

lehramt-informatik

Hermine Bschlangaul *

August 9, 2021

Contents

1	Implementation	2
1.1	aufgaben-einbinden.sty	3
1.2	aufgaben-titel.sty	4
1.3	automaten.sty	5
1.3.1	Endlicher Automat	5
1.3.2	Kellerautomat	7
1.3.3	Turingmaschine	8
1.4	basis.sty	11
1.5	baum.sty	15
1.5.1	Binärbaum	16
1.5.2	AVL-Baum	17
1.5.3	B-Baum	18
1.6	checkbox.sty	19
1.7	chomsky-normalform.sty	20
1.8	cpm.sty	22
1.8.1	Hilfsmakros für Zeitpunkt-Tabelle	23
1.9	cyk-algorithmus.sty	24
1.10	entwurfsmuster.sty	25
1.10.1	Abstrakte Fabrik	25
1.10.2	Adapter	26
1.10.3	Beobachter	27
1.10.4	Dekorierer	28
1.10.5	Einzelstück	29
1.10.6	Erbauer	30
1.10.7	Fabrikmethode	31
1.10.8	Kompositum	32
1.10.9	Modell-Präsentation-Steuerung	32
1.10.10	Zustand	33
1.11	er.sty	35
1.12	formale-sprachen.sty	37
1.13	formatierung.sty	41
1.13.1	Schriftarten / Typographie	41
1.13.2	Farben	41
1.13.3	Überschriften	41
1.13.4	Listen	41
1.13.5	Kasten	41
1.13.6	Header	41
1.14	gantt.sty	42
1.15	grafik.sty	43
1.16	graph.sty	44
1.17	hanoi.sty	46
1.18	komplexitaetstheorie.sty	47

*E-mail: hermine.bschlangaul@gmx.net

1.19	kontrollflussgraph.sty	49
1.20	literatur-dummy.sty	51
1.21	literatur.sty	52
1.22	makros.sty	53
1.23	master-theorem.sty	57
1.24	mathe.sty	60
1.25	minimierung.sty	61
1.26	normalformen.sty	64
1.27	petri.sty	66
1.28	potenzmengen-konstruktion.sty	68
1.29	pseudo.sty	70
1.30	quicksort.sty	71
1.31	relationale-algebra.sty	74
1.32	rmodell.sty	75
1.33	sortieren.sty	76
1.34	spalten.sty	77
1.35	struktogramm.sty	78
1.36	syntax.sty	79
1.37	syntaxbaum.sty	81
1.38	synthese-algorithmus.sty	82
1.39	tabelle.sty	85
1.40	typographie.sty	86
1.41	uml.sty	87
1.42	vollstaendige-induktion.sty	89
1.43	wasserfall.sty	90
1.44	wpkalkuel.sty	91

1 Implementation

1.1 aufgaben-einbinden.sty

```
1 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2 \ProvidesPackage{lehramt-informatik-aufgaben-einbinden}[2020/06/13]

\liAufgabe Eine Aufgaben mit den Pfad relativ zu Wurzelverzeichnis dieses Repository einbinden.
3 \def\liAufgabe#1{
4   \input{\LehramtInformatikRepository/#1.tex}
5 }

\liExamensAufgabe Eine Examensaufgaben mit den Pfad relativ zu Staatsexamen einbinden, z. B.
\liExamensAufgabe{46116/2015/09/Thema-2/Teilaufgabe-1/Aufgabe-3}
6 \def\liExamensAufgabe#1{
7   \input{\LehramtInformatikRepository/Staatsexamen/#1.tex}
8 }

\liExamensAufgabeTTA
9 \def\liExamensAufgabeTTA #1 / #2 / #3 : Thema #4 Teilaufgabe #5 Aufgabe #6 {
10  \input{\LehramtInformatikRepository/Staatsexamen/#1/#2/#3/Thema-#4/Teilaufgabe-
    #5/Aufgabe-#6.tex}
11 }

\liExamensAufgabeTA
12 \def\liExamensAufgabeTA #1 / #2 / #3 : Thema #4 Aufgabe #5 {
13   \input{\LehramtInformatikRepository/Staatsexamen/#1/#2/#3/Thema-#4/Aufgabe-#5.tex}
14 }

\liExamensAufgabeA
15 \def\liExamensAufgabeA #1 / #2 / #3 : Aufgabe #4 {
16   \input{\LehramtInformatikRepository/Staatsexamen/#1/#2/#3/Aufgabe-#4.tex}
17 }

18
```

1.2 aufgaben-titel.sty

```
19 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
20 \ProvidesPackage{lehramt-informatik-aufgaben-titel}[2020/07/07 Zum
21 Setzen von Aufgaben-Titel.]
```

```
22 \ExplSyntaxOn
```

`\liAufgabenTitel` Momentan eine dummy Makro das die Thematik enthält.

```
23 \def\liAufgabenTitel#1{
24 }
```

`\liSetzeAufgabenTitel` Setzen des Titels über eine plist bzw über key-values. Das Makro sollte irgendwann einmal in `\liAufgabenTitel` umbenannt werden.

Die Schlüssel-Werte-Paare sind in der Datei basis.sty definiert.

```
\liSetzeAufgabenTitel{
  Titel = Aufgabe 2,
  Thematik = Petri-Netz,
  Fussnote = sosy:pu:4,
  FussnoteSeite = Seite 11,
  RelativerPfad = Staatsexamen/46116/2016/03/Thema-2/Teilaufgabe-1/Aufgabe-2.tex,
  ExamenNummer = 46116,
  ExamenJahr = 2016,
  ExamenMonat = 03,
  ExamenThemaNr = 2,
  ExamenTeilaufgabeNr = 1,
  ExamenAufgabeNr = 2,
}
```

```
25 \def\liSetzeAufgabenTitel#1{
26
27   \_setze_variablen_zurueck:
28
29   \tl_clear:N \g_auf_examen_telaufgabe_nr_tl
30
31   \keys_set:nn { aufgabentitel } {
32     #1
33   }
34
35   \_setze_relativen_pfad:
36
37   \tl_if_empty:NTF \g_auf_titel_tl
38   {
39     \msg_fatal:nn { aufgabentitel } { kein-titel }
40   }
41   {
42   }
43
44   \_gib_examen_titel: {}
45
46   \section{\_gib_aufgaben_titel:}
47
48   \bigskip
49 }
```

```
50 \ExplSyntaxOff
```

```
51
```

1.3 automaten.sty

```
52 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
53 \ProvidesPackage{lehramt-informatik-automaten}[2021/02/14 Zum Setzen von Automaten]
```

1.3.1 Endlicher Automat

```
54 \liLadePakete{formale-sprachen}

\begin{tikzpicture}[li automat]
\node[state,initial,accepting] (0) {$z_0$};
\node[state,right of=0] (1) {$z_1$};

\path (0) edge[above] node{1} (1);
\path (0) edge[loop,above] node{0} (0);
\path (1) edge[loop,above] node{0} (1);
\end{tikzpicture}
```



```
55 \RequirePackage{tikz}
56 \usetikzlibrary{arrows,automata,positioning}
57 \liLadePakete{mathe}
58 \directlua{
59   automaten = require('lehramt-informatik-automaten')
60 }
```

`\liAutomat` `\liAutomat[automaten-name]{zustaende=Z,alphabet=Σ,delta=δ,ende=E,start=z0}`

- `\liAutomat{}`: $A = (Z, \Sigma, \delta, E, z_0)$
- `\liAutomat[A_1]{}`: $A_1 = (Z, \Sigma, \delta, E, z_0)$
- `\liAutomat{zustaende={z_0, z_1, z_2}}`: $A = (\{z_0, z_1, z_2\}, \Sigma, \delta, E, z_0)$
- `\liAutomat{alphabet={a,b}}`: $A = (Z, \{a, b\}, \delta, E, z_0)$
- `\liAutomat{delta=d}`: $A = (Z, \Sigma, d, E, z_0)$
- `\liAutomat{ende={z_0, z_1, z_2}}`: $A = (Z, \Sigma, \delta, \{z_0, z_1, z_2\}, z_0)$
- `\liAutomat{start=z_1}`: $A = (Z, \Sigma, \delta, E, z_1)$
- `\liAutomat{dea}`: $A_{DEA} = (Z, \Sigma, \delta, E, z_0)$
- `\liAutomat{nea}`: $A_{NEA} = (Z, \Sigma, \delta, E, z_0)$

```
61 \ExplSyntaxOn
62 \NewDocumentCommand {\liAutomat} { 0{A} m } {
63   \tl_set:Nn \l_zustaende_tl {Z}
64   \tl_set:Nn \l_alphabet_tl {\Sigma}
65   \tl_set:Nn \l_delta_tl {\delta}
66   \tl_set:Nn \l_ende_tl {E}
67   \tl_set:Nn \l_start_tl {z\sb{0}}
68   \tl_set:Nn \l_typ_tl {}
69
70   \keys_define:nn { automat } {
71     zustaende .code:n = {\tl_set:Nn \l_zustaende_tl {\liMenge{##1}}},
72     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
73     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
74     ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
```

```

75     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
76     dea .value_forbidden:n = true,
77     dea .code:n = {\tl_set:Nn \l_typ_tl {\sb{\text{DEA}}}},
78     nea .value_forbidden:n = true,
79     nea .code:n = {\tl_set:Nn \l_typ_tl {\sb{\text{NEA}}}},
80   }
81
82   \keys_set:nn { automat } { #2 }
83
84   $#1 \l_typ_tl = (
85     \l_zustaende_tl,
86     \l_alphabet_tl,
87     \l_delta_tl,
88     \l_ende_tl,
89     \l_start_tl
90   )$
91 }
92 \ExplSyntaxOff

```

`\liAutomatenKante` **Let-Abkürzung:** `\let\k=\liAutomatenKante`

```

93 \def\liAutomatenKante#1#2#3#4{
94   \path (#1) edge[#4] node{#3} (#2);
95 }

96 \tikzset{
97   li automat/.style={
98     ->,
99     node distance=2cm
100   },
101 }

```

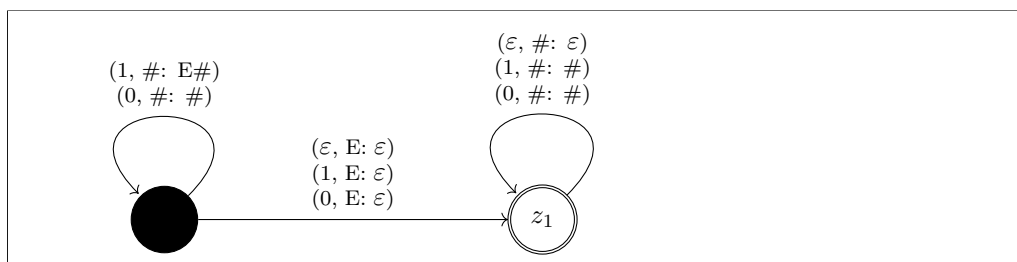
1.3.2 Kellerautomat

```
\begin{tikzpicture}[li kellerautomat,node distance=5cm]
\node[state,initial] (0) {\$z_0\$};
\node[state,right of=0,accepting] (1) {\$z_1\$};

\liKellerKante[above,loop]{0}{0}{
  1, KELLERBODEN, E KELLERBODEN;
  0, KELLERBODEN, KELLERBODEN;
}

\liKellerKante[above]{0}{1}{
  EPSILON, E, EPSILON;
  1, E, EPSILON;
  0, E, EPSILON;
}

\liKellerKante[above,loop]{1}{2}{
  EPSILON, KELLERBODEN, EPSILON;
  1, KELLERBODEN, KELLERBODEN;
  0, KELLERBODEN, KELLERBODEN;
}
\end{tikzpicture}
```



```
\liKellerAutomat \liKellerAutomat[\langle automaten-name \rangle]
{\zustaende=Z,alphabet=\Sigma,kelleralphabet=\Gamma,delta=\delta,start=z_0,kellerboden=\#,ende=E}

\liKellerAutomat{
  zustaende={z_0, z_1, z_2},
  alphabet={a, b, c},
  kelleralphabet={\#, A},
  ende={z_2},
}
```

$$K = (\{z_0, z_1, z_2\}, \{a, b, c\}, \{\#, A\}, \delta, z_0, \#, \{z_2\})$$

```
102 \ExplSyntaxOn
103 \NewDocumentCommand {\liKellerAutomat} { O{K} m } {
104   \tl_set:Nn \l_zustaende_tl {Z}
105   \tl_set:Nn \l_alphabet_tl {\Sigma}
106   \tl_set:Nn \l_kelleralphabet_tl {\Gamma}
107   \tl_set:Nn \l_delta_tl {\delta}
108   \tl_set:Nn \l_start_tl {z\sb{0}}
109   \tl_set:Nn \l_kellerboden_tl {\#}
110   \tl_set:Nn \l_ende_tl {E}
111
112   \keys_define:nn { kellerautomat } {
113     zustaende .code:n = {\tl_set:Nn \l_zustaende_tl {\liMenge{##1}}},
114     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
115     kelleralphabet .code:n = {\tl_set:Nn \l_kelleralphabet_tl {\liMenge{##1}}},
116     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
117     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
```



```

118     kellerboden .code:n = {\tl_set:Nn \l_kellerboden_tl {##1}},
119     ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
120 }
121
122 \keys_set:nn { kellerautomat } { #2 }
123
124 $#1 = (
125     \l_zustaende_tl,
126     \l_alphabet_tl,
127     \l_kelleralphabet_tl,
128     \l_delta_tl,
129     \l_start_tl,
130     \l_kellerboden_tl,
131     \l_ende_tl
132 )$
133 }
134 \ExplSyntaxOff

```

\liKellerUebergang **Let-Abkürzung:** `\let\u=\liKellerUebergang`
`\liKellerUebergang{a, KELLERBODEN, A; b KELLERBODEN, EPSILON}`

(a, #: A)
(b, #: ε)

Regulärer Ausdruck zum Konvertieren: `\((.*)`, `(.*)`, `(.*)\)` `\u{$1 $2 $3}`

```

135 \ExplSyntaxOn
136 \def\liKellerUebergang#1{
137     \directlua{automaten.drucke_keller_uebergaenge('#1')}
138 }
139 \ExplSyntaxOff

```

\liKellerKante `\liKellerKante[\tikz-optionen]{\von}{\zu}{\übergange}`
Let-Abkürzung: `\let\k=\liKellerKante`

```

140 \NewDocumentCommand{\liKellerKante} { 0{above} m m m } {
141     \path (#2) edge[#1] node{\liKellerUebergang{#4}} (#3);
142 }
143 \tikzset{
144     li keller knoten/.style={
145         text width=2cm,
146         align=center,
147         font=\footnotesize,
148     },
149     li kellerautomat/.style={
150         li automat,
151         every edge/.append style={
152             every node/.style={
153                 li keller knoten
154             }
155         }
156     }
157 }

```

1.3.3 Turingmaschine

```
158 \RequirePackage{amssymb}
```

\liTuringLeerzeichen

□

```
159 \def\liTuringLeerzeichen{\Box}
```

```

\liTuringMaschine \liTuringMaschine[\langle automaten-name \rangle]
{\zustaende=Z,alphabet= $\Sigma$ ,bandalphabet= $\Gamma$ ,delta= $\delta$ ,start= $z_0$ ,leerzeichen= $\square$ ,ende=E)}

\liTuringMaschine{
  zustaende={z_0, z_1, z_2},
  alphabet={a, b, c},
  bandalphabet={\liTuringLeerzeichen, A},
  ende={z_2},
}

```

$TM = (\{z_0, z_1, z_2\}, \{a, b, c\}, \{\square, A\}, \delta, z_0, \square, \{z_2\})$
--

```

160 \ExplSyntaxOn
161 \NewDocumentCommand {\liTuringMaschine} { O{TM} m } {
162   \tl_set:Nn \l_zustaende_tl {Z}
163   \tl_set:Nn \l_alphabet_tl {\Sigma}
164   \tl_set:Nn \l_bandalphabet_tl {\Gamma}
165   \tl_set:Nn \l_delta_tl {\delta}
166   \tl_set:Nn \l_start_tl {z\sb{0}}
167   \tl_set:Nn \l_leerzeichen_tl {\liTuringLeerzeichen}
168   \tl_set:Nn \l_ende_tl {E}
169
170   \keys_define:nn { kellerautomat } {
171     zustaende .code:n = {\tl_set:Nn \l_zustaende_tl {\liMenge{##1}}},
172     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
173     bandalphabet .code:n = {\tl_set:Nn \l_bandalphabet_tl {\liMenge{##1}}},
174     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
175     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
176     leerzeichen .code:n = {\tl_set:Nn \l_leerzeichen_tl {##1}},
177     ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
178   }
179
180   \keys_set:nn { kellerautomat } { #2 }
181
182   $\text{#1} = (
183     \l_zustaende_tl,
184     \l_alphabet_tl,
185     \l_bandalphabet_tl,
186     \l_delta_tl,
187     \l_start_tl,
188     \l_leerzeichen_tl,
189     \l_ende_tl
190   )$
191 }
192 \ExplSyntaxOff

```

\liTuringUebergangZelle Formatiert einen Zustandsübergang für eine Übergangstabelle.

Let-Abkürzung: `\let\t=\liTuringUebergangZelle`

`\liTuringUebergangZelle{z_1, LEER, R}:(z_1: \square , R) \liTuringUebergangZelle{z1, leer, l}:`
`(z1: \square , L)`

```

193 \ExplSyntaxOn
194 \def\liTuringUebergangZelle#1{
195   \directlua{tex.print(automaten.gib_einen_turing_uebergang('#1'))}
196 }
197 \ExplSyntaxOff

```

\liTuringUebergaenge Nur in den TikZ-Grafiken zu verwenden. Setzt Zeilenumbrüche ans Ende. Nicht für die Tabelle geeignet.

Let-Abkürzung: `\let\t=\liTuringUebergaenge`

`(z1: \square , L)`

`(\square : \square , R)`

```

198 \ExplSyntaxOn
199 \def\liTuringUebergaenge#1{
200   \directlua{automaten.drucke_turing_uebergaenge('#1')}
201 }
202 \ExplSyntaxOff

\liTuringKante \liTuringKante[\tikz-optionen]{\langle zustand-oder-lese \rangle}{\langle schreibe \rangle}{\langle richtung \rangle}
Let-Abkürzung: \let\t=\liTuringKante

203 \NewDocumentCommand{\liTuringKante} { 0{above} m m m } {
204   \path (#2) edge[#1] node{\liTuringUebergaenge{#4}} (#3);
205 }

\liTuringUeberfuehrung

206 \def\liTuringUeberfuehrung{
207    $\delta : Z \times \Gamma \rightarrow Z \times \Gamma \times \{ L, R, N \}$ 
208 }

209 \tikzset{
210   li turingmaschine/.style={
211     li automat,
212     every edge/.append style={
213       every node/.style={
214         li keller knoten
215       }
216     }
217   }
218 }
219

```

1.4 basis.sty

```
220 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
221 \ProvidesPackage{lehramt-informatik-basis}[2020/11/27]

222 \RequirePackage{xparse}

223 \ExplSyntaxOn

\liLadePakete

224 \NewDocumentCommand{\liLadePakete}{ m }
225 {
226   \clist_map_inline:nn { #1 } { \RequirePackage{lehramt-informatik-##1} }
227 }

\liLadeAllePakete

228 \def\liLadeAllePakete{
229   \liLadePakete{
230     aufgaben-einbinden,
231     automaten,
232     baum,
233     checkbox,
234     chomsky-normalform,
235     cpm,
236     cyk-algorithmus,
237     entwurfsmuster,
238     er,
239     formale-sprachen,
240     gantt,
241     grafik,
242     graph,
243     hanoi,
244     kontrollflussgraph,
245     makros,
246     master-theorem,
247     mathe,
248     minimierung,
249     normalformen,
250     petri,
251     potenzmengen-konstruktion,
252     pseudo,
253     quicksort,
254     relationale-algebra,
255     rmodell,
256     sortieren,
257     spalten,
258     struktogramm,
259     syntax,
260     syntaxbaum,
261     synthese-algorithmus,
262     tabelle,
263     typographie,
264     uml,
265     vollstaendige-induktion,
266     wasserfall,
267     wpkalkuel,
268   }
269 }
```

Definition einer Komma-getrennten-Liste mit deren Hilfe die vielen globalen Token-List-Variablen definiert werden können.

```
270 \clist_new:N \g_auf_schluesel_clist
271 \clist_set:Nn \g_auf_schluesel_clist {
272   titel,
273   thematik,
```

```

274 fussnote,
275 fussnote_seite,
276 relativer_pfad,
277 examen_nummer,
278 examen_jahr,
279 examen_monat,
280 examen_thema_nr,
281 examen_teilaufgabe_nr,
282 examen_aufgabe_nr,
283 }

```

Initialisierung der globalen Token-List-Variablen `\g_auf_***_tl`. `auf` steht für Aufgabe.

```

284 \clist_map_inline:Nn \g_auf_schluesssel_clist {
285   \tl_new:c {g_auf_#1_tl}
286 }

```

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

```

287 \cs_new:Npn \setze_variablen_zurueck: {
288   \clist_map_inline:Nn \g_auf_schluesssel_clist {
289     \tl_clear:c {g_auf_##1_tl}
290   }
291 }

```

```

292 \keys_define:nn { aufgabentitel }
293 {
294   Titel .tl_gset:N = \g_auf_titel_tl,
295   Thematik .tl_gset:N = \g_auf_thematik_tl,
296   Fussnote .tl_gset:N = \g_auf_fussnote_tl,
297   FussnoteSeite .tl_gset:N = \g_auf_fussnote_seite_tl,
298   RelativerPfad .tl_gset:N = \g_auf_relativer_pfad_tl,
299   ExamenNummer .tl_gset:N = \g_auf_examen_nummer_tl,
300   ExamenJahr .tl_gset:N = \g_auf_examen_jahr_tl,
301   ExamenMonat .tl_gset:N = \g_auf_examen_monat_tl,
302   ExamenThemaNr .tl_gset:N = \g_auf_examen_thema_nr_tl,
303   ExamenTeilaufgabeNr .tl_gset:N = \g_auf_examen_teilaufgabe_nr_tl,
304   ExamenAufgabeNr .tl_gset:N = \g_auf_examen_aufgabe_nr_tl,
305 }

```

```

306 \cs_gset:Npn \setze_relativen_pfad: {
307   \tl_if_empty:NTF \g_auf_relativer_pfad_tl
308   {
309     \bool_if:nTF
310     {
311       ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
312       ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
313       ! \tl_if_empty_p:N \g_auf_examen_monat_tl
314     }
315     {
316       \tl_gset:Nn \g_auf_relativer_pfad_tl {
317         Staatsexamen /
318         \g_auf_examen_nummer_tl /
319         \g_auf_examen_jahr_tl /
320         \g_auf_examen_monat_tl /
321         \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} { Thema - \g_auf_examen_thema_nr_tl / }
322         \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} { Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl / }
323         \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} { Aufgabe - \g_auf_examen_aufgabe_nr_tl / }
324       }
325     }
326   } {}
327 }
328 {}
329 }

```

```

330 \cs_gset:Npn \gib_examen_titel: {
331   \cs_set:Nn \trenner: { \, / \, }

```

```

332 \bool_if:nTF
333 {
334   ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
335   ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
336   ! \tl_if_empty_p:N \g_auf_examen_monat_tl &&
337   ! \tl_if_empty_p:N \g_auf_examen_aufgabe_nr_tl
338 }
339 {
340   {
341     \footnotesize
342     \par
343     \noindent
344     Staatsexamen ~
345     \g_auf_examen_nummer_tl \_trenner:
346     \g_auf_examen_jahr_tl \_trenner:
347
348     \tl_case:Nn \g_auf_examen_monat_tl
349     {
350       { 03 } { Frühjahr }
351       { 09 } { Herbst }
352     } \_trenner:
353
354     \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
355       Thema ~ Nr. ~ \g_auf_examen_thema_nr_tl \_trenner:
356     }
357     \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
358       Teilaufgabe ~ Nr. ~ \g_auf_examen_teilaufgabe_nr_tl \_trenner:
359     }
360     \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} {
361       Aufgabe ~ Nr. ~ \g_auf_examen_aufgabe_nr_tl
362     }
363     \par
364     \bigskip
365   }
366 }
367 }

368 \cs_new:Npn \_gib_github_url: {
369   \tl_if_empty:NTF \g_auf_relativer_pfad_tl {} {
370     Github :~\href{
371       \LehramtInformatikGithubRawDomain /
372       \LehramtInformatikGithubTexRepo /
373       \LehramtInformatikGitBranch /
374       \g_auf_relativer_pfad_tl
375     }{
376       \g_auf_relativer_pfad_tl
377     }
378   }
379 }

380 \cs_new:Npn \_gib_aufgaben_titel: {
381   \g_auf_titel_tl
382
383   \tl_if_empty:NTF \g_auf_thematik_tl
384   {}
385   {
386     \, ~ [
387       \g_auf_thematik_tl
388     ]
389   }
390 }

391 \msg_new:nnn { aufgabentitel } { kein-titel }
392 { Der~Schlüssel~ist~zwingend~notwendig }
393 \ExplSyntaxOff

