

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)	37
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)	42
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 }

```

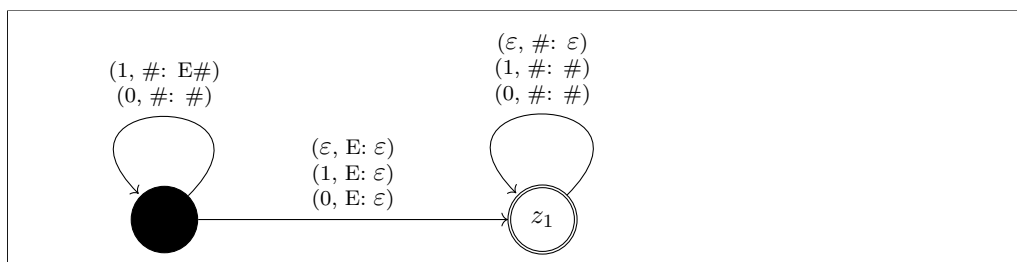
2.3.2 Kellerautomat

```
\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[⟨automaten-name⟩]
{⟨zustaende=Z,alphabet=Σ,kelleralphabet=Γ,delta=δ,start=z0,kellerboden=#,ende=E⟩}

\liKellerAutomat{
  zustaeende={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     zustaeende .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}

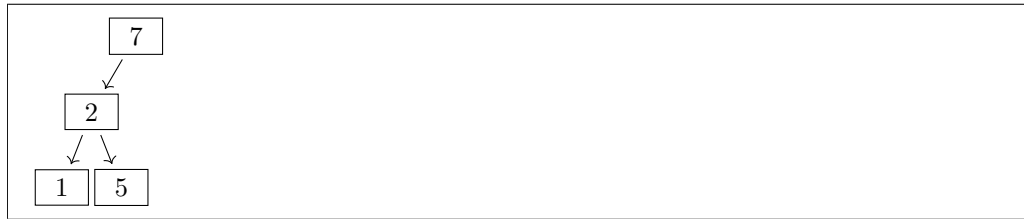
    für li binaer baum
402 \RequirePackage{tikz-qtree}

    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}

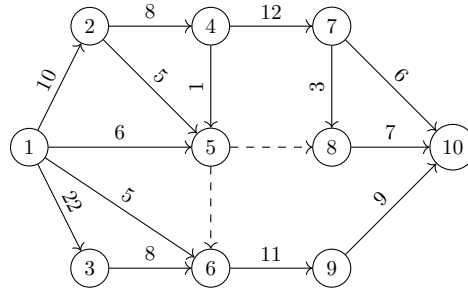
```

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

```

```

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

```

```

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

```

```

534 \ExplSyntaxOn
535 \NewDocumentCommand { \liCpmEreignis } { 0{} m m m } {
536   \tl_set:Nn \l_name_tl {}
537
538   \keys_define:nn { cpmEreignis } {
539     name .code:n = {\tl_set:Nn \l_name_tl {##1}},
540   }
541
542   \keys_set:nn { cpmEreignis } { #1 }
543
544   \tl_if_empty:NT \l_name_tl {
545     \tl_set:Nn \l_name_tl {#2}
546   }

```

```

547
548 \node[circle,draw] (\l_name_tl) at (#3,#4) {#2};
549 }
550 \ExplSyntaxOff

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

```

2.8.1 Hilfsmakros für Zeitpunkt-Tabelle

```

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

```

```

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

```

```

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

```

```

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

```

```

588   \fi%
589 }

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

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

592

```

2.9 cyk-algorithmus.sty

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

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

$R_a$ & $R_c$ & $R_b$ & $R_c$ & $R_a$ & $R_b$ \16
B      & A      & A      & B      & C \15
S      & -      & S      & S \14
-      & -      & - \13
-      & - \12
S \11
\end{tabular}
```

```
\liKurzeTabellenLinie Let-Abkürzung: \let\l=\liKurzeTabellenLinie
596 \def\liKurzeTabellenLinie#1{\ccline{1-#1}}
```

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

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

607
```

2.10 entwurfsmuster.sty

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

2.10.1 Namensschema der Entwurfsmuster-Makros:

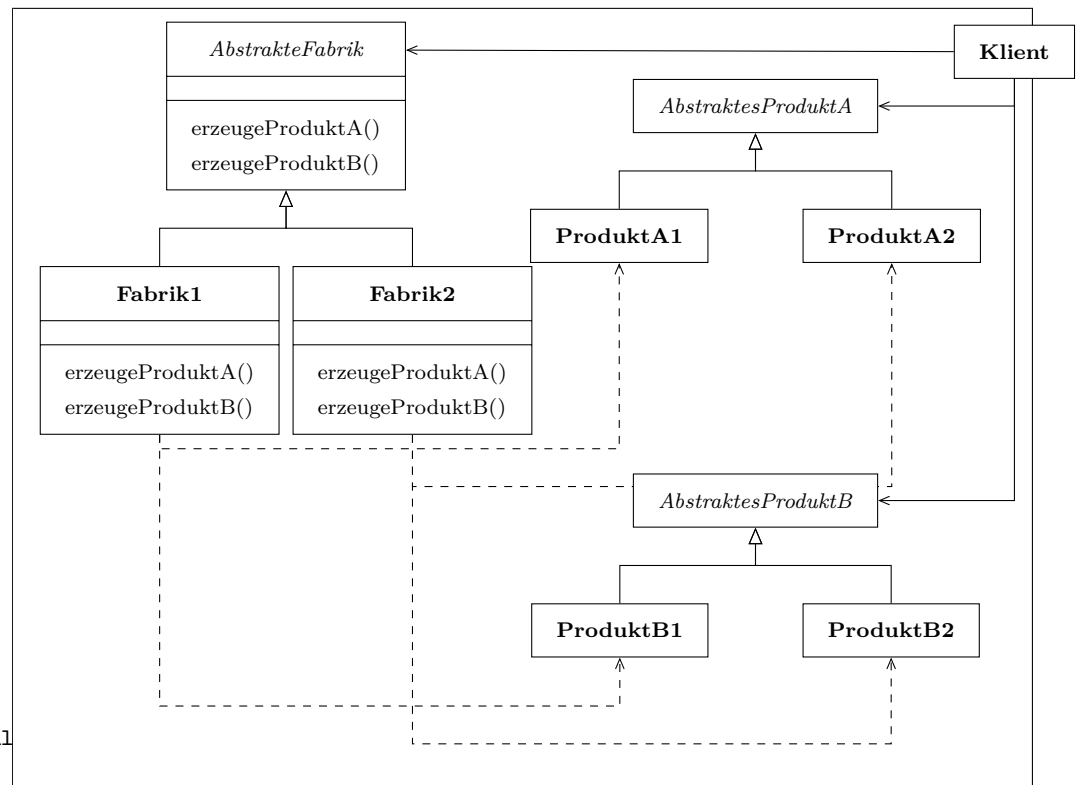
Präfix: \liEntwurfs + Name des Entwurfsmuster DeutscherName + Suffix: (ohne, Uml oder Akteure)

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 }

```

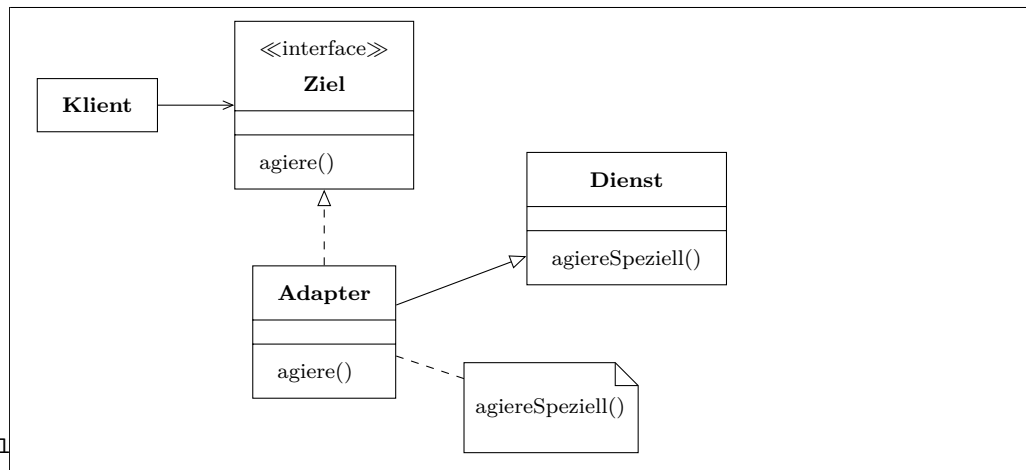
\liEntwurfsAbstrakteFabrik

```

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

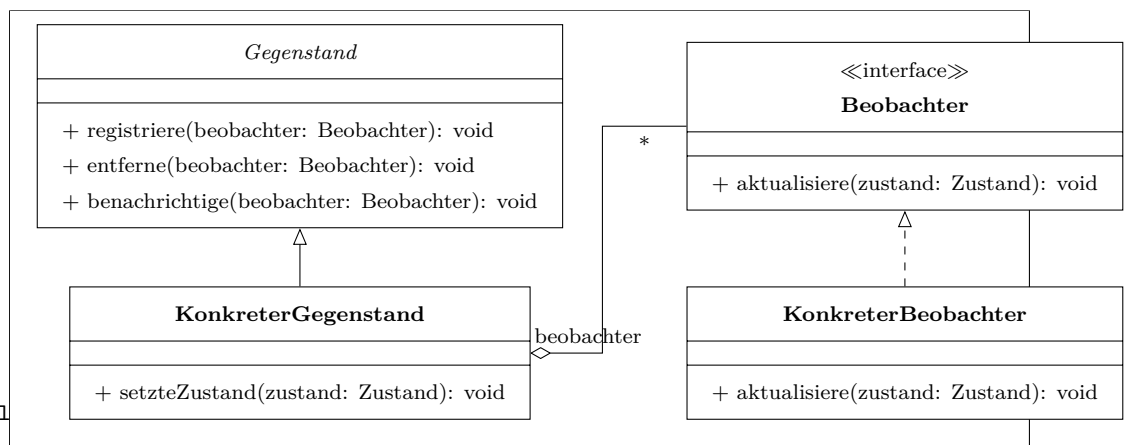
```

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 }

```

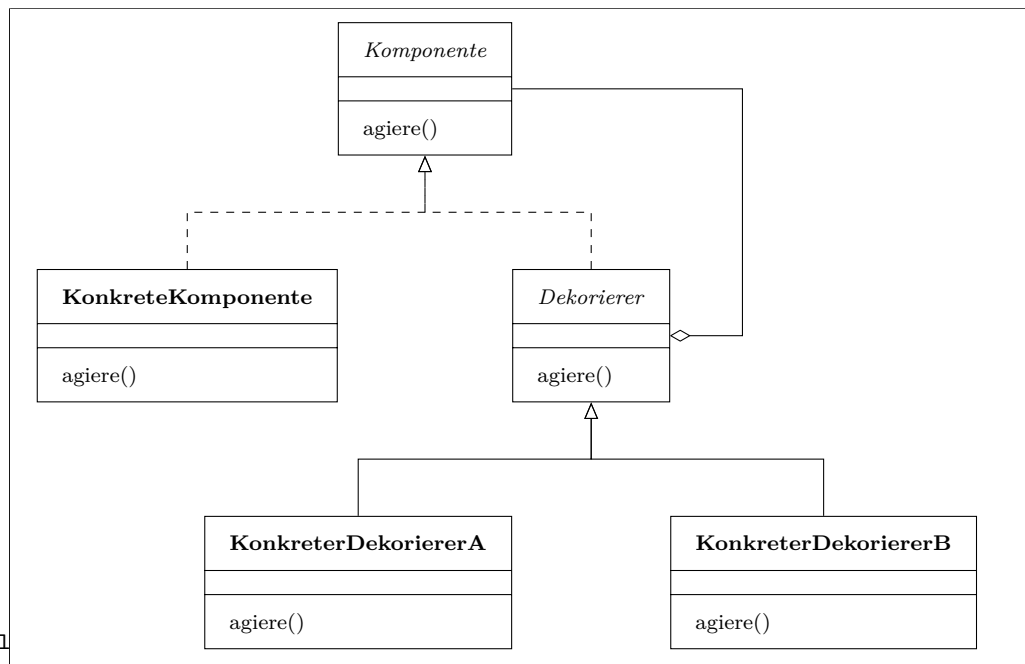
\liEntwurfsBeobachter

```

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 \umlHVHaggreg[arm1=2cm]{Dekorierer}{Komponente}
778 \footcite{wiki:dekorierer}
779 \end{tikzpicture}
780 }

```

\liEntwurfsDekorierer

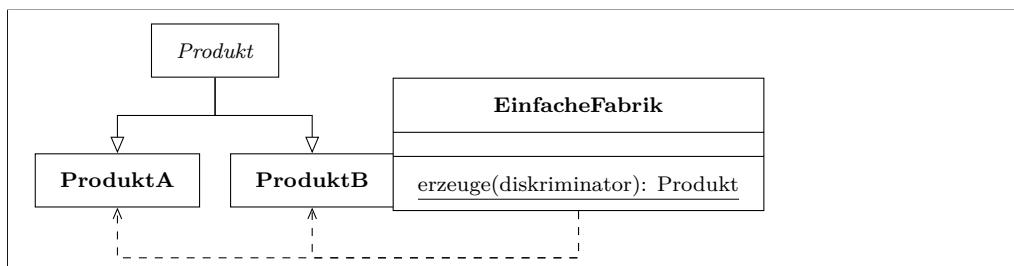
```

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 \umlsimpleclass[type=abstract]{Produkt}
788 \umlsimpleclass[below left=1cm and -0.65cm of Produkt]{ProduktA}
789 \umlsimpleclass[below right=1cm and -0.65cm of Produkt]{ProduktB}
790 \umlVHVinherit{Produkt}{ProduktA}
791 \umlVHVinherit{Produkt}{ProduktB}
792 \umlclass[below right=0cm and 1.5cm of Produkt]{EinfacheFabrik}{
793 }{
794 \umlstatic{erzeuge(diskriminator): Produkt}}\
795 }
796 \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktA}
797 \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktB}
798 \end{tikzpicture}
799 }

```

\liEntwurfsEinfacheFabrikAkteure Quelle: frei überstetzt aus GoF

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

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

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

```

800 \def\liEntwurfsEinfacheFabrikAkteure{
801 \begin{description}
802 \item[EinfacheFabrik]
803
804 Eine Klasse mit einer Erzeugungsmethode, die über eine größere
805 Bedingung verschiedene Objekt instanziiert.
806
807 \item[Produkt]
808
809 Eine abstrakte Klasse, die von den konkreten Produkten geerbt wird.
810
811 \item[KonkretesProdukt]

```

```

812
813     Ein konkretes Produkt, das von der einfachen Fabrik erzeugt wird.
814 \end{description}
815 }

```

\liEntwurfsEinfacheFabrik

```

816 \def\liEntwurfsEinfacheFabrik{
817     \liEntwurfsEinfacheFabrikUml
818     \liEntwurfsEinfacheFabrikAkteure
819 }

```

2.10.8 Einzelstück (Singleton)

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia

Einzelstück
- instanz: Einzelstück
- Einzelstück() + gibInstanz(): Einzelstück

```

820 \def\liEntwurfsEinzelstueckUml{
821     \begin{tikzpicture}
822         \umlclass{Einzelstück}{
823             \umlstatic{- instanz: Einzelstück}\\
824         }{
825             - Einzelstück()\\
826             + gibInstanz(): Einzelstück
827         }
828     \end{tikzpicture}
829 }

```

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.

```

830 \def\liEntwurfsEinzelstueckAkteure{
831     \begin{description}
832         \item[Einzelstück (Singleton)]
833
834         stellt eine statische Methode bereit, mit deren Hilfe die Klienten
835         nur auf eine einzige Instanz der Klasse zugreifen können.
836     \end{description}
837 }

```

\liEntwurfsEinzelstueck

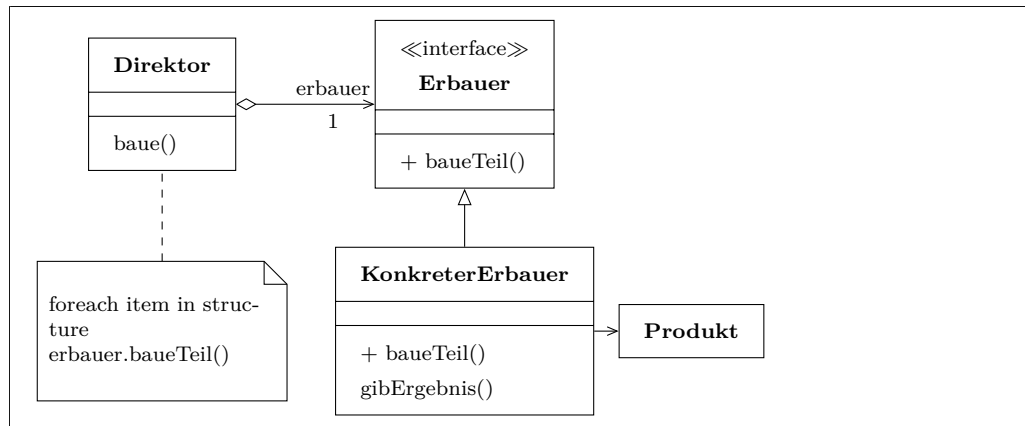
```

838 \def\liEntwurfsEinzelstueck{
839     \liEntwurfsEinzelstueckUml
840     \liEntwurfsEinzelstueckAkteure
841 }

```

2.10.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

842 \def\liEntwurfsErbauerUml{
843   \begin{tikzpicture}
844     \umlsimpleclass[x=7,y=0]{Produkt}{}{}
845     \umlclass[x=4,y=3,type=interface]{Erbauer}{}{+ baueTeil()}
846     \umlclass[x=4,y=0]{KonkreterErbauer}{}{
847       + baueTeil()\
848       gibErgebnis()}
849     \umlclass[x=0,y=3]{Direktor}{}{baue()}
850
851     \umluniagg[arg2=erbauer,pos2=0.7,mult2=1]{Direktor}{Erbauer}
852     \umluniassoc{KonkreterErbauer}{Produkt}
853     \umlinherit{KonkreterErbauer}{Erbauer}
854
855     \umlnote[x=0,y=0,width=3cm]{Direktor}{
856       foreach item in structure\
857       erbauer.baueTeil()
858     }
859   \end{tikzpicture}
860   \footcite{wiki:erbauer}
861 }

```

\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.

