

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)	33
2.10.6	Dekorierer (Decorator)	35
2.10.7	Einfache Fabrik (Simple Factory)	36
2.10.8	Einzelstück (Singleton)	36
2.10.9	Erbauer (Builder)	37
2.10.10	Fabrikmethode (Factory Method)	39
2.10.11	Kompositum (Composite)	40
2.10.12	Modell-Präsentation-Steuerung (Model-View-Controller)	41
2.10.13	Zustand (State)	41
2.11	er.sty	44
2.12	formale-sprachen.sty	46
2.13	formatierung.sty	50
2.13.1	Schriftarten / Typographie	50
2.13.2	Farben	50
2.13.3	Überschriften	50

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

2.13.4	Listen	50
2.13.5	Kasten	50
2.13.6	Header	50
2.14	gantt.sty	51
2.15	grafik.sty	52
2.16	graph.sty	53
2.17	hanoi.sty	55
2.18	komplexitaetstheorie.sty	56
2.19	kontrollflussgraph.sty	58
2.20	kopf-fusszeilen.sty	60
2.21	literatur-dummy.sty	61
2.22	literatur.sty	62
2.23	makros.sty	63
2.24	master-theorem.sty	67
2.25	mathe.sty	71
2.26	minimierung.sty	72
2.27	normalformen.sty	75
2.28	petri.sty	77
2.29	potenzmengen-konstruktion.sty	79
2.30	pseudo.sty	81
2.31	pumping-lemma.sty	82
2.32	quicksort.sty	83
2.33	relationale-algebra.sty	86
2.34	rmodell.sty	87
2.35	sortieren.sty	88
2.36	spalten.sty	90
2.37	struktogramm.sty	91
2.38	syntax.sty	92
2.39	syntaxbaum.sty	94
2.40	synthese-algorithmus.sty	95
2.41	tabelle.sty	98
2.42	typographie.sty	99
2.43	uml.sty	100
2.44	vollstaendige-induktion.sty	102
2.45	wasserfall.sty	104
2.46	wpkalkuel.sty	105

3 Index

106

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]
{\zustaende=Z,alphabet= $\Sigma$ ,bandalphabet= $\Gamma$ ,delta= $\delta$ ,start= $z_0$ ,leerzeichen= $\square$ ,ende=E)}

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

```

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

```

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

```

\liTuringUebergangZelle Formatiert einen Zustandsübergang für eine Übergangstabelle.

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

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

```

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

```

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

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

`(z1: \square , L)`

`(\square : \square , R)`

```

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

\liTuringKante \liTuringKante[\tikz-optionen]{\zustand-oder-lese}{\schreibe}{\richtung}
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-qtree.]

401 \RequirePackage{tikz}

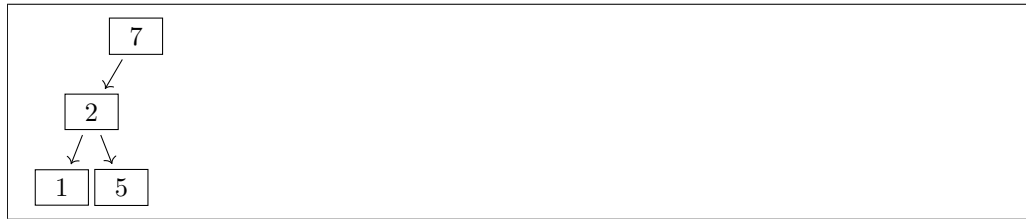
    für li binaer baum
402 \RequirePackage{tikz-qtree}

    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}

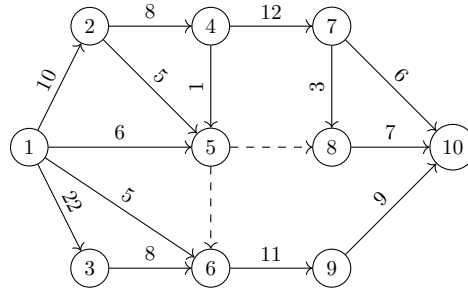
```

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}{5}
\liCpmVorgang[schein]{5}{8}{7}
\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:

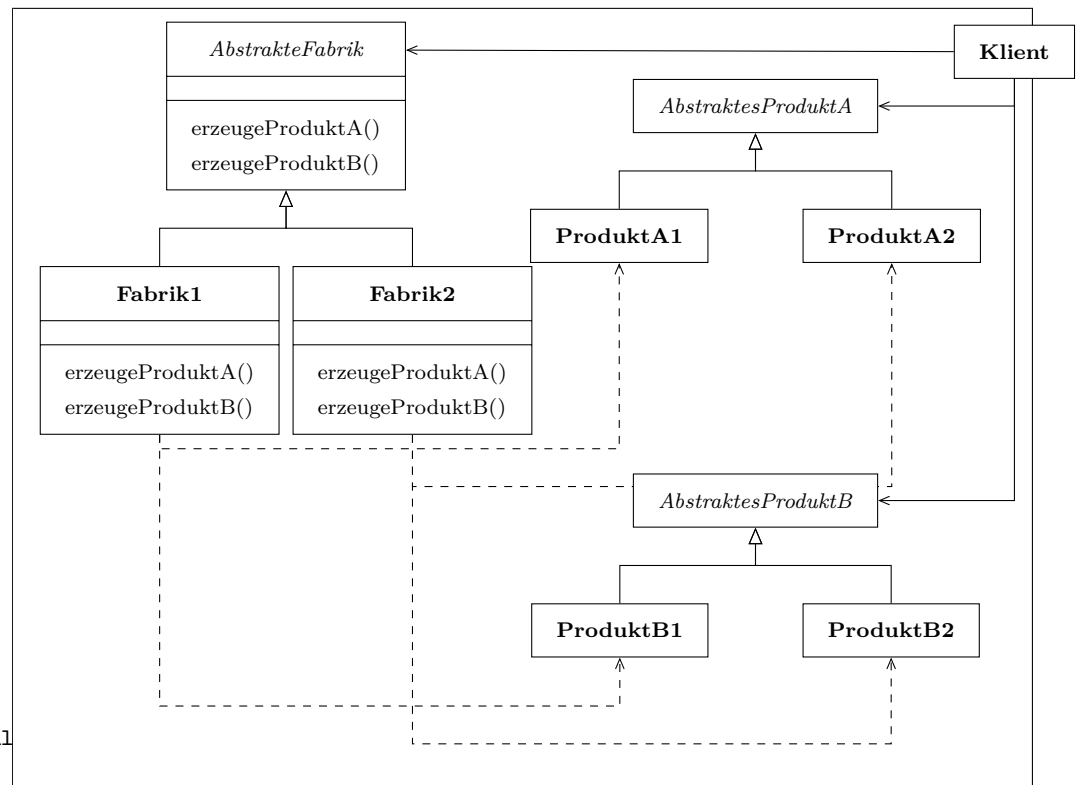
Präfix: \liEntwurfs + Name des Entwurfsmuster DeutscherName + Suffix: (ohne, Uml oder Akteure)

2.10.2 Reihenfolge

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

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

2.10.3 Abstrakte Fabrik (Abstract Factory)



```
612 \def\liEntwurfsAbstrakteFabrikUml{
613   \begin{tikzpicture}
614     \umlclass[type=abstract]{AbstrakteFabrik}{-}{-}{
615       erzeugeProduktA()\n
616       erzeugeProduktB()\n
617     }
618     \umlclass[below left=1cm and -1.5cm of AbstrakteFabrik]{Fabrik1}{-}{-}{
619       erzeugeProduktA()\n
620       erzeugeProduktB()\n
621     }
622     \umlclass[below right=1cm and -1.5cm of AbstrakteFabrik]{Fabrik2}{-}{-}{
623       erzeugeProduktA()\n
624       erzeugeProduktB()\n
625     }
```

```

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

```

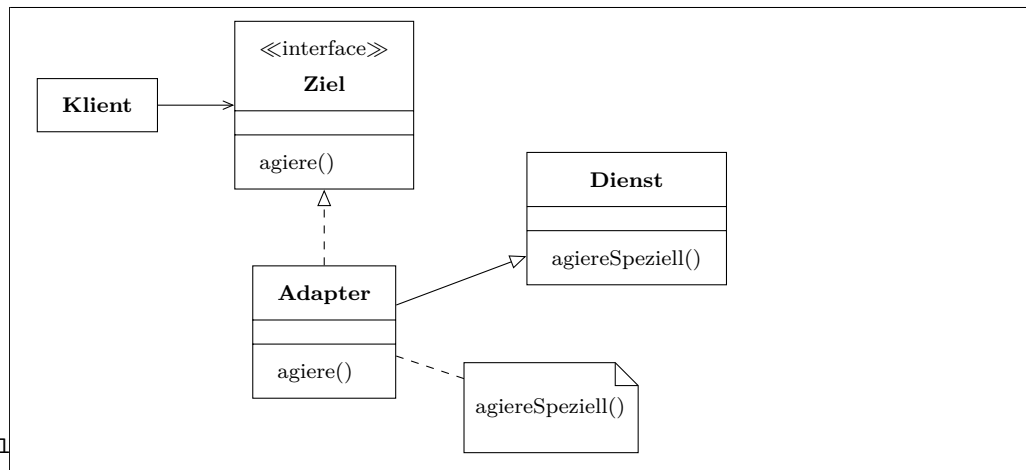
\liEntwurfsAbstrakteFabrik

```

654 \def\liEntwurfsAbstrakteFabrik{
655 \liEntwurfsAbstrakteFabrikUml
656 }

```

2.10.4 Adapter



\liEntwurfsAdapterUml

```

657 \def\liEntwurfsAdapterUml{
658 \begin{tikzpicture}
659 \umlsimpleclass[x=1,y=3]{Klient}{-}{-}
660 \umlclass[x=4,y=3,type=interface]{Ziel}{-}{agiere()}
661 \umlclass[x=4,y=0]{Adapter}{-}{agiere()}
662 \umlclass[x=8,y=1.5]{Dienst}{-}{agiereSpeziell()}
663
664 \umlreal{Adapter}{Ziel}
665 \umluniassoc{Klient}{Ziel}
666 \umlinherit{Adapter}{Dienst}
667

```



```

668     \umlnote[x=7,y=-1,width=2cm]{Adapter}{agiereSpeziell()}
669 \end{tikzpicture}
670 \footcite[so ähnlich wie GoF]{wiki:adapter}
671 }

```

\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.

```

672 \def\liEntwurfsAdapterAkteure{
673   \begin{description}
674
675     \item[Ziel (Target)]
676
677     Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.
678
679     \item[Klient (Client)]
680
681     Der Klient nutzt Dienste über inkompatible Schnittstellen und greift
682     dabei auf adaptierte Schnittstellen zurück.
683
684     \item[Dienst (Adaptee)]
685
686     Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest
687     definierter Schnittstelle an.
688
689     \item[Adapter]
690
691     Der Adapter adaptiert die Schnittstelle des Dienstes auf die
692     Schnittstelle zum Klienten.\footcite{wiki:adapter}
693
694   \end{description}
695 }

```

\liEntwurfsAdapter

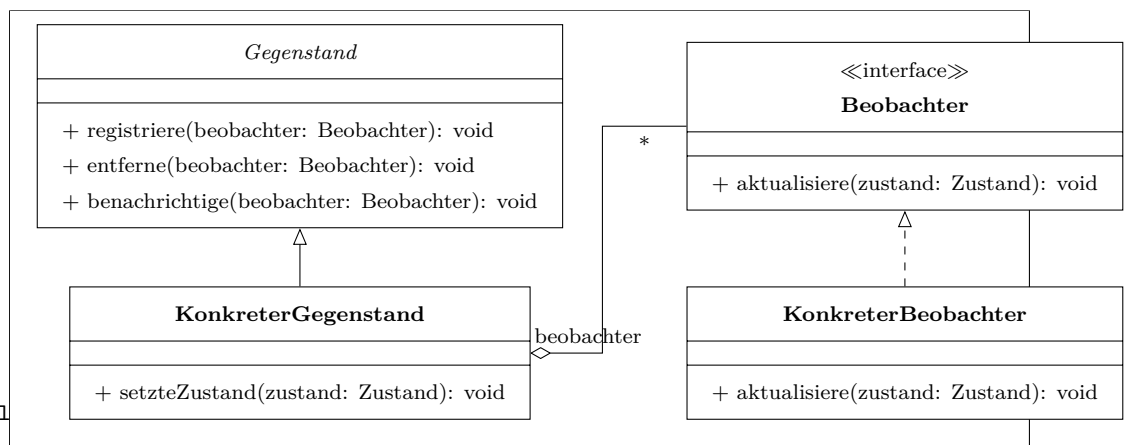
```

696 \def\liEntwurfsAdapter{
697   \liEntwurfsAdapterUml
698   \liEntwurfsAdapterAkteure
699 }

```

2.10.5 Beobachter (Observer)

\liEntwurfsBeobachterUml



```

700 \def\liEntwurfsBeobachterUml{
701   \begin{tikzpicture}
702     \umlclass[x=0,y=0,type=abstract]{Gegenstand}{-}{
703       + registriere(beobachter: Beobachter): void\\
704       + entferne(beobachter: Beobachter): void\\
705       + benachrichtige(beobachter: Beobachter): void\\
706     }
707     \umlclass[x=0,y=-3]{KonkreterGegenstand}{-}{
708       + setzZustand(zustand: Zustand): void
709     }
710     \umlinherit{KonkreterGegenstand}{Gegenstand}
711
712     \umlclass[x=8,y=0,type=interface]{Beobachter}{-}{
713       + aktualisiere(zustand: Zustand): void
714     }
715     \umlclass[x=8,y=-3]{KonkreterBeobachter}{-}{
716       + aktualisiere(zustand: Zustand): void
717     }
718     \umlreal{KonkreterBeobachter}{Beobachter}
719
720     \umlHVHaggreg[arg1=beobachter,pos1=0.8,mult2=*,pos2=2.5]
721     {KonkreterGegenstand}{Beobachter}
722   \end{tikzpicture}
723 }

```

\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.

```

724 \def\liEntwurfsBeobachterAkteure{
725   \begin{description}
726     \item[Gegenstand / Subjekt (Subject / Observable)]
727
728     Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also
729     „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne
730     deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An-
731     und Abmeldung von Beobachtern und eine Schnittstelle zur
732     Benachrichtigung von Beobachtern über Änderungen an.\footcite[Seite
733     251]{gof}
734
735     \item[Beobachter (Observer)]
736
737     Die Beobachter (auf Englisch auch subscriber, also „Abonnent“,
738     genannt) definieren eine Aktualisierungsschnittstelle.
739

```

```

740 \item[konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)]
741
742 Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert
743 den relevanten Zustand und benachrichtigt alle Beobachter bei
744 Zustandsänderungen über deren Aktualisierungsschnittstelle. Es
745 verfügt über eine Schnittstelle zur Erfragung des aktuellen
746 Zustands.
747
748 \item[Konkrete Beobachter (ConcreteObserver)]
749
750 Konkrete Beobachter verwalten die Referenz auf ein konkretes
751 Subjekt, dessen Zustand sie beobachten und speichern und dessen
752 Zustand konsistent ist. Sie implementieren eine
753 Aktualisierungsschnittstelle unter Verwendung der
754 Abfrageschnittstelle des konkreten Subjekts.
755 \footcite{wiki:beobachter}
756 \end{description}
757 }

```

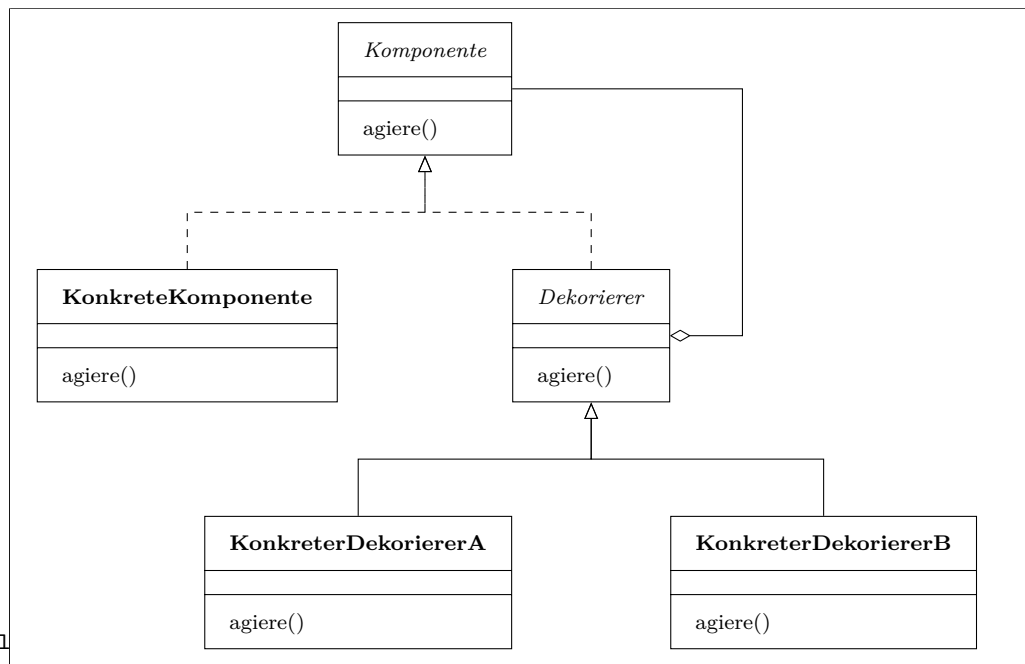
\liEntwurfsBeobachter

