173 } 3717
7174 { \__text_change_case_codepoint:nnnnn {#1} {#2} {#3} {#4} {#5} } 3717
7175 } 3717
7176 \cs_new:Npn \__text_change_case_title_nl:nnnw #1#2#3#4 \q_text_recursion_stop 3717
7177 { 3717
7178 \tl_if_head_is_N_type:nTF {#4} 3717
7179 { \__text_change_case_title_nl:nnnN } 3717
7180 { \use:c { __text_change_case_next_ #1 :nnn } } 3718
7181 {#1} {#2} {#3} #4 \q_text_recursion_stop 3718
7182 } 3718
7183 \cs_new:Npn \__text_change_case_title_nl:nnnN #1#2#3#4 3718
7184 { 3718
7185 \bool_lazy_and:nnTF 3718
7186 { ! \token_if_cs_p:N #4 } 3718
```

```
7187 {
7188     \bool_lazy_or_p:nn
7189     { \int_compare_p:nNn {\`#4} = { "004A } }
7190     { \int_compare_p:nNn {\`#4} = { "006A } }
7191 }
7192 {
7193     \__text_change_case_store:e
7194     { \char_generate:nn { "004A } { \__text_char_catcode:N #4 } }
7195     \use:c { \__text_change_case_next_ #1 :nnn } {#1} {#2} {#3}
7196 }
7197 { \use:c { \__text_change_case_next_ #1 :nnn } {#1} {#2} {#3} #4 }
7198 }
7199 \cs_new:Npn \__text_change_case_lower_tr:nnnnn #1#2#3#4#5
7200 {
7201     \__text_codepoint_compare:nNnTF {#5} = { "0049 }
7202     { \__text_change_case_lower_tr:nnnNw {#1} {#3} {#4} #5 }
7203     {
7204         \__text_codepoint_compare:nNnTF {#5} = { "0130 }
7205         {
7206             \__text_change_case_store:e
7207             {
7208                 \codepoint_generate:nn { "0069 }
7209                 { \__text_change_case_catcode:nn {#5} { "0069 } }
7210             }
7211             \__text_change_case_loop:nnnw {#1} {#3} {#4}
7212         }
7213         { \__text_change_case_codepoint:nnnnn {#1} {#2} {#3} {#4} {#5} }
7214     }
7215 }
7216 \cs_new:Npn \__text_change_case_lower_tr:nnnNw #1#2#3#4#5 \q__text_recursion_stop
7217 {
7218     \tl_if_head_is_N_type:nTF {#5}
7219     { \__text_change_case_lower_tr:NnnnN #4 {#1} {#2} {#3} }
7220     {
7221         \__text_change_case_store:e
7222         {
7223             \codepoint_generate:nn { "0131 }
7224             { \__text_change_case_catcode:nn {#4} { "0131 } }
7225         }
7226         \__text_change_case_loop:nnnw {#1} {#2} {#3}
7227     }
7228     #5 \q__text_recursion_stop
7229 }
7230 \cs_new:Npn \__text_change_case_lower_tr:NnnnN #1#2#3#4#5
7231 {
7232     \__text_codepoint_process:nN
```

```
7233 { \__text_change_case_lower_tr:Nnnnn #1 {#2} {#3} {#4} } #5 3723
7234 } 3723
7235 \cs_new:Npn \__text_change_case_lower_tr:Nnnnn #1#2#3#4#5 3723
7236 { 3723
7237 \bool_lazy_or:nnTF 3723
7238 { 3723
7239 \bool_lazy_and_p:nn 3723
7240 { \tl_if_single_p:n {#5} } 3724
7241 { \token_if_cs_p:N #5 } 3724
7242 } 3724
7243 { ! \__text_codepoint_compare_p:nNn {#5} = { "0307 } } 3724
7244 { 3724
7245 \__text_change_case_store:e 3724
7246 { 3724
7247 \codepoint_generate:nn { "0131 } 3724
7248 { \__text_change_case_catcode:nn {#1} { "0131 } } 3724
7249 } 3724
7250 \__text_change_case_loop:nnnw {#2} {#3} {#4} #5 3725
7251 } 3725
7252 { 3725
7253 \__text_change_case_store:e 3725
7254 { 3725
7255 \codepoint_generate:nn { "0069 } 3725
7256 { \__text_change_case_catcode:nn {#1} { "0069 } } 3725
7257 } 3725
7258 \__text_change_case_loop:nnnw {#2} {#3} {#4} 3725
7259 } 3725
7260 } 3726
7261 \cs_new:Npn \__text_change_case_upper_tr:nnnnn #1#2#3#4#5 3726
7262 { 3726
7263 \__text_codepoint_compare:nNnTF {#5} = { "0069 } 3726
7264 { 3726
7265 \__text_change_case_store:e 3726
7266 { 3726
7267 \codepoint_generate:nn { "0130 } 3726
7268 { \__text_change_case_catcode:nn {#5} { "0130 } } 3726
7269 } 3726
7270 \use:c { __text_change_case_next_ #2 :nnn } {#2} {#3} {#4} 3727
7271 } 3727
7272 { \__text_change_case_codepoint:nnnnn {#1} {#2} {#3} {#4} {#5} } 3727
7273 } 3727
7274 \cs_new_eq:NN \__text_change_case_lower_az:nnnnn 3727
7275 \__text_change_case_lower_tr:nnnnn 3727
7276 \cs_new_eq:NN \__text_change_case_upper_az:nnnnn 3727
7277 \__text_change_case_upper_tr:nnnnn 3727
7278 \group_begin: 3727
```



```
7279 \cs_set_protected:Npn \__text_change_case_setup:NN #1#2
7280 {
7281   \quark_if_recursion_tail_stop:N #1
7282   \tl_const:cn { c__text_lowercase_ \token_to_str:N #1 _tl }
7283   { #2 }
7284   \tl_const:cn { c__text_uppercase_ \token_to_str:N #2 _tl }
7285   { #1 }
7286   \__text_change_case_setup:NN
7287 }
7288 \__text_change_case_setup:NN
7289 \AA \aa
7290 \AE \ae
7291 \DH \dh
7292 \DJ \dj
7293 \IJ \ij
7294 \L \l
7295 \NG \ng
7296 \O \o
7297 \OE \oe
7298 \SS \ss
7299 \TH \th
7300 \q_recursion_tail ?
7301 \q_recursion_stop
7302 \tl_const:cn { c__text_uppercase_ \token_to_str:N \i _tl } { I }
7303 \tl_const:cn { c__text_uppercase_ \token_to_str:N \j _tl } { J }
7304 \group_end:
7305 \tl_if_exist:NT \@expl@finalise@setup@@
7306 {
7307   \tl_gput_right:Nn \@expl@finalise@setup@@
7308   {
7309     \tl_gput_right:Nn \@kernel@after@begindocument
7310     {
7311       \group_begin:
7312       \cs_set_protected:Npn \__text_change_case_setup:Nn #1#2
7313       {
7314         \quark_if_recursion_tail_stop:N #1
7315         \tl_if_single_token:nT {#2}
7316         {
7317           \cs_if_exist:cF
7318             { c__text_uppercase_ \token_to_str:N #1 _tl }
7319             {
7320               \tl_const:cn
7321                 { c__text_uppercase_ \token_to_str:N #1 _tl }
7322                 { #2 }
7323             }
7324           \cs_if_exist:cF
```

```
7325         { c__text_lowercase_ \token_to_str:N #2 _t1 }
7326     {
7327         \tl_const:cn
7328         { c__text_lowercase_ \token_to_str:N #2 _t1 }
7329         { #1 }
7330     }
7331 }
7332 \__text_change_case_setup:Nn
7333 }
7334 \exp_after:wN \__text_change_case_setup:Nn \@uclclist
7335 \q_recursion_tail ?
7336 \q_recursion_stop
7337 \group_end:
7338 }
7339 }
7340 }
7341 \sys_if_engine_opentype:F
7342 {
7343     \text_declare_uppercase_mapping:nn { "01F0 } { \v { J } }
7344 }
7345 %% File: l3text-map.dtx
7346 \cs_new:Npn \__text_map_function:nnN #1#2#3
7347 {
7348     \__text_map_loop:Nnnw #3 {#2} { } #1
7349     \q__text_recursion_tail \q__text_recursion_stop
7350     \prg_break_point:Nn \text_map_break: { }
7351 }
7352 \cs_generate_variant:Nn \__text_map_function:nnN { e }
7353 \cs_new:Npn \__text_map_loop:Nnnw #1#2#3#4 \q__text_recursion_stop
7354 {
7355     \tl_if_head_is_N_type:nTF {#4}
7356     { \__text_map_N_type:NnnN }
7357     {
7358         \tl_if_head_is_group:nTF {#4}
7359         { \__text_map_group:Nnnn }
7360         { \__text_map_space:Nnnw }
7361     }
7362     #1 {#2} {#3} #4 \q__text_recursion_stop
7363 }
7364 \cs_new:Npn \__text_map_group:Nnnn #1#2#3#4
7365 {
7366     \__text_map_output:Nn #1 {#3}
7367     {
7368         \__text_map_loop:Nnnw #1 {#2} { } #4
7369         \q__text_recursion_tail \q__text_recursion_stop
7370         \prg_break_point:Nn \text_map_break: { }
```

