

# 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]{zustaeende=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{zustaeende={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_zustaeende_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     zustaeende .code:n = {\tl_set:Nn \l_zustaeende_tl {\liMenge{##1}}},
77     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
78     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
79     ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
```

```

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

```

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

```

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

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

```

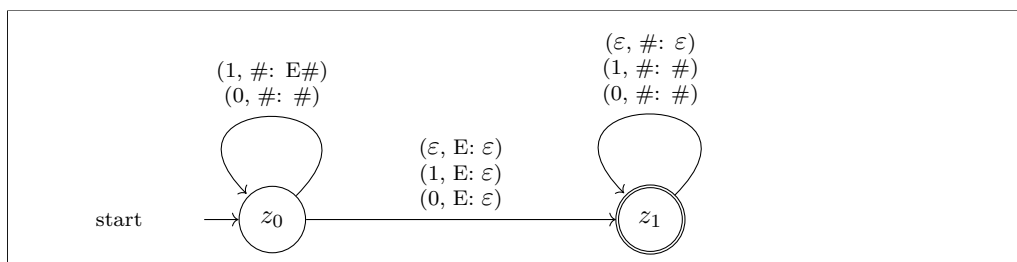
## 2.4.2 Kellerautomat

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

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

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

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



```
\liKellerAutomat \liKellerAutomat[⟨automaten-name⟩]
{⟨zustaende=Z,alphabet=Σ,kelleralphabet=Γ,delta=δ,start=z_0,kellerboden=#,ende=E⟩}

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

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

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

```

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

```

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

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

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

```

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

```

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

```

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

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

```

### 2.4.3 Turingmaschine

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

**\liTuringLeerzeichen**

□

```
164 \def\liTuringLeerzeichen{\Box}
```

```

\liTuringMaschine \liTuringMaschine[\langle automaten-name \rangle]
{\zustaende=Z,alphabet= $\Sigma$ ,bandalphabet= $\Gamma$ ,delta= $\delta$ ,start= $z_0$ ,leerzeichen= $\square$ ,ende=E)}

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

```

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

```

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

```

**\liTuringUebergangZelle** Formatiert einen Zustandsübergang für eine Übergangstabelle.

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

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

```

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

```

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

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

`(z1:  $\square$ , L)`

`( $\square$ :  $\square$ , R)`



```

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

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

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

\liTuringUeberfuehrung

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

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

```

## 2.5 basis.sty

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

227 \RequirePackage{xparse}

228 \ExplSyntaxOn

\liLadePakete

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

\liLadeAllePakete

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

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

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

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

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

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

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

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

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

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

```

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

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

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

```

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

```

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

```

Einzelprüfungsnummer / Jahr / Jahreszeit mit Trennzeichen

```

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

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

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

```

```

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

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

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

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

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

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

```

## 2.6 baum.sty

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

486 \RequirePackage{tikz}

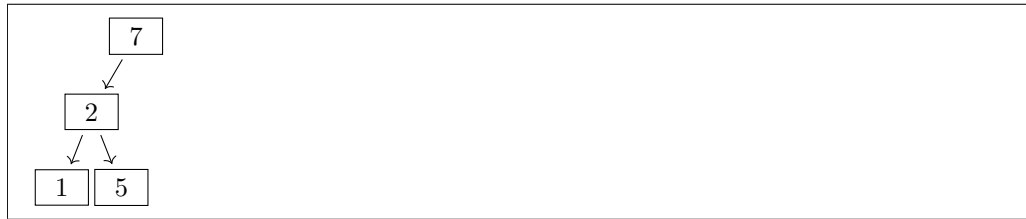
    für li binaer baum
487 \RequirePackage{tikz-qtrees}

    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}
```

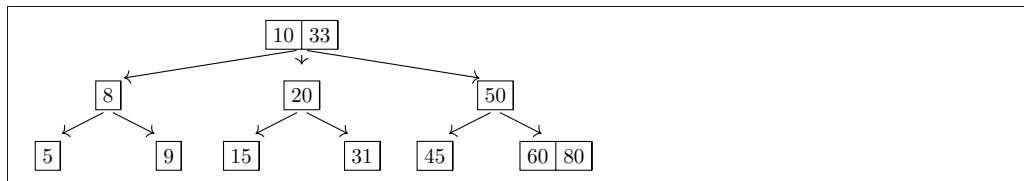


### 2.6.3 B-Baum

```

\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     \rmfamily
560     \str_case:nn {#1} {
561       {1} {Elimination-der- $\varepsilon$ -Regeln}
562       {2} {Elimination-von-Kettenregeln}
563       {3} {Separation-von-Terminalzeichen}
564       {4} {Elimination-von-mehrelementigen-Nonterminalketten}
565     }
566   }
567 }

```

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

```

568 \def\liChomsky@erklaerung@texte#1{
569   \str_case:nn {#1} {
570     %
571     {1} {
572       Alle-Regeln-der-Form- $A \rightarrow \varepsilon$ -werden-eliminiert.~
573       Die-Ersetzung-von- $A$ -wird-durch- $\varepsilon$ -in-alen-andere-
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-andere-
583       Symbolen-auftaucht,~wird-durch-ein-neues-Nonterminal-
584        $S_{\sigma}$ -ersetzt-und-die-Menge-der-Produktionen-durch-die-
585       Regel- $S_{\sigma} \rightarrow \sigma$ -ergänzt.
586     }
587     {4} {
588       Alle-Produktionen-der-Form-
589        $A \rightarrow B_{\{1\}} B_{\{2\}} \dots B_{\{n\}}$ -
590       werden-in-die-Produktionen-
591        $A \rightarrow A_{\{n-1\}} B_{\{n\}}, A_{\{n-1\}} \rightarrow A_{\{n-2\}} B_{\{n-1\}}, \dots,$ 
592        $A_{\{2\}} \rightarrow B_{\{1\}} B_{\{2\}}$ -zerteilt.~
593       Nach-der-Ersetzung-sind-alle-längere-Nonterminalketten-
594       vollständig-heruntergebrochen-und-die-Chomsky-Normalform-erreicht.
595     }
596   }
597 }
598 }

```

```

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

```

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

```

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

```

```

611 \ExplSyntaxOff
612

```

## 2.9 cpm.sty

```

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

616 \liLadePakete{mathe,typographie}

```

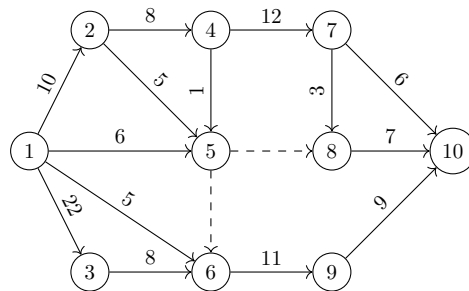
### 2.9.1 Makro-Kürzel

```

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

```

### 2.9.2 TeX-Markup-Beispiel: Graph



```

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

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

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

```

### 2.9.3 TeX-Markup-Beispiel: Ergebnistabelle

```

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

```

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

```

\liCpmFruehErklaerung
\begin{tabular}{|l|l|r|}
\hline
i & Nebenrechnung & \FZ \\
\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}
```

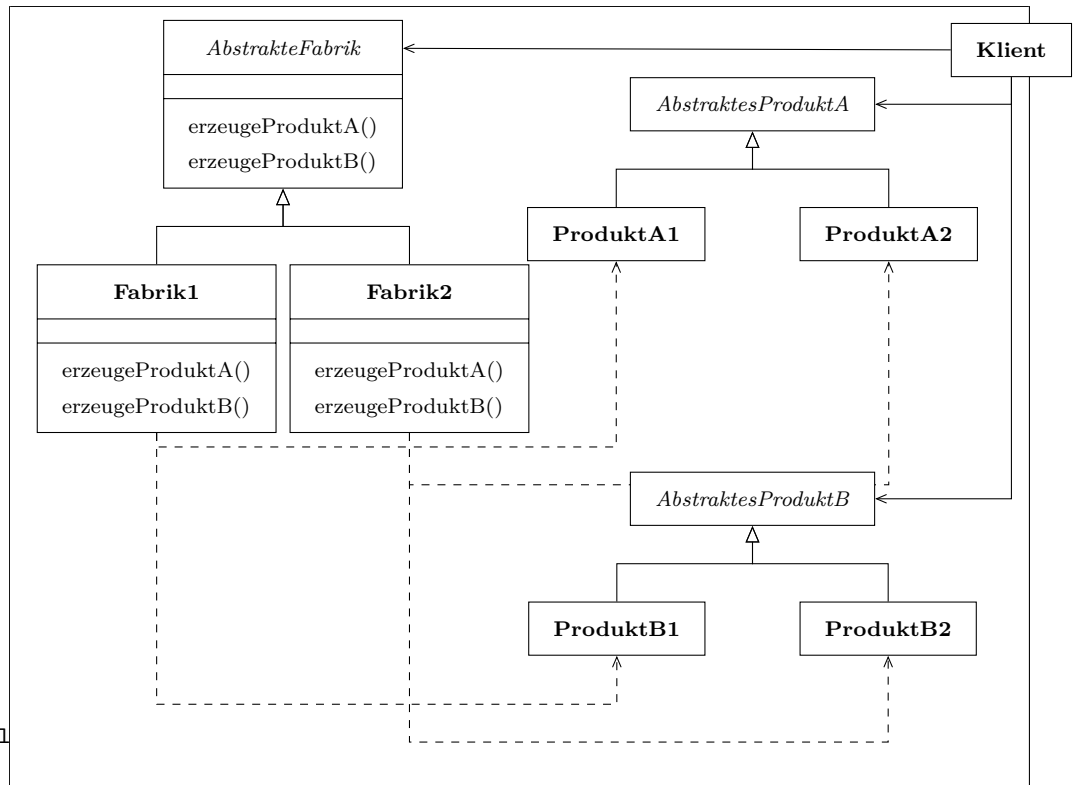
\li@EntwurfsCodeAllgemein Allgemeine Code-Beispiele zu den UML-Diagrammen und Stellvertretern

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

```

```

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

```

iEntwurfsAbstrakteFabrikCode

```

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

```

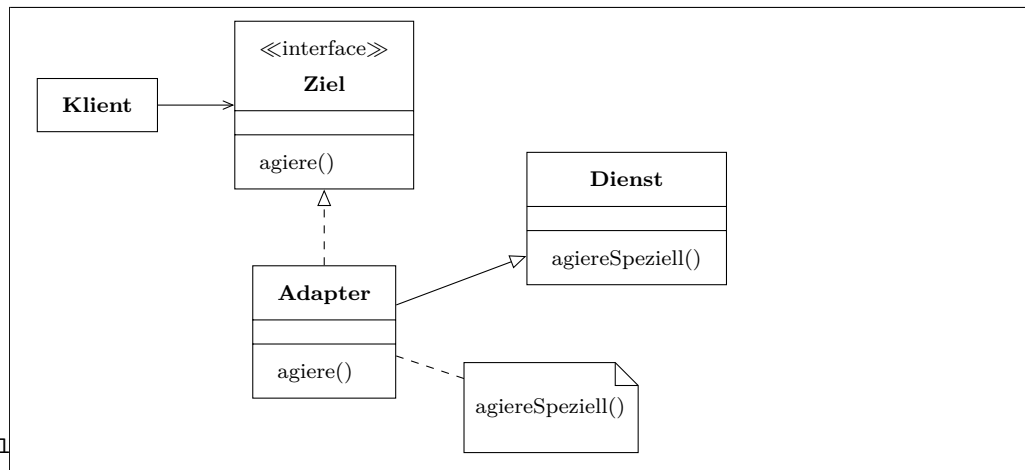
\liEntwurfsAbstrakteFabrik

```

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

```

#### 2.11.4 Adapter



\liEntwurfsAdapterUml

```

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

```

\liEntwurfsAdapterAkteure

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

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

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

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

```

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

```

\liEntwurfsAdapterCode

```

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

```

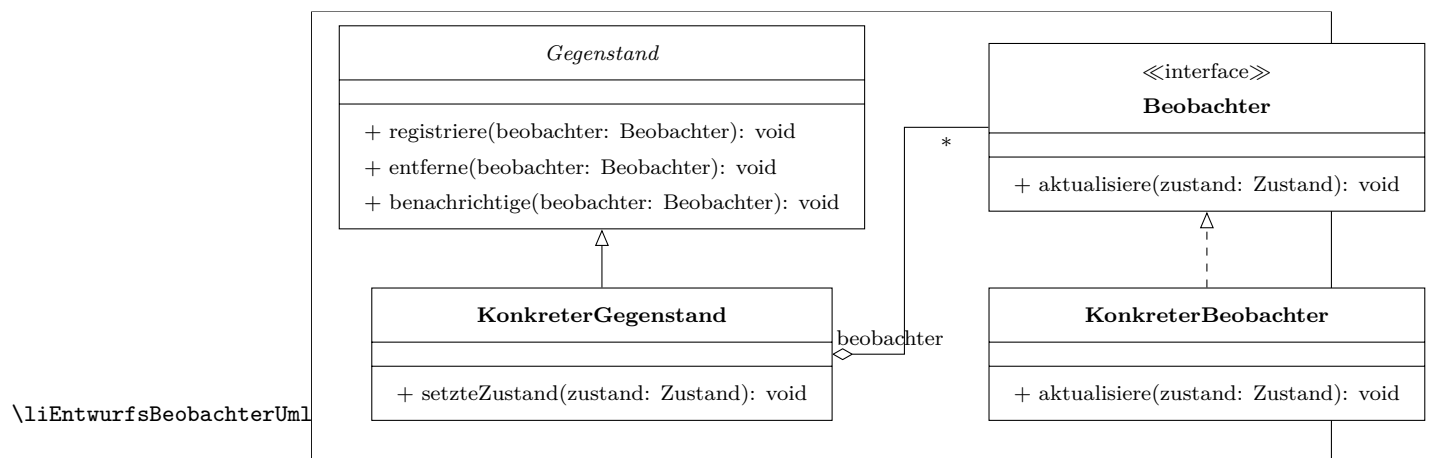
\liEntwurfsAdapter

```

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

```

### 2.11.5 Beobachter (Observer)



```

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

```

```

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

```

\liEntwurfsBeobachterAkteure

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

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

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

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

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

```

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

```



```

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

```

\liEntwurfsBeobachterCode

```

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

```

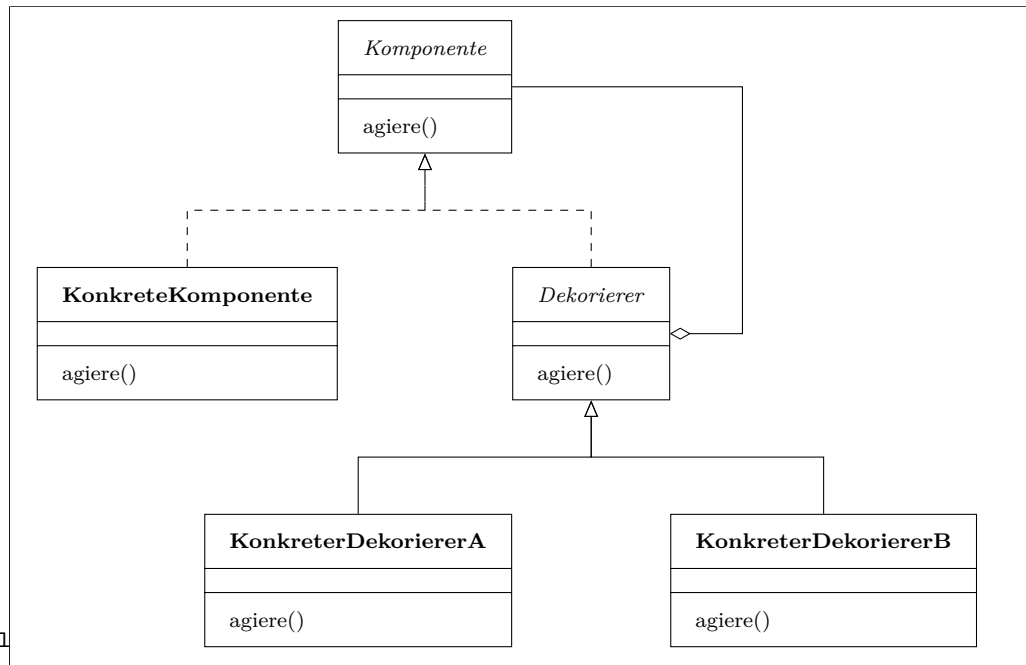
\liEntwurfsBeobachter

```

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

```

## 2.11.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

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

```

```

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

```

\liEntwurfsDekoriererCode

```

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

```

\liEntwurfsDekorierer

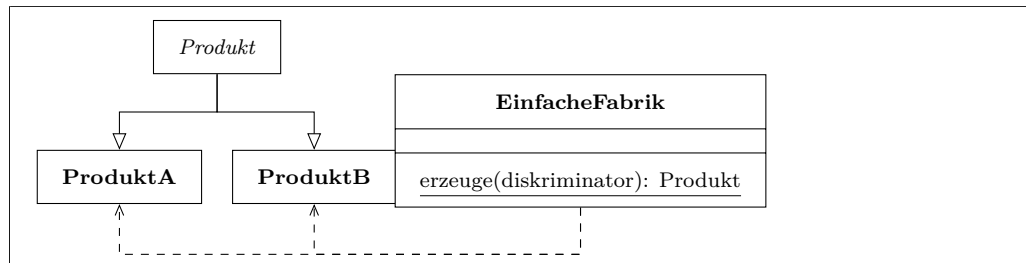
```

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

```

### 2.11.7 Einfache Fabrik (Simple Factory)

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



```

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

```

ntwurfsEinfacheFabrikAkteure Quelle: frei überstetzt aus GoF

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

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

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

```

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

\liEntwurfsEinfacheFabrik

```

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

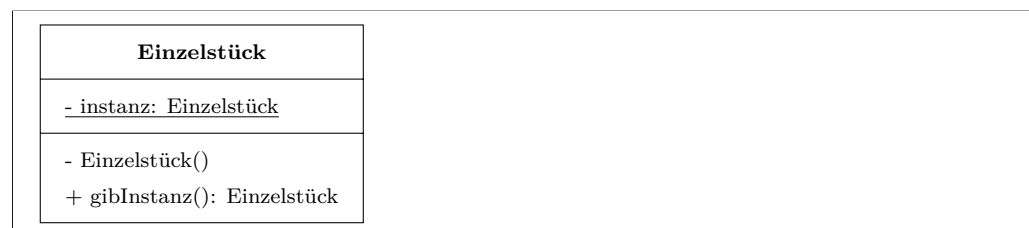
## 2.11.8 Einzelstück (Singleton)

urfsEinzelstueckBeschreibung

```

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

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```

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

iEntwurfsEinzelstueckAkteure Quelle: frei überstetzt aus GoF

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

```

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

```

\liEntwurfsEinzelstueckCode

```

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

