

# lehramt-informatik

Hermine Bschlangaul \*

September 10, 2021

## Contents

<b>1</b>	<b>Klassen</b>	<b>4</b>
1.1	Vorlage Theorie-Teil	5
1.2	Vorlage Aufgabensammlung	6
1.3	Vorlage Aufgabe	7
<b>2</b>	<b>Pakete</b>	<b>8</b>
2.1	abmessung.sty	9
2.2	aufgaben-einbinden.sty	10
2.3	aufgaben-metadaten.sty	11
2.4	automaten.sty	12
2.4.1	Endlicher Automat	12
2.4.2	Kellerautomat	14
2.4.3	Turingmaschine	15
2.5	basis.sty	18
2.6	baum.sty	23
2.6.1	Binärbaum	24
2.6.2	AVL-Baum	25
2.6.3	B-Baum	26
2.7	checkbox.sty	27
2.8	chomsky-normalform.sty	28
2.8.1	Makro-Kürzel	28
2.8.2	TeX-Markup-Grundgerüst	28
2.8.3	Konkretes TeX-Markup-Beispiel	28
2.9	cpm.sty	31
2.9.1	Makro-Kürzel	31
2.9.2	TeX-Markup-Beispiel: Graph	31
2.9.3	TeX-Markup-Beispiel: Ergebnistabelle	31
2.9.4	TeX-Markup-Beispiel: Nebenrechnungstabelle „Frühester Zeitpunkt“	31
2.9.5	TeX-Markup-Beispiel: Nebenrechnungstabelle „Spätester Zeitpunkt“	32
2.9.6	Hilfsmakros für Zeitpunkt-Tabelle	33
2.10	cyk-algorithmus.sty	35
2.10.1	Makro-Kürzel	35
2.10.2	TeX-Markup-Beispiel	35
2.11	entwurfsmuster.sty	36
2.11.1	Namensschema der Entwurfsmuster-Makros:	36
2.11.2	Reihenfolge	36
2.11.3	Abstrakte Fabrik (Abstract Factory)	36
2.11.4	Adapter	38
2.11.5	Beobachter (Observer)	39
2.11.6	Dekorierer (Decorator)	41
2.11.7	Einfache Fabrik (Simple Factory)	42
2.11.8	Einzelstück (Singleton)	43

---

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

2.11.9	Erbauer (Builder)	44
2.11.10	Fabrikmethode (Factory Method)	45
2.11.11	Kompositum (Composite)	47
2.11.12	Modell-Präsentation-Steuerung (Model-View-Controller)	47
2.11.13	Stellvertreter (Proxy)	48
2.11.14	Zustand (State)	49
2.12	er.sty	51
2.12.1	Tex-Markup-Beispiel: Komplettes Diagramm	51
2.12.2	Tex-Markup-Beispiel: EER Enhanced Entity-Relation-Modell nach Elmasri/Navante	52
2.12.3	Makro-Kürzel	52
2.13	formale-sprachen.sty	54
2.14	formatierung.sty	57
2.14.1	Schriftarten / Typographie	57
2.14.2	Farben	57
2.14.3	Überschriften	57
2.14.4	Listen	57
2.14.5	Kasten	57
2.14.6	Header	57
2.14.7	Zeilenabstände	57
2.15	gantt.sty	59
2.16	grafik.sty	60
2.17	graph.sty	61
2.18	hanoi.sty	63
2.19	klassen-konfiguration-aufgabe.sty	64
2.20	klassen-konfiguration-examen.sty	67
2.21	komplexitaetstheorie.sty	69
2.21.1	Makro-Kürzel	69
2.22	kontrollflussgraph.sty	71
2.22.1	Makro-Kürzel	71
2.22.2	TeX-Markup-Beispiel	71
2.22.3	TikZ: pin	71
2.22.4	Umgebungen	72
2.22.5	Makros	73
2.23	kopf-fusszeilen.sty	74
2.24	literatur-dummy.sty	75
2.25	literatur.sty	76
2.26	makros.sty	77
2.27	master-theorem.sty	81
2.27.1	Makro-Kürzel	81
2.28	mathe.sty	85
2.29	meta.sty	86
2.30	minimierung.sty	88
2.31	normalformen.sty	91
2.31.1	Makro-Kürzel	91
2.32	o-notation.sty	94
2.32.1	Makro-Kürzel	94
2.32.2	TeX-Markup-Beispiel: Funktionsgraphen mit pgfplots	94
2.33	petri.sty	95
2.33.1	Makro-Kürzel	95
2.34	potenzmengen-konstruktion.sty	97
2.35	pseudo.sty	99
2.36	pumping-lemma.sty	100
2.37	quicksort.sty	101
2.38	relationale-algebra.sty	104
2.39	rmodell.sty	105
2.39.1	Makro-Kürzel	105
2.40	sortieren.sty	106

2.41	spalten.sty . . . . .	108
2.42	sql.sty . . . . .	109
2.43	struktogramm.sty . . . . .	110
2.44	syntax.sty . . . . .	111
2.44.1	Makro-Kürzel . . . . .	111
2.45	syntaxbaum.sty . . . . .	114
2.46	synthese-algorithmus.sty . . . . .	115
2.46.1	Makro-Kürzel . . . . .	115
2.46.2	TeX-Markup Grundgerüst . . . . .	115
2.46.3	TeX-Markup Linksreduktion . . . . .	115
2.46.4	TeX-Markup Rechtsreduktion . . . . .	115
2.46.5	TeX-Markup Relationen formen . . . . .	116
2.47	tabelle.sty . . . . .	119
2.48	typographie.sty . . . . .	120
2.49	uml.sty . . . . .	122
2.50	vollstaendige-induktion.sty . . . . .	124
2.50.1	Makro-Kürzel . . . . .	124
2.51	wasserfall.sty . . . . .	126
2.52	wpkalkuel.sty . . . . .	127
2.52.1	Makro-Kürzel . . . . .	127

### 3 Index 128

# 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 abmessung.sty

```
1 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2 \ProvidesPackage{lehramt-informatik-baum}[2021/09/04 Einstellung der
3 Seitenabmessung mit Hilfe des geometry-Pakets.]
4 \RequirePackage{geometry}
5 \geometry{
6   a4paper,
7   margin=2cm,
8   includeheadfoot,
9   %showframe,
10  %showcrop,
11  %verbose=true,
12 }
13
```

## 2.2 aufgaben-einbinden.sty

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

\liAufgabe Eine Aufgaben mit den Pfad relativ zu Wurzelverzeichnis dieses Repository einbinden.
16 \def\liAufgabe#1{
17   \input{\LehramtInformatikRepository/#1.tex}
18 }

\liExamensAufgabe Eine Examensaufgaben mit den Pfad relativ zu Staatsexamen einbinden, z. B.
\liExamensAufgabe{46116/2015/09/Thema-2/Teilaufgabe-1/Aufgabe-3}
19 \def\liExamensAufgabe#1{
20   \input{\LehramtInformatikRepository/Staatsexamen/#1.tex}
21 }

\liExamensAufgabeTTA
22 \def\liExamensAufgabeTTA #1 / #2 / #3 : Thema #4 Teilaufgabe #5 Aufgabe #6 {
23   \input{\LehramtInformatikRepository/Staatsexamen/#1/#2/#3/Thema-#4/Teilaufgabe-
      #5/Aufgabe-#6.tex}
24 }

\liExamensAufgabeTA
25 \def\liExamensAufgabeTA #1 / #2 / #3 : Thema #4 Aufgabe #5 {
26   \input{\LehramtInformatikRepository/Staatsexamen/#1/#2/#3/Thema-#4/Aufgabe-#5.tex}
27 }

\liExamensAufgabeA
28 \def\liExamensAufgabeA #1 / #2 / #3 : Aufgabe #4 {
29   \input{\LehramtInformatikRepository/Staatsexamen/#1/#2/#3/Aufgabe-#4.tex}
30 }

31
```

## 2.3 aufgaben-metadaten.sty

```

32 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
33 \ProvidesPackage{lehramt-informatik-aufgaben-metadaten}[2020/07/07 Makros zum
34 Setzen der Aufgaben-Metadaten.]

35 \ExplSyntaxOn

\liMetaSetze Setze die Metadaten einer Aufgabe.

36 \def\liMetaSetze#1{
37   \_setze_variablen_zurueck:
38
39   \tl_clear:N \g_auf_examen_teilaufgabe_nr_tl
40
41   \keys_set:nn { aufgabenmetadaten } {
42     #1
43   }
44
45   \_setze_relativen_pfad:
46 }

\liAufgabenMetadaten Setzen der Aufgaben-Metadaten über eine plist bzw. über key-values.
  Die Schlüssel-Werte-Paare sind in der Datei basis.sty definiert. In der Typescript-
  Datei .scripts/nodejs/src/aufgaben.ts gibt es ein entsprechendes Interface AufgabenMetadaten.

\liAufgabenMetadaten{
  Titel = {Aufgabe 2},
  Thematik = {Petri-Netz},
  Stichwoerter = {Feld (Array), Implementierung in Java}
  ZitatSchluessel = sosy:pu:4,
  ZitatBeschreibung = {Seite 11},
  BearbeitungsStand = OCR,
  Korrektheit = absolut korrekt,
  RelativerPfad = Staatsexamen/46116/2016/03/Thema-2/Teilaufgabe-1/Aufgabe-2.tex,
  ExamenNummer = 46116,
  ExamenJahr = 2016,
  ExamenMonat = 03,
  ExamenThemaNr = 2,
  ExamenTeilaufgabeNr = 1,
  ExamenAufgabeNr = 2,
}

47 \def\liAufgabenMetadaten#1{
48   \liMetaSetze{#1}
49
50   \_gib_examen_titel: {}
51
52   \section{\_gib_aufgaben_titel:}
53 }

54 \ExplSyntaxOff

\liAufgabenTitel Momentan eine dummy Makro das die Thematik enthält.

55 \def\liAufgabenTitel#1{}

56

```

## 2.4 automaten.sty

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

### 2.4.1 Endlicher Automat

```
59 \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}
```



```
60 \RequirePackage{tikz}
61 \usetikzlibrary{arrows,automata,positioning}
62 \liLadePakete{mathe}
63 \directlua{
64   automaten = require('lehramt-informatik-automaten')
65 }
```

`\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}`:  $A_{NEA} = (Z, \Sigma, \delta, E, z_0)$

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

```

80     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
81     dea .value_forbidden:n = true,
82     dea .code:n = {\tl_set:Nn \l_typ_tl {\sb{\text{DEA}}}},
83     nea .value_forbidden:n = true,
84     nea .code:n = {\tl_set:Nn \l_typ_tl {\sb{\text{NEA}}}},
85   }
86
87   \keys_set:nn { automat } { #2 }
88
89   $! \l_typ_tl = (
90     \l_zustaende_tl,
91     \l_alphabet_tl,
92     \l_delta_tl,
93     \l_ende_tl,
94     \l_start_tl
95   )$
96 }
97 \ExplSyntaxOff

```

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

```

98 \def\liAutomatenKante#1#2#3#4{
99   \path (#1) edge[#4] node{#3} (#2);
100 }

101 \tikzset{
102   li automat/.style={
103     ->,
104     node distance=2cm
105   },
106 }

```

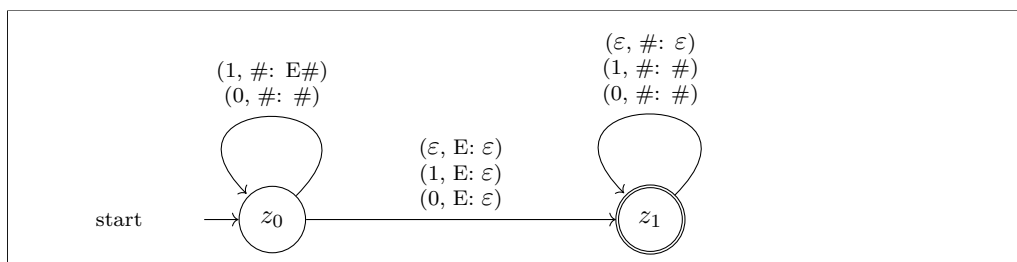
## 2.4.2 Kellerautomat

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

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

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

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



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

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

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

```
107 \ExplSyntaxOn
108 \NewDocumentCommand {\liKellerAutomat} { O{K} m } {
109   \tl_set:Nn \l_zustaeende_tl {Z}
110   \tl_set:Nn \l_alphabet_tl {\Sigma}
111   \tl_set:Nn \l_kelleralphabet_tl {\Gamma}
112   \tl_set:Nn \l_delta_tl {\delta}
113   \tl_set:Nn \l_start_tl {z\sb{0}}
114   \tl_set:Nn \l_kellerboden_tl {\#}
115   \tl_set:Nn \l_ende_tl {E}
116
117   \keys_define:nn { kellerautomat } {
118     zustaeende .code:n = {\tl_set:Nn \l_zustaeende_tl {\liMenge{##1}}},
119     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
120     kelleralphabet .code:n = {\tl_set:Nn \l_kelleralphabet_tl {\liMenge{##1}}},
121     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
122     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
```

```

123     kellerboden .code:n = {\tl_set:Nn \l_kellerboden_tl {##1}},
124     ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
125   }
126
127   \keys_set:nn { kellerautomat } { #2 }
128
129   $#1 = (
130     \l_zustaende_tl,
131     \l_alphabet_tl,
132     \l_kelleralphabet_tl,
133     \l_delta_tl,
134     \l_start_tl,
135     \l_kellerboden_tl,
136     \l_ende_tl
137   )$
138 }
139 \ExplSyntaxOff

```

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

(a, #: A)  
(b, #:  $\varepsilon$ )

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

```

140 \ExplSyntaxOn
141 \def\liKellerUebergang#1{
142   \directlua{automaten.drucke_keller_uebergaenge('#1')}
143 }
144 \ExplSyntaxOff

```

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

```

145 \NewDocumentCommand{\liKellerKante} { 0{above} m m m } {
146   \path (#2) edge[#1] node{\liKellerUebergang{#4}} (#3);
147 }

148 \tikzset{
149   li keller knoten/.style={
150     text width=2cm,
151     align=center,
152     font=\footnotesize,
153   },
154   li kellerautomat/.style={
155     li automat,
156     every edge/.append style={
157       every node/.style={
158         li keller knoten
159       }
160     }
161   }
162 }

```

### 2.4.3 Turingmaschine

```
163 \RequirePackage{amssymb}
```

**\liTuringLeerzeichen**

□

```
164 \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},
}

```

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

```

165 \ExplSyntaxOn
166 \NewDocumentCommand {\liTuringMaschine} { O{TM} m } {
167   \tl_set:Nn \l_zustaende_tl {Z}
168   \tl_set:Nn \l_alphabet_tl {\Sigma}
169   \tl_set:Nn \l_bandalphabet_tl {\Gamma}
170   \tl_set:Nn \l_delta_tl {\delta}
171   \tl_set:Nn \l_start_tl {z\sb{0}}
172   \tl_set:Nn \l_leerzeichen_tl {\liTuringLeerzeichen}
173   \tl_set:Nn \l_ende_tl {E}
174
175   \keys_define:nn { kellerautomat } {
176     zustaende .code:n = {\tl_set:Nn \l_zustaende_tl {\liMenge{##1}}},
177     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
178     bandalphabet .code:n = {\tl_set:Nn \l_bandalphabet_tl {\liMenge{##1}}},
179     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
180     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
181     leerzeichen .code:n = {\tl_set:Nn \l_leerzeichen_tl {##1}},
182     ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
183   }
184
185   \keys_set:nn { kellerautomat } { #2 }
186
187   $\text{#1} = (
188     \l_zustaende_tl,
189     \l_alphabet_tl,
190     \l_bandalphabet_tl,
191     \l_delta_tl,
192     \l_start_tl,
193     \l_leerzeichen_tl,
194     \l_ende_tl
195   )$
196 }
197 \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)`

```

198 \ExplSyntaxOn
199 \def\liTuringUebergangZelle#1{
200   \directlua{tex.print(automaten.gib_einen_turing_uebergang('#1'))}
201 }
202 \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)`



```

203 \ExplSyntaxOn
204 \def\liTuringUebergaenge#1{
205   \directlua{automaten.drucke_turing_uebergaenge('#1')}
206 }
207 \ExplSyntaxOff

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

208 \NewDocumentCommand{\liTuringKante} { 0{above} m m m } {
209   \path (#2) edge[#1] node{\liTuringUebergaenge{#4}} (#3);
210 }

\liTuringUeberfuehrung

211 \def\liTuringUeberfuehrung{
212    $\delta : Z \times \Gamma \rightarrow Z \times \Gamma \times \{ L, R, N \}$ 
213 }

214 \tikzset{
215   li turingmaschine/.style={
216     li automat,
217     every edge/.append style={
218       every node/.style={
219         li keller knoten
220       }
221     }
222   }
223 }
224

```

## 2.5 basis.sty

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

227 \RequirePackage{xparse}

228 \ExplSyntaxOn

\liLadePakete

229 \NewDocumentCommand{\liLadePakete}{ m }
230 {
231   \clist_map_inline:nn { #1 } { \RequirePackage{lehramt-informatik-##1} }
232 }

\liLadeAllePakete

233 \def\liLadeAllePakete{
234   \liLadePakete{
235     aufgaben-einbinden,
236     automaten,
237     checkbox,
238     chomsky-normalform,
239     cpm,
240     cyk-algorithmus,
241     entwurfsmuster,
242     er,
243     formale-sprachen,
244     gantt,
245     grafik,
246     graph,
247     hanoi,
248     kontrollflussgraph,
249     komplexitaetstheorie,
250     makros,
251     master-theorem,
252     mathe,
253     minimierung,
254     normalformen,
255     petri,
256     potenzmengen-konstruktion,
257     pumping-lemma,
258     pseudo,
259     quicksort,
260     relationale-algebra,
261     rmodell,
262     sortieren,
263     spalten,
264     struktogramm,
265     sql,
266     syntax,
267     syntaxbaum,
268     synthese-algorithmus,
269     tabelle,
270     typographie,
271     uml,
272     vollstaendige-induktion,
273     wasserfall,
274     wpkalkuel,
275     %
276     baum, % am Schluss sonst Fehler: undefined command \edge
277   }
278 }
```

Definition einer Komma-getrennten-Liste mit deren Hilfe die vielen globalen Token-List-Variablen definiert werden können. Die einzelnen Schlüssel sind im Interface

AufgabenMetadaten in der Typescript-Datei `.scripts/nodejs/src/aufgaben.ts` erklärt.

```
279 \clist_new:N \g_auf_schluessel_clist
280 \clist_set:Nn \g_auf_schluessel_clist {
281   titel,
282   thematik,
283   stichwoerter,
284   zitat_schluessel,
285   zitat_beschreibung,
286   %
287   bearbeitungs_stand,
288   korrektheit,
289   %
290   relativer_pfad,
291   identische_aufgabe,
292   %
293   examen_nummer,
294   examen_fach,
295   examen_jahr,
296   examen_monat,
297   examen_jahreszeit,
298   examen_thema_nr,
299   examen_teilaufgabe_nr,
300   examen_aufgabe_nr,
301 }
```

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

```
302 \clist_map_inline:Nn \g_auf_schluessel_clist {
303   \tl_new:c {g_auf_#1_tl}
304 }
```

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

```
305 \cs_new:Npn \setze_variablen_zurueck: {
306   \clist_map_inline:Nn \g_auf_schluessel_clist {
307     \tl_clear:c {g_auf_##1_tl}
308   }
309 }
```

Die einzelnen Schlüssel sind im Interface `AufgabenMetadaten` in der Typescript-Datei `.scripts/nodejs/src/aufgaben.ts` erklärt.

```
310 \keys_define:nn { aufgabenmetadaten }
311 {
312   Titel .tl_gset:N = \g_auf_titel_tl,
313   Thematik .tl_gset:N = \g_auf_thematik_tl,
314   Stichwoerter .tl_gset:N = \g_auf_stichwoerter_tl,
315   ZitatSchluessel .tl_gset:N = \g_auf_zitat_beschreibung_tl,
316   ZitatBeschreibung .tl_gset:N = \g_auf_zitat_schluessel_tl,
317   %
318   BearbeitungsStand .tl_gset:N = \g_auf_bearbeitungs_stand_tl,
319   Korrektheit .tl_gset:N = \g_auf_korrektheit_tl,
320   %
321   RelativerPfad .tl_gset:N = \g_auf_relativer_pfad_tl,
322   IdentischeAufgabe .tl_gset:N = \g_auf_identische_aufgabe_tl,
323   %
324   ExamenNummer .tl_gset:N = \g_auf_examen_nummer_tl,
325   ExamenFach .tl_gset:N = \g_auf_examen_fach_tl,
326   ExamenJahr .tl_gset:N = \g_auf_examen_jahr_tl,
327   ExamenMonat .tl_gset:N = \g_auf_examen_monat_tl,
328   ExamenJahreszeit .tl_gset:N = \g_auf_examen_jahreszeit_tl,
329   ExamenThemaNr .tl_gset:N = \g_auf_examen_thema_nr_tl,
330   ExamenTeilaufgabeNr .tl_gset:N = \g_auf_examen_teilaufgabe_nr_tl,
331   ExamenAufgabeNr .tl_gset:N = \g_auf_examen_aufgabe_nr_tl,
332 }
```

```

333 \cs_gset:Npn \_setze_relativen_pfad: {
334   \tl_if_empty:NTF \g_auf_relativer_pfad_tl
335   {
336     \bool_if:nTF
337     {
338       ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
339       ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
340       ! \tl_if_empty_p:N \g_auf_examen_monat_tl
341     }
342     {
343       \tl_gset:Nn \g_auf_relativer_pfad_tl {
344         Staatsexamen /
345         \g_auf_examen_nummer_tl /
346         \g_auf_examen_jahr_tl /
347         \g_auf_examen_monat_tl /
348         \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} { Thema - \g_auf_examen_thema_nr_tl / }
349         \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} { Teilaufgabe - \g_auf_examen_teil_
350         \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} { Aufgabe - \g_auf_examen_aufgabe_nr_
351       }
352     }
353   {}
354 }
355 {}
356 }

357 \cs_set:Nn \_trenner: {
358   \, / \,
359 }

360 \cs_gset:Npn \_gib_jahreszeit_durch_monat: #1 {
361   % \str_case funktioniert nicht mit den Tokenlist variablen.
362   \tl_case:Nn { #1 }
363   {
364     { 3 } { Frühjahr }
365     { 03 } { Frühjahr }
366     { 9 } { Herbst }
367     { 09 } { Herbst }
368   }
369 }

```

Definiert auch in .scripts/nodejs/src/examen.ts funktioniert nicht

```

370 \cs_gset:Npn \_gib_examen_fach_durch_nummer: #1 {
371   \tl_case:Nn { #1 }
372   {
373     { 46110 } { Grundlagen der Informatik (nicht vertieft) }
374     { 46111 } { Programmentwicklung / Systemprogrammierung / Datenbanksysteme (nicht vertieft) }
375     { 46112 } { Grundlagen der Informatik (nicht vertieft) }
376     { 46113 } { Theoretische Informatik (nicht vertieft) }
377     { 46114 } { Algorithmen / Datenstrukturen / Programmiermethoden (nicht vertieft) }
378     { 46115 } { Theoretische Informatik / Algorithmen / Datenstrukturen (nicht vertieft) }
379     { 46116 } { Softwaretechnologie / Datenbanksysteme (nicht vertieft) }
380     { 46118 } { Fachdidaktik (Mittelschulen) }
381     { 46119 } { Fachdidaktik (Realschulen) }
382     { 46121 } { Fachdidaktik (berufliche Schulen) }
383     { 66110 } { Automatentheorie, Algorithmische Sprache (vertieft) }
384     { 66111 } { Betriebssysteme / Datenbanksysteme / Rechnerarchitektur (vertieft) }
385     { 66112 } { Automatentheorie / Komplexität / Algorithmen (vertieft) }
386     { 66113 } { Rechnerarchitektur / Datenbanken / Betriebssysteme (vertieft) }
387     { 66114 } { Datenbank- und Betriebssysteme (vertieft) }
388     { 66115 } { Theoretische Informatik / Algorithmen (vertieft) }
389     { 66116 } { Datenbanksysteme / Softwaretechnologie (vertieft) }
390     { 66118 } { Fachdidaktik (Gymnasium) }
391   }
392 }

```

Einzelprüfungsnummer / Jahr / Jahreszeit mit Trennzeichen

```

393 \cs_gset:Npn \_gib_einzelpreuefung_trenner: {
394   Staatsexamen ~
395   \g_auf_examen_nummer_tl
396
397   \_trenner:
398
399   \g_auf_examen_jahr_tl
400
401   \_trenner:
402
403   \gib_jahreszeit_durch_monat: \g_auf_examen_monat_tl
404 }

Thema Nr.1 / Teilaufgabe Nr. 2 / Aufgabe 3
405 \cs_gset:Npn \_gib_aufgaben_pfad_trenner: {
406   \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
407     Thema ~ \g_auf_examen_thema_nr_tl \_trenner:
408   }
409   \tl_if_empty:NTF \g_auf_examen_telaufgabe_nr_tl {} {
410     Teilaufgabe ~ \g_auf_examen_telaufgabe_nr_tl \_trenner:
411   }
412   \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} {
413     Aufgabe ~ \g_auf_examen_aufgabe_nr_tl
414   }
415 }

416 \cs_gset:Npn \_gib_examen_titel: {
417   \cs_set:Nn \_trenner: { \, / \, }
418   \bool_if:nTF
419   {
420     ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
421     ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
422     ! \tl_if_empty_p:N \g_auf_examen_monat_tl &&
423     ! \tl_if_empty_p:N \g_auf_examen_aufgabe_nr_tl
424   }
425   {
426     {
427       \footnotesize
428       \par
429       \noindent
430       Staatsexamen ~
431       \g_auf_examen_nummer_tl \_trenner:
432       \g_auf_examen_jahr_tl \_trenner:
433
434       \tl_case:Nn \g_auf_examen_monat_tl
435       {
436         { 03 } { Frühjahr }
437         { 09 } { Herbst }
438       } \_trenner:
439
440       \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
441         Thema ~ Nr. ~ \g_auf_examen_thema_nr_tl \_trenner:
442       }
443       \tl_if_empty:NTF \g_auf_examen_telaufgabe_nr_tl {} {
444         Teilaufgabe ~ Nr. ~ \g_auf_examen_telaufgabe_nr_tl \_trenner:
445       }
446       \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} {
447         Aufgabe ~ Nr. ~ \g_auf_examen_aufgabe_nr_tl
448       }
449       \par
450       \bigskip
451     }
452   }
453 }

```

```

454 \cs_new:Npn \_gib_github_url: {
455   \LehramtInformatikGithubDomain /
456   \LehramtInformatikGithubTexRepo /
457   blob /
458   \LehramtInformatikGitBranch /
459   \g_auf_relativer_pfad_tl
460 }

461 \cs_new:Npn \_gib_github_url_href: {
462   \tl_if_empty:NTF \g_auf_relativer_pfad_tl {} {
463     \url{ \_gib_github_url: }
464   }
465 }

466 \cs_new:Npn \_gib_aufgaben_titel: {
467   \g_auf_titel_tl
468
469   \tl_if_empty:NTF \g_auf_thematik_tl
470   {}
471   {
472     \, ~ [
473       \g_auf_thematik_tl
474     ]
475   }
476 }

477 \msg_new:nnn { aufgabenmetadaten } { kein-titel }
478 { Der~Schlüssel~ist~zwingend~notwendig }

\def\LehramtInformatikRepository{/pfad/zum/repository}
479 \input /etc/lehramt-informatik.config.tex

biblatex not working with lualatex and babel
480 % \RequirePackage{polyglossia}
481 % \setmainlanguage{german}
482

```

## 2.6 baum.sty

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

486 \RequirePackage{tikz}

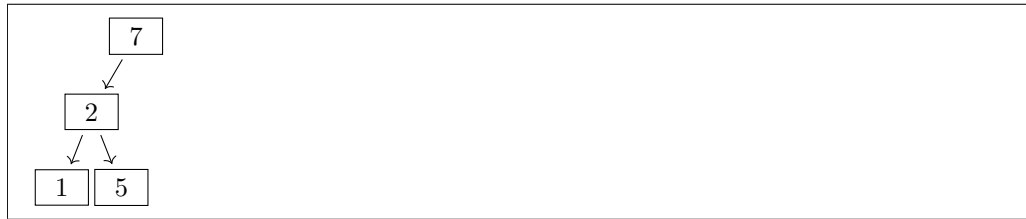
    für li binaer baum
487 \RequirePackage{tikz-qtree}