```
7371     }
7372     \__text_map_loop:Nnnw #1 {#2} { }
7373 }
7374 \use:e
7375 { \cs_new:Npn \exp_not:N \__text_map_space:Nnnw #1#2#3 \c_space_tl }
7376 {
7377     \__text_map_output:Nn #1 {#3}
7378     #1 { ~ }
7379     \__text_map_loop:Nnnw #1 {#2} { }
7380 }
7381 \cs_new:Npn \__text_map_N_type:NnnN #1#2#3#4
7382 {
7383     \__text_if_q_recursion_tail_stop_do:Nn #4
7384     {
7385         \__text_map_output:Nn #1 {#3}
7386         \text_map_break:
7387     }
7388     \token_if_cs:NTF #4
7389     {
7390         \__text_map_output:Nn #1 {#3}
7391         #1 {#4}
7392         \__text_map_loop:Nnnw #1 {#2} { }
7393     }
7394     {
7395         \__text_codepoint_process:nN
7396         { \__text_map_codepoint:Nnnn #1 {#2} {#3} } #4
7397     }
7398 }
7399 \cs_new:Npn \__text_map_codepoint:Nnnn #1#2#3#4
7400 {
7401     \__text_codepoint_compare:nNnTF {#4} = { "000D }
7402     {
7403         \__text_map_output:Nn #1 {#3}
7404         \__text_map_CR:Nnnw #1 {#2} {#4}
7405     }
7406     {
7407         \__text_codepoint_compare:nNnTF {#4} = { "200D }
7408         { \__text_map_loop:Nnnw #1 {#2} {#3#4} }
7409         { \__text_map_class:Nnnn #1 {#2} {#3} {#4} }
7410     }
7411 }
7412 \cs_new:Npn \__text_map_CR:Nnnw #1#2#3#4 \q__text_recursion_stop
7413 {
7414     \tl_if_head_is_N_type:nTF {#4}
7415     { \__text_map_CR:NnnN #1 {#2} {#3} }
7416     {
```

```
7417         #1 {#3}
7418         \__text_map_loop:Nnnw #1 {#2} { }
7419     }
7420     #4 \q__text_recursion_stop
7421 }
7422 \cs_new:Npn \__text_map_CR:NnnN #1#2#3#4
7423 {
7424     \__text_if_q_recursion_tail_stop_do:Nn #4
7425     {
7426         #1 {#3}
7427         \text_map_break:
7428     }
7429     \bool_lazy_and:nnTF
7430     { ! \token_if_cs_p:N #4 }
7431     { \int_compare_p:nNn {`#4} = { "000A } }
7432     {
7433         \__text_map_output:Nn #1 {#3#4}
7434         \__text_map_loop:Nnnw #1 {#2} { }
7435     }
7436     { \__text_map_loop:Nnnw #1 {#2} { } #3 }
7437 }
7438 \cs_new:Npn \__text_map_class:Nnnn #1#2#3#4
7439 {
7440     \exp_args:Nnnne \__text_map_class:Nnnnn #1 {#2} {#3} {#4}
7441     {
7442         \use:c { __kernel_codepoint_to_ #2 _class:n }
7443         { \__text_codepoint_from_chars:Nw #4 }
7444     }
7445 }
7446 \cs_new:Npn \__text_map_class:Nnnnn #1#2#3#4#5
7447 {
7448     \cs_if_exist_use:cF { __text_map_ #5 :Nnnn }
7449     { \__text_map_Other:Nnnn }
7450     #1 {#2} {#3} {#4}
7451 }
7452 \cs_new:Npn \__text_map_lookahead:Nnnnnw #1#2#3#4#5#6 \q__text_recursion_stop
7453 {
7454     \tl_if_head_is_N_type:nTF {#6}
7455     { \__text_map_lookahead:NnnnnN #1 {#2} {#3} {#4} {#5} }
7456     { \__text_map_loop:Nnnw #1 {#2} {#3} #4 }
7457     #6 \q__text_recursion_stop
7458 }
7459 \cs_new:Npn \__text_map_lookahead:NnnnnN #1#2#3#4#5#6
7460 {
7461     \__text_if_q_recursion_tail_stop_do:Nn #6
7462     {
```

```
7463         #1 {#3}
7464         \tl_if_blank:nF {#4} { #1 {#4} }
7465     }
7466     \token_if_cs:NTF #6
7467     {
7468         #1 {#3}
7469         \__text_map_loop:Nnnw #1 {#2} { } #4
7470     }
7471     { \__text_codepoint_process:nN { #5 #1 {#2} {#3} {#4} } }
7472     #6
7473 }
7474 \prg_new_conditional:Npnn \__text_map_if_ignorable:n #1 { TF }
7475 {
7476     \str_case:nnTF {#1}
7477     {
7478         { Extend }      { }
7479         { Format }      { }
7480         { ZWJ }         { }
7481     }
7482     \prg_return_true:
7483     \prg_return_false:
7484 }
7485 \cs_new:Npn \__text_map_output:Nn #1#2
7486 { \tl_if_blank:nF {#2} { #1 {#2} } }
7487 \cs_new:Npn \text_map_break:
7488 { \prg_map_break:Nn \text_map_break: { } }
7489 \cs_new:Npn \text_map_break:n
7490 { \prg_map_break:Nn \text_map_break: }
7491 \cs_new:Npn \__text_map_Control:Nnnn #1#2#3#4
7492 {
7493     \__text_map_output:Nn #1 {#3}
7494     \__text_map_output:Nn #1 {#4}
7495     \__text_map_loop:Nnnw #1 {#2} { }
7496 }
7497 \cs_new_eq:NN \__text_map_Newline:Nnnn \__text_map_Control:Nnnn
7498 \cs_new:Npn \__text_map_Extend:Nnnn #1#2#3#4
7499 { \__text_map_loop:Nnnw #1 {#2} {#3#4} }
7500 \cs_new_eq:NN \__text_map_Format:Nnnn \__text_map_Extend:Nnnn
7501 \cs_new_eq:NN \__text_map_SpacingMark:Nnnn \__text_map_Extend:Nnnn
7502 \cs_new:Npn \__text_map_Other:Nnnn #1#2#3#4
7503 {
7504     \__text_map_output:Nn #1 {#3}
7505     \__text_map_loop:Nnnw #1 {#2} {#4}
7506 }
7507 \cs_new:Npn \__text_map_Regional_Indicator:Nnnn #1#2#3#4
7508 {
```

```
7509     \__text_map_output:Nn #1 {#3}
7510     \__text_map_lookahead:Nnnnw #1 {#2} {#4} { }
7511     \__text_map_Regional_Indicator_aux:Nnnnn
7512 }
7513 \cs_new:Npn \__text_map_Regional_Indicator_aux:Nnnnn #1#2#3#4#5
7514 {
7515     \bool_lazy_or:nnTF
7516     { \__text_codepoint_compare_p:nNn {#5} < { "1F1E6 } }
7517     { \__text_codepoint_compare_p:nNn {#5} > { "1F1FF } }
7518     {
7519         \str_if_eq:nnTF {#2} { wordbreak }
7520         {
7521             \exp_args:Ne \__text_map_if_ignorable:nTF
7522             {
7523                 \__kernel_codepoint_to_grapheme_class:n
7524                 { \__text_codepoint_from_chars:Nw #5 }
7525             }
7526             {
7527                 \__text_map_lookahead:Nnnnw #1 {#2} {#3#5} { }
7528                 \__text_map_Regional_Indicator_aux:Nnnnn
7529             }
7530             { \__text_map_loop:Nnnw #1 {#2} {#3} #5 }
7531         }
7532         { \__text_map_loop:Nnnw #1 {#2} {#3} #5 }
7533     }
7534     { \__text_map_loop:Nnnw #1 {#2} {#3#5} }
7535 }
7536 \cs_new:Npn \text_map_function:nN #1#2
7537 {
7538     \__text_map_function:enN { \text_expand:n {#1} }
7539     { grapheme } #2
7540 }
7541 \cs_new:Npn \__text_map_Prepend:Nnnn #1#2#3#4
7542 {
7543     \__text_map_output:Nn #1 {#3}
7544     \__text_map_lookahead:Nnnnw #1 { grapheme } {#4} { }
7545     \__text_map_Prepend_aux:Nnnnn
7546 }
7547 \cs_new:Npn \__text_map_Prepend_aux:Nnnnn #1#2#3#4#5
7548 {
7549     \bool_lazy_or:nnTF
7550     { \__text_codepoint_compare_p:nNn {#5} = { "000A } }
7551     { \__text_codepoint_compare_p:nNn {#5} = { "000D } }
7552     {
7553         #1 {#3}
7554         \__text_map_loop:Nnnw #1 { grapheme } {#5}
```

```
7555     }
7556     { \__text_map_Prend:Nnn #1 {#3} {#5} }
7557 }
7558 \cs_new:Npn \__text_map_Prend:Nnn #1#2#3
7559 {
7560     \str_if_eq:eeTF
7561     { Control }
7562     {
7563         \__kernel_codepoint_to_grapheme_class:n
7564         { \__text_codepoint_from_chars:Nw #3 }
7565     }
7566     { \__text_map_loop:Nnnw #1 { grapheme } {#2} #3 }
7567     { \__text_map_loop:Nnnw #1 { grapheme } {#2#3} }
7568 }
7569 \cs_new:Npn \__text_map_L:Nnnn #1#2#3#4
7570 {
7571     \__text_map_output:Nn #1 {#3}
7572     \__text_map_hangul:Nnnw
7573     #1 {#4} { L ; V ; LV ; LVT }
7574 }
7575 \cs_new:Npn \__text_map_LV:Nnnn #1#2#3#4
7576 {
7577     \__text_map_output:Nn #1 {#3}
7578     \__text_map_hangul:Nnnw
7579     #1 {#4} { V ; T }
7580 }
7581 \cs_new_eq:NN \__text_map_V:Nnnn \__text_map_LV:Nnnn
7582 \cs_new:Npn \__text_map_LVT:Nnnn #1#2#3#4
7583 {
7584     \__text_map_output:Nn #1 {#3}
7585     \__text_map_hangul:Nnnw
7586     #1 {#4} { T }
7587 }
7588 \cs_new_eq:NN \__text_map_T:Nnnn \__text_map_LVT:Nnnn
7589 \cs_new:Npn \__text_map_hangul:Nnnw #1#2#3#4 \q__text_recursion_stop
7590 {
7591     \tl_if_head_is_N_type:nTF {#4}
7592     { \__text_map_hangul:NnnN #1 {#2} {#3} }
7593     {
7594         #1 {#2}
7595         \__text_map_loop:Nnnw #1 { grapheme } { }
7596     }
7597     #4 \q__text_recursion_stop
7598 }
7599 \cs_new:Npn \__text_map_hangul:NnnN #1#2#3#4
7600 {
```



