

# lehramt-informatik

Hermine Bschlangaul \*

September 9, 2021

## Contents

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

---

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

2.11.9	Erbauer (Builder)	44
2.11.10	Fabrikmethode (Factory Method)	45
2.11.11	Kompositum (Composite)	47
2.11.12	Modell-Präsentation-Steuerung (Model-View-Controller)	47
2.11.13	Stellvertreter (Proxy)	48
2.11.14	Zustand (State)	49
2.12	er.sty	51
2.12.1	Tex-Markup-Beispiel: Komplettes Diagramm	51
2.12.2	Tex-Markup-Beispiel: EER Enhanced Entity-Relation-Modell nach Elmasri/Navante	52
2.12.3	Makro-Kürzel	52
2.13	formale-sprachen.sty	54
2.14	formatierung.sty	57
2.14.1	Schriftarten / Typographie	57
2.14.2	Farben	57
2.14.3	Überschriften	57
2.14.4	Listen	57
2.14.5	Kasten	57
2.14.6	Header	57
2.15	gantt.sty	58
2.16	grafik.sty	59
2.17	graph.sty	60
2.18	hanoi.sty	62
2.19	klassen-konfiguration-aufgabe.sty	63
2.20	klassen-konfiguration-examen.sty	65
2.21	komplexitaetstheorie.sty	67
2.21.1	Makro-Kürzel	67
2.22	kontrollflussgraph.sty	69
2.22.1	Makro-Kürzel	69
2.22.2	TeX-Markup-Beispiel	69
2.22.3	TikZ: pin	69
2.22.4	Umgebungen	70
2.22.5	Makros	71
2.23	kopf-fusszeilen.sty	72
2.24	literatur-dummy.sty	73
2.25	literatur.sty	74
2.26	makros.sty	75
2.27	master-theorem.sty	79
2.27.1	Makro-Kürzel	79
2.28	mathe.sty	83
2.29	minimierung.sty	84
2.30	normalformen.sty	87
2.30.1	Makro-Kürzel	87
2.31	o-notation.sty	90
2.31.1	Makro-Kürzel	90
2.31.2	TeX-Markup-Beispiel: Funktionsgraphen mit pgfplots	90
2.32	petri.sty	91
2.32.1	Makro-Kürzel	91
2.33	potenzmengen-konstruktion.sty	93
2.34	pseudo.sty	95
2.35	pumping-lemma.sty	96
2.36	quicksort.sty	97
2.37	relationale-algebra.sty	100
2.38	rmodell.sty	101
2.38.1	Makro-Kürzel	101
2.39	sortieren.sty	102
2.40	spalten.sty	104
2.41	sql.sty	105

2.42	struktogramm.sty . . . . .	106
2.43	syntax.sty . . . . .	107
2.43.1	Makro-Kürzel . . . . .	107
2.44	syntaxbaum.sty . . . . .	110
2.45	synthese-algorithmus.sty . . . . .	111
2.45.1	Makro-Kürzel . . . . .	111
2.45.2	TeX-Markup Grundgerüst . . . . .	111
2.45.3	TeX-Markup Linksreduktion . . . . .	111
2.45.4	TeX-Markup Rechtsreduktion . . . . .	111
2.45.5	TeX-Markup Relationen formen . . . . .	112
2.46	tabelle.sty . . . . .	115
2.47	typographie.sty . . . . .	116
2.48	uml.sty . . . . .	118
2.49	vollstaendige-induktion.sty . . . . .	120
2.49.1	Makro-Kürzel . . . . .	120
2.50	wasserfall.sty . . . . .	122
2.51	wpkalkuel.sty . . . . .	123
2.51.1	Makro-Kürzel . . . . .	123
<b>3</b>	<b>Index</b>	<b>124</b>

# 1 Klassen

## 1.1 Vorlage Theorie-Teil

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

\begin{document}

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

\chapter{Thema des Theorie-Teils}

\literatur

\end{document}
```

## 1.2 Vorlage Aufgabensammlung

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

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

### 1.3 Vorlage Aufgabe

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

\end{document}
```

## 2 Pakete



## 2.1 abmessung.sty

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

## 2.2 aufgaben-einbinden.sty

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

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

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

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

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

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

31
```

## 2.3 aufgaben-metadaten.sty

```

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

35 \ExplSyntaxOn

\liMetaSetze Setze die Metadaten einer Aufgabe.

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

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

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

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

54 \ExplSyntaxOff

\liAufgabenTitel Momentan eine dummy Makro das die Thematik enthält.

55 \def\liAufgabenTitel#1{}

56

```

## 2.4 automaten.sty

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

### 2.4.1 Endlicher Automat

```
59 \liLadePakete{formale-sprachen}

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

\path (0) edge[above] node{1} (1);
\path (0) edge[loop,above] node{0} (0);
\path (1) edge[loop,above] node{0} (1);
\end{tikzpicture}
```



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

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

- `\liAutomat{}`:  $A = (Z, \Sigma, \delta, E, z_0)$
- `\liAutomat[A_1]{}`:  $A_1 = (Z, \Sigma, \delta, E, z_0)$
- `\liAutomat{zustaende={z_0, z_1, z_2}}`:  $A = (\{z_0, z_1, z_2\}, \Sigma, \delta, E, z_0)$
- `\liAutomat{alphabet={a,b}}`:  $A = (Z, \{a, b\}, \delta, E, z_0)$
- `\liAutomat{delta=d}`:  $A = (Z, \Sigma, d, E, z_0)$
- `\liAutomat{ende={z_0, z_1, z_2}}`:  $A = (Z, \Sigma, \delta, \{z_0, z_1, z_2\}, z_0)$
- `\liAutomat{start=z_1}`:  $A = (Z, \Sigma, \delta, E, z_1)$
- `\liAutomat{dea}`:  $A_{DEA} = (Z, \Sigma, \delta, E, z_0)$
- `\liAutomat{nea}`:  $A_{NEA} = (Z, \Sigma, \delta, E, z_0)$

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

```

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

```

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

```

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

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

```

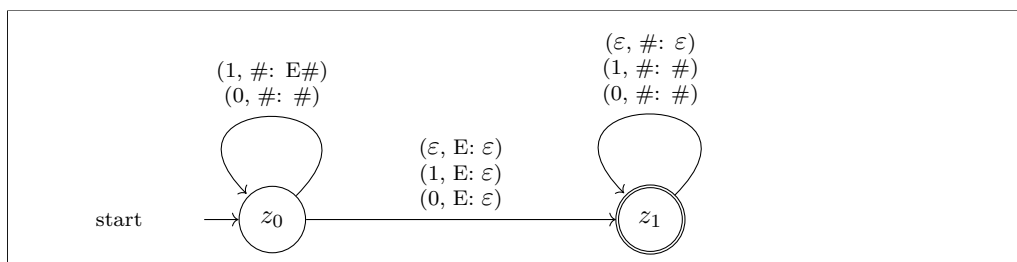
## 2.4.2 Kellerautomat

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

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

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

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



```
\liKellerAutomat \liKellerAutomat[\langle automaten-name \rangle]
{\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]{\zustand-oder-lese}{\schreibe}{\richtung}
Let-Abkürzung: \let\t=\liTuringKante

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

\liTuringUeberfuehrung

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

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

```

## 2.5 basis.sty

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

227 \RequirePackage{xparse}

228 \ExplSyntaxOn

\liLadePakete

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

\liLadeAllePakete

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

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

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

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

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

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

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

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

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

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

```

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

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

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

```

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

```

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

```

Einzelprüfungsnummer / Jahr / Jahreszeit mit Trennzeichen

```

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

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

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

```

```

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

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

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

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

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

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

```

## 2.6 baum.sty

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

486 \RequirePackage{tikz}

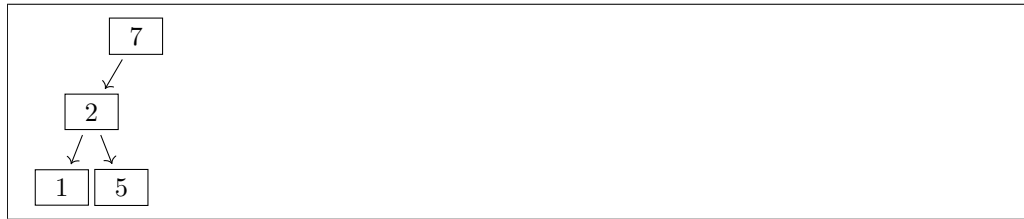
    für li binaer baum
487 \RequirePackage{tikz-qtrees}

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

### 2.6.1 Binärbaum

TikZ-Stil: li binaer baum: Knoten als Kreise, Kanten als Pfeile.

```
\begin{tikzpicture}[li binaer baum]
\Tree
[.7
  [.2
    [.1 ]
    [.5 ]
  ]
\edge[blank]; \node[blank]{};
]
\end{tikzpicture}
```



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



### 2.6.2 AVL-Baum

```
\begin{tikzpicture}[li binaer baum]
\Tree
[.\node[label=-1]{5};
  [.\node[label=-1]{2};
    [.\node[label=0]{1}; ]
    \edge[blank]; \node[blank]{};
  ]
  [.\node[label=0]{7}; ]
]
\end{tikzpicture}
```

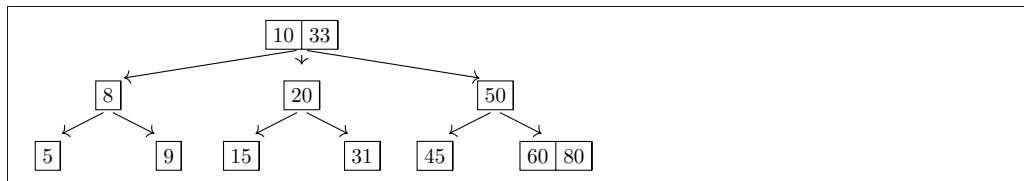


### 2.6.3 B-Baum

```

\begin{tikzpicture}[
  scale=0.8,
  transform shape,
  li bbaum,
  level 1/.style={level distance=10mm,sibling distance=32mm},
  level 2/.style={level distance=10mm,sibling distance=20mm},
]
\node {10 \nodepart{two} 33}
  child {node {8}
    child {node {5}}
    child {node {9}}
  }
  child {node {20}
    child {node {15}}
    child {node {31}}
  }
  child {node {50}
    child {node {45}}
    child {node {60 \nodepart{two} 80}}
  }
;
\end{tikzpicture}

```



```

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

```

## 2.7 checkbox.sty

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

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

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

550
```

## 2.8 chomsky-normalform.sty

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

554 \ExplSyntaxOn

555 \liLadePakete{typographie}
```

### 2.8.1 Makro-Kürzel

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

### 2.8.2 TeX-Markup-Grundgerüst

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

### 2.8.3 Konkretes TeX-Markup-Beispiel

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

\liNichtsZuTun

\item \schrittE{2}

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

\item \schrittE{3}

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

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

```

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

```

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

```

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

```

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

```

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

```

```

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

```

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

```

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

```

```

611 \ExplSyntaxOff
612

```

## 2.9 cpm.sty

```

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

616 \liLadePakete{mathe,typographie}

```

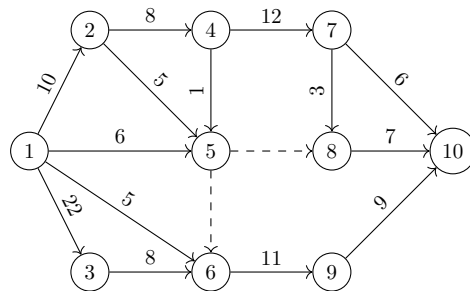
### 2.9.1 Makro-Kürzel

```

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

```

### 2.9.2 TeX-Markup-Beispiel: Graph



```

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

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

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

```

### 2.9.3 TeX-Markup-Beispiel: Ergebnistabelle

```

\begin{tabular}{|l|l|l|l|l|l|l|l|}
\hline
\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
\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}

```

```

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

```

```

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

```

```

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

```



## 2.9.6 Hilfsmakros für Zeitpunkt-Tabelle

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

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

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

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

```
673 \ExplSyntaxOn
```

\liCpmSpaetI Spätester Zeitpunkt, zu dem Ereignis  $i$  eintreten kann

**Let-Abkürzung:** \let\ SZ=\liCpmSpaetI

```
674 \NewDocumentCommand{ \liCpmSpaetI } { 0{i} } {
675 \ifmmode
676 SZ\sb{#1}
677 \else
678 $SZ\sb{#1}$
679 \fi
680 }
```

\liCpmFruehI Frühester Zeitpunkt, zu dem Ereignis  $i$  eintreten kann.

**Let-Abkürzung:** \let\ FZ=\liCpmFruehI

```
681 \NewDocumentCommand{ \liCpmFruehI } { 0{i} } {
682 \ifmmode
683 FZ\sb{#1}
684 \else
685 $FZ\sb{#1}$
```

```

686 \fi
687 }

```

\liCpmFruehErklaerung

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

```

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

```

\liCpmSpaetErklaerung

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

```

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

724 \ExplSyntaxOff
725

```

## 2.10 cyk-algorithmus.sty

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

### 2.10.1 Makro-Kürzel

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

### 2.10.2 TeX-Markup-Beispiel

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

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

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

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

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

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

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

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

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

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

```
740
```

## 2.11 entwurfsmuster.sty

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

### 2.11.1 Namensschema der Entwurfsmuster-Makros:

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

### 2.11.2 Reihenfolge

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

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

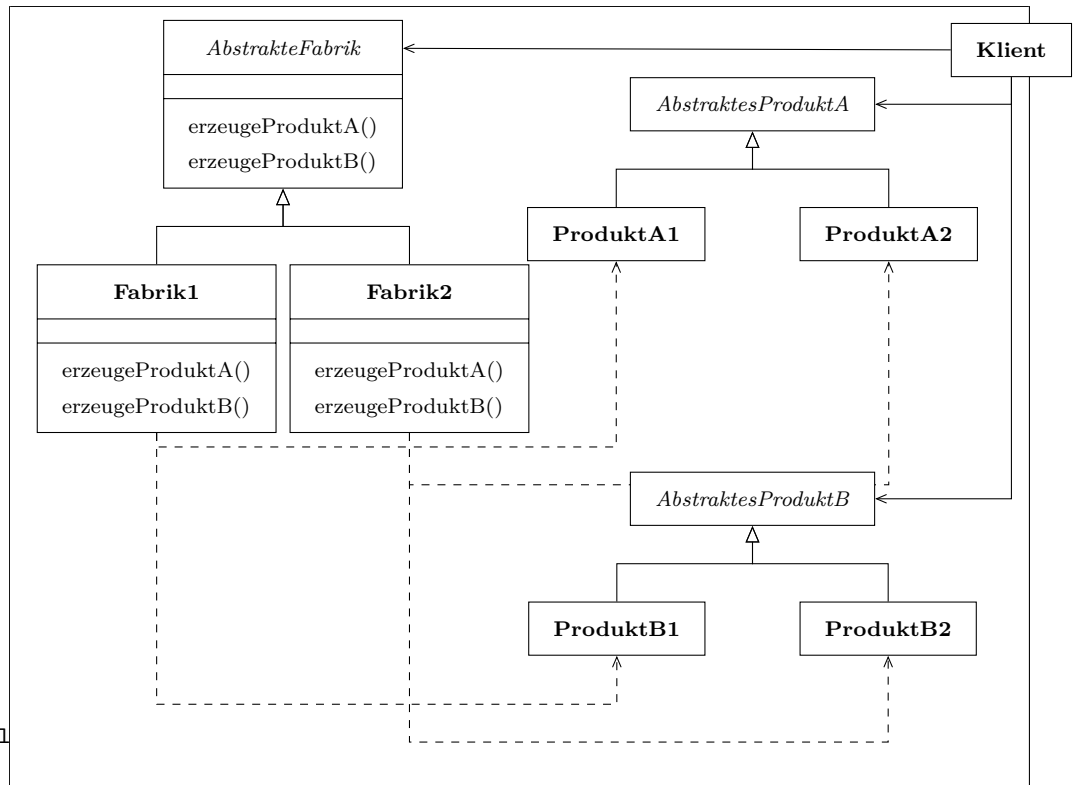
\li@EntwurfsCodeAllgemein Allgemeine Code-Beispiele zu den UML-Diagrammen und Stellvertretern

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

### 2.11.3 Abstrakte Fabrik (Abstract Factory)

sAbstrakteFabrikBeschreibung

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



```

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

```

```

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

```

iEntwurfsAbstrakteFabrikCode

```

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

```

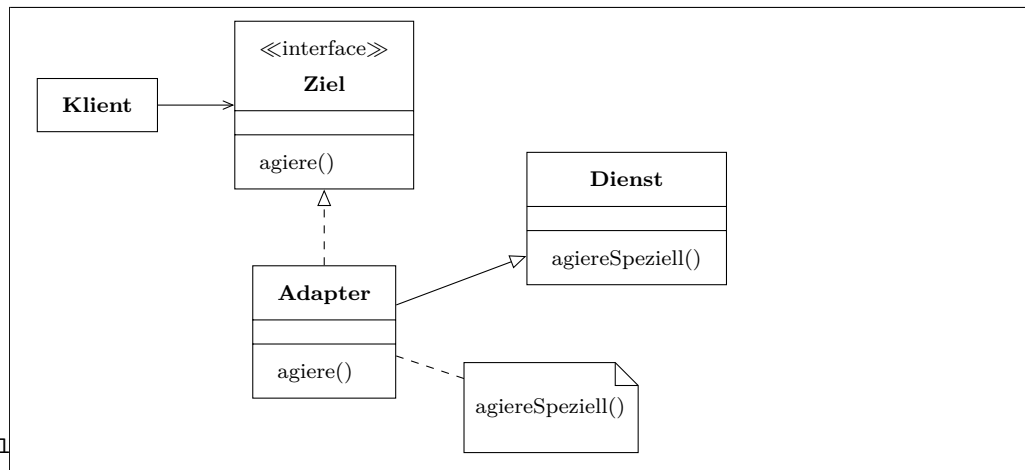
\liEntwurfsAbstrakteFabrik

```

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

```

#### 2.11.4 Adapter



\liEntwurfsAdapterUml

```

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

```

\liEntwurfsAdapterAkteure

**Ziel (Target)** Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.

**Klient (Client)** Der Klient nutzt Dienste über inkompatible Schnittstellen und greift dabei auf adaptierte Schnittstellen zurück.

**Dienst (Adaptee)** Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest definierter Schnittstelle an.

**Adapter** Der Adapter adaptiert die Schnittstelle des Dienstes auf die Schnittstelle zum Klienten.

```

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

```

\liEntwurfsAdapterCode

```

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

```

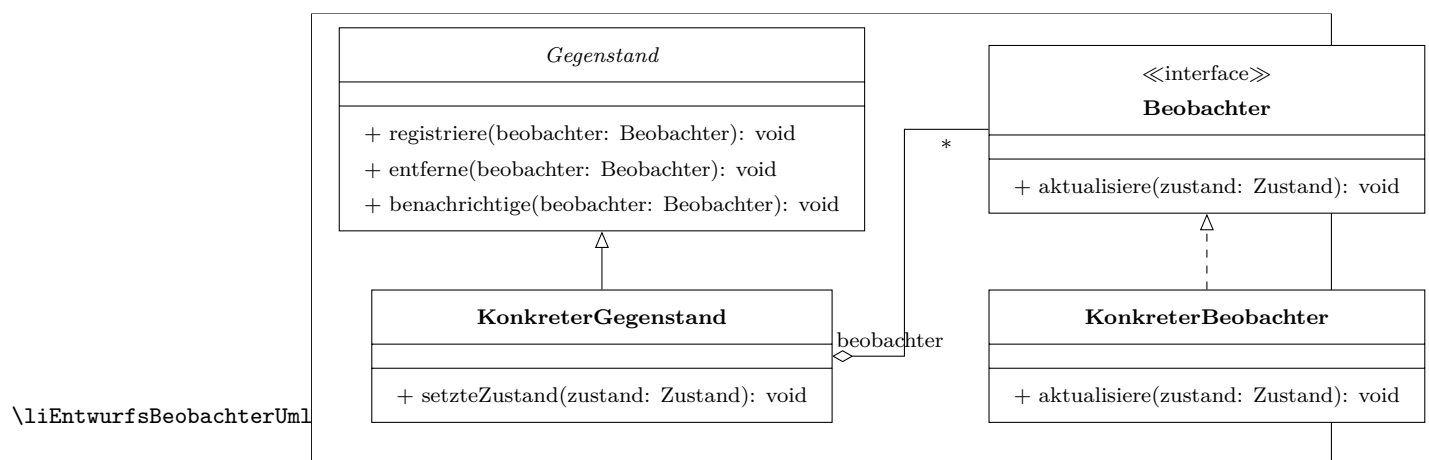
\liEntwurfsAdapter

```

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

```

### 2.11.5 Beobachter (Observer)



```

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

```

```

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

```

\liEntwurfsBeobachterAkteure

**Gegenstand / Subjekt (Subject / Observable)** Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An- und Abmeldung von Beobachtern und eine Schnittstelle zur Benachrichtigung von Beobachtern über Änderungen an.

**Beobachter (Observer)** Die Beobachter (auf Englisch auch subscriber, also „Abonnent“, genannt) definieren eine Aktualisierungsschnittstelle.

**konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)**

Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert den relevanten Zustand und benachrichtigt alle Beobachter bei Zustandsänderungen über deren Aktualisierungsschnittstelle. Es verfügt über eine Schnittstelle zur Erfragung des aktuellen Zustands.

**Konkrete Beobachter (ConcreteObserver)** Konkrete Beobachter verwalten die Referenz auf ein konkretes Subjekt, dessen Zustand sie beobachten und speichern und dessen Zustand konsistent ist. Sie implementieren eine Aktualisierungsschnittstelle unter Verwendung der Abfrageschnittstelle des konkreten Subjekts.