```

862 \def\liEntwurfsErbauerAkteure{
863   \begin{description}
864     \item[Erbauer]
865
866     Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der
867     Teile eines komplexen Objektes.
868
869     \item[KonkreterErbauer]
870
871     Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch
872     Implementierung der Schnittstelle. Außerdem definiert und verwaltet er

```

```

873     die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine
874     Schnittstelle zum Auslesen des Produkts.
875
876     \item[Direktor]
877
878     Der Direktor konstruiert ein komplexes Objekt unter Verwendung der
879     Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer
880     zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder
881     benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom
882     Klienten.
883
884     \item[Produkt]
885
886     Das Produkt repräsentiert das zu konstruierende komplexe Objekt.
887     \footcite{wiki:erbauer}
888 \end{description}
889 }

```

\liEntwurfsErbauer

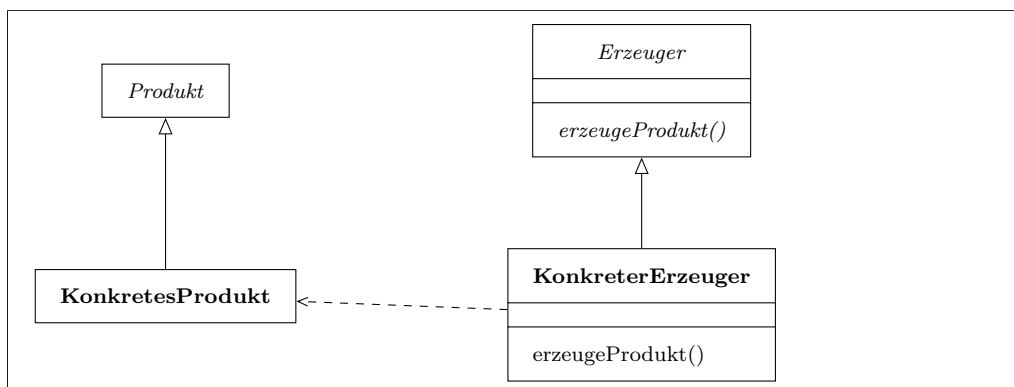
```

890 \def\liEntwurfsErbauer{
891   \liEntwurfsErbauerUml
892   \liEntwurfsErbauerAkteure
893 }

```

2.10.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

894 \def\liEntwurfsFabrikmethodeUml{
895   \begin{tikzpicture}
896     \umlsimpleclass[type=abstract]{Produkt}
897     \umlsimpleclass[below=2cm of Produkt]{KonkretesProdukt}
898     \umlinherit{KonkretesProdukt}{Produkt}
899
900     \umlclass[type=abstract,right=4cm of Produkt]{Erzeuger}{\{
901       \textit{erzeugeProdukt()}\}
902     }
903     \umlclass[below=1.2cm of Erzeuger]{KonkreterErzeuger}{\{
904       erzeugeProdukt()
905     }
906     \umlinherit{KonkreterErzeuger}{Erzeuger}
907
908     \umldep{KonkreterErzeuger}{KonkretesProdukt}
909   \end{tikzpicture}
910 }

```

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).

```

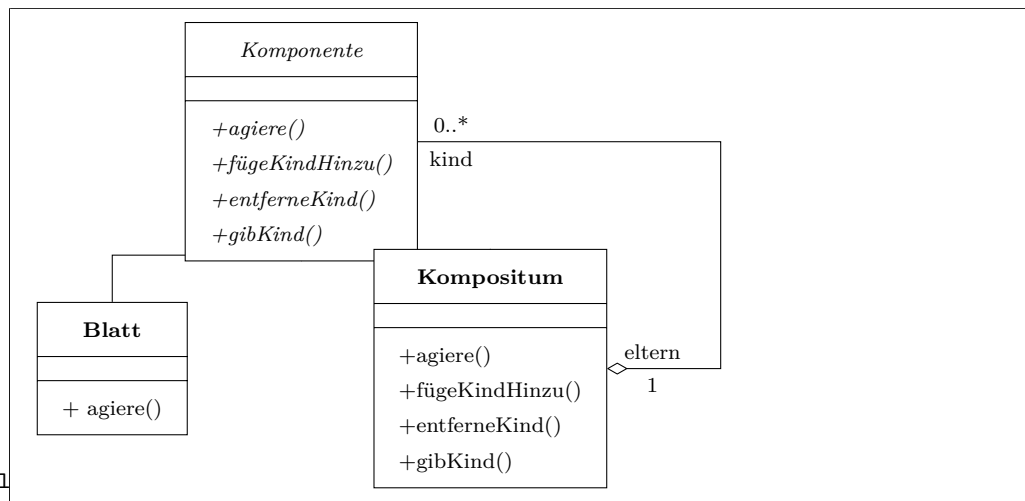
911 \def\liEntwurfsFabrikmethodeAkteure{
912   \begin{description}
913     \item[Produkt]
914
915     Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das
916     zu erzeugende Produkt.
917
918     \item[KonkretesProdukt]
919
920     KonkretesProdukt implementiert die Produkt-Schnittstelle.
921
922     \item[Erzeuger]
923
924     Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt
925     zu erzeugen und kann eine Default-Implementierung beinhalten.
926
927     \item[KonkreterErzeuger]
928
929     KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm
930     entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den
931     Konstruktor einer konkreten Produkt-Klasse aufruft).
932
933     \footcite{wiki:fabrikmethode}
934   \end{description}
935 }
```

\liEntwurfsFabrikmethode

```

936 \def\liEntwurfsFabrikmethode{
937   \liEntwurfsFabrikmethodeUml
938   \liEntwurfsFabrikmethodeAkteure
939 }
```

2.10.11 Kompositum (Composite)



\liEntwurfsKompositumUml


```

940 \def\liEntwurfsKompositumUml{
941   \begin{tikzpicture}
942     \umlclass[x=2.5,y=3,type=abstract]{Komponente}{-}{-}{
943       \textit{+agiere()}\
944       \textit{+fügeKindHinzu()}\
945       \textit{+entferneKind()}\
946       \textit{+gibKind()}
947     }
948     \umlclass[x=0]{Blatt}{-}{+ agiere()}
949     \umlclass[x=5]{Kompositum}{-}{-}{
950       +agiere()\
951       +fügeKindHinzu()\
952       +entferneKind()\
953       +gibKind()
954     }
955
956     \umlVHVinherit{Kompositum}{Komponente}
957     \umlVHVinherit{Blatt}{Komponente}
958     \umlHVHaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,]
959   \end{tikzpicture}
960 }

```

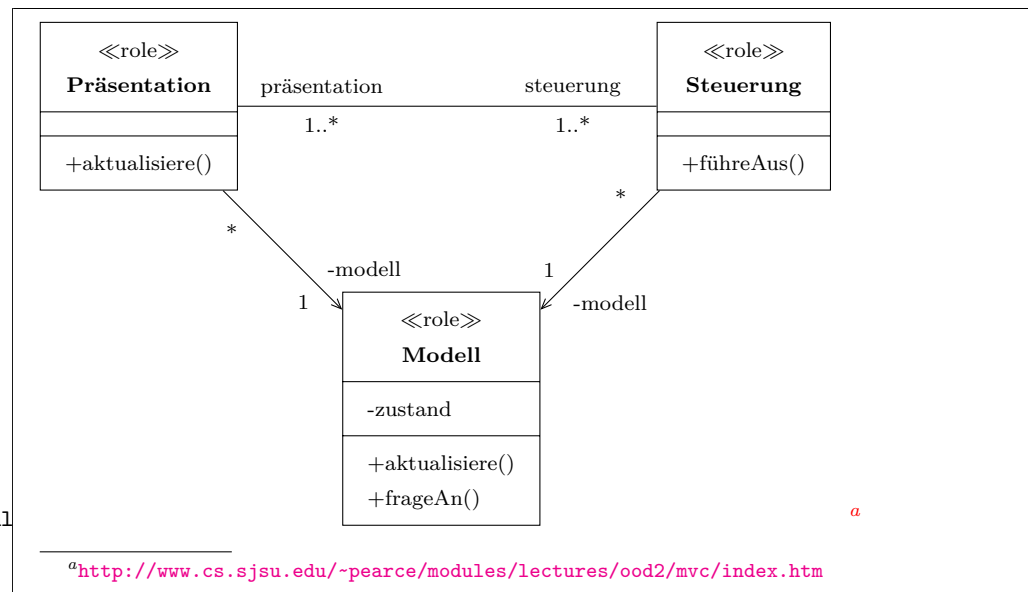
\liEntwurfsFabrikmethode

```

961 \def\liEntwurfsKompositum{
962   \liEntwurfsKompositumUml
963   \liEntwurfsKompositumAkteure
964 }

```

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



```

965 \def\liEntwurfsModellPraesentationSteuerungUml{
966   \begin{tikzpicture}
967     \umlclass[x=-4cm,type=role]{Präsentation}{-}{+aktualisiere()}
968     \umlclass[x=4cm,type=role]{Steuerung}{-}{+führeAus()}
969     \umlclass[y=-4cm,type=role]{Modell}{-}{
970       -zustand
971     }{
972       +aktualisiere()\
973       +frageAn()
974     }
975
976     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
977     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}

```

```

978 \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
979 \end{tikzpicture}
980 \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
981 }

```

ModellPraesentationSteuerung

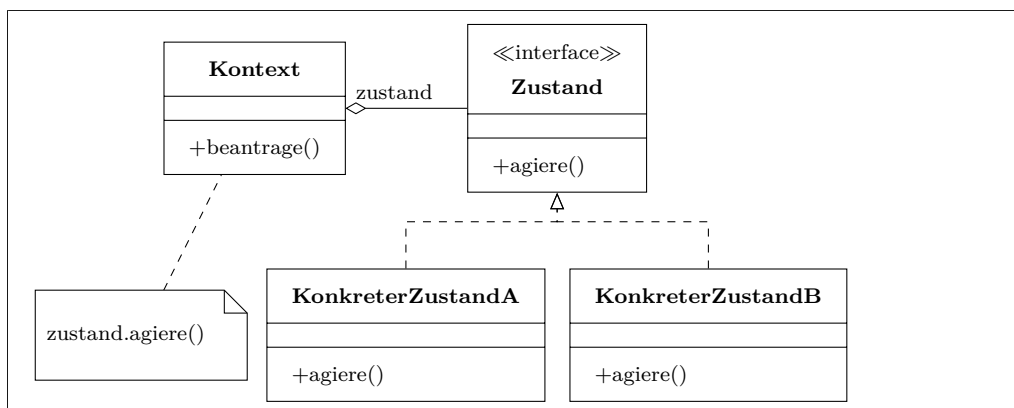
```

982 \def\liEntwurfs{
983 \liEntwurfsModellPraesentationSteuerungUml
984 \liEntwurfsModellPraesentationSteuerungAkteure
985 }

```

2.10.13 Zustand (State)

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



```

986 \def\liEntwurfsZustandUml{
987 \begin{tikzpicture}
988 \umlcclass[x=-1,y=3]{Kontext}{+beantrage()}
989 \umlcclass[x=3,y=3,type=interface]{Zustand}{+agiere()}
990 \umlcclass[x=1,y=0]{KonkreterZustandA}{+agiere()}
991 \umlcclass[x=5,y=0]{KonkreterZustandB}{+agiere()}
992
993 \umlVHVreal{KonkreterZustandA}{Zustand}
994 \umlVHVreal{KonkreterZustandB}{Zustand}
995
996 \umlagg[reg,arg=zustand,pos=0.4]{Kontext}{Zustand}
997
998 \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
999 \end{tikzpicture}
1000 }

```

\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.

```

1001 \def\liEntwurfsZustandAkteure{
1002 \begin{description}
1003 \item[Kontext (Context)]
1004
1005 definiert die clientseitige Schnittstelle und verwaltet die separaten
1006 Zustandsklassen.
1007

```

```

1008     \item[State (Zustand)]
1009
1010     definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
1011     implementiert gegebenenfalls ein Standardverhalten.
1012
1013     \item[KontreterZustand (ConcreteState)]
1014
1015     implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
1016     verbunden ist.
1017 \end{description}
1018 }

\liEntwurfsZustand

1019 \def\liEntwurfsZustand{
1020     \liEntwurfsZustandUml
1021     \liEntwurfsZustandAkteure
1022 }

1023

```

2.11 er.sty

```
1024 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1025 \ProvidesPackage{lehramt-informatik-er}[2020/06/13 Zum Setzen von
1026 ER-Diagrammen]

1027 \RequirePackage{tikz-er2}
1028 \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}

1029 \RequirePackage{soul}
1030 \RequirePackage{fontawesome}

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

1031 \ExplSyntaxOn

\liErEntity
1032 \def\liErEntity#1{\textbf{#1}}

\liErRelationship
1033 \def\liErRelationship#1{\ul{#1}}

\liErAttribute
1034 \def\liErAttribute#1{\emph{#1}}

\liErMpEntity mp = marginpar
Let-Abkürzung: \let\ e=\liErMpEntity
1035 \def\liErMpEntity#1{
1036   \liErEntity{#1}
1037   \marginpar{
1038     \liErEntity{\tiny\faSquareO{}}~E:~#1}
1039   }
1040 }
□

\liErMpRelationship Let-Abkürzung: \let\ r=\liErMpRelationship
1041 \def\liErMpRelationship#1{
1042   \liErRelationship{#1}
1043   \marginpar{
1044     \liErRelationship{\tiny\faGg{}}~R:~#1}
1045   }
1046 }

\liErMpAttribute Let-Abkürzung: \let\ a=\liErMpAttribute
1047 \def\liErMpAttribute#1{
1048   \liErAttribute{#1}
1049   \marginpar{
1050     \liErAttribute{\tiny\faCircleThin{}}~A:~#1}
1051   }
1052 }

\liErDatenbankName Let-Abkürzung: \let\ d=\liErDatenbankName
datenbank name
1053 \def\liErDatenbankName#1{
1054   {
1055     \footnotesize\texttt{(#1)}
1056   }
1057 }

1058 \ExplSyntaxOff
1059

```

2.12 formale-sprachen.sty

```

1060 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1061 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
1062 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]
1063
1064 \directlua{
1065   formale_sprachen = require('lehramt-informatik-formale-sprachen')
1066 }
1067
1068 \RequirePackage{hyperref}
1069 \RequirePackage{lehramt-informatik-mathe}

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

1070 \def\liMengeOhneMathe#1{\{ #1 \}}
1071 \def\liMenge#1{%
1072   \ifmmode%
1073     \liMengeOhneMathe{#1}%
1074   \else%
1075     $\liMengeOhneMathe{#1}$%
1076   \fi%
1077 }

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

1078 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

1079 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1080 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1081 \def\liPotenzmenge#1{$\liPotenzmengeOhneMathe{#1}$}

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

1082 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
1083 \let\liZustandsmenge=\liPotenzmenge

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

1084 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1085 \def\liUeberfuehrungsFunktion#1{
1086   \ifmmode
1087     \liUeberfuehrungsFunktionOhneMathe{#1}
1088   \else
1089     $\liUeberfuehrungsFunktionOhneMathe{#1}$
1090   \fi
1091 }

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

1092 \def\liAlphabet#1{$\Sigma = \{ #1 \}$}

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

1093 \def\liBandAlphabet#1{$\Gamma = \Sigma \cup \{ #1 \}$}

\liZustandsBuchstabe

1094 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1095 \def\liZustandsBuchstabeGross{Z}