```

\liEntwurfsEinzelstueck

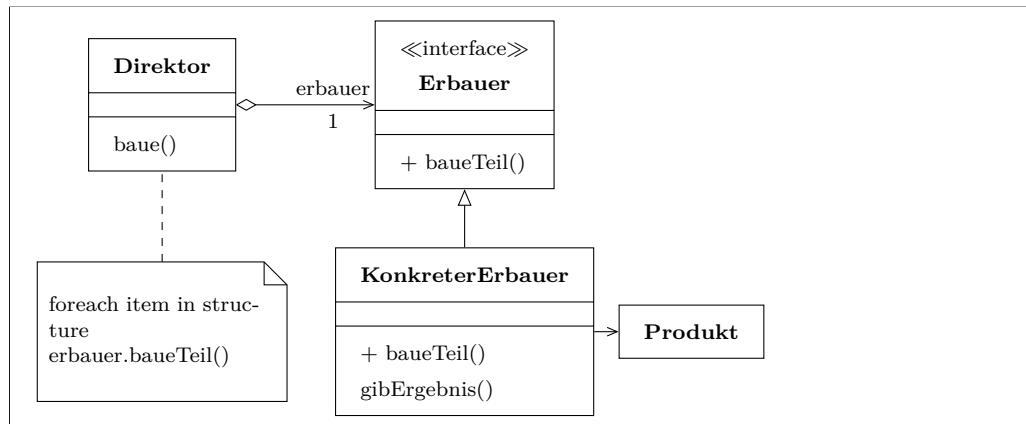
```

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

```

### 2.11.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

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

```

```

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

```

\liEntwurfsErbauerAkteure Quelle: deutsche Wikipedia

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

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

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

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

```

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

```

\liEntwurfsErbauer

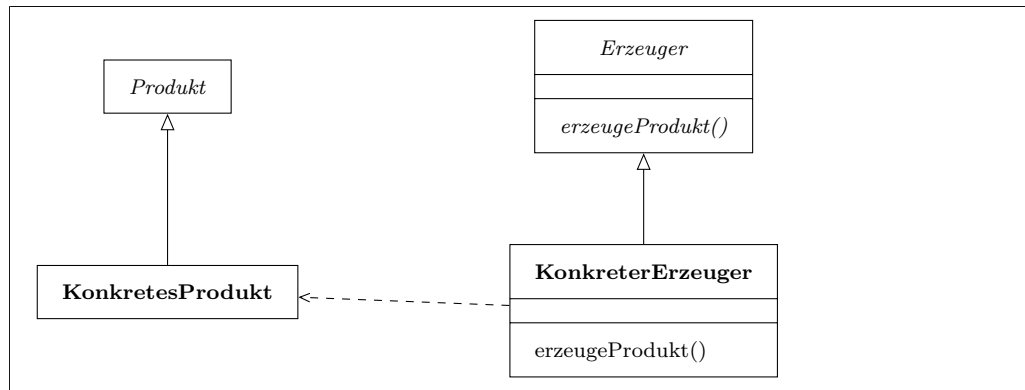
```

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

```

## 2.11.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

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

```

EntwurfsFabrikmethodeAkteure Quelle: deutsche Wikipedia

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

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

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

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

```

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

```

```

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

```

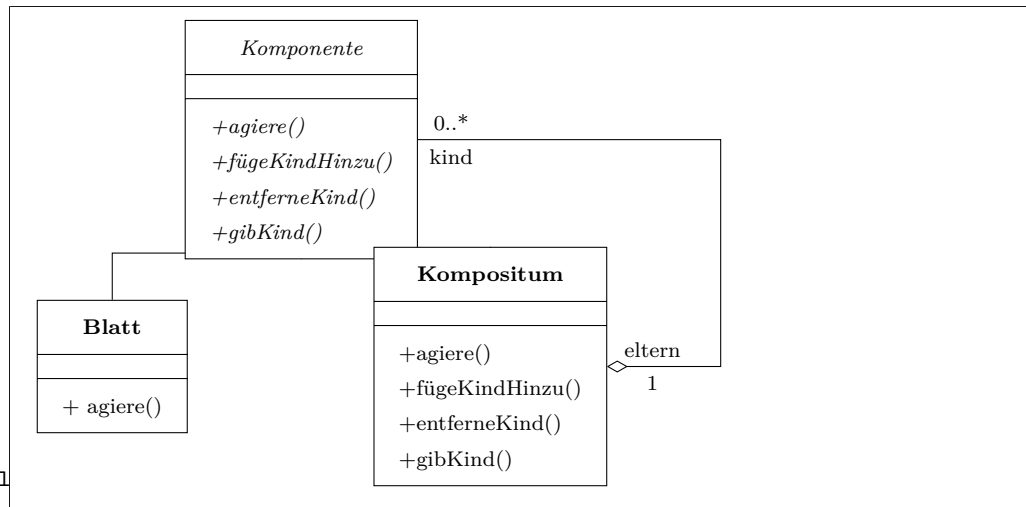
\liEntwurfsFabrikmethode

```

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

```

### 2.11.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

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

```

\liEntwurfsFabrikmethode

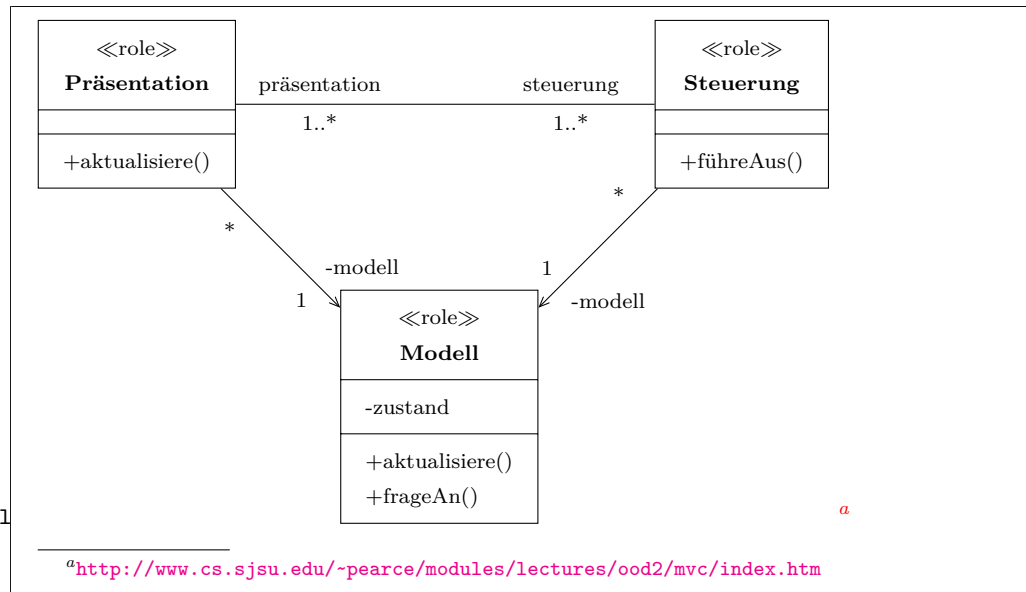
```

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

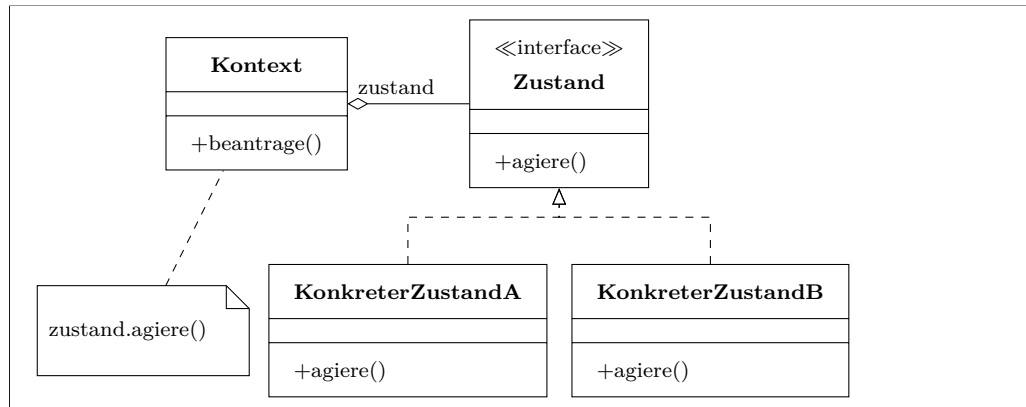
```

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.

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

```

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

```

```

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

```

\liEntwurfsZustandCode

```

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

```

\liEntwurfsZustand

```

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

```

```

1241

```

## 2.12 er.sty

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

```

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

```

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

```

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

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

```

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

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

### 2.12.3 Makro-Kürzel

```

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

```

```
1249 \ExplSyntaxOn
```

```
\liErEntity
```

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

```
\liErRelationship
```

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

```
\liErAttribute
```

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

```
\liErMpEntity mp = marginpar
```

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

```

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

```

□

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

```

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

```

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

```

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

```

```

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

1276 \ExplSyntaxOff
1277

```

## 2.13 formale-sprachen.sty

```

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

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

1284 \RequirePackage{hyperref}

1285 \liLadePakete{mathe,typographie}

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

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

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

1294 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

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

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

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

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

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

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

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

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

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

\liZustandsBuchstabe

1310 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1311 \def\liZustandsBuchstabeGross{Z}

```

```

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

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

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

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

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

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

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

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

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

1344 \ExplSyntaxOn

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

```

```

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

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

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

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

    • \liGrammatik{}:  $G = (V, \Sigma, P, S)$ 

    • \liGrammatik[G_1]{}:  $G_1 = (V, \Sigma, P, S)$ 

    • \liGrammatik{variablen={S,A,B}}:  $G = (\{S, A, B\}, \Sigma, P, S)$ 

    • \liGrammatik{alphabet={a,b}}:  $G = (V, \{a, b\}, P, S)$ 

    • \liGrammatik{start=X}:  $G = (V, \Sigma, P, X)$ 

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

```



## 2.14 formatierung.sty

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

### 2.14.1 Schriftarten / Typographie

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

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

### 2.14.2 Farben

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

### 2.14.3 Überschriften

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

### 2.14.4 Listen

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

### 2.14.5 Kasten

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

liKasten

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

### 2.14.6 Header

```
1419 \RequirePackage{fancyhdr}
1420 \fancyhead[L,C,R]{}
1421 \fancyfoot[L]{}
1422 \fancyfoot[C]{}
1423 \fancyfoot[R]{\thepage}
1424 \pagestyle{fancy}
1425 \renewcommand{\headrulewidth}{Opt}
1426 \renewcommand{\footrulewidth}{Opt}
1427
```

## 2.15 gantt.sty

```

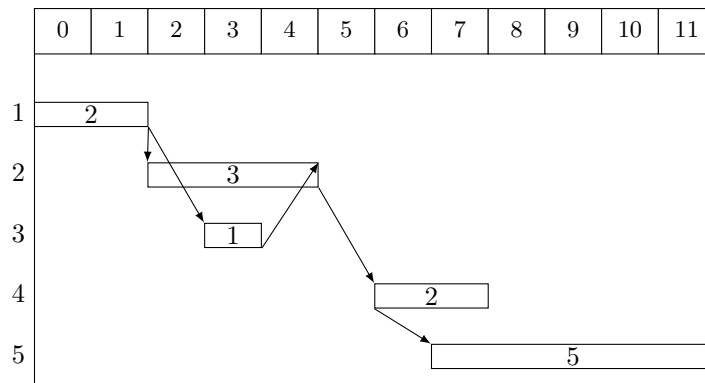
1428 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1429 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\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}

```



```

1430 \RequirePackage{tikz-uml}
1431 \RequirePackage{pgfgantt}
1432 \setganttlinklabel{f-s}{}
1433 \setganttlinklabel{s-s}{}
1434 \setganttlinklabel{f-f}{}
1435 \setganttlinklabel{s-f}{}

1436

```

## 2.16 grafik.sty

```
1437 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1438 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1439 TikZ, um Grafiken zeichnen zu können und graphicx um Bilder laden zu können.]
1440 \ExplSyntaxOn
1441 \RequirePackage{tikz}
1442 \RequirePackage{graphicx}

\liGrafikLogoPfad

1443 \def\liGrafikLogoPfad#1{
1444   \LehramtInformatikRepository / .tex / Logo / Logo_nur-Pfade.eps #1
1445 }

\liGrafikCCLizenz

1446 \NewDocumentCommand{ \liGrafikCCLizenz } { 0{ } } {
1447   \includegraphics[#1]{
1448     \liGrafikLogoPfad{CC-by-nc-sa.eps}
1449   }
1450 }

\liGrafikLogo

1451 \NewDocumentCommand{ \liGrafikLogo } { 0{ } } {
1452   \includegraphics[#1]{
1453     \liGrafikLogoPfad{Logo_nur-Pfade.eps}
1454   }
1455 }

1456 \ExplSyntaxOff
1457
```

## 2.17 graph.sty

```

1458 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1459 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1460 \RequirePackage{tikz}

```

Für die die Adjazenzliste (`xrightharpoon`)

```

1461 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

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

```

1462 \RequirePackage{blkarray}
1463 \usetikzlibrary{arrows.meta}

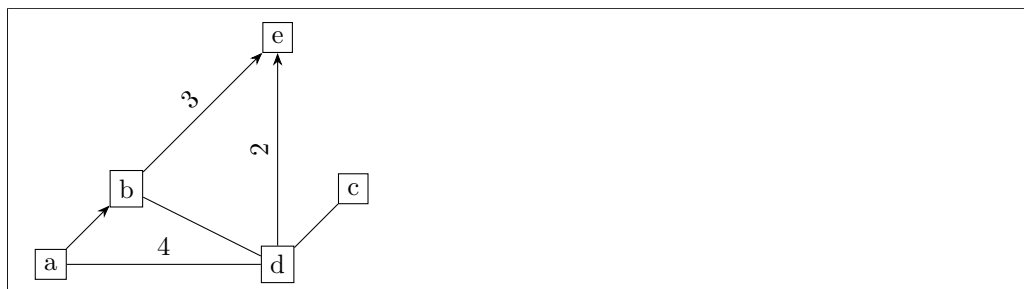
```

```

\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}

```



```

1464 \tikzset{
1465   li graph/.style={
1466     every node/.style={
1467       rectangle,
1468       draw,
1469     },
1470     every edge/.style={
1471       >={Stealth[black]},
1472       draw,
1473     },
1474     every edge/.append style={
1475       every node/.style={
1476         sloped,
1477         auto,
1478       }
1479     }
1480   },
1481   li markierung/.style={
1482     ultra thick,
1483   }
1484 }

```

**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}

```

```

1485 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1486

```

## 2.18 hanoi.sty

1487 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 1488 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen  
 1489 von Hanoi-Grafiken]

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

```
1490 \RequirePackage{tikz}
1491 \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}

1492 \def\li@mset #1[#2]=#3{%
1493 \expandafter\xdef\csname #1#2\endcsname{#3}
1494 }
1495 \def\li@mget #1[#2]{%
1496 \csname #1#2\endcsname
1497 }
1498 \def\li@minc #1[#2] += #3{%
1499 \pgfmathparse{\li@mget #1[#2] + #3}%
1500 \li@mset #1[#2] = \pgfmathresult
1501 }
1502
1503 \def\liHanoi#1#2{
1504   \edef\li@numdiscs{#1}
1505   \def\li@sequence{#2}
1506   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1507     % init colors
1508     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purple!80!black}{
1509       \li@mset col[\j]=\c;
1510     }
1511     % draw poles and init pole counters
1512     \foreach \j in {1,2,3}{
1513       \li@mset pos[\j]=0
1514       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1515     }
1516     % draw base
1517     \draw (.5,-.5) -- +(3,0);
1518     % draw discs
1519     \foreach[count=\k] \i/\j in \li@sequence{
1520       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*\i/\li@numdiscs,0);
1521       \li@minc pos[\j] += .5;
1522     }
1523   \end{tikzpicture}
1524 }
```

## 2.19 klassen-konfiguration-aufgabe.sty

```
1525 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1526 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-aufgabe}[2021/09/01 Die
1527 Klasse Aufgabe konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

Formatierung muss vor literatur sein, sonst Option clash

```
1528 \liLadePakete{
1529   formatierung,
1530   abmessung,
1531   literatur-dummy,
1532   makros,
1533   aufgaben-metadaten,
1534   kopf-fusszeilen,
1535   mathe,
1536   grafik
1537 }
1538
1539 % Komischer Option-Clash deshalb ganz am Ende, für die Silbentrennung
1540 \RequirePackage[ngerman]{babel}
1541 \ExplSyntaxOn
1542 \cs_gset:Npn \stichwoerter_formatiert: {
1543   \tl_if_empty:NTF \g_auf_stichwoerter_tl {} {
1544     \textbf{Stichwörter:} ~
1545     \g_auf_stichwoerter_tl
1546     \par
1547   }
1548 }
1549 \cs_gset:Npn \horizontale_linie: {
1550   \par
1551   \noindent
1552   \rule{\textwidth}{0.8pt}
1553   \par
1554 }
1555 \cs_gset:Npn \thematik_formatiert: {
1556   \tl_if_empty:NTF \g_auf_thematik_tl {} {
1557     \textit{
1558       ( \g_auf_thematik_tl )
1559     }
1560   }
1561 }
```

\liAufgabenMetadaten

```
1562 \def\liAufgabenMetadaten#1{
1563   \liMetaSetze{#1}
1564
1565   \noindent
1566   {\large \_gib_einzelpruefung_trenner: \par}
1567
1568   \medskip
1569
1570   \noindent
1571   {\bfseries\Large\_gib_aufgaben_pfad_trenner:}
1572   \hfill \thematik_formatiert:
1573   \par
1574
1575   \medskip
1576
1577   \noindent
1578   {\footnotesize\stichwoerter_formatiert:}
1579
1580   \horizontale_linie:
1581 }
```

```

1582 \bigskip
1583
1584 \par
1585 % Keine Einrückung
1586 \@afterindentfalse
1587 \@afterheading
1588 }

1589 \AddToHook{enddocument}{
1590 \vfill
1591 {
1592 \liGrafikLogo
1593 \liGrafikCCLizenz
1594 \tiny
1595 \noindent
1596 Hilf~mit!~
1597 Das~ist~ein~Community-Projekt.~
1598 Verbesserungsvorschläge,~Fehlerkorrekturen,~weitere~Lösungen~sind~
1599 sehr~willkommen~~~egal~wie~~~per~Pull-Request~oder~per~E-Mail~an~
1600 \href{mailto:hermine.bsclangaul@gmx.net}{hermine.bsclangaul@gmx.net}~
1601
1602 \par\noindent
1603 Der~\TeX-Quelltext~dieses~PDFs~kann~unter~folgender~URL~aufgerufen~werden:~
1604
1605 \par\noindent\hfill
1606 \_gib_github_url_href:}
1607 }

1608 \ExplSyntaxOff
1609

```



## 2.20 klassen-konfiguration-examen.sty

```
1610 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1611 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-examen}[2021/09/04 Die
1612 Klasse liSetzeExamen konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

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

```
1613 \liLadePakete{
1614   formatierung,
1615   literatur-dummy,
1616   makros,
1617   aufgaben-metadaten,
1618   abmessung,
1619   typographie,
1620   grafik
1621 }
```

Formatierung für die Überschriften setzen.