```

758 \def\liEntwurfsBeobachter{
759 \liEntwurfsBeobachterUml
760 \liEntwurfsBeobachterAkteure
761 }

```

2.10.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

762 \def\liEntwurfsDekoriererUml{
763 \begin{tikzpicture}
764 \umlclass[type=abstract]{Komponente}{-}{agiere()}
765 \umlclass[below left=1.5cm and 0cm of Komponente]{KonkreteKomponente}{-}{agiere()}
766 \umlclass[below right=1.5cm and 0cm of Komponente,type=abstract]{Dekorierer}{-}{agiere()}
767
768 \umlVHVreal{KonkreteKomponente}{Komponente}
769 \umlVHVreal{Dekorierer}{Komponente}
770
771 \umlclass[below left=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererA}{-}{agiere()}
772 \umlclass[below right=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererB}{-}{agiere()}
773
774 \umlVHVinherit{KonkreterDekoriererA}{Dekorierer}

```

```

775     \umlVHVinherit{KonkreterDekoriererB}{Dekorierer}
776
777     \umlHVVHagggreg[arm1=2cm]{Dekorierer}{Komponente}
778     \footcite{wiki:dekorierer}
779 \end{tikzpicture}
780 }

```

\liEntwurfsDekorierer

```

781 \def\liEntwurfsDekorierer{
782   \liEntwurfsDekoriererUml
783   \liEntwurfsDekoriererAkteure
784 }

```

2.10.7 Einfache Fabrik (Simple Factory)

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

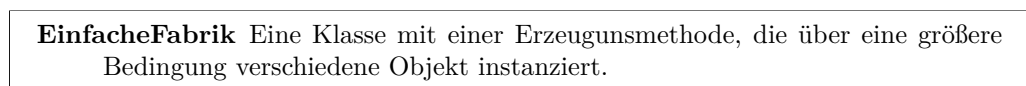


```

785 \def\liEntwurfsEinfacheFabrikUml{
786   \begin{tikzpicture}
787     \umlclass{EinfacheFabrik}{
788       {
789         \umlstatic{erzeuge(): Produkt}\\
790       }
791     \end{tikzpicture}
792 }

```

ntwurfsEinfacheFabrikAkteure Quelle: frei überstetzt aus GoF



```

793 \def\liEntwurfsEinfacheFabrikAkteure{
794   \begin{description}
795     \item[EinfacheFabrik]
796
797     Eine Klasse mit einer Erzeugungsmethode, die über eine größere
798     Bedingung verschiedene Objekt instanziiert.
799   \end{description}
800 }

```

\liEntwurfsEinfacheFabrik

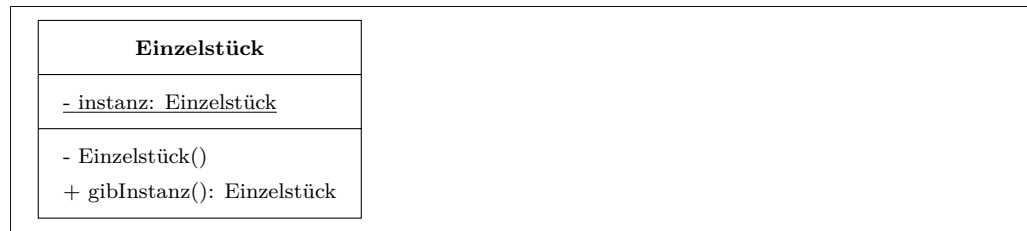
```

801 \def\liEntwurfsEinfacheFabrik{
802   \liEntwurfsEinfacheFabrikUml
803   \liEntwurfsEinfacheFabrikAkteure
804 }

```

2.10.8 Einzelstück (Singleton)

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```

805 \def\liEntwurfsEinzelstueckUml{
806   \begin{tikzpicture}
807     \umlclass{Einzelstück}{
808       \umlstatic{- instanz: Einzelstück}\\
809     }{
810       - Einzelstück()\\
811       + gibInstanz(): Einzelstück
812     }
813   \end{tikzpicture}
814 }

```

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.

```

815 \def\liEntwurfsEinzelstueckAkteure{
816   \begin{description}
817     \item[Einzelstück (Singleton)]
818
819     stellt eine statische Methode bereit, mit deren Hilfe die Klienten
820     nur auf eine einzige Instanz der Klasse zugreifen können.
821   \end{description}
822 }

```

\liEntwurfsEinzelstueck

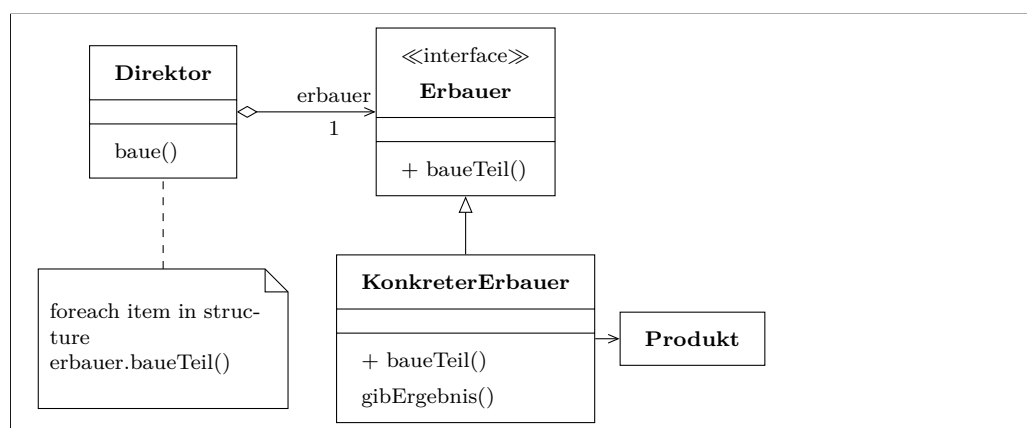
```

823 \def\liEntwurfsEinzelstueck{
824   \liEntwurfsEinzelstueckUml
825   \liEntwurfsEinzelstueckAkteure
826 }

```

2.10.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

827 \def\liEntwurfsErbauerUml{
828   \begin{tikzpicture}
829     \umlsimpleclass[x=7,y=0]{Produkt}{}{}
830     \umlclass[x=4,y=3,type=interface]{Erbauer}{}{+ baueTeil()}
831     \umlclass[x=4,y=0]{KonkreterErbauer}{}{

```

```

832     + baueTeil()\
833     gibErgebnis()
834 \umlclass[x=0,y=3]{Direktor}{-}{baue()}
835
836 \umluniagg[arg2=erbauer,pos2=0.7,mult2=1]{Direktor}{Erbauer}
837 \umluniassoc{KonkreterErbauer}{Produkt}
838 \umlinherit{KonkreterErbauer}{Erbauer}
839
840 \umlnote[x=0,y=0,width=3cm]{Direktor}{
841   foreach item in structure\
842     erbauer.baueTeil()
843 }
844 \end{tikzpicture}
845 \footcite{wiki:erbauer}
846 }

```

\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.

```

847 \def\liEntwurfsErbauerAkteure{
848   \begin{description}
849     \item[Erbauer]
850
851     Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der
852     Teile eines komplexen Objektes.
853
854     \item[KonkreterErbauer]
855
856     Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch
857     Implementierung der Schnittstelle. Außerdem definiert und verwaltet er
858     die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine
859     Schnittstelle zum Auslesen des Produkts.
860
861     \item[Direktor]
862
863     Der Direktor konstruiert ein komplexes Objekt unter Verwendung der
864     Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer
865     zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder
866     benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom
867     Klienten.
868
869     \item[Produkt]
870
871     Das Produkt repräsentiert das zu konstruierende komplexe Objekt.
872     \footcite{wiki:erbauer}
873   \end{description}
874 }

```

\liEntwurfsErbauer

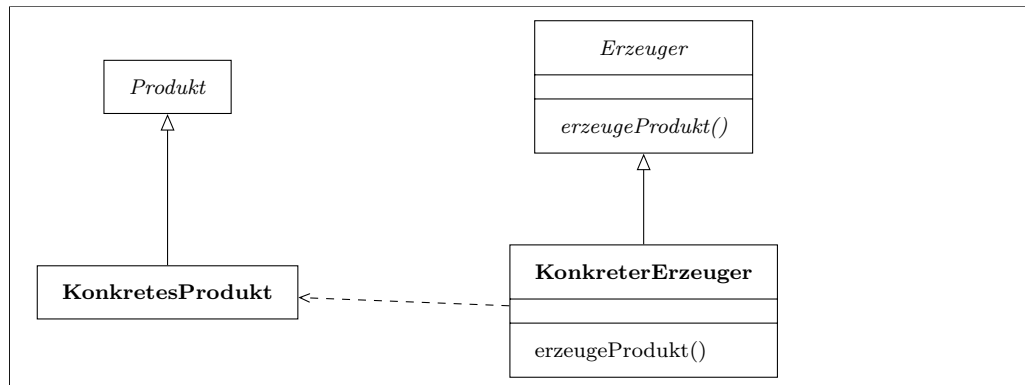
```

875 \def\liEntwurfsErbauer{
876   \liEntwurfsErbauerUml
877   \liEntwurfsErbauerAkteure
878 }

```

2.10.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

879 \def\liEntwurfsFabrikmethodeUml{
880   \begin{tikzpicture}
881     \umlsimpleclass[type=abstract]{Produkt}
882     \umlsimpleclass[below=2cm of Produkt]{KonkretesProdukt}
883     \umlinherit{KonkretesProdukt}{Produkt}
884
885     \umlclass[type=abstract,right=4cm of Produkt]{Erzeuger}{\{
886       \textit{erzeugeProdukt()}\}
887   }
888     \umlclass[below=1.2cm of Erzeuger]{KonkreterErzeuger}{\{
889       erzeugeProdukt()
890     }
891     \umlinherit{KonkreterErzeuger}{Erzeuger}
892
893     \umldep{KonkreterErzeuger}{KonkretesProdukt}
894   \end{tikzpicture}
895 }

```

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).

```

896 \def\liEntwurfsFabrikmethodeAkteure{
897   \begin{description}
898     \item[Produkt]
899
900     Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das
901     zu erzeugende Produkt.
902
903     \item[KonkretesProdukt]
904

```

```

905     KonkretesProdukt implementiert die Produkt-Schnittstelle.
906
907     \item[Erzeuger]
908
909     Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt
910     zu erzeugen und kann eine Default-Implementierung beinhalten.
911
912     \item[KonkreterErzeuger]
913
914     KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm
915     entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den
916     Konstruktor einer konkreten Produkt-Klasse aufruft).
917
918     \footcite{wiki:fabrikmethode}
919 \end{description}
920 }

```

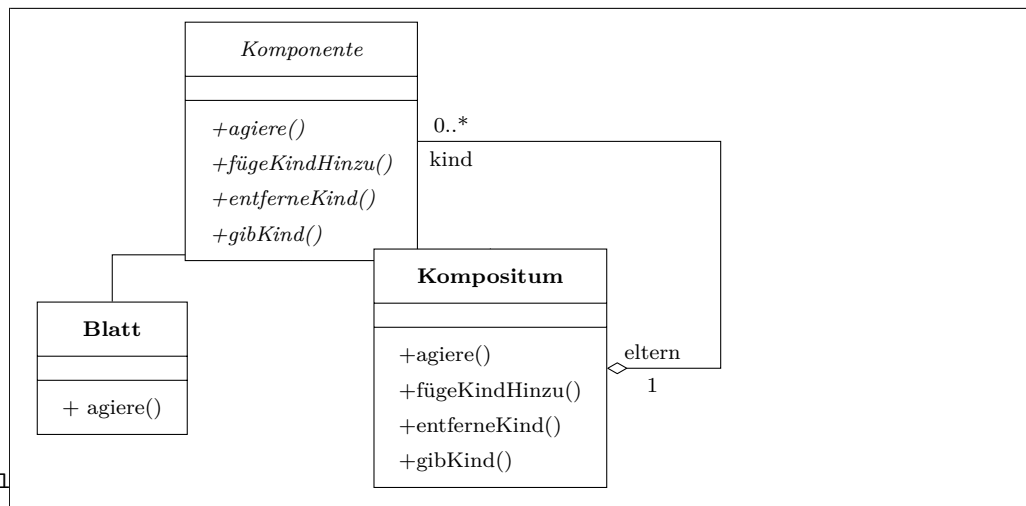
\liEntwurfsFabrikmethode

```

921 \def\liEntwurfsFabrikmethode{
922   \liEntwurfsFabrikmethodeUml
923   \liEntwurfsFabrikmethodeAkteure
924 }

```

2.10.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

925 \def\liEntwurfsKompositumUml{
926   \begin{tikzpicture}
927     \umlclass[x=2.5,y=3,type=abstract]{Komponente}{{}{
928       \textit{+agiere()}}\
929       \textit{+fügeKindHinzu()}}\
930       \textit{+entferneKind()}}\
931       \textit{+gibKind()}}
932   }
933   \umlclass[x=0]{Blatt}{{}{+ agiere()}}
934   \umlclass[x=5]{Kompositum}{{}{
935     +agiere()\
936     +fügeKindHinzu()\
937     +entferneKind()\
938     +gibKind()
939   }}
940
941   \umlVHVinherit{Kompositum}-{Komponente}
942   \umlVHVinherit{Blatt}-{Komponente}
943   \umlHVVaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,
944   \end{tikzpicture}
945 }

```

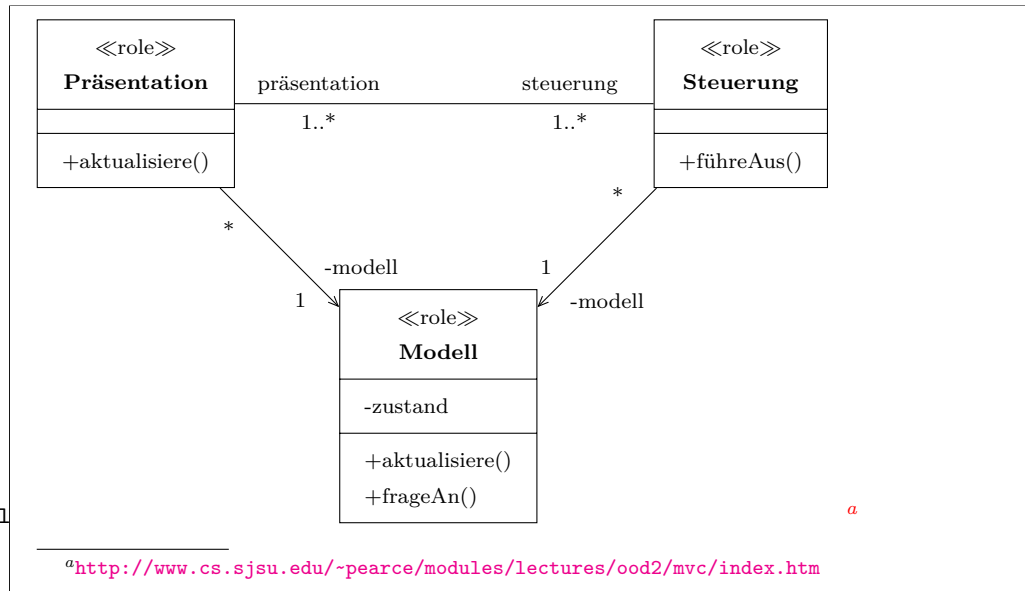

\liEntwurfsFabrikmethode

```

946 \def\liEntwurfsKompositum{
947   \liEntwurfsKompositumUml
948   \liEntwurfsKompositumAkteure
949 }

```

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



```

950 \def\liEntwurfsModellPraesentationSteuerungUml{
951   \begin{tikzpicture}
952     \umlclass[x=-4cm,type=role]{Präsentation}{+aktualisiere()}
953     \umlclass[x=4cm,type=role]{Steuerung}{+führeAus()}
954     \umlclass[y=-4cm,type=role]{Modell}{
955       -zustand
956     }{
957       +aktualisiere()\
958       +frageAn()
959     }
960
961     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
962     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}
963     \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
964   \end{tikzpicture}
965   \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
966 }

```

ModellPraesentationSteuerung

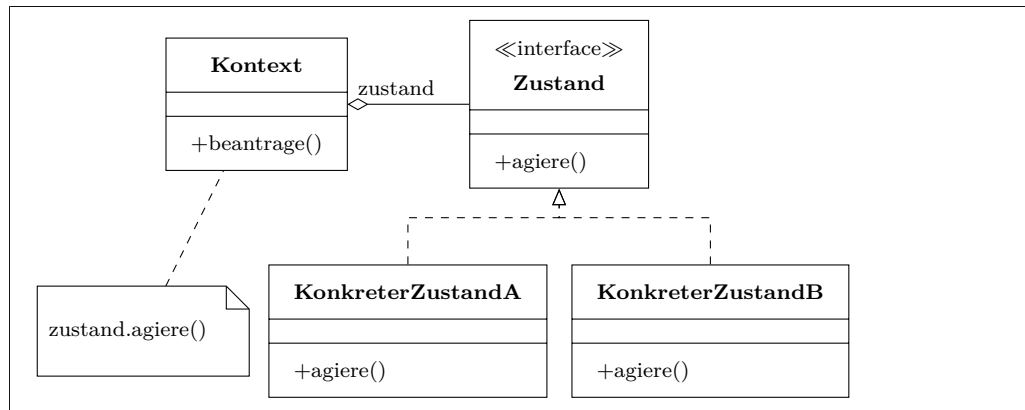
```

967 \def\liEntwurfs{
968   \liEntwurfsModellPraesentationSteuerungUml
969   \liEntwurfsModellPraesentationSteuerungAkteure
970 }