```
7601 \__text_if_q_recursion_tail_stop_do:Nn #4
7602 {
7603     #1 {#2}
7604     \text_map_break:
7605 }
7606 \token_if_cs:NTF #4
7607 {
7608     #1 {#2}
7609     \__text_map_loop:Nnnw #1 { grapheme } { }
7610 }
7611 {
7612     \__text_codepoint_process:nN
7613     { \__text_map_hangul:Nnnn #1 {#2} {#3} } #4
7614 }
7615 }
7616 \exp_args_generate:n { Nnne }
7617 \cs_new:Npn \__text_map_hangul:Nnnn #1#2#3#4
7618 {
7619     \exp_args:NNnne \__text_map_hangul_aux:Nnnnw #1 {#2} {#4}
7620     {
7621         \__kernel_codepoint_to_grapheme_class:n
7622         { \__text_codepoint_from_chars:Nw #4 }
7623     }
7624     #3 ; \q_recursion_tail ; \q_recursion_stop
7625 }
7626 \cs_new:Npn \__text_map_hangul_aux:Nnnnw #1#2#3#4#5 ;
7627 {
7628     \quark_if_recursion_tail_stop_do:nn {#5}
7629     { \__text_map_loop:Nnnw #1 { grapheme } {#2} #3 }
7630     \__text_map_hangul:Nnnnnw #1 {#2} {#3} {#4} {#5}
7631 }
7632 \cs_generate_variant:Nn \__text_map_hangul_aux:Nnnnw { Nnne }
7633 \cs_new:Npn \__text_map_hangul:Nnnnnw #1#2#3#4#5#6 \q_recursion_stop
7634 {
7635     \str_if_eq:nnTF {#4} {#5}
7636     { \use:c { __text_map_hangul_ #5 :Nnn } #1 {#2} {#3} }
7637     { \__text_map_hangul_next:Nnnnn #1 {#2} {#3} {#4} {#6} }
7638 }
7639 \cs_new:Npn \__text_map_hangul_next:Nnnnn #1#2#3#4#5
7640 { \__text_map_hangul_aux:Nnnnw #1 {#2} {#3} {#4} #5 \q_recursion_stop }
7641 \cs_new:Npn \__text_map_hangul_end:nw #1#2 \q_text_recursion_stop {#1}
7642 \cs_new:Npn \__text_map_hangul_L:Nnn #1#2#3
7643 {
7644     \__text_map_hangul:Nnnw
7645     #1 {#2#3} { L V { LV } { LVT } }
7646 }
```

```
7647 \cs_new:Npn \__text_map_hangul_LV:Nnn #1#2#3 3764
7648 { 3764
7649 \__text_map_hangul:Nnnw 3764
7650 #1 {#2#3} { VT } 3765
7651 } 3765
7652 \cs_new_eq:NN \__text_map_hangul_V:Nnn \__text_map_hangul_LV:Nnn 3765
7653 \cs_new:Npn \__text_map_hangul_LVT:Nnn #1#2#3 3765
7654 { 3765
7655 \__text_map_hangul:Nnnw 3765
7656 #1 {#2#3} { T } 3765
7657 } 3765
7658 \cs_new_eq:NN \__text_map_hangul_T:Nnn \__text_map_hangul_LVT:Nnn 3765
7659 \cs_new:Npn \text_words_map_function:nN #1#2 3765
7660 { 3766
7661 \__text_map_function:enN { \text_expand:n {#1} } 3766
7662 { wordbreak } #2 3766
7663 } 3766
7664 \cs_new:Npn \__text_map_collect:Nnnnn #1#2#3#4#5 3766
7665 { 3766
7666 \__text_map_lookahead:Nnnnnw #1 { wordbreak } {#2} { } 3766
7667 { \__text_map_collect_auxi:nnnNnnnn {#3} {#4} {#5} } 3766
7668 } 3766
7669 \cs_new:Npn \__text_map_collect_auxi:nnnNnnnn #1#2#3#4#5#6#7#8 3766
7670 { 3767
7671 \exp_args:Ne \__text_map_collect_auxii:nNnnnnn 3767
7672 { 3767
7673 \__kernel_codepoint_to_wordbreak_class:n 3767
7674 { \__text_codepoint_from_chars:Nw #8 } 3767
7675 } 3767
7676 #4 {#6} {#1} {#2} {#3} {#8} 3767
7677 } 3767
7678 \cs_new:Npn \__text_map_collect_auxii:nNnnnnn #1#2#3#4#5#6#7 3767
7679 { 3767
7680 \str_case:neTF {#1} 3768
7681 { 3768
7682 \tl_map_function:eN 3768
7683 { 3768
7684 #4 3768
7685 \str_if_eq:nnF {#4} { { WSegSpace } } { { ExtendNumLet } } 3768
7686 } 3768
7687 \__text_map_collect_auxiii:n 3768
7688 } 3768
7689 { 3768
7690 \cs_if_exist_use:cF { __text_map_ #1 :Nnnn } 3769
7691 { \__text_map_Other:Nnnn } 3769
7692 #2 { wordbreak } { } {#3#7} 3769
```

```
7693 }
7694 {
7695     \__text_map_if_ignorable:nTF {#1}
7696     { \__text_map_collect:Nnnnn #2 {#3#7} {#4} {#5} {#6} }
7697     {
7698         \str_case:neTF {#1}
7699         { \tl_map_function:nN {#5} \__text_map_collect_auxiii:n }
7700         {
7701             \__text_map_lookahead:Nnnnnw #2 { wordbreak } {#3} {#7}
7702             { \__text_map_collect_auxiv:nnNnnnn {#5} {#6} }
7703         }
7704     }
7705     \__text_map_output:Nn #2 {#3}
7706     \__text_map_loop:Nnnw #2 { wordbreak } { } #7
7707 }
7708 }
7709 }
7710 }
7711 \cs_new:Npn \__text_map_collect_auxiii:n #1
7712 { \exp_not:n { {#1} { } } }
7713 \cs_new:Npn \__text_map_collect_auxiv:nnNnnnn #1#2#3#4#5#6#7
7714 {
7715     \exp_args:Ne \__text_map_collect_auxv:nNnnnnn
7716     {
7717         \__kernel_codepoint_to_wordbreak_class:n
7718         { \__text_codepoint_from_chars:Nw #7 }
7719     }
7720     #3 {#5} {#6} {#1} {#2} {#7}
7721 }
7722 \cs_new:Npn \__text_map_collect_auxv:nNnnnnn #1#2#3#4#5#6#7
7723 {
7724     \str_case:neTF {#1}
7725     { \tl_map_function:nN {#6} \__text_map_collect_auxiii:n }
7726     { \use:c { \__text_map_ #1 :Nnnn } #2 { wordbreak } { } {#3#4#7} }
7727     {
7728         \__text_map_if_ignorable:nTF {#1}
7729         {
7730             \__text_map_lookahead:Nnnnnw #2 { wordbreak } {#3} {#4#7}
7731             { \__text_map_collect_auxiv:nnNnnnn {#5} {#6} }
7732         }
7733     }
7734     \__text_map_output:Nn #2 {#3}
7735     \__text_map_loop:Nnnw #2 { wordbreak } { } #4#7
7736 }
7737 }
7738 }
```

```
7739 \cs_new:Npn \__text_map_ALetter:Nnnn #1#2#3#4
7740 {
7741   \__text_map_output:Nn #1 {#3}
7742   \__text_map_collect:Nnnnn #1 {#4}
7743   { { ALetter } { Hebrew_Letter } { Numeric } }
7744   { { MidLetter } { MidNumLet } { Single_Quote } }
7745   { { ALetter } { Hebrew_Letter } }
7746 }
7747 \cs_new:Npn \__text_map_Hebrew_Letter:Nnnn #1#2#3#4
7748 {
7749   \__text_map_output:Nn #1 {#3}
7750   \__text_map_collect:Nnnnn #1 {#4}
7751   { { ALetter } { Hebrew_Letter } { Numeric } { Single_Quote } }
7752   { { MidLetter } { MidNumLet } { Double_Quote } }
7753   { { Hebrew_Letter } }
7754 }
7755 \cs_new:Npn \__text_map_Katakana:Nnnn #1#2#3#4
7756 {
7757   \__text_map_output:Nn #1 {#3}
7758   \__text_map_collect:Nnnnn #1 {#4} { { Katakana } } { } { }
7759 }
7760 \cs_new:Npn \__text_map_Numeric:Nnnn #1#2#3#4
7761 {
7762   \__text_map_output:Nn #1 {#3}
7763   \__text_map_collect:Nnnnn #1 {#4}
7764   { { ALetter } { Hebrew_Letter } { Numeric } }
7765   { { MidNum } { MidNumLet } { Single_Quote } }
7766   { { Numeric } }
7767 }
7768 \cs_new:Npn \__text_map_WSegSpace:Nnnn #1#2#3#4
7769 {
7770   \__text_map_output:Nn #1 {#3}
7771   \__text_map_collect:Nnnnn #1 {#4} { { WSegSpace } } { } { }
7772 }
7773 \cs_new:Npn \__text_map_ExtendNumLet:Nnnn #1#2#3#4
7774 {
7775   \__text_map_output:Nn #1 {#3}
7776   \__text_map_lookahead:Nnnnnw #1 { wordbreak } {#4} { }
7777   \__text_map_ExtendNumLet_auxi:Nnnnn
7778 }
7779 \cs_new:Npn \__text_map_ExtendNumLet_auxi:Nnnnn #1#2#3#4#5
7780 {
7781   \exp_args:Ne \__text_map_ExtendNumLet_auxii:nNnn
7782   {
7783     \__kernel_codepoint_to_wordbreak_class:n
7784     { \__text_codepoint_from_chars:Nw #5 }
```