```
1622 \RequirePackage{titlesec}
1623 \titleformat{\section}{\huge\filcenter\bfseries}{\thesection}{1em}{}
1624 \newcommand{\sectionbreak}{\clearpage}
1625 \setcounter{secnumdepth}{0}
1626 \liLadeAllePakete
```

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

```
1627 \RequirePackage[ngerman]{babel}
1628 \RequirePackage{standalone}
1629 \ExplSyntaxOn
```

`\liSetzeExamenThemaNr`

```
1630 \def\liSetzeExamenThemaNr#1{
1631   \tl_gset:Nn \g_auf_examen_thema_nr_tl { #1 }
1632   \section{Thema-Nr.~#1}
1633 }
```

`\liSetzeExamenTeilaufgabeNr`

```
1634 \def\liSetzeExamenTeilaufgabeNr#1{
1635   \tl_gset:Nn \g_auf_examen_teilaufgabe_nr_tl { #1 }
1636   \subsection{Teilaufgabe-Nr.~#1}
1637 }
```

`\liBindeAufgabeEin`

```
1638 \def\liBindeAufgabeEin#1{
1639   \tl_gset:Nn \g_auf_examen_aufgabe_nr_tl { #1 }
1640   \input{
1641     \LehramtInformatikRepository /
1642     Staatsexamen /
1643     \g_auf_examen_nummer_tl /
1644     \g_auf_examen_jahr_tl /
1645     \g_auf_examen_monat_tl /
1646     \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
1647       Thema - \g_auf_examen_thema_nr_tl /
1648     }
1649     \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
1650       Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl /
1651     }
1652     Aufgabe - \g_auf_examen_aufgabe_nr_tl .tex
1653   }
1654 }
```

`\liAufgabenMetadaten` Das Metadaten-Makro überschreiben

```
1655 \def\liAufgabenMetadaten#1{
1656   \liMetaSetze{#1}
1657   \subsubsection{\_gib_aufgaben_titel:}
1658 }
```

```

1659 \setcounter{tocdepth}{4}
1660 \RequirePackage[titles]{tocloft}
1661 \AddToHook{begindocument}{
1662   \pagestyle{empty}
1663   \begin{center}
1664     \large
1665     Erste-Staatsprüfung-für-ein-Lehramt-an-öffentlichen-Schulen \par
1666
1667     \vspace{0.5cm}
1668
1669     Fach-Informatik \par
1670
1671     \vfill
1672
1673     \liGrafikLogo[width=8cm]\par
1674     \bigskip
1675     Die-Bschlangaul-Sammlung \par
1676     {\footnotesize zusammengestellt-von-Hermine-Bschlangaul-und-Freunden} \par
1677
1678     \vfill
1679
1680     {
1681       \bfseries\Huge
1682
1683       \g_auf_examen_jahreszeit_tl \par
1684
1685       \g_auf_examen_jahr_tl \par
1686     }
1687
1688     \vspace{2cm}
1689
1690     {\LARGE \g_auf_examen_nummer_tl \par}
1691
1692     \vspace{0.5cm}
1693
1694     \g_auf_examen_fach_tl \par
1695
1696     \vspace{3cm}
1697
1698     Aufgabenstellungen-mit-Lösungsvorschlägen \par
1699
1700     \end{center}
1701
1702     % für den Abstand vor den section im Inhaltsverzeichnis
1703     % https://tex.stackexchange.com/questions/241445/how-to-control-spacing-in-toc-
1704     % for-different-sections
1705     \setlength{\cftbeforesecskip}{1.5cm}
1706     \setlength{\cftbeforesubsecskip}{0.5cm}
1707     \renewcommand{\cftsubsecafterpnum}{\vspace{\cftbeforesubsecskip}}
1708     \setlength{\cftbeforesubsubsecskip}{0.1cm}
1709
1710     \vfill
1711     \renewcommand{\contentsname}{Aufgabenübersicht}
1712     \tableofcontents
1713   }
1714 \ExplSyntaxOff
1715

```

## 2.21 komplexitaetstheorie.sty

```
1716 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1717 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1718 Setzen von Karp's NP-vollständigen Problemen, Hilfsmakros für die
1719 Polynomialzeitreduktion.]
```

### 2.21.1 Makro-Kürzel

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

```
1720 \liLadePakete{mathe}
```

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

```
1721 \RequirePackage{mdframed}
```

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

```
1722 \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`

```
1723 \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`

```
1724 \def\liProblemBeschreibung#1#2#3{
1725   \begin{mdframed}[
1726     userdefinedwidth=9cm,
1727     align=center,
1728     backgroundcolor=white!0,
1729   ]
1730   \centerline{\large\liProblemName{#1}}
1731
1732   \medskip
1733
1734   \begin{description}
1735     \item[Gegeben:] #2
1736     \item[Frage:] #3
1737   \end{description}
1738 \end{mdframed}
1739 }
```

```

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

1740 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1741 \begin{displaymath}
1742 \liProblemName{#1}
1743 \preceq_{#2}
1744 \liProblemName{#3}
1745 \end{displaymath}
1746 }

\liProblemVertexCover

1747 \def\liProblemClique{%
1748 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1749 Mindestgröße  $n$  in einem gegebenen Graphen.
1750 \footcite[Seite 76]{theo:fs:4}
1751 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1752 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1753 \footcite{wiki:cliquenproblem}
1754 }

\liProblemVertexCover

1755 \def\liProblemVertexCover{%
1756 %
1757 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1758 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1759 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1760 \footcite{wiki:knotenueberdeckung}
1761
1762 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1763 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1764 mindestens einem Knoten aus  $U$  verbunden ist.
1765 \footcite[Seite 78]{theo:fs:4}%
1766 }

\liProblemSubsetSum Kein Karp-21-Problem

1767 \def\liProblemSubsetSum{%
1768 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1769 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1770 \footcite{wiki:teilsommenproblem}
1771 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1772 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1773 größer als eine gegebene obere Schranke  $c$  ist.
1774 \footcite[Seite 74]{theo:fs:4}
1775 }

\liProblemSubsetSum Kein Karp-21-Problem

1776 \def\liProblemSat{%
1777 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1778 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1779 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1780 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1781 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1782 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1783 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1784 aufgestellt werden.
1785 \footcite[Seite 71]{theo:fs:4}
1786 }

1787

```

## 2.22 kontrollflussgraph.sty

1788 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1789 \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 = ...}]
```

```

1790 \RequirePackage{tikz}
1791 \usetikzlibrary{positioning}
1792 \tikzset{
1793   li kontrollfluss/.style={
1794     knoten/.style={
1795       circle,
1796       draw
1797     },
1798     usebox/.style={
1799       draw,
1800       rectangle,
1801       font=\scriptsize,
1802       anchor=west,
1803       align=left,
1804     },
1805     bedingung/.style={
1806       midway,
1807       draw=none,
1808       font=\scriptsize
1809     },
1810     knotenbeschriftung/.style={
1811       draw,
1812       rectangle,
1813       midway,
1814       font=\scriptsize
1815     },
1816     wahr/.style={
1817       thick
1818     },
1819     falsch/.style={
1820       dashed
1821     },
1822     every node/.style={
1823       circle,
1824       draw,
1825     },
1826     every edge/.append style={
1827       every node/.style={
1828         draw=none,
1829         bedingung,
1830       }
1831     },
1832     every path/.style={
1833       draw,
1834       ->,
1835     },
1836     every pin/.style={
1837       draw,
1838       dotted,
1839       rectangle,
1840       pin position=right
1841     },
1842     every pin edge/.style={
1843       dotted,
1844       arrows=-,
1845     }
1846   }
1847 }

```

#### 2.22.4 Umgebungen

liKontrollflussgraph

```

1848 \NewDocumentEnvironment { liKontrollflussgraph } { 0{} } {

```

```

1849 \begin{tikzpicture}[
1850     li kontrollfluss,
1851     #1
1852 ]
1853 } {
1854 \end{tikzpicture}
1855 }

```

## 2.22.5 Makros

`\liAnweisung`

```
1856 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}
```

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

```
1857 \def\liBedingung#1#2{\node[bedingung,#1]{\texttt{#2}}}
```

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

```
1858 \def\liBedingungWahr#1{\node[bedingung,#1]{\texttt{[true]}}}
```

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

```
1859 \def\liBedingungFalsch#1{\node[bedingung,#1]{\texttt{[false]}}}
```

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

```
1860 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}
```

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

```
1861 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw
```

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

```
1862 \ExplSyntaxOn
```

```
1863 \NewDocumentCommand { \liKontrollKnotenPfad } { m }
```

```
1864 {
```

```
1865   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
```

```
1866   \seq_set_map:NNn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
```

```
1867   \seq_use:Nn \l_tmpa_seq { ~~~~ }
```

```
1868 }
```

```
1869 \ExplSyntaxOff
```

```
1870
```

## 2.23 kopf-fusszeilen.sty

```
1871 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1872 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1873 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1874 \ExplSyntaxOn

1875 \fancyhead{}
1876 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1877 \fancyhead[L0,RE]{\scriptsize\today}
1878 \fancyfoot{}
1879 \fancyfoot[LE,R0]{\thepage}
1880 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1881 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1882 \renewcommand{\headrulewidth}{0.4pt}
1883 \renewcommand{\footrulewidth}{0.4pt}
1884 \setlength{\headwidth}{\textwidth}

1885 \ExplSyntaxOff

1886
```



## 2.24 literatur-dummy.sty

```
1887 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1888 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1889 \def\literatur{}

\footcite

1890 % \RequirePackage[stable,multiple]{footmisc}
1891 \NewDocumentCommand{ \footcite } { o m }{}

1892
```

## 2.25 literatur.sty

```
1893 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1894 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1895 \RequirePackage{csquotes}
1896 \RequirePackage[
1897   bibencoding=utf8,
1898   citestyle=authortitle,
1899   backend=biber,
1900 ]{biblatex}
1901 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1902 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1903 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1904 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1905 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1906 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1907 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1908 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1909 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1910 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1911 % To allow footnotes in the heading
1912 \RequirePackage[stable,multiple]{footmisc}

\literatur

1913 \def\literatur{\printbibliography[heading=subbibliography]}

1914
```

## 2.26 makros.sty

```
1915 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1916 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1917 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1918 anderen Paket passen]
1919 \RequirePackage{hyperref}
1920 \RequirePackage{graphicx}
    Für die Umgebung liQuellen benötigt.
1921 \RequirePackage{paralist}
1922 \ExplSyntaxOn

\inhaltsverzeichnis
1923 \def\inhaltsverzeichnis {
1924   \begin{mdframed}
1925     \begin{group}
1926       \let\clearpage\relax
1927       \tableofcontents
1928     \end{group}
1929   \end{mdframed}
1930 }

\memph \mephm (\marginpar and \emph)
1931 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1932 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1933 \newcommand{\liPseudoUeberschrift}[1]{
1934   \bigskip
1935
1936   \par
1937   \noindent
1938   \textsf{\textbf{#1}}
1939
1940   \medskip
1941
1942   \par
1943   % Keine Einrückung
1944   \@afterindentfalse
1945   \@afterheading
1946 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1947 \newcommand{\liBeschriftung}[1]{
1948   \par
1949   \noindent
1950   \medskip
1951   \textbf{#1}:
1952   \medskip
1953   \noindent
1954 }

\hinweis
1955 \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.
1956 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}
```

## liEinbettung

```
1957 \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.

```
1958 \NewDocumentEnvironment{ liAntwort } { O{standard} }
1959 {
1960   \str_case:nn {#1} {
1961     {standard} {
1962       \def\beschriftung{}
1963       \mdfsetup{backgroundcolor=gray!1,linecolor=gray}
1964     }
1965     {richtig} {
1966       \def\beschriftung{richtig}
1967       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1968     }
1969     {falsch} {
1970       \def\beschriftung{falsch}
1971       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1972     }
1973     {muster} {
1974       \def\beschriftung{Musterlösung}
1975       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1976     }
1977   }
1978   \ifx\beschriftung\empty\else
1979     \noindent
1980     \textbf{\beschriftung}{}:}
1981   \fi
1982   \begin{mdframed}
1983 }
1984 {\end{mdframed}}
```

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

```
1985 \NewDocumentEnvironment{ liAdditum } { o }
1986 {
1987   \begin{mdframed}[backgroundcolor=yellow!5]
1988   \IfNoValueTF {#1}
1989   { \liPseudoUeberschrift{Additum} }
1990   { \liPseudoUeberschrift{Additum:~#1} }
1991 }
1992 {\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.

```
1993 \NewDocumentEnvironment{ liExkurs }{o +b}{
1994   \vspace{0.2cm}%
1995   \begin{mdframed}[
1996     backgroundcolor=white,
1997     bottomline=false,
1998     innermargin=1cm,
1999     leftline=true,
2000     linecolor=black,
```

```

2001 linewidth=0.1cm,
2002 outermargin=1cm,
2003 rightline=false,
2004 topline=false,
2005 ]
2006 \footnotesize
2007 \noindent%
2008 \textbf{Exkurs:~#1}\par%
2009 \noindent%
2010 #2
2011 \end{mdframed}
2012 \vspace{0.2cm}
2013 }{}

```

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

```

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

```

**Weiterführende Literatur:**

- Quelle 1
- Quelle 2

```

2014 \cs_new:Npn \listen@punkt #1 {\item #1}
2015 \NewDocumentEnvironment { liQuellen }{ +b }
2016 {
2017   \seq_clear_new:N \l_quellen
2018   \seq_set_split:Nnn \l_quellen {\item} {#1}
2019   \seq_remove_all:Nn \l_quellen {}
2020   \begin{mdframed}[backgroundcolor=white!0]
2021     \footnotesize
2022     \noindent
2023     \textsf{\textbf{Weiterführende~Literatur:}}
2024     \medskip
2025     \begin{compactitem}
2026       \seq_map_function:NN \l_quellen {\listen@punkt}
2027     \end{compactitem}
2028   \end{mdframed}
2029   %
2030   \par
2031   \@afterindentfalse
2032   \@afterheading
2033 } {}

```

liLernkartei

```

2034 \NewDocumentEnvironment { liLernkartei }{ m +b }
2035 {
2036   \begin{mdframed}
2037     \footnotesize
2038     \noindent%
2039     \textbf{Lernkarteikarte:~#1}\par%
2040     \noindent%
2041     #2
2042   \end{mdframed}
2043 } {}

```

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

```

2044 \NewDocumentEnvironment { liDiagramm }{ m +b }
2045 {

```

```

2046 \begin{mdframed}[backgroundcolor=white!0]
2047 \small
2048 \noindent%
2049 \textit{\#1}:
2050 \begin{center}
2051 #2
2052 \medskip
2053 \end{center}
2054 \end{mdframed}
2055 } {}

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

2056 \NewDocumentCommand{\liFussnoteUrl} { o m } {
2057 \footnote{\url{\#2}\IfNoValueTF{\#1}{-}{ (\#1)}}
2058 }
2059

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

2060 \NewDocumentCommand{\liFussnoteLink} { o m m } {
2061 \footnote{\href{\#3}{\#2}\IfNoValueTF{\#1}{-}{ (\#1)}}
2062 }


\zB
2063 \def\zB{z.\.,B. }

\ZB
2064 \def\ZB{Z.\.,B. }

\dH
2065 \def\dH{d.\.,h. }

2066 \ExplSyntaxOff
2067
```

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

```

2056 \NewDocumentCommand{\liFussnoteUrl} { o m } {
2057   \footnote{\url{#2}\IfNoValueTF{#1}{-}{( #1)}}
2058 }
2059

```

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

```

2060 \NewDocumentCommand{\liFussnoteLink} { o m m } {
2061   \footnote{\href{#3}{#2}\IfNoValueTF{#1}{-}{ (#1)}}
2062 }

```

 $\backslash zB$ 

2063 \def\zB{z.\,B. }

 $\backslash ZB$ 

2064 \def\ZB{Z.\,B. }

 $\backslash dh$ 

```
2065 \def\dh{d.\,h. }
```

2066 \ExplSyntaxOff

2067

## 2.27 master-theorem.sty

2068 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2069 \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}

2070 \ExplSyntaxOn

2071 \RequirePackage{amsmath}

\liRundeKlammer

2072 \def\liRundeKlammer#1{

2073 \negthinspace \left( #1 \right)

2074 }

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

2075 \def\liThetaOhneMathe#1{

2076 \Theta \liRundeKlammer{#1}

2077 }

2078 \def\liTheta#1{

2079 \ifmmode

2080 \liThetaOhneMathe{#1}

2081 \else

2082 \$\liThetaOhneMathe{#1}\$

2083 \fi

2084 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
2085 \def\liOmegaOhneMathe#1{
2086 \Omega \liRundeKlammer{#1}
2087 }
2088 \def\liOmega#1{
2089 \ifmmode
2090 \liOmegaOhneMathe{#1}
2091 \else
2092 $\liOmegaOhneMathe{#1}$
2093 \fi
2094 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
2095 \def\liOOhneMathe#1{
2096 \mathcal{O} \liRundeKlammer{#1}
2097 }
2098 \def\liO#1{
2099 \ifmmode
2100 \liOOhneMathe{#1}
2101 \else
2102 $\liOOhneMathe{#1}$
2103 \fi
2104 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
2105 \def\liTOhneMathe#1#2{
2106 \tl_if_blank:nTF {#1}
2107 {}
2108 {#1 \cdot }
2109 T
2110 \liRundeKlammer{\frac{n}{#2}}
2111 }
2112 \def\liT#1#2{
2113 \ifmmode
2114 \liTOhneMathe{#1}{#2}
2115 \else
2116 $\liTOhneMathe{#1}{#2}$
2117 \fi
2118 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
2119 \def\liRekursionsGleichung{
2120 $T(n) = \liT{a}{b} + f(n)$
2121 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
2122 \def\liBedingungEins{
2123 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
2124 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
2125 \def\liBedingungZwei{
2126 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
2127 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
2128 \def\liBedingungDrei{
2129 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
2130 }
2131 \ExplSyntaxOff

```



\liMasterVariablen

```

2132 \def\liMasterVariablen{
2133   \begin{displaymath}
2134     T(n) = \liT{a}{b} + f(n)
2135   \end{displaymath}
2136
2137   \begin{itemize}
2138     \item[$a = $]
2139       Anzahl der rekursiven Aufrufe, Anzahl der Unterprobleme in der
2140       Rekursion
2141       ($a \geq 1$).
2142
2143     \item[$\textstyle{\frac{1}{b}} = $]
2144       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
2145       repräsentiert wird, Anteil an der Verkleinerung des Problems ($b > 1$).
2146
2147     \item[$f(n) = $]
2148       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
2149       die Kombination der Teillösungen entstehen. Eine von $T(n)$
2150       unabhängige und nicht negative Funktion.
2151   \end{itemize}
2152   \footcite{wiki:master-theorem}
2153   \footcite[Seite 19-35]{aud:fs:2}
2154 }
```

\liMasterFaelle

```

2155 \def\liMasterFaelle{
2156   \begin{description}
2157     \item[1. Fall:]
2158       $T(n) \in \liTheta{n^{\log\sb{b}a}}$
2159
2160       \hfill falls \liBedingungEins
2161       für $\varepsilon > 0$
2162
2163     \item[2. Fall:]
2164       $T(n) \in \liTheta{n^{\log\sb{b}a} \cdot \log n}$
2165
2166       \hfill falls \liBedingungZwei
2167
2168     \item[3. Fall:]
2169       $T(n) \in \liTheta{f(n)}$
2170
2171       \hfill falls \liBedingungDrei
2172       für $\varepsilon > 0$
2173       und ebenfalls für ein $c$ mit $0 < c < 1$ und alle hinreichend großen $n$
2174       gilt:
2175       $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$
2176   \end{description}
2177 }
```

\liMasterVariablenDeklaration

