

# lehramt-informatik

Hermine Bschlangaul \*

September 10, 2021

## Contents

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

---

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

2.11.8 Einzelstück (Singleton)	43
2.11.9 Erbauer (Builder)	44
2.11.10 Fabrikmethode (Factory Method)	45
2.11.11 Kompositum (Composite)	47
2.11.12 Modell-Präsentation-Steuerung (Model-View-Controller)	47
2.11.13 Stellvertreter (Proxy)	48
2.11.14 Zustand (State)	49
2.12 er.sty	51
2.12.1 Tex-Markup-Beispiel: Komplettes Diagramm	51
2.12.2 Tex-Markup-Beispiel: EER Enhanced Entity-Relation-Modell nach Elmasri/Navante	52
2.12.3 Makro-Kürzel	52
2.13 formale-sprachen.sty	54
2.14 formatierung.sty	58
2.14.1 Schriftarten / Typographie	58
2.14.2 Farben	58
2.14.3 Überschriften	58
2.14.4 Listen	58
2.14.5 Kasten	58
2.14.6 Header	58
2.14.7 Zeilenabstände	58
2.15 gantt.sty	60
2.16 grafik.sty	61
2.17 graph.sty	62
2.18 hanoi.sty	64
2.19 klassen-konfiguration-aufgabe.sty	65
2.20 klassen-konfiguration-examen.sty	67
2.21 komplexitaetstheorie.sty	70
2.21.1 Makro-Kürzel	70
2.22 kontrollflussgraph.sty	72
2.22.1 Makro-Kürzel	72
2.22.2 TeX-Markup-Beispiel	72
2.22.3 TikZ: pin	72
2.22.4 Umgebungen	74
2.22.5 Makros	74
2.23 kopf-fusszeilen.sty	75
2.24 literatur-dummy.sty	76
2.25 literatur.sty	77
2.26 makros.sty	78
2.27 master-theorem.sty	83
2.27.1 Makro-Kürzel	83
2.28 mathe.sty	87
2.29 meta.sty	88
2.29.1 Einfache Makros (Low level)	88
2.29.2 Zusammengesetzte Makros (High level)	89
2.30 minimierung.sty	91
2.31 normalformen.sty	94
2.31.1 Makro-Kürzel	94
2.32 o-notation.sty	97
2.32.1 Makro-Kürzel	97
2.32.2 TeX-Markup-Beispiel: Funktionsgraphen mit pgfplots	97
2.33 petri.sty	98
2.33.1 Makro-Kürzel	98
2.34 potenzmengen-konstruktion.sty	100
2.35 pseudo.sty	102
2.36 pumping-lemma.sty	103
2.37 quicksort.sty	104
2.38 relationale-algebra.sty	107

2.39	rmodell.sty	108
2.39.1	Makro-Kürzel	108
2.40	sortieren.sty	109
2.41	spalten.sty	111
2.42	sql.sty	112
2.42.1	Latex-Markup-Beispiel	112
2.43	struktogramm.sty	113
2.44	syntax.sty	114
2.44.1	Makro-Kürzel	114
2.45	syntaxbaum.sty	117
2.46	synthese-algorithmus.sty	118
2.46.1	Makro-Kürzel	118
2.46.2	TeX-Markup Grundgerüst	118
2.46.3	TeX-Markup Linksreduktion	118
2.46.4	TeX-Markup Rechtsreduktion	118
2.46.5	TeX-Markup Relationen formen	119
2.47	tabelle.sty	122
2.48	typographie.sty	123
2.49	uml.sty	125
2.50	vollstaendige-induktion.sty	127
2.50.1	Makro-Kürzel	127
2.51	wasserfall.sty	129
2.52	wpkalkuel.sty	130
2.52.1	Makro-Kürzel	130

### 3 Index 131

# 1 Klassen

## 1.1 Vorlage Theorie-Teil

```
% \documentclass{lehramt-informatik-haupt}
%
% \begin{document}
%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% % Theorie-Teil
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% \chapter{Thema des Theorie-Teils}
%
% \literatur
%
% \end{document}
%
```

## 1.2 Vorlage Aufgabensammlung

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

### 1.3 Vorlage Aufgabe

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

## 2 Pakete



## 2.1 abmessung.sty

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

## 2.2 aufgaben-einbinden.sty

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

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

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

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

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

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

31
```

## 2.3 aufgaben-metadaten.sty

```

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

35 \ExplSyntaxOn

\liMetaSetze Setze die Metadaten einer Aufgabe.

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

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

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

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

54 \ExplSyntaxOff

\liAufgabenTitel Momentan eine dummy Makro das die Thematik enthält.

55 \def\liAufgabenTitel#1{}

56

```

## 2.4 automaten.sty

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

### 2.4.1 Endlicher Automat

```
59 \liLadePakete{formale-sprachen}

% \begin{tikzpicture}[li_automat]
% \node[state,initial,accepting] (0) {$z_0$};
% \node[state,right of=0] (1) {$z_1$};
%
% \path (0) edge[above] node{1} (1);
% \path (0) edge[loop,above] node{0} (0);
% \path (1) edge[loop,above] node{0} (1);
% \end{tikzpicture}
%
```



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

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

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

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

```

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

```

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

```

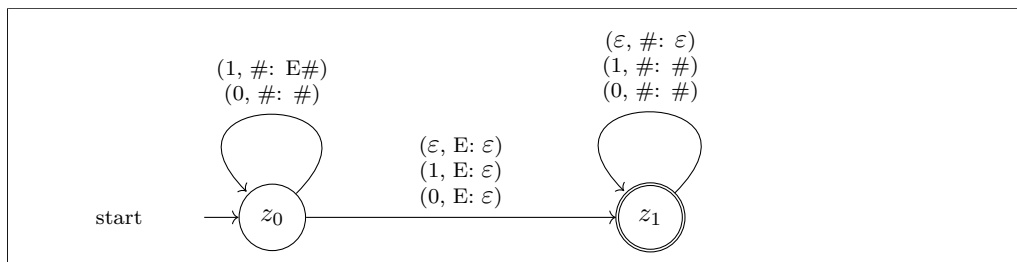
98 \def\liAutomatenKante#1#2#3#4{
99   \path (#1) edge[#4] node{#3} (#2);
100 }

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

```

## 2.4.2 Kellerautomat

```
% \begin{tikzpicture}[li kellerautomat,node distance=5cm]
% \node[state,initial] (0) {$z_0$};
% \node[state,right of=0,accepting] (1) {$z_1$};
%
% \liKellerKante[above,loop]{0}{0}{
%   1, KELLERBODEN, E KELLERBODEN;
%   0, KELLERBODEN, KELLERBODEN;
% }
%
% \liKellerKante[above]{0}{1}{
%   EPSILON, E, EPSILON;
%   1, E, EPSILON;
%   0, E, EPSILON;
% }
%
% \liKellerKante[above,loop]{1}{2}{
%   EPSILON, KELLERBODEN, EPSILON;
%   1, KELLERBODEN, KELLERBODEN;
%   0, KELLERBODEN, KELLERBODEN;
% }
% \end{tikzpicture}
%
```



```
\liKellerAutomat \liKellerAutomat[{automaten-name}]
{{zustaende=Z,alphabet=Σ,kelleralphabet=Γ,delta=δ,start=z0,kellerboden=#,ende=E}}

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

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

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

```

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

```

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

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

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

```

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

```

**\liKellerKante** `\liKellerKante[\tikz-optionen]{\von}{\zu}{\übergänge}`

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

```

145 \NewDocumentCommand{\liKellerKante} { 0{above} m m m } {
146   \path (#2) edge[#1] node{\liKellerUebergang{#4}} (#3);
147 }
148 \tikzset{
149   li keller knoten/.style={
150     text width=2cm,
151     align=center,
152     font=\footnotesize,
153   },
154   li kellerautomat/.style={
155     li automat,
156     every edge/.append style={
157       every node/.style={
158         li keller knoten
159       }
160     }
161   }
162 }

```

### 2.4.3 Turingmaschine

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

**\liTuringLeerzeichen**

$\square$

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

```

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

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

```

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

```

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

```

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

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

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

```

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

```

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

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

`(z_1: \square, L)`

`(\square: \square, R)`



```

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

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

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

\liTuringUeberfuehrung

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

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

```

## 2.5 basis.sty

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

227 \RequirePackage{xparse}

228 \ExplSyntaxOn
```

### 2.5.1 IFs

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

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

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

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

\liLadePakete

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

\liLadeAllePakete

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

```

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

```

Definition einer Komma-getrennten-Liste mit deren Hilfe die vielen globalen Token-List-Variablen definiert werden können. Die einzelnen Schlüssel sind im Interface `AufgabenMetadaten` in der Typescript-Datei `.scripts/nodejs/src/aufgaben.ts` erklärt.

```

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

```

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

```

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

```

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

```

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

```

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

```

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

```

```

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

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

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

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

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

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

```

```

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

```

Einzelprüfungsnummer / Jahr / Jahreszeit mit Trennzeichen

```

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

```

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

```

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

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

```

```

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

```

## 2.6 baum.sty

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

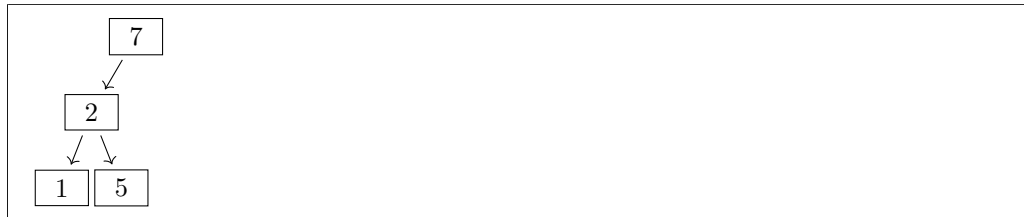
    für li binaer baum
493 \RequirePackage{tikz-qtrees}

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

### 2.6.1 Binärbaum

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

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

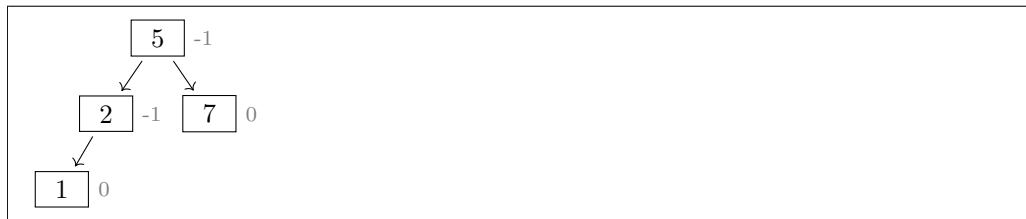


```
495 \tikzset{
496   li binaer baum/.style={
497     shorten <=2pt,
498     shorten >=2pt,
499     ->,
500     every tree node/.style={
501       minimum width=2em,
502       draw,
503       rectangle
504     },
505     blank/.style={
506       draw=none
507     },
508     edge from parent/.style={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



