gost705 и gost705s — стили оформления библиографии в соответствии с ГОСТ Р 7.0.5-2008

Станислав Кручинин

E-mail: stanislav.kruchinin@gmail.com

Версия 1.1.9

2011/11/29

Содержание

1   Исходный текст

```
1
2 ⟨∗gost705 | gost705s⟩
3
```

1.1   Поля, используемые в bib-файлах

```
4 ENTRY
5  { address
6    annote
7    author
8    booktitle
9    bookauthor
10   chapter
11   edition
12   editor
13   compiler
14   howpublished
15   institution
```

```
16      journal
17      key
18      month
19      note
20      number
21      organization
22      pages
23      numpages
24      publisher
25      school
26      series
27      title
28      type
29      volume
30      year
31      language
32      booklanguage
33      url
34      urldate
35      isbn
36      doi
37      archive
38      archivePrefix
39      primaryClass
40      eprint
41    }
42    {}
43    { label }
44
```

1.2   Объявление и инициализация переменных и констант

```
45 INTEGERS { output.state before.all mid.sentence after.sentence after.block
46 after.dblslash after.slash after.column after.semicolumn }
47
```

init.state.consts

```
48 FUNCTION {init.state.consts}
49 {
50   #0 'before.all :=
51   #1 'mid.sentence :=
52   #2 'after.sentence :=
53   #3 'after.block :=
54   #4 'after.dblslash :=
55   #5 'after.slash :=
56   #6 'after.column :=
57   #7 'after.semicolumn :=
58 }
59
60 STRINGS { s t }
61
62 STRINGS { curlanguage }
63
```

1.3   Переключение языка

change.language

```
64 FUNCTION {change.language}
65 {
66   booklanguage empty$
67     { "" }
68     { booklanguage  'curlanguage :=
69       "\selectlanguageifdefined{"
70       curlanguage *
71       "}" *
72     }
73   if$
74 }
```

1.4 Вывод библиографической записи

`output.nonnull`

```
76 FUNCTION {output.nonnull}
77 {
78   swap$
79   output.state mid.sentence =
80     { ", " * write$ }
81     { output.state after.block =
82         { add.period$ " " * write$
83           newline$
84           "\newblock " write$
85         }
86         { output.state before.all =
87           'write$
88             { output.state after.dblslash =
89               { "~//" * change.language * " " * write$ }
90               { output.state after.slash =
91                 { "~/ " * write$ }
92                 { output.state after.column =
93                   { ": " * write$ }
94                   { output.state after.semicolumn =
95                     { "; " * write$ }
96                     { add.period$ " " * write$ }
97                     if$
98                   }
99                   if$
100                 }
101                 if$
102               }
103               if$
104             }
105           if$
106         }
107       if$
108       mid.sentence 'output.state :=
109     }
110   if$
111 }
112
```

`output`

```
113 FUNCTION {output}
114 {
115   duplicate$ empty$
116     'pop$
117     'output.nonnull
118   if$
119 }
120
```

`output.check`

```
121 FUNCTION {output.check}
122 {
123   't :=
124   duplicate$ empty$
125     { pop$ "empty " t * " in " * cite$ * warning$ }
126     'output.nonnull
127   if$
128 }
129
```

1.5 Вывод блоков и ограничителей

`fin.entry`

```
           130 FUNCTION {fin.entry}
           131 {
           132   add.period$
           133   write$
           134   newline$
           135 }
           136
new.block
           137 FUNCTION {new.block}
           138 {
           139   output.state before.all =
           140     'skip$
           141     { after.block 'output.state := }
           142   if$
           143 }
           144
new.dblslash
           145 FUNCTION {new.dblslash}
           146 {
           147   output.state before.all =
           148     'skip$
           149     { after.dblslash 'output.state := }
           150   if$
           151 }
           152
new.slash
           153 FUNCTION {new.slash}
           154 {
           155   output.state before.all =
           156     'skip$
           157     { after.slash 'output.state := }
           158   if$
           159 }
           160
new.column
           161 FUNCTION {new.column}
           162 {
           163   output.state before.all =
           164     'skip$
           165     { after.column 'output.state := }
           166   if$
           167 }
           168
new.semicolumn
           169 FUNCTION {new.semicolumn}
           170 {
           171   output.state before.all =
           172     'skip$
           173     { after.semicolumn 'output.state := }
           174   if$
           175 }
           176
new.sentence
           177 FUNCTION {new.sentence}
           178 {
           179   output.state after.block =
           180     'skip$
           181     { output.state before.all =
           182         'skip$
           183         { after.sentence 'output.state := }
           184       if$
           185     }
```

```
186    if$
187 }
188
```

add.blank

```
189 FUNCTION {add.blank} { " " * before.all 'output.state := }
190
```

not

```
191 FUNCTION {not}
192 {
193     { #0 }
194     { #1 }
195   if$
196 }
197
```

and

```
198 FUNCTION {and}
199 {
200     'skip$
201     { pop$ #0 }
202   if$
203 }
204
```

or

```
205 FUNCTION {or}
206 {
207     { pop$ #1 }
208     'skip$
209   if$
210 }
211
```

non.stop

```
212 FUNCTION {non.stop}
213 {
214   duplicate$
215   "}" * add.period$
216   #-1 #1 substring$ "." =
217 }
218
```

new.block.checka

```
219 FUNCTION {new.block.checka}
220 {
221   empty$
222     'skip$
223     'new.block
224   if$
225 }
226
```

new.block.checkb

```
227 FUNCTION {new.block.checkb}
228 {
229   empty$
230   swap$ empty$
231   and
232     'skip$
233     'new.block
234   if$
235 }
236
```

new.sentence.checka

```
237 FUNCTION {new.sentence.checka}
238 {
```

```
239   empty$
240     'skip$
241     'new.sentence
242   if$
243 }
244
```

```
245 FUNCTION {new.sentence.checkb}
246 {
247   empty$
248   swap$ empty$
249   and
250     'skip$
251     'new.sentence
252   if$
253 }
254
```

```
255 FUNCTION {field.or.null}
256 {
257   duplicate$ empty$
258     { pop$ "" }
259     'skip$
260   if$
261 }
262
```

```
263 FUNCTION {emphasize}
264 {
265   duplicate$ empty$
266     { pop$ "" }
267     { "\BibEmph{" swap$ * "}" * }
268   if$
269 }
270
```

```
271 FUNCTION {space.word} { " " swap$ * " " * }
272
```

