

# lehramt-informatik

Hermine Bschlangaul \*

September 11, 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
2.5.1	IFs	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

---

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

2.11.8 Einzelstück (Singleton)	43
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	66
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
2.26.1 Umgebungen, die Inhalte aus- und einblenden können	78
2.27 master-theorem.sty	82
2.27.1 Makro-Kürzel	82
2.28 mathe.sty	86
2.29 meta.sty	87
2.29.1 Einfache Makros (Low level)	87
2.29.2 Zusammengesetzte Makros (High level)	88
2.30 minimierung.sty	90
2.31 normalformen.sty	93
2.31.1 Makro-Kürzel	93
2.32 o-notation.sty	96
2.32.1 Makro-Kürzel	96
2.32.2 TeX-Markup-Beispiel: Funktionsgraphen mit pgfplots	96
2.33 petri.sty	97
2.33.1 Makro-Kürzel	97
2.34 potenzmengen-konstruktion.sty	99
2.35 pseudo.sty	101
2.36 pumping-lemma.sty	102
2.37 quicksort.sty	103

2.38	relationale-algebra.sty	106
2.39	rmodell.sty	107
2.39.1	Makro-Kürzel	107
2.40	sortieren.sty	108
2.41	spalten.sty	110
2.42	sql.sty	111
2.42.1	Latex-Markup-Beispiel	111
2.43	struktogramm.sty	112
2.44	syntax.sty	113
2.44.1	Makro-Kürzel	113
2.45	syntaxbaum.sty	116
2.46	synthese-algorithmus.sty	117
2.46.1	Makro-Kürzel	117
2.46.2	TeX-Markup Grundgerüst	117
2.46.3	TeX-Markup Linksreduktion	117
2.46.4	TeX-Markup Rechtsreduktion	117
2.46.5	TeX-Markup Relationen formen	118
2.47	tabelle.sty	121
2.48	typographie.sty	122
2.49	uml.sty	124
2.50	vollstaendige-induktion.sty	126
2.50.1	Makro-Kürzel	126
2.51	wasserfall.sty	128
2.52	wpkalkuel.sty	129
2.52.1	Makro-Kürzel	129

### 3 Index 130

# 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}

\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   $#1 \l_typ_tl = (
90     \l_zustaende_tl,
91     \l_alphabet_tl,
92     \l_delta_tl,
93     \l_ende_tl,
94     \l_start_tl
95   )$
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 }

```

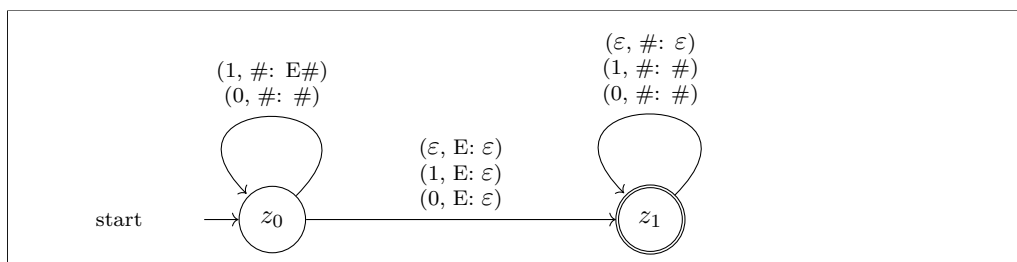
## 2.4.2 Kellerautomat

```
\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]
{\zustaende=Z,alphabet=\Sigma,kelleralphabet=\Gamma,delta=\delta,start=z_0,kellerboden=\#,ende=E}

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

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

```
107 \ExplSyntaxOn
108 \NewDocumentCommand {\liKellerAutomat} { O{K} m } {
109   \tl_set:Nn \l_zustaende_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     zustaende .code:n = {\tl_set:Nn \l_zustaende_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
```

### 2.5.1 IFs

Wir weichen von dem üblichen Namensschema ab und beginnen das `if` mit einem Großbuchstaben und schreiben das letzte Worte komplett in Großbuchstaben, damit die `if`-Befehle schöner lesbar sind, z. B. `\ifLiADDITUM`, `\LiADDITUMtrue` und `\LiADDITUMfalse`.

```
\ifLiADDITUM
\LiADDITUMtrue 229 \newif\ifLiADDITUM
\LiADDITUMfalse 230 \LiADDITUMfalse

\ifLiEXKURS
\LiEXKURStrue 231 \newif\ifLiEXKURS
\LiEXKURfalse 232 \LiEXKURStrue

\ifLiANTWORT
\LiANTWORTtrue 233 \newif\ifLiANTWORT
\LiANTWORTfalse 234 \LiANTWORTtrue

\liLadePakete

235 \NewDocumentCommand{\liLadePakete}{ m }
236 {
237   \clist_map_inline:nn { #1 } { \RequirePackage{lehramt-informatik-##1} }
238 }

\liLadeAllePakete

239 \def\liLadeAllePakete{
240   \liLadePakete{
241     aufgaben-einbinden,
242     automaten,
243     checkbox,
244     chomsky-normalform,
245     cpm,
246     cyk-algorithmus,
247     entwurfsmuster,
248     er,
249     formale-sprachen,
250     gantt,
251     grafik,
252     graph,
253     hanoi,
254     kontrollflussgraph,
255     komplexitaetstheorie,
256     makros,
257     master-theorem,
258     mathe,
259     minimierung,
260     normalformen,
261     petri,
262     potenzmengen-konstruktion,
263     pumping-lemma,
264     pseudo,
265     quicksort,
266     relationale-algebra,
267     rmodell,
268     sortieren,
269     spalten,
```

```

270     struktogramm,
271     sql,
272     syntax,
273     syntaxbaum,
274     synthese-algorithmus,
275     tabelle,
276     typographie,
277     uml,
278     vollstaendige-induktion,
279     wasserfall,
280     wpkalkuel,
281     %
282     baum, % am Schluss sonst Fehler: undefined command \edge
283 }
284 }

```

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.

```

285 \clist_new:N \g_auf_schluessel_clist
286 \clist_set:Nn \g_auf_schluessel_clist {
287     titel,
288     thematik,
289     stichwoerter,
290     zitat_schluessel,
291     zitat_beschreibung,
292     %
293     bearbeitungs_stand,
294     korrektheit,
295     %
296     relativer_pfad,
297     identische_aufgabe,
298     %
299     examen_nummer,
300     examen_fach,
301     examen_jahr,
302     examen_monat,
303     examen_jahreszeit,
304     examen_thema_nr,
305     examen_teilaufgabe_nr,
306     examen_aufgabe_nr,
307 }

```

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

```

308 \clist_map_inline:Nn \g_auf_schluessel_clist {
309     \tl_new:c {g_auf_#1_tl}
310 }

```

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

```

311 \cs_new:Npn \setze_variablen_zurueck: {
312     \clist_map_inline:Nn \g_auf_schluessel_clist {
313         \tl_clear:c {g_auf_##1_tl}
314     }
315 }

```

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

```

316 \keys_define:nn { aufgabenmetadaten }
317 {
318     Titel .tl_gset:N = \g_auf_titel_tl,
319     Thematik .tl_gset:N = \g_auf_thematik_tl,
320     Stichwoerter .tl_gset:N = \g_auf_stichwoerter_tl,

```

```

321 ZitatSchluessel .tl_gset:N = \g_auf_zitat_beschreibung_tl,
322 ZitatBeschreibung .tl_gset:N = \g_auf_zitat_schluessel_tl,
323 %
324 BearbeitungsStand .tl_gset:N = \g_auf_bearbeitungs_stand_tl,
325 Korrektheit .tl_gset:N = \g_auf_korrekttheit_tl,
326 %
327 RelativerPfad .tl_gset:N = \g_auf_relativer_pfad_tl,
328 IdentischeAufgabe .tl_gset:N = \g_auf_identische_aufgabe_tl,
329 %
330 ExamenNummer .tl_gset:N = \g_auf_examen_nummer_tl,
331 ExamenFach .tl_gset:N = \g_auf_examen_fach_tl,
332 ExamenJahr .tl_gset:N = \g_auf_examen_jahr_tl,
333 ExamenMonat .tl_gset:N = \g_auf_examen_monat_tl,
334 ExamenJahreszeit .tl_gset:N = \g_auf_examen_jahreszeit_tl,
335 ExamenThemaNr .tl_gset:N = \g_auf_examen_thema_nr_tl,
336 ExamenTeilaufgabeNr .tl_gset:N = \g_auf_examen_teilaufgabe_nr_tl,
337 ExamenAufgabeNr .tl_gset:N = \g_auf_examen_aufgabe_nr_tl,
338 }

339 \cs_gset:Npn \setze_relativen_pfad: {
340   \tl_if_empty:NTF \g_auf_relativer_pfad_tl
341   {
342     \bool_if:nTF
343     {
344       ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
345       ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
346       ! \tl_if_empty_p:N \g_auf_examen_monat_tl
347     }
348     {
349       \tl_gset:Nn \g_auf_relativer_pfad_tl {
350         Staatsexamen /
351         \g_auf_examen_nummer_tl /
352         \g_auf_examen_jahr_tl /
353         \g_auf_examen_monat_tl /
354         \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} { Thema - \g_auf_examen_thema_nr_tl / }
355         \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} { Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl / }
356         \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} { Aufgabe - \g_auf_examen_aufgabe_nr_tl / }
357       }
358     }
359   }{}
360 }
361 {}
362 }

363 \cs_set:Nn \_trenner: {
364   \, / \,
365 }

366 \cs_gset:Npn \_gib_jahreszeit_durch_monat: #1 {
367   % \str_case funktioniert nicht mit den Tokenlist variablen.
368   \tl_case:Nn { #1 }
369   {
370     { 3 } { Frühjahr }
371     { 03 } { Frühjahr }
372     { 9 } { Herbst }
373     { 09 } { Herbst }
374   }
375 }

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

376 \cs_gset:Npn \_gib_examen_fach_durch_nummer: #1 {
377   \tl_case:Nn { #1 }
378   {
379     { 46110 } { Grundlagen der Informatik (nicht vertieft) }
380     { 46111 } { Programmentwicklung / Systemprogrammierung / Datenbanksysteme (nicht vertieft) }
381     { 46112 } { Grundlagen der Informatik (nicht vertieft) }

```

```

382 { 46113 } { Theoretische Informatik (nicht vertieft) }
383 { 46114 } { Algorithmen / Datenstrukturen / Programmiermethoden (nicht vertieft) }
384 { 46115 } { Theoretische Informatik / Algorithmen / Datenstrukturen (nicht vertieft) }
385 { 46116 } { Softwaretechnologie / Datenbanksysteme (nicht vertieft) }
386 { 46118 } { Fachdidaktik (Mittelschulen) }
387 { 46119 } { Fachdidaktik (Realschulen) }
388 { 46121 } { Fachdidaktik (berufliche Schulen) }
389 { 66110 } { Automatentheorie, Algorithmische Sprache (vertieft) }
390 { 66111 } { Betriebssysteme / Datenbanksysteme / Rechnerarchitektur (vertieft) }
391 { 66112 } { Automatentheorie / Komplexität / Algorithmen (vertieft) }
392 { 66113 } { Rechnerarchitektur / Datenbanken / Betriebssysteme (vertieft) }
393 { 66114 } { Datenbank- und Betriebssysteme (vertieft) }
394 { 66115 } { Theoretische Informatik / Algorithmen (vertieft) }
395 { 66116 } { Datenbanksysteme / Softwaretechnologie (vertieft) }
396 { 66118 } { Fachdidaktik (Gymnasium) }
397 }
398 }

```

Einzelprüfungsnummer / Jahr / Jahreszeit mit Trennzeichen

```

399 \cs_gset:Npn \_gib_einzelpruefung_trenner: {
400   Staatsexamen ~
401   \g_auf_examen_nummer_tl
402
403   \_trenner:
404
405   \g_auf_examen_jahr_tl
406
407   \_trenner:
408
409   \gib_jahreszeit_durch_monat: \g_auf_examen_monat_tl
410 }

```

Thema Nr.1 / Teilaufgabe Nr. 2 / Aufgabe 3

```

411 \cs_gset:Npn \_gib_aufgaben_pfad_trenner: {
412   \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
413     Thema ~ \g_auf_examen_thema_nr_tl \_trenner:
414   }
415   \tl_if_empty:NTF \g_auf_examen_telaufgabe_nr_tl {} {
416     Teilaufgabe ~ \g_auf_examen_telaufgabe_nr_tl \_trenner:
417   }
418   \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} {
419     Aufgabe ~ \g_auf_examen_aufgabe_nr_tl
420   }
421 }

422 \cs_gset:Npn \_gib_examen_titel: {
423   \cs_set:Nn \_trenner: { \, / \, }
424   \bool_if:nTF
425   {
426     ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
427     ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
428     ! \tl_if_empty_p:N \g_auf_examen_monat_tl &&
429     ! \tl_if_empty_p:N \g_auf_examen_aufgabe_nr_tl
430   }
431   {
432     {
433       \footnotesize
434       \par
435       \noindent
436       Staatsexamen ~
437       \g_auf_examen_nummer_tl \_trenner:
438       \g_auf_examen_jahr_tl \_trenner:
439
440       \tl_case:Nn \g_auf_examen_monat_tl
441       {

```

```

442     { 03 } { Frühjahr }
443     { 09 } { Herbst }
444 } \_trenner:
445
446 \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
447     Thema ~ Nr. ~ \g_auf_examen_thema_nr_tl \_trenner:
448 }
449 \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
450     Teilaufgabe ~ Nr. ~ \g_auf_examen_teilaufgabe_nr_tl \_trenner:
451 }
452 \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} {
453     Aufgabe ~ Nr. ~ \g_auf_examen_aufgabe_nr_tl
454 }
455 \par
456 \bigskip
457 }
458 }
459 }
460 \cs_new:Npn \_gib_github_url: {
461     \LehramtInformatikGithubDomain /
462     \LehramtInformatikGithubTexRepo /
463     blob /
464     \LehramtInformatikGitBranch /
465     \g_auf_relativer_pfad_tl
466 }
467 \cs_new:Npn \_gib_github_url_href: {
468     \tl_if_empty:NTF \g_auf_relativer_pfad_tl {} {
469         \url{ \_gib_github_url: }
470     }
471 }
472 \cs_new:Npn \_gib_aufgaben_titel: {
473     \g_auf_titel_tl
474
475     \tl_if_empty:NTF \g_auf_thematik_tl
476     {}
477     {
478         \, ~ [
479             \g_auf_thematik_tl
480         ]
481     }
482 }
483 \msg_new:nnn { aufgabenmetadaten } { kein-titel }
484 { Der~Schlüssel~ist~zwingend~notwendig }
485
486 \def\LehramtInformatikRepository{/pfad/zum/repository}
487 \input /etc/lehramt-informatik.config.tex
488 biblatex not working with lualatex and babel
489 % \RequirePackage{polyglossia}
490 % \setmainlanguage{german}

```

## 2.6 baum.sty

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

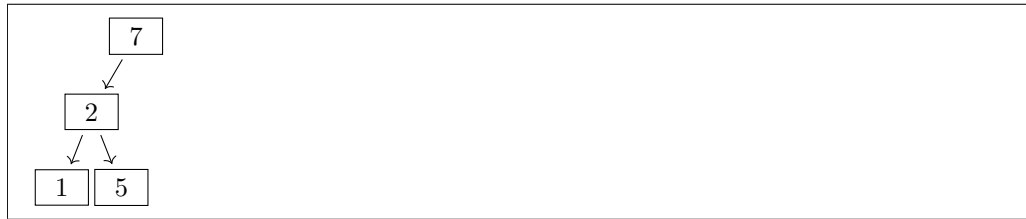
    für li binaer baum
493 \RequirePackage{tikz-qtrees}

    Für b baum
494 \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}
```



```
495 \tikzset{
496   li binaer baum/.style={
497     shorten <=2pt,
498     shorten >=2pt,
499     ->,
500     every tree node/.style={
501       minimum width=2em,
502       draw,
503       rectangle
504     },
505     blank/.style={
506       draw=none
507     },
508     edge from parent/.style={
509       draw,
510       edge from parent path={(\tikzparentnode) -- (\tikzchildnode)}
511     },
512     level distance=1cm,
513     every label/.style={
514       gray,
515       font=\footnotesize,
516       label position=0,
517       label distance=0cm,
518     }
519   },
520 }
```



### 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}
```

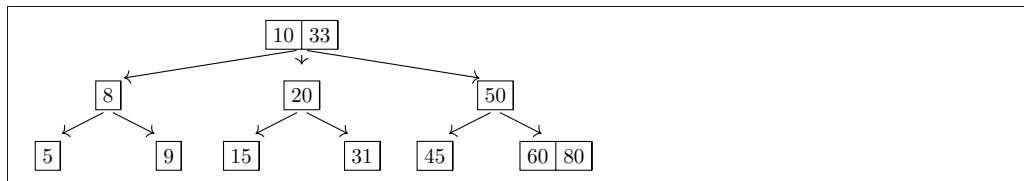


### 2.6.3 B-Baum

```

\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}

```



```

521 \tikzset{
522   li bbaum knoten/.style={
523     rectangle split parts=10,
524     rectangle split,
525     rectangle split horizontal,
526     rectangle split ignore empty parts,
527     draw,
528     fill=white
529   },
530   li bbaum/.style={
531     every node/.style={
532       li bbaum knoten
533     },
534     level 1/.style={
535       level distance=12mm,
536       sibling distance=25mm,
537     },
538     every child/.style={
539       shorten <= 2pt,
540       shorten >= 6pt,
541       ->,
542     },
543     level 2/.style={
544       level distance=9mm,
545       sibling distance=15mm,
546     },
547   }
548 }
549

```

## 2.7 checkbox.sty

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

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

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

556
```

## 2.8 chomsky-normalform.sty

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

560 \ExplSyntaxOn

561 \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

```

562 \def\liChomskyUeberschrift#1{
563   {
564     \bfseries
565     \rmfamily
566     \str_case:nn {#1} {
567       {1} {Elimination-der-\varepsilon-Regeln}
568       {2} {Elimination-von-Kettenregeln}
569       {3} {Separation-von-Terminalzeichen}
570       {4} {Elimination-von-mehrelementigen-Nonterminalketten}
571     }
572   }
573 }

```

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

```

574 \def\liChomsky@erklaerung@texte#1{
575   \str_case:nn {#1} {
576     %
577     {1} {
578       Alle-Regeln-der-Form- $A \rightarrow \varepsilon$ -werden-eliminiert.~
579       Die-Ersetzung-von- $A$ -wird-durch- $\varepsilon$ -in-alen-anderen-
580       Regeln-vorweggenommen.
581     }
582     {2} {
583       Jede-Produktion-der-Form- $A \rightarrow B$ -mit- $A, B$ -in- $S$ -wird-
584       als-Kettenregel-bezeichnet.~Diese-tragen-nicht-zur-Produktion-
585       von-Terminalzeichen-bei-und-lassen-sich-ebenfalls-eliminieren.
586     }
587     {3} {
588       Jedes-Terminalzeichen- $\sigma$ ,~das-in-Kombination-mit-anderen-
589       Symbolen-auftaucht,~wird-durch-ein-neues-Nonterminal-
590        $S_{\sigma}$ -ersetzt-und-die-Menge-der-Produktionen-durch-die-
591       Regel- $S_{\sigma} \rightarrow \sigma$ -ergänzt.
592     }
593     {4} {
594       Alle-Produktionen-der-Form-
595        $A \rightarrow B_{\{1\}} B_{\{2\}} \dots B_{\{n\}}$ -
596       werden-in-die-Produktionen-
597        $A \rightarrow$ 
598        $A_{\{n-1\}} B_{\{n\}}, A_{\{n-1\}} \rightarrow$ 
599        $A_{\{n-2\}} B_{\{n-1\}}, \dots,$ 
600        $A_{\{2\}} \rightarrow B_{\{1\}} B_{\{2\}}$ -zerteilt.~
601       Nach-der-Ersetzung-sind-alles-längeren-Nonterminalketten-
602       vollständig-heruntergebrochen-und-die-Chomsky-Normalform-erreicht.
603     }
604   }

```

```

605 }
606 \def\liChomskyErklaerung#1{
607   {
608     \itshape
609     \footnotesize
610     \liParagraphMitLinien{\li@chomsky@erklaerung@texte{#1}}
611   }
612 }

```

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

```

613 \def\liChomskyUeberErklaerung#1{
614   \liChomskyUeberschrift{#1}\par
615   \liChomskyErklaerung{#1}
616 }

```

```

617 \ExplSyntaxOff
618

```

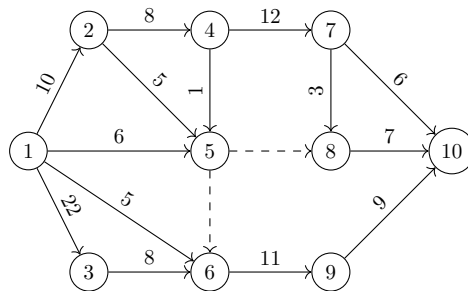
## 2.9 cpm.sty

```
619 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
620 \ProvidesPackage{lehramt-informatik-cpm}[2020/09/03]
621 \RequirePackage{tikz}
622 \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}{5}
\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 \\
\hline
SZ & 0 & 2 & 5 & 10 & 10 & 6 & 12 \\
\hline
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 \\
\hline
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  \\ \\hline
\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 \\ \\hline
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}

```

```

623 \ExplSyntaxOn
624 \NewDocumentCommand { \liCpmEreignis } { 0{} m m m } {
625   \tl_set:Nn \l_name_tl {}
626
627   \keys_define:nn { cpmEreignis } {
628     name .code:n = {\tl_set:Nn \l_name_tl {##1}},
629   }
630
631   \keys_set:nn { cpmEreignis } { #1 }
632
633   \tl_if_empty:NT \l_name_tl {
634     \tl_set:Nn \l_name_tl {#2}
635   }
636
637   \node[circle,draw] (\l_name_tl) at (#3,#4) {#2};
638 }
639 \ExplSyntaxOff

```

```

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

```

```

640 \ExplSyntaxOn
641 \NewDocumentCommand { \liCpmVorgang } { 0{} m m m } {
642   \tl_set:Nn \l_schein_tl {}
643   \tl_set:Nn \l_kritisch_tl {}
644
645   \keys_define:nn { cpmVorgang } {
646     schein .code:n = {\tl_set:Nn \l_schein_tl {dashed}},
647     kritisch .code:n = {\tl_set:Nn \l_kritisch_tl {thick}},
648   }
649
650   \keys_set:nn { cpmVorgang } { #1 }
651
652   \path[->,\l_schein_tl,\l_kritisch_tl] (#2) edge node[auto,sloped] {#4} (#3);
653 }
654 \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):  $1_{(2 \rightarrow 3)}$ 
655 \def\liCpmVonZuOhneMathe#1(#2-#3){#1_{\scriptscriptstyle(#2\rightarrow#3)}}
656 \def\liCpmVonZu#1(#2-#3){%
657 \ifmmode%
658 \liCpmVonZuOhneMathe{#1}(\#2-\#3)%
659 \else%
660 $\liCpmVonZuOhneMathe{#1}(\#2-\#3)$%
661 \fi%
662 }

\liCpmVon Let-Abkürzung: \let\v=\liCpmVon
\liCpmVon{1}(2):  $1_{(\rightarrow 2)}$ 
663 \def\liCpmVonOhneMathe#1(#2){#1_{\scriptscriptstyle(\rightarrow#2)}}
664 \def\liCpmVon#1(#2){%
665 \ifmmode%
666 \liCpmVonOhneMathe{#1}(\#2)%
667 \else%
668 $\liCpmVonOhneMathe{#1}(\#2)$%
669 \fi%
670 }

\liCpmZu Let-Abkürzung: \let\z=\liCpmZu
\liCpmZu{1}(2):  $1_{(\leftarrow 2)}$ 
671 \def\liCpmZuOhneMathe#1(#2){#1_{\scriptscriptstyle(\leftarrow#2)}}
672 \def\liCpmZu#1(#2){%
673 \ifmmode%
674 \liCpmZuOhneMathe{#1}(\#2)%
675 \else%
676 $\liCpmZuOhneMathe{#1}(\#2)$%
677 \fi%
678 }

679 \ExplSyntaxOn

\liCpmSpaetI Spätester Zeitpunkt, zu dem Ereignis  $i$  eintreten kann
Let-Abkürzung: \let\SZ=\liCpmSpaetI
680 \NewDocumentCommand{ \liCpmSpaetI } { 0{i} } {
681 \ifmmode
682 SZ\sb{#1}
683 \else
684 $\SZ\sb{#1}$
685 \fi
686 }