```

2.10.13 Zustand (State)

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



```

971 \def\liEntwurfsZustandUml{
972   \begin{tikzpicture}
973     \umlclass[x=-1,y=3]{Kontext}{+beantrage()}
974     \umlclass[x=3,y=3,type=interface]{Zustand}{+agiere()}
975     \umlclass[x=1,y=0]{KonkreterZustandA}{+agiere()}
976     \umlclass[x=5,y=0]{KonkreterZustandB}{+agiere()}
977
978     \umlVHVreal{KonkreterZustandA}{Zustand}
979     \umlVHVreal{KonkreterZustandB}{Zustand}
980
981     \umlaggreg[arg=zustand,pos=0.4]{Kontext}{Zustand}
982
983     \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
984   \end{tikzpicture}
985 }

```

\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.

```

986 \def\liEntwurfsZustandAkteure{
987   \begin{description}
988     \item[Kontext (Context)]
989
990     definiert die clientseitige Schnittstelle und verwaltet die separaten
991     Zustandsklassen.
992
993     \item[State (Zustand)]
994
995     definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
996     implementiert gegebenenfalls ein Standardverhalten.
997
998     \item[KontreterZustand (ConcreteState)]
999
1000     implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
1001     verbunden ist.
1002   \end{description}
1003 }

```

\liEntwurfsZustand

```

1004 \def\liEntwurfsZustand{
1005   \liEntwurfsZustandUml

```

```
1006 \liEntwurfsZustandAkteure
1007 }
1008
```

2.11 er.sty

```
1009 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1010 \ProvidesPackage{lehramt-informatik-er}[2020/06/13 Zum Setzen von
1011 ER-Diagrammen]

1012 \RequirePackage{tikz-er2}
1013 \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}

1014 \RequirePackage{soul}
1015 \RequirePackage{fontawesome}

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

1016 \ExplSyntaxOn

\liErEntity
1017 \def\liErEntity#1{\textbf{#1}}

\liErRelationship
1018 \def\liErRelationship#1{\ul{#1}}

\liErAttribute
1019 \def\liErAttribute#1{\emph{#1}}

\liErMpEntity mp = marginpar
Let-Abkürzung: \let\ e=\liErMpEntity
1020 \def\liErMpEntity#1{
1021   \liErEntity{#1}
1022   \marginpar{
1023     \liErEntity{\tiny\faSquareO{}}~E:~#1}
1024   }
1025 }
□

\liErMpRelationship Let-Abkürzung: \let\ r=\liErMpRelationship
1026 \def\liErMpRelationship#1{
1027   \liErRelationship{#1}
1028   \marginpar{
1029     \liErRelationship{\tiny\faGg{}}~R:~#1}
1030   }
1031 }

\liErMpAttribute Let-Abkürzung: \let\ a=\liErMpAttribute
1032 \def\liErMpAttribute#1{
1033   \liErAttribute{#1}
1034   \marginpar{
1035     \liErAttribute{\tiny\faCircleThin{}}~A:~#1}
1036   }
1037 }

\liErDatenbankName Let-Abkürzung: \let\ d=\liErDatenbankName
datenbank name
1038 \def\liErDatenbankName#1{
1039   {
1040     \footnotesize\texttt{(#1)}
1041   }
1042 }

1043 \ExplSyntaxOff
1044

```

2.12 formale-sprachen.sty

```

1045 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1046 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
1047 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]
1048
1049 \directlua{
1050   formale_sprachen = require('lehramt-informatik-formale-sprachen')
1051 }
1052
1053 \RequirePackage{hyperref}
1054 \RequirePackage{lehramt-informatik-mathe}

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

1055 \def\liMengeOhneMathe#1{\{ #1 \}}
1056 \def\liMenge#1{%
1057   \ifmmode%
1058     \liMengeOhneMathe{#1}%
1059   \else%
1060     $\liMengeOhneMathe{#1}$%
1061   \fi%
1062 }

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

1063 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

1064 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1065 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1066 \def\liPotenzmenge#1{$\liPotenzmengeOhneMathe{#1}$}

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

1067 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
1068 \let\liZustandsmenge=\liPotenzmenge

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

1069 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1070 \def\liUeberfuehrungsFunktion#1{
1071   \ifmmode
1072     \liUeberfuehrungsFunktionOhneMathe{#1}
1073   \else
1074     $\liUeberfuehrungsFunktionOhneMathe{#1}$
1075   \fi
1076 }

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

1077 \def\liAlphabet#1{$\Sigma = \{ #1 \}$}

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

1078 \def\liBandAlphabet#1{$\Gamma = \Sigma \cup \{ #1 \}$}

\liZustandsBuchstabe

1079 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1080 \def\liZustandsBuchstabeGross{Z}

```

```

\liZustandsmengeNr
1081 \def\zustandsnamens@liste#1#2{
1082   $
1083   \{
1084     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1085   \}
1086   $
1087 }
1088 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

\liZustandsmengeNrGross
1089 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

\liZustandsname \liZustandsname{1}: $z_1$
1090 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

\liZustandsnameGross \liZustandsnameGross{1}: $Z_1$
1091 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

\liAbleitung \liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1092 \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}

1093 \NewDocumentEnvironment { liProduktionsRegeln }
1094 { 0{P} +b }
1095 {
1096   \noindent
1097   $#1 = \{ $
1098   \vspace{-0.2cm}
1099   \begin{align*}
1100     \directlua{formale_sprachen.produktions_regeln('#2')}
1101   \end{align*}
1102   \vspace{-1.5cm}
1103   \begin{flushright}\}\end{flushright}
1104 } {}

\liProduktionen \liProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
1105 \def\liProduktionen#1{
1106   \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
1107 }

\liZustandsnameTiefgestellt Automatisch tiefgestellte Nummerierung \z1
Let-Abkürzung: \let\z=\liZustandsnameTiefgestellt
1108 \def\liZustandsnameTiefgestellt#1{
1109   \ifmmode
1110     \liZustandsBuchstabe\sb{#1}
1111   \else
1112     $\liZustandsBuchstabe\sb{#1}$
1113   \fi
1114 }

1115 \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}

1116 \NewDocumentCommand{ \liAusdruck } { 0{L} m m } {
1117   $
1118   \tl_if_empty:nTF {#1} {} {#1 =}
1119   \{
1120     \, #2 \,
1121     |
1122     \, #3 \,
1123   \}$
1124 }
1125 \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

1126 \def\liFlaci#1{%
1127   \par
1128   {%
1129     \scriptsize
1130     Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1131     Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1132     Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1133     \href{https://flaci.com/#1}{flaci.com/#1}%
1134   }%
1135   \par
1136 }

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

1137 \ExplSyntaxOn
1138 \NewDocumentCommand {\liGrammatik} { 0{G} m } {
1139   \tl_set:Nn \l_variablen_tl {V}
1140   \tl_set:Nn \l_alphabet_tl {\Sigma}
1141   \tl_set:Nn \l_produktionen_tl {P}
1142   \tl_set:Nn \l_start_tl {S}
1143
1144   \keys_define:nn { grammatik } {
1145     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1146     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1147     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1148     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1149   }
1150
1151   \keys_set:nn { grammatik } { #2 }
1152
1153   $#1 = (
1154     \l_variablen_tl,
1155     \l_alphabet_tl,
1156     \l_produktionen_tl,
1157     \l_start_tl
1158   )$

```



```
1159 }  
1160 \ExplSyntaxOff  
  
1161
```

2.13 formatierung.sty

```
1162 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1163 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

2.13.1 Schriftarten / Typographie

```
1164 \RequirePackage{mathpazo}
1165 \RequirePackage[no-math]{fontspec}
1166 \setmainfont{texgyrepagella}
1167 \setsansfont{QTAncientOlive}
1168 \RequirePackage{sectsty}
1169 \allsectionsfont{\fontspec{QTAncientOlive}}
```

2.13.2 Farben

```
1170 \RequirePackage{xcolor}
1171 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

2.13.3 Überschriften

```
1172 \RequirePackage{titlesec}
1173 \titleformat{\chapter}[display]{\sffamily\bfseries}{\Opt}{\LARGE}
1174 \titlespacing{\chapter}{0pt}{0pt}{*1}
1175 \titleformat{\paragraph}[hang]{\sffamily\normalsize\bfseries}{\theparagraph}{1em}{}
1176 \setcounter{secnumdepth}{0}
```

2.13.4 Listen

```
1177 \RequirePackage{paralist}
1178 \renewcommand\labelitemi{-}
1179 \renewcommand\labelitemii{-}
1180 \renewcommand\labelitemiii{-}
1181 \renewcommand\labelitemiv{-}
1182 % Counter: enumi enumii enumiii enumiv
1183 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1184 \renewcommand{\labelenumi}{(\alph{enumi})}
1185 \renewcommand{\labelenumii}{(\roman{enumii})}
```

2.13.5 Kasten

```
1186 \RequirePackage{mdframed}
1187 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1188 \NewDocumentEnvironment { liKasten }{ } {
1189   \begin{mdframed}[backgroundcolor=white!0]
1190 } {
1191   \end{mdframed}
1192 }
```

2.13.6 Header

```
1193 \RequirePackage{fancyhdr}
1194 \fancyhead[L,C,R]{}
1195 \fancyfoot[L]{}
1196 \fancyfoot[C]{}
1197 \fancyfoot[R]{\thepage}
1198 \pagestyle{fancy}
1199 \renewcommand{\headrulewidth}{0pt}
1200 \renewcommand{\footrulewidth}{0pt}
1201
```

2.14 gantt.sty

```

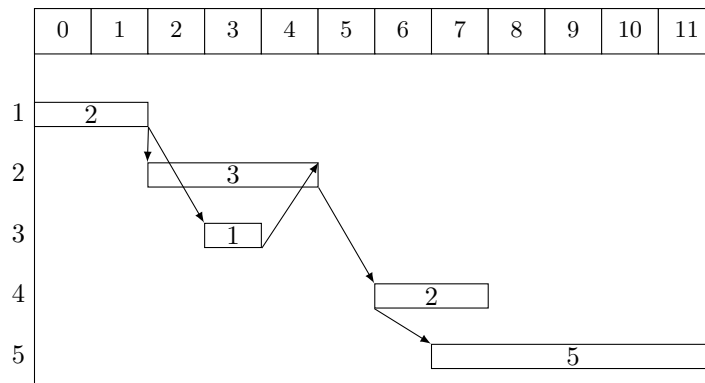
1202 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1203 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\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}

```



```

1204 \RequirePackage{tikz-uml}
1205 \RequirePackage{pgfgantt}
1206 \setganttlinklabel{f-s}{}
1207 \setganttlinklabel{s-s}{}
1208 \setganttlinklabel{f-f}{}
1209 \setganttlinklabel{s-f}{}

1210

```

2.15 grafik.sty

```
1211 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1212 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1213 TikZ, um Grafiken zeichnen zu können.]
1214 \RequirePackage{tikz}
1215
```

2.16 graph.sty

```

1216 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1217 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1218 \RequirePackage{tikz}

```

Für die die Adjazenzliste (xrightarrow)

```

1219 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

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

```

1220 \RequirePackage{blkarray}

```

```

1221 \usetikzlibrary{arrows.meta}

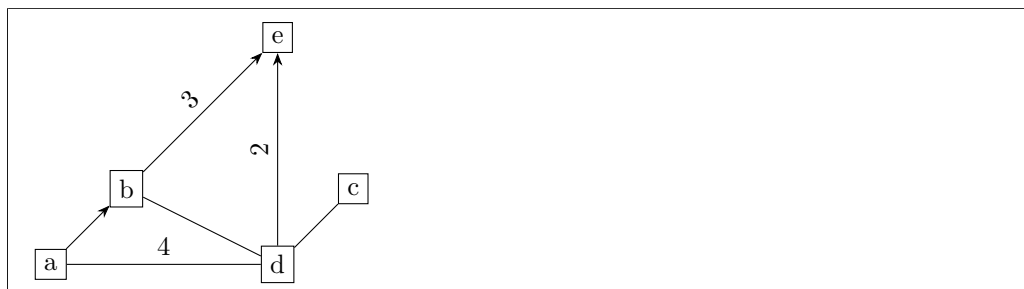
```

```

\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}

```



```

1222 \tikzset{
1223   li graph/.style={
1224     every node/.style={
1225       rectangle,
1226       draw,
1227     },
1228     every edge/.style={
1229       >={Stealth[black]},
1230       draw,
1231     },
1232     every edge/.append style={
1233       every node/.style={
1234         sloped,
1235         auto,
1236       }
1237     }
1238   },
1239   li markierung/.style={
1240     ultra thick,
1241   }
1242 }

```

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}

```

```

1243 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1244

```

2.17 hanoi.sty

```
1245 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1246 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
1247 von Hanoi-Grafiken]

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

1248 \RequirePackage{tikz}
1249 \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}

1250 \def\li@mset #1[#2]=#3{%
1251 \expandafter\xdef\csname #1#2\endcsname{#3}
1252 }
1253 \def\li@mget #1[#2]{%
1254 \csname #1#2\endcsname
1255 }
1256 \def\li@minc #1[#2] += #3{%
1257 \pgfmathparse{\li@mget #1[#2] + #3}%
1258 \li@mset #1[#2] = \pgfmathresult
1259 }
1260
1261 \def\liHanoi#1#2{
1262   \edef\li@numdiscs{#1}
1263   \def\li@sequence{#2}
1264   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1265     % init colors
1266     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purp
1267     \li@mset col[\j]={\c};
1268     % draw poles and init pole counters
1269     \foreach \j in {1,2,3}{
1270       \li@mset pos[\j]=0
1271       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1272     }
1273     % draw base
1274     \draw (.5,-.5) -- +(3,0);
1275     % draw discs
1276     \foreach[count=\k] \i/\j in \li@sequence{
1277       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*
1278       \li@minc pos[\j] += {.5}
1279     }
1280   \end{tikzpicture}
1281 }

1282
```

2.18 komplexitaetstheorie.sty

```

1283 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1284 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1285 Setzen von Karps NP-vollständigen Problemen, Hilfsmakros für die
1286 Polynomialzeitreduktion.]

```

Let-Abkürzungen

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

1287 \liLadePakete{mathe}

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

```
1288 \RequirePackage{mdframed}
```

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

```
1289 \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
```

1290 \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

```

1291 \def\liProblemBeschreibung#1#2#3{
1292   \begin{mdframed}[
1293     userdefinedwidth=9cm,
1294     align=center,
1295     backgroundcolor=white!0,
1296   ]
1297     \centerline{\large\liProblemName{#1}}
1298
1299     \medskip
1300
1301     \begin{description}
1302       \item[Gegeben:] #2
1303       \item[Frage:] #3
1304     \end{description}
1305   \end{mdframed}
1306 }

```



```

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

1307 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1308 \begin{displaymath}
1309 \liProblemName{#1}
1310 \preceq_{#2}
1311 \liProblemName{#3}
1312 \end{displaymath}
1313 }

\liProblemVertexCover

1314 \def\liProblemClique{%
1315 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1316 Mindestgröße  $n$  in einem gegebenen Graphen.
1317 \footcite[Seite 76]{theo:fs:4}
1318 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1319 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1320 \footcite{wiki:cliquenproblem}
1321 }

\liProblemVertexCover

1322 \def\liProblemVertexCover{%
1323 %
1324 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1325 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1326 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1327 \footcite{wiki:knotenueberdeckung}
1328
1329 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1330 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1331 mindestens einem Knoten aus  $U$  verbunden ist.
1332 \footcite[Seite 78]{theo:fs:4}%
1333 }

\liProblemSubsetSum Kein Karp-21-Problem

1334 \def\liProblemSubsetSum{%
1335 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1336 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1337 \footcite{wiki:teilsommenproblem}
1338 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1339 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1340 größer als eine gegebene obere Schranke  $c$  ist.
1341 \footcite[Seite 74]{theo:fs:4}
1342 }

\liProblemSubsetSum Kein Karp-21-Problem

1343 \def\liProblemSat{%
1344 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1345 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1346 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1347 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1348 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1349 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1350 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1351 aufgestellt werden.
1352 \footcite[Seite 71]{theo:fs:4}
1353 }

1354

```

2.19 kontrollflussgraph.sty

```
1355 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1356 \ProvidesPackage{lehramt-informatik-kontrollflussgraph}[2020/11/07]

1357 \RequirePackage{tikz}
1358 \usetikzlibrary{positioning}
1359 \tikzset{
1360   li kontrollfluss/.style={
1361     knoten/.style={
1362       circle,
1363       draw
1364     },
1365     usebox/.style={
1366       draw,
1367       rectangle,
1368       font=\scriptsize,
1369       anchor=west,
1370       align=left,
1371     },
1372     bedingung/.style={
1373       midway,
1374       draw=none,
1375       font=\scriptsize
1376     },
1377     knotenbeschriftung/.style={
1378       draw,
1379       rectangle,
1380       midway,
1381       font=\scriptsize
1382     },
1383     wahr/.style={
1384       thick
1385     },
1386     falsch/.style={
1387       dashed
1388     },
1389     every node/.style={
1390       circle,
1391       draw,
1392     },
1393     every edge/.append style={
1394       every node/.style={
1395         draw=none,
1396         bedingung,
1397       }
1398     },
1399     every path/.style={
1400       draw,
1401       ->,
1402     },
1403     every pin/.style={
1404       draw,
1405       dotted,
1406       rectangle,
1407       pin position=right
1408     },
1409     every pin edge/.style={
1410       dotted,
1411       arrows=-,
1412     }
1413   }
1414 }
```

liKontrollflussgraph

```

1415 \NewDocumentEnvironment { liKontrollflussgraph } { 0{ } } {
1416   \begin{tikzpicture}[
1417     li kontrollfluss,
1418     #1
1419   ]
1420 } {
1421   \end{tikzpicture}
1422 }

\liAnweisung
1423 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}

\liBedingung Let-Abkürzung: \let\b=\liBedingung
1424 \def\liBedingung#1#2{node[bedingung,#1]{\texttt{#2}}}}

\liBedingungWahr Let-Abkürzung: \let\bWahr=\liBedingungWahr
1425 \def\liBedingungWahr#1{node[bedingung,#1]{\texttt{[true]}}}}

\liBedingungFalsch Let-Abkürzung: \let\bFalsch=\liBedingungFalsch
1426 \def\liBedingungFalsch#1{node[bedingung,#1]{\texttt{[false]}}}}

\liKontrollCode Let-Abkürzung: \let\c=\liKontrollCode
1427 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}

\liKontrollTextzeileKnoten Let-Abkürzung: \let\k=\liKontrollTextzeileKnoten
1428 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw

\liKontrollKnotenPfad Let-Abkürzung: \let\p=\liKontrollKnotenPfad
1429 \ExplSyntaxOn
1430 \NewDocumentCommand { \liKontrollKnotenPfad }{ m }
1431 {
1432   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
1433   \seq_set_map:Nnn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
1434   \seq_use:Nn \l_tmpa_seq { ~~~~ }
1435 }
1436 \ExplSyntaxOff

1437

