

lehramt-informatik

Hermine Bschlangaul *

August 20, 2021

Contents

1	Klassen	3
1.1	Vorlage Theorie-Teil	4
1.2	Vorlage Aufgabensammlung	5
1.3	Vorlage Aufgabe	6
2	Pakete	7
2.1	aufgaben-einbinden.sty	8
2.2	aufgaben-titel.sty	9
2.3	automaten.sty	10
2.3.1	Endlicher Automat	10
2.3.2	Kellerautomat	12
2.3.3	Turingmaschine	13
2.4	basis.sty	16
2.5	baum.sty	20
2.5.1	Binärbaum	21
2.5.2	AVL-Baum	22
2.5.3	B-Baum	23
2.6	checkbox.sty	24
2.7	chomsky-normalform.sty	25
2.8	cpm.sty	27
2.8.1	Hilfsmakros für Zeitpunkt-Tabelle	28
2.9	cyk-algorithmus.sty	30
2.10	entwurfsmuster.sty	31
2.10.1	Namensschema der Entwurfsmuster-Makros:	31
2.10.2	Reihenfolge	31
2.10.3	Abstrakte Fabrik (Abstract Factory)	31
2.10.4	Adapter	32
2.10.5	Beobachter (Observer)	34
2.10.6	Dekorierer (Decorator)	36
2.10.7	Einfache Fabrik (Simple Factory)	37
2.10.8	Einzelstück (Singleton)	38
2.10.9	Erbauer (Builder)	38
2.10.10	Fabrikmethode (Factory Method)	40
2.10.11	Kompositum (Composite)	41
2.10.12	Modell-Präsentation-Steuerung (Model-View-Controller)	42
2.10.13	Zustand (State)	43
2.11	er.sty	45
2.12	formale-sprachen.sty	47
2.13	formatierung.sty	51
2.13.1	Schriftarten / Typographie	51
2.13.2	Farben	51
2.13.3	Überschriften	51

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

2.13.4	Listen	51
2.13.5	Kasten	51
2.13.6	Header	51
2.14	gantt.sty	52
2.15	grafik.sty	53
2.16	graph.sty	54
2.17	hanoi.sty	56
2.18	komplexitaetstheorie.sty	57
2.19	kontrollflussgraph.sty	59
2.20	kopf-fusszeilen.sty	61
2.21	literatur-dummy.sty	62
2.22	literatur.sty	63
2.23	makros.sty	64
2.24	master-theorem.sty	68
2.25	mathe.sty	72
2.26	minimierung.sty	73
2.27	normalformen.sty	76
2.28	petri.sty	78
2.29	potenzmengen-konstruktion.sty	80
2.30	pseudo.sty	82
2.31	pumping-lemma.sty	83
2.32	quicksort.sty	84
2.33	relationale-algebra.sty	87
2.34	rmodell.sty	88
2.35	sortieren.sty	89
2.36	spalten.sty	91
2.37	struktogramm.sty	92
2.38	syntax.sty	93
2.39	syntaxbaum.sty	95
2.40	synthese-algorithmus.sty	96
2.41	tabelle.sty	99
2.42	typographie.sty	100
2.43	uml.sty	101
2.44	vollstaendige-induktion.sty	103
2.45	wasserfall.sty	105
2.46	wpkalkuel.sty	106

3 Index

107

1 Klassen

1.1 Vorlage Theorie-Teil

```
\documentclass{lehramt-informatik-haupt}

\begin{document}

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Theorie-Teil
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

\chapter{Thema des Theorie-Teils}

\literatur

\end{document}
```

1.2 Vorlage Aufgabensammlung

```
\documentclass{lehramt-informatik-haupt}
\liLadeAllePakete

\begin{document}
\liAufgabe{30_AUD/06_Graphen/20_Dijkstra/Aufgabe_Graph-A-I}
\liAufgabe{30_AUD/06_Graphen/20_Dijkstra/Aufgabe_Graph-M-A-P-R-N}
\liAufgabe{30_AUD/06_Graphen/20_Dijkstra/Aufgabe_Staedte-A-F}
\liExamensAufgabe{46114/2008/09/Thema-1/Aufgabe-2}
\liExamensAufgabe{46115/2013/03/Thema-2/Aufgabe-5}
\liExamensAufgabe{66112/2004/03/Thema-1/Aufgabe-5}
\liExamensAufgabe{66115/2013/09/Thema-2/Aufgabe-9}
\liExamensAufgabe{66115/2015/03/Thema-2/Aufgabe-7}
\liExamensAufgabe{66115/2016/03/Thema-2/Aufgabe-6}
\liExamensAufgabe{66115/2017/03/Thema-1/Aufgabe-1}
\liExamensAufgabe{66115/2018/03/Thema-2/Aufgabe-9}
\liExamensAufgabe{66115/2020/09/Thema-1/Teilaufgabe-2/Aufgabe-3}
\end{document}
```

1.3 Vorlage Aufgabe

```
\documentclass{lehramt-informatik-aufgabe}
\liLadePakete{}
\begin{document}
\liAufgabenTitel{}
\section{
\index{DB}
\footcite{examen:}
}

\end{document}
```

2 Pakete

2.1 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
```


2.2 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
```

2.3 automaten.sty

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

2.3.1 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 }

```

2.3.2 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 }

```

2.3.3 Turingmaschine

```

158 \RequirePackage{amssymb}

```

\liTuringLeerzeichen

□

```

159 \def\liTuringLeerzeichen{\Box}

```

```

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

\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}:(z_1: \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`

`(z_1: \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

```

2.4 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
```

2.5 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}

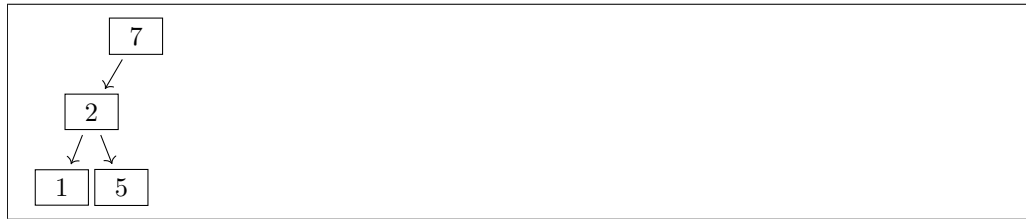
    für li binaer baum
402 \RequirePackage{tikz-qtrees}

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

2.5.1 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 }
```

2.5.2 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}
```



2.5.3 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

```

2.6 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
```


2.7 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

2.8 cpm.sty

```

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

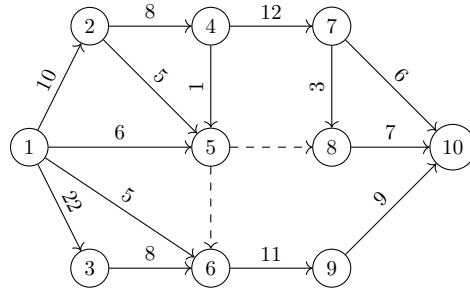
```

Let-Abkürzungen

```

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

```



```

\begin{tikzpicture}[scale=0.8,transform shape]
\liCpmEreignis{1}{0}{2}
\liCpmEreignis{2}{1}{4}
\liCpmEreignis{3}{1}{0}

\liCpmVorgang{1}{2}{10}
\liCpmVorgang{1}{3}{22}
\liCpmVorgang{1}{5}{6}

\liCpmVorgang[schein]{5}{6}{}
\liCpmVorgang[schein]{5}{8}{}
\end{tikzpicture}

```

```

\begin{tabular}{|l|l|l|l|l|l|l|l|}
\hline
$i$ & a & b & c & d & e & f & g \\
\hline
\FZ & 0 & 2 & 5 & 6 & 10 & 3 & 12 \\
\hline
\SZ & 0 & 2 & 5 & 10 & 10 & 6 & 12 \\
\hline
GP & 0 & 0 & 0 & 3 & 0 & 3 & 0 \\
\hline
\end{tabular}

```

```

\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

```

2.8.1 Hilfsmakros für Zeitpunkt-Tabelle

```

\begin{tabular}{|l||l|l|l|l|l|l|l|l|}
\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 & 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

```

2.9 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

$R_a$ & $R_c$ & $R_b$ & $R_c$ & $R_a$ & $R_b$ \16
B      & A      & A      & B      & C \15
S      & -      & S      & S \14
-      & -      & - \13
-      & - \12
S \11
\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   $\Rrightarrow #1 \in #2$
601 }
```

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

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

```
607
```

2.10 entwurfsmuster.sty

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

2.10.1 Namensschema der Entwurfsmuster-Makros:

Präfix: \liEntwurfs + Name des Entwurfsmuster DeutscherName + Suffix: (Uml, Akteure, Code, ohne)

2.10.2 Reihenfolge

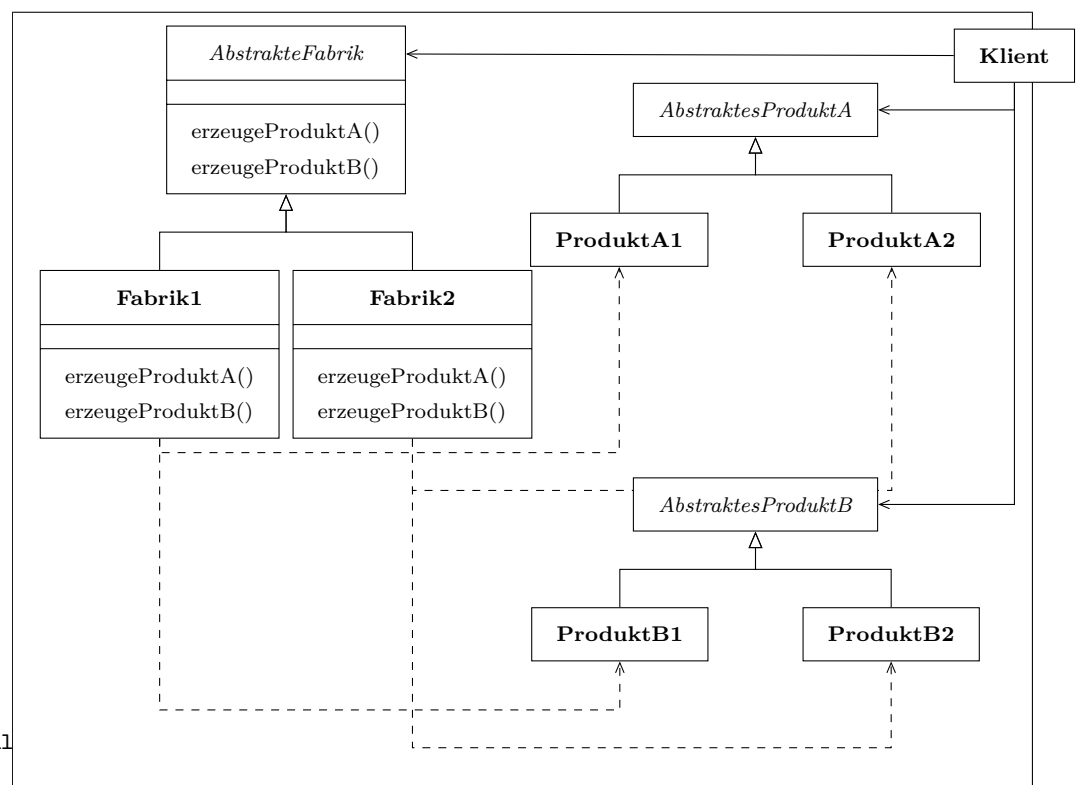
1. **Uml:** Uml-Klassendiagramm \liEntwurfsEinzelstueckUml
2. **Akteure:** Akteure, beteiligte Klassen \liEntwurfsEinzelstueckAkteure
3. **Code:** Allgemeines Code-Beispiel \liEntwurfsEinzelstueckCode
4. **ohne:** Ohne Suffix, Bündelung der einzelnen Makros eines Entwurfsmusters \liEntwurfsEinzelstueckAkteure

```
611 \RequirePackage{lehramt-informatik-uml}
```

- EntwurfsCodeAllgemein Allgemeine Code-Beispiele zu den UML-Diagrammen und Stellvertretern

```
612 \RequirePackage{lehramt-informatik-syntax}
613 \def\li@EntwurfsCode#1#2{
614   \liJavaDatei{entwurfsmuster/#1/allgemein/#2}
615 }
```

2.10.3 Abstrakte Fabrik (Abstract Factory)



```

616 \def\liEntwurfsAbstrakteFabrikUml{
617   \begin{tikzpicture}
618     \umlclass[type=abstract]{AbstrakteFabrik}{}{
619       erzeugeProduktA()\n
620       erzeugeProduktB()\n
621     }

```

```

622 \umlclass[below left=1cm and -1.5cm of AbstrakteFabrik]{Fabrik1}{-}{
623     erzeugeProduktA()\
624     erzeugeProduktB()\
625 }
626 \umlclass[below right=1cm and -1.5cm of AbstrakteFabrik]{Fabrik2}{-}{
627     erzeugeProduktA()\
628     erzeugeProduktB()\
629 }
630 \umlVHVinherit{Fabrik1}{AbstrakteFabrik}
631 \umlVHVinherit{Fabrik2}{AbstrakteFabrik}
632
633 \umlsimpleclass[right=3cm of AbstrakteFabrik,type=abstract]{AbstraktesProduktA}
634 \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktA]{ProduktA1}
635 \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktA]{ProduktA2}
636 \umlVHVinherit{ProduktA1}{AbstraktesProduktA}
637 \umlVHVinherit{ProduktA2}{AbstraktesProduktA}
638
639 \umlsimpleclass[above right=0cm and 1cm of AbstraktesProduktA]{Klient}
640
641 \umlsimpleclass[below=4.5cm of AbstraktesProduktA,type=abstract]{AbstraktesProduktB}
642 \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktB]{ProduktB1}
643 \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktB]{ProduktB2}
644 \umlVHVinherit{ProduktB1}{AbstraktesProduktB}
645 \umlVHVinherit{ProduktB2}{AbstraktesProduktB}
646
647 \umlVHVdep[arm1=-1.3cm,arm2=-1cm]{Fabrik1}{ProduktA1}
648 \umlVHVdep[arm2=-1cm]{Fabrik1}{ProduktB1}
649
650 \umlVHVdep[arm1=-1.8cm,arm2=-1.5cm]{Fabrik2}{ProduktA2}
651 \umlVHVdep[arm2=-1.5cm]{Fabrik2}{ProduktB2}
652
653 \umluniassoc[anchor2=25]{Klient}{AbstrakteFabrik}
654 \umlVHuniassoc[arm1=-1cm]{Klient}{AbstraktesProduktA}
655 \umlVHuniassoc{Klient}{AbstraktesProduktB}
656 \end{tikzpicture}
657 }

```

iEntwurfsAbstrakteFabrikCode

```

658 \def\liEntwurfsAbstrakteFabrikCode{
659     \li@EntwurfsCode{abstrakte_fabrik}{Produkte}
660     \li@EntwurfsCode{abstrakte_fabrik}{AbstrakteFabrik}
661     \li@EntwurfsCode{abstrakte_fabrik}{Klient}
662 }

```

\liEntwurfsAbstrakteFabrik

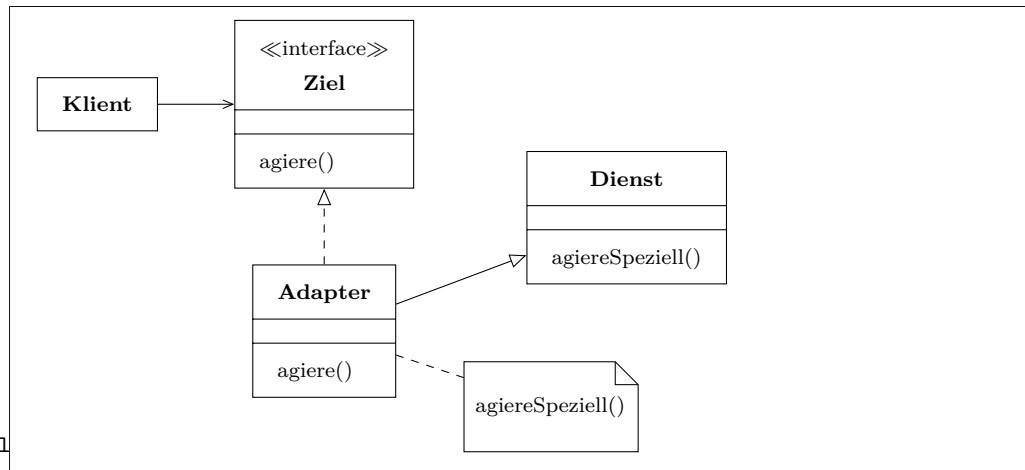
```

663 \def\liEntwurfsAbstrakteFabrik{
664     \liEntwurfsAbstrakteFabrikUml
665     \liEntwurfsAbstrakteFabrikCode
666 }

```

2.10.4 Adapter

\liEntwurfsAdapterUml



```

667 \def\liEntwurfsAdapterUml{
668   \begin{tikzpicture}
669     \umlsimpleclass[x=1,y=3]{Klient}{}{}
670     \umlclass[x=4,y=3,type=interface]{Ziel}{}{agiere()}
671     \umlclass[x=4,y=0]{Adapter}{}{agiere()}
672     \umlclass[x=8,y=1.5]{Dienst}{}{agiereSpeziell()}
673
674     \umlreal{Adapter}{Ziel}
675     \umluniassoc{Klient}{Ziel}
676     \umlinherit{Adapter}{Dienst}
677
678     \umlnote[x=7,y=-1,width=2cm]{Adapter}{agiereSpeziell()}
679   \end{tikzpicture}
680   \footcite[so ähnlich wie GoF]{wiki:adapter}
681 }

```

\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.

```

682 \def\liEntwurfsAdapterAkteure{
683   \begin{description}
684     \item[Ziel (Target)]
685
686     Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.
687
688     \item[Klient (Client)]
689
690     Der Klient nutzt Dienste über inkompatible Schnittstellen und greift
691     dabei auf adaptierte Schnittstellen zurück.
692
693     \item[Dienst (Adaptee)]
694
695     Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest
696     definierter Schnittstelle an.
697
698     \item[Adapter]
699
700     Der Adapter adaptiert die Schnittstelle des Dienstes auf die
701     Schnittstelle zum Klienten.\footcite{wiki:adapter}
702   \end{description}

```

```

703
704 \end{description}
705 }

```

\liEntwurfsAdapterCode

```

706 \def\liEntwurfsAdapterCode{
707 \li@EntwurfsCode{adapter}{Dienst}
708 \li@EntwurfsCode{adapter}{Ziel}
709 \li@EntwurfsCode{adapter}{Adapter}
710 \li@EntwurfsCode{adapter}{Klient}
711 }

```

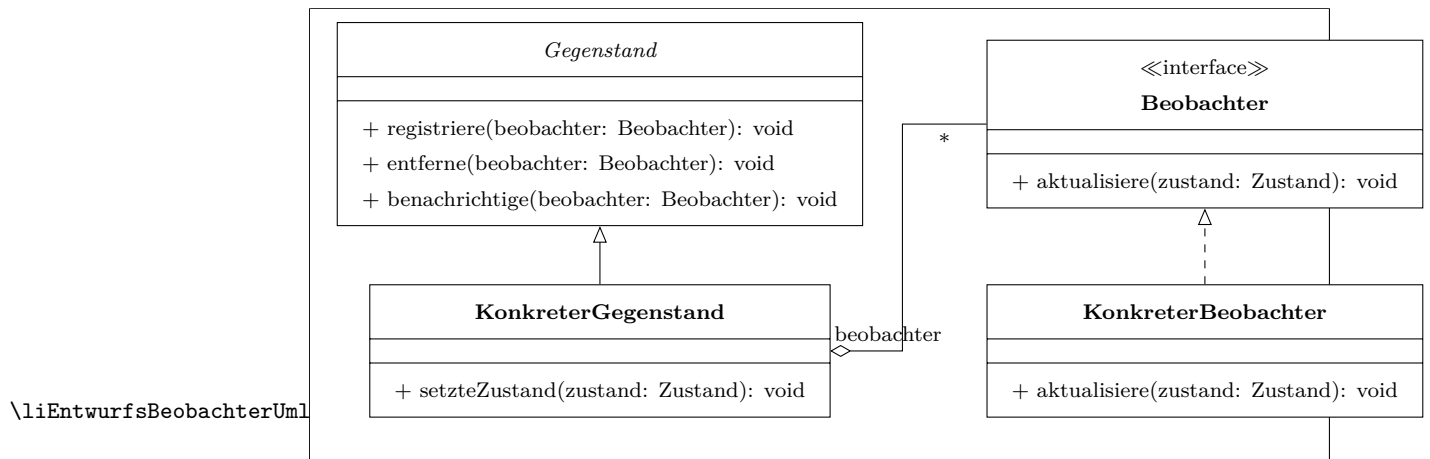
\liEntwurfsAdapter

```

712 \def\liEntwurfsAdapter{
713 \liEntwurfsAdapterUml
714 \liEntwurfsAdapterAkteure
715 \liEntwurfsAdapterCode
716 }