```
7785     }
7786     #1 {#3} {#5}
7787 }
7788 \cs_new:Npn \__text_map_ExtendNumLet_auxii:nNnn #1#2#3#4
7789 {
7790     \str_case:nnTF {#1}
7791     {
7792         { ALetter }      { }
7793         { Hebrew_Letter } { }
7794         { Numeric }      { }
7795         { Katakana }      { }
7796         { ExtendNumLet }  { }
7797     }
7798     {
7799         \cs_if_exist_use:cF { __text_map_ #1 :Nnnn } % TEMP?
7800         { \__text_map_Other:Nnnn }
7801         #2 { wordbreak } { } {#3#4}
7802     }
7803     {
7804         \__text_map_if_ignorable:nTF {#1}
7805         {
7806             \__text_map_lookahead:Nnnnnw #2 { wordbreak } {#3#4} { }
7807             \__text_map_ExtendNumLet_auxi:Nnnnn
7808         }
7809         {
7810             \__text_map_output:Nn #2 {#3}
7811             \__text_map_loop:Nnnw #2 { wordbreak } { } #4
7812         }
7813     }
7814 }
7815 \cs_new_protected:Npn \text_map_inline:nn #1#2
7816 {
7817     \int_gincr:N \g__kernel_prg_map_int
7818     \cs_gset_protected:cpn
7819     { __text_map_ \int_use:N \g__kernel_prg_map_int :w } ##1 {#2}
7820     \exp_args:Nnc \text_map_function:nN {#1}
7821     { __text_map_ \int_use:N \g__kernel_prg_map_int :w }
7822     \prg_break_point:Nn \text_map_break:
7823     { \int_gdecr:N \g__kernel_prg_map_int }
7824 }
7825 \cs_new_protected:Npn \text_words_map_inline:nn #1#2
7826 {
7827     \int_gincr:N \g__kernel_prg_map_int
7828     \cs_gset_protected:cpn
7829     { __text_map_ \int_use:N \g__kernel_prg_map_int :w } ##1 {#2}
7830     \exp_args:Nnc \text_words_map_function:nN {#1}
```

```
7831 { __text_map_ \int_use:N \g__kernel_prg_map_int :w } 3783
7832 \prg_break_point:Nn \text_map_break: 3783
7833 { \int_gdecr:N \g__kernel_prg_map_int } 3783
7834 } 3783
7835 %% File: l3text-purify.dtx 3783
7836 \__kernel_quark_new_test:N \__text_if_recursion_tail_stop:N 3783
7837 \cs_new:Npn \text_purify:n #1 3783
7838 { 3783
7839 \__kernel_exp_not:w \exp_after:wN 3783
7840 { 3784
7841 \exp:w 3784
7842 \exp_args:Ne \__text_purify:n 3784
7843 { \text_expand:n {#1} } 3784
7844 } 3784
7845 } 3784
7846 \cs_new:Npn \__text_purify:n #1 3784
7847 { 3784
7848 \group_align_safe_begin: 3784
7849 \__text_purify_loop:w #1 3784
7850 \q__text_recursion_tail \q__text_recursion_stop 3785
7851 \__text_purify_result:n { } 3785
7852 } 3785
7853 \cs_new:Npn \__text_purify_store:n #1 3785
7854 { \__text_purify_store:nw {#1} } 3785
7855 \cs_new:Npn \__text_purify_store:nw #1#2 \__text_purify_result:n #3 3785
7856 { #2 \__text_purify_result:n { #3 #1 } } 3785
7857 \cs_new:Npn \__text_purify_end:w #1 \__text_purify_result:n #2 3785
7858 { 3785
7859 \group_align_safe_end: 3785
7860 \exp_end: 3786
7861 #2 3786
7862 } 3786
7863 \cs_new:Npn \__text_purify_loop:w #1 \q__text_recursion_stop 3786
7864 { 3786
7865 \tl_if_head_is_N_type:nTF {#1} 3786
7866 { \__text_purify_N_type:N } 3786
7867 { 3786
7868 \tl_if_head_is_group:nTF {#1} 3786
7869 { \__text_purify_group:n } 3786
7870 { \__text_purify_space:w } 3787
7871 } 3787
7872 #1 \q__text_recursion_stop 3787
7873 } 3787
7874 \cs_new:Npn \__text_purify_group:n #1 { \__text_purify_loop:w #1 } 3787
7875 \exp_last_unbraced:NNo \cs_new:Npn \__text_purify_space:w \c_space_tl 3787
7876 { 3787
```

```
7877 \_text_purify_store:n { ~ } 3787
7878 \_text_purify_loop:w 3787
7879 } 3787
7880 \cs_new:Npn \_text_purify_N_type:N #1 3788
7881 { 3788
7882 \_text_if_q_recursion_tail_stop_do:Nn #1 { \_text_purify_end:w } 3788
7883 \_text_purify_N_type_aux:N #1 3788
7884 } 3788
7885 \cs_new:Npn \_text_purify_N_type_aux:N #1 3788
7886 { 3788
7887 \exp_after:wN \_text_purify_math_search:NNN 3788
7888 \exp_after:wN #1 \l_text_math_delims_tl 3788
7889 \q__text_recursion_tail ? 3788
7890 \q__text_recursion_stop 3789
7891 } 3789
7892 \cs_new:Npn \_text_purify_math_search:NNN #1#2#3 3789
7893 { 3789
7894 \_text_if_q_recursion_tail_stop_do:Nn #2 3789
7895 { \_text_purify_math_cmd:N #1 } 3789
7896 \token_if_eq_meaning:NNTF #1 #2 3789
7897 { 3789
7898 \_text_use_i_delimit_by_q_recursion_stop:nw 3789
7899 { \_text_purify_math_start:NNw #2 #3 } 3789
7900 } 3790
7901 { \_text_purify_math_search:NNN #1 } 3790
7902 } 3790
7903 \cs_new:Npn \_text_purify_math_start:NNw #1#2#3 \q__text_recursion_stop 3790
7904 { 3790
7905 \_text_purify_math_loop:NNw #1#2#3 \q__text_recursion_stop 3790
7906 \_text_purify_math_result:n { } 3790
7907 } 3790
7908 \cs_new:Npn \_text_purify_math_store:n #1 3790
7909 { \_text_purify_math_store:nw {#1} } 3790
7910 \cs_new:Npn \_text_purify_math_store:nw #1#2 \_text_purify_math_result:n #3 3791
7911 { #2 \_text_purify_math_result:n { #3 #1 } } 3791
7912 \cs_new:Npn \_text_purify_math_end:w #1 \_text_purify_math_result:n #2 3791
7913 { 3791
7914 \_text_purify_store:n { $ #2 $ } 3791
7915 \_text_purify_loop:w #1 3791
7916 } 3791
7917 \cs_new:Npn \_text_purify_math_stop:Nw #1 \_text_purify_math_result:n #2 3791
7918 { 3791
7919 \_text_purify_store:n {#1#2} 3791
7920 \_text_purify_end:w 3792
7921 } 3792
7922 \cs_new:Npn \_text_purify_math_loop:NNw #1#2#3 \q__text_recursion_stop 3792
```



```

7923 {
7924   \tl_if_head_is_N_type:nTF {#3}
7925   { \__text_purify_math_N_type:NNN }
7926   {
7927     \tl_if_head_is_group:nTF {#3}
7928     { \__text_purify_math_group:NNn }
7929     { \__text_purify_math_space:NNw }
7930   }
7931   #1#2#3 \q__text_recursion_stop
7932 }
7933 \cs_new:Npn \__text_purify_math_N_type:NNN #1#2#3
7934 {
7935   \__text_if_q_recursion_tail_stop_do:Nn #3
7936   { \__text_purify_math_stop:Nw #1 }
7937   \token_if_eq_meaning:NNTF #3 #2
7938   { \__text_purify_math_end:w }
7939   {
7940     \__text_purify_math_store:n {#3}
7941     \__text_purify_math_loop:NNw #1#2
7942   }
7943 }
7944 \cs_new:Npn \__text_purify_math_group:NNn #1#2#3
7945 {
7946   \__text_purify_math_store:n { {#3} }
7947   \__text_purify_math_loop:NNw #1#2
7948 }
7949 \exp_after:wN \cs_new:Npn \exp_after:wN \__text_purify_math_space:NNw
7950   \exp_after:wN # \exp_after:wN 1
7951   \exp_after:wN # \exp_after:wN 2 \c_space_tl
7952 {
7953   \__text_purify_math_store:n { ~ }
7954   \__text_purify_math_loop:NNw #1#2
7955 }
7956 \cs_new:Npn \__text_purify_math_cmd:N #1
7957 {
7958   \exp_after:wN \__text_purify_math_cmd:NN \exp_after:wN #1
7959   \l_text_math_arg_tl \q__text_recursion_tail \q__text_recursion_stop
7960 }
7961 \cs_new:Npn \__text_purify_math_cmd:NN #1#2
7962 {
7963   \__text_if_q_recursion_tail_stop_do:Nn #2
7964   { \__text_purify_replace:N #1 }
7965   \cs_if_eq:NNTF #2 #1
7966   {
7967     \__text_use_i_delimit_by_q_recursion_stop:nw
7968     { \__text_purify_math_cmd:n }

```