```

2.20 kopf-fusszeilen.sty

```
1438 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1439 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1440 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1441 \ExplSyntaxOn

1442 \fancyhead{}
1443 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1444 \fancyhead[L0,RE]{\scriptsize\today}
1445 \fancyfoot{}
1446 \fancyfoot[LE,R0]{\thepage}
1447 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1448 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1449 \renewcommand{\headrulewidth}{0.4pt}
1450 \renewcommand{\footrulewidth}{0.4pt}

1451 \ExplSyntaxOff

1452
```

2.21 literatur-dummy.sty

```
1453 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1454 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1455 \def\literatur{}

\footcite

1456 % \RequirePackage[stable,multiple]{footmisc}
1457 \NewDocumentCommand{ \footcite } { o m }{}

1458
```

2.22 literatur.sty

```
1459 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1460 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1461 \RequirePackage{csquotes}
1462 \RequirePackage[
1463   bibencoding=utf8,
1464   citestyle=authortitle,
1465   backend=biber,
1466 ]{biblatex}
1467 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1468 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1469 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1470 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1471 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1472 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1473 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1474 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1475 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1476 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1477 % To allow footnotes in the heading
1478 \RequirePackage[stable,multiple]{footmisc}

\literatur

1479 \def\literatur{\printbibliography[heading=subbibliography]}

1480
```

2.23 makros.sty

```
1481 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1482 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1483 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1484 anderen Paket passen]
1485 \RequirePackage{hyperref}
1486 \RequirePackage{graphicx}
    Für die Umgebung liQuellen benötigt.
1487 \RequirePackage{paralist}

\inhaltsverzeichnis
1488 \def\inhaltsverzeichnis {
1489   \begin{mdframed}
1490     \begin{group}
1491       \let\clearpage\relax
1492       \tableofcontents
1493     \end{group}
1494   \end{mdframed}
1495 }

\memph \mephm (\marginpar and \emph)
1496 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1497 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1498 \newcommand{\liPseudoUeberschrift}[1]{
1499   \bigskip
1500   \noindent
1501   \textsf{\textbf{#1}}
1502   \noindent
1503 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1504 \newcommand{\liBeschriftung}[1]{
1505   \par
1506   \noindent
1507   \medskip
1508   \textbf{#1}:
1509   \medskip
1510   \noindent
1511 }

\hinweis
1512 \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.
1513 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

liEinbettung
1514 \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.
1515 \RequirePackage{xparse}
1516 \ExplSyntaxOn
```

```

1517 \NewDocumentEnvironment{ liAntwort } { 0{standard} }
1518 {
1519   \str_case:nn {#1} {
1520     {standard} {
1521       \def\beschriftung{}
1522       \mdfsetup{backgroundcolor=gray!5,linecolor=gray}
1523     }
1524     {richtig} {
1525       \def\beschriftung{richtig}
1526       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1527     }
1528     {falsch} {
1529       \def\beschriftung{falsch}
1530       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1531     }
1532     {muster} {
1533       \def\beschriftung{Musterlösung}
1534       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1535     }
1536   }
1537   \ifx\beschriftung\empty\else
1538     \noindent
1539     \textbf{\beschriftung{:}}
1540   \fi
1541   \begin{mdframed}
1542 }
1543 {\end{mdframed}}

```

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

```

1544 \NewDocumentEnvironment{ liAdditum } { o }
1545 {
1546   \begin{mdframed}[backgroundcolor=yellow!5]
1547     \IfNoValueTF {#1}
1548     { \liPseudoUeberschrift{Additum} }
1549     { \liPseudoUeberschrift{Additum:~#1} }
1550   }
1551 {\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.

```

1552 \NewDocumentEnvironment{ liExkurs }{o +b}{
1553   \vspace{0.2cm}%
1554   \begin{mdframed}[
1555     backgroundcolor=white,
1556     bottomline=false,
1557     innermargin=1cm,
1558     leftline=true,
1559     linecolor=black,
1560     linewidth=0.1cm,
1561     outermargin=1cm,
1562     rightline=false,
1563     topline=false,
1564   ]

```



```

1565 \footnotesize
1566 \noindent%
1567 \textbf{Exkurs:~\#1}\par%
1568 \noindent%
1569 #2
1570 \end{mdframed}
1571 \vspace{0.2cm}
1572 }{}

```

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

```

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

```

Weiterführende Literatur:

- Quelle 1
- Quelle 2

```

1573 \cs_new:Npn \listen@punkt #1 {\item #1}
1574 \NewDocumentEnvironment { liQuellen }{ +b }
1575 {
1576   \seq_clear_new:N \l_quellen
1577   \seq_set_split:Nnn \l_quellen {\item} {#1}
1578   \seq_remove_all:Nn \l_quellen {}
1579   \begin{mdframed}[backgroundcolor=white!0]
1580     \footnotesize
1581     \noindent
1582     \textsf{\textbf{Weiterführende~Literatur:}}
1583     \medskip
1584     \begin{compactitem}
1585       \seq_map_function:NN \l_quellen {\listen@punkt}
1586     \end{compactitem}
1587   \end{mdframed}
1588   %
1589   \makeatletter
1590   \par\@afterindentfalse\@afterheading
1591   \makeatother
1592 } {}

```

liLernkartei

```

1593 \NewDocumentEnvironment { liLernkartei }{ m +b }
1594 {
1595   \begin{mdframed}
1596     \footnotesize
1597     \noindent%
1598     \textbf{Lernkarteikarte:~\#1}\par%
1599     \noindent%
1600     #2
1601   \end{mdframed}
1602 } {}

```

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

```

1603 \NewDocumentEnvironment { liDiagramm }{ m +b }
1604 {
1605   \begin{mdframed}[backgroundcolor=white!0]
1606     \small
1607     \noindent%
1608     \textit{#1}:
1609     \begin{center}

```

```

1610 #2
1611 \medskip
1612 \end{center}
1613 \end{mdframed}
1614 } {}
1615 \ExplSyntaxOff

\liFussnoteUrl \liFussnoteUrl[\langle zusätzlicher-text \rangle]{\langle url \rangle} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
1616 \NewDocumentCommand{\liFussnoteUrl} { o m } {
1617 \footnote{\url{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1618 }
1619

\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.
1620 \NewDocumentCommand{\liFussnoteLink} { o m m } {
1621 \footnote{\href{#3}{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1622 }

\zB
1623 \def\zB{z.\,B. }

\ZB
1624 \def\ZB{Z.\,B. }

\dh
1625 \def\dh{d.\,h. }

1626

```

2.24 master-theorem.sty

1627 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1628 \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}

1629 \ExplSyntaxOn

1630 \RequirePackage{amsmath}

\liRundeKlammer

1631 \def\liRundeKlammer#1{

1632 \negthinspace \left(#1 \right)

1633 }

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

1634 \def\liThetaOhneMathe#1{

1635 \Theta \liRundeKlammer{#1}

1636 }

1637 \def\liTheta#1{

1638 \ifmmode

1639 \liThetaOhneMathe{#1}

1640 \else

1641 \$\liThetaOhneMathe{#1}\$

1642 \fi

1643 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
1644 \def\liOmegaOhneMathe#1{
1645 \Omega \liRundeKlammer{#1}
1646 }
1647 \def\liOmega#1{
1648 \ifmmode
1649 \liOmegaOhneMathe{#1}
1650 \else
1651 $\liOmegaOhneMathe{#1}$
1652 \fi
1653 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
1654 \def\liOOhneMathe#1{
1655 \mathcal{O} \liRundeKlammer{#1}
1656 }
1657 \def\liO#1{
1658 \ifmmode
1659 \liOOhneMathe{#1}
1660 \else
1661 $\liOOhneMathe{#1}$
1662 \fi
1663 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
1664 \def\liTOhneMathe#1#2{
1665 \tl_if_blank:nTF {#1}
1666 {}
1667 {#1 \cdot }
1668 T
1669 \liRundeKlammer{\frac{n}{#2}}
1670 }
1671 \def\liT#1#2{
1672 \ifmmode
1673 \liTOhneMathe{#1}{#2}
1674 \else
1675 $\liTOhneMathe{#1}{#2}$
1676 \fi
1677 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
1678 \def\liRekursionsGleichung{
1679 $T(n) = \liT{a}{b} + f(n)$
1680 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
1681 \def\liBedingungEins{
1682 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
1683 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
1684 \def\liBedingungZwei{
1685 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
1686 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
1687 \def\liBedingungDrei{
1688 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
1689 }

1690 \ExplSyntaxOff

```

\liMasterVariablen

```

1691 \def\liMasterVariablen{
1692   \begin{displaymath}
1693     T(n) = \liT{a}{b} + f(n)
1694   \end{displaymath}
1695
1696   \begin{itemize}
1697     \item[$a = $]
1698       Anzahl der Unterprobleme in der Rekursion
1699
1700     \item[$\textstyle{\frac{1}{b}} = $]
1701       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
1702       repräsentiert wird
1703
1704     \item[$f(n) = $]
1705       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
1706       die Kombination der Teillösungen entstehen
1707   \end{itemize}
1708   \footcite{wiki:master-theorem}
1709   \footcite[Seite 19-35 (PDF 11-24)]{aud:fs:2}
1710 }

```

\liMasterFaelle

```

1711 \def\liMasterFaelle{
1712   \begin{description}
1713     \item[1. Fall:]
1714        $T(n) \in \liTheta{n^{\log\sb{b}a}}$ 
1715
1716       \hfill falls \liBedingungEins
1717       für  $\varepsilon > 0$ 
1718
1719     \item[2. Fall:]
1720        $T(n) \in \liTheta{n^{\log\sb{b}a} \cdot \log n}$ 
1721
1722       \hfill falls \liBedingungZwei
1723
1724     \item[3. Fall:]
1725        $T(n) \in \liTheta{f(n)}$ 
1726
1727       \hfill falls \liBedingungDrei
1728       für  $\varepsilon > 0$ 
1729       und ebenfalls für ein  $c$  mit  $0 < c < 1$  und alle hinreichend großen  $n$ 
1730       gilt:
1731        $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$ 
1732     \end{description}
1733 }

```

\liMasterVariablenDeklaration

```

1734 \def\liMasterVariablenDeklaration#1#2#3{
1735   \begin{description}
1736     \item[Allgemeine Rekursionsgleichung:] \strut
1737
1738     \liRekursionsGleichung
1739
1740     \item[Anzahl der rekursiven Aufrufe ( $a$ ):] \strut
1741
1742     #1
1743
1744     \item[Anteil Verkleinerung des Problems ( $b$ ):] \strut
1745
1746     um  $\frac{1}{\#2}$  also  $b = \#2$ 
1747
1748     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut

```

```

1749
1750     $#3$
1751
1752     \item[Ergibt folgende Rekursionsgleichung:] \strut
1753
1754     $T(n) = \liT{\#1}{\#2} + \#3$
1755 \end{description}
1756 }

```

\liMasterFallRechnung

```

1757 \def\liMasterFallRechnung#1#2#3{
1758   \begin{description}
1759     \item[1. Fall:] \liBedingungEins:
1760
1761     #1
1762
1763     \item[2. Fall:] \liBedingungZwei:
1764
1765     #2
1766
1767     \item[3. Fall:] \liBedingungDrei:
1768
1769     #3
1770   \end{description}
1771 }

```

\liMasterExkurs

```

1772 \def\liMasterExkurs{
1773   \begin{liExkurs}[Master-Theorem]
1774     \liMasterVariablen
1775
1776     \noindent
1777     Dann gilt:
1778
1779     \liMasterFaelle
1780   \end{liExkurs}
1781 }

```

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)

```

1782 \def\liMasterWolframLink#1{
1783   Berechne die Rekursionsgleichung auf WolframAlpha:
1784   \href{https://www.wolframalpha.com/input/?i=#1}{WolframAlpha}
1785 }
1786

```

2.25 mathe.sty

```
1787 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1788 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
1789
1790 % for example \ltimes \rtimes
1791 %\RequirePackage{amssymb}
1792 \RequirePackage{amsmath}
1793
1794 %%
1795 % \mlq \mrq
1796 %%
1797 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
1798 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
1799
```

2.26 minimierung.sty

```

1800 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1801 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
1802 Minimierungsalgorithmus von einem NEA zu einem DEA]

1803 \liLadePakete{typographie}

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

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

\liFussnoten

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

\liFussnote

1804 \def\liFussnote#1{\$x_{#1}\$}

1805 \def\li@fussnote@text#1#2{
1806 \liFussnote{#1}
1807 \quad
1808 {\footnotesize #2}
1809 }

\liFussnoteEinsText

1810 \def\liFussnoteEinsText{
1811 \li@fussnote@text{1}
1812 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
1813 }

\liFussnoteZweiText

1814 \def\liFussnoteZweiText{
1815 \li@fussnote@text{2}
1816 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
1817 }

\liFussnoteDreiText

1818 \def\liFussnoteDreiText{
1819 \li@fussnote@text{3}

```



```

1820 {In weiteren Iterationen markierte Zustände.}
1821 }

\liFussnoteVierText
1822 \def\liFussnoteVierText{
1823   \li@fussnote@text{4}
1824   {...}
1825 }

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



1826 \def\liFussnoten{
1827   \bigskip
1828
1829   \noindent
1830   \liFussnoteEinsText
1831
1832   \noindent
1833   \liFussnoteZweiText
1834
1835   \noindent
1836   \liFussnoteDreiText
1837
1838   \noindent
1839   \liFussnoteVierText
1840 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
1841 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
1842 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
1843 \def\liZustandsPaar#1#2{
1844   $(
1845     \liZustandsPaarVariablenName_#1,
1846     \liZustandsPaarVariablenName_#2
1847   )$
1848 }

liUebergangsTabelle
1849 \renewcommand{\arraystretch}{1.4}
1850 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
1851   \liPseudoUeberschrift{Übergangstabelle}
1852   \begin{center}
1853     \begin{tabular}{r|l|l}
1854       \textbf{Zustandspaar} & \textbf{\#1} & \textbf{\#2} \\ \hline
1855     \end{tabular}
1856   \end{center}
1857 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle: Minimierungstabelle (Table filling)

1859 \ExplSyntaxOn
1860 \def\liUeberschriftDreiecksTabelle{
1861   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
1862 }

```

\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. —————

```

1863 \def\liMinimierungErklaerung{
1864   %\footcite[Seite~19]{koenig}
1865   \liParagraphMitLinien{
1866     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
1867     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
1868     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
1869     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
1870      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
1871     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
1872     somit~zueinander~( $k - 1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
1873     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
1874     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
1875   }
1876 }
1877 \ExplSyntaxOff

```

1878

2.27 normalformen.sty

```

1879 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1880 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10]
1881 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
1882 Attributhülle]
1883 \liLadePakete{mathe}
1884 \directlua{
1885   helfer = require('lehramt-informatik-helfer')
1886   normalformen = require('lehramt-informatik-normalformen')
1887 }

```

Let-Abkürzungen

```

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

```

```

1888 \def\liTeilen#1{
1889   \directlua{tex.print(normalformen.teilen('#1'))}
1890 }

```

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

```

1891 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(#1)}
1892 \def\liAttributHuelle#1{
1893   \ifmmode
1894     \liAttributHuelleOhneMathe{#1}
1895   \else
1896     $\liAttributHuelleOhneMathe{#1}$
1897   \fi
1898 }

```

\liAttributMenge Let-Abkürzung: `\let\m=\liAttributMenge`
`1899 \def\liAttributMenge#1{\{ \textit{#1} \}}`

liAHuelle

```

1900 \NewDocumentEnvironment{ liAHuelle } { +b } {
1901   \begingroup
1902   \footnotesize
1903   \begin{multline*}
1904     #1
1905   \end{multline*}
1906   \endgroup
1907 } { }

```

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

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

```

1908 \def\liAttributHuelleLinksReduktion#1#2#3{
1909   \shoveleft{
1910     \liAttributHuelleOhneMathe{FA, \liAttributMenge{\liAttributMenge{#1} - \liAttributMenge{#2}}
1911   } \\\
1912   \shoveright{
1913     \liAttributMenge{#3}
1914   } \\\
1915 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
    \liFunktionaleAbhaengigkeit{A, B -> C, D} Regulärer Ausdruck zum Konvertieren:
    \$(.*) \rightarrow (.*)\$ \fa{$1 -> $2}
1916 \def\liFunktionaleAbhaengigkeit#1{%
1917   \directlua{normalformen.drucke_funk_abhaengigkeit('#1')}%
1918 }

\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;
1919 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
1920   \par
1921   \noindent
1922   #1 $= \{$
1923   \par
1924   \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
1925   \par
1926   \noindent$}\$
1927 }

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