```

```

\liZustandsmengeNr
1096 \def\zustandsnamens@liste#1#2{
1097   $
1098   \{
1099     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1100   \}
1101   $
1102 }
1103 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

\liZustandsmengeNrGross
1104 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

\liZustandsname \liZustandsname{1}: $z_1$
1105 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

\liZustandsnameGross \liZustandsnameGross{1}: $Z_1$
1106 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

\liAbleitung \liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1107 \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}

1108 \NewDocumentEnvironment { liProduktionsRegeln }
1109 { 0{P} +b }
1110 {
1111   \noindent
1112   $#1 = \{ $
1113   \vspace{-0.2cm}
1114   \begin{align*}
1115     \directlua{formale_sprachen.produktions_regeln('#2')}
1116   \end{align*}
1117   \vspace{-1.5cm}
1118   \begin{flushright}\}\end{flushright}
1119 } {}

\liProduktionen \liProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
1120 \def\liProduktionen#1{
1121   \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
1122 }

\liZustandsnameTiefgestellt Automatisch tiefgestellte Nummerierung \z1
Let-Abkürzung: \let\z=\liZustandsnameTiefgestellt
1123 \def\liZustandsnameTiefgestellt#1{
1124   \ifmmode
1125     \liZustandsBuchstabe\sb{#1}
1126   \else
1127     $\liZustandsBuchstabe\sb{#1}$
1128   \fi
1129 }

1130 \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}

1131 \NewDocumentCommand{ \liAusdruck } { O{L} m m } {
1132 $
1133 \tl_if_empty:nTF {#1} {} {#1 =}
1134 \{
1135 \, #2 \,
1136 |
1137 \, #3 \,
1138 \}$
1139 }
1140 \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

1141 \def\liFlaci#1{%
1142 \par
1143 {%
1144 \scriptsize
1145 Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1146 Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1147 Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1148 \href{https://flaci.com/#1}{flaci.com/#1}%
1149 }%
1150 \par
1151 }

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

1152 \ExplSyntaxOn
1153 \NewDocumentCommand {\liGrammatik} { O{G} m } {
1154 \tl_set:Nn \l_variablen_tl {V}
1155 \tl_set:Nn \l_alphabet_tl {\Sigma}
1156 \tl_set:Nn \l_produktionen_tl {P}
1157 \tl_set:Nn \l_start_tl {S}
1158
1159 \keys_define:nn { grammatik } {
1160 variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1161 alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1162 produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1163 start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1164 }
1165
1166 \keys_set:nn { grammatik } { #2 }
1167
1168 $#1 = (
1169 \l_variablen_tl,
1170 \l_alphabet_tl,
1171 \l_produktionen_tl,
1172 \l_start_tl
1173 )$

```



```
1174 }  
1175 \ExplSyntaxOff  
  
1176
```

2.13 formatierung.sty

```
1177 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1178 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

2.13.1 Schriftarten / Typographie

```
1179 \RequirePackage{mathpazo}
1180 \RequirePackage[no-math]{fontspec}
1181 \setmainfont{texgyrepagella}
1182 \setsansfont{QTAncientOlive}
1183 \RequirePackage{sectsty}
1184 \allsectionsfont{\fontspec{QTAncientOlive}}
```

2.13.2 Farben

```
1185 \RequirePackage{xcolor}
1186 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

2.13.3 Überschriften

```
1187 \RequirePackage{titlesec}
1188 \titleformat{\chapter}[display]{\sffamily\bfseries}{\Opt}{\LARGE}
1189 \titlespacing{\chapter}{Opt}{Opt}{*1}
1190 \titleformat{\paragraph}[hang]{\sffamily\normalsize\bfseries}{\theparagraph}{1em}{}
1191 \setcounter{secnumdepth}{0}
```

2.13.4 Listen

```
1192 \RequirePackage{paralist}
1193 \renewcommand\labelitemi{-}
1194 \renewcommand\labelitemii{-}
1195 \renewcommand\labelitemiii{-}
1196 \renewcommand\labelitemiv{-}
1197 % Counter: enumi enumii enumiii enumiv
1198 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1199 \renewcommand{\labelenumi}{(\alph{enumi})}
1200 \renewcommand{\labelenumii}{(\roman{enumii})}
```

2.13.5 Kasten

```
1201 \RequirePackage{mdframed}
1202 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1203 \NewDocumentEnvironment { liKasten }{ } {
1204   \begin{mdframed}[backgroundcolor=white!0]
1205 } {
1206   \end{mdframed}
1207 }
```

2.13.6 Header

```
1208 \RequirePackage{fancyhdr}
1209 \fancyhead[L,C,R]{}
1210 \fancyfoot[L]{}
1211 \fancyfoot[C]{}
1212 \fancyfoot[R]{\thepage}
1213 \pagestyle{fancy}
1214 \renewcommand{\headrulewidth}{Opt}
1215 \renewcommand{\footrulewidth}{Opt}
1216
```

2.14 gantt.sty

```

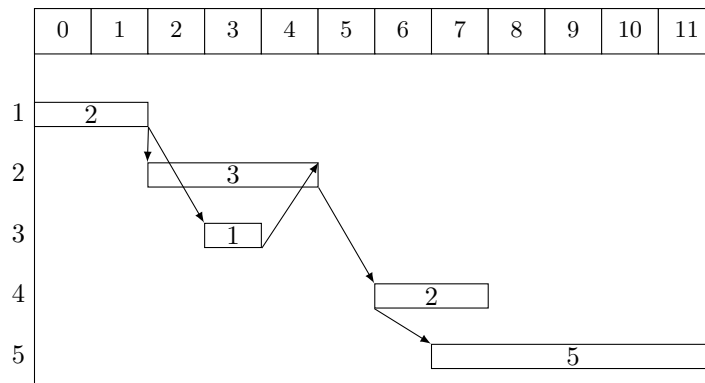
1217 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1218 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\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}

```



```

1219 \RequirePackage{tikz-uml}
1220 \RequirePackage{pgfgantt}
1221 \setganttlinklabel{f-s}{}
1222 \setganttlinklabel{s-s}{}
1223 \setganttlinklabel{f-f}{}
1224 \setganttlinklabel{s-f}{}
1225

```

2.15 grafik.sty

```
1226 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1227 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1228 TikZ, um Grafiken zeichnen zu können.]
1229 \RequirePackage{tikz}
1230
```

2.16 graph.sty

```

1231 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1232 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1233 \RequirePackage{tikz}

```

Für die die Adjazenzliste (`xrightrightarrow`)

```

1234 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

$$\begin{array}{c}
 \\
 a \\
 b \\
 c \\
 d \\
 e
 \end{array}
 \begin{array}{ccccc}
 a & b & c & d & e \\
 \left(\begin{array}{ccccc}
 0 & 1 & 0 & 4 & 0 \\
 0 & 0 & 0 & 1 & 3 \\
 0 & 0 & 0 & 1 & 0 \\
 4 & 1 & 1 & 0 & 2 \\
 0 & 0 & 0 & 0 & 0
 \end{array} \right)
 \end{array}$$

```

1235 \RequirePackage{blkarray}
1236 \usetikzlibrary{arrows.meta}

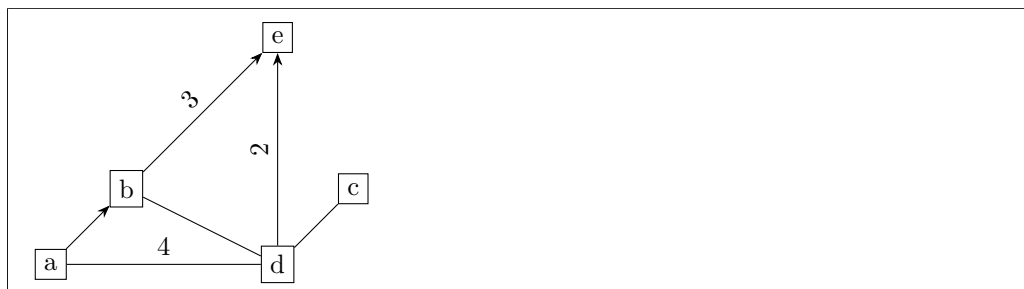
```

```

\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}

```



```

1237 \tikzset{
1238   li graph/.style={
1239     every node/.style={
1240       rectangle,
1241       draw,
1242     },
1243     every edge/.style={
1244       >={Stealth[black]},
1245       draw,
1246     },
1247     every edge/.append style={
1248       every node/.style={
1249         sloped,
1250         auto,
1251       }
1252     }
1253   },
1254   li markierung/.style={
1255     ultra thick,
1256   }
1257 }

```

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}

```

```

1258 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1259

```

2.17 hanoi.sty

```
1260 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1261 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
1262 von Hanoi-Grafiken]

Quelle: https://kleinco.de/latex-snippets/7/tikz-towers-of-hanoi-illustration-for-lat
1263 \RequirePackage{tikz}
1264 \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}

1265 \def\li@mset #1[#2]=#3{%
1266 \expandafter\xdef\csname #1#2\endcsname{#3}
1267 }
1268 \def\li@mget #1[#2]{%
1269 \csname #1#2\endcsname
1270 }
1271 \def\li@minc #1[#2]+=#3{%
1272 \pgfmathparse{\li@mget #1[#2]+#3}%
1273 \li@mset #1[#2]=\pgfmathresult
1274 }
1275
1276 \def\liHanoi#1#2{
1277   \edef\li@numdiscs{#1}
1278   \def\li@sequence{#2}
1279   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1280     % init colors
1281     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purp
1282     \li@mset col[\j]={\c};
1283     % draw poles and init pole counters
1284     \foreach \j in {1,2,3}{
1285       \li@mset pos[\j]=0
1286       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1287     }
1288     % draw base
1289     \draw (.5,-.5) -- +(3,0);
1290     % draw discs
1291     \foreach[count=\k] \i/\j in \li@sequence{
1292       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*
1293       \li@minc pos[\j]+=.5}
1294     }
1295   \end{tikzpicture}
1296 }

1297
```

2.18 komplexitaetstheorie.sty

```

1298 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1299 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1300 Setzen von Karps NP-vollständigen Problemen, Hilfsmakros für die
1301 Polynomialzeitreduktion.]

```

Let-Abkürzungen

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

1302 \liLadePakete{mathe}

Für das Makro `\liProblemBeschreibung` benötigt.

```
1303 \RequirePackage{mdframed}
```

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

```
1304 \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
```

1305 \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

```

1306 \def\liProblemBeschreibung#1#2#3{
1307   \begin{mdframed}[
1308     userdefinedwidth=9cm,
1309     align=center,
1310     backgroundcolor=white!0,
1311   ]
1312     \centerline{\large\liProblemName{#1}}
1313
1314     \medskip
1315
1316     \begin{description}
1317       \item[Gegeben:] #2
1318       \item[Frage:] #3
1319     \end{description}
1320   \end{mdframed}
1321 }

```



```

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

1322 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1323 \begin{displaymath}
1324 \liProblemName{#1}
1325 \preceq_{#2}
1326 \liProblemName{#3}
1327 \end{displaymath}
1328 }

\liProblemVertexCover

1329 \def\liProblemClique{%
1330 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1331 Mindestgröße  $n$  in einem gegebenen Graphen.
1332 \footcite[Seite 76]{theo:fs:4}
1333 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1334 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1335 \footcite{wiki:cliquenproblem}
1336 }

\liProblemVertexCover

1337 \def\liProblemVertexCover{%
1338 %
1339 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1340 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1341 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1342 \footcite{wiki:knotenueberdeckung}
1343
1344 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1345 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1346 mindestens einem Knoten aus  $U$  verbunden ist.
1347 \footcite[Seite 78]{theo:fs:4}%
1348 }

\liProblemSubsetSum Kein Karp-21-Problem

1349 \def\liProblemSubsetSum{%
1350 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1351 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1352 \footcite{wiki:teilsommenproblem}
1353 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1354 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1355 größer als eine gegebene obere Schranke  $c$  ist.
1356 \footcite[Seite 74]{theo:fs:4}
1357 }

\liProblemSubsetSum Kein Karp-21-Problem

1358 \def\liProblemSat{%
1359 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1360 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1361 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1362 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1363 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1364 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1365 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1366 aufgestellt werden.
1367 \footcite[Seite 71]{theo:fs:4}
1368 }

1369

```

2.19 kontrollflussgraph.sty

```
1370 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1371 \ProvidesPackage{lehramt-informatik-kontrollflussgraph}[2020/11/07]

1372 \RequirePackage{tikz}
1373 \usetikzlibrary{positioning}
1374 \tikzset{
1375   li kontrollfluss/.style={
1376     knoten/.style={
1377       circle,
1378       draw
1379     },
1380     usebox/.style={
1381       draw,
1382       rectangle,
1383       font=\scriptsize,
1384       anchor=west,
1385       align=left,
1386     },
1387     bedingung/.style={
1388       midway,
1389       draw=none,
1390       font=\scriptsize
1391     },
1392     knotenbeschriftung/.style={
1393       draw,
1394       rectangle,
1395       midway,
1396       font=\scriptsize
1397     },
1398     wahr/.style={
1399       thick
1400     },
1401     falsch/.style={
1402       dashed
1403     },
1404     every node/.style={
1405       circle,
1406       draw,
1407     },
1408     every edge/.append style={
1409       every node/.style={
1410         draw=none,
1411         bedingung,
1412       }
1413     },
1414     every path/.style={
1415       draw,
1416       ->,
1417     },
1418     every pin/.style={
1419       draw,
1420       dotted,
1421       rectangle,
1422       pin position=right
1423     },
1424     every pin edge/.style={
1425       dotted,
1426       arrows=-,
1427     }
1428   }
1429 }
```

liKontrollflussgraph

```

1430 \NewDocumentEnvironment { liKontrollflussgraph } { 0{ } } {
1431   \begin{tikzpicture}[
1432     li kontrollfluss,
1433     #1
1434   ]
1435 } {
1436   \end{tikzpicture}
1437 }

\liAnweisung
1438 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}

\liBedingung Let-Abkürzung: \let\b=\liBedingung
1439 \def\liBedingung#1#2{node[bedingung,#1]{\texttt{#2}}}}

\liBedingungWahr Let-Abkürzung: \let\bWahr=\liBedingungWahr
1440 \def\liBedingungWahr#1{node[bedingung,#1]{\texttt{[true]}}}}

\liBedingungFalsch Let-Abkürzung: \let\bFalsch=\liBedingungFalsch
1441 \def\liBedingungFalsch#1{node[bedingung,#1]{\texttt{[false]}}}}

\liKontrollCode Let-Abkürzung: \let\c=\liKontrollCode
1442 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}

\liKontrollTextzeileKnoten Let-Abkürzung: \let\k=\liKontrollTextzeileKnoten
1443 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw

\liKontrollKnotenPfad Let-Abkürzung: \let\p=\liKontrollKnotenPfad
1444 \ExplSyntaxOn
1445 \NewDocumentCommand { \liKontrollKnotenPfad }{ m }
1446 {
1447   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
1448   \seq_set_map:Nnn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
1449   \seq_use:Nn \l_tmpa_seq { ~~~~ }
1450 }
1451 \ExplSyntaxOff

1452

```

2.20 kopf-fusszeilen.sty

```
1453 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1454 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1455 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1456 \ExplSyntaxOn

1457 \fancyhead{}
1458 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1459 \fancyhead[L0,RE]{\scriptsize\today}
1460 \fancyfoot{}
1461 \fancyfoot[LE,R0]{\thepage}
1462 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1463 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1464 \renewcommand{\headrulewidth}{0.4pt}
1465 \renewcommand{\footrulewidth}{0.4pt}

1466 \ExplSyntaxOff

1467
```

2.21 literatur-dummy.sty

```
1468 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1469 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1470 \def\literatur{}

\footcite

1471 % \RequirePackage[stable,multiple]{footmisc}
1472 \NewDocumentCommand{ \footcite } { o m }{}

1473
```

2.22 literatur.sty

```
1474 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1475 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1476 \RequirePackage{csquotes}
1477 \RequirePackage[
1478   bibencoding=utf8,
1479   citestyle=authortitle,
1480   backend=biber,
1481 ]{biblatex}
1482 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1483 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1484 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1485 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1486 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1487 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1488 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1489 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1490 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1491 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1492 % To allow footnotes in the heading
1493 \RequirePackage[stable,multiple]{footmisc}

\literatur

1494 \def\literatur{\printbibliography[heading=subbibliography]}

1495
```

2.23 makros.sty

```

1496 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1497 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1498 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1499 anderen Paket passen]

1500 \RequirePackage{hyperref}
1501 \RequirePackage{graphicx}

    Für die Umgebung liQuellen benötigt.
1502 \RequirePackage{paralist}

\inhaltsverzeichnis
1503 \def\inhaltsverzeichnis {
1504   \begin{mdframed}
1505     \begin{group}
1506       \let\clearpage\relax
1507       \tableofcontents
1508     \end{group}
1509   \end{mdframed}
1510 }

\memph \mephm (\marginpar and \emph)
1511 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1512 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1513 \newcommand{\liPseudoUeberschrift}[1]{
1514   \bigskip
1515   \noindent
1516   \textsf{\textbf{#1}}
1517   \noindent
1518 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1519 \newcommand{\liBeschriftung}[1]{
1520   \par
1521   \noindent
1522   \medskip
1523   \textbf{#1}:
1524   \medskip
1525   \noindent
1526 }

\hinweis
1527 \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.
1528 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

liEinbettung
1529 \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.
1530 \RequirePackage{xparse}
1531 \ExplSyntaxOn

```

```

1532 \NewDocumentEnvironment{ liAntwort } { 0{standard} }
1533 {
1534   \str_case:nn {#1} {
1535     {standard} {
1536       \def\beschriftung{}
1537       \mdfsetup{backgroundcolor=gray!5,linecolor=gray}
1538     }
1539     {richtig} {
1540       \def\beschriftung{richtig}
1541       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1542     }
1543     {falsch} {
1544       \def\beschriftung{falsch}
1545       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1546     }
1547     {muster} {
1548       \def\beschriftung{Musterlösung}
1549       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1550     }
1551   }
1552   \ifx\beschriftung\empty\else
1553     \noindent
1554     \textbf{\beschriftung{:}}
1555     \fi
1556     \begin{mdframed}
1557   }
1558 {\end{mdframed}}

```

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

```

1559 \NewDocumentEnvironment{ liAdditum } { o }
1560 {
1561   \begin{mdframed}[backgroundcolor=yellow!5]
1562     \IfNoValueTF {#1}
1563     { \liPseudoUeberschrift{Additum} }
1564     { \liPseudoUeberschrift{Additum:~#1} }
1565   }
1566 {\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.