```

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

```



```

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

```

\liEntwurfsBeobachterCode

```

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

```

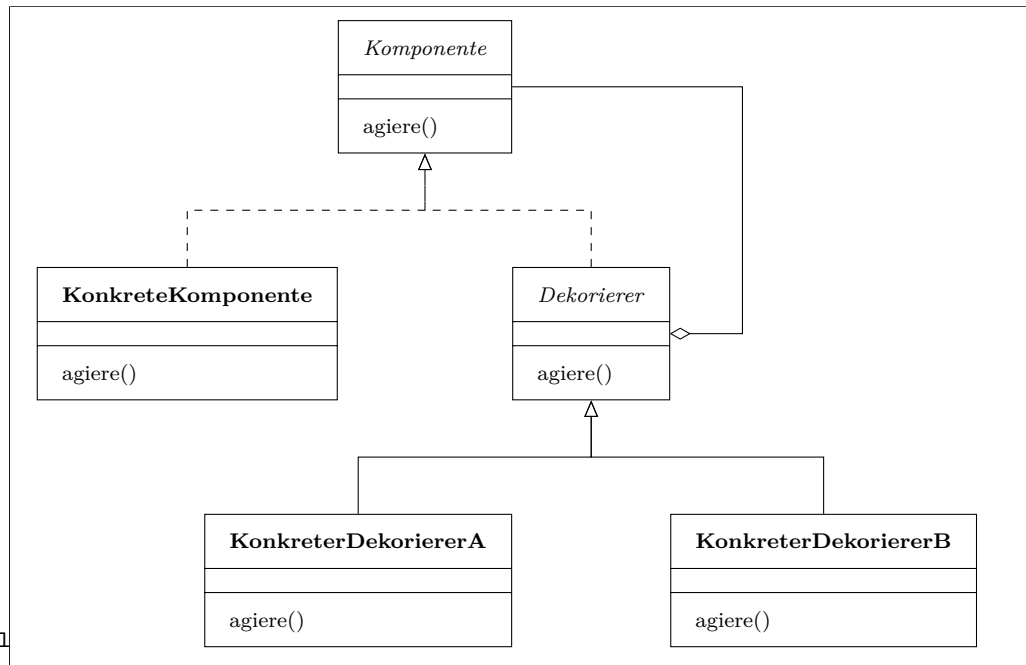
\liEntwurfsBeobachter

```

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

```

## 2.11.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

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

```

```

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

```

\liEntwurfsDekoriererCode

```

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

```

\liEntwurfsDekorierer

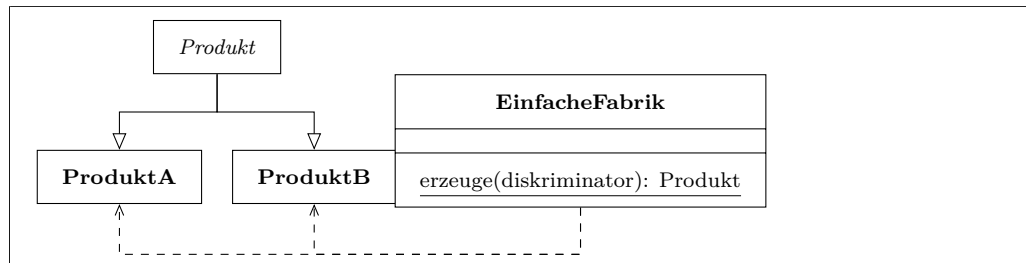
```

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

```

### 2.11.7 Einfache Fabrik (Simple Factory)

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



```

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

```

ntwurfsEinfacheFabrikAkteure Quelle: frei überstetzt aus GoF

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

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

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

```

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

\liEntwurfsEinfacheFabrik

```

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

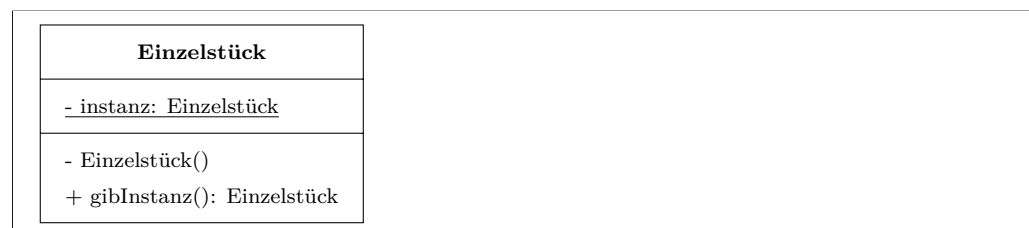
### 2.11.8 Einzelstück (Singleton)

urfsEinzelstueckBeschreibung

```

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

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```

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

iEntwurfsEinzelstueckAkteure Quelle: frei überstetzt aus GoF

**Einzelstück (Singleton)** stellt eine statische Methode bereit, mit deren Hilfe die Klienten nur auf eine einzige Instanz der Klasse zugreifen können.

```

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

```

\liEntwurfsEinzelstueckCode

```

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

```

\liEntwurfsEinzelstueck

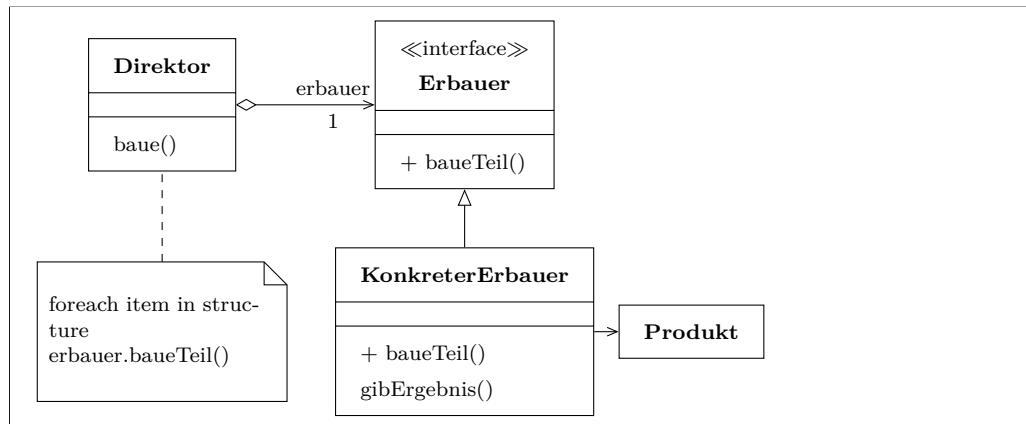
```

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

```

### 2.11.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

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

```

```

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

```

\liEntwurfsErbauerAkteure Quelle: deutsche Wikipedia

**Erbauer** Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der Teile eines komplexen Objektes.

**KonkreterErbauer** Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch Implementierung der Schnittstelle. Außerdem definiert und verwaltet er die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine Schnittstelle zum Auslesen des Produkts.

**Direktor** Der Direktor konstruiert ein komplexes Objekt unter Verwendung der Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom Klienten.

**Produkt** Das Produkt repräsentiert das zu konstruierende komplexe Objekt.

```

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

```

\liEntwurfsErbauer

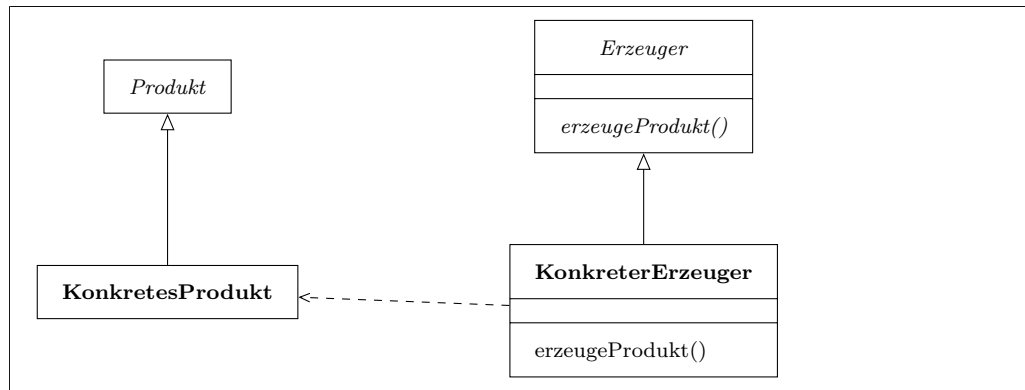
```

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

```

## 2.11.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

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

```

EntwurfsFabrikmethodeAkteure Quelle: deutsche Wikipedia

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

**KonkretesProdukt** KonkretesProdukt implementiert die Produkt-Schnittstelle.

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

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

```

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

```

```

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

```

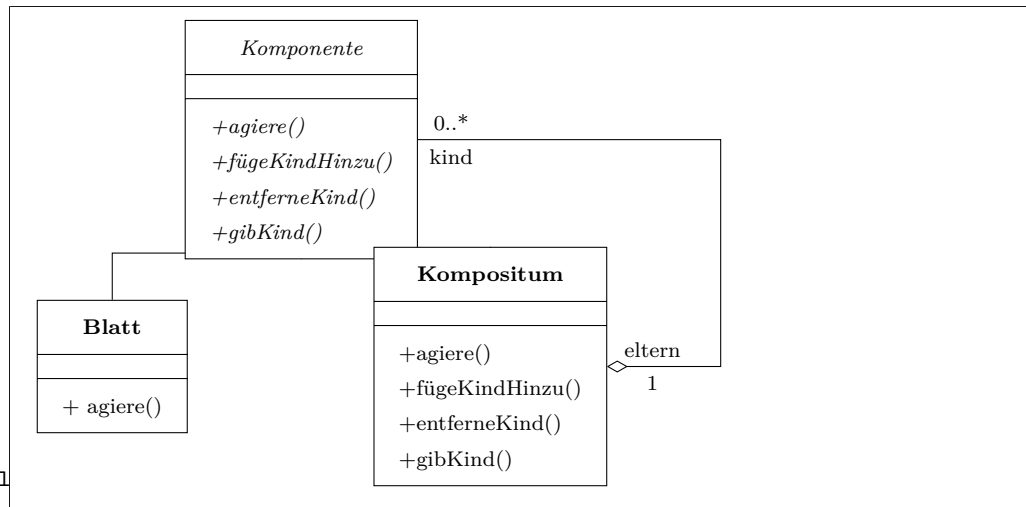
\liEntwurfsFabrikmethode

```

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

```

### 2.11.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

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

```

\liEntwurfsFabrikmethode

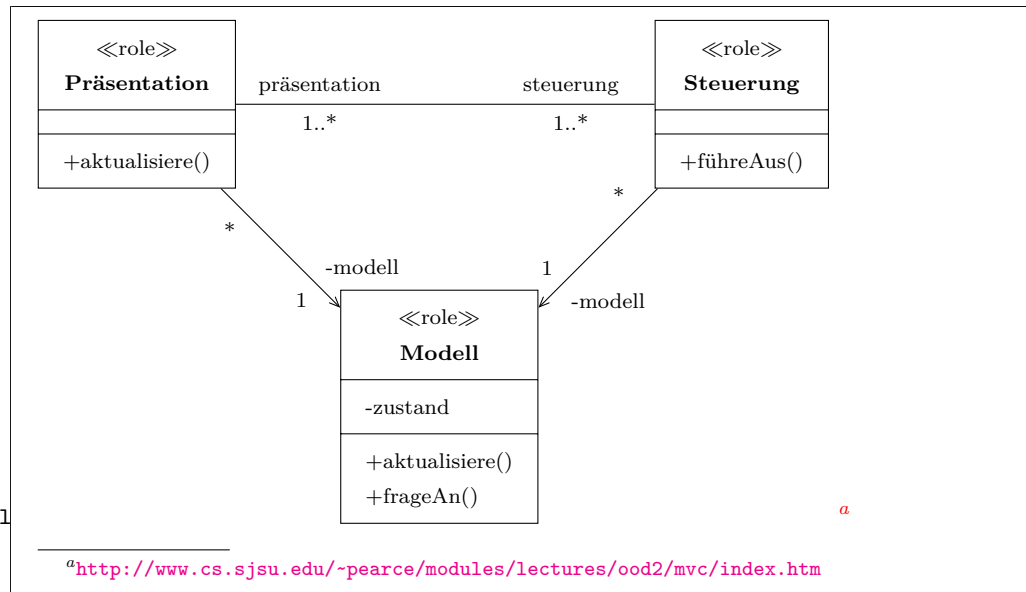
```

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

```

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

ModellPraesentationSteuerungUml



```

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

```

ModellPraesentationSteuerung

```

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

```

### 2.11.13 Stellvertreter (Proxy)

\liEntwurfsStellvertreterUml

```

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

```

\liEntwurfsStellvertreterCode



```

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

```

\liEntwurfsStellvertreter

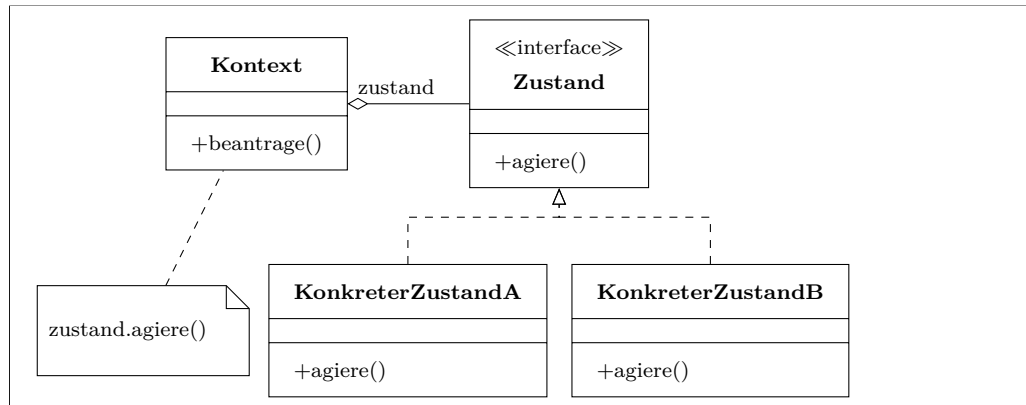
```

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

```

#### 2.11.14 Zustand (State)

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



```

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

```

\liEntwurfsZustandAkteure Quelle: Deutsche Wikipedia

**Kontext (Context)** definiert die clientseitige Schnittstelle und verwaltet die separaten Zustandsklassen.

**State (Zustand)** definiert eine einheitliche Schnittstelle aller Zustandsobjekte und implementiert gegebenenfalls ein Standardverhalten.

**KonkreterZustand (ConcreteState)** implementiert das Verhalten, das mit dem Zustand des Kontextobjektes verbunden ist.

```

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

```

```

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

```

\liEntwurfsZustandCode

```

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

```

\liEntwurfsZustand

```

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

```

```

1241

```

## 2.12 er.sty

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

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

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

```
\begin{tikzpicture}[er2,scale=0.7,transform shape]
% Person
\node[entity] (Person) {Person};
\node[attribute,right=1cm of Person] {\key{E-Mail}} edge (Person);
\node[multi attribute,above left=1cm of Person] {Vornamen} edge (Person);
\node[attribute,left=1cm of Person] {Nachnamen} edge (Person);
\node[attribute,above right=1cm of Person] {Geburtsdatum} edge (Person);

% Kunde
\node[entity,below left=1cm of Person] (Kunde) {Kunde};

% Händler
\node[entity,below right=1cm of Person] (Händler) {Händler};

\node[specialization,below=0.2cm of Person]{is-a}
  edge (Kunde) edge (Händler) edge (Person);

% Transaktion
\node[relationship,below=2cm of Person] (Transaktion) {Transaktion}
  edge node[auto]{1} (Kunde)
  edge node[auto]{1} (Händler);
\node[attribute,below=1cm of Transaktion] {Preis} edge (Transaktion);
\node[attribute,left=1cm of Transaktion,text width=2cm] {Abschlussdatum} edge (Transaktion);
\node[attribute,right=1cm of Transaktion] {Lieferadresse} edge (Transaktion);

% Zahlungsmittel
\node[entity,below=4cm of Händler] (Zahlungsmittel) {Zahlungsmittel}
  edge node[auto]{1} (Transaktion);
\node[attribute,right=1cm of Zahlungsmittel] {Inhaber} edge (Zahlungsmittel);

% Bankverbindung
\node[entity,below left=1cm and 0cm of Zahlungsmittel] (Bankverbindung) {Bankverbindung};
\node[attribute,below left=1cm of Bankverbindung] {\key{IBAN}} edge (Bankverbindung);

% Kreditkarte
\node[entity,below right=3cm and -2cm of Zahlungsmittel]
  (Kreditkarte) {Kreditkarte};

\node[attribute,below left=1cm of Kreditkarte]
  {\key{Nummer}} edge (Kreditkarte);

\node[attribute,below right=1cm of Kreditkarte,text width=2cm]
  {Ablaufdatum} edge (Kreditkarte);

\node[attribute,right=1cm of Kreditkarte]
  {Anbieter} edge (Kreditkarte);

\node[generalization,below=0.8cm of Zahlungsmittel]{is-a}
  edge (Zahlungsmittel) edge (Bankverbindung) edge (Kreditkarte);

% Produkt
\node[entity,below=4cm of Kunde] (Produkt) {Produkt}
  edge node[auto]{1} (Transaktion);
\node[attribute,left=1cm of Produkt] {\key{Bezeichnung}} edge (Produkt);
\node[attribute,below left=1cm of Produkt,text width=2cm] {Beschreibungstext} edge (Produkt);
```

```

\node[attribute,above left=1cm of Produkt] {Bewertung} edge (Produkt);
\end{tikzpicture}

```

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

```

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

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

```

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

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

### 2.12.3 Makro-Kürzel

```

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

```

```
1249 \ExplSyntaxOn
```

```
\liErEntity
```

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

```
\liErRelationship
```

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

```
\liErAttribute
```

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

```
\liErMpEntity mp = marginpar
```

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

```

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

```

□

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

```

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

```

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

```

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

```

```

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

1276 \ExplSyntaxOff
1277

```

## 2.13 formale-sprachen.sty

```

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

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

1284 \RequirePackage{hyperref}

1285 \liLadePakete{mathe,typographie}

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

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

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

1294 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

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

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

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

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

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

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

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

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

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

\liZustandsBuchstabe

1310 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1311 \def\liZustandsBuchstabeGross{Z}

```

```

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

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

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

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

\liAbleitung \liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1323 \def\liAbleitung#1{\directlua{formale_sprachen.formatiere_ableitung('#1')}}$}

liProduktionsRegeln \begin{liProduktionsRegeln}[P_1]
    S -> S A B | EPSILON,
    B A -> A B,
    A A -> a a,
    B B -> b b
\end{liProduktionsRegeln}

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

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

\liZustandsnameTiefgestellt Automatisch tiefgestellte Nummerierung \z1
Let-Abkürzung: \let\z=\liZustandsnameTiefgestellt
1337 \def\liZustandsnameTiefgestellt#1{
1338   \ifmmode
1339     \liZustandsBuchstabe\sb{#1}
1340   \else
1341     $\liZustandsBuchstabe\sb{#1}$
1342   \fi
1343 }

1344 \ExplSyntaxOn

\liAusdruck \liAusdruck[L_2]{a_1,a_2,\dots,a_n}{n \in N}:  $L_2 = \{a_1, a_2, \dots, a_n \mid n \in N\}$ 
    Ohne „=: \liAusdruck[] {x}{y}:  $\{x \mid y\}$ 
    Regulärer Ausdruck zum Konvertieren:
    \$(.*) += +\\{ *(.*) ( *\\, *)?\\ ( *\\, *)?(.*) *\\}\\$
    \liAusdruck[$1]{$2}{$5}