```

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

```

Regulärer Ausdruck zum Konvertieren: \$(R.*)\((.*)\)\$ \liRelation[$1]{$2}
1928 \NewDocumentCommand {\liRelation} { O{R} m } {
1929   $\directlua{
1930     local name = helper.konvertiere_tiefgestellt('#1')
1931     tex.print(name)
1932   }$(\textit{\,#2\,})
1933 }
1934

```

2.28 petri.sty

```
1935 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1936 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]
```

Let-Abkürzungen

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

```
1937 \RequirePackage{tikz}
1938 \usetikzlibrary{petri,arrows.meta}
```

Für die Darstellungsmatrix

```
1939 \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

```
1940 \def\liPetriSetzeSchluessel{%
1941   \def\TmpTransitionOne{}%
1942   \def\TmpTransitionTwo{}%
1943   \def\TmpTransitionThree{}%
1944   \def\TmpTransitionFour{}%
1945   \def\TmpTransitionFive{}%
1946   \def\TmpTransitionSix{}%
1947   \def\TmpTransitionSeven{}%
1948   \def\TmpTransitionEight{}%
1949   \def\TmpTransitionNine{}%
1950   \def\TmpTransitionTen{}%
1951   \pgfkeys{/petri/.cd,
1952     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
1953     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
1954     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
1955     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
1956     p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
```

```

1957 p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
1958 p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
1959 p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
1960 p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
1961 p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
1962 t1/.store in=\TmpTransitionOne,t1/.default=activated,
1963 t2/.store in=\TmpTransitionTwo,t2/.default=activated,
1964 t3/.store in=\TmpTransitionThree,t3/.default=activated,
1965 t4/.store in=\TmpTransitionFour,t4/.default=activated,
1966 t5/.store in=\TmpTransitionFive,t5/.default=activated,
1967 t6/.store in=\TmpTransitionSix,t6/.default=activated,
1968 t7/.store in=\TmpTransitionSeven,t7/.default=activated,
1969 t8/.store in=\TmpTransitionEight,t8/.default=activated,
1970 t9/.store in=\TmpTransitionNine,t9/.default=activated,
1971 t10/.store in=\TmpTransitionTen,t10/.default=activated,
1972 scale/.store in=\TmpScale,scale/.default=0.5,
1973 x/.store in=\TmpX,x/.default=5,
1974 y/.store in=\TmpY,y/.default=5,
1975 }%
1976 }

1977 \tikzset{
1978   li petri/.style={
1979     activated/.style={
1980       very thick
1981     },
1982     inhibitor/.style={
1983       {Circle[open,length=2mm,fill=white]}-
1984     }
1985   }
1986 }

```

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

```

1987 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
1988 \def\liPetriTransitionsName#1{
1989   \ifmmode
1990     \liPetriTransitionsNameOhneMathe{#1}
1991   \else
1992     $\liPetriTransitionsNameOhneMathe{#1}$
1993   \fi
1994 }

```

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

```

1995 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
1996   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
1997 }

```

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

```

1998 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

1999 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2000 }

```

2.29 potenzmengen-konstruktion.sty

```
2001 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2002 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2003 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2004 \liLadePakete{formale-sprachen}
2005 \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}
    }
  }
}
```

```
2006 \def\liZustandsMengenSammlung#1#2{
2007   \liZustandsnameGross{#1}
2008   {
2009     \footnotesize
2010     \liPotenzmenge{
2011       \str_case:nn {#1} {#2
2012       }
2013     }
2014 }
```

```
\liZustandsMengenSammlungNr
```

```
2015 \def\liZustandsMengenSammlungNr#1#2{
2016   \liZustandsnameGross{#1}
2017   {
```

```
2018     \footnotesize
2019     \liZustandsmengeNr{
2020         \str_case:nn {#1} #2
2021     }
2022 }
2023 }

2024 \ExplSyntaxOff
2025
```


2.30 pseudo.sty

2026 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2027 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen

2028 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>$E' \leftarrow \emptyset$;</p> <p>$L \leftarrow E$;</p> <p>Sortiere die Kanten in L aufsteigend nach ihrem Kantengewicht.;</p> <p>while $L \neq \emptyset$ do</p> <table border="0"> <tr> <td style="padding-right: 10px;"> </td> <td>wähle eine Kante $e \in L$ mit kleinstem Kantengewicht;</td> </tr> <tr> <td style="padding-right: 10px;"> </td> <td>entferne die Kante e aus L;</td> </tr> <tr> <td style="padding-right: 10px;"> </td> <td>if der Graph $(V, E' \cup \{e\})$ keinen Kreis enthält then</td> </tr> <tr> <td style="padding-right: 10px;"> </td> <td style="padding-left: 10px;">$E' \leftarrow E' \cup \{e\}$;</td> </tr> <tr> <td style="padding-right: 10px;"> </td> <td>end</td> </tr> </table> <p>end</p> <p>Result: $M = (V, E')$ ist ein minimaler Spannbaum von G.</p>		wähle eine Kante $e \in L$ mit kleinstem Kantengewicht;		entferne die Kante e aus L ;		if der Graph $(V, E' \cup \{e\})$ keinen Kreis enthält then		$E' \leftarrow E' \cup \{e\}$;		end
	wähle eine Kante $e \in L$ mit kleinstem Kantengewicht;									
	entferne die Kante e aus L ;									
	if der Graph $(V, E' \cup \{e\})$ keinen Kreis enthält then									
	$E' \leftarrow E' \cup \{e\}$;									
	end									

2029 \RequirePackage[german,boxruled]{algorithm2e}

2030

2.31 pumping-lemma.sty

2031 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2032 \ProvidesPackage{lehramt-informatik-pumping-lemma}[2021/08/11 Enthält die
 2033 Definitionen für das Pumping-Lemma in der Regulären Sprache und
 2034 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2035 \def\liPumpingRegulaer{%
2036   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2037   alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2038    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2039   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2040
2041   \begin{enumerate}
2042     \item  $|v| \geq 1$ 
2043     (Das Wort  $v$  ist nicht leer.)
2044
2045     \item  $|uv| \leq j$ 
2046     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2047
2048     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w$  in  $L$ 
2049     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2050     Sprache  $L$ )
2051   \end{enumerate}
2052
2053   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2054   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2055 }
```

\liPumpingKontextfrei

```
2056 \def\liPumpingKontextfrei{%
2057   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2058   sich alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  zerlegen lassen in
2059    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2060
2061   \begin{enumerate}
2062     \item  $|vx| \geq 1$ 
2063     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2064
2065     \item  $|vwx| \leq j$ 
2066     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2067
2068     \item Für alle  $i$  in  $\mathbb{N}_0$  gilt  $uv^iwx^iy$  in  $L$  (Für jede
2069     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2070     Sprache  $L$ )
2071   \end{enumerate}
2072 }
2073
```

2.32 quicksort.sty

```

2074 % https://tex.stackexchange.com/a/142634
2075 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2076 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2077
2078 %-----
2079 % USAGE:
2080 % \QSinitialize{comma, separated, numerical, values}
2081 % \loop
2082 % \QSpivotStep
2083 % \ifnum\value{pivotcount}>0
2084 %   \QSSortStep
2085 % \repeat
2086 %-----
2087
2088 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2089 \RequirePackage{xintfrac, xinttools}
2090
2091 \RequirePackage{tikz}
2092
2093 %-----
2094 % FIRST PART: TikZ styles and macros for the actual drawing
2095 \newcounter{cellcount}% used for coordinates of the node
2096 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2097
2098 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2099 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2100 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2101 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2102 % by police of LaTeX good conduct ? )
2103 \tikzset{!/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2104          o/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2105          r/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2106 % this is the "b" style as used in the image below
2107          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2108 % nicer:
2109          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2110          g/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2111
2112 % NOTE the b style was originally the same as the r(aised) style apart from
2113 % not being raised, but I find it nicer with a somewhat different
2114 % specification. I have not updated the images though.
2115
2116 % How the nodes are drawn depending on whether on the left of the pivot value
2117 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2118
2119 \def\DecoLEFT #1{%
2120   \xintFor* ##1 in {#1} \do
2121     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2122 }
2123
2124 \def\DecoINERT #1{%
2125   \xintFor* ##1 in {#1} \do
2126     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2127 }
2128
2129 \def\DecoRIGHT #1{%
2130   \xintFor* ##1 in {#1} \do
2131     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2132 }
2133
2134 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2135   \xintFor* ##1 in {#1} \do

```

```

2136     {\stepcounter{cellcount}}%
2137     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2138 }
2139
2140 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2141     \xintFor* ##1 in {#1} \do
2142     {\stepcounter{cellcount}}%
2143     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2144 }
2145
2146 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2147     \xintFor* ##1 in {#1} \do
2148     {\stepcounter{cellcount}}%
2149     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2150 }
2151
2152 %-----
2153 % SECOND PART: the actual sorting routines.
2154
2155 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2156 \def\QS@sort@b #1{\ifcase #1
2157     \expandafter\QS@sort@empty
2158     \or\expandafter\QS@sort@single
2159     \else\expandafter\QS@sort@c
2160     \fi
2161 }%
2162 \def\QS@sort@empty #1{}
2163 \def\QS@sort@single #1{\QS@Ir {#1}}
2164
2165 % This step is to pick the last as pivot.
2166 \def\QS@sort@c #1%
2167     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2168
2169 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2170 % The trick with \xintApplyUnbraced is that for example when selecting
2171 % the elements smaller than pivot, if we had been using \xintApply we
2172 % would have had at the minimum an empty brace pair. Thus we use the
2173 % "unbraced" variant, but then the \QS@select@smaller has added in
2174 % anticipation a level of braces.
2175 \def\QS@sort@d #1#2{%
2176     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2177     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2178     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2179 }%
2180 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2181 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2182 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2183
2184 %
2185 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2186 %
2187 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2188 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2189 % latter must handle correctly an empty argument.
2190
2191 %-----
2192 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QS@initialize.
2193
2194 % This draws all with suitable highlighting for the newly chosen pivots
2195 % (which will be shown raised)

```

```

2196 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2197             \let\QSIr\DecoINERT
2198             \let\QSIrr\DecoINERT
2199             \let\QSRr\DecoRIGHTwithPivot
2200 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2201             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2202             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2203 }
2204
2205 % This sorts and then draws, showing where the pivot chosen in the previous
2206 % step go. Next time they will have become "inert". If pivotcount is still at
2207 % zero on exit from \QSpivotStep, then this is the signal to stop before
2208 % executing \QSsortStep.
2209 \def\QSsortStep {\def\QSLr {\noexpand\QS@sort@a}%
2210             \def\QSRr {\noexpand\QS@sort@a}%
2211             \def\QSIr {\noexpand\QSIrr}%
2212             \let\QSIrr\relax
2213             \edef\QS@list{\QS@list}%
2214             \let\QSLr\relax
2215             \let\QSRr\relax
2216             \let\QSIr\relax
2217             \edef\QS@list{\QS@list}%
2218             \let\QSLr\DecoLEFT
2219             \let\QSIr\DecoINERTwithPivot
2220             \let\QSIrr\DecoINERT
2221             \let\QSRr\DecoRIGHT
2222 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2223             \setcounter{cellcount}{0}%
2224             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2225 }
2226
2227 \def\QSinitialize #1{%
2228     % first, we convert the comma separated values into a list of braced items
2229     % we use an \edef, and anyhow many \edef's will be used later
2230     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2231     \let\QSRr\DecoRIGHT
2232     % The \QSRr marker mutated to draw the last element as
2233     % pivot and the earlier ones with the suitable style.
2234     %
2235     % The list of marked braced items \QS@list is used both for drawing
2236     % (as here) and for doing the exchange of elements during sort.
2237     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2238             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2239 }
2240

```

2.33 relationale-algebra.sty

```

2241 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2242 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]
2243 \RequirePackage{amsmath}
2244 \RequirePackage{amssymb}

```

```

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

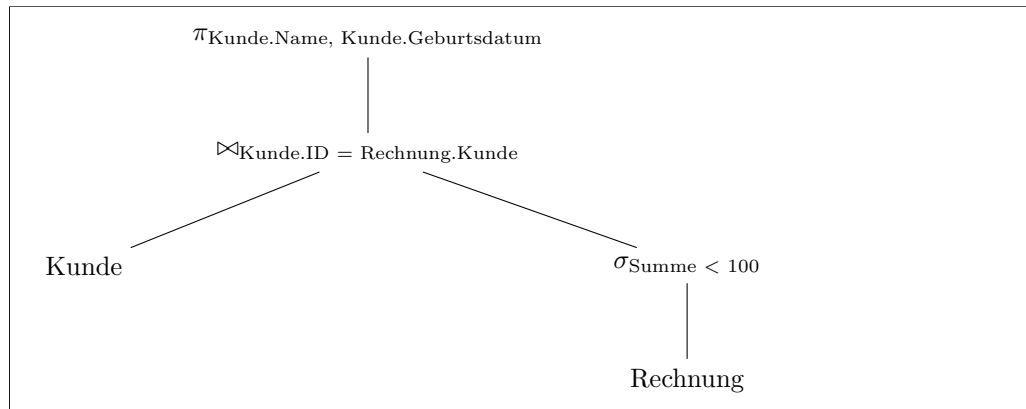
  \node[below=of pi]
    (theta join) {\bowtie_{\text{Kunde.ID = Rechnung.Kunde}}};
    edge (pi);

  \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}

```



```

2245 \RequirePackage{tikz}
2246 \usetikzlibrary{positioning}

    Privates Makros, das zwei Querstriche erzeugt.
2247 \def\o@join{\setbox0=\hbox{\bowtie}}%
2248   \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2249 }

```

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

```
2250 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}
```

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

```
2251 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}
```

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

```
2252 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
```

```
2253
```

2.34 rmodell.sty

```

2254 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2255 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2256 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2257 Datenbanken.]
2258 \RequirePackage{soul}

```

Let-Abkürzungen

```

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

```

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

```

2259 \def\liPrimaer#1{\ul{#1}}

```

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

```

2260 \def\liFremd#1{{\setul{-0.9em}{}}\ul{#1}}

```

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

```

2261 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2262 \ExplSyntaxOn
2263 \NewDocumentEnvironment { liRmodell }
2264 { +b }
2265 {
2266   \medskip
2267   {
2268     \linespread{2}
2269     \setlength{\parindent}{0pt}
2270     \li@Rmodell@Schrift#1
2271   }
2272   \medskip
2273 } {}
2274 \ExplSyntaxOff

```

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

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

```

2275 \def\liRelationMenge#1#2{
2276 \noindent
2277 #1 : \{ [ #2 ] \}
2278 \par
2279 }

```

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

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

```

2280 \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}

```

```

2281 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}

```

```

2282

```

2.35 sortieren.sty

```
2283 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2284 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
2285 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}
}
```



```
2286 \RequirePackage{tikz}
2287 \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.

```
2288 \def\liVertauschen#1{
2289   \directlua{
2290     local sortieren = require('lehramt-informatik-sortieren')
2291     sortieren('#1')
2292   }
2293 }
```

`\liSortierPfeil`

```
2294 \def\liSortierPfeil#1#2{
2295   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2296 }
```

`\liSortierPfeilUnten`

```
2297 \def\liSortierPfeilUnten#1#2{
2298   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2299 }
```

`\liSortierMarkierung`

```
2300 \def\liSortierMarkierung#1#2{\node[
2301   draw,
2302   very thick,
2303   fit=(reihe.#1) (reihe.#2),
2304   inner sep=0pt
2305 ] {}
2306 }

2307 \tikzset{
2308   li sortierung zahlenreihe/.style={
2309     draw,
2310     thin,
2311     font=\large,
2312     rectangle split horizontal,
2313     rectangle split,
2314   }
2315 }
```



```

2316 % https://tex.stackexchange.com/a/140895
2317 \RequirePackage{forest,xstring}
2318 \usetikzlibrary{calc}
2319
2320 \makeatletter
2321 \pgfmathdeclarefunction{strrepeat}{2}{%
2322   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2323   \let\pgfmathresult\pgfutil@empty
2324   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2325     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2326     \advance\pgfmath@count-1\relax
2327   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2328 \makeatother
2329
2330 \def\myNodes{}
2331
2332 \ExplSyntaxOn
2333 \newcommand*\sortList[1]{%
2334   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2335 \ExplSyntaxOff
2336
2337 \forestset{
2338   sort/.code={%
2339     \pgfmathparse{level()>\forestSortLevel}%
2340     \ifnum\pgfmathresult=0
2341       \StrSubstitute{\forestov{content}}{ }{,}{\myList}%
2342       \sortList\myList
2343       \StrSubstitute{\myList}{,}{ }{\myList}%
2344       \pgfmathparse{strrepeat("1",level())}%
2345       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-1!({\forestov{name}}$)
2346         (m\forestov{name}) {\myList}}%
2347       \pgfmathparse{level()=\forestSortLevel}%
2348       \ifnum\pgfmathresult=1
2349         \forestOget{\forestov{@first}}{name}\forestFirst
2350         \forestOget{\forestov{@last}}{name}\forestLast
2351         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2352           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2353       \fi
2354       \ifnum\forestov{@parent}=0\else
2355         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2356       \fi
2357       \gappto\myNodes{;}%
2358     \fi}}
2359
2360 \forestset{sort level/.code=%
2361   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2362   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2363

```

2.36 spalten.sty

```
2364 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2365 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2366 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2367 realisiert werden kann.]
2368 \RequirePackage{multicol}
```

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

```
2369 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2370
```

2.37 struktogramm.sty

```
2371 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2372 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2373 Paket struktex zum Setzen von Struktogrammen]
2374 \RequirePackage{struktex}
2375
```

2.38 syntax.sty

```
2376 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2377 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2378 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2379 \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
```