```

1567 \NewDocumentEnvironment{ liExkurs }{o +b}{
1568   \vspace{0.2cm}%
1569   \begin{mdframed}[
1570     backgroundcolor=white,
1571     bottomline=false,
1572     innermargin=1cm,
1573     leftline=true,
1574     linecolor=black,
1575     linewidth=0.1cm,
1576     outermargin=1cm,
1577     rightline=false,
1578     topline=false,
1579   ]

```



```

1580 \footnotesize
1581 \noindent%
1582 \textbf{Exkurs:~#1}\par%
1583 \noindent%
1584 #2
1585 \end{mdframed}
1586 \vspace{0.2cm}
1587 }{}

```

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

```

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

```

Weiterführende Literatur:

- Quelle 1
- Quelle 2

```

1588 \cs_new:Npn \listen@punkt #1 {\item #1}
1589 \NewDocumentEnvironment { liQuellen }{ +b }
1590 {
1591   \seq_clear_new:N \l_quellen
1592   \seq_set_split:Nnn \l_quellen {\item} {#1}
1593   \seq_remove_all:Nn \l_quellen {}
1594   \begin{mdframed}[backgroundcolor=white!0]
1595     \footnotesize
1596     \noindent
1597     \textsf{\textbf{Weiterführende~Literatur:}}
1598     \medskip
1599     \begin{compactitem}
1600       \seq_map_function:NN \l_quellen {\listen@punkt}
1601     \end{compactitem}
1602   \end{mdframed}
1603   %
1604   \makeatletter
1605   \par\@afterindentfalse\@afterheading
1606   \makeatother
1607 } {}

```

liLernkartei

```

1608 \NewDocumentEnvironment { liLernkartei }{ m +b }
1609 {
1610   \begin{mdframed}
1611     \footnotesize
1612     \noindent%
1613     \textbf{Lernkarteikarte:~#1}\par%
1614     \noindent%
1615     #2
1616   \end{mdframed}
1617 } {}

```

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

```

1618 \NewDocumentEnvironment { liDiagramm }{ m +b }
1619 {
1620   \begin{mdframed}[backgroundcolor=white!0]
1621     \small
1622     \noindent%
1623     \textit{#1}:
1624     \begin{center}

```

```

1625 #2
1626 \medskip
1627 \end{center}
1628 \end{mdframed}
1629 } {}
1630 \ExplSyntaxOff

\liFussnoteUrl \liFussnoteUrl[\langle zusätzlicher-text \rangle]{\langle url \rangle} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
1631 \NewDocumentCommand{\liFussnoteUrl} { o m } {
1632 \footnote{\url{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1633 }
1634

\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.
1635 \NewDocumentCommand{\liFussnoteLink} { o m m } {
1636 \footnote{\href{#3}{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1637 }

\zB
1638 \def\zB{z.\.,B. }

\ZB
1639 \def\ZB{Z.\.,B. }

\dh
1640 \def\dh{d.\.,h. }

1641

```

2.24 master-theorem.sty

1642 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1643 \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{0}(1)} % f(n)

\liMasterFallRechnung

% 1. Fall

{für $\varepsilon = 4$: \}

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

% 2. Fall

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

% 3. Fall

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

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

\liMasterVariablenDeklaration

{ } % a

{ } % b

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

\liMasterFallRechnung

% 1. Fall

{ }

% 2. Fall

{ }

% 3. Fall

{ }

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

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

1644 \ExplSyntaxOn

1645 \RequirePackage{amsmath}

\liRundeKlammer

1646 \def\liRundeKlammer#1{

1647 \negthinspace \left(#1 \right)

1648 }

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

1649 \def\liThetaOhneMathe#1{

1650 \Theta \liRundeKlammer{#1}

1651 }

1652 \def\liTheta#1{

1653 \ifmmode

1654 \liThetaOhneMathe{#1}

1655 \else

1656 \$\liThetaOhneMathe{#1}\$

1657 \fi

1658 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
1659 \def\liOmegaOhneMathe#1{
1660 \Omega \liRundeKlammer{#1}
1661 }
1662 \def\liOmega#1{
1663 \ifmmode
1664 \liOmegaOhneMathe{#1}
1665 \else
1666 $\liOmegaOhneMathe{#1}$
1667 \fi
1668 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
1669 \def\liOOhneMathe#1{
1670 \mathcal{O} \liRundeKlammer{#1}
1671 }
1672 \def\liO#1{
1673 \ifmmode
1674 \liOOhneMathe{#1}
1675 \else
1676 $\liOOhneMathe{#1}$
1677 \fi
1678 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
1679 \def\liTOhneMathe#1#2{
1680 \tl_if_blank:nTF {#1}
1681 {}
1682 {#1 \cdot }
1683 T
1684 \liRundeKlammer{\frac{n}{#2}}
1685 }
1686 \def\liT#1#2{
1687 \ifmmode
1688 \liTOhneMathe{#1}{#2}
1689 \else
1690 $\liTOhneMathe{#1}{#2}$
1691 \fi
1692 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
1693 \def\liRekursionsGleichung{
1694 $T(n) = \liT{a}{b} + f(n)$
1695 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
1696 \def\liBedingungEins{
1697 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
1698 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
1699 \def\liBedingungZwei{
1700 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
1701 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
1702 \def\liBedingungDrei{
1703 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
1704 }

1705 \ExplSyntaxOff

```

\liMasterVariablen

```

1706 \def\liMasterVariablen{
1707   \begin{displaymath}
1708     T(n) = \liT{a}{b} + f(n)
1709   \end{displaymath}
1710
1711   \begin{itemize}
1712     \item[$a = $]
1713       Anzahl der Unterprobleme in der Rekursion
1714
1715     \item[$\textstyle{\frac{1}{b}} = $]
1716       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
1717       repräsentiert wird
1718
1719     \item[$f(n) = $]
1720       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
1721       die Kombination der Teillösungen entstehen
1722   \end{itemize}
1723   \footcite{wiki:master-theorem}
1724   \footcite[Seite 19-35 (PDF 11-24)]{aud:fs:2}
1725 }
```

\liMasterFaelle

```

1726 \def\liMasterFaelle{
1727   \begin{description}
1728     \item[1. Fall:]
1729        $T(n) \in \liTheta{n^{\log_{sb{b}}a}}$ 
1730
1731       \hfill falls \liBedingungEins
1732       für  $\varepsilon > 0$ 
1733
1734     \item[2. Fall:]
1735        $T(n) \in \liTheta{n^{\log_{sb{b}}a} \cdot \log n}$ 
1736
1737       \hfill falls \liBedingungZwei
1738
1739     \item[3. Fall:]
1740        $T(n) \in \liTheta{f(n)}$ 
1741
1742       \hfill falls \liBedingungDrei
1743       für  $\varepsilon > 0$ 
1744       und ebenfalls für ein  $c$  mit  $0 < c < 1$  und alle hinreichend großen  $n$ 
1745       gilt:
1746        $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$ 
1747   \end{description}
1748 }
```

\liMasterVariablenDeklaration

```

1749 \def\liMasterVariablenDeklaration#1#2#3{
1750   \begin{description}
1751     \item[Allgemeine Rekursionsgleichung:] \strut
1752
1753     \liRekursionsGleichung
1754
1755     \item[Anzahl der rekursiven Aufrufe ( $a$ ):] \strut
1756
1757     #1
1758
1759     \item[Anteil Verkleinerung des Problems ( $b$ ):] \strut
1760
1761     um  $\frac{1}{#2}$  also  $b = #2$ 
1762
1763     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut
```

```

1764
1765     $#3$
1766
1767     \item[Ergibt folgende Rekursionsgleichung:] \strut
1768
1769     $T(n) = \liT{\#1}{\#2} + \#3$
1770 \end{description}
1771 }

```

\liMasterFallRechnung

```

1772 \def\liMasterFallRechnung#1#2#3{
1773   \begin{description}
1774     \item[1. Fall:] \liBedingungEins:
1775
1776     #1
1777
1778     \item[2. Fall:] \liBedingungZwei:
1779
1780     #2
1781
1782     \item[3. Fall:] \liBedingungDrei:
1783
1784     #3
1785   \end{description}
1786 }

```

\liMasterExkurs

```

1787 \def\liMasterExkurs{
1788   \begin{liExkurs}[Master-Theorem]
1789     \liMasterVariablen
1790
1791     \noindent
1792     Dann gilt:
1793
1794     \liMasterFaelle
1795   \end{liExkurs}
1796 }

```

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

```

1797 \def\liMasterWolframLink#1{
1798   Berechne die Rekursionsgleichung auf WolframAlpha:
1799   \href{https://www.wolframalpha.com/input/?i=#1}{WolframAlpha}
1800 }
1801

```

2.25 mathe.sty

```
1802 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1803 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
1804
1805 % for example \ltimes \rtimes
1806 %\RequirePackage{amssymb}
1807 \RequirePackage{amsmath}
1808
1809 %%
1810 % \mlq \mrq
1811 %%
1812 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
1813 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
1814
```

2.26 minimierung.sty

```

1815 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1816 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
1817 Minimierungsalgorithmus von einem NEA zu einem DEA]

1818 \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

1819 \def\liFussnote#1{\$x_{#1}\$}

1820 \def\li@fussnote@text#1#2{
1821 \liFussnote{#1}
1822 \quad
1823 {\footnotesize #2}
1824 }

\liFussnoteEinsText

1825 \def\liFussnoteEinsText{
1826 \li@fussnote@text{1}
1827 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
1828 }

\liFussnoteZweiText

1829 \def\liFussnoteZweiText{
1830 \li@fussnote@text{2}
1831 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
1832 }

\liFussnoteDreiText

1833 \def\liFussnoteDreiText{
1834 \li@fussnote@text{3}

```



```

1835 {In weiteren Iterationen markierte Zustände.}
1836 }

\liFussnoteVierText
1837 \def\liFussnoteVierText{
1838   \li@fussnote@text{4}
1839   {...}
1840 }

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



1841 \def\liFussnoten{
1842   \bigskip
1843
1844   \noindent
1845   \liFussnoteEinsText
1846
1847   \noindent
1848   \liFussnoteZweiText
1849
1850   \noindent
1851   \liFussnoteDreiText
1852
1853   \noindent
1854   \liFussnoteVierText
1855 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
1856 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
1857 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
1858 \def\liZustandsPaar#1#2{
1859   $(
1860     \liZustandsPaarVariablenName_#1,
1861     \liZustandsPaarVariablenName_#2
1862   )$
1863 }

liUebergangsTabelle
1864 \renewcommand{\arraystretch}{1.4}
1865 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
1866   \liPseudoUeberschrift{Übergangstabelle}
1867   \begin{center}
1868     \begin{tabular}{r|l|l}
1869       \textbf{Zustandspaar} & \textbf{#1} & \textbf{#2} \\ \hline
1870     \end{tabular}
1871   \end{center}
1872 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle: Minimierungstabelle (Table filling)

1874 \ExplSyntaxOn
1875 \def\liUeberschriftDreiecksTabelle{
1876   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
1877 }

```

\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. —————

```

1878 \def\liMinimierungErklaerung{
1879   %\footcite[Seite~19]{koenig}
1880   \liParagraphMitLinien{
1881     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
1882     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
1883     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
1884     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
1885      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
1886     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
1887     somit~zueinander~( $k - 1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
1888     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
1889     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
1890   }
1891 }
1892 \ExplSyntaxOff
1893

```

2.27 normalformen.sty

```

1894 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1895 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10
1896 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
1897 Attributhülle]
1898 \liLadePakete{mathe}
1899 \directlua{
1900   helfer = require('lehramt-informatik-helfer')
1901   normalformen = require('lehramt-informatik-normalformen')
1902 }

```

Let-Abkürzungen

```

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

```

```

1903 \def\liTeilen#1{
1904   \directlua{tex.print(normalformen.teilen('#1'))}
1905 }

```

```

\liAttributHuelle Let-Abkürzung: \let\ah=\liAttributHuelle
                  Regulärer Ausdruck zum Konvertieren AttrHülle\((.*)\) \ah{$1}
1906 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(#1)}
1907 \def\liAttributHuelle#1{
1908   \ifmmode
1909     \liAttributHuelleOhneMathe{#1}
1910   \else
1911     $\liAttributHuelleOhneMathe{#1}$
1912   \fi
1913 }

```

```

\liAttributMenge Let-Abkürzung: \let\m=\liAttributMenge
1914 \def\liAttributMenge#1{\{ \textit{#1} \}}

```

liAHuelle

```

1915 \NewDocumentEnvironment{ liAHuelle } { +b } {
1916   \begingroup
1917   \footnotesize
1918   \begin{multline*}
1919     #1
1920   \end{multline*}
1921   \endgroup
1922 } { }

```

```

AttributHuelleLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multiline
Let-Abkürzung: \let\ahL=\liAttributHuelleLinksReduktion
                  \ahL{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}
1923 \def\liAttributHuelleLinksReduktion#1#2#3{
1924   \shoveleft{
1925     \liAttributHuelleOhneMathe{FA, \liAttributMenge{\liAttributMenge{#1} - \liAttributMenge{#2}}
1926   } \\\
1927   \shoveright{
1928     \liAttributMenge{#3}
1929   } \\\
1930 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
    \liFunktionaleAbhaengigkeit{A, B -> C, D} Regulärer Ausdruck zum Konvertieren:
    \$(.*) \rightarrow (.*)\$ \fa{$1 -> $2}
1931 \def\liFunktionaleAbhaengigkeit#1{%
1932   \directlua{normalformen.drucke_funk_abhaengigkeit('#1')}%
1933 }

\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;
1934 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
1935   \par
1936   \noindent
1937   #1 $= \{$
1938   \par
1939   \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
1940   \par
1941   \noindent$\}$
1942 }

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

```

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

```

Regulärer Ausdruck zum Konvertieren: \$(R.*)\((.*)\)\$ \liRelation[$1]{$2}
1943 \NewDocumentCommand {\liRelation} { O{R} m } {
1944   $\directlua{
1945     local name = helper.konvertiere_tiefgestellt('#1')
1946     tex.print(name)
1947   }$(\textit{\,#2\,})
1948 }
1949

```

2.28 petri.sty

```
1950 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1951 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]
```

Let-Abkürzungen

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

```
1952 \RequirePackage{tikz}
1953 \usetikzlibrary{petri,arrows.meta}
```

Für die Darstellungsmatrix

```
1954 \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

```
1955 \def\liPetriSetzeSchluessel{%
1956   \def\TmpTransitionOne{}%
1957   \def\TmpTransitionTwo{}%
1958   \def\TmpTransitionThree{}%
1959   \def\TmpTransitionFour{}%
1960   \def\TmpTransitionFive{}%
1961   \def\TmpTransitionSix{}%
1962   \def\TmpTransitionSeven{}%
1963   \def\TmpTransitionEight{}%
1964   \def\TmpTransitionNine{}%
1965   \def\TmpTransitionTen{}%
1966   \pgfkeys{/petri/.cd,
1967     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
1968     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
1969     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
1970     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
1971     p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
```

```

1972   p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
1973   p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
1974   p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
1975   p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
1976   p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
1977   t1/.store in=\TmpTransitionOne,t1/.default=activated,
1978   t2/.store in=\TmpTransitionTwo,t2/.default=activated,
1979   t3/.store in=\TmpTransitionThree,t3/.default=activated,
1980   t4/.store in=\TmpTransitionFour,t4/.default=activated,
1981   t5/.store in=\TmpTransitionFive,t5/.default=activated,
1982   t6/.store in=\TmpTransitionSix,t6/.default=activated,
1983   t7/.store in=\TmpTransitionSeven,t7/.default=activated,
1984   t8/.store in=\TmpTransitionEight,t8/.default=activated,
1985   t9/.store in=\TmpTransitionNine,t9/.default=activated,
1986   t10/.store in=\TmpTransitionTen,t10/.default=activated,
1987   scale/.store in=\TmpScale,scale/.default=0.5,
1988   x/.store in=\TmpX,x/.default=5,
1989   y/.store in=\TmpY,y/.default=5,
1990   }%
1991 }