```

2.10.5 Beobachter (Observer)



```

717 \def\liEntwurfsBeobachterUml{
718 \begin{tikzpicture}
719 \umlclass[x=0,y=0,type=abstract]{Gegenstand}{-}{
720 + registriere(beobachter: Beobachter): void\\
721 + entferne(beobachter: Beobachter): void\\
722 + benachrichtige(beobachter: Beobachter): void\\
723 }
724 \umlclass[x=0,y=-3]{KonkreterGegenstand}{-}{
725 + setzteZustand(zustand: Zustand): void
726 }
727 \umlinherit{KonkreterGegenstand}{Gegenstand}
728
729 \umlclass[x=8,y=0,type=interface]{Beobachter}{-}{
730 + aktualisiere(zustand: Zustand): void
731 }
732 \umlclass[x=8,y=-3]{KonkreterBeobachter}{-}{
733 + aktualisiere(zustand: Zustand): void
734 }
735 \umlreal{KonkreterBeobachter}{Beobachter}
736
737 \umlHVHaggreg[arg1=beobachter,pos1=0.8,mult2=*,pos2=2.5]
738 {KonkreterGegenstand}{Beobachter}
739 \end{tikzpicture}
740 }

```

\liEntwurfsBeobachterAkteure

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.

```
741 \def\liEntwurfsBeobachterAkteure{
742   \begin{description}
743     \item[Gegenstand / Subjekt (Subject / Observable)]
744
745     Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also
746     „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne
747     deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An-
748     und Abmeldung von Beobachtern und eine Schnittstelle zur
749     Benachrichtigung von Beobachtern über Änderungen an.\footcite[Seite
750     251]{gof}
751
752     \item[Beobachter (Observer)]
753
754     Die Beobachter (auf Englisch auch subscriber, also „Abonnent“,
755     genannt) definieren eine Aktualisierungsschnittstelle.
756
757     \item[konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)]
758
759     Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert
760     den relevanten Zustand und benachrichtigt alle Beobachter bei
761     Zustandsänderungen über deren Aktualisierungsschnittstelle. Es
762     verfügt über eine Schnittstelle zur Erfragung des aktuellen
763     Zustands.
764
765     \item[Konkrete Beobachter (ConcreteObserver)]
766
767     Konkrete Beobachter verwalten die Referenz auf ein konkretes
768     Subjekt, dessen Zustand sie beobachten und speichern und dessen
769     Zustand konsistent ist. Sie implementieren eine
770     Aktualisierungsschnittstelle unter Verwendung der
771     Abfrageschnittstelle des konkreten Subjekts.
772     \footcite{wiki:beobachter}
773   \end{description}
774 }
```

\liEntwurfsBeobachterCode

```
775 \def\liEntwurfsBeobachterCode{
776   \li@EntwurfsCode{beobachter}{Gegenstand}
777   \li@EntwurfsCode{beobachter}{KonkreterGegenstand}
778   \li@EntwurfsCode{beobachter}{Beobachter}
779   \li@EntwurfsCode{beobachter}{KonkreterBeobachterA}
780   \li@EntwurfsCode{beobachter}{KonkreterBeobachterB}
```

```

781 \li@EntwurfsCode{beobachter}{Klient}
782 }

```

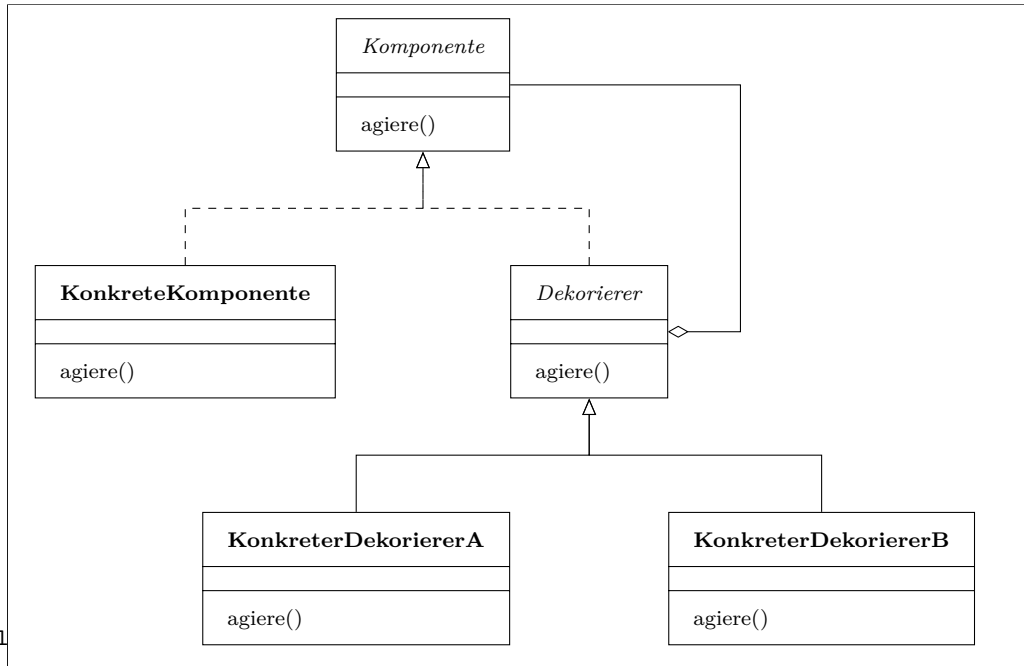
\liEntwurfsBeobachter

```

783 \def\liEntwurfsBeobachter{
784 \liEntwurfsBeobachterUml
785 \liEntwurfsBeobachterAkteure
786 \liEntwurfsBeobachterCode
787 }

```

2.10.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

788 \def\liEntwurfsDekoriererUml{
789 \begin{tikzpicture}
790 \umlclass[type=abstract]{Komponente}{-}{agiere()}
791 \umlclass[below left=1.5cm and 0cm of Komponente]{KonkreteKomponente}{-}{agiere()}
792 \umlclass[below right=1.5cm and 0cm of Komponente,type=abstract]{Dekorierer}{-}{agiere()}
793
794 \umlVHVreal{KonkreteKomponente}{Komponente}
795 \umlVHVreal{Dekorierer}{Komponente}
796
797 \umlclass[below left=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererA}{-}{agiere()}
798 \umlclass[below right=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererB}{-}{agiere()}
799
800 \umlVHVinherit{KonkreterDekoriererA}{Dekorierer}
801 \umlVHVinherit{KonkreterDekoriererB}{Dekorierer}
802
803 \umlHVVHaggreg[arm1=2cm]{Dekorierer}{Komponente}
804 \footcite{wiki:dekorierer}
805 \end{tikzpicture}
806 }

```

\liEntwurfsDekoriererCode

```

807 \def\liEntwurfsDekoriererCode{
808 \li@EntwurfsCode{dekorierer}{Komponente}
809 \li@EntwurfsCode{dekorierer}{KonkreteKomponente}
810 \li@EntwurfsCode{dekorierer}{Dekorierer}
811 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererA}
812 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererB}
813 \li@EntwurfsCode{dekorierer}{Klient}
814 }

```

\liEntwurfsDekorierer

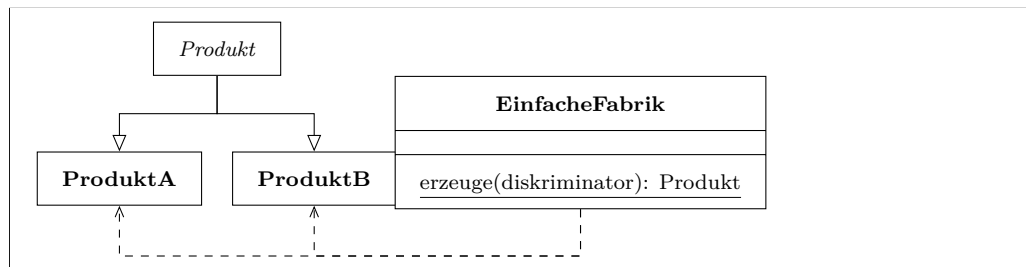
```

815 \def\liEntwurfsDekorierer{
816   \liEntwurfsDekoriererUml
817   \liEntwurfsDekoriererAkteure
818   \liEntwurfsDekoriererCode
819 }

```

2.10.7 Einfache Fabrik (Simple Factory)

\liEntwurfsEinfacheFabrikUml Quelle: <https://refactoring.guru/design-patterns/factory-comparison>



```

820 \def\liEntwurfsEinfacheFabrikUml{
821   \begin{tikzpicture}
822     \umlsimpleclass[type=abstract]{Produkt}
823     \umlsimpleclass[below left=1cm and -0.65cm of Produkt]{ProduktA}
824     \umlsimpleclass[below right=1cm and -0.65cm of Produkt]{ProduktB}
825     \umlVHVinherit{Produkt}{ProduktA}
826     \umlVHVinherit{Produkt}{ProduktB}
827     \umlclass[below right=0cm and 1.5cm of Produkt]{EinfacheFabrik}{
828     }{
829       \umlstatic{erzeuge(diskriminator): Produkt}\
830     }
831     \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktA}
832     \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktB}
833   \end{tikzpicture}
834 }

```

\liEntwurfsEinfacheFabrikAkteure Quelle: frei überstetzt aus GoF

EinfacheFabrik Eine Klasse mit einer Erzeugungsmethode, die über eine größere Bedingung verschiedene Objekt instanziiert.

Produkt Eine abstrakte Klasse, die von den konkreten Produkten geerbt wird.

KonkretesProdukt Ein konkretes Produkt, das von der einfachen Fabrik erzeugt wird.

```

835 \def\liEntwurfsEinfacheFabrikAkteure{
836   \begin{description}
837     \item[EinfacheFabrik]
838
839     Eine Klasse mit einer Erzeugungsmethode, die über eine größere
840     Bedingung verschiedene Objekt instanziiert.
841
842     \item[Produkt]
843
844     Eine abstrakte Klasse, die von den konkreten Produkten geerbt wird.
845
846     \item[KonkretesProdukt]
847
848     Ein konkretes Produkt, das von der einfachen Fabrik erzeugt wird.
849   \end{description}
850 }

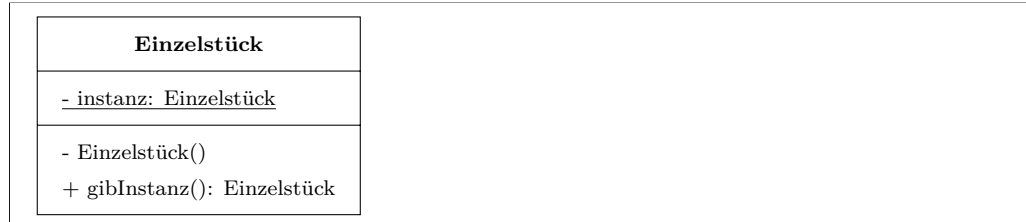
```

\liEntwurfsEinfacheFabrik

```
851 \def\liEntwurfsEinfacheFabrik{
852   \liEntwurfsEinfacheFabrikUml
853   \liEntwurfsEinfacheFabrikAkteure
854 }
```

2.10.8 Einzelstück (Singleton)

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```
855 \def\liEntwurfsEinzelstueckUml{
856   \begin{tikzpicture}
857     \umlclass{Einzelstück}{
858       \umlstatic{- instanz: Einzelstück}\\
859     }{
860       - Einzelstück()\\
861       + gibInstanz(): Einzelstück
862     }
863   \end{tikzpicture}
864 }
```

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.

```
865 \def\liEntwurfsEinzelstueckAkteure{
866   \begin{description}
867     \item[Einzelstück (Singleton)]
868
869     stellt eine statische Methode bereit, mit deren Hilfe die Klienten
870     nur auf eine einzige Instanz der Klasse zugreifen können.
871   \end{description}
872 }
```

\liEntwurfsEinzelstueckCode

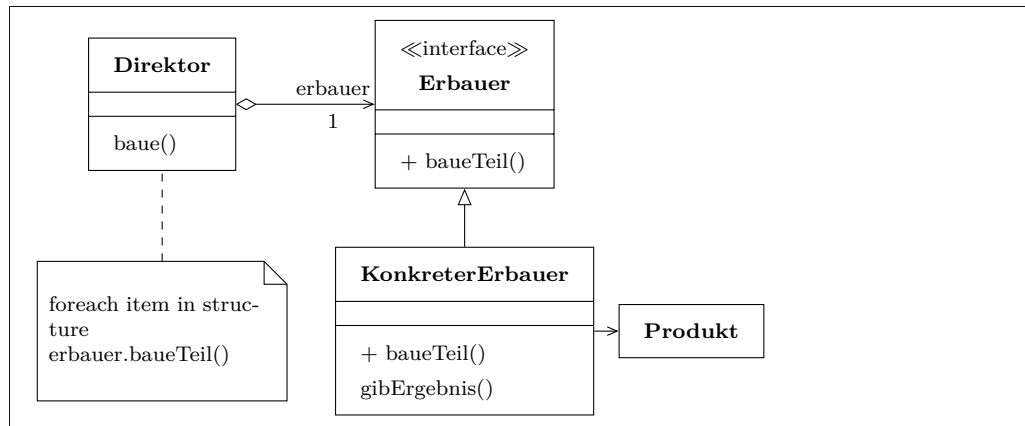
```
873 \def\liEntwurfsEinzelstueckCode{
874   \li@EntwurfsCode{einzelstueck}{Einzelstueck}
875 }
```

\liEntwurfsEinzelstueck

```
876 \def\liEntwurfsEinzelstueck{
877   \liEntwurfsEinzelstueckUml
878   \liEntwurfsEinzelstueckAkteure
879   \liEntwurfsEinzelstueckCode
880 }
```

2.10.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

881 \def\liEntwurfsErbauerUml{
882   \begin{tikzpicture}
883     \umlsimpleclass[x=7,y=0]{Produkt}{}{}
884     \umlclass[x=4,y=3,type=interface]{Erbauer}{}{+ baueTeil()}
885     \umlclass[x=4,y=0]{KonkreterErbauer}{}{
886       + baueTeil()\
887       gibErgebnis()}
888     \umlclass[x=0,y=3]{Direktor}{}{baue()}
889
890     \umluniagg[arg2=erbauer,pos2=0.7,mult2=1]{Direktor}{Erbauer}
891     \umluniassoc{KonkreterErbauer}{Produkt}
892     \umlinherit{KonkreterErbauer}{Erbauer}
893
894     \umlnote[x=0,y=0,width=3cm]{Direktor}{
895       foreach item in structure\
896       erbauer.baueTeil()
897     }
898   \end{tikzpicture}
899   \footcite{wiki:erbauer}
900 }

```

\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.

```

901 \def\liEntwurfsErbauerAkteure{
902   \begin{description}
903     \item[Erbauer]
904
905     Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der
906     Teile eines komplexen Objektes.
907
908     \item[KonkreterErbauer]
909
910     Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch
911     Implementierung der Schnittstelle. Außerdem definiert und verwaltet er

```

```

912     die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine
913     Schnittstelle zum Auslesen des Produkts.
914
915     \item[Direktor]
916
917     Der Direktor konstruiert ein komplexes Objekt unter Verwendung der
918     Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer
919     zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder
920     benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom
921     Klienten.
922
923     \item[Produkt]
924
925     Das Produkt repräsentiert das zu konstruierende komplexe Objekt.
926     \footcite{wiki:erbauer}
927 \end{description}
928 }

```

\liEntwurfsErbauer

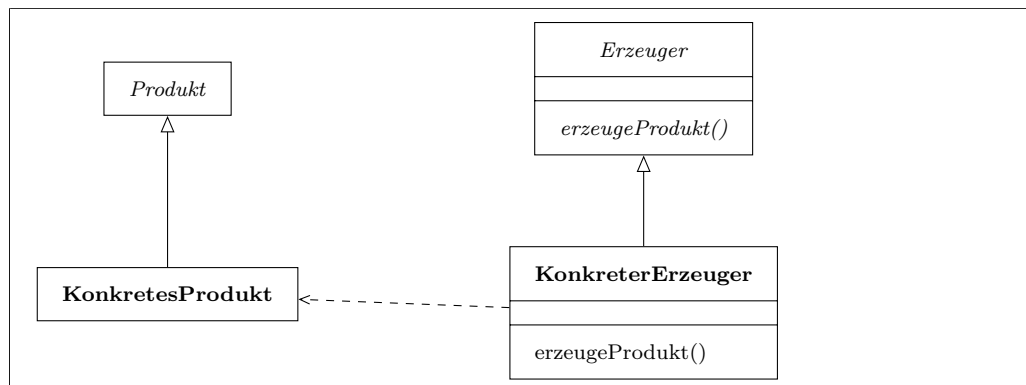
```

929 \def\liEntwurfsErbauer{
930   \liEntwurfsErbauerUml
931   \liEntwurfsErbauerAkteure
932 }

```

2.10.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

933 \def\liEntwurfsFabrikmethodeUml{
934   \begin{tikzpicture}
935     \umlsimpleclass[type=abstract]{Produkt}
936     \umlsimpleclass[below=2cm of Produkt]{KonkretesProdukt}
937     \umlinherit{KonkretesProdukt}{Produkt}
938
939     \umlclass[type=abstract,right=4cm of Produkt]{Erzeuger}{\{
940       \textit{erzeugeProdukt()}\}
941     }
942     \umlclass[below=1.2cm of Erzeuger]{KonkreterErzeuger}{\{
943       erzeugeProdukt()
944     }
945     \umlinherit{KonkreterErzeuger}{Erzeuger}
946
947     \umldep{KonkreterErzeuger}{KonkretesProdukt}
948   \end{tikzpicture}
949 }

```

EntwurfsFabrikmethodeAkteure Quelle: deutsche Wikipedia

Produkt Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das zu erzeugende Produkt.

KonkretesProdukt KonkretesProdukt implementiert die Produkt-Schnittstelle.

Erzeuger Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt zu erzeugen und kann eine Default-Implementierung beinhalten.