\liCpmFruehI Frühester Zeitpunkt, zu dem Ereignis  $i$  eintreten kann.
Let-Abkürzung: \let\FZ=\liCpmFruehI
687 \NewDocumentCommand{ \liCpmFruehI } { 0{i} } {
688 \ifmmode
689 FZ\sb{#1}
690 \else
691 $\FZ\sb{#1}$

```

```

692 \fi
693 }

```

\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. —

```

694 \def\liCpmFruehErklaerung{
695   \liParagraphMitLinien{
696     Wir~führen~eine~Vorwärtsterminierung~durch~
697     und~addieren~die~Dauern.~
698
699     Kann~ein~Ereignis~über~mehrere~Vorgänge~erreicht~
700     werden,~wählen~wir~das~Maximum~aus.~
701
702     \textbf{Erläuterungen:}~
703
704      $i$ :~
705     Ereignis~ $i$ ;~,
706
707     \liCpmFruehI{}:~
708     Frühester~Zeitpunkt,~zu~dem~Ereignis~ $i$ ~eintreten~kann
709     .
710   }
711 }

```

\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. —

```

712 \def\liCpmSpaetErklaerung{
713   \liParagraphMitLinien{
714     Wir~führen~eine~Rückwärtsterminierung~durch~
715     und~subtrahieren~die~Dauern~vom~letzten~Ereignis~aus.~
716
717     Kann~ein~Ereignis~über~mehrere~Vorgänge~erreicht~
718     werden,~wählen~wir~das~Minimum~aus.~
719
720     \textbf{Erläuterungen:}~
721
722      $i$ :~
723     Ereignis~ $i$ ;~,
724
725     \liCpmSpaetI{}:~
726     Spätester~Zeitpunkt,~zu~dem~Ereignis~ $i$ ~eintreten~kann
727     .
728   }
729 }

730 \ExplSyntaxOff
731

```

## 2.10 cyk-algorithmus.sty

```
732 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
733 \ProvidesPackage{lehramt-informatik-cyk-algorithmus}[2021/06/18 Hilfsmakros
734 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`

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

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

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

```
736 \NewDocumentCommand{ \liWortInSprache } { m O{L(G)} } {
737   \bigskip
738   \noindent
739   $\Rightarrow$ #1 \in #2$
740 }
```

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

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

```
741 \NewDocumentCommand{ \liWortNichtInSprache } { m O{L(G)} } {
742   \bigskip
743   \noindent
744   $\Rightarrow$ #1 \notin #2$
745 }
```

```
746
```

## 2.11 entwurfsmuster.sty

```
747 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
748 \ProvidesPackage{lehramt-informatik-entwurfsmuster}[2021/05/06
749 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

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

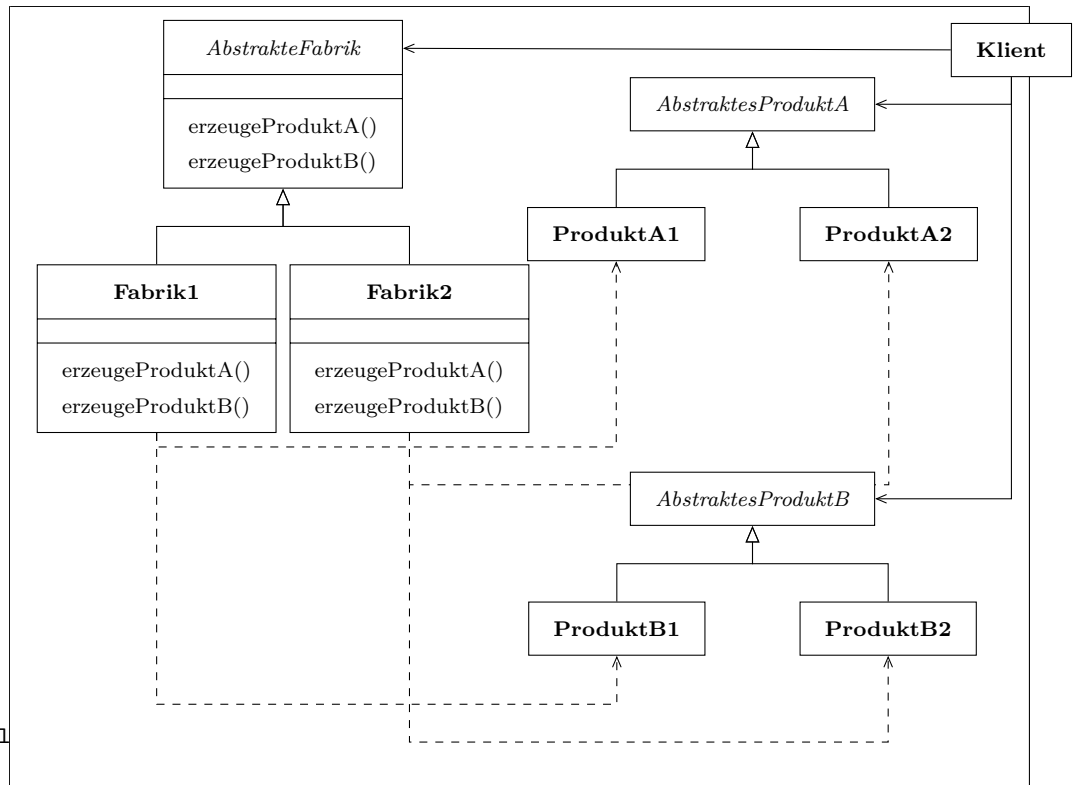
\li@EntwurfsCodeAllgemein Allgemeine Code-Beispiele zu den UML-Diagrammen und Stellvertretern

```
751 \RequirePackage{lehramt-informatik-syntax}
752 \def\li@EntwurfsCode#1#2{
753   \liJavaDatei{entwurfsmuster/#1/allgemein/#2}
754 }
```

### 2.11.3 Abstrakte Fabrik (Abstract Factory)

sAbstrakteFabrikBeschreibung

```
755 \def\liEntwurfsAbstrakteFabrikBeschreibung{
756   Es wird eine Schnittstelle bereitgestellt, um \emph{Familien
757   verbundener oder abhängiger Objekte} zu erstellen, ohne die konkreten
758   Klassen zu spezifizieren.\footcite[Seite 25]{eilebrecht}
759 }
```



```

760 \def\liEntwurfsAbstrakteFabrikUml{
761   \begin{tikzpicture}
762     \umlclass[type=abstract]{AbstrakteFabrik}{-}{-}{
763       erzeugeProduktA()\
764       erzeugeProduktB()\
765     }
766     \umlclass[below left=1cm and -1.5cm of AbstrakteFabrik]{Fabrik1}{-}{-}{
767       erzeugeProduktA()\
768       erzeugeProduktB()\
769     }
770     \umlclass[below right=1cm and -1.5cm of AbstrakteFabrik]{Fabrik2}{-}{-}{
771       erzeugeProduktA()\
772       erzeugeProduktB()\
773     }
774     \umlVHVinherit{Fabrik1}{AbstrakteFabrik}
775     \umlVHVinherit{Fabrik2}{AbstrakteFabrik}
776
777     \umlsimpleclass[right=3cm of AbstrakteFabrik,type=abstract]{AbstraktesProduktA}
778     \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktA]{ProduktA1}
779     \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktA]{ProduktA2}
780     \umlVHVinherit{ProduktA1}{AbstraktesProduktA}
781     \umlVHVinherit{ProduktA2}{AbstraktesProduktA}
782
783     \umlsimpleclass[above right=0cm and 1cm of AbstraktesProduktA]{Klient}
784
785     \umlsimpleclass[below=4.5cm of AbstraktesProduktA,type=abstract]{AbstraktesProduktB}
786     \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktB]{ProduktB1}
787     \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktB]{ProduktB2}
788     \umlVHVinherit{ProduktB1}{AbstraktesProduktB}
789     \umlVHVinherit{ProduktB2}{AbstraktesProduktB}
790
791     \umlVHVdep[arm1=-1.3cm,arm2=-1cm]{Fabrik1}{ProduktA1}
792     \umlVHVdep[arm2=-1cm]{Fabrik1}{ProduktB1}
793
794     \umlVHVdep[arm1=-1.8cm,arm2=-1.5cm]{Fabrik2}{ProduktA2}
795     \umlVHVdep[arm2=-1.5cm]{Fabrik2}{ProduktB2}
796
  
```

```

797 \umluniassoc[anchor2=25]{Klient}{AbstrakteFabrik}
798 \umlVHuniassoc[arm1=-1cm]{Klient}{AbstraktesProduktA}
799 \umlVHuniassoc{Klient}{AbstraktesProduktB}
800 \end{tikzpicture}
801 }

```

iEntwurfsAbstrakteFabrikCode

```

802 \def\liEntwurfsAbstrakteFabrikCode{
803 \li@EntwurfsCode{abstrakte_fabrik}{Produkte}
804 \li@EntwurfsCode{abstrakte_fabrik}{AbstrakteFabrik}
805 \li@EntwurfsCode{abstrakte_fabrik}{Klient}
806 }

```

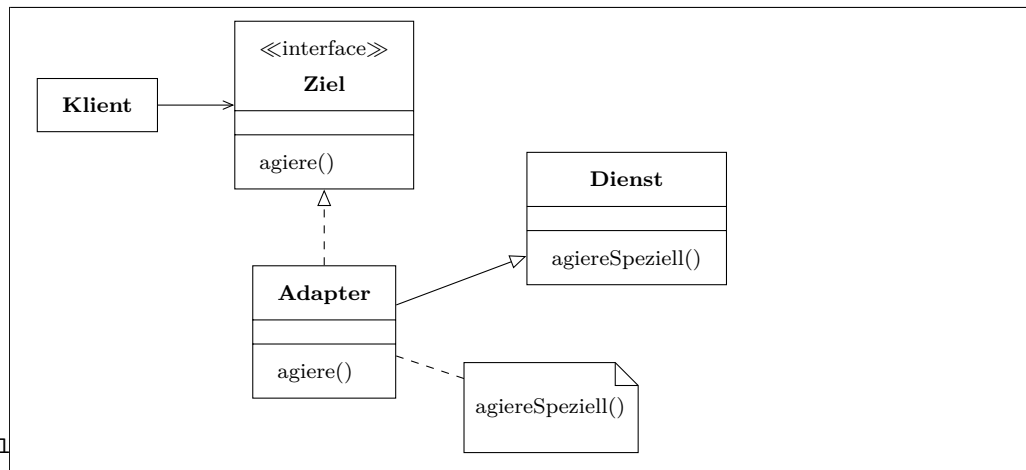
\liEntwurfsAbstrakteFabrik

```

807 \def\liEntwurfsAbstrakteFabrik{
808 \liEntwurfsAbstrakteFabrikBeschreibung
809
810 \liEntwurfsAbstrakteFabrikUml
811
812 \liEntwurfsAbstrakteFabrikCode
813 }

```

#### 2.11.4 Adapter



\liEntwurfsAdapterUml

```

814 \def\liEntwurfsAdapterUml{
815 \begin{tikzpicture}
816 \umlsimpleclass[x=1,y=3]{Klient}{}{}
817 \umlclass[x=4,y=3,type=interface]{Ziel}{}{agiere()}
818 \umlclass[x=4,y=0]{Adapter}{}{agiere()}
819 \umlclass[x=8,y=1.5]{Dienst}{}{agiereSpeziell()}
820
821 \umlreal{Adapter}{Ziel}
822 \umluniassoc{Klient}{Ziel}
823 \umlinherit{Adapter}{Dienst}
824
825 \umlnote[x=7,y=-1,width=2cm]{Adapter}{agiereSpeziell()}
826 \end{tikzpicture}
827 \footcite[so ähnlich wie GoF]{wiki:adapter}
828 }

```

\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.

```

829 \def\liEntwurfsAdapterAkteure{
830   \begin{description}
831
832     \item[Ziel (Target)]
833
834     Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.
835
836     \item[Klient (Client)]
837
838     Der Klient nutzt Dienste über inkompatible Schnittstellen und greift
839     dabei auf adaptierte Schnittstellen zurück.
840
841     \item[Dienst (Adaptee)]
842
843     Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest
844     definierter Schnittstelle an.
845
846     \item[Adapter]
847
848     Der Adapter adaptiert die Schnittstelle des Dienstes auf die
849     Schnittstelle zum Klienten.\footcite{wiki:adapter}
850
851   \end{description}
852 }

```

\liEntwurfsAdapterCode

```

853 \def\liEntwurfsAdapterCode{
854   \li@EntwurfsCode{adapter}{Dienst}
855   \li@EntwurfsCode{adapter}{Ziel}
856   \li@EntwurfsCode{adapter}{Adapter}
857   \li@EntwurfsCode{adapter}{Klient}
858 }

```

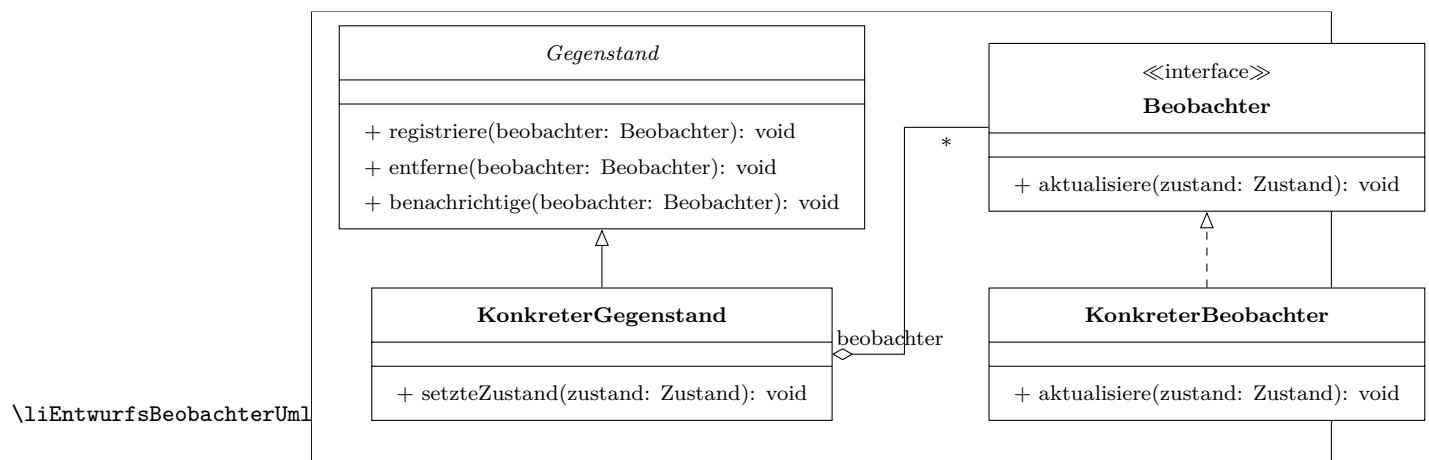
\liEntwurfsAdapter

```

859 \def\liEntwurfsAdapter{
860   \liEntwurfsAdapterUml
861   \liEntwurfsAdapterAkteure
862   \liEntwurfsAdapterCode
863 }

```

### 2.11.5 Beobachter (Observer)



```

864 \def\liEntwurfsBeobachterUml{
865   \begin{tikzpicture}
866     \umlclass[x=0,y=0,type=abstract]{Gegenstand}{}{

```

```

867     + registriere(beobachter: Beobachter): void\\
868     + entferne(beobachter: Beobachter): void\\
869     + benachrichtige(beobachter: Beobachter): void\\
870 }
871 \umlclass[x=0,y=-3]{KonkreterGegenstand}{-}{
872     + setzeZustand(zustand: Zustand): void
873 }
874 \umlinherit{KonkreterGegenstand}{Gegenstand}
875
876 \umlclass[x=8,y=0,type=interface]{Beobachter}{-}{
877     + aktualisiere(zustand: Zustand): void
878 }
879 \umlclass[x=8,y=-3]{KonkreterBeobachter}{-}{
880     + aktualisiere(zustand: Zustand): void
881 }
882 \umlreal{KonkreterBeobachter}{Beobachter}
883
884 \umlHVHaggreg[arg1=beobachter,pos1=0.8,mult2=*,pos2=2.5]
885 {KonkreterGegenstand}{Beobachter}
886 \end{tikzpicture}
887 }

```

\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.

```

888 \def\liEntwurfsBeobachterAkteure{
889   \begin{description}
890     \item[Gegenstand / Subjekt (Subject / Observable)]
891
892     Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also
893     „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne
894     deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An-
895     und Abmeldung von Beobachtern und eine Schnittstelle zur
896     Benachrichtigung von Beobachtern über Änderungen an.\footcite[Seite
897     251]{gof}
898
899     \item[Beobachter (Observer)]
900
901     Die Beobachter (auf Englisch auch subscriber, also „Abonnent“,
902     genannt) definieren eine Aktualisierungsschnittstelle.
903
904     \item[konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)]
905
906     Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert
907     den relevanten Zustand und benachrichtigt alle Beobachter bei

```



```

908 Zustandsänderungen über deren Aktualisierungsschnittstelle. Es
909 verfügt über eine Schnittstelle zur Erfragung des aktuellen
910 Zustands.
911
912 \item[Konkrete Beobachter (ConcreteObserver)]
913
914 Konkrete Beobachter verwalten die Referenz auf ein konkretes
915 Subjekt, dessen Zustand sie beobachten und speichern und dessen
916 Zustand konsistent ist. Sie implementieren eine
917 Aktualisierungsschnittstelle unter Verwendung der
918 Abfrageschnittstelle des konkreten Subjekts.
919 \footcite{wiki:beobachter}
920 \end{description}
921 }

```

\liEntwurfsBeobachterCode

```

922 \def\liEntwurfsBeobachterCode{
923 \li@EntwurfsCode{beobachter}{Gegenstand}
924 \li@EntwurfsCode{beobachter}{KonkreterGegenstand}
925 \li@EntwurfsCode{beobachter}{Beobachter}
926 \li@EntwurfsCode{beobachter}{KonkreterBeobachterA}
927 \li@EntwurfsCode{beobachter}{KonkreterBeobachterB}
928 \li@EntwurfsCode{beobachter}{Klient}
929 }

```

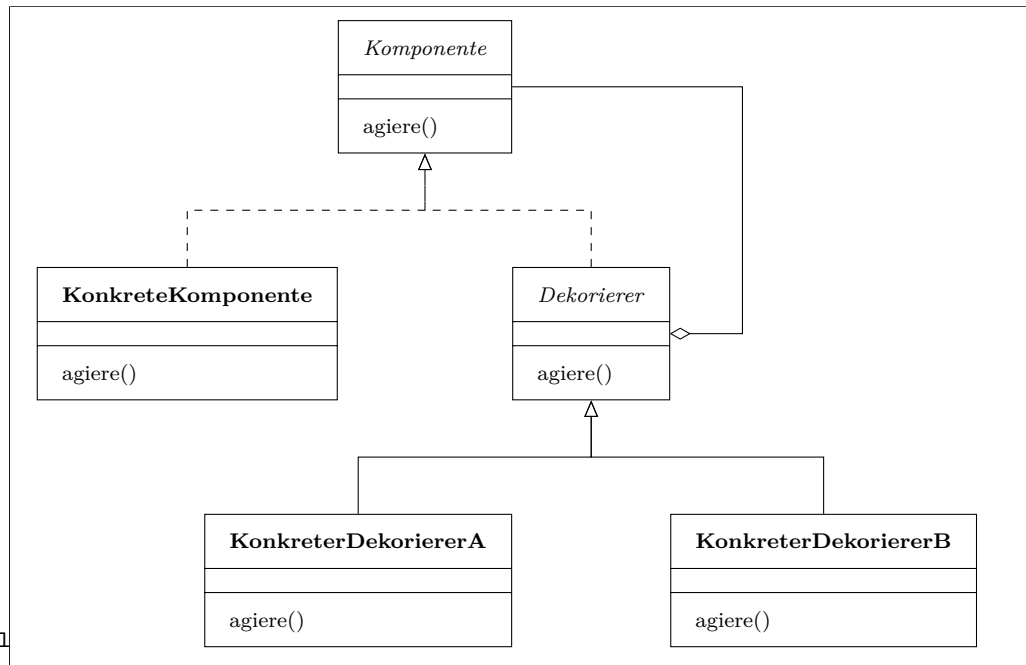
\liEntwurfsBeobachter

```

930 \def\liEntwurfsBeobachter{
931 \liEntwurfsBeobachterUml
932 \liEntwurfsBeobachterAkteure
933 \liEntwurfsBeobachterCode
934 }

```

## 2.11.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

935 \def\liEntwurfsDekoriererUml{
936 \begin{tikzpicture}
937 \umlclass[type=abstract]{Komponente}{+agiere()}
938 \umlclass[below left=1.5cm and 0cm of Komponente]{KonkreteKomponente}{+agiere()}
939 \umlclass[below right=1.5cm and 0cm of Komponente,type=abstract]{Dekorierer}{+agiere()}
940

```

```

941 \umlVHVreal{KonkreteKomponente}{Komponente}
942 \umlVHVreal{Dekorierer}{Komponente}
943
944 \umlclass[below left=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererA}{\{agiere()\}}
945 \umlclass[below right=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererB}{\{agiere()\}}
946
947 \umlVHVinherit{KonkreterDekoriererA}{Dekorierer}
948 \umlVHVinherit{KonkreterDekoriererB}{Dekorierer}
949
950 \umlHVHaggreg[arm1=2cm]{Dekorierer}{Komponente}
951 \footcite{wiki:dekorierer}
952 \end{tikzpicture}
953 }

```

\liEntwurfsDekoriererCode

```

954 \def\liEntwurfsDekoriererCode{
955 \li@EntwurfsCode{dekorierer}{Komponente}
956 \li@EntwurfsCode{dekorierer}{KonkreteKomponente}
957 \li@EntwurfsCode{dekorierer}{Dekorierer}
958 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererA}
959 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererB}
960 \li@EntwurfsCode{dekorierer}{Klient}
961 }

```

\liEntwurfsDekorierer

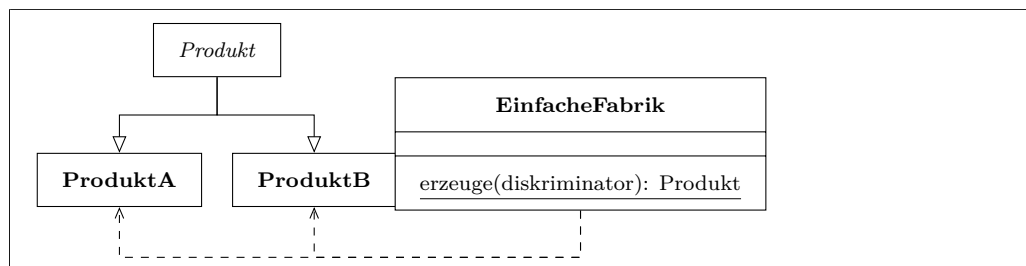
```

962 \def\liEntwurfsDekorierer{
963 \liEntwurfsDekoriererUml
964 \liEntwurfsDekoriererAkteure
965 \liEntwurfsDekoriererCode
966 }

```

### 2.11.7 Einfache Fabrik (Simple Factory)

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



```

967 \def\liEntwurfsEinfacheFabrikUml{
968 \begin{tikzpicture}
969 \umlsimpleclass[type=abstract]{Produkt}
970 \umlsimpleclass[below left=1cm and -0.65cm of Produkt]{ProduktA}
971 \umlsimpleclass[below right=1cm and -0.65cm of Produkt]{ProduktB}
972 \umlVHVinherit{Produkt}{ProduktA}
973 \umlVHVinherit{Produkt}{ProduktB}
974 \umlclass[below right=0cm and 1.5cm of Produkt]{EinfacheFabrik}{
975 }{
976 \umlstatic{erzeuge(diskriminator): Produkt}\}
977 }
978 \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktA}
979 \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktB}
980 \end{tikzpicture}
981 }

```

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.

```

982 \def\liEntwurfsEinfacheFabrikAkteure{
983   \begin{description}
984     \item[EinfacheFabrik]
985
986     Eine Klasse mit einer Erzeugungsmethode, die über eine größere
987     Bedingung verschiedene Objekt instanziert.
988
989     \item[Produkt]
990
991     Eine abstrakte Klasse, die von den konkreten Produkten geerbt wird.
992
993     \item[KonkretesProdukt]
994
995     Ein konkretes Produkt, das von der einfachen Fabrik erzeugt wird.
996   \end{description}
997 }
```

\liEntwurfsEinfacheFabrik

```

998 \def\liEntwurfsEinfacheFabrik{
999   \liEntwurfsEinfacheFabrikUml
1000   \liEntwurfsEinfacheFabrikAkteure
1001 }
```

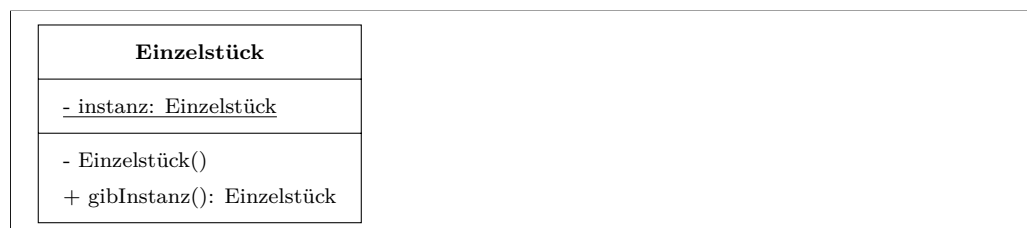
### 2.11.8 Einzelstück (Singleton)

urfsEinzelstueckBeschreibung

```

1002 \def\liEntwurfsEinzelstueckBeschreibung{
1003   Stellt sicher, dass nur \emph{genau eine Instanz einer Klasse} erzeugt
1004   wird.\footcite[Seite 38]{eilebrecht}
1005 }
1006
```

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```

1007 \def\liEntwurfsEinzelstueckUml{
1008   \begin{tikzpicture}
1009     \umlclass{Einzelstück}{
1010       \umlstatic{- instanz: Einzelstück}\\
1011     }{
1012       - Einzelstück()\\
1013       + gibInstanz(): Einzelstück
1014     }
1015   \end{tikzpicture}
1016 }
```

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.

