

# lehramt-informatik

Hermine Bschlangaul \*

September 5, 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	22
2.6.1	Binärbaum	23
2.6.2	AVL-Baum	24
2.6.3	B-Baum	25
2.7	checkbox.sty	26
2.8	chomsky-normalform.sty	27
2.8.1	Makro-Kürzel	27
2.8.2	TeX-Markup-Grundgerüst	27
2.8.3	Konkretes TeX-Markup-Beispiel	27
2.9	cpm.sty	30
2.9.1	Makro-Kürzel	30
2.9.2	Hilfsmakros für Zeitpunkt-Tabelle	31
2.10	cyk-algorithmus.sty	33
2.10.1	Makro-Kürzel	33
2.10.2	TeX-Markup-Beispiel	33
2.11	entwurfsmuster.sty	34
2.11.1	Namensschema der Entwurfsmuster-Makros:	34
2.11.2	Reihenfolge	34
2.11.3	Abstrakte Fabrik (Abstract Factory)	34
2.11.4	Adapter	35
2.11.5	Beobachter (Observer)	37
2.11.6	Dekorierer (Decorator)	39
2.11.7	Einfache Fabrik (Simple Factory)	40
2.11.8	Einzelstück (Singleton)	41
2.11.9	Erbauer (Builder)	41
2.11.10	Fabrikmethode (Factory Method)	43
2.11.11	Kompositum (Composite)	44
2.11.12	Modell-Präsentation-Steuerung (Model-View-Controller)	45

---

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

2.11.13 Zustand (State)	46
2.12 er.sty	48
2.12.1 Tex-Markup-Beispiel: Komplettes Diagramm	48
2.12.2 Tex-Markup-Beispiel: EER Enhanced Entity-Relation-Modell nach Elmasri/Navante	49
2.12.3 Makro-Kürzel	49
2.13 formale-sprachen.sty	51
2.14 formatierung.sty	54
2.14.1 Schriftarten / Typographie	54
2.14.2 Farben	54
2.14.3 Überschriften	54
2.14.4 Listen	54
2.14.5 Kasten	54
2.14.6 Header	54
2.15 gantt.sty	55
2.16 grafik.sty	56
2.17 graph.sty	57
2.18 hanoi.sty	59
2.19 klassen-konfiguration-aufgabe.sty	60
2.20 klassen-konfiguration-examen.sty	61
2.21 komplexitaetstheorie.sty	62
2.21.1 Makro-Kürzel	62
2.22 kontrollflussgraph.sty	64
2.22.1 Makro-Kürzel	64
2.22.2 TeX-Markup-Beispiel	64
2.22.3 TikZ: pin	64
2.22.4 Umgebungen	65
2.22.5 Makros	66
2.23 kopf-fusszeilen.sty	67
2.24 literatur-dummy.sty	68
2.25 literatur.sty	69
2.26 makros.sty	70
2.27 master-theorem.sty	74
2.27.1 Makro-Kürzel	74
2.28 mathe.sty	78
2.29 minimierung.sty	79
2.30 normalformen.sty	82
2.30.1 Makro-Kürzel	82
2.31 petri.sty	85
2.31.1 Makro-Kürzel	85
2.32 potenzmengen-konstruktion.sty	87
2.33 pseudo.sty	89
2.34 pumping-lemma.sty	90
2.35 quicksort.sty	91
2.36 relationale-algebra.sty	94
2.37 rmodell.sty	95
2.37.1 Makro-Kürzel	95
2.38 sortieren.sty	96
2.39 spalten.sty	98
2.40 sql.sty	99
2.41 struktogramm.sty	100
2.42 syntax.sty	101
2.42.1 Makro-Kürzel	101
2.43 syntaxbaum.sty	103
2.44 synthese-algorithmus.sty	104
2.44.1 Makro-Kürzel	104
2.44.2 TeX-Markup Grundgerüst	104
2.44.3 TeX-Markup Linksreduktion	104

2.44.4	TeX-Markup Rechtreduktion	104
2.44.5	TeX-Markup Relationen formen	105
2.45	tabelle.sty	108
2.46	typographie.sty	109
2.47	uml.sty	110
2.48	vollstaendige-induktion.sty	112
2.48.1	Makro-Kürzel	112
2.49	wasserfall.sty	114
2.50	wpkalkuel.sty	115
2.50.1	Makro-Kürzel	115
<b>3</b>	<b>Index</b>	<b>116</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

```

`\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,
}

```

```

36 \def\liAufgabenMetadaten#1{
37
38   \_setze_variablen_zurueck:
39
40   \tl_clear:N \g_auf_examen_telaufgabe_nr_tl
41
42   \keys_set:nn { aufgabenmetadaten } {
43     #1
44   }
45
46   \_setze_relativen_pfad:
47
48   \tl_if_empty:NTF \g_auf_titel_tl
49   {
50     \msg_fatal:nn { aufgabenmetadaten } { kein-titel }
51   }
52   {
53   }
54
55   \_gib_examen_titel: {}
56
57   \section{\_gib_aufgaben_titel:}
58
59   \bigskip
60 }

```

```

61 \ExplSyntaxOff

```

`\liAufgabenTitel` Momentan eine dummy Makro das die Thematik enthält.

```

62 \def\liAufgabenTitel#1{}

63

```

## 2.4 automaten.sty

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

### 2.4.1 Endlicher Automat

```
66 \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}
```



```
67 \RequirePackage{tikz}
68 \usetikzlibrary{arrows,automata,positioning}
69 \liLadePakete{mathe}
70 \directlua{
71   automaten = require('lehramt-informatik-automaten')
72 }
```

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

```
73 \ExplSyntaxOn
74 \NewDocumentCommand {\liAutomat} { 0{A} m } {
75   \tl_set:Nn \l_zustaeende_tl {Z}
76   \tl_set:Nn \l_alphabet_tl {\Sigma}
77   \tl_set:Nn \l_delta_tl {\delta}
78   \tl_set:Nn \l_ende_tl {E}
79   \tl_set:Nn \l_start_tl {z\sb{0}}
80   \tl_set:Nn \l_typ_tl {}
81
82   \keys_define:nn { automat } {
83     zustaeende .code:n = {\tl_set:Nn \l_zustaeende_tl {\liMenge{##1}}},
84     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
85     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
86     ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
```

```

87     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
88     dea .value_forbidden:n = true,
89     dea .code:n = {\tl_set:Nn \l_typ_tl {\sb{\text{DEA}}}},
90     nea .value_forbidden:n = true,
91     nea .code:n = {\tl_set:Nn \l_typ_tl {\sb{\text{NEA}}}},
92   }
93
94   \keys_set:nn { automat } { #2 }
95
96   $#1 \l_typ_tl = (
97     \l_zustaende_tl,
98     \l_alphabet_tl,
99     \l_delta_tl,
100    \l_ende_tl,
101    \l_start_tl
102  )$
103 }
104 \ExplSyntaxOff

```

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

```

105 \def\liAutomatenKante#1#2#3#4{
106   \path (#1) edge[#4] node{#3} (#2);
107 }

108 \tikzset{
109   li automat/.style={
110     ->,
111     node distance=2cm
112   },
113 }

```

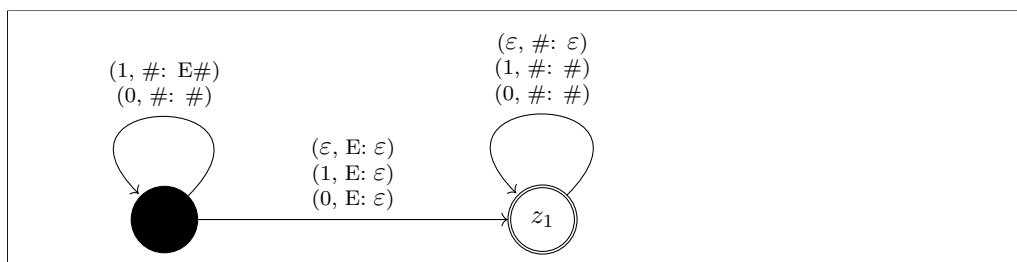
## 2.4.2 Kellerautomat

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

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

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

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



```
\liKellerAutomat \liKellerAutomat[\langle automaten-name \rangle]
{\langle zustaeende=Z,alphabet=Sigma,kelleralphabet=Gamma,delta=delta,start=z0,kellerboden=#,ende=E \rangle}

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

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

```
114 \ExplSyntaxOn
115 \NewDocumentCommand {\liKellerAutomat} { O{K} m } {
116   \tl_set:Nn \l_zustaeende_tl {Z}
117   \tl_set:Nn \l_alphabet_tl {\Sigma}
118   \tl_set:Nn \l_kelleralphabet_tl {\Gamma}
119   \tl_set:Nn \l_delta_tl {\delta}
120   \tl_set:Nn \l_start_tl {z\sb{0}}
121   \tl_set:Nn \l_kellerboden_tl {\#}
122   \tl_set:Nn \l_ende_tl {E}
123
124   \keys_define:nn { kellerautomat } {
125     zustaeende .code:n = {\tl_set:Nn \l_zustaeende_tl {\liMenge{##1}}},
126     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
127     kelleralphabet .code:n = {\tl_set:Nn \l_kelleralphabet_tl {\liMenge{##1}}},
128     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
129     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
```

```

130     kellerboden .code:n = {\tl_set:Nn \l_kellerboden_tl {##1}},
131     ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
132   }
133
134   \keys_set:nn { kellerautomat } { #2 }
135
136   $#1 = (
137     \l_zustaende_tl,
138     \l_alphabet_tl,
139     \l_kelleralphabet_tl,
140     \l_delta_tl,
141     \l_start_tl,
142     \l_kellerboden_tl,
143     \l_ende_tl
144   )$
145 }
146 \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}`

```

147 \ExplSyntaxOn
148 \def\liKellerUebergang#1{
149   \directlua{automaten.drucke_keller_uebergaenge('#1')}
150 }
151 \ExplSyntaxOff

```

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

```

152 \NewDocumentCommand{\liKellerKante} { 0{above} m m m } {
153   \path (#2) edge[#1] node{\liKellerUebergang{#4}} (#3);
154 }

155 \tikzset{
156   li keller knoten/.style={
157     text width=2cm,
158     align=center,
159     font=\footnotesize,
160   },
161   li kellerautomat/.style={
162     li automat,
163     every edge/.append style={
164       every node/.style={
165         li keller knoten
166       }
167     }
168   }
169 }

```

### 2.4.3 Turingmaschine

```

170 \RequirePackage{amssymb}

```

**\liTuringLeerzeichen**

□

```

171 \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},
}

```

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

```

172 \ExplSyntaxOn
173 \NewDocumentCommand {\liTuringMaschine} { O{TM} m } {
174   \tl_set:Nn \l_zustaende_tl {Z}
175   \tl_set:Nn \l_alphabet_tl {\Sigma}
176   \tl_set:Nn \l_bandalphabet_tl {\Gamma}
177   \tl_set:Nn \l_delta_tl {\delta}
178   \tl_set:Nn \l_start_tl {z\sb{0}}
179   \tl_set:Nn \l_leerzeichen_tl {\liTuringLeerzeichen}
180   \tl_set:Nn \l_ende_tl {E}
181
182   \keys_define:nn { kellerautomat } {
183     zustaende .code:n = {\tl_set:Nn \l_zustaende_tl {\liMenge{##1}}},
184     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
185     bandalphabet .code:n = {\tl_set:Nn \l_bandalphabet_tl {\liMenge{##1}}},
186     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
187     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
188     leerzeichen .code:n = {\tl_set:Nn \l_leerzeichen_tl {##1}},
189     ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
190   }
191
192   \keys_set:nn { kellerautomat } { #2 }
193
194   $\text{#1} = (
195     \l_zustaende_tl,
196     \l_alphabet_tl,
197     \l_bandalphabet_tl,
198     \l_delta_tl,
199     \l_start_tl,
200     \l_leerzeichen_tl,
201     \l_ende_tl
202   )$
203 }
204 \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)`

```

205 \ExplSyntaxOn
206 \def\liTuringUebergangZelle#1{
207   \directlua{tex.print(automaten.gib_einen_turing_uebergang('#1'))}
208 }
209 \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)`



```

210 \ExplSyntaxOn
211 \def\liTuringUebergaenge#1{
212   \directlua{automaten.drucke_turing_uebergaenge('#1')}
213 }
214 \ExplSyntaxOff

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

215 \NewDocumentCommand{\liTuringKante} { 0{above} m m m } {
216   \path (#2) edge[#1] node{\liTuringUebergaenge{#4}} (#3);
217 }

\liTuringUeberfuehrung

218 \def\liTuringUeberfuehrung{
219    $\delta : Z \times \Gamma \rightarrow Z \times \Gamma \times \{ L, R, N \}$ 
220 }

221 \tikzset{
222   li turingmaschine/.style={
223     li automat,
224     every edge/.append style={
225       every node/.style={
226         li keller knoten
227       }
228     }
229   }
230 }
231

```

## 2.5 basis.sty

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

234 \RequirePackage{xparse}

235 \ExplSyntaxOn

\liLadePakete

236 \NewDocumentCommand{\liLadePakete}{ m }
237 {
238   \clist_map_inline:nn { #1 } { \RequirePackage{lehramt-informatik-##1} }
239 }

\liLadeAllePakete

240 \def\liLadeAllePakete{
241   \liLadePakete{
242     aufgaben-einbinden,
243     automaten,
244     baum,
245     checkbox,
246     chomsky-normalform,
247     cpm,
248     cyk-algorithmus,
249     entwurfsmuster,
250     er,
251     formale-sprachen,
252     gantt,
253     grafik,
254     graph,
255     hanoi,
256     kontrollflussgraph,
257     makros,
258     master-theorem,
259     mathe,
260     minimierung,
261     normalformen,
262     petri,
263     potenzmengen-konstruktion,
264     pseudo,
265     quicksort,
266     relationale-algebra,
267     rmodell,
268     sortieren,
269     spalten,
270     struktogramm,
271     syntax,
272     syntaxbaum,
273     synthese-algorithmus,
274     tabelle,
275     typographie,
276     uml,
277     vollstaendige-induktion,
278     wasserfall,
279     wpkalkuel,
280   }
281 }
```

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.

```
282 \clist_new:N \g_auf_schluesssel_clist
283 \clist_set:Nn \g_auf_schluesssel_clist {
```

```

284  titel,
285  thematik,
286  stichwoerter,
287  zitat_schluessel,
288  zitat_beschreibung,
289  %
290  bearbeitungs_stand,
291  korrektheit,
292  %
293  relativer_pfad,
294  identische_aufgabe,
295  %
296  examen_nummer,
297  examen_jahr,
298  examen_monat,
299  examen_thema_nr,
300  examen_telaufgabe_nr,
301  examen_aufgabe_nr,
302 }

```

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

```

303 \clist_map_inline:Nn \g_auf_schluessel_clist {
304   \tl_new:c {g_auf_#1_tl}
305 }

```

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

```

306 \cs_new:Npn \setze_variablen_zurueck: {
307   \clist_map_inline:Nn \g_auf_schluessel_clist {
308     \tl_clear:c {g_auf_##1_tl}
309   }
310 }

```

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

```

311 \keys_define:nn { aufgabenmetadaten }
312 {
313   Titel .tl_gset:N = \g_auf_titel_tl,
314   Thematik .tl_gset:N = \g_auf_thematik_tl,
315   Stichwoerter .tl_gset:N = \g_auf_stichwoerter_tl,
316   ZitatSchluessel .tl_gset:N = \g_auf_zitat_beschreibung_tl,
317   ZitatBeschreibung .tl_gset:N = \g_auf_zitat_schluessel_tl,
318   %
319   BearbeitungsStand .tl_gset:N = \g_auf_bearbeitungs_stand_tl,
320   Korrektheit .tl_gset:N = \g_auf_korrekttheit_tl,
321   %
322   RelativerPfad .tl_gset:N = \g_auf_relativer_pfad_tl,
323   IdentischeAufgabe .tl_gset:N = \g_auf_identische_aufgabe_tl,
324   %
325   ExamenNummer .tl_gset:N = \g_auf_examen_nummer_tl,
326   ExamenJahr .tl_gset:N = \g_auf_examen_jahr_tl,
327   ExamenMonat .tl_gset:N = \g_auf_examen_monat_tl,
328   ExamenThemaNr .tl_gset:N = \g_auf_examen_thema_nr_tl,
329   ExamenTeilaufgabeNr .tl_gset:N = \g_auf_examen_telaufgabe_nr_tl,
330   ExamenAufgabeNr .tl_gset:N = \g_auf_examen_aufgabe_nr_tl,
331 }
332 \cs_gset:Npn \setze_relativen_pfad: {
333   \tl_if_empty:NTF \g_auf_relativer_pfad_tl
334   {
335     \bool_if:nTF
336     {
337       ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
338       ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
339       ! \tl_if_empty_p:N \g_auf_examen_monat_tl

```

```

340     }
341     {
342         \tl_gset:Nn \g_auf_relativer_pfad_tl {
343             Staatsexamen /
344             \g_auf_examen_nummer_tl /
345             \g_auf_examen_jahr_tl /
346             \g_auf_examen_monat_tl /
347             \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} { Thema - \g_auf_examen_thema_nr_tl / }
348             \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} { Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl / }
349             \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} { Aufgabe - \g_auf_examen_aufgabe_nr_tl / }
350         }
351     }
352     {}
353 }
354 {}
355 }

356 \cs_gset:Npn \_gib_examen_titel: {
357     \cs_set:Nn \_trenner: { \, / \, }
358     \bool_if:nTF
359     {
360         ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
361         ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
362         ! \tl_if_empty_p:N \g_auf_examen_monat_tl &&
363         ! \tl_if_empty_p:N \g_auf_examen_aufgabe_nr_tl
364     }
365     {
366         {
367             \footnotesize
368             \par
369             \noindent
370             Staatsexamen ~
371             \g_auf_examen_nummer_tl \_trenner:
372             \g_auf_examen_jahr_tl \_trenner:
373
374             \tl_case:Nn \g_auf_examen_monat_tl
375             {
376                 { 03 } { Frühjahr }
377                 { 09 } { Herbst }
378             } \_trenner:
379
380             \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
381                 Thema ~ Nr. ~ \g_auf_examen_thema_nr_tl \_trenner:
382             }
383             \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
384                 Teilaufgabe ~ Nr. ~ \g_auf_examen_teilaufgabe_nr_tl \_trenner:
385             }
386             \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} {
387                 Aufgabe ~ Nr. ~ \g_auf_examen_aufgabe_nr_tl
388             }
389             \par
390             \bigskip
391         }
392     }
393 }

394 \cs_new:Npn \_gib_github_url: {
395     \tl_if_empty:NTF \g_auf_relativer_pfad_tl {} {
396         Github :~\href{
397             \LehramtInformatikGithubRawDomain /
398             \LehramtInformatikGithubTexRepo /
399             \LehramtInformatikGitBranch /
400             \g_auf_relativer_pfad_tl
401         }{
402             \nolinkurl{\g_auf_relativer_pfad_tl}

```

```

403     }
404   }
405 }

406 \cs_new:Npn \_gib_aufgaben_titel: {
407   \g_auf_titel_tl
408
409   \tl_if_empty:NTF \g_auf_thematik_tl
410   {}
411   {
412     \, ~ [
413       \g_auf_thematik_tl
414     ]
415   }
416 }

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

\def\LehramtInformatikRepository{/pfad/zum/repository}
419 \input /etc/lehramt-informatik.config.tex
    biblatex not working with lualatex and babel
420 % \RequirePackage{polyglossia}
421 % \setmainlanguage{german}
422

```

## 2.6 baum.sty

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

426 \RequirePackage{tikz}

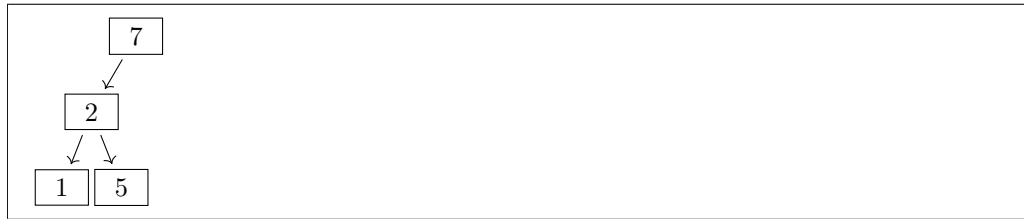
    für li binaer baum
427 \RequirePackage{tikz-qtrees}

    Für b baum
428 \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}
```



```
429 \tikzset{
430   li binaer baum/.style={
431     shorten <=2pt,
432     shorten >=2pt,
433     ->,
434     every tree node/.style={
435       minimum width=2em,
436       draw,
437       rectangle
438     },
439     blank/.style={
440       draw=none
441     },
442     edge from parent/.style={
443       draw,
444       edge from parent path={(\tikzparentnode) -- (\tikzchildnode)}
445     },
446     level distance=1cm,
447     every label/.style={
448       gray,
449       font=\footnotesize,
450       label position=0,
451       label distance=0cm,
452     }
453   },
454 }
```

### 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}
```



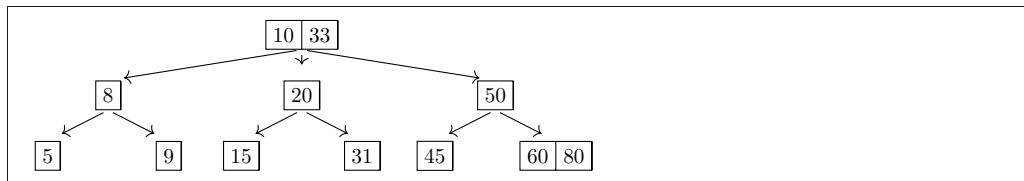


### 2.6.3 B-Baum

```

\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}

```



```

455 \tikzset{
456   li bbaum knoten/.style={
457     rectangle split parts=10,
458     rectangle split,
459     rectangle split horizontal,
460     rectangle split ignore empty parts,
461     draw,
462     fill=white
463   },
464   li bbaum/.style={
465     every node/.style={
466       li bbaum knoten
467     },
468     level 1/.style={
469       level distance=12mm,
470       sibling distance=25mm,
471     },
472     every child/.style={
473       shorten <= 2pt,
474       shorten >= 6pt,
475       ->,
476     },
477     level 2/.style={
478       level distance=9mm,
479       sibling distance=15mm,
480     },
481   }
482 }
483

```

## 2.7 checkbox.sty

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

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

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

490
```

## 2.8 chomsky-normalform.sty

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

494 \ExplSyntaxOn

495 \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

```

496 \def\liChomskyUeberschrift#1{
497   {
498     \bfseries
499     \sffamily
500     \str_case:nn {#1} {
501       {1} {Elimination-der- $\varepsilon$ -Regeln}
502       {2} {Elimination-von-Kettenregeln}
503       {3} {Separation-von-Terminalzeichen}
504       {4} {Elimination-von-mehrelementigen-Nonterminalketten}
505     }
506   }
507 }