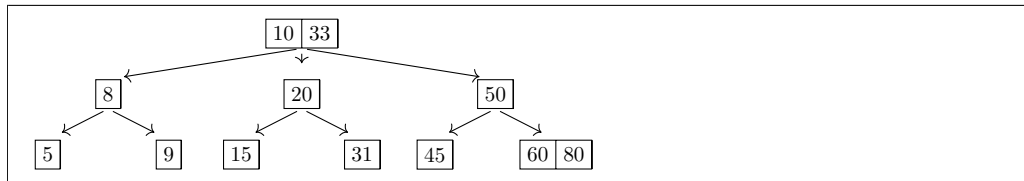
### 2.6.2 AVL-Baum

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



### 2.6.3 B-Baum

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



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

## 2.7 checkbox.sty

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

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

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

556
```

## 2.8 chomsky-normalform.sty

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

560 \ExplSyntaxOn

561 \liLadePakete{typographie}
```

### 2.8.1 Makro-Kürzel

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

### 2.8.2 TeX-Markup-Grundgerüst

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

### 2.8.3 Konkretes TeX-Markup-Beispiel

```
% \begin{enumerate}
% \item \schrittE{1}
%
% \liNichtsZuTun
%
% \item \schrittE{2}
%
% \begin{liProduktionsRegeln}
% S -> d S e | a | U c T | S b U,
% T -> d S e | a,
% U -> d S e | a | U c T,
% \end{liProduktionsRegeln}
%
% \item \schrittE{3}
%
% \begin{liProduktionsRegeln}
% S -> D S E | a | U C T | S B U,
% T -> D S E | a,
% U -> D S E | a | U C T,
% B -> b,
% C -> c,
% D -> d,
% E -> e,
% \end{liProduktionsRegeln}
%
% \item \schrittE{4}
% % S -> S S.1 | T2 S.2 | a | U S.3
% % T -> T2 S.2 | a
% % U -> T2 S.2 | a | U S.3
% % T1 -> b
% % T2 -> d
% % T3 -> e
```

```

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

```

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

```

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

```

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

```

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

```

```

600      A\sb{2}~\rightarrow B\sb{1}~B\sb{2}$~zerteilt.~
601      Nach~der~Ersetzung~sind~alle~längeren~Nonterminalketten~
602      vollständig~heruntergebrochen~und~die~Chomsky-Normalform~erreicht.
603    }
604  }
605 }
606 \def\liChomskyErklaerung#1{
607   {
608     \itshape
609     \footnotesize
610     \liParagraphMitLinien{\li@chomsky@erklaerung@texte{#1}}
611   }
612 }

```

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

```

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

617 \ExplSyntaxOff
618

```

## 2.9 cpm.sty

```

619 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
620 \ProvidesPackage{lehramt-informatik-cpm}[2020/09/03]
621 \RequirePackage{tikz}
622 \liLadePakete{mathe,typographie}

```

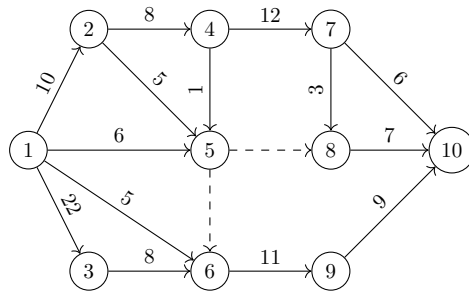
### 2.9.1 Makro-Kürzel

```

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

```

### 2.9.2 TeX-Markup-Beispiel: Graph



```

% \begin{tikzpicture}[scale=0.8,transform shape]
% \liCpmEreignis{1}{0}{2}
% \liCpmEreignis{2}{1}{4}
% \liCpmEreignis{3}{1}{0}
%
% \liCpmVorgang{1}{2}{10}
% \liCpmVorgang{1}{3}{22}
% \liCpmVorgang{1}{5}{6}
%
% \liCpmVorgang[schein]{5}{6}{}
% \liCpmVorgang[schein]{5}{8}{}
% \end{tikzpicture}
%

```

### 2.9.3 TeX-Markup-Beispiel: Ergebnistabelle

```

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

```

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

```

% \liCpmFruehErklaerung
% \begin{tabular}{|l|l|l|r|}
% \hline
% i & Nebenrechnung & & \FZ \\ \hline
% 1 & & & 0
% \end{tabular}

```

```

% 2 & & 5 \\
% 3 & & 18 \\
% 4 & & 7 \\
% 5 & & 19 \\
% 6 & & 26 \\
% 7 &  $\max(19_3, 22_4)$  & 22 \\
% 8 &  $\max(30_5, 30_6, 28_7)$  & 30 \\
% \end{tabular}
%

```

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

```

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

```

```

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

```

```

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

```

```

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

```

```

640 \ExplSyntaxOn
641 \NewDocumentCommand { \liCpmVorgang } { 0{} m m m } {
642   \tl_set:Nn \l_schein_tl {}
643   \tl_set:Nn \l_kritisch_tl {}
644
645   \keys_define:nn { cpmVorgang } {
646     schein .code:n = {\tl_set:Nn \l_schein_tl {dashed}},
647     kritisch .code:n = {\tl_set:Nn \l_kritisch_tl {thick}},
648   }
649
650   \keys_set:nn { cpmVorgang } { #1 }
651
652   \path[->,\l_schein_tl,\l_kritisch_tl] (#2) edge node[auto,sloped] {#4} (#3);

```



```

653 }
654 \ExplSyntaxOff

```

## 2.9.6 Hilfsmakros für Zeitpunkt-Tabelle

```

% \begin{tabular}{|l||l|l|l|l|l|l|l|l|l|}
% \hline
% $i$ & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & \\ \hline \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):  $1_{(2 \rightarrow 3)}$ 
655 \def\liCpmVonZuOhneMathe#1(#2-#3){#1_{\scriptscriptstyle(#2\rightarrow#3)}}
656 \def\liCpmVonZu#1(#2-#3){%
657 \ifmmode%
658 \liCpmVonZuOhneMathe{#1}(#2-#3)%
659 \else%
660 $\liCpmVonZuOhneMathe{#1}(#2-#3)$%
661 \fi%
662 }

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

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

679 \ExplSyntaxOn

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

\liCpmFruehI Frühester Zeitpunkt, zu dem Ereignis  $i$  eintreten kann.
Let-Abkürzung: \let\FZ=\liCpmFruehI
687 \NewDocumentCommand{ \liCpmFruehI } { 0{ i } } {

```

```

688 \ifmode
689   FZ\sb{#1}
690 \else
691   $FZ\sb{#1}$
692 \fi
693 }

```

\liCpmFruehErklaerung

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

```

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

```

\liCpmSpaetErklaerung

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

```

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

```

730 \ExplSyntaxOff

731

## 2.10 cyk-algorithmus.sty

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

### 2.10.1 Makro-Kürzel

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

### 2.10.2 TeX-Markup-Beispiel

```
% \begin{tabular}{|c|c|c|c|c|c|}
% a      & c      & b      & c      & a      & b \\ \hline \hline
%
% $R_a$ & $R_c$ & $R_b$ & $R_c$ & $R_a$ & $R_b$ \16
% B      & A      & A      & B      & C      & \15
% S      & -      & S      & S      & \14
% -      & -      & -      & \13
% -      & -      & \12
% S \11
% \end{tabular}
% \liWortInSprache{acbcab}
%
```

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

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

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

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

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

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

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

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

```
746
```

## 2.11 entwurfsmuster.sty

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

### 2.11.1 Namensschema der Entwurfsmuster-Makros:

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

### 2.11.2 Reihenfolge

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

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

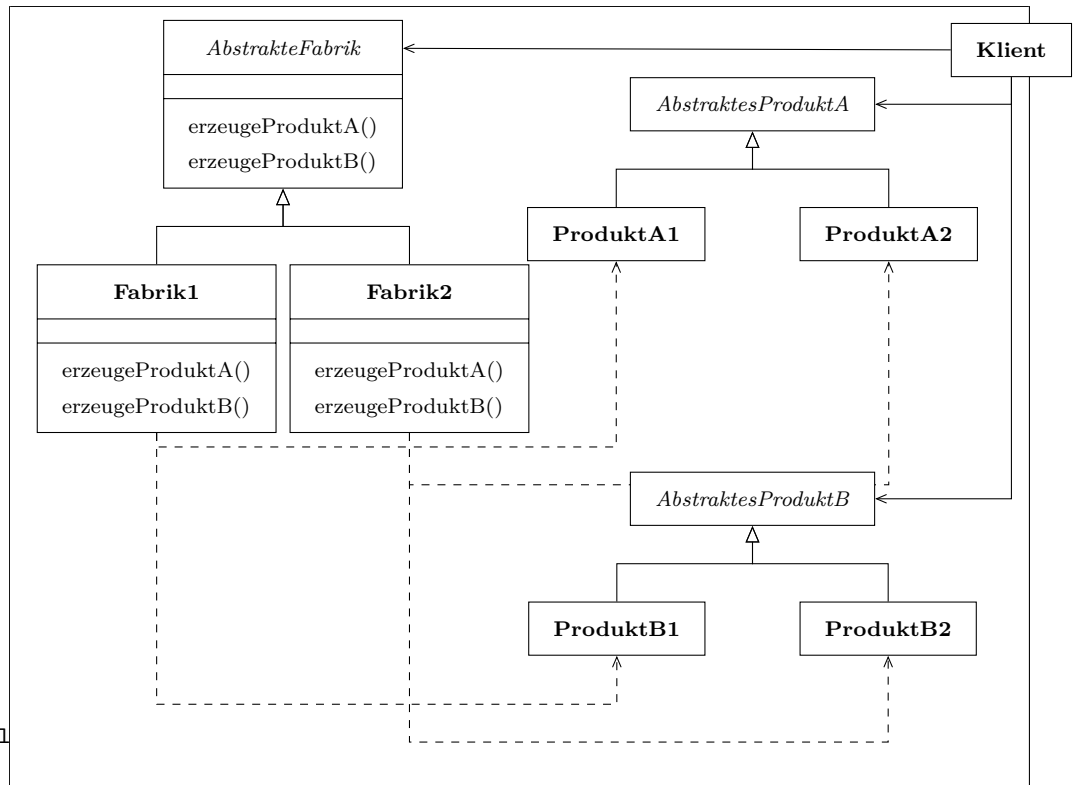
\li@EntwurfsCodeAllgemein Allgemeine Code-Beispiele zu den UML-Diagrammen und Stellvertretern

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

### 2.11.3 Abstrakte Fabrik (Abstract Factory)

sAbstrakteFabrikBeschreibung

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



```

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

```

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

```

iEntwurfsAbstrakteFabrikCode