KonkreterErzeuger KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den Konstruktor einer konkreten Produkt-Klasse aufruft).

```

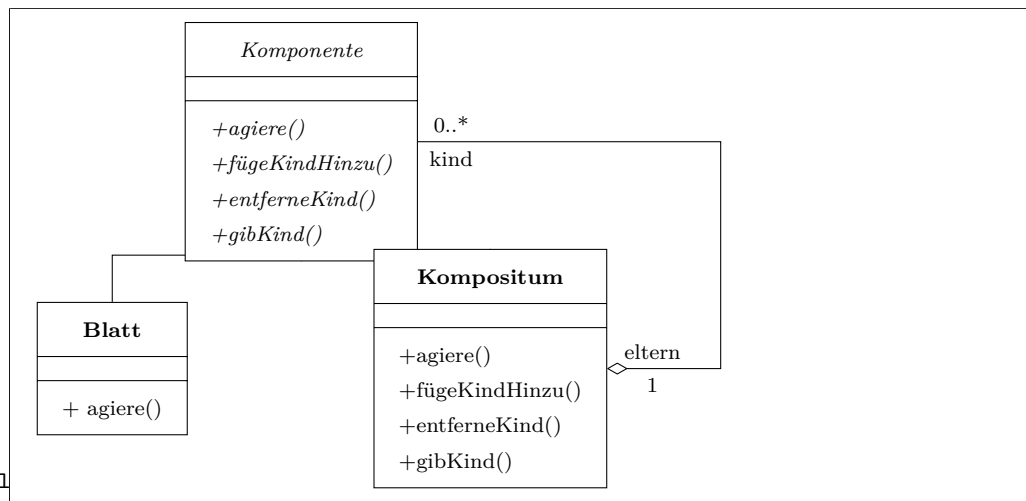
950 \def\liEntwurfsFabrikmethodeAkteure{
951   \begin{description}
952     \item[Produkt]
953
954     Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das
955     zu erzeugende Produkt.
956
957     \item[KonkretesProdukt]
958
959     KonkretesProdukt implementiert die Produkt-Schnittstelle.
960
961     \item[Erzeuger]
962
963     Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt
964     zu erzeugen und kann eine Default-Implementierung beinhalten.
965
966     \item[KonkreterErzeuger]
967
968     KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm
969     entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den
970     Konstruktor einer konkreten Produkt-Klasse aufruft).
971
972     \footcite{wiki:fabrikmethode}
973   \end{description}
974 }
```

\liEntwurfsFabrikmethode

```

975 \def\liEntwurfsFabrikmethode{
976   \liEntwurfsFabrikmethodeUml
977   \liEntwurfsFabrikmethodeAkteure
978 }
```

2.10.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

979 \def\liEntwurfsKompositumUml{
980   \begin{tikzpicture}
981     \umlclass[x=2.5,y=3,type=abstract]{Komponente}{-}{-}{
982       \textit{+agiere()}\
983       \textit{+fügeKindHinzu()}\
984       \textit{+entferneKind()}\
985       \textit{+gibKind()}
986     }
987     \umlclass[x=0]{Blatt}{-}{+ agiere()}
988     \umlclass[x=5]{Kompositum}{-}{-}{
989       +agiere()\
990       +fügeKindHinzu()\
991       +entferneKind()\
992       +gibKind()
993     }
994
995     \umlVHVinherit{Kompositum}{Komponente}
996     \umlVHVinherit{Blatt}{Komponente}
997     \umlHVVHaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,]
998   \end{tikzpicture}
999 }

```

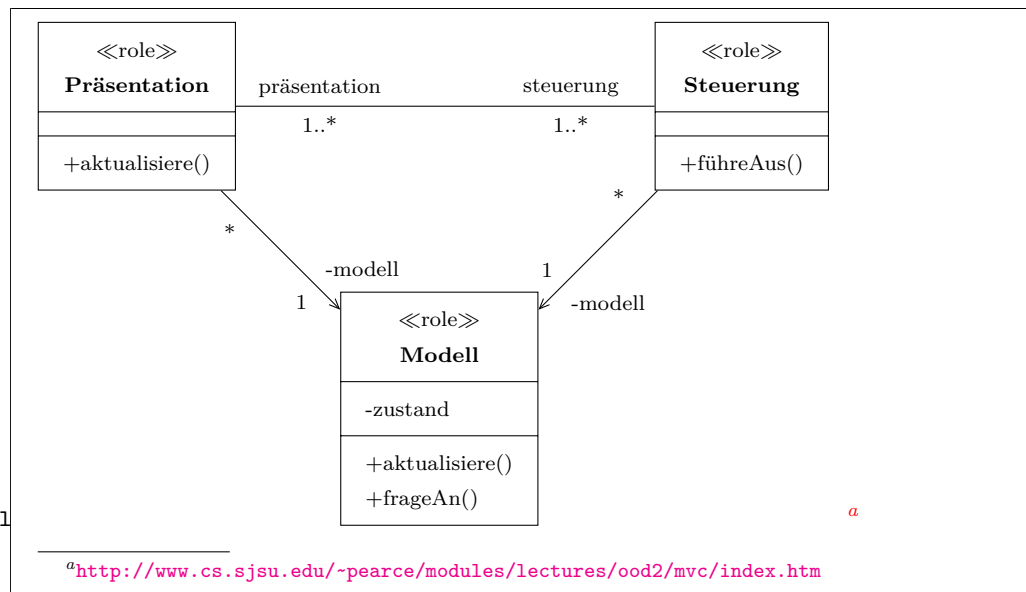
\liEntwurfsFabrikmethode

```

1000 \def\liEntwurfsKompositum{
1001   \liEntwurfsKompositumUml
1002   \liEntwurfsKompositumAkteure
1003 }

```

2.10.12 Modell-Präsentation-Steuerung (Model-View-Controller)



```

1004 \def\liEntwurfsModellPraesentationSteuerungUml{
1005   \begin{tikzpicture}
1006     \umlclass[x=-4cm,type=role]{Präsentation}{-}{+aktualisiere()}
1007     \umlclass[x=4cm,type=role]{Steuerung}{-}{+führeAus()}
1008     \umlclass[y=-4cm,type=role]{Modell}{-}{
1009       -zustand
1010     }{
1011       +aktualisiere()\
1012       +frageAn()
1013     }
1014
1015     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
1016     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}

```

```

1017 \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
1018 \end{tikzpicture}
1019 \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
1020 }

```

ModellPraesentationSteuerung

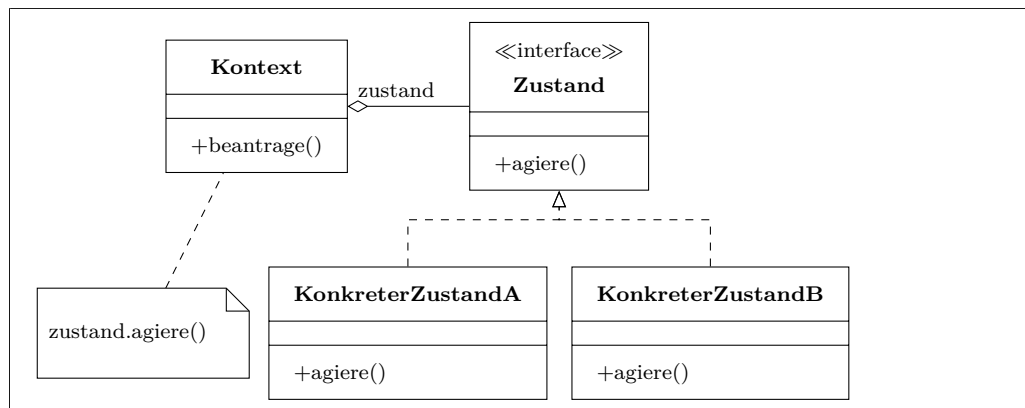
```

1021 \def\liEntwurfs{
1022 \liEntwurfsModellPraesentationSteuerungUml
1023 \liEntwurfsModellPraesentationSteuerungAkteure
1024 }

```

2.10.13 Zustand (State)

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



```

1025 \def\liEntwurfsZustandUml{
1026 \begin{tikzpicture}
1027 \umlcclass[x=-1,y=3]{Kontext}{+beantrage()}
1028 \umlcclass[x=3,y=3,type=interface]{Zustand}{+agiere()}
1029 \umlcclass[x=1,y=0]{KonkreterZustandA}{+agiere()}
1030 \umlcclass[x=5,y=0]{KonkreterZustandB}{+agiere()}
1031
1032 \umlVHVreal{KonkreterZustandA}{Zustand}
1033 \umlVHVreal{KonkreterZustandB}{Zustand}
1034
1035 \umlagg[reg[arg=zustand,pos=0.4]{Kontext}{Zustand}
1036
1037 \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
1038 \end{tikzpicture}
1039 }

```

\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.

```

1040 \def\liEntwurfsZustandAkteure{
1041 \begin{description}
1042 \item[Kontext (Context)]
1043
1044 definiert die clientseitige Schnittstelle und verwaltet die separaten
1045 Zustandsklassen.
1046

```

```

1047 \item[State (Zustand)]
1048
1049 definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
1050 implementiert gegebenenfalls ein Standardverhalten.
1051
1052 \item[KontreterZustand (ConcreteState)]
1053
1054 implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
1055 verbunden ist.
1056 \end{description}
1057 }

\liEntwurfsZustand
1058 \def\liEntwurfsZustand{
1059 \liEntwurfsZustandUml
1060 \liEntwurfsZustandAkteure
1061 }

1062

```

2.11 er.sty

```
1063 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1064 \ProvidesPackage{lehramt-informatik-er}[2020/06/13 Zum Setzen von
1065 ER-Diagrammen]

1066 \RequirePackage{tikz-er2}
1067 \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}

1068 \RequirePackage{soul}
1069 \RequirePackage{fontawesome}

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

1070 \ExplSyntaxOn

\liErEntity
1071 \def\liErEntity#1{\textbf{#1}}

\liErRelationship
1072 \def\liErRelationship#1{\ul{#1}}

\liErAttribute
1073 \def\liErAttribute#1{\emph{#1}}

\liErMpEntity mp = marginpar
Let-Abkürzung: \let\ e=\liErMpEntity
1074 \def\liErMpEntity#1{
1075   \liErEntity{#1}
1076   \marginpar{
1077     \liErEntity{\tiny\faSquareO{}}~E:~#1}
1078   }
1079 }
□

\liErMpRelationship Let-Abkürzung: \let\ r=\liErMpRelationship
1080 \def\liErMpRelationship#1{
1081   \liErRelationship{#1}
1082   \marginpar{
1083     \liErRelationship{\tiny\faGg{}}~R:~#1}
1084   }
1085 }

\liErMpAttribute Let-Abkürzung: \let\ a=\liErMpAttribute
1086 \def\liErMpAttribute#1{
1087   \liErAttribute{#1}
1088   \marginpar{
1089     \liErAttribute{\tiny\faCircleThin{}}~A:~#1}
1090   }
1091 }

\liErDatenbankName Let-Abkürzung: \let\ d=\liErDatenbankName
datenbank name
1092 \def\liErDatenbankName#1{
1093   {
1094     \footnotesize\texttt{(#1)}
1095   }
1096 }

1097 \ExplSyntaxOff
1098

```

2.12 formale-sprachen.sty

```

1099 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1100 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
1101 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]
1102
1103 \directlua{
1104   formale_sprachen = require('lehramt-informatik-formale-sprachen')
1105 }
1106
1107 \RequirePackage{hyperref}
1108 \RequirePackage{lehramt-informatik-mathe}

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

1109 \def\liMengeOhneMathe#1{\{ #1 \}}
1110 \def\liMenge#1{%
1111 \ifmmode%
1112 \liMengeOhneMathe{#1}%
1113 \else%
1114 $\liMengeOhneMathe{#1}$%
1115 \fi%
1116 }

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

1117 \def\liEpsilon{\varepsilon}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

1118 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1119 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1120 \def\liPotenzmenge#1{\liPotenzmengeOhneMathe{#1}}

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

1121 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
1122 \let\liZustandsmenge=\liPotenzmenge

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

1123 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1124 \def\liUeberfuehrungsFunktion#1{
1125 \ifmmode
1126 \liUeberfuehrungsFunktionOhneMathe{#1}
1127 \else
1128 $\liUeberfuehrungsFunktionOhneMathe{#1}$
1129 \fi
1130 }

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

1131 \def\liAlphabet#1{\Sigma = \{ #1 \}}

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

1132 \def\liBandAlphabet#1{\Gamma = \Sigma \cup \{ #1 \}}

\liZustandsBuchstabe

1133 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1134 \def\liZustandsBuchstabeGross{Z}

```

```

\liZustandsmengeNr
1135 \def\zustandsnamens@liste#1#2{
1136   $
1137   \{
1138     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1139   \}
1140   $
1141 }
1142 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

\liZustandsmengeNrGross
1143 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

\liZustandsname \liZustandsname{1}: $z_1$
1144 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

\liZustandsnameGross \liZustandsnameGross{1}: $Z_1$
1145 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

\liAbleitung \liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1146 \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}

1147 \NewDocumentEnvironment { liProduktionsRegeln }
1148 { 0{P} +b }
1149 {
1150   \noindent
1151   $#1 = \{ $
1152   \vspace{-0.2cm}
1153   \begin{align*}
1154     \directlua{formale_sprachen.produktions_regeln('#2')}
1155   \end{align*}
1156   \vspace{-1.5cm}
1157   \begin{flushright}\}\end{flushright}
1158 } {}

\liProduktionen \liProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
1159 \def\liProduktionen#1{
1160   \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
1161 }

\liZustandsnameTiefgestellt Automatisch tiefgestellte Nummerierung \z1
Let-Abkürzung: \let\z=\liZustandsnameTiefgestellt
1162 \def\liZustandsnameTiefgestellt#1{
1163   \ifmmode
1164     \liZustandsBuchstabe\sb{#1}
1165   \else
1166     $\liZustandsBuchstabe\sb{#1}$
1167   \fi
1168 }

1169 \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}

1170 \NewDocumentCommand{ \liAusdruck } { O{L} m m } {
1171   $
1172   \tl_if_empty:nTF {#1} {} {#1 =}
1173   \{
1174     \, #2 \,
1175     |
1176     \, #3 \,
1177   \}$
1178 }
1179 \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

1180 \def\liFlaci#1{%
1181   \par
1182   {%
1183     \scriptsize
1184     Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1185     Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1186     Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1187     \href{https://flaci.com/#1}{flaci.com/#1}%
1188   }%
1189   \par
1190 }

\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}:  $G = (V, \Sigma, P, X)$ 

1191 \ExplSyntaxOn
1192 \NewDocumentCommand {\liGrammatik} { O{G} m } {
1193   \tl_set:Nn \l_variablen_tl {V}
1194   \tl_set:Nn \l_alphabet_tl {\Sigma}
1195   \tl_set:Nn \l_produktionen_tl {P}
1196   \tl_set:Nn \l_start_tl {S}
1197
1198   \keys_define:nn { grammatik } {
1199     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1200     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1201     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1202     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1203   }
1204
1205   \keys_set:nn { grammatik } { #2 }
1206
1207   $#1 = (
1208     \l_variablen_tl,
1209     \l_alphabet_tl,
1210     \l_produktionen_tl,
1211     \l_start_tl
1212   )$

```

```
1213 }  
1214 \ExplSyntaxOff  
  
1215
```

2.13 formatierung.sty

```
1216 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1217 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

2.13.1 Schriftarten / Typographie

```
1218 \RequirePackage{mathpazo}
1219 \RequirePackage[no-math]{fontspec}
1220 \setmainfont{texgyrepagella}
1221 \setsansfont{QTAncientOlive}
1222 \RequirePackage{sectsty}
1223 \allsectionsfont{\fontspec{QTAncientOlive}}
```

2.13.2 Farben

```
1224 \RequirePackage{xcolor}
1225 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

2.13.3 Überschriften

```
1226 \RequirePackage{titlesec}
1227 \titleformat{\chapter}[display]{\sffamily\bfseries}{\Opt}{\LARGE}
1228 \titlespacing{\chapter}{0pt}{0pt}{*1}
1229 \titleformat{\paragraph}[hang]{\sffamily\normalsize\bfseries}{\theparagraph}{1em}{}
1230 \setcounter{secnumdepth}{0}
```

2.13.4 Listen

```
1231 \RequirePackage{paralist}
1232 \renewcommand\labelitemi{-}
1233 \renewcommand\labelitemii{-}
1234 \renewcommand\labelitemiii{-}
1235 \renewcommand\labelitemiv{-}
1236 % Counter: enumi enumii enumiii enumiv
1237 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1238 \renewcommand{\labelenumi}{(\alph{enumi})}
1239 \renewcommand{\labelenumii}{(\roman{enumii})}
```

2.13.5 Kasten

```
1240 \RequirePackage{mdframed}
1241 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1242 \NewDocumentEnvironment { liKasten }{ } {
1243   \begin{mdframed}[backgroundcolor=white!0]
1244 } {
1245   \end{mdframed}
1246 }
```

2.13.6 Header

```
1247 \RequirePackage{fancyhdr}
1248 \fancyhead[L,C,R]{}
1249 \fancyfoot[L]{}
1250 \fancyfoot[C]{}
1251 \fancyfoot[R]{\thepage}
1252 \pagestyle{fancy}
1253 \renewcommand{\headrulewidth}{0pt}
1254 \renewcommand{\footrulewidth}{0pt}
1255
```

2.14 gantt.sty

```

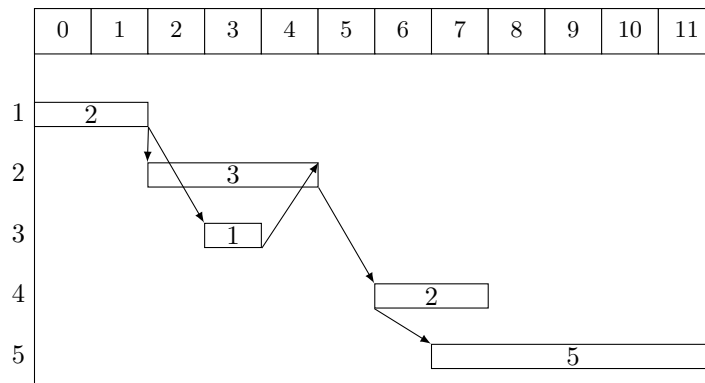
1256 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1257 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\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}

```



```

1258 \RequirePackage{tikz-uml}
1259 \RequirePackage{pgfgantt}
1260 \setganttlinklabel{f-s}{}
1261 \setganttlinklabel{s-s}{}
1262 \setganttlinklabel{f-f}{}
1263 \setganttlinklabel{s-f}{}
1264

```

2.15 grafik.sty

```
1265 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1266 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1267 TikZ, um Grafiken zeichnen zu können.]
1268 \RequirePackage{tikz}
1269
```

2.16 graph.sty

```

1270 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1271 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1272 \RequirePackage{tikz}

```

Für die die Adjazenzliste (`xrightharrow`)

```

1273 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

$$\begin{array}{c}
 \\
 a \\
 b \\
 c \\
 d \\
 e
 \end{array}
 \begin{pmatrix}
 & a & b & c & d & e \\
 \begin{pmatrix}
 0 & 1 & 0 & 4 & 0 \\
 0 & 0 & 0 & 1 & 3 \\
 0 & 0 & 0 & 1 & 0 \\
 4 & 1 & 1 & 0 & 2 \\
 0 & 0 & 0 & 0 & 0
 \end{pmatrix}
 \end{pmatrix}$$

```

1274 \RequirePackage{blkarray}
1275 \usetikzlibrary{arrows.meta}

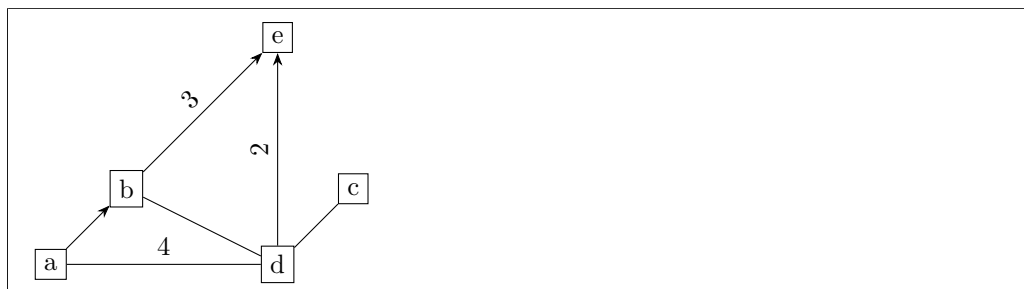
```

```

\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}

```



```

1276 \tikzset{
1277   li graph/.style={
1278     every node/.style={
1279       rectangle,
1280       draw,
1281     },
1282     every edge/.style={
1283       >={Stealth[black]},
1284       draw,
1285     },
1286     every edge/.append style={
1287       every node/.style={
1288         sloped,
1289         auto,
1290       }
1291     }
1292   },
1293   li markierung/.style={
1294     ultra thick,
1295   }
1296 }