```

```

\def\LehramtInformatikRepository{/pfad/zum/repository}
394 \input /etc/lehramt-informatik.config.tex
    biblatex not working with lualatex and babel
395 % \RequirePackage{polyglossia}
396 % \setmainlanguage{german}
397
```

1.5 baum.sty

```
398 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
399 \ProvidesPackage{lehramt-informatik-baum}[2020/06/13 Zum Setzen von
400 Binär- und AVL-Bäumen. Hüll-Paket um TikZ and tikz-qtrees.]

401 \RequirePackage{tikz}

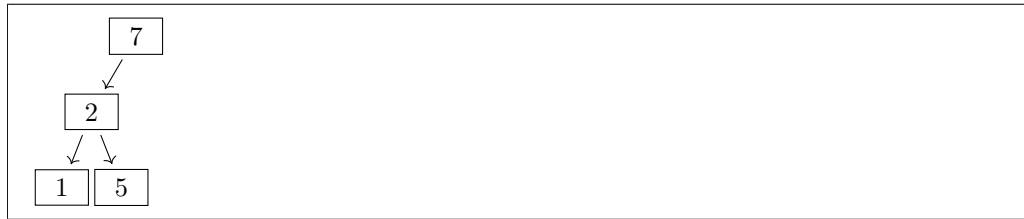
    für li binaer baum
402 \RequirePackage{tikz-qtrees}

    Für b baum
403 \usetikzlibrary{shapes.multipart}
```


1.5.1 Binärbaum

TikZ-Stil: li binaer baum: Knoten als Kreise, Kanten als Pfeile.

```
\begin{tikzpicture}[li binaer baum]
\Tree
[.7
  [.2
    [.1 ]
    [.5 ]
  ]
\edge[blank]; \node[blank]{};
]
\end{tikzpicture}
```



```
404 \tikzset{
405   li binaer baum/.style={
406     shorten <=2pt,
407     shorten >=2pt,
408     ->,
409     every tree node/.style={
410       minimum width=2em,
411       draw,
412       rectangle
413     },
414     blank/.style={
415       draw=none
416     },
417     edge from parent/.style={
418       draw,
419       edge from parent path={(\tikzparentnode) -- (\tikzchildnode)}
420     },
421     level distance=1cm,
422     every label/.style={
423       gray,
424       font=\footnotesize,
425       label position=0,
426       label distance=0cm,
427     }
428   },
429 }
```

1.5.2 AVL-Baum

```
\begin{tikzpicture}[li binaer baum]
\Tree
[.\node[label=-1]{5};
  [.\node[label=-1]{2};
    [.\node[label=0]{1}; ]
    \edge[blank]; \node[blank]{};
  ]
  [.\node[label=0]{7}; ]
]
\end{tikzpicture}
```

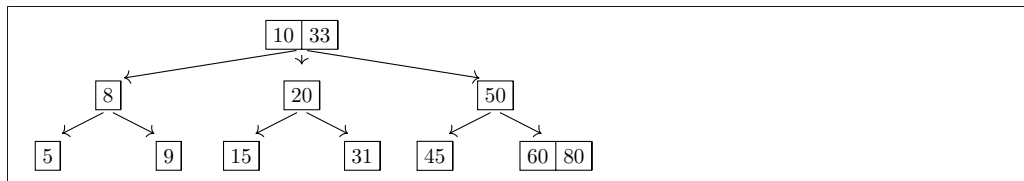


1.5.3 B-Baum

```

\begin{tikzpicture}[
  scale=0.8,
  transform shape,
  li bbaum,
  level 1/.style={level distance=10mm,sibling distance=32mm},
  level 2/.style={level distance=10mm,sibling distance=20mm},
]
\node {10 \nodepart{two} 33}
  child {node {8}
    child {node {5}}
    child {node {9}}
  }
  child {node {20}
    child {node {15}}
    child {node {31}}
  }
  child {node {50}
    child {node {45}}
    child {node {60 \nodepart{two} 80}}
  }
;
\end{tikzpicture}

```



```

430 \tikzset{
431   li bbaum knoten/.style={
432     rectangle split parts=10,
433     rectangle split,
434     rectangle split horizontal,
435     rectangle split ignore empty parts,
436     draw,
437     fill=white
438   },
439   li bbaum/.style={
440     every node/.style={
441       li bbaum knoten
442     },
443     level 1/.style={
444       level distance=12mm,
445       sibling distance=25mm,
446     },
447     every child/.style={
448       shorten <= 2pt,
449       shorten >= 6pt,
450       ->,
451     },
452     level 2/.style={
453       level distance=9mm,
454       sibling distance=15mm,
455     },
456   }
457 }
458

```

1.6 checkbox.sty

```
459 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
460 \ProvidesPackage{lehramt-informatik-checkbox}[2020/12/14 Zum Setzen von
461 Multiple-Choice-Fragen. Simulation von Kästchen zum ankreuzen]
462 \RequirePackage{amssymb}

\liRichtig Angekreuztes Kästchen (nur innerhalb der itemize-Umgebung zu verwenden.
463 \def\liRichtig{\item[$\boxtimes$]}

\liFalsch Nicht angekreuztes Kästchen (nur innerhalb der itemize-Umgebung zu verwenden.
464 \def\liFalsch{\item[$\square$]}

465
```

1.7 chomsky-normalform.sty

```

466 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
467 \ProvidesPackage{lehramt-informatik-chomsky-normalform}[2021/03/26
468 Hilfsmakros zum Setzen des Algorithmus zum Erreichen der Chomsky-Normalform]
469
470 \ExplSyntaxOn
471
472 \liLadePakete{typographie}

\liChomskyUeberschrift \let\Abkürzung:\let\schritt=\liChomskyUeberschrift

473 \def\liChomskyUeberschrift#1{
474   {
475     \bfseries
476     \sffamily
477     \str_case:nn {#1} {
478       {1} {Elimination~der~ $\varepsilon$ -Regeln}
479       {2} {Elimination~von~Kettenregeln}
480       {3} {Separation~von~Terminalzeichen}
481       {4} {Elimination~von~mehrelementigen~Nonterminalketten}
482     }
483   }
484 }

\liChomskyErklaerung \let\Abkürzung:\let\erklaerung=\liChomskyErklaerung
Hoffmann Seite 180

485 \def\li@chomsky@erklaerung@texte#1{
486   \str_case:nn {#1} {
487     %
488     {1} {
489       Alle~Regeln~der~Form~ $A \rightarrow \varepsilon$ ~werden~eliminiert.~
490       Die~Ersetzung~von~ $A$ ~wird~durch~ $\varepsilon$ ~in~allen~anderen~
491       Regeln~vorweggenommen.
492     }
493     {2} {
494       Jede~Produktion~der~Form~ $A \rightarrow B$ ~mit~ $A, B \in S$ ~wird~
495       als~Kettenregel~bezeichnet.~Diese~tragen~nicht~zur~Produktion~
496       von~Terminalzeichen~bei~und~lassen~sich~ebenfalls~eliminieren.
497     }
498     {3} {
499       Jedes~Terminalzeichen~ $\sigma$ ,~das~in~Kombination~mit~anderen~
500       Symbolen~auftaucht,~wird~durch~ein~neues~Nonterminal~
501        $S_{\sigma}$ ~ersetzt~und~die~Menge~der~Produktionen~durch~die~
502       Regel~ $S_{\sigma} \rightarrow \sigma$ ~ergänzt.
503     }
504     {4} {
505       Alle~Produktionen~der~Form~
506        $A \rightarrow B_{\{1\}} B_{\{2\}} \dots B_{\{n\}}$ ~
507       werden~in~die~Produktionen~
508        $A \rightarrow$ 
509        $A_{\{n-1\}} B_{\{n\}}, A_{\{n-1\}} \rightarrow$ 
510        $A_{\{n-2\}} B_{\{n-1\}}, \dots,$ 
511        $A_{\{2\}} \rightarrow B_{\{1\}} B_{\{2\}}$ ~zerteilt.~
512       Nach~der~Ersetzung~sind~alle~längeren~Nonterminalketten~
513       vollständig~heruntergebrochen~und~die~Chomsky-Normalform~erreicht.
514     }
515   }
516 }

517 \def\liChomskyErklaerung#1{
518   {
519     \itshape
520     \footnotesize
521     \liParagraphMitLinien{\li@chomsky@erklaerung@texte{#1}}
522   }

```

523 }

\liChomskyUeberErklaerung **Let-Abkürzung:** \let\schrittE=\liChomskyUeberErklaerung

524 \def\liChomskyUeberErklaerung#1{

525 \liChomskyUeberschrift{#1}\par

526 \liChomskyErklaerung{#1}

527 }

528 \ExplSyntaxOff

529

1.8 cpm.sty

```

530 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
531 \ProvidesPackage{lehramt-informatik-cpm}[2020/09/03]
532 \RequirePackage{tikz}
533 \liLadePakete{mathe}

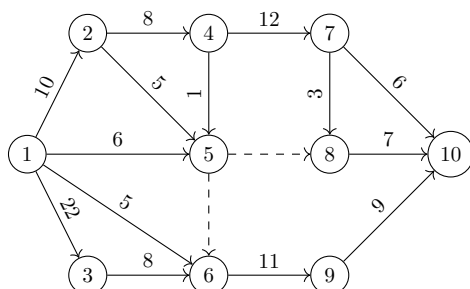
```

Let-Abkürzungen

```

\let\f=\footnotesize
\let\FZ=\liCpmFruehesterI
\let\SZ=\liCpmSpaetesterI
\let\v=\liCpmVon
\let\ vz=\liCpmVonZu
\let\z=\liCpmZu

```



```

\liCpmEreignis liCpmEreignis\{(.*)\}\{((.*),(.*))\} -> liCpmEreignis{$1}{$2}{$3}

```

```

534 \ExplSyntaxOn
535 \NewDocumentCommand { \liCpmEreignis } { 0{ } m m m } {
536   \tl_set:Nn \l_name_tl {}
537
538   \keys_define:nn { cpmEreignis } {
539     name .code:n = {\tl_set:Nn \l_name_tl {##1}},
540   }
541
542   \keys_set:nn { cpmEreignis } { #1 }
543
544   \tl_if_empty:NT \l_name_tl {
545     \tl_set:Nn \l_name_tl {#2}
546   }
547
548   \node[circle,draw] (\l_name_tl) at (#3,#4) {#2};
549 }
550 \ExplSyntaxOff

```

```

\liCpmVorgang liCpmVORGANG\{((.*)>(.*))\}\{(.*)\}

```

```

551 \ExplSyntaxOn
552 \NewDocumentCommand { \liCpmVorgang } { 0{ } m m m } {
553   \tl_set:Nn \l_schein_tl {}
554   \tl_set:Nn \l_kritisch_tl {}
555
556   \keys_define:nn { cpmVorgang } {
557     schein .code:n = {\tl_set:Nn \l_schein_tl {dashed}},
558     kritisch .code:n = {\tl_set:Nn \l_kritisch_tl {thick}},
559   }
560
561   \keys_set:nn { cpmVorgang } { #1 }
562
563   \path[->,\l_schein_tl,\l_kritisch_tl] (#2) edge node[auto,sloped] {#4} (#3);
564 }
565 \ExplSyntaxOff

```

1.8.1 Hilfsmakros für Zeitpunkt-Tabelle

```
\begin{tabular}{|1||1|1|1|1|1|1|1|1|1|}
\hline
$i$ & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & \\ \hline
\FZ & 0 & 5 & 18 & 7 & 19 & 26 & 22 & 30 & \\ \hline
\SZ & 0 & 5 & 18 & 9 & 19 & 26 & 24 & 30 & \\ \hline
GP & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 2 & 0 & \\ \hline
\end{tabular}
```

```
\liCpmVonZu Let-Abkürzung: \let\ vz=\liCpmVonZu
\liCpmVonZu{1}(2-3):  $1_{(2 \rightarrow 3)}$ 
566 \def\liCpmVonZuOhneMathe#1(#2-#3){#1_{\scriptscriptstyle(#2\rightarrow#3)}}
567 \def\liCpmVonZu#1(#2-#3){%
568 \ifmmode%
569 \liCpmVonZuOhneMathe{#1}(#2-#3)%
570 \else%
571 $\liCpmVonZuOhneMathe{#1}(#2-#3)$%
572 \fi%
573 }
```

```
\liCpmVon Let-Abkürzung: \let\ v=\liCpmVon
\liCpmVon{1}(2):  $1_{(\rightarrow 2)}$ 
574 \def\liCpmVonOhneMathe#1(#2){#1_{\scriptscriptstyle(\rightarrow#2)}}
575 \def\liCpmVon#1(#2){%
576 \ifmmode%
577 \liCpmVonOhneMathe{#1}(#2)%
578 \else%
579 $\liCpmVonOhneMathe{#1}(#2)$%
580 \fi%
581 }
```

```
\liCpmZu Let-Abkürzung: \let\ z=\liCpmZu
\liCpmZu{1}(2):  $1_{(\leftarrow 2)}$ 
582 \def\liCpmZuOhneMathe#1(#2){#1_{\scriptscriptstyle(\leftarrow#2)}}
583 \def\liCpmZu#1(#2){%
584 \ifmmode%
585 \liCpmZuOhneMathe{#1}(#2)%
586 \else%
587 $\liCpmZuOhneMathe{#1}(#2)$%
588 \fi%
589 }
```

```
\liCpmSpaetesterI Spätester Zeitpunkt, zu dem Ereignis  $i$  eintreten kann
Let-Abkürzung: \let\ SZ=\liCpmSpaetesterI
590 \def\liCpmSpaetesterI{\$SZ_i\$}
```

```
\liCpmFruehesterI Frühester Zeitpunkt, zu dem Ereignis  $i$  eintreten kann.
Let-Abkürzung: \let\ FZ=\liCpmFruehesterI
591 \def\liCpmFruehesterI{\$FZ_i\$}
```

592

1.9 cyk-algorithmus.sty

```
593 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
594 \ProvidesPackage{lehramt-informatik-cyk-algorithmus}[2021/06/18 Hilfsmakros
595 zum Setzen des CYK-Algorithmus]
```

```
\begin{tabular}{|c|c|c|c|c|c|}
a      & c      & b      & c      & a      & b \\\hline\hline

$R_a$ & $R_c$ & $R_b$ & $R_c$ & $R_a$ & $R_b$ \l6
B      & A      & A      & B      & C \l5
S      & -      & S      & S \l4
-      & -      & - \l3
-      & - \l2
S \l1
\end{tabular}
```

```
\liKurzeTabellenLinie Let-Abkürzung: \let\l=\liKurzeTabellenLinie
```

```
596 \def\liKurzeTabellenLinie#1{\ccline{1-#1}}
```

```
\liWortInSprache  $\Rightarrow abc \in L(Y)$ 
```

```
597 \NewDocumentCommand{ \liWortInSprache } { m O{L(G)} } {
598   \bigskip
599   \noindent
600   $\Rightarrow$ #1 \in #2$
601 }
```

```
\liWortNichtInSprache  $\Rightarrow abc \notin L(G)$ 
```

```
602 \NewDocumentCommand{ \liWortNichtInSprache } { m O{L(G)} } {
603   \bigskip
604   \noindent
605   $\Rightarrow$ #1 \notin #2$
606 }
```

```
607
```

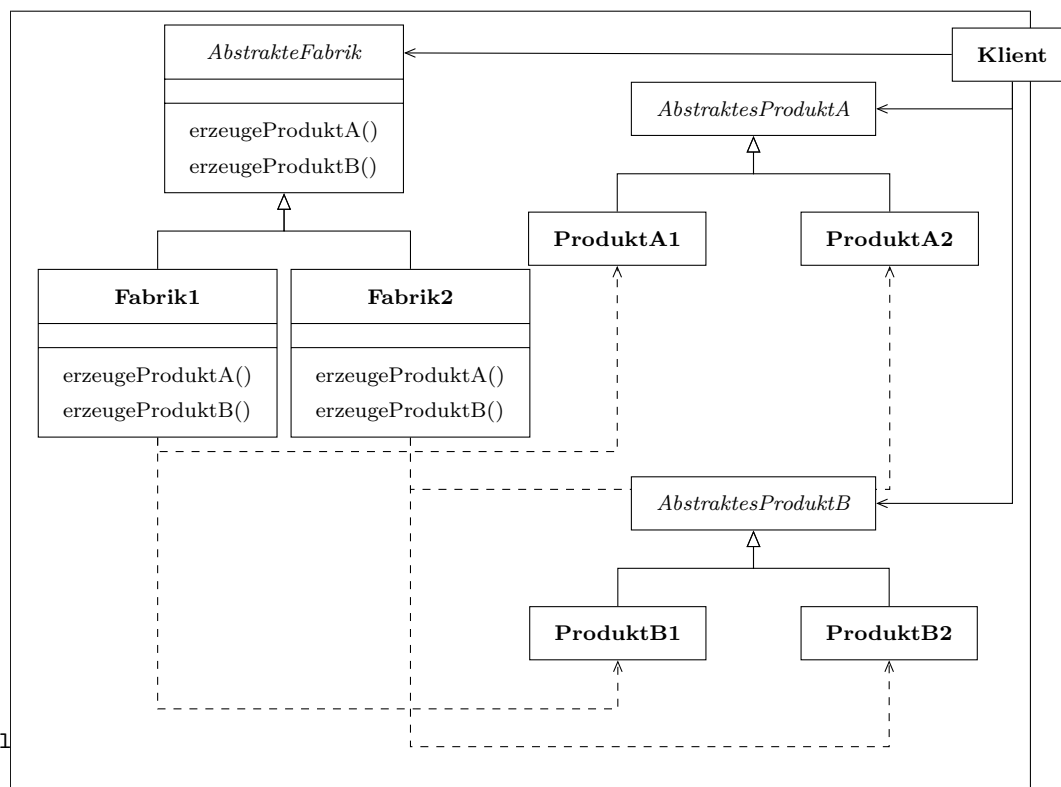
1.10 entwurfsmuster.sty

```

608 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
609 \ProvidesPackage{lehramt-informatik-entwurfsmuster}[2021/05/06 Hilfsmakros
610 zum Setzen von Entwurfsmuster/Design Patterns]
611
612 \RequirePackage{lehramt-informatik-uml}

```

1.10.1 Abstrakte Fabrik



```

613 \def\liEntwurfsAbstrakteFabrikUml{
614   \begin{tikzpicture}
615     \umlclass[type=abstract]{AbstrakteFabrik}{-}{
616       erzeugeProduktA()\
617       erzeugeProduktB()\
618     }
619     \umlclass[below left=1cm and -1.5cm of AbstrakteFabrik]{Fabrik1}{-}{
620       erzeugeProduktA()\
621       erzeugeProduktB()\
622     }
623     \umlclass[below right=1cm and -1.5cm of AbstrakteFabrik]{Fabrik2}{-}{
624       erzeugeProduktA()\
625       erzeugeProduktB()\
626     }
627     \umlVHVinherit{Fabrik1}{AbstrakteFabrik}
628     \umlVHVinherit{Fabrik2}{AbstrakteFabrik}
629
630     \umlsimpleclass[right=3cm of AbstrakteFabrik,type=abstract]{AbstraktesProduktA}
631     \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktA]{ProduktA1}
632     \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktA]{ProduktA2}
633     \umlVHVinherit{ProduktA1}{AbstraktesProduktA}
634     \umlVHVinherit{ProduktA2}{AbstraktesProduktA}
635
636     \umlsimpleclass[above right=0cm and 1cm of AbstraktesProduktA]{Klient}
637
638     \umlsimpleclass[below=4.5cm of AbstraktesProduktA,type=abstract]{AbstraktesProduktB}
639     \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktB]{ProduktB1}
640     \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktB]{ProduktB2}

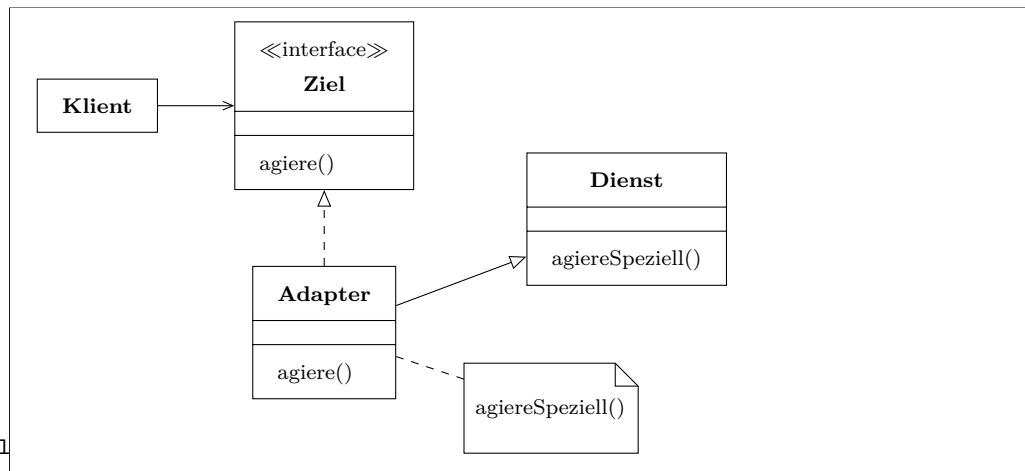
```

```

641 \umlVHVinherit{ProduktB1}{AbstraktesProduktB}
642 \umlVHVinherit{ProduktB2}{AbstraktesProduktB}
643
644 \umlVHVdep[arm1=-1.3cm,arm2=-1cm]{Fabrik1}{ProduktA1}
645 \umlVHVdep[arm2=-1cm]{Fabrik1}{ProduktB1}
646
647 \umlVHVdep[arm1=-1.8cm,arm2=-1.5cm]{Fabrik2}{ProduktA2}
648 \umlVHVdep[arm2=-1.5cm]{Fabrik2}{ProduktB2}
649
650 \umluniassoc[anchor2=25]{Klient}{AbstrakteFabrik}
651 \umlVHuniassoc[arm1=-1cm]{Klient}{AbstraktesProduktA}
652 \umlVHuniassoc{Klient}{AbstraktesProduktB}
653 \end{tikzpicture}
654 }

```

1.10.2 Adapter



\liEntwurfsAdapterUml

```

655 \def\liEntwurfsAdapterUml{
656 \begin{tikzpicture}
657 \umlsimpleclass[x=1,y=3]{Klient}{}{}
658 \umlclass[x=4,y=3,type=interface]{Ziel}{}{agiere()}
659 \umlclass[x=4,y=0]{Adapter}{}{agiere()}
660 \umlclass[x=8,y=1.5]{Dienst}{}{agiereSpeziell()}
661
662 \umlreal{Adapter}{Ziel}
663 \umluniassoc{Klient}{Ziel}
664 \umlinherit{Adapter}{Dienst}
665
666 \umlnote[x=7,y=-1,width=2cm]{Adapter}{agiereSpeziell()}
667 \end{tikzpicture}
668 \footcite[so ähnlich wie GoF]{wiki:adapter}
669 }

```

\liEntwurfsAdapterAkteure

Ziel (Target) Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.

Klient (Client) Der Klient nutzt Dienste über inkompatible Schnittstellen und greift dabei auf adaptierte Schnittstellen zurück.

Dienst (Adaptee) Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest definierter Schnittstelle an.