1.6   Общепринятые сокращения

```
273 FUNCTION {bbl.edby} % { "\bbledby{}" }
274 {
275   curlanguage "english" =
276   { "Ed. by " }
277    { curlanguage "german" =
278      { "ed. " }
279       { curlanguage "russian" =
280         { "{\cyr\CYRP\cyro\cyrd\ \cyrr\cyre\cyrd.\ }" }
281         { "language is not defined: " language  "edited by" * *
282           warning$ "edited by" }
283      if$
284      }
285    if$
286    }
287  if$
288 }
289
```

```
290 FUNCTION {bbl.cmplr}
291 {
292   curlanguage "english" =
293   { "Compiler " }
```

```
294     { curlanguage "german" =
295      { "Hrsg. " }
296       { curlanguage "russian" =
297        { "{\cyr\CYRS\cyro\cyrs\cyrt.\ }" }
298        { "language is not defined: " language  "compiler" * * warning$ "compiler" }
299       if$
300        }
301      if$
302      }
303    if$
304 }
305

bbl.edition
306 FUNCTION {bbl.edition} % { "\bbledition{}" }
307 {
308   curlanguage "english" =
309    { "edition" }
310    { curlanguage "german" =
311      { "Aufl." }
312       { curlanguage "russian" =
313        { "{\cyr\cyri\cyrz\cyrd.}" }
314        { "language is not defined: " language  "edition" * * warning$ "edition" }
315       if$
316       }
317      if$
318      }
319    if$
320 }
321

bbl.vvolume
322 FUNCTION {bbl.vvolume} % { "\bblVolume{}" }
323 {
324   curlanguage "english" =
325    { "Volume" }
326    { curlanguage "german" =
327      { "Band" }
328       { curlanguage "russian" =
329        { "{\cyr\CYRT\cyro\cyrm}" }
330        { "language is not defined: " language  "Volume" * * warning$ "Volume" }
331       if$
332       }
333      if$
334      }
335    if$
336 }
337

bbl.vvol
338 FUNCTION {bbl.vvol} % { "\bblVol{}" }
339 {
340   curlanguage "english" =
341    { "Vol." }
342    { curlanguage "german" =
343      { "Bd." }
344       { curlanguage "russian" =
345        { "{\cyr\CYRT.}" }
346        { "language is not defined: " language  "Vol." * * warning$ "Vol." }
347       if$
348       }
349      if$
350      }
351    if$
352 }
353

bbl.iissue
354 FUNCTION {bbl.iissue} % { "\bblIssue{}" }
```

```
355 {
356   curlanguage "english" =
357     { "Issue" }
358     { curlanguage "german" =
359       { "Heft" }
360       { curlanguage "russian" =
361         { "{\cyr\CYRV\cyrery\cyrp\cyru\cyrs\cyrk}" }
362         { "language is not defined: " language * warning$ "Issue" }
363       if$
364       }
365     if$
366     }
367   if$
368 }
369
```

bbl.iiss
```
370 FUNCTION {bbl.iiss} % { "\bblIss{}" }
371 {
372   curlanguage "english" =
373     { "Iss." }
374     { curlanguage "german" =
375       { "H." }
376       { curlanguage "russian" =
377         { "{\cyr\CYRV\cyrery\cyrp.}" }
378         { "language is not defined: " language * warning$ "Iss." }
379       if$
380       }
381     if$
382     }
383   if$
384 }
385
```

bbl.of
```
386 FUNCTION {bbl.of} % { "\bblof{}" }
387 {
388   curlanguage "english" =
389     { "of" }
390     { curlanguage "german" =
391       { "von" }
392       { curlanguage "russian" =
393         { "{\cyr\cyri\cyrz}" }
394         { "language is not defined: " language * warning$ "of" }
395       if$
396       }
397     if$
398     }
399   if$
400 }
401
```

bbl.etal
```
402 FUNCTION {bbl.etal} % { "\bblof{}" }
403 {
404   curlanguage "english" =
405     { "et~al." }
406     { curlanguage "german" =
407       { "u.~a." }
408       { curlanguage "russian" =
409         { "{\cyr\cyri~\cyrd\cyrr.}" }
410         { "language is not defined: " language * warning$ "etal" }
411       if$
412       }
413     if$
414     }
415   if$
416 }
```

```
                   417
bbl.nnumber
                   418 FUNCTION {bbl.nnumber} % { "\bblNumber{}" }
                   419 {
                   420   curlanguage "english" =
                   421     { "Number" }
                   422     { curlanguage "german" =
                   423       { "Heft" }
                   424       { curlanguage "russian" =
                   425         { "{\cyr\CYRN\cyro\cyrm\cyre\cyrr}" }
                   426         { "language is not defined: " language * warning$ "Number" }
                   427       if$
                   428       }
                   429     if$
                   430     }
                   431   if$
                   432 }
                   433

 bbl.number
                   434 FUNCTION {bbl.number} % { "\bblnumber{}" }
                   435 {
                   436   curlanguage "english" =
                   437     { "number" }
                   438     { curlanguage "german" =
                   439       { "Heft" }
                   440       { curlanguage "russian" =
                   441         { "{\cyr\cyrn\cyro\cyrm\cyre\cyrr}" }
                   442         { "language is not defined: " language * warning$ "number" }
                   443       if$
                   444       }
                   445     if$
                   446     }
                   447   if$
                   448 }
                   449

    bbl.nr
                   450 FUNCTION {bbl.nr} % { "\bblno{}" }
                   451 {
                   452   curlanguage "english" =
                   453     { "no." }
                   454     { curlanguage "german" =
                   455       { "H." }
                   456       { curlanguage "russian" =
                   457         { "{\cyr\textnumero}" }
                   458         { "language is not defined: " language * warning$ "no" }
                   459       if$
                   460       }
                   461     if$
                   462     }
                   463   if$
                   464 }
                   465

   bbl.nnr
                   466 FUNCTION {bbl.nnr} % { "\bblno{}" }
                   467 {
                   468   curlanguage "english" =
                   469     { "No." }
                   470     { curlanguage "german" =
                   471       { "H." }
                   472       { curlanguage "russian" =
                   473         { "{\cyr\textnumero}" }
                   474         { "language is not defined: " language * warning$ "No" }
                   475       if$
                   476       }
                   477     if$
```

```
478       }
479    if$
480 }
481
```

bbl.in
```
482 FUNCTION {bbl.in} % { "\bblin{}" }
483 {
484    curlanguage "english" =
485       { "in" }
486       { curlanguage "german" =
487          { "in" }
488          { curlanguage "russian" =
489             { "{\cyr\cyrv}" }
490             { "language is not defined: " language * warning$ "in" }
491          if$
492          }
493       if$
494       }
495    if$
496 }
497
```

bbl.iin
```
498 FUNCTION {bbl.iin} % { "\bblIn{}" }
499 {
500    curlanguage "english" =
501       { "In" }
502       { curlanguage "german" =
503          { "In" }
504          { curlanguage "russian" =
505             { "{\cyr\CYRV}" }
506             { "language is not defined: " language * warning$ "In" }
507          if$
508          }
509       if$
510       }
511    if$
512 }
513
```

bbl.pages
```
514 FUNCTION {bbl.pages} % { "\bblpp." }
515 {
516    curlanguage "english" =
517       { "p." }
518       { curlanguage "german" =
519          { "S." }
520          { curlanguage "russian" =
521             { "{\cyr\cyrs.}" }
522             { "language is not defined: " language * warning$ "p." }
523          if$
524          }
525       if$
526       }
527    if$
528 }
529
```