```

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}

```

```

1297 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1298

```

2.17 hanoi.sty

1299 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 1300 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
 1301 von Hanoi-Grafiken]

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

```
1302 \RequirePackage{tikz}
1303 \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}

1304 \def\li@mset #1[#2]=#3{%
1305 \expandafter\xdef\csname #1#2\endcsname{#3}
1306 }
1307 \def\li@mget #1[#2]{%
1308 \csname #1#2\endcsname
1309 }
1310 \def\li@minc #1[#2]+=#3{%
1311 \pgfmathparse{\li@mget #1[#2]+#3}%
1312 \li@mset #1[#2]=\pgfmathresult
1313 }
1314
1315 \def\liHanoi#1#2{
1316   \edef\li@numdiscs{#1}
1317   \def\li@sequence{#2}
1318   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1319     % init colors
1320     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purple!80!black}{
1321       \li@mset col[\j]={\c};
1322     }
1323     % draw poles and init pole counters
1324     \foreach \j in {1,2,3}{
1325       \li@mset pos[\j]=0
1326       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1327     }
1328     % draw base
1329     \draw (.5,-.5) -- +(3,0);
1330     % draw discs
1331     \foreach[count=\k] \i/\j in \li@sequence{
1332       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*\i/\li@numdiscs,0);
1333       \li@minc pos[\j]+=.5
1334     }
1335   \end{tikzpicture}
1336 }
```


2.18 komplexitaetstheorie.sty

```

1337 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1338 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1339 Setzen von Karp's NP-vollständigen Problemen, Hilfsmakros für die
1340 Polynomialzeitreduktion.]

```

Let-Abkürzungen

```

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

```

```

1341 \liLadePakete{mathe}

```

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

```

1342 \RequirePackage{mdframed}

```

```

\liStrich $L, \liStrich{L}$:  $L, L'$ 

```

```

1343 \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

```

```

1344 \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`

```

1345 \def\liProblemBeschreibung#1#2#3{
1346   \begin{mdframed}[
1347     userdefinedwidth=9cm,
1348     align=center,
1349     backgroundcolor=white!0,
1350   ]
1351     \centerline{\large\liProblemName{#1}}
1352
1353     \medskip
1354
1355     \begin{description}
1356       \item[Gegeben:] #2
1357       \item[Frage:] #3
1358     \end{description}
1359   \end{mdframed}
1360 }

```

```

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

1361 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1362 \begin{displaymath}
1363 \liProblemName{#1}
1364 \preceq_{#2}
1365 \liProblemName{#3}
1366 \end{displaymath}
1367 }

\liProblemVertexCover

1368 \def\liProblemClique{%
1369 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1370 Mindestgröße  $n$  in einem gegebenen Graphen.
1371 \footcite[Seite 76]{theo:fs:4}
1372 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1373 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1374 \footcite{wiki:cliquenproblem}
1375 }

\liProblemVertexCover

1376 \def\liProblemVertexCover{%
1377 %
1378 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1379 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1380 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1381 \footcite{wiki:knotenueberdeckung}
1382
1383 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1384 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1385 mindestens einem Knoten aus  $U$  verbunden ist.
1386 \footcite[Seite 78]{theo:fs:4}%
1387 }

\liProblemSubsetSum Kein Karp-21-Problem

1388 \def\liProblemSubsetSum{%
1389 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1390 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1391 \footcite{wiki:teilsommenproblem}
1392 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1393 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1394 größer als eine gegebene obere Schranke  $c$  ist.
1395 \footcite[Seite 74]{theo:fs:4}
1396 }

\liProblemSubsetSum Kein Karp-21-Problem

1397 \def\liProblemSat{%
1398 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1399 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1400 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1401 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1402 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1403 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1404 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1405 aufgestellt werden.
1406 \footcite[Seite 71]{theo:fs:4}
1407 }

1408

```

2.19 kontrollflussgraph.sty

```
1409 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1410 \ProvidesPackage{lehramt-informatik-kontrollflussgraph}[2020/11/07]

1411 \RequirePackage{tikz}
1412 \usetikzlibrary{positioning}
1413 \tikzset{
1414   li kontrollfluss/.style={
1415     knoten/.style={
1416       circle,
1417       draw
1418     },
1419     usebox/.style={
1420       draw,
1421       rectangle,
1422       font=\scriptsize,
1423       anchor=west,
1424       align=left,
1425     },
1426     bedingung/.style={
1427       midway,
1428       draw=none,
1429       font=\scriptsize
1430     },
1431     knotenbeschriftung/.style={
1432       draw,
1433       rectangle,
1434       midway,
1435       font=\scriptsize
1436     },
1437     wahr/.style={
1438       thick
1439     },
1440     falsch/.style={
1441       dashed
1442     },
1443     every node/.style={
1444       circle,
1445       draw,
1446     },
1447     every edge/.append style={
1448       every node/.style={
1449         draw=none,
1450         bedingung,
1451       }
1452     },
1453     every path/.style={
1454       draw,
1455       ->,
1456     },
1457     every pin/.style={
1458       draw,
1459       dotted,
1460       rectangle,
1461       pin position=right
1462     },
1463     every pin edge/.style={
1464       dotted,
1465       arrows=-,
1466     }
1467   }
1468 }
```

liKontrollflussgraph

```

1469 \NewDocumentEnvironment { liKontrollflussgraph } { 0{ } } {
1470   \begin{tikzpicture}[
1471     li kontrollfluss,
1472     #1
1473   ]
1474 } {
1475   \end{tikzpicture}
1476 }

\liAnweisung
1477 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}

\liBedingung Let-Abkürzung: \let\b=\liBedingung
1478 \def\liBedingung#1#2{node[bedingung,#1]{\texttt{#2}}}}

\liBedingungWahr Let-Abkürzung: \let\bWahr=\liBedingungWahr
1479 \def\liBedingungWahr#1{node[bedingung,#1]{\texttt{[true]}}}}

\liBedingungFalsch Let-Abkürzung: \let\bFalsch=\liBedingungFalsch
1480 \def\liBedingungFalsch#1{node[bedingung,#1]{\texttt{[false]}}}}

\liKontrollCode Let-Abkürzung: \let\c=\liKontrollCode
1481 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}

\liKontrollTextzeileKnoten Let-Abkürzung: \let\k=\liKontrollTextzeileKnoten
1482 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw

\liKontrollKnotenPfad Let-Abkürzung: \let\p=\liKontrollKnotenPfad
1483 \ExplSyntaxOn
1484 \NewDocumentCommand { \liKontrollKnotenPfad }{ m }
1485 {
1486   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
1487   \seq_set_map:Nnn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
1488   \seq_use:Nn \l_tmpa_seq { ~~~~ }
1489 }
1490 \ExplSyntaxOff

1491

```

2.20 kopf-fusszeilen.sty

```
1492 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1493 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1494 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1495 \ExplSyntaxOn

1496 \fancyhead{}
1497 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1498 \fancyhead[L0,RE]{\scriptsize\today}
1499 \fancyfoot{}
1500 \fancyfoot[LE,R0]{\thepage}
1501 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1502 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1503 \renewcommand{\headrulewidth}{0.4pt}
1504 \renewcommand{\footrulewidth}{0.4pt}

1505 \ExplSyntaxOff

1506
```

2.21 literatur-dummy.sty

```
1507 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1508 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1509 \def\literatur{}

\footcite

1510 % \RequirePackage[stable,multiple]{footmisc}
1511 \NewDocumentCommand{ \footcite } { o m }{}

1512
```

2.22 literatur.sty

```
1513 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1514 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1515 \RequirePackage{csquotes}
1516 \RequirePackage[
1517   bibencoding=utf8,
1518   citestyle=authortitle,
1519   backend=biber,
1520 ]{biblatex}
1521 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1522 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1523 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1524 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1525 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1526 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1527 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1528 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1529 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1530 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1531 % To allow footnotes in the heading
1532 \RequirePackage[stable,multiple]{footmisc}

\literatur

1533 \def\literatur{\printbibliography[heading=subbibliography]}

1534
```

2.23 makros.sty

```
1535 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1536 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1537 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1538 anderen Paket passen]
1539 \RequirePackage{hyperref}
1540 \RequirePackage{graphicx}
    Für die Umgebung liQuellen benötigt.
1541 \RequirePackage{paralist}

\inhaltsverzeichnis
1542 \def\inhaltsverzeichnis {
1543   \begin{mdframed}
1544     \begin{group}
1545       \let\clearpage\relax
1546       \tableofcontents
1547     \end{group}
1548   \end{mdframed}
1549 }

\memph \mephm (\marginpar and \emph)
1550 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1551 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1552 \newcommand{\liPseudoUeberschrift}[1]{
1553   \bigskip
1554   \noindent
1555   \textsf{\textbf{#1}}
1556   \noindent
1557 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1558 \newcommand{\liBeschriftung}[1]{
1559   \par
1560   \noindent
1561   \medskip
1562   \textbf{#1}:
1563   \medskip
1564   \noindent
1565 }

\hinweis
1566 \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.
1567 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

liEinbettung
1568 \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.
1569 \RequirePackage{xparse}
1570 \ExplSyntaxOn
```



```

1571 \NewDocumentEnvironment{ liAntwort } { 0{standard} }
1572 {
1573   \str_case:nn {#1} {
1574     {standard} {
1575       \def\beschriftung{}
1576       \mdfsetup{backgroundcolor=gray!5,linecolor=gray}
1577     }
1578     {richtig} {
1579       \def\beschriftung{richtig}
1580       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1581     }
1582     {falsch} {
1583       \def\beschriftung{falsch}
1584       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1585     }
1586     {muster} {
1587       \def\beschriftung{Musterlösung}
1588       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1589     }
1590   }
1591   \ifx\beschriftung\empty\else
1592     \noindent
1593     \textbf{\beschriftung{:}}
1594   \fi
1595   \begin{mdframed}
1596 }
1597 {\end{mdframed}}

```

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

```

1598 \NewDocumentEnvironment{ liAdditum } { o }
1599 {
1600   \begin{mdframed}[backgroundcolor=yellow!5]
1601     \IfNoValueTF {#1}
1602     { \liPseudoUeberschrift{Additum} }
1603     { \liPseudoUeberschrift{Additum:~#1} }
1604   }
1605 {\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.

```

1606 \NewDocumentEnvironment{ liExkurs }{o +b}{
1607   \vspace{0.2cm}%
1608   \begin{mdframed}[
1609     backgroundcolor=white,
1610     bottomline=false,
1611     innermargin=1cm,
1612     leftline=true,
1613     linecolor=black,
1614     linewidth=0.1cm,
1615     outermargin=1cm,
1616     rightline=false,
1617     topline=false,
1618   ]

```

```

1619 \footnotesize
1620 \noindent%
1621 \textbf{Exkurs:~#1}\par%
1622 \noindent%
1623 #2
1624 \end{mdframed}
1625 \vspace{0.2cm}
1626 }{}

```

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

```

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

```

Weiterführende Literatur:

- Quelle 1
- Quelle 2

```

1627 \cs_new:Npn \listen@punkt #1 {\item #1}
1628 \NewDocumentEnvironment { liQuellen }{ +b }
1629 {
1630 \seq_clear_new:N \l_quellen
1631 \seq_set_split:Nnn \l_quellen {\item} {#1}
1632 \seq_remove_all:Nn \l_quellen {}
1633 \begin{mdframed}[backgroundcolor=white!0]
1634 \footnotesize
1635 \noindent
1636 \textsf{\textbf{Weiterführende~Literatur:}}
1637 \medskip
1638 \begin{compactitem}
1639 \seq_map_function:NN \l_quellen {\listen@punkt}
1640 \end{compactitem}
1641 \end{mdframed}
1642 %
1643 \makeatletter
1644 \par\@afterindentfalse\@afterheading
1645 \makeatother
1646 } {}

```

liLernkartei

```

1647 \NewDocumentEnvironment { liLernkartei }{ m +b }
1648 {
1649 \begin{mdframed}
1650 \footnotesize
1651 \noindent%
1652 \textbf{Lernkarteikarte:~#1}\par%
1653 \noindent%
1654 #2
1655 \end{mdframed}
1656 } {}

```

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

```

1657 \NewDocumentEnvironment { liDiagramm }{ m +b }
1658 {
1659 \begin{mdframed}[backgroundcolor=white!0]
1660 \small
1661 \noindent%
1662 \textit{#1}:
1663 \begin{center}

```

```

1664 #2
1665 \medskip
1666 \end{center}
1667 \end{mdframed}
1668 } {}
1669 \ExplSyntaxOff

\liFussnoteUrl \liFussnoteUrl[\langle zusätzlicher-text \rangle]{\langle url \rangle} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
1670 \NewDocumentCommand{\liFussnoteUrl} { o m } {
1671 \footnote{\url{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1672 }
1673

\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.
1674 \NewDocumentCommand{\liFussnoteLink} { o m m } {
1675 \footnote{\href{#3}{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1676 }

\zB
1677 \def\zB{z.\,B. }

\ZB
1678 \def\ZB{Z.\,B. }

\dh
1679 \def\dh{d.\,h. }

1680

```

2.24 master-theorem.sty

1681 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1682 \ProvidesPackage{lehramt-informatik-master-theorem}[2021/04/13]

Let-Abkürzungen

\let\0=\li0

\let\o=\li0mega

\let\T=\liT

\let\t=\liTheta

\liMasterVariablenDeklaration

{3} % a

{3} % b

{\mathcal{O}(1)} % f(n)

\liMasterFallRechnung

% 1. Fall

{für $\varepsilon = 4$: \}

$f(n) = 5n^2 \in \mathcal{O}(n^{\log_2 8 - 4}) = \mathcal{O}(n^{\log_2 4}) = \mathcal{O}(n^2)$

% 2. Fall

$f(n) = 5n^2 \notin \mathcal{T}(n^{\log_2 8}) = \mathcal{T}(n^3)$

% 3. Fall

$f(n) = 5n^2 \notin \mathcal{O}(n^{\log_2 8 + \varepsilon})$

\liMasterWolframLink{T[n]=9T[n/3]\%2B5n^2}

\liMasterVariablenDeklaration

{ } % a

{ } % b

{ } % f(n) ohne \$mathe\$

\liMasterFallRechnung

% 1. Fall

{ }

% 2. Fall

{ }

% 3. Fall

{ }

$\rightarrow T(n) \in \mathcal{T}(n^2 \cdot \log n)$

\liMasterWolframLink{T[n]=9T[n/3]\%2B5n^2}

1683 \ExplSyntaxOn

1684 \RequirePackage{amsmath}

\liRundeKlammer

1685 \def\liRundeKlammer#1{

1686 \negthinspace \left(#1 \right)

1687 }

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

1688 \def\liThetaOhneMathe#1{

1689 \Theta \liRundeKlammer{#1}

1690 }

1691 \def\liTheta#1{

1692 \ifmmode

1693 \liThetaOhneMathe{#1}

1694 \else

1695 \$\liThetaOhneMathe{#1}\$

1696 \fi

1697 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
1698 \def\liOmegaOhneMathe#1{
1699 \Omega \liRundeKlammer{#1}
1700 }
1701 \def\liOmega#1{
1702 \ifmmode
1703 \liOmegaOhneMathe{#1}
1704 \else
1705 $\liOmegaOhneMathe{#1}$
1706 \fi
1707 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
1708 \def\liOOhneMathe#1{
1709 \mathcal{O} \liRundeKlammer{#1}
1710 }
1711 \def\liO#1{
1712 \ifmmode
1713 \liOOhneMathe{#1}
1714 \else
1715 $\liOOhneMathe{#1}$
1716 \fi
1717 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
1718 \def\liTOhneMathe#1#2{
1719 \tl_if_blank:nTF {#1}
1720 {}
1721 {#1 \cdot }
1722 T
1723 \liRundeKlammer{\frac{n}{#2}}
1724 }
1725 \def\liT#1#2{
1726 \ifmmode
1727 \liTOhneMathe{#1}{#2}
1728 \else
1729 $\liTOhneMathe{#1}{#2}$
1730 \fi
1731 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
1732 \def\liRekursionsGleichung{
1733 $T(n) = \liT{a}{b} + f(n)$
1734 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
1735 \def\liBedingungEins{
1736 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
1737 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
1738 \def\liBedingungZwei{
1739 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
1740 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
1741 \def\liBedingungDrei{
1742 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
1743 }
1744 \ExplSyntaxOff

```

\liMasterVariablen

```

1745 \def\liMasterVariablen{
1746   \begin{displaymath}
1747     T(n) = \liT{a}{b} + f(n)
1748   \end{displaymath}
1749
1750   \begin{itemize}
1751     \item[$a = $]
1752       Anzahl der Unterprobleme in der Rekursion
1753
1754     \item[$\textstyle{\frac{1}{b}} = $]
1755       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
1756       repräsentiert wird
1757
1758     \item[$f(n) = $]
1759       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
1760       die Kombination der Teillösungen entstehen
1761   \end{itemize}
1762   \footcite{wiki:master-theorem}
1763   \footcite[Seite 19-35 (PDF 11-24)]{aud:fs:2}
1764 }
```

\liMasterFaelle

```

1765 \def\liMasterFaelle{
1766   \begin{description}
1767     \item[1. Fall:]
1768        $T(n) \in \liTheta{n^{\log_{sb{b}}a}}$ 
1769
1770       \hfill falls \liBedingungEins
1771       für  $\varepsilon > 0$ 
1772
1773     \item[2. Fall:]
1774        $T(n) \in \liTheta{n^{\log_{sb{b}}a} \cdot \log n}$ 
1775
1776       \hfill falls \liBedingungZwei
1777
1778     \item[3. Fall:]
1779        $T(n) \in \liTheta{f(n)}$ 
1780
1781       \hfill falls \liBedingungDrei
1782       für  $\varepsilon > 0$ 
1783       und ebenfalls für ein  $c$  mit  $0 < c < 1$  und alle hinreichend großen  $n$ 
1784       gilt:
1785        $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$ 
1786     \end{description}
1787 }
```

\liMasterVariablenDeklaration

```

1788 \def\liMasterVariablenDeklaration#1#2#3{
1789   \begin{description}
1790     \item[Allgemeine Rekursionsgleichung:] \strut
1791
1792     \liRekursionsGleichung
1793
1794     \item[Anzahl der rekursiven Aufrufe ( $a$ ):] \strut
1795
1796     #1
1797
1798     \item[Anteil Verkleinerung des Problems ( $b$ ):] \strut
1799
1800     um  $\frac{1}{\#2}$  also  $b = \#2$ 
1801
1802     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut
```

```

1803
1804     $#3$
1805
1806     \item[Ergibt folgende Rekursionsgleichung:] \strut
1807
1808      $T(n) = T_{\#1\#2} + \#3$ 
1809     \end{description}
1810 }

\liMasterFallRechnung
1811 \def\liMasterFallRechnung#1#2#3{
1812     \begin{description}
1813     \item[1. Fall:] \liBedingungEins:
1814
1815         #1
1816
1817     \item[2. Fall:] \liBedingungZwei:
1818
1819         #2
1820
1821     \item[3. Fall:] \liBedingungDrei:
1822
1823         #3
1824     \end{description}
1825 }

\liMasterExkurs
1826 \def\liMasterExkurs{
1827     \begin{liExkurs}[Master-Theorem]
1828     \liMasterVariablen
1829
1830     \noindent
1831     Dann gilt:
1832
1833     \liMasterFaelle
1834     \end{liExkurs}
1835 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
1836 \def\liMasterWolframLink#1{
1837     Berechne die Rekursionsgleichung auf WolframAlpha:
1838     \href{https://www.wolframalpha.com/input/?i=\#1\{WolframAlpha\}}
1839 }

1840

```

2.25 mathe.sty

```
1841 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1842 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
1843
1844 % for example \ltimes \rtimes
1845 %\RequirePackage{amssymb}
1846 \RequirePackage{amsmath}
1847
1848 %%
1849 % \mlq \mrq
1850 %%
1851 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
1852 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
1853
```