Adapter Der Adapter adaptiert die Schnittstelle des Dienstes auf die Schnittstelle zum Klienten.

```

670 \def\liEntwurfsAdapterAkteure{
671 \begin{description}
672
673 \item[Ziel (Target)]

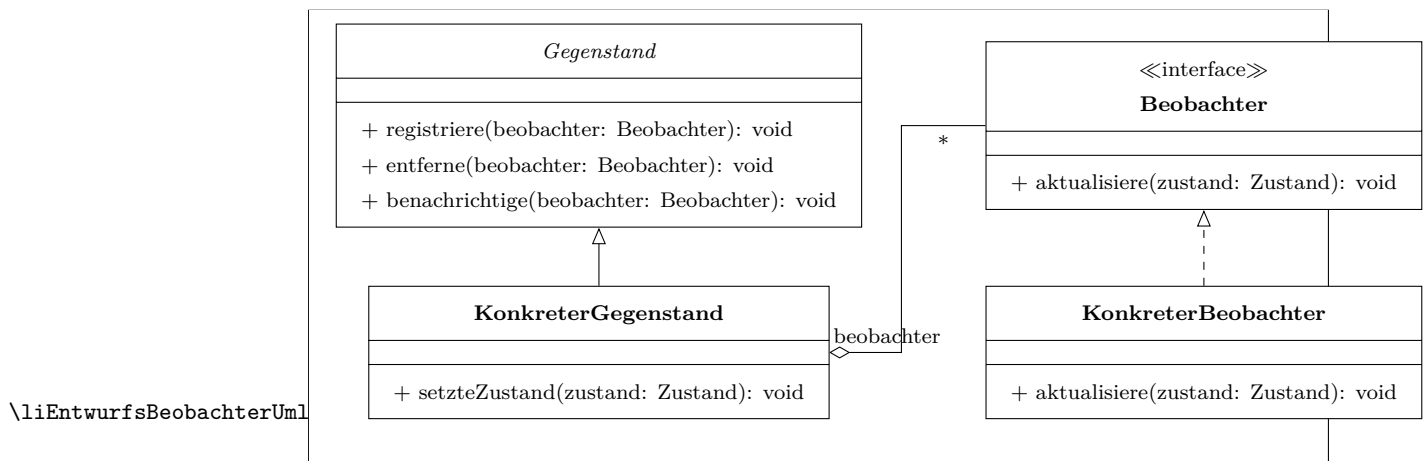
```

```

674
675 Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.
676
677 \item[Klient (Client)]
678
679 Der Klient nutzt Dienste über inkompatible Schnittstellen und greift
680 dabei auf adaptierte Schnittstellen zurück.
681
682 \item[Dienst (Adaptee)]
683
684 Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest
685 definierter Schnittstelle an.
686
687 \item[Adapter]
688
689 Der Adapter adaptiert die Schnittstelle des Dienstes auf die
690 Schnittstelle zum Klienten.\footcite{wiki:adapter}
691
692 \end{description}
693 }

```

1.10.3 Beobachter



```

694 \def\liEntwurfsBeobachterUml{
695   \begin{tikzpicture}
696     \umlclass[x=0,y=0,type=abstract]{Gegenstand}{-}{
697       + registriere(beobachter: Beobachter): void\\
698       + entferne(beobachter: Beobachter): void\\
699       + benachrichtige(beobachter: Beobachter): void\\
700     }
701     \umlclass[x=0,y=-3]{KonkreterGegenstand}{-}{
702       + setzteZustand(zustand: Zustand): void
703     }
704     \umlinherit{KonkreterGegenstand}{Gegenstand}
705
706     \umlclass[x=8,y=0,type=interface]{Beobachter}{-}{
707       + aktualisiere(zustand: Zustand): void
708     }
709     \umlclass[x=8,y=-3]{KonkreterBeobachter}{-}{
710       + aktualisiere(zustand: Zustand): void
711     }
712     \umlreal{KonkreterBeobachter}{Beobachter}
713
714     \umlHVHaggreg[arg1=beobachter,pos1=0.8,mult2=*,pos2=2.5]
715     {KonkreterGegenstand}{Beobachter}
716   \end{tikzpicture}
717 }

```

Gegenstand / Subjekt (Subject / Observable) Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An- und Abmeldung von Beobachtern und eine Schnittstelle zur Benachrichtigung von Beobachtern über Änderungen an.

Beobachter (Observer) Die Beobachter (auf Englisch auch subscriber, also „Abonnent“, genannt) definieren eine Aktualisierungsschnittstelle.

konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)

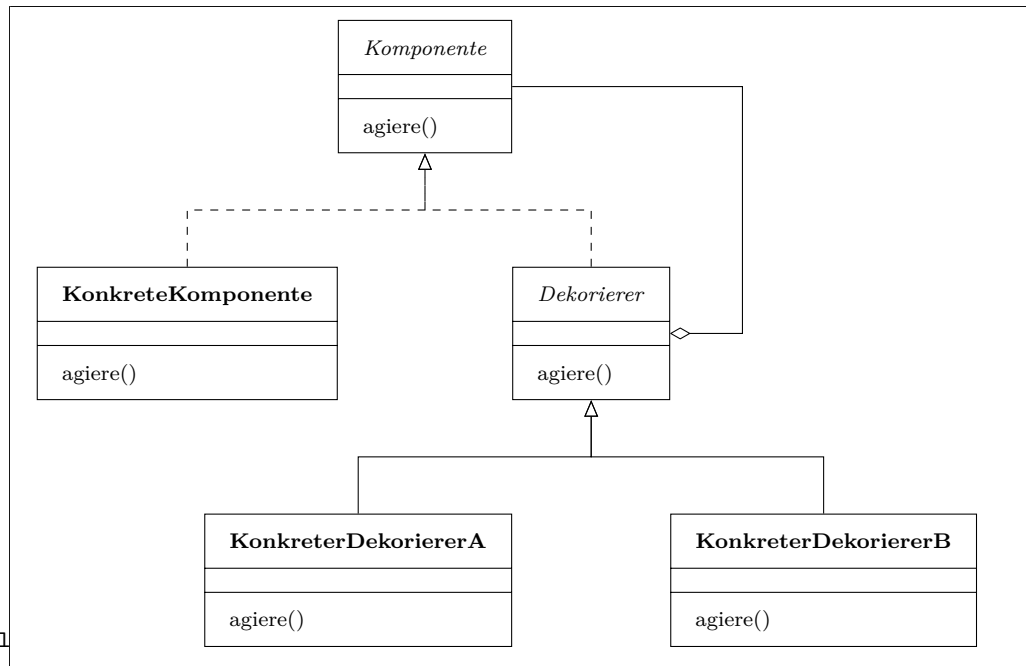
Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert den relevanten Zustand und benachrichtigt alle Beobachter bei Zustandsänderungen über deren Aktualisierungsschnittstelle. Es verfügt über eine Schnittstelle zur Erfragung des aktuellen Zustands.

Konkrete Beobachter (ConcreteObserver) Konkrete Beobachter verwalten die Referenz auf ein konkretes Subjekt, dessen Zustand sie beobachten und speichern und dessen Zustand konsistent ist. Sie implementieren eine Aktualisierungsschnittstelle unter Verwendung der Abfrageschnittstelle des konkreten Subjekts.

```

718 \def\liEntwurfsBeobachterAkteure{
719   \begin{description}
720     \item[Gegenstand / Subjekt (Subject / Observable)]
721
722     Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also
723     „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne
724     deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An-
725     und Abmeldung von Beobachtern und eine Schnittstelle zur
726     Benachrichtigung von Beobachtern über Änderungen an.\footcite[Seite
727     251]{gof}
728
729     \item[Beobachter (Observer)]
730
731     Die Beobachter (auf Englisch auch subscriber, also „Abonnent“,
732     genannt) definieren eine Aktualisierungsschnittstelle.
733
734     \item[konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)]
735
736     Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert
737     den relevanten Zustand und benachrichtigt alle Beobachter bei
738     Zustandsänderungen über deren Aktualisierungsschnittstelle. Es
739     verfügt über eine Schnittstelle zur Erfragung des aktuellen
740     Zustands.
741
742     \item[Konkrete Beobachter (ConcreteObserver)]
743
744     Konkrete Beobachter verwalten die Referenz auf ein konkretes
745     Subjekt, dessen Zustand sie beobachten und speichern und dessen
746     Zustand konsistent ist. Sie implementieren eine
747     Aktualisierungsschnittstelle unter Verwendung der
748     Abfrageschnittstelle des konkreten Subjekts.
749     \footcite{wiki:beobachter}
750   \end{description}
751 }
```

1.10.4 Dekorierer



\liEntwurfsDekoriererUml

```

752 \def\liEntwurfsDekoriererUml{
753   \begin{tikzpicture}
754     \umlclass[type=abstract]{Komponente}{-}{agiere()}
755     \umlclass[below left=1.5cm and 0cm of Komponente]{KonkreteKomponente}{-}{agiere()}
756     \umlclass[below right=1.5cm and 0cm of Komponente,type=abstract]{Dekorierer}{-}{agiere()}
757
758     \umlVHVreal{KonkreteKomponente}{Komponente}
759     \umlVHVreal{Dekorierer}{Komponente}
760
761     \umlclass[below left=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererA}{-}{agiere()}
762     \umlclass[below right=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererB}{-}{agiere()}
763
764     \umlVHVinherit{KonkreterDekoriererA}{Dekorierer}
765     \umlVHVinherit{KonkreterDekoriererB}{Dekorierer}
766
767     \umlHVVHaggreg[arm1=2cm]{Dekorierer}{Komponente}
768     \footcite{wiki:dekorierer}
769   \end{tikzpicture}
770 }
  
```

1.10.5 Einzelstück

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```

771 \def\liEntwurfsEinzelstueckUml{
772   \begin{tikzpicture}
773     \umlclass{Einzelstück}{
774       \umlstatic{- instanz: Einzelstück}\\
775     }{
776       - Einzelstück()\\
777       + gibInstanz(): Einzelstück
778     }
779   \end{tikzpicture}
  
```

780 }

iEntwurfsEinzelstueckAkteure Quelle: frei überstetzt aus GoF

Einzelstück (Singleton) stellt eine statische Methode bereit, mit deren Hilfe die Klienten nur auf eine einzige Instanz der Klasse zugreifen können.

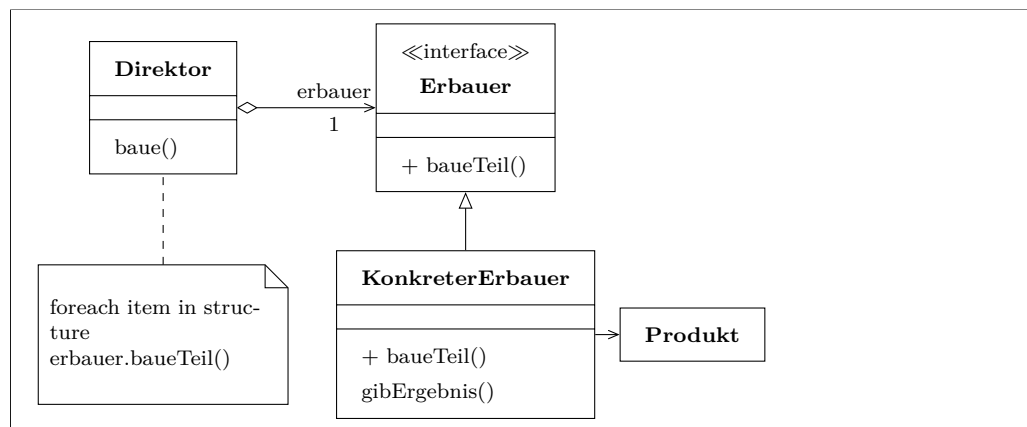
```

781 \def\liEntwurfsEinzelstueckAkteure{
782   \begin{description}
783     \item[Einzelstück (Singleton)]
784
785     stellt eine statische Methode bereit, mit deren Hilfe die Klienten
786     nur auf eine einzige Instanz der Klasse zugreifen können.
787   \end{description}
788 }

```

1.10.6 Erbauer

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

789 \def\liEntwurfsErbauerUml{
790   \begin{tikzpicture}
791     \umlsimpleclass[x=7,y=0]{Produkt}{}{}
792     \umlclass[x=4,y=3,type=interface]{Erbauer}{}{+ baueTeil()}
793     \umlclass[x=4,y=0]{KonkreterErbauer}{}{
794       + baueTeil()\
795       gibErgebnis()
796     }
797     \umlclass[x=0,y=3]{Direktor}{}{baue()}
798     \umluniagg[arg2=erbauer,pos2=0.7,mult2=1]{Direktor}{Erbauer}
799     \umluniassoc{KonkreterErbauer}{Produkt}
800     \umlinherit{KonkreterErbauer}{Erbauer}
801
802     \umlnote[x=0,y=0,width=3cm]{Direktor}{
803       foreach item in structure\
804       erbauer.baueTeil()
805     }
806   \end{tikzpicture}
807   \footcite{wiki:erbauer}
808 }

```

\liEntwurfsErbauerAkteure Quelle: deutsche Wikipedia

Erbauer Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der Teile eines komplexen Objektes.

KonkreterErbauer Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch Implementierung der Schnittstelle. Außerdem definiert und verwaltet er die von ihm erzeugte Repräsentation des Produkts. Er bietet auch

eine Schnittstelle zum Auslesen des Produkts.

Direktor Der Direktor konstruiert ein komplexes Objekt unter Verwendung der Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom Klienten.

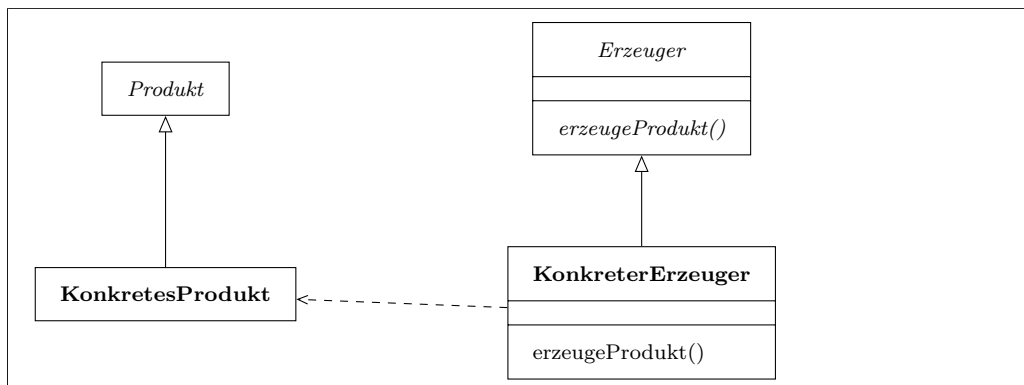
Produkt Das Produkt repräsentiert das zu konstruierende komplexe Objekt.

```

809 \def\liEntwurfsErbauerAkteure{
810   \begin{description}
811     \item[Erbauer]
812
813     Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der
814     Teile eines komplexen Objektes.
815
816     \item[KonkreterErbauer]
817
818     Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch
819     Implementierung der Schnittstelle. Außerdem definiert und verwaltet er
820     die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine
821     Schnittstelle zum Auslesen des Produkts.
822
823     \item[Direktor]
824
825     Der Direktor konstruiert ein komplexes Objekt unter Verwendung der
826     Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer
827     zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder
828     benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom
829     Klienten.
830
831     \item[Produkt]
832
833     Das Produkt repräsentiert das zu konstruierende komplexe Objekt.
834   \footcite{wiki:erbauer}
835 \end{description}
836 }
```

1.10.7 Fabrikmethode

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

837 \def\liEntwurfsFabrikmethodeUml{
838   \begin{tikzpicture}
839     \umlsimpleclass[type=abstract]{Produkt}
840     \umlsimpleclass[below=2cm of Produkt]{KonkretesProdukt}
841     \umlinherit{KonkretesProdukt}{Produkt}
842
843     \umlclass[type=abstract,right=4cm of Produkt]{Erzeuger}{
844       \textit{erzeugeProdukt()}\}
  
```

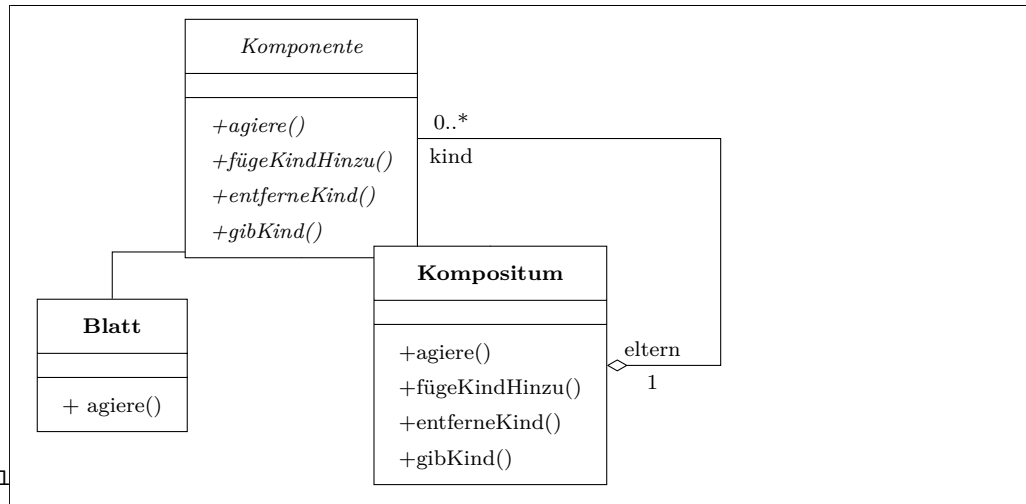


```

845     }
846     \umlclass[below=1.2cm of Erzeuger]{KonkreterErzeuger}{-}{
847     erzeugeProdukt()
848     }
849     \umlinherit{KonkreterErzeuger}{Erzeuger}
850
851     \umldep{KonkreterErzeuger}{KonkretesProdukt}
852 \end{tikzpicture}
853 }

```

1.10.8 Kompositum



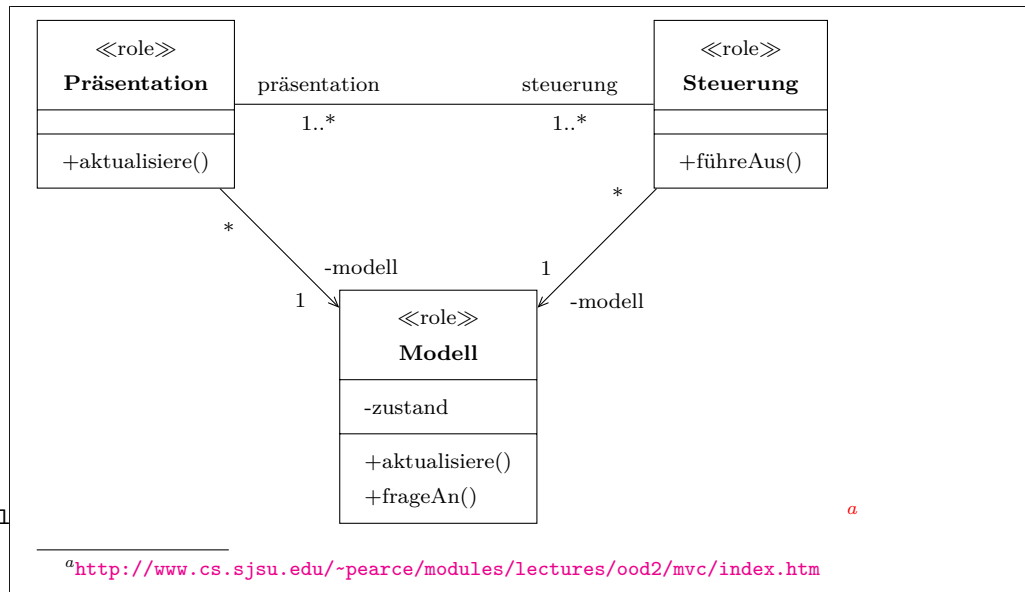
\liEntwurfsKompositumUml

```

854 \def\liEntwurfsKompositumUml{
855     \begin{tikzpicture}
856         \umlclass[x=2.5,y=3,type=abstract]{Komponente}{-}{
857             \textit{+agiere()}\
858             \textit{+fügeKindHinzu()}\
859             \textit{+entferneKind()}\
860             \textit{+gibKind()}
861         }
862         \umlclass[x=0]{Blatt}{-}{+ agiere()}
863         \umlclass[x=5]{Kompositum}{-}{
864             +agiere()\
865             +fügeKindHinzu()\
866             +entferneKind()\
867             +gibKind()
868         }
869
870         \umlVHVinherit{Kompositum}{Komponente}
871         \umlVHVinherit{Blatt}{Komponente}
872         \umlHVVaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,
873 \end{tikzpicture}
874 }

```

1.10.9 Modell-Präsentation-Steuerung



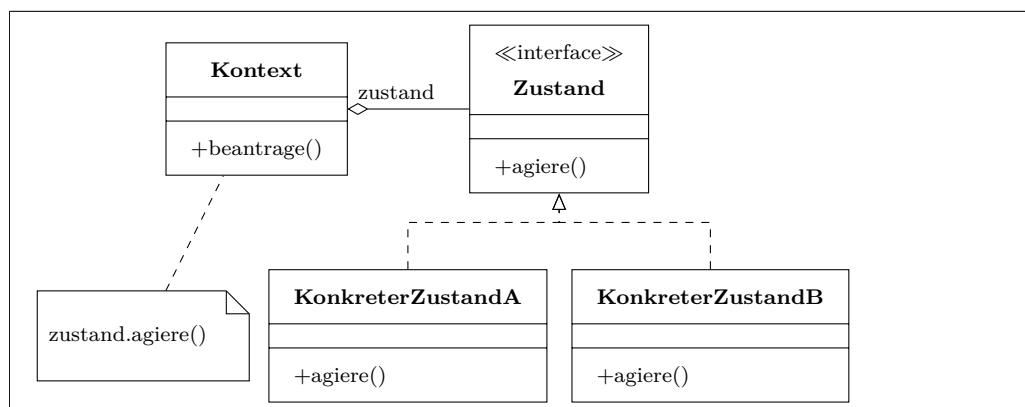
```

875 \def\liEntwurfsModellPraesentationSteuerungUml{
876   \begin{tikzpicture}
877     \umlclass[x=-4cm,type=role]{Präsentation}{+aktualisiere()}
878     \umlclass[x=4cm,type=role]{Steuerung}{+führeAus()}
879     \umlclass[y=-4cm,type=role]{Modell}{
880       -zustand
881     }{
882       +aktualisiere()\
883       +frageAn()
884     }
885
886     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
887     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}
888     \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
889   \end{tikzpicture}
890   \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
891 }

```

1.10.10 Zustand

\liEntwurfsZustandUml Quelle: Englische Wikipedia, so ähnlich wie in GoF



```

892 \def\liEntwurfsZustandUml{
893   \begin{tikzpicture}
894     \umlclass[x=-1,y=3]{Kontext}{+beantrage()}
895     \umlclass[x=3,y=3,type=interface]{Zustand}{+agiere()}
896     \umlclass[x=1,y=0]{KonkreterZustandA}{+agiere()}
897     \umlclass[x=5,y=0]{KonkreterZustandB}{+agiere()}
898

```

```

899 \umlVHVreal{KonkreterZustandA}{Zustand}
900 \umlVHVreal{KonkreterZustandB}{Zustand}
901
902 \umlaggreg[arg=zustand,pos=0.4]{Kontext}{Zustand}
903
904 \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
905 \end{tikzpicture}
906 }

```

\liEntwurfsZustandAkteure Quelle: Deutsche Wikipedia

Kontext (Context) definiert die clientseitige Schnittstelle und verwaltet die separaten Zustandsklassen.

State (Zustand) definiert eine einheitliche Schnittstelle aller Zustandsobjekte und implementiert gegebenenfalls ein Standardverhalten.

KontreterZustand (ConcreteState) implementiert das Verhalten, das mit dem Zustand des Kontextobjektes verbunden ist.