bbl.page
```
530 FUNCTION {bbl.page} % { "\bblp." }
531 {
532    curlanguage "english" =
533       { "p." }
534       { curlanguage "german" =
535          { "S." }
536          { curlanguage "russian" =
537             { "{\cyr\cyrs.}"}
538             { "language is not defined: " language * warning$ "p." }
```

```
539        if$
540          }
541      if$
542        }
543    if$
544 }
545
```

```
546 FUNCTION {bbl.ppages} % { "\bblPp." }
547 {
548    curlanguage "english" =
549      { "P." }
550      { curlanguage "german" =
551        { "S." }
552        { curlanguage "russian" =
553          { "{\cyr\CYRS.}" }
554          { "language is not defined: " language * warning$ "P." }
555        if$
556        }
557      if$
558      }
559    if$
560 }
561
```

```
562 FUNCTION {bbl.ppage} % { "\bblP." }
563 {
564    curlanguage "english" =
565      { "P." }
566      { curlanguage "german" =
567        { "S." }
568        { curlanguage "russian" =
569          { "{\cyr\CYRS.}" }
570          { "language is not defined: " language * warning$ "P." }
571        if$
572        }
573      if$
574      }
575    if$
576 }
577
```

```
578 FUNCTION {bbl.techrep}
579 {
580    curlanguage "english" =
581      { "Tech.~Rep." }
582      { curlanguage "german" =
583        { "Tech.~Bericht" }
584        { curlanguage "russian" =
585          { "{\cyr\CYRT\cyre\cyrh.\ \cyrd\cyro\cyrk\cyrl\cyra\cyrd.}" }
586          { "language is not defined: " language * warning$ "Tech Rep" }
587        if$
588        }
589      if$
590      }
591    if$
592 }
593
```

```
594 FUNCTION {bbl.mthesis}
595 {
596    curlanguage "english" =
597      { "Master's thesis" }
598      { curlanguage "german" =
599        { "Diss.~Mag." }
```

```
600    { curlanguage "russian" =
601      { "{\cyr\CYRM\cyra\cyrg\cyri\cyrs\cyrt\cyre\cyrr\cyrs\cyrk\cyra\cyrya\ "
602        "\cyrd\cyri\cyrs\cyrs\cyre\cyrr\cyrt\cyra\cyrc\cyri\cyrya}" * }
603        { "language is not defined: " language * warning$ "In" }
604      if$
605      }
606    if$
607    }
608  if$
609 }
610
```

```
611 FUNCTION {bbl.phdthesis}
612 {
613    curlanguage "english" =
614      { "Ph.\,D. thesis" }
615      { curlanguage "german" =
616        { "Diss.~Ph.\,D." }
617        { curlanguage "russian" =
618          { "{\cyr\CYRK\cyra\cyrn\cyrd\cyri\cyrd\cyra\cyrt\cyrs\cyrk\cyra\cyrya\ "
619            "\cyrd\cyri\cyrs\cyrs\cyre\cyrr\cyrt\cyra\cyrc\cyri\cyrya}" * }
620          { "language is not defined: " language * warning$ "In" }
621        if$
622        }
623      if$
624      }
625    if$
626 }
627
```

```
628 FUNCTION {bbl.docdisser}
629 {
630    curlanguage "english" =
631      { "Dr.\,Sci. dissertation" }
632      { curlanguage "german" =
633        { "Diss.~Dr." }
634        { curlanguage "russian" =
635          { "{\cyr\CYRD\cyro\cyrk\cyrt\cyro\cyrr\cyrs\cyrk\cyra\cyrya\ "
636            "\cyrd\cyri\cyrs\cyrs\cyre\cyrr\cyrt\cyra\cyrc\cyri\cyrya}" * }
637          { "language is not defined: " language * warning$ "In" }
638        if$
639        }
640      if$
641      }
642    if$
643 }
644
```

1.7 Формат элементов библиографических ссылок

```
645 INTEGERS { nameptr namesleft numnames }
646
```

```
647 FUNCTION {format.names}
648 {
649    's :=
650    #1 'nameptr :=
651    s num.names$ 'numnames :=
652    numnames 'namesleft :=
653      { namesleft #0 > }
654      { s nameptr
655        "{vv~}{ll}{~f.}{, jj}" format.name$ 't :=
656        nameptr #1 >
657          { nameptr #4 =
658            numnames #4 > and
659              { "others" 't :=
```

```
660              #1 'namesleft := }
661            'skip$
662          if$
663          namesleft #1 >
664            { ", " * t * }
665            { t "others" =
666              t "~others" =
667              or
668                { " " * bbl.etal * }
669                { ", " * t * }
670              if$
671            }
672          if$
673        }
674          't
675      if$
676      nameptr #1 + 'nameptr :=
677      namesleft #1 - 'namesleft :=
678    }
679  while$
680 }
681
```

```
682 FUNCTION {format.names.rev}
683 {
684   's :=
685   #1 'nameptr :=
686   s num.names$ 'numnames :=
687   numnames 'namesleft :=
688     { namesleft #0 > }
689     { s nameptr
690       "{f.}{~vv}{~ll}{, jj}" format.name$ 't :=
691       nameptr #1 >
692         { nameptr #4 =
693           numnames #4 > and
694             { "others" 't :=
695               #1 'namesleft := }
696             'skip$
697           if$
698           namesleft #1 >
699             { ", " * t * }
700             { t "others" =
701               t "~others" =
702               or
703                 { " " * bbl.etal * }
704                 { ", " * t * }
705               if$
706             }
707           if$
708         }
709          't
710      if$
711      nameptr #1 + 'nameptr :=
712      namesleft #1 - 'namesleft :=
713    }
714  while$
715 }
716
```

```
717 FUNCTION {format.authors}
718 {
719   author empty$
720     { "" }
721     { author format.names emphasize }
722   if$
```

```
                              723 }
                              724

      format.bookauthors
                              725 FUNCTION {format.bookauthors}
                              726 {
                              727   bookauthor empty$
                              728     { "" }
                              729     { bookauthor format.names }
                              730   if$
                              731 }
                              732

    format.authors.after
                              733 FUNCTION {format.authors.after}
                              734 {
                              735   author empty$
                              736     { "" }
                              737     { author format.names.rev }
                              738   if$
                              739 }
                              740

format.bookauthors.after
                              741 FUNCTION {format.bookauthors.after}
                              742 {
                              743   bookauthor empty$
                              744     { "" }
                              745     { bookauthor format.names.rev }
                              746   if$
                              747 }
                              748

    format.editors.after
                              749 FUNCTION {format.editors.after}
                              750 {
                              751   editor empty$
                              752     { "" }
                              753     { bbl.edby editor format.names.rev * }
                              754   if$
                              755 }
                              756

    format.compiler.after
                              757 FUNCTION {format.compiler.after}
                              758 {
                              759   compiler empty$
                              760     { "" }
                              761     { bbl.cmplr compiler format.names.rev * }
                              762   if$
                              763 }
                              764

           format.title
                              765 FUNCTION {format.title}
                              766 {
                              767   title empty$
                              768     { "" }
                              769     { title }
                              770   if$
                              771 }
                              772

            format.date
                              773 FUNCTION {format.date}
                              774 {
                              775   year empty$
                              776     { month empty$
                              777         { "" }
                              778         { "there's a month but no year in " cite$ * warning$ month }
                              779       if$
```

```
780        }
781      { month empty$
782          'year
783          { year ". " quote$ "--- " month * * * * }
784        if$
785      }
786    if$
787 }
788
```