2.26 minimierung.sty

```

1854 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1855 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
1856 Minimierungsalgorithmus von einem NEA zu einem DEA]
1857 \liLadePakete{typographie}

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


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


1858 \def\liFussnote#1{$x_{\text{\scriptsize #1}}$}

1859 \def\li@fussnote@text#1#2{
1860   \liFussnote{\scriptsize #1}
1861   \quad
1862   {\footnotesize #2}
1863 }

1864 \def\liFussnoteEinsText{
1865   \li@fussnote@text{1}
1866   {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
1867 }

1868 \def\liFussnoteZweiText{
1869   \li@fussnote@text{2}
1870   {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
1871 }

1872 \def\liFussnoteDreiText{
1873   \li@fussnote@text{3}

```

```

1874 {In weiteren Iterationen markierte Zustände.}
1875 }

\liFussnoteVierText
1876 \def\liFussnoteVierText{
1877   \li@fussnote@text{4}
1878   {...}
1879 }

\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.                          |
| $x_4$ | ...                                                                  |



1880 \def\liFussnoten{
1881   \bigskip
1882
1883   \noindent
1884   \liFussnoteEinsText
1885
1886   \noindent
1887   \liFussnoteZweiText
1888
1889   \noindent
1890   \liFussnoteDreiText
1891
1892   \noindent
1893   \liFussnoteVierText
1894 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
1895 \def\liLeereZelle{\emptyset}

\liZustandsPaarVariablenName
1896 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
1897 \def\liZustandsPaar#1#2{
1898   $(
1899     \liZustandsPaarVariablenName_#1,
1900     \liZustandsPaarVariablenName_#2
1901   )$
1902 }

liUebergangsTabelle
1903 \renewcommand{\arraystretch}{1.4}
1904 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
1905   \liPseudoUeberschrift{Übergangstabelle}
1906   \begin{center}
1907     \begin{tabular}{r|l|l}
1908       \textbf{Zustandspaar} & \textbf{#1} & \textbf{#2} \\ \hline
1909     \end{tabular}
1910   \end{center}
1911 }
1912 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle: Minimierungstabelle (Table filling)
1913 \ExplSyntaxOn
1914 \def\liUeberschriftDreiecksTabelle{
1915   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
1916 }

```

\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. —————

```

1917 \def\liMinimierungErklaerung{
1918   %\footcite[Seite~19]{koenig}
1919   \liParagraphMitLinien{
1920     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
1921     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
1922     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
1923     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
1924      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
1925     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
1926     somit~zueinander~( $k - 1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
1927     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
1928     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
1929   }
1930 }
1931 \ExplSyntaxOff
1932

```

2.27 normalformen.sty

```

1933 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1934 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10
1935 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
1936 Attributhülle]
1937 \liLadePakete{mathe}
1938 \directlua{
1939   helfer = require('lehramt-informatik-helfer')
1940   normalformen = require('lehramt-informatik-normalformen')
1941 }
```

Let-Abkürzungen

```

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

```

1942 \def\liTeilen#1{
1943   \directlua{tex.print(normalformen.teilen('#1'))}
1944 }
```

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

```

1945 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(#1)}
1946 \def\liAttributHuelle#1{
1947   \ifmmode
1948     \liAttributHuelleOhneMathe{#1}
1949   \else
1950     $\liAttributHuelleOhneMathe{#1}$
1951   \fi
1952 }
```

\liAttributMenge Let-Abkürzung: `\let\m=\liAttributMenge`
 1953 \def\liAttributMenge#1{\{ \textit{#1} \}}

liAHuelle

```

1954 \NewDocumentEnvironment{ liAHuelle } { +b } {
1955   \begingroup
1956   \footnotesize
1957   \begin{multline*}
1958     #1
1959   \end{multline*}
1960   \endgroup
1961 } { }
```

AttributHuelleLinksReduktion Nur innerhalb von `liAHuelle` zu verwenden bzw. `multline`
Let-Abkürzung: `\let\ahL=\liAttributHuelleLinksReduktion`
`\ahL{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}`

```

1962 \def\liAttributHuelleLinksReduktion#1#2#3{
1963   \shoveleft{
1964     \liAttributHuelleOhneMathe{FA, \liAttributMenge{\liAttributMenge{#1} - \liAttributMenge{#2}}
1965   } \\\
1966   \shoveright{
1967     \liAttributMenge{#3}
1968   } \\\
1969 }
```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
    \liFunktionaleAbhaengigkeit{A, B -> C, D} Regulärer Ausdruck zum Konvertieren:
    \$(.*) \rightarrow (.*)\$ \fa{$1 -> $2}
1970 \def\liFunktionaleAbhaengigkeit#1{%
1971     \directlua{normalformen.drucke_funk_abhaengigkeit('#1')}%
1972 }

\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;
1973 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
1974     \par
1975     \noindent
1976     #1 $= \{$
1977     \par
1978     \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
1979     \par
1980     \noindent$\}$
1981 }

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

```

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

```

    Regulärer Ausdruck zum Konvertieren: \$(R.*)\((.*)\)\$ \liRelation[$1]{$2}
1982 \NewDocumentCommand {\liRelation} { O{R} m } {
1983     $\directlua{
1984         local name = helper.konvertiere_tiefgestellt('#1')
1985         tex.print(name)
1986     }\textit{\,#2\,}
1987 }
1988

```

2.28 petri.sty

1989 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 1990 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]

Let-Abkürzungen

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

1991 \RequirePackage{tikz}
 1992 \usetikzlibrary{petri,arrows.meta}

Für die Darstellungsmatrix

1993 \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

```

1994 \def\liPetriSetzeSchluessel{%
1995   \def\TmpTransitionOne{}%
1996   \def\TmpTransitionTwo{}%
1997   \def\TmpTransitionThree{}%
1998   \def\TmpTransitionFour{}%
1999   \def\TmpTransitionFive{}%
2000   \def\TmpTransitionSix{}%
2001   \def\TmpTransitionSeven{}%
2002   \def\TmpTransitionEight{}%
2003   \def\TmpTransitionNine{}%
2004   \def\TmpTransitionTen{}%
2005   \pgfkeys{/petri/.cd,
2006     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2007     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2008     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2009     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
2010     p5/.store in=\TmpPlaceFive,p5/.default=0,p5,

```

```

2011     p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2012     p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2013     p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2014     p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2015     p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2016     t1/.store in=\TmpTransitionOne,t1/.default=activated,
2017     t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2018     t3/.store in=\TmpTransitionThree,t3/.default=activated,
2019     t4/.store in=\TmpTransitionFour,t4/.default=activated,
2020     t5/.store in=\TmpTransitionFive,t5/.default=activated,
2021     t6/.store in=\TmpTransitionSix,t6/.default=activated,
2022     t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2023     t8/.store in=\TmpTransitionEight,t8/.default=activated,
2024     t9/.store in=\TmpTransitionNine,t9/.default=activated,
2025     t10/.store in=\TmpTransitionTen,t10/.default=activated,
2026     scale/.store in=\TmpScale,scale/.default=0.5,
2027     x/.store in=\TmpX,x/.default=5,
2028     y/.store in=\TmpY,y/.default=5,
2029   }%
2030 }

2031 \tikzset{
2032   li petri/.style={
2033     activated/.style={
2034       very thick
2035     },
2036     inhibitor/.style={
2037       {Circle[open,length=2mm,fill=white]}-
2038     }
2039   }
2040 }

```

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

```

2041 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2042 \def\liPetriTransitionsName#1{
2043   \ifmmode
2044     \liPetriTransitionsNameOhneMathe{#1}
2045   \else
2046     $\liPetriTransitionsNameOhneMathe{#1}$
2047   \fi
2048 }

```

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

```

2049 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2050   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2051 }

```

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

```

2052 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2053 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2054 }

```

2.29 potenzmengen-konstruktion.sty

```
2055 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2056 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2057 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2058 \liLadePakete{formale-sprachen}
2059 \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}
    }
  }
}
```

```
2060 \def\liZustandsMengenSammlung#1#2{
2061   \liZustandsnameGross{#1}
2062   {
2063     \footnotesize
2064     \liPotenzmenge{
2065       \str_case:nn {#1} {#2
2066       }
2067     }
2068 }
```

```
\liZustandsMengenSammlungNr
```

```
2069 \def\liZustandsMengenSammlungNr#1#2{
2070   \liZustandsnameGross{#1}
2071   {
```



```
2072     \footnotesize
2073     \liZustandsmengeNr{
2074         \str_case:nn {#1} #2
2075     }
2076 }
2077 }

2078 \ExplSyntaxOff
2079
```

2.30 pseudo.sty

2080 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2081 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen
 2082 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> <div style="margin-left: 20px;"> <p>wähle eine Kante $e \in L$ mit kleinstem Kantengewicht;</p> <p>entferne die Kante e aus L;</p> <p>if der Graph $(V, E' \cup \{e\})$ keinen Kreis enthält then</p> <div style="margin-left: 20px;"> <p>$E' \leftarrow E' \cup \{e\}$;</p> </div> <p>end</p> </div> <p>end</p> <p>Result: $M = (V, E')$ ist ein minimaler Spannbaum von G.</p>

2083 \RequirePackage[german,boxruled]{algorithm2e}

2084

2.31 pumping-lemma.sty

2085 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2086 \ProvidesPackage{lehrant-informatik-pumping-lemma}[2021/08/11 Enthält die
 2087 Definitionen für das Pumping-Lemma in der Regulären Sprache und
 2088 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2089 \def\liPumpingRegulaer{%
2090   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2091   alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2092    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2093   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2094
2095   \begin{enumerate}
2096     \item  $|v| \geq 1$ 
2097     (Das Wort  $v$  ist nicht leer.)
2098
2099     \item  $|uv| \leq j$ 
2100     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2101
2102     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w$  in  $L$ 
2103     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2104     Sprache  $L$ )
2105   \end{enumerate}
2106
2107   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2108   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2109 }
```

\liPumpingKontextfrei

```
2110 \def\liPumpingKontextfrei{%
2111   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2112   sich alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  zerlegen lassen in
2113    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2114
2115   \begin{enumerate}
2116     \item  $|vx| \geq 1$ 
2117     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2118
2119     \item  $|vwx| \leq j$ 
2120     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2121
2122     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy$  in  $L$  (Für jede
2123     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2124     Sprache  $L$ )
2125   \end{enumerate}
2126 }
2127
```

2.32 quicksort.sty

```

2128 % https://tex.stackexchange.com/a/142634
2129 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2130 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2131
2132 %-----
2133 % USAGE:
2134 % \QSinitialize{comma, separated, numerical, values}
2135 % \loop
2136 % \QSpivotStep
2137 % \ifnum\value{pivotcount}>0
2138 %   \QSSortStep
2139 % \repeat
2140 %-----
2141
2142 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2143 \RequirePackage{xintfrac, xinttools}
2144
2145 \RequirePackage{tikz}
2146
2147 %-----
2148 % FIRST PART: TikZ styles and macros for the actual drawing
2149 \newcounter{cellcount}% used for coordinates of the node
2150 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2151
2152 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2153 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2154 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2155 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2156 % by police of LaTeX good conduct ? )
2157 \tikzset{ll/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2158          oo/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2159          rr/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2160 % this is the "b" style as used in the image below
2161          bb/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2162 % nicer:
2163          bb/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2164          gg/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2165
2166 % NOTE the b style was originally the same as the r(aised) style apart from
2167 % not being raised, but I find it nicer with a somewhat different
2168 % specification. I have not updated the images though.
2169
2170 % How the nodes are drawn depending on whether on the left of the pivot value
2171 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2172
2173 \def\DecoLEFT #1{%
2174   \xintFor* ##1 in {#1} \do
2175     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2176 }
2177
2178 \def\DecoINERT #1{%
2179   \xintFor* ##1 in {#1} \do
2180     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2181 }
2182
2183 \def\DecoRIGHT #1{%
2184   \xintFor* ##1 in {#1} \do
2185     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2186 }
2187
2188 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2189   \xintFor* ##1 in {#1} \do

```

```

2190     {\stepcounter{cellcount}}%
2191     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2192 }
2193
2194 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2195     \xintFor* ##1 in {#1} \do
2196     {\stepcounter{cellcount}}%
2197     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2198 }
2199
2200 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2201     \xintFor* ##1 in {#1} \do
2202     {\stepcounter{cellcount}}%
2203     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2204 }
2205
2206 %-----
2207 % SECOND PART: the actual sorting routines.
2208
2209 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2210 \def\QS@sort@b #1{\ifcase #1
2211     \expandafter\QS@sort@empty
2212     \or\expandafter\QS@sort@single
2213     \else\expandafter\QS@sort@c
2214     \fi
2215 }%
2216 \def\QS@sort@empty #1{}
2217 \def\QS@sort@single #1{\QS@Ir {#1}}
2218
2219 % This step is to pick the last as pivot.
2220 \def\QS@sort@c #1%
2221     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2222
2223 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2224 % The trick with \xintApplyUnbraced is that for example when selecting
2225 % the elements smaller than pivot, if we had been using \xintApply we
2226 % would have had at the minimum an empty brace pair. Thus we use the
2227 % "unbraced" variant, but then the \QS@select@smaller has added in
2228 % anticipation a level of braces.
2229 \def\QS@sort@d #1#2{%
2230     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2231     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2232     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2233 }%
2234 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2235 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2236 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2237
2238 %
2239 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2240 %
2241 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2242 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2243 % latter must handle correctly an empty argument.
2244
2245 %-----
2246 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QS@initialize.
2247
2248 % This draws all with suitable highlighting for the newly chosen pivots
2249 % (which will be shown raised)

```

```

2250 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2251             \let\QSIr\DecoINERT
2252             \let\QSIrr\DecoINERT
2253             \let\QSRr\DecoRIGHTwithPivot
2254 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2255             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2256             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2257 }
2258
2259 % This sorts and then draws, showing where the pivot chosen in the previous
2260 % step go. Next time they will have become "inert". If pivotcount is still at
2261 % zero on exit from \QSpivotStep, then this is the signal to stop before
2262 % executing \QSsortStep.
2263 \def\QSsortStep {\def\QSLr {\noexpand\QS@sort@a}%
2264             \def\QSRr {\noexpand\QS@sort@a}%
2265             \def\QSIr {\noexpand\QSIrr}%
2266             \let\QSIrr\relax
2267             \edef\QS@list{\QS@list}%
2268             \let\QSLr\relax
2269             \let\QSRr\relax
2270             \let\QSIr\relax
2271             \edef\QS@list{\QS@list}%
2272             \let\QSLr\DecoLEFT
2273             \let\QSIr\DecoINERTwithPivot
2274             \let\QSIrr\DecoINERT
2275             \let\QSRr\DecoRIGHT
2276 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2277             \setcounter{cellcount}{0}%
2278             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2279 }
2280
2281 \def\QSinitialize #1{%
2282     % first, we convert the comma separated values into a list of braced items
2283     % we use an \edef, and anyhow many \edef's will be used later
2284     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2285     \let\QSRr\DecoRIGHT
2286     % The \QSRr marker mutated to draw the last element as
2287     % pivot and the earlier ones with the suitable style.
2288     %
2289     % The list of marked braced items \QS@list is used both for drawing
2290     % (as here) and for doing the exchange of elements during sort.
2291     \par\centerline{\rule[1.5mm]{0pt}{8mm}}\setcounter{cellcount}{0}%
2292             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2293 }
2294

```

2.33 relationale-algebra.sty

```

2295 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2296 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]
2297 \RequirePackage{amsmath}
2298 \RequirePackage{amssymb}

```

```

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

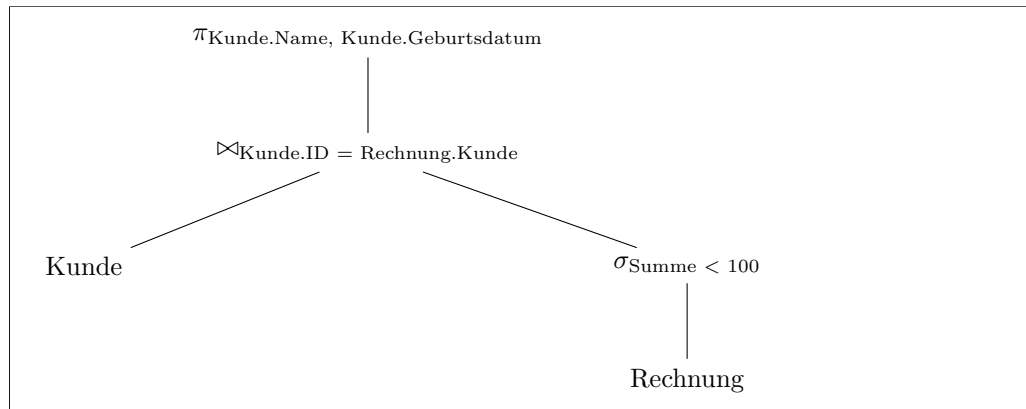
  \node[below=of pi]
    (theta join) {\bowtie_{\text{Kunde.ID = Rechnung.Kunde}}};
    edge (pi);

  \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}

```



```

2299 \RequirePackage{tikz}
2300 \usetikzlibrary{positioning}

    Privates Makros, das zwei Querstriche erzeugt.
2301 \def\o@join{\setbox0=\hbox{\bowtie}}%
2302   \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2303 }

\leftouterjoin A \leftouterjoin B: A ⋈ B
2304 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}

\rightouterjoin A \rightouterjoin B: A ⋉ B
2305 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}

\fullouterjoin A \fullouterjoin B: A ⋈⋉ B
2306 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
2307

```

2.34 rmodell.sty

```

2308 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2309 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01
2310 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2311 Datenbanken.]
2312 \RequirePackage{soul}

```

Let-Abkürzungen

```

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

```

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

```

2313 \def\liPrimaer#1{\ul{#1}}

```

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

```

2314 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}

```

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

```

2315 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2316 \ExplSyntaxOn
2317 \NewDocumentEnvironment { liRmodell }
2318 { +b }
2319 {
2320   \medskip
2321   {
2322     \linespread{2}
2323     \setlength{\parindent}{0pt}
2324     \li@Rmodell@Schrift#1
2325   }
2326   \medskip
2327 } {}
2328 \ExplSyntaxOff

```

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

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

```

2329 \def\liRelationMenge#1#2{
2330 \noindent
2331 #1 : \[ #2 ]\}
2332 \par
2333 }

```

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

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

```

2334 \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}

```

```

2335 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}

```

```

2336

```


2.35 sortieren.sty

```
2337 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2338 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
2339 für händisches Sortieren bzw. einen Schreibtischlauf setzen]
```

```
\tikz[
  rectangle split parts=5,
]{
  \node[li sortierung zahlenreihe] (reihe) {\nodepart{one} 2 \nodepart{two} 1 \nodepart{three} 3 \nodepart{four} 4 \nodepart{five} 5}
  \liSortierPfeil{one}{two}
  \liSortierPfeil{two}{three}
  \liSortierMarkierung{two split south}{three split north}
  \liSortierPfeilUnten{three}{one}
}
```



```
2340 \RequirePackage{tikz}
2341 \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.

```
2342 \def\liVertauschen#1{
2343   \directlua{
2344     local sortieren = require('lehramt-informatik-sortieren')
2345     sortieren('#1')
2346   }
2347 }
```

`\liSortierPfeil`

```
2348 \def\liSortierPfeil#1#2{
2349   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2350 }
```

`\liSortierPfeilUnten`

```
2351 \def\liSortierPfeilUnten#1#2{
2352   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2353 }
```