```

1017 \def\liEntwurfsEinzelstueckAkteure{
1018   \begin{description}
1019     \item[Einzelstück (Singleton)]
1020
1021     stellt eine statische Methode bereit, mit deren Hilfe die Klienten
1022     nur auf eine einzige Instanz der Klasse zugreifen können.
1023   \end{description}
1024 }

```

\liEntwurfsEinzelstueckCode

```

1025 \def\liEntwurfsEinzelstueckCode{
1026   \li@EntwurfsCode{einzelstueck}{Einzelstueck}
1027 }

```

\liEntwurfsEinzelstueck

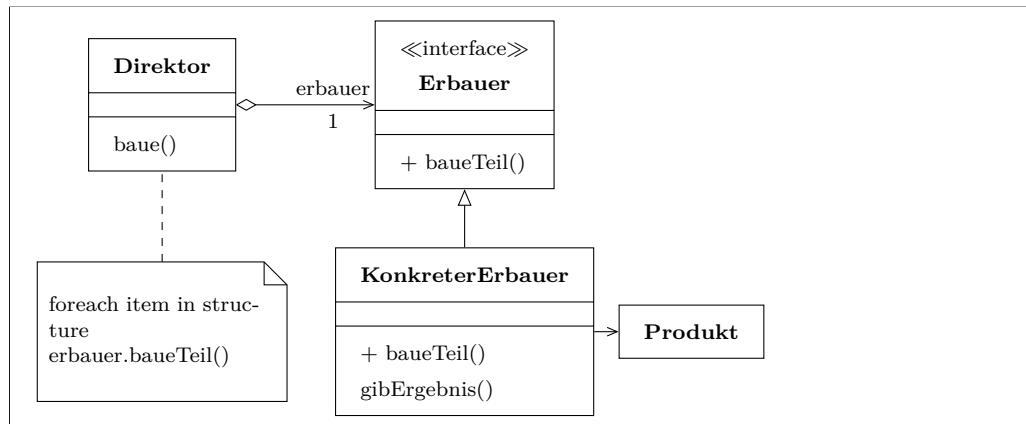
```

1028 \def\liEntwurfsEinzelstueck{
1029   \liEntwurfsEinzelstueckBeschreibung
1030
1031   \liEntwurfsEinzelstueckUml
1032
1033   \liEntwurfsEinzelstueckAkteure
1034
1035   \liEntwurfsEinzelstueckCode
1036 }

```

### 2.11.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

1037 \def\liEntwurfsErbauerUml{
1038   \begin{tikzpicture}
1039     \umlsimpleclass[x=7,y=0]{Produkt}{}{}
1040     \umlclass[x=4,y=3,type=interface]{Erbauer}{}{+ baueTeil()}
1041     \umlclass[x=4,y=0]{KonkreterErbauer}{}{
1042       + baueTeil()\n
1043       gibErgebnis()}
1044     \umlclass[x=0,y=3]{Direktor}{}{baue()}
1045
1046     \umluniagg[arg2=erbauer,pos2=0.7,mult2=1]{Direktor}{Erbauer}
1047     \umluniassoc{KonkreterErbauer}{Produkt}
1048     \umlinherit{KonkreterErbauer}{Erbauer}
1049
1050     \umlnote[x=0,y=0,width=3cm]{Direktor}{
1051       foreach item in structure\n
1052       erbauer.baueTeil()

```

```

1053 }
1054 \end{tikzpicture}
1055 \footcite{wiki:erbauer}
1056 }

```

\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.

```

1057 \def\liEntwurfsErbauerAkteure{
1058   \begin{description}
1059     \item[Erbauer]
1060
1061     Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der
1062     Teile eines komplexen Objektes.
1063
1064     \item[KonkreterErbauer]
1065
1066     Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch
1067     Implementierung der Schnittstelle. Außerdem definiert und verwaltet er
1068     die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine
1069     Schnittstelle zum Auslesen des Produkts.
1070
1071     \item[Direktor]
1072
1073     Der Direktor konstruiert ein komplexes Objekt unter Verwendung der
1074     Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer
1075     zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder
1076     benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom
1077     Klienten.
1078
1079     \item[Produkt]
1080
1081     Das Produkt repräsentiert das zu konstruierende komplexe Objekt.
1082     \footcite{wiki:erbauer}
1083   \end{description}
1084 }

```

\liEntwurfsErbauer

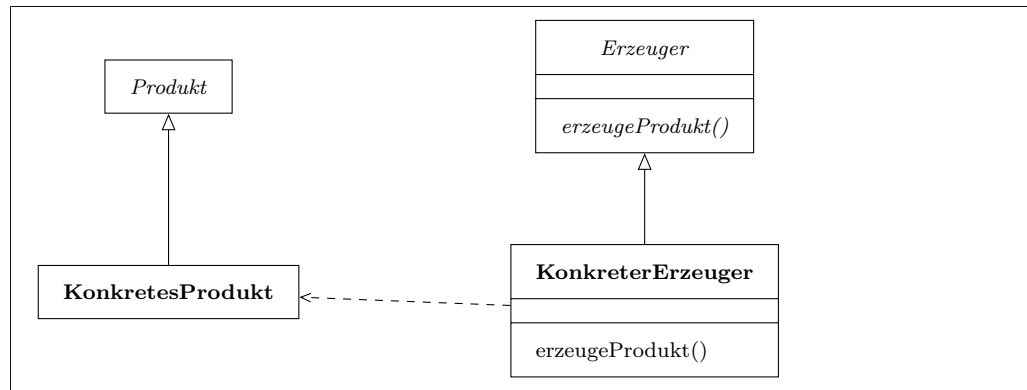
```

1085 \def\liEntwurfsErbauer{
1086   \liEntwurfsErbauerUml
1087   \liEntwurfsErbauerAkteure
1088 }

```

#### 2.11.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

1089 \def\liEntwurfsFabrikmethodeUml{
1090   \begin{tikzpicture}
1091     \umlsimpleclass[type=abstract]{Produkt}
1092     \umlsimpleclass[below=2cm of Produkt]{KonkretesProdukt}
1093     \umlinherit{KonkretesProdukt}{Produkt}
1094
1095     \umlclass[type=abstract,right=4cm of Produkt]{Erzeuger}{\{
1096       \textit{erzeugeProdukt()}\}
1097   }
1098   \umlclass[below=1.2cm of Erzeuger]{KonkreterErzeuger}{\{
1099     erzeugeProdukt()
1100   }
1101   \umlinherit{KonkreterErzeuger}{Erzeuger}
1102
1103   \umldep{KonkreterErzeuger}{KonkretesProdukt}
1104   \end{tikzpicture}
1105 }

```

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).

```

1106 \def\liEntwurfsFabrikmethodeAkteure{
1107   \begin{description}
1108     \item[Produkt]
1109
1110     Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das
1111     zu erzeugende Produkt.
1112
1113     \item[KonkretesProdukt]
1114
1115     KonkretesProdukt implementiert die Produkt-Schnittstelle.
1116
1117     \item[Erzeuger]
1118
1119     Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt
1120     zu erzeugen und kann eine Default-Implementierung beinhalten.
1121
1122     \item[KonkreterErzeuger]
1123

```

```

1124     KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm
1125     entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den
1126     Konstruktor einer konkreten Produkt-Klasse aufruft).
1127
1128     \footcite{wiki:fabrikmethode}
1129 \end{description}
1130 }

```

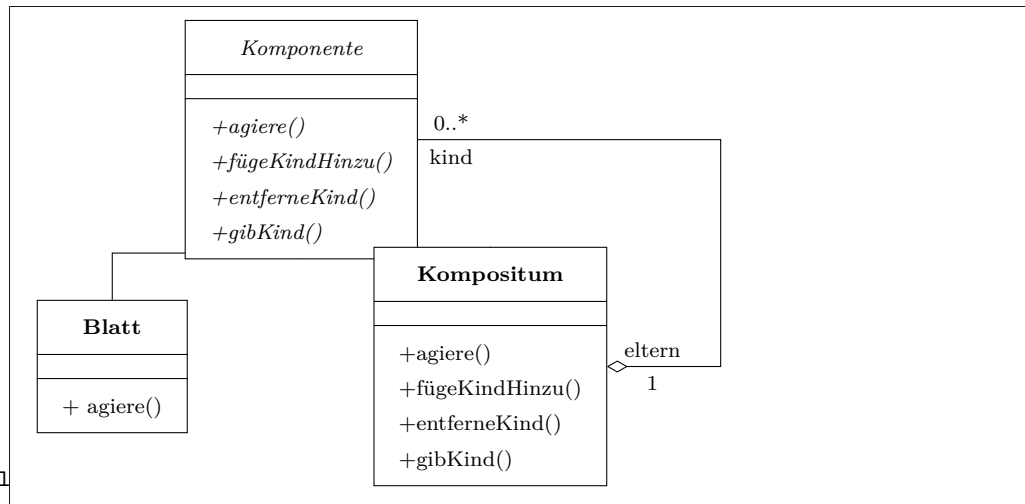
\liEntwurfsFabrikmethode

```

1131 \def\liEntwurfsFabrikmethode{
1132   \liEntwurfsFabrikmethodeUml
1133   \liEntwurfsFabrikmethodeAkteure
1134 }

```

### 2.11.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

1135 \def\liEntwurfsKompositumUml{
1136   \begin{tikzpicture}
1137     \umlclass[x=2.5,y=3,type=abstract]{Komponente}{-}{
1138       \textit{+agiere()}\}
1139     \textit{+fügeKindHinzu()}\}
1140     \textit{+entferneKind()}\}
1141     \textit{+gibKind()}\}
1142   }
1143   \umlclass[x=0]{Blatt}{-}{+ agiere()}\}
1144   \umlclass[x=5]{Kompositum}{-}{
1145     +agiere()\}
1146     +fügeKindHinzu()\}
1147     +entferneKind()\}
1148     +gibKind()}\}
1149 }
1150
1151   \umlVHVinherit{Kompositum}{Komponente}
1152   \umlVHVinherit{Blatt}{Komponente}
1153   \umlHVHaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,]
1154 \end{tikzpicture}
1155 }

```

\liEntwurfsFabrikmethode

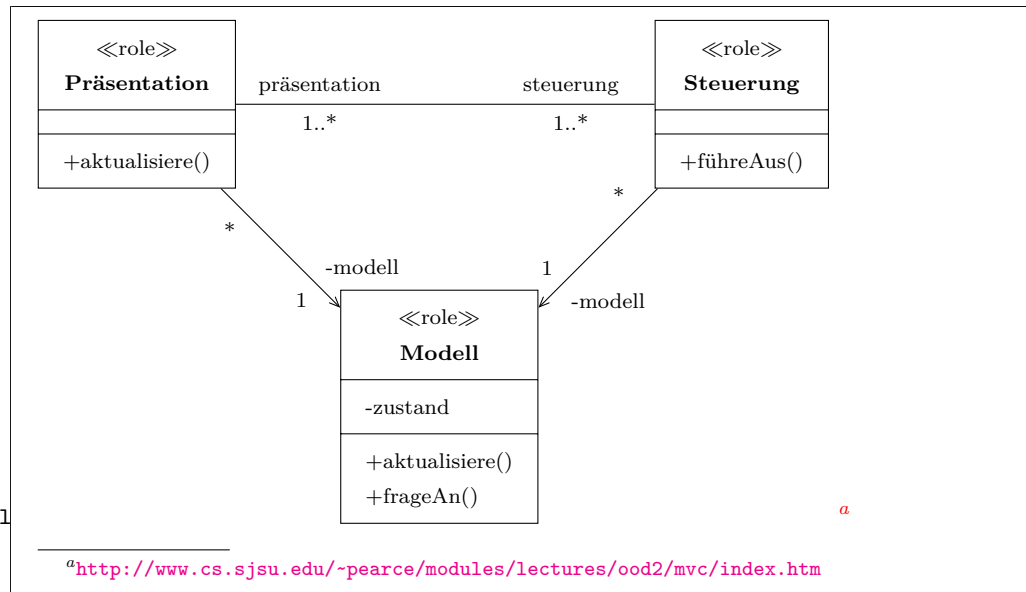
```

1156 \def\liEntwurfsKompositum{
1157   \liEntwurfsKompositumUml
1158   \liEntwurfsKompositumAkteure
1159 }

```

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

ModellPraesentationSteuerungUml



```

1160 \def\liEntwurfsModellPraesentationSteuerungUml{
1161   \begin{tikzpicture}
1162     \umlclass[x=-4cm,type=role]{Präsentation}{+aktualisiere()}
1163     \umlclass[x=4cm,type=role]{Steuerung}{+führeAus()}
1164     \umlclass[y=-4cm,type=role]{Modell}{
1165       -zustand
1166     }{
1167       +aktualisiere()\
1168       +frageAn()
1169     }
1170
1171     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
1172     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}
1173     \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
1174   \end{tikzpicture}
1175   \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
1176 }

```

ModellPraesentationSteuerung

```

1177 \def\liEntwurfs{
1178   \liEntwurfsModellPraesentationSteuerungUml
1179   \liEntwurfsModellPraesentationSteuerungAkteure
1180 }

```

### 2.11.13 Stellvertreter (Proxy)

\liEntwurfsStellvertreterUml

```

1181 \def\liEntwurfsStellvertreterUml{
1182   \begin{tikzpicture}
1183     \umlsimpleclass[x=-1,y=2]{Klient}
1184
1185     \umlclass[x=2,y=2]{Subjekt}{+ agiere()}
1186     \umlclass[x=0,y=-1]{KonkretesSubjekt}{+ agiere()}
1187     \umlclass[x=4,y=-1]{Stellvertreter}{+ agiere()}
1188
1189     \umlVHVinherit{KonkretesSubjekt}{Subjekt}
1190     \umlVHVinherit{Stellvertreter}{Subjekt}
1191     \umluniassoc{Stellvertreter}{KonkretesSubjekt}
1192     \umluniassoc{Klient}{Subjekt}
1193   \end{tikzpicture}
1194 }

```

\liEntwurfsStellvertreterCode



```

1195 \def\liEntwurfsStellvertreterCode{
1196   \li@EntwurfsCode{stellvertreter}{Subjekt}
1197   \li@EntwurfsCode{stellvertreter}{KonkretesSubjekt}
1198   \li@EntwurfsCode{stellvertreter}{Stellvertreter}
1199   \li@EntwurfsCode{stellvertreter}{Klient}
1200 }

```

\liEntwurfsStellvertreter

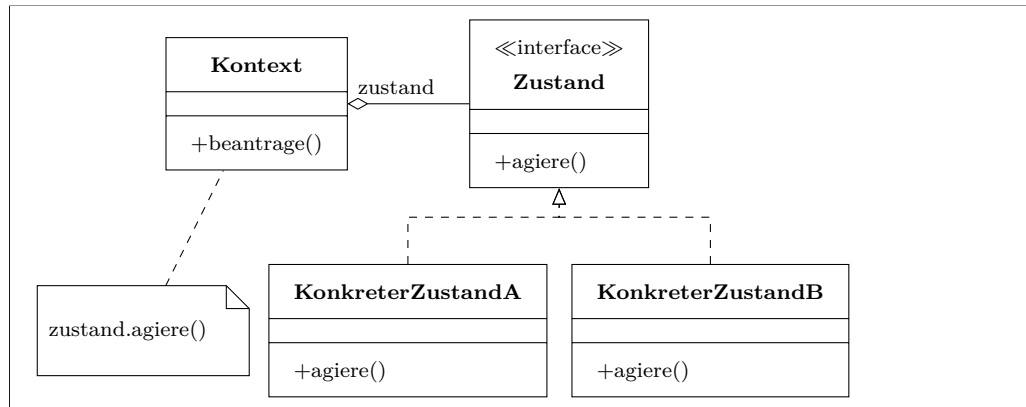
```

1201 \def\liEntwurfsStellvertreter{
1202   \liEntwurfsStellvertreterUml
1203   \liEntwurfsStellvertreterCode
1204 }

```

#### 2.11.14 Zustand (State)

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



```

1205 \def\liEntwurfsZustandUml{
1206   \begin{tikzpicture}
1207     \umlclass[x=-1,y=3]{Kontext}{+beantrage()}
1208     \umlclass[x=3,y=3,type=interface]{Zustand}{+agiere()}
1209     \umlclass[x=1,y=0]{KonkreterZustandA}{+agiere()}
1210     \umlclass[x=5,y=0]{KonkreterZustandB}{+agiere()}
1211
1212     \umlVHVreal{KonkreterZustandA}{Zustand}
1213     \umlVHVreal{KonkreterZustandB}{Zustand}
1214
1215     \umlaggreg[arg=zustand,pos=0.4]{Kontext}{Zustand}
1216
1217     \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
1218   \end{tikzpicture}
1219 }

```

\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.

```

1220 \def\liEntwurfsZustandAkteure{
1221   \begin{description}
1222     \item[Kontext (Context)]
1223
1224     definiert die clientseitige Schnittstelle und verwaltet die separaten

```

```

1225     Zustandsklassen.
1226
1227     \item[State (Zustand)]
1228
1229     definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
1230     implementiert gegebenenfalls ein Standardverhalten.
1231
1232     \item[KontreterZustand (ConcreteState)]
1233
1234     implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
1235     verbunden ist.
1236 \end{description}
1237 }

\liEntwurfsZustandCode

1238 \def\liEntwurfsZustandCode{
1239   \li@EntwurfsCode{zustand}{Kontext}
1240   \li@EntwurfsCode{zustand}{Zustand}
1241 }

\liEntwurfsZustand

1242 \def\liEntwurfsZustand{
1243   \liEntwurfsZustandUml
1244   \liEntwurfsZustandAkteure
1245   \liEntwurfsZustandCode
1246 }

1247

```

## 2.12 er.sty

```
1248 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1249 \ProvidesPackage{lehramt-informatik-er}[2020/06/13 Zum Setzen von
1250 ER-Diagrammen]
```

```
1251 \RequirePackage{tikz-er2}
1252 \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);

```

```
1253 \RequirePackage{soul}
```

```
1254 \RequirePackage{fontawesome}
```

### 2.12.3 Makro-Kürzel

```

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

```

```
1255 \ExplSyntaxOn
```

```
\liErEntity
```

```
1256 \def\liErEntity#1{\textbf{#1}}
```

```
\liErRelationship
```

```
1257 \def\liErRelationship#1{\ul{#1}}
```

```
\liErAttribute
```

```
1258 \def\liErAttribute#1{\emph{#1}}
```

```
\liErMpEntity mp = marginpar
```

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

```

1259 \def\liErMpEntity#1{
1260   \liErEntity{#1}
1261   \marginpar{
1262     \liErEntity{\tiny\faSquareO{~E:~#1}
1263   }
1264 }

```

□

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

```

1265 \def\liErMpRelationship#1{
1266   \liErRelationship{#1}
1267   \marginpar{
1268     \liErRelationship{\tiny\faGg{~R:~#1}
1269   }
1270 }

```

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

```

1271 \def\liErMpAttribute#1{
1272   \liErAttribute{#1}
1273   \marginpar{
1274     \liErAttribute{\tiny\faCircleThin{~A:~#1}
1275   }
1276 }

```

```

\liErDatenbankName Let-Abkürzung: \let\d=\liErDatenbankName
                    datenbank name
1277 \def\liErDatenbankName#1{
1278   {
1279     \footnotesize\texttt{(#1)}
1280   }
1281 }

1282 \ExplSyntaxOff
1283

```

## 2.13 formale-sprachen.sty

```

1284 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1285 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
1286 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]

1287 \directlua{
1288   formale_sprachen = require('lehramt-informatik-formale-sprachen')
1289 }

1290 \RequirePackage{hyperref}

1291 \liLadePakete{mathe,typographie}

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

1292 \def\liMengeOhneMathe#1{\{ #1 \}}
1293 \def\liMenge#1{%
1294   \ifmmode%
1295     \liMengeOhneMathe{#1}%
1296   \else%
1297     $\liMengeOhneMathe{#1}$%
1298   \fi%
1299 }

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

1300 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

1301 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1302 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1303 \def\liPotenzmenge#1{$\liPotenzmengeOhneMathe{#1}$}

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

1304 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
1305 \let\liZustandsmenge=\liPotenzmenge

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

1306 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1307 \def\liUeberfuehrungsFunktion#1{
1308   \ifmmode
1309     \liUeberfuehrungsFunktionOhneMathe{#1}
1310   \else
1311     $\liUeberfuehrungsFunktionOhneMathe{#1}$
1312   \fi
1313 }

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

1314 \def\liAlphabet#1{$\Sigma = \{ #1 \}$}

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

1315 \def\liBandAlphabet#1{$\Gamma = \Sigma \cup \{ #1 \}$}

\liZustandsBuchstabe

1316 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1317 \def\liZustandsBuchstabeGross{Z}

```

```

\liZustandsmengeNr
1318 \def\zustandsnamens@liste#1#2{
1319   $
1320   \{
1321     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1322   \}
1323   $
1324 }
1325 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

\liZustandsmengeNrGross
1326 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

\liZustandsname \liZustandsname{1}: $z_1$
1327 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

\liZustandsnameGross \liZustandsnameGross{1}: $Z_1$
1328 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

\liAbleitung \liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1329 \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}

1330 \NewDocumentEnvironment { liProduktionsRegeln }
1331 { 0{P} +b }
1332 {
1333   \liGeschweifteKlammern{#1}
1334   {
1335     \begin{align*}
1336       \directlua{formale_sprachen.produktions_regeln('#2')}
1337     \end{align*}
1338   }{-0.2cm}{-1.5cm}
1339 } {}

\liProduktionen \liProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
1340 \def\liProduktionen#1{
1341   \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
1342 }

\liZustandsnameTiefgestellt Automatisch tiefgestellte Nummerierung \z1
Let-Abkürzung: \let\z=\liZustandsnameTiefgestellt
1343 \def\liZustandsnameTiefgestellt#1{
1344   \ifmmode
1345     \liZustandsBuchstabe\sb{#1}
1346   \else
1347     $\liZustandsBuchstabe\sb{#1}$
1348   \fi
1349 }

1350 \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}

```

```

1351 \NewDocumentCommand{ \liAusdruck } { O{L} m m } {
1352   $
1353   \tl_if_empty:nTF {#1} {} {#1 =}
1354   \{
1355     \, #2 \,
1356     |
1357     \, #3 \,
1358   \}$
1359 }
1360 \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

1361 \def\liFlaci#1{%
1362   \par
1363   {%
1364     \scriptsize
1365     Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1366     Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1367     Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1368     \href{https://flaci.com/#1}{flaci.com/#1}%
1369   }%
1370   \par
1371 }

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

1372 \ExplSyntaxOn
1373 \NewDocumentCommand {\liGrammatik} { O{G} m } {
1374   \tl_set:Nn \l_variablen_tl {V}
1375   \tl_set:Nn \l_alphabet_tl {\Sigma}
1376   \tl_set:Nn \l_produktionen_tl {P}
1377   \tl_set:Nn \l_start_tl {S}
1378
1379   \keys_define:nn { grammatik } {
1380     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1381     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1382     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1383     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1384   }
1385
1386   \keys_set:nn { grammatik } { #2 }
1387
1388   $#1 = (
1389     \l_variablen_tl,
1390     \l_alphabet_tl,
1391     \l_produktionen_tl,
1392     \l_start_tl
1393   )$
1394 }
1395 \ExplSyntaxOff
1396

```



## 2.14 formatierung.sty

```
1397 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1398 \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.