```

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

```

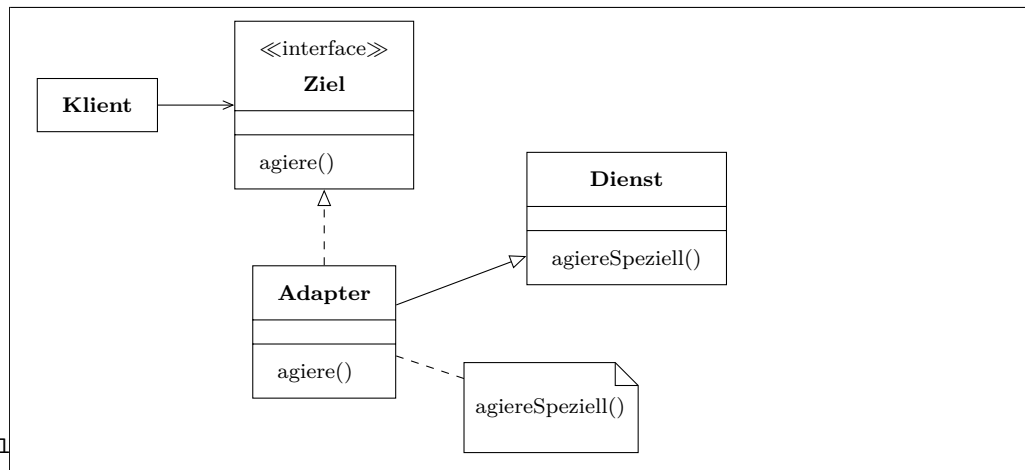
\liEntwurfsAbstrakteFabrik

```

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

```

#### 2.11.4 Adapter



\liEntwurfsAdapterUml

```

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

```

\liEntwurfsAdapterAkteure

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

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

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

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

```

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

```

\liEntwurfsAdapterCode

```

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

```

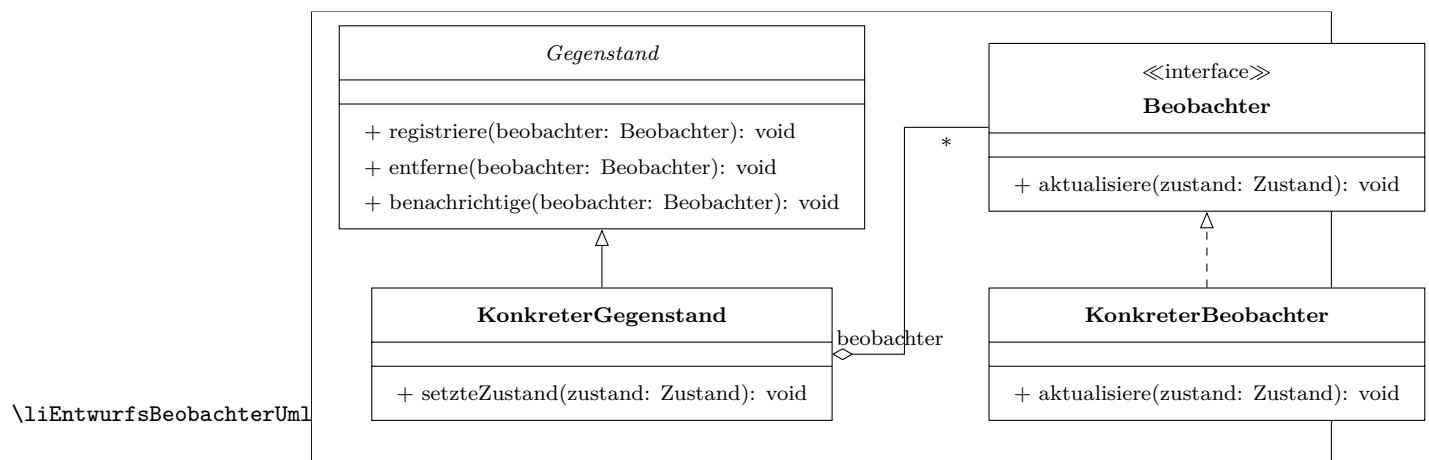
\liEntwurfsAdapter

```

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

```

### 2.11.5 Beobachter (Observer)



```

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

```

```

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

```

\liEntwurfsBeobachterAkteure

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

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

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

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

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

```

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

```



```

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

```

\liEntwurfsBeobachterCode

```

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

```

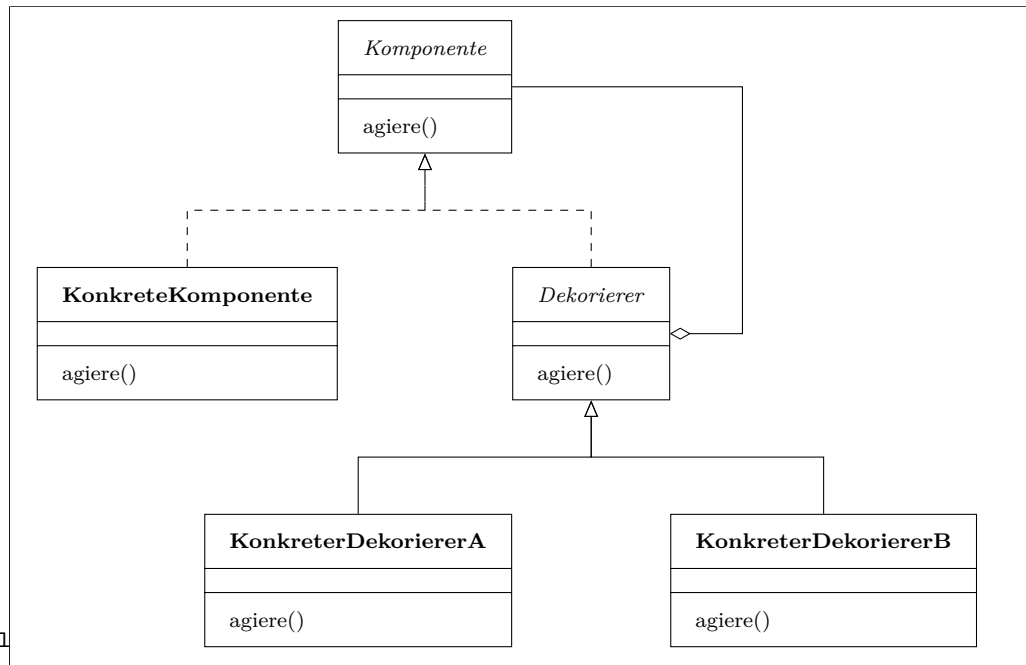
\liEntwurfsBeobachter

```

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

```

## 2.11.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

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

```

```

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

```

\liEntwurfsDekoriererCode

```

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

```

\liEntwurfsDekorierer

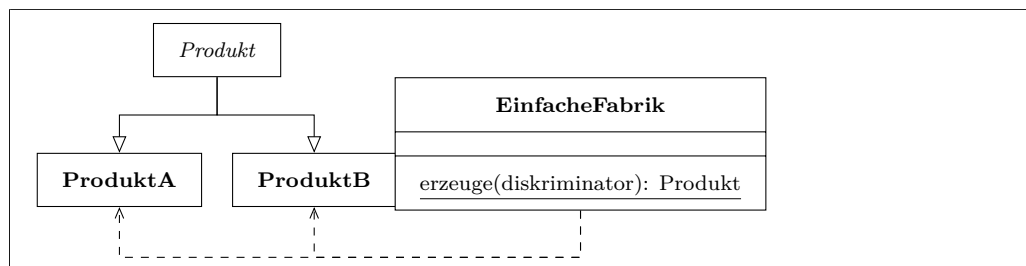
```

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

```

### 2.11.7 Einfache Fabrik (Simple Factory)

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



```

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

```

ntwurfsEinfacheFabrikAkteure Quelle: frei überstetzt aus GoF

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

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

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

```

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

\liEntwurfsEinfacheFabrik

```

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

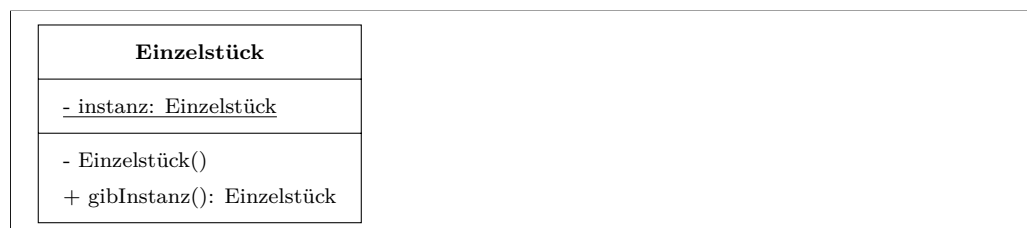
### 2.11.8 Einzelstück (Singleton)

urfsEinzelstueckBeschreibung

```

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

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```

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

iEntwurfsEinzelstueckAkteure Quelle: frei überstetzt aus GoF

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

```

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

```

\liEntwurfsEinzelstueckCode

```

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

```

\liEntwurfsEinzelstueck

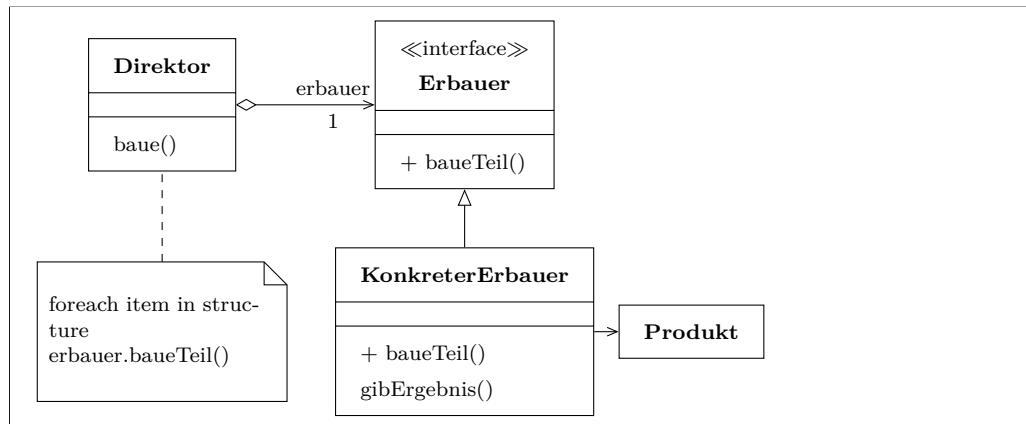
```

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

```

### 2.11.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

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

```

```

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

```

\liEntwurfsErbauerAkteure Quelle: deutsche Wikipedia

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

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

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

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

```

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