`\liSortierMarkierung`

```
2354 \def\liSortierMarkierung#1#2{\node[
2355   draw,
2356   very thick,
2357   fit=(reihe.#1) (reihe.#2),
2358   inner sep=0pt
2359 ] {}
2360 }

2361 \tikzset{
2362   li sortierung zahlenreihe/.style={
2363     draw,
2364     thin,
2365     font=\large,
2366     rectangle split horizontal,
2367     rectangle split,
2368   }
2369 }
```

```

2370 % https://tex.stackexchange.com/a/140895
2371 \RequirePackage{forest,xstring}
2372 \usetikzlibrary{calc}
2373
2374 \makeatletter
2375 \pgfmathdeclarefunction{strrepeat}{2}{%
2376   \begingroup\pgfmathint{#2}\pgfmathcount\pgfmathresult
2377   \let\pgfmathresult\pgfutil@empty
2378   \pgfutil@loop\ifnum\pgfmathcount>0\relax
2379     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2380     \advance\pgfmathcount-1\relax
2381   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2382 \makeatother
2383
2384 \def\myNodes{}
2385
2386 \ExplSyntaxOn
2387 \newcommand*\sortList[1]{%
2388   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2389 \ExplSyntaxOff
2390
2391 \forestset{
2392   sort/.code={%
2393     \pgfmathparse{level()>\forestSortLevel}%
2394     \ifnum\pgfmathresult=0
2395       \StrSubstitute{\forestov{content}}{ }{,}[\myList]%
2396       \sortList\myList
2397       \StrSubstitute{\myList}{,}{ }[\myList]%
2398       \pgfmathparse{strrepeat("1",level())}%
2399       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-1!({\forestov{name}}$)
2400         (m\forestov{name}) {\myList}}%
2401       \pgfmathparse{level()=\forestSortLevel}%
2402       \ifnum\pgfmathresult=1
2403         \forestOget{\forestov{@first}}{name}\forestFirst
2404         \forestOget{\forestov{@last}}{name}\forestLast
2405         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2406           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2407       \fi
2408       \ifnum\forestov{@parent}=0\else
2409         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2410       \fi
2411       \gappto\myNodes{;}%
2412     \fi}}
2413
2414 \forestset{sort level/.code=%
2415   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2416   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2417

```

2.36 spalten.sty

```
2418 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2419 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2420 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2421 realisiert werden kann.]
2422 \RequirePackage{multicol}
```

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

```
2423 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2424
```

2.37 struktogramm.sty

```
2425 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2426 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2427 Paket struktex zum Setzen von Struktogrammen]
2428 \RequirePackage{struktex}
2429
```

2.38 syntax.sty

```

2430 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2431 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2432 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2433 \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

```

```

2434 \ExplSyntaxOn
2435 \directlua{
2436   syntax = require('lehramt-informatik-syntax')
2437   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2438   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2439   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2440   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2441   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2442   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2443 }
2444 \RequirePackage{hyperref}
2445 \RequirePackage{minted}
2446 % pygmentize -L styles
2447 \usemintedstyle{colorful}
2448 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2449 %\AfterEndEnvironment{minted}{\end{mdframed}}
2450 %\setminted{breaklines=true,linenos}
2451 \setminted{
2452   breaklines=true,
2453   linenos,
2454   fontsize=\footnotesize,
2455 }

```

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

Let-Abkürzung: `\let\j=\liJavaCode`

```

2456 \def\liJavaCode#1{\,\mintinline{java}|#1|\,}

```

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

```

2457 \def\liLatexCode#1{\mintinline{latex}|#1|}

```

```

2458 \def\li@GithubLink#1#2{
2459   \begin{flushright}
2460     \tiny
2461     Code-Beispiel~auf~Github~ansehen:~
2462     \href{#1}{\nolinkurl{#2}}
2463   \end{flushright}
2464 }

```

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

```

2465 \NewDocumentCommand{\liJavaDatei}{ O{firstline=3} m }{
2466   \inputminted[#1]{java}{
2467     \directlua{
2468       syntax.drucke_absoluten_pfad('#2', false)
2469     }
2470   }
2471   \li@GithubLink

```

```

2472     {\directlua{syntax.drucke_github_url('#2', false)}}
2473     {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2474 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2475 \NewDocumentCommand{\liJavaTestDatei}{ 0{firstline=3} m }{
2476   \inputminted[#1]{java}{
2477     \directlua{
2478       syntax.drucke_absoluten_pfad('#2', true)
2479     }
2480   }
2481   \li@GithubLink
2482   {\directlua{syntax.drucke_github_url('#2', true)}}
2483   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2484 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([.
\liJavaExamen$1{$2}{$3}{$4}{$5}
2485 \NewDocumentCommand{\liJavaExamen}{ 0{firstline=3} m m m m }{
2486   \inputminted[#1]{java}{
2487     \directlua{
2488       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2489     }
2490   }
2491   \li@GithubLink
2492   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2493   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2494   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2495 }

\liAssemblerCode
2496 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2497 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2498   \inputminted{asm}{#1}
2499 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2500 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2501   \inputminted{componentpascal}{#1}
2502 }

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2503 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2504 \NewDocumentCommand{\liHaskellDatei}{ m }{
2505   \inputminted{haskell}{#1}
2506 }

2507 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2508 \def\liSqlCode#1{\mintinline{sql}|#1|}

2509

```

2.39 syntaxbaum.sty

```
2510 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2511 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2512 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
2513 \RequirePackage{tikz-qtrees}
2514
2515 \tikzset{li parsetree/.style={
2516     every internal node/.style={
2517         draw,circle
2518     },
2519     every leaf node/.style={
2520         draw,rectangle
2521     },
2522 }
2523 }
2524
```

2.40 synthese-algorithmus.sty

```
2525 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2526 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2527 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2528 Relation in die 3. Normalform]

2529 \liLadePakete{normalformen,mathe,typographie}
2530 \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\schritt=\liSyntheseUeberschrift

```

2531 \def\liSyntheseUeberschrift#1{
2532   {
2533     \bfseries
2534     \sffamily
2535     \str_case:nn {#1} {
2536       {1} {Kanonische-Überdeckung}
2537       {1-1} {Linksreduktion}
2538       {1-2} {Rechtsreduktion}
2539       {1-3} {Löschen-leerer-Klauseln}
2540       {1-4} {Vereinigung}
2541       {2} {Relationsschemata-formen}
2542       {3} {Schlüssel-hinzufügen}
2543       {4} {Entfernung-überflüssiger-Teilschemata}
2544     }
2545   }
2546 }

```

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

```

2547 \def\li@synthese@erklaerung@texte#1{
2548   \str_case:nn {#1} {
2549     {1} {
2550       Die-kanonische-Überdeckung---also-die-kleinst-mögliche-noch-
2551       äquivalente-Menge-von-funktionalen-Abhängigkeiten-kann-in-vier-
2552       Schritten-erreicht-werden.
2553     }
2554     {1-1} {
2555       Führe-für-jede-funktionale-Anhängigkeit-
2556       $\alpha\rightarrow\beta$-in-$F$-die-Linksreduktion-durch,-
2557       überprüfe-also-für-alle-
2558       $A\in\alpha$,~ob-$A$-überflüssig-ist,-d.h.-ob-
2559       $\beta\subseteqq\liAttributHuelle{F,\alpha\cup A}$.
2560     }
2561     {1-2} {
2562       Führe-für-jede-(verbliebene)-funktionale-Abhängigkeit-$\alpha\rightarrow\beta$-die-Rechtsreduktion-durch,-überprüfe-also-für-
2563       alle-$B\in\beta$,~ob-$B\in\liAttributHuelle{F,\alpha\rightarrow\beta\cup B}$-gilt.-In-diesem-Fall-ist-$B$-auf-der-rechten-Seite-
2564       überflüssig-und-kann-eliminiert-werden,-\dh-$\alpha\rightarrow\beta$-wird-durch-$\alpha\rightarrow\beta\cup B$-
2565       ersetzt.
2566     }
2567     {1-3} {
2568       Entferne-die-funktionalen-Abhängigkeiten-der-Form-$\alpha\rightarrow\emptyset$,~die-im-2.-Schritt-möglicherweise-
2569       entstanden-sind.
2570     }
2571     {1-4} {
2572       Fasse-mittels-der-Vereinigungsregel-funktionale-Abhängigkeiten-
2573       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.
2574     }
2575     {2} {
2576       % Kemper Seite 197
2577       Erzeuge-für-jede-funktionale-Abhängigkeit-$\alpha\rightarrow\beta\sb{c}$-ein-Relationenschema-$\mathcal{R}\sb{\alpha}$-
2578       :=$\alpha\cup\beta$.
2579     }
2580     {3} {
2581       Falls-eines-der-in-Schritt-2.-erzeugten-Schemata-$R\sb{\alpha}$-
2582       einen-Schlüsselkandidaten-von-$\mathcal{R}$-bezüglich-$F\sb{c}$-
2583     }
2584   }
2585 }

```

```

2591 enthält,~sind-wir~fertig,~sonst~wähle-einen-Schlüsselkandidaten-
2592  $\mathcal{K} \sim \text{subseq} \mathcal{R}$ ~aus-und-definiere-folgendes-
2593 zusätzliche-Schema:~ $\mathcal{R} \setminus \mathcal{K} \sim \mathcal{K}$ ~und- $\mathcal{F} \setminus \mathcal{K} \sim \emptyset$ 
2594 }
2595 {4} {
2596 Eliminiere-diejenigen-Schemata- $\mathcal{R} \setminus \alpha$ ~,~die-in-einem-
2597 anderen-Relationenschema- $\mathcal{R} \setminus \alpha'$ ~-enthalten-sind,~d.h.~
2598  $\mathcal{R} \setminus \alpha \sim \text{subseq} \mathcal{R} \setminus \alpha'$ ~.
2599 }
2600 }
2601 }
2602 }
2603 \def\liSyntheseErklaerung#1{
2604 {
2605 \itshape
2606 \footnotesize
2607 \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
2608 }
2609 }

```

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

```

2610 \def\liSyntheseUeberErklaerung#1{
2611 \liSyntheseUeberschrift{#1}\par
2612 \liSyntheseErklaerung{#1}
2613 }

2614 \ExplSyntaxOff
2615

```

2.41 tabelle.sty

2616 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2617 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]

2618 \RequirePackage{tabularx}

2619

2.42 typographie.sty

```
2620 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2621 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
2622 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
2623 formatierung.sty definiert.]
```

```
2624 \ExplSyntaxOn
```

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

```
2625 \RequirePackage{fontawesome}
```

```
\liErledigt \liErledigt: ☑
```

```
2626 \let\liErledigt=\faCheckSquareO
```

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

```
2627 \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 —

```
2628 \def\liParagraphMitLinien#1{
```

```
2629 \noindent
```

```
2630 \vrule height 2pt depth -1.6pt width 0.4cm
```

```
2631 \enspace
```

```
2632 #1
```

```
2633 \enspace
```

```
2634 \leaders\vrule height 2pt depth -1.6pt \hfill \null
```

```
2635 \par
```

```
2636 \medskip
```

```
2637 }
```

```
2638 \ExplSyntaxOff
```

```
2639
```

2.43 uml.sty

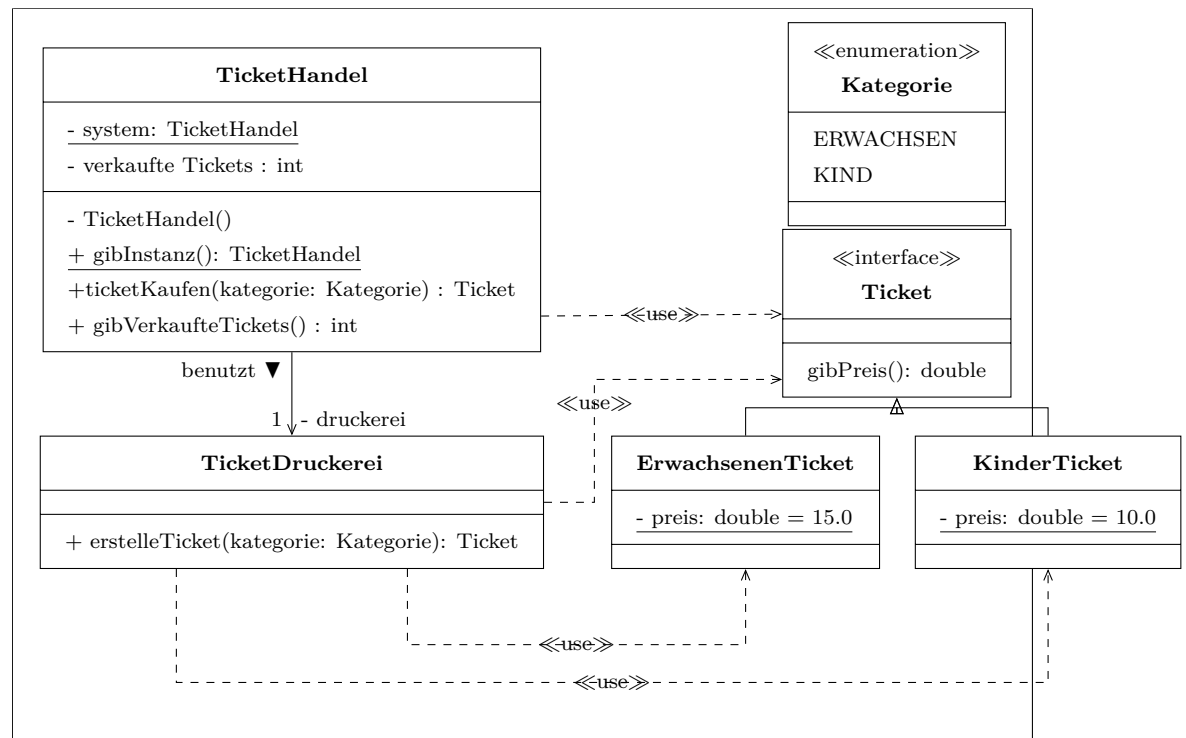
```

2640 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2641 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
2642 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
2643 Erweiterung bereitstellt]

2644 \RequirePackage{tikz-uml}
2645 \RequirePackage{tikz-uml-activity}
2646 % Not compatible with wasysym
2647 %\RequirePackage{mathabx}
2648 \RequirePackage{wasysym}
2649 \usetikzlibrary{positioning}

2650 \tikzumlset{
2651   fill class=white!0,
2652   font=\footnotesize,
2653   fill object=white!0,
2654   fill note=white!0,
2655   fill state=white!0,
2656   % Use case
2657   fill usecase=white!0,
2658   fill system=white!0,
2659 }

```



```

\liUmlLeserichtung \umluniassoc[arg1=,mult2=1,arg2=- druckerei,name=benutzt]{TicketHandel}{TicketDruckerei}
\liUmlLeserichtung[pos=below left,dir=down,distance=0cm]{benutzt}

```

```

2660 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
2661   \def\@liDirLeft{}
2662   \def\@liDirRight{ \RIGHTarrow}
2663   \pgfkeys{/lese/dir/.is choice}
2664   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
2665   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
2666   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{\LEFTarrow }}}
2667   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
2668
2669   \def\@liPos{above}
2670   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
2671

```

```

2672 \def\@liDistance{0cm}
2673 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
2674
2675 \pgfkeys{/lese/.cd,#1}
2676
2677 \node[\@liPos = \@liDistance of #2-middle] {
2678   \@liDirLeft{\footnotesize#2}\@liDirRight
2679 };
2680 }
2681

```

2.44 vollstaendige-induktion.sty

2682 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2683 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01
 2684 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
 2685 Überschriften für die einzelnen Schritte]

Let-Abkürzungen

\let\m=\liInduktionMarkierung
 \let\e=\liInduktionErklaerung

```
\begin{align*}
C_{n+1}
&= \frac{
  \{(4 \cdot (\m{n + 1} - 1) + 2) \cdot \text{cn}(\m{n + 1} - 1)\}
  \{\m{n + 1} + 1\}
}{
  \{(4\m{n} + 2) \cdot \text{cn}(\m{n})\}
  \{\m{n + 2}\}
}
&\text{\e{Java nach Mathe}}\\
\%
&= \frac{
  \{(4\m{n} + 2) \cdot \text{cn}(\m{n})\}
  \{\m{n + 2}\}
}{
  \{(4n + 2) \cdot \m{(2n)!}\}
  \{(n + 2) \cdot \m{(n + 1)!} \cdot n!\}
}
&\text{\e{für cn(n) Formel eingesetzt}}\\
\%
&= \frac{
  \{(4n + 2) \cdot (2n)! \cdot \m{\cdot (n + 1)}\}
  \{(n + 2) \cdot (n + 1)! \cdot n! \cdot \m{\cdot (n + 1)}\}
}{
  \{(n + 2) \cdot (n + 1)! \cdot n! \cdot \m{\cdot (n + 1)}\}
}
&\text{\e{\$(n + 1)\$ multipliziert}}\\
\%
&= \frac{
  \{(4n + 2) \cdot \m{(n + 1) \cdot (2n)!}\}
  \{(n + 2) \cdot (n + 1)! \cdot n! \cdot \m{(n + 1) \cdot n!}\}
}{
  \{(n + 2) \cdot (n + 1)! \cdot n! \cdot \m{(n + 1) \cdot n!}\}
}
&\text{\e{umsortiert}}\\
\%
&= \frac{
  \{\m{(2(n + 1))!}\}
  \{\m{(n + 2)! \cdot (n + 1)!}\}
}{
  \{\m{(n + 2)! \cdot (n + 1)!}\}
}
&\text{\e{Hilfsgleichungen verwendet}}\\
\%
&= \frac{
  \{(2(\m{n + 1}))!\}
  \{((\m{n + 1}) + 1)! \cdot (\m{n + 1})!\}
}{
  \{((\m{n + 1}) + 1)! \cdot (\m{n + 1})!\}
}
&\text{\e{\$(n + 1)\$ verdeutlicht}}\\
\end{align*}
```

Lade häufig benötigte Pakete

2686 \RequirePackage{lehramt-informatik-typographie}
 2687 \RequirePackage{lehramt-informatik-mathe}
 2688 \RequirePackage{lehramt-informatik-syntax}
 2689 \ExplSyntaxOn

\liInduktionMarkierung Hilfsmakro um Teile von mathematischen Formeln markieren zu können.

Let-Abkürzung: \let\m=\liInduktionMarkierung

2690 \def\liInduktionMarkierung#1{\textcolor{violet}{#1}}

\liInduktionErklaerung Gedacht für die rechte Spalte in der align-Umgebung. Das text-Makro ist dann nicht mehr nötig.