```

907 \def\liEntwurfsZustandAkteure{
908 \begin{description}
909 \item[Kontext (Context)]
910
911 definiert die clientseitige Schnittstelle und verwaltet die separaten
912 Zustandsklassen.
913
914 \item[State (Zustand)]
915
916 definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
917 implementiert gegebenenfalls ein Standardverhalten.
918
919 \item[KontreterZustand (ConcreteState)]
920
921 implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
922 verbunden ist.
923 \end{description}
924 }

```

925

1.11 er.sty

```
296 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
297 \ProvidesPackage{lehramt-informatik-er}[2020/06/13 Zum Setzen von
298 ER-Diagrammen]

299 \RequirePackage{tikz-er2}
300 \usetikzlibrary{positioning}

\begin{tikzpicture}[er2,scale=0.7,transform shape]
% Person
\node[entity] (Person) {Person};
\node[attribute,right=1cm of Person] {\key{E-Mail}} edge (Person);
\node[multi attribute,above left=1cm of Person] {Vornamen} edge (Person);
\node[attribute,left=1cm of Person] {Nachnamen} edge (Person);
\node[attribute,above right=1cm of Person] {Geburtsdatum} edge (Person);

% Kunde
\node[entity,below left=1cm of Person] (Kunde) {Kunde};

% Händler
\node[entity,below right=1cm of Person] (Händler) {Händler};

\node[specialization,below=0.2cm of Person]{is-a}
  edge (Kunde) edge (Händler) edge (Person);

% Transaktion
\node[relationship,below=2cm of Person] (Transaktion) {Transaktion}
  edge node[auto]{1} (Kunde)
  edge node[auto]{1} (Händler);
\node[attribute,below=1cm of Transaktion] {Preis} edge (Transaktion);
\node[attribute,left=1cm of Transaktion,text width=2cm] {Abschlussdatum} edge (Transaktion);
\node[attribute,right=1cm of Transaktion] {Lieferadresse} edge (Transaktion);

% Zahlungsmittel
\node[entity,below=4cm of Händler] (Zahlungsmittel) {Zahlungsmittel}
  edge node[auto]{1} (Transaktion);
\node[attribute,right=1cm of Zahlungsmittel] {Inhaber} edge (Zahlungsmittel);

% Bankverbindung
\node[entity,below left=1cm and 0cm of Zahlungsmittel] (Bankverbindung) {Bankverbindung};
\node[attribute,below left=1cm of Bankverbindung] {\key{IBAN}} edge (Bankverbindung);

% Kreditkarte
\node[entity,below right=3cm and -2cm of Zahlungsmittel]
(Kreditkarte) {Kreditkarte};

\node[attribute,below left=1cm of Kreditkarte]
{\key{Nummer}} edge (Kreditkarte);

\node[attribute,below right=1cm of Kreditkarte,text width=2cm]
{Ablaufdatum} edge (Kreditkarte);

\node[attribute,right=1cm of Kreditkarte]
{Anbieter} edge (Kreditkarte);

\node[generalization,below=0.8cm of Zahlungsmittel]{is-a}
  edge (Zahlungsmittel) edge (Bankverbindung) edge (Kreditkarte);

% Produkt
\node[entity,below=4cm of Kunde] (Produkt) {Produkt}
  edge node[auto]{1} (Transaktion);
\node[attribute,left=1cm of Produkt] {\key{Bezeichnung}} edge (Produkt);
\node[attribute,below left=1cm of Produkt,text width=2cm] {Beschreibungstext} edge (Produkt);
\node[attribute,above left=1cm of Produkt] {Bewertung} edge (Produkt);
```

```

\end{tikzpicture}

931 \RequirePackage{soul}
932 \RequirePackage{fontawesome}

Let-Abkürzungen
\let\a=\liErMpAttribute
\let\d=\liErDatenbankName
\let\e=\liErMpEntity
\let\r=\liErMpRelationship

933 \ExplSyntaxOn

\liErEntity
934 \def\liErEntity#1{\textbf{#1}}

\liErRelationship
935 \def\liErRelationship#1{\ul{#1}}

\liErAttribute
936 \def\liErAttribute#1{\emph{#1}}

\liErMpEntity mp = marginpar
Let-Abkürzung: \let\e=\liErMpEntity
937 \def\liErMpEntity#1{
938   \liErEntity{#1}
939   \marginpar{
940     \liErEntity{\tiny\faSquareO{}}~E:~#1}
941   }
942 }
□

\liErMpRelationship Let-Abkürzung: \let\r=\liErMpRelationship
943 \def\liErMpRelationship#1{
944   \liErRelationship{#1}
945   \marginpar{
946     \liErRelationship{\tiny\faGg{}}~R:~#1}
947   }
948 }

\liErMpAttribute Let-Abkürzung: \let\a=\liErMpAttribute
949 \def\liErMpAttribute#1{
950   \liErAttribute{#1}
951   \marginpar{
952     \liErAttribute{\tiny\faCircleThin{}}~A:~#1}
953   }
954 }

\liErDatenbankName Let-Abkürzung: \let\d=\liErDatenbankName
datenbank name
955 \def\liErDatenbankName#1{
956   {
957     \footnotesize\texttt{(#1)}
958   }
959 }

960 \ExplSyntaxOff
961

```

1.12 formale-sprachen.sty

```

962 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
963 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
964 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]
965
966 \directlua{
967   formale_sprachen = require('lehramt-informatik-formale-sprachen')
968 }
969
970 \RequirePackage{hyperref}
971 \RequirePackage{lehramt-informatik-mathe}

\liMenge $\liMenge{a, b, c}$:  $\{a, b, c\}$ 
Let-Abkürzung: \let\m=\liMenge

972 \def\liMengeOhneMathe#1{\{ #1 \}}
973 \def\liMenge#1{%
974   \ifmmode%
975     \liMengeOhneMathe{#1}%
976   \else%
977     $\liMengeOhneMathe{#1}$%
978   \fi%
979 }

\liEpsilon \liEpsilon:  $\varepsilon$ 
Let-Abkürzung: \let\epsilon=\liEpsilon

980 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

981 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
982 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
983 \def\liPotenzmenge#1{$\liPotenzmengeOhneMathe{#1}$}

\liZustandsmenge \liZustandsmenge{z1, z2}:  $\{z_1, z_2\}$ 

984 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
985 \let\liZustandsmenge=\liPotenzmenge

\liUeberfuehrungsFunktion \liUeberfuehrungsFunktion{z0, a}:  $\delta(z_0, a)$ 
Let-Abkürzung: \let\d=\liUeberfuehrungsFunktion

986 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
987 \def\liUeberfuehrungsFunktion#1{
988   \ifmmode
989     \liUeberfuehrungsFunktionOhneMathe{#1}
990   \else
991     $\liUeberfuehrungsFunktionOhneMathe{#1}$
992   \fi
993 }

\liAlphabet \liAlphabet{a, b} ( $\Sigma = \{a, b\}$ ):  $\Sigma = \{a, b\}$ 

994 \def\liAlphabet#1{$\Sigma = \{ #1 \}$}

\liBandAlphabet \liBandAlphabet{\liTuringLeerzeichen}:  $\Gamma = \Sigma \cup \{\square\}$ 

995 \def\liBandAlphabet#1{$\Gamma = \Sigma \cup \{ #1 \}$}

\liZustandsBuchstabe

996 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

997 \def\liZustandsBuchstabeGross{Z}

```

```

\liZustandsmengeNr
    998 \def\zustandsnamens@liste#1#2{
    999   $
    1000   \{
    1001     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
    1002   \}
    1003   $
    1004 }
    1005 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

\liZustandsmengeNrGross
    1006 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

\liZustandsname \liZustandsname{1}: $z_1$
    1007 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

\liZustandsnameGross \liZustandsnameGross{1}: $Z_1$
    1008 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

\liAbleitung \liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
    1009 \def\liAbleitung#1{\directlua{formale_sprachen.formatiere_ableitung('#1')}}$}

liProduktionsRegeln \begin{liProduktionsRegeln}[P_1]
    S -> S A B | EPSILON,
    B A -> A B,
    A A -> a a,
    B B -> b b
\end{liProduktionsRegeln}

    1010 \NewDocumentEnvironment { liProduktionsRegeln }
    1011 { 0{P} +b }
    1012 {
    1013   \noindent
    1014   $#1 = \{ $
    1015   \vspace{-0.2cm}
    1016   \begin{align*}
    1017     \directlua{formale_sprachen.produktions_regeln('#2')}
    1018   \end{align*}
    1019   \vspace{-1.5cm}
    1020   \begin{flushright}\}\end{flushright}
    1021 } {}

\liProduktionen \liProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
    1022 \def\liProduktionen#1{
    1023   \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
    1024 }

\liZustandsnameTiefgestellt Automatisch tiefgestellte Nummerierung \z1
Let-Abkürzung: \let\z=\liZustandsnameTiefgestellt

    1025 \def\liZustandsnameTiefgestellt#1{
    1026   \ifmmode
    1027     \liZustandsBuchstabe\sb{#1}
    1028   \else
    1029     $\liZustandsBuchstabe\sb{#1}$
    1030   \fi
    1031 }

    1032 \ExplSyntaxOn

\liAusdruck \liAusdruck[L_2]{a_1,a_2,\dots,a_n}{n \in N}:  $L_2 = \{a_1, a_2, \dots, a_n \mid n \in N\}$ 
    Ohne „=:“: \liAusdruck[] {x}{y}:  $\{x \mid y\}$ 
    Regulärer Ausdruck zum Konvertieren:

```

```

\$(.*) += +\\{ *(.*) ( *\\, *)?\\| ( *\\, *)? (.*?) *\\}\\$
\\liAusdruck[$1]{$2}{$5}

1033 \NewDocumentCommand{ \liAusdruck } { O{L} m m } {
1034 $
1035 \tl_if_empty:nTF {#1} {} {#1 =}
1036 \{
1037 \, #2 \,
1038 |
1039 \, #3 \,
1040 \}$
1041 }
1042 \ExplSyntaxOff

\liFlaci Link zur flaci.com Website: \liFlaci{Grxk1oczg}:
Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte Automaten, Compiler und Interpreter)
Ein Projekt der Hochschule Zittau/Görlitz und der Pädagogischen Hochschule Schwyz: flaci.com/Grxk1oczg

1043 \def\liFlaci#1{%
1044 \par
1045 {%
1046 \scriptsize
1047 Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1048 Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1049 Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1050 \href{https://flaci.com/#1}{flaci.com/#1}%
1051 }%
1052 \par
1053 }

\liGrammatik \liGrammatik[\langle grammatik-name \rangle]{\langle variablen=V,alphabet=\Sigma,produktionen=P,start=S \rangle}
\liGrammatik{variablen={}, alphabet={}}

• \liGrammatik{}:  $G = (V, \Sigma, P, S)$ 

• \liGrammatik[G_1]{}:  $G_1 = (V, \Sigma, P, S)$ 

• \liGrammatik{variablen={S,A,B}}:  $G = (\{S, A, B\}, \Sigma, P, S)$ 

• \liGrammatik{alphabet={a,b}}:  $G = (V, \{a, b\}, P, S)$ 

• \liGrammatik{start=X}:  $G = (V, \Sigma, P, X)$ 

1054 \ExplSyntaxOn
1055 \NewDocumentCommand {\liGrammatik} { O{G} m } {
1056 \tl_set:Nn \l_variablen_tl {V}
1057 \tl_set:Nn \l_alphabet_tl {\Sigma}
1058 \tl_set:Nn \l_produktionen_tl {P}
1059 \tl_set:Nn \l_start_tl {S}
1060
1061 \keys_define:nn { grammatik } {
1062 variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1063 alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1064 produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1065 start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1066 }
1067
1068 \keys_set:nn { grammatik } { #2 }
1069
1070 $#1 = (
1071 \l_variablen_tl,
1072 \l_alphabet_tl,
1073 \l_produktionen_tl,
1074 \l_start_tl
1075 )$

```



```
1076 }  
1077 \ExplSyntaxOff  
  
1078
```

1.13 formatierung.sty

```
1079 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1080 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

1.13.1 Schriftarten / Typographie

```
1081 \RequirePackage{mathpazo}
1082 \RequirePackage[no-math]{fontspec}
1083 \setmainfont{texgyrepagella}
1084 \setsansfont{QTAncientOlive}
1085 \RequirePackage{sectsty}
1086 \allsectionsfont{\fontspec{QTAncientOlive}}
```

1.13.2 Farben

```
1087 \RequirePackage{xcolor}
1088 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

1.13.3 Überschriften

```
1089 \RequirePackage{titlesec}
1090 \titleformat{\chapter}[display]{\sffamily\bfseries}{\Opt}{\LARGE}
1091 \titlespacing{\chapter}{0pt}{0pt}{*1}
1092 \titleformat{\paragraph}[hang]{\sffamily\normalsize\bfseries}{\theparagraph}{1em}{}
1093 \setcounter{secnumdepth}{0}
```

1.13.4 Listen

```
1094 \RequirePackage{paralist}
1095 \renewcommand\labelitemi{-}
1096 \renewcommand\labelitemii{-}
1097 \renewcommand\labelitemiii{-}
1098 \renewcommand\labelitemiv{-}
1099 % Counter: enumi enumii enumiii enumiv
1100 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1101 \renewcommand{\labelenumi}{(\alph{enumi})}
1102 \renewcommand{\labelenumii}{(\roman{enumii})}
```

1.13.5 Kasten

```
1103 \RequirePackage{mdframed}
1104 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1105 \NewDocumentEnvironment { liKasten }{ } {
1106   \begin{mdframed}[backgroundcolor=white!0]
1107 } {
1108   \end{mdframed}
1109 }
```

1.13.6 Header

```
1110 \RequirePackage{fancyhdr}
1111 \fancyhead[L,C,R]{}
1112 \fancyfoot[L]{}
1113 \fancyfoot[C]{}
1114 \fancyfoot[R]{\thepage}
1115 \pagestyle{fancy}
1116 \renewcommand{\headrulewidth}{0pt}
1117 \renewcommand{\footrulewidth}{0pt}
1118
```

1.14 gantt.sty

```

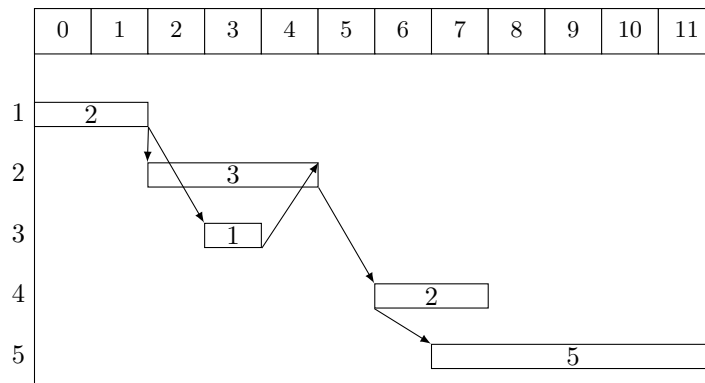
1119 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1120 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\begin{ganttchart}[x unit=0.75cm, y unit chart=0.8cm]{0}{11}
\gantttitlelist{0,...,11}{1} \\
\ganttbar[name=1]{1}{0}{1} \\
\ganttbar[name=2]{2}{2}{4} \\
\ganttbar[name=3]{3}{3}{3} \\
\ganttbar[name=4]{4}{6}{7} \\
\ganttbar[name=5]{5}{7}{11}

\node at (1) {2};
\node at (2) {3};
\node at (3) {1};
\node at (4) {2};
\node at (5) {5};

\ganttlink[link type=f-f]{3}{2}
\ganttlink[link type=f-s]{1}{2}
\ganttlink[link type=f-s]{1}{3}
\ganttlink[link type=f-s]{2}{4}
\ganttlink[link type=s-s]{4}{5}
\end{ganttchart}

```



```

1121 \RequirePackage{tikz-uml}
1122 \RequirePackage{pgfgantt}
1123 \setganttlinklabel{f-s}{}
1124 \setganttlinklabel{s-s}{}
1125 \setganttlinklabel{f-f}{}
1126 \setganttlinklabel{s-f}{}
1127

```

1.15 grafik.sty

```
1128 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1129 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1130 TikZ, um Grafiken zeichnen zu können.]
1131 \RequirePackage{tikz}
1132
```

1.16 graph.sty

```

1133 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1134 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1135 \RequirePackage{tikz}

```

Für die die Adjazenzliste (xrightarrow)

```
1136 \RequirePackage{amsmath}
```

Für Adjazenz-Matrix

```

\[
\begin{blockarray}{ccccc}
& a & b & c & d & e \\
\begin{block}{c(cccc)}
a & 0 & 1 & 0 & 4 & 0 \\
b & 0 & 0 & 0 & 1 & 3 \\
c & 0 & 0 & 0 & 1 & 0 \\
d & 4 & 1 & 1 & 0 & 2 \\
e & 0 & 0 & 0 & 0 & 0
\end{block}
\end{blockarray}
\]

```

	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>
<i>a</i>	0	1	0	4	0
<i>b</i>	0	0	0	1	3
<i>c</i>	0	0	0	1	0
<i>d</i>	4	1	1	0	2
<i>e</i>	0	0	0	0	0

```

1137 \RequirePackage{blkarray}
1138 \usetikzlibrary{arrows.meta}

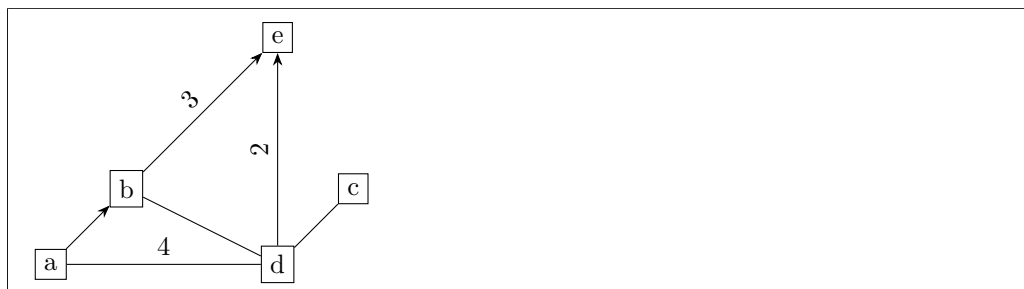
```

```

\begin{tikzpicture}[li graph]
\node (a) at (0,0) {a};
\node (b) at (1,1) {b};
\node (c) at (4,1) {c};
\node (d) at (3,0) {d};
\node (e) at (3,3) {e};

\path[->] (a) edge (b);
\path (b) edge (d);
\path[->] (b) edge node {3} (e);
\path (c) edge (d);
\path (d) edge node {4} (a);
\path[->] (d) edge node {2} (e);
\end{tikzpicture}

```



```

1139 \tikzset{
1140   li graph/.style={
1141     every node/.style={
1142       rectangle,
1143       draw,
1144     },
1145     every edge/.style={
1146       >={Stealth[black]},
1147       draw,
1148     },
1149     every edge/.append style={
1150       every node/.style={
1151         sloped,
1152         auto,
1153       }
1154     }
1155   },
1156   li markierung/.style={
1157     ultra thick,
1158   }
1159 }

```

liGraphenFormat Dummy-Umgebung, zum Parsen durch Java gedacht.

```

\begin{liGraphenFormat}
a: 0 0
b: 1 1
c: 4 1
d: 3 0
e: 2 2
a -> b
b -- d
b -> e: 3
c -- d
d -> e: 2
d -- a: 4
\end{liGraphenFormat}

```

```

1160 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1161

```

1.17 hanoi.sty

```

1162 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1163 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
1164 von Hanoi-Grafiken]

    Quelle: https://kleinco.de/latex-snippets/7/tikz-towers-of-hanoi-illustration-for-lat

1165 \RequirePackage{tikz}
1166 \RequirePackage{xcolor}

\liHanoi \liHanoi{anzahl-scheiben}{gewicht-scheibe/turm-nr,gewicht-scheibe/turm-nr},
z.B. : \liHanoi{4}{4/1,3/1,2/3,1/2}

1167 \def\li@mset #1[#2]=#3{%
1168 \expandafter\xdef\csname #1#2\endcsname{#3}
1169 }
1170 \def\li@mget #1[#2]{%
1171 \csname #1#2\endcsname
1172 }
1173 \def\li@minc #1[#2]+=#3{%
1174 \pgfmathparse{\li@mget #1[#2]+#3}%
1175 \li@mset #1[#2]=\pgfmathresult
1176 }
1177
1178 \def\liHanoi#1#2{
1179   \edef\li@numdiscs{#1}
1180   \def\li@sequence{#2}
1181   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1182     % init colors
1183     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purp
1184     \li@mset col[\j]={\c};
1185     % draw poles and init pole counters
1186     \foreach \j in {1,2,3}{
1187       \li@mset pos[\j]=0
1188       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1189     }
1190     % draw base
1191     \draw (.5,-.5) -- +(3,0);
1192     % draw discs
1193     \foreach[count=\k] \i/\j in \li@sequence{
1194       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*
1195       \li@minc pos[\j]+=.5}
1196     }
1197   \end{tikzpicture}
1198 }

1199

```

1.18 komplexitaetstheorie.sty

```

1200 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1201 \ProvidesPackage{lehrant-informatik-komplexitaetstheorie}[2021/07/08 Zum
1202 Setzen von Karps NP-vollständigen Problemen, Hilfsmakros für die
1203 Polynomialzeitreduktion.]

```

Let-Abkürzungen

```
\let\n=\liProblemName
\let\r=\liPolynomiellReduzierbar
\let\b=\liProblemBeschreibung
```

1204 \liLadePakete{mathe}

Für das Makro `\liProblemBeschreibung{ }{ }` benötigt.

1205 \RequirePackage{mdframed}

 $\backslash\mathrm{liStrich}\quad \$L, \backslash\mathrm{liStrich}\{L\}$: $L, L'$$

```
1206 \def\liStrich#1{#1^\prime}
```

`\liProblemName` Zu Setzen von Problemnamen wie zum Beispiel SAT, COL, VERTEX COVER

Let-Abkürzung: \let\n=\liProblemName

```
\liProblemName: SAT VERTEX COVER
```

1207 \def\liProblemName#1{\texttt{\textsc{#1}}}

\liProblemBeschreibung Zu setzen von Problem-Beschreibungen:

\liProblemBeschreibung

 $\{\}$ $\{\}$ $\{$

CLIQUE

Gegeben: Ein ungerichteter Graph $G = (V, E)$, eine Zahl $k \in \mathcal{N}$

Frage: Gibt es eine Menge $S \subseteq V$ mit $|S| = k$, sodass für alle Knoten $u \neq v \in V$ gilt, dass $\{u, v\}$ eine Kante in E ist?

Let-Abkürzung: \let\b=\liProblemBeschreibung

```

1208 \def\liProblemBeschreibung#1#2#3{
1209   \begin{mdframed}[
1210     userdefinedwidth=9cm,
1211     align=center,
1212     backgroundcolor=white!0,
1213   ]
1214     \centerline{\large\liProblemName{#1}}
1215
1216     \medskip
1217
1218     \begin{description}
1219       \item[Gegeben:] #2
1220       \item[Frage:] #3
1221     \end{description}
1222   \end{mdframed}
1223 }
```



```

\liPolynomiellReduzierbar Let-Abkürzung: \let\r=\liPolynomiellReduzierbar

1224 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1225 \begin{displaymath}
1226 \liProblemName{#1}
1227 \preceq_{#2}
1228 \liProblemName{#3}
1229 \end{displaymath}
1230 }

\liProblemVertexCover

1231 \def\liProblemClique{%
1232 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1233 Mindestgröße  $n$  in einem gegebenen Graphen.
1234 \footcite[Seite 76]{theo:fs:4}
1235 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1236 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1237 \footcite{wiki:cliquenproblem}
1238 }

\liProblemVertexCover

1239 \def\liProblemVertexCover{%
1240 %
1241 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1242 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1243 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1244 \footcite{wiki:knotenueberdeckung}
1245
1246 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1247 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1248 mindestens einem Knoten aus  $U$  verbunden ist.
1249 \footcite[Seite 78]{theo:fs:4}%
1250 }

\liProblemSubsetSum Kein Karp-21-Problem

1251 \def\liProblemSubsetSum{%
1252 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1253 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1254 \footcite{wiki:teilsommenproblem}
1255 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1256 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1257 größer als eine gegebene obere Schranke  $c$  ist.
1258 \footcite[Seite 74]{theo:fs:4}
1259 }

\liProblemSubsetSum Kein Karp-21-Problem

1260 \def\liProblemSat{%
1261 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1262 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1263 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1264 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1265 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1266 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1267 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1268 aufgestellt werden.
1269 \footcite[Seite 71]{theo:fs:4}
1270 }

1271

```

1.19 kontrollflussgraph.sty

```
1272 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1273 \ProvidesPackage{lehramt-informatik-kontrollflussgraph}[2020/11/07]

1274 \RequirePackage{tikz}
1275 \usetikzlibrary{positioning}
1276 \tikzset{
1277   li kontrollfluss/.style={
1278     knoten/.style={
1279       circle,
1280       draw
1281     },
1282     usebox/.style={
1283       draw,
1284       rectangle,
1285       font=\scriptsize,
1286       anchor=west,
1287       align=left,
1288     },
1289     bedingung/.style={
1290       midway,
1291       draw=none,
1292       font=\scriptsize
1293     },
1294     knotenbeschriftung/.style={
1295       draw,
1296       rectangle,
1297       midway,
1298       font=\scriptsize
1299     },
1300     wahr/.style={
1301       thick
1302     },
1303     falsch/.style={
1304       dashed
1305     },
1306     every node/.style={
1307       circle,
1308       draw,
1309     },
1310     every edge/.append style={
1311       every node/.style={
1312         draw=none,
1313         bedingung,
1314       }
1315     },
1316     every path/.style={
1317       draw,
1318       ->,
1319     },
1320     every pin/.style={
1321       draw,
1322       dotted,
1323       rectangle,
1324       pin position=right
1325     },
1326     every pin edge/.style={
1327       dotted,
1328       arrows=-,
1329     }
1330   }
1331 }
```

liKontrollflussgraph

```

1332 \NewDocumentEnvironment { liKontrollflussgraph } { 0{ } } {
1333   \begin{tikzpicture}[
1334     li kontrollfluss,
1335     #1
1336   ]
1337 } {
1338   \end{tikzpicture}
1339 }

\liAnweisung
1340 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}

\liBedingung Let-Abkürzung: \let\b=\liBedingung
1341 \def\liBedingung#1#2{node[bedingung,#1]{\texttt{#2}}}}

\liBedingungWahr Let-Abkürzung: \let\bWahr=\liBedingungWahr
1342 \def\liBedingungWahr#1{node[bedingung,#1]{\texttt{[true]}}}}

\liBedingungFalsch Let-Abkürzung: \let\bFalsch=\liBedingungFalsch
1343 \def\liBedingungFalsch#1{node[bedingung,#1]{\texttt{[false]}}}}

\liKontrollCode Let-Abkürzung: \let\c=\liKontrollCode
1344 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}

\liKontrollTextzeileKnoten Let-Abkürzung: \let\k=\liKontrollTextzeileKnoten
1345 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw

\liKontrollKnotenPfad Let-Abkürzung: \let\p=\liKontrollKnotenPfad
1346 \ExplSyntaxOn
1347 \NewDocumentCommand { \liKontrollKnotenPfad }{ m }
1348 {
1349   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
1350   \seq_set_map:Nnn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
1351   \seq_use:Nn \l_tmpa_seq { ~~~~ }
1352 }
1353 \ExplSyntaxOff

1354

```

1.20 literatur-dummy.sty

```
1355 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1356 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1357 \def\literatur{}

\footcite

1358 % \RequirePackage[stable,multiple]{footmisc}
1359 \NewDocumentCommand{ \footcite } { o m }{}

1360
```

1.21 literatur.sty

```
1361 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1362 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1363 \RequirePackage{csquotes}
1364 \RequirePackage[
1365   bibencoding=utf8,
1366   citestyle=authortitle,
1367   backend=biber,
1368 ]{biblatex}
1369 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1370 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1371 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1372 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1373 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1374 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1375 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1376 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1377 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1378 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1379 % To allow footnotes in the heading
1380 \RequirePackage[stable,multiple]{footmisc}

\literatur

1381 \def\literatur{\printbibliography[heading=subbibliography]}

1382
```

1.22 makros.sty

```

1383 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1384 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1385 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1386 anderen Paket passen]
1387 \RequirePackage{hyperref}
1388 \RequirePackage{graphicx}
    Für die Umgebung liQuellen benötigt.
1389 \RequirePackage{paralist}

\inhaltsverzeichnis
1390 \def\inhaltsverzeichnis {
1391   \begin{mdframed}
1392     \begin{group}
1393       \let\clearpage\relax
1394       \tableofcontents
1395     \end{group}
1396   \end{mdframed}
1397 }

\memph \mephm (\marginpar and \emph)
1398 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1399 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1400 \newcommand{\liPseudoUeberschrift}[1]{
1401   \bigskip
1402   \noindent
1403   \textsf{\textbf{#1}}
1404   \noindent
1405 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1406 \newcommand{\liBeschriftung}[1]{
1407   \par
1408   \noindent
1409   \medskip
1410   \textbf{#1}:
1411   \medskip
1412   \noindent
1413 }

\hinweis
1414 \def\hinweis#1{{\footnotesize#1}}

liProjektSprache \begin{liProjektSprache}{NameProjektSprache} \end{liProjektSprache}: Zum Einbetten von projekteigenen Minisprachen/DSLs (Domain-specific language) (z. B. RelationenSchema). Der Inhalt der Umgebung wird nicht von TeX kompiliert, sondern von dem Java-Kommandozeilen-Programm didaktik.java verarbeitet.
1415 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

liEinbettung
1416 \NewDocumentEnvironment { liEinbettung }{ o +b } {#2} {}

liAntwort Zum Setzen von Antworten. Sie werden mit einem Kasten umgeben. Könnten eventuell ausgeblendet werden, wenn man nur die Lösung sehen will.
1417 \RequirePackage{xparse}
1418 \ExplSyntaxOn

```

```

1419 \NewDocumentEnvironment{ liAntwort } { 0{standard} }
1420 {
1421   \str_case:nn {#1} {
1422     {standard} {
1423       \def\beschriftung{}
1424       \mdfsetup{backgroundcolor=gray!5,linecolor=gray}
1425     }
1426     {richtig} {
1427       \def\beschriftung{richtig}
1428       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1429     }
1430     {falsch} {
1431       \def\beschriftung{falsch}
1432       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1433     }
1434     {muster} {
1435       \def\beschriftung{Musterlösung}
1436       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1437     }
1438   }
1439   \ifx\beschriftung\empty\else
1440     \noindent
1441     \textbf{\beschriftung{:}}
1442     \fi
1443     \begin{mdframed}
1444   }
1445   {\end{mdframed}}

```

liAdditum Zusätzliches Material bei Aufgaben, das zum Lösen der Aufgaben nicht unbedingt nötig ist, z. b. Hintergrundinformation, Test-Dateien, komplette Code-Dateien.