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

### 2.6.1 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}
```



```
489 \tikzset{
490   li binaer baum/.style={
491     shorten <=2pt,
492     shorten >=2pt,
493     ->,
494     every tree node/.style={
495       minimum width=2em,
496       draw,
497       rectangle
498     },
499     blank/.style={
500       draw=none
501     },
502     edge from parent/.style={
503       draw,
504       edge from parent path={(\tikzparentnode) -- (\tikzchildnode)}
505     },
506     level distance=1cm,
507     every label/.style={
508       gray,
509       font=\footnotesize,
510       label position=0,
511       label distance=0cm,
512     }
513   },
514 }
```



### 2.6.2 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}
```

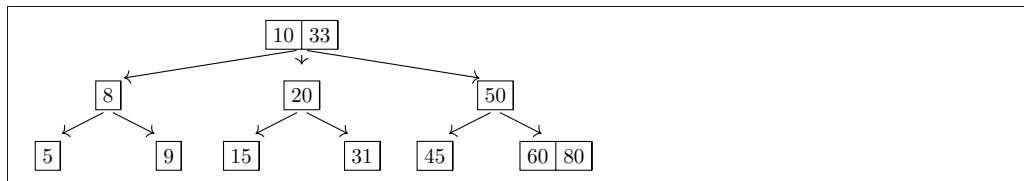


### 2.6.3 B-Baum

```

\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}

```



```

515 \tikzset{
516   li bbaum knoten/.style={
517     rectangle split parts=10,
518     rectangle split,
519     rectangle split horizontal,
520     rectangle split ignore empty parts,
521     draw,
522     fill=white
523   },
524   li bbaum/.style={
525     every node/.style={
526       li bbaum knoten
527     },
528     level 1/.style={
529       level distance=12mm,
530       sibling distance=25mm,
531     },
532     every child/.style={
533       shorten <= 2pt,
534       shorten >= 6pt,
535       ->,
536     },
537     level 2/.style={
538       level distance=9mm,
539       sibling distance=15mm,
540     },
541   }
542 }
543

```

## 2.7 checkbox.sty

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

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

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

550
```

## 2.8 chomsky-normalform.sty

```
551 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
552 \ProvidesPackage{lehramt-informatik-chomsky-normalform}[2021/03/26
553 Hilfsmakros zum Setzen des Algorithmus zum Erreichen der Chomsky-Normalform]

554 \ExplSyntaxOn

555 \liLadePakete{typographie}
```

### 2.8.1 Makro-Kürzel

```
\let\erklaerung=\liChomskyErklaerung
\let\schritt=\liChomskyUeberschrift
\let\schrittE=\liChomskyUeberErklaerung
```

### 2.8.2 TeX-Markup-Grundgerüst

```
\let\schrittE=\liChomskyUeberErklaerung
\begin{enumerate}
\item \schrittE{1}
\item \schrittE{2}
\item \schrittE{3}
\item \schrittE{4}
\end{enumerate}
```

### 2.8.3 Konkretes TeX-Markup-Beispiel

```
\begin{enumerate}
\item \schrittE{1}

\liNichtsZuTun

\item \schrittE{2}

\begin{liProduktionsRegeln}
S -> d S e | a | U c T | S b U,
T -> d S e | a,
U -> d S e | a | U c T,
\end{liProduktionsRegeln}

\item \schrittE{3}

\begin{liProduktionsRegeln}
S -> D S E | a | U C T | S B U,
T -> D S E | a,
U -> D S E | a | U C T,
B -> b,
C -> c,
D -> d,
E -> e,
\end{liProduktionsRegeln}

\item \schrittE{4}
% S -> S S.1 | T2 S.2 | a | U S.3
% T -> T2 S.2 | a
% U -> T2 S.2 | a | U S.3
% T1 -> b
% T2 -> d
% T3 -> e
% T4 -> c
% S.1 -> T1 U
% S.2 -> S T3
% S.3 -> T4 T
```

```

\begin{liProduktionsRegeln}
S -> D S_E | a | U C_T | S B_U, % S -> S S.1 | T2 S.2 | a | U S.3
T -> D S_E | a, % T -> T2 S.2 | a
U -> D S_E | a | U C_T, % U -> T2 S.2 | a | U S.3
B -> b, % T1 -> b
C -> c, % T4 -> c
D -> d, % T2 -> d
E -> e, % T3 -> e
S_E -> S E, % S.2 -> S T3
C_T -> C T, % S.3 -> T4 T
B_U -> B U, % S.1 -> T1 U
\end{liProduktionsRegeln}
\end{enumerate}

```

\liChomskyUeberschrift **Let-Abkürzung:** \let\schritt=\liChomskyUeberschrift

```

556 \def\liChomskyUeberschrift#1{
557   {
558     \bfseries
559     \rmfamily
560     \str_case:nn {#1} {
561       {1} {Elimination-der-\varepsilon-Regeln}
562       {2} {Elimination-von-Kettenregeln}
563       {3} {Separation-von-Terminalzeichen}
564       {4} {Elimination-von-mehrelementigen-Nonterminalketten}
565     }
566   }
567 }

```

\liChomskyErklaerung **Let-Abkürzung:** \let\erklaerung=\liChomskyErklaerung  
Hoffmann Seite 180

```

568 \def\liChomsky@erklaerung@texte#1{
569   \str_case:nn {#1} {
570     %
571     {1} {
572       Alle-Regeln-der-Form- $A \rightarrow \varepsilon$ -werden-eliminiert.~
573       Die-Ersetzung-von- $A$ -wird-durch- $\varepsilon$ -in-allen-anderen-
574       Regeln-vorweggenommen.
575     }
576     {2} {
577       Jede-Produktion-der-Form- $A \rightarrow B$ -mit- $A, B$ -in- $S$ -wird-
578       als-Kettenregel-bezeichnet.~Diese-tragen-nicht-zur-Produktion-
579       von-Terminalzeichen-bei-und-lassen-sich-ebenfalls-eliminieren.
580     }
581     {3} {
582       Jedes-Terminalzeichen- $\sigma$ ,~das-in-Kombination-mit-anderen-
583       Symbolen-auftaucht,~wird-durch-ein-neues-Nonterminal-
584        $S_{\sigma}$ -ersetzt-und-die-Menge-der-Produktionen-durch-die-
585       Regel- $S_{\sigma} \rightarrow \sigma$ -ergänzt.
586     }
587     {4} {
588       Alle-Produktionen-der-Form-
589        $A \rightarrow B_{\{1\}} B_{\{2\}} \dots B_{\{n\}}$ -
590       werden-in-die-Produktionen-
591        $A \rightarrow A_{\{n-1\}} B_{\{n\}}, A_{\{n-1\}} \rightarrow A_{\{n-2\}} B_{\{n-1\}}, \dots,$ 
592        $A_{\{2\}} \rightarrow B_{\{1\}} B_{\{2\}}$ -zerteilt.~
593       Nach-der-Ersetzung-sind-alle-längeren-Nonterminalketten-
594       vollständig-heruntergebrochen-und-die-Chomsky-Normalform-erreicht.
595     }
596   }
597 }
598 }

```

```

599 }
600 \def\liChomskyErklaerung#1{
601   {
602     \itshape
603     \footnotesize
604     \liParagraphMitLinien{\li@chomsky@erklaerung@texte{#1}}
605   }
606 }

```

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

```

607 \def\liChomskyUeberErklaerung#1{
608   \liChomskyUeberschrift{#1}\par
609   \liChomskyErklaerung{#1}
610 }

```

```

611 \ExplSyntaxOff
612

```

## 2.9 cpm.sty

```

613 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
614 \ProvidesPackage{lehramt-informatik-cpm}[2020/09/03]
615 \RequirePackage{tikz}

616 \liLadePakete{mathe,typographie}

```

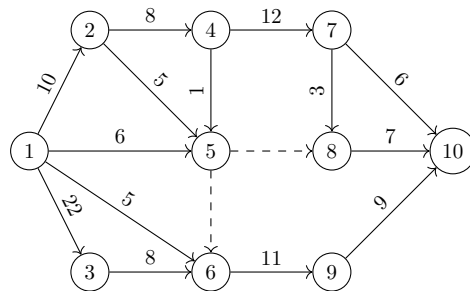
### 2.9.1 Makro-Kürzel

```

\let\f=\footnotesize
\let\FZ=\liCpmFruehI
\let\SZ=\liCpmSpaetI
\let\v=\liCpmVon
\let\ vz=\liCpmVonZu
\let\ z=\liCpmZu

```

### 2.9.2 TeX-Markup-Beispiel: Graph



```

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

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

\liCpmVorgang[schein]{5}{6}{5}
\liCpmVorgang[schein]{5}{8}{3}
\end{tikzpicture}

```

### 2.9.3 TeX-Markup-Beispiel: Ergebnistabelle

```

\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 \\
\SZ & 0 & 2 & 5 & 10 & 10 & 6 & 12 \\
GP & 0 & 0 & 0 & 3 & 0 & 3 & 0 \\
\hline
\end{tabular}

```

### 2.9.4 TeX-Markup-Beispiel: Nebenrechnungstabelle „Frühester Zeitpunkt“

```

\liCpmFruehErklaerung
\begin{tabular}{|l|l|r|}
\hline
i & Nebenrechnung & \FZ \\
1 & & 0 \\
2 & & 5 \\
3 & & 18
\end{tabular}

```

```

4 & & 7 \\
5 & & 19 \\
6 & & 26 \\
7 &  $\max(19_3, 22_4)$  & 22 \\
8 &  $\max(30_5, 30_6, 28_7)$  & 30 \\
\end{tabular}

```

## 2.9.5 TeX-Markup-Beispiel: Nebenrechnungstabelle „Spätester Zeitpunkt“

```

\liCpmSpaetErklaerung
% Absteigend nach i sortieren
\begin{tabular}{|l|l|r|}
\hline
i & Nebenrechnung & \SZ \\
8 & siehe \FZ[8] & 30 \\
7 & & 24 \\
6 & & 26 \\
5 & & 19 \\
4 & & 9 \\
3 &  $\min(18_6, 23_7)$  & 18 \\
2 & & 5 \\
1 &  $\min(0_2, 0_3, 2_4)$  & 0 \\
\hline
\end{tabular}

```

```

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

```

```

617 \ExplSyntaxOn
618 \NewDocumentCommand { \liCpmEreignis } { 0 } m m m } {
619   \tl_set:Nn \l_name_tl {}
620
621   \keys_define:nn { cpmEreignis } {
622     name .code:n = {\tl_set:Nn \l_name_tl {##1}},
623   }
624
625   \keys_set:nn { cpmEreignis } { #1 }
626
627   \tl_if_empty:NT \l_name_tl {
628     \tl_set:Nn \l_name_tl {#2}
629   }
630
631   \node[circle,draw] (\l_name_tl) at (#3,#4) {#2};
632 }
633 \ExplSyntaxOff

```

```

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

```

```

634 \ExplSyntaxOn
635 \NewDocumentCommand { \liCpmVorgang } { 0 } m m m } {
636   \tl_set:Nn \l_schein_tl {}
637   \tl_set:Nn \l_kritisch_tl {}
638
639   \keys_define:nn { cpmVorgang } {
640     schein .code:n = {\tl_set:Nn \l_schein_tl {dashed}},
641     kritisch .code:n = {\tl_set:Nn \l_kritisch_tl {thick}},
642   }
643
644   \keys_set:nn { cpmVorgang } { #1 }
645
646   \path[->,\l_schein_tl,\l_kritisch_tl] (#2) edge node[auto,sloped] {#4} (#3);
647 }
648 \ExplSyntaxOff

```



## 2.9.6 Hilfsmakros für Zeitpunkt-Tabelle

```

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

\liCpmVonZu Let-Abkürzung: \let\ vz=\liCpmVonZu
\liCpmVonZu{1}(2-3):  $1_{(2 \rightarrow 3)}$ 
649 \def\liCpmVonZuOhneMathe#1(#2-#3){#1_{\scriptscriptstyle(#2\rightarrow#3)}}
650 \def\liCpmVonZu#1(#2-#3){%
651 \ifmmode%
652 \liCpmVonZuOhneMathe{#1}(\#2-\#3)%
653 \else%
654 $\liCpmVonZuOhneMathe{#1}(\#2-\#3)$%
655 \fi%
656 }

\liCpmVon Let-Abkürzung: \let\ v=\liCpmVon
\liCpmVon{1}(2):  $1_{(\rightarrow 2)}$ 
657 \def\liCpmVonOhneMathe#1(#2){#1_{\scriptscriptstyle(\rightarrow#2)}}
658 \def\liCpmVon#1(#2){%
659 \ifmmode%
660 \liCpmVonOhneMathe{#1}(\#2)%
661 \else%
662 $\liCpmVonOhneMathe{#1}(\#2)$%
663 \fi%
664 }

\liCpmZu Let-Abkürzung: \let\ z=\liCpmZu
\liCpmZu{1}(2):  $1_{(\leftarrow 2)}$ 
665 \def\liCpmZuOhneMathe#1(#2){#1_{\scriptscriptstyle(\leftarrow#2)}}
666 \def\liCpmZu#1(#2){%
667 \ifmmode%
668 \liCpmZuOhneMathe{#1}(\#2)%
669 \else%
670 $\liCpmZuOhneMathe{#1}(\#2)$%
671 \fi%
672 }

673 \ExplSyntaxOn

\liCpmSpaetI Spätester Zeitpunkt, zu dem Ereignis  $i$  eintreten kann
Let-Abkürzung: \let\ SZ=\liCpmSpaetI
674 \NewDocumentCommand{ \liCpmSpaetI } { 0{i} } {
675 \ifmmode
676 SZ\sb{#1}
677 \else
678 $SZ\sb{#1}$
679 \fi
680 }

\liCpmFruehI Frühester Zeitpunkt, zu dem Ereignis  $i$  eintreten kann.
Let-Abkürzung: \let\ FZ=\liCpmFruehI
681 \NewDocumentCommand{ \liCpmFruehI } { 0{i} } {
682 \ifmmode
683 FZ\sb{#1}
684 \else
685 $FZ\sb{#1}$

```

```

686 \fi
687 }

```

\liCpmFruehErklaerung

— Wir führen eine Vorwärtsterminierung durch und addieren die Dauern. Kann ein Ereignis über mehrere Vorgänge erreicht werden, wählen wir das Maximum aus. **Erläuterungen:**  $i$ : Ereignis  $i$ ;  $FZ_i$ : Frühester Zeitpunkt, zu dem Ereignis  $i$  eintreten kann. —

```

688 \def\liCpmFruehErklaerung{
689   \liParagraphMitLinien{
690     Wir~führen~eine~Vorwärtsterminierung~durch~
691     und~addieren~die~Dauern.~
692
693     Kann~ein~Ereignis~über~mehrere~Vorgänge~erreicht~
694     werden,~wählen~wir~das~Maximum~aus.~
695
696     \textbf{Erläuterungen:}~
697
698      $i$ :~
699     Ereignis~ $i$ ;~,
700
701     \liCpmFruehI{}:~
702     Frühester~Zeitpunkt,~zu~dem~Ereignis~ $i$ ~eintreten~kann
703     .
704   }
705 }

```

\liCpmSpaetErklaerung

— Wir führen eine Rückwärtsterminierung durch und subtrahieren die Dauern vom letzten Ereignis aus. Kann ein Ereignis über mehrere Vorgänge erreicht werden, wählen wir das Minimum aus. **Erläuterungen:**  $i$ : Ereignis  $i$ ;  $SZ_i$ : Spätester Zeitpunkt, zu dem Ereignis  $i$  eintreten kann. —

```

706 \def\liCpmSpaetErklaerung{
707   \liParagraphMitLinien{
708     Wir~führen~eine~Rückwärtsterminierung~durch~
709     und~subtrahieren~die~Dauern~vom~letzten~Ereignis~aus.~
710
711     Kann~ein~Ereignis~über~mehrere~Vorgänge~erreicht~
712     werden,~wählen~wir~das~Minimum~aus.~
713
714     \textbf{Erläuterungen:}~
715
716      $i$ :~
717     Ereignis~ $i$ ;~,
718
719     \liCpmSpaetI{}:~
720     Spätester~Zeitpunkt,~zu~dem~Ereignis~ $i$ ~eintreten~kann
721     .
722   }
723 }

724 \ExplSyntaxOff
725

```

## 2.10 cyk-algorithmus.sty

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

### 2.10.1 Makro-Kürzel

```
\let\l=\liKurzeTabellenLinie
```

### 2.10.2 TeX-Markup-Beispiel

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

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

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

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

`\liWortInSprache` `\liWortInSprache{abc}:  $\Rightarrow abc \in L(G)$`

`\liWortInSprache{abc}[L(Z)]:  $\Rightarrow abc \in L(Z)$`

```
730 \NewDocumentCommand{ \liWortInSprache } { m O{L(G)} } {
731   \bigskip
732   \noindent
733   $\Rrightarrow #1 \in #2$
734 }
```

`\liWortNichtInSprache` `\liWortNichtInSprache{abc}:  $\Rightarrow abc \notin L(G)$`

`\liWortNichtInSprache{abc}[L(Z)]:  $\Rightarrow abc \notin L(Z)$`

```
735 \NewDocumentCommand{ \liWortNichtInSprache } { m O{L(G)} } {
736   \bigskip
737   \noindent
738   $\Rrightarrow #1 \notin #2$
739 }
```

```
740
```

## 2.11 entwurfsmuster.sty

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

### 2.11.1 Namensschema der Entwurfsmuster-Makros:

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

### 2.11.2 Reihenfolge

1. Beschreibung Kurze Beschreibung des Entwurfsmusters, z. B \liEntwurfsEinzelstueckBeschreibung
2. Uml: Uml-Klassendiagramm, z. B \liEntwurfsEinzelstueckUml
3. Akteure: Akteure, beteiligte Klassen, z. B \liEntwurfsEinzelstueckAkteure
4. Code: Allgemeines Code-Beispiel, z. B \liEntwurfsEinzelstueckCode
5. ohne: Ohne Suffix, Bündelung der einzelnen Makros eines Entwurfsmusters \liEntwurfsEinzelstueckAkteure

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

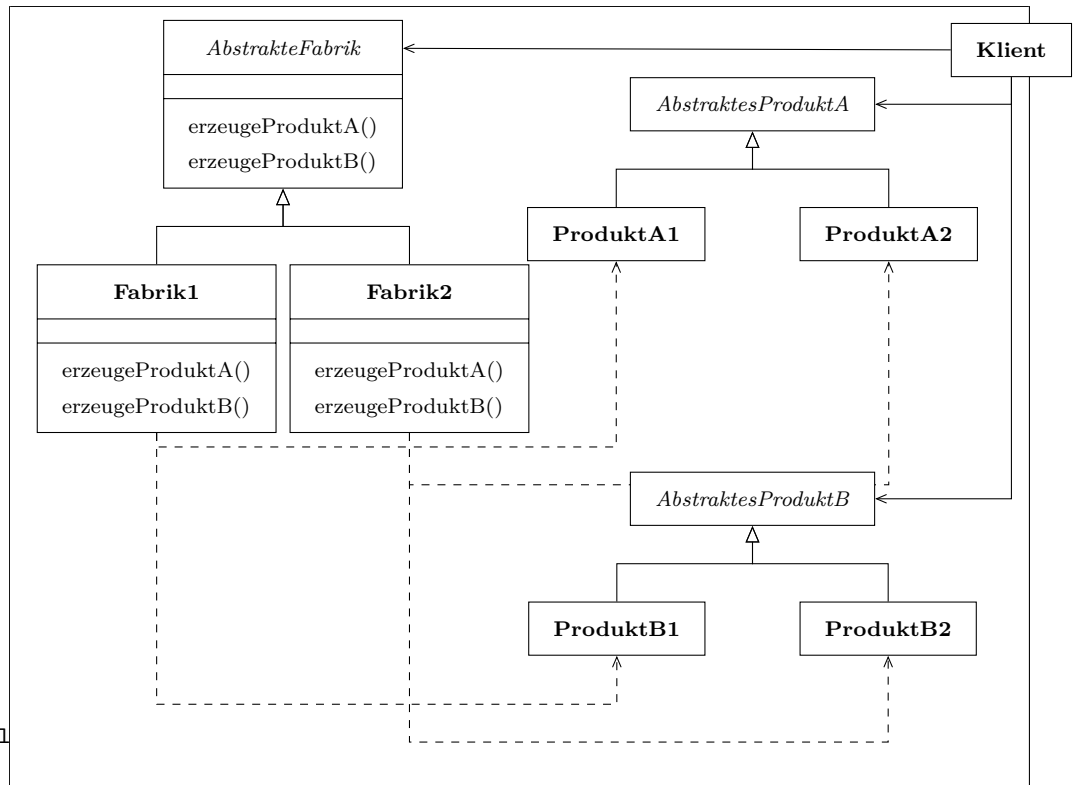
\li@EntwurfsCodeAllgemein Allgemeine Code-Beispiele zu den UML-Diagrammen und Stellvertretern

```
745 \RequirePackage{lehramt-informatik-syntax}
746 \def\li@EntwurfsCode#1#2{
747   \liJavaDatei{entwurfsmuster/#1/allgemein/#2}
748 }
```

### 2.11.3 Abstrakte Fabrik (Abstract Factory)

sAbstrakteFabrikBeschreibung

```
749 \def\liEntwurfsAbstrakteFabrikBeschreibung{
750   Es wird eine Schnittstelle bereitgestellt, um \emph{Familien
751   verbundener oder abhängiger Objekte} zu erstellen, ohne die konkreten
752   Klassen zu spezifizieren.\footcite[Seite 25]{eilebrecht}
753 }
```



```

754 \def\liEntwurfsAbstrakteFabrikUml{
755   \begin{tikzpicture}
756     \umlclass[type=abstract]{AbstrakteFabrik}{-}{-}{
757       erzeugeProduktA()\n
758       erzeugeProduktB()\n
759     }
760     \umlclass[below left=1cm and -1.5cm of AbstrakteFabrik]{Fabrik1}{-}{-}{
761       erzeugeProduktA()\n
762       erzeugeProduktB()\n
763     }
764     \umlclass[below right=1cm and -1.5cm of AbstrakteFabrik]{Fabrik2}{-}{-}{
765       erzeugeProduktA()\n
766       erzeugeProduktB()\n
767     }
768     \umlVHVinherit{Fabrik1}{AbstrakteFabrik}
769     \umlVHVinherit{Fabrik2}{AbstrakteFabrik}
770
771     \umlsimpleclass[right=3cm of AbstrakteFabrik,type=abstract]{AbstraktesProduktA}
772     \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktA]{ProduktA1}
773     \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktA]{ProduktA2}
774     \umlVHVinherit{ProduktA1}{AbstraktesProduktA}
775     \umlVHVinherit{ProduktA2}{AbstraktesProduktA}
776
777     \umlsimpleclass[above right=0cm and 1cm of AbstraktesProduktA]{Klient}
778
779     \umlsimpleclass[below=4.5cm of AbstraktesProduktA,type=abstract]{AbstraktesProduktB}
780     \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktB]{ProduktB1}
781     \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktB]{ProduktB2}
782     \umlVHVinherit{ProduktB1}{AbstraktesProduktB}
783     \umlVHVinherit{ProduktB2}{AbstraktesProduktB}
784
785     \umlVHVdep[arm1=-1.3cm,arm2=-1cm]{Fabrik1}{ProduktA1}
786     \umlVHVdep[arm2=-1cm]{Fabrik1}{ProduktB1}
787
788     \umlVHVdep[arm1=-1.8cm,arm2=-1.5cm]{Fabrik2}{ProduktA2}
789     \umlVHVdep[arm2=-1.5cm]{Fabrik2}{ProduktB2}
790

```

```

791 \umluniassoc[anchor2=25]{Klient}{AbstrakteFabrik}
792 \umlVHuniassoc[arm1=-1cm]{Klient}{AbstraktesProduktA}
793 \umlVHuniassoc{Klient}{AbstraktesProduktB}
794 \end{tikzpicture}
795 }

```

iEntwurfsAbstrakteFabrikCode

```

796 \def\liEntwurfsAbstrakteFabrikCode{
797 \li@EntwurfsCode{abstrakte_fabrik}{Produkte}
798 \li@EntwurfsCode{abstrakte_fabrik}{AbstrakteFabrik}
799 \li@EntwurfsCode{abstrakte_fabrik}{Klient}
800 }

```

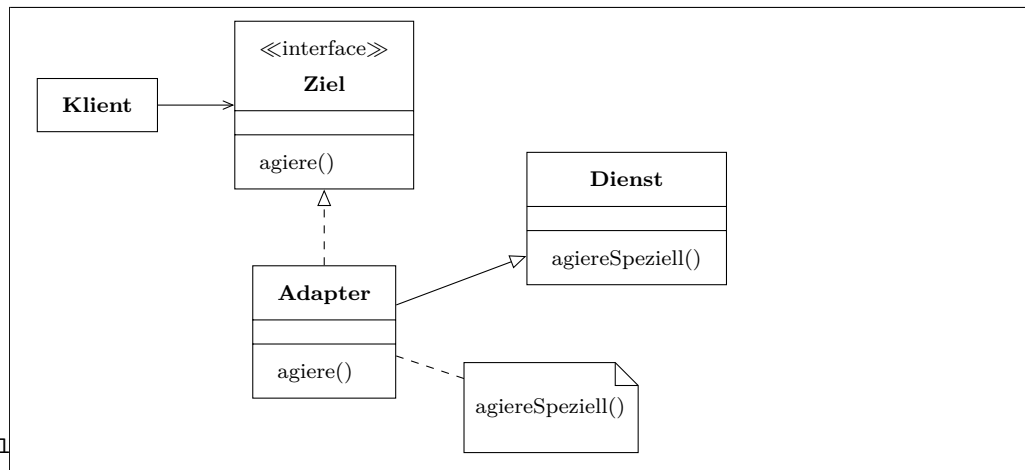
\liEntwurfsAbstrakteFabrik

```

801 \def\liEntwurfsAbstrakteFabrik{
802 \liEntwurfsAbstrakteFabrikBeschreibung
803
804 \liEntwurfsAbstrakteFabrikUml
805
806 \liEntwurfsAbstrakteFabrikCode
807 }

```

#### 2.11.4 Adapter



\liEntwurfsAdapterUml

```

808 \def\liEntwurfsAdapterUml{
809 \begin{tikzpicture}
810 \umlsimpleclass[x=1,y=3]{Klient}{}{}
811 \umlclass[x=4,y=3,type=interface]{Ziel}{}{agiere()}
812 \umlclass[x=4,y=0]{Adapter}{}{agiere()}
813 \umlclass[x=8,y=1.5]{Dienst}{}{agiereSpeziell()}
814
815 \umlreal{Adapter}{Ziel}
816 \umluniassoc{Klient}{Ziel}
817 \umlinherit{Adapter}{Dienst}
818
819 \umlnote[x=7,y=-1,width=2cm]{Adapter}{agiereSpeziell()}
820 \end{tikzpicture}
821 \footcite[so ähnlich wie GoF]{wiki:adapter}
822 }

```

\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.