```
1399 \RequirePackage{mathpazo}
1400 \RequirePackage[no-math]{fontspec}
1401 \setmainfont{texgyrepagella}
```

### 2.14.2 Farben

```
1402 \RequirePackage{xcolor}
1403 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

### 2.14.3 Überschriften

```
1404 \RequirePackage{titlesec}
1405 \titleformat{\chapter}[display]{\bfseries}{\Opt}{\LARGE}
1406 \titlespacing{\chapter}{\Opt}{\Opt}{*1}
1407 \titleformat{\paragraph}[hang]{\normalsize\bfseries}{\theparagraph}{1em}{\}
1408 \setcounter{secnumdepth}{0}
```

### 2.14.4 Listen

```
1409 \RequirePackage{paralist}
1410 \renewcommand\labelitemi{-}
1411 \renewcommand\labelitemii{-}
1412 \renewcommand\labelitemiii{-}
1413 \renewcommand\labelitemiv{-}
1414 % Counter: enumi enumii enumiii enumiv
1415 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1416 \renewcommand{\labelenumi}{(\alph{enumi})}
1417 \renewcommand{\labelenumii}{(\roman{enumii})}
```

### 2.14.5 Kasten

```
1418 \RequirePackage{mdframed}
1419 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1420 \NewDocumentEnvironment { liKasten }{ } {
1421   \begin{mdframed}[backgroundcolor=white!0]
1422 } {
1423   \end{mdframed}
1424 }
```

### 2.14.6 Header

```
1425 \RequirePackage{fancyhdr}
1426 \fancyhead[L,C,R]{\}
1427 \fancyfoot[L]{\}
1428 \fancyfoot[C]{\}
1429 \fancyfoot[R]{\thepage}
1430 \pagestyle{fancy}
1431 \renewcommand{\headrulewidth}{\Opt}
1432 \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.

```
1433 \RequirePackage{setspace}
```



## 2.15 gantt.sty

```

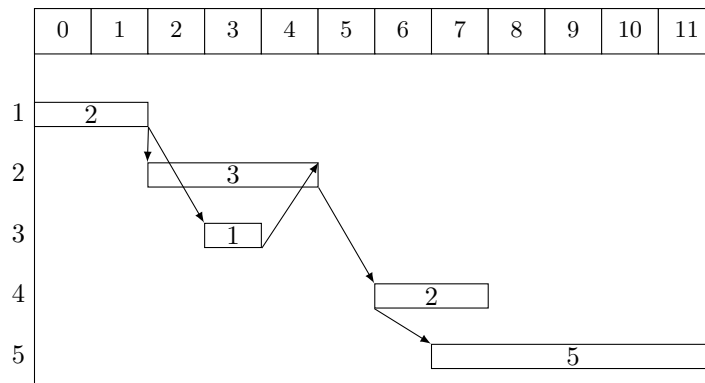
1435 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1436 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\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}

```



```

1437 \RequirePackage{tikz-uml}
1438 \RequirePackage{pgfgantt}
1439 \setganttlinklabel{f-s}{}
1440 \setganttlinklabel{s-s}{}
1441 \setganttlinklabel{f-f}{}
1442 \setganttlinklabel{s-f}{}

1443

```

## 2.16 grafik.sty

```
1444 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1445 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1446 TikZ, um Grafiken zeichnen zu können und graphicx um Bilder laden zu können.]
1447 \ExplSyntaxOn
1448 \RequirePackage{tikz}
1449 \RequirePackage{graphicx}

\liGrafikLogoPfad

1450 \def\liGrafikLogoPfad#1{
1451   \LehramtInformatikRepository / .tex / Logo / #1
1452 }

\liGrafikCCLizenz

1453 \NewDocumentCommand{ \liGrafikCCLizenz } { 0{} } {
1454   \includegraphics[#1]{
1455     \liGrafikLogoPfad{CC-by-nc-sa.eps}
1456   }
1457 }

\liGrafikLogo

1458 \NewDocumentCommand{ \liGrafikLogo } { 0{} } {
1459   \includegraphics[#1]{
1460     \liGrafikLogoPfad{Logo_nur-Pfade.eps}
1461   }
1462 }

1463 \ExplSyntaxOff
1464
```

## 2.17 graph.sty

```

1465 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1466 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1467 \RequirePackage{tikz}

```

Für die die Adjazenzliste (xrightarrow)

```

1468 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

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

```

1469 \RequirePackage{blkarray}
1470 \usetikzlibrary{arrows.meta}

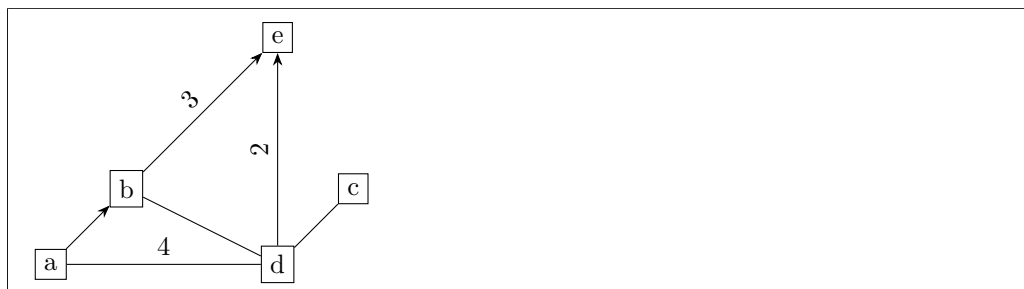
```

```

\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}

```



```

1471 \tikzset{
1472   li graph/.style={
1473     every node/.style={
1474       rectangle,
1475       draw,
1476     },
1477     every edge/.style={
1478       >={Stealth[black]},
1479       draw,
1480     },
1481     every edge/.append style={
1482       every node/.style={
1483         sloped,
1484         auto,
1485       }
1486     }
1487   },
1488   li markierung/.style={
1489     ultra thick,
1490   }
1491 }

```

**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}

```

```

1492 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1493

```

## 2.18 hanoi.sty

```
1494 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1495 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
1496 von Hanoi-Grafiken]
```

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

```
1497 \RequirePackage{tikz}
1498 \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}

1499 \def\li@mset #1[#2]=#3{%
1500 \expandafter\xdef\csname #1#2\endcsname{#3}
1501 }
1502 \def\li@mget #1[#2]{%
1503 \csname #1#2\endcsname
1504 }
1505 \def\li@minc #1[#2] += #3{%
1506 \pgfmathparse{\li@mget #1[#2] + #3}%
1507 \li@mset #1[#2] = \pgfmathresult
1508 }
1509
1510 \def\liHanoi#1#2{
1511   \edef\li@numdiscs{#1}
1512   \def\li@sequence{#2}
1513   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1514     % init colors
1515     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purp
1516     \li@mset col[\j]={\c};
1517     % draw poles and init pole counters
1518     \foreach \j in {1,2,3}{
1519       \li@mset pos[\j]=0
1520       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1521     }
1522     % draw base
1523     \draw (.5,-.5) -- +(3,0);
1524     % draw discs
1525     \foreach[count=\k] \i/\j in \li@sequence{
1526       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*
1527       \li@minc pos[\j] += {.5}
1528     }
1529   \end{tikzpicture}
1530 }

1531
```

## 2.19 klassen-konfiguration-aufgabe.sty

```
1532 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1533 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-aufgabe}[2021/09/01 Die
1534 Klasse Aufgabe konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

Formatierung muss vor literatur sein, sonst Option clash

```
1535 \liLadePakete{
1536   formatierung,
1537   abmessung,
1538   literatur-dummy,
1539   makros,
1540   aufgaben-metadaten,
1541   kopf-fusszeilen,
1542   mathe,
1543   grafik,
1544   meta
1545 }
```

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

```
1546 \RequirePackage[ngerman]{babel}
```

In Aufgaben wollen wir das Additum sehen.

```
1547 \LiADDITUMtrue
1548 \ExplSyntaxOn

1549 \cs_gset:Npn \stichwoerter_formatiert: {
1550   \tl_if_empty:NTF \g_auf_stichwoerter_tl {} {
1551     \textbf{Stichwörter:} ~
1552     \g_auf_stichwoerter_tl
1553     \par
1554   }
1555 }

1556 \cs_gset:Npn \horizontale_linie: {
1557   \par
1558   \noindent
1559   \rule{\textwidth}{0.8pt}
1560   \par
1561 }

1562 \cs_gset:Npn \thematik_formatiert: {
1563   \tl_if_empty:NTF \g_auf_thematik_tl {} {
1564     \textit{
1565       ( \g_auf_thematik_tl )
1566     }
1567   }
1568 }
```

`\liAufgabenMetadaten`

```
1569 \def\liAufgabenMetadaten#1{
1570   \liMetaSetze{#1}
1571
1572   \tl_if_empty:NTF \g_auf_examen_nummer_tl {} {
1573     {
1574       \noindent
1575       \large
1576       \gib_einzelpruefung_trenner:
1577       \par\medskip
1578     }
1579   }

1580
1581   {
1582     \noindent
1583     \bfseries
1584     \Large
1585     \tl_if_empty:NTF \g_auf_examen_nummer_tl {
```



```

1586     \g_auf_titel_tl
1587   } {
1588     \_gib_aufgaben_pfad_trenner:
1589   }
1590 }
1591 \hfill \thematik_formatiert:
1592 \par
1593
1594 \medskip
1595
1596 \noindent
1597 {\footnotesize\stichwoerter_formatiert:}
1598
1599 \horizontale_linie:
1600
1601 \bigskip
1602
1603 \par
1604 % Keine Einrückung
1605 \@afterindentfalse
1606 \@afterheading
1607 }

1608 \AddToHook{enddocument}{
1609   \vfill
1610   {
1611
1612     \liLogoTextProjekt
1613     \bigskip
1614
1615     \liLogoTextCCLizenz
1616     \bigskip
1617
1618     \begin{spacing}{1}
1619       \tiny
1620       \noindent
1621       \liMetaHilfMit
1622
1623       \liMetaQuelltext
1624       \_gib_github_url_href:
1625     \end{spacing}
1626   }
1627 }

1628 \ExplSyntaxOff
1629

```

## 2.20 klassen-konfiguration-examen.sty

```
1630 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1631 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-examen}[2021/09/04 Die
1632 Klasse liSetzeExamen konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

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

```
1633 \liLadePakete{
1634   formatierung,
1635   literatur-dummy,
1636   makros,
1637   aufgaben-metadaten,
1638   abmessung,
1639   typographie,
1640   grafik,
1641   meta
1642 }
```

Formatierung für die Überschriften setzen.

```
1643 \RequirePackage{titlesec}
1644 \titleformat{\section}{\huge\filcenter\bfseries}{\thesection}{1em}{}
1645 \newcommand{\sectionbreak}{\clearpage}
1646 \setcounter{secnumdepth}{0}
1647 \liLadeAllePakete
```

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

```
1648 \RequirePackage[ngerman]{babel}
1649 \RequirePackage{standalone}
1650 \ExplSyntaxOn
```

`\liSetzeExamenThemaNr`

```
1651 \def\liSetzeExamenThemaNr#1{
1652   \tl_gset:Nn \g_auf_examen_thema_nr_tl { #1 }
1653   \section{Thema-Nr.~#1}
1654 }
```

`\liSetzeExamenTeilaufgabeNr`

```
1655 \def\liSetzeExamenTeilaufgabeNr#1{
1656   \tl_gset:Nn \g_auf_examen_teilaufgabe_nr_tl { #1 }
1657   \subsection{Teilaufgabe-Nr.~#1}
1658 }
```

`\liBindeAufgabeEin`

```
1659 \def\liBindeAufgabeEin#1{
1660   \tl_gset:Nn \g_auf_examen_aufgabe_nr_tl { #1 }
1661   \input{
1662     \LehramtInformatikRepository /
1663     Staatsexamen /
1664     \g_auf_examen_nummer_tl /
1665     \g_auf_examen_jahr_tl /
1666     \g_auf_examen_monat_tl /
1667     \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
1668       Thema - \g_auf_examen_thema_nr_tl /
1669     }
1670     \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
1671       Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl /
1672     }
1673     Aufgabe - \g_auf_examen_aufgabe_nr_tl .tex
1674   }
1675 }
```

`\liAufgabenMetadaten` Das Metadaten-Makro überschreiben

```
1676 \def\liAufgabenMetadaten#1{
1677   \liMetaSetze{#1}
1678   \subsubsection{\_gib_aufgaben_titel:}
1679 }
```

```

1680 \cs_new:Npn \titel_seite:
1681 {
1682   \pagestyle{empty}
1683   \begin{center}
1684     \large
1685     Erste~Staatsprüfung~für~ein~Lehramt~an~öffentlichen~Schulen \par
1686
1687     \vspace{0.5cm}
1688
1689     Fach~Informatik \par
1690
1691     \vfill
1692
1693     \liGrafikLogo[width=8cm]\par
1694     \bigskip
1695     Die~Bschlangaul-Sammlung \par
1696     {\footnotesize \liMetaHermineBschlangaulAndFriends} \par
1697
1698     \vfill
1699
1700     {
1701       \bfseries\Huge
1702
1703       \g_auf_examen_jahreszeit_tl \par
1704
1705       \g_auf_examen_jahr_tl \par
1706     }
1707
1708     \vspace{2cm}
1709
1710     {\LARGE \g_auf_examen_nummer_tl \par}
1711
1712     \vspace{0.5cm}
1713
1714     \g_auf_examen_fach_tl \par
1715
1716     \vspace{3cm}
1717
1718     Aufgabenstellungen~mit~Lösungsvorschlägen \par
1719
1720   \end{center}
1721
1722   \vfill
1723 }

1724 \cs_new:Npn \inhalts_verzeichnis: {
1725   % für den Abstand vor den section im Inhaltsverzeichnis
1726   % https://tex.stackexchange.com/questions/241445/how-to-control-spacing-in-toc-for-different-sections
1727   \setlength{\cftbeforesecskip}{1.5cm}
1728   \setlength{\cftbeforesubsecskip}{0.5cm}
1729   \renewcommand{\cftsubsecafterpnum}{\vspace{\cftbeforesubsecskip}}
1730   \setlength{\cftbeforesubsubsecskip}{0.1cm}
1731   \renewcommand{\contentsname}{Aufgabenübersicht}
1732   \tableofcontents
1733 }

1734 \setcounter{tocdepth}{4}
1735 \RequirePackage[titles]{tocloft}
1736 \AddToHook{begindocument}{
1737   \titel_seite:
1738
1739   \clearpage
1740
1741   \inhalts_verzeichnis:

```

```
1742
1743 \vfill
1744
1745 \liLogoTextProjekt
1746 \bigskip
1747
1748 \liLogoTextCCLizenz
1749 \bigskip
1750
1751 \clearpage
1752 }
1753 \ExplSyntaxOff
1754
```

## 2.21 komplexitaetstheorie.sty

```
1755 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1756 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1757 Setzen von Karp's NP-vollständigen Problemen, Hilfsmakros für die
1758 Polynomialzeitreduktion.]
```

### 2.21.1 Makro-Kürzel

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

```
1759 \liLadePakete{mathe}
```

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

```
1760 \RequirePackage{mdframed}
```

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

```
1761 \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`

```
1762 \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`

```
1763 \def\liProblemBeschreibung#1#2#3{
1764   \begin{mdframed}[
1765     userdefinedwidth=9cm,
1766     align=center,
1767     backgroundcolor=white!0,
1768   ]
1769   \centerline{\large\liProblemName{#1}}
1770
1771   \medskip
1772
1773   \begin{description}
1774     \item[Gegeben:] #2
1775     \item[Frage:] #3
1776   \end{description}
1777 \end{mdframed}
1778 }
```

```

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

1779 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1780 \begin{displaymath}
1781   \liProblemName{#1}
1782   \preceq_{#2}
1783   \liProblemName{#3}
1784 \end{displaymath}
1785 }

\liProblemVertexCover

1786 \def\liProblemClique{%
1787 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1788 Mindestgröße  $n$  in einem gegebenen Graphen.
1789 \footcite[Seite 76]{theo:fs:4}
1790 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1791 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1792 \footcite{wiki:cliquenproblem}
1793 }

\liProblemVertexCover

1794 \def\liProblemVertexCover{%
1795 %
1796 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1797 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1798 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1799 \footcite{wiki:knotenueberdeckung}
1800
1801 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1802 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1803 mindestens einem Knoten aus  $U$  verbunden ist.
1804 \footcite[Seite 78]{theo:fs:4}%
1805 }

\liProblemSubsetSum Kein Karp-21-Problem

1806 \def\liProblemSubsetSum{%
1807 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1808 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1809 \footcite{wiki:teilsommenproblem}
1810 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1811 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1812 größer als eine gegebene obere Schranke  $c$  ist.
1813 \footcite[Seite 74]{theo:fs:4}
1814 }

\liProblemSubsetSum Kein Karp-21-Problem

1815 \def\liProblemSat{%
1816 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1817 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1818 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1819 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1820 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1821 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1822 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1823 aufgestellt werden.
1824 \footcite[Seite 71]{theo:fs:4}
1825 }

1826

```

## 2.22 kontrollflussgraph.sty

1827 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1828 \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 = ...}]
```

```

1829 \RequirePackage{tikz}
1830 \usetikzlibrary{positioning}
1831 \tikzset{
1832   li kontrollfluss/.style={
1833     knoten/.style={
1834       circle,
1835       draw
1836     },
1837     usebox/.style={
1838       draw,
1839       rectangle,
1840       font=\scriptsize,
1841       anchor=west,
1842       align=left,
1843     },
1844     bedingung/.style={
1845       midway,
1846       draw=none,
1847       font=\scriptsize
1848     },
1849     knotenbeschriftung/.style={
1850       draw,
1851       rectangle,
1852       midway,
1853       font=\scriptsize
1854     },
1855     wahr/.style={
1856       thick
1857     },
1858     falsch/.style={
1859       dashed
1860     },
1861     every node/.style={
1862       circle,
1863       draw,
1864     },
1865     every edge/.append style={
1866       every node/.style={
1867         draw=none,
1868         bedingung,
1869       }
1870     },
1871     every path/.style={
1872       draw,
1873       ->,
1874     },
1875     every pin/.style={
1876       draw,
1877       dotted,
1878       rectangle,
1879       pin position=right
1880     },
1881     every pin edge/.style={
1882       dotted,
1883       arrows=-,
1884     }
1885   }
1886 }

```

#### 2.22.4 Umgebungen

liKontrollflussgraph

```

1887 \NewDocumentEnvironment { liKontrollflussgraph } { 0{} } {

```



```

1888 \begin{tikzpicture}[
1889     li kontrollfluss,
1890     #1
1891 ]
1892 } {
1893 \end{tikzpicture}
1894 }

```

## 2.22.5 Makros

`\liAnweisung`

```
1895 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}
```

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

```
1896 \def\liBedingung#1#2{\node[bedingung,#1]{\texttt{#2}}}
```

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

```
1897 \def\liBedingungWahr#1{\node[bedingung,#1]{\texttt{[true]}}}
```

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

```
1898 \def\liBedingungFalsch#1{\node[bedingung,#1]{\texttt{[false]}}}
```

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

```
1899 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}
```

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

```
1900 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw
```

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

```
1901 \ExplSyntaxOn
```

```
1902 \NewDocumentCommand { \liKontrollKnotenPfad } { m }
```

```
1903 {
```

```
1904   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
```

```
1905   \seq_set_map:NNn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
```

```
1906   \seq_use:Nn \l_tmpa_seq { ~~~~ }
```

```
1907 }
```

```
1908 \ExplSyntaxOff
```

```
1909
```

## 2.23 kopf-fusszeilen.sty

```
1910 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1911 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1912 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1913 \ExplSyntaxOn

1914 \fancyhead{}
1915 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1916 \fancyhead[L0,RE]{\scriptsize\today}
1917 \fancyfoot{}
1918 \fancyfoot[LE,R0]{\thepage}
1919 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1920 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1921 \renewcommand{\headrulewidth}{0.4pt}
1922 \renewcommand{\footrulewidth}{0.4pt}
1923 \setlength{\headwidth}{\textwidth}

1924 \ExplSyntaxOff

1925
```

## 2.24 literatur-dummy.sty

```
1926 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1927 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1928 \def\literatur{}

\footcite

1929 % \RequirePackage[stable,multiple]{footmisc}
1930 \NewDocumentCommand{ \footcite } { o m }{}

1931
```

## 2.25 literatur.sty

```
1932 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1933 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1934 \RequirePackage{csquotes}
1935 \RequirePackage[
1936   bibencoding=utf8,
1937   citestyle=authortitle,
1938   backend=biber,
1939 ]{biblatex}
1940 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1941 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1942 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1943 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1944 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1945 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1946 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1947 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1948 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1949 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1950 % To allow footnotes in the heading
1951 \RequirePackage[stable,multiple]{footmisc}

\literatur

1952 \def\literatur{\printbibliography[heading=subbibliography]}

1953
```

## 2.26 makros.sty

```
1954 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1955 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1956 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1957 anderen Paket passen]
1958 \RequirePackage{hyperref}
1959 \RequirePackage{graphicx}
    Für die Umgebung liQuellen benötigt.
1960 \RequirePackage{paralist}
1961 \ExplSyntaxOn

\inhaltsverzeichnis
1962 \def\inhaltsverzeichnis {
1963   \begin{mdframed}
1964     \begin{group}
1965       \let\clearpage\relax
1966       \tableofcontents
1967     \end{group}
1968   \end{mdframed}
1969 }

\memph \mephm (\marginpar and \emph)
1970 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1971 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1972 \newcommand{\liPseudoUeberschrift}[1]{
1973   \bigskip
1974
1975   \par
1976   \noindent
1977   \textbf{#1}
1978
1979   \medskip
1980
1981   \par
1982   % Keine Einrückung
1983   \@afterindentfalse
1984   \@afterheading
1985 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1986 \newcommand{\liBeschriftung}[1]{
1987   \par
1988   \noindent
1989   \medskip
1990   \textbf{#1}:
1991   \medskip
1992   \noindent
1993 }

\hinweis
1994 \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.
1995 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}
```

liEinbettung

```
1996 \NewDocumentEnvironment { liEinbettung }{ o +b } {#2} {}
```

### 2.26.1 Umgebungen, die Inhalte aus- und einblenden können

Die einfachste Möglichkeit, um den kompletten Inhalt einer Umgebung auszublenden ist die Verwendung `+b` in einer `xparse` erzeugten Umgebung. Manchmal funktioniert diese Methode nicht. In der Dokumentation von `xparse` steht, dass dieses Feature etwas experimentell ist.