```

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

```

508 \def\liChomsky@erklaerung@texte#1{
509   \str_case:nn {#1} {
510     %
511     {1} {
512       Alle-Regeln-der-Form- $A \rightarrow \varepsilon$ -werden-eliminiert.~
513       Die-Ersetzung-von- $A$ -wird-durch- $\varepsilon$ -in-alen-andere-
514       Regeln-vorweggenommen.
515     }
516     {2} {
517       Jede-Produktion-der-Form- $A \rightarrow B$ -mit- $A, B$ -in- $S$ -wird-
518       als-Kettenregel-bezeichnet.~Diese-tragen-nicht-zur-Produktion-
519       von-Terminalzeichen-bei-und-lassen-sich-ebenfalls-eliminieren.
520     }
521     {3} {
522       Jedes-Terminalzeichen- $\sigma$ ,~das-in-Kombination-mit-andere-
523       Symbolen-auftaucht,~wird-durch-ein-neues-Nonterminal-
524        $S_{\sigma}$ -ersetzt-und-die-Menge-der-Produktionen-durch-die-
525       Regel- $S_{\sigma} \rightarrow \sigma$ -ergänzt.
526     }
527     {4} {
528       Alle-Produktionen-der-Form-
529        $A \rightarrow B_{\{1\}} B_{\{2\}} \dots B_{\{n\}}$ -
530       werden-in-die-Produktionen-
531        $A \rightarrow$ 
532        $A_{\{n-1\}} B_{\{n\}}, A_{\{n-1\}} \rightarrow$ 
533        $A_{\{n-2\}} B_{\{n-1\}}, \dots,$ 
534        $A_{\{2\}} \rightarrow B_{\{1\}} B_{\{2\}}$ -zerteilt.~
535       Nach-der-Ersetzung-sind-alle-längere-Nonterminalketten-
536       vollständig-heruntergebrochen-und-die-Chomsky-Normalform-erreicht.
537     }
538   }

```

```

539 }
540 \def\liChomskyErklaerung#1{
541   {
542     \itshape
543     \footnotesize
544     \liParagraphMitLinien{\li@chomsky@erklaerung@texte{#1}}
545   }
546 }

```

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

```

547 \def\liChomskyUeberErklaerung#1{
548   \liChomskyUeberschrift{#1}\par
549   \liChomskyErklaerung{#1}
550 }

```

```

551 \ExplSyntaxOff
552

```

## 2.9 cpm.sty

```

553 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
554 \ProvidesPackage{lehramt-informatik-cpm}[2020/09/03]
555 \RequirePackage{tikz}
556 \liLadePakete{mathe}

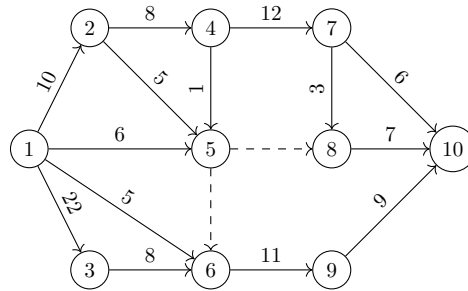
```

### 2.9.1 Makro-Kürzel

```

\let\f=\footnotesize
\let\FZ=\liCpmFruehesterI
\let\SZ=\liCpmSpaetesterI
\let\v=\liCpmVon
\let\ vz=\liCpmVonZu
\let\ z=\liCpmZu

```



```

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

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

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

```

```

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

```

```

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

```

```

557 \ExplSyntaxOn
558 \NewDocumentCommand { \liCpmEreignis } { 0{} m m m } {
559   \tl_set:Nn \l_name_tl {}
560
561   \keys_define:nn { cpmEreignis } {
562     name .code:n = {\tl_set:Nn \l_name_tl {##1}},
563   }
564
565   \keys_set:nn { cpmEreignis } { #1 }
566
567   \tl_if_empty:NT \l_name_tl {
568     \tl_set:Nn \l_name_tl {#2}
569   }

```

```

570
571 \node[circle,draw] (\l_name_tl) at (#3,#4) {#2};
572 }
573 \ExplSyntaxOff

\liCpmVorgang \liCpmVORGANG\((.*)>(.*))\{(.*)\}
574 \ExplSyntaxOn
575 \NewDocumentCommand { \liCpmVorgang } { 0{ } m m m } {
576   \tl_set:Nn \l_schein_tl {}
577   \tl_set:Nn \l_kritisch_tl {}
578
579   \keys_define:nn { cpmVorgang } {
580     schein .code:n = {\tl_set:Nn \l_schein_tl {dashed}},
581     kritisch .code:n = {\tl_set:Nn \l_kritisch_tl {thick}},
582   }
583
584   \keys_set:nn { cpmVorgang } { #1 }
585
586   \path[->,\l_schein_tl,\l_kritisch_tl] (#2) edge node[auto,sloped] {#4} (#3);
587 }
588 \ExplSyntaxOff

```

## 2.9.2 Hilfsmakros für Zeitpunkt-Tabelle

```

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

```

```

\liCpmVonZu Let-Abkürzung: \let\ vz=\liCpmVonZu
\liCpmVonZu{1}(2-3):  $l_{(2 \rightarrow 3)}$ 
589 \def\liCpmVonZuOhneMathe#1(#2-#3){#1_{\scriptscriptstyle(#2\rightarrow#3)}}
590 \def\liCpmVonZu#1(#2-#3){%
591   \ifmmode%
592     \liCpmVonZuOhneMathe{#1}(#2-#3)%
593   \else%
594     $\liCpmVonZuOhneMathe{#1}(#2-#3)$%
595   \fi%
596 }

```

```

\liCpmVon Let-Abkürzung: \let\ v=\liCpmVon
\liCpmVon{1}(2):  $l_{(\rightarrow 2)}$ 
597 \def\liCpmVonOhneMathe#1(#2){#1_{\scriptscriptstyle(\rightarrow#2)}}
598 \def\liCpmVon#1(#2){%
599   \ifmmode%
600     \liCpmVonOhneMathe{#1}(#2)%
601   \else%
602     $\liCpmVonOhneMathe{#1}(#2)$%
603   \fi%
604 }

```

```

\liCpmZu Let-Abkürzung: \let\ z=\liCpmZu
\liCpmZu{1}(2):  $l_{(\leftarrow 2)}$ 
605 \def\liCpmZuOhneMathe#1(#2){#1_{\scriptscriptstyle(\leftarrow#2)}}
606 \def\liCpmZu#1(#2){%
607   \ifmmode%
608     \liCpmZuOhneMathe{#1}(#2)%
609   \else%
610     $\liCpmZuOhneMathe{#1}(#2)$%

```

```

611 \fi%
612 }

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

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

615

```



## 2.10 cyk-algorithmus.sty

```
616 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
617 \ProvidesPackage{lehramt-informatik-cyk-algorithmus}[2021/06/18 Hilfsmakros
618 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
```

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

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

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

```
620 \NewDocumentCommand{ \liWortInSprache } { m O{L(G)} } {
621   \bigskip
622   \noindent
623   $\Rrightarrow$ #1 \in #2$
624 }
```

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

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

```
625 \NewDocumentCommand{ \liWortNichtInSprache } { m O{L(G)} } {
626   \bigskip
627   \noindent
628   $\Rrightarrow$ #1 \notin #2$
629 }
```

```
630
```

## 2.11 entwurfsmuster.sty

```
631 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
632 \ProvidesPackage{lehramt-informatik-entwurfsmuster}[2021/05/06
633 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

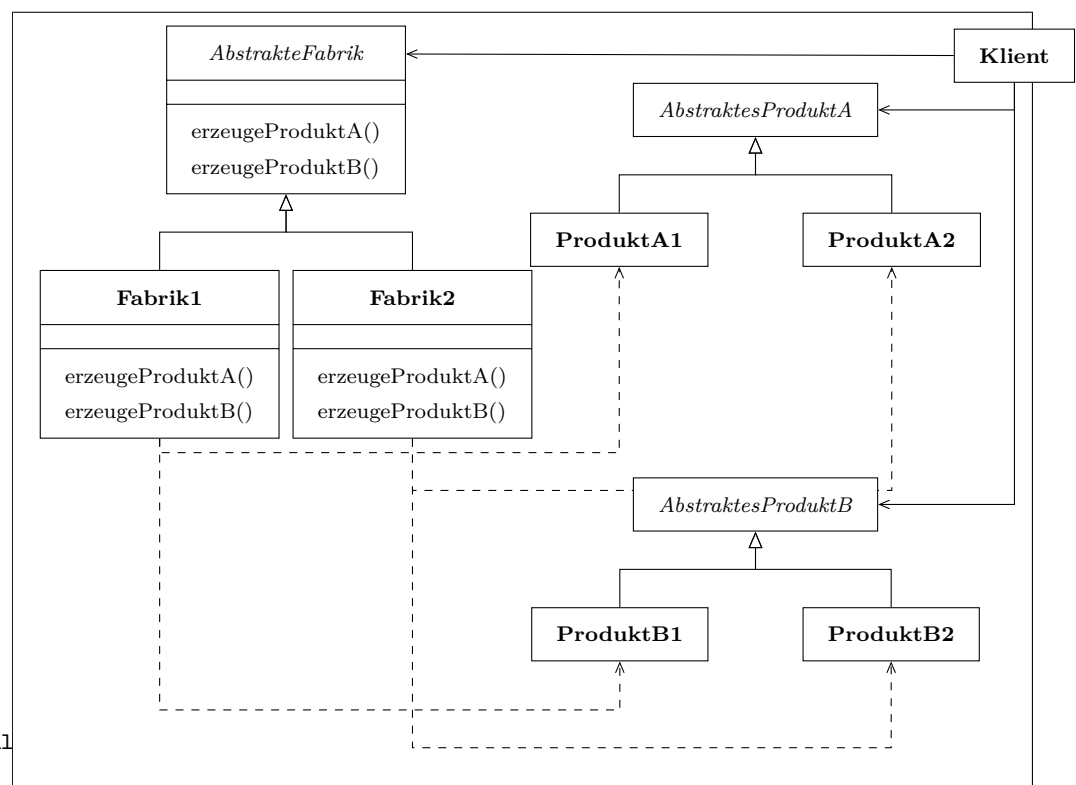
1. Uml: Uml-Klassendiagramm \liEntwurfsEinzelstueckUml
2. Akteure: Akteure, beteiligte Klassen \liEntwurfsEinzelstueckAkteure
3. Code: Allgemeines Code-Beispiel \liEntwurfsEinzelstueckCode
4. ohne: Ohne Suffix, Bündelung der einzelnen Makros eines Entwurfsmusters \liEntwurfsEinzelstueckAkteure

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

- EntwurfsCodeAllgemein Allgemeine Code-Beispiele zu den UML-Diagrammen und Stellvertretern

```
635 \RequirePackage{lehramt-informatik-syntax}
636 \def\li@EntwurfsCode#1#2{
637   \liJavaDatei{entwurfsmuster/#1/allgemein/#2}
638 }
```

### 2.11.3 Abstrakte Fabrik (Abstract Factory)



```

639 \def\liEntwurfsAbstrakteFabrikUml{
640   \begin{tikzpicture}
641     \umlclass[type=abstract]{AbstrakteFabrik}{-}{-}{
642       erzeugeProduktA()\backslash
643       erzeugeProduktB()\backslash
644     }

```

```

645 \umlclass[below left=1cm and -1.5cm of AbstrakteFabrik]{Fabrik1}{-}{
646     erzeugeProduktA()\
647     erzeugeProduktB()\
648 }
649 \umlclass[below right=1cm and -1.5cm of AbstrakteFabrik]{Fabrik2}{-}{
650     erzeugeProduktA()\
651     erzeugeProduktB()\
652 }
653 \umlVHVinherit{Fabrik1}{AbstrakteFabrik}
654 \umlVHVinherit{Fabrik2}{AbstrakteFabrik}
655
656 \umlsimpleclass[right=3cm of AbstrakteFabrik,type=abstract]{AbstraktesProduktA}
657 \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktA]{ProduktA1}
658 \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktA]{ProduktA2}
659 \umlVHVinherit{ProduktA1}{AbstraktesProduktA}
660 \umlVHVinherit{ProduktA2}{AbstraktesProduktA}
661
662 \umlsimpleclass[above right=0cm and 1cm of AbstraktesProduktA]{Klient}
663
664 \umlsimpleclass[below=4.5cm of AbstraktesProduktA,type=abstract]{AbstraktesProduktB}
665 \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktB]{ProduktB1}
666 \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktB]{ProduktB2}
667 \umlVHVinherit{ProduktB1}{AbstraktesProduktB}
668 \umlVHVinherit{ProduktB2}{AbstraktesProduktB}
669
670 \umlVHVdep[arm1=-1.3cm,arm2=-1cm]{Fabrik1}{ProduktA1}
671 \umlVHVdep[arm2=-1cm]{Fabrik1}{ProduktB1}
672
673 \umlVHVdep[arm1=-1.8cm,arm2=-1.5cm]{Fabrik2}{ProduktA2}
674 \umlVHVdep[arm2=-1.5cm]{Fabrik2}{ProduktB2}
675
676 \umluniassoc[anchor2=25]{Klient}{AbstrakteFabrik}
677 \umlVHuniassoc[arm1=-1cm]{Klient}{AbstraktesProduktA}
678 \umlVHuniassoc{Klient}{AbstraktesProduktB}
679 \end{tikzpicture}
680 }

```

iEntwurfsAbstrakteFabrikCode

```

681 \def\liEntwurfsAbstrakteFabrikCode{
682     \li@EntwurfsCode{abstrakte_fabrik}{Produkte}
683     \li@EntwurfsCode{abstrakte_fabrik}{AbstrakteFabrik}
684     \li@EntwurfsCode{abstrakte_fabrik}{Klient}
685 }

```

\liEntwurfsAbstrakteFabrik

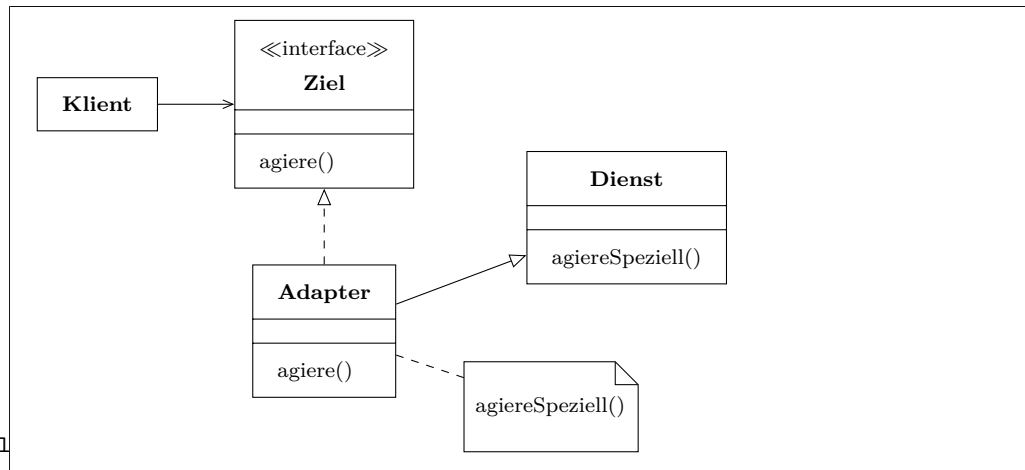
```

686 \def\liEntwurfsAbstrakteFabrik{
687     \liEntwurfsAbstrakteFabrikUml
688     \liEntwurfsAbstrakteFabrikCode
689 }

```

## 2.11.4 Adapter

\liEntwurfsAdapterUml



```

690 \def\liEntwurfsAdapterUml{
691   \begin{tikzpicture}
692     \umlsimpleclass[x=1,y=3]{Klient}{}{}
693     \umlclass[x=4,y=3,type=interface]{Ziel}{}{agiere()}
694     \umlclass[x=4,y=0]{Adapter}{}{agiere()}
695     \umlclass[x=8,y=1.5]{Dienst}{}{agiereSpeziell()}
696
697     \umlreal{Adapter}{Ziel}
698     \umluniassoc{Klient}{Ziel}
699     \umlinherit{Adapter}{Dienst}
700
701     \umlnote[x=7,y=-1,width=2cm]{Adapter}{agiereSpeziell()}
702   \end{tikzpicture}
703   \footcite[so ähnlich wie GoF]{wiki:adapter}
704 }

```

\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.

```

705 \def\liEntwurfsAdapterAkteure{
706   \begin{description}
707
708     \item[Ziel (Target)]
709
710     Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.
711
712     \item[Klient (Client)]
713
714     Der Klient nutzt Dienste über inkompatible Schnittstellen und greift
715     dabei auf adaptierte Schnittstellen zurück.
716
717     \item[Dienst (Adaptee)]
718
719     Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest
720     definierter Schnittstelle an.
721
722     \item[Adapter]
723
724     Der Adapter adaptiert die Schnittstelle des Dienstes auf die
725     Schnittstelle zum Klienten.\footcite{wiki:adapter}

```

```

726
727 \end{description}
728 }

```

\liEntwurfsAdapterCode

```

729 \def\liEntwurfsAdapterCode{
730 \li@EntwurfsCode{adapter}{Dienst}
731 \li@EntwurfsCode{adapter}{Ziel}
732 \li@EntwurfsCode{adapter}{Adapter}
733 \li@EntwurfsCode{adapter}{Klient}
734 }

```

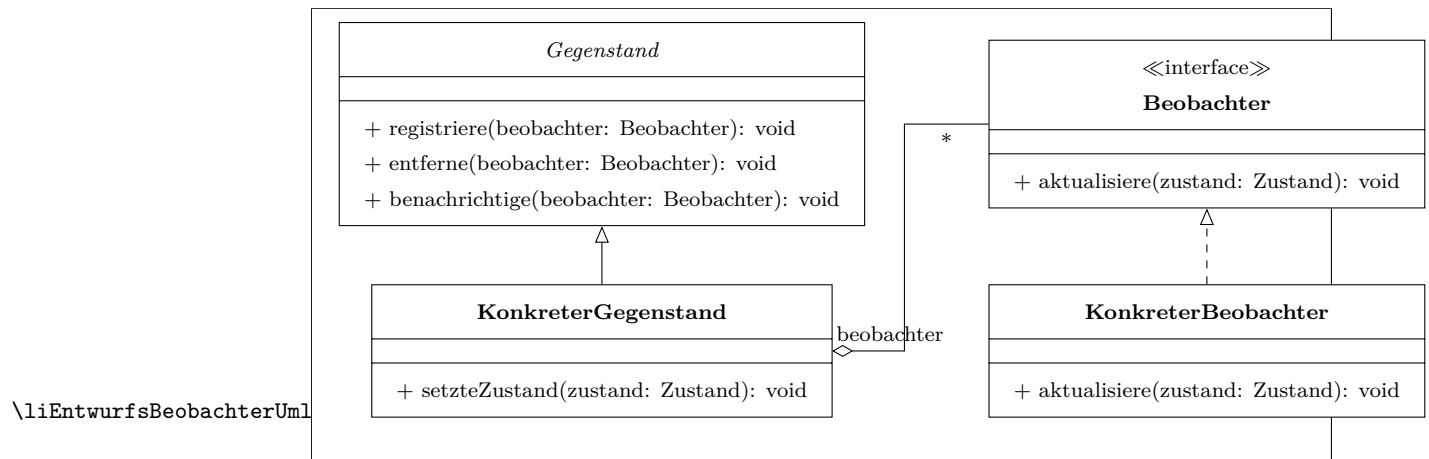
\liEntwurfsAdapter

```

735 \def\liEntwurfsAdapter{
736 \liEntwurfsAdapterUml
737 \liEntwurfsAdapterAkteure
738 \liEntwurfsAdapterCode
739 }

```

## 2.11.5 Beobachter (Observer)



```

740 \def\liEntwurfsBeobachterUml{
741 \begin{tikzpicture}
742 \umlclass[x=0,y=0,type=abstract]{Gegenstand}{-}{
743 + registriere(beobachter: Beobachter): void\\
744 + entferne(beobachter: Beobachter): void\\
745 + benachrichtige(beobachter: Beobachter): void\\
746 }
747 \umlclass[x=0,y=-3]{KonkreterGegenstand}{-}{
748 + setzteZustand(zustand: Zustand): void
749 }
750 \umlinherit{KonkreterGegenstand}{Gegenstand}
751
752 \umlclass[x=8,y=0,type=interface]{Beobachter}{-}{
753 + aktualisiere(zustand: Zustand): void
754 }
755 \umlclass[x=8,y=-3]{KonkreterBeobachter}{-}{
756 + aktualisiere(zustand: Zustand): void
757 }
758 \umlreal{KonkreterBeobachter}{Beobachter}
759
760 \umlHVHaggreg[arg1=beobachter,pos1=0.8,mult2=*,pos2=2.5]
761 {KonkreterGegenstand}{Beobachter}
762 \end{tikzpicture}
763 }

```

\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.