```

823 \def\liEntwurfsAdapterAkteure{
824   \begin{description}
825
826     \item[Ziel (Target)]
827
828     Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.
829
830     \item[Klient (Client)]
831
832     Der Klient nutzt Dienste über inkompatible Schnittstellen und greift
833     dabei auf adaptierte Schnittstellen zurück.
834
835     \item[Dienst (Adaptee)]
836
837     Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest
838     definierter Schnittstelle an.
839
840     \item[Adapter]
841
842     Der Adapter adaptiert die Schnittstelle des Dienstes auf die
843     Schnittstelle zum Klienten.\footcite{wiki:adapter}
844
845   \end{description}
846 }

```

\liEntwurfsAdapterCode

```

847 \def\liEntwurfsAdapterCode{
848   \li@EntwurfsCode{adapter}{Dienst}
849   \li@EntwurfsCode{adapter}{Ziel}
850   \li@EntwurfsCode{adapter}{Adapter}
851   \li@EntwurfsCode{adapter}{Klient}
852 }

```

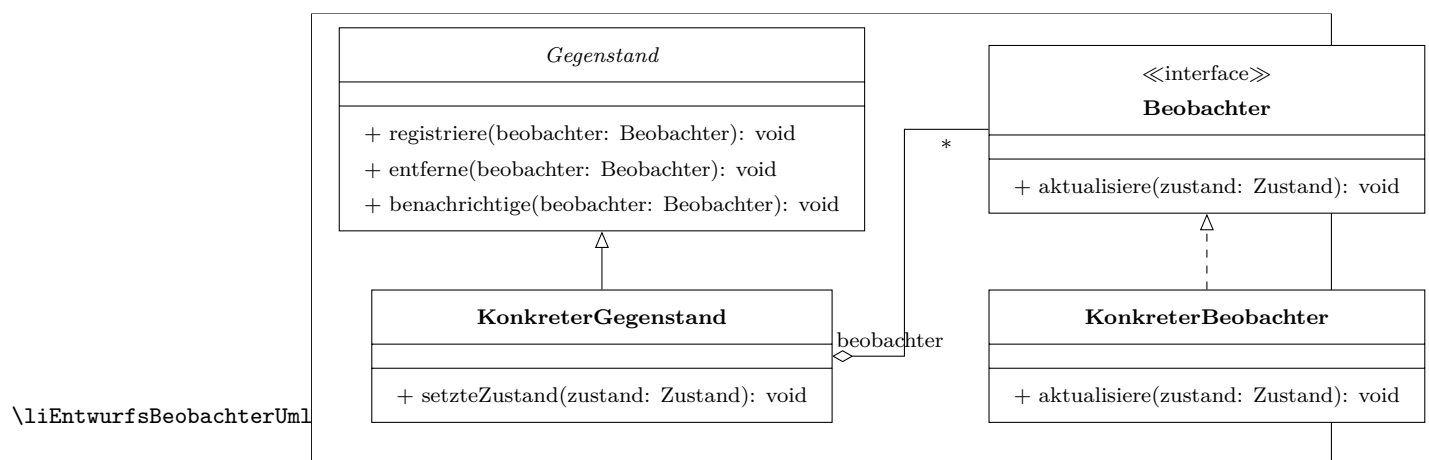
\liEntwurfsAdapter

```

853 \def\liEntwurfsAdapter{
854   \liEntwurfsAdapterUml
855   \liEntwurfsAdapterAkteure
856   \liEntwurfsAdapterCode
857 }

```

### 2.11.5 Beobachter (Observer)



```

858 \def\liEntwurfsBeobachterUml{
859   \begin{tikzpicture}
860     \umlclass[x=0,y=0,type=abstract]{Gegenstand}{}{

```

```

861     + registriere(beobachter: Beobachter): void\\
862     + entferne(beobachter: Beobachter): void\\
863     + benachrichtige(beobachter: Beobachter): void\\
864 }
865 \umlclass[x=0,y=-3]{KonkreterGegenstand}{-}{
866     + setzeZustand(zustand: Zustand): void
867 }
868 \umlinherit{KonkreterGegenstand}{Gegenstand}
869
870 \umlclass[x=8,y=0,type=interface]{Beobachter}{-}{
871     + aktualisiere(zustand: Zustand): void
872 }
873 \umlclass[x=8,y=-3]{KonkreterBeobachter}{-}{
874     + aktualisiere(zustand: Zustand): void
875 }
876 \umlreal{KonkreterBeobachter}{Beobachter}
877
878 \umlHVHaggreg[arg1=beobachter,pos1=0.8,mult2=*,pos2=2.5]
879 {KonkreterGegenstand}{Beobachter}
880 \end{tikzpicture}
881 }

```

\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.

```

882 \def\liEntwurfsBeobachterAkteure{
883   \begin{description}
884     \item[Gegenstand / Subjekt (Subject / Observable)]
885
886     Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also
887     „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne
888     deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An-
889     und Abmeldung von Beobachtern und eine Schnittstelle zur
890     Benachrichtigung von Beobachtern über Änderungen an.\footcite[Seite
891     251]{gof}
892
893     \item[Beobachter (Observer)]
894
895     Die Beobachter (auf Englisch auch subscriber, also „Abonnent“,
896     genannt) definieren eine Aktualisierungsschnittstelle.
897
898     \item[konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)]
899
900     Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert
901     den relevanten Zustand und benachrichtigt alle Beobachter bei

```



```

902 Zustandsänderungen über deren Aktualisierungsschnittstelle. Es
903 verfügt über eine Schnittstelle zur Erfragung des aktuellen
904 Zustands.
905
906 \item[Konkrete Beobachter (ConcreteObserver)]
907
908 Konkrete Beobachter verwalten die Referenz auf ein konkretes
909 Subjekt, dessen Zustand sie beobachten und speichern und dessen
910 Zustand konsistent ist. Sie implementieren eine
911 Aktualisierungsschnittstelle unter Verwendung der
912 Abfrageschnittstelle des konkreten Subjekts.
913 \footcite{wiki:beobachter}
914 \end{description}
915 }

```

\liEntwurfsBeobachterCode

```

916 \def\liEntwurfsBeobachterCode{
917 \li@EntwurfsCode{beobachter}{Gegenstand}
918 \li@EntwurfsCode{beobachter}{KonkreterGegenstand}
919 \li@EntwurfsCode{beobachter}{Beobachter}
920 \li@EntwurfsCode{beobachter}{KonkreterBeobachterA}
921 \li@EntwurfsCode{beobachter}{KonkreterBeobachterB}
922 \li@EntwurfsCode{beobachter}{Klient}
923 }

```

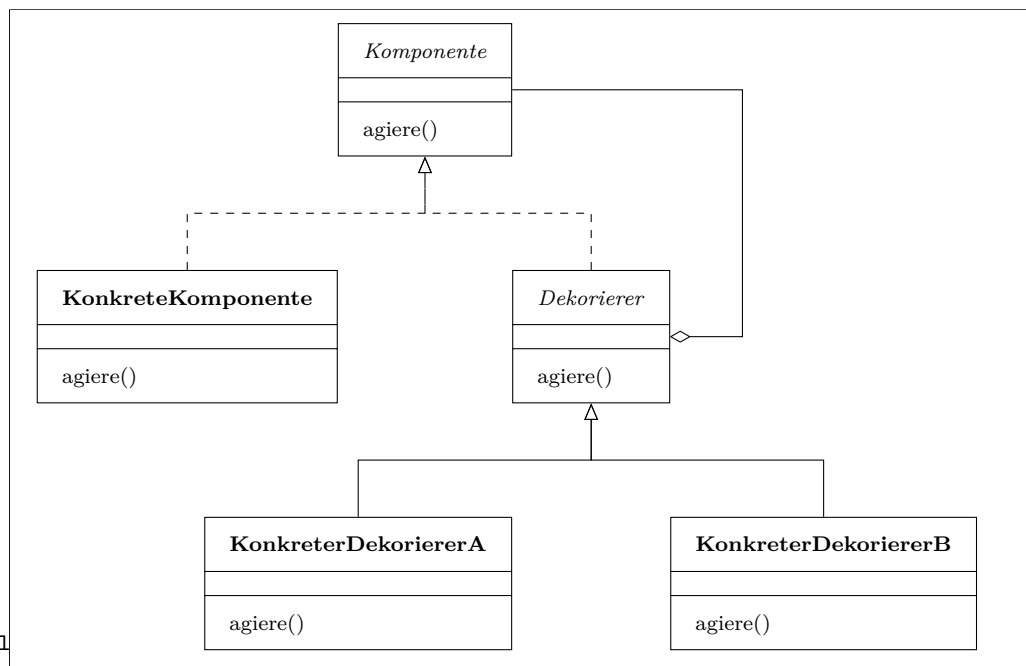
\liEntwurfsBeobachter

```

924 \def\liEntwurfsBeobachter{
925 \liEntwurfsBeobachterUml
926 \liEntwurfsBeobachterAkteure
927 \liEntwurfsBeobachterCode
928 }

```

## 2.11.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

929 \def\liEntwurfsDekoriererUml{
930 \begin{tikzpicture}
931 \umlclass[type=abstract]{Komponente}{}{agiere()}
932 \umlclass[below left=1.5cm and 0cm of Komponente]{KonkreteKomponente}{}{agiere()}
933 \umlclass[below right=1.5cm and 0cm of Komponente,type=abstract]{Dekorierer}{}{agiere()}
934

```

```

935 \umlVHVreal{KonkreteKomponente}{Komponente}
936 \umlVHVreal{Dekorierer}{Komponente}
937
938 \umlclass[below left=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererA}{-}{agiere()}
939 \umlclass[below right=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererB}{-}{agiere()}
940
941 \umlVHVinherit{KonkreterDekoriererA}{Dekorierer}
942 \umlVHVinherit{KonkreterDekoriererB}{Dekorierer}
943
944 \umlHVHaggreg[arm1=2cm]{Dekorierer}{Komponente}
945 \footcite{wiki:dekorierer}
946 \end{tikzpicture}
947 }

```

\liEntwurfsDekoriererCode

```

948 \def\liEntwurfsDekoriererCode{
949 \li@EntwurfsCode{dekorierer}{Komponente}
950 \li@EntwurfsCode{dekorierer}{KonkreteKomponente}
951 \li@EntwurfsCode{dekorierer}{Dekorierer}
952 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererA}
953 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererB}
954 \li@EntwurfsCode{dekorierer}{Klient}
955 }

```

\liEntwurfsDekorierer

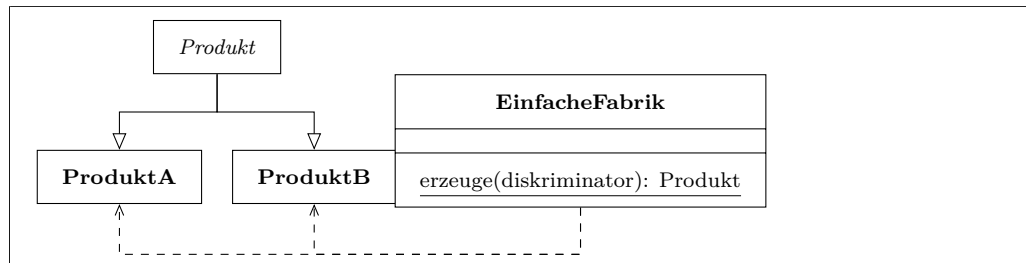
```

956 \def\liEntwurfsDekorierer{
957 \liEntwurfsDekoriererUml
958 \liEntwurfsDekoriererAkteure
959 \liEntwurfsDekoriererCode
960 }

```

### 2.11.7 Einfache Fabrik (Simple Factory)

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



```

961 \def\liEntwurfsEinfacheFabrikUml{
962 \begin{tikzpicture}
963 \umlsimpleclass[type=abstract]{Produkt}
964 \umlsimpleclass[below left=1cm and -0.65cm of Produkt]{ProduktA}
965 \umlsimpleclass[below right=1cm and -0.65cm of Produkt]{ProduktB}
966 \umlVHVinherit{Produkt}{ProduktA}
967 \umlVHVinherit{Produkt}{ProduktB}
968 \umlclass[below right=0cm and 1.5cm of Produkt]{EinfacheFabrik}{
969 }{
970 \umlstatic{erzeuge(diskriminator): Produkt}}\
971 }
972 \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktA}
973 \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktB}
974 \end{tikzpicture}
975 }

```

ntwurfsEinfacheFabrikAkteure Quelle: frei überstetzt aus GoF

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

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

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

```

976 \def\liEntwurfsEinfacheFabrikAkteure{
977   \begin{description}
978     \item[EinfacheFabrik]
979
980     Eine Klasse mit einer Erzeugungsmethode, die über eine größere
981     Bedingung verschiedene Objekt instanziert.
982
983     \item[Produkt]
984
985     Eine abstrakte Klasse, die von den konkreten Produkten geerbt wird.
986
987     \item[KonkretesProdukt]
988
989     Ein konkretes Produkt, das von der einfachen Fabrik erzeugt wird.
990   \end{description}
991 }
```

\liEntwurfsEinfacheFabrik

```

992 \def\liEntwurfsEinfacheFabrik{
993   \liEntwurfsEinfacheFabrikUml
994   \liEntwurfsEinfacheFabrikAkteure
995 }
```

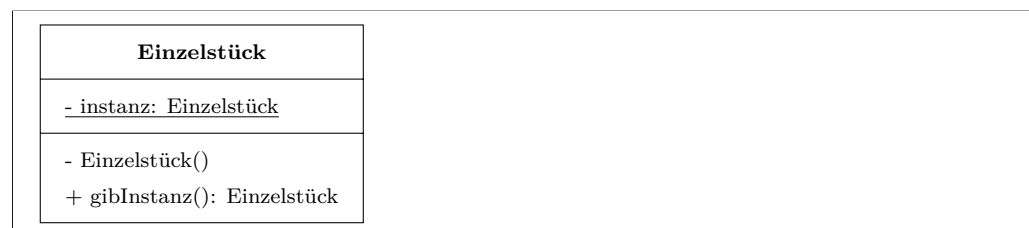
### 2.11.8 Einzelstück (Singleton)

urfsEinzelstueckBeschreibung

```

996 \def\liEntwurfsEinzelstueckBeschreibung{
997   Stellt sicher, dass nur \emph{genau eine Instanz einer Klasse} erzeugt
998   wird.\footcite[Seite 38]{eilebrecht}
999 }
1000
```

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```

1001 \def\liEntwurfsEinzelstueckUml{
1002   \begin{tikzpicture}
1003     \umlclass{Einzelstück}{
1004       \umlstatic{- instanz: Einzelstück}\\
1005     }{
1006       - Einzelstück()\\
1007       + gibInstanz(): Einzelstück
1008     }
1009   \end{tikzpicture}
1010 }
```

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.

```

1011 \def\liEntwurfsEinzelstueckAkteure{
1012   \begin{description}
1013     \item[Einzelstück (Singleton)]
1014
1015     stellt eine statische Methode bereit, mit deren Hilfe die Klienten
1016     nur auf eine einzige Instanz der Klasse zugreifen können.
1017   \end{description}
1018 }

```

\liEntwurfsEinzelstueckCode

```

1019 \def\liEntwurfsEinzelstueckCode{
1020   \li@EntwurfsCode{einzelstueck}{Einzelstueck}
1021 }

```

\liEntwurfsEinzelstueck

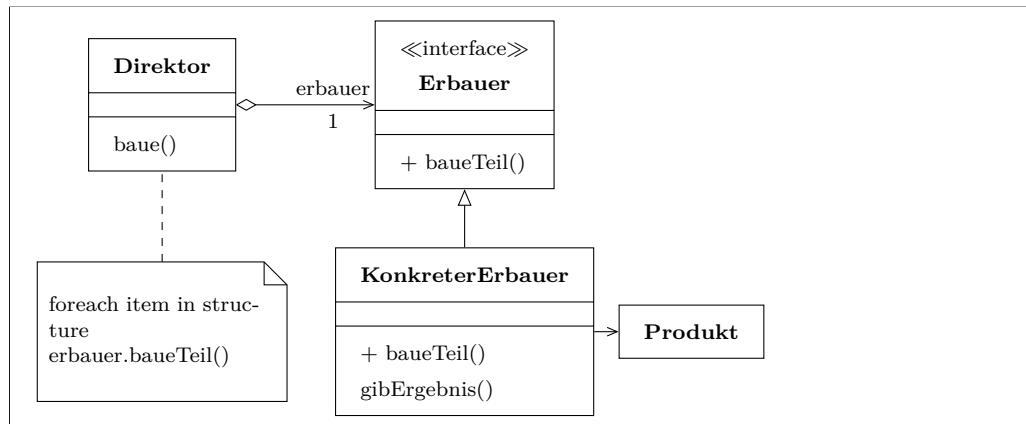
```

1022 \def\liEntwurfsEinzelstueck{
1023   \liEntwurfsEinzelstueckBeschreibung
1024
1025   \liEntwurfsEinzelstueckUml
1026
1027   \liEntwurfsEinzelstueckAkteure
1028
1029   \liEntwurfsEinzelstueckCode
1030 }

```

### 2.11.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

1031 \def\liEntwurfsErbauerUml{
1032   \begin{tikzpicture}
1033     \umlsimpleclass[x=7,y=0]{Produkt}{}{}
1034     \umlclass[x=4,y=3,type=interface]{Erbauer}{}{+ baueTeil()}
1035     \umlclass[x=4,y=0]{KonkreterErbauer}{}{
1036       + baueTeil()\n
1037       gibErgebnis()}
1038     \umlclass[x=0,y=3]{Direktor}{}{baue()}
1039
1040     \umluniagg[arg2=erbauer,pos2=0.7,mult2=1]{Direktor}{Erbauer}
1041     \umluniassoc{KonkreterErbauer}{Produkt}
1042     \umlinherit{KonkreterErbauer}{Erbauer}
1043
1044     \umlnote[x=0,y=0,width=3cm]{Direktor}{
1045       foreach item in structure\n
1046       erbauer.baueTeil()

```

```

1047 }
1048 \end{tikzpicture}
1049 \footcite{wiki:erbauer}
1050 }

```

\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.

```

1051 \def\liEntwurfsErbauerAkteure{
1052   \begin{description}
1053     \item[Erbauer]
1054
1055     Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der
1056     Teile eines komplexen Objektes.
1057
1058     \item[KonkreterErbauer]
1059
1060     Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch
1061     Implementierung der Schnittstelle. Außerdem definiert und verwaltet er
1062     die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine
1063     Schnittstelle zum Auslesen des Produkts.
1064
1065     \item[Direktor]
1066
1067     Der Direktor konstruiert ein komplexes Objekt unter Verwendung der
1068     Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer
1069     zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder
1070     benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom
1071     Klienten.
1072
1073     \item[Produkt]
1074
1075     Das Produkt repräsentiert das zu konstruierende komplexe Objekt.
1076     \footcite{wiki:erbauer}
1077   \end{description}
1078 }

```

\liEntwurfsErbauer

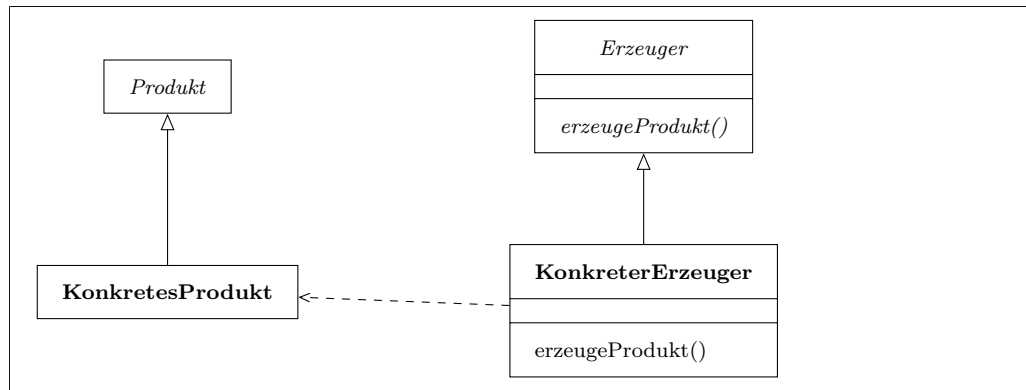
```

1079 \def\liEntwurfsErbauer{
1080   \liEntwurfsErbauerUml
1081   \liEntwurfsErbauerAkteure
1082 }

```

## 2.11.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

1083 \def\liEntwurfsFabrikmethodeUml{
1084   \begin{tikzpicture}
1085     \umlsimpleclass[type=abstract]{Produkt}
1086     \umlsimpleclass[below=2cm of Produkt]{KonkretesProdukt}
1087     \umlinherit{KonkretesProdukt}{Produkt}
1088
1089     \umlclass[type=abstract,right=4cm of Produkt]{Erzeuger}{\{
1090       \textit{erzeugeProdukt()}\}
1091   }
1092   \umlclass[below=1.2cm of Erzeuger]{KonkreterErzeuger}{\{
1093     erzeugeProdukt()
1094   }
1095   \umlinherit{KonkreterErzeuger}{Erzeuger}
1096
1097   \umldep{KonkreterErzeuger}{KonkretesProdukt}
1098 \end{tikzpicture}
1099 }

```

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).

```

1100 \def\liEntwurfsFabrikmethodeAkteure{
1101   \begin{description}
1102     \item[Produkt]
1103
1104     Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das
1105     zu erzeugende Produkt.
1106
1107     \item[KonkretesProdukt]
1108
1109     KonkretesProdukt implementiert die Produkt-Schnittstelle.
1110
1111     \item[Erzeuger]
1112
1113     Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt
1114     zu erzeugen und kann eine Default-Implementierung beinhalten.
1115
1116     \item[KonkreterErzeuger]
1117

```

```

1118     KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm
1119     entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den
1120     Konstruktor einer konkreten Produkt-Klasse aufruft).
1121
1122     \footcite{wiki:fabrikmethode}
1123 \end{description}
1124 }

```

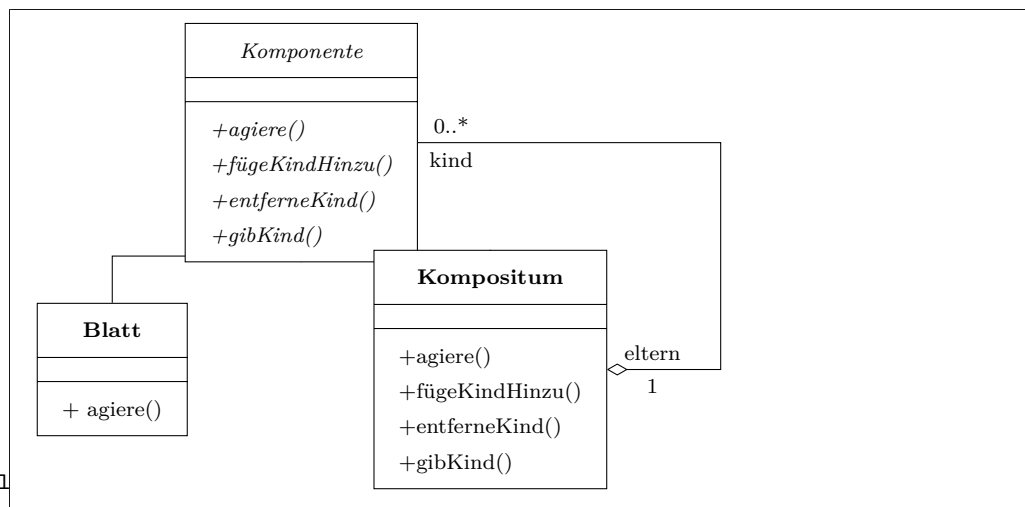
\liEntwurfsFabrikmethode

```

1125 \def\liEntwurfsFabrikmethode{
1126   \liEntwurfsFabrikmethodeUml
1127   \liEntwurfsFabrikmethodeAkteure
1128 }

```

### 2.11.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

1129 \def\liEntwurfsKompositumUml{
1130   \begin{tikzpicture}
1131     \umlclass[x=2.5,y=3,type=abstract]{Komponente}{-}{
1132       \textit{+agiere()}\
1133       \textit{+fügeKindHinzu()}\
1134       \textit{+entferneKind()}\
1135       \textit{+gibKind()}
1136     }
1137     \umlclass[x=0]{Blatt}{-}{+ agiere()}
1138     \umlclass[x=5]{Kompositum}{-}{
1139       +agiere()\
1140       +fügeKindHinzu()\
1141       +entferneKind()\
1142       +gibKind()
1143     }
1144
1145     \umlVHVinherit{Kompositum}{Komponente}
1146     \umlVHVinherit{Blatt}{Komponente}
1147     \umlHVHaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,]
1148   \end{tikzpicture}
1149 }

```

\liEntwurfsFabrikmethode

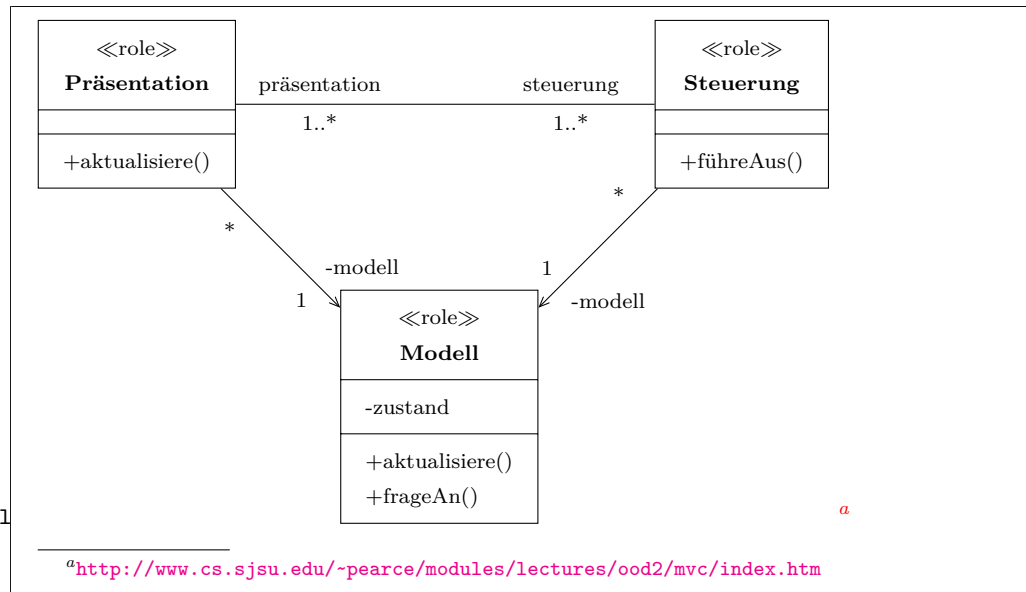
```

1150 \def\liEntwurfsKompositum{
1151   \liEntwurfsKompositumUml
1152   \liEntwurfsKompositumAkteure
1153 }

```

### 2.11.12 Modell-Präsentation-Steuerung (Model-View-Controller)

ModellPraesentationSteuerungUml



```

1154 \def\liEntwurfsModellPraesentationSteuerungUml{
1155   \begin{tikzpicture}
1156     \umlclass[x=-4cm,type=role]{Präsentation}{+aktualisiere()}
1157     \umlclass[x=4cm,type=role]{Steuerung}{+führeAus()}
1158     \umlclass[y=-4cm,type=role]{Modell}{
1159       -zustand
1160     }{
1161       +aktualisiere()\
1162       +frageAn()
1163     }
1164
1165     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
1166     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}
1167     \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
1168   \end{tikzpicture}
1169   \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
1170 }

```

ModellPraesentationSteuerung

```

1171 \def\liEntwurfs{
1172   \liEntwurfsModellPraesentationSteuerungUml
1173   \liEntwurfsModellPraesentationSteuerungAkteure
1174 }

```

### 2.11.13 Stellvertreter (Proxy)

\liEntwurfsStellvertreterUml

```

1175 \def\liEntwurfsStellvertreterUml{
1176   \begin{tikzpicture}
1177     \umlsimpleclass[x=-1,y=2]{Klient}
1178
1179     \umlclass[x=2,y=2]{Subjekt}{+ agiere()}
1180     \umlclass[x=0,y=-1]{KonkretesSubjekt}{+ agiere()}
1181     \umlclass[x=4,y=-1]{Stellvertreter}{+ agiere()}
1182
1183     \umlVHVinherit{KonkretesSubjekt}{Subjekt}
1184     \umlVHVinherit{Stellvertreter}{Subjekt}
1185     \umluniassoc{Stellvertreter}{KonkretesSubjekt}
1186     \umluniassoc{Klient}{Subjekt}
1187   \end{tikzpicture}
1188 }

```

\liEntwurfsStellvertreterCode



```

1189 \def\liEntwurfsStellvertreterCode{
1190   \li@EntwurfsCode{stellvertreter}{Subjekt}
1191   \li@EntwurfsCode{stellvertreter}{KonkretesSubjekt}
1192   \li@EntwurfsCode{stellvertreter}{Stellvertreter}
1193   \li@EntwurfsCode{stellvertreter}{Klient}
1194 }