Eine andere Methode verwendet das `exam`-Package. Die Inhalt wird in eine Box verschoben, die dann einfach ignoriert wird.

```
\NewDocumentEnvironment{ liAdditum } { o }
{
  \ifLiADDITUM
  \else
    % Alles in eine Box verschieben und die dann ignorieren.
    \setbox 0 \vbox
    \bgroup
    \fi

    \begin{frame}
  } {
    \end{frame}

    \ifLiADDITUM
    \else
      \egroup
    \fi
}
```

**liAntwort** Zum Setzen von Antworten. Sie werden mit einem Kasten umgeben. Könnten eventuell ausgeblendet werden, wenn man nur die Lösung sehen will.

```
1997 \NewDocumentEnvironment{ liAntwort } { O{standard} }
1998 {
1999   \ifLiANTWORT
2000   \else
2001     \setbox 0 \vbox
2002     \bgroup
2003     \fi
2004
2005   \str_case:nn {#1} {
2006     {standard} {
2007       \def\beschriftung{}
2008       \mdfsetup{backgroundcolor=gray!1,linecolor=gray}
2009     }
2010     {richtig} {
2011       \def\beschriftung{richtig}
2012       \mdfsetup{backgroundcolor=green!5,linecolor=green}
2013     }
2014     {falsch} {
2015       \def\beschriftung{falsch}
2016       \mdfsetup{backgroundcolor=red!3,linecolor=red}
2017     }
2018     {muster} {
2019       \def\beschriftung{Musterlösung}
2020       \mdfsetup{backgroundcolor=green!12,linecolor=green}
2021     }
2022   }
2023   \ifx\beschriftung\empty\else
2024     \noindent
2025     \textbf{\beschriftung{}}
  }
```

```

2026 \fi
2027 \begin{mdframed}
2028 }
2029 {
2030 \end{mdframed}
2031 \ifLiANTWORT
2032 \else
2033 \egroup
2034 \fi
2035 }

```

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

```

2036 \NewDocumentEnvironment{ liAdditum } { o }
2037 {
2038 \ifLiADDITUM
2039 \else
2040 \setbox 0 \vbox
2041 \bgroup
2042 \fi
2043
2044 \begin{mdframed}[backgroundcolor=yellow!5]
2045 \IfNoValueTF {#1}
2046 {
2047 \liPseudoUeberschrift{Additum}
2048 }
2049 {
2050 \liPseudoUeberschrift{Additum:~#1}
2051 }
2052 }
2053 {
2054 \end{mdframed}
2055
2056 \ifLiADDITUM
2057 \else
2058 \egroup
2059 \fi
2060 }

```

**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.

```

2061 \NewDocumentEnvironment{ liExkurs }{o +b}{
2062 \ifLiEXKURS
2063 \vspace{0.2cm}%
2064 \begin{mdframed}[
2065 backgroundcolor=white,
2066 bottomline=false,
2067 innermargin=1cm,
2068 leftline=true,
2069 linecolor=black,
2070 linewidth=0.1cm,
2071 outermargin=1cm,
2072 rightline=false,
2073 topline=false,

```

```

2074 ]
2075 \footnotesize
2076 \noindent%
2077 \textbf{Exkurs:~#1}\par%
2078 \noindent%
2079 #2
2080 \end{mdframed}
2081 \vspace{0.2cm}
2082 \else
2083 \fi
2084 }{}

```

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

```

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

```

**Weiterführende Literatur:**

- Quelle 1
- Quelle 2

```

2085 \cs_new:Npn \listen@punkt #1 {\item #1}
2086 \NewDocumentEnvironment { liQuellen }{ +b }
2087 {
2088   \seq_clear_new:N \l_quellen
2089   \seq_set_split:Nnn \l_quellen {\item} {#1}
2090   \seq_remove_all:Nn \l_quellen {}
2091   \begin{mdframed}[backgroundcolor=white!0]
2092     \footnotesize
2093     \noindent
2094     \textsf{\textbf{Weiterführende~Literatur:}}
2095     \medskip
2096     \begin{compactitem}
2097       \seq_map_function:NN \l_quellen {\listen@punkt}
2098     \end{compactitem}
2099   \end{mdframed}
2100   %
2101   \par
2102   \@afterindentfalse
2103   \@afterheading
2104 } {}

```

liLernkartei

```

2105 \NewDocumentEnvironment { liLernkartei }{ m +b }
2106 {
2107   \begin{mdframed}
2108     \footnotesize
2109     \noindent%
2110     \textbf{Lernkarteikarte:~#1}\par%
2111     \noindent%
2112     #2
2113   \end{mdframed}
2114 } {}

```

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

```

2115 \NewDocumentEnvironment { liDiagramm }{ m +b }
2116 {
2117   \begin{mdframed}[backgroundcolor=white!0]
2118     \small

```



```

2119 \noindent%
2120 \textit{#1}:
2121 \begin{center}
2122 #2
2123 \medskip
2124 \end{center}
2125 \end{mdframed}
2126 } {}

\liFussnoteUrl \liFussnoteUrl[zusätzlicher-text]{url} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
2127 \NewDocumentCommand{\liFussnoteUrl} { o m } {
2128 \footnote{\url{#2}\IfNoValueTF{#1}{}{ (#1)}}
2129 }
2130

\liFussnoteLink \liFussnoteLink[zusätzlicher-text]{link-text}{url} \liFussnoteLink[zusätzlicher Text]{tex
Einen Link, d. h. einen Link-Text und eine URL als Fußnote setzen.
2131 \NewDocumentCommand{\liFussnoteLink} { o m m } {
2132 \footnote{\href{#3}{#2}\IfNoValueTF{#1}{}{ (#1)}}
2133 }

\zB
2134 \def\zB{z.\,B. }

\ZB
2135 \def\ZB{Z.\,B. }

\dh
2136 \def\dh{d.\,h. }

2137 \ExplSyntaxOff
2138

```

## 2.27 master-theorem.sty

2139 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2140 \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}

2141 \ExplSyntaxOn

2142 \RequirePackage{amsmath}

\liRundeKlammer

2143 \def\liRundeKlammer#1{

2144 \negthinspace \left( #1 \right)

2145 }

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

2146 \def\liThetaOhneMathe#1{

2147 \Theta \liRundeKlammer{#1}

2148 }

2149 \def\liTheta#1{

2150 \ifmmode

2151 \liThetaOhneMathe{#1}

2152 \else

2153 \$\liThetaOhneMathe{#1}\$

2154 \fi

2155 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
2156 \def\liOmegaOhneMathe#1{
2157 \Omega \liRundeKlammer{#1}
2158 }
2159 \def\liOmega#1{
2160 \ifmmode
2161 \liOmegaOhneMathe{#1}
2162 \else
2163 $\liOmegaOhneMathe{#1}$
2164 \fi
2165 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
2166 \def\liOOhneMathe#1{
2167 \mathcal{O} \liRundeKlammer{#1}
2168 }
2169 \def\liO#1{
2170 \ifmmode
2171 \liOOhneMathe{#1}
2172 \else
2173 $\liOOhneMathe{#1}$
2174 \fi
2175 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
2176 \def\liTOhneMathe#1#2{
2177 \tl_if_blank:nTF {#1}
2178 {}
2179 {#1 \cdot }
2180 T
2181 \liRundeKlammer{\frac{n}{#2}}
2182 }
2183 \def\liT#1#2{
2184 \ifmmode
2185 \liTOhneMathe{#1}{#2}
2186 \else
2187 $\liTOhneMathe{#1}{#2}$
2188 \fi
2189 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
2190 \def\liRekursionsGleichung{
2191 $T(n) = \liT{a}{b} + f(n)$
2192 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
2193 \def\liBedingungEins{
2194 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
2195 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
2196 \def\liBedingungZwei{
2197 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
2198 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
2199 \def\liBedingungDrei{
2200 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
2201 }
2202 \ExplSyntaxOff

```

\liMasterVariablen

```

2203 \def\liMasterVariablen{
2204   \begin{displaymath}
2205     T(n) = \liT{a}{b} + f(n)
2206   \end{displaymath}
2207
2208   \begin{itemize}
2209     \item[$a = $]
2210       Anzahl der rekursiven Aufrufe, Anzahl der Unterprobleme in der
2211       Rekursion
2212       ($a \geq 1$).
2213
2214     \item[$\textstyle{\frac{1}{b}} = $]
2215       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
2216       repräsentiert wird, Anteil an der Verkleinerung des Problems ($b > 1$).
2217
2218     \item[$f(n) = $]
2219       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
2220       die Kombination der Teillösungen entstehen. Eine von $T(n)$
2221       unabhängige und nicht negative Funktion.
2222   \end{itemize}
2223   \footcite{wiki:master-theorem}
2224   \footcite[Seite 19-35]{aud:fs:2}
2225 }
```

\liMasterFaelle

```

2226 \def\liMasterFaelle{
2227   \begin{description}
2228     \item[1. Fall:]
2229       $T(n) \in \liTheta{n^{\log\sb{b}a}}$
2230
2231       \hfill falls \liBedingungEins
2232       für $\varepsilon > 0$
2233
2234     \item[2. Fall:]
2235       $T(n) \in \liTheta{n^{\log\sb{b}a} \cdot \log n}$
2236
2237       \hfill falls \liBedingungZwei
2238
2239     \item[3. Fall:]
2240       $T(n) \in \liTheta{f(n)}$
2241
2242       \hfill falls \liBedingungDrei
2243       für $\varepsilon > 0$
2244       und ebenfalls für ein $c$ mit $0 < c < 1$ und alle hinreichend großen $n$
2245       gilt:
2246       $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$
2247   \end{description}
2248 }
```

\liMasterVariablenDeklaration

```

2249 \def\liMasterVariablenDeklaration#1#2#3{
2250   \begin{description}
2251     \item[Allgemeine Rekursionsgleichung:] \strut
2252
2253     \liRekursionsGleichung
2254
2255     \item[Anzahl der rekursiven Aufrufe ($a$):] \strut
2256
2257     #1
2258
2259     \item[Anteil Verkleinerung des Problems ($b$):] \strut
2260   \end{description}

```

```

2261     um  $\frac{1}{b}$  also  $b = \frac{1}{a}$ 
2262
2263     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut
2264
2265      $T(n) = \frac{1}{b} + T(\frac{n}{b})$ 
2266
2267     \item[Ergibt folgende Rekursionsgleichung:] \strut
2268
2269      $T(n) = \frac{1}{b} + T(\frac{n}{b})$ 
2270 \end{description}
2271 }

\liMasterFallRechnung
2272 \def\liMasterFallRechnung#1#2#3{
2273   \begin{description}
2274     \item[1. Fall:] \liBedingungEins:
2275
2276     #1
2277
2278     \item[2. Fall:] \liBedingungZwei:
2279
2280     #2
2281
2282     \item[3. Fall:] \liBedingungDrei:
2283
2284     #3
2285   \end{description}
2286 }

\liMasterExkurs
2287 \def\liMasterExkurs{
2288   \begin{liExkurs}[Master-Theorem]
2289     \liMasterVariablen
2290
2291     \noindent
2292     Dann gilt:
2293
2294     \liMasterFaelle
2295   \end{liExkurs}
2296 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
2297 \def\liMasterWolframLink#1{
2298   Berechne die Rekursionsgleichung auf WolframAlpha:
2299   \href{https://www.wolframalpha.com/input/?i=#1}{WolframAlpha}
2300 }

2301

```

## 2.28 mathe.sty

```
2302 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2303 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
2304
2305 % for example \ltimes \rtimes
2306 %\RequirePackage{amssymb}
2307 \RequirePackage{amsmath}
2308
2309 %%
2310 % \mlq \mrq
2311 %%
2312 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
2313 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
2314
```

## 2.29 meta.sty

```
2315 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2316 \ProvidesPackage{lehramt-informatik-klassen-meta}[2021/09/10 Sammlung
2317 von Textschnipseln, die das Projekt beschreiben]
```

```
2318 \ExplSyntaxOn
```

```
2319 \liLadePakete{grafik}
```

### 2.29.1 Einfache Makros (Low level)

```
\liMetaBschlangaulSammlung
```

```
2320 \def\liMetaBschlangaulSammlung{
2321   Die~Bschlangaul-Sammlung
2322 }
```

```
HermineBschlangaulAndFriends
```

```
2323 \def\liMetaHermineBschlangaulAndFriends{
2324   Hermine~Bschlangaul~and~Friends
2325 }
```

```
\liMetaUeberDasProjekt
```

```
2326 \def\liMetaUeberDasProjekt{
2327   Eine~freie~Aufgabensammlung~mit~Lösungen~
2328   von~Studierenden~für~Studierende~
2329   zur~Vorbereitung~auf~die~1.~Staatsexamensprüfungen~
2330   des~Lehramts~Informatik~in~Bayern.
2331 }
```

```
\liMetaCCLink
```

```
2332 \def\liMetaCCLink{
2333   Diese~Materialsammlung~unterliegt~den~Bestimmungen~der~
2334   \href{
2335     https://creativecommons.org/licenses/by-nc-sa/4.0/deed.de
2336   }
2337   {
2338     Creative~Commons~Namensnennung~Nicht-kommerziell~Share~Alike~4.0~
2339     International-Lizenz
2340   }.
2341 }
```

```
\liMetaEmailLink
```

```
2342 \def\liMetaEmailLink{
2343   \href{
2344     mailto:hermine.bschlangaul@gmx.net
2345   }{
2346     hermine.bschlangaul@gmx.net
2347   }
2348 }
```

```
\liMetaHilfMit
```

```
2349 \def\liMetaHilfMit{
2350   Hilf~mit!~
2351
2352   Die~Hermine~schafft~das~nicht~alleine!~
2353
2354   Das~ist~ein~Community-Projekt.~
2355
2356   Verbesserungsvorschläge,~Fehlerkorrekturen,~weitere~Lösungen~sind~
2357   herzlich~willkommen~~~egal~wie~~~per~Pull-Request~oder~per~E-Mail~an~
2358   \liMetaEmailLink.
2359 }
```

`\liMetaHilfMit`

```
2360 \def\liMetaQuelltext{
2361   Der~\TeX-Quelltext~dieses~Dokuments~kann~unter~folgender~
2362   URL~aufgerufen~werden:~
2363 }
```

## 2.29.2 Zusammengesetzte Makros (High level)

Plaziert zwei Minipages nebeneinander. Die erste Umgebung ist für eine Logo gedacht, die zweite für einen Text

```
2364 \cs_new:Npn \logo_dann_text:nn #1 #2 {
2365   \begin{center}
2366     \begin{minipage}[c]{5.5cm}
2367       #1
2368     \end{minipage}
2369
2370     \begin{minipage}[c]{10cm}
2371       #2
2372     \end{minipage}
2373   \end{center}
2374 }
```

`\liLogoTextProjekt`

```
2375 \def\liLogoTextProjekt
2376 {
2377   \logo_dann_text:nn
2378   {
2379     \liGrafikLogo[width=5cm]
2380   }
2381   {
2382     {
2383       \bfseries
2384       \liMetaBschlangaulSammlung
2385     }
2386     \par
2387
2388     \liMetaHermineBschlangaulAndFriends
2389     \par
2390
2391     \medskip
2392
2393     \begin{spacing}{1}
2394       \footnotesize
2395       \liMetaUeberDasProjekt
2396     \end{spacing}
2397   }
2398 }
```

`\liLogoTextCCLizenz`

```
2399 \def\liLogoTextCCLizenz
2400 {
2401   \logo_dann_text:nn
2402   {
2403     \centerline{\liGrafikCCLizenz[width=3cm]}
2404   }
2405   {
2406     \begin{spacing}{1}
2407       \scriptsize
2408       \liMetaCCLink
2409     \end{spacing}
2410   }
2411 }
```



2412 \ExplSyntaxOff

2413

## 2.30 minimierung.sty

```

2414 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2415 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
2416 Minimierungsalgorithmus von einem NEA zu einem DEA]

2417 \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

2418 \def\liFussnote#1{\$x_{#1}\$}

2419 \def\li@fussnote@text#1#2{
2420 \liFussnote{#1}
2421 \quad
2422 {\footnotesize #2}
2423 }

\liFussnoteEinsText
2424 \def\liFussnoteEinsText{
2425 \li@fussnote@text{1}
2426 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
2427 }

\liFussnoteZweiText
2428 \def\liFussnoteZweiText{
2429 \li@fussnote@text{2}
2430 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
2431 }

\liFussnoteDreiText
2432 \def\liFussnoteDreiText{
2433 \li@fussnote@text{3}

```

```

2434 {In weiteren Iterationen markierte Zustände.}
2435 }

\liFussnoteVierText
2436 \def\liFussnoteVierText{
2437   \li@fussnote@text{4}
2438   {...}
2439 }

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



2440 \def\liFussnoten{
2441   \bigskip
2442
2443   \noindent
2444   \liFussnoteEinsText
2445
2446   \noindent
2447   \liFussnoteZweiText
2448
2449   \noindent
2450   \liFussnoteDreiText
2451
2452   \noindent
2453   \liFussnoteVierText
2454 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
2455 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
2456 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
2457 \def\liZustandsPaar#1#2{
2458   $(
2459     \liZustandsPaarVariablenName_#1,
2460     \liZustandsPaarVariablenName_#2
2461   )$
2462 }

liUebergangsTabelle
2463 \renewcommand{\arraystretch}{1.4}
2464 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
2465   \liPseudoUeberschrift{Übergangstabelle}
2466   \begin{center}
2467     \begin{tabular}{r|l|l}
2468       \textbf{Zustandspaar} & \textbf{\#1} & \textbf{\#2} \\ \hline
2469     \end{tabular}
2470   \end{center}
2471 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle:

Minimierungstabelle (Table filling)
2473 \ExplSyntaxOn

```

```

2474 \def\liUeberschriftDreiecksTabelle{
2475   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
2476 }

```

**\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. —

```

2477 \def\liMinimierungErklaerung{
2478   %\footcite[Seite~19]{koenig}
2479   \liParagraphMitLinien{
2480     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
2481     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
2482     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
2483     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
2484      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
2485     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
2486     somit~zueinander~( $k - 1$ )-äquivalent,~aber~nicht~ $k$ -äquivalent~und~
2487     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
2488     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
2489   }
2490 }
2491 \ExplSyntaxOff

```

2492

## 2.31 normalformen.sty

```

2493 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2494 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10
2495 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
2496 Attributhülle]

```

Mathe für textit tex etc, typographie für geschweifte Klammer

```

2497 \liLadePakete{mathe,typographie}
2498 \directlua{
2499   helfer = require('lehramt-informatik-helfer')
2500   normalformen = require('lehramt-informatik-normalformen')
2501 }

```

### 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

```

```

2502 \def\liTeilen#1{
2503   \directlua{tex.print(normalformen.teilen('#1'))}
2504 }

```

```

\liAttributHuelle Let-Abkürzung: \let\ah=\liAttributHuelle
                  \ah{F, \m{A, B}} AttrHülle( $F, \{A, B\}$ ) Regulärer Ausdruck zum Konvertieren
                  AttrHülle((.*)\ ) \ah{$1}
2505 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(\#1)}
2506 \def\liAttributHuelle#1{
2507   \ifmmode
2508     \liAttributHuelleOhneMathe{\#1}
2509   \else
2510     $\liAttributHuelleOhneMathe{\#1}$
2511   \fi
2512 }

```

```

\liAttributMenge Let-Abkürzung: \let\m=\liAttributMenge
2513 \def\liAttributMenge#1{\{ \textit{\#1} \}}

```

liAHuelle

```

2514 \NewDocumentEnvironment{ liAHuelle } { +b } {
2515   \begingroup
2516   \footnotesize
2517   \begin{multline*}
2518     \#1
2519   \end{multline*}
2520   \endgroup
2521 } { }

```

```

\liLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multline
Let-Abkürzung: \let\ahL=\liLinksReduktion
                  \ahL{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}
2522 \def\liLinksReduktion#1#2#3{
2523   \shoveleft{
2524     \liAttributHuelleOhneMathe{FA,
2525       \liAttributMenge{\#1 \string\ #2}} =
2526   } \\\

```

```

2527 \shoveright{
2528   \liAttributMenge{#3}
2529 } \\\
2530 }

```

```

\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( $F, \{A, E \setminus E\}$ ) =  $\{A, E, F, B, D\}$ 
2531 \def\liLinksReduktionInline#1#2#3{%
2532   {%
2533     \footnotesize%
2534     $\liAttributHuelleOhneMathe{F,
2535       \liAttributMenge{#1} \string\ #2}} =
2536       \liAttributMenge{#3}$
2537   }
2538 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahr=\liLinksReduktionInline
    \ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
2539 \def\liRechtsReduktionInline#1#2#3#4{%
2540   {%
2541     \footnotesize%
2542     $\liAttributHuelleOhneMathe{
2543       F \setminus
2544       \liFunktionaleAbhaengigkeit{#1}
2545       \def\tmp{#2}\ifx\tmp\empty
2546       \else
2547         \cup \liFunktionaleAbhaengigkeit{#2}
2548       \fi
2549       ,
2550       \liAttributMenge{#3}
2551     } =
2552     \liAttributMenge{#4}$
2553   }
2554 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
    \fa{A, B -> C, D}:  $\{A, B\} \rightarrow \{C, D\}$  Regulärer Ausdruck zum Konvertieren:
    \$(.*?) \\\rightarrow (.*)\$ \fa{$1 -> $2}
2555 \def\liFunktionaleAbhaengigkeit#1{%
2556   \directlua{normalformen.drucke_funk_abhaengigkeit('#1', false)}%
2557 }

```

```

\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;

```

```

2558 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
2559   \liGeschweifteKlammern
2560   {#1}
2561   {
2562     \begin{align*}
2563       \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
2564     \end{align*}
2565   }
2566   {-0.5cm}
2567   {-1.7cm}
2568 }

\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}

2569 \NewDocumentCommand {\liRelation} { O{R} m } {
2570   $\directlua{
2571     local name = helper.konvertiere_tiefgestellt('#1')
2572     tex.print(name)
2573   }$(\textit{\, #2\,})
2574 }

2575

```

## 2.32 o-notation.sty

2576 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2577 \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}
```

2578 \ExplSyntaxOn

2579 \RequirePackage{amssymb}

2580 \RequirePackage{pgfplots}

Für echte Teilmenge \subsetneq:  $\subsetneq$

2581 \RequirePackage{amssymb}

\liRundeKlammer

```
2582 \def\liRundeKlammer#1{
2583   \negthinspace \left( #1 \right)
2584 }
```

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

```
2585 \cs_new:Npn \o_notation_O:n #1 {
2586   \mathcal{O} \liRundeKlammer{#1}
2587 }
2588 \def\liONotationO#1{
2589   \ifmmode
2590     \o_notation_O:n { #1 }
2591   \else
2592     $ \o_notation_O:n { #1 } $
2593   \fi
2594 }
```

2595



## 2.33 petri.sty

2596 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2597 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]

### 2.33.1 Makro-Kürzel

\let\t=\liPetriTransitionsName

\let\tp=\liPetriTransPfeile

\let\k=\liPetriErreichKnotenDrei

2598 \RequirePackage{tikz}

2599 \usetikzlibrary{petri,arrows.meta}

Für die Darstellungsmatrix

2600 \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

```
2601 \def\liPetriSetzeSchluessel{%
2602   \def\TmpTransitionOne{}%
2603   \def\TmpTransitionTwo{}%
2604   \def\TmpTransitionThree{}%
2605   \def\TmpTransitionFour{}%
2606   \def\TmpTransitionFive{}%
2607   \def\TmpTransitionSix{}%
2608   \def\TmpTransitionSeven{}%
2609   \def\TmpTransitionEight{}%
2610   \def\TmpTransitionNine{}%
2611   \def\TmpTransitionTen{}%
2612   \pgfkeys{/petri/.cd,
2613     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2614     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2615     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2616     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
```

```

2617 p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
2618 p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2619 p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2620 p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2621 p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2622 p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2623 t1/.store in=\TmpTransitionOne,t1/.default=activated,
2624 t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2625 t3/.store in=\TmpTransitionThree,t3/.default=activated,
2626 t4/.store in=\TmpTransitionFour,t4/.default=activated,
2627 t5/.store in=\TmpTransitionFive,t5/.default=activated,
2628 t6/.store in=\TmpTransitionSix,t6/.default=activated,
2629 t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2630 t8/.store in=\TmpTransitionEight,t8/.default=activated,
2631 t9/.store in=\TmpTransitionNine,t9/.default=activated,
2632 t10/.store in=\TmpTransitionTen,t10/.default=activated,
2633 scale/.store in=\TmpScale,scale/.default=0.5,
2634 x/.store in=\TmpX,x/.default=5,
2635 y/.store in=\TmpY,y/.default=5,
2636 }%
2637 }

```

```

2638 \tikzset{
2639   li petri/.style={
2640     activated/.style={
2641       very thick
2642     },
2643     inhibitor/.style={
2644       {Circle[open,length=2mm,fill=white]}-
2645     }
2646   }
2647 }

```

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

```

2648 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2649 \def\liPetriTransitionsName#1{
2650   \ifmmode
2651     \liPetriTransitionsNameOhneMathe{#1}
2652   \else
2653     $\liPetriTransitionsNameOhneMathe{#1}$
2654   \fi
2655 }

```

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

```

2656 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2657   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2658 }

```

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

```

2659 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2660 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2661 }

```

## 2.34 potenzmengen-konstruktion.sty

```
2662 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2663 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2664 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2665 \liLadePakete{formale-sprachen}
2666 \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}
    }
  }
}
```

```
2667 \def\liZustandsMengenSammlung#1#2{
2668   \liZustandsnameGross{#1}
2669   {
2670     \footnotesize
2671     \liPotenzmenge{
2672       \str_case:nn {#1} {#2
2673       }
2674     }
2675 }
```

```
\liZustandsMengenSammlungNr
```

```
2676 \def\liZustandsMengenSammlungNr#1#2{
2677   \liZustandsnameGross{#1}
2678   {
```

```
2679     \footnotesize
2680     \liZustandsmengeNr{
2681         \str_case:nn {#1} #2
2682     }
2683 }
2684 }