```

```

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

\liFlaci Link zur flaci.com Website: \liFlaci{Grxk1oczg}:
    Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte Automaten, Compiler und Interpreter)
    Ein Projekt der Hochschule Zittau/Görlitz und der Pädagogischen Hochschule Schwyz: flaci.com/Grxk1oczg

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

\liGrammatik \liGrammatik[\langle grammatik-name \rangle]{\langle variablen=V,alphabet=\Sigma,produktionen=P,start=S \rangle}
    \liGrammatik{variablen={}, alphabet={}}

    • \liGrammatik{}:  $G = (V, \Sigma, P, S)$ 
    • \liGrammatik[G_1]{}:  $G_1 = (V, \Sigma, P, S)$ 
    • \liGrammatik{variablen={S,A,B}}:  $G = (\{S, A, B\}, \Sigma, P, S)$ 
    • \liGrammatik{alphabet={a,b}}:  $G = (V, \{a, b\}, P, S)$ 
    • \liGrammatik{start=X}:  $G = (V, \Sigma, P, X)$ 

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

```



## 2.14 formatierung.sty

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

### 2.14.1 Schriftarten / Typographie

```
1393 \RequirePackage{mathpazo}
1394 \RequirePackage[no-math]{fontspec}
1395 \setmainfont{texgyrepagella}
1396 \setsansfont{QTAncientOlive}
1397 \RequirePackage{sectsty}
1398 \allsectionsfont{\fontspec{QTAncientOlive}}
```

### 2.14.2 Farben

```
1399 \RequirePackage{xcolor}
1400 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

### 2.14.3 Überschriften

```
1401 \RequirePackage{titlesec}
1402 \titleformat{\chapter}[display]{\sffamily\bfseries}{\Opt}{\LARGE}
1403 \titlespacing{\chapter}{0pt}{0pt}{*1}
1404 \titleformat{\paragraph}[hang]{\sffamily\normalsize\bfseries}{\theparagraph}{1em}{\}
1405 \setcounter{secnumdepth}{0}
```

### 2.14.4 Listen

```
1406 \RequirePackage{paralist}
1407 \renewcommand\labelitemi{-}
1408 \renewcommand\labelitemii{-}
1409 \renewcommand\labelitemiii{-}
1410 \renewcommand\labelitemiv{-}
1411 % Counter: enumi enumii enumiii enumiv
1412 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1413 \renewcommand{\labelenumi}{(\alph{enumi})}
1414 \renewcommand{\labelenumii}{(\roman{enumii})}
```

### 2.14.5 Kasten

```
1415 \RequirePackage{mdframed}
1416 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1417 \NewDocumentEnvironment { liKasten }{ } {
1418   \begin{mdframed}[backgroundcolor=white!0]
1419 } {
1420   \end{mdframed}
1421 }
```

### 2.14.6 Header

```
1422 \RequirePackage{fancyhdr}
1423 \fancyhead[L,C,R]{}
1424 \fancyfoot[L]{}
1425 \fancyfoot[C]{}
1426 \fancyfoot[R]{\thepage}
1427 \pagestyle{fancy}
1428 \renewcommand{\headrulewidth}{0pt}
1429 \renewcommand{\footrulewidth}{0pt}
1430
```

## 2.15 gantt.sty

```

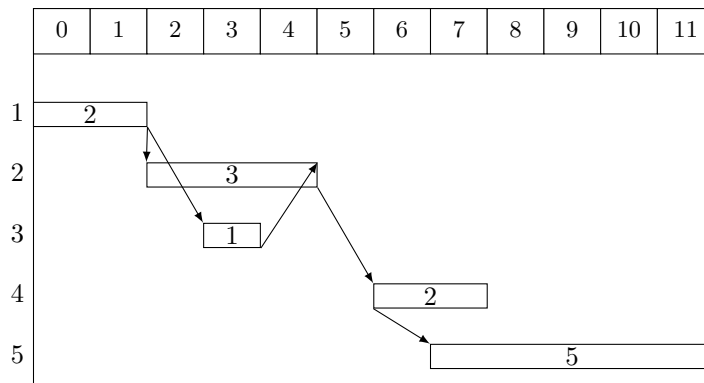
1431 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1432 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\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}

```



```

1433 \RequirePackage{tikz-uml}
1434 \RequirePackage{pgfgantt}
1435 \setganttlinklabel{f-s}{}
1436 \setganttlinklabel{s-s}{}
1437 \setganttlinklabel{f-f}{}
1438 \setganttlinklabel{s-f}{}

1439

```

## 2.16 grafik.sty

```
1440 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1441 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1442 TikZ, um Grafiken zeichnen zu können.]
1443 \RequirePackage{tikz}
1444
```

## 2.17 graph.sty

```

1445 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1446 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1447 \RequirePackage{tikz}

```

Für die die Adjazenzliste (`xrightrightarrow`)

```

1448 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

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

```

1449 \RequirePackage{blkarray}
1450 \usetikzlibrary{arrows.meta}

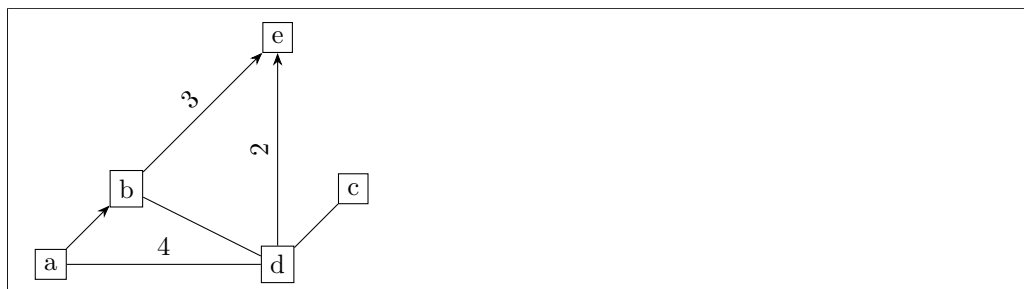
```

```

\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}

```



```

1451 \tikzset{
1452   li graph/.style={
1453     every node/.style={
1454       rectangle,
1455       draw,
1456     },
1457     every edge/.style={
1458       >={Stealth[black]},
1459       draw,
1460     },
1461     every edge/.append style={
1462       every node/.style={
1463         sloped,
1464         auto,
1465       }
1466     }
1467   },
1468   li markierung/.style={
1469     ultra thick,
1470   }
1471 }

```

**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}

```

```

1472 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1473

```

## 2.18 hanoi.sty

```
1474 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1475 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
1476 von Hanoi-Grafiken]
```

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

```
1477 \RequirePackage{tikz}
1478 \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}

1479 \def\li@mset #1[#2]=#3{%
1480 \expandafter\xdef\csname #1#2\endcsname{#3}
1481 }
1482 \def\li@mget #1[#2]{%
1483 \csname #1#2\endcsname
1484 }
1485 \def\li@minc #1[#2]+=#3{%
1486 \pgfmthparse{\li@mget #1[#2]+#3}%
1487 \li@mset #1[#2]=\pgfmthresult
1488 }
1489
1490 \def\liHanoi#1#2{
1491   \edef\li@numdiscs{#1}
1492   \def\li@sequence{#2}
1493   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1494     % init colors
1495     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purple!80!black}{
1496       \li@mset col[\j]={\c};
1497     }
1498     % draw poles and init pole counters
1499     \foreach \j in {1,2,3}{
1500       \li@mset pos[\j]=0
1501       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1502     }
1503     % draw base
1504     \draw (.5,-.5) -- +(3,0);
1505     % draw discs
1506     \foreach[count=\k] \i/\j in \li@sequence{
1507       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*\i/\li@numdiscs,0);
1508       \li@minc pos[\j]+=.5
1509     }
1510   \end{tikzpicture}
1511 }
```

## 2.19 klassen-konfiguration-aufgabe.sty

```
1512 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1513 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-aufgabe}[2021/09/01 Die
1514 Klasse Aufgabe konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

Formatierung muss vor literatur sein, sonst Option clash

```
1515 \liLadePakete{
1516   formatierung,
1517   abmessung,
1518   literatur-dummy,
1519   makros,
1520   aufgaben-metadaten,
1521   kopf-fusszeilen,
1522   mathe
1523 }
1524
1525 % Komischer Option-Clash deshalb ganz am Ende, für die Silbentrennung
1526 \RequirePackage[ngerman]{babel}
1527 \ExplSyntaxOn
1528 \cs_gset:Npn \stichwoerter_formatiert: {
1529   \tl_if_empty:NTF \g_auf_stichwoerter_tl {} {
1530     \textbf{Stichwörter:} ~
1531     \g_auf_stichwoerter_tl
1532     \par
1533   }
1534 }
1535 \cs_gset:Npn \horizontale_linie: {
1536   \par
1537   \noindent
1538   \rule{\textwidth}{0.8pt}
1539   \par
1540 }
1541 \cs_gset:Npn \thematik_formatiert: {
1542   \tl_if_empty:NTF \g_auf_thematik_tl {} {
1543     \textit{
1544       ( \g_auf_thematik_tl )
1545     }
1546   }
1547 }
```

\liAufgabenMetadaten

```
1548 \def\liAufgabenMetadaten#1{
1549   \liMetaSetze{#1}
1550
1551   \noindent
1552   {\large \_gib_einzelpruefung_trenner: \par}
1553
1554   \medskip
1555
1556   \noindent
1557   {\bfseries\Large\_gib_aufgaben_pfad_trenner:}
1558   \hfill \thematik_formatiert:
1559   \par
1560
1561   \medskip
1562
1563   \noindent
1564   {\footnotesize\stichwoerter_formatiert:}
1565
1566   \horizontale_linie:
1567
1568   \bigskip
```

```

1569
1570 \par
1571 % Keine Einrückung
1572 \@afterindentfalse
1573 \@afterheading
1574 }

1575 \AddToHook{enddocument}{
1576 \vfill
1577 {
1578 \tiny
1579 \noindent
1580 Hilf~mit!~
1581 Das~ist~ein~Community~Projekt.~
1582 Verbesserungsvorschläge,~Fehlerkorrekturen,~weitere~Lösungen~sind~
1583 sehr~willkommen~~~egal~wie~~~per~Pull-Request~oder~per~E-Mail~an~
1584 \href{mailto:hermine.bsclangaul@gmx.net}{hermine.bsclangaul@gmx.net}~
1585
1586 \par\noindent
1587 Der~\TeX-Quelltext~dieses~PDFs~kann~unter~folgender~URL~aufgerufen~werden:~
1588
1589 \par\noindent\hfill
1590 \_gib_github_url_href:}
1591 }

1592 \ExplSyntaxOff
1593

```



## 2.20 klassen-konfiguration-examen.sty

```
1594 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1595 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-examen}[2021/09/04 Die
1596 Klasse liSetzeExamen konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

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

```
1597 \liLadePakete{
1598   formatierung,
1599   literatur-dummy,
1600   makros,
1601   aufgaben-metadaten,
1602   abmessung,
1603   typographie
1604 }
```

Formatierung für die Überschriften setzen.

```
1605 \RequirePackage{titlesec}
1606 \titleformat{\section}{\sffamily\huge\filcenter\bfseries}{\thesection}{1em}{}
1607 \newcommand{\sectionbreak}{\clearpage}
1608 \setcounter{secnumdepth}{0}
1609 \liLadeAllePakete
```

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

```
1610 \RequirePackage[ngerman]{babel}
1611 \RequirePackage{standalone}
1612 \ExplSyntaxOn
```

\liSetzeExamenThemaNr

```
1613 \def\liSetzeExamenThemaNr#1{
1614   \tl_gset:Nn \g_auf_examen_thema_nr_tl { #1 }
1615   \section{Thema-Nr.~#1}
1616 }
```

\liSetzeExamenTeilaufgabeNr

```
1617 \def\liSetzeExamenTeilaufgabeNr#1{
1618   \tl_gset:Nn \g_auf_examen_teilaufgabe_nr_tl { #1 }
1619   \subsection{Teilaufgabe-Nr.~#1}
1620 }
```

\liBindeAufgabeEin

```
1621 \def\liBindeAufgabeEin#1{
1622   \tl_gset:Nn \g_auf_examen_aufgabe_nr_tl { #1 }
1623   \input{
1624     \LehramtInformatikRepository /
1625     Staatsexamen /
1626     \g_auf_examen_nummer_tl /
1627     \g_auf_examen_jahr_tl /
1628     \g_auf_examen_monat_tl /
1629     \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
1630       Thema - \g_auf_examen_thema_nr_tl /
1631     }
1632     \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
1633       Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl /
1634     }
1635     Aufgabe - \g_auf_examen_aufgabe_nr_tl .tex
1636   }
1637 }
```

\liAufgabenMetadaten Das Metadaten-Makro überschreiben

```
1638 \def\liAufgabenMetadaten#1{
1639   \liMetaSetze{#1}
1640   \subsubsection{\_gib_aufgaben_titel:}
1641 }
```

```

1642 \setcounter{tocdepth}{4}
1643 \RequirePackage[titles]{tocloft}
1644 \AddToHook{begindocument}{
1645   \pagestyle{empty}
1646   \begin{center}
1647     \large
1648     Erste-Staatsprüfung-für-ein-Lehramt-an-öffentlichen-Schulen \par
1649
1650     \vspace{0.5cm}
1651
1652     Fach-Informatik \par
1653
1654     \vfill
1655
1656     {
1657       \sffamily\bfseries\Huge
1658
1659       \g_auf_examen_jahreszeit_tl \par
1660
1661       \g_auf_examen_jahr_tl \par
1662     }
1663
1664     \vspace{2cm}
1665
1666     {\LARGE \g_auf_examen_nummer_tl \par}
1667
1668     \vspace{0.5cm}
1669
1670     \g_auf_examen_fach_tl \par
1671
1672     \vspace{3cm}
1673
1674     Aufgabenstellungen-mit-Lösungsvorschlägen \par
1675
1676     \end{center}
1677
1678     % für den Abstand vor den section im Inhaltsverzeichnis
1679     % https://tex.stackexchange.com/questions/241445/how-to-control-spacing-in-toc-
1680     % for-different-sections
1680     \setlength{\cftbeforesecskip}{1.5cm}
1681     \setlength{\cftbeforesubsecskip}{0.5cm}
1682     \renewcommand{\cftsubsecafterpnum}{\vspace{\cftbeforesubsecskip}}
1683     \setlength{\cftbeforesubsubsecskip}{0.1cm}
1684
1685     \vfill
1686     \renewcommand{\contentsname}{Aufgabenübersicht}
1687     \tableofcontents
1688     \clearpage
1689 }
1690 \ExplSyntaxOff
1691

```

## 2.21 komplexitaetstheorie.sty

```

1692 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1693 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1694 Setzen von Karps NP-vollständigen Problemen, Hilfsmakros für die
1695 Polynomialzeitreduktion.]

```

### 2.21.1 Makro-Kürzel

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

1696 \liLadePakete{mathe}

Für das Makro `\liProblemBeschreibung` benötigt.

1697 \RequirePackage{mdframed}

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

```
1698 \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
```

1699 \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

```

1700 \def\liProblemBeschreibung#1#2#3{
1701   \begin{mdframed}[
1702     userdefinedwidth=9cm,
1703     align=center,
1704     backgroundcolor=white!0,
1705   ]
1706     \centerline{\large\liProblemName{#1}}
1707
1708     \medskip
1709
1710     \begin{description}
1711       \item[Gegeben:] #2
1712       \item[Frage:] #3
1713     \end{description}
1714   \end{mdframed}
1715 }
```

```

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

1716 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1717 \begin{displaymath}
1718 \liProblemName{#1}
1719 \preceq_{#2}
1720 \liProblemName{#3}
1721 \end{displaymath}
1722 }

\liProblemVertexCover

1723 \def\liProblemClique{%
1724 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1725 Mindestgröße  $n$  in einem gegebenen Graphen.
1726 \footcite[Seite 76]{theo:fs:4}
1727 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1728 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1729 \footcite{wiki:cliquenproblem}
1730 }

\liProblemVertexCover

1731 \def\liProblemVertexCover{%
1732 %
1733 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1734 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1735 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1736 \footcite{wiki:knotenueberdeckung}
1737
1738 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1739 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1740 mindestens einem Knoten aus  $U$  verbunden ist.
1741 \footcite[Seite 78]{theo:fs:4}%
1742 }

\liProblemSubsetSum Kein Karp-21-Problem

1743 \def\liProblemSubsetSum{%
1744 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1745 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1746 \footcite{wiki:teilsommenproblem}
1747 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1748 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1749 größer als eine gegebene obere Schranke  $c$  ist.
1750 \footcite[Seite 74]{theo:fs:4}
1751 }

\liProblemSubsetSum Kein Karp-21-Problem

1752 \def\liProblemSat{%
1753 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1754 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1755 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1756 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1757 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1758 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1759 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1760 aufgestellt werden.
1761 \footcite[Seite 71]{theo:fs:4}
1762 }

1763

```

## 2.22 kontrollflussgraph.sty

1764 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1765 \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 = ...}]
```

```

1766 \RequirePackage{tikz}
1767 \usetikzlibrary{positioning}
1768 \tikzset{
1769   li kontrollfluss/.style={
1770     knoten/.style={
1771       circle,
1772       draw
1773     },
1774     usebox/.style={
1775       draw,
1776       rectangle,
1777       font=\scriptsize,
1778       anchor=west,
1779       align=left,
1780     },
1781     bedingung/.style={
1782       midway,
1783       draw=none,
1784       font=\scriptsize
1785     },
1786     knotenbeschriftung/.style={
1787       draw,
1788       rectangle,
1789       midway,
1790       font=\scriptsize
1791     },
1792     wahr/.style={
1793       thick
1794     },
1795     falsch/.style={
1796       dashed
1797     },
1798     every node/.style={
1799       circle,
1800       draw,
1801     },
1802     every edge/.append style={
1803       every node/.style={
1804         draw=none,
1805         bedingung,
1806       }
1807     },
1808     every path/.style={
1809       draw,
1810       ->,
1811     },
1812     every pin/.style={
1813       draw,
1814       dotted,
1815       rectangle,
1816       pin position=right
1817     },
1818     every pin edge/.style={
1819       dotted,
1820       arrows=-,
1821     }
1822   }
1823 }

```

#### 2.22.4 Umgebungen

liKontrollflussgraph

```

1824 \NewDocumentEnvironment { liKontrollflussgraph } { 0{} } {

```

```

1825 \begin{tikzpicture}[
1826     li kontrollfluss,
1827     #1
1828 ]
1829 } {
1830 \end{tikzpicture}
1831 }

```

## 2.22.5 Makros

`\liAnweisung`

```
1832 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}
```

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

```
1833 \def\liBedingung#1#2{\node[bedingung,#1]{\texttt{#2}}}
```

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

```
1834 \def\liBedingungWahr#1{\node[bedingung,#1]{\texttt{[true]}}}
```

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

```
1835 \def\liBedingungFalsch#1{\node[bedingung,#1]{\texttt{[false]}}}
```

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

```
1836 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}
```

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

```
1837 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw
```

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

```
1838 \ExplSyntaxOn
```

```
1839 \NewDocumentCommand { \liKontrollKnotenPfad } { m }
```

```
1840 {
```

```
1841   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
```

```
1842   \seq_set_map:NNn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
```

```
1843   \seq_use:Nn \l_tmpa_seq { ~~~~ }
```

```
1844 }
```

```
1845 \ExplSyntaxOff
```

```
1846
```

## 2.23 kopf-fusszeilen.sty

```
1847 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1848 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1849 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1850 \ExplSyntaxOn

1851 \fancyhead{}
1852 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1853 \fancyhead[L0,RE]{\scriptsize\today}
1854 \fancyfoot{}
1855 \fancyfoot[LE,R0]{\thepage}
1856 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1857 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1858 \renewcommand{\headrulewidth}{0.4pt}
1859 \renewcommand{\footrulewidth}{0.4pt}
1860 \setlength{\headwidth}{\textwidth}

1861 \ExplSyntaxOff

1862
```