```

\liEntwurfsErbauer

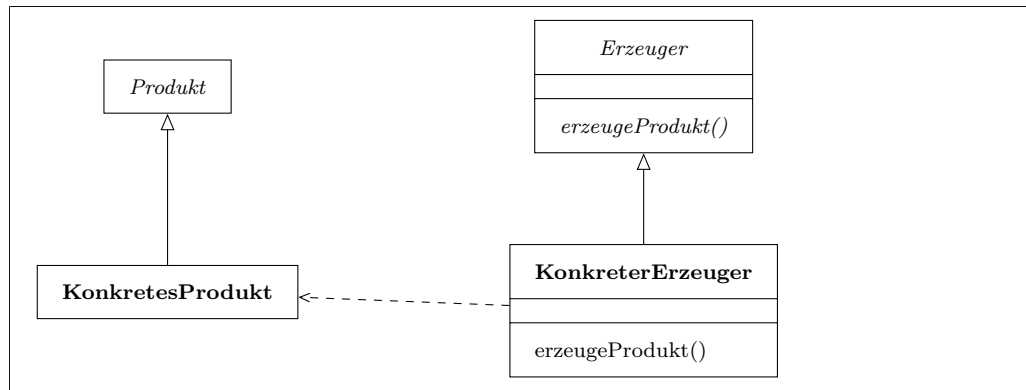
```

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

```

## 2.11.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

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

```

EntwurfsFabrikmethodeAkteure Quelle: deutsche Wikipedia

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

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

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

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

```

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

```

```

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

```

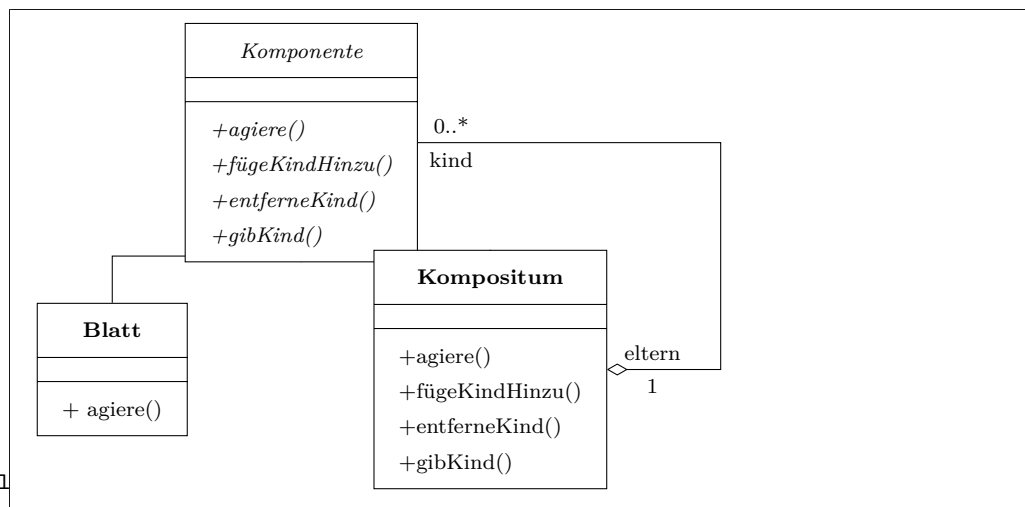
\liEntwurfsFabrikmethode

```

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

```

### 2.11.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

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

```

\liEntwurfsFabrikmethode

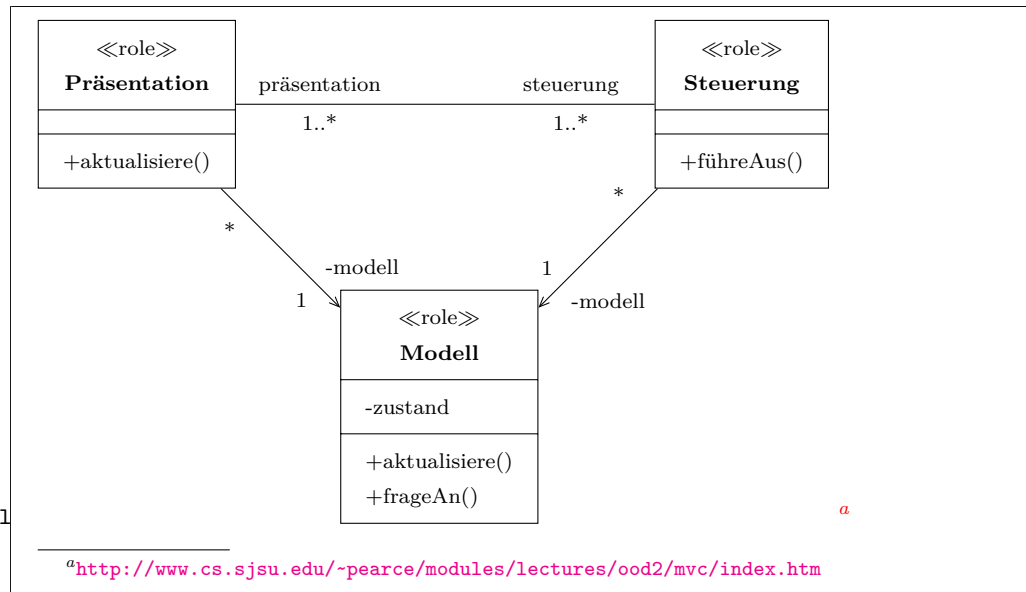
```

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

```

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

ModellPraesentationSteuerungUml



```

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

```

ModellPraesentationSteuerung

```

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

```

### 2.11.13 Stellvertreter (Proxy)

\liEntwurfsStellvertreterUml

```

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

```

\liEntwurfsStellvertreterCode



```

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

```

\liEntwurfsStellvertreter

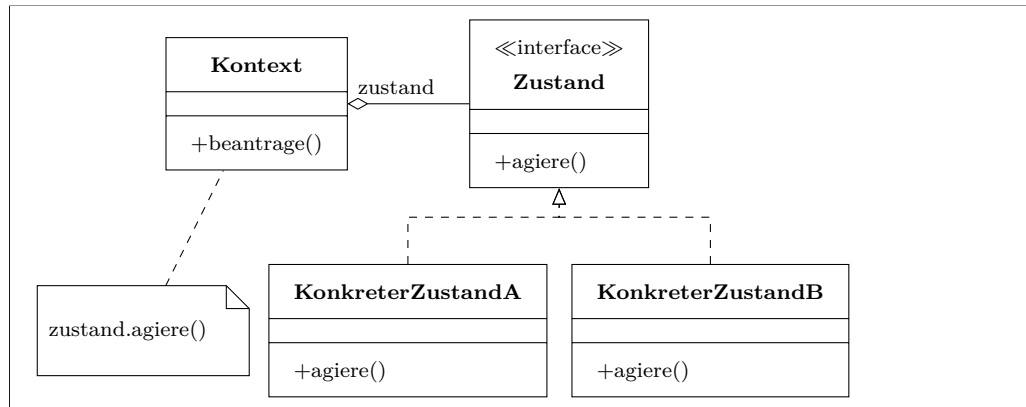
```

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

```

#### 2.11.14 Zustand (State)

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



```

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

```

\liEntwurfsZustandAkteure Quelle: Deutsche Wikipedia

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

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

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

```

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

```

```

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

\liEntwurfsZustandCode

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

\liEntwurfsZustand

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

1247

```

## 2.12 er.sty

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

```
1251 \RequirePackage{tikz-er2}
1252 \usetikzlibrary{positioning}
```

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

```
% \begin{tikzpicture}[er2,scale=0.7,transform shape]
% % Person
% \node[entity] (Person) {Person};
% \node[attribute,right=1cm of Person] {\key{E-Mail}} edge (Person);
% \node[multi attribute,above left=1cm of Person] {Vornamen} edge (Person);
% \node[attribute,left=1cm of Person] {Nachnamen} edge (Person);
% \node[attribute,above right=1cm of Person] {Geburtsdatum} edge (Person);
%
% % Kunde
% \node[entity,below left=1cm of Person] (Kunde) {Kunde};
%
% % Händler
% \node[entity,below right=1cm of Person] (Händler) {Händler};
%
% \node[specialization,below=0.2cm of Person]{is-a}
%   edge (Kunde) edge (Händler) edge (Person);
%
% % Transaktion
% \node[relationship,below=2cm of Person] (Transaktion) {Transaktion}
%   edge node[auto]{1} (Kunde)
%   edge node[auto]{1} (Händler);
% \node[attribute,below=1cm of Transaktion] {Preis} edge (Transaktion);
% \node[attribute,left=1cm of Transaktion,text width=2cm] {Abschlussdatum} edge (Transaktion);
% \node[attribute,right=1cm of Transaktion] {Lieferadresse} edge (Transaktion);
%
% % Zahlungsmittel
% \node[entity,below=4cm of Händler] (Zahlungsmittel) {Zahlungsmittel}
%   edge node[auto]{1} (Transaktion);
% \node[attribute,right=1cm of Zahlungsmittel] {Inhaber} edge (Zahlungsmittel);
%
% % Bankverbindung
% \node[entity,below left=1cm and 0cm of Zahlungsmittel] (Bankverbindung) {Bankverbindung}
% \node[attribute,below left=1cm of Bankverbindung] {\key{IBAN}} edge (Bankverbindung);
%
% % Kreditkarte
% \node[entity,below right=3cm and -2cm of Zahlungsmittel]
% (Kreditkarte) {Kreditkarte};
%
% \node[attribute,below left=1cm of Kreditkarte]
% {\key{Nummer}} edge (Kreditkarte);
%
% \node[attribute,below right=1cm of Kreditkarte,text width=2cm]
% {Ablaufdatum} edge (Kreditkarte);
%
% \node[attribute,right=1cm of Kreditkarte]
% {Anbieter} edge (Kreditkarte);
%
% \node[generalization,below=0.8cm of Zahlungsmittel]{is-a}
%   edge (Zahlungsmittel) edge (Bankverbindung) edge (Kreditkarte);
%
```

```

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

```

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

```

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

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

```

```

1253 \RequirePackage{soul}
1254 \RequirePackage{fontawesome}

```

### 2.12.3 Makro-Kürzel

```

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

```

```

1255 \ExplSyntaxOn

```

**\liErEntity**

```

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

```

**\liErRelationship**

```

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

```

**\liErAttribute**

```

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

```

**\liErMpEntity**

```

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

```

```

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

```

□

**\liErMpRelationship**

```

Let-Abkürzung: \let\r=\liErMpRelationship

```

```

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

```

```

\liErMpAttribute Let-Abkürzung: \let\a=\liErMpAttribute
1271 \def\liErMpAttribute#1{
1272   \liErAttribute{#1}
1273   \marginpar{
1274     \liErAttribute{\tiny\faCircleThin{}}~A:~#1}
1275   }
1276 }

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

1282 \ExplSyntaxOff
1283

```

## 2.13 formale-sprachen.sty

```

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

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

1290 \RequirePackage{hyperref}

1291 \liLadePakete{mathe,typographie}

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

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

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

1300 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

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

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

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

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

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

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

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

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

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

\liZustandsBuchstabe

1316 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1317 \def\liZustandsBuchstabeGross{Z}

```

\liZustandsmengeNr

```

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

```

\liZustandsmengeNrGross

```

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

```

\liZustandsname

```

\liZustandsname{1}: $Z_1$
1327 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

```

\liZustandsnameGross

```

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

```

\liAbleitung

```

\liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1329 \def\liAbleitung#1{\directlua{formale_sprachen.formatiere_ableitung('#1')}}}