```

\liEntwurfsStellvertreter

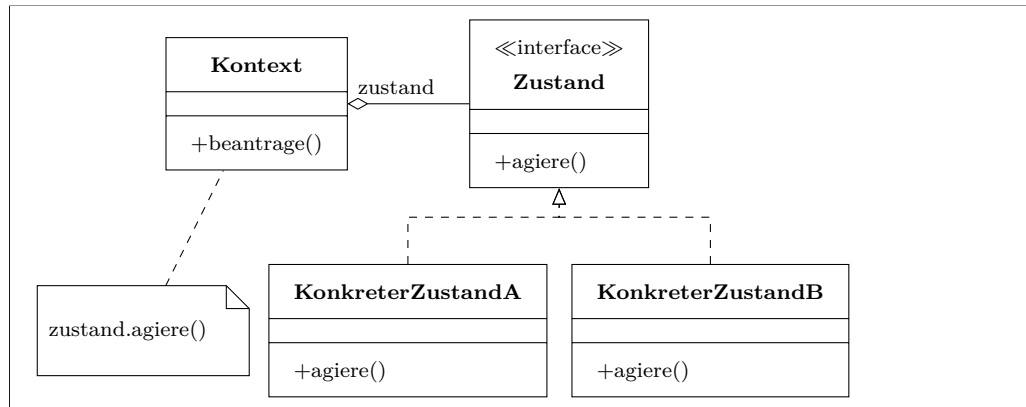
```

1195 \def\liEntwurfsStellvertreter{
1196   \liEntwurfsStellvertreterUml
1197   \liEntwurfsStellvertreterCode
1198 }

```

#### 2.11.14 Zustand (State)

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



```

1199 \def\liEntwurfsZustandUml{
1200   \begin{tikzpicture}
1201     \umlclass[x=-1,y=3]{Kontext}{{+beantrage()}}
1202     \umlclass[x=3,y=3,type=interface]{Zustand}{{+agiere()}}
1203     \umlclass[x=1,y=0]{KonkreterZustandA}{{+agiere()}}
1204     \umlclass[x=5,y=0]{KonkreterZustandB}{{+agiere()}}
1205
1206     \umlVHVreal{KonkreterZustandA}{Zustand}
1207     \umlVHVreal{KonkreterZustandB}{Zustand}
1208
1209     \umlagg[arg=zustand,pos=0.4]{Kontext}{Zustand}
1210
1211     \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
1212   \end{tikzpicture}
1213 }

```

\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.

```

1214 \def\liEntwurfsZustandAkteure{
1215   \begin{description}
1216     \item[Kontext (Context)]
1217
1218     definiert die clientseitige Schnittstelle und verwaltet die separaten

```

```

1219 Zustandsklassen.
1220
1221 \item[State (Zustand)]
1222
1223 definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
1224 implementiert gegebenenfalls ein Standardverhalten.
1225
1226 \item[KontreterZustand (ConcreteState)]
1227
1228 implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
1229 verbunden ist.
1230 \end{description}
1231 }

```

\liEntwurfsZustandCode

```

1232 \def\liEntwurfsZustandCode{
1233 \li@EntwurfsCode{zustand}{Kontext}
1234 \li@EntwurfsCode{zustand}{Zustand}
1235 }

```

\liEntwurfsZustand

```

1236 \def\liEntwurfsZustand{
1237 \liEntwurfsZustandUml
1238 \liEntwurfsZustandAkteure
1239 \liEntwurfsZustandCode
1240 }

```

```

1241

```

## 2.12 er.sty

```
1242 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1243 \ProvidesPackage{lehramt-informatik-er}[2020/06/13 Zum Setzen von
1244 ER-Diagrammen]
```

```
1245 \RequirePackage{tikz-er2}
1246 \usetikzlibrary{positioning}
```

### 2.12.1 Tex-Markup-Beispiel: Komplettes Diagramm

```
\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}

```

### 2.12.2 Tex-Markup-Beispiel: EER Enhanced Entity-Relation-Modell nach Elmasri/Navante

```

\node[below=1cm of Forscher,circle,draw] (union) {u}
  edge (Sekretär) edge (Forscher) edge (Techniker);

\node[below=1cm of union,entity] (AktiveMitarbeiter) {Aktive Mitarbeiter}
  edge node {$\bigcup$} (union);

```

```
1247 \RequirePackage{soul}
```

```
1248 \RequirePackage{fontawesome}
```

### 2.12.3 Makro-Kürzel

```

\let\a=\liErMpAttribute
\let\d=\liErDatenbankName
\let\e=\liErMpEntity
\let\r=\liErMpRelationship

```

```
1249 \ExplSyntaxOn
```

```
\liErEntity
```

```
1250 \def\liErEntity#1{\textbf{#1}}
```

```
\liErRelationship
```

```
1251 \def\liErRelationship#1{\ul{#1}}
```

```
\liErAttribute
```

```
1252 \def\liErAttribute#1{\emph{#1}}
```

```
\liErMpEntity mp = marginpar
```

**Let-Abkürzung:** \let\e=\liErMpEntity

```

1253 \def\liErMpEntity#1{
1254   \liErEntity{#1}
1255   \marginpar{
1256     \liErEntity{\tiny\faSquareO{~E:~#1}
1257   }
1258 }

```

□

```
\liErMpRelationship Let-Abkürzung: \let\r=\liErMpRelationship
```

```

1259 \def\liErMpRelationship#1{
1260   \liErRelationship{#1}
1261   \marginpar{
1262     \liErRelationship{\tiny\faGg{~R:~#1}
1263   }
1264 }

```

```
\liErMpAttribute Let-Abkürzung: \let\a=\liErMpAttribute
```

```

1265 \def\liErMpAttribute#1{
1266   \liErAttribute{#1}
1267   \marginpar{
1268     \liErAttribute{\tiny\faCircleThin{~A:~#1}
1269   }
1270 }

```

```

\liErDatenbankName Let-Abkürzung: \let\d=\liErDatenbankName
                    datenbank name
1271 \def\liErDatenbankName#1{
1272   {
1273     \footnotesize\texttt{(#1)}
1274   }
1275 }

1276 \ExplSyntaxOff
1277

```

## 2.13 formale-sprachen.sty

```

1278 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1279 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
1280 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]

1281 \directlua{
1282   formale_sprachen = require('lehramt-informatik-formale-sprachen')
1283 }

1284 \RequirePackage{hyperref}

1285 \liLadePakete{mathe,typographie}

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

1286 \def\liMengeOhneMathe#1{\{ #1 \}}
1287 \def\liMenge#1{%
1288   \ifmode%
1289     \liMengeOhneMathe{#1}%
1290   \else%
1291     $\liMengeOhneMathe{#1}$%
1292   \fi%
1293 }

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

1294 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

1295 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1296 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1297 \def\liPotenzmenge#1{$\liPotenzmengeOhneMathe{#1}$}

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

1298 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
1299 \let\liZustandsmenge=\liPotenzmenge

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

1300 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1301 \def\liUeberfuehrungsFunktion#1{
1302   \ifmode
1303     \liUeberfuehrungsFunktionOhneMathe{#1}
1304   \else
1305     $\liUeberfuehrungsFunktionOhneMathe{#1}$
1306   \fi
1307 }

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

1308 \def\liAlphabet#1{$\Sigma = \{ #1 \}$}

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

1309 \def\liBandAlphabet#1{$\Gamma = \Sigma \cup \{ #1 \}$}

\liZustandsBuchstabe

1310 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1311 \def\liZustandsBuchstabeGross{Z}

```

\liZustandsmengeNr

```

1312 \def\zustandsnamens@liste#1#2{
1313   $
1314   \{
1315     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1316   \}
1317   $
1318 }
1319 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

```

\liZustandsmengeNrGross

```

1320 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

```

\liZustandsname

```

\liZustandsname{1}: $z_1$
1321 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

```

\liZustandsnameGross

```

\liZustandsnameGross{1}: $Z_1$
1322 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

```

\liAbleitung

```

\liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1323 \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}

1324 \NewDocumentEnvironment { liProduktionsRegeln }
1325 { 0{P} +b }
1326 {
1327   \liGeschweifteKlammern{#1}
1328   {
1329     \begin{align*}
1330       \directlua{formale_sprachen.produktions_regeln('#2')}
1331     \end{align*}
1332   }{-0.2cm}{-1.5cm}
1333 } {}

```

\liProduktionen

```

\liProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
1334 \def\liProduktionen#1{
1335   \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
1336 }

```

\liZustandsnameTiefgestellt

Automatisch tiefgestellte Nummerierung \z1

**Let-Abkürzung:** \let\z=\liZustandsnameTiefgestellt

```

1337 \def\liZustandsnameTiefgestellt#1{
1338   \ifmmode
1339     \liZustandsBuchstabe\sb{#1}
1340   \else
1341     $\liZustandsBuchstabe\sb{#1}$
1342   \fi
1343 }

```

```

1344 \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}

```

```

1345 \NewDocumentCommand{ \liAusdruck } { O{L} m m } {
1346   $
1347   \tl_if_empty:nTF {#1} {} {#1 =}
1348   \{
1349     \, #2 \,
1350     |
1351     \, #3 \,
1352   \}$
1353 }
1354 \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

1355 \def\liFlaci#1{%
1356   \par
1357   {%
1358     \scriptsize
1359     Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1360     Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1361     Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1362     \href{https://flaci.com/#1}{flaci.com/#1}%
1363   }%
1364   \par
1365 }

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

1366 \ExplSyntaxOn
1367 \NewDocumentCommand {\liGrammatik} { O{G} m } {
1368   \tl_set:Nn \l_variablen_tl {V}
1369   \tl_set:Nn \l_alphabet_tl {\Sigma}
1370   \tl_set:Nn \l_produktionen_tl {P}
1371   \tl_set:Nn \l_start_tl {S}
1372
1373   \keys_define:nn { grammatik } {
1374     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1375     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1376     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1377     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1378   }
1379
1380   \keys_set:nn { grammatik } { #2 }
1381
1382   $#1 = (
1383     \l_variablen_tl,
1384     \l_alphabet_tl,
1385     \l_produktionen_tl,
1386     \l_start_tl
1387   )$
1388 }
1389 \ExplSyntaxOff
1390

```



## 2.14 formatierung.sty

```
1391 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1392 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

### 2.14.1 Schriftarten / Typographie

The package mathpazo Loading this package changes the default roman font family to Adobe Palatino, and the virtual ‘mathpazo’ fonts will be used for math. These virtual fonts are made up basically from Palatino Italic, with the missing math symbols coming from the CM and Pazo math fonts.

```
1393 \RequirePackage{mathpazo}
1394 \RequirePackage[no-math]{fontspec}
1395 \setmainfont{texgyrepagella}
```

### 2.14.2 Farben

```
1396 \RequirePackage{xcolor}
1397 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

### 2.14.3 Überschriften

```
1398 \RequirePackage{titlesec}
1399 \titleformat{\chapter}[display]{\bfseries}{\Opt}{\LARGE}
1400 \titlespacing{\chapter}{\Opt}{\Opt}{*1}
1401 \titleformat{\paragraph}[hang]{\normalsize\bfseries}{\theparagraph}{1em}{\}
1402 \setcounter{secnumdepth}{0}
```

### 2.14.4 Listen

```
1403 \RequirePackage{paralist}
1404 \renewcommand\labelitemi{-}
1405 \renewcommand\labelitemii{-}
1406 \renewcommand\labelitemiii{-}
1407 \renewcommand\labelitemiv{-}
1408 % Counter: enumi enumii enumiii enumiv
1409 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1410 \renewcommand{\labelenumi}{(\alph{enumi})}
1411 \renewcommand{\labelenumii}{(\roman{enumii})}
```

### 2.14.5 Kasten

```
1412 \RequirePackage{mdframed}
1413 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1414 \NewDocumentEnvironment { liKasten }{ } {
1415   \begin{mdframed}[backgroundcolor=white!0]
1416 } {
1417   \end{mdframed}
1418 }
```

### 2.14.6 Header

```
1419 \RequirePackage{fancyhdr}
1420 \fancyhead[L,C,R]{\}
1421 \fancyfoot[L]{\}
1422 \fancyfoot[C]{\}
1423 \fancyfoot[R]{\thepage}
1424 \pagestyle{fancy}
1425 \renewcommand{\headrulewidth}{\Opt}
1426 \renewcommand{\footrulewidth}{\Opt}
```

### 2.14.7 Zeilenabstände

Werden kleinere Schriften verwendet, passt sich der Zeilenabstand nicht entsprechend an. Mit der Umgebung spacing funktioniert es dann.

```
1427 \RequirePackage{setspace}
```



## 2.15 gantt.sty

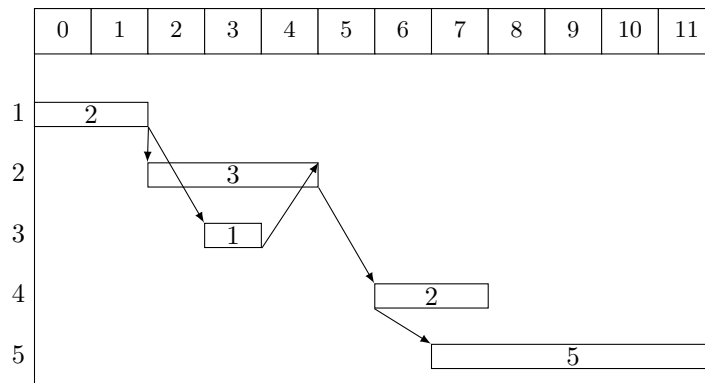
1429 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1430 \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}
```



```
1431 \RequirePackage{tikz-uml}
1432 \RequirePackage{pgfgantt}
1433 \setganttlinklabel{f-s}{}
1434 \setganttlinklabel{s-s}{}
1435 \setganttlinklabel{f-f}{}
1436 \setganttlinklabel{s-f}{}
1437
```

## 2.16 grafik.sty

```
1438 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1439 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1440 TikZ, um Grafiken zeichnen zu können und graphicx um Bilder laden zu können.]
1441 \ExplSyntaxOn
1442 \RequirePackage{tikz}
1443 \RequirePackage{graphicx}

\liGrafikLogoPfad

1444 \def\liGrafikLogoPfad#1{
1445   \LehramtInformatikRepository / .tex / Logo / #1
1446 }

\liGrafikCCLizenz

1447 \NewDocumentCommand{ \liGrafikCCLizenz } { 0{ } } {
1448   \includegraphics[#1]{
1449     \liGrafikLogoPfad{CC-by-nc-sa.eps}
1450   }
1451 }

\liGrafikLogo

1452 \NewDocumentCommand{ \liGrafikLogo } { 0{ } } {
1453   \includegraphics[#1]{
1454     \liGrafikLogoPfad{Logo_nur-Pfade.eps}
1455   }
1456 }

1457 \ExplSyntaxOff
1458
```

## 2.17 graph.sty

```

1459 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1460 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1461 \RequirePackage{tikz}

```

Für die die Adjazenzliste (`xrightharrow`)

```

1462 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

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

```

1463 \RequirePackage{blkarray}

```

```

1464 \usetikzlibrary{arrows.meta}

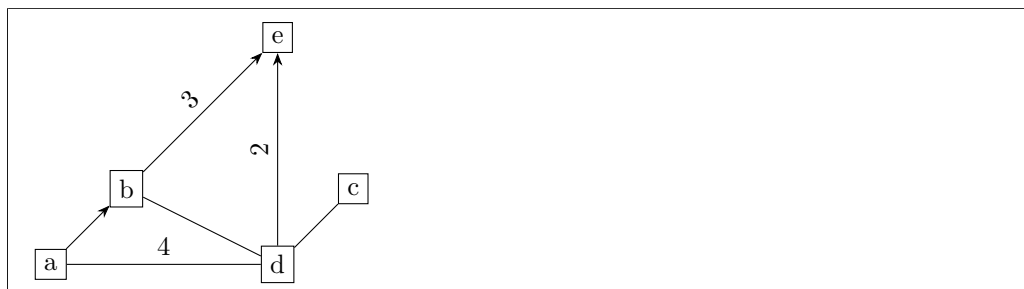
```

```

\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}

```



```

1465 \tikzset{
1466   li graph/.style={
1467     every node/.style={
1468       rectangle,
1469       draw,
1470     },
1471     every edge/.style={
1472       >={Stealth[black]},
1473       draw,
1474     },
1475     every edge/.append style={
1476       every node/.style={
1477         sloped,
1478         auto,
1479       }
1480     }
1481   },
1482   li markierung/.style={
1483     ultra thick,
1484   }
1485 }

```

**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}

```

```

1486 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1487

```

## 2.18 hanoi.sty

```
1488 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1489 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
1490 von Hanoi-Grafiken]
```

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

```
1491 \RequirePackage{tikz}
1492 \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}

1493 \def\li@mset #1[#2]=#3{%
1494 \expandafter\xdef\csname #1#2\endcsname{#3}
1495 }
1496 \def\li@mget #1[#2]{%
1497 \csname #1#2\endcsname
1498 }
1499 \def\li@minc #1[#2] += #3{%
1500 \pgfmthparse{\li@mget #1[#2] + #3}%
1501 \li@mset #1[#2] = \pgfmthresult
1502 }
1503
1504 \def\liHanoi#1#2{
1505   \edef\li@numdiscs{#1}
1506   \def\li@sequence{#2}
1507   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1508     % init colors
1509     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purp
1510     \li@mset col[\j]={\c};
1511     % draw poles and init pole counters
1512     \foreach \j in {1,2,3}{
1513       \li@mset pos[\j]=0
1514       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1515     }
1516     % draw base
1517     \draw (.5,-.5) -- +(3,0);
1518     % draw discs
1519     \foreach[count=\k] \i/\j in \li@sequence{
1520       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*
1521       \li@minc pos[\j] += {.5}
1522     }
1523   \end{tikzpicture}
1524 }

1525
```

## 2.19 klassen-konfiguration-aufgabe.sty

```
1526 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1527 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-aufgabe}[2021/09/01 Die
1528 Klasse Aufgabe konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

Formatierung muss vor literatur sein, sonst Option clash

```
1529 \liLadePakete{
1530   formatierung,
1531   abmessung,
1532   literatur-dummy,
1533   makros,
1534   aufgaben-metadaten,
1535   kopf-fusszeilen,
1536   mathe,
1537   grafik,
1538   meta
1539 }
1540
1541 % Komischer Option-Clash deshalb ganz am Ende, für die Silbentrennung
1542 \RequirePackage[ngerman]{babel}
1543 \ExplSyntaxOn
1544 \cs_gset:Npn \stichwoerter_formatiert: {
1545   \tl_if_empty:NTF \g_auf_stichwoerter_tl {} {
1546     \textbf{Stichwörter:} ~
1547     \g_auf_stichwoerter_tl
1548   \par
1549   }
1550 }
1551 \cs_gset:Npn \horizontale_linie: {
1552   \par
1553   \noindent
1554   \rule{\textwidth}{0.8pt}
1555   \par
1556 }
1557 \cs_gset:Npn \thematik_formatiert: {
1558   \tl_if_empty:NTF \g_auf_thematik_tl {} {
1559     \textit{
1560       ( \g_auf_thematik_tl )
1561     }
1562   }
1563 }
```

\liAufgabenMetadaten

```
1564 \def\liAufgabenMetadaten#1{
1565   \liMetaSetze{#1}
1566
1567   \tl_if_empty:NTF \g_auf_examen_nummer_tl {} {
1568     {
1569       \noindent
1570       \large
1571       \_gib_einzelpruefung_trenner:
1572       \par\medskip
1573     }
1574   }
1575
1576   {
1577     \noindent
1578     \bfseries
1579     \Large
1580     \tl_if_empty:NTF \g_auf_examen_nummer_tl {
1581       \g_auf_titel_tl
1582     } {
```



```

1583     \_gib_aufgaben_pfad_trenner:
1584   }
1585 }
1586 \hfill \thematik_formatiert:
1587 \par
1588
1589 \medskip
1590
1591 \noindent
1592 {\footnotesize\stichwoerter_formatiert:}
1593
1594 \horizontale_linie:
1595
1596 \bigskip
1597
1598 \par
1599 % Keine Einrückung
1600 \@afterindentfalse
1601 \@afterheading
1602 }

1603 \AddToHook{enddocument}{
1604   \vfill
1605   {
1606
1607     \begin{minipage}{5.5cm}
1608       \liGrafikLogo[width=5cm]
1609     \end{minipage}
1610     \begin{minipage}{10cm}
1611       {
1612         \bfseries
1613         \liMetaBschlangaulSammlung
1614       }\par
1615
1616       \liMetaHermineBschlangaulAndFriends\par
1617       \medskip\par
1618
1619       \begin{spacing}{1}
1620         \footnotesize
1621         \liMetaUeberDasProjekt
1622       \end{spacing}
1623     \end{minipage}
1624
1625   \par
1626
1627   \bigskip
1628
1629   \begin{minipage}{5.5cm}
1630     \centerline{\liGrafikCCLLizenz[width=3cm]}
1631   \end{minipage}
1632   \begin{minipage}{10cm}
1633     \begin{spacing}{1}
1634       \scriptsize
1635       \liMetaCCLink
1636     \end{spacing}
1637   \end{minipage}
1638
1639   \bigskip
1640
1641   \begin{spacing}{1}
1642     \tiny
1643     \noindent
1644     \liMetaHilfMit
1645

```

```
1646      \liMetaQuelltext
1647      \_gib_github_url_href:
1648      \end{spacing}
1649    }
1650 }
1651 \ExplSyntaxOff
1652
```

## 2.20 klassen-konfiguration-examen.sty

```
1653 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1654 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-examen}[2021/09/04 Die
1655 Klasse liSetzeExamen konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

Lade die wichtigsten Pakete. Formatierung muss vor literatur sein, sonst Option clash

```
1656 \liLadePakete{
1657   formatierung,
1658   literatur-dummy,
1659   makros,
1660   aufgaben-metadaten,
1661   abmessung,
1662   typographie,
1663   grafik
1664 }
```

Formatierung für die Überschriften setzen.

```
1665 \RequirePackage{titlesec}
1666 \titleformat{\section}{\huge\filcenter\bfseries}{\thesection}{1em}{}
1667 \newcommand{\sectionbreak}{\clearpage}
1668 \setcounter{secnumdepth}{0}
1669 \liLadeAllePakete
```

Komischer Option-Clash deshalb ganz am Ende, für die Silbentrennung

```
1670 \RequirePackage[ngerman]{babel}
1671 \RequirePackage{standalone}
1672 \ExplSyntaxOn
```

`\liSetzeExamenThemaNr`

```
1673 \def\liSetzeExamenThemaNr#1{
1674   \tl_gset:Nn \g_auf_examen_thema_nr_tl { #1 }
1675   \section{Thema-Nr.~#1}
1676 }
```

`\liSetzeExamenTeilaufgabeNr`

```
1677 \def\liSetzeExamenTeilaufgabeNr#1{
1678   \tl_gset:Nn \g_auf_examen_teilaufgabe_nr_tl { #1 }
1679   \subsection{Teilaufgabe-Nr.~#1}
1680 }
```

`\liBindeAufgabeEin`

```
1681 \def\liBindeAufgabeEin#1{
1682   \tl_gset:Nn \g_auf_examen_aufgabe_nr_tl { #1 }
1683   \input{
1684     \LehramtInformatikRepository /
1685     Staatsexamen /
1686     \g_auf_examen_nummer_tl /
1687     \g_auf_examen_jahr_tl /
1688     \g_auf_examen_monat_tl /
1689     \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
1690       Thema - \g_auf_examen_thema_nr_tl /
1691     }
1692     \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
1693       Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl /
1694     }
1695     Aufgabe - \g_auf_examen_aufgabe_nr_tl .tex
1696   }
1697 }
```

`\liAufgabenMetadaten` Das Metadaten-Makro überschreiben

```
1698 \def\liAufgabenMetadaten#1{
1699   \liMetaSetze{#1}
1700   \subsubsection{\_gib_aufgaben_titel:}
1701 }
```

```

1702 \setcounter{tocdepth}{4}
1703 \RequirePackage[titles]{tocloft}
1704 \AddToHook{begindocument}{
1705   \pagestyle{empty}
1706   \begin{center}
1707     \large
1708     Erste-Staatsprüfung-für-ein-Lehramt-an-öffentlichen-Schulen \par
1709
1710     \vspace{0.5cm}
1711
1712     Fach-Informatik \par
1713
1714     \vfill
1715
1716     \liGrafikLogo[width=8cm]\par
1717     \bigskip
1718     Die-Bschlangaul-Sammlung \par
1719     {\footnotesize zusammengestellt-von-Hermine-Bschlangaul-und-Freunden} \par
1720
1721     \vfill
1722
1723     {
1724       \bfseries\Huge
1725
1726       \g_auf_examen_jahreszeit_tl \par
1727
1728       \g_auf_examen_jahr_tl \par
1729     }
1730
1731     \vspace{2cm}
1732
1733     {\LARGE \g_auf_examen_nummer_tl \par}
1734
1735     \vspace{0.5cm}
1736
1737     \g_auf_examen_fach_tl \par
1738
1739     \vspace{3cm}
1740
1741     Aufgabenstellungen-mit-Lösungsvorschlägen \par
1742
1743     \end{center}
1744
1745     % für den Abstand vor den section im Inhaltsverzeichnis
1746     % https://tex.stackexchange.com/questions/241445/how-to-control-spacing-in-toc-
1747     % for-different-sections
1747     \setlength{\cftbeforesecskip}{1.5cm}
1748     \setlength{\cftbeforesubsecskip}{0.5cm}
1749     \renewcommand{\cftsubsecafterpnum}{\vspace{\cftbeforesubsecskip}}
1750     \setlength{\cftbeforesubsubsecskip}{0.1cm}
1751
1752     \vfill
1753     \renewcommand{\contentsname}{Aufgabenübersicht}
1754     \tableofcontents
1755     \clearpage
1756 }
1757 \ExplSyntaxOff
1758

```

## 2.21 komplexitaetstheorie.sty

```

1759 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1760 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1761 Setzen von Karps NP-vollständigen Problemen, Hilfsmakros für die
1762 Polynomialzeitreduktion.]

```

### 2.21.1 Makro-Kürzel

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

1763 \liLadePakete{mathe}

Für das Makro `\liProblemBeschreibung` benötigt.

```
1764 \RequirePackage{mdframed}
```

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

```
1765 \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
```

1766 \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

```

1767 \def\liProblemBeschreibung#1#2#3{
1768   \begin{mdframed}[
1769     userdefinedwidth=9cm,
1770     align=center,
1771     backgroundcolor=white!0,
1772   ]
1773     \centerline{\large\liProblemName{#1}}
1774
1775     \medskip
1776
1777     \begin{description}
1778       \item[Gegeben:] #2
1779       \item[Frage:] #3
1780     \end{description}
1781   \end{mdframed}
1782 }
```

```

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

1783 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1784 \begin{displaymath}
1785 \liProblemName{#1}
1786 \preceq_{#2}
1787 \liProblemName{#3}
1788 \end{displaymath}
1789 }

\liProblemVertexCover

1790 \def\liProblemClique{%
1791 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1792 Mindestgröße  $n$  in einem gegebenen Graphen.
1793 \footcite[Seite 76]{theo:fs:4}
1794 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1795 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1796 \footcite{wiki:cliquenproblem}
1797 }

\liProblemVertexCover

1798 \def\liProblemVertexCover{%
1799 %
1800 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1801 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1802 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1803 \footcite{wiki:knotenueberdeckung}
1804
1805 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1806 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1807 mindestens einem Knoten aus  $U$  verbunden ist.
1808 \footcite[Seite 78]{theo:fs:4}%
1809 }

\liProblemSubsetSum Kein Karp-21-Problem

1810 \def\liProblemSubsetSum{%
1811 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1812 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1813 \footcite{wiki:teilsommenproblem}
1814 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1815 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1816 größer als eine gegebene obere Schranke  $c$  ist.
1817 \footcite[Seite 74]{theo:fs:4}
1818 }

\liProblemSubsetSum Kein Karp-21-Problem

1819 \def\liProblemSat{%
1820 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1821 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1822 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1823 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1824 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1825 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1826 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1827 aufgestellt werden.
1828 \footcite[Seite 71]{theo:fs:4}
1829 }

1830

```

## 2.22 kontrollflussgraph.sty

1831 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1832 \ProvidesPackage{lehramt-informatik-kontrollflussgraph}[2020/11/07]

### 2.22.1 Makro-Kürzel

```
\let\b=\liBedingung
\let\c=\liKontrollCode
\let\f=\liBedingungFalsch
\let\k=\liKontrollTextzeileKnoten
\let\p=\liKontrollKnotenPfad
\let\w=\liBedingungWahr
```

### 2.22.2 TeX-Markup-Beispiel

```
\begin{liKontrollflussgraph}[xscale=1,yscale=-1.6]
\node[knoten] at (0,0) (S) {S};

\node[pin=\c{boolean resultat = false;}] at (0,1) (2) {2};
\node[pin=\c{if (wort != null)}] at (0,2) (3) {3};
\node[pin=\c{int laenge = wort.length;}] at (1,3) (4) {4};
\node[pin=\c{if (laenge >= 2)}] at (1,4) (5) {5};
\node[pin=\c{resultat = true; int i = 0;}] at (2,5) (6) {6};
\node[pin=\c{for (i < laenge / 2;)}] at (2,6) (7) {7};
\node[pin=\c{char c1; char c2 ...}] at (3,7) (8) {8};
\node[pin=\c{if (Char...) }] at (2,8) (10) {10};
\node[pin=\c{resultat = false; break;}] at (2,9) (12) {12};
\node[pin=\c{if (laenge == 1)}] at (1,10) (17) {17};
\node[pin=\c{resultat = true;}] at (1,11) (18) {18};
\node[pin=180:\c{return resultat;}] at (-1,9) (22) {22};
\node[knoten] at (-1,10) (E) {E};

\path (S) -- (2);
\path (2) -- (3);
\path[wahr] (3) -- (4) \b{right}{wort != null};
\path[falsch] (3) -- (22) \b{left}{wort == null};
\path (4) -- (5);
\path[wahr] (5) -- (6) \b{right}{laenge >= 2};
\path[falsch] (5) -- (17) \b{left,rotate=70,pos=0.2}{laenge < 2};
\path (6) -- (7);
\path[wahr] (7) -- (8) \b{right}{i < laenge / 2};
\path[falsch] (7) -- (22) \b{left}{i >= laenge / 2};
\path (8) -- (10);
\path[wahr] (10) -- (12) \b{right}{c != c};
\path[falsch] (10) -- (7) \b{left,rotate=70,pos=0.8}{c == c};
\path (12) -- (22);
\path[wahr] (17) -- (18) \b{right}{laenge == 1};
\path[falsch] (17) -- (22) \b{right,rotate=-20,pos=0.99}{laenge != 1};
\path (18) -- (22);
\path (22) -- (E);
\end{liKontrollflussgraph}
```

### 2.22.3 TikZ: pin

Die Knoten erhalten sogenannte Pins, in denen Code-Ausschnitte der entsprechenden Anweisung zu sehen ist.