```
7969     }
7970     { \_text_purify_math_cmd:NN #1 }
7971 }
7972 \cs_new:Npn \_text_purify_math_cmd:n #1
7973 { \_text_purify_math_end:w \_text_purify_math_result:n {#1} }
7974 \cs_new:Npn \_text_purify_replace:N #1
7975 {
7976   \bool_lazy_and:nnTF
7977   { \cs_if_exist_p:c { l\_text_purify\_token_to_str:N #1 _tl } }
7978   {
7979     \bool_lazy_or_p:nn
7980     { \token_if_cs_p:N #1 }
7981     { \token_if_active_p:N #1 }
7982   }
7983   {
7984     \exp_args:Nv \_text_purify_replace_auxi:n
7985     { l\_text_purify\_token_to_str:N #1 _tl }
7986   }
7987   {
7988     \bool_lazy_or:nnTF
7989     { \token_if_group_begin_p:N #1 }
7990     { \token_if_group_end_p:N #1 }
7991     { \_text_purify_loop:w }
7992     {
7993       \exp_args:Ne \_text_purify_replace_auxii:n
7994       { \_text_token_to_explicit:N #1 }
7995     }
7996   }
7997 }
7998 \cs_new:Npn \_text_purify_replace_auxi:n #1 { \_text_purify_loop:w #1 }
7999 \cs_new:Npn \_text_purify_replace_auxii:n #1
8000 {
8001   \token_if_cs:NNTF #1
8002   { \_text_purify_expand:N #1 }
8003   {
8004     \_text_purify_store:n {#1}
8005     \_text_purify_loop:w
8006   }
8007 }
8008 \cs_new:Npn \_text_purify_expand:N #1
8009 {
8010   \str_if_eq:nnTF {#1} { \protect }
8011   { \_text_purify_protect:N }
8012   { \_text_purify_encoding:N #1 }
8013 }
8014 \cs_new:Npn \_text_purify_protect:N #1
```

```
8015 { 38015
8016 \_text_if_q_recursion_tail_stop_do:Nn #1 { \_text_purify_end:w } 38016
8017 \_text_purify_loop:w 38017
8018 } 38018
8019 \cs_new:Npn \_text_purify_encoding:N #1 38019
8020 { 38020
8021 \bool_lazy_or:nnTF 38021
8022 { \cs_if_eq_p:NN #1 \@current@cmd } 38022
8023 { \cs_if_eq_p:NN #1 \@changed@cmd } 38023
8024 { \_text_purify_encoding_escape:NN } 38024
8025 { 38025
8026 \_text_if_expandable:NTF #1 38026
8027 { \exp_after:wN \_text_purify_loop:w #1 } 38027
8028 { \_text_purify_loop:w } 38028
8029 } 38029
8030 } 38030
8031 \cs_new:Npn \_text_purify_encoding_escape:NN #1#2 38031
8032 { 38032
8033 \_text_purify_store:n {#1} 38033
8034 \_text_purify_loop:w 38034
8035 } 38035
8036 \cs_new_protected:Npn \text_declare_purify_equivalent:Nn #1#2 38036
8037 { 38037
8038 \tl_clear_new:c { l__text_purify_ \token_to_str:N #1 _tl } 38038
8039 \tl_set:cn { l__text_purify_ \token_to_str:N #1 _tl } {#2} 38039
8040 } 38040
8041 \cs_generate_variant:Nn \text_declare_purify_equivalent:Nn { Ne } 38041
8042 \tl_map_inline:nn 38042
8043 { 38043
8044 \fontencoding 38044
8045 \fontfamily 38045
8046 \fontseries 38046
8047 \fontshape 38047
8048 } 38048
8049 { \text_declare_purify_equivalent:Nn #1 { \use_none:n } } 38049
8050 \text_declare_purify_equivalent:Nn \fontsize { \use_none:nn } 38050
8051 \text_declare_purify_equivalent:Nn \selectfont { } 38051
8052 \text_declare_purify_equivalent:Nn \usefont { \use_none:nnnn } 38052
8053 \exp_args:Nc \text_declare_purify_equivalent:Nn 38053
8054 { @protected@testopt } { \use_none:nnn } 38054
8055 \text_declare_purify_equivalent:Nn \begin { \use:c } 38055
8056 \text_declare_purify_equivalent:Nn \end { \_text_end_env:n } 38056
8057 \cs_new:Npn \_text_end_env:n #1 { \cs:w end #1 \cs_end: } 38057
8058 \text_declare_purify_equivalent:Nn \\\ { } 38058
8059 \tl_map_inline:nn 38059
8060 { \{ \} \# \$ \% \_ } 38060
```

8061	{ \text_declare_purify_equivalent:Ne #1 { \cs_to_str:N #1 } }	38061
8062	\text_declare_purify_equivalent:Nn \label { \use_none:n }	38062
8063	\group_begin:	38063
8064	\char_set_catcode_active:N \~	38064
8065	\use:n	38065
8066	{	38066
8067	\group_end:	38067
8068	\text_declare_purify_equivalent:Ne ~ { \c_space_tl }	38068
8069	}	38069
8070	\text_declare_purify_equivalent:Nn \nobreakspace { ~ }	38070
8071	\text_declare_purify_equivalent:Nn _ { ~ }	38071
8072	\text_declare_purify_equivalent:Nn \, { ~ }	38072
8073	\cs_set_protected:Npn __text_loop:Nn #1#2	38073
8074	{	38074
8075	\quark_if_recursion_tail_stop:N #1	38075
8076	\text_declare_purify_equivalent:Ne #1	38076
8077	{	38077
8078	\codepoint_generate:nn {"#2}	38078
8079	{ \char_value_catcode:n {"#2} }	38079
8080	}	38080
8081	__text_loop:Nn	38081
8082	}	38082
8083	__text_loop:Nn	38083
8084	\AA { 00C5 }	38084
8085	\AE { 00C6 }	38085
8086	\DH { 00D0 }	38086
8087	\DJ { 0110 }	38087
8088	\IJ { 0132 }	38088
8089	\L { 0141 }	38089
8090	\NG { 014A }	38090
8091	\O { 00D8 }	38091
8092	\OE { 0152 }	38092
8093	\TH { 00DE }	38093
8094	\aa { 00E5 }	38094
8095	\ae { 00E6 }	38095
8096	\dh { 00F0 }	38096
8097	\dj { 0111 }	38097
8098	\i { 0131 }	38098
8099	\j { 0237 }	38099
8100	\ij { 0132 }	38100
8101	\l { 0142 }	38101
8102	\ng { 014B }	38102
8103	\o { 00F8 }	38103
8104	\oe { 0153 }	38104
8105	\ss { 00DF }	38105
8106	\th { 00FE }	38106