```

1446 \NewDocumentEnvironment{ liAdditum } { o }
1447 {
1448   \begin{mdframed}[backgroundcolor=yellow!5]
1449     \IfNoValueTF {#1}
1450     { \liPseudoUeberschrift{Additum} }
1451     { \liPseudoUeberschrift{Additum:~#1} }
1452   }
1453   {\end{mdframed}}

```

liExkurs \begin{liExkurs}[Linear rekursiv]
 Die häufigste Rekursionsform ist die lineare Rekursion, bei der in jedem Fall der rekursiven Definition höchstens ein rekursiver Aufruf vorkommen darf.
 \end{liExkurs}

Exkurs: Linear rekursiv

Die häufigste Rekursionsform ist die lineare Rekursion, bei der in jedem Fall der rekursiven Definition höchstens ein rekursiver Aufruf vorkommen darf.

```

1454 \NewDocumentEnvironment{ liExkurs }{o +b}{
1455   \vspace{0.2cm}%
1456   \begin{mdframed}[
1457     backgroundcolor=white,
1458     bottomline=false,
1459     innermargin=1cm,
1460     leftline=true,
1461     linecolor=black,
1462     linewidth=0.1cm,
1463     outermargin=1cm,
1464     rightline=false,
1465     topline=false,
1466   ]

```

```

1467 \footnotesize
1468 \noindent%
1469 \textbf{Exkurs:~#1}\par%
1470 \noindent%
1471 #2
1472 \end{mdframed}
1473 \vspace{0.2cm}
1474 }{}

```

liQuellen <https://tex.stackexchange.com/a/229004>

```

\begin{liQuellen}
\item Quelle 1
\item Quelle 2
\end{liQuellen}

```

Weiterführende Literatur:

- Quelle 1
- Quelle 2

```

1475 \cs_new:Npn \listen@punkt #1 {\item #1}
1476 \NewDocumentEnvironment { liQuellen }{ +b }
1477 {
1478   \seq_clear_new:N \l_quellen
1479   \seq_set_split:Nnn \l_quellen {\item} {#1}
1480   \seq_remove_all:Nn \l_quellen {}
1481   \begin{mdframed}[backgroundcolor=white!0]
1482     \footnotesize
1483     \noindent
1484     \textsf{\textbf{Weiterführende~Literatur:}}
1485     \medskip
1486     \begin{compactitem}
1487       \seq_map_function:NN \l_quellen {\listen@punkt}
1488     \end{compactitem}
1489   \end{mdframed}
1490   %
1491   \makeatletter
1492   \par\@afterindentfalse\@afterheading
1493   \makeatother
1494 } {}

```

liLernkartei

```

1495 \NewDocumentEnvironment { liLernkartei }{ m +b }
1496 {
1497   \begin{mdframed}
1498     \footnotesize
1499     \noindent%
1500     \textbf{Lernkarteikarte:~#1}\par%
1501     \noindent%
1502     #2
1503   \end{mdframed}
1504 } {}

```

liDiagramm `\begin{liDiagramm}{beschriftung}\end{liDiagramm}`: Zu setzen einer Graphik bzw eines Diagramms.

```

1505 \NewDocumentEnvironment { liDiagramm }{ m +b }
1506 {
1507   \begin{mdframed}[backgroundcolor=white!0]
1508     \small
1509     \noindent%
1510     \textit{#1}:
1511     \begin{center}

```



```

1512 #2
1513 \medskip
1514 \end{center}
1515 \end{mdframed}
1516 } {}
1517 \ExplSyntaxOff

\liFussnoteUrl \liFussnoteUrl[\langle zusätzlicher-text \rangle]{\langle url \rangle} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
1518 \NewDocumentCommand{\liFussnoteUrl} { o m } {
1519 \footnote{\url{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1520 }
1521

\liFussnoteLink \liFussnoteLink[\langle zusätzlicher-text \rangle]{\langle link-text \rangle}{\langle url \rangle} \liFussnoteLink[zusätzlicher Text]{tex
Einen Link, d. h. einen Link-Text und eine URL als Fußnote setzen.
1522 \NewDocumentCommand{\liFussnoteLink} { o m m } {
1523 \footnote{\href{#3}{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1524 }

\zB
1525 \def\zB{z.\,B. }

\ZB
1526 \def\ZB{Z.\,B. }

\dh
1527 \def\dh{d.\,h. }

1528

```

1.23 master-theorem.sty

1529 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1530 \ProvidesPackage{lehramt-informatik-master-theorem}[2021/04/13]

Let-Abkürzungen

\let\0=\li0

\let\o=\li0omega

\let\T=\liT

\let\t=\liTheta

1531 \ExplSyntaxOn

1532 \RequirePackage{amsmath}

\liRundeKlammer

1533 \def\liRundeKlammer#1{

1534 \negthinspace \left(#1 \right)

1535 }

\liTheta \liTheta{n^2}: $\Theta(n^2)$

1536 \def\liThetaOhneMathe#1{

1537 \Theta \liRundeKlammer{#1}

1538 }

1539 \def\liTheta#1{

1540 \ifmmode

1541 \liThetaOhneMathe{#1}

1542 \else

1543 \$\liThetaOhneMathe{#1}\$

1544 \fi

1545 }

\li0omega \li0omega{n^2}: $\Omega(n^2)$

1546 \def\li0omegaOhneMathe#1{

1547 \Omega \liRundeKlammer{#1}

1548 }

1549 \def\li0omega#1{

1550 \ifmmode

1551 \li0omegaOhneMathe{#1}

1552 \else

1553 \$\li0omegaOhneMathe{#1}\$

1554 \fi

1555 }

\li0 \li0{n^2}: $\mathcal{O}(n^2)$

1556 \def\li0OhneMathe#1{

1557 \mathcal{O} \liRundeKlammer{#1}

1558 }

1559 \def\li0#1{

1560 \ifmmode

1561 \li0OhneMathe{#1}

1562 \else

1563 \$\li0OhneMathe{#1}\$

1564 \fi

1565 }

\liT **Let-Abkürzung:** \let\T=\liT

\liT{16}{2}: $16 \cdot T\left(\frac{n}{2}\right)$ \liT{}{2}: $T\left(\frac{n}{2}\right)$

1566 \def\liTOhneMathe#1#2{

1567 \tl_if_blank:nTF {#1}

1568 {}

1569 {#1 \cdot }

1570 T

1571 \liRundeKlammer{\frac{n}{#2}}

```

1572 }
1573 \def\liT#1#2{
1574   \ifmmode
1575     \liTOhneMathe{#1}{#2}
1576   \else
1577     $\liTOhneMathe{#1}{#2}$
1578   \fi
1579 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
1580 \def\liRekursionsGleichung{
1581   $T(n) = \liT{a}{b} + f(n)$
1582 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
1583 \def\liBedingungEins{
1584   $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
1585 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
1586 \def\liBedingungZwei{
1587   $f(n) \in \liTheta{n^{\log\sb{b}a}}$
1588 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
1589 \def\liBedingungDrei{
1590   $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
1591 }

1592 \ExplSyntaxOff

\liMasterVariablen
1593 \def\liMasterVariablen{
1594   \begin{displaymath}
1595     T(n) = \liT{a}{b} + f(n)
1596   \end{displaymath}
1597
1598   \begin{itemize}
1599     \item[$a = $]
1600     Anzahl der Unterprobleme in der Rekursion
1601
1602     \item[$\textstyle\frac{1}{b} = $]
1603     Teil des Originalproblems, welches wiederum durch alle Unterprobleme
1604     repräsentiert wird
1605
1606     \item[$f(n) = $]
1607     Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
1608     die Kombination der Teillösungen entstehen
1609   \end{itemize}
1610   \footcite{wiki:master-theorem}
1611   \footcite[Seite 19-35 (PDF 11-24)]{aud:fs:2}
1612 }

\liMasterFaelle
1613 \def\liMasterFaelle{
1614   \begin{description}
1615     \item[1. Fall:]
1616     $T(n) \in \liT{n^{\log\sb{b}a}}$
1617
1618     \hfill falls \liBedingungEins
1619     für $\varepsilon > 0$
1620
1621     \item[2. Fall:]

```

```

1622 $T(n) \in \mathcal{O}(\log^a n \cdot \log n)$
1623
1624 \hfill falls \liBedingungZwei
1625
1626 \item[3. Fall:]
1627 $T(n) \in \mathcal{O}(f(n))$
1628
1629 \hfill falls \liBedingungDrei
1630 für $\varepsilon > 0$
1631 und ebenfalls für ein $c$ mit $0 < c < 1$ und alle hinreichend großen $n$
1632 gilt:
1633 $a \cdot f(\frac{n}{b}) \leq c \cdot f(n)$
1634 \end{description}
1635 }
1636

```

1.24 mathe.sty

```
1637 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1638 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
1639
1640 % for example \ltimes \rtimes
1641 %\RequirePackage{amssymb}
1642 \RequirePackage{amsmath}
1643
1644 %%
1645 % \mlq \mrq
1646 %%
1647 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
1648 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
1649
```

1.25 minimierung.sty

```

1650 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1651 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
1652 Minimierungsalgorithmus von einem NEA zu einem DEA]

1653 \liLadePakete{typographie}

\let\z=\liZustandsnameTiefgestellt
\let\f=\liFussnote
\let\l=\liLeereZelle
\let\Z=\liZustandsPaar
\let\erklaerung=\liMinimierungErklaerung

\begin{tabular}{|c||c|c|c|c|c|c|c|c|c|}
\hline
\z0 & \l & \l & \l & \l & \l & \l & \l & \l & \l & \l \\
\z1 & & \l & \l & \l & \l & \l & \l & \l & \l & \l \\
\z2 & & & \l & \l & \l & \l & \l & \l & \l & \l \\
\z3 & & & & \l & \l & \l & \l & \l & \l & \l \\
\z4 & & & & & \l & \l & \l & \l & \l & \l \\
\z5 & & & & & & \l & \l & \l & \l & \l \\
\z6 & & & & & & & \l & \l & \l & \l \\
\z7 & & & & & & & & \l & \l & \l \\
\z8 & & & & & & & & & \l & \l \\
& \z0 & \z1 & \z2 & \z3 & \z4 & \z5 & \z6 & \z7 & \z8 & \l \\
\end{tabular}

\liFussnoten

\begin{liUebergangsTabelle}{0}{1}
\Z01 & \Z10 & \Z23 & \\
\Z05 & \Z15 & \Z25 & \f2 \\
\Z15 & \Z05 & \Z35 & \f2 \\
\Z23 & \Z44 & \Z55 & \\
\Z24 & \Z44 & \Z55 & \\
\Z34 & \Z44 & \Z55 & \\
\end{liUebergangsTabelle}

\liFussnote

1654 \def\liFussnote#1{\$x_{#1}\$}

1655 \def\li@fussnote@text#1#2{
1656 \liFussnote{#1}
1657 \quad
1658 {\footnotesize #2}
1659 }

\liFussnoteEinsText
1660 \def\liFussnoteEinsText{
1661 \li@fussnote@text{1}
1662 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
1663 }

\liFussnoteZweiText
1664 \def\liFussnoteZweiText{
1665 \li@fussnote@text{2}
1666 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
1667 }

\liFussnoteDreiText
1668 \def\liFussnoteDreiText{
1669 \li@fussnote@text{3}

```

```

1670 {In weiteren Iterationen markierte Zustände.}
1671 }

\liFussnoteVierText
1672 \def\liFussnoteVierText{
1673   \li@fussnote@text{4}
1674   {...}
1675 }

\liFussnoten


|       |                                                                      |
|-------|----------------------------------------------------------------------|
| $x_1$ | Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.          |
| $x_2$ | Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt. |
| $x_3$ | In weiteren Iterationen markierte Zustände.                          |
| $x_4$ | ...                                                                  |



1676 \def\liFussnoten{
1677   \bigskip
1678
1679   \noindent
1680   \liFussnoteEinsText
1681
1682   \noindent
1683   \liFussnoteZweiText
1684
1685   \noindent
1686   \liFussnoteDreiText
1687
1688   \noindent
1689   \liFussnoteVierText
1690 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
1691 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
1692 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
1693 \def\liZustandsPaar#1#2{
1694   $(
1695     \liZustandsPaarVariablenName_#1,
1696     \liZustandsPaarVariablenName_#2
1697   )$
1698 }

liUebergangsTabelle
1699 \renewcommand{\arraystretch}{1.4}
1700 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
1701   \liPseudoUeberschrift{Übergangstabelle}
1702   \begin{center}
1703     \begin{tabular}{r|l|l}
1704       \textbf{Zustandspaar} & \textbf{\#1} & \textbf{\#2} \\ \hline
1705     \end{tabular}
1706   \end{center}
1707 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle: Minimierungstabelle (Table filling)

1709 \ExplSyntaxOn
1710 \def\liUeberschriftDreiecksTabelle{
1711   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
1712 }

```

\liMinimierungErklaerung **Let-Abkürzung:** `\let\verklaerung=\liMinimierungErklaerung`

— Der Minimierungs-Algorithmus (auch Table-Filling-Algorithmus genannt) trägt in seinem Verlauf eine Markierung in alle diejenigen Zellen der Tabelle ein, die zueinander nicht äquivalente Zustände bezeichnen. Die Markierung „ x_n “ in einer Tabellenzelle (i, j) bedeutet dabei, dass das Zustandspaar (i, j) in der k -ten Iteration des Algorithmus markiert wurde und die Zustände i und j somit zueinander $(k - 1)$ -äquivalent, aber nicht k -äquivalent und somit insbesondere nicht äquivalent sind. Bleibt eine Zelle bis zum Ende unmarkiert, sind die entsprechenden Zustände zueinander äquivalent. —————

```

1713 \def\liMinimierungErklaerung{
1714   %\footcite[Seite~19]{koenig}
1715   \liParagraphMitLinien{
1716     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
1717     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
1718     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
1719     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
1720      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
1721     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
1722     somit~zueinander~( $k - 1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
1723     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
1724     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
1725   }
1726 }
1727 \ExplSyntaxOff
1728

```


1.26 normalformen.sty

```

1729 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1730 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10
1731 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
1732 Attributhülle]
1733 \liLadePakete{mathe}
1734 \directlua{
1735   helfer = require('lehramt-informatik-helfer')
1736   normalformen = require('lehramt-informatik-normalformen')
1737 }

```

Let-Abkürzungen

```

\let\ah=\liAttributHuelle
\let\ahL=\liAttributHuelleLinksReduktion
\let\fa=\liFunktionaleAbhaengigkeit
\let\FA=\liFunktionaleAbhaengigkeiten
\let\m=\liAttributMenge
\let\r=\liRelation

```

```

1738 \def\liTeilen#1{
1739   \directlua{tex.print(normalformen.teilen('#1'))}
1740 }

```

\liAttributHuelle Let-Abkürzung: `\let\ah=\liAttributHuelle`
 Regulärer Ausdruck zum Konvertieren `AttrHülle\((.*)\)` in `\ah{$1}`

```

1741 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(#1)}
1742 \def\liAttributHuelle#1{
1743   \ifmmode
1744     \liAttributHuelleOhneMathe{#1}
1745   \else
1746     $\liAttributHuelleOhneMathe{#1}$
1747   \fi
1748 }

```

\liAttributMenge Let-Abkürzung: `\let\m=\liAttributMenge`
 1749 `\def\liAttributMenge#1{\{ \textit{#1} \}}`

liAHuelle

```

1750 \NewDocumentEnvironment{ liAHuelle } { +b } {
1751   \begingroup
1752   \footnotesize
1753   \begin{multline*}
1754     #1
1755   \end{multline*}
1756   \endgroup
1757 } { }

```

AttributHuelleLinksReduktion Nur innerhalb von `liAHuelle` zu verwenden bzw. `multline`

Let-Abkürzung: `\let\ahL=\liAttributHuelleLinksReduktion`
`\ahL{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}`

```

1758 \def\liAttributHuelleLinksReduktion#1#2#3{
1759   \shoveleft{
1760     \liAttributHuelleOhneMathe{FA, \liAttributMenge{\liAttributMenge{#1} - \liAttributMenge{#2}}
1761   } \\\
1762   \shoveright{
1763     \liAttributMenge{#3}
1764   } \\\
1765 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
    \liFunktionaleAbhaengigkeit{A, B -> C, D} Regulärer Ausdruck zum Konvertieren:
    \$(.*) \rightarrow (.*)\$ \fa{$1 -> $2}
1766 \def\liFunktionaleAbhaengigkeit#1{%
1767   \directlua{normalformen.drucke_funk_abhaengigkeit('#1')}%
1768 }

\liFunktionaleAbhaengigkeiten \liFunktionaleAbhaengigkeiten[F]{
    M -> M;
    M -> N;
    V -> T, P, PN;
    P -> PN;
}

Let-Abkürzung: \let\FA=\liFunktionaleAbhaengigkeiten
    Regulärer Ausdruck zum Konvertieren: \item \$(.*) \rightarrow (.*)\$ \$(.*) \rightarrow
    $1 -> $2;
1769 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
1770   \par
1771   \noindent
1772   #1 $= \{$
1773   \par
1774   \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
1775   \par
1776   \noindent$}\$
1777 }

\liRelation Let-Abkürzung: \let\r=\liRelation

```

$$R_3(A, B, C)$$

```

    Regulärer Ausdruck zum Konvertieren: \$(R.*)\((.*)\)\$ \liRelation[$1]{$2}
1778 \NewDocumentCommand {\liRelation} { O{R} m } {
1779   $\directlua{
1780     local name = helper.konvertiere_tiefgestellt('#1')
1781     tex.print(name)
1782   }$(\textit{\,#2\,})
1783 }
1784

```

1.27 petri.sty

```
1785 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1786 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]
```

Let-Abkürzungen

```
\let\t=\liPetriTransitionsName
\let\tp=\liPetriTransPfeile
\let\k=\liPetriErreichKnotenDrei
```

```
1787 \RequirePackage{tikz}
1788 \usetikzlibrary{petri,arrows.meta}
```

Für die Darstellungsmatrix

```
1789 \RequirePackage{blkarray}

\def\TmpA#1{
  \liPetriSetzeSchluessel%
  \pgfkeys{/petri/.cd,#1}%
  \begin{tikzpicture}[li petri]
    \node at (-0.25,-0.25) {};
    \node at (\TmpX,\TmpY) {};

    \begin{scope}[transform canvas={scale=\TmpScale},x=2cm,y=2cm,]
      \node[place,tokens=\TmpPlaceOne,label=$p_1$] at (0,1) (p1) {};
      \node[place,tokens=\TmpPlaceTwo,label=$p_2$] at (2,2) (p2) {};
      \node[place,tokens=\TmpPlaceThree,label=east:$p_3$] at (2,0) (p3) {};

      \node[transition,label=east:$t_1$, \TmpTransitionOne] at (2,1) {}
        edge[pre] (p2)
        edge[post] (p3);
      \node[transition,label=$t_2$, \TmpTransitionTwo] at (1,1.5) {}
        edge[pre] (p1)
        edge[post] (p2);
      \node[transition,label=$t_3$, \TmpTransitionThree] at (1,0.5) {}
        edge[pre] (p3)
        edge[post] (p1);
      \node[transition,label=$t_4$, \TmpTransitionFour] at (1,1) {}
        edge[pre] (p2)
        edge[pre] (p3)
        edge[post] (p1);
    \end{scope}
  \end{tikzpicture}
}
```

\liPetriSetzeSchluessel

```
1790 \def\liPetriSetzeSchluessel{%
1791   \def\TmpTransitionOne{}%
1792   \def\TmpTransitionTwo{}%
1793   \def\TmpTransitionThree{}%
1794   \def\TmpTransitionFour{}%
1795   \def\TmpTransitionFive{}%
1796   \def\TmpTransitionSix{}%
1797   \def\TmpTransitionSeven{}%
1798   \def\TmpTransitionEight{}%
1799   \def\TmpTransitionNine{}%
1800   \def\TmpTransitionTen{}%
1801   \pgfkeys{/petri/.cd,
1802     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
1803     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
1804     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
1805     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
1806     p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
```

```

1807     p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
1808     p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
1809     p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
1810     p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
1811     p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
1812     t1/.store in=\TmpTransitionOne,t1/.default=activated,
1813     t2/.store in=\TmpTransitionTwo,t2/.default=activated,
1814     t3/.store in=\TmpTransitionThree,t3/.default=activated,
1815     t4/.store in=\TmpTransitionFour,t4/.default=activated,
1816     t5/.store in=\TmpTransitionFive,t5/.default=activated,
1817     t6/.store in=\TmpTransitionSix,t6/.default=activated,
1818     t7/.store in=\TmpTransitionSeven,t7/.default=activated,
1819     t8/.store in=\TmpTransitionEight,t8/.default=activated,
1820     t9/.store in=\TmpTransitionNine,t9/.default=activated,
1821     t10/.store in=\TmpTransitionTen,t10/.default=activated,
1822     scale/.store in=\TmpScale,scale/.default=0.5,
1823     x/.store in=\TmpX,x/.default=5,
1824     y/.store in=\TmpY,y/.default=5,
1825   }%
1826 }

1827 \tikzset{
1828   li petri/.style={
1829     activated/.style={
1830       very thick
1831     },
1832     inhibitor/.style={
1833       {Circle[open,length=2mm,fill=white]}-
1834     }
1835   }
1836 }

```

\liPetriTransitionsName **Let-Abkürzung:** \let\t=\liPetriTransitionsName
 \t_(\d+)\\$ \t\$1

```

1837 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
1838 \def\liPetriTransitionsName#1{
1839   \ifmmode
1840     \liPetriTransitionsNameOhneMathe{#1}
1841   \else
1842     $\liPetriTransitionsNameOhneMathe{#1}$
1843   \fi
1844 }

```

\liPetriErreichTransition **Let-Abkürzung:** \let\t=\liPetriErreichTransition