```
pin=[options]angle:text}
```

```
\node[pin=\c{resultat = true;}] at (1,11) {18};
\node[pin=180:\c{return resultat;}] at (-1,9) {22};
\node[pin=[pin distance=3cm]\c{resultat = ...}]
```

```

1833 \RequirePackage{tikz}
1834 \usetikzlibrary{positioning}
1835 \tikzset{
1836   li kontrollfluss/.style={
1837     knoten/.style={
1838       circle,
1839       draw
1840     },
1841     usebox/.style={
1842       draw,
1843       rectangle,
1844       font=\scriptsize,
1845       anchor=west,
1846       align=left,
1847     },
1848     bedingung/.style={
1849       midway,
1850       draw=none,
1851       font=\scriptsize
1852     },
1853     knotenbeschriftung/.style={
1854       draw,
1855       rectangle,
1856       midway,
1857       font=\scriptsize
1858     },
1859     wahr/.style={
1860       thick
1861     },
1862     falsch/.style={
1863       dashed
1864     },
1865     every node/.style={
1866       circle,
1867       draw,
1868     },
1869     every edge/.append style={
1870       every node/.style={
1871         draw=none,
1872         bedingung,
1873       }
1874     },
1875     every path/.style={
1876       draw,
1877       ->,
1878     },
1879     every pin/.style={
1880       draw,
1881       dotted,
1882       rectangle,
1883       pin position=right
1884     },
1885     every pin edge/.style={
1886       dotted,
1887       arrows=-,
1888     }
1889   }
1890 }

```

#### 2.22.4 Umgebungen

liKontrollflussgraph

```

1891 \NewDocumentEnvironment { liKontrollflussgraph } { 0{} } {

```



```

1892 \begin{tikzpicture}[
1893     li kontrollfluss,
1894     #1
1895 ]
1896 } {
1897 \end{tikzpicture}
1898 }

```

## 2.22.5 Makros

\liAnweisung

```
1899 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}
```

\liBedingung **Let-Abkürzung:** \let\b=\liBedingung

```
1900 \def\liBedingung#1#2{\node[bedingung,#1]{\texttt{#2}}}
```

\liBedingungWahr **Let-Abkürzung:** \let\w=\liBedingungWahr

```
1901 \def\liBedingungWahr#1{\node[bedingung,#1]{\texttt{[true]}}}
```

\liBedingungFalsch **Let-Abkürzung:** \let\f=\liBedingungFalsch

```
1902 \def\liBedingungFalsch#1{\node[bedingung,#1]{\texttt{[false]}}}
```

\liKontrollCode **Let-Abkürzung:** \let\c=\liKontrollCode

```
1903 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}
```

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

```
1904 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw
```

\liKontrollKnotenPfad **Let-Abkürzung:** \let\p=\liKontrollKnotenPfad

```
1905 \ExplSyntaxOn
```

```
1906 \NewDocumentCommand { \liKontrollKnotenPfad } { m }
```

```
1907 {
```

```
1908   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
```

```
1909   \seq_set_map:NNn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
```

```
1910   \seq_use:Nn \l_tmpa_seq { ~~~~ }
```

```
1911 }
```

```
1912 \ExplSyntaxOff
```

```
1913
```

## 2.23 kopf-fusszeilen.sty

```
1914 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1915 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1916 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1917 \ExplSyntaxOn

1918 \fancyhead{}
1919 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1920 \fancyhead[L0,RE]{\scriptsize\today}
1921 \fancyfoot{}
1922 \fancyfoot[LE,R0]{\thepage}
1923 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1924 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1925 \renewcommand{\headrulewidth}{0.4pt}
1926 \renewcommand{\footrulewidth}{0.4pt}
1927 \setlength{\headwidth}{\textwidth}

1928 \ExplSyntaxOff

1929
```

## 2.24 literatur-dummy.sty

```
1930 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1931 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1932 \def\literatur{}

\footcite

1933 % \RequirePackage[stable,multiple]{footmisc}
1934 \NewDocumentCommand{ \footcite } { o m }{}

1935
```

## 2.25 literatur.sty

```
1936 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1937 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1938 \RequirePackage{csquotes}
1939 \RequirePackage[
1940   bibencoding=utf8,
1941   citestyle=authortitle,
1942   backend=biber,
1943 ]{biblatex}
1944 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1945 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1946 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1947 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1948 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1949 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1950 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1951 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1952 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1953 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1954 % To allow footnotes in the heading
1955 \RequirePackage[stable,multiple]{footmisc}

\literatur

1956 \def\literatur{\printbibliography[heading=subbibliography]}

1957
```

## 2.26 makros.sty

```
1958 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1959 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1960 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1961 anderen Paket passen]
1962 \RequirePackage{hyperref}
1963 \RequirePackage{graphicx}

    Für die Umgebung liQuellen benötigt.
1964 \RequirePackage{paralist}
1965 \ExplSyntaxOn

\inhaltsverzeichnis

1966 \def\inhaltsverzeichnis {
1967   \begin{mdframed}
1968     \begin{group}
1969       \let\clearpage\relax
1970       \tableofcontents
1971     \end{group}
1972   \end{mdframed}
1973 }

\memph \mephm (\marginpar and \emph)
1974 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH

1975 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1976 \newcommand{\liPseudoUeberschrift}[1]{
1977   \bigskip
1978
1979   \par
1980   \noindent
1981   \textbf{#1}
1982
1983   \medskip
1984
1985   \par
1986   % Keine Einrückung
1987   \@afterindentfalse
1988   \@afterheading
1989 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1990 \newcommand{\liBeschriftung}[1]{
1991   \par
1992   \noindent
1993   \medskip
1994   \textbf{#1}:
1995   \medskip
1996   \noindent
1997 }

\hinweis

1998 \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.
1999 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}
```

## liEinbettung

```
2000 \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.

```
2001 \NewDocumentEnvironment{ liAntwort } { O{standard} }
2002 {
2003   \str_case:nn {#1} {
2004     {standard} {
2005       \def\beschriftung{}
2006       \mdfsetup{backgroundcolor=gray!1,linecolor=gray}
2007     }
2008     {richtig} {
2009       \def\beschriftung{richtig}
2010       \mdfsetup{backgroundcolor=green!5,linecolor=green}
2011     }
2012     {falsch} {
2013       \def\beschriftung{falsch}
2014       \mdfsetup{backgroundcolor=red!3,linecolor=red}
2015     }
2016     {muster} {
2017       \def\beschriftung{Musterlösung}
2018       \mdfsetup{backgroundcolor=green!12,linecolor=green}
2019     }
2020   }
2021   \ifx\beschriftung\empty\else
2022     \noindent
2023     \textbf{\beschriftung}{}:}
2024   \fi
2025   \begin{mdframed}
2026 }
2027 {\end{mdframed}}
```

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

```
2028 \NewDocumentEnvironment{ liAdditum } { o }
2029 {
2030   \begin{mdframed}[backgroundcolor=yellow!5]
2031   \IfNoValueTF {#1}
2032   { \liPseudoUeberschrift{Additum} }
2033   { \liPseudoUeberschrift{Additum:~#1} }
2034 }
2035 {\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.

```
2036 \NewDocumentEnvironment{ liExkurs }{o +b}{
2037   \vspace{0.2cm}%
2038   \begin{mdframed}[
2039     backgroundcolor=white,
2040     bottomline=false,
2041     innermargin=1cm,
2042     leftline=true,
2043     linecolor=black,
```

```

2044 linewidth=0.1cm,
2045 outermargin=1cm,
2046 rightline=false,
2047 topline=false,
2048 ]
2049 \footnotesize
2050 \noindent%
2051 \textbf{Exkurs:~\#1}\par%
2052 \noindent%
2053 #2
2054 \end{mdframed}
2055 \vspace{0.2cm}
2056 }{}

```

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

```

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

```

**Weiterführende Literatur:**

- Quelle 1
- Quelle 2

```

2057 \cs_new:Npn \listen@punkt #1 {\item #1}
2058 \NewDocumentEnvironment { liQuellen }{ +b }
2059 {
2060   \seq_clear_new:N \l_quellen
2061   \seq_set_split:Nnn \l_quellen {\item} {#1}
2062   \seq_remove_all:Nn \l_quellen {}
2063   \begin{mdframed}[backgroundcolor=white!0]
2064     \footnotesize
2065     \noindent
2066     \textsf{\textbf{Weiterführende~Literatur:}}
2067     \medskip
2068     \begin{compactitem}
2069       \seq_map_function:NN \l_quellen {\listen@punkt}
2070     \end{compactitem}
2071   \end{mdframed}
2072   %
2073   \par
2074   \@afterindentfalse
2075   \@afterheading
2076 } {}

```

liLernkartei

```

2077 \NewDocumentEnvironment { liLernkartei }{ m +b }
2078 {
2079   \begin{mdframed}
2080     \footnotesize
2081     \noindent%
2082     \textbf{Lernkarteikarte:~\#1}\par%
2083     \noindent%
2084     #2
2085   \end{mdframed}
2086 } {}

```

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

```

2087 \NewDocumentEnvironment { liDiagramm }{ m +b }
2088 {

```

```

2089 \begin{mdframed}[backgroundcolor=white!0]
2090 \small
2091 \noindent%
2092 \textit{\#1}:
2093 \begin{center}
2094 #2
2095 \medskip
2096 \end{center}
2097 \end{mdframed}
2098 } {}

\liFussnoteUrl \liFussnoteUrl[\langlezusätzlicher-text\rangle]{\langle url\rangle}\liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.

2099 \NewDocumentCommand{\liFussnoteUrl}{ o m } {
2100 \footnote{\url{\#2}\IfNoValueTF{\#1}{ }\{ (\#1)\}}
2101 }
2102

\liFussnoteLink \liFussnoteLink[\langlezusätzlicher-text\rangle]{\langle link-text\rangle}{\langle url\rangle}\liFussnoteLink[zusätzlicher Text]{text}
Einen Link, d. h. einen Link-Text und eine URL als Fußnote setzen.

2103 \NewDocumentCommand{\liFussnoteLink}{ o m m } {
2104 \footnote{\href{\#3}{\#2}\IfNoValueTF{\#1}{ }\{ (\#1)\}}
2105 }


\zB
2106 \def\zB{z.\.,B. }

\ZB
2107 \def\ZB{Z.\.,B. }

\dH
2108 \def\dH{d.\.,h. }

2109 \ExplSyntaxOff
2110
```

`\liFussnoteUrl`    `\liFussnoteUrl[[zusätzlicher-text]]{<url>}`    `\liFussnoteUrl[zusätzlicher Text]{url}`:  
Eine HTTP-URL als Fußnote setzen.

```

2099 \NewDocumentCommand{\liFussnoteUrl} { o m } {
2100   \footnote{\url{#2}\IfNoValueTF{#1}{}{ (#1)}}
2101 }
2102

```

`\liFussnoteLink`    `\liFussnoteLink[<zusätzlicher-text>]{<link-text>}{<url>}`    `\liFussnoteLink[zusätzlicher Text]{text}`  
Einen Link, d. h. einen Link-Text und eine URL als Fußnote setzen.

```

2103 \NewDocumentCommand{\liFussnoteLink} { o m m } {
2104   \footnote{\href{#3}{#2}\IfNoValueTF{#1}{-}{( #1)}}
2105 }

```

 $\setminus zB$ 

2106 \def\zB{z.\,B. }

\ZB

```
2107 \def\ZB{Z.\,B. }
```

 $\backslash dh$ 

```
2108 \def\dh{d.\,h. }
```

2109 \ExplSyntaxOff

2110



## 2.27 master-theorem.sty

2111 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2112 \ProvidesPackage{lehramt-informatik-master-theorem}[2021/04/13]

### 2.27.1 Makro-Kürzel

\let\O=\liO

\let\o=\liOmega

\let\T=\liT

\let\t=\liTheta

\liMasterVariablenDeklaration

{3} % a

{3} % b

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

\liMasterFallRechnung

% 1. Fall

{für  $\varepsilon = 4$ : \}

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

% 2. Fall

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

% 3. Fall

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

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

\liMasterVariablenDeklaration

{ } % a

{ } % b

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

\liMasterFallRechnung

% 1. Fall

{ }

% 2. Fall

{ }

% 3. Fall

{ }

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

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

2113 \ExplSyntaxOn

2114 \RequirePackage{amsmath}

\liRundeKlammer

2115 \def\liRundeKlammer#1{

2116 \negthinspace \left( #1 \right)

2117 }

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

2118 \def\liThetaOhneMathe#1{

2119 \Theta \liRundeKlammer{#1}

2120 }

2121 \def\liTheta#1{

2122 \ifmmode

2123 \liThetaOhneMathe{#1}

2124 \else

2125 \$\liThetaOhneMathe{#1}\$

2126 \fi

2127 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
2128 \def\liOmegaOhneMathe#1{
2129 \Omega \liRundeKlammer{#1}
2130 }
2131 \def\liOmega#1{
2132 \ifmmode
2133 \liOmegaOhneMathe{#1}
2134 \else
2135 $\liOmegaOhneMathe{#1}$
2136 \fi
2137 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
2138 \def\liOOhneMathe#1{
2139 \mathcal{O} \liRundeKlammer{#1}
2140 }
2141 \def\liO#1{
2142 \ifmmode
2143 \liOOhneMathe{#1}
2144 \else
2145 $\liOOhneMathe{#1}$
2146 \fi
2147 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
2148 \def\liTOhneMathe#1#2{
2149 \tl_if_blank:nTF {#1}
2150 {}
2151 {#1 \cdot }
2152 T
2153 \liRundeKlammer{\frac{n}{#2}}
2154 }
2155 \def\liT#1#2{
2156 \ifmmode
2157 \liTOhneMathe{#1}{#2}
2158 \else
2159 $\liTOhneMathe{#1}{#2}$
2160 \fi
2161 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
2162 \def\liRekursionsGleichung{
2163 $T(n) = \liT{a}{b} + f(n)$
2164 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
2165 \def\liBedingungEins{
2166 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
2167 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
2168 \def\liBedingungZwei{
2169 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
2170 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
2171 \def\liBedingungDrei{
2172 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
2173 }

2174 \ExplSyntaxOff

```

\liMasterVariablen

```

2175 \def\liMasterVariablen{
2176   \begin{displaymath}
2177     T(n) = \liT{a}{b} + f(n)
2178   \end{displaymath}
2179
2180   \begin{itemize}
2181     \item[$a = $]
2182       Anzahl der rekursiven Aufrufe, Anzahl der Unterprobleme in der
2183       Rekursion
2184       ($a \geq 1$).
2185
2186     \item[$\textstyle{\frac{1}{b}} = $]
2187       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
2188       repräsentiert wird, Anteil an der Verkleinerung des Problems ($b > 1$).
2189
2190     \item[$f(n) = $]
2191       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
2192       die Kombination der Teillösungen entstehen. Eine von $T(n)$
2193       unabhängige und nicht negative Funktion.
2194   \end{itemize}
2195   \footcite{wiki:master-theorem}
2196   \footcite[Seite 19-35]{aud:fs:2}
2197 }
```

\liMasterFaelle

```

2198 \def\liMasterFaelle{
2199   \begin{description}
2200     \item[1. Fall:]
2201       $T(n) \in \liTheta{n^{\log\sb{b}a}}$
2202
2203       \hfill falls \liBedingungEins
2204       für $\varepsilon > 0$
2205
2206     \item[2. Fall:]
2207       $T(n) \in \liTheta{n^{\log\sb{b}a} \cdot \log n}$
2208
2209       \hfill falls \liBedingungZwei
2210
2211     \item[3. Fall:]
2212       $T(n) \in \liTheta{f(n)}$
2213
2214       \hfill falls \liBedingungDrei
2215       für $\varepsilon > 0$
2216       und ebenfalls für ein $c$ mit $0 < c < 1$ und alle hinreichend großen $n$
2217       gilt:
2218       $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$
2219     \end{description}
2220 }
```

\liMasterVariablenDeklaration

```

2221 \def\liMasterVariablenDeklaration#1#2#3{
2222   \begin{description}
2223     \item[Allgemeine Rekursionsgleichung:] \strut
2224
2225     \liRekursionsGleichung
2226
2227     \item[Anzahl der rekursiven Aufrufe ($a$):] \strut
2228
2229     #1
2230
2231     \item[Anteil Verkleinerung des Problems ($b$):] \strut
2232   \end{description}
}
```

```

2233     um  $\frac{1}{b}$  also  $b = \frac{1}{a}$ 
2234
2235     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut
2236
2237      $T(n) = \frac{1}{b} + T(\frac{n}{b})$ 
2238
2239     \item[Ergibt folgende Rekursionsgleichung:] \strut
2240
2241      $T(n) = \frac{1}{b} + T(\frac{n}{b})$ 
2242 \end{description}
2243 }

\liMasterFallRechnung
2244 \def\liMasterFallRechnung#1#2#3{
2245   \begin{description}
2246     \item[1. Fall:] \liBedingungEins:
2247
2248     #1
2249
2250     \item[2. Fall:] \liBedingungZwei:
2251
2252     #2
2253
2254     \item[3. Fall:] \liBedingungDrei:
2255
2256     #3
2257   \end{description}
2258 }

\liMasterExkurs
2259 \def\liMasterExkurs{
2260   \begin{liExkurs}[Master-Theorem]
2261     \liMasterVariablen
2262
2263     \noindent
2264     Dann gilt:
2265
2266     \liMasterFaelle
2267   \end{liExkurs}
2268 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
2269 \def\liMasterWolframLink#1{
2270   Berechne die Rekursionsgleichung auf WolframAlpha:
2271   \href{https://www.wolframalpha.com/input/?i=#1}{WolframAlpha}
2272 }

2273

```

## 2.28 mathe.sty

```
2274 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2275 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
2276
2277 % for example \ltimes \rtimes
2278 %\RequirePackage{amssymb}
2279 \RequirePackage{amsmath}
2280
2281 %%
2282 % \mlq \mrq
2283 %%
2284 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
2285 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
2286
```

## 2.29 meta.sty

```
2287 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2288 \ProvidesPackage{lehramt-informatik-klassen-meta}[2021/09/10 Sammlung
2289 von Textschnipseln, die das Projekt beschreiben]
2290 \ExplSyntaxOn
```

\liMetaBschlangaulSammlung

```
2291 \def\liMetaBschlangaulSammlung{
2292   Die~Bschlangaul-Sammlung
2293 }
```

HermineBschlangaulAndFriends

```
2294 \def\liMetaHermineBschlangaulAndFriends{
2295   Hermine~Bschlangaul-and~Friends
2296 }
```

\liMetaUeberDasProjekt

```
2297 \def\liMetaUeberDasProjekt{
2298   Eine~freie~Aufgabensammlung~mit~Lösungen~
2299   von~Studierenden~für~Studierende~
2300   zur~Vorbereitung~auf~die~1.~Staatsexamensprüfungen~
2301   des~Lehramts~Informatik~in~Bayern.
2302 }
```

\liMetaCCLink

```
2303 \def\liMetaCCLink{
2304   Diese~Materialsammlung~unterliegt~den~Bestimmungen~der~
2305   \href{
2306     https://creativecommons.org/licenses/by-nc-sa/4.0/deed.de
2307   }
2308   {
2309     Creative~Commons~Namensnennung~Nicht~kommerziell~Share~Alike~4.0~
2310     International~Lizenz
2311   }.
2312 }
```

\liMetaEmailLink

```
2313 \def\liMetaEmailLink{
2314   \href{
2315     mailto:hermine.bschlangaul@gmx.net
2316   }{
2317     hermine.bschlangaul@gmx.net
2318   }
2319 }
```

\liMetaHilfMit

```
2320 \def\liMetaHilfMit{
2321   Hilf~mit!~
2322
2323   Die~Hermine~schafft~das~nicht~alleine!~
2324
2325   Das~ist~ein~Community-Projekt.~
2326
2327   Verbesserungsvorschläge,~Fehlerkorrekturen,~weitere~Lösungen~sind~
2328   herzlich~willkommen~~~egal~wie~~~per~Pull-Request~oder~per~E-Mail~an~
2329   \liMetaEmailLink.
2330 }
```

\liMetaHilfMit

```
2331 \def\liMetaQuelltext{
2332   Der~\TeX-Quelltext~dieses~Dokuments~kann~unter~folgender~
2333   URL~aufgerufen~werden:~
2334 }
```

2335 \ExplSyntaxOff

2336

## 2.30 minimierung.sty

```

2337 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2338 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
2339 Minimierungsalgorithmus von einem NEA zu einem DEA]

2340 \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

2341 \def\liFussnote#1{\$x_{#1}\$}

2342 \def\li@fussnote@text#1#2{
2343 \liFussnote{#1}
2344 \quad
2345 {\footnotesize #2}
2346 }

\liFussnoteEinsText
2347 \def\liFussnoteEinsText{
2348 \li@fussnote@text{1}
2349 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
2350 }

\liFussnoteZweiText
2351 \def\liFussnoteZweiText{
2352 \li@fussnote@text{2}
2353 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
2354 }

\liFussnoteDreiText
2355 \def\liFussnoteDreiText{
2356 \li@fussnote@text{3}

```



```

2357 {In weiteren Iterationen markierte Zustände.}
2358 }

\liFussnoteVierText
2359 \def\liFussnoteVierText{
2360   \li@fussnote@text{4}
2361   {...}
2362 }

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



2363 \def\liFussnoten{
2364   \bigskip
2365
2366   \noindent
2367   \liFussnoteEinsText
2368
2369   \noindent
2370   \liFussnoteZweiText
2371
2372   \noindent
2373   \liFussnoteDreiText
2374
2375   \noindent
2376   \liFussnoteVierText
2377 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
2378 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
2379 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
2380 \def\liZustandsPaar#1#2{
2381   $(
2382     \liZustandsPaarVariablenName_#1,
2383     \liZustandsPaarVariablenName_#2
2384   )$
2385 }

liUebergangsTabelle
2386 \renewcommand{\arraystretch}{1.4}
2387 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
2388   \liPseudoUeberschrift{Übergangstabelle}
2389   \begin{center}
2390     \begin{tabular}{r|l|l}
2391       \textbf{Zustandspaar} & \textbf{\#1} & \textbf{\#2} \\ \hline
2392     \end{tabular}
2393   \end{center}
2394 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle:

Minimierungstabelle (Table filling)
2396 \ExplSyntaxOn

```

```

2397 \def\liUeberschriftDreiecksTabelle{
2398   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
2399 }

```

\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. —

```

2400 \def\liMinimierungErklaerung{
2401   %\footcite[Seite~19]{koenig}
2402   \liParagraphMitLinien{
2403     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
2404     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
2405     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
2406     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
2407      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
2408     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
2409     somit~zueinander~( $k - 1$ )-äquivalent,~aber~nicht~ $k$ -äquivalent~und~
2410     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
2411     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
2412   }
2413 }
2414 \ExplSyntaxOff

```

2415

## 2.31 normalformen.sty

```
2416 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2417 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10
2418 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
2419 Attributhülle]
```

Mathe für textit tex etc, typographie für geschweifte Klammer

```
2420 \liLadePakete{mathe,typographie}
2421 \directlua{
2422   helper = require('lehramt-informatik-helfer')
2423   normalformen = require('lehramt-informatik-normalformen')
2424 }
```

### 2.31.1 Makro-Kürzel

```
\let\ah=\liAttributHuelle
\let\ahL=\liLinksReduktion
\let\ahl=\liLinksReduktionInline
\let\ahr=\liRechtsReduktionInline
\let\fa=\liFunktionaleAbhaengigkeit
\let\FA=\liFunktionaleAbhaengigkeiten
\let\m=\liAttributMenge
\let\r=\liRelation
\let\u=\underline
```

```
2425 \def\liTeilen#1{
2426   \directlua{tex.print(normalformen.teilen('#1'))}
2427 }
```

```
\liAttributHuelle Let-Abkürzung: \let\ah=\liAttributHuelle
                  \ah{F, \m{A, B}} AttrHülle( $F, \{A, B\}$ ) Regulärer Ausdruck zum Konvertieren
                  AttrHülle((.*)) \ah{$1}
2428 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(\#1)}
2429 \def\liAttributHuelle#1{
2430   \ifmmode
2431     \liAttributHuelleOhneMathe{\#1}
2432   \else
2433     $\liAttributHuelleOhneMathe{\#1}$
2434   \fi
2435 }
```

```
\liAttributMenge Let-Abkürzung: \let\m=\liAttributMenge
2436 \def\liAttributMenge#1{\{ \textit{\#1} \}}
```

liAHuelle

```
2437 \NewDocumentEnvironment{ liAHuelle } { +b } {
2438   \begingroup
2439   \footnotesize
2440   \begin{multline*}
2441     \#1
2442   \end{multline*}
2443   \endgroup
2444 } { }
```

```
\liLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multline
Let-Abkürzung: \let\ahL=\liLinksReduktion
                  \ahL{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}
2445 \def\liLinksReduktion#1#2#3{
2446   \shoveleft{
2447     \liAttributHuelleOhneMathe{FA,
2448       \liAttributMenge{\#1 \string\ #2}} =
2449   } \}
```

```

2450 \shoveright{
2451 \liAttributMenge{#3}
2452 } \\\
2453 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahL=\liLinksReduktionInline
\ahL{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}
\ahL{A, E}{E}{A, E, F, B, \textbf{D}}: AttrHülle( $F, \{A, E \setminus E\}$ ) =  $\{A, E, F, B, D\}$ 
2454 \def\liLinksReduktionInline#1#2#3{%
2455 {%
2456 \footnotesize%
2457 $\liAttributHuelleOhneMathe{F,
2458 \liAttributMenge{#1 \string\ #2}} =
2459 \liAttributMenge{#3}$
2460 }
2461 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahr=\liLinksReduktionInline
\ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
2462 \def\liRechtsReduktionInline#1#2#3#4{%
2463 {%
2464 \footnotesize%
2465 $\liAttributHuelleOhneMathe{
2466 F \setminus
2467 \liFunktionaleAbhaengigkeit{#1}
2468 \def\tmp{#2}\ifx\tmp\empty
2469 \else
2470 \cup \liFunktionaleAbhaengigkeit{#2}
2471 \fi
2472 ,
2473 \liAttributMenge{#3}
2474 } =
2475 \liAttributMenge{#4}$
2476 }
2477 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
\fa{A, B -> C, D}:  $\{A, B\} \rightarrow \{C, D\}$  Regulärer Ausdruck zum Konvertieren:
\$(.*?) \\\rightarrow (.*)\$ \fa{$1 -> $2}
2478 \def\liFunktionaleAbhaengigkeit#1{%
2479 \directlua{normalformen.drucke_funk_abhaengigkeit('#1', false)}%
2480 }