```
764 \def\liEntwurfsBeobachterAkteure{
765   \begin{description}
766     \item[Gegenstand / Subjekt (Subject / Observable)]
767
768     Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also
769     „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne
770     deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An-
771     und Abmeldung von Beobachtern und eine Schnittstelle zur
772     Benachrichtigung von Beobachtern über Änderungen an.\footcite[Seite
773     251]{gof}
774
775     \item[Beobachter (Observer)]
776
777     Die Beobachter (auf Englisch auch subscriber, also „Abonnent“,
778     genannt) definieren eine Aktualisierungsschnittstelle.
779
780     \item[konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)]
781
782     Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert
783     den relevanten Zustand und benachrichtigt alle Beobachter bei
784     Zustandsänderungen über deren Aktualisierungsschnittstelle. Es
785     verfügt über eine Schnittstelle zur Erfragung des aktuellen
786     Zustands.
787
788     \item[Konkrete Beobachter (ConcreteObserver)]
789
790     Konkrete Beobachter verwalten die Referenz auf ein konkretes
791     Subjekt, dessen Zustand sie beobachten und speichern und dessen
792     Zustand konsistent ist. Sie implementieren eine
793     Aktualisierungsschnittstelle unter Verwendung der
794     Abfrageschnittstelle des konkreten Subjekts.
795     \footcite{wiki:beobachter}
796   \end{description}
797 }
```

\liEntwurfsBeobachterCode

```
798 \def\liEntwurfsBeobachterCode{
799   \li@EntwurfsCode{beobachter}{Gegenstand}
800   \li@EntwurfsCode{beobachter}{KonkreterGegenstand}
801   \li@EntwurfsCode{beobachter}{Beobachter}
802   \li@EntwurfsCode{beobachter}{KonkreterBeobachterA}
803   \li@EntwurfsCode{beobachter}{KonkreterBeobachterB}
```

```

804 \li@EntwurfsCode{beobachter}{Klient}
805 }

```

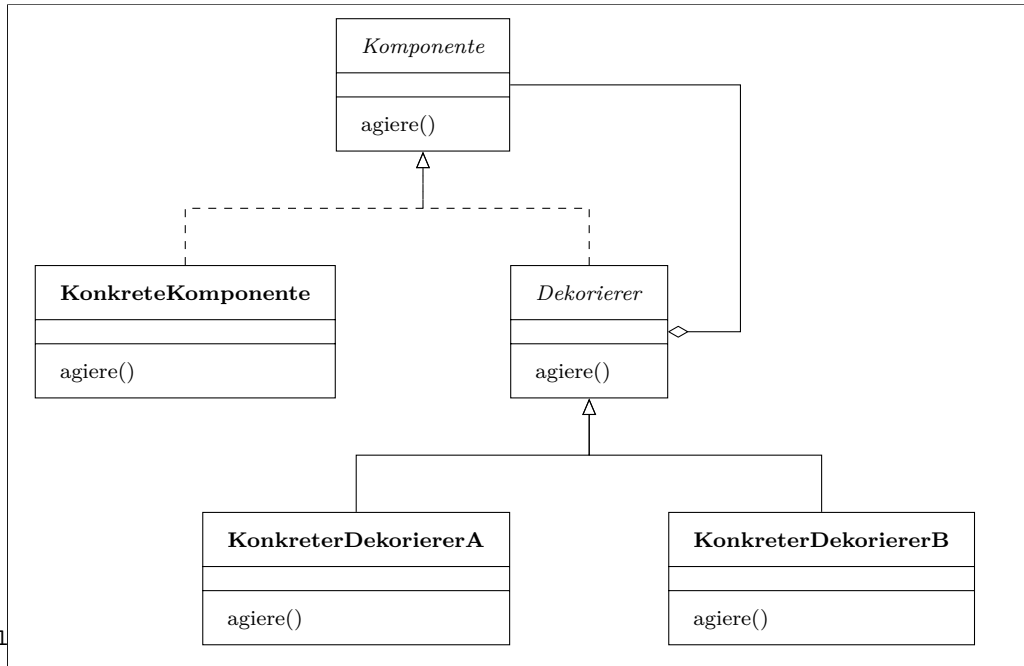
\liEntwurfsBeobachter

```

806 \def\liEntwurfsBeobachter{
807 \liEntwurfsBeobachterUml
808 \liEntwurfsBeobachterAkteure
809 \liEntwurfsBeobachterCode
810 }

```

### 2.11.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

811 \def\liEntwurfsDekoriererUml{
812 \begin{tikzpicture}
813 \umlclass[type=abstract]{Komponente}{-}{agiere()}
814 \umlclass[below left=1.5cm and 0cm of Komponente]{KonkreteKomponente}{-}{agiere()}
815 \umlclass[below right=1.5cm and 0cm of Komponente,type=abstract]{Dekorierer}{-}{agiere()}
816
817 \umlVHVreal{KonkreteKomponente}{Komponente}
818 \umlVHVreal{Dekorierer}{Komponente}
819
820 \umlclass[below left=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererA}{-}{agiere()}
821 \umlclass[below right=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererB}{-}{agiere()}
822
823 \umlVHVinherit{KonkreterDekoriererA}{Dekorierer}
824 \umlVHVinherit{KonkreterDekoriererB}{Dekorierer}
825
826 \umlHVVHaggreg[arm1=2cm]{Dekorierer}{Komponente}
827 \footcite{wiki:dekorierer}
828 \end{tikzpicture}
829 }

```

\liEntwurfsDekoriererCode

```

830 \def\liEntwurfsDekoriererCode{
831 \li@EntwurfsCode{dekorierer}{Komponente}
832 \li@EntwurfsCode{dekorierer}{KonkreteKomponente}
833 \li@EntwurfsCode{dekorierer}{Dekorierer}
834 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererA}
835 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererB}
836 \li@EntwurfsCode{dekorierer}{Klient}
837 }

```

\liEntwurfsDekorierer

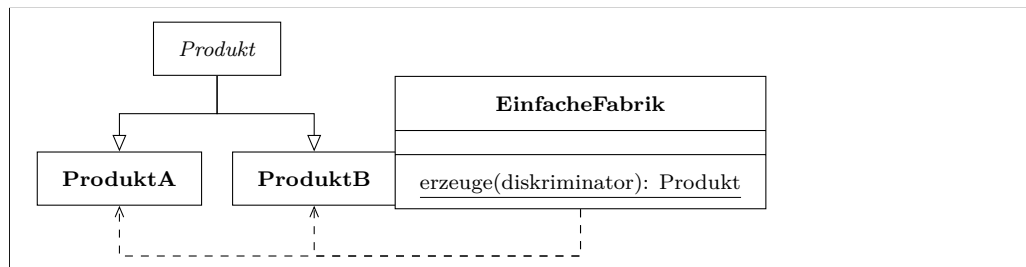
```

838 \def\liEntwurfsDekorierer{
839   \liEntwurfsDekoriererUml
840   \liEntwurfsDekoriererAkteure
841   \liEntwurfsDekoriererCode
842 }

```

### 2.11.7 Einfache Fabrik (Simple Factory)

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



```

843 \def\liEntwurfsEinfacheFabrikUml{
844   \begin{tikzpicture}
845     \umlsimpleclass[type=abstract]{Produkt}
846     \umlsimpleclass[below left=1cm and -0.65cm of Produkt]{ProduktA}
847     \umlsimpleclass[below right=1cm and -0.65cm of Produkt]{ProduktB}
848     \umlVHVinherit{Produkt}{ProduktA}
849     \umlVHVinherit{Produkt}{ProduktB}
850     \umlclass[below right=0cm and 1.5cm of Produkt]{EinfacheFabrik}{
851     }{
852       \umlstatic{erzeuge(diskriminator): Produkt}\\
853     }
854     \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktA}
855     \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktB}
856   \end{tikzpicture}
857 }

```

ntwurfsEinfacheFabrikAkteure Quelle: frei überstetzt aus GoF

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

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

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

```

858 \def\liEntwurfsEinfacheFabrikAkteure{
859   \begin{description}
860     \item[EinfacheFabrik]
861
862     Eine Klasse mit einer Erzeugungsmethode, die über eine größere
863     Bedingung verschiedene Objekt instanziiert.
864
865     \item[Produkt]
866
867     Eine abstrakte Klasse, die von den konkreten Produkten geerbt wird.
868
869     \item[KonkretesProdukt]
870
871     Ein konkretes Produkt, das von der einfachen Fabrik erzeugt wird.
872   \end{description}
873 }

```

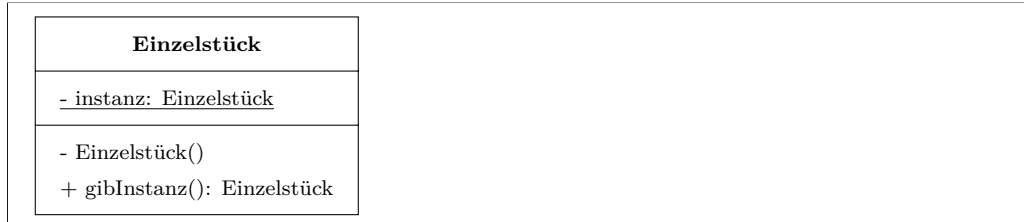


\liEntwurfsEinfacheFabrik

```
874 \def\liEntwurfsEinfacheFabrik{
875   \liEntwurfsEinfacheFabrikUml
876   \liEntwurfsEinfacheFabrikAkteure
877 }
```

### 2.11.8 Einzelstück (Singleton)

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```
878 \def\liEntwurfsEinzelstueckUml{
879   \begin{tikzpicture}
880     \umlclass{Einzelstück}{
881       \umlstatic{- instanz: Einzelstück}\\
882     }{
883       - Einzelstück()\\
884       + gibInstanz(): Einzelstück
885     }
886   \end{tikzpicture}
887 }
```

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.

```
888 \def\liEntwurfsEinzelstueckAkteure{
889   \begin{description}
890     \item[Einzelstück (Singleton)]
891
892     stellt eine statische Methode bereit, mit deren Hilfe die Klienten
893     nur auf eine einzige Instanz der Klasse zugreifen können.
894   \end{description}
895 }
```

\liEntwurfsEinzelstueckCode

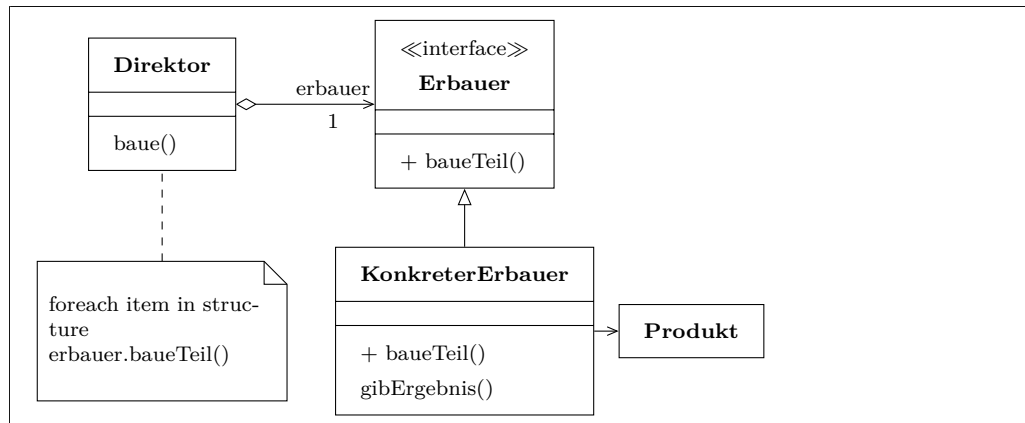
```
896 \def\liEntwurfsEinzelstueckCode{
897   \li@EntwurfsCode{einzelstueck}{Einzelstueck}
898 }
```

\liEntwurfsEinzelstueck

```
899 \def\liEntwurfsEinzelstueck{
900   \liEntwurfsEinzelstueckUml
901   \liEntwurfsEinzelstueckAkteure
902   \liEntwurfsEinzelstueckCode
903 }
```

### 2.11.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

904 \def\liEntwurfsErbauerUml{
905   \begin{tikzpicture}
906     \umlsimpleclass[x=7,y=0]{Produkt}{}{}
907     \umlclass[x=4,y=3,type=interface]{Erbauer}{}{+ baueTeil()}
908     \umlclass[x=4,y=0]{KonkreterErbauer}{}{
909       + baueTeil()\
910       gibErgebnis()}
911     \umlclass[x=0,y=3]{Direktor}{}{baue()}
912
913     \umluniagg[arg2=erbauer,pos2=0.7,mult2=1]{Direktor}{Erbauer}
914     \umluniassoc{KonkreterErbauer}{Produkt}
915     \umlinherit{KonkreterErbauer}{Erbauer}
916
917     \umlnote[x=0,y=0,width=3cm]{Direktor}{
918       foreach item in structure\
919       erbauer.baueTeil()
920   }
921   \end{tikzpicture}
922   \footcite{wiki:erbauer}
923 }

```

\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.

```

924 \def\liEntwurfsErbauerAkteure{
925   \begin{description}
926     \item[Erbauer]
927
928     Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der
929     Teile eines komplexen Objektes.
930
931     \item[KonkreterErbauer]
932
933     Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch
934     Implementierung der Schnittstelle. Außerdem definiert und verwaltet er

```

```

935     die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine
936     Schnittstelle zum Auslesen des Produkts.
937
938     \item[Direktor]
939
940     Der Direktor konstruiert ein komplexes Objekt unter Verwendung der
941     Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer
942     zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder
943     benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom
944     Klienten.
945
946     \item[Produkt]
947
948     Das Produkt repräsentiert das zu konstruierende komplexe Objekt.
949     \footcite{wiki:erbauer}
950 \end{description}
951 }

```

\liEntwurfsErbauer

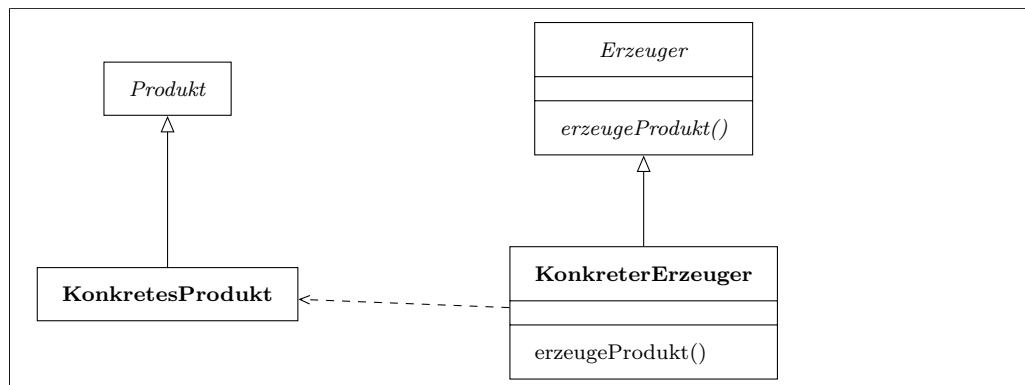
```

952 \def\liEntwurfsErbauer{
953   \liEntwurfsErbauerUml
954   \liEntwurfsErbauerAkteure
955 }

```

## 2.11.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

956 \def\liEntwurfsFabrikmethodeUml{
957   \begin{tikzpicture}
958     \umlsimpleclass[type=abstract]{Produkt}
959     \umlsimpleclass[below=2cm of Produkt]{KonkretesProdukt}
960     \umlinherit{KonkretesProdukt}{Produkt}
961
962     \umlclass[type=abstract,right=4cm of Produkt]{Erzeuger}{\{
963       \textit{erzeugeProdukt()}\}
964     }
965     \umlclass[below=1.2cm of Erzeuger]{KonkreterErzeuger}{\{
966       erzeugeProdukt()
967     }
968     \umlinherit{KonkreterErzeuger}{Erzeuger}
969
970     \umldep{KonkreterErzeuger}{KonkretesProdukt}
971   \end{tikzpicture}
972 }

```

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).

```

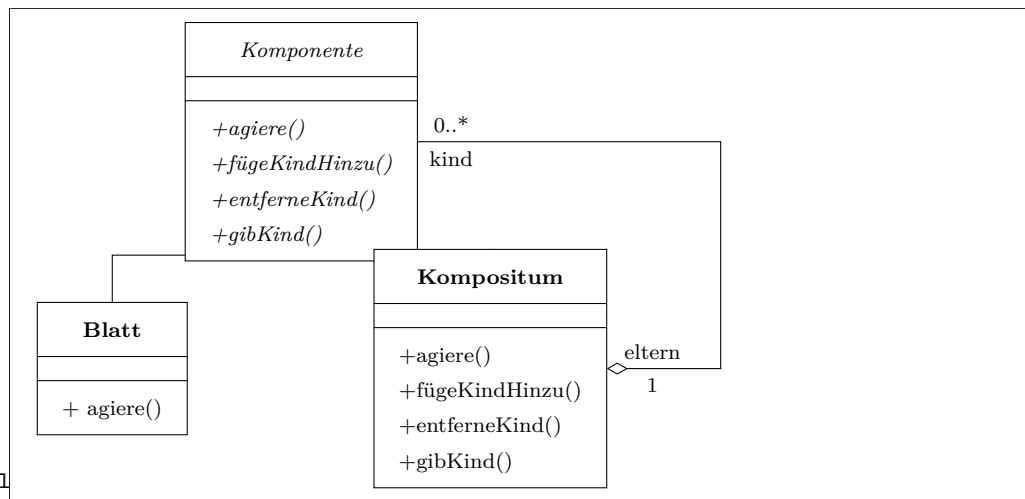
973 \def\liEntwurfsFabrikmethodeAkteure{
974   \begin{description}
975     \item[Produkt]
976
977     Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das
978     zu erzeugende Produkt.
979
980     \item[KonkretesProdukt]
981
982     KonkretesProdukt implementiert die Produkt-Schnittstelle.
983
984     \item[Erzeuger]
985
986     Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt
987     zu erzeugen und kann eine Default-Implementierung beinhalten.
988
989     \item[KonkreterErzeuger]
990
991     KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm
992     entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den
993     Konstruktor einer konkreten Produkt-Klasse aufruft).
994
995     \footcite{wiki:fabrikmethode}
996   \end{description}
997 }
```

\liEntwurfsFabrikmethode

```

998 \def\liEntwurfsFabrikmethode{
999   \liEntwurfsFabrikmethodeUml
1000   \liEntwurfsFabrikmethodeAkteure
1001 }
```

### 2.11.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

1002 \def\liEntwurfsKompositumUml{
1003   \begin{tikzpicture}
1004     \umlclass[x=2.5,y=3,type=abstract]{Komponente}{-}{-}{
1005       \textit{+agiere()}\
1006       \textit{+fügeKindHinzu()}\
1007       \textit{+entferneKind()}\
1008       \textit{+gibKind()}
1009     }
1010     \umlclass[x=0]{Blatt}{-}{+ agiere()}
1011     \umlclass[x=5]{Kompositum}{-}{-}{
1012       +agiere()\
1013       +fügeKindHinzu()\
1014       +entferneKind()\
1015       +gibKind()
1016     }
1017
1018     \umlVHVinherit{Kompositum}{Komponente}
1019     \umlVHVinherit{Blatt}{Komponente}
1020     \umlHVHaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,]
1021   \end{tikzpicture}
1022 }

```

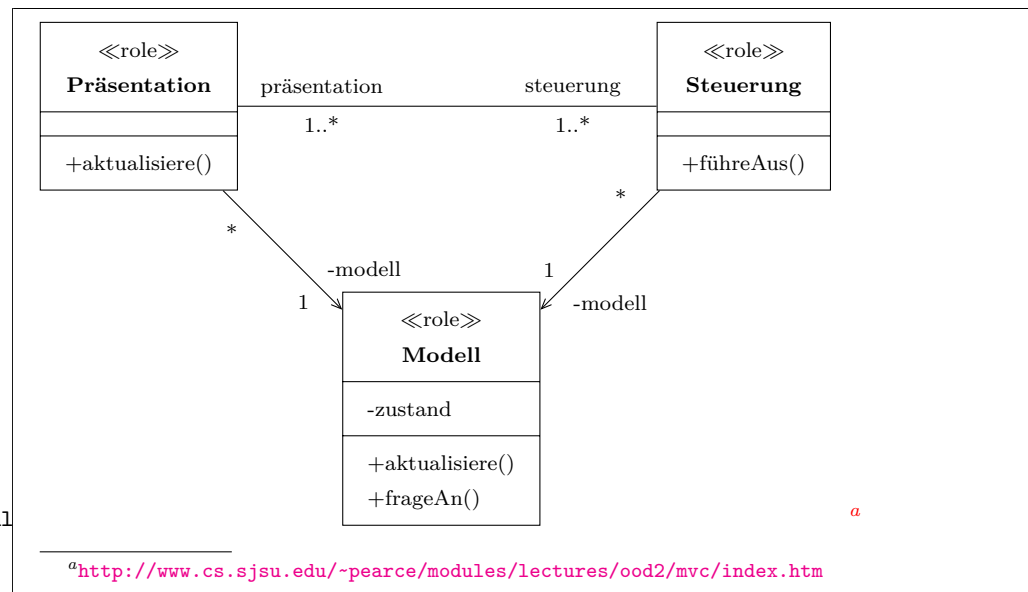
\liEntwurfsFabrikmethode

```

1023 \def\liEntwurfsKompositum{
1024   \liEntwurfsKompositumUml
1025   \liEntwurfsKompositumAkteure
1026 }

```

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



```

1027 \def\liEntwurfsModellPraesentationSteuerungUml{
1028   \begin{tikzpicture}
1029     \umlclass[x=-4cm,type=role]{Präsentation}{-}{+aktualisiere()}
1030     \umlclass[x=4cm,type=role]{Steuerung}{-}{+führeAus()}
1031     \umlclass[y=-4cm,type=role]{Modell}{-}{-}{
1032       -zustand
1033     }{
1034       +aktualisiere()\
1035       +frageAn()
1036     }
1037
1038     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
1039     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}

```

```

1040 \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
1041 \end{tikzpicture}
1042 \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
1043 }

```

ModellPraesentationSteuerung

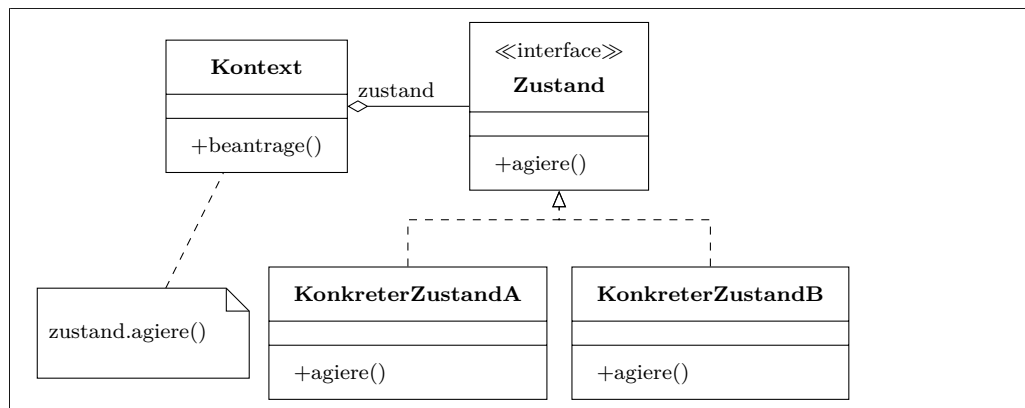
```

1044 \def\liEntwurfs{
1045 \liEntwurfsModellPraesentationSteuerungUml
1046 \liEntwurfsModellPraesentationSteuerungAkteure
1047 }

```

### 2.11.13 Zustand (State)

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



```

1048 \def\liEntwurfsZustandUml{
1049 \begin{tikzpicture}
1050 \umlcclass[x=-1,y=3]{Kontext}{+beantrage()}
1051 \umlcclass[x=3,y=3,type=interface]{Zustand}{+agiere()}
1052 \umlcclass[x=1,y=0]{KonkreterZustandA}{+agiere()}
1053 \umlcclass[x=5,y=0]{KonkreterZustandB}{+agiere()}
1054
1055 \umlVHVreal{KonkreterZustandA}{Zustand}
1056 \umlVHVreal{KonkreterZustandB}{Zustand}
1057
1058 \umlagg[reg[arg=zustand,pos=0.4]{Kontext}{Zustand}
1059
1060 \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
1061 \end{tikzpicture}
1062 }

```

\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.

```

1063 \def\liEntwurfsZustandAkteure{
1064 \begin{description}
1065 \item[Kontext (Context)]
1066
1067 definiert die clientseitige Schnittstelle und verwaltet die separaten
1068 Zustandsklassen.
1069

```

```

1070 \item[State (Zustand)]
1071
1072 definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
1073 implementiert gegebenenfalls ein Standardverhalten.
1074
1075 \item[KontreterZustand (ConcreteState)]
1076
1077 implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
1078 verbunden ist.
1079 \end{description}
1080 }

```

\liEntwurfsZustand

```

1081 \def\liEntwurfsZustand{
1082 \liEntwurfsZustandUml
1083 \liEntwurfsZustandAkteure
1084 }
1085

```

## 2.12 er.sty

```
1086 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1087 \ProvidesPackage{lehramt-informatik-er}[2020/06/13 Zum Setzen von
1088 ER-Diagrammen]
```

```
1089 \RequirePackage{tikz-er2}
1090 \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);
```

```
1091 \RequirePackage{soul}
```

```
1092 \RequirePackage{fontawesome}
```

### 2.12.3 Makro-Kürzel

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

```
1093 \ExplSyntaxOn
```

```
\liErEntity
```

```
1094 \def\liErEntity#1{\textbf{#1}}
```

```
\liErRelationship
```

```
1095 \def\liErRelationship#1{\ul{#1}}
```

```
\liErAttribute
```

```
1096 \def\liErAttribute#1{\emph{#1}}
```

```
\liErMpEntity
```

```
mp = marginpar
Let-Abkürzung: \let\e=\liErMpEntity
```

```
1097 \def\liErMpEntity#1{
1098   \liErEntity{#1}
1099   \marginpar{
1100     \liErEntity{\tiny\faSquareO{~E:~#1}
1101   }
1102 }
```

□

```
\liErMpRelationship
```

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

```
1103 \def\liErMpRelationship#1{
1104   \liErRelationship{#1}
1105   \marginpar{
1106     \liErRelationship{\tiny\faGg{~R:~#1}
1107   }
1108 }
```

```
\liErMpAttribute
```

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

```
1109 \def\liErMpAttribute#1{
1110   \liErAttribute{#1}
1111   \marginpar{
1112     \liErAttribute{\tiny\faCircleThin{~A:~#1}
1113   }
1114 }
```

```

\liErDatenbankName Let-Abkürzung: \let\d=\liErDatenbankName
                    datenbank name
1115 \def\liErDatenbankName#1{
1116   {
1117     \footnotesize\texttt{(#1)}
1118   }
1119 }

1120 \ExplSyntaxOff
1121