output.address.publisher.date

```
789 FUNCTION {output.address.publisher.date}
790 {
791   address empty$
792     'skip$
793     { address output
794       publisher empty$
795         'skip$
796         { new.column }
797       if$
798     }
799   if$
800   publisher output
801   format.date output
802 }
803
```

output.bibitem

```
804 FUNCTION {output.bibitem}
805 {
806   newline$
807   "\bibitem" "{" * write$
808   cite$ write$
809   "}" write$
810   newline$
811   language empty$
812     { "english" 'curlanguage := }
813     { language  'curlanguage := }
814   if$
815   "\selectlanguageifdefined{" write$
816   curlanguage write$
817   "}" write$ newline$
818   ""
819   before.all 'output.state :=
820 }
821
```

n.dashify

```
822 FUNCTION {n.dashify}
823 {
824   't :=
825   ""
826     { t empty$ not }
827     { t #1 #1 substring$ "-" =
828       { t #1 #2 substring$ "--" = not
829           { "--" *
830             t #2 global.max$ substring$ 't :=
831           }
832           {   { t #1 #1 substring$ "-" = }
833               { "-" *
834                 t #2 global.max$ substring$ 't :=
835               }
836             while$
837           }
838         if$
839       }
840       { t #1 #1 substring$ *
841         t #2 global.max$ substring$ 't :=
```

```
842          }
843        if$
844      }
845    while$
846 }
847
```

word.in

```
848 FUNCTION {word.in} { bbl.iin " " * }
849
```

format.btitle

```
850 FUNCTION {format.btitle} { title }
851
```

tie.or.space.connect

```
852 FUNCTION {tie.or.space.connect}
853 {
854    duplicate$ text.length$ #3 <
855      { "~" }
856      { " " }
857    if$
858    swap$ * *
859 }
860
```

tie.connect

```
861 FUNCTION {tie.connect} { "~" swap$ * * }
862
```

either.or.check

```
863 FUNCTION {either.or.check}
864 {
865    empty$
866      'pop$
867      { "can't use both " swap$ * " fields in " * cite$ * warning$ }
868    if$
869 }
870
```

format.bvolume

```
871 FUNCTION {format.bvolume}
872 {
873    volume empty$
874      { "" }
875      { bbl.vvol volume tie.connect
876        series empty$
877          'skip$
878          { bbl.of space.word * series emphasize * }
879        if$
880        "volume and number" number either.or.check
881      }
882    if$
883 }
884
```

format.number.series

```
885 FUNCTION {format.number.series}
886 {
887    volume empty$
888      { number empty$
889          { series field.or.null }
890          { series empty$
891              { "there's a number but no series in " cite$ * warning$
892                bbl.nnr
893              }
894              { new.sentence
895                bbl.iiss series tie.or.space.connect
896                ", " * bbl.nr
897                tie.or.space.connect
898              }
```

```
899        if$
900          number tie.or.space.connect
901        }
902      if$
903    }
904    { "" }
905    if$
906 }
907
```

is.num

```
908 FUNCTION {is.num}
909 {
910   chr.to.int$
911   duplicate$ "0" chr.to.int$ < not
912   swap$ "9" chr.to.int$ > not and
913 }
914
```

extract.num

```
915 FUNCTION {extract.num}
916 {
917   duplicate$ 't :=
918   "" 's :=
919     { t empty$ not }
920     { t #1 #1 substring$
921       t #2 global.max$ substring$ 't :=
922       duplicate$ is.num
923         { s swap$ * 's := }
924         { pop$ "" 't := }
925       if$
926     }
927   while$
928   s empty$
929     'skip$
930     { pop$ s }
931   if$
932 }
933
```

convert.edition

```
934 FUNCTION {convert.edition} { edition }
935
```

format.edition

```
936 FUNCTION {format.edition}
937 {
938   edition empty$
939     { "" }
940     { output.state mid.sentence =
941         { convert.edition "l" change.case$ " " * bbl.edition * }
942         { convert.edition "t" change.case$ " " * bbl.edition * }
943       if$
944     }
945   if$
946 }
947
948 INTEGERS { multiresult }
949
```

multi.page.check

```
950 FUNCTION {multi.page.check}
951 {
952   't :=
953   #0 'multiresult :=
954     { multiresult not
955       t empty$ not
956       and
957     }
```

```
958     { t #1 #1 substring$
959       duplicate$ "-" =
960       swap$ duplicate$ "," =
961       swap$ "+" =
962       or or
963         { #1 'multiresult := }
964         { t #2 global.max$ substring$ 't := }
965       if$
966     }
967   while$
968   multiresult
969 }
970
```

```
971 FUNCTION {format.pages}
972 {
973   pages empty$
974     { "" }
975     { pages multi.page.check
976         { bbl.ppages pages n.dashify tie.connect }
977         { bbl.ppage pages tie.connect }
978       if$
979     }
980   if$
981 }
982
```

```
983 FUNCTION {format.pages.page}
984 {
985   pages empty$
986     { numpages empty$
987         { "" }
988         { numpages bbl.pages tie.connect }
989       if$
990     }
991     { format.pages }
992   if$
993 }
994
```

```
995 FUNCTION {format.vol.num.pages}
996 {
997   volume field.or.null
998   number empty$
999     'skip$
1000    { ", no." number tie.or.space.connect *
1001      volume empty$
1002        { "there's a number but no volume in " cite$ * warning$ }
1003        'skip$
1004      if$
1005    }
1006  if$
1007  pages empty$
1008    'skip$
1009    { duplicate$ empty$
1010        { pop$ format.pages }
1011        { ": " * pages n.dashify * }
1012      if$
1013    }
1014  if$
1015 }
1016
```

```
1017 FUNCTION {format.volume}
1018 {
```

```
                  1019   volume empty$
                  1020     { "" }
                  1021     { bbl.vvol volume tie.or.space.connect }
                  1022   if$
                  1023 }
                  1024

format.number
                  1025 FUNCTION {format.number}
                  1026 {
                  1027   number empty$
                  1028     { "" }
                  1029     { bbl.nr number tie.or.space.connect }
                  1030   if$
                  1031 }
                  1032

empty.misc.check
                  1033 FUNCTION {empty.misc.check}
                  1034 {
                  1035   author empty$ title empty$ howpublished empty$
                  1036   month empty$ year empty$ note empty$
                  1037   and and and and and
                  1038   key empty$ not and
                  1039     { "all relevant fields are empty in " cite$ * warning$ }
                  1040     'skip$
                  1041   if$
                  1042 }
                  1043

format.thesis.type
                  1044 FUNCTION {format.thesis.type}
                  1045 {
                  1046   type empty$
                  1047     'skip$
                  1048     { pop$
                  1049       type "t" change.case$
                  1050     }
                  1051   if$
                  1052 }
                  1053

format.tr.number
                  1054 FUNCTION {format.tr.number}
                  1055 {
                  1056   type empty$
                  1057     { bbl.techrep }
                  1058     'type
                  1059   if$
                  1060   number empty$
                  1061     { "t" change.case$ }
                  1062     { number tie.or.space.connect }
                  1063   if$
                  1064 }
                  1065

author.before
                  1066 FUNCTION {author.before}
                  1067 {
                  1068   author empty$
                  1069     'skip$
                  1070     { format.authors output
                  1071       new.sentence
                  1072     }
                  1073   if$
                  1074 }
                  1075

bookauthor.before
                  1076 FUNCTION {bookauthor.before}
```