```
2380 \ExplSyntaxOn
2381 \directlua{
2382   syntax = require('lehramt-informatik-syntax')
2383   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2384   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2385   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2386   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2387   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2388   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2389 }
2390 \RequirePackage{hyperref}
2391 \RequirePackage{minted}
2392 % pygmentize -L styles
2393 \usemintedstyle{colorful}
2394 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2395 %\AfterEndEnvironment{minted}{\end{mdframed}}
2396 %\setminted{breaklines=true,linenos}
2397 \setminted{
2398   breaklines=true,
2399   linenos,
2400   fontsize=\footnotesize,
2401 }
```

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

Let-Abkürzung: `\let\j=\liJavaCode`

```
2402 \def\liJavaCode#1{\,\mintinline{java}|#1|\,}
```

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

```
2403 \def\liLatexCode#1{\mintinline{latex}|#1|}
```

```
2404 \def\li@GithubLink#1#2{
2405   \begin{flushright}
2406     \tiny
2407     Code-Beispiel~auf~Github~ansehen:~
2408     \href{#1}{\nolinkurl{#2}}
2409   \end{flushright}
2410 }
```

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

```
2411 \NewDocumentCommand{\liJavaDatei}{ O{firstline=3} m }{
2412   \inputminted[#1]{java}{
2413     \directlua{
2414       syntax.drucke_absoluten_pfad('#2', false)
2415     }
2416   }
2417   \li@GithubLink
```

```

2418     {\directlua{syntax.drucke_github_url('#2', false)}}
2419     {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2420 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2421 \NewDocumentCommand{\liJavaTestDatei}{ O{firstline=3} m }{
2422   \inputminted[#1]{java}{
2423     \directlua{
2424       syntax.drucke_absoluten_pfad('#2', true)
2425     }
2426   }
2427   \li@GithubLink
2428   {\directlua{syntax.drucke_github_url('#2', true)}}
2429   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2430 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([.
\liJavaExamen$1{$2}{$3}{$4}{$5}
2431 \NewDocumentCommand{\liJavaExamen}{ O{firstline=3} m m m m }{
2432   \inputminted[#1]{java}{
2433     \directlua{
2434       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2435     }
2436   }
2437
2438   \li@GithubLink
2439   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2440   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2441 }

\liAssemblerCode
2442 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2443 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2444   \inputminted{asm}{#1}
2445 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2446 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2447   \inputminted{componentpascal}{#1}
2448 }

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2449 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2450 \NewDocumentCommand{\liHaskellDatei}{ m }{
2451   \inputminted{haskell}{#1}
2452 }

2453 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2454 \def\liSqlCode#1{\mintinline{sql}|#1|}

2455

```

2.39 syntaxbaum.sty

```
2456 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2457 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2458 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
2459 \RequirePackage{tikz-qtrees}
2460
2461 \tikzset{li parsetree/.style={
2462     every internal node/.style={
2463         draw,circle
2464     },
2465     every leaf node/.style={
2466         draw,rectangle
2467     },
2468 }
2469 }
2470
```

2.40 synthese-algorithmus.sty

```
2471 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2472 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2473 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2474 Relation in die 3. Normalform]

2475 \liLadePakete{normalformen,mathe,typographie}
2476 \ExplSyntaxOn

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

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

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

1. Kanonische Überdeckung

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

(a) Linksreduktion

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

(b) Rechtsreduktion

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

(c) Löschen leerer Klauseln

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

(d) Vereinigung

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

2. Relationsschemata formen

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

3. Schlüssel hinzufügen

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

4. Entfernung überflüssiger Teilschemata

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

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

```

2477 \def\liSyntheseUeberschrift#1{
2478   {
2479     \bfseries
2480     \sffamily
2481     \str_case:nn {#1} {
2482       {1} {Kanonische-Überdeckung}
2483       {1-1} {Linksreduktion}
2484       {1-2} {Rechtsreduktion}
2485       {1-3} {Löschen-leerer-Klauseln}
2486       {1-4} {Vereinigung}
2487       {2} {Relationsschemata-formen}
2488       {3} {Schlüssel-hinzufügen}
2489       {4} {Entfernung-überflüssiger-Teilschemata}
2490     }
2491   }
2492 }

```

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

```

2493 \def\li@synthese@erklaerung@texte#1{
2494   \str_case:nn {#1} {
2495     {1} {
2496       Die-kanonische-Überdeckung---also-die-kleinst-mögliche-noch-
2497       äquivalente-Menge-von-funktionalen-Abhängigkeiten-kann-in-vier-
2498       Schritten-erreicht-werden.
2499     }
2500     {1-1} {
2501       Führe-für-jede-funktionale-Anhängigkeit-
2502       $\alpha\rightarrow\beta$-in-$F$-die-Linksreduktion-durch,-
2503       überprüfe-also-für-alle-
2504       $A\in\alpha$,~ob-$A$-überflüssig-ist,-d.h.-ob-
2505       $\beta\subseteqq\liAttributHuelle{F,\alpha\cup A}$.
2506     }
2507     {1-2} {
2508       Führe-für-jede-(verbliebene)-funktionale-Abhängigkeit-$\alpha\rightarrow\beta$-die-Rechtsreduktion-durch,-überprüfe-also-für-
2509       alle-$B\in\beta$,~ob-$B\in\liAttributHuelle{F,\alpha\rightarrow\beta}$,-
2510       $\alpha\rightarrow\beta\cup(\alpha\rightarrow\beta)$,-
2511       $\alpha$-gilt.-In-diesem-Fall-ist-$B$-auf-der-rechten-Seite-
2512       überflüssig-und-kann-eliminiert-werden,-dh-$\alpha\rightarrow\beta$-wird-durch-$\alpha\rightarrow(\beta\cup B)$-
2513       ersetzt.
2514     }
2515     {1-3} {
2516       Entferne-die-funktionalen-Abhängigkeiten-der-Form-$\alpha\rightarrow\emptyset$,~die-im-2.-Schritt-möglicherweise-
2517       entstanden-sind.
2518     }
2519     {1-4} {
2520       Fasse-mittels-der-Vereinigungsregel-funktionale-Abhängigkeiten-
2521       der-Form-$\alpha\rightarrow\beta_1,\dots,\alpha\rightarrow\beta_n$,~so-dass-$\alpha\rightarrow\beta_1\cup\dots\cup\beta_n$-verbleibt.
2522     }
2523     % Kemper Seite 197
2524     {2} {
2525       Erzeuge-für-jede-funktionale-Abhängigkeit-$\alpha\rightarrow\beta$-in-$F$-ein-Relationenschema-$\mathcal{R}_{\alpha}$-
2526       :=$\alpha\cup\beta$.
2527     }
2528     {3} {
2529       Falls-eines-der-in-Schritt-2.-erzeugten-Schemata-$\mathcal{R}_{\alpha}$-
2530       einen-Schlüsselkandidaten-von-$\mathcal{R}$-bezüglich-$F$-

```



```

2537 enthält,~sind-wir~fertig,~sonst~wähle-einen-Schlüsselkandidaten-
2538  $\mathcal{K} \sim \text{subseq} \mathcal{R}$ ~aus-und-definiere-folgendes-
2539 zusätzliche-Schema:~ $\mathcal{R} \setminus \mathcal{K} \sim \mathcal{K}$ ~und-
2540  $\mathcal{F} \setminus \mathcal{K} \sim \emptyset$ 
2541 }
2542 {4} {
2543 Eliminiere-diejenigen-Schemata- $\mathcal{R} \setminus \alpha$ ~,~die-in-einem-
2544 anderen-Relationenschema- $\mathcal{R} \setminus \alpha'$ ~-enthalten-sind,~d.h.~
2545  $\mathcal{R} \setminus \alpha \sim \text{subseq} \mathcal{R} \setminus \alpha'$ ~.
2546 }
2547 }
2548 }
2549 \def\liSyntheseErklaerung#1{
2550 {
2551 \itshape
2552 \footnotesize
2553 \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
2554 }
2555 }

```

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

```

2556 \def\liSyntheseUeberErklaerung#1{
2557 \liSyntheseUeberschrift{#1}\par
2558 \liSyntheseErklaerung{#1}
2559 }

```

```

2560 \ExplSyntaxOff
2561

```

2.41 tabelle.sty

2562 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2563 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]

2564 \RequirePackage{tabularx}

2565

2.42 typographie.sty

```
2566 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2567 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
2568 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
2569 formatierung.sty definiert.]
```

```
2570 \ExplSyntaxOn
```

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

```
2571 \RequirePackage{fontawesome}
```

```
\liErledigt \liErledigt: ☑
```

```
2572 \let\liErledigt=\faCheckSquareO
```

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

```
2573 \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 —

```
2574 \def\liParagraphMitLinien#1{
```

```
2575   \noindent
```

```
2576   \vrule height 2pt depth -1.6pt width 0.4cm
```

```
2577   \enspace
```

```
2578   #1
```

```
2579   \enspace
```

```
2580   \leaders\vrule height 2pt depth -1.6pt \hfill \null
```

```
2581   \par
```

```
2582   \medskip
```

```
2583 }
```

```
2584 \ExplSyntaxOff
```

```
2585
```

2.43 uml.sty

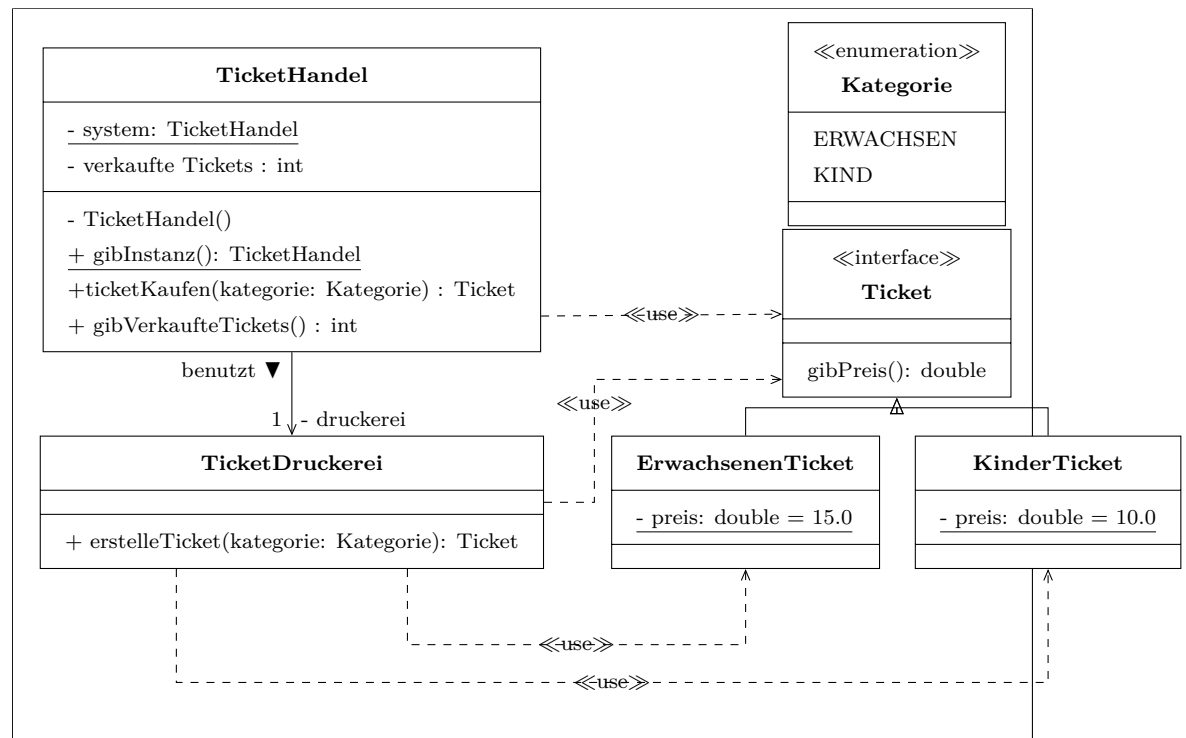
```

2586 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2587 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
2588 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
2589 Erweiterung bereitstellt]

2590 \RequirePackage{tikz-uml}
2591 \RequirePackage{tikz-uml-activity}
2592 % Not compatible with wasysym
2593 %\RequirePackage{mathabx}
2594 \RequirePackage{wasysym}
2595 \usetikzlibrary{positioning}

2596 \tikzumlset{
2597   fill class=white!0,
2598   font=\footnotesize,
2599   fill object=white!0,
2600   fill note=white!0,
2601   fill state=white!0,
2602   % Use case
2603   fill usecase=white!0,
2604   fill system=white!0,
2605 }

```



```

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

```

```

2606 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
2607   \def\@liDirLeft{}
2608   \def\@liDirRight{ \RIGHTarrow}
2609   \pgfkeys{/lese/dir/.is choice}
2610   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
2611   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
2612   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{ \LEFTarrow }}}
2613   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
2614
2615   \def\@liPos{above}
2616   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
2617

```

```

2618 \def\@liDistance{0cm}
2619 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
2620
2621 \pgfkeys{/lese/.cd,#1}
2622
2623 \node[\@liPos = \@liDistance of #2-middle] {
2624   \@liDirLeft{\footnotesize#2}\@liDirRight
2625 };
2626 }
2627

```

2.44 vollstaendige-induktion.sty

```
2628 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2629 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01
2630 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
2631 Überschriften für die einzelnen Schritte]
```

Let-Abkürzungen

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

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

Lade häufig benötigte Pakete

```
2632 \RequirePackage{lehramt-informatik-typographie}
2633 \RequirePackage{lehramt-informatik-mathe}
2634 \RequirePackage{lehramt-informatik-syntax}
2635 \ExplSyntaxOn
```

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

Let-Abkürzung: `\let\m=\liInduktionMarkierung`

```
2636 \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

```
2637 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
2638 \def\liInduktionAnfang{
```

```
2639   \liPseudoUeberschrift{Induktionsanfang}
```

```
2640
```

```
2641   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
2642   \liParagraphMitLinien{
```

```
2643     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
```

```
2644   }
```

```
2645 }
```

\liInduktionVoraussetzung

```
2646 \def\liInduktionVoraussetzung{
```

```
2647   \liPseudoUeberschrift{Induktionsvoraussetzung}
```

```
2648
```

```
2649   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
2650   \liParagraphMitLinien{
```

```
2651     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ ~.
```

```
2652   }
```

```
2653 }
```

\liInduktionSchritt

```
2654 \def\liInduktionSchritt{
```

```
2655   \liPseudoUeberschrift{Induktionsschritt}
```

```
2656
```

```
2657   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
2658   \liParagraphMitLinien{
```

```
2659     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
```

```
2660     auch~ $A(n=k+1)$ ~wahr~sein~muss.
```

```
2661   }
```

```
2662 }
```

```
2663 \ExplSyntaxOff
```

```
2664
```

2.45 wasserfall.sty

```
2665 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2666 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
2667 \RequirePackage{tikz}
2668 \tikzset{wasserfall/.style={
2669   >=stealth,
2670   node distance = 2mm and -8mm,
2671   start chain = A going below right,
2672   every node/.style = {
2673     draw,
2674     text width=24mm,
2675     minimum height=12mm,
2676     align=center,
2677     inner sep=1mm,
2678     fill=white,
2679     drop shadow={fill=black},
2680     on chain=A
2681   },
2682 }}
2683 \usetikzlibrary{chains,positioning,shadows}
2684
```


2.46 wpkalkuel.sty

2685 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2686 \ProvidesPackage{lehramt-informatik-wpkalkuel}[2020/06/13]

Let-Abkürzungen

\let\wp=\liWpKalkuel

\let\equivalent=\liWpEquivalent

\let\erklaerung=\liWpErklaerung

2687 \RequirePackage{amsmath}

2688 \ExplSyntaxOn

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

2689 \def\liWpKalkuelOhneMathe#1#2{

2690 \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)

2691 }

2692 \def\liWpKalkuel#1#2{

2693 \ifmmode

2694 \liWpKalkuelOhneMathe{#1}{#2}

2695 \else

2696 \$\liWpKalkuelOhneMathe{#1}{#2}\$

2697 \fi

2698 }

\MatheEnv

2699 \def\MatheEnv#1{

2700 \medskip

2701

2702 \hspace{1em}#1

2703

2704 \medskip

2705 }

\Mathe

2706 \def\Mathe#1{

2707 \MatheEnv{#1\$}

2708 }

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

2709 \def\liWpEquivalent#1{

2710 \MatheEnv{\$\equiv\$\hspace{1em}#1\$}

2711 }

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

2712 \newlength{@Skip@Erklaerung@Reset}

2713 \def\liWpErklaerung#1{

2714 \setlength{@Skip@Erklaerung@Reset}{\leftskip}

2715 \setlength{\leftskip}{0.5cm}

2716

2717 \par

2718 \noindent

2719 {

2720 \scriptsize

2721 #1

2722 }

2723 \par

2724

2725 \setlength{\leftskip}{@Skip@Erklaerung@Reset}

2726 }

\liWpErklaerungVerzweigung

```

2727 \def\liWpErklaerungVerzweigung{
2728   $\liWpKalkuelOhneMathe{if~\{-b~\}~then~\{-a1~\}~else~\{-a2~\}}{Q}
2729   \equiv
2730   (b \land \liWpKalkuelOhneMathe{a1}{Q})
2731   \lor
2732   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
2733 }

2734 \ExplSyntaxOff

2735