```

## 2.13 formale-sprachen.sty

```

1122 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1123 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
1124 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]

1125 \directlua{
1126   formale_sprachen = require('lehramt-informatik-formale-sprachen')
1127 }

1128 \RequirePackage{hyperref}

1129 \liLadePakete{mathe,typographie}

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

1130 \def\liMengeOhneMathe#1{\{ #1 \}}
1131 \def\liMenge#1{%
1132   \ifmode%
1133     \liMengeOhneMathe{#1}%
1134   \else%
1135     $\liMengeOhneMathe{#1}$%
1136   \fi%
1137 }

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

1138 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

1139 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1140 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1141 \def\liPotenzmenge#1{$\liPotenzmengeOhneMathe{#1}$}

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

1142 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
1143 \let\liZustandsmenge=\liPotenzmenge

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

1144 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1145 \def\liUeberfuehrungsFunktion#1{
1146   \ifmode
1147     \liUeberfuehrungsFunktionOhneMathe{#1}
1148   \else
1149     $\liUeberfuehrungsFunktionOhneMathe{#1}$
1150   \fi
1151 }

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

1152 \def\liAlphabet#1{$\Sigma = \{ #1 \}$}

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

1153 \def\liBandAlphabet#1{$\Gamma = \Sigma \cup \{ #1 \}$}

\liZustandsBuchstabe

1154 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1155 \def\liZustandsBuchstabeGross{Z}

```

\liZustandsmengeNr

```

1156 \def\zustandsnamens@liste#1#2{
1157   $
1158   \{
1159     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1160   \}
1161   $
1162 }
1163 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

```

\liZustandsmengeNrGross

```

1164 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

```

\liZustandsname

```

\liZustandsname{1}: $z_1$
1165 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

```

\liZustandsnameGross

```

\liZustandsnameGross{1}: $Z_1$
1166 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

```

\liAbleitung

```

\liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1167 \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}

1168 \NewDocumentEnvironment { liProduktionsRegeln }
1169 { 0{P} +b }
1170 {
1171   \liGeschweifteKlammern{#1}
1172   {
1173     \begin{align*}
1174       \directlua{formale_sprachen.produktions_regeln('#2')}
1175     \end{align*}
1176   }{-0.2cm}{-1.5cm}
1177 } {}

```

\liProduktionen

```

\liProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
1178 \def\liProduktionen#1{
1179   \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
1180 }

```

\liZustandsnameTiefgestellt

Automatisch tiefgestellte Nummerierung \z1  
**Let-Abkürzung:** \let\z=\liZustandsnameTiefgestellt

```

1181 \def\liZustandsnameTiefgestellt#1{
1182   \ifmmode
1183     \liZustandsBuchstabe\sb{#1}
1184   \else
1185     $\liZustandsBuchstabe\sb{#1}$
1186   \fi
1187 }

```

```

1188 \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}

```

```

1189 \NewDocumentCommand{ \liAusdruck } { O{L} m m } {
1190   $
1191   \tl_if_empty:nTF {#1} {} {#1 =}
1192   \{
1193     \, #2 \,
1194     |
1195     \, #3 \,
1196   \}$
1197 }
1198 \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

1199 \def\liFlaci#1{%
1200   \par
1201   {%
1202     \scriptsize
1203     Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1204     Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1205     Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1206     \href{https://flaci.com/#1}{flaci.com/#1}%
1207   }%
1208   \par
1209 }

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

1210 \ExplSyntaxOn
1211 \NewDocumentCommand {\liGrammatik} { O{G} m } {
1212   \tl_set:Nn \l_variablen_tl {V}
1213   \tl_set:Nn \l_alphabet_tl {\Sigma}
1214   \tl_set:Nn \l_produktionen_tl {P}
1215   \tl_set:Nn \l_start_tl {S}
1216
1217   \keys_define:nn { grammatik } {
1218     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1219     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1220     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1221     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1222   }
1223
1224   \keys_set:nn { grammatik } { #2 }
1225
1226   $#1 = (
1227     \l_variablen_tl,
1228     \l_alphabet_tl,
1229     \l_produktionen_tl,
1230     \l_start_tl
1231   )$
1232 }
1233 \ExplSyntaxOff

1234

```

## 2.14 formatierung.sty

```
1235 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1236 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

### 2.14.1 Schriftarten / Typographie

```
1237 \RequirePackage{mathpazo}
1238 \RequirePackage[no-math]{fontspec}
1239 \setmainfont{texgyrepagella}
1240 \setsansfont{QTAncientOlive}
1241 \RequirePackage{sectsty}
1242 \allsectionsfont{\fontspec{QTAncientOlive}}
```

### 2.14.2 Farben

```
1243 \RequirePackage{xcolor}
1244 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

### 2.14.3 Überschriften

```
1245 \RequirePackage{titlesec}
1246 \titleformat{\chapter}[display]{\sffamily\bfseries}{\Opt}{\LARGE}
1247 \titlespacing{\chapter}{0pt}{0pt}{*1}
1248 \titleformat{\paragraph}[hang]{\sffamily\normalsize\bfseries}{\theparagraph}{1em}{}
1249 \setcounter{secnumdepth}{0}
```

### 2.14.4 Listen

```
1250 \RequirePackage{paralist}
1251 \renewcommand\labelitemi{-}
1252 \renewcommand\labelitemii{-}
1253 \renewcommand\labelitemiii{-}
1254 \renewcommand\labelitemiv{-}
1255 % Counter: enumi enumii enumiii enumiv
1256 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1257 \renewcommand{\labelenumi}{(\alph{enumi})}
1258 \renewcommand{\labelenumii}{(\roman{enumii})}
```

### 2.14.5 Kasten

```
1259 \RequirePackage{mdframed}
1260 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1261 \NewDocumentEnvironment { liKasten }{ } {
1262   \begin{mdframed}[backgroundcolor=white!0]
1263 } {
1264   \end{mdframed}
1265 }
```

### 2.14.6 Header

```
1266 \RequirePackage{fancyhdr}
1267 \fancyhead[L,C,R]{}
1268 \fancyfoot[L]{}
1269 \fancyfoot[C]{}
1270 \fancyfoot[R]{\thepage}
1271 \pagestyle{fancy}
1272 \renewcommand{\headrulewidth}{0pt}
1273 \renewcommand{\footrulewidth}{0pt}
1274
```

## 2.15 gantt.sty

```

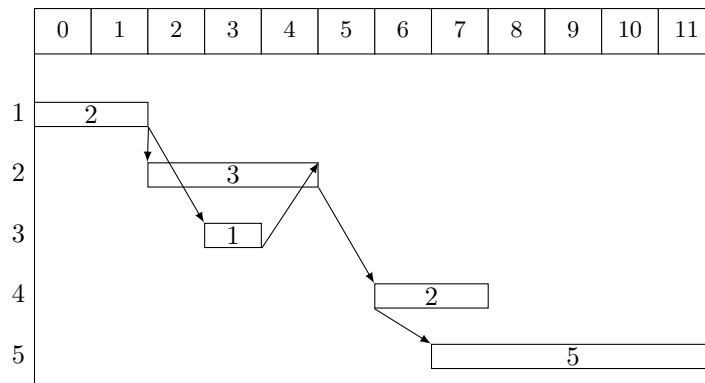
1275 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1276 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\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}

```



```

1277 \RequirePackage{tikz-uml}
1278 \RequirePackage{pgfgantt}
1279 \setganttlinklabel{f-s}{}
1280 \setganttlinklabel{s-s}{}
1281 \setganttlinklabel{f-f}{}
1282 \setganttlinklabel{s-f}{}
1283

```

## 2.16 grafik.sty

```
1284 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1285 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1286 TikZ, um Grafiken zeichnen zu können.]
1287 \RequirePackage{tikz}
1288
```



## 2.17 graph.sty

```

1289 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1290 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1291 \RequirePackage{tikz}

```

Für die die Adjazenzliste (`xrightarrow`)

```

1292 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

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

```

1293 \RequirePackage{blkarray}
1294 \usetikzlibrary{arrows.meta}

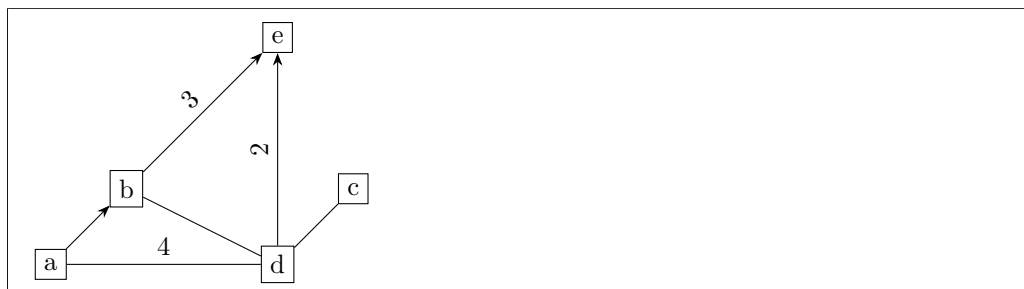
```

```

\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}

```



```

1295 \tikzset{
1296   li graph/.style={
1297     every node/.style={
1298       rectangle,
1299       draw,
1300     },
1301     every edge/.style={
1302       >={Stealth[black]},
1303       draw,
1304     },
1305     every edge/.append style={
1306       every node/.style={
1307         sloped,
1308         auto,
1309       }
1310     }
1311   },
1312   li markierung/.style={
1313     ultra thick,
1314   }
1315 }

```

**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}

```

```

1316 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1317

```

## 2.18 hanoi.sty

```
1318 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1319 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
1320 von Hanoi-Grafiken]
```

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

```
1321 \RequirePackage{tikz}
1322 \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}

1323 \def\li@mset #1[#2]=#3{%
1324 \expandafter\xdef\csname #1#2\endcsname{#3}
1325 }
1326 \def\li@mget #1[#2]{%
1327 \csname #1#2\endcsname
1328 }
1329 \def\li@minc #1[#2] += #3{%
1330 \pgfmathparse{\li@mget #1[#2] + #3}%
1331 \li@mset #1[#2] = \pgfmathresult
1332 }
1333
1334 \def\liHanoi#1#2{
1335   \edef\li@numdiscs{#1}
1336   \def\li@sequence{#2}
1337   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1338     % init colors
1339     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purple!80!black}{
1340       \li@mset col[\j]=\c;
1341     }
1342     % draw poles and init pole counters
1343     \foreach \j in {1,2,3}{
1344       \li@mset pos[\j]=0
1345       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1346     }
1347     % draw base
1348     \draw (.5,-.5) -- +(3,0);
1349     % draw discs
1350     \foreach[count=\k] \i/\j in \li@sequence{
1351       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*\i/\li@numdiscs,0);
1352       \li@minc pos[\j] += {.5}
1353     }
1354   \end{tikzpicture}
1355 }
```

## 2.19 klassen-konfiguration-aufgabe.sty

```
1356 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1357 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-aufgabe}[2021/09/01 Die
1358 Klasse Aufgabe konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]

    Formatierung muss vor literatur sein, sonst Option clash
1359 \liLadePakete{
1360   formatierung,
1361   abmessung,
1362   literatur-dummy,
1363   makros,
1364   aufgaben-metadaten,
1365   kopf-fusszeilen,
1366   mathe
1367 }
1368
1369 % Komischer Option-Clash deshalb ganz am Ende, für die Silbentrennung
1370 \RequirePackage[ngerman]{babel}
1371 \ExplSyntaxOn
1372 \AddToHook{enddocument}{
1373   \_gib_github_url:
1374 }
1375 \ExplSyntaxOff
1376
```

## 2.20 klassen-konfiguration-examen.sty

```
1377 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1378 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-examen}[2021/09/04 Die
1379 Klasse liSetzeExamen konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

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

```
1380 \liLadePakete{
1381   formatierung,
1382   literatur-dummy,
1383   makros,
1384   aufgaben-metadaten,
1385   abmessung
1386 }
1387 \liLadeAllePakete
```

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

```
1388 \RequirePackage[ngerman]{babel}
1389 \RequirePackage{standalone}
1390 \ExplSyntaxOn
```

\liSetzeExamen

```
1391 \def\liSetzeExamen#1#2#3{
1392   \tl_gset:Nn \g_auf_examen_nummer_tl { #1 }
1393   \tl_gset:Nn \g_auf_examen_jahr_tl { #2 }
1394   \tl_gset:Nn \g_auf_examen_monat_tl { #3 }
1395 }
```

\liSetzeExamenThemaNr

```
1396 \def\liSetzeExamenThemaNr#1{
1397   \tl_gset:Nn \g_auf_examen_thema_nr_tl { #1 }
1398   \section{Thema-Nr.~#1}
1399 }
```

\liSetzeExamenTeilaufgabeNr

```
1400 \def\liSetzeExamenTeilaufgabeNr#1{
1401   \tl_gset:Nn \g_auf_examen_teilaufgabe_nr_tl { #1 }
1402   \subsection{Teilaufgabe-Nr.~#1}
1403 }
```

\liBindeAufgabeEin

```
1404 \def\liBindeAufgabeEin#1{
1405   \tl_gset:Nn \g_auf_examen_aufgabe_nr_tl { #1 }
1406   \input{
1407     \LehramtInformatikRepository /
1408     Staatsexamen /
1409     \g_auf_examen_nummer_tl /
1410     \g_auf_examen_jahr_tl /
1411     \g_auf_examen_monat_tl /
1412     \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
1413       Thema - \g_auf_examen_thema_nr_tl /
1414     }
1415     \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
1416       Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl /
1417     }
1418     Aufgabe - \g_auf_examen_aufgabe_nr_tl .tex
1419   }
1420 }
```

```
1421 \ExplSyntaxOff
```

```
1422
```

## 2.21 komplexitaetstheorie.sty

```
1423 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1424 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1425 Setzen von Karp's NP-vollständigen Problemen, Hilfsmakros für die
1426 Polynomialzeitreduktion.]
```

### 2.21.1 Makro-Kürzel

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

```
1427 \liLadePakete{mathe}
```

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

```
1428 \RequirePackage{mdframed}
```

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

```
1429 \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`

```
1430 \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`

```
1431 \def\liProblemBeschreibung#1#2#3{
1432   \begin{mdframed}[
1433     userdefinedwidth=9cm,
1434     align=center,
1435     backgroundcolor=white!0,
1436   ]
1437   \centerline{\large\liProblemName{#1}}
1438
1439   \medskip
1440
1441   \begin{description}
1442     \item[Gegeben:] #2
1443     \item[Frage:] #3
1444   \end{description}
1445 \end{mdframed}
1446 }
```

```

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

1447 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1448 \begin{displaymath}
1449 \liProblemName{#1}
1450 \preceq_{#2}
1451 \liProblemName{#3}
1452 \end{displaymath}
1453 }

\liProblemVertexCover

1454 \def\liProblemClique{%
1455 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1456 Mindestgröße  $n$  in einem gegebenen Graphen.
1457 \footcite[Seite 76]{theo:fs:4}
1458 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1459 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1460 \footcite{wiki:cliquenproblem}
1461 }

\liProblemVertexCover

1462 \def\liProblemVertexCover{%
1463 %
1464 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1465 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1466 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1467 \footcite{wiki:knotenueberdeckung}
1468
1469 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1470 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1471 mindestens einem Knoten aus  $U$  verbunden ist.
1472 \footcite[Seite 78]{theo:fs:4}%
1473 }

\liProblemSubsetSum Kein Karp-21-Problem

1474 \def\liProblemSubsetSum{%
1475 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1476 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1477 \footcite{wiki:teilsommenproblem}
1478 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1479 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1480 größer als eine gegebene obere Schranke  $c$  ist.
1481 \footcite[Seite 74]{theo:fs:4}
1482 }

\liProblemSubsetSum Kein Karp-21-Problem

1483 \def\liProblemSat{%
1484 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1485 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1486 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1487 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1488 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1489 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1490 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1491 aufgestellt werden.
1492 \footcite[Seite 71]{theo:fs:4}
1493 }

1494

```

## 2.22 kontrollflussgraph.sty

1495 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1496 \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 = ...}]
```



```

1497 \RequirePackage{tikz}
1498 \usetikzlibrary{positioning}
1499 \tikzset{
1500   li kontrollfluss/.style={
1501     knoten/.style={
1502       circle,
1503       draw
1504     },
1505     usebox/.style={
1506       draw,
1507       rectangle,
1508       font=\scriptsize,
1509       anchor=west,
1510       align=left,
1511     },
1512     bedingung/.style={
1513       midway,
1514       draw=none,
1515       font=\scriptsize
1516     },
1517     knotenbeschriftung/.style={
1518       draw,
1519       rectangle,
1520       midway,
1521       font=\scriptsize
1522     },
1523     wahr/.style={
1524       thick
1525     },
1526     falsch/.style={
1527       dashed
1528     },
1529     every node/.style={
1530       circle,
1531       draw,
1532     },
1533     every edge/.append style={
1534       every node/.style={
1535         draw=none,
1536         bedingung,
1537       }
1538     },
1539     every path/.style={
1540       draw,
1541       ->,
1542     },
1543     every pin/.style={
1544       draw,
1545       dotted,
1546       rectangle,
1547       pin position=right
1548     },
1549     every pin edge/.style={
1550       dotted,
1551       arrows=-,
1552     }
1553   }
1554 }

```

## 2.22.4 Umgebungen

liKontrollflussgraph

```

1555 \NewDocumentEnvironment { liKontrollflussgraph } { 0{} } {

```

```

1556 \begin{tikzpicture}[
1557     li kontrollfluss,
1558     #1
1559 ]
1560 } {
1561 \end{tikzpicture}
1562 }

```

## 2.22.5 Makros

\liAnweisung

```
1563 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}
```

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

```
1564 \def\liBedingung#1#2{\node[bedingung,#1]{\texttt{#2}}}
```

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

```
1565 \def\liBedingungWahr#1{\node[bedingung,#1]{\texttt{[true]}}}
```

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

```
1566 \def\liBedingungFalsch#1{\node[bedingung,#1]{\texttt{[false]}}}
```

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

```
1567 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}
```

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

```
1568 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw
```

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

```
1569 \ExplSyntaxOn
```

```
1570 \NewDocumentCommand { \liKontrollKnotenPfad } { m }
```

```
1571 {
```

```
1572 \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
```

```
1573 \seq_set_map:NNn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
```

```
1574 \seq_use:Nn \l_tmpa_seq { ~~~~ }
```

```
1575 }
```

```
1576 \ExplSyntaxOff
```

```
1577
```

## 2.23 kopf-fusszeilen.sty

```
1578 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1579 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1580 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1581 \ExplSyntaxOn

1582 \fancyhead{}
1583 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1584 \fancyhead[L0,RE]{\scriptsize\today}
1585 \fancyfoot{}
1586 \fancyfoot[LE,R0]{\thepage}
1587 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1588 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1589 \renewcommand{\headrulewidth}{0.4pt}
1590 \renewcommand{\footrulewidth}{0.4pt}
1591 \setlength{\headwidth}{\textwidth}

1592 \ExplSyntaxOff

1593
```

## 2.24 literatur-dummy.sty

```
1594 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1595 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1596 \def\literatur{}

\footcite

1597 % \RequirePackage[stable,multiple]{footmisc}
1598 \NewDocumentCommand{ \footcite } { o m }{}

1599
```

## 2.25 literatur.sty

```
1600 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1601 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1602 \RequirePackage{csquotes}
1603 \RequirePackage[
1604   bibencoding=utf8,
1605   citestyle=authortitle,
1606   backend=biber,
1607 ]{biblatex}
1608 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1609 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1610 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1611 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1612 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1613 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1614 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1615 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1616 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1617 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1618 % To allow footnotes in the heading
1619 \RequirePackage[stable,multiple]{footmisc}

\literatur

1620 \def\literatur{\printbibliography[heading=subbibliography]}

1621
```

## 2.26 makros.sty

```
1622 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1623 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1624 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1625 anderen Paket passen]
1626 \RequirePackage{hyperref}
1627 \RequirePackage{graphicx}
    Für die Umgebung liQuellen benötigt.
1628 \RequirePackage{paralist}

\inhaltsverzeichnis
1629 \def\inhaltsverzeichnis {
1630   \begin{mdframed}
1631     \begin{group}
1632       \let\clearpage\relax
1633       \tableofcontents
1634     \end{group}
1635   \end{mdframed}
1636 }

\memph \mephm (\marginpar and \emph)
1637 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1638 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1639 \newcommand{\liPseudoUeberschrift}[1]{
1640   \bigskip
1641   \noindent
1642   \textsf{\textbf{#1}}
1643   \noindent
1644 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1645 \newcommand{\liBeschriftung}[1]{
1646   \par
1647   \noindent
1648   \medskip
1649   \textbf{#1}:
1650   \medskip
1651   \noindent
1652 }

\hinweis
1653 \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.
1654 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

liEinbettung
1655 \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.
1656 \RequirePackage{xparse}
1657 \ExplSyntaxOn
```

```

1658 \NewDocumentEnvironment{ liAntwort } { 0{standard} }
1659 {
1660   \str_case:nn {#1} {
1661     {standard} {
1662       \def\beschriftung{}
1663       \mdfsetup{backgroundcolor=gray!1,linecolor=gray}
1664     }
1665     {richtig} {
1666       \def\beschriftung{richtig}
1667       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1668     }
1669     {falsch} {
1670       \def\beschriftung{falsch}
1671       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1672     }
1673     {muster} {
1674       \def\beschriftung{Musterlösung}
1675       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1676     }
1677   }
1678   \ifx\beschriftung\empty\else
1679     \noindent
1680     \textbf{\beschriftung{:}}
1681   \fi
1682   \begin{mdframed}
1683 }
1684 {\end{mdframed}}

```

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

```

1685 \NewDocumentEnvironment{ liAdditum } { o }
1686 {
1687   \begin{mdframed}[backgroundcolor=yellow!5]
1688     \IfNoValueTF {#1}
1689     { \liPseudoUeberschrift{Additum} }
1690     { \liPseudoUeberschrift{Additum:~#1} }
1691   }
1692 {\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.