19

```
1077 {
1078   bookauthor empty$
1079     'skip$
1080     { format.bookauthors output
1081       new.sentence
1082     }
1083   if$
1084 }
1085
```

author.after

```
1086 FUNCTION {author.after}
1087 {
1088   author empty$
1089     'skip$
1090     { author num.names$ #3 >
1091       { format.authors.after output
1092         new.semicolumn
1093       }
1094       'skip$
1095     if$
1096     }
1097   if$
1098 }
1099
```

bookauthor.after

```
1100 FUNCTION {bookauthor.after}
1101 {
1102   bookauthor empty$
1103     'skip$
1104     { bookauthor num.names$ #3 >
1105       { format.bookauthors.after output
1106         new.semicolumn
1107       }
1108       'skip$
1109     if$
1110     }
1111   if$
1112 }
1113
```

editor.organization.after

```
1114 FUNCTION {editor.organization.after}
1115 {
1116   compiler empty$
1117     'skip$
1118     { format.compiler.after output
1119       new.semicolumn
1120     }
1121   if$
1122   editor empty$
1123     'skip$
1124     { format.editors.after output
1125       new.semicolumn
1126     }
1127   if$
1128   organization empty$
1129     'skip$
1130     { organization output
1131       new.semicolumn
1132     }
1133   if$
1134 }
1135
```

format.annote

```
1136 FUNCTION {format.annote}
1137 {
```

```
1138  annote empty$
1139    { "" }
1140    { after.sentence 'output.state :=
1141      "\BibAnnote{" annote add.period$ * "}" *
1142    }
1143  if$
1144 }
1145
```

1.8   Создание гиперссылок

format.url

```
1146 FUNCTION {format.url}
1147 {
1148   url empty$
1149    { "" }
1150    { "\BibUrl{" url * "}" * }
1151  if$
1152 }
1153
```

format.urldate

```
1154 FUNCTION {format.urldate}
1155 {
1156   urldate empty$
1157    { "" }
1158    { add.blank
1159      "\BibUrlDate{" urldate * "}" *
1160    }
1161  if$
1162 }
1163
```

output.url

```
1164 FUNCTION {output.url}
1165 {
1166   url empty$
1167    'skip$
1168    { format.url output
1169      format.urldate output
1170    }
1171  if$
1172 }
1173
```

format.isbn

```
1174 FUNCTION {format.isbn}
1175 {
1176   isbn empty$
1177    { "" }
1178    { "ISBN:~\href{http://isbndb.com/search-all.html?kw=" isbn *
1179      "}{" * isbn * "}" *
1180    }
1181  if$
1182 }
1183
```

add.doi

```
1184 FUNCTION {add.doi}
1185 { duplicate$ empty$
1186    'skip$
1187    { doi empty$
1188       'skip$
1189       { "\href{http://dx.doi.org/" doi * "}{" * swap$ * "}" * }
1190      if$
1191    }
1192  if$
1193 }
1194
```

format.eprint

```
1195 FUNCTION {format.eprint}
1196 { eprint empty$
1197    { "" }
1198    { primaryClass empty$
1199       { " \href{http://arxiv.org/abs/" eprint * "}" *
1200       "{" * eprint * "}" *
1201       }
1202       { archivePrefix empty$
1203          { " \href{http://arxiv.org/abs/" eprint * "}" *
1204          "{" * primaryClass * "/" * eprint * "}" *
1205          }
1206          { " \href{http://arxiv.org/abs/" eprint * "}" *
1207          "{" * archivePrefix * ":" * primaryClass * "/" * eprint * "}" *
1208          }
1209          if$
1210       }
1211    if$
1212 }
1213    if$
1214 }
1215
```

1.9   Описание типов библиографических ссылок

article

```
1216 FUNCTION {article}
1217 {
1218   output.bibitem
1219   author.before
1220   format.title "title" output.check
1221   new.dblslash
1222   journal emphasize add.doi "journal" output.check
1223   new.block
1224   format.date output
1225   new.block
1226   format.volume output
1227   format.number output
1228   new.block
1229   format.pages.page output
1230   new.block
1231   note output
1232   new.sentence
1233   format.eprint output
1234   new.block
1235   output.url
1236   format.annote output
1237   fin.entry
1238 }
1239
```

book

```
1240 FUNCTION {book}
1241 {
1242   output.bibitem
1243   author.before
1244   format.btitle add.doi "title" output.check
1245   new.slash
1246   editor.organization.after
1247   new.sentence
1248   format.number.series output
1249   new.block
1250   format.edition output
1251   new.block
1252   output.address.publisher.date
1253   new.block
```

```
1254    format.bvolume output
1255    new.block
1256    format.pages.page output
1257    new.block
1258    format.isbn output
1259    new.block
1260    note output
1261    new.sentence
1262    format.eprint output
1263    new.block
1264    output.url
1265    format.annote output
1266    fin.entry
1267 }
1268
```

booklet

```
1269 FUNCTION {booklet}
1270 {
1271    output.bibitem
1272    author.before
1273    format.title "title" output.check
1274    editor.organization.after
1275    new.block
1276    howpublished output
1277    address output
1278    format.date output
1279    new.block
1280    note output
1281    new.sentence
1282    format.eprint output
1283    new.block
1284    output.url
1285    format.annote output
1286    fin.entry
1287 }
1288
```

inbook

```
1289 FUNCTION {inbook}
1290 {
1291    output.bibitem
1292    author.before
1293    format.btitle add.doi "title" output.check
1294    new.dblslash
1295    bookauthor.before
1296    booktitle "booktitle" output.check
1297    editor.organization.after
1298    new.block
1299    format.edition output
1300    new.block
1301    format.number.series output
1302    new.sentence
1303    output.address.publisher.date
1304    new.block
1305    format.bvolume output
1306    new.block
1307    format.pages.page output
1308    new.block
1309    format.isbn output
1310    new.block
1311    note output
1312    new.sentence
1313    format.eprint output
1314    new.block
1315    output.url
1316    format.annote output
```

incollection

```
1317   fin.entry
1318 }
1319
1320 FUNCTION {incollection}
1321 {
1322   output.bibitem
1323   author.before
1324   new.sentence
1325   format.title add.doi "title" output.check
1326   new.dblslash
1327   booktitle "booktitle" output.check
1328   new.slash
1329   editor.organization.after
1330   new.block
1331   output.address.publisher.date
1332   new.block
1333   format.bvolume output
1334   format.number.series output
1335   new.block
1336   format.pages.page output
1337   new.block
1338   note output
1339   new.block
1340   format.eprint output
1341   new.block
1342   output.url
1343   format.annote output
1344   fin.entry
1345 }
1346
```