2685 \ExplSyntaxOff
2686
```

## 2.35 pseudo.sty

2687 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2688 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen  
 2689 von Pseudo-Code, Hüll-Paket um algorithm2e]

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

Algorithmus 1: Minimaler Spannbaum nach Kruskal
---

<p><b>Data:</b> <math>G = (V, E, w)</math>: ein zusammenhängender, ungerichteter, kantengewichteter Graph kruskal(<math>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;"> <p>wähle eine Kante <math>e \in L</math> mit kleinstem Kantengewicht;</p> <p>entferne die Kante <math>e</math> aus <math>L</math>;</p> <p><b>if</b> der Graph <math>(V, E' \cup \{e\})</math> keinen Kreis enthält <b>then</b></p> <div style="margin-left: 20px;"> <p><math>E' \leftarrow E' \cup \{e\}</math>;</p> </div> <p><b>end</b></p> </div> <p><b>end</b></p> <p><b>Result:</b> <math>M = (V, E')</math> ist ein minimaler Spannbaum von <math>G</math>.</p>
---

2690 \RequirePackage[german,boxruled]{algorithm2e}

2691

## 2.36 pumping-lemma.sty

2692 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2693 \ProvidesPackage{lehrant-informatik-pumping-lemma}[2021/08/11 Enthält die  
 2694 Definitionen für das Pumping-Lemma in der Regulären Sprache und  
 2695 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2696 \def\liPumpingRegulaer{%
2697   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2698   alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2699    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2700   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2701
2702   \begin{enumerate}
2703     \item  $|v| \geq 1$ 
2704     (Das Wort  $v$  ist nicht leer.)
2705
2706     \item  $|uv| \leq j$ 
2707     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2708
2709     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w$  in  $L$ 
2710     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2711     Sprache  $L$ )
2712   \end{enumerate}
2713
2714   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2715   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2716 }
```

\liPumpingKontextfrei

```
2717 \def\liPumpingKontextfrei{%
2718   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2719   sich alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  zerlegen lassen in
2720    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2721
2722   \begin{enumerate}
2723     \item  $|vx| \geq 1$ 
2724     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2725
2726     \item  $|vwx| \leq j$ 
2727     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2728
2729     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy$  in  $L$  (Für jede
2730     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2731     Sprache  $L$ )
2732   \end{enumerate}
2733 }
2734
```

## 2.37 quicksort.sty

```

2735 % https://tex.stackexchange.com/a/142634
2736 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2737 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2738
2739 %-----
2740 % USAGE:
2741 % \QSinitialize{comma, separated, numerical, values}
2742 % \loop
2743 % \QSpivotStep
2744 % \ifnum\value{pivotcount}>0
2745 %   \QSSortStep
2746 % \repeat
2747 %-----
2748
2749 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2750 \RequirePackage{xintfrac, xinttools}
2751
2752 \RequirePackage{tikz}
2753
2754 %-----
2755 % FIRST PART: TikZ styles and macros for the actual drawing
2756 \newcounter{cellcount}% used for coordinates of the node
2757 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2758
2759 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2760 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2761 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2762 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2763 % by police of LaTeX good conduct ? )
2764 \tikzset{l/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2765          o/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2766          r/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2767 % this is the "b" style as used in the image below
2768          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2769 % nicer:
2770          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2771          g/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2772
2773 % NOTE the b style was originally the same as the r(aised) style apart from
2774 % not being raised, but I find it nicer with a somewhat different
2775 % specification. I have not updated the images though.
2776
2777 % How the nodes are drawn depending on whether on the left of the pivot value
2778 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2779
2780 \def\DecoLEFT #1{%
2781   \xintFor* ##1 in {#1} \do
2782     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2783 }
2784
2785 \def\DecoINERT #1{%
2786   \xintFor* ##1 in {#1} \do
2787     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2788 }
2789
2790 \def\DecoRIGHT #1{%
2791   \xintFor* ##1 in {#1} \do
2792     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2793 }
2794
2795 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2796   \xintFor* ##1 in {#1} \do

```

```

2797     {\stepcounter{cellcount}}%
2798     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2799 }
2800
2801 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2802     \xintFor* ##1 in {#1} \do
2803     {\stepcounter{cellcount}}%
2804     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2805 }
2806
2807 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2808     \xintFor* ##1 in {#1} \do
2809     {\stepcounter{cellcount}}%
2810     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2811 }
2812
2813 %-----
2814 % SECOND PART: the actual sorting routines.
2815
2816 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2817 \def\QS@sort@b #1{\ifcase #1
2818     \expandafter\QS@sort@empty
2819     \or\expandafter\QS@sort@single
2820     \else\expandafter\QS@sort@c
2821     \fi
2822 }%
2823 \def\QS@sort@empty #1{}
2824 \def\QS@sort@single #1{\QS@Ir {#1}}
2825
2826 % This step is to pick the last as pivot.
2827 \def\QS@sort@c #1%
2828     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2829
2830 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2831 % The trick with \xintApplyUnbraced is that for example when selecting
2832 % the elements smaller than pivot, if we had been using \xintApply we
2833 % would have had at the minimum an empty brace pair. Thus we use the
2834 % "unbraced" variant, but then the \QS@select@smaller has added in
2835 % anticipation a level of braces.
2836 \def\QS@sort@d #1#2{%
2837     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2838     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2839     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2840 }%
2841 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{#{#2}}{ }}% space will stop a f-
    expansion
2842 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{#{#2}}{ }}% space will stop a f-
    expansion
2843 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{#{#2}}{ }}% space will stop a f-
    expansion
2844
2845 %
2846 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2847 %
2848 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2849 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2850 % latter must handle correctly an empty argument.
2851
2852 %-----
2853 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QSinitialize.
2854
2855 % This draws all with suitable highlighting for the newly chosen pivots
2856 % (which will be shown raised)

```



```

2857 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2858             \let\QSIr\DecoINERT
2859             \let\QSIrr\DecoINERT
2860             \let\QSRr\DecoRIGHTwithPivot
2861 \par\centerline{\rule[1.5mm]{0pt}{8mm}%
2862             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2863             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2864 }
2865
2866 % This sorts and then draws, showing where the pivot chosen in the previous
2867 % step go. Next time they will have become "inert". If pivotcount is still at
2868 % zero on exit from \QSpivotStep, then this is the signal to stop before
2869 % executing \QSsortStep.
2870 \def\QSsortStep {\def\QSLr {\noexpand\QS@sort@a}%
2871             \def\QSRr {\noexpand\QS@sort@a}%
2872             \def\QSIr {\noexpand\QSIrr}%
2873             \let\QSIrr\relax
2874             \edef\QS@list{\QS@list}%
2875             \let\QSLr\relax
2876             \let\QSRr\relax
2877             \let\QSIr\relax
2878             \edef\QS@list{\QS@list}%
2879             \let\QSLr\DecoLEFT
2880             \let\QSIr\DecoINERTwithPivot
2881             \let\QSIrr\DecoINERT
2882             \let\QSRr\DecoRIGHT
2883 \par\centerline{\rule[1.5mm]{0pt}{8mm}%
2884             \setcounter{cellcount}{0}%
2885             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2886 }
2887
2888 \def\QSinitialize #1{%
2889     % first, we convert the comma separated values into a list of braced items
2890     % we use an \edef, and anyhow many \edef's will be used later
2891     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2892     \let\QSRr\DecoRIGHT
2893     % The \QSRr marker mutated to draw the last element as
2894     % pivot and the earlier ones with the suitable style.
2895     %
2896     % The list of marked braced items \QS@list is used both for drawing
2897     % (as here) and for doing the exchange of elements during sort.
2898     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2899             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2900 }
2901

```

## 2.38 relationale-algebra.sty

2902 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2903 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]

2904 \RequirePackage{amsmath}

2905 \RequirePackage{amssymb}

Zum Zeichnen von Operatorenbäumen verwenden wir TikZ

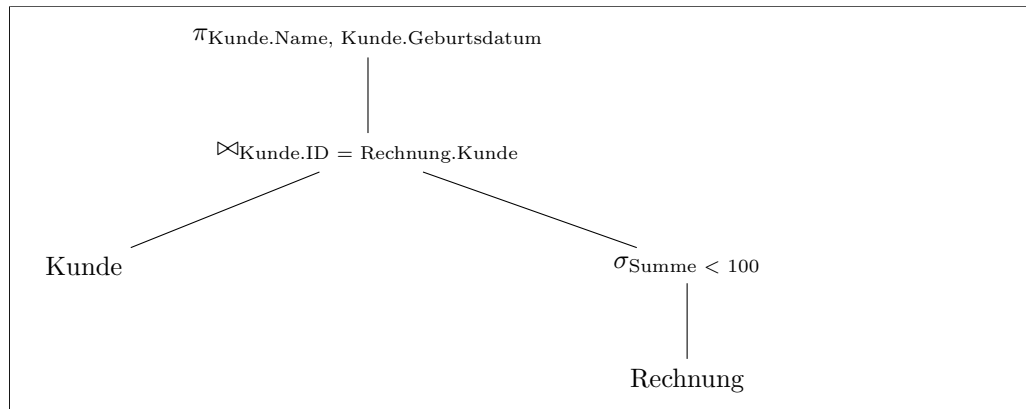
```
\begin{tikzpicture}
  \node
    (pi) {\pi_{\text{Kunde.Name, Kunde.Geburtsdatum}}};

  \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}
```



2906 \RequirePackage{tikz}

2907 \usetikzlibrary{positioning}

Privates Makros, das zwei Querstriche erzeugt.

```
2908 \def\o@join{\setbox0=\hbox{\bowtie}}%
2909 \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2910 }
```

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

```
2911 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}
```

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

```
2912 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}
```

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

```
2913 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
```

2914

## 2.39 rmodell.sty

```
2915 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2916 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2917 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2918 Datenbanken.]
2919 \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
2920 \def\liPrimaer#1{\ul{#1}}
```

```
\liFremd \liFremd{text}: Überstreichung für den Fremdschlüssel
2921 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

```
liRmodell \begin{liRmodell} \end{liRmodell}: Kleinere Schrift und Schreibmaschinenschrift.
2922 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2923 \ExplSyntaxOn
2924 \NewDocumentEnvironment { liRmodell }
2925 { +b }
2926 {
2927   \medskip
2928   {
2929     \linespread{2}
2930     \setlength{\parindent}{0pt}
2931     \li@Rmodell@Schrift#1
2932   }
2933   \medskip
2934 } {}
2935 \ExplSyntaxOff
```

```
\liRelationMenge Let-Abkürzung: \let\r=\liRelationMenge
\liRelationMenge{name}{attribut, attribut}: Umhüllen der Attribute mit geschweiften
und dann eckigen Klammern.
2936 \def\liRelationMenge#1#2{
2937 \noindent
2938 #1 : \{ [ #2 ] \}
2939 \par
2940 }
```

```
\liAttribut Let-Abkürzung: \let\a=\liAttribut
\liAttribut{text}: Gleiche Schrift wie Umgebung liRmodell
2941 \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}
```

```
2942 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2943
```

## 2.40 sortieren.sty

```
2944 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2945 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
2946 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}
}
```



```
2947 \RequirePackage{tikz}
2948 \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.

```
2949 \def\liVertauschen#1{
2950   \directlua{
2951     local sortieren = require('lehramt-informatik-sortieren')
2952     sortieren('#1')
2953   }
2954 }
```

`\liSortierPfeil`

```
2955 \def\liSortierPfeil#1#2{
2956   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2957 }
```

`\liSortierPfeilUnten`

```
2958 \def\liSortierPfeilUnten#1#2{
2959   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2960 }
```

`\liSortierMarkierung`

```
2961 \def\liSortierMarkierung#1#2{\node[
2962   draw,
2963   very thick,
2964   fit=(reihe.#1) (reihe.#2),
2965   inner sep=0pt
2966 ] {}
2967 }

2968 \tikzset{
2969   li sortierung zahlenreihe/.style={
2970     draw,
2971     thin,
2972     font=\large,
2973     rectangle split horizontal,
2974     rectangle split,
2975   }
2976 }
```

```

2977 % https://tex.stackexchange.com/a/140895
2978 \RequirePackage{forest,xstring}
2979 \usetikzlibrary{calc}
2980
2981 \makeatletter
2982 \pgfmathdeclarefunction{strrepeat}{2}{%
2983   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2984   \let\pgfmathresult\pgfutil@empty
2985   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2986     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2987     \advance\pgfmath@count-1\relax
2988   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2989 \makeatother
2990
2991 \def\myNodes{}
2992
2993 \ExplSyntaxOn
2994 \newcommand*\sortList[1]{%
2995   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2996 \ExplSyntaxOff
2997
2998 \forestset{
2999   sort/.code={%
3000     \pgfmathparse{level()}>\forestSortLevel}%
3001     \ifnum\pgfmathresult=0
3002       \StrSubstitute{\forestov{content}}{ }{,}{\myList}%
3003       \sortList\myList
3004       \StrSubstitute{\myList}{,}{ }{\myList}%
3005       \pgfmathparse{strrepeat("1",level())}%
3006       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-1!({\forestov{name}}$)
3007         (m\forestov{name}) {\myList}}%
3008       \pgfmathparse{level()}==\forestSortLevel}%
3009       \ifnum\pgfmathresult=1
3010         \forestOget{\forestov{@first}}{name}\forestFirst
3011         \forestOget{\forestov{@last}}{name}\forestLast
3012         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
3013           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
3014       \fi
3015       \ifnum\forestov{@parent}=0\else
3016         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
3017       \fi
3018       \gappto\myNodes{;}%
3019     \fi}}
3020
3021 \forestset{sort level/.code=%
3022   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
3023   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
3024

```

## 2.41 spalten.sty

```
3025 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3026 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
3027 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
3028 realisiert werden kann.]
3029 \RequirePackage{multicol}
```

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

```
3030 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
3031
```

## 2.42 sql.sty

3032 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

3033 \ProvidesPackage{lehramt-informatik-tabelle}[2021/09/02 Zu Setzen von SQL]

### 2.42.1 Latex-Markup-Beispiel

```
\begin{liAdditum}[Übungsdatenbank]
% Datenbankname: Personalverwaltung
\begin{minted}{sql}
CREATE TABLE Abteilung(
  AbteilungsID INTEGER PRIMARY KEY,
  Bezeichnung VARCHAR(30)
);

CREATE TABLE Mitarbeiter(
  MitarbeiterID INTEGER PRIMARY KEY,
  Vorname VARCHAR(30),
  Vorgesetzter INTEGER REFERENCES Mitarbeiter(MitarbeiterID),
  Telefonnummer VARCHAR(50),
  Gehalt DOUBLE PRECISION
);

INSERT INTO Abteilung VALUES
  (1, 'Buchhaltung');

INSERT INTO Mitarbeiter
  (MitarbeiterID, Vorname, Nachname, Vorgesetzter, AbteilungsID, Telefonnummer, Gehalt)
VALUES
  (1, 'Hans', 'Meier', 11, 4, '023/13432', 2335),
  (2, 'Fred', 'Wolitz', 11, 2, '0233/413432', 1233);
\end{minted}
\index{SQL mit Übungsdatenbank}
\end{liAdditum}
```

3034 \liLadePakete{syntax}

3035 \RequirePackage{fancyvrb}

3036 \DefineVerbatimEnvironment{liSqlErgebnis}{Verbatim}

3037 {fontsize=\footnotesize}

3038

## 2.43 struktogramm.sty

```
3039 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3040 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
3041 Paket struktex zum Setzen von Struktogrammen]
3042 \RequirePackage{struktex}
3043
```



## 2.44 syntax.sty

```

3044 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3045 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
3046 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
3047 \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

```

```

3048 \ExplSyntaxOn
3049 \directlua{
3050   syntax = require('lehramt-informatik-syntax')
3051   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
3052   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
3053   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
3054   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
3055   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
3056   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
3057 }
3058 \RequirePackage{hyperref}
3059 \RequirePackage{minted}
3060 % pygmentize -L styles
3061 \usemintedstyle{colorful}
3062 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
3063 %\AfterEndEnvironment{minted}{\end{mdframed}}
3064 %\setminted{breaklines=true,linenos}
3065 \setminted{
3066   breaklines=true,
3067   linenos=false,
3068   fontsize=\footnotesize,
3069 }

```

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

```

3070 \newminted[liJavaAngabe]{java}{
3071   xleftmargin=1cm
3072 }

```

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

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

```

3073 \def\liJavaCode#1{
3074   \,
3075   \textcolor{blue}{
3076     \mintinline[
3077       fontsize=\normalsize,
3078       breakanywhere % https://github.com/gpoore/minted/issues/31#issuecomment-
458640242
3079     ]{java}|#1|
3080   }
3081   \,
3082 }

```

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

```

3083 \def\liLatexCode#1{\mintinline{latex}|#1|}

```

```

3084 \def\li@GithubLink#1#2{
3085   \begin{flushright}
3086     \tiny
3087     Code-Beispiel~auf~Github~ansehen:~
3088     \href{#1}{\nolinkurl{#2}}
3089   \end{flushright}
3090 }

\liJavaDatei Eine komplette Java-Datei einbinden, die Verzeichnis ./Code/src/main/java/org/bschlangaul
liegt.
3091 \NewDocumentCommand{\liJavaDatei}{ O{firstline=3} m }{
3092   \inputminted[#1]{java}{
3093     \directlua{
3094       syntax.drucke_absoluten_pfad('#2', false)
3095     }
3096   }
3097   \li@GithubLink
3098   {\directlua{syntax.drucke_github_url('#2', false)}}
3099   {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
3100 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
3101 \NewDocumentCommand{\liJavaTestDatei}{ O{firstline=3} m }{
3102   \inputminted[#1]{java}{
3103     \directlua{
3104       syntax.drucke_absoluten_pfad('#2', true)
3105     }
3106   }
3107   \li@GithubLink
3108   {\directlua{syntax.drucke_github_url('#2', true)}}
3109   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
3110 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([
\liJavaExamen$1{$2}{$3}{$4}{$5}
3111 \NewDocumentCommand{\liJavaExamen}{ O{firstline=3} m m m m }{
3112   \inputminted[#1]{java}{
3113     \directlua{
3114       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
3115     }
3116   }
3117   \li@GithubLink
3118   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
3119   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
3120 }
3121 }

\liAssemblerCode
3122 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
3123 \NewDocumentCommand{\liAssemblerDatei}{ m }{
3124   \inputminted{asm}{#1}
3125 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
3126 \NewDocumentCommand{\liMinispracheDatei}{ m }{
3127   \inputminted{componentpascal}{#1}
3128 }

```

```

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
3129 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
3130 \NewDocumentCommand{\liHaskellDatei}{ m }{
3131   \inputminted{haskell}{#1}
3132 }

3133 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
3134 \def\liSqlCode#1{\mintinline{sql}|#1|}

3135

```

## 2.45 syntaxbaum.sty

```
3136 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3137 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
3138 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
3139 \RequirePackage{tikz-qtrees}
3140
3141 \tikzset{li parsetree/.style={
3142     every internal node/.style={
3143         draw,circle
3144     },
3145     every leaf node/.style={
3146         draw,rectangle
3147     },
3148 }
3149 }
3150
```

## 2.46 synthese-algorithmus.sty

```
3151 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3152 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
3153 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
3154 Relation in die 3. Normalform]

3155 \liLadePakete{normalformen,mathe,typographie}
3156 \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}}

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

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

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

### 2.46.4 TeX-Markup Rechtsreduktion

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

\liPseudoUeberschrift{F}

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

\liPseudoUeberschrift{A}

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

## 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

```
3157 \def\liSyntheseUeberschrift#1{
3158   {
3159     \bfseries
3160     \rmfamily
3161     \str_case:nn {#1} {
3162       {1} {Kanonische-Überdeckung}
3163       {1-1} {Linksreduktion}
3164       {1-2} {Rechtsreduktion}
3165       {1-3} {Löschen-leerer-Klauseln}
3166       {1-4} {Vereinigung}
3167       {2} {Relationsschemata-formen}
3168       {3} {Schlüssel-hinzufügen}
3169       {4} {Entfernung-überflüssiger-Teilschemata}
3170     }
```

```

3171 }
3172 }

\liSyntheseErklaerung Let-Abkürzung: \let\erklaerung=\liSyntheseErklaerung
3173 \def\li@synthese@erklaerung@texte#1{
3174   \str_case:nn {#1} {
3175     {1} {
3176       Die~kanonische~Überdeckung~---also~die~kleinst~mögliche~noch~
3177       äquivalente~Menge~von~funktionalen~Abhängigkeiten~kann~in~vier~
3178       Schritten~erreicht~werden.
3179     }
3180     {1-1} {
3181       Führe~für~jede~funktionale~Anhängigkeit~
3182        $\alpha \rightarrow \beta$ ~die~Linksreduktion~durch,~
3183       überprüfe~also~für~alle~
3184        $A \in \alpha$ ,~ob~ $A$ ~überflüssig~ist,~d.h.~ob~
3185        $\beta \subseteqq \text{liAttributHuelle}\{F, \alpha \cup A\}$ .
3186     }
3187     {1-2} {
3188       Führe~für~jede~(verbliebene)~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~die~Rechtsreduktion~durch,~überprüfe~also~für~
3189       alle~ $B \in \beta$ ,~ob~ $B \in \text{liAttributHuelle}\{F \cup (\alpha \rightarrow \beta) \cup (\alpha \rightarrow \beta) \cup (\alpha \rightarrow \beta) \}$ ,~
3190        $\alpha$ ~gilt.~In~diesem~Fall~ist~B~auf~der~rechten~Seite~
3191       überflüssig~und~kann~eliminiert~werden,~d.h.~ $\alpha \rightarrow \beta$ ~
3192       wird~durch~ $\alpha \rightarrow \beta$ ~ersetzt.
3193     }
3194     {1-3} {
3195       Entferne~die~funktionalen~Abhängigkeiten~der~Form~ $\alpha \rightarrow \emptyset$ ,~die~im~2.~Schritt~möglicherweise~
3196       entstanden~sind.
3197     }
3198     {1-4} {
3199       Fasse~mittels~der~Vereinigungsregel~funktionale~Abhängigkeiten~
3200       der~Form~ $\alpha \rightarrow \beta_1, \dots, \alpha \rightarrow \beta_n$ ~so~dass~ $\alpha \rightarrow \beta_1 \cup \dots \cup \beta_n$ ~
3201       verbleibt.
3202     }
3203     % Kemper Seite 197
3204     {2} {
3205       Erzeuge~für~jede~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~ein~Relationenschema~ $\mathcal{R}_{\alpha \rightarrow \beta}$ ~
3206        $:= \alpha \cup \beta$ .
3207     }
3208     {3} {
3209       Falls~eines~der~in~Schritt~2.~erzeugten~Schemata~ $\mathcal{R}_{\alpha \rightarrow \beta}$ ~
3210       einen~Schlüsselkandidaten~von~ $\mathcal{R}_{\alpha \rightarrow \beta}$ ~bezüglich~ $F \subseteq c$ ~
3211       enthält,~sind~wir~fertig,~sonst~wähle~einen~Schlüsselkandidaten~
3212        $\mathcal{K} \subseteq \mathcal{R}_{\alpha \rightarrow \beta}$ ~aus~und~definiere~folgendes~
3213       zusätzliche~Schema:~ $\mathcal{R}_{\alpha \rightarrow \beta} \setminus \mathcal{K} := \mathcal{K}$ ~
3214       und~ $\mathcal{F} \setminus \mathcal{K} := \emptyset$ 
3215     }
3216     {4} {
3217       Eliminiere~diejenigen~Schemata~ $\mathcal{R}_{\alpha \rightarrow \beta}$ ,~die~in~einem~
3218       anderen~Relationenschema~ $\mathcal{R}_{\alpha' \rightarrow \beta'}$ ~enthalten~sind,~d.h.~
3219        $\mathcal{R}_{\alpha \rightarrow \beta} \subseteq \mathcal{R}_{\alpha' \rightarrow \beta'}$ .
3220     }
3221   }
3222 }
3223 \def\liSyntheseErklaerung#1{
3224   {
3225     \itshape
3226     \footnotesize

```

```

3233     \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
3234   }
3235 }

```

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

```

3236 \def\liSyntheseUeberErklaerung#1{
3237   \liSyntheseUeberschrift{#1}\par
3238   \liSyntheseErklaerung{#1}
3239 }

```

```

3240 \ExplSyntaxOff
3241

```



## 2.47 tabelle.sty

3242 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

3243 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]

3244 \RequirePackage{tabularx}

3245

## 2.48 typography.sty

```

3246 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3247 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
3248 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
3249 formatierung.sty definiert.]

```

```

3250 \ExplSyntaxOn

```

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

```

3251 \RequirePackage{fontawesome}

```

`\liErledigt` `\liErledigt:` ☑

```

3252 \let\liErledigt=\faCheckSquareO

```

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

```

3253 \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 —

```

3254 \def\liParagraphMitLinien#1{
3255   \noindent
3256   \vrule height 2pt depth -1.6pt width 0.4cm
3257   \enspace
3258   #1
3259   \enspace
3260   \leaders\vrule height 2pt depth -1.6pt \hfill \null
3261   \par
3262   \medskip
3263 }