```

2178 \def\liMasterVariablenDeklaration#1#2#3{
2179   \begin{description}
2180     \item[Allgemeine Rekursionsgleichung:] \strut
2181
2182     \liRekursionsGleichung
2183
2184     \item[Anzahl der rekursiven Aufrufe ($a$):] \strut
2185
2186     #1
2187
2188     \item[Anteil Verkleinerung des Problems ($b$):] \strut
2189   \end{description}
```

```

2190     um  $\frac{1}{b}$  also  $b = \frac{1}{a}$ 
2191
2192     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut
2193
2194      $T(n) = \frac{1}{b} + T(\frac{n}{b})$ 
2195
2196     \item[Ergibt folgende Rekursionsgleichung:] \strut
2197
2198      $T(n) = \frac{1}{b} + T(\frac{n}{b})$ 
2199 \end{description}
2200 }

\liMasterFallRechnung
2201 \def\liMasterFallRechnung#1#2#3{
2202   \begin{description}
2203     \item[1. Fall:] \liBedingungEins:
2204
2205     #1
2206
2207     \item[2. Fall:] \liBedingungZwei:
2208
2209     #2
2210
2211     \item[3. Fall:] \liBedingungDrei:
2212
2213     #3
2214   \end{description}
2215 }

\liMasterExkurs
2216 \def\liMasterExkurs{
2217   \begin{liExkurs}[Master-Theorem]
2218     \liMasterVariablen
2219
2220     \noindent
2221     Dann gilt:
2222
2223     \liMasterFaelle
2224   \end{liExkurs}
2225 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
2226 \def\liMasterWolframLink#1{
2227   Berechne die Rekursionsgleichung auf WolframAlpha:
2228   \href{https://www.wolframalpha.com/input/?i=#1}{WolframAlpha}
2229 }

2230

```

## 2.28 mathe.sty

```
2231 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2232 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
2233
2234 % for example \ltimes \rtimes
2235 %\RequirePackage{amssymb}
2236 \RequirePackage{amsmath}
2237
2238 %%
2239 % \mlq \mrq
2240 %%
2241 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
2242 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
2243
```

## 2.29 minimierung.sty

```

2244 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2245 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
2246 Minimierungsalgorithmus von einem NEA zu einem DEA]

2247 \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

2248 \def\liFussnote#1{\$x_{#1}\$}

2249 \def\li@fussnote@text#1#2{
2250 \liFussnote{#1}
2251 \quad
2252 {\footnotesize #2}
2253 }

\liFussnoteEinsText
2254 \def\liFussnoteEinsText{
2255 \li@fussnote@text{1}
2256 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
2257 }

\liFussnoteZweiText
2258 \def\liFussnoteZweiText{
2259 \li@fussnote@text{2}
2260 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
2261 }

\liFussnoteDreiText
2262 \def\liFussnoteDreiText{
2263 \li@fussnote@text{3}

```

```

2264 {In weiteren Iterationen markierte Zustände.}
2265 }

\liFussnoteVierText
2266 \def\liFussnoteVierText{
2267   \li@fussnote@text{4}
2268   {...}
2269 }

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



2270 \def\liFussnoten{
2271   \bigskip
2272
2273   \noindent
2274   \liFussnoteEinsText
2275
2276   \noindent
2277   \liFussnoteZweiText
2278
2279   \noindent
2280   \liFussnoteDreiText
2281
2282   \noindent
2283   \liFussnoteVierText
2284 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
2285 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
2286 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
2287 \def\liZustandsPaar#1#2{
2288   $(
2289     \liZustandsPaarVariablenName_#1,
2290     \liZustandsPaarVariablenName_#2
2291   )$
2292 }

liUebergangsTabelle
2293 \renewcommand{\arraystretch}{1.4}
2294 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
2295   \liPseudoUeberschrift{Übergangstabelle}
2296   \begin{center}
2297     \begin{tabular}{r|l|l}
2298       \textbf{Zustandspaar} & \textbf{\#1} & \textbf{\#2} \\ \hline
2299     \end{tabular}
2300   \end{center}
2301 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle:

Minimierungstabelle (Table filling)
2303 \ExplSyntaxOn

```

```

2304 \def\liUeberschriftDreiecksTabelle{
2305   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
2306 }

```

\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. —

```

2307 \def\liMinimierungErklaerung{
2308   %\footcite[Seite~19]{koenig}
2309   \liParagraphMitLinien{
2310     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
2311     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
2312     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
2313     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
2314      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
2315     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
2316     somit~zueinander~( $k-1$ )-äquivalent,~aber~nicht~ $k$ -äquivalent~und~
2317     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
2318     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
2319   }
2320 }
2321 \ExplSyntaxOff

```

2322

## 2.30 normalformen.sty

```

2323 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2324 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10]
2325 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
2326 Attributhülle]

```

Mathe für textit tex etc, typographie für geschweifte Klammer

```

2327 \liLadePakete{mathe,typographie}
2328 \directlua{
2329   helfer = require('lehramt-informatik-helfer')
2330   normalformen = require('lehramt-informatik-normalformen')
2331 }

```

### 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

```

```

2332 \def\liTeilen#1{
2333   \directlua{tex.print(normalformen.teilen('#1'))}
2334 }

```

```

\liAttributHuelle Let-Abkürzung: \let\ah=\liAttributHuelle
                  \ah{F, \m{A, B}} AttrHülle( $F, \{A, B\}$ ) Regulärer Ausdruck zum Konvertieren
                  AttrHülle((.*)\ ) \ah{$1}
2335 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(#1)}
2336 \def\liAttributHuelle#1{
2337   \ifmmode
2338     \liAttributHuelleOhneMathe{#1}
2339   \else
2340     $\liAttributHuelleOhneMathe{#1}$
2341   \fi
2342 }

```

```

\liAttributMenge Let-Abkürzung: \let\m=\liAttributMenge
2343 \def\liAttributMenge#1{\{ \textit{#1} \}}

```

liAHuelle

```

2344 \NewDocumentEnvironment{ liAHuelle } { +b } {
2345   \begingroup
2346   \footnotesize
2347   \begin{multline*}
2348     #1
2349   \end{multline*}
2350   \endgroup
2351 } { }

```

```

\liLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multline
Let-Abkürzung: \let\ahL=\liLinksReduktion
                  \ahL{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}
2352 \def\liLinksReduktion#1#2#3{
2353   \shoveleft{
2354     \liAttributHuelleOhneMathe{FA,
2355       \liAttributMenge{#1 \string\ #2}} =
2356   } \\\

```

```

2357 \shoveright{
2358   \liAttributMenge{#3}
2359 } \\\
2360 }

```

```

\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( $F, \{A, E \setminus E\}$ ) =  $\{A, E, F, B, D\}$ 
2361 \def\liLinksReduktionInline#1#2#3{%
2362   {%
2363     \footnotesize%
2364     $\liAttributHuelleOhneMathe{F,
2365       \liAttributMenge{#1} \string\ #2}} =
2366       \liAttributMenge{#3}$
2367   }
2368 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahr=\liLinksReduktionInline
    \ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
2369 \def\liRechtsReduktionInline#1#2#3#4{%
2370   {%
2371     \footnotesize%
2372     $\liAttributHuelleOhneMathe{
2373       F \setminus
2374       \liFunktionaleAbhaengigkeit{#1}
2375       \def\tmp{#2}\ifx\tmp\empty
2376       \else
2377         \cup \liFunktionaleAbhaengigkeit{#2}
2378       \fi
2379       ,
2380       \liAttributMenge{#3}
2381     } =
2382     \liAttributMenge{#4}$
2383   }
2384 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
    \fa{A, B -> C, D}:  $\{A, B\} \rightarrow \{C, D\}$  Regulärer Ausdruck zum Konvertieren:
    \$(.*?) \\\rightarrow (.*)\$ \fa{$1 -> $2}
2385 \def\liFunktionaleAbhaengigkeit#1{%
2386   \directlua{normalformen.drucke_funk_abhaengigkeit('#1', false)}%
2387 }

```

```

\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{.$1 -> $2}

```



```

2388 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
2389   \liGeschweifteKlammern
2390   {#1}
2391   {
2392     \begin{align*}
2393       \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
2394     \end{align*}
2395   }
2396   {-0.5cm}
2397   {-1.7cm}
2398 }

\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}

2399 \NewDocumentCommand {\liRelation} { O{R} m } {
2400   $\directlua{
2401     local name = helper.konvertiere_tiefgestellt('#1')
2402     tex.print(name)
2403   }$(\textit{\, #2\,})
2404 }

2405

```

## 2.31 o-notation.sty

2406 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2407 \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}
```

2408 \ExplSyntaxOn

2409 \RequirePackage{amssymb}

2410 \RequirePackage{pgfplots}

Für echte Teilmenge \subsetneq:  $\subsetneq$

2411 \RequirePackage{amssymb}

\liRundeKlammer

```
2412 \def\liRundeKlammer#1{
2413   \negthinspace \left( #1 \right)
2414 }
```

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

```
2415 \cs_new:Npn \o_notation_O:n #1 {
2416   \mathcal{O} \liRundeKlammer{#1}
2417 }
2418 \def\liONotationO#1{
2419   \ifmmode
2420     \o_notation_O:n { #1 }
2421   \else
2422     $ \o_notation_O:n { #1 } $
2423   \fi
2424 }
```

2425

## 2.32 petri.sty

2426 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2427 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]

### 2.32.1 Makro-Kürzel

\let\t=\liPetriTransitionsName

\let\tp=\liPetriTransPfeile

\let\k=\liPetriErreichKnotenDrei

2428 \RequirePackage{tikz}

2429 \usetikzlibrary{petri,arrows.meta}

Für die Darstellungsmatrix

2430 \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

```
2431 \def\liPetriSetzeSchluessel{%
2432   \def\TmpTransitionOne{}%
2433   \def\TmpTransitionTwo{}%
2434   \def\TmpTransitionThree{}%
2435   \def\TmpTransitionFour{}%
2436   \def\TmpTransitionFive{}%
2437   \def\TmpTransitionSix{}%
2438   \def\TmpTransitionSeven{}%
2439   \def\TmpTransitionEight{}%
2440   \def\TmpTransitionNine{}%
2441   \def\TmpTransitionTen{}%
2442   \pgfkeys{/petri/.cd,
2443     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2444     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2445     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2446     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
```

```

2447 p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
2448 p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2449 p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2450 p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2451 p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2452 p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2453 t1/.store in=\TmpTransitionOne,t1/.default=activated,
2454 t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2455 t3/.store in=\TmpTransitionThree,t3/.default=activated,
2456 t4/.store in=\TmpTransitionFour,t4/.default=activated,
2457 t5/.store in=\TmpTransitionFive,t5/.default=activated,
2458 t6/.store in=\TmpTransitionSix,t6/.default=activated,
2459 t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2460 t8/.store in=\TmpTransitionEight,t8/.default=activated,
2461 t9/.store in=\TmpTransitionNine,t9/.default=activated,
2462 t10/.store in=\TmpTransitionTen,t10/.default=activated,
2463 scale/.store in=\TmpScale,scale/.default=0.5,
2464 x/.store in=\TmpX,x/.default=5,
2465 y/.store in=\TmpY,y/.default=5,
2466 }%
2467 }

```

```

2468 \tikzset{
2469   li petri/.style={
2470     activated/.style={
2471       very thick
2472     },
2473     inhibitor/.style={
2474       {Circle[open,length=2mm,fill=white]}-
2475     }
2476   }
2477 }

```

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

```

2478 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2479 \def\liPetriTransitionsName#1{
2480   \ifmmode
2481     \liPetriTransitionsNameOhneMathe{#1}
2482   \else
2483     $\liPetriTransitionsNameOhneMathe{#1}$
2484   \fi
2485 }

```

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

```

2486 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2487   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2488 }

```

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

```

2489 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2490 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2491 }

```

## 2.33 potenzmengen-konstruktion.sty

```
2492 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2493 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2494 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2495 \liLadePakete{formale-sprachen}
2496 \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}
    }
  }
}
```

```
2497 \def\liZustandsMengenSammlung#1#2{
2498   \liZustandsnameGross{#1}
2499   {
2500     \footnotesize
2501     \liPotenzmenge{
2502       \str_case:nn {#1} {#2
2503       }
2504     }
2505 }
```

```
\liZustandsMengenSammlungNr
```

```
2506 \def\liZustandsMengenSammlungNr#1#2{
2507   \liZustandsnameGross{#1}
2508   {
```

```
2509     \footnotesize
2510     \liZustandsmengeNr{
2511         \str_case:nn {#1} #2
2512     }
2513 }
2514 }

2515 \ExplSyntaxOff
2516
```

## 2.34 pseudo.sty

```

2517 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2518 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen
2519 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>
---

```

2520 \RequirePackage[german,boxruled]{algorithm2e}

```

```

2521

```

## 2.35 pumping-lemma.sty

2522 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2523 \ProvidesPackage{lehramt-informatik-pumping-lemma}[2021/08/11 Enthält die  
 2524 Definitionen für das Pumping-Lemma in der Regulären Sprache und  
 2525 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2526 \def\liPumpingRegulaer{%
2527   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2528   alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2529    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2530   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2531
2532   \begin{enumerate}
2533     \item  $|v| \geq 1$ 
2534     (Das Wort  $v$  ist nicht leer.)
2535
2536     \item  $|uv| \leq j$ 
2537     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2538
2539     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w$  in  $L$ 
2540     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2541     Sprache  $L$ )
2542   \end{enumerate}
2543
2544   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2545   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2546 }
```

\liPumpingKontextfrei

```
2547 \def\liPumpingKontextfrei{%
2548   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2549   sich alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  zerlegen lassen in
2550    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2551
2552   \begin{enumerate}
2553     \item  $|vx| \geq 1$ 
2554     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2555
2556     \item  $|vwx| \leq j$ 
2557     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2558
2559     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy$  in  $L$  (Für jede
2560     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2561     Sprache  $L$ )
2562   \end{enumerate}
2563 }
2564
```



## 2.36 quicksort.sty

```

2565 % https://tex.stackexchange.com/a/142634
2566 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2567 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2568
2569 %-----
2570 % USAGE:
2571 % \QSinitialize{comma, separated, numerical, values}
2572 % \loop
2573 % \QSpivotStep
2574 % \ifnum\value{pivotcount}>0
2575 %   \QSSortStep
2576 % \repeat
2577 %-----
2578
2579 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2580 \RequirePackage{xintfrac, xinttools}
2581
2582 \RequirePackage{tikz}
2583
2584 %-----
2585 % FIRST PART: TikZ styles and macros for the actual drawing
2586 \newcounter{cellcount}% used for coordinates of the node
2587 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2588
2589 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2590 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2591 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2592 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2593 % by police of LaTeX good conduct ? )
2594 \tikzset{l/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2595          o/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2596          r/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2597 % this is the "b" style as used in the image below
2598          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2599 % nicer:
2600          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2601          g/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2602
2603 % NOTE the b style was originally the same as the r(aised) style apart from
2604 % not being raised, but I find it nicer with a somewhat different
2605 % specification. I have not updated the images though.
2606
2607 % How the nodes are drawn depending on whether on the left of the pivot value
2608 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2609
2610 \def\DecoLEFT #1{%
2611   \xintFor* ##1 in {#1} \do
2612     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2613 }
2614
2615 \def\DecoINERT #1{%
2616   \xintFor* ##1 in {#1} \do
2617     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2618 }
2619
2620 \def\DecoRIGHT #1{%
2621   \xintFor* ##1 in {#1} \do
2622     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2623 }
2624
2625 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2626   \xintFor* ##1 in {#1} \do

```

```

2627     {\stepcounter{cellcount}}%
2628     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2629 }
2630
2631 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2632     \xintFor* ##1 in {#1} \do
2633     {\stepcounter{cellcount}}%
2634     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2635 }
2636
2637 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2638     \xintFor* ##1 in {#1} \do
2639     {\stepcounter{cellcount}}%
2640     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2641 }
2642
2643 %-----
2644 % SECOND PART: the actual sorting routines.
2645
2646 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2647 \def\QS@sort@b #1{\ifcase #1
2648     \expandafter\QS@sort@empty
2649     \or\expandafter\QS@sort@single
2650     \else\expandafter\QS@sort@c
2651     \fi
2652 }%
2653 \def\QS@sort@empty #1{}
2654 \def\QS@sort@single #1{\QSIr {#1}}
2655
2656 % This step is to pick the last as pivot.
2657 \def\QS@sort@c #1%
2658     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2659
2660 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2661 % The trick with \xintApplyUnbraced is that for example when selecting
2662 % the elements smaller than pivot, if we had been using \xintApply we
2663 % would have had at the minimum an empty brace pair. Thus we use the
2664 % "unbraced" variant, but then the \QS@select@smaller has added in
2665 % anticipation a level of braces.
2666 \def\QS@sort@d #1#2{%
2667     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2668     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2669     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2670 }%
2671 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2672 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2673 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2674
2675 %
2676 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2677 %
2678 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2679 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2680 % latter must handle correctly an empty argument.
2681
2682 %-----
2683 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QSinitialize.
2684
2685 % This draws all with suitable highlighting for the newly chosen pivots
2686 % (which will be shown raised)

```

```

2687 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2688             \let\QSIr\DecoINERT
2689             \let\QSIrr\DecoINERT
2690             \let\QSRr\DecoRIGHTwithPivot
2691 \par\centerline{\rule[1.5mm]{0pt}{8mm}%
2692             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2693             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2694 }
2695
2696 % This sorts and then draws, showing where the pivot chosen in the previous
2697 % step go. Next time they will have become "inert". If pivotcount is still at
2698 % zero on exit from \QSpivotStep, then this is the signal to stop before
2699 % executing \QSortStep.
2700 \def\QSortStep {\def\QSLr {\noexpand\QS@sort@a}%
2701             \def\QSRr {\noexpand\QS@sort@a}%
2702             \def\QSIr {\noexpand\QSIrr}%
2703             \let\QSIrr\relax
2704             \edef\QS@list{\QS@list}%
2705             \let\QSLr\relax
2706             \let\QSRr\relax
2707             \let\QSIr\relax
2708             \edef\QS@list{\QS@list}%
2709             \let\QSLr\DecoLEFT
2710             \let\QSIr\DecoINERTwithPivot
2711             \let\QSIrr\DecoINERT
2712             \let\QSRr\DecoRIGHT
2713 \par\centerline{\rule[1.5mm]{0pt}{8mm}%
2714             \setcounter{cellcount}{0}%
2715             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2716 }
2717
2718 \def\QSinitialize #1{%
2719     % first, we convert the comma separated values into a list of braced items
2720     % we use an \edef, and anyhow many \edef's will be used later
2721     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2722     \let\QSRr\DecoRIGHT
2723     % The \QSRr marker mutated to draw the last element as
2724     % pivot and the earlier ones with the suitable style.
2725     %
2726     % The list of marked braced items \QS@list is used both for drawing
2727     % (as here) and for doing the exchange of elements during sort.
2728     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2729             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2730 }
2731

```

## 2.37 relationale-algebra.sty

2732 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2733 \ProvidesPackage{lehrant-informatik-relationale-algebra}[2020/12/11]

2734 \RequirePackage{amsmath}

2735 \RequirePackage{amssymb}

Zum Zeichnen von Operatorenbäumen verwenden wir TikZ

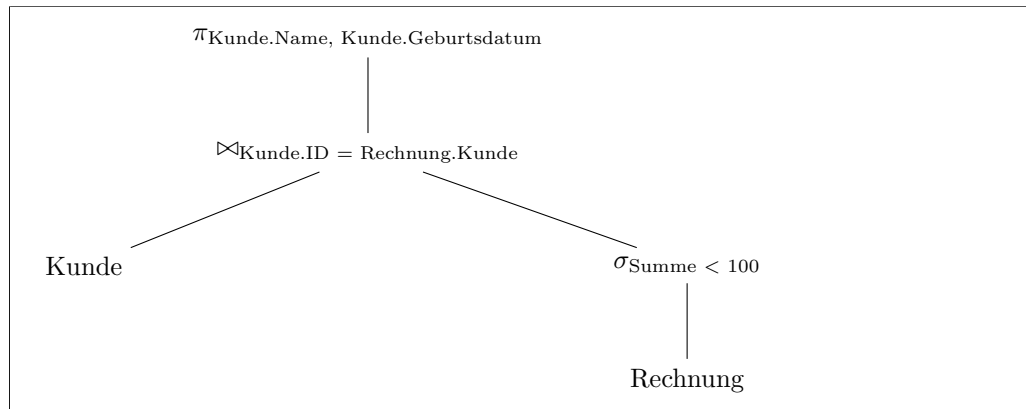
```
\begin{tikzpicture}
  \node
    (pi) {\pi_{\text{Kunde.Name, Kunde.Geburtsdatum}}};