inproceedings

```
1347 FUNCTION {inproceedings}
1348 {
1349   output.bibitem
1350   author.before
1351   new.sentence
1352   format.title add.doi "title" output.check
1353   new.dblslash
1354   booktitle "booktitle" output.check
1355   new.slash
1356   editor.organization.after
1357   new.block
1358   format.bvolume output
1359   format.number.series output
1360   new.block
1361   address empty$
1362     { publisher output
1363       format.date "year" output.check
1364     }
1365     { address output.nonnull
1366       new.column
1367       publisher output
1368       format.date "year" output.check
1369     }
1370   if$
1371   new.block
1372   format.pages.page output
1373   new.block
1374   note output
1375   new.sentence
1376   format.eprint output
1377   new.block
1378   output.url
1379   format.annote output
```

```
1380    fin.entry
1381 }
1382
```

conference

```
1383 FUNCTION {conference} { inproceedings }
1384
```

manual

```
1385 FUNCTION {manual}
1386 {
1387   output.bibitem
1388   author empty$
1389     { organization empty$
1390         'skip$
1391         { organization output.nonnull
1392             address output
1393         }
1394       if$
1395     }
1396     { format.authors output.nonnull }
1397   if$
1398   new.block
1399   format.btitle "title" output.check
1400   author empty$
1401     { organization empty$
1402         { address new.block.checka
1403             address output
1404         }
1405         'skip$
1406       if$
1407     }
1408     { organization address new.block.checkb
1409         organization output
1410         address output
1411     }
1412   if$
1413   format.edition output
1414   format.date output
1415   new.block
1416   note output
1417   new.sentence
1418   format.eprint output
1419   new.block
1420   output.url
1421   format.annote output
1422   fin.entry
1423 }
1424
```

mastersthesis

```
1425 FUNCTION {mastersthesis}
1426 {
1427   output.bibitem
1428   format.authors "author" output.check
1429   new.block
1430   format.btitle add.doi "title" output.check
1431   new.block
1432   bbl.mthesis format.thesis.type output.nonnull
1433   school "school" output.check
1434   address output
1435   format.date "year" output.check
1436   new.block
1437   note output
1438   new.block
1439   format.eprint output
1440   new.block
```

misc

```
1441    output.url
1442    format.annote output
1443    fin.entry
1444 }
1445

1446 FUNCTION {misc}
1447 {
1448    output.bibitem
1449    format.authors "author" output.check
1450    new.block
1451    format.title "title" output.check
1452    new.block
1453    howpublished output
1454    new.block
1455    format.date output
1456    new.block
1457    note output
1458    new.sentence
1459    format.eprint output
1460    new.block
1461    output.url
1462    format.annote output
1463    fin.entry
1464 }
1465
```

phdthesis

```
1466 FUNCTION {phdthesis}
1467 {
1468    output.bibitem
1469    format.authors "author" output.check
1470    new.sentence
1471    format.btitle add.doi "title" output.check
1472    new.column
1473    bbl.phdthesis format.thesis.type output.nonnull
1474    new.slash
1475    school "school" output.check
1476    new.block
1477    output.address.publisher.date
1478    new.block
1479    format.pages.page output
1480    note output
1481    new.sentence
1482    format.eprint output
1483    new.block
1484    output.url
1485    format.annote output
1486    fin.entry
1487 }
1488
```

docdisser

```
1489 FUNCTION {docdisser}
1490 {
1491    output.bibitem
1492    format.authors "author" output.check
1493    new.sentence
1494    format.btitle add.doi "title" output.check
1495    new.column
1496    bbl.docdisser format.thesis.type output.nonnull
1497    new.slash
1498    school "school" output.check
1499    new.block
1500    output.address.publisher.date
1501    new.block
```

```
1502    format.pages.page output
1503    note output
1504    new.sentence
1505    format.eprint output
1506    new.block
1507    output.url
1508    format.annote output
1509    fin.entry
1510 }
1511
```

proceedings

```
1512 FUNCTION {proceedings}
1513 {
1514    output.bibitem
1515    format.btitle add.doi "title" output.check
1516    new.slash
1517    editor.organization.after
1518    new.block
1519    format.bvolume output
1520    format.number.series output
1521    address empty$
1522      { publisher output
1523        format.date "year" output.check
1524      }
1525      { address output.nonnull
1526        format.date "year" output.check
1527        new.sentence
1528        publisher output
1529      }
1530    if$
1531    new.block
1532    note output
1533    new.sentence
1534    format.eprint output
1535    new.block
1536    output.url
1537    format.annote output
1538    fin.entry
1539 }
1540
```

techreport

```
1541 FUNCTION {techreport}
1542 {
1543    output.bibitem
1544    author.before
1545    new.sentence
1546    format.title add.doi "title" output.check
1547    new.column
1548    format.tr.number output.nonnull
1549    editor.organization.after
1550    new.block
1551    address output
1552    new.column
1553    institution "institution" output.check
1554    format.date "year" output.check
1555    new.block
1556    note output
1557    new.sentence
1558    format.eprint output
1559    new.block
1560    output.url
1561    format.annote output
1562    fin.entry
1563 }
1564
```

unpublished

```
1565 FUNCTION {unpublished}
1566 {
1567   output.bibitem
1568   author.before
1569   format.btitle "title" output.check
1570   editor.organization.after
1571   new.block
1572   note "note" output.check
1573   new.sentence
1574   format.eprint output
1575   new.block
1576   output.url
1577   format.annote output
1578   fin.entry
1579 }
1580
```

default.type

```
1581 FUNCTION {default.type} { misc }
1582
```

1.10 Сокращения названий месяцев

```
1583 MACRO {jan} {"\bbljan{}"}
1584 MACRO {feb} {"\bblfeb{}"}
1585 MACRO {mar} {"\bblmar{}"}
1586 MACRO {apr} {"\bblapr{}"}
1587 MACRO {may} {"\bblmay{}"}
1588 MACRO {jun} {"\bbljun{}"}
1589 MACRO {jul} {"\bbljul{}"}
1590 MACRO {aug} {"\bblaug{}"}
1591 MACRO {sep} {"\bblsep{}"}
1592 MACRO {oct} {"\bbloct{}"}
1593 MACRO {nov} {"\bblnov{}"}
1594 MACRO {dec} {"\bbldec{}"}
1595
```