1992 \tikzset{
1993   li petri/.style={
1994     activated/.style={
1995       very thick
1996     },
1997     inhibitor/.style={
1998       {Circle[open,length=2mm,fill=white]}-
1999     }
2000   }
2001 }

```

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

```

2002 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2003 \def\liPetriTransitionsName#1{
2004   \ifmmode
2005     \liPetriTransitionsNameOhneMathe{#1}
2006   \else
2007     $\liPetriTransitionsNameOhneMathe{#1}$
2008   \fi
2009 }

```

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

```

2010 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2011   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2012 }

```

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

```

2013 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2014 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm} }
2015

```

2.29 potenzmengen-konstruktion.sty

2016 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2017 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
 2018 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]

2019 \liLadePakete{formale-sprachen}
 2020 \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}
    }
  }
}
```

```
2021 \def\liZustandsMengenSammlung#1#2{
2022   \liZustandsnameGross{#1}
2023   {
2024     \footnotesize
2025     \liPotenzmenge{
2026       \str_case:nn {#1} {#2
2027       }
2028     }
2029 }
```

\liZustandsMengenSammlungNr

```
2030 \def\liZustandsMengenSammlungNr#1#2{
2031   \liZustandsnameGross{#1}
2032   {
```

```
2033     \footnotesize
2034     \liZustandsmengeNr{
2035         \str_case:nn {#1} #2
2036     }
2037 }
2038 }

2039 \ExplSyntaxOff
2040
```


2.30 pseudo.sty

```

2041 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2042 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen
2043 von Pseudo-Code, Hüll-Paket um algorithm2e]

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

```

2044 \RequirePackage[german,boxruled]{algorithm2e}

```

```

2045

```

2.31 pumping-lemma.sty

2046 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2047 \ProvidesPackage{lehramt-informatik-pumping-lemma}[2021/08/11 Enthält die
2048 Definitionen für das Pumping-Lemma in der Regulären Sprache und
2049 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2050 \def\liPumpingRegulaer{%
2051   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2052   alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2053    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2054   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2055
2056   \begin{enumerate}
2057     \item  $|v| \geq 1$ 
2058     (Das Wort  $v$  ist nicht leer.)
2059
2060     \item  $|uv| \leq j$ 
2061     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2062
2063     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w$  in  $L$ 
2064     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2065     Sprache  $L$ )
2066   \end{enumerate}
2067
2068   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2069   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2070 }
```

\liPumpingKontextfrei

```
2071 \def\liPumpingKontextfrei{%
2072   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2073   sich alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  zerlegen lassen in
2074    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2075
2076   \begin{enumerate}
2077     \item  $|vx| \geq 1$ 
2078     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2079
2080     \item  $|vwx| \leq j$ 
2081     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2082
2083     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy$  in  $L$  (Für jede
2084     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2085     Sprache  $L$ )
2086   \end{enumerate}
2087 }
2088
```

2.32 quicksort.sty

```

2089 % https://tex.stackexchange.com/a/142634
2090 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2091 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2092
2093 %-----
2094 % USAGE:
2095 % \QSinitialize{comma, separated, numerical, values}
2096 % \loop
2097 % \QSpivotStep
2098 % \ifnum\value{pivotcount}>0
2099 %   \QSSortStep
2100 % \repeat
2101 %-----
2102
2103 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2104 \RequirePackage{xintfrac, xinttools}
2105
2106 \RequirePackage{tikz}
2107
2108 %-----
2109 % FIRST PART: TikZ styles and macros for the actual drawing
2110 \newcounter{cellcount}% used for coordinates of the node
2111 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2112
2113 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2114 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2115 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2116 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2117 % by police of LaTeX good conduct ? )
2118 \tikzset{ll/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2119          oo/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2120          rr/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2121 % this is the "b" style as used in the image below
2122          bb/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2123 % nicer:
2124          bb/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2125          gg/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2126
2127 % NOTE the b style was originally the same as the r(aised) style apart from
2128 % not being raised, but I find it nicer with a somewhat different
2129 % specification. I have not updated the images though.
2130
2131 % How the nodes are drawn depending on whether on the left of the pivot value
2132 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2133
2134 \def\DecoLEFT #1{%
2135   \xintFor* ##1 in {#1} \do
2136     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2137 }
2138
2139 \def\DecoINERT #1{%
2140   \xintFor* ##1 in {#1} \do
2141     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2142 }
2143
2144 \def\DecoRIGHT #1{%
2145   \xintFor* ##1 in {#1} \do
2146     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2147 }
2148
2149 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2150   \xintFor* ##1 in {#1} \do

```

```

2151     {\stepcounter{cellcount}}%
2152     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2153 }
2154
2155 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2156     \xintFor* ##1 in {#1} \do
2157     {\stepcounter{cellcount}}%
2158     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2159 }
2160
2161 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2162     \xintFor* ##1 in {#1} \do
2163     {\stepcounter{cellcount}}%
2164     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2165 }
2166
2167 %-----
2168 % SECOND PART: the actual sorting routines.
2169
2170 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2171 \def\QS@sort@b #1{\ifcase #1
2172     \expandafter\QS@sort@empty
2173     \or\expandafter\QS@sort@single
2174     \else\expandafter\QS@sort@c
2175     \fi
2176 }%
2177 \def\QS@sort@empty #1{}
2178 \def\QS@sort@single #1{\QSIr {#1}}
2179
2180 % This step is to pick the last as pivot.
2181 \def\QS@sort@c #1%
2182     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2183
2184 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2185 % The trick with \xintApplyUnbraced is that for example when selecting
2186 % the elements smaller than pivot, if we had been using \xintApply we
2187 % would have had at the minimum an empty brace pair. Thus we use the
2188 % "unbraced" variant, but then the \QS@select@smaller has added in
2189 % anticipation a level of braces.
2190 \def\QS@sort@d #1#2{%
2191     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2192     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2193     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2194 }%
2195 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2196 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2197 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2198
2199 %
2200 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2201 %
2202 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2203 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2204 % latter must handle correctly an empty argument.
2205
2206 %-----
2207 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QSinitialize.
2208
2209 % This draws all with suitable highlighting for the newly chosen pivots
2210 % (which will be shown raised)

```

```

2211 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2212             \let\QSIr\DecoINERT
2213             \let\QSIrr\DecoINERT
2214             \let\QSRr\DecoRIGHTwithPivot
2215 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2216             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2217             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2218 }
2219
2220 % This sorts and then draws, showing where the pivot chosen in the previous
2221 % step go. Next time they will have become "inert". If pivotcount is still at
2222 % zero on exit from \QSpivotStep, then this is the signal to stop before
2223 % executing \QSSortStep.
2224 \def\QSSortStep {\def\QSLr {\noexpand\QS@sort@a}%
2225             \def\QSRr {\noexpand\QS@sort@a}%
2226             \def\QSIr {\noexpand\QSIrr}%
2227             \let\QSIrr\relax
2228             \edef\QS@list{\QS@list}%
2229             \let\QSLr\relax
2230             \let\QSRr\relax
2231             \let\QSIr\relax
2232             \edef\QS@list{\QS@list}%
2233             \let\QSLr\DecoLEFT
2234             \let\QSIr\DecoINERTwithPivot
2235             \let\QSIrr\DecoINERT
2236             \let\QSRr\DecoRIGHT
2237 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2238             \setcounter{cellcount}{0}%
2239             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2240 }
2241
2242 \def\QSinitialize #1{%
2243     % first, we convert the comma separated values into a list of braced items
2244     % we use an \edef, and anyhow many \edef's will be used later
2245     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2246     \let\QSRr\DecoRIGHT
2247     % The \QSRr marker mutated to draw the last element as
2248     % pivot and the earlier ones with the suitable style.
2249     %
2250     % The list of marked braced items \QS@list is used both for drawing
2251     % (as here) and for doing the exchange of elements during sort.
2252     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2253             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2254 }
2255

```

2.33 relationale-algebra.sty

2256 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2257 \ProvidesPackage{lehrant-informatik-relationale-algebra}[2020/12/11]

2258 \RequirePackage{amsmath}

2259 \RequirePackage{amssymb}

Zum Zeichnen von Operatorenbäumen verwenden wir TikZ

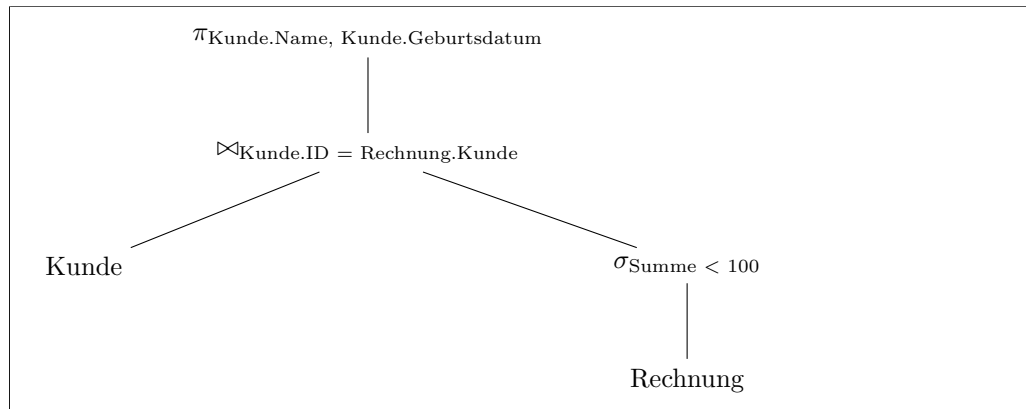
```
\begin{tikzpicture}
  \node
    (pi) {\pi_{\text{Kunde.Name, Kunde.Geburtsdatum}}};

  \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}
```



2260 \RequirePackage{tikz}

2261 \usetikzlibrary{positioning}

Privates Makros, das zwei Querstriche erzeugt.

```
2262 \def\o@join{\setbox0=\hbox{\bowtie}}%
2263 \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2264 }
```

\leftouterjoin A \leftouterjoin B: $A \bowtie B$

```
2265 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}
```

\rightouterjoin A \rightouterjoin B: $A \bowtie B$

```
2266 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}
```

\fullouterjoin A \fullouterjoin B: $A \bowtie B$

```
2267 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
```

2268

2.34 rmodell.sty

```
2269 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2270 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2271 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2272 Datenbanken.]
2273 \RequirePackage{soul}
```

Let-Abkürzungen

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

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

```
2274 \def\liPrimaer#1{\ul{#1}}
```

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

```
2275 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

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

```
2276 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2277 \ExplSyntaxOn
2278 \NewDocumentEnvironment { liRmodell }
2279 { +b }
2280 {
2281   \medskip
2282   {
2283     \linespread{2}
2284     \setlength{\parindent}{0pt}
2285     \li@Rmodell@Schrift#1
2286   }
2287   \medskip
2288 } {}
2289 \ExplSyntaxOff
```

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

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

```
2290 \def\liRelationMenge#1#2{
2291 \noindent
2292 #1 : \[ #2 ]\}
2293 \par
2294 }
```

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

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

```
2295 \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}
```

```
2296 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2297
```

2.35 sortieren.sty

2298 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2299 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
 2300 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}
}
```



2301 \RequirePackage{tikz}
 2302 \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.

```
2303 \def\liVertauschen#1{
2304   \directlua{
2305     local sortieren = require('lehramt-informatik-sortieren')
2306     sortieren('#1')
2307   }
2308 }
```

\liSortierPfeil

```
2309 \def\liSortierPfeil#1#2{
2310   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2311 }
```

\liSortierPfeilUnten

```
2312 \def\liSortierPfeilUnten#1#2{
2313   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2314 }
```

\liSortierMarkierung

```
2315 \def\liSortierMarkierung#1#2{\node[
2316   draw,
2317   very thick,
2318   fit=(reihe.#1) (reihe.#2),
2319   inner sep=0pt
2320 ] {}
2321 }

2322 \tikzset{
2323   li sortierung zahlenreihe/.style={
2324     draw,
2325     thin,
2326     font=\large,
2327     rectangle split horizontal,
2328     rectangle split,
2329   }
2330 }
```



```

2331 % https://tex.stackexchange.com/a/140895
2332 \RequirePackage{forest,xstring}
2333 \usetikzlibrary{calc}
2334
2335 \makeatletter
2336 \pgfmathdeclarefunction{strrepeat}{2}{%
2337   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2338   \let\pgfmathresult\pgfutil@empty
2339   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2340     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2341     \advance\pgfmath@count-1\relax
2342   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2343 \makeatother
2344
2345 \def\myNodes{}
2346
2347 \ExplSyntaxOn
2348 \newcommand*\sortList[1]{%
2349   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2350 \ExplSyntaxOff
2351
2352 \forestset{
2353   sort/.code={%
2354     \pgfmathparse{level()}>\forestSortLevel}%
2355     \ifnum\pgfmathresult=0
2356       \StrSubstitute{\forestov{content}}{ }{,}{\myList}%
2357       \sortList\myList
2358       \StrSubstitute{\myList}{,}{ }{\myList}%
2359       \pgfmathparse{strrepeat("1",level())}%
2360       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-!({\forestov{name}}$)
2361         (m\forestov{name}) {\myList}}%
2362       \pgfmathparse{level()}==\forestSortLevel}%
2363       \ifnum\pgfmathresult=1
2364         \forestOget{\forestov{@first}}{name}\forestFirst
2365         \forestOget{\forestov{@last}}{name}\forestLast
2366         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2367           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2368       \fi
2369       \ifnum\forestov{@parent}=0\else
2370         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2371       \fi
2372       \gappto\myNodes{;}%
2373     \fi}}
2374
2375 \forestset{sort level/.code=%
2376   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2377   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2378
2379

```

2.36 spalten.sty

```
2379 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2380 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2381 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2382 realisiert werden kann.]
2383 \RequirePackage{multicol}
```

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

```
2384 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2385
```

2.37 struktogramm.sty

```
2386 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2387 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2388 Paket struktex zum Setzen von Struktogrammen]
2389 \RequirePackage{struktex}
2390
```

2.38 syntax.sty

```
2391 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2392 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2393 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2394 \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
```

```
2395 \ExplSyntaxOn
2396 \directlua{
2397   syntax = require('lehramt-informatik-syntax')
2398   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2399   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2400   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2401   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2402   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2403   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2404 }
2405 \RequirePackage{hyperref}
2406 \RequirePackage{minted}
2407 % pygmentize -L styles
2408 \usemintedstyle{colorful}
2409 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2410 %\AfterEndEnvironment{minted}{\end{mdframed}}
2411 %\setminted{breaklines=true,linenos}
2412 \setminted{
2413   breaklines=true,
2414   linenos,
2415   fontsize=\footnotesize,
2416 }
```

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

Let-Abkürzung: `\let\j=\liJavaCode`

```
2417 \def\liJavaCode#1{\,\mintinline{java}|#1|\,}
```

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

```
2418 \def\liLatexCode#1{\mintinline{latex}|#1|}
```

```
2419 \def\li@GithubLink#1#2{
2420   \begin{flushright}
2421     \tiny
2422     Code-Beispiel~auf~Github~ansehen:~
2423     \href{#1}{\nolinkurl{#2}}
2424   \end{flushright}
2425 }
```

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

```
2426 \NewDocumentCommand{\liJavaDatei}{ O{firstline=3} m }{
2427   \inputminted[#1]{java}{
2428     \directlua{
2429       syntax.drucke_absoluten_pfad('#2', false)
2430     }
2431   }
2432   \li@GithubLink
```

```

2433     {\directlua{syntax.drucke_github_url('#2', false)}}
2434     {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2435 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2436 \NewDocumentCommand{\liJavaTestDatei}{ O{firstline=3} m }{
2437   \inputminted[#1]{java}{
2438     \directlua{
2439       syntax.drucke_absoluten_pfad('#2', true)
2440     }
2441   }
2442   \li@GithubLink
2443   {\directlua{syntax.drucke_github_url('#2', true)}}
2444   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2445 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([.
\liJavaExamen$1{$2}{$3}{$4}{$5}
2446 \NewDocumentCommand{\liJavaExamen}{ O{firstline=3} m m m m }{
2447   \inputminted[#1]{java}{
2448     \directlua{
2449       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2450     }
2451   }
2452
2453   \li@GithubLink
2454   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2455   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2456 }

\liAssemblerCode
2457 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2458 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2459   \inputminted{asm}{#1}
2460 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2461 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2462   \inputminted{componentpascal}{#1}
2463 }

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2464 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2465 \NewDocumentCommand{\liHaskellDatei}{ m }{
2466   \inputminted{haskell}{#1}
2467 }

2468 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2469 \def\liSqlCode#1{\mintinline{sql}|#1|}
2470