```
8107 \q_recursion_tail ?
8108 \q_recursion_stop
8109 \text_declare_purify_equivalent:Nn \SS { SS }
8110 \cs_new:Npn \__text_purify_accent:NN #1#2
8111 {
8112   \cs_if_exist:cTF
8113     { c__text_purify_ \token_to_str:N #1 _ \token_to_str:N #2 _tl }
8114     {
8115       \exp_not:v
8116         { c__text_purify_ \token_to_str:N #1 _ \token_to_str:N #2 _tl }
8117     }
8118     {
8119       \exp_not:n {#2}
8120       \exp_not:v { c__text_purify_ \token_to_str:N #1 _tl }
8121     }
8122 }
8123 \tl_map_inline:nn { \` \' \^ \~ \= \u \. \" \r \H \v \d \c \k \b \t }
8124 { \text_declare_purify_equivalent:Nn #1 { \__text_purify_accent:NN #1 } }
8125 \group_begin:
8126 \cs_set_protected:Npn \__text_loop:Nn #1#2
8127 {
8128   \quark_if_recursion_tail_stop:N #1
8129   \tl_const:ce { c__text_purify_ \token_to_str:N #1 _tl }
8130   { \codepoint_generate:nn {"#2} { \char_value_catcode:n { "#2 } } }
8131   \__text_loop:Nn
8132 }
8133 \__text_loop:Nn
8134 \` { 0300 }
8135 \' { 0301 }
8136 \^ { 0302 }
8137 \~ { 0303 }
8138 \= { 0304 }
8139 \u { 0306 }
8140 \. { 0307 }
8141 \" { 0308 }
8142 \r { 030A }
8143 \H { 030B }
8144 \v { 030C }
8145 \d { 0323 }
8146 \c { 0327 }
8147 \k { 0328 }
8148 \b { 0331 }
8149 \t { 0361 }
8150 \q_recursion_tail { }
8151 \q_recursion_stop
8152 \cs_set_protected:Npn \__text_loop:NNn #1#2#3
```

8153	{	3815
8154	\quark_if_recursion_tail_stop:N #1	3815
8155	\tl_const:ce	3815
8156	{ c__text_purify_ \token_to_str:N #1 _ \token_to_str:N #2 _tl }	3815
8157	{ \codepoint_generate:nn {"#3"} { \char_value_catcode:n { "#3" } } }	3815
8158	__text_loop:NNn	3815
8159	}	3815
8160	__text_loop:NNn	3816
8161	\` A { 00C0 }	3816
8162	\` A { 00C1 }	3816
8163	\^ A { 00C2 }	3816
8164	\~ A { 00C3 }	3816
8165	\" A { 00C4 }	3816
8166	\r A { 00C5 }	3816
8167	\c C { 00C7 }	3816
8168	\` E { 00C8 }	3816
8169	\` E { 00C9 }	3816
8170	\^ E { 00CA }	3817
8171	\" E { 00CB }	3817
8172	\` I { 00CC }	3817
8173	\` I { 00CD }	3817
8174	\^ I { 00CE }	3817
8175	\" I { 00CF }	3817
8176	\~ N { 00D1 }	3817
8177	\` O { 00D2 }	3817
8178	\` O { 00D3 }	3817
8179	\^ O { 00D4 }	3817
8180	\~ O { 00D5 }	3818
8181	\" O { 00D6 }	3818
8182	\` U { 00D9 }	3818
8183	\` U { 00DA }	3818
8184	\^ U { 00DB }	3818
8185	\" U { 00DC }	3818
8186	\` Y { 00DD }	3818
8187	\` a { 00E0 }	3818
8188	\` a { 00E1 }	3818
8189	\^ a { 00E2 }	3818
8190	\~ a { 00E3 }	3819
8191	\" a { 00E4 }	3819
8192	\r a { 00E5 }	3819
8193	\c c { 00E7 }	3819
8194	\` e { 00E8 }	3819
8195	\` e { 00E9 }	3819
8196	\^ e { 00EA }	3819
8197	\" e { 00EB }	3819
8198	\` i { 00EC }	3819

8199	\` \i	{ 00EC }	38199
8200	\' i	{ 00ED }	38200
8201	\' \i	{ 00ED }	38201
8202	\^ i	{ 00EE }	38202
8203	\^ \i	{ 00EE }	38203
8204	\" i	{ 00EF }	38204
8205	\" \i	{ 00EF }	38205
8206	\~ n	{ 00F1 }	38206
8207	\` o	{ 00F2 }	38207
8208	\' o	{ 00F3 }	38208
8209	\^ o	{ 00F4 }	38209
8210	\~ o	{ 00F5 }	38210
8211	\" o	{ 00F6 }	38211
8212	\` u	{ 00F9 }	38212
8213	\' u	{ 00FA }	38213
8214	\^ u	{ 00FB }	38214
8215	\" u	{ 00FC }	38215
8216	\' y	{ 00FD }	38216
8217	\" y	{ 00FF }	38217
8218	\= A	{ 0100 }	38218
8219	\= a	{ 0101 }	38219
8220	\u A	{ 0102 }	38220
8221	\u a	{ 0103 }	38221
8222	\k A	{ 0104 }	38222
8223	\k a	{ 0105 }	38223
8224	\' C	{ 0106 }	38224
8225	\' c	{ 0107 }	38225
8226	\^ C	{ 0108 }	38226
8227	\^ c	{ 0109 }	38227
8228	\. C	{ 010A }	38228
8229	\. c	{ 010B }	38229
8230	\v C	{ 010C }	38230
8231	\v c	{ 010D }	38231
8232	\v D	{ 010E }	38232
8233	\v d	{ 010F }	38233
8234	\= E	{ 0112 }	38234
8235	\= e	{ 0113 }	38235
8236	\u E	{ 0114 }	38236
8237	\u e	{ 0115 }	38237
8238	\. E	{ 0116 }	38238
8239	\. e	{ 0117 }	38239
8240	\k E	{ 0118 }	38240
8241	\k e	{ 0119 }	38241
8242	\v E	{ 011A }	38242
8243	\v e	{ 011B }	38243
8244	\^ G	{ 011C }	38244

8245	\^ g	{ 011D }	38245
8246	\u G	{ 011E }	38246
8247	\u g	{ 011F }	38247
8248	\. G	{ 0120 }	38248
8249	\. g	{ 0121 }	38249
8250	\c G	{ 0122 }	38250
8251	\c g	{ 0123 }	38251
8252	\^ H	{ 0124 }	38252
8253	\^ h	{ 0125 }	38253
8254	\~ I	{ 0128 }	38254
8255	\~ i	{ 0129 }	38255
8256	\~ \i	{ 0129 }	38256
8257	\= I	{ 012A }	38257
8258	\= i	{ 012B }	38258
8259	\= \i	{ 012B }	38259
8260	\u I	{ 012C }	38260
8261	\u i	{ 012D }	38261
8262	\u \i	{ 012D }	38262
8263	\k I	{ 012E }	38263
8264	\k i	{ 012F }	38264
8265	\k \i	{ 012F }	38265
8266	\. I	{ 0130 }	38266
8267	\^ J	{ 0134 }	38267
8268	\^ j	{ 0135 }	38268
8269	\^ \j	{ 0135 }	38269
8270	\c K	{ 0136 }	38270
8271	\c k	{ 0137 }	38271
8272	\' L	{ 0139 }	38272
8273	\' l	{ 013A }	38273
8274	\c L	{ 013B }	38274
8275	\c l	{ 013C }	38275
8276	\v L	{ 013D }	38276
8277	\v l	{ 013E }	38277
8278	\. L	{ 013F }	38278
8279	\. l	{ 0140 }	38279
8280	\' N	{ 0143 }	38280
8281	\' n	{ 0144 }	38281
8282	\c N	{ 0145 }	38282
8283	\c n	{ 0146 }	38283
8284	\v N	{ 0147 }	38284
8285	\v n	{ 0148 }	38285
8286	\= O	{ 014C }	38286
8287	\= o	{ 014D }	38287
8288	\u O	{ 014E }	38288
8289	\u o	{ 014F }	38289
8290	\H O	{ 0150 }	38290

8291	\H o	{ 0151 }	3829
8292	\' R	{ 0154 }	3829
8293	\' r	{ 0155 }	3829
8294	\c R	{ 0156 }	3829
8295	\c r	{ 0157 }	3829
8296	\v R	{ 0158 }	3829
8297	\v r	{ 0159 }	3829
8298	\' S	{ 015A }	3829
8299	\' s	{ 015B }	3829
8300	\^ S	{ 015C }	3830
8301	\^ s	{ 015D }	3830
8302	\c S	{ 015E }	3830
8303	\c s	{ 015F }	3830
8304	\v S	{ 0160 }	3830
8305	\v s	{ 0161 }	3830
8306	\c T	{ 0162 }	3830
8307	\c t	{ 0163 }	3830
8308	\v T	{ 0164 }	3830
8309	\v t	{ 0165 }	3830
8310	\~ U	{ 0168 }	3831
8311	\~ u	{ 0169 }	3831
8312	\= U	{ 016A }	3831
8313	\= u	{ 016B }	3831
8314	\u U	{ 016C }	3831
8315	\u u	{ 016D }	3831
8316	\r U	{ 016E }	3831
8317	\r u	{ 016F }	3831
8318	\H U	{ 0170 }	3831
8319	\H u	{ 0171 }	3831
8320	\k U	{ 0172 }	3832
8321	\k u	{ 0173 }	3832
8322	\^ W	{ 0174 }	3832
8323	\^ w	{ 0175 }	3832
8324	\^ Y	{ 0176 }	3832
8325	\^ y	{ 0177 }	3832
8326	\" Y	{ 0178 }	3832
8327	\' Z	{ 0179 }	3832
8328	\' z	{ 017A }	3832
8329	\. Z	{ 017B }	3832
8330	\. z	{ 017C }	3833
8331	\v Z	{ 017D }	3833
8332	\v z	{ 017E }	3833
8333	\v A	{ 01CD }	3833
8334	\v a	{ 01CE }	3833
8335	\v I	{ 01CF }	3833
8336	\v \i	{ 01D0 }	3833

8337	\v i { 01D0 }	3833
8338	\v O { 01D1 }	3833
8339	\v o { 01D2 }	3833
8340	\v U { 01D3 }	3834
8341	\v u { 01D4 }	3834
8342	\v G { 01E6 }	3834
8343	\v g { 01E7 }	3834
8344	\v K { 01E8 }	3834
8345	\v k { 01E9 }	3834
8346	\k O { 01EA }	3834
8347	\k o { 01EB }	3834
8348	\v \j { 01F0 }	3834
8349	\v j { 01F0 }	3834
8350	\' G { 01F4 }	3835
8351	\' g { 01F5 }	3835
8352	\` N { 01F8 }	3835
8353	\` n { 01F9 }	3835
8354	\' \AE { 01FC }	3835
8355	\' \ae { 01FD }	3835
8356	\' \O { 01FE }	3835
8357	\' \o { 01FF }	3835
8358	\v H { 021E }	3835
8359	\v h { 021F }	3835
8360	\. A { 0226 }	3836
8361	\. a { 0227 }	3836
8362	\c E { 0228 }	3836
8363	\c e { 0229 }	3836
8364	\. O { 022E }	3836
8365	\. o { 022F }	3836
8366	\= Y { 0232 }	3836
8367	\= y { 0233 }	3836
8368	\q_recursion_tail ? { }	3836
8369	\q_recursion_stop	3836
8370	\group_end:	3837
8371	%% File: l3legacy.dtx	3837
8372	\prg_new_conditional:Npnn \legacy_if:n #1 { p , T , F , TF }	3837
8373	{	3837
8374	\exp_after:wN \reverse_if:N	3837
8375	\cs:w if#1 \cs_end:	3837
8376	\prg_return_false:	3837
8377	\else:	3837
8378	\prg_return_true:	3837
8379	\fi:	3837
8380	}	3838
8381	\cs_new_protected:Npn \legacy_if_set_true:n #1	3838
8382	{ \cs_set_eq:cN { if#1 } \if_true: }	3838

```
8383 \cs_new_protected:Npn \legacy_if_set_false:n #1
8384 { \cs_set_eq:cN { if#1 } \if_false: }
8385 \cs_new_protected:Npn \legacy_if_gset_true:n #1
8386 { \cs_gset_eq:cN { if#1 } \if_true: }
8387 \cs_new_protected:Npn \legacy_if_gset_false:n #1
8388 { \cs_gset_eq:cN { if#1 } \if_false: }
8389 \cs_new_protected:Npn \legacy_if_set:nn #1#2
8390 {
8391   \bool_if:nTF {#2} \legacy_if_set_true:n \legacy_if_set_false:n
8392   {#1}
8393 }
8394 \cs_new_protected:Npn \legacy_if_gset:nn #1#2
8395 {
8396   \bool_if:nTF {#2} \legacy_if_gset_true:n \legacy_if_gset_false:n
8397   {#1}
8398 }
8399 %% File: l3deprecation.dtx
8400 \cs_new_protected:Npn \__kernel_patch_deprecation:nnNNpn #1#2#3#4#5#
8401 { \__deprecation_patch_aux:nnNNnn {#1} {#2} #3 #4 {#5} }
8402 \cs_new_protected:Npn \__deprecation_patch_aux:nnNNnn #1#2#3#4#5#6
8403 {
8404   \__kernel_deprecation_code:nn
8405   {
8406     \tex_let:D #4 \scan_stop:
8407     \__kernel_deprecation_error:Nnn #4 {#2} {#1}
8408   }
8409   { \tex_let:D #4 \scan_stop: }
8410   \cs_if_eq:NNTF #3 \cs_gset_protected:Npn
8411   { \__deprecation_warn_once:nnNnn {#1} {#2} #4 {#5} {#6} }
8412   { \__deprecation_patch_aux:Nn #3 { #4 #5 {#6} } }
8413 }
8414 \cs_new_protected:Npn \__deprecation_warn_once:nnNnn #1#2#3#4#5
8415 {
8416   \cs_gset_protected:Npe #3
8417   {
8418     \__kernel_if_debug:TF
8419     {
8420       \exp_not:N \msg_warning:nneee
8421       { deprecation } { deprecated-command }
8422       {#1}
8423       { \token_to_str:N #3 }
8424       { \tl_to_str:n {#2} }
8425     }
8426     { }
8427     \exp_not:n { \cs_gset_protected:Npn #3 #4 {#5} }
8428     \exp_not:N #3
```