1.11 Сокращения названий журналов

1.11.1 Физика и астрономия

Код данной секции создан на основе файла physjour.mbs пакета custom-bib.

```
1596 MACRO {aa}{"Astron. \& Astrophys."}
1597 MACRO {aasup}{"Astron. \& Astrophys. Suppl. Ser."}
1598 MACRO {aj} {"Astron. J."}
1599 MACRO {aph} {"Acta Phys."}
1600 MACRO {advp} {"Adv. Phys."}
1601 MACRO {ajp} {"Amer. J. Phys."}
1602 MACRO {ajm} {"Amer. J. Math."}
1603 MACRO {amsci} {"Amer. Sci."}
1604 MACRO {anofd} {"Ann. Fluid Dyn."}
1605 MACRO {am} {"Ann. Math."}
1606 MACRO {ap} {"Ann. Phys. (NY)"}
1607 MACRO {adp} {"Ann. Phys. (Leipzig)"}
1608 MACRO {ao} {"Appl. Opt."}
1609 MACRO {apl} {"Appl. Phys. Lett."}
1610 MACRO {app} {"Astroparticle Phys."}
1611 MACRO {apj} {"Astrophys. J."}
1612 MACRO {apjsup} {"Astrophys. J. Suppl."}
1613 MACRO {apss} {"Astrophys. Space Sci."}
1614 MACRO {araa} {"Ann. Rev. Astron. Astrophys."}
1615 MACRO {baas} {"Bull. Amer. Astron. Soc."}
1616 MACRO {baps} {"Bull. Amer. Phys. Soc."}
1617 MACRO {cmp} {"Comm. Math. Phys."}
1618 MACRO {cpam} {"Commun. Pure Appl. Math."}
1619 MACRO {cppcf} {"Comm. Plasma Phys. \& Controlled Fusion"}
```

```
1620 MACRO {cpc} {"Comp. Phys. Comm."}
1621 MACRO {cqg} {"Class. Quant. Grav."}
1622 MACRO {cra} {"C. R. Acad. Sci. A"}
1623 MACRO {fed} {"Fusion Eng. \& Design"}
1624 MACRO {ft} {"Fusion Tech."}
1625 MACRO {grg} {"Gen. Relativ. Gravit."}
1626 MACRO {ieeens} {"IEEE Trans. Nucl. Sci."}
1627 MACRO {ieeeps} {"IEEE Trans. Plasma Sci."}
1628 MACRO {ijimw} {"Interntl. J. Infrared \& Millimeter Waves"}
1629 MACRO {ip} {"Infrared Phys."}
1630 MACRO {irp} {"Infrared Phys."}
1631 MACRO {jap} {"J. Appl. Phys."}
1632 MACRO {jasa} {"J. Acoust. Soc. America"}
1633 MACRO {jcp} {"J. Comp. Phys."}
1634 MACRO {jchp} {"J. Chem. Phys."}
1635 MACRO {jetp} {"Sov. Phys.--JETP"}
1636 MACRO {jfe} {"J. Fusion Energy"}
1637 MACRO {jfm} {"J. Fluid Mech."}
1638 MACRO {jmp} {"J. Math. Phys."}
1639 MACRO {jne} {"J. Nucl. Energy"}
1640 MACRO {jnec} {"J. Nucl. Energy, C: Plasma Phys., Accelerators, Thermonucl. Res."}
1641 MACRO {jnm} {"J. Nucl. Mat."}
1642 MACRO {jpc} {"J. Phys. Chem."}
1643 MACRO {jpp} {"J. Plasma Phys."}
1644 MACRO {jpsj} {"J. Phys. Soc. Japan"}
1645 MACRO {jsi} {"J. Sci. Instrum."}
1646 MACRO {jvst} {"J. Vac. Sci. \& Tech."}
1647 MACRO {nat} {"Nature"}
1648 MACRO {nature} {"Nature"}
1649 MACRO {nedf} {"Nucl. Eng. \& Design/Fusion"}
1650 MACRO {nf} {"Nucl. Fusion"}
1651 MACRO {nim} {"Nucl. Inst. \& Meth."}
1652 MACRO {nimpr} {"Nucl. Inst. \& Meth. in Phys. Res."}
1653 MACRO {np} {"Nucl. Phys."}
1654 MACRO {npb} {"Nucl. Phys. B"}
1655 MACRO {nt/f} {"Nucl. Tech./Fusion"}
1656 MACRO {npbpc} {"Nucl. Phys. B (Proc. Suppl.)"}
1657 MACRO {inc} {"Nuovo Cimento"}
1658 MACRO {nc} {"Nuovo Cimento"}
1659 MACRO {pf} {"Phys. Fluids"}
1660 MACRO {pfa} {"Phys. Fluids A: Fluid Dyn."}
1661 MACRO {pfb} {"Phys. Fluids B: Plasma Phys."}
1662 MACRO {pl} {"Phys. Lett."}
1663 MACRO {pla} {"Phys. Lett. A"}
1664 MACRO {plb} {"Phys. Lett. B"}
1665 MACRO {prep} {"Phys. Rep."}
1666 MACRO {pnas} {"Proc. Nat. Acad. Sci. USA"}
1667 MACRO {pp} {"Phys. Plasmas"}
1668 MACRO {ppcf} {"Plasma Phys. \& Controlled Fusion"}
1669 MACRO {phitrsl} {"Philos. Trans. Roy. Soc. London"}
1670 MACRO {prl} {"Phys. Rev. Lett."}
1671 MACRO {pr} {"Phys. Rev."}
1672 MACRO {physrev} {"Phys. Rev."}
1673 MACRO {pra} {"Phys. Rev. A"}
1674 MACRO {prb} {"Phys. Rev. B"}
1675 MACRO {prc} {"Phys. Rev. C"}
1676 MACRO {prd} {"Phys. Rev. D"}
1677 MACRO {pre} {"Phys. Rev. E"}
1678 MACRO {ps} {"Phys. Scripta"}
1679 MACRO {procrsl} {"Proc. Roy. Soc. London"}
1680 MACRO {rmp} {"Rev. Mod. Phys."}
1681 MACRO {rsi} {"Rev. Sci. Inst."}
1682 MACRO {science} {"Science"}
1683 MACRO {sciam} {"Sci. Am."}
1684 MACRO {sam} {"Stud. Appl. Math."}
1685 MACRO {sjpp} {"Sov. J. Plasma Phys."}
```

```
1686 MACRO {spd} {"Sov. Phys.--Doklady"}
1687 MACRO {sptp} {"Sov. Phys.--Tech. Phys."}
1688 MACRO {spu} {"Sov. Phys.--Uspekhi"}
1689 MACRO {st} {"Sky and Telesc."}
1690
```

1.11.2 Оптика

Из файла photjour.mbs.

```
1691 MACRO {appopt} {"Appl. Opt."}
1692 MACRO {bell} {"Bell Syst. Tech. J."}
1693 MACRO {ell} {"Electron. Lett."}
1694 MACRO {jasp} {"J. Appl. Spectr."}
1695 MACRO {jqe} {"IEEE J. Quantum Electron."}
1696 MACRO {jlwt} {"J. Lightwave Technol."}
1697 MACRO {jmo} {"J. Mod. Opt."}
1698 MACRO {josa} {"J. Opt. Soc. America"}
1699 MACRO {josaa} {"J. Opt. Soc. Amer.~A"}
1700 MACRO {josab} {"J. Opt. Soc. Amer.~B"}
1701 MACRO {jdp} {"J. Phys. (Paris)"}
1702 MACRO {oc} {"Opt. Commun."}
1703 MACRO {ol} {"Opt. Lett."}
1704 MACRO {os} {"Opt. Spectrosc."}
1705 MACRO {phtl} {"IEEE Photon. Technol. Lett."}
1706 MACRO {pspie} {"Proc. Soc. Photo-Opt. Instrum. Eng."}
1707 MACRO {sjot} {"Sov. J. Opt. Technol."}
1708 MACRO {sjqe} {"Sov. J. Quantum Electron."}
1709 MACRO {sleb} {"Sov. Phys.--Leb. Inst. Rep."}
1710 MACRO {stph} {"Sov. Phys.--Techn. Phys."}
1711 MACRO {stphl} {"Sov. Techn. Phys. Lett."}
1712 MACRO {vr} {"Vision Res."}
1713 MACRO {zph} {"Z. f. Physik"}
1714 MACRO {zphb} {"Z. f. Physik~B"}
1715 MACRO {zphd} {"Z. f. Physik~D"}
1716
```