```

liProduktionsRegeln

```

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

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

```

\liProduktionen

```

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

```

\liZustandsnameTiefgestellt

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

```

1343 \def\liZustandsnameTiefgestellt#1{
1344   \ifmmode
1345     \liZustandsBuchstabe\sb{#1}
1346   \else
1347     $\liZustandsBuchstabe\sb{#1}$
1348   \fi
1349 }

```

```

1350 \ExplSyntaxOn

```

\liAusdruck

```

\liAusdruck[L_2]{a_1,a_2,\dots,a_n}{n \in N}:  $L_2 = \{a_1, a_2, \dots, a_n \mid n \in N\}$ 
  Ohne „=:“: \liAusdruck[] {x}{y}:  $\{x \mid y\}$ 
  Regulärer Ausdruck zum Konvertieren:

```

```

% \$(.*) += +\\{ *(.*) ( *\\, *)?\\( *\\, *)?(.*) *\\}\\$
% \\liAusdruck[$1]{$2}{$5}
%

1351 \NewDocumentCommand{ \liAusdruck } { 0{L} m m } {
1352   $
1353   \tl_if_empty:nTF {#1} {} {#1 =}
1354   \{
1355     \, #2 \,
1356     |
1357     \, #3 \,
1358   \}$
1359 }
1360 \ExplSyntaxOff

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

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

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



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



1372 \ExplSyntaxOn
1373 \NewDocumentCommand {\liGrammatik} { 0{G} m } {
1374   \tl_set:Nn \l_variablen_tl {V}
1375   \tl_set:Nn \l_alphabet_tl {\Sigma}
1376   \tl_set:Nn \l_produktionen_tl {P}
1377   \tl_set:Nn \l_start_tl {S}
1378
1379   \keys_define:nn { grammatik } {
1380     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1381     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1382     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1383     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1384   }
1385
1386   \keys_set:nn { grammatik } { #2 }
1387
1388   $#1 = (
1389     \l_variablen_tl,
1390     \l_alphabet_tl,
1391     \l_produktionen_tl,
1392     \l_start_tl

```



```
1393    )$  
1394 }  
1395 \ExplSyntaxOff  
  
1396
```

## 2.14 formatierung.sty

```
1397 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1398 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

### 2.14.1 Schriftarten / Typographie

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

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

### 2.14.2 Farben

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

### 2.14.3 Überschriften

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

### 2.14.4 Listen

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

### 2.14.5 Kasten

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

liKasten

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

### 2.14.6 Header

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

### 2.14.7 Zeilenabstände

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

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

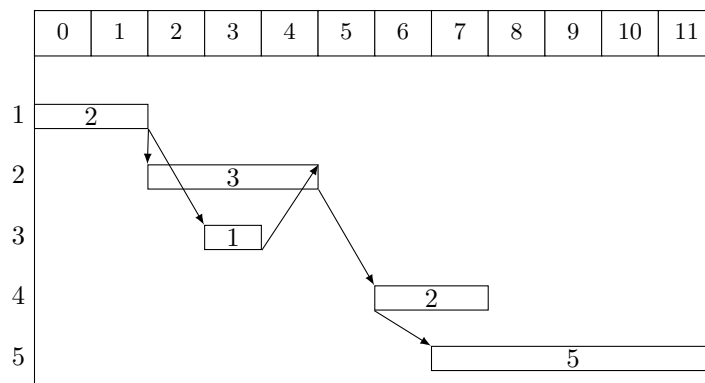


## 2.15 gantt.sty

```

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

% \begin{ganttchart}[x unit=0.75cm, y unit chart=0.8cm]{0}{11}
% \gantttitlelist{0,...,11}{1} \\
% \ganttbar[name=1]{1}{0}{1} \\
% \ganttbar[name=2]{2}{2}{4} \\
% \ganttbar[name=3]{3}{3}{3} \\
% \ganttbar[name=4]{4}{6}{7} \\
% \ganttbar[name=5]{5}{7}{11}
%
% \node at (1) {2};
% \node at (2) {3};
% \node at (3) {1};
% \node at (4) {2};
% \node at (5) {5};
%
% \ganttlink[link type=f-f]{3}{2}
% \ganttlink[link type=f-s]{1}{2}
% \ganttlink[link type=f-s]{1}{3}
% \ganttlink[link type=f-s]{2}{4}
% \ganttlink[link type=s-s]{4}{5}
% \end{ganttchart}
%
```



```

1437 \RequirePackage{tikz-uml}
1438 \RequirePackage{pgfgantt}
1439 \setganttlinklabel{f-s}{}
1440 \setganttlinklabel{s-s}{}
1441 \setganttlinklabel{f-f}{}
1442 \setganttlinklabel{s-f}{}
1443
```

## 2.16 grafik.sty

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

\liGrafikLogoPfad

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

\liGrafikCCLizenz

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

\liGrafikLogo

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

1463 \ExplSyntaxOff
1464
```

## 2.17 graph.sty

```

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

```

Für die die Adjazenzliste (xrightarrow)

```

1468 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

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

```

1469 \RequirePackage{blkarray}

```

```

1470 \usetikzlibrary{arrows.meta}

```

```

% \begin{tikzpicture}[li graph]
% \node (a) at (0,0) {a};
% \node (b) at (1,1) {b};
% \node (c) at (4,1) {c};
% \node (d) at (3,0) {d};
% \node (e) at (3,3) {e};
%
% \path[->] (a) edge (b);
% \path (b) edge (d);
% \path[->] (b) edge node {3} (e);
% \path (c) edge (d);
% \path (d) edge node {4} (a);
% \path[->] (d) edge node {2} (e);
% \end{tikzpicture}
%

```



```

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

```

**liGraphenFormat** Dummy-Umgebung, zum Parsen durch Java gedacht.

```

% \begin{liGraphenFormat}
% a: 0 0
% b: 1 1
% c: 4 1
% d: 3 0
% e: 2 2
% a -> b
% b -- d
% b -> e: 3
% c -- d
% d -> e: 2
% d -- a: 4
% \end{liGraphenFormat}
%

1492 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}
1493

```

## 2.18 hanoi.sty

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

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

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

\liHanoi \liHanoi{anzahl-scheiben}{gewicht-scheibe/turm-nr,gewicht-scheibe/turm-nr},
z. B.: \liHanoi{4}{4/1,3/1,2/3,1/2}

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

1531
```



## 2.19 klassen-konfiguration-aufgabe.sty

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

Formatierung muss vor literatur sein, sonst Option clash

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

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

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

In Aufgaben wollen wir das Additum sehen.

```
1547 \LiADDITUMtrue
1548 \ExplSyntaxOn

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

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

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

`\liAufgabenMetadaten`

```
1569 \def\liAufgabenMetadaten#1{
1570   \liMetaSetze{#1}
1571
1572   \tl_if_empty:NTF \g_auf_examen_nummer_tl {} {
1573     {
1574       \noindent
1575       \large
1576       \gib_einzelpruefung_trenner:
1577       \par\medskip
1578     }
1579   }
1580
1581   {
1582     \noindent
1583     \bfseries
1584     \Large
1585     \tl_if_empty:NTF \g_auf_examen_nummer_tl {
```

```

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

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

1628 \ExplSyntaxOff
1629

```

## 2.20 klassen-konfiguration-examen.sty

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

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

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

Formatierung für die Überschriften setzen.

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

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

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

`\liSetzeExamenThemaNr`

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

`\liSetzeExamenTeilaufgabeNr`

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

`\liBindeAufgabeEin`

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

`\liAufgabenMetadaten` Das Metadaten-Makro überschreiben

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

```

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

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

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

```

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

## 2.21 komplexitaetstheorie.sty

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

### 2.21.1 Makro-Kürzel

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

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

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

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

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

```
1761 \def\liStrich#1{#1'\prime}
```

`\liProblemName` Zu Setzen von Problemnamen wie zum Beispiel SAT, COL, VERTEX COVER

**Let-Abkürzung:** `\let\n=\liProblemName`

`\liProblemName: SAT VERTEX COVER`

```
1762 \def\liProblemName#1{\texttt{\textsc{#1}}}
```

`\liProblemBeschreibung` Zu setzen von Problem-Beschreibungen:

```
% \liProblemBeschreibung
% {}
% {}
% {}
%
```

<p><b>CLIQUE</b></p> <p><b>Gegeben:</b> Ein ungerichteter Graph <math>G = (V, E)</math>, eine Zahl <math>k \in \mathcal{N}</math></p> <p><b>Frage:</b> Gibt es eine Menge <math>S \subseteq V</math> mit <math> S  = k</math>, sodass für alle Knoten <math>u \neq v \in V</math> gilt, dass <math>\{u, v\}</math> eine Kante in <math>E</math> ist?</p>
--

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

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

```

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

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

\liProblemVertexCover

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

\liProblemVertexCover

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

\liProblemSubsetSum Kein Karp-21-Problem

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

\liProblemSubsetSum Kein Karp-21-Problem

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

1826

```

## 2.22 kontrollflussgraph.sty

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

1828 \ProvidesPackage{lehramt-informatik-kontrollflussgraph}[2020/11/07]

### 2.22.1 Makro-Kürzel

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

### 2.22.2 TeX-Markup-Beispiel

```
% \begin{liKontrollflussgraph}[xscale=1,yscale=-1.6]
% \node[knoten] at (0,0) (S) {S};
%
% \node[pin=\c{boolean resultat = false;}] at (0,1) (2) {2};
% \node[pin=\c{if (wort != null)}] at (0,2) (3) {3};
% \node[pin=\c{int laenge = wort.length;}] at (1,3) (4) {4};
% \node[pin=\c{if (laenge >= 2)}] at (1,4) (5) {5};
% \node[pin=\c{resultat = true; int i = 0;}] at (2,5) (6) {6};
% \node[pin=\c{for (i < laenge / 2;)}] at (2,6) (7) {7};
% \node[pin=\c{char c1; char c2 ...}] at (3,7) (8) {8};
% \node[pin=\c{if (Char...) }] at (2,8) (10) {10};
% \node[pin=\c{resultat = false; break;}] at (2,9) (12) {12};
% \node[pin=\c{if (laenge == 1)}] at (1,10) (17) {17};
% \node[pin=\c{resultat = true;}] at (1,11) (18) {18};
% \node[pin=180:\c{return resultat;}] at (-1,9) (22) {22};
% \node[knoten] at (-1,10) (E) {E};
%
% \path (S) -- (2);
% \path (2) -- (3);
% \path[wahr] (3) -- (4) \b{right}{wort != null};
% \path[falsch] (3) -- (22) \b{left}{wort == null};
% \path (4) -- (5);
% \path[wahr] (5) -- (6) \b{right}{laenge >= 2};
% \path[falsch] (5) -- (17) \b{left,rotate=70,pos=0.2}{laenge < 2};
% \path (6) -- (7);
% \path[wahr] (7) -- (8) \b{right}{i < laenge / 2};
% \path[falsch] (7) -- (22) \b{left}{i >= laenge / 2};
% \path (8) -- (10);
% \path[wahr] (10) -- (12) \b{right}{c != c};
% \path[falsch] (10) -- (7) \b{left,rotate=70,pos=0.8}{c == c};
% \path (12) -- (22);
% \path[wahr] (17) -- (18) \b{right}{laenge == 1};
% \path[falsch] (17) -- (22) \b{right,rotate=-20,pos=0.99}{laenge != 1};
% \path (18) -- (22);
% \path (22) -- (E);
% \end{liKontrollflussgraph}
%
```

### 2.22.3 TikZ: pin

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

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