```

8429 }
8430 \__kernel_deprecation_code:nn { }
8431 { \cs_set_protected:Npn #3 #4 {#5} }
8432 }
8433 \cs_new_protected:Npn \__deprecation_patch_aux:Nn #1#2
8434 {
8435   #1 #2
8436   \cs_if_eq:NNTF #1 \cs_gset_protected:Npn
8437     { \__kernel_deprecation_code:nn { } { \cs_set_protected:Npn #2 } }
8438     { \__kernel_deprecation_code:nn { } { \cs_set:Npn #2 } }
8439 }
8440 \cs_new_protected:Npn \__kernel_deprecation_error:Nnn #1#2#3
8441 {
8442   \tex_protected:D \tex_outer:D \tex_edef:D #1
8443   {
8444     \exp_not:N \msg_expandable_error:nnnnn
8445       { deprecation } { deprecated-command }
8446       { \tl_to_str:n {#3} } { \token_to_str:N #1 } { \tl_to_str:n {#2} }
8447     \exp_not:N \msg_error:nneee
8448       { deprecation } { deprecated-command }
8449       { \tl_to_str:n {#3} } { \token_to_str:N #1 } { \tl_to_str:n {#2} }
8450   }
8451 }
8452 \msg_new:nnn { deprecation } { deprecated-command }
8453 {
8454   \tl_if_blank:nF {#3} { Use~ \tl_trim_spaces:n {#3} ~not~ }
8455   #2~deprecated~on~#1.
8456 }
8457 \cs_new:Npn \cs_argument_spec:N { \cs_parameter_spec:N }
8458 \cs_new_protected:Npn \iow_shipout_x:Nn { \iow_shipout_e:Nn }
8459 \cs_generate_variant:Nn \iow_shipout_x:Nn { Nx , c, cx }
8460 \cs_new_protected:cpn { \c__keys_props_root_str .str_set_x:N } #1
8461   { \__keys_variable_set:NnnN #1 { str } { } x }
8462 \cs_new_protected:cpn { \c__keys_props_root_str .str_set_x:c } #1
8463   { \__keys_variable_set:cnnN {#1} { str } { } x }
8464 \cs_new_protected:cpn { \c__keys_props_root_str .str_gset_x:N } #1
8465   { \__keys_variable_set:NnnN #1 { str } { g } x }
8466 \cs_new_protected:cpn { \c__keys_props_root_str .str_gset_x:c } #1
8467   { \__keys_variable_set:cnnN {#1} { str } { g } x }
8468 \cs_new_protected:cpn { \c__keys_props_root_str .tl_set_x:N } #1
8469   { \__keys_variable_set:NnnN #1 { tl } { } x }
8470 \cs_new_protected:cpn { \c__keys_props_root_str .tl_set_x:c } #1
8471   { \__keys_variable_set:cnnN {#1} { tl } { } x }
8472 \cs_new_protected:cpn { \c__keys_props_root_str .tl_gset_x:N } #1
8473   { \__keys_variable_set:NnnN #1 { tl } { g } x }
8474 \cs_new_protected:cpn { \c__keys_props_root_str .tl_gset_x:c } #1

```

```
8475 { \__keys_variable_set:cnnN {#1} { t1 } { g } x } 38475
8476 \cs_new_protected:Npn \keys_set_filter:nnn { \keys_set_exclude_groups:nnn } 38476
8477 \cs_generate_variant:Nn \keys_set_filter:nnn { nnV , nnv , nno } 38477
8478 \cs_new_protected:Npn \keys_set_filter:nnnN { \keys_set_exclude_groups:nnnN } 38478
8479 \cs_generate_variant:Nn \keys_set_filter:nnnN { nnV , nnv , nno } 38479
8480 \cs_new_protected:Npn \keys_set_filter:nnnnN { \keys_set_exclude_groups:nnnnN } 38480
8481 \cs_generate_variant:Nn \keys_set_filter:nnnnN { nnV , nnv , nno } 38481
8482 \__kernel_patch_deprecation:nnNNpn { 2024-02-13 } { \msg_set:nnnn } 38482
8483 \cs_new_protected:Npn \msg_gset:nnnn { \msg_set:nnnn } 38483
8484 \__kernel_patch_deprecation:nnNNpn { 2024-02-13 } { \msg_set:nnn } 38484
8485 \cs_new_protected:Npn \msg_gset:nnn { \msg_set:nnn } 38485
8486 \prop_new:N \g__pdf_object_prop 38486
8487 \__kernel_patch_deprecation:nnNNpn { 2022-08-30 } { [\pdf_object_new:n] } 38487
8488 \cs_new_protected:Npn \pdf_object_new:nn #1#2 38488
8489 { 38489
8490     \prop_gput:Nnn \g__pdf_object_prop {#1} {#2} 38490
8491     \pdf_object_new:n {#1} 38491
8492 } 38492
8493 \__kernel_patch_deprecation:nnNNpn { 2022-08-30 } { [\pdf_object_write:n] } 38493
8494 \cs_new_protected:Npn \pdf_object_write:nn #1#2 38494
8495 { 38495
8496     \exp_args:Nee \__pdf_backend_object_write:nnn 38496
8497     { \__pdf_object_retrieve:n {#1} } 38497
8498     { \prop_item:Nn \g__pdf_object_prop {#1} } {#2} 38498
8499     \bool_gset_true:N \g__pdf_init_bool 38499
8500 } 38500
8501 \cs_generate_variant:Nn \pdf_object_write:nn { nx } 38501
8502 \__kernel_patch_deprecation:nnNNpn { 2023-05-03 } { \bool_case:n } 38502
8503 \cs_new:Npn \bool_case_true:n { \bool_case:n } 38503
8504 \__kernel_patch_deprecation:nnNNpn { 2023-05-03 } { \bool_case:nT } 38504
8505 \cs_new:Npn \bool_case_true:nT { \bool_case:nT } 38505
8506 \__kernel_patch_deprecation:nnNNpn { 2023-05-03 } { \bool_case:nF } 38506
8507 \cs_new:Npn \bool_case_true:nF { \bool_case:nF } 38507
8508 \__kernel_patch_deprecation:nnNNpn { 2023-05-03 } { \bool_case:nTF } 38508
8509 \cs_new:Npn \bool_case_true:nTF { \bool_case:nTF } 38509
8510 \__kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \str_lowercase:n } 38510
8511 \cs_new:Npn \str_lower_case:n { \str_lowercase:n } 38511
8512 \__kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \str_lowercase:f } 38512
8513 \cs_new:Npn \str_lower_case:f { \str_lowercase:f } 38513
8514 \__kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \str_uppercase:n } 38514
8515 \cs_new:Npn \str_upper_case:n { \str_uppercase:n } 38515
8516 \__kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \str_uppercase:f } 38516
8517 \cs_new:Npn \str_upper_case:f { \str_uppercase:f } 38517
8518 \__kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \str_casefold:n } 38518
8519 \cs_new:Npn \str_fold_case:n { \str_casefold:n } 38519
8520 \__kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \str_casefold:V } 38520
```


8521 \cs_new:Npn \str_fold_case:V { \str_casefold:V } 38521
8522 __kernel_patch_deprecation:nnNNpn { 2020-10-17 } { \str_casefold:n } 38522
8523 \cs_new:Npn \str_foldcase:n { \str_casefold:n } 38523
8524 __kernel_patch_deprecation:nnNNpn { 2022-10-17 } { \str_casefold:V } 38524
8525 \cs_new:Npn \str_foldcase:V { \str_casefold:V } 38525
8526 __kernel_patch_deprecation:nnNNpn { 2020-08-20 } { } 38526
8527 \cs_new_protected:Npn \str_declare_eight_bit_encoding:nnn #1 38527
8528 { __str_declare_eight_bit_encoding:nnnn {#1} { 1114112 } } 38528
8529 __kernel_patch_deprecation:nnNNpn { 2020-06-18 } { \seq_map_indexed_inline:Nn } 38529
8530 \cs_new_protected:Npn \seq_indexed_map_inline:Nn { \seq_map_indexed_inline:Nn } 38530
8531 __kernel_patch_deprecation:nnNNpn { 2020-06-18 } { \seq_map_indexed_function:NN } 38531
8532 \cs_new:Npn \seq_indexed_map_function:NN { \seq_map_indexed_function:NN } 38532
8533 __kernel_patch_deprecation:nnNNpn { 2023-05-10 } { \seq_map_pairwise_function:NNN } 38533
8534 \cs_new:Npn \seq_mapthread_function:NNN { \seq_map_pairwise_function:NNN } 38534
8535 __kernel_patch_deprecation:nnNNpn { 2023-10-26 } { \seq_set_map_e:NNn } 38535
8536 \cs_new_protected:Npn \seq_set_map_x:NNn { \seq_set_map_e:NNn } 38536
8537 __kernel_patch_deprecation:nnNNpn { 2023-10-26 } { \seq_gset_map_e:NNn } 38537
8538 \cs_new_protected:Npn \seq_gset_map_x:NNn { \seq_gset_map_e:NNn } 38538
8539 __kernel_patch_deprecation:nnNNpn { 2021-01-11 } { (no~longer~required) } 38539
8540 \cs_new_protected:Npn \sys_load_deprecation: { } 38540
8541 __kernel_patch_deprecation:nnNNpn { 2025-03-26 } { (no~longer~required) } 38541
8542 \cs_new:Npn \sys_if_timer_exist:T #1 {#1} 38542
8543 __kernel_patch_deprecation:nnNNpn { 2025-03-26 } { (no~longer~required) } 38543
8544 \cs_new:Npn \sys_if_timer_exist:F #1 { } 38544
8545 __kernel_patch_deprecation:nnNNpn { 2025-03-26 } { (no~longer~required) } 38545
8546 \cs_new:Npn \sys_if_timer_exist:TF #1#2 {#1} 38546
8547 __kernel_patch_deprecation:nnNNpn { 2025-03-26 } { (no~longer~required) } 38547
8548 \cs_new:Npn \sys_if_timer_exist_p: { \c_true_bool } 38548
8549 __kernel_patch_deprecation:nnNNpn { 2023-07-08 } { \text_titlecase_first:n } 38549
8550 \cs_new:Npn \text_titlecase:n #1 38550
8551 { \text_titlecase_first:n { \text_lowercase:n {#1} } } 38551
8552 __kernel_patch_deprecation:nnNNpn { 2023-07-08 } { \text_titlecase_first:nn } 38552
8553 \cs_new:Npn \text_titlecase:nn #1#2 38553
8554 { \text_titlecase_first:nn {#1} { \text_lowercase:n {#2} } } 38554
8555 __kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \text_lowercase:n } 38555
8556 \cs_new:Npn \tl_lower_case:n #1 38556
8557 { \text_lowercase:n {#1} } 38557
8558 __kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \text_lowercase:nn } 38558
8559 \cs_new:Npn \tl_lower_case:nn #1#2 38559
8560 { \text_lowercase:nn {#1} {#2} } 38560
8561 __kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \text_uppercase:n } 38561
8562 \cs_new:Npn \tl_upper_case:n #1 38562
8563 { \text_uppercase:n {#1} } 38563
8564 __kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \text_uppercase:nn } 38564
8565 \cs_new:Npn \tl_upper_case:nn #1#2 38565
8566 { \text_uppercase:nn {#1} {#2} } 38566

8567 __kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \text_titlecase_first:n } 3856
8568 \cs_new:Npn \tl_mixed_case:n #1 3856
8569 { \text_titlecase_first:n { \text_lowercase:n {#1} } } 3856
8570 __kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \text_titlecase_first:nn } 3857
8571 \cs_new:Npn \tl_mixed_case:nn #1#2 3857
8572 { \text_titlecase_first:nn {#1} { \text_lowercase:n {#2} } } 3857
8573 __kernel_patch_deprecation:nnNNpn { 2022-05-23 } { \token_case_meaning:Nn } 3857
8574 \cs_new:Npn \tl_case:Nn { \token_case_meaning:Nn } 3857
8575 __kernel_patch_deprecation:nnNNpn { 2022-05-23 } { \token_case_meaning:NnT } 3857
8576 \cs_new:Npn \tl_case:NnT { \token_case_meaning:NnT } 3857
8577 __kernel_patch_deprecation:nnNNpn { 2022-05-23 } { \token_case_meaning:NnF } 3857
8578 \cs_new:Npn \tl_case:NnF { \token_case_meaning:NnF } 3857
8579 __kernel_patch_deprecation:nnNNpn { 2022-05-23 } { \token_case_meaning:NnTF } 3857
8580 \cs_new:Npn \tl_case:NnTF { \token_case_meaning:NnTF } 3858
8581 \cs_generate_variant:Nn \tl_case:Nn { c } 3858
8582 \prg_generate_conditional_variant:Nnn \tl_case:Nn 3858
8583 { c } { T , F , TF } 3858
8584 __kernel_patch_deprecation:nnNNpn { 2023-10-18 } { \tl_build_begin:N } 3858
8585 \cs_new_protected:Npn \tl_build_clear:N { \tl_build_begin:N } 3858
8586 __kernel_patch_deprecation:nnNNpn { 2023-10-18 } { \tl_build_gbegin:N } 3858
8587 \cs_new_protected:Npn \tl_build_gclear:N { \tl_build_gbegin:N } 3858
8588 __kernel_patch_deprecation:nnNNpn { 2023-10-25 } { \tl_build_get_intermediate:NN } 3858
8589 \cs_new_protected:Npn \tl_build_get:NN { \tl_build_get_intermediate:NN } 3858
8590 __kernel_patch_deprecation:nnNNpn { 2022-10-09 } { [\codepoint_generate:nn] } 3859
8591 \cs_new:Npn \char_to_utfviii_bytes:n { __kernel_codepoint_to_bytes:n } 3859
8592 __kernel_patch_deprecation:nnNNpn { 2022-10-09 } { \codepoint_to_nfd:n } 3859
8593 \cs_new:Npn \char_to_nfd:N #1 { \codepoint_to_nfd:n {`#1} } 3859
8594 __kernel_patch_deprecation:nnNNpn { 2022-10-09 } { \codepoint_to_nfd:n } 3859
8595 \cs_new:Npn \char_to_nfd:n { \codepoint_to_nfd:n } 3859
8596 __kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \text_lowercase:n } 3859
8597 \cs_new:Npn \char_lower_case:N { \text_lowercase:n } 3859
8598 __kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \text_uppercase:n } 3859
8599 \cs_new:Npn \char_upper_case:N { \text_uppercase:n } 3859
8600 __kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \text_titlecase_first:n } 3860
8601 \cs_new:Npn \char_mixed_case:N { \text_titlecase_first:n } 3860
8602 __kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \str_casefold:n } 3860
8603 \cs_new:Npn \char_fold_case:N { \str_casefold:n } 3860
8604 __kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \str_lowercase:n } 3860
8605 \cs_new:Npn \char_str_lower_case:N { \str_lowercase:n } 3860
8606 __kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \str_uppercase:n } 3860
8607 \cs_new:Npn \char_str_upper_case:N { \str_uppercase:n } 3860
8608 __kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \str_titlecase:n } 3860
8609 \cs_new:Npn \char_str_mixed_case:N { \str_titlecase:n } 3860
8610 __kernel_patch_deprecation:nnNNpn { 2020-01-03 } { \str_casefold:n } 3861
8611 \cs_new:Npn \char_str_fold_case:N { \str_casefold:n } 3861
8612 __kernel_patch_deprecation:nnNNpn { 2022-10-17 } { \text_lowercase:n } 3861