1.11.3 Физика конденсированного состояния

```
1717 MACRO {sse} {"Solid-State Electron."}
1718 MACRO {pss} {"Phys. Sol. State"}
1719 MACRO {sst} {"Semicond. Sci. Tech."}
1720 MACRO {nl} {"Nano Lett."}
1721
1722 READ
1723
```

1.12 Сортировка списка литературы по именам авторов и названиям

```
1724 ⟨∗gost705s⟩
```

sortify
```
1725 FUNCTION {sortify}
1726 {
1727   purify$
1728   "l" change.case$
1729 }
1730
1731 INTEGERS { len }
1732
```

chop.word
```
1733 FUNCTION {chop.word}
1734 {
1735   's :=
1736   'len :=
1737   s #1 len substring$ =
```

```
1738     { s len #1 + global.max$ substring$ }
1739     's
1740    if$
1741 }
1742
1743 FUNCTION {sort.format.names}
1744 {
1745   's :=
1746   #1 'nameptr :=
1747   ""
1748   s num.names$ 'numnames :=
1749   numnames 'namesleft :=
1750     { namesleft #0 > }
1751     { nameptr #1 >
1752         { "   " * }
1753         'skip$
1754       if$
1755       s nameptr
1756       "{vv{ } }{ll{ }}{  f{ }}{  jj{ }}"
1757       format.name$ 't :=
1758       nameptr numnames = t "others" = and
1759         { "et al" * }
1760         { t sortify * }
1761       if$
1762       nameptr #1 + 'nameptr :=
1763       namesleft #1 - 'namesleft :=
1764     }
1765   while$
1766 }
1767
```

```
1768 FUNCTION {sort.format.title}
1769 {
1770   't :=
1771   "A " #2
1772     "An " #3
1773       "The " #4 t chop.word
1774     chop.word
1775   chop.word
1776   sortify
1777   #1 global.max$ substring$
1778 }
1779
```

```
1780 FUNCTION {author.sort}
1781 {
1782   author empty$
1783     { key empty$
1784         { "to sort, need author or key in " cite$ * warning$ "" }
1785         { key sortify }
1786       if$
1787     }
1788     { author sort.format.names }
1789   if$
1790 }
1791
```

```
1792 FUNCTION {author.title.sort}
1793 {
1794   author empty$
1795     { title empty$
1796         { key empty$
1797             { "to sort, need author, title, or key in " cite$ * warning$ "" }
1798             { key sortify }
```

```
1799        if$
1800        }
1801        { title sort.format.title }
1802      if$
1803      }
1804      { author sort.format.names }
1805    if$
1806 }
1807
```

presort

```
1808 FUNCTION {presort}
1809 {
1810   author.title.sort
1811   "    "
1812   *
1813   year field.or.null sortify
1814   *
1815   "    "
1816   *
1817   title field.or.null
1818   sort.format.title
1819   *
1820   #1 entry.max$ substring$
1821   'sort.key$ :=
1822 }
1823
1824 ITERATE {presort}
1825
1826 SORT
1827
1828 ⟨/gost705s⟩
1829
```

1.13 Вычисление длины табуляции между номером и текстом

```
1830 STRINGS { longest.label }
1831
1832 INTEGERS { number.label longest.label.width }
1833
```

initialize.longest.label

```
1834 FUNCTION {initialize.longest.label}
1835 {
1836   "" 'longest.label :=
1837   #1 'number.label :=
1838   #0 'longest.label.width :=
1839 }
1840
```

longest.label.pass

```
1841 FUNCTION {longest.label.pass}
1842 {
1843   number.label int.to.str$ 'label :=
1844   number.label #1 + 'number.label :=
1845   label width$ longest.label.width >
1846     { label 'longest.label :=
1847       label width$ 'longest.label.width :=
1848     }
1849     'skip$
1850   if$
1851 }
1852
1853 EXECUTE {initialize.longest.label}
1854
1855 ITERATE {longest.label.pass}
1856
```

1.14 Формирование окружения thebibliography

1.14.1 Преамбула

`begin.bib`

```
1857 FUNCTION {begin.bib}
1858 {
1859   preamble$ empty$
1860     'skip$
1861     { preamble$ write$ newline$ }
1862   if$
1863   "\ifx\undefined\BibEmph\def\BibEmph#1{#1}\else\fi"
1864   write$ newline$
1865   "\ifx\undefined\href\def\href#1#2{#2}\else\fi"
1866   write$ newline$
1867   "\ifx\undefined\url\def\url#1{\texttt{#1}}\else\fi"
1868   write$ newline$
1869   "\ifx\undefined\urlprefix\def\urlprefix{URL: }\else\fi"
1870   write$ newline$
1871   "\ifx\undefined\BibUrl\def\BibUrl#1{\urlprefix\url{#1}}\else\fi"
1872   write$ newline$
1873   "\ifx\undefined\BibUrlDate\long\def\BibUrlDate#1{({%"
1874   write$ newline$
1875   "\cyr\cyrd\cyra\cyrt\cyra\ %"
1876   write$ newline$
1877   "\cyro\cyrb\cyrr\cyra\cyrshch\cyre\cyrn\cyri\cyrya}: #1)}\else\fi"
1878   write$ newline$
1879   "\ifx\undefined\BibAnnote\long\def\BibAnnote#1{#1}\else\fi"
1880   write$ newline$
1881   "\begin{thebibliography}{" longest.label  * "}" *
1882   write$ newline$
1883   "\def\selectlanguageifdefined#1{"
1884   write$ newline$
1885   "\expandafter\ifx\csname date#1\endcsname\relax"
1886   write$ newline$
1887   "\else\language\csname l@#1\endcsname\fi}"
1888   write$ newline$
1889 }
1890
```

1.14.2 Начало окружения

```
1891 EXECUTE {begin.bib}
1892
1893 EXECUTE {init.state.consts}
1894
1895 ITERATE {call.type$}
1896
```

1.14.3 Конец окружения

`end.bib`

```
1897 FUNCTION {end.bib}
1898 {
1899   newline$
1900   "\end{thebibliography}" write$ newline$
1901 }
1902
1903 EXECUTE {end.bib}
1904
1905 ⟨/gost705 | gost705s⟩
```

2 Предметный указатель