```

2.39 syntaxbaum.sty

```
2471 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2472 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2473 Syntaxbäumen mit Hilfe des Pakets tikz-qtree]
2474 \RequirePackage{tikz-qtree}
2475
2476 \tikzset{li parsetree/.style={
2477     every internal node/.style={
2478         draw,circle
2479     },
2480     every leaf node/.style={
2481         draw,rectangle
2482     },
2483 }
2484 }
2485
```

2.40 synthese-algorithmus.sty

```

2486 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2487 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2488 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2489 Relation in die 3. Normalform]

2490 \liLadePakete{normalformen,mathe,typographie}
2491 \ExplSyntaxOn

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

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

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

```

1. Kanonische Überdeckung

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

(a) Linksreduktion

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

(b) Rechtsreduktion

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

(c) Löschen leerer Klauseln

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

(d) Vereinigung

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

2. Relationsschemata formen

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

3. Schlüssel hinzufügen

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

4. Entfernung überflüssiger Teilschemata

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

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

```

2492 \def\liSyntheseUeberschrift#1{
2493   {
2494     \bfseries
2495     \sffamily
2496     \str_case:nn {#1} {
2497       {1} {Kanonische-Überdeckung}
2498       {1-1} {Linksreduktion}
2499       {1-2} {Rechtsreduktion}
2500       {1-3} {Löschen-leerer-Klauseln}
2501       {1-4} {Vereinigung}
2502       {2} {Relationsschemata-formen}
2503       {3} {Schlüssel-hinzufügen}
2504       {4} {Entfernung-überflüssiger-Teilschemata}
2505     }
2506   }
2507 }

```

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

```

2508 \def\li@synthese@erklaerung@texte#1{
2509   \str_case:nn {#1} {
2510     {1} {
2511       Die~kanonische-Überdeckung---also~die~kleinst~mögliche~noch~
2512       äquivalente~Menge~von~funktionalen~Abhängigkeiten~kann~in~vier~
2513       Schritten~erreicht~werden.
2514     }
2515     {1-1} {
2516       Führe~für~jede~funktionale~Anhängigkeit~
2517       $\alpha\rightarrow\beta$~in~F$~die~Linksreduktion~durch,~
2518       überprüfe~also~für~alle~
2519       $A\in\alpha$,~ob~$A$~überflüssig~ist,~d.~h.~ob~
2520       $\beta\subseteqq\liAttributHuelle{F,\alpha\cup A}$.
2521     }
2522     {1-2} {
2523       Führe~für~jede~(verbliebene)~funktionale~Abhängigkeit~$\alpha\rightarrow\beta$~die~Rechtsreduktion~durch,~überprüfe~also~für~
2524       alle~$B\in\beta$,~ob~$B\in\liAttributHuelle{F,\alpha\rightarrow\beta\cup B}$,~
2525       $\alpha\rightarrow\beta\cup B$~gilt.~In~diesem~Fall~ist~B~auf~der~rechten~Seite~
2526       überflüssig~und~kann~eliminiert~werden,~\dh~$\alpha\rightarrow\beta$~wird~durch~$\alpha\rightarrow\beta\cup B$~
2527       ersetzt.
2528     }
2529     {1-3} {
2530       Entferne~die~funktionalen~Abhängigkeiten~der~Form~$\alpha\rightarrow\emptyset$,~die~im~2.~Schritt~möglicherweise~
2531       entstanden~sind.
2532     }
2533     {1-4} {
2534       Fasse~mittels~der~Vereinigungsregel~funktionale~Abhängigkeiten~
2535       der~Form~$\alpha\rightarrow\beta\sb{1},\dots,\alpha\rightarrow\beta\sb{n}$,~so~dass~$\alpha\rightarrow\beta\sb{1}\cup\dots\cup\beta\sb{n}$~verbleibt.
2536     }
2537     % Kemper Seite 197
2538     {2} {
2539       Erzeuge~für~jede~funktionale~Abhängigkeit~$\alpha\rightarrow\beta\in F\sb{c}$~ein~Relationenschema~$\mathcal{R}\sb{\alpha}$~
2540       :=~$\alpha\cup\beta$.
2541     }
2542     {3} {
2543       Falls~eines~der~in~Schritt~2.~erzeugten~Schemata~$\mathcal{R}\sb{\alpha}$~
2544       einen~Schlüsselkandidaten~von~$\mathcal{R}$~bezüglich~$F\sb{c}$~

```



```

2552 enthält,~sind~wir~fertig,~sonst~wähle~einen~Schlüsselkandidaten~
2553  $\mathcal{K} \sim \text{subseq} \mathcal{R}$ ~aus~und~definiere~folgendes~
2554 zusätzliche~Schema:~ $\mathcal{R} \setminus \mathcal{K} \sim \mathcal{K}$ ~
2555 und~ $\mathcal{F} \setminus \mathcal{K} \sim \emptyset$ 
2556 }
2557 {4} {
2558 Eliminiere~diejenigen~Schemata~ $\mathcal{R} \setminus \alpha$ ~,~die~in~einem~
2559 anderen~Relationenschema~ $\mathcal{R} \setminus \alpha'$ ~enthalten~sind,~d.h.~
2560  $\mathcal{R} \setminus \alpha \sim \text{subseq} \mathcal{R} \setminus \alpha'$ ~.
2561 }
2562 }
2563 }
2564 \def\liSyntheseErklaerung#1{
2565 {
2566 \itshape
2567 \footnotesize
2568 \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
2569 }
2570 }

```

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

```

2571 \def\liSyntheseUeberErklaerung#1{
2572 \liSyntheseUeberschrift{#1}\par
2573 \liSyntheseErklaerung{#1}
2574 }

```

```

2575 \ExplSyntaxOff
2576

```

2.41 tabelle.sty

2577 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2578 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]

2579 \RequirePackage{tabularx}

2580

2.42 typographie.sty

```
2581 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2582 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
2583 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
2584 formatierung.sty definiert.]
```

```
2585 \ExplSyntaxOn
```

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

```
2586 \RequirePackage{fontawesome}
```

```
\liErledigt \liErledigt: ☑
```

```
2587 \let\liErledigt=\faCheckSquareO
```

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

```
2588 \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 —

```
2589 \def\liParagraphMitLinien#1{
2590   \noindent
2591   \vrule height 2pt depth -1.6pt width 0.4cm
2592   \enspace
2593   #1
2594   \enspace
2595   \leaders\vrule height 2pt depth -1.6pt \hfill \null
2596   \par
2597   \medskip
2598 }
```

```
2599 \ExplSyntaxOff
```

```
2600
```

2.43 uml.sty

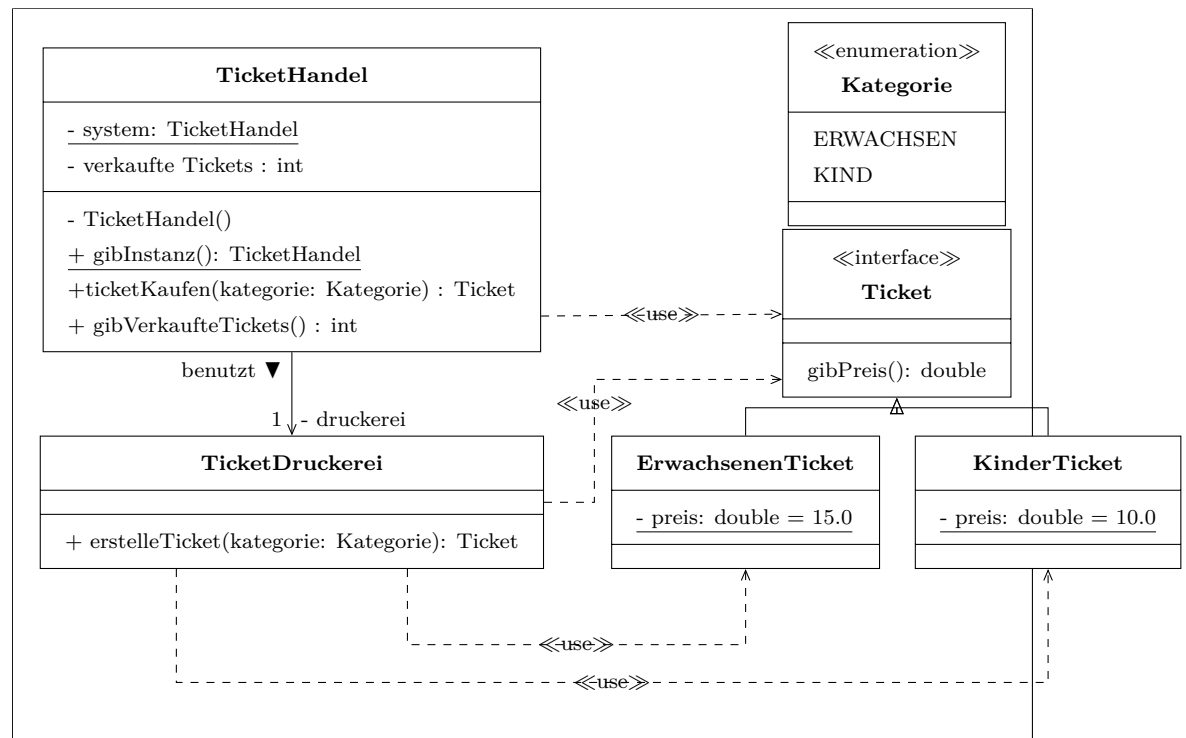
```

2601 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2602 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
2603 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
2604 Erweiterung bereitstellt]

2605 \RequirePackage{tikz-uml}
2606 \RequirePackage{tikz-uml-activity}
2607 % Not compatible with wasysym
2608 %\RequirePackage{mathabx}
2609 \RequirePackage{wasysym}
2610 \usetikzlibrary{positioning}

2611 \tikzumlset{
2612   fill class=white!0,
2613   font=\footnotesize,
2614   fill object=white!0,
2615   fill note=white!0,
2616   fill state=white!0,
2617   % Use case
2618   fill usecase=white!0,
2619   fill system=white!0,
2620 }

```



```

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

```

```

2621 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
2622   \def\@liDirLeft{}
2623   \def\@liDirRight{ \RIGHTarrow}
2624   \pgfkeys{/lese/dir/.is choice}
2625   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
2626   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
2627   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{ \LEFTarrow }}}
2628   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
2629
2630   \def\@liPos{above}
2631   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
2632

```

```

2633 \def\@liDistance{0cm}
2634 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
2635
2636 \pgfkeys{/lese/.cd,#1}
2637
2638 \node[\@liPos = \@liDistance of #2-middle] {
2639   \@liDirLeft{\footnotesize#2}\@liDirRight
2640 };
2641 }
2642

```

2.44 vollstaendige-induktion.sty

2643 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2644 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01
 2645 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
 2646 Überschriften für die einzelnen Schritte]

Let-Abkürzungen

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

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

Lade häufig benötigte Pakete

2647 \RequirePackage{lehramt-informatik-typographie}
 2648 \RequirePackage{lehramt-informatik-mathe}
 2649 \RequirePackage{lehramt-informatik-syntax}
 2650 \ExplSyntaxOn

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

Let-Abkürzung: \let\m=\liInduktionMarkierung

2651 \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

```
2652 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
2653 \def\liInduktionAnfang{
```

```
2654   \liPseudoUeberschrift{Induktionsanfang}
```

```
2655
```

```
2656   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
2657   \liParagraphMitLinien{
```

```
2658     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
```

```
2659   }
```

```
2660 }
```

\liInduktionVoraussetzung

```
2661 \def\liInduktionVoraussetzung{
```

```
2662   \liPseudoUeberschrift{Induktionsvoraussetzung}
```

```
2663
```

```
2664   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
2665   \liParagraphMitLinien{
```

```
2666     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ ~.
```

```
2667   }
```

```
2668 }
```

\liInduktionSchritt

```
2669 \def\liInduktionSchritt{
```

```
2670   \liPseudoUeberschrift{Induktionsschritt}
```

```
2671
```

```
2672   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
2673   \liParagraphMitLinien{
```

```
2674     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
```

```
2675     auch~ $A(n=k+1)$ ~wahr~sein~muss.
```

```
2676   }
```

```
2677 }
```

```
2678 \ExplSyntaxOff
```

```
2679
```

2.45 wasserfall.sty

```
2680 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2681 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
2682 \RequirePackage{tikz}
2683 \tikzset{wasserfall/.style={
2684   >=stealth,
2685   node distance = 2mm and -8mm,
2686   start chain = A going below right,
2687   every node/.style = {
2688     draw,
2689     text width=24mm,
2690     minimum height=12mm,
2691     align=center,
2692     inner sep=1mm,
2693     fill=white,
2694     drop shadow={fill=black},
2695     on chain=A
2696   },
2697 }}
2698 \usetikzlibrary{chains,positioning,shadows}
2699
```


2.46 wpkalkuel.sty

```
2700 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2701 \ProvidesPackage{lehramt-informatik-wpkalkuel}[2020/06/13]
```

Let-Abkürzungen

```
\let\wp=\liWpKalkuel
\let\equivalent=\liWpEquivalent
\let\erklaerung=\liWpErklaerung
```

```
2702 \RequirePackage{amsmath}
```

```
2703 \ExplSyntaxOn
```

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

```
2704 \def\liWpKalkuelOhneMathe#1#2{
2705   \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)
2706 }
2707 \def\liWpKalkuel#1#2{
2708   \ifmmode
2709     \liWpKalkuelOhneMathe{#1}{#2}
2710   \else
2711     $\liWpKalkuelOhneMathe{#1}{#2}$
2712   \fi
2713 }
```

```
\MatheEnv
```

```
2714 \def\MatheEnv#1{
2715   \medskip
2716
2717   \hspace{1em}#1
2718
2719   \medskip
2720 }
```

```
\Mathe
```

```
2721 \def\Mathe#1{
2722   \MatheEnv{#1$}
2723 }
```

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

```
2724 \def\liWpEquivalent#1{
2725   \MatheEnv{$\equiv$\hspace{1em}$#1$}
2726 }
```

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

```
2727 \newlength{\@Skip@Erklaerung@Reset}
2728 \def\liWpErklaerung#1{
2729   \setlength{\@Skip@Erklaerung@Reset}{\leftskip}
2730   \setlength{\leftskip}{0.5cm}
2731
2732   \par
2733   \noindent
2734   {
2735     \scriptsize
2736     #1
2737   }
2738   \par
2739
2740   \setlength{\leftskip}{\@Skip@Erklaerung@Reset}
2741 }
```

\liWpErklaerungVerzweigung

```

2742 \def\liWpErklaerungVerzweigung{
2743   $\liWpKalkuelOhneMathe{if~\{-b~\}~then~\{-a1~\}~else~\{-a2~\}}{Q}
2744   \equiv
2745   (b \land \liWpKalkuelOhneMathe{a1}{Q})
2746   \lor
2747   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
2748 }