## 2.24 literatur-dummy.sty

```
1863 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1864 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1865 \def\literatur{}

\footcite

1866 % \RequirePackage[stable,multiple]{footmisc}
1867 \NewDocumentCommand{ \footcite } { o m }{}

1868
```

## 2.25 literatur.sty

```
1869 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1870 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1871 \RequirePackage{csquotes}
1872 \RequirePackage[
1873   bibencoding=utf8,
1874   citestyle=authortitle,
1875   backend=biber,
1876 ]{biblatex}
1877 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1878 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1879 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1880 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1881 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1882 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1883 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1884 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1885 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1886 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1887 % To allow footnotes in the heading
1888 \RequirePackage[stable,multiple]{footmisc}

\literatur

1889 \def\literatur{\printbibliography[heading=subbibliography]}

1890
```

## 2.26 makros.sty

```

1891 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1892 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1893 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1894 anderen Paket passen]
1895 \RequirePackage{hyperref}
1896 \RequirePackage{graphicx}
    Für die Umgebung liQuellen benötigt.
1897 \RequirePackage{paralist}
1898 \ExplSyntaxOn

\inhaltsverzeichnis
1899 \def\inhaltsverzeichnis {
1900   \begin{mdframed}
1901     \begin{group}
1902       \let\clearpage\relax
1903       \tableofcontents
1904     \end{group}
1905   \end{mdframed}
1906 }

\memph \mephm (\marginpar and \emph)
1907 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1908 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1909 \newcommand{\liPseudoUeberschrift}[1]{
1910   \bigskip
1911
1912   \par
1913   \noindent
1914   \textsf{\textbf{#1}}
1915
1916   \medskip
1917
1918   \par
1919   % Keine Einrückung
1920   \@afterindentfalse
1921   \@afterheading
1922 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1923 \newcommand{\liBeschriftung}[1]{
1924   \par
1925   \noindent
1926   \medskip
1927   \textbf{#1}:
1928   \medskip
1929   \noindent
1930 }

\hinweis
1931 \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.
1932 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

```

## liEinbettung

```
1933 \NewDocumentEnvironment { liEinbettung }{ o +b } {#2} {}
```

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

```
1934 \NewDocumentEnvironment{ liAntwort } { O{standard} }
1935 {
1936   \str_case:nn {#1} {
1937     {standard} {
1938       \def\beschriftung{}
1939       \mdfsetup{backgroundcolor=gray!1,linecolor=gray}
1940     }
1941     {richtig} {
1942       \def\beschriftung{richtig}
1943       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1944     }
1945     {falsch} {
1946       \def\beschriftung{falsch}
1947       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1948     }
1949     {muster} {
1950       \def\beschriftung{Musterlösung}
1951       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1952     }
1953   }
1954   \ifx\beschriftung\empty\else
1955     \noindent
1956     \textbf{\beschriftung{:}}
1957   \fi
1958   \begin{mdframed}
1959 }
1960 {\end{mdframed}}
```

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

```
1961 \NewDocumentEnvironment{ liAdditum } { o }
1962 {
1963   \begin{mdframed}[backgroundcolor=yellow!5]
1964   \IfNoValueTF {#1}
1965   { \liPseudoUeberschrift{Additum} }
1966   { \liPseudoUeberschrift{Additum:~#1} }
1967 }
1968 {\end{mdframed}}
```

**liExkurs** \begin{liExkurs}[Linear rekursiv]  
Die häufigste Rekursionsform ist die lineare Rekursion, bei der in jedem Fall der rekursiven Definition höchstens ein rekursiver Aufruf vorkommen darf.  
\end{liExkurs}

### Exkurs: Linear rekursiv

Die häufigste Rekursionsform ist die lineare Rekursion, bei der in jedem Fall der rekursiven Definition höchstens ein rekursiver Aufruf vorkommen darf.

```
1969 \NewDocumentEnvironment{ liExkurs }{o +b}{
1970   \vspace{0.2cm}%
1971   \begin{mdframed}[
1972     backgroundcolor=white,
1973     bottomline=false,
1974     innermargin=1cm,
1975     leftline=true,
1976     linecolor=black,
```

```

1977 linewidth=0.1cm,
1978 outermargin=1cm,
1979 rightline=false,
1980 topline=false,
1981 ]
1982 \footnotesize
1983 \noindent%
1984 \textbf{Exkurs:~#1}\par%
1985 \noindent%
1986 #2
1987 \end{mdframed}
1988 \vspace{0.2cm}
1989 }{}

```

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

```

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

```

**Weiterführende Literatur:**

- Quelle 1
- Quelle 2

```

1990 \cs_new:Npn \listen@punkt #1 {\item #1}
1991 \NewDocumentEnvironment { liQuellen }{ +b }
1992 {
1993   \seq_clear_new:N \l_quellen
1994   \seq_set_split:Nnn \l_quellen {\item} {#1}
1995   \seq_remove_all:Nn \l_quellen {}
1996   \begin{mdframed}[backgroundcolor=white!0]
1997     \footnotesize
1998     \noindent
1999     \textsf{\textbf{Weiterführende~Literatur:}}
2000     \medskip
2001     \begin{compactitem}
2002       \seq_map_function:NN \l_quellen {\listen@punkt}
2003     \end{compactitem}
2004   \end{mdframed}
2005   %
2006   \par
2007   \@afterindentfalse
2008   \@afterheading
2009 } {}

```

liLernkartei

```

2010 \NewDocumentEnvironment { liLernkartei }{ m +b }
2011 {
2012   \begin{mdframed}
2013     \footnotesize
2014     \noindent%
2015     \textbf{Lernkarteikarte:~#1}\par%
2016     \noindent%
2017     #2
2018   \end{mdframed}
2019 } {}

```

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

```

2020 \NewDocumentEnvironment { liDiagramm }{ m +b }
2021 {

```

```

2022 \begin{mdframed}[backgroundcolor=white!0]
2023 \small
2024 \noindent%
2025 \textit{\#1}:
2026 \begin{center}
2027 #2
2028 \medskip
2029 \end{center}
2030 \end{mdframed}
2031 } {}

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

2032 \NewDocumentCommand{\liFussnoteUrl} { o m } {
2033 \footnote{\url{\#2}\IfNoValueTF{\#1}{-}{ ( \#1 ) }}
2034 }
2035

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

2036 \NewDocumentCommand{\liFussnoteLink} { o m m } {
2037 \footnote{\href{\#3}{\#2}\IfNoValueTF{\#1}{-}{ ( \#1 ) }}
2038 }

\zB
2039 \def\zB{z.\,B. }

\ZB
2040 \def\ZB{Z.\,B. }

\dh
2041 \def\dh{d.\,h. }

2042 \ExplSyntaxOff
2043
```

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

```

2032 \NewDocumentCommand{\liFussnoteUrl} { o m } {
2033   \footnote{\url{#2}\IfNoValueTF{#1}{}{ (#1)}}
2034 }
2035

```

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

```

2036 \NewDocumentCommand{\liFussnoteLink} { o m m } {
2037   \footnote{\href{#3}{#2}\IfNoValueTF{#1}{-}{ ( #1 ) }}
2038 }

```

 $\setminus zB$ 

2039 \def\zB{z.\,B. }

\ZB

2040 \def\ZB{Z.\,B. }

 $\backslash dh$ 

```
2041 \def\dh{d.\,h. }
```

2042 \ExplSyntaxOff

2043

## 2.27 master-theorem.sty

2044 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2045 \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}

2046 \ExplSyntaxOn

2047 \RequirePackage{amsmath}

\liRundeKlammer

2048 \def\liRundeKlammer#1{

2049 \negthinspace \left( #1 \right)

2050 }

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

2051 \def\liThetaOhneMathe#1{

2052 \Theta \liRundeKlammer{#1}

2053 }

2054 \def\liTheta#1{

2055 \ifmmode

2056 \liThetaOhneMathe{#1}

2057 \else

2058 \$\liThetaOhneMathe{#1}\$

2059 \fi

2060 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
2061 \def\liOmegaOhneMathe#1{
2062 \Omega \liRundeKlammer{#1}
2063 }
2064 \def\liOmega#1{
2065 \ifmmode
2066 \liOmegaOhneMathe{#1}
2067 \else
2068 $\liOmegaOhneMathe{#1}$
2069 \fi
2070 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
2071 \def\liOOhneMathe#1{
2072 \mathcal{O} \liRundeKlammer{#1}
2073 }
2074 \def\liO#1{
2075 \ifmmode
2076 \liOOhneMathe{#1}
2077 \else
2078 $\liOOhneMathe{#1}$
2079 \fi
2080 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
2081 \def\liTOhneMathe#1#2{
2082 \tl_if_blank:nTF {#1}
2083 {}
2084 {#1 \cdot }
2085 T
2086 \liRundeKlammer{\frac{n}{#2}}
2087 }
2088 \def\liT#1#2{
2089 \ifmmode
2090 \liTOhneMathe{#1}{#2}
2091 \else
2092 $\liTOhneMathe{#1}{#2}$
2093 \fi
2094 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
2095 \def\liRekursionsGleichung{
2096 $T(n) = \liT{a}{b} + f(n)$
2097 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
2098 \def\liBedingungEins{
2099 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
2100 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
2101 \def\liBedingungZwei{
2102 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
2103 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
2104 \def\liBedingungDrei{
2105 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
2106 }

2107 \ExplSyntaxOff

```



\liMasterVariablen

```

2108 \def\liMasterVariablen{
2109   \begin{displaymath}
2110     T(n) = \liT{a}{b} + f(n)
2111   \end{displaymath}
2112
2113   \begin{itemize}
2114     \item[$a = $]
2115       Anzahl der rekursiven Aufrufe, Anzahl der Unterprobleme in der
2116       Rekursion
2117       ($a \geq 1$).
2118
2119     \item[$\textstyle{\frac{1}{b}} = $]
2120       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
2121       repräsentiert wird, Anteil an der Verkleinerung des Problems ($b > 1$).
2122
2123     \item[$f(n) = $]
2124       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
2125       die Kombination der Teillösungen entstehen. Eine von $T(n)$
2126       unabhängige und nicht negative Funktion.
2127   \end{itemize}
2128   \footcite{wiki:master-theorem}
2129   \footcite[Seite 19-35]{aud:fs:2}
2130 }
```

\liMasterFaelle

```

2131 \def\liMasterFaelle{
2132   \begin{description}
2133     \item[1. Fall:]
2134       $T(n) \in \liTheta{n^{\log\sb{b}a}}$
2135
2136       \hfill falls \liBedingungEins
2137       für $\varepsilon > 0$
2138
2139     \item[2. Fall:]
2140       $T(n) \in \liTheta{n^{\log\sb{b}a} \cdot \log n}$
2141
2142       \hfill falls \liBedingungZwei
2143
2144     \item[3. Fall:]
2145       $T(n) \in \liTheta{f(n)}$
2146
2147       \hfill falls \liBedingungDrei
2148       für $\varepsilon > 0$
2149       und ebenfalls für ein $c$ mit $0 < c < 1$ und alle hinreichend großen $n$
2150       gilt:
2151       $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$
2152   \end{description}
2153 }
```

\liMasterVariablenDeklaration

```

2154 \def\liMasterVariablenDeklaration#1#2#3{
2155   \begin{description}
2156     \item[Allgemeine Rekursionsgleichung:] \strut
2157
2158     \liRekursionsGleichung
2159
2160     \item[Anzahl der rekursiven Aufrufe ($a$):] \strut
2161
2162     #1
2163
2164     \item[Anteil Verkleinerung des Problems ($b$):] \strut
2165   \end{description}
}
```

```

2166     um  $\frac{1}{\#2}$  also  $b = \#2$ 
2167
2168     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut
2169
2170      $\#3$ 
2171
2172     \item[Ergibt folgende Rekursionsgleichung:] \strut
2173
2174      $T(n) = \text{\liT{\#1}{\#2}} + \#3$ 
2175 \end{description}
2176 }

\liMasterFallRechnung
2177 \def\liMasterFallRechnung#1#2#3{
2178   \begin{description}
2179     \item[1. Fall:] \liBedingungEins:
2180
2181     \#1
2182
2183     \item[2. Fall:] \liBedingungZwei:
2184
2185     \#2
2186
2187     \item[3. Fall:] \liBedingungDrei:
2188
2189     \#3
2190   \end{description}
2191 }

\liMasterExkurs
2192 \def\liMasterExkurs{
2193   \begin{liExkurs}[Master-Theorem]
2194     \liMasterVariablen
2195
2196     \noindent
2197     Dann gilt:
2198
2199     \liMasterFaelle
2200   \end{liExkurs}
2201 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
2202 \def\liMasterWolframLink#1{
2203   Berechne die Rekursionsgleichung auf WolframAlpha:
2204   \href{https://www.wolframalpha.com/input/?i=\#1}{WolframAlpha}
2205 }

2206

```

## 2.28 mathe.sty

```
2207 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2208 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
2209
2210 % for example \ltimes \rtimes
2211 %\RequirePackage{amssymb}
2212 \RequirePackage{amsmath}
2213
2214 %%
2215 % \mlq \mrq
2216 %%
2217 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
2218 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
2219
```

## 2.29 minimierung.sty

```

2220 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2221 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
2222 Minimierungsalgorithmus von einem NEA zu einem DEA]

2223 \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

2224 \def\liFussnote#1{\$x_{#1}\$}

2225 \def\li@fussnote@text#1#2{
2226 \liFussnote{#1}
2227 \quad
2228 {\footnotesize #2}
2229 }

\liFussnoteEinsText
2230 \def\liFussnoteEinsText{
2231 \li@fussnote@text{1}
2232 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
2233 }

\liFussnoteZweiText
2234 \def\liFussnoteZweiText{
2235 \li@fussnote@text{2}
2236 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
2237 }

\liFussnoteDreiText
2238 \def\liFussnoteDreiText{
2239 \li@fussnote@text{3}

```

```

2240 {In weiteren Iterationen markierte Zustände.}
2241 }

\liFussnoteVierText
2242 \def\liFussnoteVierText{
2243   \li@fussnote@text{4}
2244   {...}
2245 }

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



2246 \def\liFussnoten{
2247   \bigskip
2248
2249   \noindent
2250   \liFussnoteEinsText
2251
2252   \noindent
2253   \liFussnoteZweiText
2254
2255   \noindent
2256   \liFussnoteDreiText
2257
2258   \noindent
2259   \liFussnoteVierText
2260 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
2261 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
2262 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
2263 \def\liZustandsPaar#1#2{
2264   $(
2265     \liZustandsPaarVariablenName_#1,
2266     \liZustandsPaarVariablenName_#2
2267   )$
2268 }

liUebergangsTabelle
2269 \renewcommand{\arraystretch}{1.4}
2270 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
2271   \liPseudoUeberschrift{Übergangstabelle}
2272   \begin{center}
2273     \begin{tabular}{r|l|l}
2274       \textbf{Zustandspaar} & \textbf{\#1} & \textbf{\#2} \\ \hline
2275     \end{tabular}
2276   \end{center}
2277 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle:

Minimierungstabelle (Table filling)
2279 \ExplSyntaxOn

```

```

2280 \def\liUeberschriftDreiecksTabelle{
2281   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
2282 }

```

\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. —

```

2283 \def\liMinimierungErklaerung{
2284   %\footcite[Seite~19]{koenig}
2285   \liParagraphMitLinien{
2286     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
2287     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
2288     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
2289     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
2290      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
2291     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
2292     somit~zueinander~( $k - 1$ )-äquivalent,~aber~nicht~ $k$ -äquivalent~und~
2293     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
2294     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
2295   }
2296 }
2297 \ExplSyntaxOff

```

2298

## 2.30 normalformen.sty

```
2299 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2300 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10]
2301 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
2302 Attributhülle]
```

Mathe für textit tex etc, typographie für geschweifte Klammer

```
2303 \liLadePakete{mathe,typographie}
2304 \directlua{
2305   helfer = require('lehramt-informatik-helfer')
2306   normalformen = require('lehramt-informatik-normalformen')
2307 }
```

### 2.30.1 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
```

```
2308 \def\liTeilen#1{
2309   \directlua{tex.print(normalformen.teilen('#1'))}
2310 }
```

```
\liAttributHuelle Let-Abkürzung: \let\ah=\liAttributHuelle
                  \ah{F, \m{A, B}} AttrHülle( $F, \{A, B\}$ ) Regulärer Ausdruck zum Konvertieren
                  AttrHülle((.*)\)\ \ah{$1}
2311 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(\#1)}
2312 \def\liAttributHuelle#1{
2313   \ifmmode
2314     \liAttributHuelleOhneMathe{\#1}
2315   \else
2316     $\liAttributHuelleOhneMathe{\#1}$
2317   \fi
2318 }
```

```
\liAttributMenge Let-Abkürzung: \let\m=\liAttributMenge
2319 \def\liAttributMenge#1{\{ \textit{\#1} \}}
```

liAHuelle

```
2320 \NewDocumentEnvironment{ liAHuelle } { +b } {
2321   \begingroup
2322   \footnotesize
2323   \begin{multline*}
2324     \#1
2325   \end{multline*}
2326   \endgroup
2327 } { }
```

```
\liLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multline
Let-Abkürzung: \let\ahL=\liLinksReduktion
                  \ahL{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}
2328 \def\liLinksReduktion#1#2#3{
2329   \shoveleft{
2330     \liAttributHuelleOhneMathe{FA,
2331       \liAttributMenge{\#1 \string\ #2}} =
2332   } \\\
```

```

2333 \shoveright{
2334 \liAttributMenge{#3}
2335 } \\\
2336 }

```

```

\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( $F, \{A, E \setminus E\}$ ) =  $\{A, E, F, B, D\}$ 
2337 \def\liLinksReduktionInline#1#2#3{%
2338 {%
2339 \footnotesize%
2340 $\liAttributHuelleOhneMathe{F,
2341 \liAttributMenge{#1 \string\ #2}} =
2342 \liAttributMenge{#3}$
2343 }
2344 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahr=\liLinksReduktionInline
\ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
2345 \def\liRechtsReduktionInline#1#2#3#4{%
2346 {%
2347 \footnotesize%
2348 $\liAttributHuelleOhneMathe{
2349 F \setminus
2350 \liFunktionaleAbhaengigkeit{#1}
2351 \def\tmp{#2}\ifx\tmp\empty
2352 \else
2353 \cup \liFunktionaleAbhaengigkeit{#2}
2354 \fi
2355 ,
2356 \liAttributMenge{#3}
2357 } =
2358 \liAttributMenge{#4}$
2359 }
2360 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
\fa{A, B -> C, D}:  $\{A, B\} \rightarrow \{C, D\}$  Regulärer Ausdruck zum Konvertieren:
\$(.*?) \\\rightarrow (.*)\$ \fa{$1 -> $2}
2361 \def\liFunktionaleAbhaengigkeit#1{%
2362 \directlua{normalformen.drucke_funk_abhaengigkeit('#1', false)}%
2363 }

```

```

\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;

```



```

2364 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
2365   \liGeschweifteKlammern
2366   {#1}
2367   {
2368     \begin{align*}
2369       \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
2370     \end{align*}
2371   }
2372   {-0.5cm}
2373   {-1.7cm}
2374 }

\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}

2375 \NewDocumentCommand {\liRelation} { O{R} m } {
2376   $\directlua{
2377     local name = helper.konvertiere_tiefgestellt('#1')
2378     tex.print(name)
2379   }$(\textit{\, #2\,})
2380 }

2381

```

## 2.31 o-notation.sty

2382 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2383 \ProvidesPackage{lehramt-informatik-o-notation}[2021/09/08]

### 2.31.1 Makro-Kürzel

\let\O=\liONotationO

### 2.31.2 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}
```

2384 \ExplSyntaxOn

2385 \RequirePackage{amssymb}

2386 \RequirePackage{pgfplots}

Für echte Teilmenge \subsetneq:  $\subsetneq$

2387 \RequirePackage{amssymb}

\liRundeKlammer

```
2388 \def\liRundeKlammer#1{
2389   \negthinspace \left( #1 \right)
2390 }
```

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