Let-Abkürzung: \let\e=\liInduktionErklaerung

```
2691 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
2692 \def\liInduktionAnfang{
2693   \liPseudoUeberschrift{Induktionsanfang}
2694
2695   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2696   \liParagraphMitLinien{
2697     Beweise,~dass~$A(1)$~eine~wahre~Aussage~ist.
2698   }
2699 }
```

\liInduktionVoraussetzung

```
2700 \def\liInduktionVoraussetzung{
2701   \liPseudoUeberschrift{Induktionsvoraussetzung}
2702
2703   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2704   \liParagraphMitLinien{
2705     Die~Aussage~$A(k)$~ist~wahr~für~ein~beliebiges~$k$ \in \mathbb{N}$$.
2706   }
2707 }
```

\liInduktionSchritt

```
2708 \def\liInduktionSchritt{
2709   \liPseudoUeberschrift{Induktionsschritt}
2710
2711   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2712   \liParagraphMitLinien{
2713     Beweise,~dass~wenn~$A(n=k)$~wahr~ist,~
2714     auch~$A(n=k+1)$~wahr~sein~muss.
2715   }
2716 }

2717 \ExplSyntaxOff
2718
```


2.45 wasserfall.sty

```
2719 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2720 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
2721 \RequirePackage{tikz}
2722 \tikzset{wasserfall/.style={
2723   >=stealth,
2724   node distance = 2mm and -8mm,
2725   start chain = A going below right,
2726   every node/.style = {
2727     draw,
2728     text width=24mm,
2729     minimum height=12mm,
2730     align=center,
2731     inner sep=1mm,
2732     fill=white,
2733     drop shadow={fill=black},
2734     on chain=A
2735   },
2736 }}
2737 \usetikzlibrary{chains,positioning,shadows}
2738
```

2.46 wpkalkuel.sty

2739 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2740 \ProvidesPackage{lehramt-informatik-wpkalkuel}[2020/06/13]

Let-Abkürzungen

\let\wp=\liWpKalkuel

\let\equivalent=\liWpEquivalent

\let\erklaerung=\liWpErklaerung

2741 \RequirePackage{amsmath}

2742 \ExplSyntaxOn

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

2743 \def\liWpKalkuelOhneMathe#1#2{

2744 \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)

2745 }

2746 \def\liWpKalkuel#1#2{

2747 \ifmmode

2748 \liWpKalkuelOhneMathe{#1}{#2}

2749 \else

2750 \$\liWpKalkuelOhneMathe{#1}{#2}\$

2751 \fi

2752 }

\MatheEnv

2753 \def\MatheEnv#1{

2754 \medskip

2755

2756 \hspace{1em}#1

2757

2758 \medskip

2759 }

\Mathe

2760 \def\Mathe#1{

2761 \MatheEnv{#1\$}

2762 }

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

2763 \def\liWpEquivalent#1{

2764 \MatheEnv{\$\equiv\$\hspace{1em}\$#1\$}

2765 }

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

2766 \newlength{@Skip@Erklaerung@Reset}

2767 \def\liWpErklaerung#1{

2768 \setlength{@Skip@Erklaerung@Reset}{\leftskip}

2769 \setlength{\leftskip}{0.5cm}

2770

2771 \par

2772 \noindent

2773 {

2774 \scriptsize

2775 #1

2776 }

2777 \par

2778

2779 \setlength{\leftskip}{@Skip@Erklaerung@Reset}

2780 }

\liWpErklaerungVerzweigung

```

2781 \def\liWpErklaerungVerzweigung{
2782   $\liWpKalkuelOhneMathe{if~\{-b~\}~then~\{-a1~\}~else~\{-a2~\}}{Q}
2783   \equiv
2784   (b \land \liWpKalkuelOhneMathe{a1}{Q})
2785   \lor
2786   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
2787 }

2788 \ExplSyntaxOff

2789

```

3 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		
\# 109	\alpha 2556, 2558, 2559,	\bowtie 2301, 2304, 2305, 2306
\, 331, 386,	2562, 2564, 2565,	\Box 159
1174, 1176, 1677,	2566, 2567, 2568,	\boxtimes 463
1678, 1679, 1986, 2456	2572, 2578, 2579,	
\@Skip@Erklaerung@Reset	2584, 2585, 2586,	
... 2766, 2768, 2779	2589, 2597, 2598, 2599	C
\@afterheading 1644	\arabic 1237, 2175, 2180,	\c 1320, 1321
\@afterindentfalse . 1644	2185, 2191, 2197, 2203	\cdot 1721, 1774, 1785
\@liDirLeft 2661, 2666, 2678	\arraystretch 1903	\centerline 1351, 2254, 2276, 2291
\@liDirRight 2662, 2664,		\chapter 1227, 1228
2665, 2666, 2667, 2678	B	\char 1551
\@liDistance 2672, 2673, 2677	\BeforeBeginEnvironment	\clearpage 1545
\@liPos .. 2669, 2670, 2677 2448	\cline 596
\\ .. 596, 619, 620, 623,	\begin 617, 668,	\clist 226, 270,
624, 627, 628, 720,	683, 718, 742, 789,	271, 284, 288, 2388
721, 722, 829, 858,	821, 836, 856, 866,	\columnbreak 2423
860, 886, 895, 940,	882, 902, 934, 951,	\cs 287, 306, 330,
982, 983, 984, 989,	980, 1005, 1026,	331, 368, 380, 1627
990, 991, 1011,	1041, 1153, 1157,	\csname 1305, 1308
1551, 1908, 1965, 1968	1243, 1318, 1346,	\cup 1132, 2565, 2579, 2586
\{ 207, 1109, 1119,	1355, 1362, 1470,	
1131, 1132, 1137,	1543, 1595, 1600,	D
1151, 1173, 1392,	1608, 1633, 1638,	\DeclareMathSymbol ..
1953, 1976, 2331, 2782	1649, 1659, 1663, 1851, 1852
\} 207, 1109, 1119,	1746, 1750, 1766,	\DecoINERT 2178, 2251, 2252, 2274
1131, 1132, 1139,	1789, 1812, 1827,	\DecoINERTwithPivot .
1157, 1177, 1393,	1906, 1907, 1957, 2194, 2273
1953, 1980, 2331, 2782	2095, 2115, 2256,	\DecoLEFT 2173, 2272
_ 27, 35, 44, 46, 287, 306,	2278, 2292, 2448, 2459	\DecoLEFTwithPivot ..
330, 331, 345, 346,	\begingroup 1544, 1955, 2376 2188, 2250
352, 355, 358, 368, 380	\beschriftung 1583, 1587, 1591, 1593	\DecoRIGHT 2183, 2275, 2285
		\DecoRIGHTwithPivot .
A	\beta 2556, 2200, 2253
\addbibresource 2559, 2563, 2564,	2565, 2568, 2578,	\definecolor 1225
1521, 1522, 1523,	2579, 2580, 2585, 2586	\delta 65, 107, 165, 207, 1123
1524, 1525, 1526,	\bf 2153, 2154, 2155	\dh 1679, 2567
1527, 1528, 1529, 1530	\bfseries .. 475, 1227,	\directlua 58, 137, 195, 200,
\advance 2380	1229, 2153, 2159,	1103, 1118, 1138,
\AfterEndEnvironment 2449	2161, 2163, 2164, 2533	1146, 1154, 1160,
\allsectionsfont ... 1223	\bigskip 48, 364,	1938, 1943, 1971,
\Alph 1237	598, 603, 1553, 1881	1978, 1983, 2343,
\alph 1237, 1238	\bool 309, 332	

2435, 2467, 2472, 2473, 2477, 2482, 2483, 2487, 2493, 2494	\erzeuge@tieffgestellt ... 1118, 1119, 1123	\forestov . 2395, 2399, 2400, 2403, 2404, 2405, 2406, 2408, 2409
\do 2174, 2179, 2184, 2189, 2195, 2201	\expandafter 1305, 2209, 2211, 2212, 2213, 2221, 2379	\forestset 2391, 2414 \forestSortLevel 2393, 2401, 2415, 2416
\dots 506, 510, 1392, 2102, 2578, 2579	\ExplSyntaxOff 50, 92, 134, 139, 192, 197, 202, 393, 528, 550, 565, 1097, 1179, 1214, 1490, 1505, 1669, 1744, 1931, 2078, 2328, 2389, 2507, 2614, 2638, 2717, 2788	\frac 1723, 1754, 1785, 1800 \fullouterjoin <u>2306</u>
\DOWNarrow 2665		
\draw 1325, 1328, 1331, 2050, 2349, 2352		
E		G
\edef 1316, 2267, 2271, 2283, 2284		\g 29, 37, 270, 271, 284, 288, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 307, 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
\else 570, 578, 586, 1113, 1127, 1165, 1591, 1694, 1704, 1714, 1728, 1949, 2045, 2213, 2406, 2408, 2749	\ExplSyntaxOn 22, 61, 102, 135, 160, 193, 198, 223, 470, 534, 551, 1070, 1169, 1191, 1483, 1495, 1570, 1683, 1913, 2059, 2316, 2386, 2434, 2530, 2624, 2689, 2742	\Gamma . 106, 164, 207, 1132 \gappto 2411 \geq 1399, 2091, 2096, 2112, 2116
\emph 1073, 1373, 1402, 1404, 1550		
\empty 1591		
\emptyset 1895, 2573, 2594		
\end 656, 679, 704, 739, 773, 805, 833, 849, 863, 871, 898, 927, 948, 973, 998, 1018, 1038, 1056, 1155, 1157, 1245, 1334, 1358, 1359, 1366, 1475, 1548, 1597, 1605, 1624, 1640, 1641, 1655, 1666, 1667, 1748, 1761, 1786, 1809, 1824, 1834, 1910, 1911, 1959, 2105, 2125, 2256, 2278, 2292, 2449, 2463	F \faCheckSquare0 2626, 2627 \faCircleThin 1089 \faGg 1083 \fancyfoot 1249, 1250, 1251, 1499, 1500, 1501, 1502 \fancyhead 1248, 1496, 1497, 1498 \faSquare0 1077 \fi 572, 580, 588, 1115, 1129, 1167, 1594, 1696, 1706, 1716, 1730, 1951, 2047, 2214, 2406, 2407, 2410, 2412, 2751 \fontspec 1223 \footcite 680, 702, 749, 772, 804, 899, 926, 972, 1371, 1374, 1381, 1386, 1391, 1395, 1401, 1406, 1510, 1762, 1763, 1918, 2108 \footnote 1671, 1675 \footnotesize 147, 341, 424, 520, 1094, 1566, 1619, 1634, 1650, 1862, 1956, 2063, 2072, 2315, 2454, 2606, 2652, 2678 \footrulewidth . 1254, 1504 \foreach . 1320, 1323, 1330 \forestFirst .. 2403, 2406 \forestLast ... 2404, 2406 \forestOget ... 2403, 2404 \forestOnes 2416 \forestOv 2405, 2406, 2409	
\endcsname 1305, 1308		H \hbox 2301 \headrulewidth . 1253, 1503 \hfill 1770, 1776, 1781, 2634 \hinweis <u>1566</u> \hline 1908 \href 370, 1187, 1675, 1838, 2462 \hspace .. 2053, 2756, 2764 \ht 2302
\endgroup 1547, 1960, 2381		
\enspace 2631, 2633		I \i 1330, 1331 \ifcase 2210 \ifmmode 568, 576, 584, 1111, 1125, 1163, 1692, 1702, 1712, 1726, 1947, 2043, 2747 \IfNoValueTF 1601, 1671, 1675 \ifnum 2137, 2378, 2394, 2402, 2408 \ifx 1591, 2406 \in 494, 600, 1399, 1736, 1739, 1742, 1768, 1774, 1779, 2091, 2102, 2112, 2122, 2556, 2558, 2564, 2585, 2705 \inhaltsverzeichnis <u>1542</u> \input . 4, 7, 10, 13, 16, 394 \inputminted 2466, 2476, 2486, 2498, 2501, 2505 \int 2388 \item 463, 464, 685, 689, 694, 699, 743, 752, 757,
environments:		
liAdditum <u>1598</u>		
liAHuelle <u>1954</u>		
liAntwort <u>1569</u>		
liDiagramm <u>1657</u>		
liEinbettung <u>1568</u>		
liExkurs <u>1606</u>		
liGraphenFormat . <u>1297</u>		
liKasten <u>1242</u>		
liKontrollflussgraph <u>1469</u>		
liLernkartei <u>1647</u>		
liProduktionsRegeln <u>1147</u>		
liProjektSprache <u>1567</u>		
liQuellen <u>1627</u>		
liRelationenSchemaFormat <u>2335</u>		
liRmodell <u>2315</u>		
liUebergangsTabelle <u>1903</u>		
\equiv 2764, 2783		

765, 837, 842, 846, 867, 903, 908, 915, 923, 952, 957, 961, 966, 1042, 1047, 1052, 1356, 1357, 1627, 1631, 1751, 1754, 1758, 1767, 1773, 1778, 1790, 1794, 1798, 1802, 1806, 1813, 1817, 1821, 2096, 2099, 2102, 2116, 2119, 2122	\itshape 519, 2605		
J			
\j 1320, 1321, 1323, 1324, 1325, 1330, 1331, 1332			
K			
\k 1330			
\keys 31, 70, 82, 112, 122, 170, 180, 292, 538, 542, 556, 561, 1198, 1205			
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, 1193, 1194, 1195, 1196, 1199, 1200, 1201, 1202, 1208, 1209, 1210, 1211, 1486, 1487, 1488, 1630, 1631, 1632, 1639			
\labelenumi 1238			
\labelenumii 1239			
\labelitemi 1232			
\labelitemii 1233			
\labelitemiii 1234			
\labelitemiv 1235			
\land 2784, 2786			
\LARGE 1227			
\large 1351, 2365			
\leaders 2634			
\left 1686			
\LEFTarrow 2666			
\leftarrow 582			
\leftouterjoin 2304			
\leftskip 2768, 2769, 2779			
		\LehramtInformatikAutorEmail\liAnweisung 1477	
	 1502 \liAssemblerCode . . . 2496	
		\LehramtInformatikAutorName \liAssemblerDatei . . 2497	
	 1501 \liAttribut 2334	
		\LehramtInformatikGitBranch \liAttributHuelle . . .	
	 373, 2442 . . . 1945, 2559, 2564	
		\LehramtInformatikGithubCodeRepository\liAttributHuelleLinksReduktion	
	 2441 1962	
		\LehramtInformatikGithubDomain\liAttributHuelleOhneMathe	
	 2438 1945, 1948, 1950, 1964	
		\LehramtInformatikGithubRawDomain\liAttributMenge	
	 371, 2439 . . . 1953, 1964, 1967	
		\LehramtInformatikGithubRepository\liAufgabe 3	
	 372, 2440 \liAufgabenTitel 23	
		\LehramtInformatikRepository\liAusdruck 1170	
	 4, \liAutomat 61	
		7, 10, 13, 16, 1521, \liAutomatenKante 93	
		1522, 1523, 1524, \liBandAlphabet 1132	
		1525, 1526, 1527, \liBedingung 1478	
		1528, 1529, 1530, 2437 \liBedingungDrei	
		\LehramtInformatikTitel 1741, 1781, 1821	
	 1497 \liBedingungEins	
		\leq 1785, 2099, 2119 . . . 1735, 1770, 1813	
		\let 1121, \liBedingungFalsch . 1480	
		1122, 1545, 2250, \liBedingungWahr . . 1479	
		2251, 2252, 2253, \liBedingungZwei	
		2266, 2268, 2269, . . . 1738, 1776, 1817	
		2270, 2272, 2273, \liBeschriftung 1558	
		2274, 2275, 2285, \liChomskyErklaerung	
		2377, 2415, 2416, 2626 485, 526	
		\li@chomsky@erklae rung@texte\liChomskyUeberErklaerung	
	 485, 521 524	
		\li@EntwurfsCode \liChomskyUeberschrift	
	 613, 659, 660, 661, 473, 525	
		707, 708, 709, 710, \liCpmEreignis 534	
		776, 777, 778, 779, \liCpmFruehesterI . . . 591	
		780, 781, 808, 809, \liCpmSpaetesterI . . . 590	
		810, 811, 812, 813, 874 \liCpmVon 574	
		\li@EntwurfsCodeAllgemein \liCpmVonOhneMathe . .	
	 612 574, 577, 579	
		\li@fussnote@text 1859, \liCpmVonZu 566	
		1865, 1869, 1873, 1877 \liCpmVonZuOhneMathe	
		\li@GithubLink 566, 569, 571	
	 2458, 2471, 2481, 2492 \liCpmVorgang 551	
		\li@mget . 1307, 1311, 1331 \liCpmZu 582	
		\li@minc 1310, 1332 \liCpmZuOhneMathe . .	
		\li@mset 582, 585, 587	
	 1304, 1312, 1321, 1324 liDiagramm (environ-	
		\li@numdiscs ment) 1657	
		. . . 1316, 1325, 1331 liEinbettung (environ-	
		\li@Rmodell@Schrift ment) 1568	
		. . . 2315, 2324, 2334 \liEntwurfs 1021	
		\li@sequence . . 1317, 1330 \liEntwurfsAbstrakteFabrik	
		\li@synthese@erklae rung@texte 663	
	 2547, 2607 \liEntwurfsAbstrakteFabrikCode	
		\liAbleitung 1146 658, 665	
		liAdditum (environment) \liEntwurfsAbstrakteFabrikUml	
	 1598 616, 664	
		liAHuelle (environment) \liEntwurfsAdapter . . 712	
	 1954 \liEntwurfsAdapterAkteure	
		\liAlphabet 1131 682, 714	
		liAntwort (environment) \liEntwurfsAdapterCode	
	 1569 706, 715	

\liEntwurfsAdapterUml	\liErledigt	2626	\liLadePakete	54, 57, 224, 229,
..... 667, 713	\liErMpAttribute	1086	472, 533, 1341,	
\liEntwurfsBeobachter 783	\liErMpEntity	1074	1857, 1937, 2058, 2529	
\liEntwurfsBeobachterAkteure	\liErMpRelationship	1080	\liLatexCode	2457
..... 741, 785	\liErRelationship	...	\liLeereZelle	1895
\liEntwurfsBeobachterCode	... 1072, 1081, 1083		liLernkartei (environ-	1647
..... 775, 786	\liExamensAufgabe	6	ment)	
\liEntwurfsBeobachterUml	\liExamensAufgabeA	15	\liMasterExkurs	1826
..... 717, 784	\liExamensAufgabeTA	12	\liMasterFaelle	1765, 1833
\liEntwurfsDekorierer 815	\liExamensAufgabeTTA	9	\liMasterFallRechnung	
\liEntwurfsDekoriererAkteure	liExkurs (environment)	1606 1811	
..... 817	\liFalsch	464	\liMasterVariablen	1745, 1828
\liEntwurfsDekoriererCode	\liFlaci	1180	\liMasterVariablenDeklaration	1788
..... 807, 818	\liFremd	2314 1836	
\liEntwurfsDekoriererUml	\liFunktionaleAbhaengigkeit	1970	\liMasterWolframLink	1836
..... 788, 816 1970		\liMenge	71, 72, 74,
\liEntwurfsEinfacheFabrik	\liFunktionaleAbhaengigkeiten	1973	113, 114, 115, 119,	
..... 851 1973		171, 172, 173, 177,	
\liEntwurfsEinfacheFabrikAkteure	\liFussnote	1858, 1860	1109, 1160, 1199, 1200	
..... 835, 853	\liFussnoteDreiText	1872, 1890	\liMengeOhneMathe	1109, 1112, 1114
\liEntwurfsEinfacheFabrikUml 1872, 1890	 1197	
..... 820, 852	\liFussnoteEinsText	1864, 1884	\liMinimierungErklaerung	1917
\liEntwurfsEinzelstueck 1864, 1884	 2500	
..... 876	\liFussnoteLink	1674	\linespread	2322
\liEntwurfsEinzelstueckAkteure	\liFussnoten	1880	\liNichtsZuTun	2627
..... 865, 878	\liFussnoteUrl	1019, 1670	\liO	1708, 1736
\liEntwurfsEinzelstueckCode	\liFussnoteVierText	1876, 1893	\liOmega	1698, 1742
..... 873, 879 1876, 1893		\liOmegaOhneMathe	1698, 1703, 1705
\liEntwurfsEinzelstueckUml	\liFussnoteZweiText	1868, 1887	\liOOhneMathe	1708, 1713, 1715
..... 855, 877 1868, 1887	 2041, 2053	
\liEntwurfsErbauer	\liGrammatik	1191	\liParagrafMitLinien	521, 1919, 2607,
\liEntwurfsErbauerAkteure	liGraphenFormat (envi-	1297	2628, 2696, 2704, 2712	
..... 901, 931	ronment) 1297		\liPetriErreichKnotenDrei	2052
\liEntwurfsErbauerUml	\liHanoi	1304	\liPetriErreichTransition	2049
..... 881, 930	\liHaskellCode	2503 1994	
\liEntwurfsFabrikmethode	\liHaskellDatei	2504	\liPetriSetzeSchluessel	1994
..... 975, 1000	\liInduktionAnfang	2692 2041, 2053	
\liEntwurfsFabrikmethodeAkteure	\liInduktionErklaerung	2691	\liPetriTransitionsName	2041, 2044, 2046
..... 950, 977 2691	 2053	
\liEntwurfsFabrikmethodeUml	\liInduktionMarkierung	2690	\liPetriTransPfeile	2053
..... 933, 976 2690		\liPolynomiellReduzierbar	1361
\liEntwurfsKompositum	\liInduktionSchritt	2708	\liPotenzmenge	1118, 1122, 2064
..... 1000	\liInduktionVoraussetzung	2700 1119, 1120, 1121	
\liEntwurfsKompositumAkteure 2700		\liPrimaer	2313
..... 1002	\liJavaCode	2456	\liProblemBeschreibung	1345
\liEntwurfsKompositumUml	\liJavaDatei	614, 2465	\liProblemClique	1368
..... 979, 1001	\liJavaExamen	2485	\liProblemName	1344, 1351,
\liEntwurfsModellPraesentation	\liJavaTestDatei	2475	1363, 1365, 1378,	
..... 1021	liKnoten (environment)	1242	1389, 1390, 1398, 1399	
\liEntwurfsModellPraesentationAkteure	\liKellerAutomat	102	\liProblemSat	1397
..... 1023	\liKellerKarte	140		
\liEntwurfsModellPraesentationUml	\liKellerUebergang	135, 141		
..... 1004, 1022 135, 141			
\liEntwurfsZustand	\liKontrollCode	1481		
\liEntwurfsZustandAkteure	liKontrollflussgraph	1469		
..... 1040, 1060	(environment)			
\liEntwurfsZustandUml	\liKontrollKnotenPfad	1483		
..... 1025, 1059 1483			
\liEpsilon	\liKontrollTextzeileKnoten	1482, 1487		
..... 1117 1482, 1487			
\liErAttribute	\liKurzeTabellenLinie	596		
..... 1073, 1087, 1089 596			
\liErDatenbankName	\liLadeAllePakete	228		
\liErEntity 228			
1071, 1075, 1077				