```

1693 \NewDocumentEnvironment{ liExkurs }{o +b}{
1694   \vspace{0.2cm}%
1695   \begin{mdframed}[
1696     backgroundcolor=white,
1697     bottomline=false,
1698     innermargin=1cm,
1699     leftline=true,
1700     linecolor=black,
1701     linewidth=0.1cm,
1702     outermargin=1cm,
1703     rightline=false,
1704     topline=false,
1705   ]

```

```

1706 \footnotesize
1707 \noindent%
1708 \textbf{Exkurs:~#1}\par%
1709 \noindent%
1710 #2
1711 \end{mdframed}
1712 \vspace{0.2cm}
1713 }{}

```

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

```

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

```

**Weiterführende Literatur:**

- Quelle 1
- Quelle 2

```

1714 \cs_new:Npn \listen@punkt #1 {\item #1}
1715 \NewDocumentEnvironment { liQuellen }{ +b }
1716 {
1717   \seq_clear_new:N \l_quellen
1718   \seq_set_split:Nnn \l_quellen {\item} {#1}
1719   \seq_remove_all:Nn \l_quellen {}
1720   \begin{mdframed}[backgroundcolor=white!0]
1721     \footnotesize
1722     \noindent
1723     \textsf{\textbf{Weiterführende~Literatur:}}
1724     \medskip
1725     \begin{compactitem}
1726       \seq_map_function:NN \l_quellen {\listen@punkt}
1727     \end{compactitem}
1728   \end{mdframed}
1729   %
1730   \makeatletter
1731   \par\@afterindentfalse\@afterheading
1732   \makeatother
1733 } {}

```

liLernkartei

```

1734 \NewDocumentEnvironment { liLernkartei }{ m +b }
1735 {
1736   \begin{mdframed}
1737     \footnotesize
1738     \noindent%
1739     \textbf{Lernkarteikarte:~#1}\par%
1740     \noindent%
1741     #2
1742   \end{mdframed}
1743 } {}

```

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

```

1744 \NewDocumentEnvironment { liDiagramm }{ m +b }
1745 {
1746   \begin{mdframed}[backgroundcolor=white!0]
1747     \small
1748     \noindent%
1749     \textit{#1}:
1750     \begin{center}

```



```

1751 #2
1752 \medskip
1753 \end{center}
1754 \end{mdframed}
1755 } {}
1756 \ExplSyntaxOff

\liFussnoteUrl \liFussnoteUrl[\langle zusätzlicher-text \rangle]{\langle url \rangle} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
1757 \NewDocumentCommand{\liFussnoteUrl} { o m } {
1758   \footnote{\url{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1759 }
1760

\liFussnoteLink \liFussnoteLink[\langle zusätzlicher-text \rangle]{\langle link-text \rangle}{\langle url \rangle} \liFussnoteLink[zusätzlicher Text]{tex
Einen Link, d. h. einen Link-Text und eine URL als Fußnote setzen.
1761 \NewDocumentCommand{\liFussnoteLink} { o m m } {
1762   \footnote{\href{#3}{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1763 }

\zB
1764 \def\zB{z.\,B. }

\ZB
1765 \def\ZB{Z.\,B. }

\dh
1766 \def\dh{d.\,h. }

1767

```

## 2.27 master-theorem.sty

1768 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1769 \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}

1770 \ExplSyntaxOn

1771 \RequirePackage{amsmath}

\liRundeKlammer

1772 \def\liRundeKlammer#1{

1773 \negthinspace \left( #1 \right)

1774 }

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

1775 \def\liThetaOhneMathe#1{

1776 \Theta \liRundeKlammer{#1}

1777 }

1778 \def\liTheta#1{

1779 \ifmmode

1780 \liThetaOhneMathe{#1}

1781 \else

1782 \$\liThetaOhneMathe{#1}\$

1783 \fi

1784 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
1785 \def\liOmegaOhneMathe#1{
1786 \Omega \liRundeKlammer{#1}
1787 }
1788 \def\liOmega#1{
1789 \ifmmode
1790 \liOmegaOhneMathe{#1}
1791 \else
1792 $\liOmegaOhneMathe{#1}$
1793 \fi
1794 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
1795 \def\liOOhneMathe#1{
1796 \mathcal{O} \liRundeKlammer{#1}
1797 }
1798 \def\liO#1{
1799 \ifmmode
1800 \liOOhneMathe{#1}
1801 \else
1802 $\liOOhneMathe{#1}$
1803 \fi
1804 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
1805 \def\liTOhneMathe#1#2{
1806 \tl_if_blank:nTF {#1}
1807 {}
1808 {#1 \cdot }
1809 T
1810 \liRundeKlammer{\frac{n}{#2}}
1811 }
1812 \def\liT#1#2{
1813 \ifmmode
1814 \liTOhneMathe{#1}{#2}
1815 \else
1816 $\liTOhneMathe{#1}{#2}$
1817 \fi
1818 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
1819 \def\liRekursionsGleichung{
1820 $T(n) = \liT{a}{b} + f(n)$
1821 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
1822 \def\liBedingungEins{
1823 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
1824 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
1825 \def\liBedingungZwei{
1826 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
1827 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
1828 \def\liBedingungDrei{
1829 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
1830 }

1831 \ExplSyntaxOff

```

\liMasterVariablen

```

1832 \def\liMasterVariablen{
1833   \begin{displaymath}
1834     T(n) = \liT{a}{b} + f(n)
1835   \end{displaymath}
1836
1837   \begin{itemize}
1838     \item[$a = $]
1839       Anzahl der rekursiven Aufrufe, Anzahl der Unterprobleme in der
1840       Rekursion
1841       ($a \geq 1$).
1842
1843     \item[$\textstyle{\frac{1}{b}} = $]
1844       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
1845       repräsentiert wird, Anteil an der Verkleinerung des Problems ($b > 1$).
1846
1847     \item[$f(n) = $]
1848       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
1849       die Kombination der Teillösungen entstehen. Eine von $T(n)$
1850       unabhängige und nicht negative Funktion.
1851   \end{itemize}
1852   \footcite{wiki:master-theorem}
1853   \footcite[Seite 19-35]{aud:fs:2}
1854 }
```

\liMasterFaelle

```

1855 \def\liMasterFaelle{
1856   \begin{description}
1857     \item[1. Fall:]
1858       $T(n) \in \liTheta{n^{\log\sb{b}a}}$
1859
1860       \hfill falls \liBedingungEins
1861       für $\varepsilon > 0$
1862
1863     \item[2. Fall:]
1864       $T(n) \in \liTheta{n^{\log\sb{b}a} \cdot \log n}$
1865
1866       \hfill falls \liBedingungZwei
1867
1868     \item[3. Fall:]
1869       $T(n) \in \liTheta{f(n)}$
1870
1871       \hfill falls \liBedingungDrei
1872       für $\varepsilon > 0$
1873       und ebenfalls für ein $c$ mit $0 < c < 1$ und alle hinreichend großen $n$
1874       gilt:
1875       $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$
1876   \end{description}
1877 }
```

\liMasterVariablenDeklaration

```

1878 \def\liMasterVariablenDeklaration#1#2#3{
1879   \begin{description}
1880     \item[Allgemeine Rekursionsgleichung:] \strut
1881
1882     \liRekursionsGleichung
1883
1884     \item[Anzahl der rekursiven Aufrufe ($a$):] \strut
1885
1886     #1
1887
1888     \item[Anteil Verkleinerung des Problems ($b$):] \strut
1889   \end{description}
```

```

1890     um  $\frac{1}{b}$  also  $b = \frac{1}{a}$ 
1891
1892     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut
1893
1894      $T(n) = \frac{1}{b} + T(\frac{n}{a})$ 
1895
1896     \item[Ergibt folgende Rekursionsgleichung:] \strut
1897
1898      $T(n) = \frac{1}{b} + T(\frac{n}{a})$ 
1899 \end{description}
1900 }

\liMasterFallRechnung
1901 \def\liMasterFallRechnung#1#2#3{
1902   \begin{description}
1903     \item[1. Fall:] \liBedingungEins:
1904
1905     #1
1906
1907     \item[2. Fall:] \liBedingungZwei:
1908
1909     #2
1910
1911     \item[3. Fall:] \liBedingungDrei:
1912
1913     #3
1914   \end{description}
1915 }

\liMasterExkurs
1916 \def\liMasterExkurs{
1917   \begin{liExkurs}[Master-Theorem]
1918     \liMasterVariablen
1919
1920     \noindent
1921     Dann gilt:
1922
1923     \liMasterFaelle
1924   \end{liExkurs}
1925 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
1926 \def\liMasterWolframLink#1{
1927   Berechne die Rekursionsgleichung auf WolframAlpha:
1928   \href{https://www.wolframalpha.com/input/?i=#1}{WolframAlpha}
1929 }

1930

```

## 2.28 mathe.sty

```
1931 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1932 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
1933
1934 % for example \ltimes \rtimes
1935 %\RequirePackage{amssymb}
1936 \RequirePackage{amsmath}
1937
1938 %%
1939 % \mlq \mrq
1940 %%
1941 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
1942 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
1943
```

## 2.29 minimierung.sty

```

1944 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1945 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
1946 Minimierungsalgorithmus von einem NEA zu einem DEA]

1947 \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

1948 \def\liFussnote#1{\$x_{#1}\$}

1949 \def\li@fussnote@text#1#2{
1950 \liFussnote{#1}
1951 \quad
1952 {\footnotesize #2}
1953 }

\liFussnoteEinsText

1954 \def\liFussnoteEinsText{
1955 \li@fussnote@text{1}
1956 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
1957 }

\liFussnoteZweiText

1958 \def\liFussnoteZweiText{
1959 \li@fussnote@text{2}
1960 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
1961 }

\liFussnoteDreiText

1962 \def\liFussnoteDreiText{
1963 \li@fussnote@text{3}

```

```

1964 {In weiteren Iterationen markierte Zustände.}
1965 }

\liFussnoteVierText
1966 \def\liFussnoteVierText{
1967   \li@fussnote@text{4}
1968   {...}
1969 }

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



1970 \def\liFussnoten{
1971   \bigskip
1972
1973   \noindent
1974   \liFussnoteEinsText
1975
1976   \noindent
1977   \liFussnoteZweiText
1978
1979   \noindent
1980   \liFussnoteDreiText
1981
1982   \noindent
1983   \liFussnoteVierText
1984 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
1985 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
1986 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
1987 \def\liZustandsPaar#1#2{
1988   $(
1989     \liZustandsPaarVariablenName_#1,
1990     \liZustandsPaarVariablenName_#2
1991   )$
1992 }

liUebergangsTabelle
1993 \renewcommand{\arraystretch}{1.4}
1994 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
1995   \liPseudoUeberschrift{Übergangstabelle}
1996   \begin{center}
1997     \begin{tabular}{r|l|l}
1998       \textbf{Zustandspaar} & \textbf{#1} & \textbf{#2} \\ \hline
1999     \end{tabular}
2000   \end{center}
2001
2002 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle: Minimierungstabelle (Table filling)

2003 \ExplSyntaxOn
2004 \def\liUeberschriftDreiecksTabelle{
2005   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
2006 }

```



**\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. —————

```

2007 \def\liMinimierungErklaerung{
2008   %\footcite[Seite~19]{koenig}
2009   \liParagraphMitLinien{
2010     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
2011     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
2012     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
2013     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
2014      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
2015     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
2016     somit~zueinander~( $k - 1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
2017     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
2018     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
2019   }
2020 }
2021 \ExplSyntaxOff
2022

```

## 2.30 normalformen.sty

```
2023 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2024 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10
2025 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
2026 Attributhülle]
```

Mathe für textit tex etc, typographie für geschweifte Klammer

```
2027 \liLadePakete{mathe,typographie}
2028 \directlua{
2029   helfer = require('lehramt-informatik-helfer')
2030   normalformen = require('lehramt-informatik-normalformen')
2031 }
```

### 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
```

```
2032 \def\liTeilen#1{
2033   \directlua{tex.print(normalformen.teilen('#1'))}
2034 }
```

**\liAttributHuelle Let-Abkürzung:**  $\text{let}\ \text{ah}=\text{liAttributHuelle}$   
 $\text{ah}\{F, \text{m}\{A, B\}\}$  AttrHülle( $F, \{A, B\}$ ) Regulärer Ausdruck zum Konvertieren  
 $\text{AttrHülle}((.*)\backslash)\ \backslash\text{ah}\{\$1\}$

```
2035 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(\#1)}
2036 \def\liAttributHuelle#1{
2037   \ifmmode
2038     \liAttributHuelleOhneMathe{\#1}
2039   \else
2040     $\liAttributHuelleOhneMathe{\#1}$
2041   \fi
2042 }
```

**\liAttributMenge Let-Abkürzung:**  $\text{let}\ \text{m}=\text{liAttributMenge}$   
2043  $\backslash\text{def}\ \backslash\text{liAttributMenge}\#1\{\backslash\{\ \backslash\text{textit}\{\#1\}\ \backslash\}$

liAHuelle

```
2044 \NewDocumentEnvironment{ liAHuelle } { +b } {
2045   \begingroup
2046   \footnotesize
2047   \begin{multline*}
2048     \#1
2049   \end{multline*}
2050   \endgroup
2051 } { }
```

**\liLinksReduktion** Nur innerhalb von liAHuelle zu verwenden bzw. multline

**Let-Abkürzung:**  $\text{let}\ \text{ahL}=\text{liLinksReduktion}$   
 $\text{ahL}\{\text{ursprüngliche linke Attributmenge}\}\{\text{ohne dieses Attribut}\}\{\text{Ergebnis}\}$   
2052  $\backslash\text{def}\ \backslash\text{liLinksReduktion}\#1\#2\#3\{$   
2053  $\backslash\text{shoveleft}\{$   
2054  $\backslash\text{liAttributHuelleOhneMathe}\{\text{FA},$   
2055  $\backslash\text{liAttributMenge}\{\#1\ \backslash\text{string}\ \#2\}\} =$   
2056  $\}\ \backslash\backslash$

```

2057 \shoveright{
2058   \liAttributMenge{#3}
2059 } \\\
2060 }

```

```

\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( $F, \{A, E \setminus E\}$ ) =  $\{A, E, F, B, D\}$ 
2061 \def\liLinksReduktionInline#1#2#3{%
2062   {%
2063     \footnotesize%
2064     $\liAttributHuelleOhneMathe{F,
2065       \liAttributMenge{#1} \string\ #2}} =
2066     \liAttributMenge{#3}$
2067   }
2068 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahr=\liLinksReduktionInline
    \ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
2069 \def\liRechtsReduktionInline#1#2#3#4{%
2070   {%
2071     \footnotesize%
2072     $\liAttributHuelleOhneMathe{
2073       F \setminus
2074       \liFunktionaleAbhaengigkeit{#1}
2075       \def\tmp{#2}\ifx\tmp\empty
2076       \else
2077         \cup \liFunktionaleAbhaengigkeit{#2}
2078       \fi
2079       ,
2080       \liAttributMenge{#3}
2081     } =
2082     \liAttributMenge{#4}$
2083   }
2084 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
    \fa{A, B -> C, D}:  $\{A, B\} \rightarrow \{C, D\}$  Regulärer Ausdruck zum Konvertieren:
    \$(.*?) \\\rightarrow (.*)\$ \fa{$1 -> $2}
2085 \def\liFunktionaleAbhaengigkeit#1{%
2086   \directlua{normalformen.drucke_funk_abhaengigkeit('#1', false)}%
2087 }

```

```

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

```

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

```

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

```

```

2088 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
2089   \liGeschweifteKlammern
2090   {#1}
2091   {
2092     \begin{align*}
2093       \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
2094     \end{align*}
2095   }
2096   {-0.5cm}
2097   {-1.7cm}
2098 }

\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}

2099 \NewDocumentCommand {\liRelation} { O{R} m } {
2100   $\directlua{
2101     local name = helper.konvertiere_tiefgestellt('#1')
2102     tex.print(name)
2103   }$(\textit{\, #2\,})
2104 }

2105

```

## 2.31 petri.sty

2106 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2107 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]

### 2.31.1 Makro-Kürzel

\let\t=\liPetriTransitionsName

\let\tp=\liPetriTransPfeile

\let\k=\liPetriErreichKnotenDrei

2108 \RequirePackage{tikz}

2109 \usetikzlibrary{petri,arrows.meta}

Für die Darstellungsmatrix

2110 \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

```
2111 \def\liPetriSetzeSchluessel{%
2112   \def\TmpTransitionOne{}%
2113   \def\TmpTransitionTwo{}%
2114   \def\TmpTransitionThree{}%
2115   \def\TmpTransitionFour{}%
2116   \def\TmpTransitionFive{}%
2117   \def\TmpTransitionSix{}%
2118   \def\TmpTransitionSeven{}%
2119   \def\TmpTransitionEight{}%
2120   \def\TmpTransitionNine{}%
2121   \def\TmpTransitionTen{}%
2122   \pgfkeys{/petri/.cd,
2123     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2124     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2125     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2126     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
```

```

2127 p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
2128 p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2129 p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2130 p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2131 p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2132 p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2133 t1/.store in=\TmpTransitionOne,t1/.default=activated,
2134 t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2135 t3/.store in=\TmpTransitionThree,t3/.default=activated,
2136 t4/.store in=\TmpTransitionFour,t4/.default=activated,
2137 t5/.store in=\TmpTransitionFive,t5/.default=activated,
2138 t6/.store in=\TmpTransitionSix,t6/.default=activated,
2139 t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2140 t8/.store in=\TmpTransitionEight,t8/.default=activated,
2141 t9/.store in=\TmpTransitionNine,t9/.default=activated,
2142 t10/.store in=\TmpTransitionTen,t10/.default=activated,
2143 scale/.store in=\TmpScale,scale/.default=0.5,
2144 x/.store in=\TmpX,x/.default=5,
2145 y/.store in=\TmpY,y/.default=5,
2146 }%
2147 }

```

```

2148 \tikzset{
2149   li petri/.style={
2150     activated/.style={
2151       very thick
2152     },
2153     inhibitor/.style={
2154       {Circle[open,length=2mm,fill=white]}-
2155     }
2156   }
2157 }

```

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

```

2158 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2159 \def\liPetriTransitionsName#1{
2160   \ifmmode
2161     \liPetriTransitionsNameOhneMathe{#1}
2162   \else
2163     $\liPetriTransitionsNameOhneMathe{#1}$
2164   \fi
2165 }

```

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

```

2166 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2167   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2168 }

```

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

```

2169 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2170 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2171 }

```

## 2.32 potenzmengen-konstruktion.sty

```
2172 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2173 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2174 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2175 \liLadePakete{formale-sprachen}
2176 \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}
    }
  }
}
```

```
2177 \def\liZustandsMengenSammlung#1#2{
2178   \liZustandsnameGross{#1}
2179   {
2180     \footnotesize
2181     \liPotenzmenge{
2182       \str_case:nn {#1} {#2
2183       }
2184     }
2185 }
```

```
\liZustandsMengenSammlungNr
```

```
2186 \def\liZustandsMengenSammlungNr#1#2{
2187   \liZustandsnameGross{#1}
2188   {
```

```
2189     \footnotesize
2190     \liZustandsmengeNr{
2191         \str_case:nn {#1} #2
2192     }
2193 }
2194 }

2195 \ExplSyntaxOff
2196
```



## 2.33 pseudo.sty

2197 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2198 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen  
 2199 von Pseudo-Code, Hüll-Paket um algorithm2e]

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

Algorithmus 1: Minimaler Spannbaum nach Kruskal
---

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

2200 \RequirePackage[german,boxruled]{algorithm2e}

2201

## 2.34 pumping-lemma.sty

2202 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2203 \ProvidesPackage{lehramt-informatik-pumping-lemma}[2021/08/11 Enthält die  
 2204 Definitionen für das Pumping-Lemma in der Regulären Sprache und  
 2205 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2206 \def\liPumpingRegulaer{%
2207   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2208   alle Wörter  $\omega \in L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2209    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2210   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2211
2212   \begin{enumerate}
2213     \item  $|v| \geq 1$ 
2214     (Das Wort  $v$  ist nicht leer.)
2215
2216     \item  $|uv| \leq j$ 
2217     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2218
2219     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w \in L$ 
2220     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2221     Sprache  $L$ )
2222   \end{enumerate}
2223
2224   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2225   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2226 }
```

\liPumpingKontextfrei

```
2227 \def\liPumpingKontextfrei{%
2228   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2229   sich alle Wörter  $\omega \in L$  mit  $|\omega| \geq j$  zerlegen lassen in
2230    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2231
2232   \begin{enumerate}
2233     \item  $|vx| \geq 1$ 
2234     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2235
2236     \item  $|vwx| \leq j$ 
2237     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2238
2239     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy \in L$  (Für jede
2240     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2241     Sprache  $L$ )
2242   \end{enumerate}
2243 }
2244
```

## 2.35 quicksort.sty

```

2245 % https://tex.stackexchange.com/a/142634
2246 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2247 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2248
2249 %-----
2250 % USAGE:
2251 % \QSinitialize{comma, separated, numerical, values}
2252 % \loop
2253 % \QSpivotStep
2254 % \ifnum\value{pivotcount}>0
2255 %   \QSSortStep
2256 % \repeat
2257 %-----
2258
2259 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2260 \RequirePackage{xintfrac, xinttools}
2261
2262 \RequirePackage{tikz}
2263
2264 %-----
2265 % FIRST PART: TikZ styles and macros for the actual drawing
2266 \newcounter{cellcount}% used for coordinates of the node
2267 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2268
2269 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2270 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2271 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2272 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2273 % by police of LaTeX good conduct ? )
2274 \tikzset{l/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2275          o/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2276          r/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2277 % this is the "b" style as used in the image below
2278          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2279 % nicer:
2280          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2281          g/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2282
2283 % NOTE the b style was originally the same as the r(aised) style apart from
2284 % not being raised, but I find it nicer with a somewhat different
2285 % specification. I have not updated the images though.
2286
2287 % How the nodes are drawn depending on whether on the left of the pivot value
2288 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2289
2290 \def\DecoLEFT #1{%
2291   \xintFor* ##1 in {#1} \do
2292     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2293 }
2294
2295 \def\DecoINERT #1{%
2296   \xintFor* ##1 in {#1} \do
2297     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2298 }
2299
2300 \def\DecoRIGHT #1{%
2301   \xintFor* ##1 in {#1} \do
2302     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2303 }
2304
2305 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2306   \xintFor* ##1 in {#1} \do

```

```

2307     {\stepcounter{cellcount}}%
2308     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2309 }
2310
2311 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2312     \xintFor* ##1 in {#1} \do
2313     {\stepcounter{cellcount}}%
2314     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2315 }
2316
2317 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2318     \xintFor* ##1 in {#1} \do
2319     {\stepcounter{cellcount}}%
2320     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2321 }
2322
2323 %-----
2324 % SECOND PART: the actual sorting routines.
2325
2326 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2327 \def\QS@sort@b #1{\ifcase #1
2328     \expandafter\QS@sort@empty
2329     \or\expandafter\QS@sort@single
2330     \else\expandafter\QS@sort@c
2331     \fi
2332 }%
2333 \def\QS@sort@empty #1{}
2334 \def\QS@sort@single #1{\QSIr {#1}}
2335
2336 % This step is to pick the last as pivot.
2337 \def\QS@sort@c #1%
2338     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2339
2340 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2341 % The trick with \xintApplyUnbraced is that for example when selecting
2342 % the elements smaller than pivot, if we had been using \xintApply we
2343 % would have had at the minimum an empty brace pair. Thus we use the
2344 % "unbraced" variant, but then the \QS@select@smaller has added in
2345 % anticipation a level of braces.
2346 \def\QS@sort@d #1#2{%
2347     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2348     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2349     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2350 }%
2351 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{#{#2}}{ }}% space will stop a f-
    expansion
2352 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{#{#2}}{ }}% space will stop a f-
    expansion
2353 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{#{#2}}{ }}% space will stop a f-
    expansion
2354
2355 %
2356 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2357 %
2358 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2359 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2360 % latter must handle correctly an empty argument.
2361
2362 %-----
2363 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QSinitialize.
2364
2365 % This draws all with suitable highlighting for the newly chosen pivots
2366 % (which will be shown raised)

```

```

2367 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2368             \let\QSIr\DecoINERT
2369             \let\QSIrr\DecoINERT
2370             \let\QSRr\DecoRIGHTwithPivot
2371 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2372             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2373             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2374 }
2375
2376 % This sorts and then draws, showing where the pivot chosen in the previous
2377 % step go. Next time they will have become "inert". If pivotcount is still at
2378 % zero on exit from \QSpivotStep, then this is the signal to stop before
2379 % executing \QSsortStep.
2380 \def\QSsortStep {\def\QSLr {\noexpand\QS@sort@a}%
2381             \def\QSRr {\noexpand\QS@sort@a}%
2382             \def\QSIr {\noexpand\QSIrr}%
2383             \let\QSIrr\relax
2384             \edef\QS@list{\QS@list}%
2385             \let\QSLr\relax
2386             \let\QSRr\relax
2387             \let\QSIr\relax
2388             \edef\QS@list{\QS@list}%
2389             \let\QSLr\DecoLEFT
2390             \let\QSIr\DecoINERTwithPivot
2391             \let\QSIrr\DecoINERT
2392             \let\QSRr\DecoRIGHT
2393 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2394             \setcounter{cellcount}{0}%
2395             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2396 }
2397
2398 \def\QSinitialize #1{%
2399     % first, we convert the comma separated values into a list of braced items
2400     % we use an \edef, and anyhow many \edef's will be used later
2401     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2402     \let\QSRr\DecoRIGHT
2403     % The \QSRr marker mutated to draw the last element as
2404     % pivot and the earlier ones with the suitable style.
2405     %
2406     % The list of marked braced items \QS@list is used both for drawing
2407     % (as here) and for doing the exchange of elements during sort.
2408     \par\centerline{\rule[1.5mm]{0pt}{8mm}}\setcounter{cellcount}{0}%
2409             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2410 }
2411