  \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}
```



2736 \RequirePackage{tikz}

2737 \usetikzlibrary{positioning}

Privates Makros, das zwei Querstriche erzeugt.

2738 \def\o@join{\setbox0=\hbox{\bowtie}%

2739 \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%

2740 }

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

2741 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}

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

2742 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}

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

2743 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}

2744

## 2.38 rmodell.sty

```
2745 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2746 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2747 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2748 Datenbanken.]
2749 \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

```
2750 \def\liPrimaer#1{\ul{#1}}
```

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

```
2751 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

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

```
2752 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2753 \ExplSyntaxOn
2754 \NewDocumentEnvironment { liRmodell }
2755 { +b }
2756 {
2757   \medskip
2758   {
2759     \linespread{2}
2760     \setlength{\parindent}{0pt}
2761     \li@Rmodell@Schrift#1
2762   }
2763   \medskip
2764 } {}
2765 \ExplSyntaxOff
```

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

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

```
2766 \def\liRelationMenge#1#2{
2767   \noindent
2768   #1 : \[ #2 ]\}
2769   \par
2770 }
```

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

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

```
2771 \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}
```

```
2772 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2773
```

## 2.39 sortieren.sty

```
2774 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2775 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
2776 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}
}
```



```
2777 \RequirePackage{tikz}
2778 \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.

```
2779 \def\liVertauschen#1{
2780   \directlua{
2781     local sortieren = require('lehramt-informatik-sortieren')
2782     sortieren('#1')
2783   }
2784 }
```

`\liSortierPfeil`

```
2785 \def\liSortierPfeil#1#2{
2786   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2787 }
```

`\liSortierPfeilUnten`

```
2788 \def\liSortierPfeilUnten#1#2{
2789   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2790 }
```

`\liSortierMarkierung`

```
2791 \def\liSortierMarkierung#1#2{\node[
2792   draw,
2793   very thick,
2794   fit=(reihe.#1) (reihe.#2),
2795   inner sep=0pt
2796 ] {}
2797 }

2798 \tikzset{
2799   li sortierung zahlenreihe/.style={
2800     draw,
2801     thin,
2802     font=\large,
2803     rectangle split horizontal,
2804     rectangle split,
2805   }
2806 }
```

```

2807 % https://tex.stackexchange.com/a/140895
2808 \RequirePackage{forest,xstring}
2809 \usetikzlibrary{calc}
2810
2811 \makeatletter
2812 \pgfmathdeclarefunction{strrepeat}{2}{%
2813   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2814   \let\pgfmathresult\pgfutil@empty
2815   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2816     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2817     \advance\pgfmath@count-1\relax
2818   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2819 \makeatother
2820
2821 \def\myNodes{}
2822
2823 \ExplSyntaxOn
2824 \newcommand*\sortList[1]{%
2825   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2826 \ExplSyntaxOff
2827
2828 \forestset{
2829   sort/.code={%
2830     \pgfmathparse{level()>\forestSortLevel}%
2831     \ifnum\pgfmathresult=0
2832       \StrSubstitute{\forestov{content}}{ }{,}{\myList}%
2833       \sortList\myList
2834       \StrSubstitute{\myList}{,}{ }{\myList}%
2835       \pgfmathparse{strrepeat("1",level())}%
2836       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-1!({\forestov{name}}$)
2837         (m\forestov{name}) {\myList}}%
2838       \pgfmathparse{level()=\forestSortLevel}%
2839       \ifnum\pgfmathresult=1
2840         \forestOget{\forestov{@first}}{name}\forestFirst
2841         \forestOget{\forestov{@last}}{name}\forestLast
2842         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2843           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2844       \fi
2845       \ifnum\forestov{@parent}=0\else
2846         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2847       \fi
2848       \gappto\myNodes{;}%
2849     \fi}}
2850
2851 \forestset{sort level/.code=%
2852   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2853   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2854

```

## 2.40 spalten.sty

```
2855 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2856 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2857 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2858 realisiert werden kann.]
2859 \RequirePackage{multicol}
```

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

```
2860 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2861
```



## 2.41 sql.sty

```
2862 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2863 \ProvidesPackage{lehramt-informatik-tabelle}[2021/09/02 Zu Setzen von SQL]
2864 \liLadePakete{syntax}
2865 \RequirePackage{fancyvrb}
2866 \DefineVerbatimEnvironment{liSqlErgebnis}{Verbatim}
2867 {fontsize=\footnotesize}
2868
```

## 2.42 struktogramm.sty

```
2869 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2870 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2871 Paket struktex zum Setzen von Struktogrammen]
2872 \RequirePackage{struktex}
2873
```

## 2.43 syntax.sty

```
2874 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2875 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2876 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2877 \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
```

```
2878 \ExplSyntaxOn
2879 \directlua{
2880   syntax = require('lehramt-informatik-syntax')
2881   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2882   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2883   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2884   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2885   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2886   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2887 }
2888 \RequirePackage{hyperref}
2889 \RequirePackage{minted}
2890 % pygmentize -L styles
2891 \usemintedstyle{colorful}
2892 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2893 %\AfterEndEnvironment{minted}{\end{mdframed}}
2894 %\setminted{breaklines=true,linenos}
2895 \setminted{
2896   breaklines=true,
2897   linenos,
2898   fontsize=\footnotesize,
2899 }
```

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

```
2900 \newminted[liJavaAngabe]{java}{
2901   xleftmargin=1cm,
2902   linenos=false
2903 }
```

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

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

```
2904 \def\liJavaCode#1{
2905   \,
2906   \textcolor{blue}{
2907     \mintinline[
2908       fontsize=\normalsize,
2909       breakanywhere % https://github.com/gpoore/minted/issues/31#issuecomment-458640242
2910     ]{java}|#1|
2911   }
2912   \,
2913 }
```

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

```
2914 \def\liLatexCode#1{\mintinline{latex}|#1|}
```

```

2915 \def\li@GithubLink#1#2{
2916   \begin{flushright}
2917     \tiny
2918     Code-Beispiel~auf~Github~ansehen:~
2919     \href{#1}{\nolinkurl{#2}}
2920   \end{flushright}
2921 }

\liJavaDatei Eine komplette Java-Datei einbinden, die Verzeichnis ./Code/src/main/java/org/bschlangaul
liegt.
2922 \NewDocumentCommand{\liJavaDatei}{ O{firstline=3} m }{
2923   \inputminted[#1]{java}{
2924     \directlua{
2925       syntax.drucke_absoluten_pfad('#2', false)
2926     }
2927   }
2928   \li@GithubLink
2929   {\directlua{syntax.drucke_github_url('#2', false)}}
2930   {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2931 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2932 \NewDocumentCommand{\liJavaTestDatei}{ O{firstline=3} m }{
2933   \inputminted[#1]{java}{
2934     \directlua{
2935       syntax.drucke_absoluten_pfad('#2', true)
2936     }
2937   }
2938   \li@GithubLink
2939   {\directlua{syntax.drucke_github_url('#2', true)}}
2940   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2941 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([
\liJavaExamen$1{$2}{$3}{$4}{$5}
2942 \NewDocumentCommand{\liJavaExamen}{ O{firstline=3} m m m m }{
2943   \inputminted[#1]{java}{
2944     \directlua{
2945       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2946     }
2947   }
2948   \li@GithubLink
2949   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2950   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2951 }

\liAssemblerCode
2953 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2954 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2955   \inputminted{asm}{#1}
2956 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2957 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2958   \inputminted{componentpascal}{#1}
2959 }

```

```

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2960 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2961 \NewDocumentCommand{\liHaskellDatei}{ m }{
2962   \inputminted{haskell}{#1}
2963 }

2964 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2965 \def\liSqlCode#1{\mintinline{sql}|#1|}

2966

```

## 2.44 syntaxbaum.sty

```
2967 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2968 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2969 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
2970 \RequirePackage{tikz-qtrees}
2971
2972 \tikzset{li parsetree/.style={
2973     every internal node/.style={
2974         draw,circle
2975     },
2976     every leaf node/.style={
2977         draw,rectangle
2978     },
2979 }
2980 }
2981
```

## 2.45 synthese-algorithmus.sty

```
2982 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2983 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2984 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2985 Relation in die 3. Normalform]

2986 \liLadePakete{normalformen,mathe,typographie}
2987 \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 Abhängigkeit  $\alpha \rightarrow \beta \in F$  die Linksreduktion durch, überprüfe also für alle  $A \in \alpha$ , ob  $A$  überflüssig ist, d. h. ob  $\beta \subseteq \text{AttrHülle}(F, \alpha - A)$ .

##### (b) Rechtsreduktion

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

##### (c) Löschen leerer Klauseln

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

##### (d) Vereinigung

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

#### 2. Relationsschemata formen

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

#### 3. Schlüssel hinzufügen

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

#### 4. Entfernung überflüssiger Teilschemata

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

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

```
2988 \def\liSyntheseUeberschrift#1{
2989   {
2990     \bfseries
2991     \rmfamily
2992     \str_case:nn {#1} {
2993       {1} {Kanonische-Überdeckung}
2994       {1-1} {Linksreduktion}
2995       {1-2} {Rechtsreduktion}
2996       {1-3} {Löschen-leerer-Klauseln}
2997       {1-4} {Vereinigung}
2998       {2} {Relationsschemata-formen}
2999       {3} {Schlüssel-hinzufügen}
3000       {4} {Entfernung-überflüssiger-Teilschemata}
3001     }
```



```
3002 }
3003 }
```

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

```
3004 \def\li@synthese@erklaerung@texte#1{
3005   \str_case:nn {#1} {
3006     {1} {
3007       Die~kanonische~Überdeckung---also~die~kleinst~mögliche~noch~
3008       äquivalente~Menge~von~funktionalen~Abhängigkeiten~kann~in~vier~
3009       Schritten~erreicht~werden.
3010     }
3011     {1-1} {
3012       Führe~für~jede~funktionale~Anhängigkeit~
3013        $\alpha \rightarrow \beta$ ~in~F~die~Linksreduktion~durch,~
3014       überprüfe~also~für~alle~
3015        $A \in \alpha$ ,~ob~ $A$ ~überflüssig~ist,~d.h.~ob~
3016        $\beta \subseteqq \text{liAttributHuelle}\{F, \alpha\}$ .
3017     }
3018     {1-2} {
3019       Führe~für~jede~(verbliebene)~funktionale~Abhängigkeit~ $\alpha$ ~
3020        $\rightarrow \beta$ ~die~Rechtsreduktion~durch,~überprüfe~also~für~
3021       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~
3022       überflüssig~und~kann~eliminiert~werden,~dh~ $\alpha$ ~
3023        $\rightarrow \beta$ ~wird~durch~ $\alpha \rightarrow (\beta \cup B)$ ~
3024       ersetzt.
3025     }
3026   }
3027   {1-3} {
3028     Entferne~die~funktionalen~Abhängigkeiten~der~Form~ $\alpha$ ~
3029      $\rightarrow \emptyset$ ,~die~im~2.~Schritt~möglicherweise~
3030     entstanden~sind.
3031   }
3032   {1-4} {
3033     Fasse~mittels~der~Vereinigungsregel~funktionale~Abhängigkeiten~
3034     der~Form~ $\alpha \rightarrow \beta_{\{1\}}, \dots, \alpha \rightarrow \beta_{\{n\}}$ ,~so~dass~ $\alpha \rightarrow \beta_{\{1\}} \cup \dots \cup \beta_{\{n\}}$ ~
3035     verbleibt.
3036   }
3037   % Kemper Seite 197
3038   {2} {
3039     Erzeuge~für~jede~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~
3040      $\in F_{\{c\}}$ ~ein~Relationenschema~ $\mathcal{R}_{\{\alpha\}}$ ~
3041      $:= \alpha \cup \beta$ .
3042   }
3043   {3} {
3044     Falls~eines~der~in~Schritt~2.~erzeugten~Schemata~ $\mathcal{R}_{\{\alpha\}}$ ~
3045     einen~Schlüsselkandidaten~von~ $\mathcal{R}_{\{\alpha\}}$ ~bezüglich~ $F_{\{c\}}$ ~
3046     enthält,~sind~wir~fertig,~sonst~wähle~einen~Schlüsselkandidaten~
3047      $\mathcal{K} \subseteqq \mathcal{R}_{\{\alpha\}}$ ~aus~und~definiere~folgendes~
3048     zusätzliche~Schema:~ $\mathcal{R}_{\{\alpha\}} \setminus \mathcal{K} := \mathcal{K}$ ~
3049     und~ $\mathcal{F}_{\{\mathcal{K}\}} := \emptyset$ 
3050   }
3051   {4} {
3052     Eliminiere~diejenigen~Schemata~ $\mathcal{R}_{\{\alpha\}}$ ,~die~in~einem~
3053     anderen~Relationenschema~ $\mathcal{R}_{\{\alpha'\}}$ ~enthalten~sind,~d.h.~
3054      $\mathcal{R}_{\{\alpha\}} \subseteqq \mathcal{R}_{\{\alpha'\}}$ .
3055   }
3056 }
3057 }
3058 }
3059 }
3060 \def\liSyntheseErklaerung#1{
3061   {
3062     \itshape
3063     \footnotesize
```

```

3064     \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
3065   }
3066 }

```

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

```

3067 \def\liSyntheseUeberErklaerung#1{
3068   \liSyntheseUeberschrift{#1}\par
3069   \liSyntheseErklaerung{#1}
3070 }

```

```

3071 \ExplSyntaxOff
3072

```

## 2.46 tabelle.sty

```
3073 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3074 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]
3075 \RequirePackage{tabularx}
3076
```

## 2.47 typographie.sty

```
3077 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3078 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
3079 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
3080 formatierung.sty definiert.]
```

```
3081 \ExplSyntaxOn
```

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

```
3082 \RequirePackage{fontawesome}
```

```
\liErledigt \liErledigt: ☑
```

```
3083 \let\liErledigt=\faCheckSquareO
```

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

```
3084 \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 —

```
3085 \def\liParagraphMitLinien#1{
```

```
3086 \noindent
```

```
3087 \vrule height 2pt depth -1.6pt width 0.4cm
```

```
3088 \enspace
```

```
3089 #1
```

```
3090 \enspace
```

```
3091 \leaders\vrule height 2pt depth -1.6pt \hfill \null
```

```
3092 \par
```

```
3093 \medskip
```

```
3094 }
```

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

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

```
3095 \def\liGeschweifteKlammern#1#2#3#4{
```

```
3096 \par
```

```
3097 \medskip
```

```
3098 \noindent
```

```
3099 #1 \, \$= \Bigl\{ \$
```

```
3100 \vspace{#3}
```

```
3101 #2
```

```
3102 \vspace{#4}
```

```
3103 \begin{flushright} \$\Bigr\} \$\end{flushright}
```

```
3104 \par
```

```
3105 }
```

```
\liTypoUeberschrift
```

```
3106 \def\liTypoUeberschrift#1{
```

```
3107 {
```

```
3108 \bfseries\rmfamily
```

```
3109 #1
```

```
3110 }
```

```
3111 }
```

\liTypoUeberGross

```
3112 \def\liTypoUeberGross#1{
3113   {
3114     \huge
3115     \liTypoUeberschrift{#1}
3116   }
3117 }
```

\liTypoUeberGross

```
3118 \def\liTypoUeberGROSS#1{
3119   {
3120     \Huge
3121     \liTypoUeberschrift{#1}
3122   }
3123 }
```

3124 \ExplSyntaxOff

3125

## 2.48 uml.sty

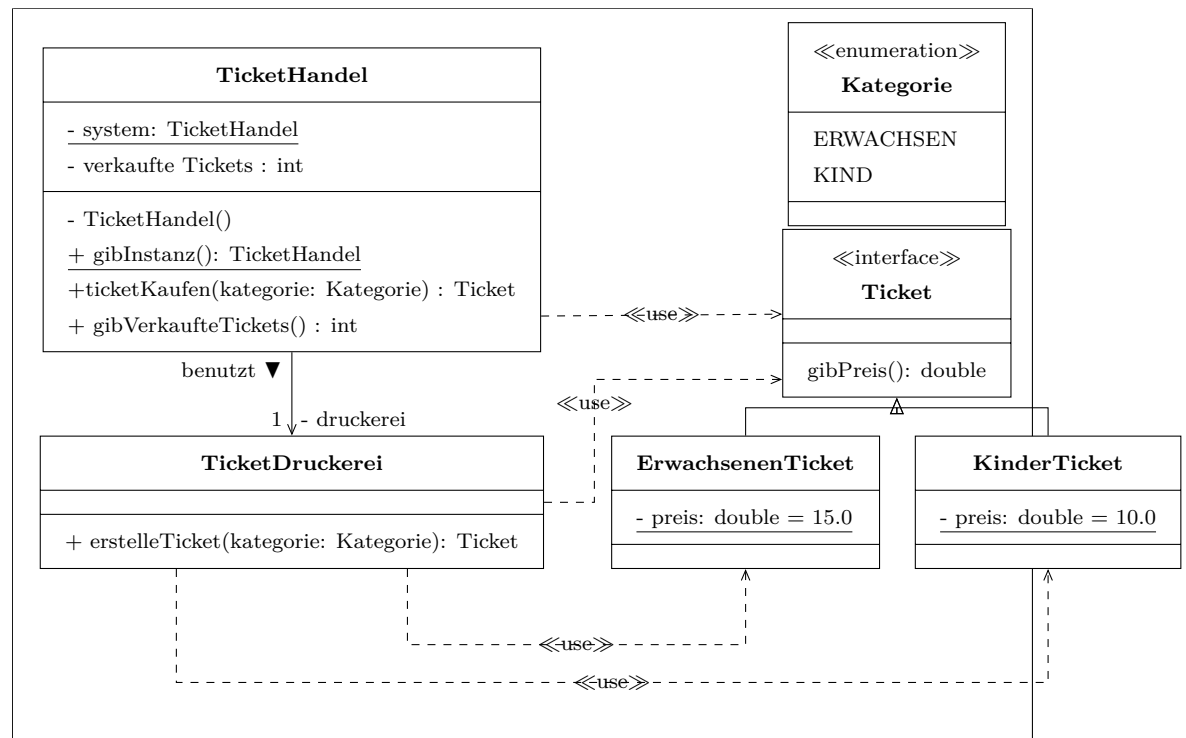
```

3126 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3127 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
3128 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
3129 Erweiterung bereitstellt]

3130 \RequirePackage{tikz-uml}
3131 \RequirePackage{tikz-uml-activity}
3132 % Not compatible with wasysym
3133 %\RequirePackage{mathabx}
3134 \RequirePackage{wasysym}
3135 \usetikzlibrary{positioning}

3136 \tikzumlset{
3137   fill class=white!0,
3138   font=\footnotesize,
3139   fill object=white!0,
3140   fill note=white!0,
3141   fill state=white!0,
3142   % Use case
3143   fill usecase=white!0,
3144   fill system=white!0,
3145 }