```

```

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

```

$$F = \left\{ \begin{array}{l} \{M\} \rightarrow \{M\}, \\ \{M\} \rightarrow \{N\}, \\ \{V\} \rightarrow \{T, P, PN\}, \\ \{P\} \rightarrow \{PN\}, \end{array} \right\}$$

```

Let-Abkürzung: \let\FA=\liFunktionaleAbhaengigkeiten
Regulärer Ausdruck zum Konvertieren: \item \$(.*) \\\rightarrow (.*)\$ \fa{(.*) \\\rightarrow
$1 -> $2;

```

```

2481 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
2482   \liGeschweifteKlammern
2483   {#1}
2484   {
2485     \begin{align*}
2486       \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
2487     \end{align*}
2488   }
2489   {-0.5cm}
2490   {-1.7cm}
2491 }

\liRelation Let-Abkürzung: \let\r=\liRelation
      \r[R3]{\u{A}, B, C}:  $R_3(\underline{A}, B, C)$  Regulärer Ausdruck zum Konvertieren:
      \$(R.*)\((.*)\)\$ \liRelation[$1]{$2}

2492 \NewDocumentCommand {\liRelation} { O{R} m } {
2493   $\directlua{
2494     local name = helper.konvertiere_tiefgestellt('#1')
2495     tex.print(name)
2496   }$(\textit{\,#2\,})
2497 }

2498

```

## 2.32 o-notation.sty

2499 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2500 \ProvidesPackage{lehramt-informatik-o-notation}[2021/09/08]

### 2.32.1 Makro-Kürzel

\let\O=\liONotationO

### 2.32.2 TeX-Markup-Beispiel: Funktionsgraphen mit pgfplots

```
\begin{tikzpicture}
  \begin{axis}[
    xlabel=$n$,
    legend entries={\f a, \f b, \f c, \f d, \f e},
    ymax=500,
    xmin=0,
    xmax=7,
    legend pos=north west,
    domain=0:7
  ]
    \addplot{sqrt(x^5) + (4 * x) - 5};
    \addplot{log2(log2(x))};
    \addplot{2^x};
    \addplot{x^2 * log10(x) + (2 * x)};
    \addplot{4^x / (log2(x))};
  \end{axis}
\end{tikzpicture}
```

2501 \ExplSyntaxOn

2502 \RequirePackage{amssymb}

2503 \RequirePackage{pgfplots}

Für echte Teilmenge \subsetneq:  $\subsetneq$

2504 \RequirePackage{amssymb}

\liRundeKlammer

```
2505 \def\liRundeKlammer#1{
2506   \negthinspace \left( #1 \right)
2507 }
```

\liONotationO  $\mathcal{O}(n^2)$

```
2508 \cs_new:Npn \o_notation_O:n #1 {
2509   \mathcal{O} \liRundeKlammer{#1}
2510 }
2511 \def\liONotationO#1{
2512   \ifmmode
2513     \o_notation_O:n { #1 }
2514   \else
2515     $ \o_notation_O:n { #1 } $
2516   \fi
2517 }
```

2518

## 2.33 petri.sty

2519 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2520 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]

### 2.33.1 Makro-Kürzel

\let\t=\liPetriTransitionsName

\let\tp=\liPetriTransPfeile

\let\k=\liPetriErreichKnotenDrei

2521 \RequirePackage{tikz}

2522 \usetikzlibrary{petri,arrows.meta}

Für die Darstellungsmatrix

2523 \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

```
2524 \def\liPetriSetzeSchluessel{%
2525   \def\TmpTransitionOne{}%
2526   \def\TmpTransitionTwo{}%
2527   \def\TmpTransitionThree{}%
2528   \def\TmpTransitionFour{}%
2529   \def\TmpTransitionFive{}%
2530   \def\TmpTransitionSix{}%
2531   \def\TmpTransitionSeven{}%
2532   \def\TmpTransitionEight{}%
2533   \def\TmpTransitionNine{}%
2534   \def\TmpTransitionTen{}%
2535   \pgfkeys{/petri/.cd,
2536     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2537     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2538     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2539     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
```

```

2540 p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
2541 p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2542 p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2543 p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2544 p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2545 p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2546 t1/.store in=\TmpTransitionOne,t1/.default=activated,
2547 t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2548 t3/.store in=\TmpTransitionThree,t3/.default=activated,
2549 t4/.store in=\TmpTransitionFour,t4/.default=activated,
2550 t5/.store in=\TmpTransitionFive,t5/.default=activated,
2551 t6/.store in=\TmpTransitionSix,t6/.default=activated,
2552 t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2553 t8/.store in=\TmpTransitionEight,t8/.default=activated,
2554 t9/.store in=\TmpTransitionNine,t9/.default=activated,
2555 t10/.store in=\TmpTransitionTen,t10/.default=activated,
2556 scale/.store in=\TmpScale,scale/.default=0.5,
2557 x/.store in=\TmpX,x/.default=5,
2558 y/.store in=\TmpY,y/.default=5,
2559 }%
2560 }

```

```

2561 \tikzset{
2562   li petri/.style={
2563     activated/.style={
2564       very thick
2565     },
2566     inhibitor/.style={
2567       {Circle[open,length=2mm,fill=white]}-
2568     }
2569   }
2570 }

```

**\liPetriTransitionsName** **Let-Abkürzung:** \let\t=\liPetriTransitionsName  
 \t\_{\d+)\\$ \t\$1

```

2571 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2572 \def\liPetriTransitionsName#1{
2573   \ifmmode
2574     \liPetriTransitionsNameOhneMathe{#1}
2575   \else
2576     $\liPetriTransitionsNameOhneMathe{#1}$
2577   \fi
2578 }

```

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

```

2579 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2580   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2581 }

```

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

```

2582 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2583 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2584 }

```



## 2.34 potenzmengen-konstruktion.sty

```
2585 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2586 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2587 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2588 \liLadePakete{formale-sprachen}
2589 \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}
    }
  }
}
```

```
2590 \def\liZustandsMengenSammlung#1#2{
2591   \liZustandsnameGross{#1}
2592   {
2593     \footnotesize
2594     \liPotenzmenge{
2595       \str_case:nn {#1} {#2
2596       }
2597     }
2598 }
```

```
\liZustandsMengenSammlungNr
```

```
2599 \def\liZustandsMengenSammlungNr#1#2{
2600   \liZustandsnameGross{#1}
2601   {
```

```
2602     \footnotesize
2603     \liZustandsmengeNr{
2604         \str_case:nn {#1} #2
2605     }
2606 }
2607 }

2608 \ExplSyntaxOff
2609
```

## 2.35 pseudo.sty

```

2610 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2611 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen
2612 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><b>Data:</b> <math>G = (V, E, w)</math>: ein zusammenhängender, ungerichteter, kantengewichteter Graph <math>\text{kruskal}(G)</math></p> <p><math>E' \leftarrow \emptyset</math>;  <math>L \leftarrow E</math>;  Sortiere die Kanten in <math>L</math> aufsteigend nach ihrem Kantengewicht.;</p> <p><b>while</b> <math>L \neq \emptyset</math> <b>do</b></p> <div style="margin-left: 20px;"> wähle eine Kante <math>e \in L</math> mit kleinstem Kantengewicht;  entferne die Kante <math>e</math> aus <math>L</math>;  <b>if</b> der Graph <math>(V, E' \cup \{e\})</math> keinen Kreis enthält <b>then</b>    <math>E' \leftarrow E' \cup \{e\}</math>;  <b>end</b> </div> <p><b>end</b></p> <p><b>Result:</b> <math>M = (V, E')</math> ist ein minimaler Spannbaum von <math>G</math>.</p>
---

```

2613 \RequirePackage[german,boxruled]{algorithm2e}

```

```

2614

```

## 2.36 pumping-lemma.sty

2615 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2616 \ProvidesPackage{lehramt-informatik-pumping-lemma}[2021/08/11 Enthält die  
 2617 Definitionen für das Pumping-Lemma in der Regulären Sprache und  
 2618 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2619 \def\liPumpingRegulaer{%
2620   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2621   alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2622    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2623   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2624
2625   \begin{enumerate}
2626     \item  $|v| \geq 1$ 
2627     (Das Wort  $v$  ist nicht leer.)
2628
2629     \item  $|uv| \leq j$ 
2630     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2631
2632     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w$  in  $L$ 
2633     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2634     Sprache  $L$ )
2635   \end{enumerate}
2636
2637   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2638   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2639 }
```

\liPumpingKontextfrei

```
2640 \def\liPumpingKontextfrei{%
2641   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2642   sich alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  zerlegen lassen in
2643    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2644
2645   \begin{enumerate}
2646     \item  $|vx| \geq 1$ 
2647     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2648
2649     \item  $|vwx| \leq j$ 
2650     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2651
2652     \item Für alle  $i$  in  $\mathbb{N}_0$  gilt  $uv^iwx^iy$  in  $L$  (Für jede
2653     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2654     Sprache  $L$ )
2655   \end{enumerate}
2656 }
2657
```

## 2.37 quicksort.sty

```

2658 % https://tex.stackexchange.com/a/142634
2659 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2660 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2661
2662 %-----
2663 % USAGE:
2664 % \QSinitialize{comma, separated, numerical, values}
2665 % \loop
2666 % \QSpivotStep
2667 % \ifnum\value{pivotcount}>0
2668 %   \QSSortStep
2669 % \repeat
2670 %-----
2671
2672 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2673 \RequirePackage{xintfrac, xinttools}
2674
2675 \RequirePackage{tikz}
2676
2677 %-----
2678 % FIRST PART: TikZ styles and macros for the actual drawing
2679 \newcounter{cellcount}% used for coordinates of the node
2680 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2681
2682 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2683 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2684 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2685 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2686 % by police of LaTeX good conduct ? )
2687 \tikzset{
2688   l/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2689   o/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2690   r/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2691   b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2692   n/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2693   g/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2694 }
2695
2696 % NOTE the b style was originally the same as the r(aised) style apart from
2697 % not being raised, but I find it nicer with a somewhat different
2698 % specification. I have not updated the images though.
2699
2700 % How the nodes are drawn depending on whether on the left of the pivot value
2701 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2702
2703 \def\DecoLEFT #1{%
2704   \xintFor* ##1 in {#1} \do
2705     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2706 }
2707
2708 \def\DecoINERT #1{%
2709   \xintFor* ##1 in {#1} \do
2710     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2711 }
2712
2713 \def\DecoRIGHT #1{%
2714   \xintFor* ##1 in {#1} \do
2715     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2716 }
2717
2718 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2719   \xintFor* ##1 in {#1} \do

```

```

2720     {\stepcounter{cellcount}}%
2721     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2722 }
2723
2724 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2725     \xintFor* ##1 in {#1} \do
2726     {\stepcounter{cellcount}}%
2727     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2728 }
2729
2730 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2731     \xintFor* ##1 in {#1} \do
2732     {\stepcounter{cellcount}}%
2733     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2734 }
2735
2736 %-----
2737 % SECOND PART: the actual sorting routines.
2738
2739 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2740 \def\QS@sort@b #1{\ifcase #1
2741     \expandafter\QS@sort@empty
2742     \or\expandafter\QS@sort@single
2743     \else\expandafter\QS@sort@c
2744     \fi
2745 }%
2746 \def\QS@sort@empty #1{}
2747 \def\QS@sort@single #1{\QSIr {#1}}
2748
2749 % This step is to pick the last as pivot.
2750 \def\QS@sort@c #1%
2751     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2752
2753 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2754 % The trick with \xintApplyUnbraced is that for example when selecting
2755 % the elements smaller than pivot, if we had been using \xintApply we
2756 % would have had at the minimum an empty brace pair. Thus we use the
2757 % "unbraced" variant, but then the \QS@select@smaller has added in
2758 % anticipation a level of braces.
2759 \def\QS@sort@d #1#2{%
2760     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2761     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2762     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2763 }%
2764 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2765 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2766 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2767
2768 %
2769 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2770 %
2771 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2772 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2773 % latter must handle correctly an empty argument.
2774
2775 %-----
2776 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QSinitialize.
2777
2778 % This draws all with suitable highlighting for the newly chosen pivots
2779 % (which will be shown raised)

```

```

2780 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2781             \let\QSIr\DecoINERT
2782             \let\QSIrr\DecoINERT
2783             \let\QSRr\DecoRIGHTwithPivot
2784 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2785             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2786             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2787 }
2788
2789 % This sorts and then draws, showing where the pivot chosen in the previous
2790 % step go. Next time they will have become "inert". If pivotcount is still at
2791 % zero on exit from \QSpivotStep, then this is the signal to stop before
2792 % executing \QSsortStep.
2793 \def\QSsortStep {\def\QSLr {\noexpand\QS@sort@a}%
2794             \def\QSRr {\noexpand\QS@sort@a}%
2795             \def\QSIr {\noexpand\QSIrr}%
2796             \let\QSIrr\relax
2797             \edef\QS@list{\QS@list}%
2798             \let\QSLr\relax
2799             \let\QSRr\relax
2800             \let\QSIr\relax
2801             \edef\QS@list{\QS@list}%
2802             \let\QSLr\DecoLEFT
2803             \let\QSIr\DecoINERTwithPivot
2804             \let\QSIrr\DecoINERT
2805             \let\QSRr\DecoRIGHT
2806 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2807             \setcounter{cellcount}{0}%
2808             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2809 }
2810
2811 \def\QSinitialize #1{%
2812     % first, we convert the comma separated values into a list of braced items
2813     % we use an \edef, and anyhow many \edef's will be used later
2814     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2815     \let\QSRr\DecoRIGHT
2816     % The \QSRr marker mutated to draw the last element as
2817     % pivot and the earlier ones with the suitable style.
2818     %
2819     % The list of marked braced items \QS@list is used both for drawing
2820     % (as here) and for doing the exchange of elements during sort.
2821     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2822             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2823 }
2824

```

## 2.38 relationale-algebra.sty

```

2825 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2826 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]
2827 \RequirePackage{amsmath}
2828 \RequirePackage{amssymb}

```

```

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

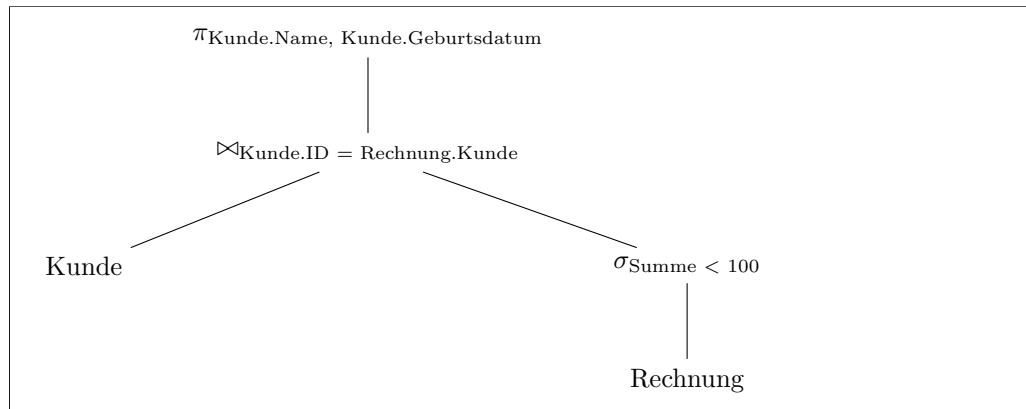
  \node[below=of pi]
    (theta join) {\bowtie_{\text{Kunde.ID = Rechnung.Kunde}}};
    edge (pi);

  \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}

```



```

2829 \RequirePackage{tikz}
2830 \usetikzlibrary{positioning}

    Privates Makros, das zwei Querstriche erzeugt.
2831 \def\o@join{\setbox0=\hbox{\bowtie}}%
2832 \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2833 }

```

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

```
2834 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}
```

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

```
2835 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}
```

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

```
2836 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
```

```
2837
```



## 2.39 rmodell.sty

```
2838 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2839 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2840 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2841 Datenbanken.]
2842 \RequirePackage{soul}
```

### 2.39.1 Makro-Kürzel

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

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

```
2843 \def\liPrimaer#1{\ul{#1}}
```

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

```
2844 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

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

```
2845 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2846 \ExplSyntaxOn
2847 \NewDocumentEnvironment { liRmodell }
2848 { +b }
2849 {
2850   \medskip
2851   {
2852     \linespread{2}
2853     \setlength{\parindent}{0pt}
2854     \li@Rmodell@Schrift#1
2855   }
2856   \medskip
2857 } {}
2858 \ExplSyntaxOff
```

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

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

```
2859 \def\liRelationMenge#1#2{
2860 \noindent
2861 #1 : \[ #2 ]\}
2862 \par
2863 }
```

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

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

```
2864 \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}
```

```
2865 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2866
```

## 2.40 sortieren.sty

```
2867 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2868 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
2869 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}
}
```



```
2870 \RequirePackage{tikz}
2871 \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.

```
2872 \def\liVertauschen#1{
2873   \directlua{
2874     local sortieren = require('lehramt-informatik-sortieren')
2875     sortieren('#1')
2876   }
2877 }
```

`\liSortierPfeil`

```
2878 \def\liSortierPfeil#1#2{
2879   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2880 }
```

`\liSortierPfeilUnten`

```
2881 \def\liSortierPfeilUnten#1#2{
2882   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2883 }
```

`\liSortierMarkierung`

```
2884 \def\liSortierMarkierung#1#2{\node[
2885   draw,
2886   very thick,
2887   fit=(reihe.#1) (reihe.#2),
2888   inner sep=0pt
2889 ] {}
2890 }

2891 \tikzset{
2892   li sortierung zahlenreihe/.style={
2893     draw,
2894     thin,
2895     font=\large,
2896     rectangle split horizontal,
2897     rectangle split,
2898   }
2899 }
```

```

2900 % https://tex.stackexchange.com/a/140895
2901 \RequirePackage{forest,xstring}
2902 \usetikzlibrary{calc}
2903
2904 \makeatletter
2905 \pgfmathdeclarefunction{strrepeat}{2}{%
2906   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2907   \let\pgfmathresult\pgfutil@empty
2908   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2909     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2910     \advance\pgfmath@count-1\relax
2911   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2912 \makeatother
2913
2914 \def\myNodes{}
2915
2916 \ExplSyntaxOn
2917 \newcommand*\sortList[1]{%
2918   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2919 \ExplSyntaxOff
2920
2921 \forestset{
2922   sort/.code={%
2923     \pgfmathparse{level()>\forestSortLevel}%
2924     \ifnum\pgfmathresult=0
2925       \StrSubstitute{\forestov{content}}{ }{,}{\myList}%
2926       \sortList\myList
2927       \StrSubstitute{\myList}{,}{ }{\myList}%
2928       \pgfmathparse{strrepeat("1",level())}%
2929       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-1!({\forestov{name}}$)
2930         (m\forestov{name}) {\myList}}%
2931       \pgfmathparse{level()=\forestSortLevel}%
2932       \ifnum\pgfmathresult=1
2933         \forestOget{\forestov{@first}}{name}\forestFirst
2934         \forestOget{\forestov{@last}}{name}\forestLast
2935         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2936           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2937       \fi
2938       \ifnum\forestov{@parent}=0\else
2939         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2940       \fi
2941       \gappto\myNodes{;}%
2942     \fi}}
2943
2944 \forestset{sort level/.code=%
2945   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2946   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2947

```

## 2.41 spalten.sty

```
2948 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2949 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2950 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2951 realisiert werden kann.]
2952 \RequirePackage{multicol}
```

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

```
2953 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2954
```

## 2.42 sql.sty

```
2955 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2956 \ProvidesPackage{lehramt-informatik-tabelle}[2021/09/02 Zu Setzen von SQL]
2957 \liLadePakete{syntax}
2958 \RequirePackage{fancyvrb}
2959 \DefineVerbatimEnvironment{liSqlErgebnis}{Verbatim}
2960 {fontsize=\footnotesize}
2961
```

## 2.43 struktogramm.sty

```
2962 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2963 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2964 Paket struktex zum Setzen von Struktogrammen]
2965 \RequirePackage{struktex}
2966
```

## 2.44 syntax.sty

```

2967 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2968 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2969 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2970 \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.

### 2.44.1 Makro-Kürzel

```

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

```

```

2971 \ExplSyntaxOn
2972 \directlua{
2973   syntax = require('lehramt-informatik-syntax')
2974   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2975   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2976   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2977   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2978   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2979   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2980 }
2981 \RequirePackage{hyperref}
2982 \RequirePackage{minted}
2983 % pygmentize -L styles
2984 \usemintedstyle{colorful}
2985 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2986 %\AfterEndEnvironment{minted}{\end{mdframed}}
2987 %\setminted{breaklines=true,linenos}
2988 \setminted{
2989   breaklines=true,
2990   linenos,
2991   fontsize=\footnotesize,
2992 }

```

**liJavaAngabe** Eine Umgebung für Java-Code, ohne Zeilennummer und etwas eingerückt um den Java-Code in Angaben / Aufgabestellungen zu setzen.

```

2993 \newminted[liJavaAngabe]{java}{
2994   xleftmargin=1cm,
2995   linenos=false
2996 }

```

**\liJavaCode** Im Zeilenfluss einen kurzen Java-Code-Ausschnitt setzen. Es werden automatische Zeilenumbrüche gemacht.

**Let-Abkürzung:** `\let\j=\liJavaCode`

```

2997 \def\liJavaCode#1{
2998   \,
2999   \textcolor{blue}{
3000     \mintinline[
3001       fontsize=\normalsize,
3002       breakanywhere % https://github.com/gpoore/minted/issues/31#issuecomment-
3003         458640242
3004     ]{java}|#1|
3005   }
3006 }

```

**\liLatexCode** Im Zeilenfluss einen kurzen L<sup>A</sup>T<sub>E</sub>X-Code-Ausschnitt setzen.

```

3007 \def\liLatexCode#1{\mintinline{latex}|#1|}

```

```

3008 \def\li@GithubLink#1#2{
3009   \begin{flushright}
3010     \tiny
3011     Code-Beispiel~auf~Github~ansehen:~
3012     \href{#1}{\nolinkurl{#2}}
3013   \end{flushright}
3014 }

\liJavaDatei Eine komplette Java-Datei einbinden, die Verzeichnis ./Code/src/main/java/org/bschlangaul
liegt.
3015 \NewDocumentCommand{\liJavaDatei}{ O{firstline=3} m }{
3016   \inputminted[#1]{java}{
3017     \directlua{
3018       syntax.drucke_absoluten_pfad('#2', false)
3019     }
3020   }
3021   \li@GithubLink
3022   {\directlua{syntax.drucke_github_url('#2', false)}}
3023   {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
3024 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
3025 \NewDocumentCommand{\liJavaTestDatei}{ O{firstline=3} m }{
3026   \inputminted[#1]{java}{
3027     \directlua{
3028       syntax.drucke_absoluten_pfad('#2', true)
3029     }
3030   }
3031   \li@GithubLink
3032   {\directlua{syntax.drucke_github_url('#2', true)}}
3033   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
3034 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([
\liJavaExamen$1{$2}{$3}{$4}{$5}
3035 \NewDocumentCommand{\liJavaExamen}{ O{firstline=3} m m m m }{
3036   \inputminted[#1]{java}{
3037     \directlua{
3038       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
3039     }
3040   }
3041   \li@GithubLink
3042   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
3043   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
3044 }

\liAssemblerCode
3046 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
3047 \NewDocumentCommand{\liAssemblerDatei}{ m }{
3048   \inputminted{asm}{#1}
3049 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
3050 \NewDocumentCommand{\liMinispracheDatei}{ m }{
3051   \inputminted{componentpascal}{#1}
3052 }

```



```

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
3053 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
3054 \NewDocumentCommand{\liHaskellDatei}{ m }{
3055   \inputminted{haskell}{#1}
3056 }

3057 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
3058 \def\liSqlCode#1{\mintinline{sql}|#1|}

3059

```

## 2.45 syntaxbaum.sty

```
3060 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3061 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
3062 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
3063 \RequirePackage{tikz-qtrees}
3064
3065 \tikzset{li parsetree/.style={
3066     every internal node/.style={
3067         draw,circle
3068     },
3069     every leaf node/.style={
3070         draw,rectangle
3071     },
3072 }
3073 }
3074
```

## 2.46 synthese-algorithmus.sty

```
3075 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3076 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
3077 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
3078 Relation in die 3. Normalform]

3079 \liLadePakete{normalformen,mathe,typographie}
3080 \ExplSyntaxOn
```

### 2.46.1 Makro-Kürzel

```
\let\schrittE=\liSyntheseUeberErklaerung
```

### 2.46.2 TeX-Markup Grundgerüst

```
\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}
```

### 2.46.3 TeX-Markup Linksreduktion

```
\let\ahl=\liLinksReduktionInline
\let\fa=\liFunktionaleAbhaengigkeit
\let\m=\liAttributMenge
\let\b=\textbf

\liPseudoUeberschrift{\fa{C, E -> D, X}}

 $\m{D, X} \in \ahl{C, E}{E}{A, C, B, \b{D, X}} \setminus$ 
 $\m{D, X} \notin \ahl{C, E}{C}{E, F}$ 

\liPseudoUeberschrift{\fa{C, E -> F}}

 $F \notin \ahl{C, E}{E}{A, C, B} \setminus$ 
 $F \in \ahl{C, E}{C}{E, \b{F}}$ 
```

### 2.46.4 TeX-Markup Rechtsreduktion

```
\let\ahr=\liRechtsReduktionInline
\let\fa=\liFunktionaleAbhaengigkeit

\liPseudoUeberschrift{F}

 $F \in \ahr{E -> F, X}{E -> F}{E}{E, \b{F}}$ 

\liPseudoUeberschrift{A}

 $A \notin \ahr{B -> A}{B}{B} \setminus$ 
 $A \in \ahr{C -> A}{C}{\b{A}, B, C}$ 
```

## 2.46.5 TeX-Markup Relationen formen

```
\let\r=\liRelation
\let\u=\underline

\r[R1]{\u{A, D}, E}\
\r[R2]{\u{B, C}, A, E}\
\r[R3]{\u{D}, B}
```

### 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  $\mathcal{R}_\alpha$  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  $\mathcal{R}_\alpha$ , 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

```
3081 \def\liSyntheseUeberschrift#1{
3082   {
3083     \bfseries
3084     \rmfamily
3085     \str_case:nn {#1} {
3086       {1} {Kanonische-Überdeckung}
3087       {1-1} {Linksreduktion}
3088       {1-2} {Rechtsreduktion}
3089       {1-3} {Löschen-leerer-Klauseln}
3090       {1-4} {Vereinigung}
3091       {2} {Relationsschemata-formen}
3092       {3} {Schlüssel-hinzufügen}
3093       {4} {Entfernung-überflüssiger-Teilschemata}
3094     }
```

```

3095   }
3096 }

\liSyntheseErklaerung Let-Abkürzung: \let\erklaerung=\liSyntheseErklaerung
3097 \def\li@synthese@erklaerung@texte#1{
3098   \str_case:nn {#1} {
3099     {1} {
3100       Die~kanonische~Überdeckung---also~die~kleinst~mögliche~noch~
3101       äquivalente~Menge~von~funktionalen~Abhängigkeiten~kann~in~vier~
3102       Schritten~erreicht~werden.
3103     }
3104     {1-1} {
3105       Führe~für~jede~funktionale~Anhängigkeit~
3106        $\alpha \rightarrow \beta$ ~in~F~die~Linksreduktion~durch,~
3107       überprüfe~also~für~alle~
3108        $A \in \alpha$ ,~ob~ $A$ ~überflüssig~ist,~d.h.~ob~
3109        $\beta \subseteqq \text{liAttributHuelle}\{F, \alpha \cup A\}$ .
3110     }
3111     {1-2} {
3112       Führe~für~jede~(verbliebene)~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~die~Rechtsreduktion~durch,~überprüfe~also~für~
3113       alle~ $B \in \beta$ ,~ob~ $B \in \text{liAttributHuelle}\{F \cup (\alpha \rightarrow \beta) \cup (\alpha \rightarrow B)\}$ ,~
3114        $\alpha$ ~gilt.~In~diesem~Fall~ist~B~auf~der~rechten~Seite~
3115       überflüssig~und~kann~eliminiert~werden,~d.h.~ $\alpha \rightarrow \beta$ ~wird~durch~ $\alpha \rightarrow (\beta \cup B)$ ~
3116       ersetzt.
3117     }
3118     {1-3} {
3119       Entferne~die~funktionalen~Abhängigkeiten~der~Form~ $\alpha \rightarrow \emptyset$ ,~die~im~2.~Schritt~möglicherweise~
3120       entstanden~sind.
3121     }
3122     {1-4} {
3123       Fasse~mittels~der~Vereinigungsregel~funktionale~Abhängigkeiten~
3124       der~Form~ $\alpha \rightarrow \beta_{\{1\}}, \dots, \alpha \rightarrow \beta_{\{n\}}$ ,~so~dass~ $\alpha \rightarrow \beta_{\{1\}} \cup \dots \cup \beta_{\{n\}}$ ~
3125       verbleibt.
3126     }
3127     % Kemper Seite 197
3128     {2} {
3129       Erzeuge~für~jede~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~in~F~ein~Relationenschema~ $\mathcal{R}_{\alpha}$ ~
3130        $:= \alpha \cup \beta$ .
3131     }
3132     {3} {
3133       Falls~eines~der~in~Schritt~2.~erzeugten~Schemata~ $\mathcal{R}_{\alpha}$ ~einen~Schlüsselkandidaten~von~ $\mathcal{R}$ ~bezüglich~ $F$ ~
3134       enthält,~sind~wir~fertig,~sonst~wähle~einen~Schlüsselkandidaten~
3135        $\mathcal{K} \subseteqq \mathcal{R}$ ~aus~und~definiere~folgendes~
3136       zusätzliche~Schema:~ $\mathcal{R}_{\mathcal{K}} := \mathcal{K} \cup \{F\}$ ~
3137       und~ $\mathcal{R}_{\mathcal{K}} \subseteqq \mathcal{R}$ .
3138     }
3139     {4} {
3140       Eliminiere~diejenigen~Schemata~ $\mathcal{R}_{\alpha}$ ,~die~in~einem~
3141       anderen~Relationenschema~ $\mathcal{R}_{\alpha'}$ ~enthalten~sind,~d.h.~
3142        $\mathcal{R}_{\alpha} \subseteqq \mathcal{R}_{\alpha'}$ .
3143     }
3144   }
3145 }
3146 }
3147 \def\liSyntheseErklaerung#1{
3148 {
3149   \itshape
3150   \footnotesize

```

```

3157     \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
3158   }
3159 }

```

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

```

3160 \def\liSyntheseUeberErklaerung#1{
3161   \liSyntheseUeberschrift{#1}\par
3162   \liSyntheseErklaerung{#1}
3163 }

```

```

3164 \ExplSyntaxOff
3165

```

## 2.47 tabelle.sty

```
3166 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3167 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]
3168 \RequirePackage{tabularx}
3169
```

## 2.48 typographie.sty

```
3170 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3171 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
3172 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
3173 formatierung.sty definiert.]
```

```
3174 \ExplSyntaxOn
```

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

```
3175 \RequirePackage{fontawesome}
```

`\liErledigt` `\liErledigt:` ✓

```
3176 \let\liErledigt=\faCheckSquareO
```

`\liNichtsZuTun` `\liNichtsZuTun:` ∅ Nichts zu tun

```
3177 \def\liNichtsZuTun{${\emptyset}$-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 —

```
3178 \def\liParagraphMitLinien#1{
3179   \noindent
3180   \vrule height 2pt depth -1.6pt width 0.4cm
3181   \enspace
3182   #1
3183   \enspace
3184   \leaders\vrule height 2pt depth -1.6pt \hfill \null
3185   \par
3186   \medskip
3187 }
```

`\liGeschweifteKlammern` Große geschweifte Klammer mit Istgleich-Zeichen.