```

## 2.36 relationale-algebra.sty

```

2412 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2413 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]
2414 \RequirePackage{amsmath}
2415 \RequirePackage{amssymb}

```

```

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

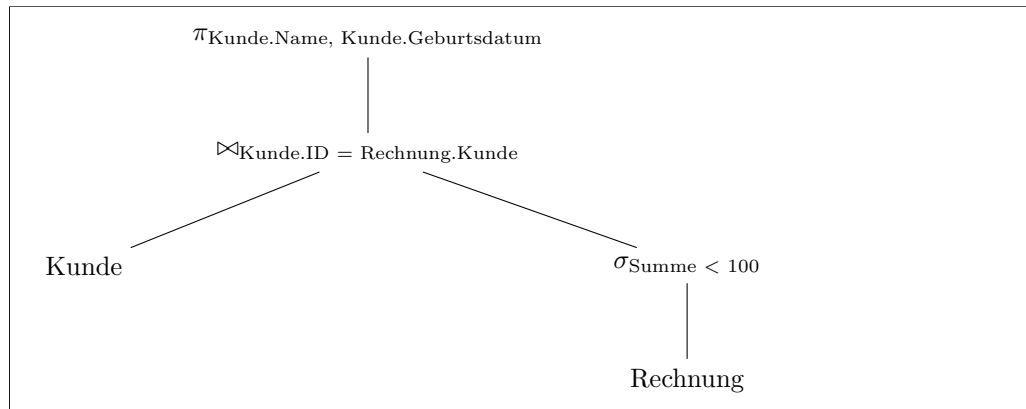
  \node[below=of pi]
    (theta join) {\$\bowtie_{\text{Kunde.ID = Rechnung.Kunde}}\$}
    edge (pi);

  \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}

```



```

2416 \RequirePackage{tikz}
2417 \usetikzlibrary{positioning}

    Privates Makros, das zwei Querstriche erzeugt.
2418 \def\o@join{\setbox0=\hbox{\$\bowtie\$}%
2419   \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2420 }

\leftouterjoin A \leftouterjoin B: A ⋈ B
2421 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}

\rightouterjoin A \rightouterjoin B: A ⋉ B
2422 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}

\fullouterjoin A \fullouterjoin B: A ⋈⋉ B
2423 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
2424

```

## 2.37 rmodell.sty

```
2425 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2426 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2427 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2428 Datenbanken.]
2429 \RequirePackage{soul}
```

### 2.37.1 Makro-Kürzel

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

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

```
2430 \def\liPrimaer#1{\ul{#1}}
```

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

```
2431 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

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

```
2432 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2433 \ExplSyntaxOn
2434 \NewDocumentEnvironment { liRmodell }
2435 { +b }
2436 {
2437   \medskip
2438   {
2439     \linespread{2}
2440     \setlength{\parindent}{0pt}
2441     \li@Rmodell@Schrift#1
2442   }
2443   \medskip
2444 } {}
2445 \ExplSyntaxOff
```

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

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

```
2446 \def\liRelationMenge#1#2{
2447   \noindent
2448   #1 : \[ #2 ]\}
2449   \par
2450 }
```

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

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

```
2451 \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}
```

```
2452 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2453
```

## 2.38 sortieren.sty

```
2454 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2455 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
2456 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}
}
```



```
2457 \RequirePackage{tikz}
2458 \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.

```
2459 \def\liVertauschen#1{
2460   \directlua{
2461     local sortieren = require('lehramt-informatik-sortieren')
2462     sortieren('#1')
2463   }
2464 }
```

`\liSortierPfeil`

```
2465 \def\liSortierPfeil#1#2{
2466   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2467 }
```

`\liSortierPfeilUnten`

```
2468 \def\liSortierPfeilUnten#1#2{
2469   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2470 }
```

`\liSortierMarkierung`

```
2471 \def\liSortierMarkierung#1#2{\node[
2472   draw,
2473   very thick,
2474   fit=(reihe.#1) (reihe.#2),
2475   inner sep=0pt
2476 ] {}
2477 }

2478 \tikzset{
2479   li sortierung zahlenreihe/.style={
2480     draw,
2481     thin,
2482     font=\large,
2483     rectangle split horizontal,
2484     rectangle split,
2485   }
2486 }
```



```

2487 % https://tex.stackexchange.com/a/140895
2488 \RequirePackage{forest,xstring}
2489 \usetikzlibrary{calc}
2490
2491 \makeatletter
2492 \pgfmathdeclarefunction{strrepeat}{2}{%
2493   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2494   \let\pgfmathresult\pgfutil@empty
2495   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2496     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2497     \advance\pgfmath@count-1\relax
2498   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2499 \makeatother
2500
2501 \def\myNodes{}
2502
2503 \ExplSyntaxOn
2504 \newcommand*\sortList[1]{%
2505   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2506 \ExplSyntaxOff
2507
2508 \forestset{
2509   sort/.code={%
2510     \pgfmathparse{level()>\forestSortLevel}%
2511     \ifnum\pgfmathresult=0
2512       \StrSubstitute{\forestov{content}}{ }{,}{\myList}%
2513       \sortList\myList
2514       \StrSubstitute{\myList}{,}{ }{\myList}%
2515       \pgfmathparse{strrepeat("1",level())}%
2516       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-!({\forestov{name}}$)
2517         (m\forestov{name}) {\myList}}%
2518       \pgfmathparse{level()=\forestSortLevel}%
2519       \ifnum\pgfmathresult=1
2520         \forestOget{\forestov{@first}}{name}\forestFirst
2521         \forestOget{\forestov{@last}}{name}\forestLast
2522         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2523           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2524       \fi
2525       \ifnum\forestov{@parent}=0\else
2526         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2527       \fi
2528       \gappto\myNodes{;}%
2529     \fi}}
2530
2531 \forestset{sort level/.code=%
2532   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2533   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2534

```

## 2.39 spalten.sty

```
2535 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2536 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2537 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2538 realisiert werden kann.]
2539 \RequirePackage{multicol}
```

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

```
2540 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2541
```

## 2.40 sql.sty

```
2542 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2543 \ProvidesPackage{lehramt-informatik-tabelle}[2021/09/02 Zu Setzen von SQL]
2544 \liLadePakete{syntax}
2545 \RequirePackage{fancyvrb}
2546 \DefineVerbatimEnvironment{liSqlErgebnis}{Verbatim}
2547 {fontsize=\footnotesize}
2548
```

## 2.41 struktogramm.sty

```
2549 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2550 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2551 Paket struktex zum Setzen von Struktogrammen]
2552 \RequirePackage{struktex}
2553
```

## 2.42 syntax.sty

```
2554 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2555 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2556 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2557 \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.

### 2.42.1 Makro-Kürzel

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

```
2558 \ExplSyntaxOn
2559 \directlua{
2560   syntax = require('lehramt-informatik-syntax')
2561   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2562   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2563   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2564   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2565   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2566   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2567 }
2568 \RequirePackage{hyperref}
2569 \RequirePackage{minted}
2570 % pygmentize -L styles
2571 \usemintedstyle{colorful}
2572 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2573 %\AfterEndEnvironment{minted}{\end{mdframed}}
2574 %\setminted{breaklines=true,linenos}
2575 \setminted{
2576   breaklines=true,
2577   linenos,
2578   fontsize=\footnotesize,
2579 }
```

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

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

```
2580 \def\liJavaCode#1{\,\mintinline{java}|#1|\,}
```

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

```
2581 \def\liLatexCode#1{\mintinline{latex}|#1|}
```

```
2582 \def\li@GithubLink#1#2{
2583   \begin{flushright}
2584     \tiny
2585     Code-Beispiel~auf~Github~ansehen:~
2586     \href{#1}{\nolinkurl{#2}}
2587   \end{flushright}
2588 }
```

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

```
2589 \NewDocumentCommand{\liJavaDatei}{0{firstline=3} m }{
2590   \inputminted[#1]{java}{
2591     \directlua{
2592       syntax.drucke_absoluten_pfad('#2', false)
2593     }
2594   }
2595   \li@GithubLink
```

```

2596     {\directlua{syntax.drucke_github_url('#2', false)}}
2597     {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2598 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2599 \NewDocumentCommand{\liJavaTestDatei}{ 0{firstline=3} m }{
2600   \inputminted[#1]{java}{
2601     \directlua{
2602       syntax.drucke_absoluten_pfad('#2', true)
2603     }
2604   }
2605   \li@GithubLink
2606   {\directlua{syntax.drucke_github_url('#2', true)}}
2607   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2608 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([.
\liJavaExamen$1{$2}{$3}{$4}{$5}
2609 \NewDocumentCommand{\liJavaExamen}{ 0{firstline=3} m m m m }{
2610   \inputminted[#1]{java}{
2611     \directlua{
2612       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2613     }
2614   }
2615   \li@GithubLink
2616   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2617   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2618   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2619 }

\liAssemblerCode
2620 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2621 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2622   \inputminted{asm}{#1}
2623 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2624 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2625   \inputminted{componentpascal}{#1}
2626 }

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2627 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2628 \NewDocumentCommand{\liHaskellDatei}{ m }{
2629   \inputminted{haskell}{#1}
2630 }

2631 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2632 \def\liSqlCode#1{\mintinline{sql}|#1|}
2633

```

## 2.43 syntaxbaum.sty

```
2634 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2635 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2636 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
2637 \RequirePackage{tikz-qtrees}
2638
2639 \tikzset{li parsetree/.style={
2640     every internal node/.style={
2641         draw,circle
2642     },
2643     every leaf node/.style={
2644         draw,rectangle
2645     },
2646 }
2647 }
2648
```

## 2.44 synthese-algorithmus.sty

```
2649 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2650 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2651 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2652 Relation in die 3. Normalform]
2653 \liLadePakete{normalformen,mathe,typographie}
2654 \ExplSyntaxOn
```

### 2.44.1 Makro-Kürzel

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

### 2.44.2 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}
```

### 2.44.3 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}}

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

### 2.44.4 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}

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



## 2.44.5 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  $R_\alpha := \alpha \cup \beta$ . —

### 3. Schlüssel hinzufügen

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

### 4. Entfernung überflüssiger Teilschemata

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

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

```
2655 \def\liSyntheseUeberschrift#1{
2656   {
2657     \bfseries
2658     \sffamily
2659     \str_case:nn {#1} {
2660       {1} {Kanonische-Überdeckung}
2661       {1-1} {Linksreduktion}
2662       {1-2} {Rechtsreduktion}
2663       {1-3} {Löschen-leerer-Klauseln}
2664       {1-4} {Vereinigung}
2665       {2} {Relationsschemata-formen}
2666       {3} {Schlüssel-hinzufügen}
2667       {4} {Entfernung-überflüssiger-Teilschemata}
2668     }
```

```

2669 }
2670 }

\liSyntheseErklaerung Let-Abkürzung: \let\verklaerung=\liSyntheseErklaerung
2671 \def\li@synthese@erklaerung@texte#1{
2672   \str_case:nn {#1} {
2673     {1} {
2674       Die~kanonische~Überdeckung~---also~die~kleinst~mögliche~noch~
2675       äquivalente~Menge~von~funktionalen~Abhängigkeiten~kann~in~vier~
2676       Schritten~erreicht~werden.
2677     }
2678     {1-1} {
2679       Führe~für~jede~funktionale~Anhängigkeit~
2680        $\alpha \rightarrow \beta$ ~in~F~die~Linksreduktion~durch,~
2681       überprüfe~also~für~alle~
2682        $A \in \alpha$ ,~ob~ $A$ ~überflüssig~ist,~d.h.~ob~
2683        $\beta \subseteqq \text{liAttributHuelle}\{F, \alpha \cup A\}$ .
2684     }
2685     {1-2} {
2686       Führe~für~jede~(verbliebene)~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~die~Rechtsreduktion~durch,~überprüfe~also~für~
2687       alle~ $B \in \beta$ ,~ob~ $B \in \text{liAttributHuelle}\{F \cup (\alpha \rightarrow \beta) \cup (\alpha \rightarrow \beta) \cup (\alpha \rightarrow \beta) \}$ ~gilt.~In~diesem~Fall~ist~B~auf~der~rechten~Seite~
2688       überflüssig~und~kann~eliminiert~werden,~d.h.~ $\alpha \rightarrow \beta$ ~wird~durch~ $\alpha \rightarrow (\beta \cup B)$ ~
2689       ersetzt.
2690     }
2691     {1-3} {
2692       Entferne~die~funktionalen~Abhängigkeiten~der~Form~ $\alpha \rightarrow \emptyset$ ,~die~im~2.~Schritt~möglicherweise~
2693       entstanden~sind.
2694     }
2695     {1-4} {
2696       Fasse~mittels~der~Vereinigungsregel~funktionale~Abhängigkeiten~
2697       der~Form~ $\alpha \rightarrow \beta_1, \dots, \alpha \rightarrow \beta_n$ ~so~dass~ $\alpha \rightarrow \beta_1 \cup \dots \cup \beta_n$ ~
2698       verbleibt.
2699     }
2700     % Kemper Seite 197
2701     {2} {
2702       Erzeuge~für~jede~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~in~F~ein~Relationenschema~ $\text{mathcal{R}}_{\alpha \rightarrow \beta}$ ~
2703        $:= \alpha \cup \beta$ .
2704     }
2705     {3} {
2706       Falls~eines~der~in~Schritt~2.~erzeugten~Schemata~ $\text{mathcal{R}}_{\alpha \rightarrow \beta}$ ~
2707       einen~Schlüsselkandidaten~von~ $\text{mathcal{R}}_{\alpha \rightarrow \beta}$ ~bezüglich~ $F \subseteq c$ ~
2708       enthält,~sind~wir~fertig,~sonst~wähle~einen~Schlüsselkandidaten~
2709        $\text{mathcal{K}} \subseteq \text{mathcal{R}}_{\alpha \rightarrow \beta}$ ~aus~und~definiere~folgendes~
2710       zusätzliche~Schema:~ $\text{mathcal{R}}_{\alpha \rightarrow \beta \cup \text{mathcal{K}}} := \text{mathcal{K}} \cup \text{mathcal{F}}$ ~
2711       und~ $\text{mathcal{R}}_{\alpha \rightarrow \beta \cup \text{mathcal{K}}} := \emptyset$ 
2712     }
2713     {4} {
2714       Eliminiere~diejenigen~Schemata~ $\text{mathcal{R}}_{\alpha \rightarrow \beta}$ ,~die~in~einem~
2715       anderen~Relationenschema~ $\text{mathcal{R}}_{\alpha' \rightarrow \beta'}$ ~enthalten~sind,~d.h.~
2716        $\text{mathcal{R}}_{\alpha \rightarrow \beta} \subseteq \text{mathcal{R}}_{\alpha' \rightarrow \beta'}$ .
2717     }
2718   }
2719 }
2720 }
2721 \def\liSyntheseErklaerung#1{
2722 {
2723   \itshape
2724   \footnotesize

```

```

2731     \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
2732   }
2733 }

```

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

```

2734 \def\liSyntheseUeberErklaerung#1{
2735   \liSyntheseUeberschrift{#1}\par
2736   \liSyntheseErklaerung{#1}
2737 }

```

```

2738 \ExplSyntaxOff
2739

```

## 2.45 tabelle.sty

2740 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2741 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]

2742 \RequirePackage{tabularx}

2743

## 2.46 typographie.sty

```
2744 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2745 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
2746 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
2747 formatierung.sty definiert.]
```

```
2748 \ExplSyntaxOn
```

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

```
2749 \RequirePackage{fontawesome}
```

`\liErledigt` `\liErledigt:` ☑

```
2750 \let\liErledigt=\faCheckSquareO
```

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

```
2751 \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 —

```
2752 \def\liParagraphMitLinien#1{
```

```
2753   \noindent
```

```
2754   \vrule height 2pt depth -1.6pt width 0.4cm
```

```
2755   \enspace
```

```
2756   #1
```

```
2757   \enspace
```

```
2758   \leaders\vrule height 2pt depth -1.6pt \hfill \null
```

```
2759   \par
```

```
2760   \medskip
```

```
2761 }
```

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

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

```
2762 \def\liGeschweifteKlammern#1#2#3#4{
```

```
2763   \par
```

```
2764   \medskip
```

```
2765   \noindent
```

```
2766   #1 \, $\= \Bigl\{ $\
```

```
2767   \vspace{#3}
```

```
2768   #2
```

```
2769   \vspace{#4}
```

```
2770   \begin{flushright} $\Bigr\} $\end{flushright}
```

```
2771   \par
```

```
2772 }
```

```
2773 \ExplSyntaxOff
```

```
2774
```

## 2.47 uml.sty

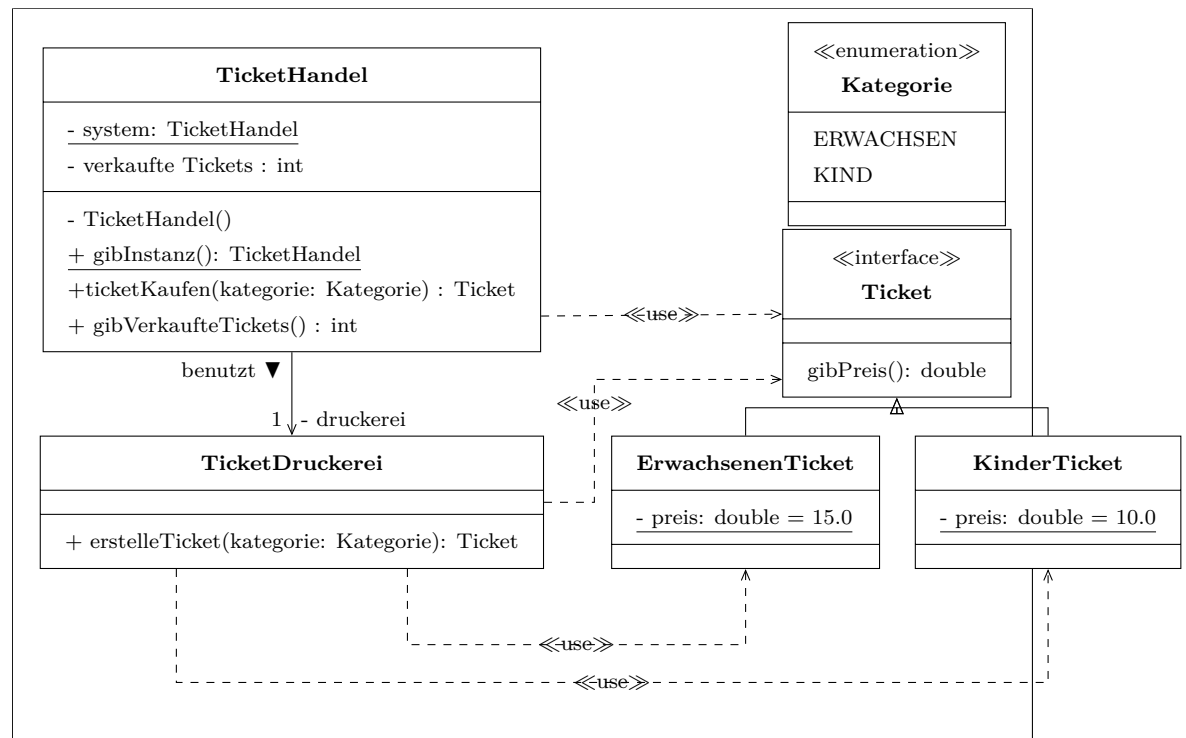
```

2775 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2776 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
2777 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
2778 Erweiterung bereitstellt]

2779 \RequirePackage{tikz-uml}
2780 \RequirePackage{tikz-uml-activity}
2781 % Not compatible with wasysym
2782 %\RequirePackage{mathabx}
2783 \RequirePackage{wasysym}
2784 \usetikzlibrary{positioning}

2785 \tikzumlset{
2786   fill class=white!0,
2787   font=\footnotesize,
2788   fill object=white!0,
2789   fill note=white!0,
2790   fill state=white!0,
2791   % Use case
2792   fill usecase=white!0,
2793   fill system=white!0,
2794 }

```



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

```

2795 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
2796   \def\@liDirLeft{}
2797   \def\@liDirRight{ \RIGHTarrow}
2798   \pgfkeys{/lese/dir/.is choice}
2799   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
2800   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
2801   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{ \LEFTarrow }}}
2802   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
2803
2804   \def\@liPos{above}
2805   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
2806

```

```

2807 \def\@liDistance{0cm}
2808 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}
2809
2810 \pgfkeys{/lese/.cd,#1}
2811
2812 \node[\@liPos = \@liDistance of #2-middle] {
2813   \@liDirLeft{\footnotesize#2}\@liDirRight
2814 };
2815 }
2816

```

## 2.48 vollstaendige-induktion.sty

2817 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2818 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01  
 2819 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die  
 2820 Überschriften für die einzelnen Schritte]

### 2.48.1 Makro-Kürzel

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

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

Lade häufig benötigte Pakete

2821 \RequirePackage{lehramt-informatik-typographie}  
 2822 \RequirePackage{lehramt-informatik-mathe}  
 2823 \RequirePackage{lehramt-informatik-syntax}  
 2824 \ExplSyntaxOn

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

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

2825 \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

```
2826 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
2827 \def\liInduktionAnfang{
2828   \liPseudoUeberschrift{Induktionsanfang}
2829
2830   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2831   \liParagraphMitLinien{
2832     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
2833   }
2834 }
```

\liInduktionVoraussetzung

```
2835 \def\liInduktionVoraussetzung{
2836   \liPseudoUeberschrift{Induktionsvoraussetzung}
2837
2838   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2839   \liParagraphMitLinien{
2840     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ $.
2841   }
2842 }
```

\liInduktionSchritt

```
2843 \def\liInduktionSchritt{
2844   \liPseudoUeberschrift{Induktionsschritt}
2845
2846   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2847   \liParagraphMitLinien{
2848     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
2849     auch~ $A(n=k+1)$ ~wahr~sein~muss.
2850   }
2851 }
```

```
2852 \ExplSyntaxOff
```

```
2853
```

## 2.49 wasserfall.sty

```
2854 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2855 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
2856 \RequirePackage{tikz}
2857 \tikzset{wasserfall/.style={
2858   >=stealth,
2859   node distance = 2mm and -8mm,
2860   start chain = A going below right,
2861   every node/.style = {
2862     draw,
2863     text width=24mm,
2864     minimum height=12mm,
2865     align=center,
2866     inner sep=1mm,
2867     fill=white,
2868     drop shadow={fill=black},
2869     on chain=A
2870   },
2871 }}
2872 \usetikzlibrary{chains,positioning,shadows}
2873
```

## 2.50 wpkalkuel.sty

```
2874 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2875 \ProvidesPackage{lehramt-informatik-wpkalkuel}[2020/06/13]
```

### 2.50.1 Makro-Kürzel

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