```

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 2502, 2504, 2505,	\bowtie 2247, 2250, 2251, 2252
\, 331, 386,	2508, 2510, 2511,	\Box 159
1120, 1122, 1623,	2512, 2513, 2514,	\boxtimes 463
1624, 1625, 1932, 2402	2518, 2524, 2525,	
\@Skip@Erklaerung@Reset	2530, 2531, 2532,	
... 2712, 2714, 2725	2535, 2543, 2544, 2545	C
\@afterheading 1590	\arabic 1183, 2121, 2126,	\c 1266, 1267
\@afterindentfalse . 1590	2131, 2137, 2143, 2149	\cdot 1667, 1720, 1731
\@liDirLeft 2607, 2612, 2624	\arraystretch 1849	\centerline 1297, 2200, 2222, 2237
\@liDirRight 2608, 2610,		\chapter 1173, 1174
2611, 2612, 2613, 2624	B	\char 1497
\@liDistance 2618, 2619, 2623	\BeforeBeginEnvironment	\clearpage 1491
\@liPos .. 2615, 2616, 2623 2394	\cline 596
\\ 596, 615, 616,	\begin 613,	\clist 226, 270,
619, 620, 623, 624,	658, 673, 701, 725,	271, 284, 288, 2334
703, 704, 705, 789,	763, 786, 794, 806,	\columnbreak 2369
808, 810, 832, 841,	816, 828, 848, 880,	\cs 287, 306, 330,
886, 928, 929, 930,	897, 926, 951, 972,	331, 368, 380, 1573
935, 936, 937, 957,	987, 1099, 1103,	\csname 1251, 1254
1497, 1854, 1911, 1914	1189, 1264, 1292,	\cup 1078, 2511, 2525, 2532
\{ 207, 1055, 1065,	1301, 1308, 1416,	
1077, 1078, 1083,	1489, 1541, 1546,	D
1097, 1119, 1338,	1554, 1579, 1584,	\DeclareMathSymbol ..
1899, 1922, 2277, 2728	1595, 1605, 1609, 1797, 1798
\} 207, 1055, 1065,	1692, 1696, 1712,	\DecoINERT 2124, 2197, 2198, 2220
1077, 1078, 1085,	1735, 1758, 1773,	\DecoINERTwithPivot .
1103, 1123, 1339,	1852, 1853, 1903, 2140, 2219
1899, 1926, 2277, 2728	2041, 2061, 2202,	\DecoLEFT 2119, 2218
_ 27, 35, 44, 46, 287, 306,	2224, 2238, 2394, 2405	\DecoLEFTwithPivot ..
330, 331, 345, 346,	\begingroup 1490, 1901, 2322 2134, 2196
352, 355, 358, 368, 380	\beschriftung 1529, 1533, 1537, 1539	\DecoRIGHT 2129, 2221, 2231
		\DecoRIGHTwithPivot .
A	\beta 2502, 2146, 2199
\addbibresource 1467, 1468, 1469,	2505, 2509, 2510,	\definecolor 1171
1470, 1471, 1472,	2511, 2514, 2524,	\delta 65, 107, 165, 207, 1069
1473, 1474, 1475, 1476	2525, 2526, 2531, 2532	\dh 1625, 2513
\advance 2326	\bf 2099, 2100, 2101	\directlua 58, 137, 195, 200,
\AfterEndEnvironment 2395	\bfseries .. 475, 1173,	1049, 1064, 1084,
\allsectionsfont ... 1169	1175, 2099, 2105,	1092, 1100, 1106,
\Alph 1183	2107, 2109, 2110, 2479	1884, 1889, 1917,
\alph 1183, 1184	\bigskip 48, 364,	1924, 1929, 2289,
	598, 603, 1499, 1827	
	\bool 309, 332	

2381, 2413, 2418, 2419, 2423, 2428, 2429, 2433, 2439, 2440	\do 2120, 2125, 2130, 2135, 2141, 2147	\erzeuge@tiefgestellt . . . 1064, 1065, 1069	\forestov . 2341, 2345, 2346, 2349, 2350, 2351, 2352, 2354, 2355
\dots 506, 510, 1338, 2048, 2524, 2525	\DOWNarrow 2611	\expandafter 1251, 2155, 2157, 2158, 2159, 2167, 2325	\forestset 2337, 2360
\draw 1271, 1274, 1277, 1996, 2295, 2298		\ExplSyntaxOff 50, 92, 134, 139, 192, 197, 202, 393, 528, 550, 565, 1043, 1125, 1160, 1436, 1451, 1615, 1690, 1877, 2024, 2274, 2335, 2453, 2560, 2584, 2663, 2734	\forestSortLevel 2339, 2347, 2361, 2362
E		\ExplSyntaxOn 22, 61, 102, 135, 160, 193, 198, 223, 470, 534, 551, 1016, 1115, 1137, 1429, 1441, 1516, 1629, 1859, 2005, 2262, 2332, 2380, 2476, 2570, 2635, 2688	\frac 1669, 1700, 1731, 1746
\edef 1262, 2213, 2217, 2229, 2230	\else 570, 578, 586, 1059, 1073, 1111, 1537, 1640, 1650, 1660, 1674, 1895, 1991, 2159, 2352, 2354, 2695		\fullouterjoin 2252
\emph 1019, 1319, 1348, 1350, 1496	\empty 1537		G
\emptyset 1841, 2519, 2540	\end 652, 669, 694, 722, 756, 779, 791, 799, 813, 821, 844, 873, 894, 919, 944, 964, 984, 1002, 1101, 1103, 1191, 1280, 1304, 1305, 1312, 1421, 1494, 1543, 1551, 1570, 1586, 1587, 1601, 1612, 1613, 1694, 1707, 1732, 1755, 1770, 1780, 1856, 1857, 1905, 2051, 2071, 2202, 2224, 2238, 2395, 2409		\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
\endcsname 1251, 1254	\endgroup 1493, 1906, 2327	F	\Gamma . 106, 164, 207, 1078
\enspace 2577, 2579	environments:	\faCheckSquareO 2572, 2573	\gappto 2357
liAdditum 1544	liAHuelle 1900	\faCircleThin 1035	\geq 1345, 2037, 2042, 2058, 2062
liAntwort 1515	liDiagramm 1603	\faGg 1029	H
liEinbettung 1514	liExkurs 1552	\fancyfoot 1195, 1196, 1197, 1445, 1446, 1447, 1448	\hbox 2247
liGraphenFormat . 1243	liKasten 1188	\fancyhead 1194, 1442, 1443, 1444	\headrulewidth . 1199, 1449
liKontrollflussgraph 1415	liLernkartei 1593	\faSquareO 1023	\hfill 1716, 1722, 1727, 2580
liProduktionsRegeln 1093	liProjektSprache 1513	\fi 572, 580, 588, 1061, 1075, 1113, 1540, 1642, 1652, 1662, 1676, 1897, 1993, 2160, 2352, 2353, 2356, 2358, 2697	\hinweis 1512
liQuellen 1573	liRelationenSchemaFormat 2281	\fontspec 1169	\hline 1854
liRmodell 2261	liUebergangsTabelle 1849	\footcite 670, 692, 732, 755, 778, 845, 872, 918, 1317, 1320, 1327, 1332, 1337, 1341, 1347, 1352, 1456, 1708, 1709, 1864, 2054	\href 370, 1133, 1621, 1784, 2408
\equiv 2710, 2729		\footnote 1617, 1621	\hspace . . 1999, 2702, 2710
		\footnotesize 147, 341, 424, 520, 1040, 1512, 1565, 1580, 1596, 1808, 1902, 2009, 2018, 2261, 2400, 2552, 2598, 2624	\ht 2248
		\footrulewidth . 1200, 1450	I
		\foreach . 1266, 1269, 1276	\i 1276, 1277
		\forestFirst . . 2349, 2352	\ifcase 2156
		\forestLast . . . 2350, 2352	\ifmode 568, 576, 584, 1057, 1071, 1109, 1638, 1648, 1658, 1672, 1893, 1989, 2693
		\forestOget . . . 2349, 2350	\IfNoValueTF 1547, 1617, 1621
		\forestOnes 2362	\ifnum 2083, 2324, 2340, 2348, 2354
		\forestOv 2351, 2352, 2355	\ifx 1537, 2352
			\in 494, 600, 1345, 1682, 1685, 1688, 1714, 1720, 1725, 2037, 2048, 2058, 2068, 2502, 2504, 2510, 2531, 2651
			\inhaltsverzeichnis 1488
			\input . 4, 7, 10, 13, 16, 394
			\inputminted 2412, 2422, 2432, 2444, 2447, 2451
			\int 2334
			\item 463, 464, 675, 679, 684, 689, 726, 735, 740, 748,

795, 817, 849, 854, 861, 869, 898, 903, 907, 912, 988, 993, 998, 1302, 1303, 1573, 1577, 1697, 1700, 1704, 1713, 1719, 1724, 1736, 1740, 1744, 1748, 1752, 1759, 1763, 1767, 2042, 2045, 2048, 2062, 2065, 2068	\itshape 519, 2551	\LehramtInformatikAutorEmail\liAttributHuelleOhneMathe 1448 1891, 1894, 1896, 1910
		\LehramtInformatikAutorName \liAttributMenge 1447 . . . 1899, 1910, 1913
		\LehramtInformatikGitBranch \liAufgabe 3 373, 2388 \liAufgabenTitel 23
		\LehramtInformatikGithubCodeRepositoryAusdruck 1116 2387 \liAutomat 61
		\LehramtInformatikGithubDomaine\liAutomatenKante 93 2384 \liBandAlphabet 1078
		\LehramtInformatikGithubRawDomaineBedingung 1424 371, 2385 \liBedingungDrei
		\LehramtInformatikGithubTexRepo . . . 1687, 1727, 1767 372, 2386 \liBedingungEins
		\LehramtInformatikRepository . . . 1681, 1716, 1759 4, \liBedingungFalsch . 1426
		7, 10, 13, 16, 1467, \liBedingungWahr . . 1425
		1468, 1469, 1470, \liBedingungZwei 1684, 1722, 1763
		1471, 1472, 1473, \liBeschriftung . . . 1504
		1474, 1475, 1476, 2383 \liChomskyErklaerung 1443 485, 526
		\leq . . . 1731, 2045, 2065 \liChomskyUeberErklaerung 1067, 524
		\let 1067, \liChomskyUeberschrift 1068, 1491, 2196, 473, 525
		2197, 2198, 2199, \liCpmEreignis 534
		2212, 2214, 2215, \liCpmFruehesterI . . 591
		2216, 2218, 2219, \liCpmSpaetesterI . . 590
		2220, 2221, 2231, \liCpmVon 574
		2323, 2361, 2362, 2572 \liCpmVonOhneMathe 485, 521 574, 577, 579
		\li@chomsky@erklaerung@texte \liCpmVonZu 566
		\li@fussnote@text 1805, \liCpmVonZuOhneMathe 1811, 1815, 1819, 1823 566, 569, 571
		\li@GithubLink 2404, 2417, 2427, 2438 \liCpmVorgang 551
		\li@mget . 1253, 1257, 1277 \liCpmZu 582
		\li@minc 1256, 1278 \liCpmZuOhneMathe 582, 585, 587
		\li@mset 1250, 1258, 1267, 1270 liDiagramm (environ- 1262, 1271, 1277 ment) 1603
		\li@numdiscs 1262, 1271, 1277 liEinbettung (environ- 2261, 2270, 2280 ment) 1514
		\li@Rmodell@Schrift 2261, 2270, 2280 \liEntwurfs 967
		\li@sequence . . 1263, 1276 \liEntwurfsAbstrakteFabrik 2493, 2553 654
		\li@synthese@erklaerung@texte \liEntwurfsAbstrakteFabrikUml 2493, 2553 612, 655
		\liAbleitung 1092 \liEntwurfsAdapter . . 696
		liAdditum (environment) 1544 \liEntwurfsAdapterAkteure 1544 672, 698
		liAHuelle (environment) 1900 \liEntwurfsAdapterUml 1900 657, 697
		\liAlphabet 1077 \liEntwurfsBeobachter 758
		liAntwort (environment) 1515 \liEntwurfsBeobachterAkteure 1423 724, 760
		\liAnweisung 1423 \liEntwurfsBeobachterUml 2442 700, 759
		\liAssemblerCode . . 2442 \liEntwurfsDekorierer 781
		\liAssemblerDatei . . 2443 \liEntwurfsDekoriererAkteure 2280 783
		\liAttribut 1891, 2505, 2510 \liEntwurfsDekoriererUml 1891, 2505, 2510 762, 782
		\liAttributHuelle 1908
		\liAttributHuelleLinksReduktion

\liEntwurfsEinfacheFabrik	\liFussnoteDreiText	171, 172, 173, 177, 1055, 1106, 1145, 1146
..... 801 1818, 1836	
\liEntwurfsEinfacheFabrikAkteure	\liFussnoteEinsText	\liMengeOhneMathe ...
..... 793, 803 1810, 1830	... 1055, 1058, 1060
\liEntwurfsEinfacheFabrikUml	\liFussnoteLink	\liMinimierungErklaerung
..... 785, 802 1620 1863
\liEntwurfsEinzelstueck	\liFussnoten	\liMinispracheDatei 2446
..... 823	\liFussnoteUrl . 965, 1616	\linespread 2268
\liEntwurfsEinzelstueckAkteure	\liFussnoteVierText	\liNichtsZuTun 2573
..... 815, 825 1822, 1839	\liO 1654, 1682
\liEntwurfsEinzelstueckUml	\liFussnoteZweiText	\liOmega 1644, 1688
..... 805, 824 1814, 1833	\liOmegaOhneMathe ...
\liEntwurfsErbauer .. 875	\liGrammatik 1137	... 1644, 1649, 1651
\liEntwurfsErbauerAkteure	liGraphenFormat (envi- ronment) 1243	\liOOhneMathe
..... 847, 877	\liHanoi 1250	... 1654, 1659, 1661
\liEntwurfsErbauerUml	\liHaskellCode 2449	\liParagraphMitLinien
..... 827, 876	\liHaskellDatei 2450	. 521, 1865, 2553,
\liEntwurfsFabrikmethode	\liInduktionAnfang . 2638	2574, 2642, 2650, 2658
..... 921, 946	\liInduktionErklaerung	\liPetriErreichKnotenDrei
\liEntwurfsFabrikmethodeAkteure 2637 1998
..... 896, 923	\liInduktionMarkierung	\liPetriErreichTransition
\liEntwurfsFabrikmethodeUml 2636 1995
..... 879, 922	\liInduktionSchritt 2654	\liPetriSetzeSchluessel
\liEntwurfsKompositum 946	\liInduktionVoraussetzung 1940
\liEntwurfsKompositumAkteure 2646	\liPetriTransitionsName
..... 948	\liJavaCode 2402 1987, 1999
\liEntwurfsKompositumUml	\liJavaDatei 2411	\liPetriTransitionsNameOhneMathe
..... 925, 947	\liJavaExamen 2431	... 1987, 1990, 1992
\liEntwurfsModellPraesentationSteuerung	\liJavaTestDatei ... 2421	\liPetriTransPfeile 1999
..... 967	liKasten (environment) 1188	\liPolynomiellReduzierbar
\liEntwurfsModellPraesentationSteuerungAkteure	\liKellerAutomat 102 1307
..... 969	\liKellerKante 140	\liPotenzmenge
\liEntwurfsModellPraesentationSteuerungUml	\liKellerUebergang 1064, 1068, 2010
..... 950, 968 135, 141	\liPotenzmengeOhneMathe
\liEntwurfsZustand . 1004	\liKontrollCode 1427	... 1065, 1066, 1067
\liEntwurfsZustandAkteure	liKontrollflussgraph	\liPrimaer 2259
..... 986, 1006	(environment) 1415	\liProblemBeschreibung
\liEntwurfsZustandUml	\liKontrollKnotenPfad 1291
..... 971, 1005 1429	\liProblemClique ... 1314
\liEpsilon 1063	\liKontrollTextzeileKnoten	\liProblemName
\liErAttribute 1428, 1433 1290, 1297,
... 1019, 1033, 1035	\liKurzeTabellenLinie 596	1309, 1311, 1324,
\liErDatenbankName . 1038	\liLadeAllePakete ... 228	1335, 1336, 1344, 1345
\liErEntity 1017, 1021, 1023	\liLadePakete	\liProblemSat 1343
\liErledigt 2572	. 54, 57, 224, 229,	\liProblemSubsetSum .
\liErMpAttribute ... 1032	472, 533, 1287, 1334, 1343
\liErMpEntity 1020	1803, 1883, 2004, 2475	\liProblemVertexCover
\liErMpRelationship 1026	\liLatexCode 2403 1314, 1322
\liErRelationship ...	\liLeereZelle 1841	\liProduktionen 1105, 1147
... 1018, 1027, 1029	liLernkartei (environ- ment) 1593	liProduktionsRegeln
\liExamensAufgabe 6	\liMasterExkurs 1772	(environment) 1093
\liExamensAufgabeA ... 15	\liMasterFaelle 1711, 1779	liProjektSprache (envi- ronment) 1513
\liExamensAufgabeTA .. 12	\liMasterFallRechnung	\liPseudoUeberschrift
\liExamensAufgabeTTA .. 9 1757 1498,
liExkurs (environment) 1552	\liMasterVariablen ..	1548, 1549, 1851,
\liFalsch 464 1691, 1774	1861, 2639, 2647, 2655
\liFlaci 1126	\liMasterVariablenDeklaration	\liPumpingKontextfrei
\liFremd 2260 1734 2056
\liFunktionaleAbhaengigkeit	\liMasterWolframLink 1782	\liPumpingRegulaer . 2035
..... 1916	\liMenge ... 71, 72, 74,	liQuellen (environment)
\liFunktionaleAbhaengigkeiten	113, 114, 115, 119, 1573
..... 1919		
\liFussnote ... 1804, 1806		