```



```

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

```

```

3146 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
3147   \def\@liDirLeft{}
3148   \def\@liDirRight{ \RIGHTarrow}
3149   \pgfkeys{/lese/dir/.is choice}
3150   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
3151   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
3152   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{\LEFTarrow }}}
3153   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
3154
3155   \def\@liPos{above}
3156   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
3157

```

```

3158 \def\@liDistance{0cm}
3159 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
3160
3161 \pgfkeys{/lese/.cd,#1}
3162
3163 \node[\@liPos = \@liDistance of #2-middle] {
3164   \@liDirLeft{\footnotesize#2}\@liDirRight
3165 };
3166 }
3167

```

## 2.49 vollstaendige-induktion.sty

```
3168 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3169 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01]
3170 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
3171 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

```
3172 \RequirePackage{lehramt-informatik-typographie}
3173 \RequirePackage{lehramt-informatik-mathe}
3174 \RequirePackage{lehramt-informatik-syntax}
3175 \ExplSyntaxOn
```

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

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

```
3176 \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

```
3177 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
3178 \def\liInduktionAnfang{
3179   \liPseudoUeberschrift{Induktionsanfang}
3180
3181   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
3182   \liParagraphMitLinien{
3183     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
3184   }
3185 }
```

\liInduktionVoraussetzung

```
3186 \def\liInduktionVoraussetzung{
3187   \liPseudoUeberschrift{Induktionsvoraussetzung}
3188
3189   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
3190   \liParagraphMitLinien{
3191     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ $.
3192   }
3193 }
```

\liInduktionSchritt

```
3194 \def\liInduktionSchritt{
3195   \liPseudoUeberschrift{Induktionsschritt}
3196
3197   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
3198   \liParagraphMitLinien{
3199     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
3200     auch~ $A(n=k+1)$ ~wahr~sein~muss.
3201   }
3202 }

3203 \ExplSyntaxOff
3204
```

## 2.50 wasserfall.sty

```
3205 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3206 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
3207 \RequirePackage{tikz}
3208 \tikzset{wasserfall/.style={
3209   >=stealth,
3210   node distance = 2mm and -8mm,
3211   start chain = A going below right,
3212   every node/.style = {
3213     draw,
3214     text width=24mm,
3215     minimum height=12mm,
3216     align=center,
3217     inner sep=1mm,
3218     fill=white,
3219     drop shadow={fill=black},
3220     on chain=A
3221   },
3222 }}
3223 \usetikzlibrary{chains,positioning,shadows}
3224
```

## 2.51 wpkalkuel.sty

```
3225 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3226 \ProvidesPackage{lehramt-informatik-wpkalkuel}[2020/06/13]
```

### 2.51.1 Makro-Kürzel

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

```
3227 \RequirePackage{amsmath}
```

```
3228 \ExplSyntaxOn
```

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

```
3229 \def\liWpKalkuelOhneMathe#1#2{
3230   \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)
3231 }
3232 \def\liWpKalkuel#1#2{
3233   \ifmmode
3234     \liWpKalkuelOhneMathe{#1}{#2}
3235   \else
3236     $\liWpKalkuelOhneMathe{#1}{#2}$
3237   \fi
3238 }
```

**\MatheEnv**

```
3239 \def\MatheEnv#1{
3240   \medskip
3241
3242   \hspace{1em}#1
3243
3244   \medskip
3245 }
```

**\Mathe**

```
3246 \def\Mathe#1{
3247   \MatheEnv{${#1}$}
3248 }
```

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

```
3249 \def\liWpEquivalent#1{
3250   \MatheEnv{${\equiv}\hspace{1em}${#1}$}
3251 }
```

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

```
3252 \newlength{\@Skip@Erklaerung@Reset}
3253 \def\liWpErklaerung#1{
3254   \setlength{\@Skip@Erklaerung@Reset}{\leftskip}
3255   \setlength{\leftskip}{0.5cm}
3256
3257   \par
3258   \noindent
3259   {
3260     \scriptsize
3261     #1
3262   }
3263   \par
3264
3265   \setlength{\leftskip}{\@Skip@Erklaerung@Reset}
3266 }
```

\liWpErklaerungVerzweigung

```

3267 \def\liWpErklaerungVerzweigung{
3268   $\liWpKalkuelOhneMathe{if~\{-b~\}~then~\{-a1~\}~else~\{-a2~\}}{Q}
3269   \equiv
3270   (b \land \liWpKalkuelOhneMathe{a1}{Q})
3271   \lor
3272   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
3273 }

3274 \ExplSyntaxOff

3275

```

### 3 Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

Symbols			
\#	..... 114	\sqcup	..... 2355, 2365
\,	.. 358, 417, 472, 699, 717, 1349, 1351, 2063, 2064, 2065, 2403, 2905, 2912, 3099		
\@Skip@Erklaerung@Reset	... 3252, 3254, 3265		
\@afterheading	..... ... 1587, 1945, 2032		
\@afterindentfalse	.. ... 1586, 1944, 2031		
\@liDirLeft	3147, 3152, 3164		
\@liDirRight	3148, 3150, 3151, 3152, 3153, 3164		
\@liDistance	..... ... 3158, 3159, 3163		
\@liPos	.. 3155, 3156, 3163		
\\	..... 729, 757, 758, 761, 762, 765, 766, 861, 862, 863, 970, 1004, 1006, 1036, 1045, 1090, 1132, 1133, 1134, 1139, 1140, 1141, 1161, 1932, 2298, 2356, 2359		
\{	..... 212, 1286, 1296, 1308, 1309, 1314, 1348, 1771, 2343, 2768, 3099, 3268		
\}	..... 212, 1286, 1296, 1308, 1309, 1316, 1352, 1772, 2343, 2768, 3103, 3268		
\_	..... 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, 1566, 1571, 1606, 1657		
		\addbibresource	..... 1901, 1902, 1903, 1904, 1905, 1906, 1907, 1908, 1909, 1910
		\AddToHook	.... 1589, 1661
		\advance	..... 2817
		\AfterEndEnvironment	2893
		\Alph	..... 1409
		\alph	..... 1409, 1410
		\alpha	3013, 3015, 3016, 3019, 3021, 3022, 3023, 3024, 3025, 3029, 3035, 3036, 3041, 3042, 3043, 3046, 3054, 3055, 3056
		\arabic	1409, 2612, 2617, 2622, 2628, 2634, 2640
		\arraystretch	..... 2293
		B	
		\BeforeBeginEnvironment	..... 2892
		\begin	755, 809, 824, 859, 883, 930, 962, 977, 1002, 1012, 1032, 1052, 1084, 1101, 1130, 1155, 1176, 1200, 1215, 1329, 1415, 1506, 1663, 1725, 1734, 1741, 1849, 1924, 1982, 1987, 1995, 2020, 2025, 2036, 2046, 2050, 2133, 2137, 2156, 2179, 2202, 2217, 2296, 2297, 2347, 2392, 2532, 2552, 2693, 2715, 2729, 2892, 2916, 3103
		\begingroup	1925, 2345, 2813
		\beschriftung	..... ..... 1962, 1966, 1970, 1974, 1978, 1980
		\beta	..... 3013, 3016, 3020, 3021, 3022, 3025, 3035, 3036, 3037, 3042, 3043
		\bf	..... 2590, 2591, 2592
		\bfseries	..... .. 558, 1399, 1401, 1571, 1623, 1681, 2590, 2596, 2598, 2600, 2601, 2990, 3108
		\Bigl	..... 3099
		\Bigr	..... 3103
		\bigskip	450, 731, 736, 1582, 1674, 1934, 2271
		\bool	..... 336, 418
		\bowtie	..... 2738, 2741, 2742, 2743
		\Box	..... 164
		\boxtimes	..... 548
		C	
		\c	..... 1508, 1509
		\cdot	.... 2108, 2164, 2175
		\centerline	..... 1730, 2691, 2713, 2728
		\cftbeforesecskip	.. 1704
		\cftbeforesubsecskip	..... 1705, 1706
		\cftbeforesubsubsecskip	..... 1707
		\cftsubsecafterpnum	1706
		\chapter	..... 1399, 1400
		\char	..... 1932
		\clearpage	1624, 1712, 1926
		\cline	..... 729
		\clist	..... 231, 279, 280, 302, 306, 2825
		\columnbreak	..... 2860
		\contentsname	..... 1710
		\cs	..... 305, 333, 357, 360, 370, 393,

405, 416, 417, 454, 461, 466, 1542, 1549, 1555, 2014, 2415	1417, 1522, 1700, 1737, 1738, 1745, 1854, 1929, 1984, 1992, 2011, 2027, 2028, 2042, 2053, 2054, 2135, 2151, 2176, 2199, 2214, 2224, 2300, 2301, 2349, 2394, 2542, 2562, 2693, 2715, 2729, 2893, 2920, 3103	<b>F</b> <code>\faCheckSquare0</code> . . . . 3083 <code>\faCircleThin</code> . . . . . 1268 <code>\faGg</code> . . . . . 1262 <code>\fancyfoot</code> . . . . . 1421, 1422, 1423, 1878, 1879, 1880, 1881 <code>\fancyhead</code> . . . . . 1420, 1875, 1876, 1877 <code>\faSquare0</code> . . . . . 1256 <code>\fi</code> . 655, 663, 671, 679, 686, 1292, 1306, 1342, 1981, 2083, 2093, 2103, 2117, 2341, 2378, 2423, 2484, 2651, 2843, 2844, 2847, 2849, 3237 <code>\filcenter</code> . . . . . 1623 <code>\footcite</code> 752, 821, 843, 890, 913, 945, 998, 1049, 1076, 1122, 1750, 1753, 1760, 1765, 1770, 1774, 1780, 1785, 1890, 2152, 2153, 2308, 2545 <code>\footnote</code> . . . . . 2057, 2061 <code>\footnotesize</code> . . . . . 152, 427, 509, 603, 1273, 1578, 1676, 1955, 2006, 2021, 2037, 2252, 2346, 2363, 2371, 2500, 2509, 2752, 2867, 2898, 3063, 3138, 3164 <code>\footrulewidth</code> . 1426, 1883 <code>\foreach</code> . 1508, 1511, 1518 <code>\forestFirst</code> . . 2840, 2843 <code>\forestLast</code> . . . 2841, 2843 <code>\forestOget</code> . . . 2840, 2841 <code>\forestOnes</code> . . . . . 2853 <code>\forestOv</code> 2842, 2843, 2846 <code>\forestov</code> . 2832, 2836, 2837, 2840, 2841, 2842, 2843, 2845, 2846 <code>\forestset</code> . . . . 2828, 2851 <code>\forestSortLevel</code> . . . . 2830, 2838, 2852, 2853 <code>\frac</code> 2110, 2143, 2175, 2190 <code>\fullouterjoin</code> . . . . . 2743
<b>D</b> <code>\DeclareMathSymbol</code> . . . . . . . 2241, 2242 <code>\DecoINERT</code> . . . . . 2615, 2688, 2689, 2711 <code>\DecoINERTwithPivot</code> . . . . . . 2631, 2710 <code>\DecoLEFT</code> . . . . . 2610, 2709 <code>\DecoLEFTwithPivot</code> . . . . . . . 2625, 2687 <code>\DecoRIGHT</code> 2620, 2712, 2722 <code>\DecoRIGHTwithPivot</code> . . . . . . 2637, 2690 <code>\definecolor</code> . . . . . 1397 <code>\DefineVerbatimEnvironment</code> . . . . . 2866 <code>\delta</code> 70, 112, 170, 212, 1300 <code>\dh</code> . . . . . 2065, 3024 <code>\directlua</code> . . . . . 63, 142, 200, 205, 1281, 1295, 1315, 1323, 1330, 1335, 2328, 2333, 2386, 2393, 2400, 2780, 2879, 2924, 2929, 2930, 2934, 2939, 2940, 2944, 2950, 2951 <code>\do</code> . . . . . 2611, 2616, 2621, 2626, 2632, 2638 <code>\dots</code> . . . . . 589, 593, 1771, 2539, 3035, 3036 <code>\DOWNarrow</code> . . . . . 3151 <code>\draw</code> . . . . . 1513, 1516, 1519, 2487, 2786, 2789	<b>E</b> <code>\edef</code> . . . . . 1504, 2704, 2708, 2720, 2721 <code>\edge</code> . . . . . 276 <code>\else</code> . . . . . 653, 661, 669, 677, 684, 1290, 1304, 1340, 1978, 2081, 2091, 2101, 2115, 2339, 2376, 2421, 2482, 2650, 2843, 2845, 3235 <code>\emph</code> . . 750, 997, 1252, 1752, 1781, 1783, 1931 <code>\empty</code> . . . . . 1978, 2375 <code>\emptyset</code> . . . . . 2285, 3030, 3051, 3084 <code>\end</code> 794, 820, 845, 880, 914, 946, 974, 990, 1009, 1017, 1048, 1077, 1098, 1123, 1148, 1168, 1187, 1212, 1230, 1331, 1493, 1496 <code>\endcsname</code> . . . . 1493, 1496 <code>\endgroup</code> 1928, 2350, 2818 <code>\enspace</code> . . . . . 3088, 3090 environments: <code>liAdditum</code> . . . . . 1985 <code>liAHuelle</code> . . . . . 2344 <code>liAntwort</code> . . . . . 1958 <code>liDiagramm</code> . . . . . 2044 <code>liEinbettung</code> . . . . . 1957 <code>liExkurs</code> . . . . . 1993 <code>liGraphenFormat</code> . 1485 <code>liJavaAngabe</code> . . . . 2900 <code>liKasten</code> . . . . . 1414 <code>liKontrollflussgraph</code> . . . . . 1848 <code>liLernkartei</code> . . . . 2034 <code>liProduktionsRegeln</code> . . . . . 1324 <code>liProjektSprache</code> 1956 <code>liQuellen</code> . . . . . 2014 <code>liRelationenSchemaFormat</code> . . . . . 2772 <code>liRmodell</code> . . . . . 2752 <code>liUebergangsTabelle</code> . . . . . 2293 <code>\equiv</code> . . . . . 3250, 3269 <code>\verzeuge@tiefgestellt</code> . . . 1295, 1296, 1300 <code>\expandafter</code> . . . . . 1493, 2646, 2648, 2649, 2650, 2658, 2816 <code>\ExplSyntaxOff</code> . . . . . . . . . . 54, 97, 139, 144, 197, 202, 207, 611, 633, 648, 724, 1276, 1354, 1389, 1456, 1608, 1714, 1869, 1885, 2066, 2131, 2321, 2515, 2765, 2826, 2964, 3071, 3124, 3203, 3274 <code>\ExplSyntaxOn</code> . . 35, 66, 107, 140, 165, 198, 203, 228, 554, 617, 634, 673, 1249, 1344, 1366, 1440, 1541, 1629, 1862, 1874, 1922, 2070, 2303, 2408, 2496, 2753, 2823, 2878, 2987, 3081, 3175, 3228	<b>G</b> <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, 434, 440, 441, 443,

444, 446, 447, 459, 462, 467, 469, 473, 1543, 1545, 1556, 1558, 1631, 1635, 1639, 1643, 1644, 1645, 1646, 1647, 1649, 1650, 1652, 1683, 1685, 1690, 1694	1226, 1735, 1736, 2014, 2018, 2138, 2143, 2147, 2157, 2163, 2168, 2180, 2184, 2188, 2192, 2196, 2203, 2207, 2211, 2533, 2536, 2539, 2553, 2556, 2559	\LehramtInformatikAutorName ..... 1880 \LehramtInformatikGitBranch ..... 458, 2886 \LehramtInformatikGithubCodeRepo ..... 2885 \LehramtInformatikGithubDomain ..... 455, 2882 \LehramtInformatikGithubRawDomain ..... 2883 \LehramtInformatikGithubTexRepo ..... 456, 2884 \LehramtInformatikRepository 17, 20, 23, 26, 29, 1444, 1641, 1901, 1902, 1903, 1904, 1905, 1906, 1907, 1908, 1909, 1910, 2881 \LehramtInformatikTitel ..... 1876 \leq .... 2175, 2536, 2556 \let ..... 1298, 1299, 1926, 2687, 2688, 2689, 2690, 2703, 2705, 2706, 2707, 2709, 2710, 2711, 2712, 2722, 2814, 2852, 2853, 3083 \li@chomsky@erklaerung@texte ..... 568, 604 \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 \li@EntwurfsCodeAllgemein ..... 745 \li@fussnote@text 2249, 2255, 2259, 2263, 2267 \li@GithubLink ..... 2915, 2928, 2938, 2949 \li@mget . 1495, 1499, 1519 \li@minc ..... 1498, 1520 \li@mset ..... 1492, 1500, 1509, 1512 \li@numdiscs ..... ... 1504, 1513, 1519 \li@Rmodell@Schrift . ... 2752, 2761, 2771 \li@sequence .. 1505, 1518 \li@synthese@erklaerung@texte ..... 3004, 3064 \liAbleitung ..... 1323 liAdditum (environment) ..... 1985 liAHuelle (environment) ..... 2344 \liAlphabet ..... 1308 \LehramtInformatikAutorEmailIliAntwort (environment) ..... 1881 ..... 1958
\Gamma . 111, 169, 212, 1309 \gappto ..... 2848 \geometry ..... 5 \geq ..... 1778, 2141, 2528, 2533, 2549, 2553	\itshape ..... 602, 3062  <b>J</b> \j 1508, 1509, 1511, 1512, 1513, 1518, 1519, 1520	
<b>H</b> \hbox ..... 2738 \headrulewidth . 1425, 1882 \headwidth ..... 1884 \hfill .... 1572, 1605, 2160, 2166, 2171, 3091 \hinweis ..... 1955 \hline ..... 2298 \horizontale .. 1549, 1580 \href ..... 1362, 1600, 2061, 2228, 2919 \hspace .. 2490, 3242, 3250 \ht ..... 2739 \Huge ..... 1681, 3120 \huge ..... 1623, 3114	<b>K</b> \k ..... 1518 \keys ..... 41, 75, 87, 117, 127, 175, 185, 310, 621, 625, 639, 644, 1373, 1380	
<b>I</b> \i ..... 1518, 1519 \ifcase ..... 2647 \ifmmode 651, 659, 667, 675, 682, 1288, 1302, 1338, 2079, 2089, 2099, 2113, 2337, 2419, 2480, 3233 \IfNoValueTF ..... ... 1988, 2057, 2061 \ifnum ..... 2574, 2815, 2831, 2839, 2845 \ifx .... 1978, 2375, 2843 \in ..... 577, 733, 1778, 2123, 2126, 2129, 2158, 2164, 2169, 2528, 2539, 2549, 2559, 3013, 3015, 3021, 3042, 3191 \includegraphics .... ..... 1447, 1452 \inhaltsverzeichnis 1923 \input ..... 17, 20, 23, 26, 29, 479, 1640 \inputminted 2923, 2933, 2943, 2955, 2958, 2962 \int ..... 2825 \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,	<b>L</b> \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, 1865, 1866, 1867, 2017, 2018, 2019, 2026 \labelenumi ..... 1410 \labelenumii ..... 1411 \labelitemi ..... 1404 \labelitemii ..... 1405 \labelitemiii ..... 1406 \labelitemiv ..... 1407 \land ..... 3270, 3272 \LARGE ..... 1399, 1690 \Large ..... 1571 \large 1566, 1664, 1730, 2802 \leaders ..... 3091 \left ..... 2073, 2413 \LEFTarrow ..... 3152 \leftarrow ..... 665 \leftouterjoin ..... 2741 \leftskip 3254, 3255, 3265 \LehramtInformatikAutorEmailIliAntwort (environment) ..... 1881 ..... 1958	