```

1845 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
1846   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
1847 }

```

\liPetriErreichKnotenDrei **Let-Abkürzung:** \let\k=\liPetriErreichKnotenDrei

```

1848 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

\liPetriTransPfeile **Let-Abkürzung:** \let\tp=\liPetriTransPfeile

```

1849 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm} }
1850

```

1.28 potenzmengen-konstruktion.sty

```
1851 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1852 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
1853 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
1854 \liLadePakete{formale-sprachen}
1855 \ExplSyntaxOn
```

```
\def\z#1{
  \liZustandsMengenSammlungNr{#1}{
    {
      {0} {0}
      {1} {0,1}
      {2} {0,2}
      {3} {0,1,3}
      {4} {0,2,3}
      {5} {0,3}
    }
  }
}

\let\s=\liZustandsnameGross

\begin{tabular}{l|l|l}
Zustandsmenge & Eingabe $a$ & Eingabe $b$ \\ \hline
\z0 & \z0 & \z1 \\
\z1 & \z2 & \z1 \\
\z2 & \z0 & \z3 \\
\z3 & \z4 & \z3 \\
\z4 & \z5 & \z3 \\
\z5 & \z5 & \z3 \\
\end{tabular}
```

```
\liZustandsMengenSammlung \liZustandsMengenSammlung{nummer}{latex3 str_case:nn}
```

```
\def\z#1{
  \liZustandsMengenSammlung{#1}{
    {
      {0} {z0}
      {1} {z0, z1}
      {2} {z0, z1, z2}
      {3} {z0, z2}
      {4} {z0, z1, z2, z3}
      {5} {z0, z3}
      {6} {z0, z2, z3}
      {7} {z0, z1, z3}
    }
  }
}
```

```
1856 \def\liZustandsMengenSammlung#1#2{
1857   \liZustandsnameGross{#1}
1858   {
1859     \footnotesize
1860     \liPotenzmenge{
1861       \str_case:nn {#1} {#2
1862       }
1863     }
1864 }
```

```
\liZustandsMengenSammlungNr
```

```
1865 \def\liZustandsMengenSammlungNr#1#2{
1866   \liZustandsnameGross{#1}
1867   {
```

```
1868     \footnotesize
1869     \liZustandsmengeNr{
1870         \str_case:nn {#1} #2
1871     }
1872 }
1873 }

1874 \ExplSyntaxOff
1875
```

1.29 pseudo.sty

1876 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 1877 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen
 1878 von Pseudo-Code, Hüll-Paket um algorithm2e]

```
\begin{algorithm}[H]
\KwData{$G = (V, E, w)$: ein zusammenhängender, ungerichteter,
kantengewichteter Graph kruskal(G)}
$E' \leftarrow \emptyset$;
$L \leftarrow E$;
Sortiere die Kanten in $L$ aufsteigend nach ihrem Kantengewicht.;
\While{$L \neq \emptyset$}{
  wähle eine Kante $e \in L$ mit kleinstem Kantengewicht;
  entferne die Kante $e$ aus $L$;
  \If{der Graph $(V, E' \cup \{e\})$ keinen Kreis enthält}{
    $E' \leftarrow E' \cup \{e\}$;
  }
}
\KwResult{$M = (V, E')$ ist ein minimaler Spannbaum von $G$.}
\caption{Minimaler Spannbaum nach Kruskal\footcite{wiki:kruskal}}
\end{algorithm}
```

Algorithmus 1: Minimaler Spannbaum nach Kruskal

<p>Data: $G = (V, E, w)$: ein zusammenhängender, ungerichteter, kantengewichteter Graph kruskal(G)</p> <p>$E' \leftarrow \emptyset$; $L \leftarrow E$; Sortiere die Kanten in L aufsteigend nach ihrem Kantengewicht.;</p> <p>while $L \neq \emptyset$ do</p> <table border="0"> <tr> <td style="padding-right: 10px;"> </td> <td>wähle eine Kante $e \in L$ mit kleinstem Kantengewicht;</td> </tr> <tr> <td style="padding-right: 10px;"> </td> <td>entferne die Kante e aus L;</td> </tr> <tr> <td style="padding-right: 10px;"> </td> <td>if der Graph $(V, E' \cup \{e\})$ keinen Kreis enthält then</td> </tr> <tr> <td style="padding-right: 10px;"> </td> <td style="padding-left: 10px;">$E' \leftarrow E' \cup \{e\}$;</td> </tr> <tr> <td style="padding-right: 10px;"> </td> <td>end</td> </tr> </table> <p>end</p> <p>Result: $M = (V, E')$ ist ein minimaler Spannbaum von G.</p>		wähle eine Kante $e \in L$ mit kleinstem Kantengewicht;		entferne die Kante e aus L ;		if der Graph $(V, E' \cup \{e\})$ keinen Kreis enthält then		$E' \leftarrow E' \cup \{e\}$;		end
	wähle eine Kante $e \in L$ mit kleinstem Kantengewicht;									
	entferne die Kante e aus L ;									
	if der Graph $(V, E' \cup \{e\})$ keinen Kreis enthält then									
	$E' \leftarrow E' \cup \{e\}$;									
	end									

1879 \RequirePackage[german,boxruled]{algorithm2e}

1880

1.30 quicksort.sty

```

1881 % https://tex.stackexchange.com/a/142634
1882 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1883 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
1884
1885 %-----
1886 % USAGE:
1887 % \QSinitialize{comma, separated, numerical, values}
1888 % \loop
1889 % \QSpivotStep
1890 % \ifnum\value{pivotcount}>0
1891 %   \QSSortStep
1892 % \repeat
1893 %-----
1894
1895 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
1896 \RequirePackage{xintfrac, xinttools}
1897
1898 \RequirePackage{tikz}
1899
1900 %-----
1901 % FIRST PART: TikZ styles and macros for the actual drawing
1902 \newcounter{cellcount}% used for coordinates of the node
1903 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
1904
1905 % Styles defined by Tom Bombaldi. (modified: all share the same size)
1906 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
1907 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
1908 % if I want to use \bf, and know what I am doing, why should I get HARASSED
1909 % by police of LaTeX good conduct ? )
1910 \tikzset{!/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
1911          o/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
1912          r/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
1913 % this is the "b" style as used in the image below
1914          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
1915 % nicer:
1916          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
1917          g/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
1918
1919 % NOTE the b style was originally the same as the r(aised) style apart from
1920 % not being raised, but I find it nicer with a somewhat different
1921 % specification. I have not updated the images though.
1922
1923 % How the nodes are drawn depending on whether on the left of the pivot value
1924 % or on the right, or is a pivot value, or a raised pivot during selection phase.
1925
1926 \def\DecoLEFT #1{%
1927   \xintFor* ##1 in {#1} \do
1928     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
1929 }
1930
1931 \def\DecoINERT #1{%
1932   \xintFor* ##1 in {#1} \do
1933     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
1934 }
1935
1936 \def\DecoRIGHT #1{%
1937   \xintFor* ##1 in {#1} \do
1938     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
1939 }
1940
1941 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
1942   \xintFor* ##1 in {#1} \do

```



```

1943     {\stepcounter{cellcount}}%
1944     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
1945 }
1946
1947 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
1948     \xintFor* ##1 in {#1} \do
1949     {\stepcounter{cellcount}}%
1950     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
1951 }
1952
1953 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
1954     \xintFor* ##1 in {#1} \do
1955     {\stepcounter{cellcount}}%
1956     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
1957 }
1958
1959 %-----
1960 % SECOND PART: the actual sorting routines.
1961
1962 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
1963 \def\QS@sort@b #1{\ifcase #1
1964     \expandafter\QS@sort@empty
1965     \or\expandafter\QS@sort@single
1966     \else\expandafter\QS@sort@c
1967     \fi
1968 }%
1969 \def\QS@sort@empty #1{}
1970 \def\QS@sort@single #1{\QS@Ir {#1}}
1971
1972 % This step is to pick the last as pivot.
1973 \def\QS@sort@c #1%
1974     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
1975
1976 % Here \QSLr, \QS@Ir, \QSR have been let to \relax.
1977 % The trick with \xintApplyUnbraced is that for example when selecting
1978 % the elements smaller than pivot, if we had been using \xintApply we
1979 % would have had at the minimum an empty brace pair. Thus we use the
1980 % "unbraced" variant, but then the \QS@select@smaller has added in
1981 % anticipation a level of braces.
1982 \def\QS@sort@d #1#2{%
1983     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
1984     \QS@Ir {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
1985     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
1986 }%
1987 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
1988 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
1989 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
1990
1991 %
1992 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
1993 %
1994 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
1995 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
1996 % latter must handle correctly an empty argument.
1997
1998 %-----
1999 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QS@initialize.
2000
2001 % This draws all with suitable highlighting for the newly chosen pivots
2002 % (which will be shown raised)

```

```

2003 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2004             \let\QSIr\DecoINERT
2005             \let\QSIrr\DecoINERT
2006             \let\QSRr\DecoRIGHTwithPivot
2007 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2008             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2009             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2010 }
2011
2012 % This sorts and then draws, showing where the pivot chosen in the previous
2013 % step go. Next time they will have become "inert". If pivotcount is still at
2014 % zero on exit from \QSpivotStep, then this is the signal to stop before
2015 % executing \QSsortStep.
2016 \def\QSsortStep {\def\QSLr {\noexpand\QS@sort@a}%
2017             \def\QSRr {\noexpand\QS@sort@a}%
2018             \def\QSIr {\noexpand\QSIrr}%
2019             \let\QSIrr\relax
2020             \edef\QS@list{\QS@list}%
2021             \let\QSLr\relax
2022             \let\QSRr\relax
2023             \let\QSIr\relax
2024             \edef\QS@list{\QS@list}%
2025             \let\QSLr\DecoLEFT
2026             \let\QSIr\DecoINERTwithPivot
2027             \let\QSIrr\DecoINERT
2028             \let\QSRr\DecoRIGHT
2029 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2030             \setcounter{cellcount}{0}%
2031             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2032 }
2033
2034 \def\QSinitialize #1{%
2035     % first, we convert the comma separated values into a list of braced items
2036     % we use an \edef, and anyhow many \edef's will be used later
2037     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2038     \let\QSRr\DecoRIGHT
2039     % The \QSRr marker mutated to draw the last element as
2040     % pivot and the earlier ones with the suitable style.
2041     %
2042     % The list of marked braced items \QS@list is used both for drawing
2043     % (as here) and for doing the exchange of elements during sort.
2044     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2045             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2046 }
2047

```

1.31 relationale-algebra.sty

```

2048 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2049 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]
2050 \RequirePackage{amsmath}
2051 \RequirePackage{amssymb}

```

```

    Zum Zeichnen von Operatorenbäumen verwenden wir TikZ
\begin{tikzpicture}
  \node
    (pi) {\pi_{\text{Kunde.Name, Kunde.Geburtsdatum}}};

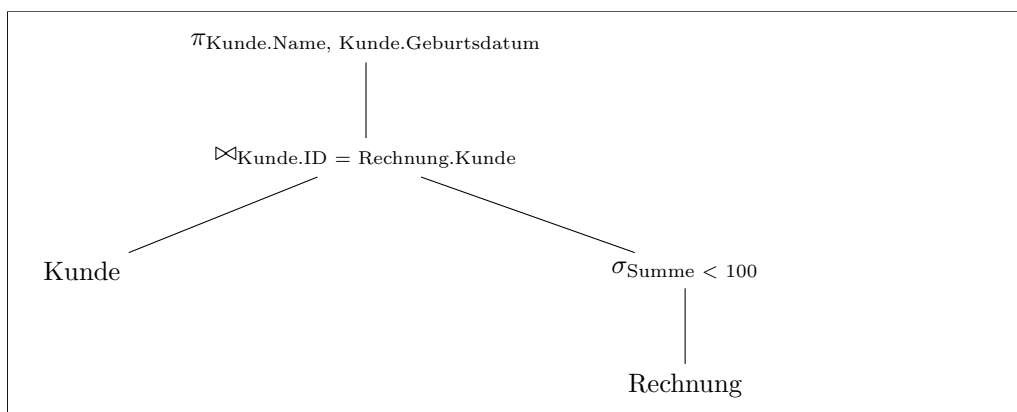
  \node[below=of pi]
    (theta join) {\bowtie_{\text{Kunde.ID = Rechnung.Kunde}}};
    edge (pi);

  \node[below left=of theta join]
    {Kunde}
    edge(theta join);

  \node[below right=of theta join]
    (sigma rechnung) {\sigma_{\text{Summe < 100}}};
    edge (theta join);

  \node[below=of sigma rechnung]
    {Rechnung}
    edge(sigma rechnung);
\end{tikzpicture}

```



```

2052 \RequirePackage{tikz}
2053 \usetikzlibrary{positioning}

    Privates Makros, das zwei Querstriche erzeugt.
2054 \def\o@join{\setbox0=\hbox{\bowtie}}%
2055   \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2056 }

```

```
\leftouterjoin A \leftouterjoin B: A ⋈ B
```

```
2057 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}
```

```
\rightouterjoin A \rightouterjoin B: A ⋉ B
```

```
2058 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}
```

```
\fullouterjoin A \fullouterjoin B: A ⋈⋉ B
```

```
2059 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
```

```
2060
```

1.32 rmodell.sty

```
2061 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2062 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01
2063 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2064 Datenbanken.]
2065 \RequirePackage{soul}
```

Let-Abkürzungen

```
\let\a=\liAttribut
\let\f=\liFremd
\let\p=\liPrimaer
\let\r=\liRelationMenge
```

`\liPrimaer` `\liPrimaer{text}`: Unterstreichung für den Primärschlüssel

```
2066 \def\liPrimaer#1{\ul{#1}}
```

`\liFremd` `\liFremd{text}`: Überstreichung für den Fremdschlüssel

```
2067 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

`liRmodell` `\begin{liRmodell}` `\end{liRmodell}`: Kleinere Schrift und Schreibmaschinenschrift.

```
2068 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2069 \ExplSyntaxOn
2070 \NewDocumentEnvironment { liRmodell }
2071 { +b }
2072 {
2073   \medskip
2074   {
2075     \linespread{2}
2076     \setlength{\parindent}{0pt}
2077     \li@Rmodell@Schrift#1
2078   }
2079   \medskip
2080 } {}
2081 \ExplSyntaxOff
```

`\liRelationMenge` **Let-Abkürzung:** `\let\r=\liRelationMenge`

`\liRelationMenge{name}{attribut, attribut}`: Umhüllen der Attribute mit geschweiften und dann eckigen Klammern.

```
2082 \def\liRelationMenge#1#2{
2083 \noindent
2084 #1 : \[ #2 ]\}
2085 \par
2086 }
```

`\liAttribut` **Let-Abkürzung:** `\let\a=\liAttribut`

`\liAttribut{text}`: Gleiche Schrift wie Umgebung `liRmodell`

```
2087 \def\liAttribut#1{{\li@Rmodell@Schrift#1}}
```

`liRelationenSchemaFormat` Dummy-Umgebung, zum Parsen durch Java gedacht.

```
\begin{liRelationenSchemaFormat}
Springer(Startnummer*, Nachname, Vorname, Geburtsdatum, Körpergröße)
Sprung(SID*, Beschreibung, Schwierigkeit)
springt(SID[Sprung], Startnummer[Springer], Durchgang)
\end{liRelationenSchemaFormat}
```

```
2088 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2089
```

1.33 sortieren.sty

```

2090 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2091 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
2092 für händisches Sortieren bzw. einen Schreibtischlauf setzen]
2093 \RequirePackage{tikz}
2094 \usetikzlibrary{shapes.multipart,positioning}

\liVertauschen \liVertauschen{1 2 >4 <3 5}: Setze ein Schaubild mit Hilfe von TikZ. < und > werden
dazu verwendet, um den Vertauschprozess zu visualisieren.

2095 \def\liVertauschen#1{
2096   \directlua{
2097     local sortieren = require('lehramt-informatik-sortieren')
2098     sortieren('#1')
2099   }
2100 }

2101 % https://tex.stackexchange.com/a/140895
2102 \RequirePackage{forest,xstring}
2103 \usetikzlibrary{calc}
2104
2105 \makeatletter
2106 \pgfmathdeclarefunction{strrepeat}{2}{%
2107   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2108   \let\pgfmathresult\pgfutil@empty
2109   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2110     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2111     \advance\pgfmath@count-1\relax
2112   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2113 \makeatother
2114
2115 \def\myNodes{}
2116
2117 \ExplSyntaxOn
2118 \newcommand*\sortList[1]{%
2119   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2120 \ExplSyntaxOff
2121
2122 \forestset{
2123   sort/.code={%
2124     \pgfmathparse{level()>\forestSortLevel}%
2125     \ifnum\pgfmathresult=0
2126       \StrSubstitute{\forestov{content}}{ }{,}{\myList}%
2127       \sortList\myList
2128       \StrSubstitute{\myList}{,}{ }{\myList}%
2129       \pgfmathparse{strrepeat("1",level())}%
2130       \xappto\myNodes{\noexpand\node at ($(\forestov{name})|-m)!-1!(\forestov{name}))$}
2131       (m\forestov{name}) {\myList}}%
2132     \pgfmathparse{level()=\forestSortLevel}%
2133     \ifnum\pgfmathresult=1
2134       \forestOget{\forestov{@first}}{name}\forestFirst
2135       \forestOget{\forestov{@last}}{name}\forestLast
2136       \xappto\myNodes{[<-]edge (\forestOv{\forestov{@first}}{name})
2137         \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2138     \fi
2139     \ifnum\forestov{@parent}=0\else
2140       \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2141     \fi
2142     \gappto\myNodes{;}%
2143   \fi}}
2144
2145 \forestset{sort level/.code=%
2146   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2147   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2148

```

1.34 spalten.sty

```
2149 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2150 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2151 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2152 realisiert werden kann.]
2153 \RequirePackage{multicol}
```

`\liSpaltenUmbruch` `\liSpaltenUmbruch`: Spezieller Spaltenumbruch, der den Inhalt mit Hilfe von `\vfill\strut` nach oben schiebt.

```
2154 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2155
```

1.35 struktogramm.sty

```
2156 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2157 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2158 Paket struktex zum Setzen von Struktogrammen]
2159 \RequirePackage{struktex}
2160
```

1.36 syntax.sty

```
2161 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2162 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2163 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2164 \RequirePackage{xparse}
```

Um das Paket `minted` benutzen zu können, muss `pygmentize` installiert sein. Außerdem müssen die TeX-Dateien mit der Kommandozeilen-Option `--shell-escape` kompiliert werden.

Let-Abkürzungen

```
\let\j=\liJavaCode
\let\s=\liSqlCode
```

```
2165 \ExplSyntaxOn
2166 \directlua{
2167   syntax = require('lehramt-informatik-syntax')
2168   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2169   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2170   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2171   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2172   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2173   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2174 }
2175 \RequirePackage{hyperref}
2176 \RequirePackage{minted}
2177 % pygmentize -L styles
2178 \usemintedstyle{colorful}
2179 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2180 %\AfterEndEnvironment{minted}{\end{mdframed}}
2181 %\setminted{breaklines=true,linenos}
2182 \setminted{
2183   breaklines=true,
2184   linenos,
2185   fontsize=\footnotesize,
2186 }
```

`\liJavaCode` Im Zeilenfluss einen kurzen Java-Code-Ausschnitt setzen.

Let-Abkürzung: `\let\j=\liJavaCode`

```
2187 \def\liJavaCode#1{\mintinline{java}|#1|}
```

`\liLatexCode` Im Zeilenfluss einen kurzen L^AT_EX-Code-Ausschnitt setzen.

```
2188 \def\liLatexCode#1{\mintinline{latex}|#1|}
```

```
2189 \def\li@GithubLink#1#2{
2190   \begin{flushright}
2191     \tiny
2192     Code-Beispiel~auf~Github~ansehen:~
2193     \href{#1}{\nolinkurl{#2}}
2194   \end{flushright}
2195 }
```

`\liJavaDatei` Eine komplette Java-Datei einbinden, die Verzeichnis `./Code/src/main/java/org/bschlangaul` liegt.

```
2196 \NewDocumentCommand{\liJavaDatei}{ O{firstline=3} m }{
2197   \inputminted[#1]{java}{
2198     \directlua{
2199       syntax.drucke_absoluten_pfad('#2', false)
2200     }
2201   }
2202   \li@GithubLink
```



```

2203     {\directlua{syntax.drucke_github_url('#2', false)}}
2204     {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2205 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2206 \NewDocumentCommand{\liJavaTestDatei}{ 0{firstline=3} m }{
2207   \inputminted[#1]{java}{
2208     \directlua{
2209       syntax.drucke_absoluten_pfad('#2', true)
2210     }
2211   }
2212   \li@GithubLink
2213   {\directlua{syntax.drucke_github_url('#2', true)}}
2214   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2215 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei(\[.
\liJavaExamen$1{$2}{$3}{$4}{$5}
2216 \NewDocumentCommand{\liJavaExamen}{ 0{firstline=3} m m m m }{
2217   \inputminted[#1]{java}{
2218     \directlua{
2219       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2220     }
2221   }
2222
2223   \li@GithubLink
2224   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2225   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2226 }

\liAssemblerCode
2227 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2228 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2229   \inputminted{asm}{#1}
2230 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2231 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2232   \inputminted{componentpascal}{#1}
2233 }

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2234 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2235 \NewDocumentCommand{\liHaskellDatei}{ m }{
2236   \inputminted{haskell}{#1}
2237 }

2238 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2239 \def\liSqlCode#1{\mintinline{sql}|#1|}
2240

```

1.37 syntaxbaum.sty

```
2241 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2242 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2243 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
2244 \RequirePackage{tikz-qtrees}
2245
2246 \tikzset{li parsetree/.style={
2247     every internal node/.style={
2248         draw,circle
2249     },
2250     every leaf node/.style={
2251         draw,rectangle
2252     },
2253 }
2254 }
2255
```

1.38 synthese-algorithmus.sty

```

2256 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2257 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2258 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2259 Relation in die 3. Normalform]

2260 \liLadePakete{normalformen,mathe,typographie}
2261 \ExplSyntaxOn

\let\schrittE=\liSyntheseUeberErklaerung
\begin{enumerate}
\item \schrittE{1}

\begin{enumerate}
\item \schrittE{1-1}
\item \schrittE{1-2}
\item \schrittE{1-3}
\item \schrittE{1-4}
\end{enumerate}

\item \schrittE{2}
\item \schrittE{3}
\item \schrittE{4}
\end{enumerate}

```

1. Kanonische Überdeckung

— Die kanonische Überdeckung - also die kleinst mögliche noch äquivalente Menge von funktionalen Abhängigkeiten kann in vier Schritten erreicht werden. —

(a) Linksreduktion

— Führe für jede funktionale Abhängigkeit $\alpha \rightarrow \beta \in F$ die Linksreduktion durch, überprüfe also für alle $A \in \alpha$, ob A überflüssig ist, d. h. ob $\beta \subseteq \text{AttrHülle}(F, \alpha - A)$.

(b) Rechtsreduktion

— Führe für jede (verbliebene) funktionale Abhängigkeit $\alpha \rightarrow \beta$ die Rechtsreduktion durch, überprüfe also für alle $B \in \beta$, ob $B \in \text{AttrHülle}(F - (\alpha \rightarrow \beta) \cup (\alpha \rightarrow (\beta - B)), \alpha)$ gilt. In diesem Fall ist B auf der rechten Seite überflüssig und kann eliminiert werden, d. h. $\alpha \rightarrow \beta$ wird durch $\alpha \rightarrow (\beta - B)$ ersetzt. —

(c) Löschen leerer Klauseln

— Entferne die funktionalen Abhängigkeiten der Form $\alpha \rightarrow \emptyset$, die im 2. Schritt möglicherweise entstanden sind. —

(d) Vereinigung

— Fasse mittels der Vereinigungsregel funktionale Abhängigkeiten der Form $\alpha \rightarrow \beta_1, \dots, \alpha \rightarrow \beta_n$, so dass $\alpha \rightarrow \beta_1 \cup \dots \cup \beta_n$ verbleibt. —

2. Relationsschemata formen

— Erzeuge für jede funktionale Abhängigkeit $\alpha \rightarrow \beta \in F_c$ ein Relationenschema $\mathcal{R}_\alpha := \alpha \cup \beta$. —

3. Schlüssel hinzufügen

— Falls eines der in Schritt 2. erzeugten Schemata \mathcal{R}_α einen Schlüsselkandidaten von \mathcal{R} bezüglich F_c enthält, sind wir fertig, sonst wähle einen Schlüsselkandidaten $\mathcal{K} \subseteq \mathcal{R}$ aus und definiere folgendes zusätzliche Schema: $\mathcal{R}_\mathcal{K} := \mathcal{K}$ und $\mathcal{F}_\mathcal{K} := \emptyset$ —

4. Entfernung überflüssiger Teilschemata

— Eliminiere diejenigen Schemata \mathcal{R}_α , die in einem anderen Relationenschema $\mathcal{R}_{\alpha'}$ enthalten sind, d. h. $\mathcal{R}_\alpha \subseteq \mathcal{R}_{\alpha'}$. —

\liSyntheseUeberschrift Let-Abkürzung: \let\schrift=\liSyntheseUeberschrift