```
2391 \cs_new:Npn \o_notation_O:n #1 {
2392   \mathcal{O} \liRundeKlammer{#1}
2393 }
2394 \def\liONotationO#1{
2395   \ifmmode
2396     \o_notation_O:n { #1 }
2397   \else
2398     $ \o_notation_O:n { #1 } $
2399   \fi
2400 }
```

2401

## 2.32 petri.sty

2402 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2403 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]

### 2.32.1 Makro-Kürzel

\let\t=\liPetriTransitionsName

\let\tp=\liPetriTransPfeile

\let\k=\liPetriErreichKnotenDrei

2404 \RequirePackage{tikz}

2405 \usetikzlibrary{petri,arrows.meta}

Für die Darstellungsmatrix

2406 \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

```
2407 \def\liPetriSetzeSchluessel{%
2408   \def\TmpTransitionOne{}%
2409   \def\TmpTransitionTwo{}%
2410   \def\TmpTransitionThree{}%
2411   \def\TmpTransitionFour{}%
2412   \def\TmpTransitionFive{}%
2413   \def\TmpTransitionSix{}%
2414   \def\TmpTransitionSeven{}%
2415   \def\TmpTransitionEight{}%
2416   \def\TmpTransitionNine{}%
2417   \def\TmpTransitionTen{}%
2418   \pgfkeys{/petri/.cd,
2419     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2420     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2421     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2422     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
```

```

2423 p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
2424 p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2425 p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2426 p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2427 p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2428 p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2429 t1/.store in=\TmpTransitionOne,t1/.default=activated,
2430 t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2431 t3/.store in=\TmpTransitionThree,t3/.default=activated,
2432 t4/.store in=\TmpTransitionFour,t4/.default=activated,
2433 t5/.store in=\TmpTransitionFive,t5/.default=activated,
2434 t6/.store in=\TmpTransitionSix,t6/.default=activated,
2435 t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2436 t8/.store in=\TmpTransitionEight,t8/.default=activated,
2437 t9/.store in=\TmpTransitionNine,t9/.default=activated,
2438 t10/.store in=\TmpTransitionTen,t10/.default=activated,
2439 scale/.store in=\TmpScale,scale/.default=0.5,
2440 x/.store in=\TmpX,x/.default=5,
2441 y/.store in=\TmpY,y/.default=5,
2442 }%
2443 }

```

```

2444 \tikzset{
2445   li petri/.style={
2446     activated/.style={
2447       very thick
2448     },
2449     inhibitor/.style={
2450       {Circle[open,length=2mm,fill=white]}-
2451     }
2452   }
2453 }

```

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

```

2454 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2455 \def\liPetriTransitionsName#1{
2456   \ifmmode
2457     \liPetriTransitionsNameOhneMathe{#1}
2458   \else
2459     $\liPetriTransitionsNameOhneMathe{#1}$
2460   \fi
2461 }

```

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

```

2462 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2463   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2464 }

```

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

```

2465 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2466 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2467 }

```

## 2.33 potenzmengen-konstruktion.sty

```
2468 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2469 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2470 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2471 \liLadePakete{formale-sprachen}
2472 \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}
    }
  }
}
```

```
2473 \def\liZustandsMengenSammlung#1#2{
2474   \liZustandsnameGross{#1}
2475   {
2476     \footnotesize
2477     \liPotenzmenge{
2478       \str_case:nn {#1} {#2
2479       }
2480     }
2481 }
```

```
\liZustandsMengenSammlungNr
```

```
2482 \def\liZustandsMengenSammlungNr#1#2{
2483   \liZustandsnameGross{#1}
2484   {
```

```
2485     \footnotesize
2486     \liZustandsmengeNr{
2487         \str_case:nn {#1} #2
2488     }
2489 }
2490 }

2491 \ExplSyntaxOff
2492
```

## 2.34 pseudo.sty

```

2493 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2494 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen
2495 von Pseudo-Code, Hüll-Paket um algorithm2e]

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

```

Algorithmus 1: Minimaler Spannbaum nach Kruskal
---

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

```

2496 \RequirePackage[german,boxruled]{algorithm2e}

```

```

2497

```

## 2.35 pumping-lemma.sty

2498 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2499 \ProvidesPackage{lehramt-informatik-pumping-lemma}[2021/08/11 Enthält die  
 2500 Definitionen für das Pumping-Lemma in der Regulären Sprache und  
 2501 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2502 \def\liPumpingRegulaer{%
2503   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2504   alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2505    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2506   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2507
2508   \begin{enumerate}
2509     \item  $|v| \geq 1$ 
2510     (Das Wort  $v$  ist nicht leer.)
2511
2512     \item  $|uv| \leq j$ 
2513     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2514
2515     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w$  in  $L$ 
2516     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2517     Sprache  $L$ )
2518   \end{enumerate}
2519
2520   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2521   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2522 }
```

\liPumpingKontextfrei

```
2523 \def\liPumpingKontextfrei{%
2524   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2525   sich alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  zerlegen lassen in
2526    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2527
2528   \begin{enumerate}
2529     \item  $|vx| \geq 1$ 
2530     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2531
2532     \item  $|vwx| \leq j$ 
2533     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2534
2535     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy$  in  $L$  (Für jede
2536     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2537     Sprache  $L$ )
2538   \end{enumerate}
2539 }
```

2540



## 2.36 quicksort.sty

```

2541 % https://tex.stackexchange.com/a/142634
2542 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2543 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2544
2545 %-----
2546 % USAGE:
2547 % \QSinitialize{comma, separated, numerical, values}
2548 % \loop
2549 % \QSpivotStep
2550 % \ifnum\value{pivotcount}>0
2551 %   \QSSortStep
2552 % \repeat
2553 %-----
2554
2555 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2556 \RequirePackage{xintfrac, xinttools}
2557
2558 \RequirePackage{tikz}
2559
2560 %-----
2561 % FIRST PART: TikZ styles and macros for the actual drawing
2562 \newcounter{cellcount}% used for coordinates of the node
2563 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2564
2565 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2566 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2567 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2568 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2569 % by police of LaTeX good conduct ? )
2570 \tikzset{l/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2571          o/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2572          r/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2573 % this is the "b" style as used in the image below
2574          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2575 % nicer:
2576          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2577          g/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2578
2579 % NOTE the b style was originally the same as the r(aised) style apart from
2580 % not being raised, but I find it nicer with a somewhat different
2581 % specification. I have not updated the images though.
2582
2583 % How the nodes are drawn depending on whether on the left of the pivot value
2584 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2585
2586 \def\DecoLEFT #1{%
2587   \xintFor* ##1 in {#1} \do
2588     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2589 }
2590
2591 \def\DecoINERT #1{%
2592   \xintFor* ##1 in {#1} \do
2593     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2594 }
2595
2596 \def\DecoRIGHT #1{%
2597   \xintFor* ##1 in {#1} \do
2598     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2599 }
2600
2601 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2602   \xintFor* ##1 in {#1} \do

```

```

2603     {\stepcounter{cellcount}}%
2604     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2605 }
2606
2607 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2608     \xintFor* ##1 in {#1} \do
2609     {\stepcounter{cellcount}%
2610     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2611 }
2612
2613 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2614     \xintFor* ##1 in {#1} \do
2615     {\stepcounter{cellcount}%
2616     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2617 }
2618
2619 %-----
2620 % SECOND PART: the actual sorting routines.
2621
2622 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2623 \def\QS@sort@b #1{\ifcase #1
2624     \expandafter\QS@sort@empty
2625     \or\expandafter\QS@sort@single
2626     \else\expandafter\QS@sort@c
2627     \fi
2628 }%
2629 \def\QS@sort@empty #1{}
2630 \def\QS@sort@single #1{\QS@Ir {#1}}
2631
2632 % This step is to pick the last as pivot.
2633 \def\QS@sort@c #1%
2634     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2635
2636 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2637 % The trick with \xintApplyUnbraced is that for example when selecting
2638 % the elements smaller than pivot, if we had been using \xintApply we
2639 % would have had at the minimum an empty brace pair. Thus we use the
2640 % "unbraced" variant, but then the \QS@select@smaller has added in
2641 % anticipation a level of braces.
2642 \def\QS@sort@d #1#2{%
2643     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2644     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2645     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2646 }%
2647 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2648 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2649 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2650
2651 %
2652 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2653 %
2654 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2655 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2656 % latter must handle correctly an empty argument.
2657
2658 %-----
2659 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QS@initialize.
2660
2661 % This draws all with suitable highlighting for the newly chosen pivots
2662 % (which will be shown raised)

```

```

2663 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2664             \let\QSIr\DecoINERT
2665             \let\QSIrr\DecoINERT
2666             \let\QSRr\DecoRIGHTwithPivot
2667 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2668             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2669             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2670 }
2671
2672 % This sorts and then draws, showing where the pivot chosen in the previous
2673 % step go. Next time they will have become "inert". If pivotcount is still at
2674 % zero on exit from \QSpivotStep, then this is the signal to stop before
2675 % executing \QSsortStep.
2676 \def\QSsortStep {\def\QSLr {\noexpand\QS@sort@a}%
2677             \def\QSRr {\noexpand\QS@sort@a}%
2678             \def\QSIr {\noexpand\QSIrr}%
2679             \let\QSIrr\relax
2680             \edef\QS@list{\QS@list}%
2681             \let\QSLr\relax
2682             \let\QSRr\relax
2683             \let\QSIr\relax
2684             \edef\QS@list{\QS@list}%
2685             \let\QSLr\DecoLEFT
2686             \let\QSIr\DecoINERTwithPivot
2687             \let\QSIrr\DecoINERT
2688             \let\QSRr\DecoRIGHT
2689 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2690             \setcounter{cellcount}{0}%
2691             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2692 }
2693
2694 \def\QSinitialize #1{%
2695     % first, we convert the comma separated values into a list of braced items
2696     % we use an \edef, and anyhow many \edef's will be used later
2697     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2698     \let\QSRr\DecoRIGHT
2699     % The \QSRr marker mutated to draw the last element as
2700     % pivot and the earlier ones with the suitable style.
2701     %
2702     % The list of marked braced items \QS@list is used both for drawing
2703     % (as here) and for doing the exchange of elements during sort.
2704     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2705             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2706 }
2707

```

## 2.37 relationale-algebra.sty

2708 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2709 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]

2710 \RequirePackage{amsmath}

2711 \RequirePackage{amssymb}

Zum Zeichnen von Operatorenbäumen verwenden wir TikZ

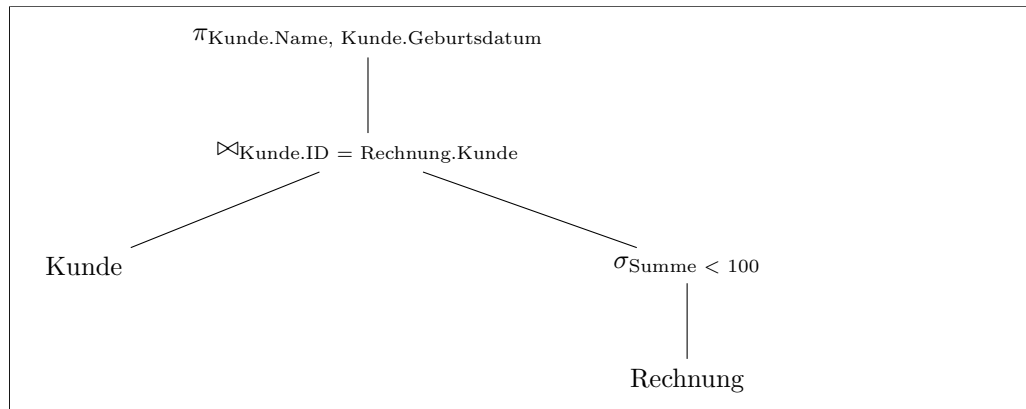
```
\begin{tikzpicture}
  \node
    (pi) {\pi_{\text{Kunde.Name, Kunde.Geburtsdatum}}};

  \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}
```



2712 \RequirePackage{tikz}

2713 \usetikzlibrary{positioning}

Privates Makros, das zwei Querstriche erzeugt.

```
2714 \def\o@join{\setbox0=\hbox{\bowtie}%
2715 \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2716 }
```

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

```
2717 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}
```

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

```
2718 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}
```

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

```
2719 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
```

2720

## 2.38 rmodell.sty

```
2721 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2722 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2723 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2724 Datenbanken.]
2725 \RequirePackage{soul}
```

### 2.38.1 Makro-Kürzel

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

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

```
2726 \def\liPrimaer#1{\ul{#1}}
```

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

```
2727 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

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

```
2728 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2729 \ExplSyntaxOn
2730 \NewDocumentEnvironment { liRmodell }
2731 { +b }
2732 {
2733   \medskip
2734   {
2735     \linespread{2}
2736     \setlength{\parindent}{0pt}
2737     \li@Rmodell@Schrift#1
2738   }
2739   \medskip
2740 } {}
2741 \ExplSyntaxOff
```

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

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

```
2742 \def\liRelationMenge#1#2{
2743   \noindent
2744   #1 : \[ #2 ]\}
2745   \par
2746 }
```

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

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

```
2747 \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}
```

```
2748 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2749
```

## 2.39 sortieren.sty

2750 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2751 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder

2752 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}
}
```



2753 \RequirePackage{tikz}

2754 \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.

2755 \def\liVertauschen#1{

2756 \directlua{

2757 local sortieren = require('lehramt-informatik-sortieren')

2758 sortieren('#1')

2759 }

2760 }

\liSortierPfeil

2761 \def\liSortierPfeil#1#2{

2762 \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);

2763 }

\liSortierPfeilUnten

2764 \def\liSortierPfeilUnten#1#2{

2765 \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);

2766 }

\liSortierMarkierung

2767 \def\liSortierMarkierung#1#2{\node[

2768 draw,

2769 very thick,

2770 fit=(reihe.#1) (reihe.#2),

2771 inner sep=0pt

2772 ] {};

2773 }

2774 \tikzset{

2775 li sortierung zahlenreihe/.style={

2776 draw,

2777 thin,

2778 font=\large,

2779 rectangle split horizontal,

2780 rectangle split,

2781 }

2782 }

```

2783 % https://tex.stackexchange.com/a/140895
2784 \RequirePackage{forest,xstring}
2785 \usetikzlibrary{calc}
2786
2787 \makeatletter
2788 \pgfmathdeclarefunction{strrepeat}{2}{%
2789   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2790   \let\pgfmathresult\pgfutil@empty
2791   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2792     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2793     \advance\pgfmath@count-1\relax
2794   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2795 \makeatother
2796
2797 \def\myNodes{}
2798
2799 \ExplSyntaxOn
2800 \newcommand*\sortList[1]{%
2801   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2802 \ExplSyntaxOff
2803
2804 \forestset{
2805   sort/.code={%
2806     \pgfmathparse{level()>\forestSortLevel}%
2807     \ifnum\pgfmathresult=0
2808       \StrSubstitute{\forestov{content}}{ }{,}{\myList}%
2809       \sortList\myList
2810       \StrSubstitute{\myList}{,}{ }{\myList}%
2811       \pgfmathparse{strrepeat("1",level())}%
2812       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-1!({\forestov{name}}$)
2813         (m\forestov{name}) {\myList}}%
2814       \pgfmathparse{level()=\forestSortLevel}%
2815       \ifnum\pgfmathresult=1
2816         \forestOget{\forestov{@first}}{name}\forestFirst
2817         \forestOget{\forestov{@last}}{name}\forestLast
2818         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2819           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2820       \fi
2821       \ifnum\forestov{@parent}=0\else
2822         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2823       \fi
2824       \gappto\myNodes{;}%
2825     \fi}}
2826
2827 \forestset{sort level/.code=%
2828   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2829   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2830

```

## 2.40 spalten.sty

```
2831 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2832 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2833 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2834 realisiert werden kann.]
2835 \RequirePackage{multicol}
```

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

```
2836 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2837
```



## 2.41 sql.sty

```
2838 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2839 \ProvidesPackage{lehramt-informatik-tabelle}[2021/09/02 Zu Setzen von SQL]

2840 \liLadePakete{syntax}

2841 \RequirePackage{fancyvrb}
2842 \DefineVerbatimEnvironment{liSqlErgebnis}{Verbatim}
2843 {fontsize=\footnotesize}

2844
```

## 2.42 struktogramm.sty

```
2845 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2846 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2847 Paket struktex zum Setzen von Struktogrammen]
2848 \RequirePackage{struktex}
2849
```

## 2.43 syntax.sty

```

2850 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2851 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2852 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2853 \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.

### 2.43.1 Makro-Kürzel

```

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

```

```

2854 \ExplSyntaxOn
2855 \directlua{
2856   syntax = require('lehramt-informatik-syntax')
2857   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2858   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2859   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2860   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2861   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2862   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2863 }
2864 \RequirePackage{hyperref}
2865 \RequirePackage{minted}
2866 % pygmentize -L styles
2867 \usemintedstyle{colorful}
2868 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2869 %\AfterEndEnvironment{minted}{\end{mdframed}}
2870 %\setminted{breaklines=true,linenos}
2871 \setminted{
2872   breaklines=true,
2873   linenos,
2874   fontsize=\footnotesize,
2875 }

```

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

```

2876 \newminted[liJavaAngabe]{java}{
2877   xleftmargin=1cm,
2878   linenos=false
2879 }

```

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

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

```

2880 \def\liJavaCode#1{
2881   \,
2882   \textcolor{blue}{
2883     \mintinline[
2884       fontsize=\normalsize,
2885       breakanywhere % https://github.com/gpoore/minted/issues/31#issuecomment-
2886         458640242
2887     ]{java}|#1|
2888   }
2889 }

```

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

```

2890 \def\liLatexCode#1{\mintinline{latex}|#1|}

```

```

2891 \def\li@GithubLink#1#2{
2892   \begin{flushright}
2893     \tiny
2894     Code-Beispiel~auf~Github~ansehen:~
2895     \href{#1}{\nolinkurl{#2}}
2896   \end{flushright}
2897 }

\liJavaDatei Eine komplette Java-Datei einbinden, die Verzeichnis ./Code/src/main/java/org/bschlangaul
liegt.
2898 \NewDocumentCommand{\liJavaDatei}{ O{firstline=3} m }{
2899   \inputminted[#1]{java}{
2900     \directlua{
2901       syntax.drucke_absoluten_pfad('#2', false)
2902     }
2903   }
2904   \li@GithubLink
2905   {\directlua{syntax.drucke_github_url('#2', false)}}
2906   {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2907 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2908 \NewDocumentCommand{\liJavaTestDatei}{ O{firstline=3} m }{
2909   \inputminted[#1]{java}{
2910     \directlua{
2911       syntax.drucke_absoluten_pfad('#2', true)
2912     }
2913   }
2914   \li@GithubLink
2915   {\directlua{syntax.drucke_github_url('#2', true)}}
2916   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2917 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([
\liJavaExamen$1{$2}{$3}{$4}{$5}
2918 \NewDocumentCommand{\liJavaExamen}{ O{firstline=3} m m m m }{
2919   \inputminted[#1]{java}{
2920     \directlua{
2921       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2922     }
2923   }
2924   \li@GithubLink
2925   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2926   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2927 }

\liAssemblerCode
2929 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2930 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2931   \inputminted{asm}{#1}
2932 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2933 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2934   \inputminted{componentpascal}{#1}
2935 }

```

```

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2936 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2937 \NewDocumentCommand{\liHaskellDatei}{ m }{
2938   \inputminted{haskell}{#1}
2939 }

2940 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2941 \def\liSqlCode#1{\mintinline{sql}|#1|}

2942

```

## 2.44 syntaxbaum.sty

```
2943 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2944 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2945 Syntaxbäumen mit Hilfe des Pakets tikz-qtree]
2946 \RequirePackage{tikz-qtree}
2947
2948 \tikzset{li parsetree/.style={
2949     every internal node/.style={
2950         draw,circle
2951     },
2952     every leaf node/.style={
2953         draw,rectangle
2954     },
2955 }
2956 }
2957
```

## 2.45 synthese-algorithmus.sty

```
2958 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2959 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2960 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2961 Relation in die 3. Normalform]
2962 \liLadePakete{normalformen,mathe,typographie}
2963 \ExplSyntaxOn
```

### 2.45.1 Makro-Kürzel

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

### 2.45.2 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}
```

### 2.45.3 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}}

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

### 2.45.4 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}

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

## 2.45.5 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 Anhängigkeit  $\alpha \rightarrow \beta \in F$  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  $R_\alpha := \alpha \cup \beta$ . —

### 3. Schlüssel hinzufügen

— Falls eines der in Schritt 2. erzeugten Schemata  $R_\alpha$  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  $R_\alpha$ , die in einem anderen Relationenschema  $R_{\alpha'}$  enthalten sind, d. h.  $R_\alpha \subseteq R_{\alpha'}$ . —