\liAnweisung	1856	\liEntwurfsAbstrakteFabrikUml	754, 804	\liEntwurfsStellvertreter	1195
\liAssemblerCode	2953	\liEntwurfsAdapter	853	\liEntwurfsStellvertreterCode	1189, 1197
\liAssemblerDatei	2954	\liEntwurfsAdapterAkteure	823, 855	\liEntwurfsStellvertreterUml	1175, 1196
\liAttribut	2771	\liEntwurfsAdapterCode	847, 856	\liEntwurfsZustand	1236
\liAttributHuelle	2335, 3016, 3021	\liEntwurfsAdapterUml	808, 854	\liEntwurfsZustandAkteure	1214, 1238
\liAttributHuelleOhneMathe	2335, 2338, 2340, 2354, 2364, 2372	\liEntwurfsBeobachter	924	\liEntwurfsZustandCode	1232, 1239
\liAttributMenge	2343, 2355, 2358, 2365, 2366, 2380, 2382	\liEntwurfsBeobachterAkteure	882, 926	\liEntwurfsZustandUml	1199, 1237
\liAufgabe	16	\liEntwurfsBeobachterCode	916, 927	\liEpsilon	1294
\liAufgabenMetadaten	47, 1562, 1655	\liEntwurfsBeobachterUml	858, 925	\liErAttribute	1252, 1266, 1268
\liAufgabenTitel	55	\liEntwurfsDekorierer	956	\liErDatenbankName	1271
\liAusdruck	1345	\liEntwurfsDekoriererAkteure	958	\liErEntity	1250, 1254, 1256
\liAutomat	66	\liEntwurfsDekoriererCode	948, 959	\liErledigt	3083
\liAutomatenKante	98	\liEntwurfsDekoriererUml	929, 957	\liErMpAttribute	1265
\liBandAlphabet	1309	\liEntwurfsEinfacheFabrik	992	\liErMpEntity	1253
\liBedingung	1857	\liEntwurfsEinfacheFabrikAkteure	976, 994	\liErMpRelationship	1259
\liBedingungDrei	2128, 2171, 2211	\liEntwurfsEinfacheFabrikUml	961, 993	\liErRelationship	1251, 1260, 1262
\liBedingungEins	2122, 2160, 2203	\liEntwurfsEinzelstueck	1022	\liExamensAufgabe	19
\liBedingungFalsch	1859	\liEntwurfsEinzelstueckAkteure	1011, 1027	\liExamensAufgabeA	28
\liBedingungWahr	1858	\liEntwurfsEinzelstueckBeschreibung	996, 1023	\liExamensAufgabeTA	25
\liBedingungZwei	2125, 2166, 2207	\liEntwurfsEinzelstueckCode	1019, 1029	\liExamensAufgabeTTA	22
\liBeschriftung	1947	\liEntwurfsEinzelstueckUml	1001, 1025	\liExkurs (environment)	1993
\liBindeAufgabeEin	1638	\liEntwurfsErbauer	1079	\liFalsch	549
\liChomskyErklaerung	568, 609	\liEntwurfsErbauerAkteure	1051, 1081	\liFlaci	1355
\liChomskyUeberErklaerung	607	\liEntwurfsErbauerUml	1031, 1080	\liFremd	2751
\liChomskyUeberschrift	556, 608	\liEntwurfsFabrikmethode	1125, 1150	\liFunktionaleAbhaengigkeit	2374, 2377, 2385
\liCpmEreignis	617	\liEntwurfsFabrikmethodeAkteure	1100, 1127	\liFunktionaleAbhaengigkeiten	2388
\liCpmFruehErklaerung	688	\liEntwurfsFabrikmethodeUml	1083, 1126	\liFussnote	2248, 2250
\liCpmFruehI	681, 701	\liEntwurfsKompositum	1150	\liFussnoteDreiText	2262, 2280
\liCpmSpaetErklaerung	706	\liEntwurfsKompositumAkteure	1152	\liFussnoteEinsText	2254, 2274
\liCpmSpaetI	674, 719	\liEntwurfsKompositumUml	1129, 1151	\liFussnoteLink	2060
\liCpmVon	657	\liEntwurfsModellPraesentation	1171	\liFussnoten	2270
\liCpmVonOhneMathe	657, 660, 662	\liEntwurfsModellPraesentationAkteure	1173	\liFussnoteUrl	1169, 2056
\liCpmVonZu	649	\liEntwurfsModellPraesentationUml	1154, 1172	\liFussnoteVierText	2266, 2283
\liCpmVonZuOhneMathe	649, 652, 654	\liEntwurfsModellPraesentationStapelFormat (environment)	1485	\liFussnoteZweiText	2258, 2277
\liCpmVorgang	634	\liEntwurfsModellPraesentationStapelFormatAkteure	1492	\liGeschweifteKlammern	1327, 2389, 3095
\liCpmZu	665	\liEntwurfsModellPraesentationStapelFormatUml	1443, 1448, 1453	\liGrafikCCLizenz	1446, 1593
\liCpmZuOhneMathe	665, 668, 670	\liEntwurfsModellPraesentationStapelFormatUmlAkteure	1451, 1592, 1673	\liGrafikLogo	1451, 1592, 1673
liDiagramm (environment)	2044	\liEntwurfsModellPraesentationStapelFormatUmlCode	1443, 1448, 1453	\liGrafikLogoPfad	1443, 1448, 1453
liEinbettung (environment)	1957	\liEntwurfsModellPraesentationStapelFormatUmlCodeAkteure	1443, 1448, 1453	\liGrammatik	1366
\liEntwurfs	1171	\liEntwurfsModellPraesentationStapelFormatUmlCodeUml	1443, 1448, 1453	\liHaskellCode	2960
\liEntwurfsAbstrakteFabrik	801	\liEntwurfsModellPraesentationStapelFormatUmlCodeUmlAkteure	1443, 1448, 1453	\liHaskellCodeUml	2961
\liEntwurfsAbstrakteFabrikBeschreibung	749, 802	\liEntwurfsModellPraesentationStapelFormatUmlCodeUmlAkteureCode	1443, 1448, 1453	\liInduktionAnfang	3178
\liEntwurfsAbstrakteFabrikCode	796, 806				

<code>\liInduktionErklaerung</code>	<code>\liOmega</code> . . . . .	<code>liRmodell (environment)</code>
. . . . . <a href="#">3177</a>	<code>\liOmegaOhneMathe</code> . . . . .	. . . . . <a href="#">2752</a>
<code>\liInduktionMarkierung</code>	. . . <a href="#">2085</a> , <a href="#">2090</a> , <a href="#">2092</a>	<code>\liRundeKlammer</code> . . . . .
. . . . . <a href="#">3176</a>	<code>\liONotation0</code> . . . . . <a href="#">2415</a>	. . . . . <a href="#">2072</a> , <a href="#">2076</a> , <a href="#">2086</a> ,
<code>\liInduktionSchritt</code> <a href="#">3194</a>	<code>\liOOhneMathe</code> . . . . .	. . . . . <a href="#">2096</a> , <a href="#">2110</a> , <a href="#">2412</a> , <a href="#">2416</a>
<code>\liInduktionVoraussetzung</code>	. . . <a href="#">2095</a> , <a href="#">2100</a> , <a href="#">2102</a>	<code>\liSetzeExamenTeilaufgabeNr</code>
. . . . . <a href="#">3186</a>	<code>\liParagraphMitLinien</code>	. . . . . <a href="#">1634</a>
<code>liJavaAngabe (environ-</code>	. . . . . <a href="#">604</a> , <a href="#">689</a> ,	<code>\liSetzeExamenThemaNr</code>
<code>ment)</code> . . . . . <a href="#">2900</a>	. . . . . <a href="#">707</a> , <a href="#">2309</a> , <a href="#">3064</a> ,	. . . . . <a href="#">1630</a>
<code>\liJavaCode</code> . . . . . <a href="#">2904</a>	. . . . . <a href="#">3085</a> , <a href="#">3182</a> , <a href="#">3190</a> , <a href="#">3198</a>	<code>\liSortierMarkierung</code> <a href="#">2791</a>
<code>\liJavaDatei</code> . . <a href="#">747</a> , <a href="#">2922</a>	<code>\liPetriErreichKnotenDrei</code>	<code>\liSortierPfeil</code> . . . <a href="#">2785</a>
<code>\liJavaExamen</code> . . . . . <a href="#">2942</a>	. . . . . <a href="#">2489</a>	<code>\liSortierPfeilUnten</code> <a href="#">2788</a>
<code>\liJavaTestDatei</code> . . <a href="#">2932</a>	<code>\liPetriErreichTransition</code>	<code>\liSpaltenUmbruch</code> . . <a href="#">2860</a>
<code>liKasten (environment)</code> <a href="#">1414</a>	. . . . . <a href="#">2486</a>	<code>\liSqlCode</code> . . . . . <a href="#">2965</a>
<code>\liKellerAutomat</code> . . . . <a href="#">107</a>	<code>\liPetriSetzeSchluessel</code>	<code>\listen@punkt</code> . . <a href="#">2014</a> , <a href="#">2026</a>
<code>\liKellerKante</code> . . . . . <a href="#">145</a>	. . . . . <a href="#">2431</a>	<code>\liStrich</code> . . . . . <a href="#">1722</a>
<code>\liKellerUebergang</code> . .	<code>\liPetriTransitionsName</code>	<code>\liSyntheseErklaerung</code>
. . . . . <a href="#">140</a> , <a href="#">146</a>	. . . . . <a href="#">2478</a> , <a href="#">2490</a>	. . . . . <a href="#">3004</a> , <a href="#">3069</a>
<code>\liKontrollCode</code> . . . . <a href="#">1860</a>	<code>\liPetriTransitionsNameOhneMathe</code>	<code>\liSyntheseUeberErklaerung</code>
<code>liKontrollflussgraph</code>	. . . <a href="#">2478</a> , <a href="#">2481</a> , <a href="#">2483</a>	. . . . . <a href="#">3067</a>
(environment) <a href="#">1848</a>	<code>\liPetriTransPfeile</code> <a href="#">2490</a>	<code>\liSyntheseUeberschrift</code>
<code>\liKontrollKnotenPfad</code>	<code>\liPolynomiellReduzierbar</code>	. . . . . <a href="#">2988</a> , <a href="#">3068</a>
. . . . . <a href="#">1862</a>	. . . . . <a href="#">1740</a>	<code>\liT</code> <a href="#">2105</a> , <a href="#">2120</a> , <a href="#">2134</a> , <a href="#">2198</a>
<code>\liKontrollTextzeileKnoten</code>	<code>\liPotenzmenge</code> . . . . .	<code>\liTeilen</code> . . . . . <a href="#">2332</a>
. . . . . <a href="#">1861</a> , <a href="#">1866</a>	. . . <a href="#">1295</a> , <a href="#">1299</a> , <a href="#">2501</a>	<code>\literatur</code> . . . . <a href="#">1889</a> , <a href="#">1913</a>
<code>\liKurzeTabellenLinie</code> <a href="#">729</a>	<code>\liPotenzmengeOhneMathe</code>	<code>\liTheta</code> . . . . . <a href="#">2075</a> ,
<code>\liLadeAllePakete</code> . . .	. . . <a href="#">1296</a> , <a href="#">1297</a> , <a href="#">1298</a>	. . . <a href="#">2126</a> , <a href="#">2158</a> , <a href="#">2164</a> , <a href="#">2169</a>
. . . . . <a href="#">233</a> , <a href="#">1626</a>	<code>\liPrimaer</code> . . . . . <a href="#">2750</a>	<code>\liThetaOhneMathe</code> . . .
<code>\liLadePakete</code> . . . . <a href="#">59</a> ,	<code>\liProblemBeschreibung</code>	. . . <a href="#">2075</a> , <a href="#">2080</a> , <a href="#">2082</a>
. . . <a href="#">62</a> , <a href="#">229</a> , <a href="#">234</a> , <a href="#">555</a> ,	. . . . . <a href="#">1724</a>	<code>\liTOhneMathe</code> . . . . .
. . . <a href="#">616</a> , <a href="#">1285</a> , <a href="#">1528</a> ,	<code>\liProblemClique</code> . . <a href="#">1747</a>	. . . <a href="#">2105</a> , <a href="#">2114</a> , <a href="#">2116</a>
. . . <a href="#">1613</a> , <a href="#">1720</a> , <a href="#">2247</a> ,	<code>\liProblemName</code> . . . . .	<code>\liTuringKante</code> . . . . . <a href="#">208</a>
. . . <a href="#">2327</a> , <a href="#">2495</a> , <a href="#">2864</a> , <a href="#">2986</a>	. . . . . <a href="#">1723</a> , <a href="#">1730</a> ,	<code>\liTuringLeerzeichen</code>
<code>\liLatexCode</code> . . . . . <a href="#">2914</a>	. . . <a href="#">1742</a> , <a href="#">1744</a> , <a href="#">1757</a> ,	. . . . . <a href="#">164</a> , <a href="#">172</a>
<code>\liLeereZelle</code> . . . . . <a href="#">2285</a>	. . . <a href="#">1768</a> , <a href="#">1769</a> , <a href="#">1777</a> , <a href="#">1778</a>	<code>\liTuringMaschine</code> . . <a href="#">165</a>
<code>liLernkartei (environ-</code>	<code>\liProblemSat</code> . . . . . <a href="#">1776</a>	<code>\liTuringUeberfuehrung</code>
<code>ment)</code> . . . . . <a href="#">2034</a>	<code>\liProblemSubsetSum</code> .	. . . . . <a href="#">211</a>
<code>\liLinksReduktion</code> . . <a href="#">2352</a>	. . . . . <a href="#">1767</a> , <a href="#">1776</a>	<code>\liTuringUebergaenge</code>
<code>\liLinksReduktionInline</code>	<code>\liProblemVertexCover</code>	. . . . . <a href="#">203</a> , <a href="#">209</a>
. . . . . <a href="#">2361</a> , <a href="#">2369</a>	. . . . . <a href="#">1747</a> , <a href="#">1755</a>	<code>\liTuringUebergangZelle</code>
<code>\liMasterExkurs</code> . . . . <a href="#">2216</a>	<code>\liProduktionen</code> <a href="#">1334</a> , <a href="#">1376</a>	. . . . . <a href="#">198</a>
<code>\liMasterFaelle</code> <a href="#">2155</a> , <a href="#">2223</a>	<code>liProduktionsRegeln</code>	<code>\liTypoUeberGROSS</code> . . <a href="#">3118</a>
<code>\liMasterFallRechnung</code>	(environment) <a href="#">1324</a>	<code>\liTypoUeberGross</code> . . .
. . . . . <a href="#">2201</a>	<code>liProjektSprache (envi-</code>	. . . . . <a href="#">3112</a> , <a href="#">3118</a>
<code>\liMasterVariablen</code> . .	<code>ronment)</code> . . . . <a href="#">1956</a>	<code>\liTypoUeberschrift</code> .
. . . . . <a href="#">2132</a> , <a href="#">2218</a>	<code>\liPseudoUeberschrift</code>	. . . <a href="#">3106</a> , <a href="#">3115</a> , <a href="#">3121</a>
<code>\liMasterVariablenDeklaration</code>	. . . . . <a href="#">1933</a> ,	<code>\liUeberfuehrungsFunktion</code>
. . . . . <a href="#">2178</a>	. . . <a href="#">1989</a> , <a href="#">1990</a> , <a href="#">2295</a> ,	. . . . . <a href="#">1300</a>
<code>\liMasterWolframLink</code> <a href="#">2226</a>	. . . <a href="#">2305</a> , <a href="#">3179</a> , <a href="#">3187</a> , <a href="#">3195</a>	<code>\liUeberfuehrungsFunktionOhneMathe</code>
<code>\liMenge</code> . . . <a href="#">76</a> , <a href="#">77</a> , <a href="#">79</a> ,	<code>\liPumpingKontextfrei</code>	. . . <a href="#">1300</a> , <a href="#">1303</a> , <a href="#">1305</a>
. . . <a href="#">118</a> , <a href="#">119</a> , <a href="#">120</a> , <a href="#">124</a> ,	. . . . . <a href="#">2547</a>	<code>liUebergangsTabelle</code>
. . . <a href="#">176</a> , <a href="#">177</a> , <a href="#">178</a> , <a href="#">182</a> ,	<code>\liPumpingRegulaer</code> . <a href="#">2526</a>	(environment) <a href="#">2293</a>
. . . <a href="#">1286</a> , <a href="#">1335</a> , <a href="#">1374</a> , <a href="#">1375</a>	<code>liQuellen (environment)</code>	<code>\liUeberschriftDreiecksTabelle</code>
<code>\liMengeOhneMathe</code> . . .	. . . . . <a href="#">2014</a>	. . . . . <a href="#">2303</a>
. . . <a href="#">1286</a> , <a href="#">1289</a> , <a href="#">1291</a>	<code>\liRechtsReduktionInline</code>	<code>\liUmlLeserichtung</code> . <a href="#">3146</a>
<code>\liMetaSetze</code> . . . . .	. . . . . <a href="#">2369</a>	<code>\liVertauschen</code> . . . . <a href="#">2779</a>
. . . <a href="#">36</a> , <a href="#">48</a> , <a href="#">1563</a> , <a href="#">1656</a>	<code>\liRekursionsGleichung</code>	<code>\liWortInSprache</code> . . . <a href="#">730</a>
<code>\liMinimierungErklaerung</code>	. . . . . <a href="#">2119</a> , <a href="#">2182</a>	<code>\liWortNichtInSprache</code> <a href="#">735</a>
. . . . . <a href="#">2307</a>	<code>\liRelation</code> . . . . . <a href="#">2399</a>	<code>\liWpEquivalent</code> . . . <a href="#">3249</a>
<code>\liMinispracheDatei</code> <a href="#">2957</a>	<code>liRelationenSchemaFormat</code>	<code>\liWpErklaerung</code> . . . <a href="#">3252</a>
<code>\linespread</code> . . . . . <a href="#">2759</a>	(environment) <a href="#">2772</a>	<code>\liWpErklaerungVerzweigung</code>
<code>\liNichtsZuTun</code> . . . . <a href="#">3084</a>	<code>\liRelationMenge</code> . . <a href="#">2766</a>	. . . . . <a href="#">3267</a>
<code>\liO</code> . . . . . <a href="#">2095</a> , <a href="#">2123</a>	<code>\liRichtig</code> . . . . . <a href="#">548</a>	<code>\liWpKalkuel</code> . . . . . <a href="#">3229</a>