```
8613 \cs_new:Npn \char_lowercase:N { \text_lowercase:n } 3861
8614 \__kernel_patch_deprecation:nnNNpn { 2022-10-17 } { \text_uppercase:n } 3861
8615 \cs_new:Npn \char_uppercase:N { \text_uppercase:n } 3861
8616 \__kernel_patch_deprecation:nnNNpn { 2022-10-17 } { \text_titlecase_first:n } 3861
8617 \cs_new:Npn \char_titlecase:N { \text_titlecase_first:n } 3861
8618 \__kernel_patch_deprecation:nnNNpn { 2022-10-17 } { \str_casefold:n } 3861
8619 \cs_new:Npn \char_foldcase:N { \str_casefold:n } 3861
8620 \__kernel_patch_deprecation:nnNNpn { 2022-10-17 } { \str_lowercase:n } 3862
8621 \cs_new:Npn \char_str_lowercase:N { \str_lowercase:n } 3862
8622 \__kernel_patch_deprecation:nnNNpn { 2022-10-17 } 3862
8623 { \tl_to_str:e { \text_titlecase_first:n } } 3862
8624 \cs_new:Npn \char_str_titlecase:N #1 3862
8625 { \tl_to_str:e { \text_titlecase_first:n {#1} } } 3862
8626 \__kernel_patch_deprecation:nnNNpn { 2022-10-17 } { \str_uppercase:n } 3862
8627 \cs_new:Npn \char_str_uppercase:N { \str_uppercase:n } 3862
8628 \__kernel_patch_deprecation:nnNNpn { 2022-10-17 } { \str_casefold:n } 3862
8629 \cs_new:Npn \char_str_foldcase:N { \str_casefold:n } 3862
8630 \tl_map_inline:nn 3863
8631 { 3863
8632 { catcode } { catcode_remove } 3863
8633 { charcode } { charcode_remove } 3863
8634 { meaning } { meaning_remove } 3863
8635 } 3863
8636 { 3863
8637 \use:e 3863
8638 { 3863
8639 \__kernel_patch_deprecation:nnNNpn { 2022-01-11 } { \peek_remove_spaces:n } 3863
8640 \cs_gset_protected:Npn \exp_not:c { peek_ #1 _ignore_spaces:NTF } ##1##2##3 3864
8641 { 3864
8642 \peek_remove_spaces:n 3864
8643 { \exp_not:c { peek_ #1 :NTF } ##1 {##2} {##3} } 3864
8644 } 3864
8645 \__kernel_patch_deprecation:nnNNpn { 2022-01-11 } { \peek_remove_spaces:n } 3864
8646 \cs_gset_protected:Npn \exp_not:c { peek_ #1 _ignore_spaces:NT } ##1##2 3864
8647 { 3864
8648 \peek_remove_spaces:n 3864
8649 { \exp_not:c { peek_ #1 :NT } ##1 {##2} } 3864
8650 } 3865
8651 \__kernel_patch_deprecation:nnNNpn { 2022-01-11 } { \peek_remove_spaces:n } 3865
8652 \cs_gset_protected:Npn \exp_not:c { peek_ #1 _ignore_spaces:NF } ##1##2 3865
8653 { 3865
8654 \peek_remove_spaces:n 3865
8655 { \exp_not:c { peek_ #1 :NF } ##1 {##2} } 3865
8656 } 3865
8657 } 3865
8658 } 3865
```

```
8659 \cs_new_protected:Npn \prop_put_if_new:Nnn { \prop_put_if_not_in:Nnn }
8660 \cs_new_protected:Npn \prop_gput_if_new:Nnn { \prop_gput_if_not_in:Nnn }
8661 \cs_generate_variant:Nn \prop_put_if_new:Nnn
8662 { NnV , NV , c , cnV , cV }
8663 \cs_generate_variant:Nn \prop_gput_if_new:Nnn
8664 { NnV , NV , c , cnV , cV }
8665 %%
8666 %%
8667 %% End of file `expl3-code.tex'.
```