```

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

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

```

3264 \def\liGeschweifteKlammern#1#2#3#4{
3265   \par
3266   \medskip
3267   \noindent
3268   #1 \, $= \Bigl\{ \$
3269   \vspace{#3}
3270   #2
3271   \vspace{#4}
3272   \begin{flushright} \$\Bigr\} \$\end{flushright}
3273   \par
3274 }

```

`\liTypoUeberschrift`

```

3275 \def\liTypoUeberschrift#1{
3276   {
3277     \bfseries\rmfamily
3278     #1
3279   }
3280 }

```

\liTypoUeberGross

```
3281 \def\liTypoUeberGross#1{
3282   {
3283     \huge
3284     \liTypoUeberschrift{#1}
3285   }
3286 }
```

\liTypoUeberGross

```
3287 \def\liTypoUeberGROSS#1{
3288   {
3289     \Huge
3290     \liTypoUeberschrift{#1}
3291   }
3292 }
```

3293 \ExplSyntaxOff

3294

## 2.49 uml.sty

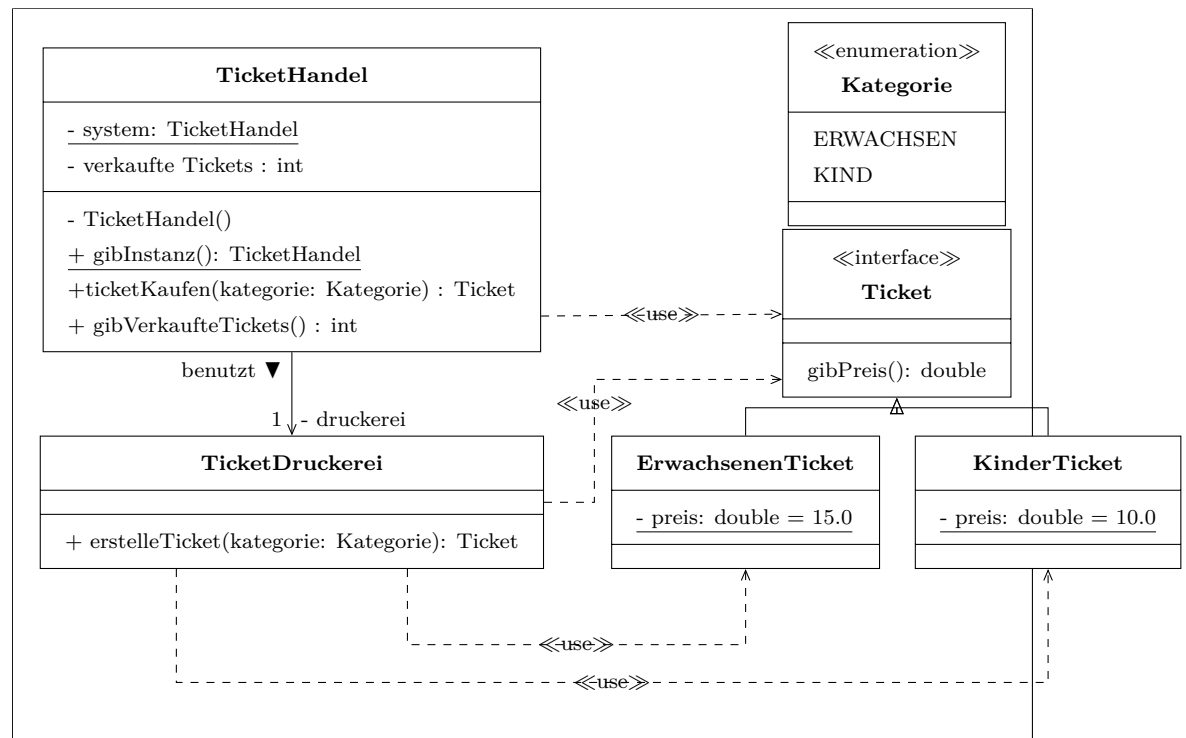
```

3295 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3296 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
3297 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
3298 Erweiterung bereitstellt]

3299 \RequirePackage{tikz-uml}
3300 \RequirePackage{tikz-uml-activity}
3301 % Not compatible with wasysym
3302 %\RequirePackage{mathabx}
3303 \RequirePackage{wasysym}
3304 \usetikzlibrary{positioning}

3305 \tikzumlset{
3306   fill class=white!0,
3307   font=\footnotesize,
3308   fill object=white!0,
3309   fill note=white!0,
3310   fill state=white!0,
3311   % Use case
3312   fill usecase=white!0,
3313   fill system=white!0,
3314 }

```



```

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

```

```

3315 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
3316   \def\@liDirLeft{}
3317   \def\@liDirRight{ \RIGHTarrow}
3318   \pgfkeys{/lese/dir/.is choice}
3319   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
3320   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
3321   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{\LEFTarrow }}}
3322   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
3323
3324   \def\@liPos{above}
3325   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
3326

```

```

3327 \def\@liDistance{0cm}
3328 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
3329
3330 \pgfkeys{/lese/.cd,#1}
3331
3332 \node[\@liPos = \@liDistance of #2-middle] {
3333   \@liDirLeft{\footnotesize#2}\@liDirRight
3334 };
3335 }
3336

```

## 2.50 vollstaendige-induktion.sty

3337 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 3338 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01  
 3339 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die  
 3340 Überschriften für die einzelnen Schritte]

### 2.50.1 Makro-Kürzel

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

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

Lade häufig benötigte Pakete

3341 \RequirePackage{lehramt-informatik-typographie}  
 3342 \RequirePackage{lehramt-informatik-mathe}  
 3343 \RequirePackage{lehramt-informatik-syntax}  
 3344 \ExplSyntaxOn

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

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

3345 \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

```
3346 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
3347 \def\liInduktionAnfang{
```

```
3348   \liPseudoUeberschrift{Induktionsanfang}
```

```
3349
```

```
3350   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
3351   \liParagraphMitLinien{
```

```
3352     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
```

```
3353   }
```

```
3354 }
```

\liInduktionVoraussetzung

```
3355 \def\liInduktionVoraussetzung{
```

```
3356   \liPseudoUeberschrift{Induktionsvoraussetzung}
```

```
3357
```

```
3358   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
3359   \liParagraphMitLinien{
```

```
3360     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ ~.
```

```
3361   }
```

```
3362 }
```

\liInduktionSchritt

```
3363 \def\liInduktionSchritt{
```

```
3364   \liPseudoUeberschrift{Induktionsschritt}
```

```
3365
```

```
3366   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
3367   \liParagraphMitLinien{
```

```
3368     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
```

```
3369     auch~ $A(n=k+1)$ ~wahr~sein~muss.
```

```
3370   }
```

```
3371 }
```

```
3372 \ExplSyntaxOff
```

```
3373
```

## 2.51 wasserfall.sty

```
3374 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3375 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
3376 \RequirePackage{tikz}
3377 \tikzset{wasserfall/.style={
3378   >=stealth,
3379   node distance = 2mm and -8mm,
3380   start chain = A going below right,
3381   every node/.style = {
3382     draw,
3383     text width=24mm,
3384     minimum height=12mm,
3385     align=center,
3386     inner sep=1mm,
3387     fill=white,
3388     drop shadow={fill=black},
3389     on chain=A
3390   },
3391 }}
3392 \usetikzlibrary{chains,positioning,shadows}
3393
```



## 2.52 wpkalkuel.sty

```
3394 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3395 \ProvidesPackage{lehrant-informatik-wpkalkuel}[2020/06/13]
```

### 2.52.1 Makro-Kürzel

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

```
3396 \RequirePackage{amsmath}
```

```
3397 \ExplSyntaxOn
```

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

```
3398 \def\liWpKalkuelOhneMathe#1#2{
3399   \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)
3400 }
3401 \def\liWpKalkuel#1#2{
3402   \ifmmode
3403     \liWpKalkuelOhneMathe{#1}{#2}
3404   \else
3405     $\liWpKalkuelOhneMathe{#1}{#2}$
3406   \fi
3407 }
```

**\MatheEnv**

```
3408 \def\MatheEnv#1{
3409   \medskip
3410
3411   \hspace{1em}#1
3412
3413   \medskip
3414 }
```

**\Mathe**

```
3415 \def\Mathe#1{
3416   \MatheEnv{${#1}$}
3417 }
```

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

```
3418 \def\liWpEquivalent#1{
3419   \MatheEnv{${\equiv}\hspace{1em}${#1}$}
3420 }
```

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

```
3421 \newlength{\@Skip@Erklaerung@Reset}
3422 \def\liWpErklaerung#1{
3423   \setlength{\@Skip@Erklaerung@Reset}{\leftskip}
3424   \setlength{\leftskip}{0.5cm}
3425
3426   \par
3427   \noindent
3428   {
3429     \scriptsize
3430     #1
3431   }
3432   \par
3433
3434   \setlength{\leftskip}{\@Skip@Erklaerung@Reset}
3435 }
```

```

3436 \def\liWpErklaerungVerzweigung{
3437   $\liWpKalkuel0hneMathe{if~\{{b~}\}~then~\{{a1~}\}~else~\{{a2~}\}}{Q}
3438   \equiv
3439   (b \land \liWpKalkuel0hneMathe{a1}{Q})
3440   \lor
3441   (\neg b \land \liWpKalkuel0hneMathe{a2}{Q})$
3442 }

3443 \ExplSyntaxOff

3444

```

### 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		2722, 2863, 2885,
\#	..... 114	\_ 2525, 2535
\,	.. 364, 423, 478, 705,	2899, 3062, 3085, 3272
	723, 1355, 1357,	\begingroup 1964, 2515, 2983
	2134, 2135, 2136,	\beschriftung .....
	2573, 3074, 3081, 3268	..... 2007, 2011,
\@Skip@Erklaerung@Reset		2015, 2019, 2023, 2025
...	3421, 3423, 3434	\beta ..... 3182,
\@afterheading .....		3185, 3189, 3190,
...	1606, 1984, 2103	3191, 3194, 3204,
\@afterindentfalse ..		3205, 3206, 3211, 3212
...	1605, 1983, 2102	\bf ..... 2760, 2761, 2762
\@liDirLeft 3316, 3321, 3333		\bfseries ..... 564,
\@liDirRight 3317, 3319,		1405, 1407, 1583,
3320, 3321, 3322, 3333		1644, 1701, 2383,
\@liDistance .....		2760, 2766, 2768,
...	3327, 3328, 3332	2770, 2771, 3159, 3277
\@liPos .. 3324, 3325, 3332		\bgroup ..... 2002, 2041
\\	..... 735, 763, 764,	\Bigl ..... 3268
	767, 768, 771, 772,	\Bigr ..... 3272
	867, 868, 869, 976,	\bigskip ..... 456,
	1010, 1012, 1042,	737, 742, 1601,
	1051, 1096, 1138,	1613, 1616, 1694,
	1139, 1140, 1145,	1746, 1749, 1973, 2441
	1146, 1147, 1167,	\bool ..... 342, 424
	1971, 2468, 2526, 2529	\bowtie .....
\{	..... 212, 1292,	2908, 2911, 2912, 2913
	1302, 1314, 1315,	\Box ..... 164
	1320, 1354, 1810,	\boxtimes ..... 554
	2513, 2938, 3268, 3437	
\}	..... 212, 1292,	
	1302, 1314, 1315,	
	1322, 1358, 1811,	
	2513, 2938, 3272, 3437	
\_	..... 37, 45, 50,	
	52, 311, 339, 363,	
	366, 376, 399, 403,	
	407, 409, 411, 413,	
	416, 422, 423, 437,	
	438, 444, 447, 450,	
	460, 467, 469, 472,	
	1576, 1588, 1624, 1678	



<code>\frac</code>	2181, 2214, 2246, 2261	<code>\ifnum</code>	2744, 1374, 1375, 1376,
<code>\fullouterjoin</code>	2913		1377, 1380, 1381,
<b>G</b>		<code>\ifx</code>	2023, 2545, 3013
<code>\g</code>	39, 285, 286, 308,	<code>\in</code>	583, 739, 1390, 1391, 1392,
	312, 318, 319, 320,		1904, 1905, 1906,
	321, 322, 324, 325,		2088, 2089, 2090, 2097
	327, 328, 330, 331,	<code>\includegraphics</code>	1416
	332, 333, 334, 335,		<code>\labelenumii</code>
	336, 337, 340, 344,		1417
	345, 346, 349, 351,		<code>\labelitemi</code>
	352, 353, 354, 355,		1410
	356, 401, 405, 409,	<code>\inhalts</code>	1411
	412, 413, 415, 416,		<code>\labelitemiii</code>
	418, 419, 426, 427,	<code>\inhaltsverzeichnis</code>	1412
	428, 429, 437, 438,		<code>\labelitemiv</code>
	440, 446, 447, 449,	<code>\input</code>	1413
	450, 452, 453, 465,		<code>\land</code>
	468, 473, 475, 479,		3439, 3441
	1550, 1552, 1563,	<code>\inputminted</code>	<code>\LARGE</code>
	1565, 1572, 1585,		1405, 1710
	1586, 1652, 1656,		<code>\Large</code>
	1660, 1664, 1665,		1584
	1666, 1667, 1668,		<code>\large</code>
	1670, 1671, 1673,		1575, 1684, 1769, 2972
	1703, 1705, 1710, 1714	<code>\int</code>	<code>\leaders</code>
<code>\Gamma</code>	111, 169, 212, 1315		3260
<code>\gappto</code>	3018	<code>\item</code>	<code>\left</code>
<code>\geometry</code>	5		2144, 2583
<code>\geq</code>	1817, 2212,		<code>\LEFTTarrow</code>
	2698, 2703, 2719, 2723		3321
<b>H</b>			<code>\leftarrow</code>
<code>\hbox</code>	2908		671
<code>\headrulewidth</code>	1431, 1921		<code>\leftouterjoin</code>
<code>\headwidth</code>	1923		2911
<code>\hfill</code>	1591,		<code>\leftskip</code>
	2231, 2237, 2242, 3260		3423, 3424, 3434
<code>\hinweis</code>	1994		<code>\LehramtInformatikAutorEmail</code>
<code>\hline</code>	2468		1920
<code>\horizontale</code>	1556, 1599		<code>\LehramtInformatikAutorName</code>
<code>\href</code>	1368, 2132,		1919
	2299, 2334, 2343, 3088		<code>\LehramtInformatikGitBranch</code>
<code>\hspace</code>	2660, 3411, 3419		464, 3056
<code>\ht</code>	2909		<code>\LehramtInformatikGithubCodeRepo</code>
<code>\Huge</code>	1701, 3289		3055
<code>\huge</code>	1644, 3283		<code>\LehramtInformatikGithubDomain</code>
<b>I</b>			461, 3052
<code>\i</code>	1525, 1526		<code>\LehramtInformatikGithubRawDomain</code>
<code>\ifcase</code>	2817		3053
<code>\ifLiADDITUM</code>	229, 2038, 2056		<code>\LehramtInformatikGithubTexRepo</code>
<code>\ifLiANTWORT</code>	233, 1999, 2031		462, 3054
<code>\ifLiEXKURS</code>	231, 2062		<code>\LehramtInformatikRepository</code>
<code>\ifmmode</code>	657, 665, 673,		17, 20, 23, 26, 29,
	681, 688, 1294,		1451, 1662, 1940,
	1308, 1344, 2150,		1941, 1942, 1943,
	2160, 2170, 2184,		1944, 1945, 1946,
	2507, 2589, 2650, 3402		1947, 1948, 1949, 3051
<code>\IfNoValueTF</code>	2045, 2128, 2132		<code>\LehramtInformatikTitel</code>
			1915
<b>J</b>			<code>\leq</code>
<code>\j</code>	1515, 1516, 1518, 1519,		2246, 2706, 2726
	1520, 1525, 1526, 1527		<code>\let</code>
<b>K</b>			1304,
<code>\k</code>	1525		1305, 1965, 2857,
<code>\keys</code>	41, 75,		2858, 2859, 2860,
	87, 117, 127, 175,		2873, 2875, 2876,
	185, 316, 627, 631,		2877, 2879, 2880,
	645, 650, 1379, 1386		2881, 2882, 2892,
<b>L</b>			2984, 3022, 3023, 3252
<code>\l</code>	68, 69, 70, 71,		<code>\li@chomsky@erklaerung@texte</code>
	72, 73, 76, 77, 78,		574, 610
	79, 80, 82, 84, 89,		<code>\li@EntwurfsCode</code>
	90, 91, 92, 93, 94,		752, 803, 804, 805,
	109, 110, 111, 112,		854, 855, 856, 857,
	113, 114, 115, 118,		923, 924, 925, 926,
	119, 120, 121, 122,		927, 928, 955, 956,
	123, 124, 130, 131,		957, 958, 959, 960,
	132, 133, 134, 135,		1026, 1196, 1197,
	136, 167, 168, 169,		1198, 1199, 1239, 1240
	170, 171, 172, 173,		
	176, 177, 178, 179,		
	180, 181, 182, 188,		
	189, 190, 191, 192,		
	193, 194, 625, 628,		
	633, 634, 637, 642,		
	643, 646, 647, 652,		

<code>\li@EntwurfsCodeAllgemein</code>	<code>\liChomskyUeberErklaerung</code>	<code>\liEntwurfsEinzelstueckBeschreibung</code>
..... <a href="#">751</a>	..... <a href="#">613</a>	..... <a href="#">1002</a> , <a href="#">1029</a>
<code>\li@fussnote@text</code> <a href="#">2419</a> ,	<code>\liChomskyUeberschrift</code>	<code>\liEntwurfsEinzelstueckCode</code>
<a href="#">2425</a> , <a href="#">2429</a> , <a href="#">2433</a> , <a href="#">2437</a>	..... <a href="#">562</a> , <a href="#">614</a>	..... <a href="#">1025</a> , <a href="#">1035</a>
<code>\li@GithubLink</code> .....	<code>\liCpmEreignis</code> .....	<code>\liEntwurfsEinzelstueckUml</code>
<a href="#">3084</a> , <a href="#">3097</a> , <a href="#">3107</a> , <a href="#">3118</a>	<a href="#">623</a>	..... <a href="#">1007</a> , <a href="#">1031</a>
<code>\li@mget</code> . <a href="#">1502</a> , <a href="#">1506</a> , <a href="#">1526</a>	<code>\liCpmFruehErklaerung</code> <a href="#">694</a>	<code>\liEntwurfsErbauer</code> . <a href="#">1085</a>
<code>\li@minc</code> .....	<code>\liCpmFruehI</code> ... <a href="#">687</a> , <a href="#">707</a>	<code>\liEntwurfsErbauerAkteure</code>
<a href="#">1505</a> , <a href="#">1527</a>	<code>\liCpmSpaetErklaerung</code> <a href="#">712</a>	..... <a href="#">1057</a> , <a href="#">1087</a>
<code>\li@mset</code> .....	<code>\liCpmSpaetI</code> ... <a href="#">680</a> , <a href="#">725</a>	<code>\liEntwurfsErbauerUml</code>
<a href="#">1499</a> , <a href="#">1507</a> , <a href="#">1516</a> , <a href="#">1519</a>	<code>\liCpmVon</code> .....	..... <a href="#">1037</a> , <a href="#">1086</a>
<code>\li@numdiscs</code> .....	<code>\liCpmVonOhneMathe</code> ..	<code>\liEntwurfsFabrikmethode</code>
... <a href="#">1511</a> , <a href="#">1520</a> , <a href="#">1526</a>	..... <a href="#">663</a> , <a href="#">666</a> , <a href="#">668</a>	..... <a href="#">1131</a> , <a href="#">1156</a>
<code>\li@Rmodell@Schrift</code> .	<code>\liCpmVonZu</code> .....	<code>\liEntwurfsFabrikmethodeAkteure</code>
... <a href="#">2922</a> , <a href="#">2931</a> , <a href="#">2941</a>	<a href="#">655</a>	..... <a href="#">1106</a> , <a href="#">1133</a>
<code>\li@sequence</code> .. <a href="#">1512</a> , <a href="#">1525</a>	<code>\liCpmVonZuOhneMathe</code>	<code>\liEntwurfsFabrikmethodeUml</code>
<code>\li@synthese@erklaerung@text</code>	..... <a href="#">655</a> , <a href="#">658</a> , <a href="#">660</a>	..... <a href="#">1089</a> , <a href="#">1132</a>
..... <a href="#">3173</a> , <a href="#">3233</a>	<code>\liCpmVorgang</code> .....	<code>\liEntwurfsKompositum</code>
<code>\liAbleitung</code> .....	<code>\liCpmZu</code> .....	..... <a href="#">1156</a>
<a href="#">1329</a>	<code>\liCpmZuOhneMathe</code> ...	<code>\liEntwurfsKompositumAkteure</code>
<code>liAdditum (environment)</code>	..... <a href="#">671</a> , <a href="#">674</a> , <a href="#">676</a>	..... <a href="#">1158</a>
..... <a href="#">2036</a>	<code>liDiagramm (environ-</code>	<code>\liEntwurfsKompositumUml</code>
<code>\LiADDITUMfalse</code> .....	<code>ment)</code> .....	..... <a href="#">1135</a> , <a href="#">1157</a>
<a href="#">229</a>	<a href="#">2115</a>	<code>\liEntwurfsModellPraesentationSteuerung</code>
<code>\LiADDITUMtrue</code> . <a href="#">229</a> , <a href="#">1547</a>	<code>liEinbettung (environ-</code>	<code>\liEntwurfsModellPraesentationSteuerungAk</code>
<code>liAHuelle (environment)</code>	<code>ment)</code> .....	<code>\liEntwurfsModellPraesentationSteuerungUml</code>
..... <a href="#">2514</a>	<a href="#">1996</a>	..... <a href="#">1160</a> , <a href="#">1178</a>
<code>\liAlphabet</code> .....	<code>\liEntwurfs</code> .....	<code>\liEntwurfsStellvertreter</code>
<a href="#">1314</a>	<a href="#">1177</a>	..... <a href="#">1201</a>
<code>liAntwort (environment)</code>	<code>\liEntwurfsAbstrakteFabrik</code>	<code>\liEntwurfsStellvertreterCode</code>
..... <a href="#">1997</a>	..... <a href="#">807</a>	..... <a href="#">1195</a> , <a href="#">1203</a>
<code>\LiANTWORTfalse</code> .....	<code>\liEntwurfsAbstrakteFabrikBeschreibung</code> .....	<code>\liEntwurfsStellvertreterUml</code>
<a href="#">233</a>	..... <a href="#">755</a> , <a href="#">808</a>	..... <a href="#">1181</a> , <a href="#">1202</a>
<code>\LiANTWORTtrue</code> .....	<code>\liEntwurfsAbstrakteFabrikCode</code>	<code>\liEntwurfsZustand</code> . <a href="#">1242</a>
<a href="#">233</a>	..... <a href="#">802</a> , <a href="#">812</a>	<code>\liEntwurfsZustandAkteure</code>
<code>\liAnweisung</code> .....	<code>\liEntwurfsAbstrakteFabrikUml</code>	..... <a href="#">1220</a> , <a href="#">1244</a>
<a href="#">1895</a>	..... <a href="#">760</a> , <a href="#">810</a>	<code>\liEntwurfsZustandCode</code>
<code>\liAssemblerCode</code> ... <a href="#">3122</a>	<code>\liEntwurfsAdapter</code> .. <a href="#">859</a>	..... <a href="#">1238</a> , <a href="#">1245</a>
<code>\liAssemblerDatei</code> .. <a href="#">3123</a>	<code>\liEntwurfsAdapterAkteure</code>	<code>\liEntwurfsZustandUml</code>
<code>\liAttribut</code> .....	..... <a href="#">829</a> , <a href="#">861</a>	..... <a href="#">1205</a> , <a href="#">1243</a>
<a href="#">2941</a>	<code>\liEntwurfsAdapterCode</code>	<code>\liEpsilon</code> .....
<code>\liAttributHuelle</code> ...	..... <a href="#">853</a> , <a href="#">862</a>	<a href="#">1300</a>
... <a href="#">2505</a> , <a href="#">3185</a> , <a href="#">3190</a>	<code>\liEntwurfsAdapterUml</code>	<code>\liErAttribute</code> .....
<code>\liAttributHuelleOhneMathe</code>	..... <a href="#">814</a> , <a href="#">860</a>	... <a href="#">1258</a> , <a href="#">1272</a> , <a href="#">1274</a>
..... <a href="#">2505</a> , <a href="#">2508</a> ,	<code>\liEntwurfsBeobachter</code> <a href="#">930</a>	<code>\liErDatenbankName</code> . <a href="#">1277</a>
<a href="#">2510</a> , <a href="#">2524</a> , <a href="#">2534</a> , <a href="#">2542</a>	<code>\liEntwurfsBeobachterAkteure</code>	<code>\liErEntity</code> <a href="#">1256</a> , <a href="#">1260</a> , <a href="#">1262</a>
<code>\liAttributMenge</code> ....	..... <a href="#">888</a> , <a href="#">932</a>	<code>\liErledigt</code> .....
<a href="#">2513</a> , <a href="#">2525</a> , <a href="#">2528</a> ,	<code>\liEntwurfsBeobachterCode</code>	<a href="#">1271</a>
<a href="#">2535</a> , <a href="#">2536</a> , <a href="#">2550</a> , <a href="#">2552</a>	..... <a href="#">922</a> , <a href="#">933</a>	<code>\liErMpAttribute</code> ... <a href="#">1271</a>
<code>\liAufgabe</code> .....	<code>\liEntwurfsBeobachterUml</code>	<code>\liErMpEntity</code> .....
<a href="#">16</a>	..... <a href="#">864</a> , <a href="#">931</a>	<a href="#">1259</a>
<code>\liAufgabenMetadaten</code>	<code>\liEntwurfsDekorierer</code> <a href="#">962</a>	<code>\liErMpRelationship</code> <a href="#">1265</a>
..... <a href="#">47</a> , <a href="#">1569</a> , <a href="#">1676</a>	<code>\liEntwurfsDekoriererAkteure</code>	<code>\liErRelationship</code> ...
<code>\liAufgabenTitel</code> .... <a href="#">55</a>	..... <a href="#">964</a>	... <a href="#">1257</a> , <a href="#">1266</a> , <a href="#">1268</a>
<code>\liAusdruck</code> .....	<code>\liEntwurfsDekoriererCode</code>	<code>\liExamensAufgabe</code> .... <a href="#">19</a>
<a href="#">1351</a>	..... <a href="#">954</a> , <a href="#">965</a>	<code>\liExamensAufgabeA</code> ... <a href="#">28</a>
<code>\liAutomat</code> .....	<code>\liEntwurfsDekoriererUml</code>	<code>\liExamensAufgabeTA</code> .. <a href="#">25</a>
<a href="#">66</a>	..... <a href="#">935</a> , <a href="#">963</a>	<code>\liExamensAufgabeTTA</code> . <a href="#">22</a>
<code>\liAutomatenKante</code> .... <a href="#">98</a>	<code>\liEntwurfsEinfacheFabrik</code>	<code>\liExkurs (environment)</code> <a href="#">2061</a>
<code>\liBandAlphabet</code> .... <a href="#">1315</a>	..... <a href="#">998</a>	<code>\LiEXKURSfalse</code> .....
<code>\liBedingung</code> .....	<code>\liEntwurfsEinfacheFabrikAkteure</code>	<a href="#">231</a>
<a href="#">1896</a>	..... <a href="#">982</a> , <a href="#">1000</a>	<code>\LiEXKURStrue</code> .....
<code>\liBedingungDrei</code> ....	<code>\liEntwurfsEinfacheFabrikUml</code>	<a href="#">555</a>
... <a href="#">2199</a> , <a href="#">2242</a> , <a href="#">2282</a>	..... <a href="#">967</a> , <a href="#">999</a>	<code>\liFlaci</code> .....
<code>\liBedingungEins</code> ....	<code>\liEntwurfsEinzelstueck</code>	<a href="#">1361</a>
... <a href="#">2193</a> , <a href="#">2231</a> , <a href="#">2274</a>	..... <a href="#">1028</a>	<code>\liFremd</code> .....
<code>\liBedingungFalsch</code> . <a href="#">1898</a>	<code>\liEntwurfsEinzelstueckAkteure</code>	<a href="#">2921</a>
<code>\liBedingungWahr</code> ... <a href="#">1897</a>	..... <a href="#">1017</a> , <a href="#">1033</a>	
<code>\liBedingungZwei</code> ....		
... <a href="#">2196</a> , <a href="#">2237</a> , <a href="#">2278</a>		
<code>\liBeschriftung</code> .... <a href="#">1986</a>		
<code>\liBindeAufgabeEin</code> . <a href="#">1659</a>		
<code>\liChomskyErklaerung</code>		
..... <a href="#">574</a> , <a href="#">615</a>		