<code>\liWpKalkuelOhneMathe</code>	<code>\myNodes</code> . . . . . 2821,	<b>O</b>
. . . . . 3229, 3234,	2836, 2842, 2846, 2848	<code>\o</code> . . . . . 2415, 2420, 2422
3236, 3268, 3270, 3272		<code>\o@join</code> . . . . .
<code>\liZustandsBuchstabe</code>	<b>N</b>	2738, 2741, 2742, 2743
. . . . . 1310,	<code>\NeedsTeXFormat</code> . . . . .	<code>\Omega</code> . . . . . 2086
1319, 1321, 1339, 1341	. . . . . 1, 14, 32,	<code>\omega</code> 2528, 2529, 2549, 2550
<code>\liZustandsBuchstabeGross</code>	57, 225, 483, 544,	<code>\or</code> . . . . . 2649
. . . 1311, 1320, 1322	551, 613, 726, 741,	
<code>\liZustandsmenge</code> . . . 1298	1242, 1278, 1391,	<b>P</b>
<code>\liZustandsmengeNr</code> . .	1428, 1437, 1458,	<code>\pagestyle</code> . . . . 1424, 1662
. . . . . 1312, 2510	1487, 1525, 1610,	<code>\par</code> . . . . . 428, 449,
<code>\liZustandsmengeNrGross</code>	1716, 1788, 1871,	608, 1356, 1364,
. . . . . 1320	1887, 1893, 1915,	1546, 1550, 1553,
<code>\liZustandsMengenSammlung</code>	2068, 2231, 2244,	1566, 1573, 1584,
. . . . . 2497	2323, 2406, 2426,	1602, 1605, 1665,
<code>\liZustandsMengenSammlungNr</code>	2492, 2517, 2522,	1669, 1673, 1675,
. . . . . 2506	2566, 2732, 2745,	1676, 1683, 1685,
<code>\liZustandsmengeOhneMathe</code>	2774, 2855, 2862,	1690, 1694, 1698,
. . . . . 1298	2869, 2874, 2967,	1936, 1942, 1948,
<code>\liZustandsname</code> . . . 1321	2982, 3073, 3077,	2008, 2030, 2039,
<code>\liZustandsnameGross</code>	3126, 3168, 3205, 3225	2691, 2713, 2728,
. . . 1322, 2498, 2507	<code>\neg</code> . . . . . 3272	2769, 3068, 3092,
<code>\liZustandsnameTiefgestellt</code>	<code>\negthinspace</code> . . 2073, 2413	3096, 3104, 3257, 3263
. . . . . 1337	<code>\newcounter</code> . . . 2586, 2587	<code>\paragraph</code> . . . . . 1401
<code>\liZustandsPaar</code> . . . 2287	<code>\NewDocumentCommand</code> .	<code>\parindent</code> . . . . . 2760
<code>\liZustandsPaarVariablenName</code>	67, 108, 145, 166,	<code>\path</code> . . . . 99, 146, 209, 646
. . . 2286, 2289, 2290	208, 229, 618, 635,	<code>\pgfkeys</code> . . 2442, 3149,
<code>\llap</code> . . . . . 2739	674, 681, 730, 735,	3150, 3151, 3152,
<code>\log</code> . . . . . 2123,	1345, 1367, 1446,	3153, 3156, 3159, 3161
2126, 2129, 2158, 2164	1451, 1740, 1863,	<code>\pgfmath@count</code> . . . . .
<code>\loop</code> . . . . . 2572	1891, 2056, 2060,	. . . 2813, 2815, 2817
<code>\lor</code> . . . . . 3271	2388, 2399, 2486,	<code>\pgfmath@smuggleone</code> 2818
<code>\ltimes</code> . . . . . 2234	2922, 2932, 2942,	<code>\pgfmathdeclarefunction</code>
	2954, 2957, 2961, 3146	. . . . . 2812
<b>M</b>	<code>\NewDocumentEnvironment</code>	<code>\pgfmathhint</code> . . . . . 2813
<code>\makeatletter</code> . . . . . 2811	1324, 1414, 1485,	<code>\pgfmathparse</code> . . . . .
<code>\makeatother</code> . . . . . 2819	1848, 1956, 1957,	. . . . 1499, 2830,
<code>\marginpar</code> . . . . .	1958, 1985, 1993,	2835, 2838, 2852, 2853
1255, 1261, 1267, 1931	2015, 2034, 2044,	<code>\pgfmathresult</code> . . . . .
<code>\mathbb</code> . . 1778, 2559, 3191	2294, 2344, 2754, 2772	. . . . 1500, 2813,
<code>\mathbin</code> . 2741, 2742, 2743	<code>\newlength</code> . . . . . 3252	2814, 2816, 2818,
<code>\mathcal</code> . . . . .	<code>\newminted</code> . . . . . 2900	2831, 2839, 2852, 2853
2096, 2416, 3042,	<code>\node</code> . . . . . 631, 1856,	<code>\pgfutil@empty</code> . . . . . 2814
3047, 3049, 3050, 3051	1861, 2612, 2617,	<code>\pgfutil@loop</code> . . . . . 2815
<code>\Mathe</code> . . . . . 3246	2622, 2628, 2634,	<code>\pgfutil@repeat</code> . . . . 2818
<code>\MatheEnv</code> 3239, 3247, 3250	2640, 2791, 2836, 3163	<code>\preceq</code> . . . . . 1743
<code>\mathord</code> . . . . . 2241, 2242	<code>\noexpand</code> . . . . . 2700,	<code>\prime</code> . . . . . 1722
<code>\mdfsetup</code> . . . . . 1413,	2701, 2702, 2721, 2836	<code>\printbibliography</code> . 1913
1963, 1967, 1971, 1975	<code>\noindent</code> . . . . . 429,	<code>\ProvidesPackage</code> . . . . .
<code>\medskip</code> . . . . . 1568,	732, 737, 1551,	. . . . . 2, 15, 33,
1575, 1732, 1940,	1565, 1570, 1577,	58, 226, 484, 545,
1950, 1952, 2024,	1595, 1602, 1605,	552, 614, 727, 742,
2052, 2757, 2763,	1937, 1949, 1953,	1243, 1279, 1392,
3093, 3097, 3240, 3244	1979, 2007, 2009,	1429, 1438, 1459,
<code>\memph</code> . . . . . 1931	2022, 2038, 2040,	1488, 1526, 1611,
<code>\mintinline</code> . . . . 2907,	2048, 2220, 2273,	1717, 1789, 1872,
2914, 2953, 2960, 2965	2276, 2279, 2282,	1888, 1894, 1916,
<code>\mkern</code> . . . 2741, 2742, 2743	2767, 3086, 3098, 3258	2069, 2232, 2245,
<code>\mlq</code> . . . . . 2239, 2241	<code>\nolinkurl</code> . . . . . 2919	2324, 2407, 2427,
<code>\mrq</code> . . . . . 2239, 2242	<code>\normalsize</code> . . . 1401, 2908	2493, 2518, 2523,
<code>\msg</code> . . . . . 477	<code>\notin</code> . . . . . 738	2567, 2733, 2746,
<code>\myList</code> . . . . .	<code>\null</code> . . . . . 3091	2775, 2856, 2863,
2832, 2833, 2834, 2837		2870, 2875, 2968,

2983, 3074, 3078, 3127, 3169, 3206, 3226	1920, 1921, 2071, 2235, 2236, 2409, 2410, 2411, 2428, 2430, 2520, 2580, 2582, 2734, 2735, 2736, 2749, 2777, 2808, 2859, 2865, 2872, 2877, 2888, 2889, 2970, 3075, 3082, 3130, 3131, 3133, 3134, 3172, 3173, 3174, 3207, 3227	\setminusminus ..... 2373 \setul ..... 2751 \shoveleft ..... 2353 \shoveright ..... 2357 \Sigma ..... 69, 110, 168, 1308, 1309, 1369 \sigma ..... 582, 584, 585 \SLASH ..... 1932 \small ..... 2047 \sort ..... 2825 \sortList ..... 2824, 2833 \square ..... 549 \stepcounter 2612, 2617, 2622, 2625, 2627, 2631, 2633, 2637, 2639 \stichwoerter .. 1542, 1578 \str 361, 560, 569, 1960, 2502, 2511, 2992, 3005 \string ..... 2355, 2365 \StrSubstitute . 2832, 2834 \strut .... 2180, 2184, 2188, 2192, 2196, 2860 \subsection ..... 1636 \subseteq 3016, 3049, 3056 \subsubsection ..... 1657	
<b>Q</b>		<b>T</b>	
\QS@list ..... 2693, 2704, 2708, 2715, 2721, 2726, 2729	\right ..... 2073, 2413 \RIGHTarrow ... 3148, 3153 \Rightarrow .... 733, 738 \rightarrow ..... 212, 572, 577, 585, 589, 591, 592, 594, 649, 657, 2490, 3013, 3020, 3022, 3025, 3030, 3035, 3036, 3041 \rightouterjoin .... 2742 \rmfamily ..... 559, 1519, 2991, 3108 \Roman ..... 1409 \roman ..... 1409, 1411 \romannumeral ..... 2658 \rtimes ..... 2234 \rule ..... 1552, 2691, 2713, 2728, 2739	\tableofcontents .... ..... 1711, 1927 \TeX ..... 1603 \text ..... 82, 84, 187, 2335, 3177, 3230 \textbf ..... 696, 714, 1250, 1544, 1748, 1757, 1768, 1777, 1938, 1951, 1980, 2008, 2023, 2039, 2298 \textcolor 1860, 2906, 3176 \textit ... 1090, 1132, 1133, 1134, 1135, 1557, 2049, 2343, 2403 \textsc ..... 1723 \textsf ..... 1938, 2023 \textstyle .... 2143, 2175 \texttt 1273, 1723, 1857, 1858, 1859, 1860, 3230 \textwidth .... 1552, 1884 \thematik ..... 1555, 1572 \thepage ..... 1423, 1879 \theparagraph ..... 1401 \thesection ..... 1623 \Theta ..... 2076 \thinspace ..... 3230 \tikz ..... 1861 tikz: bbaum ..... 26 tikz: li binaer baum .... 24 \tikzchildnode ..... 504 \tikzparentnode ..... 504 \tikzset ..... 101, 148, 214, 489, 515, 1464, 1792, 2468, 2594, 2798, 2972, 3208 \tikzumlset ..... 3136 \times ..... 212	
\QS@select@equal .... ..... 2668, 2672			
\QS@select@greater .. ..... 2669, 2673			
\QS@select@smaller .. ... 2664, 2667, 2671			
\QS@sort@a ..... 2646, 2679, 2700, 2701			
\QS@sort@b .... 2646, 2647			
\QS@sort@c .... 2650, 2657			
\QS@sort@d .... 2658, 2666			
\QS@sort@empty . 2648, 2653			
\QS@sort@single 2649, 2654			
\QSinitialize ..... ... 2571, 2683, 2718			
\QSIr . 2654, 2660, 2668, 2688, 2702, 2707, 2710			
\QSIrr 2689, 2702, 2703, 2711			
\QSLr ..... 2660, 2667, 2678, 2679, 2687, 2700, 2705, 2709			
\QSPivotStep ..... 2573, 2683, 2687, 2698			
\QSR ..... 2660			
\QSRr ..... 2669, 2690, 2701, 2706, 2712, 2721, 2722, 2723	<b>S</b> \sb ..... 72, 82, 84, 113, 171, 584, 585, 589, 592, 593, 594, 676, 678, 683, 685, 1339, 1341, 2123, 2126, 2129, 2158, 2164, 2313, 2478, 2487, 3035, 3036, 3037, 3042, 3046, 3047, 3050, 3051, 3054, 3055, 3056 \scriptscriptstyle .. ..... 649, 657, 665 \scriptsize .... 1358, 1801, 1808, 1814, 1876, 1877, 1880, 1881, 3177, 3230, 3260 \section ... 52, 1623, 1632 \sectionbreak ..... 1624 \seq . 1865, 1866, 1867, 2017, 2018, 2019, 2026 \setbox ..... 2738 \setcounter 1402, 1625, 1659, 2692, 2714, 2728 \setganttlinklabel .. 1432, 1433, 1434, 1435 \setlength ..... 1704, 1705, 1707, 1884, 2760, 3254, 3255, 3265 \setmainfont ..... 1395 \setmainlanguage .... 481 \setminted .... 2894, 2895		
<b>R</b>			
\raisebox ..... 1861			
\relax ..... 1926, 2660, 2703, 2705, 2706, 2707, 2815, 2817			
\renewcommand ..... 1404, 1405, 1406, 1407, 1410, 1411, 1425, 1426, 1706, 1710, 1882, 1883, 2293			
\repeat ..... 2576			
\RequirePackage ..... ..... 4, 60, 163, 227, 231, 480, 486, 487, 547, 615, 744, 745, 1245, 1247, 1248, 1284, 1393, 1394, 1396, 1398, 1403, 1412, 1419, 1430, 1431, 1441, 1442, 1460, 1461, 1462, 1490, 1491, 1540, 1622, 1627, 1628, 1660, 1721, 1790, 1890, 1895, 1896, 1912, 1919,			

<code>\tiny</code> . 1256, 1262, 1268, 1594, 1860, 1931, 2917	<code>\TmpTransitionOne</code> ... ..... 2432, 2453	<code>\umlVHVdep</code> ..... 785, 786, 788, 789, 972, 973
<code>\titleformat</code> ..... ... 1399, 1401, 1623	<code>\TmpTransitionSeven</code> . ..... 2438, 2459	<code>\umlVHVinherit</code> 768, 769, 774, 775, 782, 783, 941, 942, 966, 967, 1145, 1146, 1183, 1184
<code>\titlespacing</code> ..... 1400	<code>\TmpTransitionSix</code> ... ..... 2437, 2458	<code>\umlVHVreal</code> ..... . 935, 936, 1206, 1207
<code>\tl</code> ..... 39, 68, 69, 70, 71, 72, 73, 76, 77, 78, 79, 80, 82, 84, 109, 110, 111, 112, 113, 114, 115, 118, 119, 120, 121, 122, 123, 124, 167, 168, 169, 170, 171, 172, 173, 176, 177, 178, 179, 180, 181, 182, 303, 307, 334, 338, 339, 340, 343, 348, 349, 350, 362, 371, 406, 409, 412, 420, 421, 422, 423, 434, 440, 443, 446, 462, 469, 619, 622, 627, 628, 636, 637, 640, 641, 1347, 1368, 1369, 1370, 1371, 1374, 1375, 1376, 1377, 1543, 1556, 1631, 1635, 1639, 1646, 1649, 2106	<code>\TmpTransitionTen</code> ... ..... 2441, 2462	<code>\UParrow</code> ..... 3150
<code>\tmp</code> ..... 2375	<code>\TmpTransitionThree</code> . ..... 2434, 2455	<code>\url</code> ..... 463, 2057
<code>\TmpPlaceEight</code> ..... 2450	<code>\TmpTransitionTwo</code> ... ..... 2433, 2454	<code>\usemintedstyle</code> .... 2891
<code>\TmpPlaceFive</code> ..... 2447	<code>\TmpX</code> ..... 2464	<code>\usetikzlibrary</code> .. 61, 488, 1246, 1463, 1791, 2429, 2737, 2778, 2809, 3135, 3223
<code>\TmpPlaceFour</code> ..... 2446	<code>\TmpY</code> ..... 2465	
<code>\TmpPlaceNine</code> ..... 2451	<code>\today</code> ..... 1877	
<code>\TmpPlaceOne</code> ..... 2443	<code>\ttfamily</code> ..... 2752	
<code>\TmpPlaceSeven</code> ..... 2449		<b>V</b>
<code>\TmpPlaceSix</code> ..... 2448	<b>U</b>	<code>\value</code> ..... 2574
<code>\TmpPlaceTen</code> ..... 2452	<code>\ul</code> ..... 1251, 2750, 2751	<code>\varepsilon</code> ..... 561, 572, 573, 1294, 2123, 2129, 2161, 2172
<code>\TmpPlaceThree</code> ..... 2445	<code>\umlagg</code> ..... 1209	<code>\vfill</code> ..... 1590, 1671, 1678, 1709, 2860
<code>\TmpPlaceTwo</code> ..... 2444	<code>\umlassoc</code> ..... 1167	<code>\vrule</code> ..... 3087, 3091
<code>\TmpScale</code> ..... 2463	<code>\umlclass</code> ..... 756, 760, 764, 811, 812, 813, 860, 865, 870, 873, 931, 932, 933, 938, 939, 968, 1003, 1034, 1035, 1038, 1089, 1092, 1131, 1137, 1138, 1156, 1157, 1158, 1179, 1180, 1181, 1201, 1202, 1203, 1204	<code>\vspace</code> ... 1667, 1688, 1692, 1696, 1706, 1994, 2012, 3100, 3102
<code>\TmpTransitionEight</code> . ..... 2439, 2460	<code>\umldep</code> ..... 1097	<b>X</b>
<code>\TmpTransitionFive</code> .. ..... 2436, 2457	<code>\umlHVHagg</code> ..... ..... 878, 944, 1147	<code>\xappto</code> .. 2836, 2842, 2846
<code>\TmpTransitionFour</code> .. ..... 2435, 2456	<code>\umlinherit</code> ..... 817, 868, 1042, 1087, 1095	<code>\xdef</code> ..... 1493
<code>\TmpTransitionNine</code> .. ..... 2440, 2461	<code>\umlnote</code> .. 819, 1044, 1211	<code>\xintApply</code> ..... 2662
	<code>\umlreal</code> ..... 815, 876	<code>\xintApplyUnbraced</code> .. 2661, 2667, 2668, 2669
	<code>\umlsimpleclass</code> ..... ... 771, 772, 773, 777, 779, 780, 781, 810, 963, 964, 965, 1033, 1085, 1086, 1177	<code>\xintCSVtoList</code> ..... 2721
	<code>\umlstatic</code> .... 970, 1004	<code>\xintFor</code> ..... 2611, 2616, 2621, 2626, 2632, 2638, 2679
	<code>\umluniagg</code> ..... 1040	<code>\xintIfEq</code> ..... 2672
	<code>\umluniassoc</code> ..... ... 791, 816, 1041, 1165, 1166, 1185, 1186	<code>\xintIfForLast</code> ..... ... 2628, 2634, 2640
	<code>\umlVHuniassoc</code> .. 792, 793	<code>\xintIfGt</code> ..... 2673
		<code>\xintIfLt</code> ..... 2671
		<code>\xintLength</code> ..... 2646
		<code>\xintntheft</code> ..... 2658
		<b>Z</b>
		<code>\ZB</code> ..... 2064
		<code>\zB</code> ..... 2063
		<code>\zustandsnamens@liste</code> ... 1312, 1319, 1320