```
2876 \RequirePackage{amsmath}
2877 \ExplSyntaxOn
```

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

```
2878 \def\liWpKalkuelOhneMathe#1#2{
2879   \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)
2880 }
2881 \def\liWpKalkuel#1#2{
2882   \ifmmode
2883     \liWpKalkuelOhneMathe{#1}{#2}
2884   \else
2885     $\liWpKalkuelOhneMathe{#1}{#2}$
2886   \fi
2887 }
```

**\MatheEnv**

```
2888 \def\MatheEnv#1{
2889   \medskip
2890
2891   \hspace{1em}#1
2892
2893   \medskip
2894 }
```

**\Mathe**

```
2895 \def\Mathe#1{
2896   \MatheEnv{#1$}
2897 }
```

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

```
2898 \def\liWpEquivalent#1{
2899   \MatheEnv{$\equiv$\hspace{1em}$#1$}
2900 }
```

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

```
2901 \newlength{\@Skip@Erklaerung@Reset}
2902 \def\liWpErklaerung#1{
2903   \setlength{\@Skip@Erklaerung@Reset}{\leftskip}
2904   \setlength{\leftskip}{0.5cm}
2905
2906   \par
2907   \noindent
2908   {
2909     \scriptsize
2910     #1
2911   }
2912   \par
2913
2914   \setlength{\leftskip}{\@Skip@Erklaerung@Reset}
2915 }
```

\liWpErklaerungVerzweigung

```

2916 \def\liWpErklaerungVerzweigung{
2917   $\liWpKalkuelOhneMathe{if~\{-b~\}~then~\{-a1~\}~else~\{-a2~\}}{Q}
2918   \equiv
2919   (b \land \liWpKalkuelOhneMathe{a1}{Q})
2920   \lor
2921   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
2922 }