<code>\liProblemSubsetSum</code> 1388 , 1397	<code>\liUeberfuehrungsFunktion</code> 1123	<code>\mdfsetup</code> 1241 , 1576 , 1580 , 1584 , 1588
<code>\liProblemVertexCover</code> 1368 , 1376	<code>\liUeberfuehrungsFunktionOhneMathe</code> ... 1123 , 1126 , 1128	<code>\newtheorem</code> 1353 , 1561 , 1563 ,
<code>\liProduktionen</code> 1159 , 1201	<code>liUebergangsTabelle</code> (environment) 1903	1637 , 1665 , 2320 ,
<code>liProduktionsRegeln</code> (environment) 1147	<code>\liUeberschriftDreiecksTabelle</code> 1913	2326 , 2636 , 2754 , 2758
<code>liProjektSprache</code> (envi- ronment) 1567	<code>\liUmlLeserichtung</code> . 2660	<code>\emph</code> 1550
<code>\liPseudoUeberschrift</code> 1552 , 1602 , 1603 , 1905 , 1915 , 2693 , 2701 , 2709	<code>\liVertauschen</code> 2342	<code>\mintinline</code> 2456 , 2457 , 2496 , 2503 , 2508
<code>\liPumpingKontextfrei</code> 2110	<code>\liWortInSprache</code> 597	<code>\mkern</code> ... 2304 , 2305 , 2306
<code>\liPumpingRegulaer</code> . 2089	<code>\liWortNichtInSprache</code> 602	<code>\mlq</code> 1849 , 1851
<code>liQuellen</code> (environment) 1627	<code>\liWpEquivalent</code> 2763	<code>\mrq</code> 1849 , 1852
<code>\liRekursionsGleichung</code> 1732 , 1792	<code>\liWpErklaerung</code> 2766	<code>\msg</code> 39 , 391
<code>\liRelation</code> 1982	<code>\liWpErklaerungVerzweigung</code> 2781	<code>\myList</code> 2395 , 2396 , 2397 , 2400
<code>liRelationenSchemaFormat</code> (environment) 2335	<code>\liWpKalkuel</code> 2743	<code>\myNodes</code> 2384 , 2399 , 2405 , 2409 , 2411
<code>\liRelationMenge</code> ... 2329	<code>\liWpKalkuelOhneMathe</code> 2743 , 2748 , 2750 , 2782 , 2784 , 2786	N
<code>\liRichtig</code> 463	<code>\liZustandsBuchstabe</code> 1133 , 1142 , 1144 , 1164 , 1166	<code>\NeedsTeXFormat</code> 1 , 19 , 52 , 220 , 398 ,
<code>liRmodell</code> (environment) 2315	<code>\liZustandsBuchstabeGross</code> ... 1134 , 1143 , 1145	459 , 466 , 530 , 593 ,
<code>\liRundeKlammer</code> . 1685 , 1689 , 1699 , 1709 , 1723	<code>\liZustandsmenge</code> ... 1121	608 , 1063 , 1099 ,
<code>\liSetzeAufgabenTitel</code> . 25	<code>\liZustandsmengeNr</code> 1135 , 2073	1216 , 1256 , 1265 ,
<code>\liSortierMarkierung</code> 2354	<code>\liZustandsmengeNrGross</code> 1143	1270 , 1299 , 1337 ,
<code>\liSortierPfeil</code> 2348	<code>\liZustandsMengenSammlung</code> 2060	1409 , 1492 , 1507 ,
<code>\liSortierPfeilUnten</code> 2351	<code>\liZustandsMengenSammlungNr</code> 2069	1513 , 1535 , 1681 ,
<code>\liSpaltenUmbruch</code> .. 2423	<code>\liZustandsmengeOhneMathe</code> 1121	1841 , 1854 , 1933 ,
<code>\liSqlCode</code> 2508	<code>\liZustandsname</code> 1144	1989 , 2055 , 2080 ,
<code>\listen@punkt</code> .. 1627 , 1639	<code>\liZustandsnameGross</code> ... 1145 , 2061 , 2070	2085 , 2129 , 2295 ,
<code>\liStrich</code> 1343	<code>\liZustandsnameTiefgestellt</code> 1162	2308 , 2337 , 2418 ,
<code>\liSyntheseErklaerung</code> 2547 , 2612	<code>\liZustandsPaar</code> 1897	2425 , 2430 , 2510 ,
<code>\liSyntheseUeberErklaerung</code> 2610	<code>\liZustandsPaarVariablenName</code> ... 1896 , 1899 , 1900	2525 , 2616 , 2620 ,
<code>\liSyntheseUeberschrift</code> 2531 , 2611	<code>\llap</code> 2302	2640 , 2682 , 2719 , 2739
<code>\liT</code> 1718 , 1733 , 1747 , 1808	<code>\log</code> 1736 ,	<code>\neg</code> 2786
<code>\liTeilen</code> 1942	1739 , 1742 , 1768 , 1774	<code>\negthinspace</code> 1686
<code>\literatur</code> 1509 , 1533	<code>\loop</code> 2135	<code>\newcounter</code> ... 2149 , 2150
<code>\liTheta</code> 1688 , 1739 , 1768 , 1774 , 1779	<code>\lor</code> 2785	<code>\NewDocumentCommand</code> . 62 , 103 , 140 , 161 ,
<code>\liThetaOhneMathe</code> 1688 , 1693 , 1695	<code>\ltimes</code> 1844	203 , 224 , 535 , 552 ,
<code>\liTOhneMathe</code> 1718 , 1727 , 1729	M	597 , 602 , 1170 ,
<code>\liTuringKante</code> 203	<code>\makeatletter</code> .. 1643 , 2374	1192 , 1361 , 1484 ,
<code>\liTuringLeerzeichen</code> 159 , 167	<code>\makeatother</code> .. 1645 , 2382	1511 , 1670 , 1674 ,
<code>\liTuringMaschine</code> ... 160	<code>\marginpar</code> 1076 , 1082 , 1088 , 1550	1973 , 1982 , 2049 ,
<code>\liTuringUeberfuehrung</code> 206	<code>\mathbb</code> .. 1399 , 2122 , 2705	2465 , 2475 , 2485 ,
<code>\liTuringUebergaenge</code> 198 , 204	<code>\mathbin</code> . 2304 , 2305 , 2306	2497 , 2500 , 2504 , 2660
<code>\liTuringUebergangZelle</code> 193	<code>\mathcal</code> .. 1709 , 2585 , 2590 , 2592 , 2593 , 2594	<code>\NewDocumentEnvironment</code> 1147 , 1242 , 1297 ,
	<code>\Mathe</code> 2760	1469 , 1567 , 1568 ,
	<code>\MatheEnv</code> 2753 , 2761 , 2764	1571 , 1598 , 1606 ,
	<code>\mathord</code> 1851 , 1852	1628 , 1647 , 1657 ,
		1904 , 1954 , 2317 , 2335
		<code>\newlength</code> 2766
		<code>\node</code> 548 , 1477 ,
		1482 , 2175 , 2180 ,
		2185 , 2191 , 2197 ,
		2203 , 2354 , 2399 , 2677
		<code>\noexpand</code> 2263 ,
		2264 , 2265 , 2284 , 2399
		<code>\noindent</code> 343 , 599 , 604 ,
		1150 , 1554 , 1556 ,
		1560 , 1564 , 1592 ,
		1620 , 1622 , 1635 ,
		1651 , 1653 , 1661 ,

1830, 1883, 1886, 1889, 1892, 1975, 1980, 2330, 2629, 2772			
\nolinkurl	2462		
\normalsize	1229		
\notin	605		
\null	2634		
O			
\o@join			
2301, 2304, 2305, 2306			
\Omega	1699		
\omega 2091, 2092, 2112, 2113			
\or	2212		
P			
\pagestyle	1252		
\par 342, 363, 525, 1181, 1189, 1559, 1621, 1644, 1652, 1974, 1977, 1979, 2254, 2276, 2291, 2332, 2611, 2635, 2771, 2777			
\paragraph	1229		
\parindent	2323		
\path 94, 141, 204, 563			
\pgfkeys	2005, 2663, 2664, 2665, 2666, 2667, 2670, 2673, 2675		
\pgfmath@count			
.	2376, 2378, 2380		
\pgfmath@smuggleone 2381			
\pgfmathdeclarefunction	2375		
\pgfmathhint	2376		
\pgfmathparse			
.	1311, 2393, 2398, 2401, 2415, 2416		
\pgfmathresult			
.	1312, 2376, 2377, 2379, 2381, 2394, 2402, 2415, 2416		
\pgfutil@empty	2377		
\pgfutil@loop	2378		
\pgfutil@repeat	2381		
\preceq	1364		
\prime	1343		
\printbibliography	1533		
\ProvidesPackage			
2, 20, 53, 221, 399, 460, 467, 531, 594, 609, 1064, 1100, 1217, 1257, 1266, 1271, 1300, 1338, 1410, 1493, 1508, 1514, 1536, 1682, 1842, 1855, 1934, 1990, 2056, 2081, 2086, 2130, 2296, 2309, 2338, 2419, 2426, 2431, 2511, 2526, 2617, 2621, 2641, 2683, 2720, 2740			
Q			
\QS@list			
2256, 2267, 2271, 2278, 2284, 2289, 2292			
\QS@select@equal	2231, 2235		
\QS@select@greater	2232, 2236		
\QS@select@smaller	2227, 2230, 2234		
\QS@sort@a	2209, 2242, 2263, 2264		
\QS@sort@b	2209, 2210		
\QS@sort@c	2213, 2220		
\QS@sort@d	2221, 2229		
\QS@sort@empty	2211, 2216		
\QS@sort@single 2212, 2217			
\QS@initialize	2134, 2246, 2281		
\QS@Ir	2217, 2223, 2231, 2251, 2265, 2270, 2273		
\QS@Irr 2252, 2265, 2266, 2274			
\QS@Lr	2223, 2230, 2241, 2242, 2250, 2263, 2268, 2272		
\QS@pivotStep	2136, 2246, 2250, 2261		
\QS@r	2223		
\QS@Rr	2232, 2253, 2264, 2269, 2275, 2284, 2285, 2286		
\QS@sortStep	2138, 2246, 2262, 2263		
\quad	1861		
R			
\raisebox	1482		
\relax	1545, 2223, 2266, 2268, 2269, 2270, 2378, 2380		
\renewcommand	1232, 1233, 1234, 1235, 1238, 1239, 1253, 1254, 1503, 1504, 1903		
\repeat	2139		
\RequirePackage	55, 158, 222, 226, 395, 401, 402, 462, 532, 611, 612, 1066, 1068, 1069, 1107, 1108, 1218, 1219, 1222, 1224, 1226, 1231, 1240, 1247, 1258, 1259, 1268, 1272, 1273, 1274, 1302, 1303, 1342, 1411, 1510, 1515, 1516, 1532, 1539, 1540, 1541, 1569, 1684, 1845, 1846, 1991, 1993, 2083, 2143, 2145, 2297, 2298, 2299, 2312, 2340, 2371, 2422,		
2428, 2433, 2444, 2445, 2513, 2618, 2625, 2644, 2645, 2647, 2648, 2686, 2687, 2688, 2721, 2741			
\right	1686		
\RIGHTarrow	2662, 2667		
\Rightarrow	600, 605		
\rightarrow	207, 489, 494, 502, 506, 508, 509, 511, 566, 574, 2053, 2556, 2563, 2565, 2568, 2573, 2578, 2579, 2584		
\rightrightarrows	2305		
\Roman	1237		
\roman	1237, 1239		
\romannumeral	2221		
\rtimes	1844		
\rule 2254, 2276, 2291, 2302			
S			
\sb	67, 77, 79, 108, 166, 501, 502, 506, 509, 510, 511, 1164, 1166, 1736, 1739, 1742, 1768, 1774, 1923, 2041, 2050, 2578, 2579, 2580, 2585, 2589, 2590, 2593, 2594, 2597, 2598, 2599		
\scriptscriptstyle	566, 574, 582		
\scriptsize	1183, 1422, 1429, 1435, 1497, 1498, 1501, 1502, 2691, 2744, 2774		
\section	46		
\seq	1486, 1487, 1488, 1630, 1631, 1632, 1639		
\setbox	2301		
\setcounter	1230, 2255, 2277, 2291		
\setganttlinklabel	1260, 1261, 1262, 1263		
\setlength	2323, 2768, 2769, 2779		
\setmainfont	1220		
\setmainlanguage	396		
\setminted	2450, 2451		
\setsansfont	1221		
\setul	2314		
\sffamily	476, 1227, 1229, 1331, 2534		
\shoveleft	1963		
\shoveright	1966		
\Sigma	64, 105, 163, 1131, 1132, 1194		
\sigma	499, 501, 502		
\SLASH	1551		
\small	1660		
\sort	2388		
\sortList	2387, 2396		

<code>\square</code>	464	336, 337, 348, 354,	<code>\umlnote</code> . . .	678, 894, 1037
<code>\stepcounter</code> 2175, 2180,		357, 360, 369, 383,	<code>\umlreal</code>	674, 735
2185, 2188, 2190,		536, 539, 544, 545,	<code>\umlsimpleclass</code> . .	633,
2194, 2196, 2200, 2202		553, 554, 557, 558,	634, 635, 639, 641,	
<code>\str</code> . .	477, 486, 1573,	1172, 1193, 1194,	642, 643, 669, 822,	
2065, 2074, 2535, 2548		1195, 1196, 1199,	823, 824, 883, 935, 936	
<code>\StrSubstitute</code> .	2395, 2397	1200, 1201, 1202, 1719	<code>\umlstatic</code>	829, 858
<code>\strut</code>	1790, 1794,	<code>\TmpPlaceEight</code>	<code>\umluniagggreg</code>	890
1798, 1802, 1806, 2423		<code>\TmpPlaceFive</code>	<code>\umluniassoc</code>	653,
<code>\subseteq</code> 2559, 2592, 2599		<code>\TmpPlaceFour</code>	675, 891, 1015, 1016	
		<code>\TmpPlaceNine</code>	<code>\umlVHuniassoc</code> . .	654, 655
T		<code>\TmpPlaceOne</code>	<code>\umlVHVdep</code>	647,
<code>\tableofcontents</code> . . .	1546	<code>\TmpPlaceSeven</code>	648, 650, 651, 831, 832	
<code>\text</code>	77, 79,	<code>\TmpPlaceSix</code>	<code>\umlVHVinherit</code>	
182, 1945, 2691, 2744		<code>\TmpPlaceTen</code> 630, 631, 636,	
<code>\textbf</code> . . .	1071, 1369,	<code>\TmpPlaceThree</code>	637, 644, 645, 800,	
1378, 1389, 1398,		<code>\TmpPlaceTwo</code>	801, 825, 826, 995, 996	
1555, 1562, 1593,		<code>\TmpScale</code>	<code>\umlVHVreal</code>	
1621, 1636, 1652, 1908		<code>\TmpTransitionEight</code> .	. 794, 795, 1032, 1033	
<code>\textcolor</code>	1481, 2690 2002, 2023	<code>\UParrow</code>	2664
<code>\textit</code>		<code>\TmpTransitionFive</code> . .	<code>\url</code>	1671
940, 982, 983, 984,	 1999, 2020	<code>\usemintedstyle</code>	2447
985, 1662, 1953, 1986		<code>\TmpTransitionFour</code> . .	<code>\usetikzlibrary</code> . .	56,
<code>\textsc</code>	1344 1998, 2019	403, 1067, 1275,	
<code>\textsf</code>	1555, 1636	<code>\TmpTransitionNine</code> . .	1412, 1992, 2300,	
<code>\textstyle</code>	1754, 1785 2003, 2024	2341, 2372, 2649, 2737	
<code>\texttt</code> 1094, 1344, 1478,		<code>\TmpTransitionOne</code> . . .		
1479, 1480, 1481, 2744	 1995, 2016	V	
<code>\thepage</code>	1251, 1500	<code>\TmpTransitionSeven</code> .	<code>\value</code>	2137
<code>\theparagraph</code>	1229 2001, 2022	<code>\varepsilon</code>	478,
<code>\Theta</code>	1689	<code>\TmpTransitionSix</code> . . .	489, 490, 1117,	
<code>\thinspace</code>	2744 2000, 2021	1736, 1742, 1771, 1782	
<code>\tikz</code>	1482	<code>\TmpTransitionTen</code> . . .	<code>\vfill</code>	2423
<code>tikz: bbaum</code>	23 2004, 2025	<code>\vrule</code>	2630, 2634
<code>tikz: li binaer baum</code>	21	<code>\TmpTransitionThree</code> .	<code>\vspace</code>	
<code>\tikzchildnode</code>	419 1997, 2018	1152, 1156, 1607, 1625	
<code>\tikzparentnode</code>	419	<code>\TmpTransitionTwo</code> . . .		
<code>\tikzset</code>	96, 1996, 2017	X	
143, 209, 404, 430,		<code>\TmpX</code>	<code>\xappto</code> . .	2399, 2405, 2409
1276, 1413, 2031,		<code>\TmpY</code>	<code>\xdef</code>	1305
2157, 2361, 2515, 2722		<code>\today</code>	<code>\xintApply</code>	2225
<code>\tikzumlset</code>	2650	<code>\ttfamily</code>	<code>\xintApplyUnbraced</code> . .	
<code>\times</code>	207		2224, 2230, 2231, 2232	
<code>\tiny</code>	1077, 1083,	U	<code>\xintCSVtoList</code>	2284
1089, 1481, 1550, 2460		<code>\ul</code>	<code>\xintFor</code>	
<code>\titleformat</code> . .	1227, 1229	1072, 2313, 2314	2174, 2179, 2184,	
<code>\titlespacing</code>	1228	<code>\umlaggreg</code>	2189, 2195, 2201, 2242	
<code>\tl</code> .	29, 37, 63, 64, 65,	<code>\umlassoc</code>	<code>\xintifEq</code>	2235
66, 67, 68, 71, 72,		1017	<code>\xintifForLast</code>	
73, 74, 75, 77, 79,		<code>\umlclass</code> 2191, 2197, 2203	
104, 105, 106, 107,		618,	<code>\xintifGt</code>	2236
108, 109, 110, 113,		622, 626, 670, 671,	<code>\xintifLt</code>	2234
114, 115, 116, 117,		672, 719, 724, 729,	<code>\xintLength</code>	2209
118, 119, 162, 163,		732, 790, 791, 792,	<code>\xintntheft</code>	2221
164, 165, 166, 167,		797, 798, 827, 857,		
168, 171, 172, 173,		884, 885, 888, 939,	Z	
174, 175, 176, 177,		942, 981, 987, 988,	<code>\ZB</code>	1678
285, 289, 307, 311,		1006, 1007, 1008,	<code>\zB</code>	1677
312, 313, 316, 321,		1027, 1028, 1029, 1030	<code>\zustandsnamens@liste</code>	
322, 323, 334, 335,		<code>\umldep</code> 1135, 1142, 1143	
		947		
		<code>\umlHVHagggreg</code> 737, 803, 997		
		<code>\umlinherit</code>		
		676, 727, 892, 937, 945		