```

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

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

```

## 2.22.4 Umgebungen

liKontrollflussgraph

```
1887 \NewDocumentEnvironment { liKontrollflussgraph } { 0{} } {
1888   \begin{tikzpicture}[
1889     li kontrollfluss,
1890     #1
1891   ]
1892 } {
1893   \end{tikzpicture}
1894 }
```

## 2.22.5 Makros

\liAnweisung

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

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

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

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

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

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

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

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

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

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

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

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

```
1901 \ExplSyntaxOn
1902 \NewDocumentCommand { \liKontrollKnotenPfad }{ m }
1903 {
1904   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
1905   \seq_set_map:NNn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
1906   \seq_use:Nn \l_tmpa_seq { ~~~~ }
1907 }
1908 \ExplSyntaxOff
1909
```

## 2.23 kopf-fusszeilen.sty

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

1913 \ExplSyntaxOn

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

1924 \ExplSyntaxOff

1925
```

## 2.24 literatur-dummy.sty

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

\literatur

1928 \def\literatur{}

\footcite

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

1931
```

## 2.25 literatur.sty

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

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

\literatur

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

1953
```

## 2.26 makros.sty

```
1954 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1955 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1956 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1957 anderen Paket passen]
```

```
1958 \RequirePackage{hyperref}
```

```
1959 \RequirePackage{graphicx}
```

Für die Umgebung `liQuellen` benötigt.

```
1960 \RequirePackage{paralist}
```

Damit wir Umgebungen ausblenden können. Liefert `\comment` und `\endcomment`  
<https://tex.stackexchange.com/a/37945>

```
1961 \RequirePackage{verbatim}
```

```
1962 \ExplSyntaxOn
```

`\inhaltsverzeichnis`

```
1963 \def\inhaltsverzeichnis {
1964   \begin{mdframed}
1965     \begingroup
1966     \let\clearpage\relax
1967     \tableofcontents
1968     \endgroup
1969   \end{mdframed}
1970 }
```

`\memph` `\mephm` (`\marginpar` and `\emph`)

```
1971 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}
```

`\SLASH`

```
1972 \newcommand\SLASH{\char`\\}
```

`\liPseudoUeberschrift` Text, der sich wie eine Überschrift verhält.

```
1973 \newcommand{\liPseudoUeberschrift}[1]{
1974   \bigskip
1975
1976   \par
1977   \noindent
1978   \textbf{#1}
1979
1980   \medskip
1981
1982   \par
1983   % Keine Einrückung
1984   \@afterindentfalse
1985   \@afterheading
1986 }
```

`\liBeschriftung` Ähnlich dem Makro `\liPseudoUeberschrift`. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.

```
1987 \newcommand{\liBeschriftung}[1]{
1988   \par
1989   \noindent
1990   \medskip
1991   \textbf{#1}:
1992   \medskip
1993   \noindent
1994 }
```

`\hinweis`

```
1995 \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.
1996 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

liEinbettung
1997 \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.
1998 \NewDocumentEnvironment{ liAntwort } { O{standard} }
1999 {
2000   \ifLiANTWORT
2001     \str_case:nn {#1} {
2002       {standard} {
2003         \def\beschriftung{}
2004         \mdfsetup{backgroundcolor=gray!1,linecolor=gray}
2005       }
2006       {richtig} {
2007         \def\beschriftung{richtig}
2008         \mdfsetup{backgroundcolor=green!5,linecolor=green}
2009       }
2010       {falsch} {
2011         \def\beschriftung{falsch}
2012         \mdfsetup{backgroundcolor=red!3,linecolor=red}
2013       }
2014       {muster} {
2015         \def\beschriftung{Musterlösung}
2016         \mdfsetup{backgroundcolor=green!12,linecolor=green}
2017       }
2018     }
2019     \ifx\beschriftung\empty\else
2020       \noindent
2021       \textbf{\beschriftung{:}}
2022     \fi
2023     \begin{mdframed}
2024   \else
2025     \fi
2026 }
2027 {
2028   \ifLiANTWORT
2029     \end{mdframed}
2030   \else
2031     \fi
2032 }

liAdditum Zusätzliches Material bei Aufgaben, das zum Lösen der Aufgaben nicht unbedingt nötig
ist, z. b. Hintergrundinformation, Test-Dateien, komplette Code-Dateien.
2033 \NewDocumentEnvironment{ liAdditum } { o }
2034 {
2035   \ifLiADDITUM\else
2036     % das Verwenden von +b hat nicht funktioniert.
2037     \expandafter\comment % https://tex.stackexchange.com/a/37945
2038   \fi
2039   \begin{mdframed}[backgroundcolor=yellow!5]
2040     \IfNoValueTF {#1}
2041     {
2042       \liPseudoUeberschrift{Additum}
2043     }
2044     {
2045       \liPseudoUeberschrift{Additum:~#1}
2046     }

```

```

2047 } {
2048   \end{mdframed}
2049   \ifLiADDITUM\else
2050     \expandafter\endcomment % https://tex.stackexchange.com/a/37945
2051   \fi
2052 }

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.

```

2053 \NewDocumentEnvironment{ liExkurs }{o +b}{
2054   \ifLiEXKURS
2055     \vspace{0.2cm}%
2056     \begin{mdframed}[
2057       backgroundcolor=white,
2058       bottomline=false,
2059       innermargin=1cm,
2060       leftline=true,
2061       linecolor=black,
2062       linewidth=0.1cm,
2063       outermargin=1cm,
2064       rightline=false,
2065       topline=false,
2066     ]
2067     \footnotesize
2068     \noindent%
2069     \textbf{Exkurs:~#1}\par%
2070     \noindent%
2071     #2
2072   \end{mdframed}
2073   \vspace{0.2cm}
2074   \else
2075     \fi
2076 }{}

```

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

```

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

```

#### Weiterführende Literatur:

- Quelle 1
- Quelle 2

```

2077 \cs_new:Npn \listen@punkt #1 {\item #1}
2078 \NewDocumentEnvironment { liQuellen }{ +b }
2079 {
2080   \seq_clear_new:N \l_quellen
2081   \seq_set_split:Nnn \l_quellen {\item} {#1}
2082   \seq_remove_all:Nn \l_quellen {}
2083   \begin{mdframed}[backgroundcolor=white!0]

```



```

2084 \footnotesize
2085 \noindent
2086 \textsf{\textbf{Weiterführende~Literatur:}}
2087 \medskip
2088 \begin{compactitem}
2089 \seq_map_function:NN \l_quellen {\listen@punkt}
2090 \end{compactitem}
2091 \end{mdframed}
2092 %
2093 \par
2094 \@afterindentfalse
2095 \@afterheading
2096 } {}

liLernkartei

2097 \NewDocumentEnvironment { liLernkartei }{ m +b }
2098 {
2099 \begin{mdframed}
2100 \footnotesize
2101 \noindent%
2102 \textbf{Lernkarteikarte:~#1}\par%
2103 \noindent%
2104 #2
2105 \end{mdframed}
2106 } {}

liDiagramm \begin{liDiagramm}{beschriftung}\end{liDiagramm}: Zu setzen einer Graphik bzw
eines Diagramms.
2107 \NewDocumentEnvironment { liDiagramm }{ m +b }
2108 {
2109 \begin{mdframed}[backgroundcolor=white!0]
2110 \small
2111 \noindent%
2112 \textit{#1}:
2113 \begin{center}
2114 #2
2115 \medskip
2116 \end{center}
2117 \end{mdframed}
2118 } {}

\liFussnoteUrl \liFussnoteUrl[zusätzlicher-text]{url} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
2119 \NewDocumentCommand{\liFussnoteUrl} { o m } {
2120 \footnote{\url{#2}\IfNoValueTF{#1}{-}{ ( #1 ) }}
2121 }
2122

\liFussnoteLink \liFussnoteLink[zusätzlicher-text]{link-text}{url} \liFussnoteLink[zusätzlicher Text]{tex
Einen Link, d. h. einen Link-Text und eine URL als Fußnote setzen.
2123 \NewDocumentCommand{\liFussnoteLink} { o m m } {
2124 \footnote{\href{#3}{#2}\IfNoValueTF{#1}{-}{ ( #1 ) }}
2125 }

\zB
2126 \def\zB{z.\,B. }

\ZB
2127 \def\ZB{Z.\,B. }

\dh
2128 \def\dh{d.\,h. }

```

2129 \ExplSyntaxOff

2130

## 2.27 master-theorem.sty

2131 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2132 \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}

%

2133 \ExplSyntaxOn

2134 \RequirePackage{amsmath}

\liRundeKlammer

2135 \def\liRundeKlammer#1{

2136 \negthinspace \left( #1 \right)

2137 }

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

2138 \def\liThetaOhneMathe#1{

2139 \Theta \liRundeKlammer{#1}

2140 }

2141 \def\liTheta#1{

2142 \ifmmode

2143 \liThetaOhneMathe{#1}

```

2144 \else
2145     $\liThetaOhneMathe{#1}$
2146 \fi
2147 }

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
2148 \def\liOmegaOhneMathe#1{
2149     \Omega \liRundeKlammer{#1}
2150 }
2151 \def\liOmega#1{
2152     \ifmmode
2153         \liOmegaOhneMathe{#1}
2154     \else
2155         $\liOmegaOhneMathe{#1}$
2156     \fi
2157 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
2158 \def\liOOhneMathe#1{
2159     \mathcal{O} \liRundeKlammer{#1}
2160 }
2161 \def\liO#1{
2162     \ifmmode
2163         \liOOhneMathe{#1}
2164     \else
2165         $\liOOhneMathe{#1}$
2166     \fi
2167 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
2168 \def\liTOhneMathe#1#2{
2169     \tl_if_blank:nTF {#1}
2170     {}
2171     {#1 \cdot }
2172     T
2173     \liRundeKlammer{\frac{n}{#2}}
2174 }
2175 \def\liT#1#2{
2176     \ifmmode
2177         \liTOhneMathe{#1}{#2}
2178     \else
2179         $\liTOhneMathe{#1}{#2}$
2180     \fi
2181 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
2182 \def\liRekursionsGleichung{
2183     $T(n) = \liT{a}{b} + f(n)$
2184 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
2185 \def\liBedingungEins{
2186     $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
2187 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
2188 \def\liBedingungZwei{
2189     $f(n) \in \liTheta{n^{\log\sb{b}a}}$
2190 }

```