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

```
2964 \def\liSyntheseUeberschrift#1{
2965   {
2966     \bfseries
2967     \sffamily
2968     \str_case:nn {#1} {
2969       {1} {Kanonische-Überdeckung}
2970       {1-1} {Linksreduktion}
2971       {1-2} {Rechtsreduktion}
2972       {1-3} {Löschen-leerer-Klauseln}
2973       {1-4} {Vereinigung}
2974       {2} {Relationsschemata-formen}
2975       {3} {Schlüssel-hinzufügen}
2976       {4} {Entfernung-überflüssiger-Teilschemata}
2977     }
```



```

2978 }
2979 }

\liSyntheseErklaerung Let-Abkürzung: \let\erklaerung=\liSyntheseErklaerung
2980 \def\li@synthese@erklaerung@texte#1{
2981   \str_case:nn {#1} {
2982     {1} {
2983       Die~kanonische~Überdeckung~---also~die~kleinst~mögliche~noch~
2984       äquivalente~Menge~von~funktionalen~Abhängigkeiten~kann~in~vier~
2985       Schritten~erreicht~werden.
2986     }
2987     {1-1} {
2988       Führe~für~jede~funktionale~Anhängigkeit~
2989        $\alpha \rightarrow \beta$ ~die~Linksreduktion~durch,~
2990       überprüfe~also~für~alle~
2991        $A \in \alpha$ ,~ob~ $A$ ~überflüssig~ist,~d.h.~ob~
2992        $\beta \subseteqq \text{\liAttributHuelle}\{F, \alpha\}$ .
2993     }
2994     {1-2} {
2995       Führe~für~jede~(verbliebene)~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~die~Rechtsreduktion~durch,~überprüfe~also~für~
2996       alle~ $B \in \beta$ ,~ob~ $B \in \text{\liAttributHuelle}\{F \cup (\alpha \rightarrow \beta), \alpha\}$ ~gilt.~In~diesem~Fall~ist~ $B$ ~auf~der~rechten~Seite~
2997       überflüssig~und~kann~eliminiert~werden,~d.h.~ $\alpha \rightarrow \beta$ ~wird~durch~ $\alpha \rightarrow (\beta \cup B)$ ~
2998       ersetzt.
2999     }
3000     {1-3} {
3001       Entferne~die~funktionalen~Abhängigkeiten~der~Form~ $\alpha \rightarrow \emptyset$ ,~die~im~2.~Schritt~möglicherweise~
3002       entstanden~sind.
3003     }
3004     {1-4} {
3005       Fasse~mittels~der~Vereinigungsregel~funktionale~Abhängigkeiten~
3006       der~Form~ $\alpha \rightarrow \beta_{\{1\}}, \dots, \alpha \rightarrow \beta_{\{n\}}$ ,~so~dass~ $\alpha \rightarrow \beta_{\{1\}} \cup \dots \cup \beta_{\{n\}}$ ~
3007       verbleibt.
3008     }
3009     % Kemper Seite 197
3010     {2} {
3011       Erzeuge~für~jede~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~ein~Relationenschema~ $\mathcal{R}_{\alpha}$ ~
3012        $:= \alpha \cup \beta$ .
3013     }
3014     {3} {
3015       Falls~eines~der~in~Schritt~2.~erzeugten~Schemata~ $\mathcal{R}_{\alpha}$ ~
3016       einen~Schlüsselkandidaten~von~ $\mathcal{R}$ ~bezüglich~ $F$ ~
3017       enthält,~sind~wir~fertig,~sonst~wähle~einen~Schlüsselkandidaten~
3018        $\mathcal{K} \subseteqq \mathcal{R}$ ~aus~und~definiere~folgendes~
3019       zusätzliche~Schema:~ $\mathcal{R}_{\mathcal{K}} := \mathcal{K}$ ~
3020       und~ $\mathcal{F}_{\mathcal{K}} := \emptyset$ 
3021     }
3022     {4} {
3023       Eliminiere~diejenigen~Schemata~ $\mathcal{R}_{\alpha}$ ,~die~in~einem~
3024       anderen~Relationenschema~ $\mathcal{R}_{\alpha'}$ ~enthalten~sind,~d.h.~
3025        $\mathcal{R}_{\alpha} \subseteqq \mathcal{R}_{\alpha'}$ .
3026     }
3027   }
3028 }
3029 \def\liSyntheseErklaerung#1{
3030 {
3031   \itshape
3032   \footnotesize

```

```

3040     \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
3041   }
3042 }

```

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

```

3043 \def\liSyntheseUeberErklaerung#1{
3044   \liSyntheseUeberschrift{#1}\par
3045   \liSyntheseErklaerung{#1}
3046 }

```

```

3047 \ExplSyntaxOff
3048

```

## 2.46 tabelle.sty

```
3049 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3050 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]
3051 \RequirePackage{tabularx}
3052
```

## 2.47 typographie.sty

```

3053 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3054 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
3055 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
3056 formatierung.sty definiert.]
3057 \ExplSyntaxOn

```

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

```
3058 \RequirePackage{fontawesome}
```

`\liErledigt` `\liErledigt:` ☑

```
3059 \let\liErledigt=\faCheckSquareO
```

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

```
3060 \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 —

```

3061 \def\liParagraphMitLinien#1{
3062   \noindent
3063   \vrule height 2pt depth -1.6pt width 0.4cm
3064   \enspace
3065   #1
3066   \enspace
3067   \leaders\vrule height 2pt depth -1.6pt \hfill \null
3068   \par
3069   \medskip
3070 }

```

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

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

```

3071 \def\liGeschweifteKlammern#1#2#3#4{
3072   \par
3073   \medskip
3074   \noindent
3075   #1 \, $= \Bigl\{ \$
3076   \vspace{#3}
3077   #2
3078   \vspace{#4}
3079   \begin{flushright} \$\Bigr\} \$\end{flushright}
3080   \par
3081 }

```

`\liTypoUeberschrift`

```

3082 \def\liTypoUeberschrift#1{
3083   {
3084     \bfseries\sffamily
3085     #1
3086   }
3087 }

```

\liTypoUeberGross

```
3088 \def\liTypoUeberGross#1{
3089   {
3090     \huge
3091     \liTypoUeberschrift{#1}
3092   }
3093 }
```

\liTypoUeberGross

```
3094 \def\liTypoUeberGROSS#1{
3095   {
3096     \Huge
3097     \liTypoUeberschrift{#1}
3098   }
3099 }
```

3100 \ExplSyntaxOff

3101

## 2.48 uml.sty

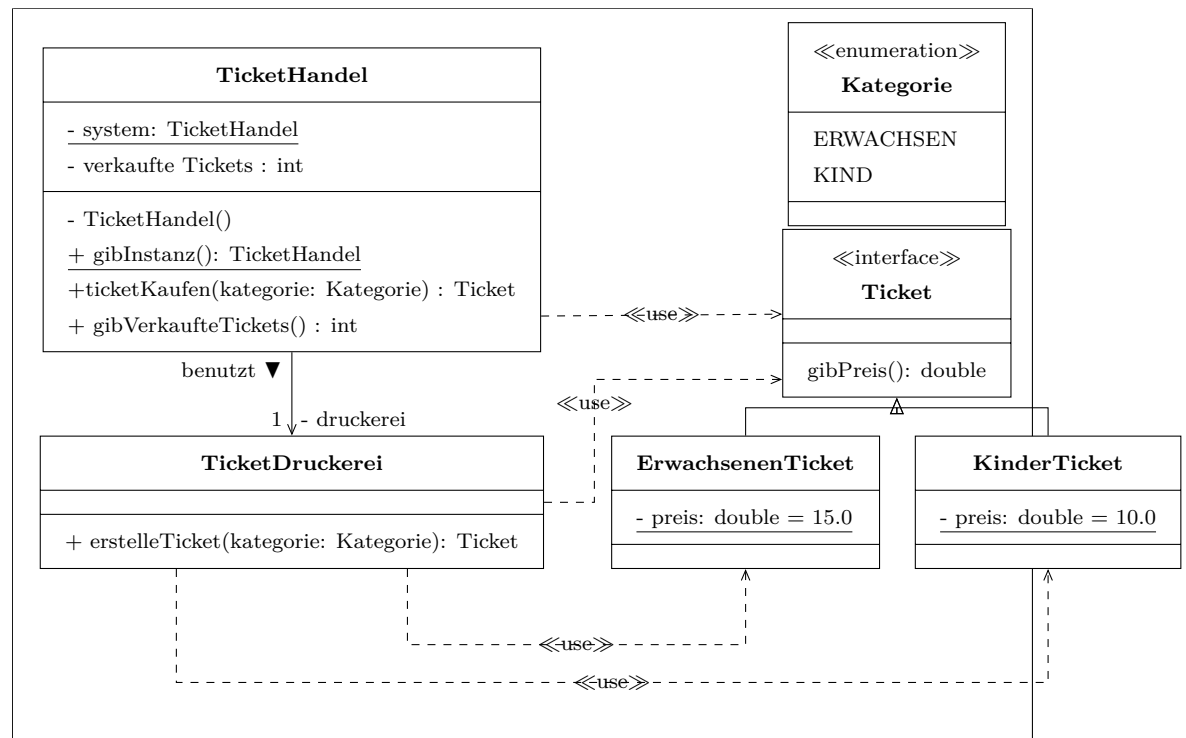
```

3102 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3103 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
3104 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
3105 Erweiterung bereitstellt]

3106 \RequirePackage{tikz-uml}
3107 \RequirePackage{tikz-uml-activity}
3108 % Not compatible with wasysym
3109 %\RequirePackage{mathabx}
3110 \RequirePackage{wasysym}
3111 \usetikzlibrary{positioning}

3112 \tikzumlset{
3113   fill class=white!0,
3114   font=\footnotesize,
3115   fill object=white!0,
3116   fill note=white!0,
3117   fill state=white!0,
3118   % Use case
3119   fill usecase=white!0,
3120   fill system=white!0,
3121 }

```



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

```

3122 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
3123   \def\@liDirLeft{}
3124   \def\@liDirRight{ \RIGHTarrow}
3125   \pgfkeys{/lese/dir/.is choice}
3126   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
3127   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
3128   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{ \LEFTarrow }}}
3129   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
3130
3131   \def\@liPos{above}
3132   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
3133

```

```

3134 \def\@liDistance{0cm}
3135 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
3136
3137 \pgfkeys{/lese/.cd,#1}
3138
3139 \node[\@liPos = \@liDistance of #2-middle] {
3140   \@liDirLeft{\footnotesize#2}\@liDirRight
3141 };
3142 }
3143

```

## 2.49 vollstaendige-induktion.sty

```
3144 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3145 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01]
3146 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
3147 Überschriften für die einzelnen Schritte]
```

### 2.49.1 Makro-Kürzel

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

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

Lade häufig benötigte Pakete

```
3148 \RequirePackage{lehramt-informatik-typographie}
3149 \RequirePackage{lehramt-informatik-mathe}
3150 \RequirePackage{lehramt-informatik-syntax}
3151 \ExplSyntaxOn
```

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

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

```
3152 \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

```
3153 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
3154 \def\liInduktionAnfang{
```

```
3155   \liPseudoUeberschrift{Induktionsanfang}
```

```
3156
```

```
3157   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
3158   \liParagraphMitLinien{
```

```
3159     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
```

```
3160   }
```

```
3161 }
```

\liInduktionVoraussetzung

```
3162 \def\liInduktionVoraussetzung{
```

```
3163   \liPseudoUeberschrift{Induktionsvoraussetzung}
```

```
3164
```

```
3165   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
3166   \liParagraphMitLinien{
```

```
3167     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ .
```

```
3168   }
```

```
3169 }
```

\liInduktionSchritt

```
3170 \def\liInduktionSchritt{
```

```
3171   \liPseudoUeberschrift{Induktionsschritt}
```

```
3172
```

```
3173   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
3174   \liParagraphMitLinien{
```

```
3175     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
```

```
3176     auch~ $A(n=k+1)$ ~wahr~sein~muss.
```

```
3177   }
```

```
3178 }
```

```
3179 \ExplSyntaxOff
```

```
3180
```

## 2.50 wasserfall.sty

```
3181 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3182 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
3183 \RequirePackage{tikz}
3184 \tikzset{wasserfall/.style={
3185   >=stealth,
3186   node distance = 2mm and -8mm,
3187   start chain = A going below right,
3188   every node/.style = {
3189     draw,
3190     text width=24mm,
3191     minimum height=12mm,
3192     align=center,
3193     inner sep=1mm,
3194     fill=white,
3195     drop shadow={fill=black},
3196     on chain=A
3197   },
3198 }}
3199 \usetikzlibrary{chains,positioning,shadows}
3200
```

## 2.51 wpkalkuel.sty

```
3201 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3202 \ProvidesPackage{lehramt-informatik-wpkalkuel}[2020/06/13]
```

### 2.51.1 Makro-Kürzel

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

```
3203 \RequirePackage{amsmath}
3204 \ExplSyntaxOn
```

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

```
3205 \def\liWpKalkuelOhneMathe#1#2{
3206   \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)
3207 }
3208 \def\liWpKalkuel#1#2{
3209   \ifmmode
3210     \liWpKalkuelOhneMathe{#1}{#2}
3211   \else
3212     $\liWpKalkuelOhneMathe{#1}{#2}$
3213   \fi
3214 }
```

**\MatheEnv**

```
3215 \def\MatheEnv#1{
3216   \medskip
3217
3218   \hspace{1em}#1
3219
3220   \medskip
3221 }
```

**\Mathe**

```
3222 \def\Mathe#1{
3223   \MatheEnv{${#1}$}
3224 }
```

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

```
3225 \def\liWpEquivalent#1{
3226   \MatheEnv{${\equiv}\hspace{1em}${#1}$}
3227 }
```

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

```
3228 \newlength{\@Skip@Erklaerung@Reset}
3229 \def\liWpErklaerung#1{
3230   \setlength{\@Skip@Erklaerung@Reset}{\leftskip}
3231   \setlength{\leftskip}{0.5cm}
3232
3233   \par
3234   \noindent
3235   {
3236     \scriptsize
3237     #1
3238   }
3239   \par
3240
3241   \setlength{\leftskip}{\@Skip@Erklaerung@Reset}
3242 }
```

```

3243 \def\liWpErklaerungVerzweigung{
3244   $\liWpKalkuel0hneMathe{if~\{~b~\}~then~\{~a1~\}~else~\{~a2~\}}{Q}
3245   \equiv
3246   (b \land \liWpKalkuel0hneMathe{a1}{Q})
3247   \lor
3248   (\neg b \land \liWpKalkuel0hneMathe{a2}{Q})$
3249 }

3250 \ExplSyntaxOff

3251

```

### 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		\beginngroup 1901, 2321, 2789
\# ..... 114	\_ ..... 2331, 2341	\beschriftung ..... ..... 1938, 1942, 1946, 1950, 1954, 1956
\, .. 358, 417, 472, 699, 717, 1349, 1351, 2039, 2040, 2041, 2379, 2881, 2888, 3075	A \addbibresource ..... 1877, 1878, 1879, 1880, 1881, 1882, 1883, 1884, 1885, 1886	\beta ..... 2989, 2992, 2996, 2997, 2998, 3001, 3011, 3012, 3013, 3018, 3019
\@Skip@Erklaerung@Reset ... 3228, 3230, 3241	\AddToHook .... 1575, 1644	\bf ..... 2566, 2567, 2568
\@afterheading ..... ... 1573, 1921, 2008	\advance ..... 2793	\bfseries ..... .. 558, 1402, 1404, 1557, 1606, 1657, 2566, 2572, 2574, 2576, 2577, 2966, 3084
\@afterindentfalse .. ... 1572, 1920, 2007	\AfterEndEnvironment 2869	\Bigl ..... 3075
\@liDirLeft 3123, 3128, 3140	\allsectionsfont ... 1398	\Bigr ..... 3079
\@liDirRight 3124, 3126, 3127, 3128, 3129, 3140	\Alph ..... 1412	\bigskip .... 450, 731, 736, 1568, 1910, 2247
\@liDistance ..... ... 3134, 3135, 3139	\alph ..... 1412, 1413	\bool ..... 336, 418
\@liPos .. 3131, 3132, 3139	\alpha 2989, 2991, 2992, 2995, 2997, 2998, 2999, 3000, 3001, 3005, 3011, 3012, 3017, 3018, 3019, 3022, 3030, 3031, 3032	\bowtie ..... 2714, 2717, 2718, 2719
\\ ..... 729, 757, 758, 761, 762, 765, 766, 861, 862, 863, 970, 1004, 1006, 1036, 1045, 1090, 1132, 1133, 1134, 1139, 1140, 1141, 1161, 1908, 2274, 2332, 2335	\arabic 1412, 2588, 2593, 2598, 2604, 2610, 2616	\Box ..... 164
{ ..... 212, 1286, 1296, 1308, 1309, 1314, 1348, 1747, 2319, 2744, 3075, 3244	\arraystretch ..... 2269	\boxtimes ..... 548
} ..... 212, 1286, 1296, 1308, 1309, 1316, 1352, 1748, 2319, 2744, 3079, 3244	B \BeforeBeginEnvironment ..... 2868	C \c ..... 1495, 1496
_ ..... 37, 45, 50, 52, 305, 333, 357, 360, 370, 393, 397, 401, 403, 405, 407, 410, 416, 417, 431, 432, 438, 441, 444, 454, 461, 463, 466, 1552, 1557, 1590, 1640	\begin 755, 809, 824, 859, 883, 930, 962, 977, 1002, 1012, 1032, 1052, 1084, 1101, 1130, 1155, 1176, 1200, 1215, 1329, 1418, 1493, 1646, 1701, 1710, 1717, 1825, 1900, 1958, 1963, 1971, 1996, 2001, 2012, 2022, 2026, 2109, 2113, 2132, 2155, 2178, 2193, 2272, 2273, 2323, 2368, 2508, 2528, 2669, 2691, 2705, 2868, 2892, 3079	\cdot .... 2084, 2140, 2151 \centerline ..... 1706, 2667, 2689, 2704 \cftbeforesecskip .. 1680 \cftbeforesubsecskip ..... 1681, 1682 \cftbeforesubsubsecskip ..... 1683 \cftsubsecafterpnum 1682 \chapter ..... 1402, 1403 \char ..... 1908 \clearpage 1607, 1688, 1902 \cline ..... 729 \clist ..... 231, 279, 280, 302, 306, 2801 \columnbreak ..... 2836 \contentsname ..... 1686