<code>\liRekursionsGleichung</code>	<code>\liWpKalkuelOhneMathe</code>	N
..... 1678, 1738 2689, 2694,	<code>\NeedsTeXFormat</code>
<code>\liRelation</code> 1928	2696, 2728, 2730, 2732	1, 19, 52, 220, 398,
<code>liRelationenSchemaFormat</code>	<code>\liZustandsBuchstabe</code>	459, 466, 530, 593,
(environment) 2281 1079,	608, 1009, 1045,
<code>\liRelationMenge</code> ... 2275	1088, 1090, 1110, 1112	1162, 1202, 1211,
<code>\liRichtig</code> 463	<code>\liZustandsBuchstabeGross</code>	1216, 1245, 1283,
<code>liRmodell</code> (environment)	... 1080, 1089, 1091	1355, 1438, 1453,
..... 2261	<code>\liZustandsmenge</code> ... 1067	1459, 1481, 1627,
<code>\liRundeKlammer</code> . 1631,	<code>\liZustandsmengeNr</code> ..	1787, 1800, 1879,
1635, 1645, 1655, 1669 1081, 2019	1935, 2001, 2026,
<code>\liSetzeAufgabenTitel</code> . 25	<code>\liZustandsmengeNrGross</code>	2031, 2075, 2241,
<code>\liSortierMarkierung</code> 2300 1089	2254, 2283, 2364,
<code>\liSortierPfeil</code> 2294	<code>\liZustandsMengenSammlung</code>	2371, 2376, 2456,
<code>\liSortierPfeilUnten</code> 2297 2006	2471, 2562, 2566,
<code>\liSpaltenUmbruch</code> .. 2369	<code>\liZustandsMengenSammlungNr</code>	2586, 2628, 2665, 2685
<code>\liSqlCode</code> 2454 2015	<code>\neg</code> 2732
<code>\listen@punkt</code> .. 1573, 1585	<code>\liZustandsmengeOhneMathe</code>	<code>\negthinspace</code> 1632
<code>\liStrich</code> 1289 1067	<code>\newcounter</code> ... 2095, 2096
<code>\liSyntheseErklaerung</code>	<code>\liZustandsname</code> 1090	<code>\NewDocumentCommand</code> .
..... 2493, 2558	<code>\liZustandsnameGross</code>	62, 103, 140, 161,
<code>\liSyntheseUeberErklaerung</code>	... 1091, 2007, 2016	203, 224, 535, 552,
..... 2556	<code>\liZustandsnameTiefgestellt</code>	597, 602, 1116,
<code>\liSyntheseUeberschrift</code> 1108	1138, 1307, 1430,
..... 2477, 2557	<code>\liZustandsPaar</code> 1843	1457, 1616, 1620,
<code>\liT</code> 1664, 1679, 1693, 1754	<code>\liZustandsPaarVariablenName</code>	1919, 1928, 1995,
<code>\liTeilen</code> 1888	... 1842, 1845, 1846	2411, 2421, 2431,
<code>\literatur</code> 1455, 1479	<code>\llap</code> 2248	2443, 2446, 2450, 2606
<code>\liTheta</code> 1634,	<code>\log</code> 1682,	<code>\NewDocumentEnvironment</code>
1685, 1714, 1720, 1725	1685, 1688, 1714, 1720	1093, 1188, 1243,
<code>\liThetaOhneMathe</code> ...	<code>\loop</code> 2081	1415, 1513, 1514,
... 1634, 1639, 1641	<code>\lor</code> 2731	1517, 1544, 1552,
<code>\liTOhneMathe</code>	<code>\ltimes</code> 1790	1574, 1593, 1603,
... 1664, 1673, 1675	M	1850, 1900, 2263, 2281
<code>\liTuringKante</code> 203	<code>\makeatletter</code> .. 1589, 2320	<code>\newlength</code> 2712
<code>\liTuringLeerzeichen</code>	<code>\makeatother</code> .. 1591, 2328	<code>\node</code> 548, 1423,
..... 159, 167	<code>\marginpar</code>	1428, 2121, 2126,
<code>\liTuringMaschine</code> ... 160	1022, 1028, 1034, 1496	2131, 2137, 2143,
<code>\liTuringUeberfuehrung</code>	<code>\mathbb</code> .. 1345, 2068, 2651	2149, 2300, 2345, 2623
..... 206	<code>\mathbin</code> . 2250, 2251, 2252	<code>\noexpand</code> 2209,
<code>\liTuringUebergaeenge</code>	<code>\mathcal</code> .. 1655, 2531,	2210, 2211, 2230, 2345
..... 198, 204	2536, 2538, 2539, 2540	<code>\noindent</code> 343, 599, 604,
<code>\liTuringUebergangZelle</code>	<code>\Mathe</code> 2706	1096, 1500, 1502,
..... 193	<code>\MatheEnv</code> 2699, 2707, 2710	1506, 1510, 1538,
<code>\liUeberfuehrungsFunktion</code>	<code>\mathord</code> 1797, 1798	1566, 1568, 1581,
..... 1069	<code>\mdfsetup</code> 1187,	1597, 1599, 1607,
<code>\liUeberfuehrungsFunktionOhneMathe</code>	1522, 1526, 1530, 1534	1776, 1829, 1832,
... 1069, 1072, 1074	<code>\medskip</code>	1835, 1838, 1921,
<code>liUebergangsTabelle</code>	1299, 1507, 1509,	1926, 2276, 2575, 2718
(environment) 1849	1583, 1611, 2266,	<code>\nolinkurl</code> 2408
<code>\liUeberschriftDreiecksTabelle</code>	2272, 2582, 2700, 2704	<code>\normalsize</code> 1175
..... 1859	<code>\memph</code> 1496	<code>\notin</code> 605
<code>\liUmlLeserichtung</code> . 2606	<code>\mintinline</code> 2402,	<code>\null</code> 2580
<code>\liVertauschen</code> 2288	2403, 2442, 2449, 2454	O
<code>\liWortInSprache</code> 597	<code>\mkern</code> ... 2250, 2251, 2252	<code>\o@join</code>
<code>\liWortNichtInSprache</code> 602	<code>\mlq</code> 1795, 1797	2247, 2250, 2251, 2252
<code>\liWpEquivalent</code> 2709	<code>\mrq</code> 1795, 1798	<code>\Omega</code> 1645
<code>\liWpErklaerung</code> 2712	<code>\msg</code> 39, 391	<code>\omega</code> 2037, 2038, 2058, 2059
<code>\liWpErklaerungVerzweigung</code>	<code>\myList</code>	<code>\or</code> 2158
..... 2727	2341, 2342, 2343, 2346	P
<code>\liWpKalkuel</code> 2689	<code>\myNodes</code> 2330,	<code>\pagestyle</code> 1198
	2345, 2351, 2355, 2357	

<code>\par</code> 342, 363, 525, 1127, 1135, 1505, 1567, 1590, 1598, 1920, 1923, 1925, 2200, 2222, 2237, 2278, 2557, 2581, 2717, 2723	<code>\QsInitialize</code> 2080, 2192, 2227	<code>\romannumeral</code> 2167
<code>\paragraph</code> 1175	<code>\QSIr</code> . 2163, 2169, 2177, 2197, 2211, 2216, 2219	<code>\rtimes</code> 1790
<code>\parindent</code> 2269	<code>\QSIrr</code> 2198, 2211, 2212, 2220	<code>\rule</code> 2200, 2222, 2237, 2248
<code>\path</code> 94, 141, 204, 563	<code>\QSLr</code> 2169, 2176, 2187, 2188, 2196, 2209, 2214, 2218	S
<code>\pgfkeys</code> .. 1951, 2609, 2610, 2611, 2612, 2613, 2616, 2619, 2621	<code>\QSpivotStep</code> 2082, 2192, 2196, 2207	<code>\sb</code> 67, 77, 79, 108, 166, 501, 502, 506, 509, 510, 511, 1110, 1112, 1682, 1685, 1688, 1714, 1720, 1869, 1987, 1996, 2524, 2525, 2526, 2531, 2535, 2536, 2539, 2540, 2543, 2544, 2545
<code>\pgfmath@count</code> 2322, 2324, 2326	<code>\QSR</code> 2169	<code>\scriptscriptstyle</code> 566, 574, 582
<code>\pgfmath@smuggleone</code> 2327	<code>\QSRr</code> 2178, 2199, 2210, 2215, 2221, 2230, 2231, 2232	<code>\scriptsize</code> 1129, 1368, 1375, 1381, 1443, 1444, 1447, 1448, 2637, 2690, 2720
<code>\pgfmathdeclarefunction</code> 2321	<code>\QSSortStep</code> 2084, 2192, 2208, 2209	<code>\section</code> 46
<code>\pgfmathhint</code> 2322	<code>\quad</code> 1807	<code>\seq</code> . 1432, 1433, 1434, 1576, 1577, 1578, 1585
<code>\pgfmathparse</code> 1257, 2339, 2344, 2347, 2361, 2362	R	<code>\setbox</code> 2247
<code>\pgfmathresult</code> 1258, 2322, 2323, 2325, 2327, 2340, 2348, 2361, 2362	<code>\raisebox</code> 1428	<code>\setcounter</code> 1176, 2201, 2223, 2237
<code>\pgfutil@empty</code> 2323	<code>\relax</code> 1491, 2169, 2212, 2214, 2215, 2216, 2324, 2326	<code>\setganttlinklabel</code> .. 1206, 1207, 1208, 1209
<code>\pgfutil@loop</code> 2324	<code>\renewcommand</code> ... 1178, 1179, 1180, 1181, 1184, 1185, 1199, 1200, 1449, 1450, 1849	<code>\setlength</code> 2269, 2714, 2715, 2725
<code>\pgfutil@repeat</code> 2327	<code>\repeat</code> 2085	<code>\setmainfont</code> 1166
<code>\preceq</code> 1310	<code>\RequirePackage</code> 55, 158, 222, 226, 395, 401, 402, 462, 532, 611, 1012, 1014, 1015, 1053, 1054, 1164, 1165, 1168, 1170, 1172, 1177, 1186, 1193, 1204, 1205, 1214, 1218, 1219, 1220, 1248, 1249, 1288, 1357, 1456, 1461, 1462, 1478, 1485, 1486, 1487, 1515, 1630, 1791, 1792, 1937, 1939, 2029, 2089, 2091, 2243, 2244, 2245, 2258, 2286, 2317, 2368, 2374, 2379, 2390, 2391, 2459, 2564, 2571, 2590, 2591, 2593, 2594, 2632, 2633, 2634, 2667, 2687	<code>\setmainlanguage</code> 396
<code>\prime</code> 1289	<code>\right</code> 1632	<code>\setminted</code> 2396, 2397
<code>\printbibliography</code> . 1479	<code>\RIGHTarrow</code> ... 2608, 2613	<code>\setsansfont</code> 1167
<code>\ProvidesPackage</code> 2, 20, 53, 221, 399, 460, 467, 531, 594, 609, 1010, 1046, 1163, 1203, 1212, 1217, 1246, 1284, 1356, 1439, 1454, 1460, 1482, 1628, 1788, 1801, 1880, 1936, 2002, 2027, 2032, 2076, 2242, 2255, 2284, 2365, 2372, 2377, 2457, 2472, 2563, 2567, 2587, 2629, 2666, 2686	<code>\rightarrow</code> 207, 489, 494, 502, 506, 508, 509, 511, 566, 574, 1999, 2502, 2509, 2511, 2514, 2519, 2524, 2525, 2530	<code>\setul</code> 2260
Q	<code>\Roman</code> 1183	<code>\sffamily</code> 476, 1173, 1175, 1277, 2480
<code>\QS@list</code> 2202, 2213, 2217, 2224, 2230, 2235, 2238	<code>\romannumeral</code> 1183, 1185	<code>\shoveleft</code> 1909
<code>\QS@select@equal</code> 2177, 2181	<code>\rightouterjoin</code> 2251	<code>\shoveright</code> 1912
<code>\QS@select@greater</code> 2178, 2182	<code>\Roman</code> 1183	<code>\Sigma</code> 64, 105, 163, 1077, 1078, 1140
<code>\QS@select@smaller</code> 2173, 2176, 2180	<code>\romannumeral</code> 1183, 1185	<code>\sigma</code> 499, 501, 502
<code>\QS@sort@a</code> 2155, 2188, 2209, 2210	<code>\Roman</code> 1183	<code>\SLASH</code> 1497
<code>\QS@sort@b</code> 2155, 2156	<code>\romannumeral</code> 1183, 1185	<code>\small</code> 1606
<code>\QS@sort@c</code> 2159, 2166	<code>\romannumeral</code> 1183, 1185	<code>\sort</code> 2334
<code>\QS@sort@d</code> 2167, 2175	<code>\romannumeral</code> 1183, 1185	<code>\sortList</code> 2333, 2342
<code>\QS@sort@empty</code> . 2157, 2162	<code>\romannumeral</code> 1183, 1185	<code>\square</code> 464
<code>\QS@sort@single</code> 2158, 2163	<code>\romannumeral</code> 1183, 1185	<code>\stepcounter</code> 2121, 2126, 2131, 2134, 2136, 2140, 2142, 2146, 2148
		<code>\str</code> .. 477, 486, 1519, 2011, 2020, 2481, 2494
		<code>\StrSubstitute</code> . 2341, 2343
		<code>\strut</code> 1736, 1740, 1744, 1748, 1752, 2369
		<code>\subseteq</code> 2505, 2538, 2545
		T
		<code>\tableofcontents</code> ... 1492
		<code>\text</code> 77, 79, 182, 1891, 2637, 2690
		<code>\textbf</code> ... 1017, 1315, 1324, 1335, 1344,

1501, 1508, 1539,	\TmpPlaceOne	1952	\umlstatic	789, 808
1567, 1582, 1598, 1854	\TmpPlaceSeven	1958	\umluniaggreg	836
\textcolor 1427, 2636	\TmpPlaceSix	1957	\umluniassoc	
\textit	\TmpPlaceTen	1961		649, 665, 837, 961, 962
886, 928, 929, 930,	\TmpPlaceThree	1954	\umlVHuniassoc . .	650, 651
931, 1608, 1899, 1932	\TmpPlaceTwo	1953	\umlVHVdep	
\textsc 1290	\TmpScale	1972		643, 644, 646, 647
\textsf 1501, 1582	\TmpTransitionEight .		\umlVHVinherit . .	626,
\textstyle 1700, 1731		1948, 1969		627, 632, 633, 640,
\texttt 1040, 1290, 1424,	\TmpTransitionFive . .			641, 774, 775, 941, 942
1425, 1426, 1427, 2690		1945, 1966	\umlVHVreal	
\thepage 1197, 1446	\TmpTransitionFour . .			768, 769, 978, 979
\theparagraph 1175		1944, 1965	\UParrow	2610
\Theta 1635	\TmpTransitionNine . .		\url	1617
\thinspace 2690		1949, 1970	\usemintedstyle . . .	2393
\tikz 1428	\TmpTransitionOne . . .		\usetikzlibrary . .	56,
tikz: bbaum 23		1941, 1962		403, 1013, 1221,
tikz: li binaer baum 21	\TmpTransitionSeven .			1358, 1938, 2246,
\tikzchildnode 419		1947, 1968		2287, 2318, 2595, 2683
\tikzparentnode 419	\TmpTransitionSix . . .			
\tikzset 96,		1946, 1967		V
143, 209, 404, 430,	\TmpTransitionTen . . .		\value	2083
1222, 1359, 1977,		1950, 1971	\varepsilon	478,
2103, 2307, 2461, 2668	\TmpTransitionThree .			489, 490, 1063,
\tikzumlset 2596		1943, 1964		1682, 1688, 1717, 1728
\times 207	\TmpTransitionTwo . . .		\vfill	2369
\tiny 1023, 1029,		1942, 1963	\vrule	2576, 2580
1035, 1427, 1496, 2406	\TmpX	1973	\vspace	
\titleformat . . 1173, 1175	\TmpY	1974		1098, 1102, 1553, 1571
\titlespacing 1174	\today	1444		
\tl 29, 37, 63, 64, 65,	\ttfamily	2261		X
66, 67, 68, 71, 72,			\xappto	2345, 2351, 2355
73, 74, 75, 77, 79,	U		\xdef	1251
104, 105, 106, 107,	\ul	1018, 2259, 2260	\xintApply	2171
108, 109, 110, 113,	\umlaggreg	981	\xintApplyUnbraced . .	
114, 115, 116, 117,	\umlassoc	963		2170, 2176, 2177, 2178
118, 119, 162, 163,	\umlclass 614, 618, 622,		\xintCSVtoList	2230
164, 165, 166, 167,		660, 661, 662, 702,	\xintFor	
168, 171, 172, 173,		707, 712, 715, 764,		2120, 2125, 2130,
174, 175, 176, 177,		765, 766, 771, 772,		2135, 2141, 2147, 2188
285, 289, 307, 311,		787, 807, 830, 831,	\xintifEq	2181
312, 313, 316, 321,		834, 885, 888, 927,	\xintifForLast	
322, 323, 334, 335,		933, 934, 952, 953,		2137, 2143, 2149
336, 337, 348, 354,		954, 973, 974, 975, 976	\xintifGt	2182
357, 360, 369, 383,	\umldep	893	\xintifLt	2180
536, 539, 544, 545,	\umlHVHaggreg 720, 777, 943		\xintLength	2155
553, 554, 557, 558,	\umlinherit		\xintntheft	2167
1118, 1139, 1140,		666, 710, 838, 883, 891		
1141, 1142, 1145,	\umlnote	668, 840, 983		Z
1146, 1147, 1148, 1665	\umlreal	664, 718		
\TmpPlaceEight 1959	\umlsimpleclass		\ZB	1624
\TmpPlaceFive 1956		629, 630,	\zB	1623
\TmpPlaceFour 1955		631, 635, 637, 638,	\zustandsnamens@liste	
\TmpPlaceNine 1960		639, 659, 829, 881, 882		1081, 1088, 1089