$\text{Variable} = \left\{ \begin{array}{l} \text{Inhalt} \end{array} \right\}$
---

```
3188 \def\liGeschweifteKlammern#1#2#3#4{
3189   \par
3190   \medskip
3191   \noindent
3192   #1 \, $= \Bigl\{ $
3193   \vspace{#3}
3194   #2
3195   \vspace{#4}
3196   \begin{flushright} $\Bigr\}$ \end{flushright}
3197   \par
3198 }
```

`\liTypoUeberschrift`

```
3199 \def\liTypoUeberschrift#1{
3200   {
3201     \bfseries\rmfamily
3202     #1
3203   }
3204 }
```



\liTypoUeberGross

```
3205 \def\liTypoUeberGross#1{
3206   {
3207     \huge
3208     \liTypoUeberschrift{#1}
3209   }
3210 }
```

\liTypoUeberGross

```
3211 \def\liTypoUeberGROSS#1{
3212   {
3213     \Huge
3214     \liTypoUeberschrift{#1}
3215   }
3216 }
```

3217 \ExplSyntaxOff

3218

## 2.49 uml.sty

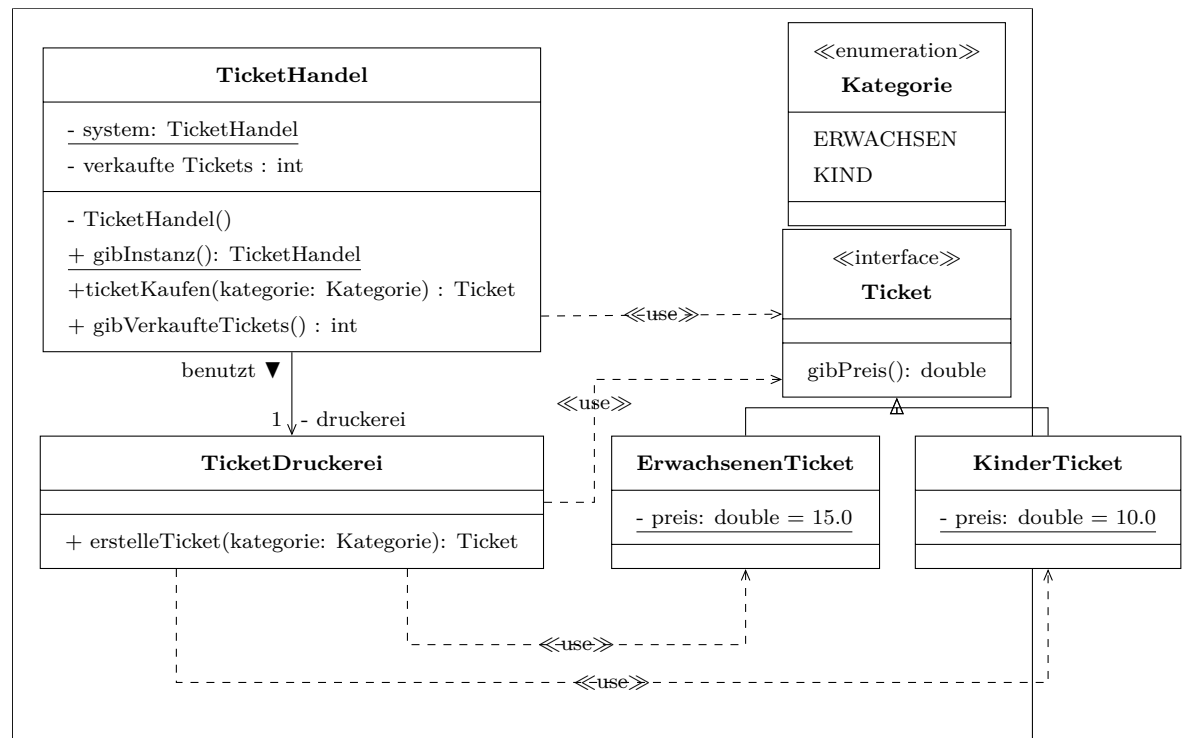
```

3219 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3220 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
3221 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
3222 Erweiterung bereitstellt]

3223 \RequirePackage{tikz-uml}
3224 \RequirePackage{tikz-uml-activity}
3225 % Not compatible with wasysym
3226 %\RequirePackage{mathabx}
3227 \RequirePackage{wasysym}
3228 \usetikzlibrary{positioning}

3229 \tikzumlset{
3230   fill class=white!0,
3231   font=\footnotesize,
3232   fill object=white!0,
3233   fill note=white!0,
3234   fill state=white!0,
3235   % Use case
3236   fill usecase=white!0,
3237   fill system=white!0,
3238 }

```



```

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

```

```

3239 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
3240   \def\@liDirLeft{}
3241   \def\@liDirRight{ \RIGHTarrow}
3242   \pgfkeys{/lese/dir/.is choice}
3243   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
3244   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
3245   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{\LEFTarrow }}}
3246   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
3247
3248   \def\@liPos{above}
3249   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
3250

```

```

3251 \def\@liDistance{0cm}
3252 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
3253
3254 \pgfkeys{/lese/.cd,#1}
3255
3256 \node[\@liPos = \@liDistance of #2-middle] {
3257   \@liDirLeft{\footnotesize#2}\@liDirRight
3258 };
3259 }
3260

```

## 2.50 vollstaendige-induktion.sty

```
3261 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3262 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01
3263 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
3264 Überschriften für die einzelnen Schritte]
```

### 2.50.1 Makro-Kürzel

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

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

Lade häufig benötigte Pakete

```
3265 \RequirePackage{lehramt-informatik-typographie}
3266 \RequirePackage{lehramt-informatik-mathe}
3267 \RequirePackage{lehramt-informatik-syntax}
3268 \ExplSyntaxOn
```

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

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

```
3269 \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

```
3270 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
3271 \def\liInduktionAnfang{
```

```
3272   \liPseudoUeberschrift{Induktionsanfang}
```

```
3273
```

```
3274   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
3275   \liParagraphMitLinien{
```

```
3276     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
```

```
3277   }
```

```
3278 }
```

\liInduktionVoraussetzung

```
3279 \def\liInduktionVoraussetzung{
```

```
3280   \liPseudoUeberschrift{Induktionsvoraussetzung}
```

```
3281
```

```
3282   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
3283   \liParagraphMitLinien{
```

```
3284     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ ~.
```

```
3285   }
```

```
3286 }
```

\liInduktionSchritt

```
3287 \def\liInduktionSchritt{
```

```
3288   \liPseudoUeberschrift{Induktionsschritt}
```

```
3289
```

```
3290   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
3291   \liParagraphMitLinien{
```

```
3292     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
```

```
3293     auch~ $A(n=k+1)$ ~wahr~sein~muss.
```

```
3294   }
```

```
3295 }
```

```
3296 \ExplSyntaxOff
```

```
3297
```

## 2.51 wasserfall.sty

```
3298 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3299 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
3300 \RequirePackage{tikz}
3301 \tikzset{wasserfall/.style={
3302   >=stealth,
3303   node distance = 2mm and -8mm,
3304   start chain = A going below right,
3305   every node/.style = {
3306     draw,
3307     text width=24mm,
3308     minimum height=12mm,
3309     align=center,
3310     inner sep=1mm,
3311     fill=white,
3312     drop shadow={fill=black},
3313     on chain=A
3314   },
3315 }}
3316 \usetikzlibrary{chains,positioning,shadows}
3317
```

## 2.52 wpkalkuel.sty

```
3318 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3319 \ProvidesPackage{lehramt-informatik-wpkalkuel}[2020/06/13]
```

### 2.52.1 Makro-Kürzel

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

```
3320 \RequirePackage{amsmath}
```

```
3321 \ExplSyntaxOn
```

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

```
3322 \def\liWpKalkuelOhneMathe#1#2{
3323   \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)
3324 }
3325 \def\liWpKalkuel#1#2{
3326   \ifmmode
3327     \liWpKalkuelOhneMathe{#1}{#2}
3328   \else
3329     $\liWpKalkuelOhneMathe{#1}{#2}$
3330   \fi
3331 }
```

```
\MatheEnv
```

```
3332 \def\MatheEnv#1{
3333   \medskip
3334
3335   \hspace{1em}#1
3336
3337   \medskip
3338 }
```

```
\Mathe
```

```
3339 \def\Mathe#1{
3340   \MatheEnv{${#1}$}
3341 }
```

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

```
3342 \def\liWpEquivalent#1{
3343   \MatheEnv{${equiv$\hspace{1em}${#1}$}
3344 }
```

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

```
3345 \newlength{\@Skip@Erklaerung@Reset}
3346 \def\liWpErklaerung#1{
3347   \setlength{\@Skip@Erklaerung@Reset}{\leftskip}
3348   \setlength{\leftskip}{0.5cm}
3349
3350   \par
3351   \noindent
3352   {
3353     \scriptsize
3354     #1
3355   }
3356   \par
3357
3358   \setlength{\leftskip}{\@Skip@Erklaerung@Reset}
3359 }
```

\liWpErklaerungVerzweigung

```

3360 \def\liWpErklaerungVerzweigung{
3361   $\liWpKalkuelOhneMathe{if~\{-b~\}~then~\{-a1~\}~else~\{-a2~\}}{Q}
3362   \equiv
3363   (b \land \liWpKalkuelOhneMathe{a1}{Q})
3364   \lor
3365   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
3366 }