<code>\cs</code> . . . . .	305, 333, 357, 360, 370, 393, 405, 416, 417, 454, 461, 466, 1528, 1535, 1541, 1990, 2391	1148, 1168, 1187, 1212, 1230, 1331, 1420, 1509, 1676, 1713, 1714, 1721, 1830, 1905, 1960, 1968, 1987, 2003, 2004, 2018, 2029, 2030, 2111, 2127, 2152, 2175, 2190, 2200, 2276, 2277, 2325, 2370, 2518, 2538, 2669, 2691, 2705, 2869, 2896, 3079	<b>F</b>	
<code>\csname</code> . . . . .	1480, 1483	1480, 1483	<code>\faCheckSquare0</code> . . . .	3059
<code>\cup</code> . . . . .	1309, 2353, 2998, 3012, 3019	2004, 2018, 2029, 2030, 2111, 2127, 2152, 2175, 2190, 2200, 2276, 2277, 2325, 2370, 2518, 2538, 2669, 2691, 2705, 2869, 2896, 3079	<code>\faCircleThin</code> . . . . .	1268
<b>D</b>			<code>\faGg</code> . . . . .	1262
<code>\DeclareMathSymbol</code> . .	2217, 2218	2325, 2370, 2518, 2538, 2669, 2691, 2705, 2869, 2896, 3079	<code>\fancyfoot</code> . . . . .	1424, 1425, 1426, 1854, 1855, 1856, 1857
<code>\DecoINERT</code> . . . . .	2591, 2664, 2665, 2687	<code>\endcsname</code> . . . .	<code>\fancyhead</code> . . . . .	1423, 1851, 1852, 1853
<code>\DecoINERTwithPivot</code> .	2607, 2686	<code>\endgroup</code> . . . .	<code>\faSquare0</code> . . . . .	1256
<code>\DecoLEFT</code> . . . . .	2586, 2685	<code>\enspace</code> . . . . .	<code>\fi</code> . . . . .	655, 663, 671, 679, 686, 1292, 1306, 1342, 1957, 2059, 2069, 2079, 2093, 2317, 2354, 2399, 2460, 2627, 2819, 2820, 2823, 2825, 3213
<code>\DecoLEFTwithPivot</code> . .	2601, 2663	environments:	<code>\filcenter</code> . . . . .	1606
<code>\DecoRIGHT</code> . . . . .	2596, 2688, 2698	<code>liAdditum</code> . . . . .	<code>\fontspec</code> . . . . .	1398
<code>\DecoRIGHTwithPivot</code> .	2613, 2666	<code>liAHuelle</code> . . . . .	<code>\footcite</code> . . . . .	752, 821, 843, 890, 913, 945, 998, 1049, 1076, 1122, 1726, 1729, 1736, 1741, 1746, 1750, 1756, 1761, 1866, 2128, 2129, 2284, 2521
<code>\definecolor</code> . . . . .	1400	<code>liAntwort</code> . . . . .	<code>\footnote</code> . . . . .	2033, 2037
<code>\DefineVerbatimEnvironment</code>	2842	<code>liDiagramm</code> . . . .	<code>\footnotesize</code> . . . . .	152, 427, 509, 603, 1273, 1564, 1931, 1982, 1997, 2013, 2228, 2322, 2339, 2347, 2476, 2485, 2728, 2843, 2874, 3039, 3114, 3140
<code>\delta</code> . . . . .	70, 112, 170, 212, 1300	<code>liEinbettung</code> . . . .	<code>\footrulewidth</code> . . . .	1429, 1859
<code>\dh</code> . . . . .	2041, 3000	<code>liExkurs</code> . . . . .	<code>\foreach</code> . . . . .	1495, 1498, 1505
<code>\directlua</code> . . . . .	63, 142, 200, 205, 1281, 1295, 1315, 1323, 1330, 1335, 2304, 2309, 2362, 2369, 2376, 2756, 2855, 2900, 2905, 2906, 2910, 2915, 2916, 2920, 2926, 2927	<code>liGraphenFormat</code> . .	<code>\forestFirst</code> . . . .	2816, 2819
<code>\do</code> . . . . .	2587, 2592, 2597, 2602, 2608, 2614	<code>liJavaAngabe</code> . . . .	<code>\forestLast</code> . . . .	2817, 2819
<code>\dots</code> . . . . .	589, 593, 1747, 2515, 3011, 3012	<code>liKasten</code> . . . . .	<code>\forestOget</code> . . . .	2816, 2817
<code>\DOWNarrow</code> . . . . .	3127	<code>liKontrollflussgraph</code>	<code>\forestOnes</code> . . . . .	2829
<code>\draw</code> . . . . .	1500, 1503, 1506, 2463, 2762, 2765	1824 <code>liLernkartei</code> . . . . .	<code>\forestOv</code> . . . . .	2818, 2819, 2822
<b>E</b>		2010 <code>liProduktionsRegeln</code>	<code>\foresttov</code> . . . . .	2808, 2812, 2813, 2816, 2817, 2818, 2819, 2821, 2822
<code>\edef</code> . . . . .	1491, 2680, 2684, 2696, 2697	1324 <code>liProjektSprache</code> . .	<code>\forestset</code> . . . . .	2804, 2827
<code>\edge</code> . . . . .	276	1932 <code>liQuellen</code> . . . . .	<code>\forestSortLevel</code> . . . .	2806, 2814, 2828, 2829
<code>\else</code> . . . . .	653, 661, 669, 677, 684, 1290, 1304, 1340, 1954, 2057, 2067, 2077, 2091, 2315, 2352, 2397, 2458, 2626, 2819, 2821, 3211	1990 <code>liRelationenSchemaFormat</code>	<code>\frac</code> . . . . .	2086, 2119, 2151, 2166
<code>\emph</code> . . . . .	750, 997, 1252, 1728, 1757, 1759, 1907	2748 <code>liRmodell</code> . . . . .	<code>\fullouterjoin</code> . . . .	2719
<code>\empty</code> . . . . .	1954, 2351	2728 <code>liUebergangsTabelle</code>	<b>G</b>	
<code>\emptyset</code> . . . . .	2261, 3006, 3027, 3060	2269 <code>\equiv</code> . . . . .	<code>\g</code> . . . . .	39, 279, 280, 302, 306, 312, 313, 314, 315, 316, 318, 319, 321, 322, 324, 325, 326, 327, 328, 329, 330, 331, 334, 338, 339, 340, 343, 345, 346, 347, 348, 349, 350, 395, 399, 403, 406, 407, 409, 410, 412, 413, 420, 421, 422, 423, 431, 432,
<code>\end</code> . . . . .	794, 820, 845, 880, 914, 946, 974, 990, 1009, 1017, 1048, 1077, 1098, 1123,	3226, 3245 <code>\erzeuge@tiefgestellt</code>		

434, 440, 441, 443, 444, 446, 447, 459, 462, 467, 469, 473, 1529, 1531, 1542, 1544, 1614, 1618, 1622, 1626, 1627, 1628, 1629, 1630, 1632, 1633, 1635, 1659, 1661, 1666, 1670	1990, 1994, 2114, 2119, 2123, 2133, 2139, 2144, 2156, 2160, 2164, 2168, 2172, 2179, 2183, 2187, 2509, 2512, 2515, 2529, 2532, 2535	\LehramtInformatikGitBranch ..... 458, 2862
\Gamma . 111, 169, 212, 1309		\LehramtInformatikGithubCodeRepo ..... 2861
\gappto ..... 2824		\LehramtInformatikGithubDomain ..... 455, 2858
\geometry ..... 5		\LehramtInformatikGithubRawDomain ..... 2859
\geq ..... 1754, 2117, 2504, 2509, 2525, 2529	\itshape ..... 602, 3038	\LehramtInformatikGithubTexRepo ..... 456, 2860
	<b>J</b>	\LehramtInformatikRepository ..... 17, 20, 23, 26, 29, 1624, 1877, 1878, 1879, 1880, 1881, 1882, 1883, 1884, 1885, 1886, 2857
<b>H</b>	<b>K</b>	\LehramtInformatikTitel ..... 1852
\hbox ..... 2714	\k ..... 1505	\leq .... 2151, 2512, 2532
\headrulewidth . 1428, 1858	\keys ..... 41, 75, 87, 117, 127, 175, 185, 310, 621, 625, 639, 644, 1373, 1380	\let ..... 1298, 1299, 1902, 2663, 2664, 2665, 2666, 2679, 2681, 2682, 2683, 2685, 2686, 2687, 2688, 2698, 2790, 2828, 2829, 3059
\headwidth ..... 1860		\li@chomsky@erklaerung@texte ..... 568, 604
\hfill .... 1558, 1589, 2136, 2142, 2147, 3067	<b>L</b>	\li@EntwurfsCode .... 746, 797, 798, 799, 848, 849, 850, 851, 917, 918, 919, 920, 921, 922, 949, 950, 951, 952, 953, 954, 1020, 1190, 1191, 1192, 1193, 1233, 1234
\hinweis ..... 1931	\l ..... 68, 69, 70, 71, 72, 73, 76, 77, 78, 79, 80, 82, 84, 89, 90, 91, 92, 93, 94, 109, 110, 111, 112, 113, 114, 115, 118, 119, 120, 121, 122, 123, 124, 130, 131, 132, 133, 134, 135, 136, 167, 168, 169, 170, 171, 172, 173, 176, 177, 178, 179, 180, 181, 182, 188, 189, 190, 191, 192, 193, 194, 619, 622, 627, 628, 631, 636, 637, 640, 641, 646, 1368, 1369, 1370, 1371, 1374, 1375, 1376, 1377, 1383, 1384, 1385, 1386, 1841, 1842, 1843, 1993, 1994, 1995, 2002	\li@EntwurfsCodeAllgemein ..... 745
\hline ..... 2274		\li@fussnote@text 2225, 2231, 2235, 2239, 2243
\horizontale .. 1535, 1566		\li@GithubLink ..... 2891, 2904, 2914, 2925
\href ..... 1362, 1584, 2037, 2204, 2895	\labelenumi ..... 1413	\li@mget . 1482, 1486, 1506
\hspace .. 2466, 3218, 3226	\labelenumii ..... 1414	\li@minc ..... 1485, 1507
\ht ..... 2715	\labelitemi ..... 1407	\li@mset ..... 1479, 1487, 1496, 1499
\Huge ..... 1657, 3096	\labelitemii ..... 1408	\li@numdiscs ..... ... 1491, 1500, 1506
\huge ..... 1606, 3090	\labelitemiii ..... 1409	\li@Rmodell@Schrift . ... 2728, 2737, 2747
	\labelitemiv ..... 1410	\li@sequence .. 1492, 1505
<b>I</b>	\land ..... 3246, 3248	\li@synthese@erklaerung@texte ..... 2980, 3040
\i ..... 1505, 1506	\LARGE ..... 1402, 1666	\li@Ableitung ..... 1323
\ifcase ..... 2623	\Large ..... 1557	liAdditum (environment) ..... 1961
\ifmmode 651, 659, 667, 675, 682, 1288, 1302, 1338, 2055, 2065, 2075, 2089, 2313, 2395, 2456, 3209	\large 1552, 1647, 1706, 2778	liAHuelle (environment) ..... 2320
\IfNoValueTF ..... ... 1964, 2033, 2037	\leaders ..... 3067	\li@Alphabet ..... 1308
\ifnum ..... 2550, 2791, 2807, 2815, 2821	\left ..... 2049, 2389	\LehramtInformatikAutorEmailIliAntwort (environment) ..... 1857
\ifx .... 1954, 2351, 2819	\LEFTarrow ..... 3128	\LehramtInformatikAutorName ..... 1856
\in ..... 577, 733, 1754, 2099, 2102, 2105, 2134, 2140, 2145, 2504, 2515, 2525, 2535, 2989, 2991, 2997, 3018, 3167	\leftarrow ..... 665	\li@Anweisung ..... 1832
\inhaltsverzeichnis 1899	\leftouterjoin .... 2717	\li@AssemblerCode ... 2929
\input ..... 17, 20, 23, 26, 29, 479, 1623	\leftskip 3230, 3231, 3241	
\inputminted 2899, 2909, 2919, 2931, 2934, 2938	\LehramtInformatikAutorEmailIliAntwort (environment) ..... 1857	
\int ..... 2801	\LehramtInformatikAutorName ..... 1856	
\item 548, 549, 826, 830, 835, 840, 884, 893, 898, 906, 978, 983, 987, 1013, 1053, 1058, 1065, 1073, 1102, 1107, 1111, 1116, 1216, 1221, 1226, 1711, 1712,		

\liAssemblerDatei ..	<a href="#">2930</a>	\liEntwurfsAdapter ..	<a href="#">853</a>	\liEntwurfsStellvertreterCode	
\liAttribut .....	<a href="#">2747</a>	\liEntwurfsAdapterAkteure		.....	<a href="#">1189, 1197</a>
\liAttributHuelle ...		.....	<a href="#">823, 855</a>	\liEntwurfsStellvertreterUml	
... ..	<a href="#">2311, 2992, 2997</a>	\liEntwurfsAdapterCode		.....	<a href="#">1175, 1196</a>
\liAttributHuelleOhneMathe		.....	<a href="#">847, 856</a>	\liEntwurfsZustand ..	<a href="#">1236</a>
.....	<a href="#">2311, 2314,</a>	\liEntwurfsAdapterUml		\liEntwurfsZustandAkteure	
<a href="#">2316, 2330, 2340, 2348</a>		.....	<a href="#">808, 854</a>	.....	<a href="#">1214, 1238</a>
\liAttributMenge ....		\liEntwurfsBeobachter	<a href="#">924</a>	\liEntwurfsZustandCode	
.....	<a href="#">2319, 2331, 2334,</a>	\liEntwurfsBeobachterAkteure		.....	<a href="#">1232, 1239</a>
<a href="#">2341, 2342, 2356, 2358</a>		.....	<a href="#">882, 926</a>	\liEntwurfsZustandUml	
\liAufgabe .....	<a href="#">16</a>	\liEntwurfsBeobachterCode		.....	<a href="#">1199, 1237</a>
\liAufgabenMetadaten		.....	<a href="#">916, 927</a>	\liEpsilon .....	<a href="#">1294</a>
.....	<a href="#">47, 1548, 1638</a>	\liEntwurfsBeobachterUml		\liErAttribute .....	
\liAufgabenTitel .....	<a href="#">55</a>	.....	<a href="#">858, 925</a>	... ..	<a href="#">1252, 1266, 1268</a>
\liAusdruck .....	<a href="#">1345</a>	\liEntwurfsDekorierer	<a href="#">956</a>	\liErDatenbankName ..	<a href="#">1271</a>
\liAutomat .....	<a href="#">66</a>	\liEntwurfsDekoriererAkteure		\liErEntity <a href="#">1250, 1254, 1256</a>	
\liAutomatenKante ....	<a href="#">98</a>	.....	<a href="#">958</a>	\liErledigt .....	<a href="#">3059</a>
\liBandAlphabet ....	<a href="#">1309</a>	\liEntwurfsDekoriererCode		\liErMpAttribute ...	<a href="#">1265</a>
\liBedingung .....	<a href="#">1833</a>	.....	<a href="#">948, 959</a>	\liErMpEntity .....	<a href="#">1253</a>
\liBedingungDrei ....		\liEntwurfsDekoriererUml		\liErMpRelationship	<a href="#">1259</a>
... ..	<a href="#">2104, 2147, 2187</a>	.....	<a href="#">929, 957</a>	\liErRelationship ...	
\liBedingungEins ....		\liEntwurfsEinfacheFabrik		... ..	<a href="#">1251, 1260, 1262</a>
... ..	<a href="#">2098, 2136, 2179</a>	.....	<a href="#">992</a>	\liExamensAufgabe ....	<a href="#">19</a>
\liBedingungFalsch ..	<a href="#">1835</a>	\liEntwurfsEinfacheFabrikAkteure		\liExamensAufgabeA ...	<a href="#">28</a>
\liBedingungWahr ...	<a href="#">1834</a>	.....	<a href="#">976, 994</a>	\liExamensAufgabeTA ..	<a href="#">25</a>
\liBedingungZwei ....		\liEntwurfsEinfacheFabrikUml		\liExamensAufgabeTTA ..	<a href="#">22</a>
... ..	<a href="#">2101, 2142, 2183</a>	.....	<a href="#">961, 993</a>	liExkurs (environment)	<a href="#">1969</a>
\liBeschriftung ....	<a href="#">1923</a>	\liEntwurfsEinzelstueck		\liFalsch .....	<a href="#">549</a>
\liBindeAufgabeEin ..	<a href="#">1621</a>	.....	<a href="#">1022</a>	\liFlaci .....	<a href="#">1355</a>
\liChomskyErklaerung		\liEntwurfsEinzelstueckAkteure		\liFremd .....	<a href="#">2727</a>
.....	<a href="#">568, 609</a>	.....	<a href="#">1011, 1027</a>	\liFunktionaleAbhaengigkeit	
\liChomskyUeberErklaerung		\liEntwurfsEinzelstueckBeschreibung ..	<a href="#">2350, 2353, 2361</a>	\liFunktionaleAbhaengigkeiten	
.....	<a href="#">607</a>	.....	<a href="#">996, 1023</a>	\liFussnote ...	<a href="#">2224, 2226</a>
\liChomskyUeberschrift		\liEntwurfsEinzelstueckCode		.....	<a href="#">2238, 2256</a>
.....	<a href="#">556, 608</a>	.....	<a href="#">1019, 1029</a>	\liFussnoteEinsText ..	
\liCpmEreignis .....	<a href="#">617</a>	\liEntwurfsEinzelstueckUml		.....	<a href="#">2230, 2250</a>
\liCpmFruehErklaerung	<a href="#">688</a>	.....	<a href="#">1001, 1025</a>	\liFussnoteLink ....	<a href="#">2036</a>
\liCpmFruehI ...	<a href="#">681, 701</a>	\liEntwurfsErbauer ..	<a href="#">1079</a>	\liFussnoten .....	<a href="#">2246</a>
\liCpmSpaetErklaerung	<a href="#">706</a>	\liEntwurfsErbauerAkteure		\liFussnoteUrl ..	<a href="#">1169, 2032</a>
\liCpmSpaetI ...	<a href="#">674, 719</a>	.....	<a href="#">1051, 1081</a>	\liFussnoteVierText ..	
\liCpmVon .....	<a href="#">657</a>	\liEntwurfsErbauerUml		.....	<a href="#">2242, 2259</a>
\liCpmVonOhneMathe ..		.....	<a href="#">1031, 1080</a>	\liFussnoteZweiText ..	
.....	<a href="#">657, 660, 662</a>	\liEntwurfsFabrikmethode		.....	<a href="#">2234, 2253</a>
\liCpmVonZu .....	<a href="#">649</a>	.....	<a href="#">1125, 1150</a>	\liGeschweifteKlammern	
\liCpmVonZuOhneMathe		\liEntwurfsFabrikmethodeAkteure		... ..	<a href="#">1327, 2365, 3071</a>
.....	<a href="#">649, 652, 654</a>	.....	<a href="#">1100, 1127</a>	\liGrammatik .....	<a href="#">1366</a>
\liCpmVorgang .....	<a href="#">634</a>	\liEntwurfsFabrikmethodeUml		liGraphenFormat (envi-	
\liCpmZu .....	<a href="#">665</a>	.....	<a href="#">1083, 1126</a>	ronment) ....	<a href="#">1472</a>
\liCpmZuOhneMathe ...		\liEntwurfsKompositum		\liHanoi .....	<a href="#">1479</a>
.....	<a href="#">665, 668, 670</a>	.....	<a href="#">1150</a>	\liHaskellCode .....	<a href="#">2936</a>
liDiagramm (environ-		\liEntwurfsKompositumAkteure		\liHaskellDatei ....	<a href="#">2937</a>
ment) .....	<a href="#">2020</a>	.....	<a href="#">1152</a>	\liInduktionAnfang ..	<a href="#">3154</a>
liEinbettung (environ-		\liEntwurfsKompositumUml		\liInduktionErklaerung	
ment) .....	<a href="#">1933</a>	.....	<a href="#">1129, 1151</a>	.....	<a href="#">3153</a>
\liEntwurfs .....	<a href="#">1171</a>	\liEntwurfsModellPraesentation		\liInduktionMarkierung	
\liEntwurfsAbstrakteFabrik		.....	<a href="#">1171</a>	.....	<a href="#">3152</a>
.....	<a href="#">801</a>	\liEntwurfsModellPraesentationAkteure		\liInduktionSchritt	<a href="#">3170</a>
\liEntwurfsAbstrakteFabrikBeschreibung		.....	<a href="#">1173</a>	\liInduktionVoraussetzung	
.....	<a href="#">749, 802</a>	\liEntwurfsModellPraesentationUml		.....	<a href="#">3162</a>
\liEntwurfsAbstrakteFabrikCode		.....	<a href="#">1154, 1172</a>		
.....	<a href="#">796, 806</a>	\liEntwurfsStellvertreter			
\liEntwurfsAbstrakteFabrikUml		.....	<a href="#">1195</a>		
.....	<a href="#">754, 804</a>				