```

2262 \def\liSyntheseUeberschrift#1{
2263   {
2264     \bfseries
2265     \sffamily
2266     \str_case:nn {#1} {
2267       {1} {Kanonische-Überdeckung}
2268       {1-1} {Linksreduktion}
2269       {1-2} {Rechtsreduktion}
2270       {1-3} {Löschen-leerer-Klauseln}
2271       {1-4} {Vereinigung}
2272       {2} {Relationsschemata-formen}
2273       {3} {Schlüssel-hinzufügen}
2274       {4} {Entfernung-überflüssiger-Teilschemata}
2275     }
2276   }
2277 }

```

\liSyntheseErklaerung Let-Abkürzung: \let\erklaerung=\liSyntheseErklaerung

```

2278 \def\li@synthese@erklaerung@texte#1{
2279   \str_case:nn {#1} {
2280     {1} {
2281       Die-kanonische-Überdeckung---also-die-kleinst-mögliche-noch-
2282       äquivalente-Menge-von-funktionalen-Abhängigkeiten-kann-in-vier-
2283       Schritten-erreicht-werden.
2284     }
2285     {1-1} {
2286       Führe-für-jede-funktionale-Anhängigkeit-
2287       $\alpha\rightarrow\beta$-in-$F$-die-Linksreduktion-durch,-
2288       überprüfe-also-für-alle-
2289       $A\in\alpha$,~ob-$A$-überflüssig-ist,-d.h.-ob-
2290       $\beta\subseteqq\liAttributHuelle{F,\alpha\cup A}$.
2291     }
2292     {1-2} {
2293       Führe-für-jede-(verbliebene)-funktionale-Abhängigkeit-$\alpha\rightarrow\beta$-die-Rechtsreduktion-durch,-überprüfe-also-für-
2294       alle-$B\in\beta$,~ob-$B\in\liAttributHuelle{F,\alpha\rightarrow\beta\cup B}$,-
2295       $\alpha\rightarrow\beta\cup B$-gilt.-In-diesem-Fall-ist-$B$-auf-der-rechten-Seite-
2296       überflüssig-und-kann-eliminiert-werden,-\dh-$\alpha\rightarrow\beta$-wird-durch-$\alpha\rightarrow\beta\cup B$-
2297       ersetzt.
2298     }
2299     {1-3} {
2300       Entferne-die-funktionalen-Abhängigkeiten-der-Form-$\alpha\rightarrow\emptyset$,~die-im-2.-Schritt-möglicherweise-
2301       entstanden-sind.
2302     }
2303     {1-4} {
2304       Fasse-mittels-der-Vereinigungsregel-funktionale-Abhängigkeiten-
2305       der-Form-$\alpha\rightarrow\beta\sb{1},\dots,\alpha\rightarrow\beta\sb{n}$,-so-dass-$\alpha\rightarrow\beta\sb{1}\cup\dots\cup\beta\sb{n}$-verbleibt.
2306     }
2307     {2} {
2308       % Kemper Seite 197
2309       Erzeuge-für-jede-funktionale-Abhängigkeit-$\alpha\rightarrow\beta\in F\sb{c}$-ein-Relationenschema-$\mathcal{R}\sb{\alpha}$-
2310       :=$\alpha\cup\beta$.
2311     }
2312     {3} {
2313       Falls-eines-der-in-Schritt-2.-erzeugten-Schemata-$R\sb{\alpha}$-
2314       einen-Schlüsselkandidaten-von-$\mathcal{R}$-bezüglich-$F\sb{c}$-

```

```

2322 enthält,~sind-wir~fertig,~sonst~wähle-einen-Schlüsselkandidaten-
2323  $\mathcal{K} \sim \text{subseq} \mathcal{R}$ ~aus-und-definiere-folgendes-
2324 zusätzliche-Schema:~ $\mathcal{R} \setminus \mathcal{K} \sim \mathcal{K}$ ~und- $\mathcal{F} \setminus \mathcal{K} \sim \emptyset$ 
2325 }
2326 {4} {
2327 Eliminiere-diejenigen-Schemata- $\mathcal{R} \setminus \alpha$ ~,~die-in-einem-
2328 anderen-Relationenschema- $\mathcal{R} \setminus \alpha'$ ~-enthalten-sind,~d.h.~
2329  $\mathcal{R} \setminus \alpha \sim \text{subseq} \mathcal{R} \setminus \alpha'$ ~.
2330 }
2331 }
2332 }
2333 }
2334 \def\liSyntheseErklaerung#1{
2335 {
2336 \itshape
2337 \footnotesize
2338 \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
2339 }
2340 }

```

\liSyntheseUeberErklaerung **Let-Abkürzung:** \let\schrittE=\liSyntheseUeberErklaerung

```

2341 \def\liSyntheseUeberErklaerung#1{
2342 \liSyntheseUeberschrift{#1}\par
2343 \liSyntheseErklaerung{#1}
2344 }

2345 \ExplSyntaxOff
2346

```

1.39 tabelle.sty

2347 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2348 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]

2349 \RequirePackage{tabularx}

2350

1.40 typographie.sty

```
2351 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2352 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
2353 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
2354 formatierung.sty definiert.]
```

```
2355 \ExplSyntaxOn
```

Mit dem Packet wasysym gab es Unverträglichkeiten, deshalb verwenden wir fontawesome.

```
2356 \RequirePackage{fontawesome}
```

```
\liErledigt \liErledigt: ☑
```

```
2357 \let\liErledigt=\faCheckSquareO
```

```
\liNichtsZuTun \liNichtsZuTun: ☑ Nichts zu tun
```

```
2358 \def\liNichtsZuTun{\faCheckSquareO{~Nichts~zu~tun}}
```

```
\liParagraphMitLinien \liParagraphMitLinien{Lorem ipsum...}:
```

— Lorem ipsum dolor sit, ipsum dolor sit, ipsum dolor sit, ipsum dolor sit, ipsum dolor sit, ipsum dolor sit —

```
2359 \def\liParagraphMitLinien#1{
```

```
2360 \noindent
```

```
2361 \vrule height 2pt depth -1.6pt width 0.4cm
```

```
2362 \enspace
```

```
2363 #1
```

```
2364 \enspace
```

```
2365 \leaders\vrule height 2pt depth -1.6pt \hfill \null
```

```
2366 \par
```

```
2367 \medskip
```

```
2368 }
```

```
2369 \ExplSyntaxOff
```

```
2370
```

1.41 uml.sty

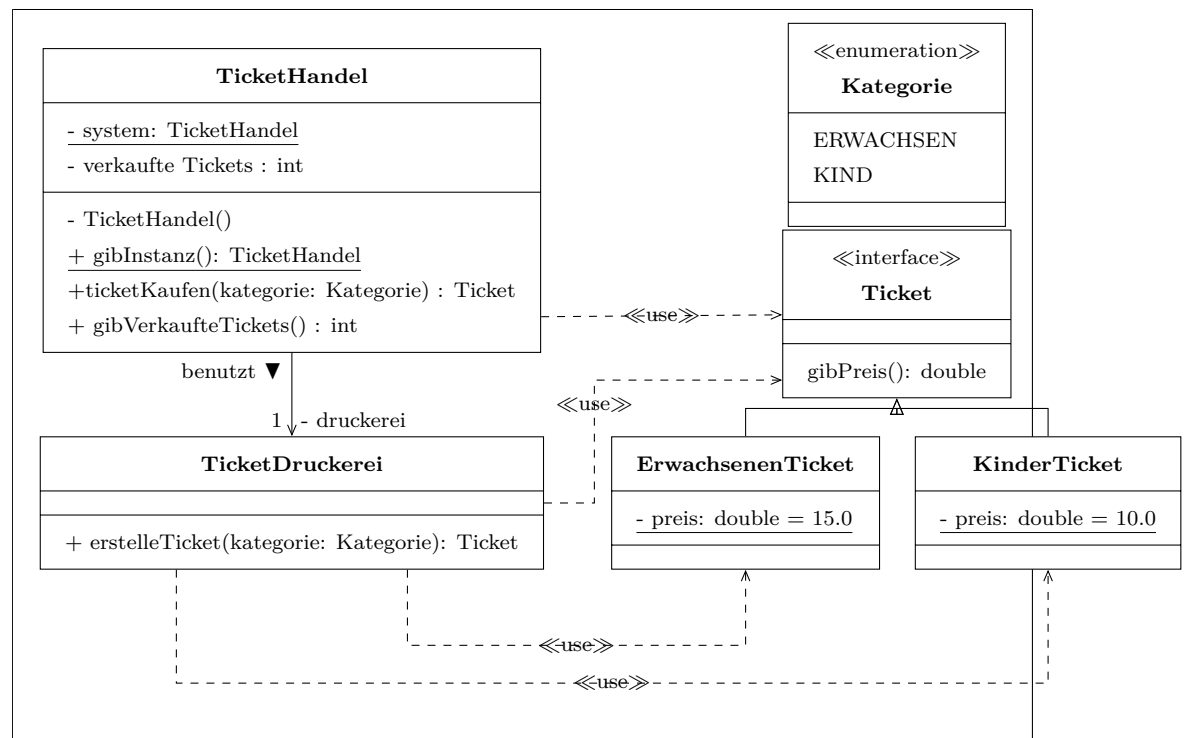
```

2371 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2372 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
2373 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
2374 Erweiterung bereitstellt]

2375 \RequirePackage{tikz-uml}
2376 \RequirePackage{tikz-uml-activity}
2377 % Not compatible with wasysym
2378 %\RequirePackage{mathabx}
2379 \RequirePackage{wasysym}
2380 \usetikzlibrary{positioning}

2381 \tikzumlset{
2382   fill class=white!0,
2383   font=\footnotesize,
2384   fill object=white!0,
2385   fill note=white!0,
2386   fill state=white!0,
2387   % Use case
2388   fill usecase=white!0,
2389   fill system=white!0,
2390 }

```



\liUmlLeserichtung

```

2391 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
2392   \def\@liDirLeft{}
2393   \def\@liDirRight{ \RIGHTarrow}
2394   \pgfkeys{/lese/dir/.is choice}
2395   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
2396   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
2397   \pgfkeys{/lese/dir/left/.code={\@liDirRight{}\def\@liDirLeft{ \LEFTarrow }}}
2398   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
2399
2400   \def\@liPos{above}
2401   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
2402
2403   \def\@liDistance{0cm}
2404   \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}

```



```

2405
2406 \pgfkeys{/lese/.cd,#1}
2407
2408 \node[\@liPos = \@liDistance of #2-middle] {
2409   \@liDirLeft{\footnotesize#2}\@liDirRight
2410 };
2411 }
2412

```

1.42 vollstaendige-induktion.sty

```
2413 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2414 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01
2415 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
2416 Überschriften für die einzelnen Schritte]
```

Lade häufig benötigte Pakete

```
2417 \RequirePackage{lehramt-informatik-typographie}
2418 \RequirePackage{lehramt-informatik-mathe}
2419 \RequirePackage{lehramt-informatik-syntax}
2420 \ExplSyntaxOn
```

`\liInduktionAnfang`

```
2421 \def\liInduktionAnfang{
2422   \liPseudoUeberschrift{Induktionsanfang}
2423
2424   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
2425   \liParagraphMitLinien{
2426     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
2427   }
2428 }
```

`\liInduktionVoraussetzung`

```
2429 \def\liInduktionVoraussetzung{
2430   \liPseudoUeberschrift{Induktionsvoraussetzung}
2431
2432   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
2433   \liParagraphMitLinien{
2434     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ $.
2435   }
2436 }
```

`\liInduktionSchritt`

```
2437 \def\liInduktionSchritt{
2438   \liPseudoUeberschrift{Induktionsschritt}
2439
2440   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
2441   \liParagraphMitLinien{
2442     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
2443     auch~ $A(n=k+1)$ ~wahr~sein~muss.
2444   }
2445 }

2446 \ExplSyntaxOff
2447
```

1.43 wasserfall.sty

```
2448 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2449 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
2450 \RequirePackage{tikz}
2451 \tikzset{wasserfall/.style={
2452   >=stealth,
2453   node distance = 2mm and -8mm,
2454   start chain = A going below right,
2455   every node/.style = {
2456     draw,
2457     text width=24mm,
2458     minimum height=12mm,
2459     align=center,
2460     inner sep=1mm,
2461     fill=white,
2462     drop shadow={fill=black},
2463     on chain=A
2464   },
2465 }}
2466 \usetikzlibrary{chains,positioning,shadows}
2467
```

1.44 wpkalkuel.sty

2468 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2469 \ProvidesPackage{lehramt-informatik-wpkalkuel}[2020/06/13]

Let-Abkürzungen

\let\wp=\liWpKalkuel

\let\equivalent=\liWpEquivalent

\let\erklaerung=\liWpErklaerung

2470 \RequirePackage{amsmath}

2471 \ExplSyntaxOn

\liWpKalkuel **Let-Abkürzung:** \let\wp=\liWpKalkuel

2472 \def\liWpKalkuelOhneMathe#1#2{

2473 \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)

2474 }

2475 \def\liWpKalkuel#1#2{

2476 \ifmmode

2477 \liWpKalkuelOhneMathe{#1}{#2}

2478 \else

2479 \$\liWpKalkuelOhneMathe{#1}{#2}\$

2480 \fi

2481 }

\MatheEnv

2482 \def\MatheEnv#1{

2483 \medskip

2484

2485 \hspace{1em}#1

2486

2487 \medskip

2488 }

\Mathe

2489 \def\Mathe#1{

2490 \MatheEnv{#1\$}

2491 }

\liWpEquivalent **Let-Abkürzung:** \let\equivalent=\liWpEquivalent

2492 \def\liWpEquivalent#1{

2493 \MatheEnv{\$\equiv\$\hspace{1em}#1\$}

2494 }

\liWpErklaerung **Let-Abkürzung:** \let\erklaerung=\liWpErklaerung

2495 \newlength{@Skip@Erklaerung@Reset}

2496 \def\liWpErklaerung#1{

2497 \setlength{@Skip@Erklaerung@Reset}{\leftskip}

2498 \setlength{\leftskip}{0.5cm}

2499

2500 \par

2501 \noindent

2502 {

2503 \scriptsize

2504 #1

2505 }

2506 \par

2507

2508 \setlength{\leftskip}{@Skip@Erklaerung@Reset}

2509 }

\liWpErklaerungVerzweigung

```

2510 \def\liWpErklaerungVerzweigung{
2511   $\liWpKalkuelOhneMathe{if~\{-b~\}~then~\{-a1~\}~else~\{-a2~\}}{Q}
2512   \equiv
2513   (b \land \liWpKalkuelOhneMathe{a1}{Q})
2514   \lor
2515   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
2516 }