2749 \ExplSyntaxOff

2750

```

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 2517, 2519, 2520,	\bowtie 2262, 2265, 2266, 2267
\, 331, 386,	2523, 2525, 2526,	\Box 159
1135, 1137, 1638,	2527, 2528, 2529,	\boxtimes 463
1639, 1640, 1947, 2417	2533, 2539, 2540,	
\@Skip@Erklaerung@Reset	2545, 2546, 2547,	
... 2727, 2729, 2740	2550, 2558, 2559, 2560	C
\@afterheading 1605	\arabic 1198, 2136, 2141,	\c 1281, 1282
\@afterindentfalse . 1605	2146, 2152, 2158, 2164	\cdot 1682, 1735, 1746
\@liDirLeft 2622, 2627, 2639	\arraystretch 1864	\centerline 1312, 2215, 2237, 2252
\@liDirRight 2623, 2625,		\chapter 1188, 1189
2626, 2627, 2628, 2639	B	\char 1512
\@liDistance 2633, 2634, 2638	\BeforeBeginEnvironment	\clearpage 1506
\@liPos .. 2630, 2631, 2638 2409	\cline 596
\\ 596, 615, 616,	\begin 613,	\clist 226, 270,
619, 620, 623, 624,	658, 673, 701, 725,	271, 284, 288, 2349
703, 704, 705, 794,	763, 786, 801, 821,	\columnbreak 2384
823, 825, 847, 856,	831, 843, 863, 895,	\cs 287, 306, 330,
901, 943, 944, 945,	912, 941, 966, 987,	331, 368, 380, 1588
950, 951, 952, 972,	1002, 1114, 1118,	\csname 1266, 1269
1512, 1869, 1926, 1929	1204, 1279, 1307,	\cup 1093, 2526, 2540, 2547
\{ 207, 1070, 1080,	1316, 1323, 1431,	
1092, 1093, 1098,	1504, 1556, 1561,	D
1112, 1134, 1353,	1569, 1594, 1599,	\DeclareMathSymbol ..
1914, 1937, 2292, 2743	1610, 1620, 1624, 1812, 1813
\} 207, 1070, 1080,	1707, 1711, 1727,	\DecoINERT 2139, 2212, 2213, 2235
1092, 1093, 1100,	1750, 1773, 1788,	\DecoINERTwithPivot .
1118, 1138, 1354,	1867, 1868, 1918, 2155, 2234
1914, 1941, 2292, 2743	2056, 2076, 2217,	\DecoLEFT 2134, 2233
_ 27, 35, 44, 46, 287, 306,	2239, 2253, 2409, 2420	\DecoLEFTwithPivot ..
330, 331, 345, 346,	\begingroup 1505, 1916, 2337 2149, 2211
352, 355, 358, 368, 380	\beschriftung 1544, 1548, 1552, 1554	\DecoRIGHT 2144, 2236, 2246
	\beta 2517,	\DecoRIGHTwithPivot .
A	2520, 2524, 2525, 2161, 2214
\addbibresource 1482, 1483, 1484,	2526, 2529, 2539,	\definecolor 1186
1485, 1486, 1487,	2540, 2541, 2546, 2547	\delta 65, 107, 165, 207, 1084
1488, 1489, 1490, 1491	\bf 2114, 2115, 2116	\dh 1640, 2528
\advance 2341	\bfseries .. 475, 1188,	\directlua 58, 137, 195, 200,
\AfterEndEnvironment 2410	1190, 2114, 2120,	1064, 1079, 1099,
\allsectionsfont ... 1184	2122, 2124, 2125, 2494	1107, 1115, 1121,
\Alpha 1198	\bigskip 48, 364,	1899, 1904, 1932,
\alph 1198, 1199	598, 603, 1514, 1842	1939, 1944, 2304,
	\bool 309, 332	

	2396, 2428, 2433, 2434, 2438, 2443, 2444, 2448, 2454, 2455				
\do	2135, 2140, 2145, 2150, 2156, 2162				
\dots	506, 510, 1353, 2063, 2539, 2540				
\DOWNarrow	2626				
\draw	1286, 1289, 1292, 2011, 2310, 2313				
E					
\edef	1277, 2228, 2232, 2244, 2245				
\else	570, 578, 586, 1074, 1088, 1126, 1552, 1655, 1665, 1675, 1689, 1910, 2006, 2174, 2367, 2369, 2710				
\emph	1034, 1334, 1363, 1365, 1511				
\empty	1552				
\emptyset	1856, 2534, 2555				
\end	652, 669, 694, 722, 756, 779, 798, 814, 828, 836, 859, 888, 909, 934, 959, 979, 999, 1017, 1116, 1118, 1206, 1295, 1319, 1320, 1327, 1436, 1509, 1558, 1566, 1585, 1601, 1602, 1616, 1627, 1628, 1709, 1722, 1747, 1770, 1785, 1795, 1871, 1872, 1920, 2066, 2086, 2217, 2239, 2253, 2410, 2424				
\endcsname	1266, 1269				
\endgroup	1508, 1921, 2342				
\enspace	2592, 2594				
environments:					
liAdditum	1559				
liAHuelle	1915				
liAntwort	1530				
liDiagramm	1618				
liEinbettung	1529				
liExkurs	1567				
liGraphenFormat	1258				
liKasten	1203				
liKontrollflussgraph	1430				
liLernkartei	1608				
liProduktionsRegeln	1108				
liProjektSprache	1528				
liQuellen	1588				
liRelationenSchemaFormat	2296				
liRmodell	2276				
liUebergangsTabelle	1864				
\equiv	2725, 2744				
\erzeuge@tiefgestellt	1079, 1080, 1084				
\expandafter	1266, 2170, 2172, 2173, 2174, 2182, 2340				
\ExplSyntaxOff	50, 92, 134, 139, 192, 197, 202, 393, 528, 550, 565, 1058, 1140, 1175, 1451, 1466, 1630, 1705, 1892, 2039, 2289, 2350, 2468, 2575, 2599, 2678, 2749				
\ExplSyntaxOn	22, 61, 102, 135, 160, 193, 198, 223, 470, 534, 551, 1031, 1130, 1152, 1444, 1456, 1531, 1644, 1874, 2020, 2277, 2347, 2395, 2491, 2585, 2650, 2703				
F					
\faCheckSquare	2587, 2588				
\faCircleThin	1050				
\faGg	1044				
\fancyfoot	1210, 1211, 1212, 1460, 1461, 1462, 1463				
\fancyhead	1209, 1457, 1458, 1459				
\faSquare	1038				
\fi	572, 580, 588, 1076, 1090, 1128, 1555, 1657, 1667, 1677, 1691, 1912, 2008, 2175, 2367, 2368, 2371, 2373, 2712				
\fontspec	1184				
\footcite	670, 692, 732, 755, 778, 860, 887, 933, 1332, 1335, 1342, 1347, 1352, 1356, 1362, 1367, 1471, 1723, 1724, 1879, 2069				
\footnote	1632, 1636				
\footnotesize	147, 341, 424, 520, 1055, 1527, 1580, 1595, 1611, 1823, 1917, 2024, 2033, 2276, 2415, 2567, 2613, 2639				
\footrulewidth	1215, 1465				
\foreach	1281, 1284, 1291				
\forestFirst	2364, 2367				
\forestLast	2365, 2367				
\forestOget	2364, 2365				
\forestOnes	2377				
\forestOv	2366, 2367, 2370				
\forestov	2356, 2360, 2361, 2364, 2365, 2366, 2367, 2369, 2370				
\forestset	2352, 2375				
\forestSortLevel	2354, 2362, 2376, 2377				
\frac	1684, 1715, 1746, 1761				
\fullouterjoin	2267				
G					
\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				
\Gamma	106, 164, 207, 1093				
\gappto	2372				
\geq	1360, 2052, 2057, 2073, 2077				
H					
\hbox	2262				
\headrulewidth	1214, 1464				
\hfill	1731, 1737, 1742, 2595				
\hinweis	1527				
\hline	1869				
\href	370, 1148, 1636, 1799, 2423				
\hspace	2014, 2717, 2725				
\ht	2263				
I					
\i	1291, 1292				
\ifcase	2171				
\ifmode	568, 576, 584, 1072, 1086, 1124, 1653, 1663, 1673, 1687, 1908, 2004, 2708				
\IfNoValueTF	1562, 1632, 1636				
\ifnum	2098, 2339, 2355, 2363, 2369				
\ifx	1552, 2367				
\in	494, 600, 1360, 1697, 1700, 1703, 1729, 1735, 1740, 2052, 2063, 2073, 2083, 2517, 2519, 2525, 2546, 2666				
\inhaltsverzeichnis	1503				
\input	4, 7, 10, 13, 16, 394				
\inputminted	2427, 2437, 2447, 2459, 2462, 2466				
\int	2349				
\item	463, 464, 675, 679, 684, 689, 726, 735, 740,				

748, 802, 807, 811, 832, 864, 869, 876, 884, 913, 918, 922, 927, 1003, 1008, 1013, 1317, 1318, 1588, 1592, 1712, 1715, 1719, 1728, 1734, 1739, 1751, 1755, 1759, 1763, 1767, 1774, 1778, 1782, 2057, 2060, 2063, 2077, 2080, 2083	\itshape 519, 2566	\LehramtInformatikAutorEmail\liAttributHuelleOhneMathe 1463 1906, 1909, 1911, 1925
		\LehramtInformatikAutorName \liAttributMenge 1462 . . . 1914, 1925, 1928
		\LehramtInformatikGitBranch \liAufgabe 3 373, 2403 \liAufgabenTitel 23
		\LehramtInformatikGithubCodeRepositoryAusdruck 1131 2402 \liAutomat 61
		\LehramtInformatikGithubDomaine\liAutomatenKante 93 2399 \liBandAlphabet 1093
		\LehramtInformatikGithubRawDomaineBedingung 1439 371, 2400 \liBedingungDrei
		\LehramtInformatikGithubTexRepo . . . 1702, 1742, 1782 372, 2401 \liBedingungEins
		\LehramtInformatikRepository . . . 1696, 1731, 1774 4, \liBedingungFalsch . 1441
		7, 10, 13, 16, 1482, \liBedingungWahr . . . 1440
		1483, 1484, 1485, \liBedingungZwei
		1486, 1487, 1488, . . . 1699, 1737, 1778
		1489, 1490, 1491, 2398 \liBeschriftung . . . 1519
		\LehramtInformatikTitel \liChomskyErklaerung 1458 485, 526
		\leq . . . 1746, 2060, 2080 \liChomskyUeberErklaerung \let 1082, 524
		1083, 1506, 2211, \liChomskyUeberschrift 2212, 2213, 2214, 473, 525
		2227, 2229, 2230, \liCpmEreignis 534
		2231, 2233, 2234, \liCpmFruehesterI . . . 591
		2235, 2236, 2246, \liCpmSpaetesterI . . . 590
		2338, 2376, 2377, 2587 \liCpmVon 574
		\li@chomsky@erklaerung@texte\liCpmVonOhneMathe 485, 521 574, 577, 579
		\li@fussnote@text 1820, \liCpmVonZu 566
		1826, 1830, 1834, 1838 \liCpmVonZuOhneMathe 566, 569, 571
		\li@GithubLink 2419, 2432, 2442, 2453 \liCpmVorgang 551
		\li@mget . 1268, 1272, 1292 \liCpmZu 582
		\li@minc 1271, 1293 \liCpmZuOhneMathe 582, 585, 587
		\li@mset 1265, 1273, 1282, 1285 liDiagramm (environ- ment) 1618
		\li@numdiscs 1277, 1286, 1292 liEinbettung (environ- 2276, 2285, 2295 ment) 1529
		\li@Rmodell@Schrift 1278, 1291 \liEntwurfs 982
		\li@sequence . . 1278, 1291 \liEntwurfsAbstrakteFabrik 2508, 2568 \liEntwurfsAbstrakteFabrikUml 1107 612, 655
		\liAbleitung 1559 \liEntwurfsAdapter . . 696
		liAdditum (environment) 1915 \liEntwurfsAdapterAkteure 1092 672, 698
		liAHuelle (environment) 1530 \liEntwurfsAdapterUml 1438 657, 697
		\liAlphabet 2457 \liEntwurfsBeobachter 758
		liAntwort (environment) 2458 \liEntwurfsBeobachterAkteure 2295 700, 759
		\liAnweisung 1906, 2520, 2525 \liEntwurfsDekorierer 781
		\liAssemblerCode . . . 1923 \liEntwurfsDekoriererAkteure 1923 783
		\liAssemblerDatei . . 1923 \liEntwurfsDekoriererUml 1923 762, 782
		\liAttribut 1923
		\liAttributHuelle 1923
		\liAttributHuelleLinksReduktion
		\liAttributHuelleLinksReduktion

\liEntwurfsEinfacheFabrik	\liFussnoteDreiText	171, 172, 173, 177, 1070, 1121, 1160, 1161
..... 816 1833, 1851	
\liEntwurfsEinfacheFabrikAkteure	\liFussnoteEinsText	\liMengeOhneMathe ...
..... 800, 818 1825, 1845	... 1070, 1073, 1075
\liEntwurfsEinfacheFabrikUml	\liFussnoteLink	\liMinimierungErklaerung
..... 785, 817 1635 1878
\liEntwurfsEinzelstueck	\liFussnoten	\liMinispracheDatei 2461
..... 838	\liFussnoteUrl . 980, 1631	\linespread 2283
\liEntwurfsEinzelstueckAkteure	\liFussnoteVierText	\liNichtsZuTun 2588
..... 830, 840 1837, 1854	\liO 1669, 1697
\liEntwurfsEinzelstueckUml	\liFussnoteZweiText	\liOmega 1659, 1703
..... 820, 839 1829, 1848	\liOmegaOhneMathe ...
\liEntwurfsErbauer .. 890	\liGrammatik 1152	... 1659, 1664, 1666
\liEntwurfsErbauerAkteure	liGraphenFormat (envi- ronment) 1258	\liOOhneMathe
..... 862, 892	\liHanoi 1265	... 1669, 1674, 1676
\liEntwurfsErbauerUml	\liHaskellCode 2464	\liParagraphMitLinien
..... 842, 891	\liHaskellDatei 2465	. 521, 1880, 2568,
\liEntwurfsFabrikmethode	\liInduktionAnfang . 2653	2589, 2657, 2665, 2673
..... 936, 961	\liInduktionErklaerung	\liPetriErreichKnotenDrei
\liEntwurfsFabrikmethodeAkteure 2652 2013
..... 911, 938	\liInduktionMarkierung	\liPetriErreichTransition
\liEntwurfsFabrikmethodeUml 2651 2010
..... 894, 937	\liInduktionSchritt 2669	\liPetriSetzeSchluessel
\liEntwurfsKompositum 961	\liInduktionVoraussetzung 1955
\liEntwurfsKompositumAkteure 2661	\liPetriTransitionsName
..... 963	\liJavaCode 2417 2002, 2014
\liEntwurfsKompositumUml	\liJavaDatei 2426	\liPetriTransitionsNameOhneMathe
..... 940, 962	\liJavaExamen 2446	... 2002, 2005, 2007
\liEntwurfsModellPraesentationSteuerung	\liJavaTestDatei ... 2436	\liPetriTransPfeile 2014
..... 982	liKasten (environment) 1203	\liPolynomiellReduzierbar
\liEntwurfsModellPraesentationSteuerungAkteure	\liKellerAutomat 102 1322
..... 984	\liKellerKante 140	\liPotenzmenge
\liEntwurfsModellPraesentationSteuerungUml	\liKellerUebergang 1079, 1083, 2025
..... 965, 983 135, 141	\liPotenzmengeOhneMathe
\liEntwurfsZustand . 1019	\liKontrollCode 1442	... 1080, 1081, 1082
\liEntwurfsZustandAkteure	liKontrollflussgraph	\liPrimaer 2274
..... 1001, 1021	(environment) 1430	\liProblemBeschreibung
\liEntwurfsZustandUml	\liKontrollKnotenPfad 1306
..... 986, 1020 1444	\liProblemClique ... 1329
\liEpsilon 1078	\liKontrollTextzeileKnoten	\liProblemName
\liErAttribute 1443, 1448 1305, 1312,
... 1034, 1048, 1050	\liKurzeTabellenLinie 596	1324, 1326, 1339,
\liErDatenbankName . 1053	\liLadeAllePakete ... 228	1350, 1351, 1359, 1360
\liErEntity 1032, 1036, 1038	\liLadePakete	\liProblemSat 1358
\liErledigt 2587	. 54, 57, 224, 229,	\liProblemSubsetSum .
\liErMpAttribute ... 1047	472, 533, 1302, 1349, 1358
\liErMpEntity 1035	1818, 1898, 2019, 2490	\liProblemVertexCover
\liErMpRelationship 1041	\liLatexCode 2418 1329, 1337
\liErRelationship ...	\liLeereZelle 1856	\liProduktionen 1120, 1162
... 1033, 1042, 1044	liLernkartei (environ- ment) 1608	liProduktionsRegeln
\liExamensAufgabe 6	\liMasterExkurs 1787	(environment) 1108
\liExamensAufgabeA ... 15	\liMasterFaelle 1726, 1794	liProjektSprache (envi- ronment) 1528
\liExamensAufgabeTA .. 12	\liMasterFallRechnung	\liPseudoUeberschrift
\liExamensAufgabeTTA .. 9 1772 1513,
liExkurs (environment) 1567	\liMasterVariablen ..	1563, 1564, 1866,
\liFalsch 464 1706, 1789	1876, 2654, 2662, 2670
\liFlaci 1141	\liMasterVariablenDeklaration	\liPumpingKontextfrei
\liFremd 2275 1749 2071
\liFunktionaleAbhaengigkeit	\liMasterWolframLink 1797	\liPumpingRegulaer . 2050
..... 1931	\liMenge ... 71, 72, 74,	liQuellen (environment)
\liFunktionaleAbhaengigkeiten	113, 114, 115, 119, 1588
..... 1934		
\liFussnote ... 1819, 1821		

<code>\liRekursionsGleichung</code>	<code>\liWpKalkuelOhneMathe</code>	N