<code>\liFunktionaleAbhaengigkeit</code>	1759, 2319, 2417, ... 2544, 2547, 2555	<code>\liPetriSetzeSchluessel</code>	2601
<code>\liFunktionaleAbhaengigkeiten</code>	2558	<code>\liPetriTransitionsName</code>	2648, 2660
<code>\liFussnote</code>	2418, 2420	<code>\liPetriTransitionsNameOhneMathe</code>	2648, 2651, 2653
<code>\liFussnoteDreiText</code>	2432, 2450	<code>\liPetriTransPfeile</code>	2660
<code>\liFussnoteEinsText</code>	2424, 2444	<code>\liPolynomiellReduzierbar</code>	1779
<code>\liFussnoteLink</code>	2131	<code>\liPotenzmenge</code>	1301, 1305, 2671
<code>\liFussnoten</code>	2440	<code>\liPotenzmengeOhneMathe</code>	1302, 1303, 1304
<code>\liFussnoteUrl</code>	1175, 2127	<code>\liPrimaer</code>	2920
<code>\liFussnoteVierText</code>	2436, 2453	<code>\liProblemBeschreibung</code>	1763
<code>\liFussnoteZweiText</code>	2428, 2447	<code>\liProblemClique</code>	1786
<code>\liGeschweifteKlammern</code>	1333, 2559, 3264	<code>\liProblemName</code>	1762, 1769, 1781, 1783, 1796, 1807, 1808, 1816, 1817
<code>\liGrafikCCLizenz</code>	1453, 2403	<code>\liProblemSat</code>	1815
<code>\liGrafikLogo</code>	1458, 1693, 2379	<code>\liProblemSubsetSum</code>	1806, 1815
<code>\liGrafikLogoPfad</code>	1450, 1455, 1460	<code>\liProblemVertexCover</code>	1786, 1794
<code>\liGrammatik</code>	1372	<code>\liProduktionen</code>	1340, 1382
<code>liGraphenFormat</code> (environment)	1492	<code>liProduktionsRegeln</code> (environment)	1330
<code>\liHanoi</code>	1499	<code>liProjektSprache</code> (environment)	1995
<code>\liHaskellCode</code>	3129	<code>\liPseudoUeberschrift</code>	1972, 2047, 2050, 2465, 2475, 3348, 3356, 3364
<code>\liHaskellDatei</code>	3130	<code>\liPumpingKontextfrei</code>	2717
<code>\liInduktionAnfang</code>	3347	<code>\liPumpingRegulaer</code>	2696
<code>\liInduktionErklaerung</code>	3346	<code>liQuellen</code> (environment)	2085
<code>\liInduktionMarkierung</code>	3345	<code>\liRechtsReduktionInline</code>	2539
<code>\liInduktionSchritt</code>	3363	<code>\liRekursionsGleichung</code>	2190, 2253
<code>\liInduktionVoraussetzung</code>	3355	<code>\liRelation</code>	2569
<code>liJavaAngabe</code> (environment)	3070	<code>liRelationenSchemaFormat</code> (environment)	2942
<code>\liJavaCode</code>	3073	<code>\liRelationMenge</code>	2936
<code>\liJavaDatei</code>	753, 3091	<code>\liRichtig</code>	554
<code>\liJavaExamen</code>	3111	<code>liRmodell</code> (environment)	2922
<code>\liJavaTestDatei</code>	3101	<code>\liRundeKlammer</code>	2143, 2147, 2157, 2167, 2181, 2582, 2586
<code>liKasten</code> (environment)	1420	<code>\liSetzeExamenTeilaufgabeNr</code>	1655
<code>\liKellerAutomat</code>	107	<code>\liSetzeExamenThemaNr</code>	1651
<code>\liKellerKante</code>	145	<code>\liSortierMarkierung</code>	2961
<code>\liKellerUebergang</code>	140, 146	<code>\liSortierPfeil</code>	2955
<code>\liKontrollCode</code>	1899	<code>\liSortierPfeilUnten</code>	2958
<code>liKontrollflussgraph</code> (environment)	1887	<code>\liSpaltenUmbruch</code>	3030
<code>\liKontrollKnotenPfad</code>	1901	<code>\liSqlCode</code>	3134
<code>\liKontrollTextzeileKnoten</code>	1900, 1905		
<code>\liKurzeTabellenLinie</code>	735		
<code>\liLadeAllePakete</code>	239, 1647		
<code>\liLadePakete</code>	59, 62, 235, 240, 561, 622, 1291, 1535, 1633,		
<code>\liLatexCode</code>	3083		
<code>\liLeereZelle</code>	2455		
<code>liLernkartei</code> (environment)	2105		
<code>\liLinksReduktion</code>	2522		
<code>\liLinksReduktionInline</code>	2531, 2539		
<code>\liLogoTextCCLizenz</code>	1615, 1748, 2399		
<code>\liLogoTextProjekt</code>	1612, 1745, 2375		
<code>\liMasterExkurs</code>	2287		
<code>\liMasterFaelle</code>	2226, 2294		
<code>\liMasterFallRechnung</code>	2272		
<code>\liMasterVariablen</code>	2203, 2289		
<code>\liMasterVariablenDeklaration</code>	2249		
<code>\liMasterWolframLink</code>	2297		
<code>\liMenge</code>	76, 77, 79, 118, 119, 120, 124, 176, 177, 178, 182, 1292, 1341, 1380, 1381		
<code>\liMengeOhneMathe</code>	1292, 1295, 1297		
<code>\liMetaBsclangaulSammlung</code>	2320, 2384		
<code>\liMetaCCLink</code>	2332, 2408		
<code>\liMetaEmailLink</code>	2342, 2358		
<code>\liMetaHermineBsclangaulAndFriends</code>	1696, 2323, 2388		
<code>\liMetaHilfMit</code>	1621, 2349, 2360		
<code>\liMetaQuelltext</code>	1623, 2360		
<code>\liMetaSetze</code>	36, 48, 1570, 1677		
<code>\liMetaUeberDasProjekt</code>	2326, 2395		
<code>\liMinimierungErklaerung</code>	2477		
<code>\liMinispracheDatei</code>	3126		
<code>\linespread</code>	2929		
<code>\liNichtsZuTun</code>	3253		
<code>\liO</code>	2166, 2194		
<code>\liOmega</code>	2156, 2200		
<code>\liOmegaOhneMathe</code>	2156, 2161, 2163		
<code>\liONotationO</code>	2585		
<code>\liOOhneMathe</code>	2166, 2171, 2173		
<code>\liParagraphMitLinien</code>	610, 695, 713, 2479, 3233, 3254, 3351, 3359, 3367		
<code>\liPetriErreichKnotenDrei</code>	2659		
<code>\liPetriErreichTransition</code>	2656		





1685, 1689, 1693, 1695, 1696, 1703, 1705, 1710, 1714, 1718, 1975, 1981, 1987, 2077, 2101, 2110, 2386, 2389, 2861, 2883, 2898, 2939, 3237, 3261, 3265, 3273, 3426, 3432	\QS@sort@b . . . . . 2816, 2817 \QS@sort@c . . . . . 2820, 2827 \QS@sort@d . . . . . 2828, 2836 \QS@sort@empty . 2818, 2823 \QS@sort@single 2819, 2824 \QSinitialize . . . . . ... 2741, 2853, 2888 \QSIr . 2824, 2830, 2838, 2858, 2872, 2877, 2880 \QSIrr 2859, 2872, 2873, 2881 \QSLr . . . . . 2830, 2837, 2848, 2849, 2857, 2870, 2875, 2879 \QSpivotStep . . . . . 2743, 2853, 2857, 2868 \QSR . . . . . 2830 \QSRr . . . . . 2839, 2860, 2871, 2876, 2882, 2891, 2892, 2893 \QSSortStep . . . . . 2745, 2853, 2869, 2870 \quad . . . . . 2421	\rightarrow . . . . . 212, 578, 583, 591, 595, 597, 598, 600, 655, 663, 2660, 3182, 3189, 3191, 3194, 3199, 3204, 3205, 3210 \rightouterjoin . . . . 2912 \rmfamily . . . . . 565, 1526, 3160, 3277 \Roman . . . . . 1415 \roman . . . . . 1415, 1417 \romannumeral . . . . . 2828 \rtimes . . . . . 2305 \rule . . . . . 1559, 2861, 2883, 2898, 2909
\paragraph . . . . . 1407 \parindent . . . . . 2930 \path . . . . . 99, 146, 209, 652 \pgfkeys . . 2612, 3318, 3319, 3320, 3321, 3322, 3325, 3328, 3330 \pgfmath@count . . . . . ... 2983, 2985, 2987 \pgfmath@smuggleone 2988 \pgfmathdeclarefunction . . . . . 2982 \pgfmathhint . . . . . 2983 \pgfmathparse . . . . . . . . . . 1506, 3000, 3005, 3008, 3022, 3023 \pgfmathresult . . . . . . . . . . 1507, 2983, 2984, 2986, 2988, 3001, 3009, 3022, 3023 \pgfutil@empty . . . . . 2984 \pgfutil@loop . . . . . 2985 \pgfutil@repeat . . . . 2988 \preceq . . . . . 1782 \prime . . . . . 1761 \printbibliography . 1952 \ProvidesPackage . . . . 2, 15, 33, 58, 226, 490, 551, 558, 620, 733, 748, 1249, 1285, 1398, 1436, 1445, 1466, 1495, 1533, 1631, 1756, 1828, 1911, 1927, 1933, 1955, 2140, 2303, 2316, 2415, 2494, 2577, 2597, 2663, 2688, 2693, 2737, 2903, 2916, 2945, 3026, 3033, 3040, 3045, 3137, 3152, 3243, 3247, 3296, 3338, 3375, 3395	\R . . . . . 2 R \raisebox . . . . . 1900 \relax . . . . . 1965, 2830, 2873, 2875, 2876, 2877, 2985, 2987 \renewcommand . . . . . 1410, 1411, 1412, 1413, 1416, 1417, 1431, 1432, 1729, 1731, 1921, 1922, 2463 \repeat . . . . . 2746 \RequirePackage . . . . . 4, 60, 163, 227, 237, 486, 492, 493, 553, 621, 750, 751, 1251, 1253, 1254, 1290, 1399, 1400, 1402, 1404, 1409, 1418, 1425, 1433, 1437, 1438, 1448, 1449, 1467, 1468, 1469, 1497, 1498, 1546, 1643, 1648, 1649, 1735, 1760, 1829, 1929, 1934, 1935, 1951, 1958, 1959, 1960, 2142, 2306, 2307, 2579, 2580, 2581, 2598, 2600, 2690, 2750, 2752, 2904, 2905, 2906, 2919, 2947, 2978, 3029, 3035, 3042, 3047, 3058, 3059, 3139, 3244, 3251, 3299, 3300, 3302, 3303, 3341, 3342, 3343, 3376, 3396 \right . . . . . 2144, 2583 \RIGHTarrow . . . 3317, 3322 \Rightarrow . . . . . 739, 744	S \sb . . . . . 72, 82, 84, 113, 171, 590, 591, 595, 598, 599, 600, 682, 684, 689, 691, 1345, 1347, 2194, 2197, 2200, 2229, 2235, 2483, 2648, 2657, 3204, 3205, 3206, 3211, 3215, 3216, 3219, 3220, 3223, 3224, 3225 \scriptscriptstyle . . . . . . . 655, 663, 671 \scriptsize 1364, 1840, 1847, 1853, 1915, 1916, 1919, 1920, 2407, 3346, 3399, 3429 \section . . . 52, 1644, 1653 \sectionbreak . . . . . 1645 \seq . 1904, 1905, 1906, 2088, 2089, 2090, 2097 \setbox . . 2001, 2040, 2908 \setcounter 1408, 1646, 1734, 2862, 2884, 2898 \setganttlinklabel . . 1439, 1440, 1441, 1442 \setlength . . . . . 1727, 1728, 1730, 1923, 2930, 3423, 3424, 3434 \setmainfont . . . . . 1401 \setmainlanguage . . . . 487 \setminted . . . . 3064, 3065 \setminus . . . . . 2543 \setul . . . . . 2921 \shoveleft . . . . . 2523 \shoveright . . . . . 2527 \Sigma . . . . . 69, 110, 168, 1314, 1315, 1375 \sigma . . . . . 588, 590, 591 \SLASH . . . . . 1971 \small . . . . . 2118 \sort . . . . . 2995 \sortList . . . . . 2994, 3003 \square . . . . . 555 \stepcounter 2782, 2787, 2792, 2795, 2797, 2801, 2803, 2807, 2809
Q \QS@list . . . . . 2863, 2874, 2878, 2885, 2891, 2896, 2899 \QS@select@equal . . . . . . . . . 2838, 2842 \QS@select@greater . . . . . . . 2839, 2843 \QS@select@smaller . . . . . . . 2834, 2837, 2841 \QS@sort@a . . . . . 2816, 2849, 2870, 2871		



<code>\stichwoerter</code> .. 1549, 1597	180, 181, 182, 309,	1162, 1163, 1164,
<code>\str</code> 367, 566, 575, 2005,	313, 340, 344, 345,	1185, 1186, 1187,
2672, 2681, 3161, 3174	346, 349, 354, 355,	1207, 1208, 1209, 1210
<code>\string</code> ..... 2525, 2535	356, 368, 377, 412,	<code>\umldep</code> ..... 1103
<code>\StrSubstitute</code> . 3002, 3004	415, 418, 426, 427,	<code>\umlHVVHaggreg</code> .....
<code>\strut</code> .... 2251, 2255,	428, 429, 440, 446,	..... 884, 950, 1153
2259, 2263, 2267, 3030	449, 452, 468, 475,	<code>\umlinherit</code> ..... 823,
<code>\subsection</code> ..... 1657	625, 628, 633, 634,	874, 1048, 1093, 1101
<code>\subseteq</code> 3185, 3218, 3225	642, 643, 646, 647,	<code>\umlnote</code> .. 825, 1050, 1217
<code>\subsubsection</code> ..... 1678	1353, 1374, 1375,	<code>\umlreal</code> ..... 821, 882
	1376, 1377, 1380,	<code>\umlsimpleclass</code> .....
	1381, 1382, 1383,	... 777, 778, 779,
<b>T</b>	1550, 1563, 1572,	783, 785, 786, 787,
<code>\tableofcontents</code> ....	1585, 1652, 1656,	816, 969, 970, 971,
..... 1732, 1966	1660, 1667, 1670, 2177	1039, 1091, 1092, 1183
<code>\TeX</code> ..... 2361	<code>\tmp</code> ..... 2545	<code>\umlstatic</code> .... 976, 1010
<code>\text</code> ..... 82, 84,	<code>\TmpPlaceEight</code> ..... 2620	<code>\umluniaggreg</code> ..... 1046
187, 2505, 3346, 3399	<code>\TmpPlaceFive</code> ..... 2617	<code>\umluniassoc</code> .....
<code>\textbf</code> ..... 702, 720,	<code>\TmpPlaceFour</code> ..... 2616	... 797, 822, 1047,
1256, 1551, 1787,	<code>\TmpPlaceNine</code> ..... 2621	1171, 1172, 1191, 1192
1796, 1807, 1816,	<code>\TmpPlaceOne</code> ..... 2613	<code>\umlVHuniassoc</code> .. 798, 799
1977, 1990, 2025,	<code>\TmpPlaceSeven</code> ..... 2619	<code>\umlVHVdep</code> ..... 791,
2077, 2094, 2110, 2468	<code>\TmpPlaceSix</code> ..... 2618	792, 794, 795, 978, 979
<code>\textcolor</code> 1899, 3075, 3345	<code>\TmpPlaceTen</code> ..... 2622	<code>\umlVHVinherit</code> 774, 775,
<code>\textit</code> ... 1096, 1138,	<code>\TmpPlaceThree</code> ..... 2615	780, 781, 788, 789,
1139, 1140, 1141,	<code>\TmpPlaceTwo</code> ..... 2614	947, 948, 972, 973,
1564, 2120, 2513, 2573	<code>\TmpScale</code> ..... 2633	1151, 1152, 1189, 1190
<code>\textsc</code> ..... 1762	<code>\TmpTransitionEight</code> .	<code>\umlVHVreal</code> .....
<code>\textsf</code> ..... 2094	..... 2609, 2630	... 941, 942, 1212, 1213
<code>\textstyle</code> .... 2214, 2246	<code>\TmpTransitionFive</code> ..	<code>\UParrow</code> ..... 3319
<code>\texttt</code> 1279, 1762, 1896,	..... 2606, 2627	<code>\url</code> ..... 469, 2128
1897, 1898, 1899, 3399	<code>\TmpTransitionFour</code> ..	<code>\usemintedstyle</code> .... 3061
<code>\textwidth</code> .... 1559, 1923	..... 2605, 2626	<code>\usetikzlibrary</code> .. 61,
<code>\thematik</code> ..... 1562, 1591	<code>\TmpTransitionNine</code> ..	494, 1252, 1470,
<code>\thepage</code> ..... 1429, 1918	..... 2610, 2631	1830, 2599, 2907,
<code>\theparagraph</code> ..... 1407	<code>\TmpTransitionOne</code> ...	2948, 2979, 3304, 3392
<code>\thesection</code> ..... 1644	..... 2602, 2623	
<code>\Theta</code> ..... 2147	<code>\TmpTransitionSeven</code> .	<b>V</b>
<code>\thinspace</code> ..... 3399	..... 2608, 2629	<code>\value</code> ..... 2744
<code>\tikz</code> ..... 1900	<code>\TmpTransitionSix</code> ...	<code>\varepsilon</code> ..... 567,
<code>tikz: bbaum</code> ..... 26	..... 2607, 2628	578, 579, 1300,
<code>tikz: li binaer baum</code> .... 24	<code>\TmpTransitionTen</code> ...	2194, 2200, 2232, 2243
<code>\tikzchildnode</code> ..... 510	..... 2611, 2632	<code>\vbox</code> ..... 2001, 2040
<code>\tikzparentnode</code> ..... 510	<code>\TmpTransitionThree</code> .	<code>\vfill</code> .... 1609, 1691,
<code>\tikzset</code> ..... 101,	..... 2604, 2625	1698, 1722, 1743, 3030
148, 214, 495, 521,	<code>\TmpTransitionTwo</code> ...	<code>\vrule</code> ..... 3256, 3260
1471, 1831, 2638,	..... 2603, 2624	<code>\vspace</code> ... 1687, 1708,
2764, 2968, 3141, 3377	<code>\TmpX</code> ..... 2634	1712, 1716, 1729,
<code>\tikzumlset</code> ..... 3305	<code>\TmpY</code> ..... 2635	2063, 2081, 3269, 3271
<code>\times</code> ..... 212	<code>\today</code> ..... 1916	
<code>\tiny</code> . 1262, 1268, 1274,	<code>\ttfamily</code> ..... 2922	<b>X</b>
1619, 1899, 1970, 3086		<code>\xappto</code> .. 3006, 3012, 3016
<code>\titel</code> ..... 1680, 1737		<code>\xdef</code> ..... 1500
<code>\titleformat</code> .....	<b>U</b>	<code>\xintApply</code> ..... 2832
... 1405, 1407, 1644	<code>\ul</code> ..... 1257, 2920, 2921	<code>\xintApplyUnbraced</code> ..
<code>\titlespacing</code> ..... 1406	<code>\umlaggreg</code> ..... 1215	2831, 2837, 2838, 2839
<code>\tl</code> . 39, 68, 69, 70, 71,	<code>\umlassoc</code> ..... 1173	<code>\xintCSVtoList</code> ..... 2891
72, 73, 76, 77, 78,	<code>\umlclass</code> .....	<code>\xintFor</code> .....
79, 80, 82, 84, 109,	762, 766, 770, 817,	2781, 2786, 2791,
110, 111, 112, 113,	818, 819, 866, 871,	2796, 2802, 2808, 2849
114, 115, 118, 119,	876, 879, 937, 938,	<code>\xintifEq</code> ..... 2842
120, 121, 122, 123,	939, 944, 945, 974,	<code>\xintifForLast</code> .....
124, 167, 168, 169,	1009, 1040, 1041,	... 2798, 2804, 2810
170, 171, 172, 173,	1044, 1095, 1098,	<code>\xintifGt</code> ..... 2843
176, 177, 178, 179,	1137, 1143, 1144,	

\xintifLt	.....	2841		<b>Z</b>		\zustandsnamens@liste
\xintLength	.....	2816	\ZB	.....	<u>2135</u>	... 1318, 1325, 1326
\xintnthelt	.....	2828	\zB	.....	<u>2134</u>	