<code>\liJavaAngabe</code> (environment) . . . . .	<a href="#">2876</a>	<code>\liParagraphMitLinien</code> . . . . .	<a href="#">604, 689, 707, 2285, 3040, 3061, 3158, 3166, 3174</a>	<code>\liSetzeExamenTeilaufgabeNr</code> . . . . .	<a href="#">1617</a>
<code>\liJavaCode</code> . . . . .	<a href="#">2880</a>	<code>\liPetriErreichKnotenDrei</code> . . . . .	<a href="#">2465</a>	<code>\liSetzeExamenThemaNr</code> . . . . .	<a href="#">1613</a>
<code>\liJavaDatei</code> . . . . .	<a href="#">747, 2898</a>	<code>\liPetriErreichTransition</code> . . . . .	<a href="#">2462</a>	<code>\liSortierMarkierung</code> . . . . .	<a href="#">2767</a>
<code>\liJavaExamen</code> . . . . .	<a href="#">2918</a>	<code>\liPetriSetzeSchluessel</code> . . . . .	<a href="#">2407</a>	<code>\liSortierPfeil</code> . . . . .	<a href="#">2761</a>
<code>\liJavaTestDatei</code> . . . . .	<a href="#">2908</a>	<code>\liPetriTransitionsName</code> . . . . .	<a href="#">2454, 2466</a>	<code>\liSortierPfeilUnten</code> . . . . .	<a href="#">2764</a>
<code>liKasten</code> (environment) . . . . .	<a href="#">1417</a>	<code>\liPetriTransitionsNameOhneMathe</code> . . . . .	<a href="#">2454, 2457, 2459</a>	<code>\liSpaltenUmbruch</code> . . . . .	<a href="#">2836</a>
<code>\liKellerAutomat</code> . . . . .	<a href="#">107</a>	<code>\liPetriTransPfeile</code> . . . . .	<a href="#">2466</a>	<code>\liSqlCode</code> . . . . .	<a href="#">2941</a>
<code>\liKellerKante</code> . . . . .	<a href="#">145</a>	<code>\liPolynomiellReduzierbar</code> . . . . .	<a href="#">1716</a>	<code>\listen@punkt</code> . . . . .	<a href="#">1990, 2002</a>
<code>\liKellerUebergang</code> . . . . .	<a href="#">140, 146</a>	<code>\liPotenzmenge</code> . . . . .	<a href="#">1295, 1299, 2477</a>	<code>\liStrich</code> . . . . .	<a href="#">1698</a>
<code>\liKontrollCode</code> . . . . .	<a href="#">1836</a>	<code>\liPotenzmengeOhneMathe</code> . . . . .	<a href="#">1296, 1297, 1298</a>	<code>\liSyntheseErklaerung</code> . . . . .	<a href="#">2980, 3045</a>
<code>liKontrollflussgraph</code> (environment) . . . . .	<a href="#">1824</a>	<code>\liPrimaer</code> . . . . .	<a href="#">2726</a>	<code>\liSyntheseUeberErklaerung</code> . . . . .	<a href="#">3043</a>
<code>\liKontrollKnotenPfad</code> . . . . .	<a href="#">1838</a>	<code>\liProblemBeschreibung</code> . . . . .	<a href="#">1700</a>	<code>\liSyntheseUeberschrift</code> . . . . .	<a href="#">2964, 3044</a>
<code>\liKontrollTextzeileKnoten</code> . . . . .	<a href="#">1837, 1842</a>	<code>\liProblemClique</code> . . . . .	<a href="#">1723</a>	<code>\liT</code> . . . . .	<a href="#">2081, 2096, 2110, 2174</a>
<code>\liKurzeTabellenLinie</code> . . . . .	<a href="#">729</a>	<code>\liProblemName</code> . . . . .	<a href="#">1699, 1706, 1718, 1720, 1733, 1744, 1745, 1753, 1754</a>	<code>\liTeilen</code> . . . . .	<a href="#">2308</a>
<code>\liLadeAllePakete</code> . . . . .	<a href="#">233, 1609</a>	<code>\liProjektSprache</code> (environment) . . . . .	<a href="#">1932</a>	<code>\liLiteratur</code> . . . . .	<a href="#">1865, 1889</a>
<code>\liLadePakete</code> . . . . .	<a href="#">59, 62, 229, 234, 555, 616, 1285, 1515, 1597, 1696, 2223, 2303, 2471, 2840, 2962</a>	<code>\liPseudoUeberschrift</code> . . . . .	<a href="#">1909, 1965, 1966, 2271, 2281, 3155, 3163, 3171</a>	<code>\liTheta</code> . . . . .	<a href="#">2051, 2102, 2134, 2140, 2145</a>
<code>\liLatexCode</code> . . . . .	<a href="#">2890</a>	<code>\liPumpingKontextfrei</code> . . . . .	<a href="#">2523</a>	<code>\liThetaOhneMathe</code> . . . . .	<a href="#">2051, 2056, 2058</a>
<code>\liLeereZelle</code> . . . . .	<a href="#">2261</a>	<code>\liPumpingRegulaer</code> . . . . .	<a href="#">2502</a>	<code>\liTOhneMathe</code> . . . . .	<a href="#">2081, 2090, 2092</a>
<code>liLernkartei</code> (environment) . . . . .	<a href="#">2010</a>	<code>liQuellen</code> (environment) . . . . .	<a href="#">1990</a>	<code>\liTuringKante</code> . . . . .	<a href="#">208</a>
<code>\liLinksReduktion</code> . . . . .	<a href="#">2328</a>	<code>\liRechtsReduktionInline</code> . . . . .	<a href="#">2345</a>	<code>\liTuringLeerzeichen</code> . . . . .	<a href="#">164, 172</a>
<code>\liLinksReduktionInline</code> . . . . .	<a href="#">2337, 2345</a>	<code>\liRekursionsGleichung</code> . . . . .	<a href="#">2095, 2158</a>	<code>\liTuringMaschine</code> . . . . .	<a href="#">165</a>
<code>\liMasterExkurs</code> . . . . .	<a href="#">2192</a>	<code>\liRelation</code> . . . . .	<a href="#">2375</a>	<code>\liTuringUeberfuehrung</code> . . . . .	<a href="#">211</a>
<code>\liMasterFaelle</code> . . . . .	<a href="#">2131, 2199</a>	<code>liRelationenSchemaFormat</code> (environment) . . . . .	<a href="#">2748</a>	<code>\liTuringUebergaenge</code> . . . . .	<a href="#">203, 209</a>
<code>\liMasterFallRechnung</code> . . . . .	<a href="#">2177</a>	<code>\liRelationMenge</code> . . . . .	<a href="#">2742</a>	<code>\liTuringUebergangZelle</code> . . . . .	<a href="#">198</a>
<code>\liMasterVariablen</code> . . . . .	<a href="#">2108, 2194</a>	<code>\liRichtig</code> . . . . .	<a href="#">548</a>	<code>\liTypoUeberGROSS</code> . . . . .	<a href="#">3094</a>
<code>\liMasterVariablenDeklaration</code> . . . . .	<a href="#">2154</a>	<code>liRmodell</code> (environment) . . . . .	<a href="#">2728</a>	<code>\liTypoUeberGross</code> . . . . .	<a href="#">3088, 3094</a>
<code>\liMasterWolframLink</code> . . . . .	<a href="#">2202</a>	<code>\liRundeKlammer</code> . . . . .	<a href="#">2048, 2052, 2062, 2072, 2086, 2388, 2392</a>	<code>\liTypoUeberschrift</code> . . . . .	<a href="#">3082, 3091, 3097</a>
<code>\liMenge</code> . . . . .	<a href="#">76, 77, 79, 118, 119, 120, 124, 176, 177, 178, 182, 1286, 1335, 1374, 1375</a>			<code>\liUeberfuehrungsFunktion</code> . . . . .	<a href="#">1300</a>
<code>\liMengeOhneMathe</code> . . . . .	<a href="#">1286, 1289, 1291</a>			<code>\liUeberfuehrungsFunktionOhneMathe</code> . . . . .	<a href="#">1300, 1303, 1305</a>
<code>\liMetaSetze</code> . . . . .	<a href="#">36, 48, 1549, 1639</a>			<code>liUebergangsTabelle</code> (environment) . . . . .	<a href="#">2269</a>
<code>\liMinimierungErklaerung</code> . . . . .	<a href="#">2283</a>			<code>\liUeberschriftDreiecksTabelle</code> . . . . .	<a href="#">2279</a>
<code>\liMinispracheDatei</code> . . . . .	<a href="#">2933</a>			<code>\liUmlLeserichtung</code> . . . . .	<a href="#">3122</a>
<code>\linespread</code> . . . . .	<a href="#">2735</a>			<code>\liVertauschen</code> . . . . .	<a href="#">2755</a>
<code>\liNichtsZuTun</code> . . . . .	<a href="#">3060</a>			<code>\liWortInSprache</code> . . . . .	<a href="#">730</a>
<code>\liO</code> . . . . .	<a href="#">2071, 2099</a>			<code>\liWortNichtInSprache</code> . . . . .	<a href="#">735</a>
<code>\liOmega</code> . . . . .	<a href="#">2061, 2105</a>			<code>\liWpEquivalent</code> . . . . .	<a href="#">3225</a>
<code>\liOmegaOhneMathe</code> . . . . .	<a href="#">2061, 2066, 2068</a>			<code>\liWpErklaerung</code> . . . . .	<a href="#">3228</a>
<code>\liONotationO</code> . . . . .	<a href="#">2391</a>			<code>\liWpErklaerungVerzweigung</code> . . . . .	<a href="#">3243</a>
<code>\liOOhneMathe</code> . . . . .	<a href="#">2071, 2076, 2078</a>			<code>\liWpKalkuel</code> . . . . .	<a href="#">3205</a>
				<code>\liWpKalkuelOhneMathe</code> . . . . .	<a href="#">3205, 3210, 3212, 3244, 3246, 3248</a>



<code>\liZustandsBuchstabe</code>	N	<code>\Omega</code>	2062
..... 1310,	<code>\NeedsTeXFormat</code>	<code>\omega</code>	2504, 2505, 2525, 2526
1319, 1321, 1339, 1341	..... 1, 14, 32,	<code>\or</code>	2625
<code>\liZustandsBuchstabeGross</code>	57, 225, 483, 544,		
... 1311, 1320, 1322	551, 613, 726, 741,	P	
<code>\liZustandsmenge</code>	1242, 1278, 1391,	<code>\pagestyle</code>	1427, 1645
<code>\liZustandsmengeNr</code>	1431, 1440, 1445,	<code>\par</code>	428, 449,
..... 1312, 2486	1474, 1512, 1594,	608, 1356, 1364,	
<code>\liZustandsmengeNrGross</code>	1692, 1764, 1847,	1532, 1536, 1539,	
..... 1320	1863, 1869, 1891,	1552, 1559, 1570,	
<code>\liZustandsMengenSammlung</code>	2044, 2207, 2220,	1586, 1589, 1648,	
..... 2473	2299, 2382, 2402,	1652, 1659, 1661,	
<code>\liZustandsMengenSammlungNr</code>	2468, 2493, 2498,	1666, 1670, 1674,	
..... 2482	2542, 2708, 2721,	1912, 1918, 1924,	
<code>\liZustandsmengeOhneMathe</code>	2750, 2831, 2838,	1984, 2006, 2015,	
..... 1298	2845, 2850, 2943,	2667, 2689, 2704,	
<code>\liZustandsname</code>	2958, 3049, 3053,	2745, 3044, 3068,	
<code>\liZustandsnameGross</code>	3102, 3144, 3181, 3201	3072, 3080, 3233, 3239	
... 1322, 2474, 2483	<code>\neg</code>	<code>\paragraph</code>	1404
<code>\liZustandsnameTiefgestellt</code>	..... 3248	<code>\parindent</code>	2736
..... 1337	<code>\negthinspace</code>	<code>\path</code>	99, 146, 209, 646
<code>\liZustandsPaar</code>	... 2049, 2389	<code>\pgfkeys</code>	2418, 3125,
<code>\liZustandsPaarVariablenName</code>	<code>\newcounter</code>	3126, 3127, 3128,	
... 2262, 2265, 2266	... 2562, 2563	3129, 3132, 3135, 3137	
<code>\llap</code>	<code>\NewDocumentCommand</code>	<code>\pgfmath@count</code>	2789, 2791, 2793
<code>\log</code>	..... 67, 108,	2790, 2792, 2794,	
2102, 2105, 2134, 2140	145, 166, 208, 229,	2807, 2815, 2828, 2829	
<code>\loop</code>	618, 635, 674, 681,	<code>\pgfmath@empty</code>	2790
<code>\lor</code>	730, 735, 1345,	<code>\pgfmath@loop</code>	2791
<code>\ltimes</code>	1367, 1716, 1839,	<code>\pgfmath@repeat</code>	2794
	1867, 2032, 2036,	<code>\preceq</code>	1719
	2364, 2375, 2462,	<code>\prime</code>	1698
	2898, 2908, 2918,	<code>\printbibliography</code>	1889
	2930, 2933, 2937, 3122	<code>\ProvidesPackage</code>	2, 15, 33,
	<code>\NewDocumentEnvironment</code>	58, 226, 484, 545,	
M	1324, 1417, 1472,	1243, 1279, 1392,	
<code>\makeatletter</code>	1824, 1932, 1933,	1432, 1441, 1446,	
<code>\makeatother</code>	1934, 1961, 1969,	1475, 1513, 1595,	
<code>\marginpar</code>	1991, 2010, 2020,	1693, 1765, 1848,	
1255, 1261, 1267, 1907	2270, 2320, 2730, 2748	1864, 1870, 1892,	
<code>\mathbb</code>	<code>\newlength</code>	2045, 2208, 2221,	
1754, 2535, 3167	..... 3228	2300, 2383, 2403,	
<code>\mathbin</code>	<code>\newminted</code>	2469, 2494, 2499,	
2717, 2718, 2719	..... 2876	2543, 2709, 2722,	
<code>\mathcal</code>	<code>\node</code>	2751, 2832, 2839,	
2072, 2392, 3018,	..... 631, 1832,	2846, 2851, 2944,	
3023, 3025, 3026, 3027	1837, 2588, 2593,	2959, 3050, 3054,	
<code>\Mathe</code>	2598, 2604, 2610,	3103, 3145, 3182, 3202	
<code>\MatheEnv</code>	2616, 2767, 2812, 3139		
<code>\mathord</code>	<code>\noexpand</code>		
2028, 2733, 2739,	..... 2676,		
3069, 3073, 3216, 3220	2677, 2678, 2697, 2812		
<code>\memph</code>	<code>\noindent</code>		
<code>\mintinline</code>	..... 429,		
2883,	732, 737, 1537,		
2890, 2929, 2936, 2941	1551, 1556, 1563,		
<code>\mkern</code>	1579, 1586, 1589,		
2717, 2718, 2719	1913, 1925, 1929,		
<code>\mlq</code>	1955, 1983, 1985,		
2215, 2217	1998, 2014, 2016,		
<code>\mrq</code>	2024, 2196, 2249,		
2215, 2218	2252, 2255, 2258,		
<code>\msg</code>	2743, 3062, 3074, 3234		
477	<code>\nolinkurl</code>		
<code>\myList</code>	..... 2895		
2808, 2809, 2810, 2813	<code>\normalsize</code>		
<code>\myNodes</code>	... 1404, 2884		
2812, 2818, 2822, 2824	<code>\notin</code>		
	..... 738		
	<code>\null</code>		
	..... 3067		
	O		
	<code>\o</code>		
	..... 2391, 2396, 2398		
	<code>\o@join</code>		
	..... 2714, 2717, 2718, 2719		
	Q		
	<code>\QS@list</code>		
	..... 2669, 2680, 2684,		

2691, 2697, 2702, 2705	2784, 2835, 2841,	\Sigma ..... 69, 110,
\QS@select@equal ....	2848, 2853, 2864,	168, 1308, 1309, 1369
..... 2644, 2648	2865, 2946, 3051,	\sigma ..... 582, 584, 585
\QS@select@greater ..	3058, 3106, 3107,	\SLASH ..... 1908
..... 2645, 2649	3109, 3110, 3148,	\small ..... 2023
\QS@select@smaller ..	3149, 3150, 3183, 3203	\sort ..... 2801
... 2640, 2643, 2647	\right ..... 2049, 2389	\sortList ..... 2800, 2809
\QS@sort@a ..... 2622, 2655, 2676, 2677	\RIGHTarrow .... 3124, 3129	\square ..... 549
\QS@sort@b .... 2622, 2623	\Rightarrow .... 733, 738	\stepcounter 2588, 2593,
\QS@sort@c .... 2626, 2633	\rightarrow .... 212,	2598, 2601, 2603,
\QS@sort@d .... 2634, 2642	572, 577, 585, 589,	2607, 2609, 2613, 2615
\QS@sort@empty . 2624, 2629	591, 592, 594, 649,	\stichwoerter .. 1528, 1564
\QS@sort@single 2625, 2630	657, 2466, 2989,	\str 361, 560, 569, 1936,
\QS@initialize ..... 3006, 3011, 3012, 3017	2996, 2998, 3001,	2478, 2487, 2968, 2981
... 2547, 2659, 2694	\rightouterjoin .... 2718	\string ..... 2331, 2341
\QSIr . 2630, 2636, 2644,	\Roman ..... 1412	\StrSubstitute . 2808, 2810
2664, 2678, 2683, 2686	\roman ..... 1412, 1414	\strut .... 2156, 2160,
\QSIrr 2665, 2678, 2679, 2687	\romannumeral ..... 2634	2164, 2168, 2172, 2836
\QSLr ..... 2636,	\rtimes ..... 2210	\subsection ..... 1619
2643, 2654, 2655,	\rule ..... 1538,	\subseq 2992, 3025, 3032
2663, 2676, 2681, 2685	2667, 2689, 2704, 2715	\subsubsection ..... 1640
\QSPivotStep ..... 2549, 2659, 2663, 2674		
\QSR ..... 2636	S	T
\QSRr ..... 2645,	\sb ..... 72, 82,	\tableofcontents ....
2666, 2677, 2682,	84, 113, 171, 584,	..... 1687, 1903
2688, 2697, 2698, 2699	585, 589, 592, 593,	\TeX ..... 1587
\QSSortStep ..... 685, 1339, 1341,	594, 676, 678, 683,	\text ..... 82, 84,
2551, 2659, 2675, 2676	2099, 2102, 2105,	187, 2311, 3153, 3206
\quad ..... 2227	2134, 2140, 2289,	\textbf ..... 696, 714,
	2454, 2463, 3011,	1250, 1530, 1724,
	3012, 3013, 3018,	1733, 1744, 1753,
R	3022, 3023, 3026,	1914, 1927, 1956,
\raisebox ..... 1837	3027, 3030, 3031, 3032	1984, 1999, 2015, 2274
\relax ..... 1902,	\scriptscriptstyle ..	\textcolor 1836, 2882, 3152
2636, 2679, 2681,	..... 649, 657, 665	\textit ... 1090, 1132,
2682, 2683, 2791, 2793	\scriptsize .... 1358,	1133, 1134, 1135,
\renewcommand ..... 1407, 1408, 1409,	1777, 1784, 1790,	1543, 2025, 2319, 2379
1410, 1413, 1414,	1852, 1853, 1856,	\textsc ..... 1699
1428, 1429, 1682,	1857, 3153, 3206, 3236	\textsf ..... 1914, 1999
1686, 1858, 1859, 2269	\section ... 52, 1606, 1615	\textstyle .... 2119, 2151
\repeat ..... 2552	\sectionbreak ..... 1607	\texttt 1273, 1699, 1833,
\RequirePackage ..... 4, 60, 163,	\seq . 1841, 1842, 1843,	1834, 1835, 1836, 3206
227, 231, 480, 486,	1993, 1994, 1995, 2002	\textwidth .... 1538, 1860
487, 547, 615, 744,	\setbox ..... 2714	\thematik ..... 1541, 1558
745, 1245, 1247,	\setcounter 1405, 1608,	\thepage ..... 1426, 1855
1248, 1284, 1393,	1642, 2668, 2690, 2704	\theparagraph ..... 1404
1394, 1397, 1399,	\setganttlinklabel ..	\thesection ..... 1606
1401, 1406, 1415,	1435, 1436, 1437, 1438	\Theta ..... 2052
1422, 1433, 1434,	\setlength ..... 1680,	\thinspace ..... 3206
1443, 1447, 1448,	1681, 1683, 1860,	\tikz ..... 1837
1449, 1477, 1478,	2736, 3230, 3231, 3241	tikz: bbaum ..... 26
1526, 1605, 1610,	\setmainfont ..... 1395	tikz: li binaer baum .... 24
1611, 1643, 1697,	\setmainlanguage .... 481	\tikzchildnode ..... 504
1766, 1866, 1871,	\setminted .... 2870, 2871	\tikzparentnode ..... 504
1872, 1888, 1895,	\setminus ..... 2349	\tikzset ..... 101,
1896, 1897, 2047,	\setsansfont ..... 1396	148, 214, 489, 515,
2211, 2212, 2385,	\setul ..... 2727	1451, 1768, 2444,
2386, 2387, 2404,	\sffamily ..... 559,	2570, 2774, 2948, 3184
2406, 2496, 2556,	1402, 1404, 1506,	\tikzumlset ..... 3112
2558, 2710, 2711,	1606, 1657, 2967, 3084	\times ..... 212
2712, 2725, 2753,	\shoveleft ..... 2329	\tiny . 1256, 1262, 1268,
	\shoveright ..... 2333	1578, 1836, 1907, 2893
		\titleformat ..... 1402, 1404, 1606