..... 1693 , 1753 2704 , 2709 ,	<code>\NeedsTeXFormat</code>
<code>\liRelation</code>	2711 , 2743 , 2745 , 2747	1 , 19 , 52 , 220 , 398 ,
<code>liRelationenSchemaFormat</code>	<code>\liZustandsBuchstabe</code>	459 , 466 , 530 , 593 ,
(environment) 2296 1094 ,	608 , 1024 , 1060 ,
<code>\liRelationMenge</code> ... 2290	1103 , 1105 , 1125 , 1127	1177 , 1217 , 1226 ,
<code>\liRichtig</code>	<code>\liZustandsBuchstabeGross</code>	1231 , 1260 , 1298 ,
463	... 1095 , 1104 , 1106	1370 , 1453 , 1468 ,
<code>liRmodell (environment)</code>	<code>\liZustandsmenge</code> ... 1082	1474 , 1496 , 1642 ,
..... 2276	<code>\liZustandsmengeNr</code> ..	1802 , 1815 , 1894 ,
<code>\liRundeKlammer</code> . 1646 , 1096 , 2034	1950 , 2016 , 2041 ,
1650 , 1660 , 1670 , 1684	<code>\liZustandsmengeNrGross</code>	2046 , 2090 , 2256 ,
<code>\liSetzeAufgabenTitel</code> . 25 1104	2269 , 2298 , 2379 ,
<code>\liSortierMarkierung</code> 2315	<code>\liZustandsMengenSammlung</code>	2386 , 2391 , 2471 ,
<code>\liSortierPfeil</code> 2309 2021	2486 , 2577 , 2581 ,
<code>\liSortierPfeilUnten</code> 2312	<code>\liZustandsMengenSammlungNr</code>	2601 , 2643 , 2680 , 2700
<code>\liSpaltenUmbruch</code> .. 2384 2030	<code>\neg</code>
<code>\liSqlCode</code>	<code>\liZustandsmengeOhneMathe</code> 2747
2469 1082	<code>\negthinspace</code>
<code>\listen@punkt</code> .. 1588 , 1600	<code>\liZustandsname</code> 1105 1647
<code>\liStrich</code>	<code>\liZustandsnameGross</code>	<code>\newcounter</code> ... 2110 , 2111
1304	... 1106 , 2022 , 2031	<code>\NewDocumentCommand</code> .
<code>\liSyntheseErklaerung</code> 1123	62 , 103 , 140 , 161 ,
..... 2508 , 2573	<code>\liZustandsnameTiefgestellt</code>	203 , 224 , 535 , 552 ,
<code>\liSyntheseUeberErklaerung</code> 1858	597 , 602 , 1131 ,
..... 2571	<code>\liZustandsPaar</code> 1857 , 1860 , 1861	1153 , 1322 , 1445 ,
<code>\liSyntheseUeberschrift</code>	... 2263	1472 , 1631 , 1635 ,
..... 2492 , 2572	<code>\llap</code>	1934 , 1943 , 2010 ,
<code>\liT</code> 1679 , 1694 , 1708 , 1769	<code>\log</code>	2426 , 2436 , 2446 ,
<code>\liTeilen</code>	1697 ,	2458 , 2461 , 2465 , 2621
1903	1700 , 1703 , 1729 , 1735	<code>\NewDocumentEnvironment</code>
<code>\literatur</code> 1470 , 1494	<code>\loop</code>	1108 , 1203 , 1258 ,
<code>\liTheta</code>	2096	1430 , 1528 , 1529 ,
1649 ,	<code>\lor</code>	1532 , 1559 , 1567 ,
1700 , 1729 , 1735 , 1740	2746	1589 , 1608 , 1618 ,
<code>\liThetaOhneMathe</code> ...	<code>\ltimes</code>	1865 , 1915 , 2278 , 2296
... 1649 , 1654 , 1656		<code>\newlength</code>
<code>\liTOhneMathe</code>	M 2727
... 1679 , 1688 , 1690	<code>\makeatletter</code> .. 1604 , 2335	<code>\node</code>
<code>\liTuringKante</code>	<code>\makeatother</code> .. 1606 , 2343	548 , 1438 ,
203	<code>\marginpar</code>	1443 , 2136 , 2141 ,
<code>\liTuringLeerzeichen</code>	1037 , 1043 , 1049 , 1511	2146 , 2152 , 2158 ,
..... 159 , 167	<code>\mathbb</code> .. 1360 , 2083 , 2666	2164 , 2315 , 2360 , 2638
<code>\liTuringMaschine</code> ... 160	<code>\mathbin</code> . 2265 , 2266 , 2267	<code>\noexpand</code>
<code>\liTuringUeberfuehrung</code>	<code>\mathcal</code> .. 1670 , 2546 ,	2224 ,
..... 206	2551 , 2553 , 2554 , 2555	2225 , 2226 , 2245 , 2360
<code>\liTuringUebergange</code>	<code>\Mathe</code>	<code>\noindent</code> 343 , 599 , 604 ,
..... 198 , 204	2721	1111 , 1515 , 1517 ,
<code>\liTuringUebergangZelle</code>	<code>\MatheEnv</code> 2714 , 2722 , 2725	1521 , 1525 , 1553 ,
..... 193	<code>\mathord</code>	1581 , 1583 , 1596 ,
<code>\liUeberfuehrungsFunktion</code>	1812 , 1813	1612 , 1614 , 1622 ,
..... 1084	<code>\mdfsetup</code>	1791 , 1844 , 1847 ,
<code>\liUeberfuehrungsFunktionOhneMathe</code>	1537 , 1541 , 1545 , 1549	1850 , 1853 , 1936 ,
... 1084 , 1087 , 1089	<code>\medskip</code>	1941 , 2291 , 2590 , 2733
<code>liUebergangsTabelle</code>	1314 , 1522 , 1524 ,	<code>\nolinkurl</code>
(environment) 1864	1598 , 1626 , 2281 ,	2423
<code>\liUeberschriftDreiecksTabelle</code>	2287 , 2597 , 2715 , 2719	<code>\normalsize</code>
..... 1874	<code>\memph</code>	1190
<code>\liUmlLeserichtung</code> . 2621	1511	<code>\notin</code>
<code>\liVertauschen</code> 2303	<code>\mintinline</code> 2417 ,	605
<code>\liWortInSprache</code> 597	2418 , 2457 , 2464 , 2469	<code>\null</code>
<code>\liWortNichtInSprache</code> 602	<code>\mkern</code> ... 2265 , 2266 , 2267	2595
<code>\liWpEquivalent</code> 2724	<code>\mlq</code>	O
<code>\liWpErklaerung</code> 2727	1810 , 1812	<code>\o@join</code>
<code>\liWpErklaerungVerzweigung</code>	<code>\mrq</code>	2262 , 2265 , 2266 , 2267
..... 2742	1810 , 1813	<code>\Omega</code>
<code>\liWpKalkuel</code>	<code>\msg</code>	1660
2704	39 , 391	<code>\omega</code> 2052 , 2053 , 2073 , 2074
	<code>\myList</code>	<code>\or</code>
	2356 , 2357 , 2358 , 2361	2173
	<code>\myNodes</code>	P
	2345 ,	<code>\pagestyle</code>
	2360 , 2366 , 2370 , 2372	1213

<code>\par</code> 342, 363, 525, 1142, 1150, 1520, 1582, 1605, 1613, 1935, 1938, 1940, 2215, 2237, 2252, 2293, 2572, 2596, 2732, 2738	<code>\QSinitialize</code> 2095, 2207, 2242	<code>\romannumeral</code> 2182
<code>\paragraph</code> 1190	<code>\QSIr</code> . 2178, 2184, 2192, 2212, 2226, 2231, 2234	<code>\rtimes</code> 1805
<code>\parindent</code> 2284	<code>\QSIrr</code> 2213, 2226, 2227, 2235	<code>\rule</code> 2215, 2237, 2252, 2263
<code>\path</code> 94, 141, 204, 563	<code>\QSLr</code> 2184, 2191, 2202, 2203, 2211, 2224, 2229, 2233	S
<code>\pgfkeys</code> .. 1966, 2624, 2625, 2626, 2627, 2628, 2631, 2634, 2636	<code>\QSpivotStep</code> 2097, 2207, 2211, 2222	<code>\sb</code> 67, 77, 79, 108, 166, 501, 502, 506, 509, 510, 511, 1125, 1127, 1697, 1700, 1703, 1729, 1735, 1884, 2002, 2011, 2539, 2540, 2541, 2546, 2550, 2551, 2554, 2555, 2558, 2559, 2560
<code>\pgfmath@count</code> 2337, 2339, 2341	<code>\QSR</code> 2184	<code>\scriptscriptstyle</code> 566, 574, 582
<code>\pgfmath@smuggleone</code> 2342	<code>\QSRr</code> 2193, 2214, 2225, 2230, 2236, 2245, 2246, 2247	<code>\scriptsize</code> 1144, 1383, 1390, 1396, 1458, 1459, 1462, 1463, 2652, 2705, 2735
<code>\pgfmathdeclarefunction</code> 2336	<code>\QSSortStep</code> 2099, 2207, 2223, 2224	<code>\section</code> 46
<code>\pgfmathhint</code> 2337	<code>\quad</code> 1822	<code>\seq</code> . 1447, 1448, 1449, 1591, 1592, 1593, 1600
<code>\pgfmathparse</code> 1272, 2354, 2359, 2362, 2376, 2377	R	<code>\setbox</code> 2262
<code>\pgfmathresult</code> 1273, 2337, 2338, 2340, 2342, 2355, 2363, 2376, 2377	<code>\raisebox</code> 1443	<code>\setcounter</code> 1191, 2216, 2238, 2252
<code>\pgfutil@empty</code> 2338	<code>\relax</code> 1506, 2184, 2227, 2229, 2230, 2231, 2339, 2341	<code>\setganttlinklabel</code> .. 1221, 1222, 1223, 1224
<code>\pgfutil@loop</code> 2339	<code>\renewcommand</code> ... 1193, 1194, 1195, 1196, 1199, 1200, 1214, 1215, 1464, 1465, 1864	<code>\setlength</code> 2284, 2729, 2730, 2740
<code>\pgfutil@repeat</code> 2342	<code>\repeat</code> 2100	<code>\setmainfont</code> 1181
<code>\preceq</code> 1325	<code>\RequirePackage</code> 55, 158, 222, 226, 395, 401, 402, 462, 532, 611, 1027, 1029, 1030, 1068, 1069, 1179, 1180, 1183, 1185, 1187, 1192, 1201, 1208, 1219, 1220, 1229, 1233, 1234, 1235, 1263, 1264, 1303, 1372, 1471, 1476, 1477, 1493, 1500, 1501, 1502, 1530, 1645, 1806, 1807, 1952, 1954, 2044, 2104, 2106, 2258, 2259, 2260, 2273, 2301, 2332, 2383, 2389, 2394, 2405, 2406, 2474, 2579, 2586, 2605, 2606, 2608, 2609, 2647, 2648, 2649, 2682, 2702	<code>\setmainlanguage</code> 396
<code>\prime</code> 1304	<code>\right</code> 1647	<code>\setminted</code> 2411, 2412
<code>\printbibliography</code> . 1494	<code>\RIGHTarrow</code> ... 2623, 2628	<code>\setsansfont</code> 1182
<code>\ProvidesPackage</code> 2, 20, 53, 221, 399, 460, 467, 531, 594, 609, 1025, 1061, 1178, 1218, 1227, 1232, 1261, 1299, 1371, 1454, 1469, 1475, 1497, 1643, 1803, 1816, 1895, 1951, 2017, 2042, 2047, 2091, 2257, 2270, 2299, 2380, 2387, 2392, 2472, 2487, 2578, 2582, 2602, 2644, 2681, 2701	<code>\Rightarrow</code> 600, 605	<code>\setul</code> 2275
Q	<code>\rightarrow</code> 207, 489, 494, 502, 506, 508, 509, 511, 566, 574, 2014, 2517, 2524, 2526, 2529, 2534, 2539, 2540, 2545	<code>\sffamily</code> 476, 1188, 1190, 1292, 2495
<code>\QS@list</code> 2217, 2228, 2232, 2239, 2245, 2250, 2253	<code>\rightouterjoin</code> 2266	<code>\shoveleft</code> 1924
<code>\QS@select@equal</code> 2192, 2196	<code>\Roman</code> 1198	<code>\shoveright</code> 1927
<code>\QS@select@greater</code> 2193, 2197	<code>\roman</code> 1198, 1200	<code>\Sigma</code> 64, 105, 163, 1092, 1093, 1155
<code>\QS@select@smaller</code> 2188, 2191, 2195		<code>\sigma</code> 499, 501, 502
<code>\QS@sort@a</code> 2170, 2203, 2224, 2225		<code>\SLASH</code> 1512
<code>\QS@sort@b</code> 2170, 2171		<code>\small</code> 1621
<code>\QS@sort@c</code> 2174, 2181		<code>\sort</code> 2349
<code>\QS@sort@d</code> 2182, 2190		<code>\sortList</code> 2348, 2357
<code>\QS@sort@empty</code> . 2172, 2177		<code>\square</code> 464
<code>\QS@sort@single</code> 2173, 2178		<code>\stepcounter</code> 2136, 2141, 2146, 2149, 2151, 2155, 2157, 2161, 2163
		<code>\str</code> .. 477, 486, 1534, 2026, 2035, 2496, 2509
		<code>\StrSubstitute</code> . 2356, 2358
		<code>\strut</code> 1751, 1755, 1759, 1763, 1767, 2384
		<code>\subseteq</code> 2520, 2553, 2560
		T
		<code>\tableofcontents</code> ... 1507
		<code>\text</code> 77, 79, 182, 1906, 2652, 2705
		<code>\textbf</code> ... 1032, 1330, 1339, 1350, 1359,

1516, 1523, 1554, 1582, 1597, 1613, 1869	\textcolor 1442, 2651	\TmpPlaceOne 1967	\umlstatic 794, 823
\textit	901, 943, 944, 945, 946, 1623, 1914, 1947	\TmpPlaceSeven 1973	\umluniagggreg 851
\textsc 1305	\textsf 1516, 1597	\TmpPlaceSix 1972	\umluniassoc
\textstyle 1715, 1746	\texttt 1055, 1305, 1439, 1440, 1441, 1442, 2705	\TmpPlaceTen 1976	649, 665, 852, 976, 977
\thepage 1212, 1461	\theparagraph 1190	\TmpPlaceThree 1969	\umlVHuniassoc . . . 650, 651
\Theta 1650	\thinspace 2705	\TmpPlaceTwo 1968	\umlVHVdep 643,
\thinspace 2705	\tikz 1443	\TmpScale 1987	644, 646, 647, 796, 797
\tikz: bbaum 23	tikz: li binaer baum 21	\TmpTransitionEight 1963, 1984	\umlVHVinherit
\tikzchildnode 419	\tikzparentnode 419	\TmpTransitionFive 1960, 1981	. . . 626, 627, 632,
\tikzset 96, 143, 209, 404, 430, 1237, 1374, 1992, 2118, 2322, 2476, 2683	\tikzumlset 2611	\TmpTransitionFour 1959, 1980	633, 640, 641, 774, 775, 790, 791, 956, 957
\times 207	\tiny 1038, 1044, 1050, 1442, 1511, 2421	\TmpTransitionNine 1964, 1985	\umlVHVreal
\titleformat . . . 1188, 1190	\titlespacing 1189	\TmpTransitionOne 1956, 1977	. . . 768, 769, 993, 994
\tl . . . 29, 37, 63, 64, 65, 66, 67, 68, 71, 72, 73, 74, 75, 77, 79, 104, 105, 106, 107, 108, 109, 110, 113, 114, 115, 116, 117, 118, 119, 162, 163, 164, 165, 166, 167, 168, 171, 172, 173, 174, 175, 176, 177, 285, 289, 307, 311, 312, 313, 316, 321, 322, 323, 334, 335, 336, 337, 348, 354, 357, 360, 369, 383, 536, 539, 544, 545, 553, 554, 557, 558, 1133, 1154, 1155, 1156, 1157, 1160, 1161, 1162, 1163, 1680	\tmp . . . 29, 37, 63, 64, 65, 66, 67, 68, 71, 72, 73, 74, 75, 77, 79, 104, 105, 106, 107, 108, 109, 110, 113, 114, 115, 116, 117, 118, 119, 162, 163, 164, 165, 166, 167, 168, 171, 172, 173, 174, 175, 176, 177, 285, 289, 307, 311, 312, 313, 316, 321, 322, 323, 334, 335, 336, 337, 348, 354, 357, 360, 369, 383, 536, 539, 544, 545, 553, 554, 557, 558, 1133, 1154, 1155, 1156, 1157, 1160, 1161, 1162, 1163, 1680	\TmpTransitionSeven 1962, 1983	\UParrow 2625
\TmpPlaceEight 1974	\TmpPlaceFive 1971	\TmpTransitionSix 1961, 1982	\url 1632
\TmpPlaceFour 1970	\TmpPlaceNine 1975	\TmpTransitionTen 1965, 1986	\usemintedstyle 2408
		\TmpTransitionThree 1958, 1979	\usetikzlibrary . . . 56, 403, 1028, 1236, 1373, 1953, 2261, 2302, 2333, 2610, 2698
		\TmpTransitionTwo 1957, 1978	
		\TmpX 1988	V
		\TmpY 1989	\value 2098
		\today 1459	\varepsilon 478, 489, 490, 1078, 1697, 1703, 1732, 1743
		\ttfamily 2276	\vfill 2384
			\vrule 2591, 2595
			\vspace
			1113, 1117, 1568, 1586
			X
			\xappto . . . 2360, 2366, 2370
			\xdef 1266
			\xintApply 2186
			\xintApplyUnbraced
			2185, 2191, 2192, 2193
			\xintCSVtoList 2245
			\xintFor
			2135, 2140, 2145, 2150, 2156, 2162, 2203
			\xintifEq 2196
			\xintifForLast
			. . . 2152, 2158, 2164
			\xintifGt 2197
			\xintifLt 2195
			\xintLength 2170
			\xintnthelt 2182
			Z
			\ZB 1639
			\zB 1638
			\zustandsnamens@liste
			1096, 1103, 1104