2923 \ExplSyntaxOff

2924

```

### 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.

<b>Symbols</b>	<b>\AddToHook</b> . . . . . 1372	<b>\bfseries</b> . . 498, 1246,
<b>\#</b> . . . . . 121	<b>\advance</b> . . . . . 2497	1248, 2270, 2276,
<b>\,</b> . . . . 357, 412, 1193,	<b>\AfterEndEnvironment</b> 2573	2278, 2280, 2281, 2657
1195, 1764, 1765,	<b>\allsectionsfont</b> . . . 1242	<b>\Bigl</b> . . . . . 2766
1766, 2103, 2580, 2766	<b>\Alph</b> . . . . . 1256	<b>\Bigr</b> . . . . . 2770
<b>\@Skip@Erklaerung@Reset</b>	<b>\alph</b> . . . . . 1256, 1257	<b>\bigskip</b> . . . . 59, 390,
. . . 2901, 2903, 2914	<b>\alpha</b> 2680, 2682, 2683,	621, 626, 1640, 1971
<b>\@afterheading</b> . . . . 1731	2686, 2688, 2689,	<b>\bool</b> . . . . . 335, 358
<b>\@afterindentfalse</b> . 1731	2690, 2691, 2692,	<b>\bowtie</b> . . . . .
<b>\@liDirLeft</b> 2796, 2801, 2813	2696, 2702, 2703,	2418, 2421, 2422, 2423
<b>\@liDirRight</b> 2797, 2799,	2708, 2709, 2710,	<b>\Box</b> . . . . . 171
2800, 2801, 2802, 2813	2713, 2721, 2722, 2723	<b>\boxtimes</b> . . . . . 488
<b>\@liDistance</b> . . . . .	<b>\arabic</b> 1256, 2292, 2297,	<b>C</b>
. . . 2807, 2808, 2812	2302, 2308, 2314, 2320	<b>\c</b> . . . . . 1339, 1340
<b>\@liPos</b> . . 2804, 2805, 2812	<b>\arraystretch</b> . . . . 1993	<b>\cdot</b> . . . . 1808, 1864, 1875
<b>\</b> . . . . . 619, 642,	<b>B</b>	<b>\centerline</b> . . . . .
643, 646, 647, 650,	<b>\BeforeBeginEnvironment</b>	1437, 2371, 2393, 2408
651, 743, 744, 745,	. . . . . 2572	<b>\chapter</b> . . . . 1246, 1247
852, 881, 883, 909,	<b>\begin</b> 640, 691, 706, 741,	<b>\char</b> . . . . . 1638
918, 963, 1005,	765, 812, 844, 859,	<b>\clearpage</b> . . . . . 1632
1006, 1007, 1012,	879, 889, 905, 925,	<b>\cline</b> . . . . . 619
1013, 1014, 1034,	957, 974, 1003,	<b>\clist</b> . . . . . 238, 282,
1638, 1998, 2056, 2059	1028, 1049, 1064,	283, 303, 307, 2505
<b>\{</b> . . . . . 219, 1130,	1173, 1262, 1337,	<b>\columnbreak</b> . . . . 2540
1140, 1152, 1153,	1432, 1441, 1448,	<b>\cs</b> . . . . 306, 332, 356,
1158, 1192, 1478,	1556, 1630, 1682,	357, 394, 406, 1714
2043, 2448, 2766, 2917	1687, 1695, 1720,	<b>\csname</b> . . . . . 1324, 1327
<b>\}</b> . . . . . 219, 1130,	1725, 1736, 1746,	<b>\cup</b> . . . . . 1153,
1140, 1152, 1153,	1750, 1833, 1837,	2077, 2689, 2703, 2710
1160, 1196, 1479,	1856, 1879, 1902,	<b>D</b>
2043, 2448, 2770, 2917	1917, 1996, 1997,	<b>\DeclareMathSymbol</b> . .
<b>\_</b> . . . . 38, 46, 55, 57,	2047, 2092, 2212,	. . . . . 1941, 1942
306, 332, 356, 357,	2232, 2373, 2395,	<b>\DecoINERT</b> . . . . .
371, 372, 378, 381,	2409, 2572, 2583, 2770	2295, 2368, 2369, 2391
384, 394, 406, 1373	<b>\begingroup</b> 1631, 2045, 2493	<b>\DecoINERTwithPivot</b> .
<b>\_</b> . . . . . 2055, 2065	<b>\beschriftung</b> . . . . .	. . . . . 2311, 2390
<b>A</b>	. . . . 1662, 1666,	<b>\DecoLEFT</b> . . . . 2290, 2389
<b>\addbibresource</b> . . . .	1670, 1674, 1678, 1680	<b>\DecoLEFTwithPivot</b> . .
1608, 1609, 1610,	<b>\beta</b> . . . . . 2680,	. . . . . 2305, 2367
1611, 1612, 1613,	2683, 2687, 2688,	<b>\DecoRIGHT</b> 2300, 2392, 2402
1614, 1615, 1616, 1617	2689, 2692, 2702,	<b>\DecoRIGHTwithPivot</b> .
	2703, 2704, 2709, 2710	. . . . . 2317, 2370
	<b>\bf</b> . . . . 2270, 2271, 2272	

\definecolor .....	1244
\DefineVerbatimEnvironment .....	1555
.....	2546
\delta	77, 119, 177, 219, 1144
\dh .....	1766, 2691
\directlua .....	70, 149, 207, 212, 1125, 1139, 1159, 1167, 1174, 1179, 2028, 2033, 2086, 2093, 2100, 2460, 2559, 2591, 2596, 2597, 2601, 2606, 2607, 2611, 2617, 2618
\do .....	2291, 2296, 2301, 2306, 2312, 2318
\dots .....	529, 533, 1478, 2219, 2702, 2703
\DOWNarrow .....	2800
\draw .....	1344, 1347, 1350, 2167, 2466, 2469
E	
\edef .....	1335, 2384, 2388, 2400, 2401
\else .....	593, 601, 609, 1134, 1148, 1184, 1678, 1781, 1791, 1801, 1815, 2039, 2076, 2162, 2330, 2523, 2525, 2884
\emph .....	1096, 1459, 1488, 1490, 1637
\empty .....	1678, 2075
\emptyset .....	1985, 2697, 2718, 2751
\end .....	679, 702, 727, 762, 796, 828, 856, 872, 886, 894, 921, 950, 971, 996, 1021, 1041, 1061, 1079, 1175, 1264, 1353, 1444, 1445, 1452, 1561, 1635, 1684, 1692, 1711, 1727, 1728, 1742, 1753, 1754, 1835, 1851, 1876, 1899, 1914, 1924, 2000, 2001, 2049, 2094, 2222, 2242, 2373, 2395, 2409, 2573, 2587, 2770
\endscsname ....	1324, 1327
\endgroup	1634, 2050, 2498
\enspace .....	2755, 2757
environments:	
liAdditum .....	1685
liAHuelle .....	2044
liAntwort .....	1656
liDiagramm .....	1744
liEinbettung ....	1655
liExkurs .....	1693
liGraphenFormat .	1316
liKasten .....	1261
liKonfliktflussgraph	
liLernkartei ....	1734
liProduktionsRegeln	
liProjektSprache	1654
liQuellen .....	1714
liRelationenSchemaFormat	2452
liRmodell .....	2432
liUebergangsTabelle	1993
\equiv .....	2899, 2918
\erzeuge@tiefgestellt	1139, 1140, 1144
\expandafter .....	1324, 2326, 2328, 2329, 2330, 2338, 2496
\ExplSyntaxOff ...	61, 104, 146, 151, 204, 209, 214, 551, 573, 588, 1120, 1198, 1233, 1375, 1421, 1576, 1592, 1756, 1831, 2021, 2195, 2445, 2506, 2631, 2738, 2773, 2852, 2923
\ExplSyntaxOn ..	35, 73, 114, 147, 172, 205, 210, 235, 494, 557, 574, 1093, 1188, 1210, 1371, 1390, 1569, 1581, 1657, 1770, 2003, 2176, 2433, 2503, 2558, 2654, 2748, 2824, 2877
F	
\faCheckSquare0 ....	2750
\faCircleThin .....	1112
\faGg .....	1106
\fancyfoot .....	1268, 1269, 1270, 1585, 1586, 1587, 1588
\fancyhead .....	1267, 1582, 1583, 1584
\faSquare0 .....	1100
\fi	595, 603, 611, 1136, 1150, 1186, 1681, 1783, 1793, 1803, 1817, 2041, 2078, 2164, 2331, 2523, 2524, 2527, 2529, 2886
\fontspec .....	1242
\footcite .....	703, 725, 772, 795, 827, 922, 949, 995, 1457, 1460, 1467, 1472, 1477, 1481, 1487, 1492, 1597, 1852, 1853, 2008, 2225
\footnote .....	1758, 1762
\footnotesize	159, 367,
	449, 543, 1117,
1653, 1706, 1721,	
1737, 1952, 2046,	
2063, 2071, 2180,	
2189, 2432, 2547,	
2578, 2730, 2787, 2813	
\footrulewidth .	1273, 1590
\foreach .	1339, 1342, 1349
\forestFirst ..	2520, 2523
\forestLast ...	2521, 2523
\forestOget ...	2520, 2521
\forestOnes .....	2533
\forestOv	2522, 2523, 2526
\forestov .	2512, 2516, 2517, 2520, 2521, 2522, 2523, 2525, 2526
\forestset ....	2508, 2531
\forestSortLevel ....	2510, 2518, 2532, 2533
\frac	1810, 1843, 1875, 1890
\fullouterjoin .....	2423
G	
\g	40, 48, 282, 283, 303, 307, 313, 314, 315, 316, 317, 319, 320, 322, 323, 325, 326, 327, 328, 329, 330, 333, 337, 338, 339, 342, 344, 345, 346, 347, 348, 349, 360, 361, 362, 363, 371, 372, 374, 380, 381, 383, 384, 386, 387, 395, 400, 402, 407, 409, 413, 1392, 1393, 1394, 1397, 1401, 1405, 1409, 1410, 1411, 1412, 1413, 1415, 1416, 1418
\Gamma .	118, 176, 219, 1153
\gappto .....	2528
\geometry .....	5
\geq .....	1485, 1841, 2208, 2213, 2229, 2233
H	
\hbox .....	2418
\headrulewidth .	1272, 1589
\headwidth .....	1591
\hfill	1860, 1866, 1871, 2758
\hinweis .....	1653
\hline .....	1998
\href .....	396, 1206, 1762, 1928, 2586
\hspace ..	2170, 2891, 2899
\ht .....	2419
I	
\i .....	1349, 1350
\ifcase .....	2327
\iffmode	591, 599, 607, 1132, 1146, 1182, 1779, 1789, 1799, 1813, 2037, 2160, 2882

<code>\IfNoValueTF</code> .....	1215, 1218, 1219,	<code>\li@GithubLink</code> .....
... 1688, 1758, 1762	1220, 1221, 1227,	2582, 2595, 2605, 2616
<code>\ifnum</code> .....	1228, 1229, 1230,	<code>\li@mgget</code> . 1326, 1330, 1350
2495, 2511, 2519, 2525	1572, 1573, 1574,	<code>\li@minc</code> .....
<code>\ifx</code> .... 1678, 2075, 2523	1717, 1718, 1719, 1726	1329, 1351
<code>\in</code> .....	<code>\labelenumi</code> .....	<code>\li@msset</code> .....
517, 623,	1257	1323, 1331, 1340, 1343
1485, 1823, 1826,	<code>\labelenumii</code> .....	<code>\li@numdiscs</code> .....
1829, 1858, 1864,	1258	... 1335, 1344, 1350
1869, 2208, 2219,	<code>\labelitemi</code> .....	<code>\li@Rmodell@Schrift</code> .
2229, 2239, 2680,	1251	... 2432, 2441, 2451
2682, 2688, 2709, 2840	<code>\labelitemii</code> .....	<code>\li@sequence</code> .. 1336, 1349
<code>\inhaltsverzeichnis</code> <u>1629</u>	<code>\labelitemiii</code> .....	<code>\li@synthese@erklaerung@texte</code>
<code>\input</code> .....	<code>\labelitemiv</code> .....	2671, 2731
17, 20,	<code>\land</code> .....	<code>\li@Ableitung</code> .....
23, 26, 29, 419, 1406	<code>\LARGE</code> .....	<u>1167</u>
<code>\inputminted</code> 2590, 2600,	<code>\large</code> .....	<code>liAdditum</code> (environment)
2610, 2622, 2625, 2629	1437, 2482	..... <u>1685</u>
<code>\int</code> .....	<code>\leaders</code> .....	<code>liAHuelle</code> (environment)
2505	<code>\left</code> .....	..... <u>2044</u>
<code>\item</code> .....	<code>\LEFTarrow</code> .....	<code>\li@Alphabet</code> .....
488,	2801	<u>1152</u>
489, 708, 712, 717,	<code>\leftarrow</code> .....	<code>liAntwort</code> (environment)
722, 766, 775, 780,	605	..... <u>1656</u>
788, 860, 865, 869,	<code>\leftouterjoin</code> ....	<code>\li@Anweisung</code> .....
890, 926, 931, 938,	2421	<u>1563</u>
946, 975, 980, 984,	<code>\leftskip</code> 2903, 2904, 2914	<code>\li@AssemblerCode</code> ...
989, 1065, 1070,	<code>\LehramtInformatikAutorEmail</code>	<u>2620</u>
1075, 1442, 1443,	..... 1588	<code>\li@AssemblerDatei</code> ..
1714, 1718, 1838,	<code>\LehramtInformatikAutorName</code>	<u>2621</u>
1843, 1847, 1857,	..... 1587	<code>\li@Attribut</code> .....
1863, 1868, 1880,	<code>\LehramtInformatikGitBranch</code>	<u>2451</u>
1884, 1888, 1892,	..... 399, 2566	<code>\li@AttributHuelle</code> ...
1896, 1903, 1907,	<code>\LehramtInformatikGithubCodeRepo</code>	... <u>2035, 2683, 2688</u>
1911, 2213, 2216,	..... 2565	..... <u>2035, 2038,</u>
2219, 2233, 2236, 2239	<code>\LehramtInformatikGithubDomain</code>	<code>\li@AttributMenge</code> ....
<code>\itshape</code> .....	2562	2043, 2055, 2058,
542, 2729	<code>\LehramtInformatikGithubRawDomain</code>	2065, 2066, 2080, 2082
	397, 2563	<code>\LehramtInformatikGithubTexRepo</code>
<b>J</b>	<code>\LehramtInformatikGithubTexRepoAufgabe</code> .....	<u>16</u>
<code>\j</code> 1339, 1340, 1342, 1343,	..... 398, 2564	<code>\li@AufgabenMetadaten</code> .
1344, 1349, 1350, 1351	<code>\LehramtInformatikRepository</code>	<u>36</u>
	..... 17, 20, 23,	<code>\li@AufgabenTitel</code> ....
<b>K</b>	26, 29, 1407, 1608,	<u>62</u>
<code>\k</code> .....	1609, 1610, 1611,	<code>\li@Ausdruck</code> .....
1349	1612, 1613, 1614,	<u>1189</u>
<code>\keys</code> .....	1615, 1616, 1617, 2561	<code>\li@Automat</code> .....
42, 82,	<code>\LehramtInformatikTitel</code>	<u>73</u>
94, 124, 134, 182,	..... 1583	<code>\li@AutomatenKante</code> ...
192, 311, 561, 565,	<code>\leq</code> .... 1875, 2216, 2236	<u>105</u>
579, 584, 1217, 1224	<code>\let</code> .....	<code>\li@BandAlphabet</code> ....
	1142,	<u>1153</u>
<b>L</b>	1143, 1632, 2367,	<code>\li@Bedingung</code> .....
<code>\l</code> .. 75, 76, 77, 78, 79,	2368, 2369, 2370,	<u>1564</u>
80, 83, 84, 85, 86,	2383, 2385, 2386,	<code>\li@BedingungDrei</code> ....
87, 89, 91, 96, 97,	2387, 2389, 2390,	... <u>1828, 1871, 1911</u>
98, 99, 100, 101,	2391, 2392, 2402,	<code>\li@BedingungEins</code> ....
116, 117, 118, 119,	2494, 2532, 2533, 2750	... <u>1822, 1860, 1903</u>
120, 121, 122, 125,	<code>\li@chomsky@erklaerung@texte</code>	<code>\li@BedingungFalsch</code> .
126, 127, 128, 129,	..... 508, 544	<u>1566</u>
130, 131, 137, 138,	<code>\li@EntwurfsCode</code> ....	<code>\li@BedingungWahr</code> ...
139, 140, 141, 142,	636, 682, 683, 684,	<u>1565</u>
143, 174, 175, 176,	730, 731, 732, 733,	<code>\li@BedingungZwei</code> ....
177, 178, 179, 180,	799, 800, 801, 802,	... <u>1825, 1866, 1907</u>
183, 184, 185, 186,	803, 804, 831, 832,	<code>\li@Beschriftung</code> ....
187, 188, 189, 195,	833, 834, 835, 836, 897	<u>1645</u>
196, 197, 198, 199,	<code>\li@EntwurfsCodeAllgemein</code>	<code>\li@BindeAufgabeEin</code> .
200, 201, 559, 562,	..... <u>635</u>	<u>1404</u>
567, 568, 571, 576,	<code>\li@fussnote@text</code> 1949,	<code>\li@ChomskyErklaerung</code>
577, 580, 581, 586,	1955, 1959, 1963, 1967	..... 508, 549
1212, 1213, 1214,		<code>\li@ChomskyUeberErklaerung</code>
		..... <u>547</u>
		<code>\li@ChomskyUeberschrift</code>
		..... <u>496, 548</u>
		<code>\li@CpmEreignis</code> .....
		<u>557</u>
		<code>\li@CpmFruehesterI</code> ...
		<u>614</u>
		<code>\li@CpmSpaetesterI</code> ...
		<u>613</u>
		<code>\li@CpmVon</code> .....
		<u>597</u>
		<code>\li@CpmVonOhneMathe</code> ..
		..... <u>597, 600, 602</u>

\liCpmVonZu	589	\liEntwurfsFabrikmethodeUml	956, 999	\liInduktionErklaerung	2826
\liCpmVonZuOhneMathe	589, 592, 594	\liEntwurfsKompositum	1023	\liInduktionMarkierung	2825
\liCpmVorgang	574	\liEntwurfsKompositumAkteure	1025	\liInduktionSchritt	2843
\liCpmZu	605	\liEntwurfsKompositumUml	1002, 1024	\liInduktionVoraussetzung	2835
\liCpmZuOhneMathe	605, 608, 610	\liEntwurfsModellPraesentation	1044	\liJavaCode	2580
liDiagramm (environment)	1744	\liEntwurfsModellPraesentationAkteure	1046	\liJavaDatei	637, 2589
liEinbettung (environment)	1655	\liEntwurfsModellPraesentationSteuerung	1027, 1045	\liJavaExamen	2609
\liEntwurfs	1044	\liEntwurfsModellPraesentationSteuerungUml	1081	\liJavarescater	2599
\liEntwurfsAbstrakteFabrik	686	\liEntwurfsZustand	1063, 1083	liKasten (environment)	1261
\liEntwurfsAbstrakteFabrikCode	681, 688	\liEntwurfsZustandAkteure	1048, 1082	liKellerAutomat	114
\liEntwurfsAbstrakteFabrikUml	639, 687	\liEntwurfsZustandUml	1138	liKellerKante	152
\liEntwurfsAdapter	735	\liEpsilon	1096, 1110, 1112	liKellerUebergang	147, 153
\liEntwurfsAdapterAkteure	705, 737	\liErAttribute	1115	\liKontrollCode	1567
\liEntwurfsAdapterCode	729, 738	\liErDatenbankName	1094, 1098, 1100	liKontrollflussgraph (environment)	1555
\liEntwurfsAdapterUml	690, 736	\liErledigt	2750	\liKontrollKnotenPfad	1569
\liEntwurfsBeobachter	806	\liErMpAttribute	1109	\liKontrollTextzeileKnoten	1568, 1573
\liEntwurfsBeobachterAkteure	764, 808	\liErMpEntity	1097	\liKurzeTabellenLinie	619
\liEntwurfsBeobachterCode	798, 809	\liErMpRelationship	1103	\liLadeAllePakete	240, 1387
\liEntwurfsBeobachterUml	740, 807	\liErRelationship	1095, 1104, 1106	\liLadePakete	66, 69, 236, 241, 495, 556, 1129, 1359, 1380, 1427, 1947, 2027, 2175, 2544, 2653
\liEntwurfsDekorierer	838	\liExamensAufgabe	19	\liLatexCode	2581
\liEntwurfsDekoriererAkteure	840	\liExamensAufgabeA	28	\liLeereZelle	1985
\liEntwurfsDekoriererCode	830, 841	\liExamensAufgabeTA	25	liLernkartei (environment)	1734
\liEntwurfsDekoriererUml	811, 839	\liExamensAufgabeTTA	22	\liLinksReduktion	2052
\liEntwurfsEinfacheFabrik	874	liExkurs (environment)	1693	\liLinksReduktionInline	2061, 2069
\liEntwurfsEinfacheFabrikAkteure	858, 876	\liFalsch	489	\liMasterExkurs	1916
\liEntwurfsEinfacheFabrikUml	843, 875	\liFlaci	1199	\liMasterFaelle	1855, 1923
\liEntwurfsEinzelstueck	899	\liFremd	2431	\liMasterFallRechnung	1901
\liEntwurfsEinzelstueckAkteure	888, 901	\liFunktionaleAbhaengigkeit	2074, 2077, 2085	\liMasterVariablen	1832, 1918
\liEntwurfsEinzelstueckCode	896, 902	\liFunktionaleAbhaengigkeiten	2088	\liMasterVariablenDeklaration	1878
\liEntwurfsEinzelstueckUml	878, 900	\liFussnote	1948, 1950	\liMasterWolframLink	1926
\liEntwurfsErbauer	952	\liFussnoteDreiText	1962, 1980	\liMenge	83, 84, 86, 125, 126, 127, 131, 183, 184, 185, 189, 1130, 1179, 1218, 1219
\liEntwurfsErbauerAkteure	924, 954	\liFussnoteEinsText	1954, 1974	\liMengeOhneMathe	1130, 1133, 1135
\liEntwurfsErbauerUml	904, 953	\liFussnoteLink	1761	\liMinimierungErklaerung	2007
\liEntwurfsFabrikmethode	998, 1023	\liFussnoteLink	1970	\liMinispracheDatei	2624
\liEntwurfsFabrikmethodeAkteure	973, 1000	\liFussnoten	1042, 1757	\liLinespread	2439
		\liFussnoteUrl	1966, 1983	\liNichtsZuTun	2751
		\liFussnoteVierText	1958, 1977	liO	1795, 1823
		\liFussnoteZweiText	1171, 2089, 2762	liOmega	1785, 1829
		\liGeschweifteKlammern	1210	liOmegaOhneMathe	1785, 1790, 1792
		\liGrammatik	1316		
		liGraphenFormat (environment)	1323		
		\liHanoi	2627		
		\liHaskellCode	2628		
		\liHaskellDatei	2827		
		\liInduktionAnfang			

<code>\li0OhneMathe</code> . . . . .	<code>\liSetzeExamenTeilaufgabeNr</code> . . . . .	<code>\liZustandsmengeNr</code> . . . . .
. . . 1795, 1800, 1802	. . . . . 1400	. . . . . 1156, 2190
<code>\liParagraphMitLinien</code> . . . . .	<code>\liSetzeExamenThemaNr</code> . . . . .	<code>\liZustandsmengeNrGross</code> . . . . .
. . . 544, 2009, 2731,	. . . . . 1396	. . . . . 1164
2752, 2831, 2839, 2847	<code>\liSortierMarkierung</code> 2471	<code>\liZustandsMengenSammlung</code> . . . . .
<code>\liPetriErreichKnotenDrei</code> . . . . .	<code>\liSortierPfeil</code> . . . . .	. . . . . 2177
. . . . . 2169	<code>\liSortierPfeilUnten</code> 2468	<code>\liZustandsMengenSammlungNr</code> . . . . .
<code>\liPetriErreichTransition</code> . . . . .	<code>\liSpaltenUmbruch</code> . . . . .	. . . . . 2186
. . . . . 2166	<code>\liSqlCode</code> . . . . .	<code>\liZustandsmengeOhneMathe</code> . . . . .
<code>\liPetriSetzeSchluessel</code> . . . . .	<code>\listen@punkt</code> . . . . .	. . . . . 1142
. . . . . 2111	<code>\liStrich</code> . . . . .	<code>\liZustandsname</code> . . . . .
<code>\liPetriTransitionsName</code> . . . . .	<code>\liSyntheseErklaerung</code> . . . . .	<code>\liZustandsnameGross</code> . . . . .
. . . . . 2158, 2170	. . . . . 2671, 2736	. . . . . 1166, 2178, 2187
<code>\liPetriTransitionsNameOhneMathe</code> . . . . .	<code>\liSyntheseUeberErklaerung</code> . . . . .	<code>\liZustandsnameTiefgestellt</code> . . . . .
. . . . . 2158, 2161, 2163	. . . . . 2734	. . . . . 1181
<code>\liPetriTransPfeile</code> 2170	<code>\liSyntheseUeberschrift</code> . . . . .	<code>\liZustandsPaar</code> . . . . .
<code>\liPolynomiellReduzierbar</code> . . . . .	. . . . . 2655, 2735	<code>\liZustandsPaarVariablenName</code> . . . . .
. . . . . 1447	<code>\liT</code> 1805, 1820, 1834, 1898	. . . . . 1986, 1989, 1990
<code>\liPotenzmenge</code> . . . . .	<code>\liTeilen</code> . . . . .	<code>\llap</code> . . . . .
. . . . . 1139, 1143, 2181	. . . . . 2032	. . . . . 2419
<code>\liPotenzmengeOhneMathe</code> . . . . .	<code>\literatur</code> . . . . .	<code>\log</code> . . . . .
. . . . . 1140, 1141, 1142	. . . . . 1596, 1620	. . . . . 1823,
<code>\liPrimaer</code> . . . . .	<code>\liTheta</code> . . . . .	1826, 1829, 1858, 1864
<code>\liProblemBeschreibung</code> . . . . .	. . . . . 1775,	<code>\loop</code> . . . . .
. . . . . 1431	1826, 1858, 1864, 1869	. . . . . 2252
<code>\liProblemClique</code> . . . . .	<code>\liThetaOhneMathe</code> . . . . .	<code>\lor</code> . . . . .
. . . . . 1454	. . . . . 1775, 1780, 1782	. . . . . 2920
<code>\liProblemName</code> . . . . .	<code>\liTOhneMathe</code> . . . . .	<code>\ltimes</code> . . . . .
. . . . . 1430, 1437,	. . . . . 1805, 1814, 1816	. . . . . 1934
1449, 1451, 1464,	<code>\liTuringKante</code> . . . . .	
1475, 1476, 1484, 1485	. . . . . 215	
<code>\liProblemSat</code> . . . . .	<code>\liTuringLeerzeichen</code> . . . . .	
. . . . . 1483	. . . . . 171, 179	
<code>\liProblemSubsetSum</code> . . . . .	<code>\liTuringMaschine</code> . . . . .	
. . . . . 1474, 1483	. . . . . 172	
<code>\liProblemVertexCover</code> . . . . .	<code>\liTuringUeberfuehrung</code> . . . . .	
. . . . . 1454, 1462	. . . . . 218	
<code>\liProduktionen</code> 1178, 1220	<code>\liTuringUebergaenge</code> . . . . .	
<code>liProduktionsRegeln</code> (environment) . . . . .	. . . . . 210, 216	
. . . . . 1168	<code>\liTuringUebergangZelle</code> . . . . .	
<code>liProjektSprache</code> (environment) . . . . .	. . . . . 205	
. . . . . 1654	<code>\liUeberfuehrungsFunktion</code> . . . . .	
<code>\liPseudoUeberschrift</code> . . . . .	. . . . . 1144	
. . . . . 1639,	<code>\liUeberfuehrungsFunktionOhneMathe</code> . . . . .	
1689, 1690, 1995,	. . . . . 1144, 1147, 1149	
2005, 2828, 2836, 2844	<code>liUebergangsTabelle</code> (environment) . . . . .	
<code>\liPumpingKontextfrei</code> . . . . .	. . . . . 1993	
. . . . . 2227	<code>\liUeberschriftDreiecksTabelle</code> . . . . .	
<code>\liPumpingRegulaer</code> . . . . .	. . . . . 2003	
. . . . . 2206	<code>\liUmlLeserichtung</code> . . . . .	
<code>liQuellen</code> (environment) . . . . .	. . . . . 2795	
. . . . . 1714	<code>\liVertauschen</code> . . . . .	
<code>\liRechtsReduktionInline</code> . . . . .	. . . . . 2459	
. . . . . 2069	<code>\liWortInSprache</code> . . . . .	
<code>\liRekursionsGleichung</code> . . . . .	. . . . . 620	
. . . . . 1819, 1882	<code>\liWortNichtInSprache</code> . . . . .	
<code>\liRelation</code> . . . . .	. . . . . 625	
. . . . . 2099	<code>\liWpEquivalent</code> . . . . .	
<code>liRelationenSchemaFormat</code> (environment) . . . . .	. . . . . 2898	
. . . . . 2452	<code>\liWpErklaerung</code> . . . . .	
<code>\liRelationMenge</code> . . . . .	. . . . . 2901	
. . . . . 2446	<code>\liWpErklaerungVerzweigung</code> . . . . .	
<code>\liRichtig</code> . . . . .	. . . . . 2916	
. . . . . 488	<code>\liWpKalkuel</code> . . . . .	
<code>liRmodell</code> (environment) . . . . .	. . . . . 2878	
. . . . . 2432	<code>\liWpKalkuelOhneMathe</code> . . . . .	
<code>\liRundeKlammer</code> . . . . .	. . . . . 2878, 2883,	
. . . . . 1772,	2885, 2917, 2919, 2921	
1776, 1786, 1796, 1810	<code>\liZustandsBuchstabe</code> . . . . .	
<code>\liSetzeExamen</code> . . . . .	. . . . . 1154,	
. . . . . 1391	1163, 1165, 1183, 1185	
	<code>\liZustandsBuchstabeGross</code> . . . . .	
	. . . . . 1155, 1164, 1166	
	<code>\liZustandsmenge</code> . . . . .	
	. . . . . 1142	

## M

<code>\makeatletter</code> . . . . .	1730, 2491
<code>\makeatother</code> . . . . .	1732, 2499
<code>\marginpar</code> . . . . .	1099, 1105, 1111, 1637
<code>\mathbb</code> . . . . .	1485, 2239, 2840
<code>\mathbin</code> . . . . .	2421, 2422, 2423
<code>\mathcal</code> . . . . .	1796, 2709,
. . . . .	2714, 2716, 2717, 2718
<code>\Mathe</code> . . . . .	2895
<code>\MatheEnv</code> . . . . .	2888, 2896, 2899
<code>\mathord</code> . . . . .	1941, 1942
<code>\mdfsetup</code> . . . . .	1260,
. . . . .	1663, 1667, 1671, 1675
<code>\medskip</code> . . . . .	1439,
. . . . .	1648, 1650, 1724,
. . . . .	1752, 2437, 2443,
. . . . .	2760, 2764, 2889, 2893
<code>\emph</code> . . . . .	1637
<code>\mintinline</code> . . . . .	2580,
. . . . .	2581, 2620, 2627, 2632
<code>\mkern</code> . . . . .	2421, 2422, 2423
<code>\mlq</code> . . . . .	1939, 1941
<code>\mrq</code> . . . . .	1939, 1942
<code>\msg</code> . . . . .	50, 417
<code>\myList</code> . . . . .	2512, 2513, 2514, 2517
<code>\myNodes</code> . . . . .	2501,
. . . . .	2516, 2522, 2526, 2528

## N

<code>\NeedsTeXFormat</code> . . . . .	1, 14,
. . . . .	32, 64, 232, 423,
. . . . .	484, 491, 553, 616,
. . . . .	631, 1086, 1122,
. . . . .	1235, 1275, 1284,
. . . . .	1289, 1318, 1356,
. . . . .	1377, 1423, 1495,
. . . . .	1578, 1594, 1600,



1622, 1768, 1931, 1944, 2023, 2106, 2172, 2197, 2202, 2246, 2412, 2425, 2454, 2535, 2542, 2549, 2554, 2634, 2649, 2740, 2744, 2775, 2817, 2854, 2874	\neg ..... 2921	\pgfkeys .. 2122, 2798, 2799, 2800, 2801, 2802, 2805, 2808, 2810	\QSpivotStep ..... 2253, 2363, 2367, 2378
\negthinspace ..... 1773	\NewDocumentCommand . 74, 115, 152, 173, 215, 236, 558, 575, 620, 625, 1189, 1211, 1447, 1570, 1598, 1757, 1761, 2088, 2099, 2166, 2589, 2599, 2609, 2621, 2624, 2628, 2795	\pgfmath@count ..... ... 2493, 2495, 2497	\QSr ..... 2340
\NewDocumentEnvironment 1168, 1261, 1316, 1555, 1654, 1655, 1658, 1685, 1693, 1715, 1734, 1744, 1994, 2044, 2434, 2452	\newlength ..... 2901	\pgfmath@smuggleone 2498	\QSRr ..... 2349, 2370, 2381, 2386, 2392, 2401, 2402, 2403
\newlength ..... 2901	\node ..... 571, 1563, 1568, 2292, 2297, 2302, 2308, 2314, 2320, 2471, 2516, 2812	\pgfmathdeclarefunction ..... 2492	\QSSortStep ..... 2255, 2363, 2379, 2380
\noexpand ..... 2380, 2381, 2382, 2401, 2516	\noindent ..... 369, 622, 627, 1641, 1643, 1647, 1651, 1679, 1707, 1709, 1722, 1738, 1740, 1748, 1920, 1973, 1976, 1979, 1982, 2447, 2753, 2765, 2907	\pgfmathhint ..... 2493	\quad ..... 1951
\nolinkurl .... 402, 2586	\normalize ..... 1248	\pgfmathparse ..... ..... 1330, 2510, 2515, 2518, 2532, 2533	R
\notin ..... 628	\null ..... 2758	\pgfmathresult ..... ..... 1331, 2493, 2494, 2496, 2498, 2511, 2519, 2532, 2533	\raisebox ..... 1568
O	\o@join ..... 2418, 2421, 2422, 2423	\pgfutil@empty ..... 2494	\relax ..... 1632, 2340, 2383, 2385, 2386, 2387, 2495, 2497
\Omega ..... 1786	\Omega ..... 2208, 2209, 2229, 2230	\pgfutil@loop ..... 2495	\renewcommand ... 1251, 1252, 1253, 1254, 1257, 1258, 1272, 1273, 1589, 1590, 1993
\omega ..... 2329	\or ..... 2329	\pgfutil@repeat .... 2498	\repeat ..... 2256
P	\pagestyle ..... 1271	\preceq ..... 1450	\RequirePackage . 4, 67, 170, 234, 238, 420, 426, 427, 487, 555, 634, 635, 1089, 1091, 1092, 1128, 1237, 1238, 1241, 1243, 1245, 1250, 1259, 1266, 1277, 1278, 1287, 1291, 1292, 1293, 1321, 1322, 1370, 1388, 1389, 1428, 1497, 1597, 1602, 1603, 1619, 1626, 1627, 1628, 1656, 1771, 1935, 1936, 2108, 2110, 2200, 2260, 2262, 2414, 2415, 2416, 2429, 2457, 2488, 2539, 2545, 2552, 2557, 2568, 2569, 2637, 2742, 2749, 2779, 2780, 2782, 2783, 2821, 2822, 2823, 2856, 2876
\par ... 368, 389, 548, 1200, 1208, 1646, 1708, 1731, 1739, 2371, 2393, 2408, 2449, 2735, 2759, 2763, 2771, 2906, 2912	\paragraph ..... 1248	\prime ..... 1429	\right ..... 1773
\parindent ..... 2440	\path ... 106, 153, 216, 586	\printbibliography . 1620	\RIGHTarrow ... 2797, 2802
		\ProvidesPackage 2, 15, 33, 65, 233, 424, 485, 492, 554, 617, 632, 1087, 1123, 1236, 1276, 1285, 1290, 1319, 1357, 1378, 1424, 1496, 1579, 1595, 1601, 1623, 1769, 1932, 1945, 2024, 2107, 2173, 2198, 2203, 2247, 2413, 2426, 2455, 2536, 2543, 2550, 2555, 2635, 2650, 2741, 2745, 2776, 2818, 2855, 2875	\Rightarrow .... 623, 628
		Q	\rightarrow ..... 219, 512, 517, 525, 529, 531, 532, 534, 589, 597, 2170, 2680, 2687, 2689, 2692, 2697, 2702, 2703, 2708
		\QS@list ..... 2373, 2384, 2388, 2395, 2401, 2406, 2409	\rightouterjoin ... 2422
		\QS@select@equal .... ..... 2348, 2352	\Roman ..... 1256
		\QS@select@greater .. ..... 2349, 2353	\roman ..... 1256, 1258
		\QS@select@smaller .. ... 2344, 2347, 2351	\romannumeral ..... 2338
		\QS@sort@a ..... 2326, 2359, 2380, 2381	\rtimes ..... 1934
		\QS@sort@b .... 2326, 2327	\rule 2371, 2393, 2408, 2419
		\QS@sort@c .... 2330, 2337	
		\QS@sort@d .... 2338, 2346	S
		\QS@sort@empty . 2328, 2333	\sb ..... 79, 89, 91, 120, 178, 524,
		\QS@sort@single 2329, 2334	
		\QSinitialize ..... ... 2251, 2363, 2398	
		\QSIr . 2334, 2340, 2348, 2368, 2382, 2387, 2390	
		\QSIrr 2369, 2382, 2383, 2391	
		\QSLr ..... 2340, 2347, 2358, 2359, 2367, 2380, 2385, 2389	

525, 529, 532, 533,	\textit . . . . .	963,	\TmpTransitionEight .
534, 1183, 1185,	1005, 1006, 1007,		. . . . . 2119, 2140
1823, 1826, 1829,	1008, 1749, 2043, 2103		\TmpTransitionFive ..
1858, 1864, 2013,	\textsc . . . . .	1430	. . . . . 2116, 2137
2158, 2167, 2702,	\textsf . . . . .	1642, 1723	\TmpTransitionFour ..
2703, 2704, 2709,	\textstyle . . . .	1843, 1875	. . . . . 2115, 2136
2713, 2714, 2717,	\texttt 1117, 1430, 1564,		\TmpTransitionNine ..
2718, 2721, 2722, 2723	1565, 1566, 1567, 2879		. . . . . 2120, 2141
\scriptscriptstyle ..	\textwidth . . . . .	1591	\TmpTransitionOne ...
. . . . . 589, 597, 605	\the page . . . . .	1270, 1586	. . . . . 2112, 2133
\scriptsize .... 1202,	\the paragraph . . . . .	1248	\TmpTransitionSeven .
1508, 1515, 1521,	\Theta . . . . .	1776	. . . . . 2118, 2139
1583, 1584, 1587,	\thinspace . . . . .	2879	\TmpTransitionSix ...
1588, 2826, 2879, 2909	\tikz . . . . .	1568	. . . . . 2117, 2138
\section . . . . . 57, 1398	tikz: bbaum . . . . .	25	\TmpTransitionTen ...
\seq . 1572, 1573, 1574,	tikz: li binaer baum ....	23	. . . . . 2121, 2142
1717, 1718, 1719, 1726	\tikzchildnode . . . . .	444	\TmpTransitionThree .
\setbox . . . . . 2418	\tikzparentnode . . . . .	444	. . . . . 2114, 2135
\setcounter . . . . .	\tikzset . . . . .	108,	\TmpTransitionTwo ...
1249, 2372, 2394, 2408	155, 221, 429, 455,		. . . . . 2113, 2134
\setganttlinklabel ..	1295, 1499, 2148,		\TmpX . . . . . 2144
1279, 1280, 1281, 1282	2274, 2478, 2639, 2857		\TmpY . . . . . 2145
\setlength . . . . . 1591,	\tikzumlset . . . . .	2785	\today . . . . . 1584
2440, 2903, 2904, 2914	\times . . . . .	219	\ttfamily . . . . . 2432
\setmainfont . . . . . 1239	\tiny . . . . . 1100, 1106,		
\setmainlanguage .... 421	1112, 1567, 1637, 2584		U
\setminted .... 2574, 2575	\titleformat .. 1246, 1248		\ul . . . . . 1095, 2430, 2431
\setminus . . . . . 2073	\titlespacing . . . . .	1247	\umlaggreg . . . . . 1058
\setsansfont . . . . . 1240	\tl . . . . . 40, 48, 75, 76,		\umlassoc . . . . . 1040
\setul . . . . . 2431	77, 78, 79, 80, 83,		\umlclass ... 641, 645,
\sffamily . . . . . 499,	84, 85, 86, 87, 89,		649, 693, 694, 695,
1246, 1248, 1350, 2658	91, 116, 117, 118,		742, 747, 752, 755,
\shoveleft . . . . . 2053	119, 120, 121, 122,		813, 814, 815, 820,
\shoveright . . . . . 2057	125, 126, 127, 128,		821, 850, 880, 907,
\Sigma . . . . . 76, 117,	129, 130, 131, 174,		908, 911, 962, 965,
175, 1152, 1153, 1213	175, 176, 177, 178,		1004, 1010, 1011,
\sigma . . . . . 522, 524, 525	179, 180, 183, 184,		1029, 1030, 1031,
\SLASH . . . . . 1638	185, 186, 187, 188,		1050, 1051, 1052, 1053
\small . . . . . 1747	189, 304, 308, 333,		\umldep . . . . . 970
\sort . . . . . 2505	337, 338, 339, 342,		\umlHVHaggreg . . . . .
\sortList . . . . . 2504, 2513	347, 348, 349, 360,		. . . . . 760, 826, 1020
\square . . . . . 489	361, 362, 363, 374,		\umlinherit . . . . .
\stepcounter 2292, 2297,	380, 383, 386, 395,		699, 750, 915, 960, 968
2302, 2305, 2307,	409, 559, 562, 567,		\umlnote ... 701, 917, 1060
2311, 2313, 2317, 2319	568, 576, 577, 580,		\umlreal . . . . . 697, 758
\str .. 500, 509, 1660,	581, 1191, 1212,		\umlsimpleclass .. 656,
2182, 2191, 2659, 2672	1213, 1214, 1215,		657, 658, 662, 664,
\string . . . . . 2055, 2065	1218, 1219, 1220,		665, 666, 692, 845,
\StrSubstitute . 2512, 2514	1221, 1392, 1393,		846, 847, 906, 958, 959
\strut .... 1880, 1884,	1394, 1397, 1401,		\umlstatic . . . . . 852, 881
1888, 1892, 1896, 2540	1405, 1412, 1415, 1806		\umluniaggreg . . . . . 913
\subsection . . . . . 1402	\tmp . . . . . 2075		\umluniassoc .... 676,
\subsetq 2683, 2716, 2723	\TmpPlaceEight . . . . .	2130	698, 914, 1038, 1039
	\TmpPlaceFive . . . . .	2127	\umlVHuniassoc .. 677, 678
	\TmpPlaceFour . . . . .	2126	\umlVHVdep . . . . . 670,
	\TmpPlaceNine . . . . .	2131	671, 673, 674, 854, 855
	\TmpPlaceOne . . . . .	2123	\umlVHVinherit . . . . .
	\TmpPlaceSeven . . . . .	2129	653, 654, 659, 660,
	\TmpPlaceSix . . . . .	2128	667, 668, 823, 824,
	\TmpPlaceTen . . . . .	2132	848, 849, 1018, 1019
	\TmpPlaceThree . . . . .	2125	\umlVHVreal . . . . .
	\TmpPlaceTwo . . . . .	2124	. 817, 81

<code>\usemintedstyle</code> .... 2571	<code>\vspace</code> ..... 1694, 1712, 2767, 2769	<code>\xintifEq</code> ..... 2352
<code>\usetikzlibrary</code> .. 68, 428, 1090, 1294, 1498, 2109, 2417, 2458, 2489, 2784, 2872		<code>\xintifForLast</code> ..... ... 2308, 2314, 2320
	<b>X</b>	<code>\xintifGt</code> ..... 2353
	<code>\xappto</code> .. 2516, 2522, 2526	<code>\xintifLt</code> ..... 2351
	<code>\xdef</code> ..... 1324	<code>\xintLength</code> ..... 2326
<b>V</b>	<code>\xintApply</code> ..... 2342	<code>\xintnthelt</code> ..... 2338
<code>\value</code> ..... 2254	<code>\xintApplyUnbraced</code> .. 2341, 2347, 2348, 2349	<b>Z</b>
<code>\varepsilon</code> ..... 501, 512, 513, 1138, 1823, 1829, 1861, 1872	<code>\xintCSVtoList</code> ..... 2401	<code>\ZB</code> ..... 1765
	<code>\xintFor</code> ..... 2291, 2296, 2301, 2306, 2312, 2318, 2359	<code>\zB</code> ..... 1764
<code>\vfill</code> ..... 2540		<code>\zustandsnamens@liste</code> ... 1156, 1163, 1164
<code>\vrule</code> ..... 2754, 2758		