3367 \ExplSyntaxOff

3368

```

### 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		
\# ..... 114	\_ ..... 2448, 2458	2645, 2786, 2808, 2822, 2985, 3009, 3196
\, .. 358, 417, 472, 699, 717, 1349, 1351, 2106, 2107, 2108, 2496, 2998, 3005, 3192	<b>A</b>	\beginngroup 1968, 2438, 2906
\@Skip@Erklaerung@Reset ... 3345, 3347, 3358	\addbibresource ..... 1944, 1945, 1946, 1947, 1948, 1949, 1950, 1951, 1952, 1953	\beschriftung ..... ..... 2005, 2009, 2013, 2017, 2021, 2023
\@afterheading ..... ... 1601, 1988, 2075	\AddToHook .... 1603, 1704	\beta ..... 3106, 3109, 3113, 3114, 3115, 3118, 3128, 3129, 3130, 3135, 3136
\@afterindentfalse .. ... 1600, 1987, 2074	\advance ..... 2910	\bf ..... 2683, 2684, 2685
\@liDirLeft 3240, 3245, 3257	\AfterEndEnvironment 2986	\bfseries ..... 558, 1399, 1401, 1578, 1612, 1666, 1724, 2683, 2689, 2691, 2693, 2694, 3083, 3201
\@liDirRight 3241, 3243, 3244, 3245, 3246, 3257	\Alph ..... 1409	\Bigl ..... 3192
\@liDistance ..... ... 3251, 3252, 3256	\alph ..... 1409, 1410	\Bigr ..... 3196
\@liPos .. 3248, 3249, 3256	\alpha 3106, 3108, 3109, 3112, 3114, 3115, 3116, 3117, 3118, 3122, 3128, 3129, 3134, 3135, 3136, 3139, 3147, 3148, 3149	\bigskip .... 450, 731, 736, 1596, 1627, 1639, 1717, 1977, 2364
\\ ..... 729, 757, 758, 761, 762, 765, 766, 861, 862, 863, 970, 1004, 1006, 1036, 1045, 1090, 1132, 1133, 1134, 1139, 1140, 1141, 1161, 1975, 2391, 2449, 2452	\arabic 1409, 2705, 2710, 2715, 2721, 2727, 2733	\bool ..... 336, 418
\{ ..... 212, 1286, 1296, 1308, 1309, 1314, 1348, 1814, 2436, 2861, 3192, 3361	\arraystretch ..... 2386	\bowtie ..... 2831, 2834, 2835, 2836
\} ..... 212, 1286, 1296, 1308, 1309, 1316, 1352, 1815, 2436, 2861, 3196, 3361	<b>B</b>	\Box ..... 164
\_ ..... 37, 45, 50, 52, 305, 333, 357, 360, 370, 393, 397, 401, 403, 405, 407, 410, 416, 417, 431, 432, 438, 441, 444, 454, 461, 463, 466, 1571, 1583, 1647, 1700	\BeforeBeginEnvironment ..... 2985	\boxtimes ..... 548
	\begin ..... 755, 809, 824, 859, 883, 930, 962, 977, 1002, 1012, 1032, 1052, 1084, 1101, 1130, 1155, 1176, 1200, 1215, 1329, 1415, 1507, 1607, 1610, 1619, 1629, 1632, 1633, 1641, 1706, 1768, 1777, 1784, 1892, 1967, 2025, 2030, 2038, 2063, 2068, 2079, 2089, 2093, 2176, 2180, 2199, 2222, 2245, 2260, 2389, 2390, 2440, 2485, 2625,	<b>C</b>
		\c ..... 1509, 1510
		\cdot .... 2151, 2207, 2218
		\centerline .... 1630, 1773, 2784, 2806, 2821
		\cftbeforesecskip .. 1747
		\cftbeforesubsecskip ..... 1748, 1749
		\cftbeforesubsubsecskip ..... 1750
		\cftsubsecafterpnum 1749
		\chapter ..... 1399, 1400
		\char ..... 1975
		\clearpage 1667, 1755, 1969
		\cline ..... 729
		\clist ..... 231, 279, 280, 302, 306, 2918



<code>\columnbreak</code> . . . . .	2953	974, 990, 1009,	673, 1249, 1344,
<code>\contentsname</code> . . . . .	1753	1017, 1048, 1077,	1366, 1441, 1543,
<code>\cs</code> . . . . .	305, 333,	1098, 1123, 1148,	1672, 1905, 1917,
	357, 360, 370, 393,	1168, 1187, 1212,	1965, 2113, 2290,
	405, 416, 417, 454,	1230, 1331, 1417,	2396, 2501, 2589,
	461, 466, 1544,	1523, 1609, 1622,	2846, 2916, 2971,
	1551, 1557, 2057, 2508	1623, 1631, 1636,	3080, 3174, 3268, 3321
<code>\csname</code> . . . . .	1494, 1497	1637, 1648, 1743,	
<code>\cup</code> . . . . .	1309,	1780, 1781, 1788,	<b>F</b>
	2470, 3115, 3129, 3136	1897, 1972, 2027,	<code>\faCheckSquare0</code> . . . .
		2035, 2054, 2070,	3176
<b>D</b>		2071, 2085, 2096,	<code>\faCircleThin</code> . . . . .
<code>\DeclareMathSymbol</code> . .		2097, 2178, 2194,	1268
	2284, 2285	2219, 2242, 2257,	<code>\faGg</code> . . . . .
<code>\DecoINERT</code> . . . . .		2267, 2393, 2394,	1262
	2708, 2781, 2782, 2804	2442, 2487, 2635,	<code>\fancyfoot</code> . . . . .
<code>\DecoINERTwithPivot</code> .		2655, 2786, 2808,	1421, 1422, 1423,
	2724, 2803	2822, 2986, 3013, 3196	1921, 1922, 1923, 1924
<code>\DecoLEFT</code> . . . . .	2703, 2802	<code>\endcsname</code> . . . .	<code>\fancyhead</code> . . . . .
<code>\DecoLEFTwithPivot</code> . .		1494, 1497	1420, 1918, 1919, 1920
	2718, 2780	<code>\endgroup</code> 1971, 2443, 2911	<code>\faSquare0</code> . . . . .
<code>\DecoRIGHT</code> 2713, 2805, 2815		<code>\enspace</code> . . . . .	1256
<code>\DecoRIGHTwithPivot</code> .		3181, 3183	<code>\fi</code> . 655, 663, 671, 679,
	2730, 2783	environments:	686, 1292, 1306,
<code>\definecolor</code> . . . . .	1397	<code>liAdditum</code> . . . . .	1342, 2024, 2126,
<code>\DefineVerbatimEnvironment</code>	2959	2028	2136, 2146, 2160,
<code>\delta</code> 70, 112, 170, 212, 1300		<code>liAHuelle</code> . . . . .	2434, 2471, 2516,
<code>\dh</code> . . . . .	2108, 3117	2437	2577, 2744, 2936,
<code>\directlua</code> . . . . .		<code>liAntwort</code> . . . . .	2937, 2940, 2942, 3330
	63, 142, 200, 205,	2001	<code>\filcenter</code> . . . . .
	1281, 1295, 1315,	<code>liDiagramm</code> . . . .	1666
	1323, 1330, 1335,	2087	<code>\footcite</code> 752, 821, 843,
	2421, 2426, 2479,	<code>liEinbettung</code> . . . .	890, 913, 945, 998,
	2486, 2493, 2873,	2000	1049, 1076, 1122,
	2972, 3017, 3022,	<code>liExkurs</code> . . . . .	1793, 1796, 1803,
	3023, 3027, 3032,	2036	1808, 1813, 1817,
	3033, 3037, 3043, 3044	<code>liGraphenFormat</code> .	1823, 1828, 1933,
<code>\do</code> . . . . .	2704, 2709,	1486	2195, 2196, 2401, 2638
	2714, 2719, 2725, 2731	<code>liJavaAngabe</code> . . . .	<code>\footnote</code> . . . . .
<code>\dots</code> . . . . .	589, 593,	2993	2100, 2104
	1814, 2632, 3128, 3129	<code>liKasten</code> . . . . .	<code>\footnotesize</code> 152, 427,
<code>\DOWNarrow</code> . . . . .	3244	1414	509, 603, 1273,
<code>\draw</code> . . . . .	1514, 1517,	<code>liKontrollflussgraph</code>	1592, 1620, 1719,
	1520, 2580, 2879, 2882	1891	1998, 2049, 2064,
		<code>liLernkartei</code> . . . .	2080, 2345, 2439,
<b>E</b>		2077	2456, 2464, 2593,
<code>\edef</code> . . . . .	1505,	<code>liProduktionsRegeln</code>	2602, 2845, 2960,
	2797, 2801, 2813, 2814	1324	2991, 3156, 3231, 3257
<code>\edge</code> . . . . .	276	<code>liProjektSprache</code> 1999	<code>\footrulewidth</code> . 1426, 1926
<code>\else</code> . . . . .	653,	<code>liQuellen</code> . . . . .	<code>\foreach</code> . 1509, 1512, 1519
	661, 669, 677, 684,	2057	<code>\forestFirst</code> . . 2933, 2936
	1290, 1304, 1340,	<code>liRelationenSchemaFormat</code>	<code>\forestLast</code> . . . 2934, 2936
	2021, 2124, 2134,	2865	<code>\forestOget</code> . . . 2933, 2934
	2144, 2158, 2432,	<code>liRmodell</code> . . . . .	<code>\forestOnes</code> . . . . .
	2469, 2514, 2575,	2845	2946
	2743, 2936, 2938, 3328	<code>liUebergangsTabelle</code>	<code>\forestOv</code> 2935, 2936, 2939
<code>\emph</code> . . 750, 997, 1252,		2386	<code>\forestov</code> . 2925, 2929,
	1795, 1824, 1826, 1974	<code>\equiv</code> . . . . .	2930, 2933, 2934,
<code>\empty</code> . . . . .	2021, 2468	3343, 3362	2935, 2936, 2938, 2939
<code>\emptyset</code> . . . . .	2378, 3123, 3144, 3177	<code>\erzeuge@tiefgestellt</code>	<code>\forestset</code> . . . . 2921, 2944
<code>\end</code> . . . . .	794, 820,	1295, 1296, 1300	<code>\forestSortLevel</code> . . . .
	845, 880, 914, 946,	<code>\expandafter</code> . . . . .	2923, 2931, 2945, 2946
		1494, 2739, 2741,	<code>\frac</code> 2153, 2186, 2218, 2233
		2742, 2743, 2751, 2909	<code>\fullouterjoin</code> . . . . .
		<code>\ExplSyntaxOff</code> . . . 54,	2836
		97, 139, 144, 197,	<b>G</b>
		202, 207, 611, 633,	<code>\g</code> . . . 39, 279, 280, 302,
		648, 724, 1276,	306, 312, 313, 314,
		1354, 1389, 1457,	315, 316, 318, 319,
		1651, 1757, 1912,	321, 322, 324, 325,
		1928, 2109, 2174,	326, 327, 328, 329,
		2335, 2414, 2608,	
		2858, 2919, 3057,	
		3164, 3217, 3296, 3367	
		<code>\ExplSyntaxOn</code> . . . . .	
		35, 66, 107,	
		140, 165, 198, 203,	
		228, 554, 617, 634,	

330, 331, 334, 338, 339, 340, 343, 345, 346, 347, 348, 349, 350, 395, 399, 403, 406, 407, 409, 410, 412, 413, 420, 421, 422, 423, 431, 432, 434, 440, 441, 443, 444, 446, 447, 459, 462, 467, 469, 473, 1545, 1547, 1558, 1560, 1567, 1580, 1581, 1674, 1678, 1682, 1686, 1687, 1688, 1689, 1690, 1692, 1693, 1695, 1726, 1728, 1733, 1737	<code>\inputmed</code> 3016, 3026, 3036, 3048, 3051, 3055 <code>\int</code> ..... 2918 <code>\item</code> 548, 549, 826, 830, 835, 840, 884, 893, 898, 906, 978, 983, 987, 1013, 1053, 1058, 1065, 1073, 1102, 1107, 1111, 1116, 1216, 1221, 1226, 1778, 1779, 2057, 2061, 2181, 2186, 2190, 2200, 2206, 2211, 2223, 2227, 2231, 2235, 2239, 2246, 2250, 2254, 2626, 2629, 2632, 2646, 2649, 2652 <code>\itshape</code> ..... 602, 3155	<code>\large</code> 1570, 1707, 1773, 2895 <code>\leaders</code> ..... 3184 <code>\left</code> ..... 2116, 2506 <code>\LEFTarrow</code> ..... 3245 <code>\leftarrow</code> ..... 665 <code>\leftouterjoin</code> ..... 2834 <code>\leftskip</code> 3347, 3348, 3358 <code>\LehramtInformatikAutorEmail</code> ..... 1924 <code>\LehramtInformatikAutorName</code> ..... 1923 <code>\LehramtInformatikGitBranch</code> ..... 458, 2979 <code>\LehramtInformatikGithubCodeRepo</code> ..... 2978 <code>\LehramtInformatikGithubDomain</code> ..... 455, 2975 <code>\LehramtInformatikGithubRawDomain</code> ..... 2976 <code>\LehramtInformatikGithubTexRepo</code> ..... 456, 2977 <code>\LehramtInformatikRepository</code> 17, 20, 23, 26, 29, 1445, 1684, 1944, 1945, 1946, 1947, 1948, 1949, 1950, 1951, 1952, 1953, 2974 <code>\LehramtInformatikTitel</code> ..... 1919 <code>\leq</code> .... 2218, 2629, 2649 <code>\let</code> ..... 1298, 1299, 1969, 2780, 2781, 2782, 2783, 2796, 2798, 2799, 2800, 2802, 2803, 2804, 2805, 2815, 2907, 2945, 2946, 3176 <code>\li@chomsky@erklaerung@texte</code> ..... 568, 604 <code>\li@EntwurfsCode</code> .... 746, 797, 798, 799, 848, 849, 850, 851, 917, 918, 919, 920, 921, 922, 949, 950, 951, 952, 953, 954, 1020, 1190, 1191, 1192, 1193, 1233, 1234 <code>\li@EntwurfsCodeAllgemein</code> ..... 745 <code>\li@fussnote@text</code> 2342, 2348, 2352, 2356, 2360 <code>\li@GithubLink</code> ..... 3008, 3021, 3031, 3042 <code>\li@mget</code> . 1496, 1500, 1520 <code>\li@minc</code> ..... 1499, 1521 <code>\li@mset</code> ..... 1493, 1501, 1510, 1513 <code>\li@numdiscs</code> ..... ... 1505, 1514, 1520 <code>\li@Rmodell@Schrift</code> . ... 2845, 2854, 2864 <code>\li@sequence</code> .. 1506, 1519
<code>\Gamma</code> . 111, 169, 212, 1309 <code>\gappto</code> ..... 2941 <code>\geometry</code> ..... 5 <code>\geq</code> ..... 1821, 2184, 2621, 2626, 2642, 2646	<b>H</b> <code>\hbox</code> ..... 2831 <code>\headrulewidth</code> . 1425, 1925 <code>\headwidth</code> ..... 1927 <code>\hfill</code> ..... 1586, 2203, 2209, 2214, 3184 <code>\hinweis</code> ..... 1998 <code>\hline</code> ..... 2391 <code>\horizontale</code> .. 1551, 1594 <code>\href</code> ..... 1362, 2104, 2271, 2305, 2314, 3012 <code>\hspace</code> .. 2583, 3335, 3343 <code>\ht</code> ..... 2832 <code>\Huge</code> ..... 1724, 3213 <code>\huge</code> ..... 1666, 3207	<b>J</b> <code>\j</code> 1509, 1510, 1512, 1513, 1514, 1519, 1520, 1521
<b>K</b> <code>\k</code> ..... 1519 <code>\keys</code> ..... 41, 75, 87, 117, 127, 175, 185, 310, 621, 625, 639, 644, 1373, 1380	<b>L</b> <code>\l</code> ..... 68, 69, 70, 71, 72, 73, 76, 77, 78, 79, 80, 82, 84, 89, 90, 91, 92, 93, 94, 109, 110, 111, 112, 113, 114, 115, 118, 119, 120, 121, 122, 123, 124, 130, 131, 132, 133, 134, 135, 136, 167, 168, 169, 170, 171, 172, 173, 176, 177, 178, 179, 180, 181, 182, 188, 189, 190, 191, 192, 193, 194, 619, 622, 627, 628, 631, 636, 637, 640, 641, 646, 1368, 1369, 1370, 1371, 1374, 1375, 1376, 1377, 1383, 1384, 1385, 1386, 1908, 1909, 1910, 2060, 2061, 2062, 2069 <code>\labelenumi</code> ..... 1410 <code>\labelenumii</code> ..... 1411 <code>\labelitemi</code> ..... 1404 <code>\labelitemii</code> ..... 1405 <code>\labelitemiii</code> ..... 1406 <code>\labelitemiv</code> ..... 1407 <code>\land</code> ..... 3363, 3365 <code>\LARGE</code> ..... 1399, 1733 <code>\Large</code> ..... 1579	<b>I</b> <code>\i</code> ..... 1519, 1520 <code>\ifcase</code> ..... 2740 <code>\ifmmode</code> 651, 659, 667, 675, 682, 1288, 1302, 1338, 2122, 2132, 2142, 2156, 2430, 2512, 2573, 3326 <code>\IfNoValueTF</code> ..... ... 2031, 2100, 2104 <code>\ifnum</code> ..... 2667, 2908, 2924, 2932, 2938 <code>\ifx</code> .... 2021, 2468, 2936 <code>\in</code> ..... 577, 733, 1821, 2166, 2169, 2172, 2201, 2207, 2212, 2621, 2632, 2642, 2652, 3106, 3108, 3114, 3135, 3284 <code>\includegraphics</code> .... ..... 1448, 1453 <code>\inhaltsverzeichnis</code> <u>1966</u> <code>\input</code> ..... 17, 20, 23, 26, 29, 479, 1683

\li@synthese@erklaerung@text	3097, 3157	\liDiagramm (environment)	2087	\liEntwurfsKompositumAkteure	1152
\liAbleitung	1323	\liEinbettung (environment)	2000	\liEntwurfsKompositumUml	1129, 1151
\liAdditum (environment)	2028	\liEntwurfs	1171	\liEntwurfsModellPraesentationSteuerung	
\liAHuelle (environment)	2437	\liEntwurfsAbstrakteFabrik	801	\liEntwurfsModellPraesentationSteuerungAk	
\liAlphabet	1308	\liEntwurfsAbstrakteFabrikBeschreibung	1173	\liEntwurfsModellPraesentationSteuerungUml	
\liAntwort (environment)	2001	\liEntwurfsAbstrakteFabrikCode	749, 802	\liEntwurfsModellPraesentationSteuerungUml	
\liAnweisung	1899	\liEntwurfsAbstrakteFabrikUml	796, 806	\liEntwurfsStellvertreter	
\liAssemblerCode	3046	\liEntwurfsAbstrakteFabrikUml	754, 804	\liEntwurfsStellvertreterCode	1195
\liAssemblerDatei	3047	\liEntwurfsAdapter	853	\liEntwurfsStellvertreterCode	1189, 1197
\liAttribut	2864	\liEntwurfsAdapterAkteure	823, 855	\liEntwurfsStellvertreterUml	1175, 1196
\liAttributHuelle	2428, 3109, 3114	\liEntwurfsAdapterCode	847, 856	\liEntwurfsZustand	1236
\liAttributHuelleOhneMathe	2428, 2431, 2433, 2447, 2457, 2465	\liEntwurfsAdapterUml	808, 854	\liEntwurfsZustandAkteure	1214, 1238
\liAttributMenge	2436, 2448, 2451, 2458, 2459, 2473, 2475	\liEntwurfsBeobachter	924	\liEntwurfsZustandCode	1232, 1239
\liAufgabe	16	\liEntwurfsBeobachterAkteure	882, 926	\liEntwurfsZustandUml	1199, 1237
\liAufgabenMetadaten	47, 1564, 1698	\liEntwurfsBeobachterCode	916, 927	\liEntwurfsZustandUml	1294
\liAufgabenTitel	55	\liEntwurfsBeobachterUml	858, 925	\liEpsilon	1252, 1266, 1268
\liAusdruck	1345	\liEntwurfsBeobachterUml	956	\liErAttribute	
\liAutomat	66	\liEntwurfsDekorierer	958	\liErDatenbankName	1271
\liAutomatenKante	98	\liEntwurfsDekoriererAkteure	948, 959	\liErEntity	1250, 1254, 1256
\liBandAlphabet	1309	\liEntwurfsDekoriererCode	929, 957	\liErErledigt	3176
\liBedingung	1900	\liEntwurfsDekoriererUml	992	\liErErledigt	1265
\liBedingungDrei	2171, 2214, 2254	\liEntwurfsDekoriererUml	976, 994	\liErErledigt	1253
\liBedingungEins	2165, 2203, 2246	\liEntwurfsEinfacheFabrik	961, 993	\liErErledigt	1259
\liBedingungFalsch	1902	\liEntwurfsEinfacheFabrikUml	1011, 1027	\liErErledigt	1251, 1260, 1262
\liBedingungWahr	1901	\liEntwurfsEinfacheFabrikUml	1022	\liExamensAufgabe	19
\liBedingungZwei	2168, 2209, 2250	\liEntwurfsEinfacheFabrikUml	1011, 1027	\liExamensAufgabeA	28
\liBeschriftung	1990	\liEntwurfsEinfacheFabrikUml	1011, 1027	\liExamensAufgabeTA	25
\liBindeAufgabeEin	1681	\liEntwurfsEinfacheFabrikUml	1011, 1027	\liExamensAufgabeTTA	22
\liChomskyErklaerung	568, 609	\liEntwurfsEinzelstueck	1022	\liExkurs (environment)	2036
\liChomskyUeberErklaerung	607	\liEntwurfsEinzelstueckAkteure	1011, 1027	\liFalsch	549
\liChomskyUeberschrift	556, 608	\liEntwurfsEinzelstueckBeschreibung	996, 1023	\liFlaci	1355
\liCpmEreignis	617	\liEntwurfsEinzelstueckCode	1019, 1029	\liFremd	2844
\liCpmFruehErklaerung	688	\liEntwurfsEinzelstueckUml	1001, 1025	\liFunktionaleAbhaengigkeit	2467, 2470, 2478
\liCpmFruehI	681, 701	\liEntwurfsErbauer	1079	\liFunktionaleAbhaengigkeiten	2481
\liCpmSpaetErklaerung	706	\liEntwurfsErbauerAkteure	1051, 1081	\liFussnote	2341, 2343
\liCpmSpaetI	674, 719	\liEntwurfsErbauerUml	1031, 1080	\liFussnoteDreiText	2355, 2373
\liCpmVon	657	\liEntwurfsFabrikmethode	1125, 1150	\liFussnoteEinsText	2347, 2367
\liCpmVonOhneMathe	657, 660, 662	\liEntwurfsFabrikmethodeAkteure	1100, 1127	\liFussnoteLink	2103
\liCpmVonZu	649	\liEntwurfsFabrikmethodeUml	1083, 1126	\liFussnoteNoten	2363
\liCpmVonZuOhneMathe	649, 652, 654	\liEntwurfsKompositum	1150	\liFussnoteUrl	1169, 2099
\liCpmVorgang	634			\liFussnoteVierText	2359, 2376
\liCpmZu	665			\liFussnoteZweiText	2351, 2370
\liCpmZuOhneMathe	665, 668, 670			\liGeschweifteKlammern	1327, 2482, 3188
				\liGrafikCCLlizen	1447, 1630

\liGrafikLogo .....	176, 177, 178, 182,	\liProblemVertexCover	1790, 1798
... 1452, 1608, 1716	1286, 1335, 1374, 1375	.....	
\liGrafikLogoPfad ...	\liMengeOhneMathe ...	\liProduktionen	1334, 1376
... 1444, 1449, 1454	... 1286, 1289, 1291	liProduktionsRegeln	
\liGrammatik .....	\liMetaBschlangaulSammlung	(environment)	1324
liGraphenFormat (envi-	..... 1613, 2291	liProjektSprache (envi-	
ronment) .... 1486	\liMetaCCLink .. 1635, 2303	ronment) .... 1999	
\liHanoi .....	\liMetaEmailLink ....	\liPseudoUeberschrift	
..... 1493	..... 2313, 2329	..... 1976,	
\liHaskellCode .... 3053	\liMetaHermineBschlangaulAndFriends	2032, 2033, 2388,	
\liHaskellDatei .... 3054	..... 1616, 2294	2398, 3272, 3280, 3288	
\liInduktionAnfang . 3271	\liMetaHilfMit .....	\liPumpingKontextfrei	
..... 3270	... 1644, 2320, 2331	..... 2640	
\liInduktionMarkierung	\liMetaQuelltext ....	\liPumpingRegulaer . 2619	
..... 3269	..... 1646, 2331	liQuellen (environment)	
\liInduktionSchritt 3287	\liMetaSetze .....	..... 2057	
\liInduktionVoraussetzung	.. 36, 48, 1565, 1699	\liRechtsReduktionInline	
..... 3279	\liMetaUeberDasProjekt	..... 2462	
liJavaAngabe (environ-	..... 1621, 2297	\liRekursionsGleichung	
ment) .... 2993	\liMinimierungErklaerung	..... 2162, 2225	
\liJavaCode .....	..... 2400	\liRelation .....	2492
\liJavaDatei .. 747, 3015	\liMinispracheDatei 3050	liRelationenSchemaFormat	
\liJavaExamen .....	\linespread .....	(environment) 2865	
..... 3035	..... 2852	\liRelationMenge ... 2859	
\liJavaTestDatei ... 3025	\liNichtsZuTun .... 3177	\liRichtig .....	548
liKasten (environment) 1414	\liO .....	liRmodell (environment)	
\liKellerAutomat .... 107	..... 2138, 2166	..... 2845	
\liKellerKante .....	\liOmega .....	\liRundeKlammer .....	
..... 145	..... 2128, 2172	..... 2115, 2119, 2129,	
\liKellerUebergang ..	\liOmegaOhneMathe ...	2139, 2153, 2505, 2509	
..... 140, 146	... 2128, 2133, 2135	\liSetzeExamenTeilaufgabeNr	
\liKontrollCode .... 1903	\liONotationO .....	..... 1677	
liKontrollflussgraph	..... 2508	\liSetzeExamenThemaNr	
(environment) 1891	\liOOhneMathe .....	..... 1673	
\liKontrollKnotenPfad	... 2138, 2143, 2145	\liSortierMarkierung 2884	
..... 1905	\liParagraphMitLinien	\liSortierPfeil .... 2878	
\liKontrollTextzeileKnoten	..... 604, 689,	\liSortierPfeilUnten 2881	
..... 1904, 1909	707, 2402, 3157,	\liSpaltenUmbruch .. 2953	
\liKurzeTabellenLinie 729	3178, 3275, 3283, 3291	\liSqlCode .....	3058
\liLadeAllePakete ...	\liPetriErreichKnotenDrei	\listen@punkt .. 2057, 2069	
..... 233, 1669	..... 2582	\liStrich .....	1765
\liLadePakete .... 59,	\liPetriErreichTransition	\liSyntheseErklaerung	
62, 229, 234, 555,	..... 2579	..... 3097, 3162	
616, 1285, 1529,	\liPetriSetzeSchluessel	\liSyntheseUeberErklaerung	
1656, 1763, 2340,	..... 2524	..... 3160	
2420, 2588, 2957, 3079	\liPetriTransitionsName	\liSyntheseUeberschrift	
\liLatexCode .....	..... 2571, 2583	..... 3081, 3161	
..... 3007	\liPetriTransitionsNameOhneMathe	\liT 2148, 2163, 2177, 2241	
\liLeereZelle .....	... 2571, 2574, 2576	\liTeilen .....	2425
..... 2378	\liPetriTransPfeile 2583	\literatur .... 1932, 1956	
liLernkartei (environ-	\liPolynomiellReduzierbar	\liTheta .....	2118,
ment) .... 2077	..... 1783	2169, 2201, 2207, 2212	
\liLinksReduktion .. 2445	\liPotenzmenge .....	\liThetaOhneMathe ...	
\liLinksReduktionInline	... 1295, 1299, 2594	... 2118, 2123, 2125	
..... 2454, 2462	\liPotenzmengeOhneMathe	\liTOhneMathe .....	
\liMasterExkurs .... 2259	... 1296, 1297, 1298	... 2148, 2157, 2159	
\liMasterFaelle 2198, 2266	\liPrimaer .....	\liTuringKante .....	208
\liMasterFallRechnung	..... 2843	\liTuringLeerzeichen	
..... 2244	\liProblemBeschreibung	..... 164, 172	
\liMasterVariablen ..	..... 1767	\liTuringMaschine ... 165	
..... 2175, 2261	\liProblemClique ... 1790	\liTuringUeberfuehrung	
\liMasterVariablenDeklaration	\liProblemName .....	..... 211	
..... 2221	..... 1766, 1773,		
\liMasterWolframLink 2269	1785, 1787, 1800,		
\liMenge ... 76, 77, 79,	1811, 1812, 1820, 1821		
118, 119, 120, 124,	\liProblemSat .....		
	..... 1819		
	\liProblemSubsetSum .		
	..... 1810, 1819		





2907, 2909, 2911, 2924, 2932, 2945, 2946	\pgfutil@empty . . . . . 2907	\pgfutil@loop . . . . . 2908	\pgfutil@repeat . . . . . 2911	\preceq . . . . . 1786	\prime . . . . . 1765	\printbibliography . 1956	\ProvidesPackage . . . . . 2, 15, 33, 58, 226, 484, 545, 552, 614, 727, 742, 1243, 1279, 1392, 1430, 1439, 1460, 1489, 1527, 1654, 1760, 1832, 1915, 1931, 1937, 1959, 2112, 2275, 2288, 2338, 2417, 2500, 2520, 2586, 2611, 2616, 2660, 2826, 2839, 2868, 2949, 2956, 2963, 2968, 3061, 3076, 3167, 3171, 3220, 3262, 3299, 3319	Q	\QS@list . . . . . 2786, 2797, 2801, 2808, 2814, 2819, 2822	\QS@select@equal . . . . . 2761, 2765	\QS@select@greater . . . . . 2762, 2766	\QS@select@smaller . . . . . 2757, 2760, 2764	\QS@sort@a . . . . . 2739, 2772, 2793, 2794	\QS@sort@b . . . . . 2739, 2740	\QS@sort@c . . . . . 2743, 2750	\QS@sort@d . . . . . 2751, 2759	\QS@sort@empty . 2741, 2746	\QS@sort@single 2742, 2747	\QS@initialize . . . . . 2664, 2776, 2811	\QS@Ir . 2747, 2753, 2761, 2781, 2795, 2800, 2803	\QS@Irr 2782, 2795, 2796, 2804	\QS@Lr . . . . . 2753, 2760, 2771, 2772, 2780, 2793, 2798, 2802	\QS@pivotStep . . . . . 2666, 2776, 2780, 2791	\QS@r . . . . . 2753	\QS@Rr . . . . . 2762, 2783, 2794, 2799, 2805, 2814, 2815, 2816	\QS@sortStep . . . . . 2668, 2776, 2792, 2793	\quad . . . . . 2344	R	\raisebox . . . . . 1904	\relax . . . . . 1969, 2753, 2796, 2798, 2799, 2800, 2908, 2910	\renewcommand . . . . . 1404, 1405, 1406, 1407, 1410, 1411, 1425, 1426, 1749, 1753, 1925, 1926, 2386	\repeat . . . . . 2669	\RequirePackage . . . . . 4, 60, 163, 227, 231, 480, 486, 487, 547, 615, 744, 745, 1245, 1247, 1248, 1284, 1393, 1394, 1396, 1398, 1403, 1412, 1419, 1427, 1431, 1432, 1442, 1443, 1461, 1462, 1463, 1491, 1492, 1542, 1665, 1670, 1671, 1703, 1764, 1833, 1933, 1938, 1939, 1955, 1962, 1963, 1964, 2114, 2278, 2279, 2502, 2503, 2504, 2521, 2523, 2613, 2673, 2675, 2827, 2828, 2829, 2842, 2870, 2901, 2952, 2958, 2965, 2970, 2981, 2982, 3063, 3168, 3175, 3223, 3224, 3226, 3227, 3265, 3266, 3267, 3300, 3320	\right . . . . . 2116, 2506	\RIGHTTarrow . . . 3241, 3246	\Rightarrow . . . . . 733, 738	\rightarrow . . . . . 212, 572, 577, 585, 589, 591, 592, 594, 649, 657, 2583, 3106, 3113, 3115, 3118, 3123, 3128, 3129, 3134	\rightrightarrow . . . . . 2835	\rmfamily . . . . . 559, 1520, 3084, 3201	\Roman . . . . . 1409	\roman . . . . . 1409, 1411	\romannumeral . . . . . 2751	\rtimes . . . . . 2277	\rule . . . . . 1554, 2784, 2806, 2821, 2832	S	\sb . . . . . 72, 82, 84, 113, 171, 584, 585, 589, 592, 593, 594, 676, 678, 683, 685, 1339, 1341, 2166, 2169, 2172, 2201, 2207, 2406, 2571, 2580, 3128,	3129, 3130, 3135, 3139, 3140, 3143, 3144, 3147, 3148, 3149	\scriptscriptstyle . . . . . 649, 657, 665	\scriptsize 1358, 1634, 1844, 1851, 1857, 1919, 1920, 1923, 1924, 3270, 3323, 3353	\section . . . 52, 1666, 1675	\sectionbreak . . . . . 1667	\seq . 1908, 1909, 1910, 2060, 2061, 2062, 2069	\setbox . . . . . 2831	\setcounter 1402, 1668, 1702, 2785, 2807, 2821	\setganttlinklabel . . . . . 1433, 1434, 1435, 1436	\setlength . . . . . 1747, 1748, 1750, 1927, 2853, 3347, 3348, 3358	\setmainfont . . . . . 1395	\setmainlanguage . . . . . 481	\setminted . . . . . 2987, 2988	\setminus . . . . . 2466	\setul . . . . . 2844	\shoveleft . . . . . 2446	\shoveright . . . . . 2450	\Sigma . . . . . 69, 110, 168, 1308, 1309, 1369	\sigma . . . . . 582, 584, 585	\SLASH . . . . . 1975	\small . . . . . 2090	\sort . . . . . 2918	\sortList . . . . . 2917, 2926	\square . . . . . 549	\stepcounter 2705, 2710, 2715, 2718, 2720, 2724, 2726, 2730, 2732	\stichwoerter . . 1544, 1592	\str 361, 560, 569, 2003, 2595, 2604, 3085, 3098	\string . . . . . 2448, 2458	\StrSubstitute . 2925, 2927	\strut . . . . . 2223, 2227, 2231, 2235, 2239, 2953	\subsection . . . . . 1679	\subseteq 3109, 3142, 3149	\subsubsection . . . . . 1700	T	\tableofcontents . . . . . 1754, 1970	\TeX . . . . . 2332	\text . . . . . 82, 84, 187, 2428, 3270, 3323	\textbf . . . . . 696, 714, 1250, 1546, 1791, 1800, 1811, 1820, 1981, 1994, 2023, 2051, 2066, 2082, 2391	\textcolor 1903, 2999, 3269
---	-------------------------------	------------------------------	--------------------------------	------------------------	-----------------------	---------------------------	---	---	---	--	--	--	--	---------------------------------	---------------------------------	---------------------------------	-----------------------------	----------------------------	--	--	--------------------------------	---	---	----------------------	---	--	----------------------	---	--------------------------	---	--	------------------------	---	-----------------------------	-------------------------------	--------------------------------	---	---------------------------------	--	-----------------------	-----------------------------	------------------------------	------------------------	---	---	--	--	---	---	-------------------------------	------------------------------	--	------------------------	---	--	---	-----------------------------	--------------------------------	---------------------------------	--------------------------	-----------------------	---------------------------	----------------------------	--	--------------------------------	-----------------------	-----------------------	----------------------	--------------------------------	-----------------------	---	------------------------------	---	------------------------------	-----------------------------	--	----------------------------	----------------------------	-------------------------------	---	--	---------------------	--	--	-----------------------------

<code>\textit</code> . . .	1090, 1132, 1133, 1134, 1135, 1559, 2092, 2436, 2496
<code>\textsc</code> . . . . .	1766
<code>\textsf</code> . . . . .	2066
<code>\textstyle</code> . . . .	2186, 2218
<code>\texttt</code> 1273, 1766, 1900, 1901, 1902, 1903, 3323	
<code>\textwidth</code> . . . .	1554, 1927
<code>\thematik</code> . . . .	1557, 1586
<code>\thepage</code> . . . . .	1423, 1922
<code>\theparagraph</code> . . . .	1401
<code>\thesection</code> . . . . .	1666
<code>\Theta</code> . . . . .	2119
<code>\thinspace</code> . . . . .	3323
<code>\tikz</code> . . . . .	1904
<code>tikz: bbaum</code> . . . . .	26
<code>tikz: li binaer baum</code> . . . .	24
<code>\tikzchildnode</code> . . . . .	504
<code>\tikzparentnode</code> . . . . .	504
<code>\tikzset</code> . . . . .	101, 148, 214, 489, 515, 1465, 1835, 2561, 2687, 2891, 3065, 3301
<code>\tikzumlset</code> . . . . .	3229
<code>\times</code> . . . . .	212
<code>\tiny</code> . 1256, 1262, 1268, 1642, 1903, 1974, 3010	
<code>\titleformat</code> . . . . .	1399, 1401, 1666
<code>\titlespacing</code> . . . . .	1400
<code>\tl</code> . 39, 68, 69, 70, 71, 72, 73, 76, 77, 78, 79, 80, 82, 84, 109, 110, 111, 112, 113, 114, 115, 118, 119, 120, 121, 122, 123, 124, 167, 168, 169, 170, 171, 172, 173, 176, 177, 178, 179, 180, 181, 182, 303, 307, 334, 338, 339, 340, 343, 348, 349, 350, 362, 371, 406, 409, 412, 420, 421, 422, 423, 434, 440, 443, 446, 462, 469, 619, 622, 627, 628, 636, 637, 640, 641, 1347, 1368, 1369, 1370, 1371, 1374, 1375, 1376, 1377, 1545, 1558, 1567, 1580, 1674, 1678, 1682, 1689, 1692, 2149	
<code>\tmp</code> . . . . .	2468
<code>\TmpPlaceEight</code> . . . . .	2543
<code>\TmpPlaceFive</code> . . . . .	2540
<code>\TmpPlaceFour</code> . . . . .	2539
<code>\TmpPlaceNine</code> . . . . .	2544
<code>\TmpPlaceOne</code> . . . . .	2536
<code>\TmpPlaceSeven</code> . . . . .	2542
<code>\TmpPlaceSix</code> . . . . .	2541
<code>\TmpPlaceTen</code> . . . . .	2545
<code>\TmpPlaceThree</code> . . . . .	2538
<code>\TmpPlaceTwo</code> . . . . .	2537
<code>\TmpScale</code> . . . . .	2556
<code>\TmpTransitionEight</code> . . . . . .	2532, 2553
<code>\TmpTransitionFive</code> . . . . . . .	2529, 2550
<code>\TmpTransitionFour</code> . . . . . . .	2528, 2549
<code>\TmpTransitionNine</code> . . . . . . .	2533, 2554
<code>\TmpTransitionOne</code> . . . . . . . .	2525, 2546
<code>\TmpTransitionSeven</code> . . . . . . .	2531, 2552
<code>\TmpTransitionSix</code> . . . . . . . .	2530, 2551
<code>\TmpTransitionTen</code> . . . . . . . .	2534, 2555
<code>\TmpTransitionThree</code> . . . . . . .	2527, 2548
<code>\TmpTransitionTwo</code> . . . . . . . .	2526, 2547
<code>\TmpX</code> . . . . .	2557
<code>\TmpY</code> . . . . .	2558
<code>\today</code> . . . . .	1920
<code>\ttfamily</code> . . . . .	2845
<b>U</b>	
<code>\ul</code> . . . . .	1251, 2843, 2844
<code>\umlagg</code> . . . . .	1209
<code>\umlassoc</code> . . . . .	1167
<code>\umlclass</code> . . . . .	756, 760, 764, 811, 812, 813, 860, 865, 870, 873, 931, 932, 933, 938, 939, 968, 1003, 1034, 1035, 1038, 1089, 1092, 1131, 1137, 1138, 1156, 1157, 1158, 1179, 1180, 1181, 1201, 1202, 1203, 1204
<code>\umldep</code> . . . . .	1097
<code>\umlHVHagg</code> . . . . .	878, 944, 1147
<code>\umlinherit</code> . . . . .	817, 868, 1042, 1087, 1095
<code>\umlnote</code> . . . . .	819, 1044, 1211
<code>\umlreal</code> . . . . .	815, 876
<code>\umlsimpleclass</code> . . . . .	771, 772, 773, 777, 779, 780, 781,
<code>\umlstatic</code> . . . . .	810, 963, 964, 965, 1033, 1085, 1086, 1177
<code>\umluniagg</code> . . . . .	970, 1004
<code>\umluniagg</code> . . . . .	1040
<code>\umluniassoc</code> . . . . .	791, 816, 1041, 1165, 1166, 1185, 1186
<code>\umlVHuniassoc</code> . . . . .	792, 793
<code>\umlVHVdep</code> . . . . .	785, 786, 788, 789, 972, 973
<code>\umlVHVinherit</code> 768, 769, 774, 775, 782, 783, 941, 942, 966, 967, 1145, 1146, 1183, 1184	
<code>\umlVHVreal</code> . . . . .	935, 936, 1206, 1207
<code>\UParrow</code> . . . . .	3243
<code>\url</code> . . . . .	463, 2100
<code>\usemintedstyle</code> . . . .	2984
<code>\usetikzlibrary</code> . . . .	61, 488, 1246, 1464, 1834, 2522, 2830, 2871, 2902, 3228, 3316
<b>V</b>	
<code>\value</code> . . . . .	2667
<code>\varepsilon</code> . . . . .	561, 572, 573, 1294, 2166, 2172, 2204, 2215
<code>\vfill</code> . . . . .	1604, 1714, 1721, 1752, 2953
<code>\vrule</code> . . . . .	3180, 3184
<code>\vspace</code> . . . . .	1710, 1731, 1735, 1739, 1749, 2037, 2055, 3193, 3195
<b>X</b>	
<code>\xappto</code> . . . . .	2929, 2935, 2939
<code>\xdef</code> . . . . .	1494
<code>\xintApply</code> . . . . .	2755
<code>\xintApplyUnbraced</code> . . . . . . . .	2754, 2760, 2761, 2762
<code>\xintCSVtoList</code> . . . . .	2814
<code>\xintFor</code> . . . . .	2704, 2709, 2714, 2719, 2725, 2731, 2772
<code>\xintIfEq</code> . . . . .	2765
<code>\xintIfForLast</code> . . . . .	2721, 2727, 2733
<code>\xintIfGt</code> . . . . .	2766
<code>\xintIfLt</code> . . . . .	2764
<code>\xintLength</code> . . . . .	2739
<code>\xintntheft</code> . . . . .	2751
<b>Z</b>	
<code>\ZB</code> . . . . .	2107
<code>\zB</code> . . . . .	2106
<code>\zustandsnamen@liste</code> . . . . .	1312, 1319, 1320