2517 \ExplSyntaxOff

2518

```

2 Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

Symbols	\alpha 2287, 2289, 2290, 2293, 2295, 2296, 2297, 2298, 2299, 2303, 2309, 2310, 2315, 2316, 2317, 2320, 2328, 2329, 2330	\boxtimes 463
\# 109		C
\, 331, 386, 1037, 1039, 1525, 1526, 1527, 1782		\c 1183, 1184
\@Skip@Erklaerung@Reset ... 2495, 2497, 2508		\cdot 1569, 1622, 1633
\@afterheading 1492	\arabic 1100, 1928, 1933, 1938, 1944, 1950, 1956	\centerline 1214, 2007, 2029, 2044
\@afterindentfalse . 1492	\arraystretch 1699	\chapter 1090, 1091
\@liDirLeft 2392, 2397, 2409		\char 1399
\@liDirRight 2393, 2395, 2396, 2397, 2398, 2409	B	\clearpage 1393
\@liDistance 2403, 2404, 2408	\BeforeBeginEnvironment 2179	\cline 596
\@liPos .. 2400, 2401, 2408	\begin .. 614, 656, 671, 695, 719, 753, 772, 782, 790, 810, 838, 855, 876, 893, 908, 1016, 1020, 1106, 1181, 1209, 1218, 1225, 1333, 1391, 1443, 1448, 1456, 1481, 1486, 1497, 1507, 1511, 1594, 1598, 1614, 1702, 1703, 1753, 2009, 2031, 2045, 2179, 2190	\clist 226, 270, 271, 284, 288, 2119
\ 596, 616, 617, 620, 621, 624, 625, 697, 698, 699, 774, 776, 794, 803, 844, 857, 858, 859, 864, 865, 866, 882, 1399, 1704, 1761, 1764	\begingroup 1392, 1751, 2107	\columnbreak 2154
\{ 207, 972, 982, 994, 995, 1000, 1014, 1036, 1255, 1749, 1772, 2084, 2511	\beschriftung 1423, 1427, 1431, 1435, 1439, 1441	\cs 287, 306, 330, 331, 368, 380, 1475
\} 207, 972, 982, 994, 995, 1002, 1020, 1040, 1256, 1749, 1776, 2084, 2511	\beta 2287, 2290, 2294, 2295, 2296, 2299, 2309, 2310, 2311, 2316, 2317	\csname 1168, 1171
_ 27, 35, 44, 46, 287, 306, 330, 331, 345, 346, 352, 355, 358, 368, 380	\bf 1906, 1907, 1908	\cup 995, 2296, 2310, 2317
A	\bfseries .. 475, 1090, 1092, 1906, 1912, 1914, 1916, 1917, 2264	D
\addbibresource 1369, 1370, 1371, 1372, 1373, 1374, 1375, 1376, 1377, 1378	\bigskip 48, 364, 598, 603, 1401, 1677	\DeclareMathSymbol 1647, 1648
\advance 2111	\bool 309, 332	\DecoINERT 1931, 2004, 2005, 2027
\AfterEndEnvironment 2180	\bowtie 2054, 2057, 2058, 2059	\DecoINERTwithPivot 1947, 2026
\allsectionsfont ... 1086	\Box 159	\DecoLEFT 1926, 2025
\Alph 1100		\DecoLEFTwithPivot 1941, 2003
\alph 1100, 1101		\DecoRIGHT 1936, 2028, 2038
		\DecoRIGHTwithPivot 1953, 2006
		\definecolor 1088
		\delta 65, 107, 165, 207, 986
		\dh 1527, 2298
		\directlua 58, 137, 195, 200, 966, 981, 1001, 1009, 1017, 1023, 1734, 1739, 1767, 1774, 1779, 2096, 2166, 2198, 2203, 2204, 2208, 2213, 2214, 2218, 2224, 2225

<code>\do</code>	1927, 1932, 1937, 1942, 1948, 1954	202, 393, 528, 550, 565, 960, 1042, 1077, 1353, 1517, 1592, 1727, 1874, 2081, 2120, 2238, 2345, 2369, 2446, 2517	311, 312, 313, 316, 318, 319, 320, 321, 322, 323, 334, 335, 336, 337, 345, 346, 348, 354, 355, 357, 358, 360, 361, 369, 374, 376, 381, 383, 387
<code>\dots</code>	506, 510, 1255, 2309, 2310		
<code>\DOWNarrow</code>	2396		
<code>\draw</code> 1188, 1191, 1194, 1846			
E			
<code>\edef</code>	1179, 2020, 2024, 2036, 2037	102, 135, 160, 193, 198, 223, 470, 534, 551, 933, 1032, 1054, 1346, 1418, 1531, 1709, 1855, 2069, 2117, 2165, 2261, 2355, 2420, 2471	<code>\Gamma</code> 106, 164, 207, 995 <code>\gappto</code> 2142 <code>\geq</code> 1262
<code>\else</code> 570, 578, 586, 976, 990, 1028, 1439, 1542, 1552, 1562, 1576, 1745, 1841, 1966, 2137, 2139, 2478			
<code>\emph</code>	936, 1236, 1265, 1267, 1398		
<code>\empty</code>	1439		
<code>\emptyset</code> 1691, 2304, 2325			
<code>\end</code>	653, 667, 692, 716, 750, 769, 779, 787, 806, 835, 852, 873, 889, 905, 923, 1018, 1020, 1108, 1197, 1221, 1222, 1229, 1338, 1396, 1445, 1453, 1472, 1488, 1489, 1503, 1514, 1515, 1596, 1609, 1634, 1706, 1707, 1755, 2009, 2031, 2045, 2180, 2194	F <code>\faCheckSquareO</code> 2357, 2358 <code>\faCircleThin</code> 952 <code>\faGg</code> 946 <code>\fancyfoot</code> 1112, 1113, 1114 <code>\fancyhead</code> 1111 <code>\faSquareO</code> 940 <code>\fi</code> 572, 580, 588, 978, 992, 1030, 1442, 1544, 1554, 1564, 1578, 1747, 1843, 1967, 2137, 2138, 2141, 2143, 2480 <code>\fontspec</code> 1086 <code>\footcite</code> 668, 690, 726, 749, 768, 807, 834, 1234, 1237, 1244, 1249, 1254, 1258, 1264, 1269, 1358, 1610, 1611, 1714 <code>\footnote</code> 1519, 1523 <code>\footnotesize</code> 147, 341, 424, 520, 957, 1414, 1467, 1482, 1498, 1658, 1752, 1859, 1868, 2068, 2185, 2337, 2383, 2409 <code>\footrulewidth</code> 1117 <code>\foreach</code> 1183, 1186, 1193 <code>\forestFirst</code> 2134, 2137 <code>\forestLast</code> 2135, 2137 <code>\forestOget</code> 2134, 2135 <code>\forestOnes</code> 2147 <code>\forestOv</code> 2136, 2137, 2140 <code>\forestov</code> 2126, 2130, 2131, 2134, 2135, 2136, 2137, 2139, 2140 <code>\forestset</code> 2122, 2145 <code>\forestSortLevel</code> 2124, 2132, 2146, 2147 <code>\frac</code> 1571, 1602, 1633 <code>\fullouterjoin</code> 2059	H <code>\hbox</code> 2054 <code>\headrulewidth</code> 1116 <code>\hfill</code> 1618, 1624, 1629, 2365 <code>\hinweis</code> 1414 <code>\hline</code> 1704 <code>\href</code> 370, 1050, 1523, 2193 <code>\hspace</code> 1849, 2485, 2493 <code>\ht</code> 2055
<code>\endcsname</code>	1168, 1171		
<code>\endgroup</code> 1395, 1756, 2112			
<code>\enspace</code>	2362, 2364		
environments:			
<code>liAdditum</code>	1446		
<code>liAHuelle</code>	1750		
<code>liAntwort</code>	1417		
<code>liDiagramm</code>	1505		
<code>liEinbettung</code>	1416		
<code>liExkurs</code>	1454		
<code>liGraphenFormat</code>	1160		
<code>liKasten</code>	1105		
<code>liKontrollflussgraph</code>	1332		
<code>liLernkartei</code>	1495		
<code>liProduktionsRegeln</code>	1010		
<code>liProjektSprache</code>	1415		
<code>liQuellen</code>	1475		
<code>liRelationenSchemaFormat</code>	2088		
<code>liRmodell</code>	2068		
<code>liUebergangsTabelle</code>	1699		
<code>\equiv</code>	2493, 2512		
<code>\verzeuge@tiefgestellt</code>	981, 982, 986		
<code>\expandafter</code>	1168, 1962, 1964, 1965, 1966, 1974, 2110		
<code>\ExplSyntaxOff</code>	50, 92, 134, 139, 192, 197,		
<code>\g</code>	29, 37, 270, 271, 284, 288, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 307,		
<code>\k</code>	1193		
<code>\keys</code>	31, 70, 82, 112, 122, 170, 180, 292, 538, 542, 556, 561, 1061, 1068		
<code>\i</code>	1193, 1194		
<code>\ifcase</code>	1963		
<code>\ifmode</code> 568, 576, 584, 974, 988, 1026, 1540, 1550, 1560, 1574, 1743, 1839, 2476			
<code>\IfNoValueTF</code>			
<code>\ifnum</code>	1890, 2109, 2125, 2133, 2139		
<code>\ifx</code>	1439, 2137		
<code>\in</code> 494, 600, 1262, 1584, 1587, 1590, 1616, 1622, 1627, 2287, 2289, 2295, 2316, 2434			
<code>\inhaltsverzeichnis</code>	1390		
<code>\input</code>	4, 7, 10, 13, 16, 394		
<code>\inputminted</code> 2197, 2207, 2217, 2229, 2232, 2236			
<code>\int</code>	2119		
<code>\item</code>	463, 464, 673, 677, 682, 687, 720, 729, 734, 742, 783, 811, 816, 823, 831, 909, 914, 919, 1219, 1220, 1475, 1479, 1599, 1602, 1606, 1615, 1621, 1626		
<code>\itshape</code>	519, 2336		
J			
<code>\j</code> 1183, 1184, 1186, 1187, 1188, 1193, 1194, 1195			
K			

L	
\l	63, 64, 65, 66, 67, 68, 71, 72, 73, 74, 75, 77, 79, 84, 85, 86, 87, 88, 89, 104, 105, 106, 107, 108, 109, 110, 113, 114, 115, 116, 117, 118, 119, 125, 126, 127, 128, 129, 130, 131, 162, 163, 164, 165, 166, 167, 168, 171, 172, 173, 174, 175, 176, 177, 183, 184, 185, 186, 187, 188, 189, 536, 539, 544, 545, 548, 553, 554, 557, 558, 563, 1056, 1057, 1058, 1059, 1062, 1063, 1064, 1065, 1071, 1072, 1073, 1074, 1349, 1350, 1351, 1478, 1479, 1480, 1487
\labelenumi	1101
\labelenumii	1102
\labelitemi	1095
\labelitemii	1096
\labelitemiii	1097
\labelitemiv	1098
\land	2513, 2515
\LARGE	1090
\large	1214
\leaders	2365
\left	1534
\LEFTarrow	2397
\leftarrow	582
\leftouterjoin	2057
\leftskip	2497, 2498, 2508
\LehramtInformatikGitBranch	373, 2173
\LehramtInformatikGithubCodeRepo	2172
\LehramtInformatikGithubDomaine	2169
\LehramtInformatikGithubRawDomaine	371, 2170
\LehramtInformatikGithubTexRepo	372, 2171
\LehramtInformatikRepository	4, 7, 10, 13, 16, 1369, 1370, 1371, 1372, 1373, 1374, 1375, 1376, 1377, 1378, 2168
\leq	1633
\let	984, 985, 1393, 2003, 2004, 2005, 2006, 2019, 2021, 2022, 2023, 2025, 2026, 2027, 2028, 2038, 2108, 2146, 2147, 2357
\li@chomsky@erklaerung@texte	485, 521
\li@fussnote@text	1655, 1661, 1665, 1669, 1673
\li@GithubLink	2189, 2202, 2212, 2223
\li@mget	1170, 1174, 1194
\li@minc	1173, 1195
\li@mset	1167, 1175, 1184, 1187
\li@numdiscs	1179, 1188, 1194
\li@Rmodell@Schrift	2068, 2077, 2087
\li@sequence	1180, 1193
\li@synthese@erklaerung@texte	2278, 2338
\liAbleitung	1009
liAdditum (environment)	1446
liAHuelle (environment)	1750
\liAlphabet	994
liAntwort (environment)	1417
\liAnweisung	1340
\liAssemblerCode	2227
\liAssemblerDatei	2228
\liAttribut	2087
\liAttributHuelle	1741, 2290, 2295
\liAttributHuelleLinksReduktion	1758
\liAttributHuelleOhneMathe	1741, 1744, 1746, 1760
\liAttributMenge	1749, 1760, 1763
\liAufgabe	3
\liAufgabenTitel	23
\liAusdruck	1033
\liAutomat	61
\liAutomatenKante	93
\liBandAlphabet	995
\liBedingung	1341
\liBedingungDrei	1589, 1629
\liBedingungEins	1583, 1618
\liBedingungFalsch	1343
\liBedingungWahr	1342
\liBedingungZwei	1586, 1624
\liBeschriftung	1406
\liChomskyErklaerung	485, 526
\liChomskyUeberErklaerung	524
\liChomskyUeberschrift	473, 525
\liCpmEreignis	534
\liCpmFruehesterI	591
\liCpmSpaetesterI	590
\liCpmVon	574
\liCpmVonOhneMathe	574, 577, 579
\liCpmVonZu	566
\liCpmVonZuOhneMathe	566, 569, 571
\liCpmVorgang	551
\liCpmZu	582
\liCpmZuOhneMathe	582, 585, 587
liDiagramm (environment)	1505
liEinbettung (environment)	1416
\liEntwurfsAbstrakteFabrikUml	613
\liEntwurfsAdapterAkteure	670
\liEntwurfsAdapterUml	655
\liEntwurfsBeobachterAkteure	718
\liEntwurfsBeobachterUml	694
\liEntwurfsDekoriererUml	752
\liEntwurfsEinzelstueckAkteure	781
\liEntwurfsEinzelstueckUml	771
\liEntwurfsErbauerAkteure	809
\liEntwurfsErbauerUml	789
\liEntwurfsFabrikmethodeUml	837
\liEntwurfsKompositumUml	854
\liEntwurfsModellPraesentationSteuerungUml	875
\liEntwurfsZustandAkteure	907
\liEntwurfsZustandUml	892
\liEpsilon	980
\liErAttribute	936, 950, 952
\liErDatenbankName	955
\liErEntity	934, 938, 940
\liErledigt	2357
\liErMpAttribute	949
\liErMpEntity	937
\liErMpRelationship	943
\liErRelationship	935, 944, 946
\liExamensAufgabe	6
\liExamensAufgabeA	15
\liExamensAufgabeTA	12
\liExamensAufgabeTTA	9
liExkurs (environment)	1454
\liFalsch	464
\liFlaci	1043
\liFremd	2067
\liFunktionaleAbhaengigkeit	1766
\liFunktionaleAbhaengigkeiten	1769

<code>\liFussnote</code> ... 1654 , 1656	<code>\liOmegaOhneMathe</code> 1546 , 1551 , 1553	<code>\liStrich</code> 1206
<code>\liFussnoteDreiText</code> 1668 , 1686	<code>\liOOhneMathe</code> 1556 , 1561 , 1563	<code>\liSyntheseErklaerung</code> 2278 , 2343
<code>\liFussnoteEinsText</code> 1660 , 1680	<code>\liParagraphMitLinien</code> ... 521 , 1715 , 2338 , 2359 , 2425 , 2433 , 2441	<code>\liSyntheseUeberErklaerung</code> 2341
<code>\liFussnoteLink</code> 1522	<code>\liPetriErreichKnotenDrei</code> 1848	<code>\liSyntheseUeberschrift</code> 2262 , 2342
<code>\liFussnoten</code> 1676	<code>\liPetriErreichTransition</code> 1845	<code>\liT</code> 1566 , 1581
<code>\liFussnoteUrl</code> . 890 , 1518	<code>\liPetriSetzeSchluessel</code> 1790	<code>\liTeilen</code> 1738
<code>\liFussnoteVierText</code> 1672 , 1689	<code>\liPetriTransitionsName</code> 1837 , 1849	<code>\liLiteratur</code> 1357 , 1381
<code>\liFussnoteZweiText</code> 1664 , 1683	<code>\liPetriTransitionsNameOhneMathe</code> ... 1837 , 1840 , 1842	<code>\liTheta</code> 1536 , 1587
<code>\liGrammatik</code> 1054	<code>\liPetriTransPfeile</code> 1849	<code>\liThetaOhneMathe</code> 1536 , 1541 , 1543
<code>liGraphenFormat</code> (envi- ronment) 1160	<code>\liPolynomiellReduzierbar</code> 1224	<code>\liTOhneMathe</code> 1566 , 1575 , 1577
<code>\liHanoi</code> 1167	<code>\liPotenzmenge</code> 981 , 985 , 1860	<code>\liTuringKante</code> 203
<code>\liHaskellCode</code> 2234	<code>\liPotenzmengeOhneMathe</code> 982 , 983 , 984	<code>\liTuringLeerzeichen</code> 159 , 167
<code>\liHaskellDatei</code> 2235	<code>\liPrimaer</code> 2066	<code>\liTuringMaschine</code> ... 160
<code>\liInduktionAnfang</code> . 2421	<code>\liProblemBeschreibung</code> 1208	<code>\liTuringUeberfuehrung</code> 206
<code>\liInduktionSchritt</code> 2437	<code>\liProblemClique</code> ... 1231	<code>\liTuringUebergange</code> 198 , 204
<code>\liInduktionVoraussetzung</code> 2429	<code>\liProblemName</code> 1207 , 1214 , 1226 , 1228 , 1241 , 1252 , 1253 , 1261 , 1262	<code>\liTuringUebergangZelle</code> 193
<code>\liJavaCode</code> 2187	<code>\liProblemSat</code> 1260	<code>\liUeberfuehrungsFunktion</code> 986
<code>\liJavaDatei</code> 2196	<code>\liProblemSubsetSum</code> 1251 , 1260	<code>\liUeberfuehrungsFunktionOhneMathe</code> 986 , 989 , 991
<code>\liJavaExamen</code> 2216	<code>\liProblemVertexCover</code> 1231 , 1239	<code>liUebergangsTabelle</code> (environment) 1699
<code>\liJavaTestDatei</code> ... 2206	<code>\liProduktionen</code> 1022 , 1064	<code>\liUeberschriftDreiecksTabelle</code> 1709
<code>liKasten</code> (environment) 1105	<code>liProduktionsRegeln</code> (environment) 1010	<code>\liUmlLeserichtung</code> . 2391
<code>\liKellerAutomat</code> 102	<code>liProjektSprache</code> (envi- ronment) 1415	<code>\liVertauschen</code> 2095
<code>\liKellerKante</code> 140	<code>\liPseudoUeberschrift</code> 1400 , 1450 , 1451 , 1701 , 1711 , 2422 , 2430 , 2438	<code>\liWortInSprache</code> 597
<code>\liKellerUebergang</code> 135 , 141	<code>liQuellen</code> (environment) 1475	<code>\liWortNichtInSprache</code> 602
<code>\liKontrollCode</code> 1344	<code>\liRekursionsGleichung</code> 1580	<code>\liWpEquivalent</code> 2492
<code>liKontrollflussgraph</code> (environment) 1332	<code>\liRelation</code> 1778	<code>\liWpErklaerung</code> 2495
<code>\liKontrollKnotenPfad</code> 1346	<code>liRelationenSchemaFormat</code> (environment) 2088	<code>\liWpErklaerungVerzweigung</code> 2510
<code>\liKontrollTextzeileKnoten</code> 1345 , 1350	<code>\liRelationMenge</code> ... 2082	<code>\liWpKalkuel</code> 2472
<code>\liKurzeTabellenLinie</code> 596	<code>\liRichtig</code> 463	<code>\liWpKalkuelOhneMathe</code> 2472 , 2477 , 2479 , 2511 , 2513 , 2515
<code>\liLadeAllePakete</code> ... 228	<code>liRmodell</code> (environment) 2068	<code>\liZustandsBuchstabe</code> 996 , 1005 , 1007 , 1027 , 1029
<code>\liLadePakete</code> 54 , 57 , 224 , 229 , 472 , 533 , 1204 , 1653 , 1733 , 1854 , 2260	<code>\liRundeKlammer</code> . 1533 , 1537 , 1547 , 1557 , 1571	<code>\liZustandsBuchstabeGross</code> 997 , 1006 , 1008
<code>\liLatexCode</code> 2188	<code>\liSetzeAufgabenTitel</code> . 25	<code>\liZustandsmenge</code> 984
<code>\liLeereZelle</code> 1691	<code>\liSpaltenUmbruch</code> .. 2154	<code>\liZustandsmengeNr</code> 998 , 1869
<code>liLernkartei</code> (environ- ment) 1495	<code>\liSqlCode</code> 2239	<code>\liZustandsmengeNrGross</code> 1006
<code>\liMasterFaelle</code> 1613	<code>\listen@punkt</code> .. 1475 , 1487	<code>\liZustandsMengenSammlung</code> 1856
<code>\liMasterVariablen</code> . 1593		<code>\liZustandsMengenSammlungNr</code> 1865
<code>\liMenge</code> ... 71 , 72 , 74 , 113 , 114 , 115 , 119 , 171 , 172 , 173 , 177 , 972 , 1023 , 1062 , 1063		<code>\liZustandsmengeOhneMathe</code> 984
<code>\liMengeOhneMathe</code> 972 , 975 , 977		<code>\liZustandsname</code> 1007
<code>\liMinimierungErklaerung</code> 1713		
<code>\liMinispracheDatei</code> 2231		
<code>\linespread</code> 2075		
<code>\liNichtsZuTun</code> 2358		
<code>\liO</code> 1556 , 1584		
<code>\liOmega</code> 1546 , 1590		

<code>\liZustandsnameGross</code>	203, 224, 535, 552,	2108, 2110, 2112,
... 1008 , 1857 , 1866	597, 602, 1033,	2125, 2133, 2146, 2147
<code>\liZustandsnameTiefgestellt</code>	1055, 1224, 1347,	<code>\pgfutil@empty</code> 2108
... 1025	1359, 1518, 1522,	<code>\pgfutil@loop</code> 2109
<code>\liZustandsPaar</code> 1693	1769, 1778, 1845,	<code>\pgfutil@repeat</code> 2112
<code>\liZustandsPaarVariablenName</code>	2196, 2206, 2216,	<code>\preceq</code> 1227
... 1692 , 1695 , 1696	2228, 2231, 2235, 2391	<code>\prime</code> 1206
<code>\llap</code> 2055	<code>\NewDocumentEnvironment</code>	<code>\printbibliography</code> . 1381
<code>\log</code> 1584,	1010, 1105, 1160,	<code>\ProvidesPackage</code> 2, 20,
1587 , 1590 , 1616 , 1622	1332, 1415, 1416,	53, 221, 399, 460,
<code>\loop</code> 1888	1419, 1446, 1454,	467, 531, 594, 609,
<code>\lor</code> 2514	1476, 1495, 1505,	927, 963, 1080,
<code>\ltimes</code> 1640	1700, 1750, 2070, 2088	1120, 1129, 1134,
	<code>\newlength</code> 2495	1163, 1201, 1273,
M	<code>\node</code> 548,	1356, 1362, 1384,
<code>\makeatletter</code> . . 1491, 2105	1340, 1345, 1928,	1530, 1638, 1651,
<code>\makeatother</code> . . 1493, 2113	1933, 1938, 1944,	1730, 1786, 1852,
<code>\marginpar</code>	1950, 1956, 2130, 2408	1877, 1883, 2049,
. . 939, 945, 951, 1398	<code>\noexpand</code> 2016,	2062, 2091, 2150,
<code>\mathbb</code> 1262, 2434	2017, 2018, 2037, 2130	2157, 2162, 2242,
<code>\mathbin</code> . 2057, 2058, 2059	<code>\noindent</code> . . . 343, 599,	2257, 2348, 2352,
<code>\mathcal</code> . . 1557, 2316,	604, 1013, 1402,	2372, 2414, 2449, 2469
2321 , 2323 , 2324 , 2325	1404, 1408, 1412,	
<code>\Mathe</code> 2489	1440, 1468, 1470,	Q
<code>\MatheEnv</code> 2482 , 2490 , 2493	1483, 1499, 1501,	<code>\QS@list</code>
<code>\mathord</code> 1647, 1648	1509, 1679, 1682,	2009, 2020, 2024,
<code>\mdfsetup</code> 1104,	1685, 1688, 1771,	2031, 2037, 2042, 2045
1424 , 1428 , 1432 , 1436	1776, 2083, 2360, 2501	<code>\QS@select@equal</code>
<code>\medskip</code>	<code>\nolinkurl</code> 2193 1984, 1988
1216 , 1409 , 1411 ,	<code>\normalsize</code> 1092	<code>\QS@select@greater</code> . .
1485 , 1513 , 2073 ,	<code>\notin</code> 605 1985, 1989
2079 , 2367 , 2483 , 2487	<code>\null</code> 2365	<code>\QS@select@smaller</code> . .
<code>\memp</code> 1398		. . . 1980, 1983, 1987
<code>\mintinline</code> 2187,	O	<code>\QS@sort@a</code>
2188 , 2227 , 2234 , 2239	<code>\o@join</code>	1962, 1995, 2016, 2017
<code>\mkern</code> . . . 2057, 2058, 2059	2054, 2057, 2058, 2059	<code>\QS@sort@b</code> 1962, 1963
<code>\mlq</code> 1645, 1647	<code>\Omega</code> 1547	<code>\QS@sort@c</code> 1966, 1973
<code>\mrq</code> 1645, 1648	<code>\or</code> 1965	<code>\QS@sort@d</code> 1974, 1982
<code>\msg</code> 39, 391		<code>\QS@sort@empty</code> . 1964, 1969
<code>\myList</code>	P	<code>\QS@sort@single</code> 1965, 1970
2126 , 2127 , 2128 , 2131	<code>\pagestyle</code> 1115	<code>\QS@initialize</code>
<code>\myNodes</code> 2115,	<code>\par</code> 342, 363, 525, 1044,	. . . 1887, 1999, 2034
2130 , 2136 , 2140 , 2142	1052, 1407, 1469,	<code>\QS@Ir</code> . 1970, 1976, 1984,
N	1492, 1500, 1770,	2004, 2018, 2023, 2026
<code>\NeedsTeXFormat</code> . 1, 19,	1773, 1775, 2007,	<code>\QS@Irr</code> 2005, 2018, 2019, 2027
52, 220, 398, 459,	2029, 2044, 2085,	<code>\QSLr</code> 1976,
466, 530, 593, 608,	2342, 2366, 2500, 2506	1983, 1994, 1995,
926, 962, 1079,	<code>\paragraph</code> 1092	2003, 2016, 2021, 2025
1119, 1128, 1133,	<code>\parindent</code> 2076	<code>\QSPivotStep</code>
1162, 1200, 1272,	<code>\path</code> 94, 141, 204, 563	1889, 1999, 2003, 2014
1355, 1361, 1383,	<code>\pgfkeys</code> . . 1801, 2394,	<code>\QSR</code> 1976
1529, 1637, 1650,	2395, 2396, 2397,	<code>\QSRr</code> 1985,
1729, 1785, 1851,	2398, 2401, 2404, 2406	2006, 2017, 2022,
1876, 1882, 2048,	<code>\pgfmath@count</code>	2028, 2037, 2038, 2039
2061, 2090, 2149,	. . . 2107, 2109, 2111	<code>\QS@sortStep</code>
2156, 2161, 2241,	<code>\pgfmath@smuggleone</code> 2112	1891, 1999, 2015, 2016
2256, 2347, 2351,	<code>\pgfmathdeclarefunction</code>	<code>\quad</code> 1657
2371 , 2413 , 2448 , 2468 2106	
<code>\neg</code> 2515	<code>\pgfmathhint</code> 2107	R
<code>\negthinspace</code> 1534	<code>\pgfmathparse</code>	<code>\raisebox</code> 1345
<code>\newcounter</code> . . . 1902, 1903 1174, 1214,	<code>\relax</code> 1393,
<code>\NewDocumentCommand</code> .	2129, 2132, 2146, 2147	1976, 2019, 2021,
62, 103, 140, 161,	<code>\pgfmathresult</code>	2022, 2023, 2109, 2111
 1175, 2107,	

\renewcommand	\setganttlinklabel ..	\tiny	
.....	1095, 1096,	1123, 1124, 1125, 1126	940, 946,	
1097, 1098, 1101,			952, 1344, 1398, 2191	
1102, 1116, 1117, 1699				
\repeat 1892	\setlength	
\RequirePackage	55, 158,	2076, 2497, 2498, 2508	\titleformat .. 1090, 1092	
222, 226, 395, 401,			\titlespacing 1091
402, 462, 532, 612,			\tl	. 29, 37, 63, 64, 65,
929, 931, 932, 970,			66, 67, 68, 71, 72,	
971, 1081, 1082,			73, 74, 75, 77, 79,	
1085, 1087, 1089,			104, 105, 106, 107,	
1094, 1103, 1110,			108, 109, 110, 113,	
1121, 1122, 1131,			114, 115, 116, 117,	
1135, 1136, 1137,			118, 119, 162, 163,	
1165, 1166, 1205,			164, 165, 166, 167,	
1274, 1358, 1363,			168, 171, 172, 173,	
1364, 1380, 1387,			174, 175, 176, 177,	
1388, 1389, 1417,			285, 289, 307, 311,	
1532, 1641, 1642,			312, 313, 316, 321,	
1787, 1789, 1879,			322, 323, 334, 335,	
1896, 1898, 2050,			336, 337, 348, 354,	
2051, 2052, 2065,			357, 360, 369, 383,	
2093, 2102, 2153,			536, 539, 544, 545,	
2159, 2164, 2175,			553, 554, 557, 558,	
2176, 2244, 2349,			1035, 1056, 1057,	
2356, 2375, 2376,			1058, 1059, 1062,	
2378, 2379, 2417,			1063, 1064, 1065, 1567	
2418, 2419, 2450, 2470			\TmpPlaceEight 1809
\right 1534		\TmpPlaceFive 1806
\RIGHTarrow	... 2393, 2398		\TmpPlaceFour 1805
\rightarrow 600, 605		\TmpPlaceNine 1810
\rightarrow 207,		\TmpPlaceOne 1802
489, 494, 502, 506,			\TmpPlaceSeven 1808
508, 509, 511, 566,			\TmpPlaceSix 1807
574, 1849, 2287,			\TmpPlaceTen 1811
2294, 2296, 2299,			\TmpPlaceThree 1804
2304, 2309, 2310, 2315			\TmpPlaceTwo 1803
\rightouterjoin 2058		\TmpScale 1822
\Roman 1100		\TmpTransitionEight	.
\roman 1100, 1102		1798, 1819
\romannumeral 1974		\TmpTransitionFive	..
\rtimes 1640		1795, 1816
\rule	2007, 2029, 2044, 2055		\TmpTransitionFour	..
S			1794, 1815
\sb 67, 77, 79,		\TmpTransitionNine	..
108, 166, 501, 502,			1799, 1820
506, 509, 510, 511,			\TmpTransitionOne	...
1027, 1029, 1584,			1791, 1812
1587, 1590, 1719,			\TmpTransitionSeven	.
1837, 1846, 2309,			1797, 1818
2310, 2311, 2316,			\TmpTransitionSix	...
2320, 2321, 2324,			1796, 1817
2325, 2328, 2329, 2330			\TmpTransitionTen	...
\scriptscriptstyle	1800, 1821
.....	566, 574, 582		\TmpTransitionThree	.
\scriptsize	1046, 1285,		1793, 1814
1292, 1298, 2473, 2503			\TmpTransitionTwo	...
\section 46		1792, 1813
\seq	. 1349, 1350, 1351,		\TmpX 1823
1478, 1479, 1480, 1487			\TmpY 1824
\setbox 2054		\ttfamily 2068
\setcounter		U	
1093, 2008, 2030, 2044			\ul 935, 2066, 2067
\setganttlinklabel	..		\unlagger 902
1123, 1124, 1125, 1126			\unlassoc 888
\setlength			
2076, 2497, 2498, 2508				
\setmainfont 1083			
\setmainlanguage 396			
\setminted 2181, 2182			
\setsansfont 1084			
\setul 2067			
\sffamily 476,			
1090, 1092, 1194, 2265				
\shoveleft 1759			
\shoveright 1762			
\Sigma 64, 105,			
163, 994, 995, 1057				
\sigma 499, 501, 502			
\SLASH 1399		</	

<code>\umlclass</code> ... 615, 619, 623, 658, 659, 660, 696, 701, 706, 709, 754, 755, 756, 761, 762, 773, 792, 793, 796, 843, 846, 856, 862, 863, 877, 878, 879, 894, 895, 896, 897	<code>\umlVHVdep</code> 644, 645, 647, 648	X <code>\xappto</code> .. 2130, 2136, 2140
<code>\umldep</code> 851	<code>\umlVHVinherit</code> ... 627, 628, 633, 634, 641, 642, 764, 765, 870, 871	<code>\xdef</code> 1168
<code>\umlHVVHagggreg</code> 714, 767, 872	<code>\umlVHVreal</code> 758, 759, 899, 900	<code>\xintApply</code> 1978
<code>\umlinherit</code> 664, 704, 800, 841, 849	<code>\UParrow</code> 2395	<code>\xintApplyUnbraced</code> .. 1977, 1983, 1984, 1985
<code>\umlnote</code> 666, 802, 904	<code>\url</code> 1519	<code>\xintCSVtoList</code> 2037
<code>\umlreal</code> 662, 712	<code>\usemintedstyle</code> 2178	<code>\xintFor</code> 1927, 1932, 1937, 1942, 1948, 1954, 1995
<code>\umlsimpleclass</code> 630, 631, 632, 636, 638, 639, 640, 657, 791, 839, 840	<code>\usetikzlibrary</code> 56, 403, 930, 1138, 1275, 1788, 2053, 2094, 2103, 2380, 2466	<code>\xintIfEq</code> 1988
<code>\umlstatic</code> 774	V <code>\value</code> 1890	<code>\xintIfForLast</code> 1944, 1950, 1956
<code>\umluniagggreg</code> 798	<code>\varepsilon</code> 478, 489, 490, 980, 1584, 1590, 1619, 1630	<code>\xintIfGt</code> 1989
<code>\umluniassoc</code> 650, 663, 799, 886, 887	<code>\vfill</code> 2154	<code>\xintIfLt</code> 1987
<code>\umlVHuniassoc</code> .. 651, 652	<code>\vrule</code> 2361, 2365	<code>\xintLength</code> 1962
	<code>\vspace</code> 1015, 1019, 1455, 1473	<code>\xintntheft</code> 1974
		Z
		<code>\ZB</code> <u>1526</u>
		<code>\zB</code> <u>1525</u>
		<code>\zustandsnamens@liste</code> 998, 1005, 1006