```

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \epsilon})$ 
2191 \def\liBedingungDrei{
2192    $f(n) \in \Omega(n^{\log_{\sb{b}} a + \varepsilon})$ 
2193 }

2194 \ExplSyntaxOff

\liMasterVariablen
2195 \def\liMasterVariablen{
2196   \begin{displaymath}
2197     T(n) = \liT{a}{b} + f(n)
2198   \end{displaymath}
2199
2200   \begin{itemize}
2201     \item[ $a =$ ]
2202       Anzahl der rekursiven Aufrufe, Anzahl der Unterprobleme in der
2203       Rekursion
2204       ( $a \geq 1$ ).
2205
2206     \item[ $\textstyle\frac{1}{b} =$ ]
2207       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
2208       repräsentiert wird, Anteil an der Verkleinerung des Problems ( $b > 1$ ).
2209
2210     \item[ $f(n) =$ ]
2211       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
2212       die Kombination der Teillösungen entstehen. Eine von  $T(n)$ 
2213       unabhängige und nicht negative Funktion.
2214   \end{itemize}
2215   \footcite{wiki:master-theorem}
2216   \footcite[Seite 19-35]{aud:fs:2}
2217 }

\liMasterFaelle
2218 \def\liMasterFaelle{
2219   \begin{description}
2220     \item[1. Fall:]
2221        $T(n) \in \liTheta{n^{\log_{\sb{b}} a}}$ 
2222
2223       \hfill falls \liBedingungEins
2224       für  $\varepsilon > 0$ 
2225
2226     \item[2. Fall:]
2227        $T(n) \in \liTheta{n^{\log_{\sb{b}} a} \cdot \log n}$ 
2228
2229       \hfill falls \liBedingungZwei
2230
2231     \item[3. Fall:]
2232        $T(n) \in \liTheta{f(n)}$ 
2233
2234       \hfill falls \liBedingungDrei
2235       für  $\varepsilon > 0$ 
2236       und ebenfalls für ein  $c$  mit  $0 < c < 1$  und alle hinreichend großen  $n$ 
2237       gilt:
2238        $a \cdot f(\textstyle\frac{n}{b}) \leq c \cdot f(n)$ 
2239     \end{description}
2240 }

\liMasterVariablenDeklaration
2241 \def\liMasterVariablenDeklaration#1#2#3{
2242   \begin{description}
2243     \item[Allgemeine Rekursionsgleichung:] \strut
2244
2245     \liRekursionsGleichung

```

```

2246
2247 \item[Anzahl der rekursiven Aufrufe ( $a_n$ ):] \strut
2248
2249 #1
2250
2251 \item[Anteil Verkleinerung des Problems ( $b_n$ ):] \strut
2252
2253 um  $\frac{1}{n}$  also  $b_n = \frac{1}{n}$ 
2254
2255 \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut
2256
2257  $\Theta(n^3)$ 
2258
2259 \item[Ergibt folgende Rekursionsgleichung:] \strut
2260
2261  $T(n) = T(n-1) + \frac{1}{n}$ 
2262 \end{description}
2263 }

```

\liMasterFallRechnung

```

2264 \def\liMasterFallRechnung#1#2#3{
2265 \begin{description}
2266 \item[1. Fall:] \liBedingungEins:
2267
2268 #1
2269
2270 \item[2. Fall:] \liBedingungZwei:
2271
2272 #2
2273
2274 \item[3. Fall:] \liBedingungDrei:
2275
2276 #3
2277 \end{description}
2278 }

```

\liMasterExkurs

```

2279 \def\liMasterExkurs{
2280 \begin{liExkurs}[Master-Theorem]
2281 \liMasterVariablen
2282
2283 \noindent
2284 Dann gilt:
2285
2286 \liMasterFaelle
2287 \end{liExkurs}
2288 }

```

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)

```

2289 \def\liMasterWolframLink#1{
2290 Berechne die Rekursionsgleichung auf WolframAlpha:
2291 \href{https://www.wolframalpha.com/input/?i=#1}{WolframAlpha}
2292 }
2293

```

## 2.28 mathe.sty

```
2294 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2295 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
2296
2297 % for example \ltimes \rtimes
2298 %\RequirePackage{amssymb}
2299 \RequirePackage{amsmath}
2300
2301 %%
2302 % \mlq \mrq
2303 %%
2304 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
2305 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
2306
```

## 2.29 meta.sty

```
2307 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2308 \ProvidesPackage{lehramt-informatik-klassen-meta}[2021/09/10 Sammlung
2309 von Textschnipseln, die das Projekt beschreiben]
```

```
2310 \ExplSyntaxOn
```

```
2311 \liLadePakete{grafik}
```

### 2.29.1 Einfache Makros (Low level)

```
\liMetaBschlangaulSammlung
```

```
2312 \def\liMetaBschlangaulSammlung{
2313   Die~Bschlangaul-Sammlung
2314 }
```

```
HermineBschlangaulAndFriends
```

```
2315 \def\liMetaHermineBschlangaulAndFriends{
2316   Hermine~Bschlangaul~and~Friends
2317 }
```

```
\liMetaUeberDasProjekt
```

```
2318 \def\liMetaUeberDasProjekt{
2319   Eine~freie~Aufgabensammlung~mit~Lösungen~
2320   von~Studierenden~für~Studierende~
2321   zur~Vorbereitung~auf~die~1.~Staatsexamensprüfungen~
2322   des~Lehramts~Informatik~in~Bayern.
2323 }
```

```
\liMetaCCLink
```

```
2324 \def\liMetaCCLink{
2325   Diese~Materialsammlung~unterliegt~den~Bestimmungen~der~
2326   \href{
2327     https://creativecommons.org/licenses/by-nc-sa/4.0/deed.de
2328   }
2329   {
2330     Creative~Commons~Namensnennung~Nicht~kommerziell~Share~Alike~4.0~
2331     International~Lizenz
2332   }.
2333 }
```

```
\liMetaEmailLink
```

```
2334 \def\liMetaEmailLink{
2335   \href{
2336     mailto:hermine.bschlangaul@gmx.net
2337   }{
2338     hermine.bschlangaul@gmx.net
2339   }
2340 }
```

```
\liMetaHilfMit
```

```
2341 \def\liMetaHilfMit{
2342   Hilf~mit!~
2343
2344   Die~Hermine~schafft~das~nicht~alleine!~
2345
2346   Das~ist~ein~Community-Projekt.~
2347
2348   Verbesserungsvorschläge,~Fehlerkorrekturen,~weitere~Lösungen~sind~
2349   herzlich~willkommen~~~egal~wie~~~per~Pull-Request~oder~per~E-Mail~an~
2350   \liMetaEmailLink.
2351 }
```



`\liMetaHilfMit`

```
2352 \def\liMetaQuelltext{
2353   Der~\TeX-Quelltext~dieses~Dokuments~kann~unter~folgender~
2354   URL~aufgerufen~werden:~
2355 }
```

## 2.29.2 Zusammengesetzte Makros (High level)

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

```
2356 \cs_new:Npn \logo_dann_text:nn #1 #2 {
2357   \begin{center}
2358     \begin{minipage}[c]{5.5cm}
2359       #1
2360     \end{minipage}
2361
2362     \begin{minipage}[c]{10cm}
2363       #2
2364     \end{minipage}
2365   \end{center}
2366 }
```

`\liLogoTextProjekt`

```
2367 \def\liLogoTextProjekt
2368 {
2369   \logo_dann_text:nn
2370   {
2371     \liGrafikLogo[width=5cm]
2372   }
2373   {
2374     {
2375       \bfseries
2376       \liMetaBschlangaulSammlung
2377     }
2378     \par
2379
2380     \liMetaHermineBschlangaulAndFriends
2381     \par
2382
2383     \medskip
2384
2385     \begin{spacing}{1}
2386       \footnotesize
2387       \liMetaUeberDasProjekt
2388     \end{spacing}
2389   }
2390 }
```

`\liLogoTextCCLizenz`

```
2391 \def\liLogoTextCCLizenz
2392 {
2393   \logo_dann_text:nn
2394   {
2395     \centerline{\liGrafikCCLizenz[width=3cm]}
2396   }
2397   {
2398     \begin{spacing}{1}
2399       \scriptsize
2400       \liMetaCCLink
2401     \end{spacing}
2402   }
2403 }
```

2404 \ExplSyntaxOff

2405



\liFussnoteDreiText

```
2424 \def\liFussnoteDreiText{
2425   \li@fussnote@text{3}
2426   {In weiteren Iterationen markierte Zustände.}
2427 }
```

\liFussnoteVierText

```
2428 \def\liFussnoteVierText{
2429   \li@fussnote@text{4}
2430   {...}
2431 }
```

\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.
$x_4$	...

```
2432 \def\liFussnoten{
2433   \bigskip
2434
2435   \noindent
2436   \liFussnoteEinsText
2437
2438   \noindent
2439   \liFussnoteZweiText
2440
2441   \noindent
2442   \liFussnoteDreiText
2443
2444   \noindent
2445   \liFussnoteVierText
2446 }
```

\liLeereZelle

```
\liLeereZelle: {}
Let-Abkürzung: \let\l=\liLeereZelle
2447 \def\liLeereZelle{\$\emptyset$}
```

\liZustandsPaarVariablenName

```
2448 \def\liZustandsPaarVariablenName{z}
```

\liZustandsPaar

```
2449 \def\liZustandsPaar#1#2{
2450   $(
2451     \liZustandsPaarVariablenName_#1,
2452     \liZustandsPaarVariablenName_#2
2453   )$
2454 }
```

liUebergangsTabelle

```
2455 \renewcommand{\arraystretch}{1.4}
2456 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
2457   \liPseudoUeberschrift{Übergangstabelle}
2458   \begin{center}
2459     \begin{tabular}{r|l|l}
2460       \textbf{Zustandspaar} & \textbf{#1} & \textbf{#2} \\ \hline
2461     \end{tabular}
2462   \end{center}
2463 }
2464 }
```

iUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle:

### Minimierungstabelle (Table filling)

```
2465 \ExplSyntaxOn
2466 \def\liUeberschriftDreiecksTabelle{
2467   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
2468 }
```

\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. —————

```
2469 \def\liMinimierungErklaerung{
2470   %\footcite[Seite~19]{koenig}
2471   \liParagraphMitLinien{
2472     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
2473     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
2474     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
2475     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
2476      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
2477     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
2478     somit~zueinander~( $k-1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
2479     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
2480     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
2481   }
2482 }
2483 \ExplSyntaxOff
2484
```

## 2.31 normalformen.sty

```

2485 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2486 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10
2487 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
2488 Attributhülle]

```

Mathe für textit tex etc, typographie für geschweifte Klammer

```

2489 \liLadePakete{mathe,typographie}
2490 \directlua{
2491   helfer = require('lehramt-informatik-helfer')
2492   normalformen = require('lehramt-informatik-normalformen')
2493 }

```

### 2.31.1 Makro-Kürzel

```

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

```

```

2494 \def\liTeilen#1{
2495   \directlua{tex.print(normalformen.teilen('#1'))}
2496 }

```

```

\liAttributHuelle Let-Abkürzung: \let\ah=\liAttributHuelle
                  \ah{F, \m{A, B}} AttrHülle( $F, \{A, B\}$ ) Regulärer Ausdruck zum Konvertieren
                  AttrHülle((.*)\ ) \ah{$1}
2497 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(\#1)}
2498 \def\liAttributHuelle#1{
2499   \ifmmode
2500     \liAttributHuelleOhneMathe{\#1}
2501   \else
2502     $\liAttributHuelleOhneMathe{\#1}$
2503   \fi
2504 }

```

```

\liAttributMenge Let-Abkürzung: \let\m=\liAttributMenge
2505 \def\liAttributMenge#1{\{ \textit{\#1} \}}

```

liAHuelle

```

2506 \NewDocumentEnvironment{ liAHuelle } { +b } {
2507   \begingroup
2508   \footnotesize
2509   \begin{multline*}
2510     \#1
2511   \end{multline*}
2512   \endgroup
2513 } { }

```

```

\liLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multline
Let-Abkürzung: \let\ahL=\liLinksReduktion
                  \ahL{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}
2514 \def\liLinksReduktion#1#2#3{
2515   \shoveleft{
2516     \liAttributHuelleOhneMathe{FA,
2517       \liAttributMenge{\#1 \string\ #2}} =
2518   } \\\

```

```

2519 \shoveright{
2520   \liAttributMenge{#3}
2521 } \\\
2522 }

```

```

\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( $F, \{A, E \setminus E\} = \{A, E, F, B, D\}$ )
2523 \def\liLinksReduktionInline#1#2#3{%
2524   {%
2525     \footnotesize%
2526     $\liAttributHuelleOhneMathe{F,
2527       \liAttributMenge{#1 \string\ #2}} =
2528       \liAttributMenge{#3}$
2529   }
2530 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahr=\liLinksReduktionInline
    \ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
2531 \def\liRechtsReduktionInline#1#2#3#4{%
2532   {%
2533     \footnotesize%
2534     $\liAttributHuelleOhneMathe{
2535       F \setminus
2536       \liFunktionaleAbhaengigkeit{#1}
2537       \def\tmp{#2}\ifx\tmp\empty
2538       \else
2539         \cup \liFunktionaleAbhaengigkeit{#2}
2540       \fi
2541       ,
2542       \liAttributMenge{#3}
2543     } =
2544     \liAttributMenge{#4}$
2545   }
2546 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
    \fa{A, B -> C, D}:  $\{A, B\} \rightarrow \{C, D\}$  Regulärer Ausdruck zum Konvertieren:
    \$(.*?) \rightarrow (.*?)\$ \fa{$1 -> $2}
2547 \def\liFunktionaleAbhaengigkeit#1{%
2548   \directlua{normalformen.drucke_funk_abhaengigkeit('#1', false)}%
2549 }

```

```

\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 (.*)\$\$(.*) \\rightarrow
$1 -> $2;
2550 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
2551   \liGeschweifteKlammern
2552   {#1}
2553   {
2554     \begin{align*}
2555       \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
2556     \end{align*}
2557   }
2558   {-0.5cm}
2559   {-1.7cm}
2560 }

\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}
2561 \NewDocumentCommand {\liRelation} { O{R} m } {
2562   $\directlua{
2563     local name = helper.konvertiere_tiefgestellt('#1')
2564     tex.print(name)
2565   }$(\textit{\, #2\,})
2566 }

2567

```



## 2.32 o-notation.sty

2568 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2569 \ProvidesPackage{lehramt-informatik-o-notation}[2021/09/08]

### 2.32.1 Makro-Kürzel

\let\O=\liONotationO

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

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

2570 \ExplSyntaxOn

2571 \RequirePackage{amssymb}

2572 \RequirePackage{pgfplots}

Für echte Teilmenge \subsetneq:  $\subsetneq$

2573 \RequirePackage{amssymb}

\liRundeKlammer

```
2574 \def\liRundeKlammer#1{
2575   \negthinspace \left( #1 \right)
2576 }
```

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

```
2577 \cs_new:Npn \o_notation_O:n #1 {
2578   \mathcal{O} \liRundeKlammer{#1}
2579 }
2580 \def\liONotationO#1{
2581   \ifmmode
2582     \o_notation_O:n { #1 }
2583   \else
2584     $ \o_notation_O:n { #1 } $
2585   \fi
2586 }
```

2587

## 2.33 petri.sty

2588 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2589 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]

### 2.33.1 Makro-Kürzel

\let\t=\liPetriTransitionsName

\let\tp=\liPetriTransPfeile

\let\k=\liPetriErreichKnotenDrei

2590 \RequirePackage{tikz}

2591 \usetikzlibrary{petri,arrows.meta}

Für die Darstellungsmatrix

2592 \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

```
2593 \def\liPetriSetzeSchluessel{%
2594   \def\TmpTransitionOne{}%
2595   \def\TmpTransitionTwo{}%
2596   \def\TmpTransitionThree{}%
2597   \def\TmpTransitionFour{}%
2598   \def\TmpTransitionFive{}%
2599   \def\TmpTransitionSix{}%
2600   \def\TmpTransitionSeven{}%
2601   \def\TmpTransitionEight{}%
2602   \def\TmpTransitionNine{}%
2603   \def\TmpTransitionTen{}%
2604   \pgfkeys{/petri/.cd,
2605     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2606     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
```

```

2607 p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2608 p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
2609 p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
2610 p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2611 p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2612 p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2613 p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2614 p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2615 t1/.store in=\TmpTransitionOne,t1/.default=activated,
2616 t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2617 t3/.store in=\TmpTransitionThree,t3/.default=activated,
2618 t4/.store in=\TmpTransitionFour,t4/.default=activated,
2619 t5/.store in=\TmpTransitionFive,t5/.default=activated,
2620 t6/.store in=\TmpTransitionSix,t6/.default=activated,
2621 t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2622 t8/.store in=\TmpTransitionEight,t8/.default=activated,
2623 t9/.store in=\TmpTransitionNine,t9/.default=activated,
2624 t10/.store in=\TmpTransitionTen,t10/.default=activated,
2625 scale/.store in=\TmpScale,scale/.default=0.5,
2626 x/.store in=\TmpX,x/.default=5,
2627 y/.store in=\TmpY,y/.default=5,
2628 }%
2629 }

2630 \tikzset{
2631   li petri/.style={
2632     activated/.style={
2633       very thick
2634     },
2635     inhibitor/.style={
2636       {Circle[open,length=2mm,fill=white]}-
2637     }
2638   }
2639 }

```

**\liPetriTransitionsName** **Let-Abkürzung:** `\let\t=\liPetriTransitionsName`  
`\$t_(\d+)\$ \t$1`

```

2640 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2641 \def\liPetriTransitionsName#1{
2642   \ifmmode
2643     \liPetriTransitionsNameOhneMathe{#1}
2644   \else
2645     $\liPetriTransitionsNameOhneMathe{#1}$
2646   \fi
2647 }

```

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

```

2648 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2649   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{\$t\sb{#3}$} (#2);
2650 }

```

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

```

2651 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2652 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2653 }

```

## 2.34 potenzmengen-konstruktion.sty

```
2654 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2655 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2656 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2657 \liLadePakete{formale-sprachen}
2658 \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}
%     }
%   }
% }
%
```

```
2659 \def\liZustandsMengenSammlung#1#2{
2660   \liZustandsnameGross{#1}
2661   {
2662     \footnotesize
2663     \liPotenzmenge{
2664       \str_case:nn {#1} {#2
2665       }
2666     }
2667 }
```

```

\liZustandsMengenSammlungNr
2668 \def\liZustandsMengenSammlungNr#1#2{
2669   \liZustandsnameGross{#1}
2670   {
2671     \footnotesize
2672     \liZustandsmengeNr{
2673       \str_case:nn {#1} #2
2674     }
2675   }
2676 }

2677 \ExplSyntaxOff
2678

```

## 2.35 pseudo.sty

```

2679 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2680 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen
2681 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

**Data:**  $G = (V, E, w)$ : ein zusammenhängender, ungerichteter, kantengewichteter Graph kruskal( $G$ )

$E' \leftarrow \emptyset$ ;  
 $L \leftarrow E$ ;  
 Sortiere die Kanten in  $L$  aufsteigend nach ihrem Kantengewicht.;

**while**  $L \neq \emptyset$  **do**

- 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* **then**
  - $E' \leftarrow E' \cup \{e\}$ ;
- end**

**end**

**Result:**  $M = (V, E')$  ist ein minimaler Spannbaum von  $G$ .

```

2682 \RequirePackage[german,boxruled]{algorithm2e}

```

```

2683

```

## 2.36 pumping-lemma.sty

2684 \NeedsTeXFormat{LaTeX2e}[1995/12/01]  
 2685 \ProvidesPackage{lehrant-informatik-pumping-lemma}[2021/08/11 Enthält die  
 2686 Definitionen für das Pumping-Lemma in der Regulären Sprache und  
 2687 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2688 \def\liPumpingRegulaer{%
2689   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2690   alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2691    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2692   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2693
2694   \begin{enumerate}
2695     \item  $|v| \geq 1$ 
2696     (Das Wort  $v$  ist nicht leer.)
2697
2698     \item  $|uv| \leq j$ 
2699     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2700
2701     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w$  in  $L$ 
2702     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2703     Sprache  $L$ )
2704   \end{enumerate}
2705
2706   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2707   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2708 }
```

\liPumpingKontextfrei

```
2709 \def\liPumpingKontextfrei{%
2710   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2711   sich alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  zerlegen lassen in
2712    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2713
2714   \begin{enumerate}
2715     \item  $|vx| \geq 1$ 
2716     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2717
2718     \item  $|vwx| \leq j$ 
2719     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2720
2721     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy$  in  $L$  (Für jede
2722     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2723     Sprache  $L$ )
2724   \end{enumerate}
2725 }
2726
```

## 2.37 quicksort.sty

```
2727 % https://tex.stackexchange.com/a/142634
2728 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2729 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2730
2731 %-----
2732 % USAGE:
2733 % \QSinitialize{comma, separated, numerical, values}
2734 % \loop
2735 % \QSpivotStep
2736 % \ifnum\value{pivotcount}>0
2737 %   \QSSortStep
2738 % \repeat
2739 %-----
2740
2741 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2742 \RequirePackage{xintfrac, xinttools}
2743
2744 \RequirePackage{tikz}
2745
2746 %-----
2747 % FIRST PART: TikZ styles and macros for the actual drawing
2748 \newcounter{cellcount}% used for coordinates of the node
2749 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2750
2751 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2752 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2753 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2754 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2755 % by police of LaTeX good conduct ? )
2756 \tikzset{l/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2757          o/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2758          r/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, t
2759 % this is the "b" style as used in the image below
2760 %   b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black,
2761 % nicer:
2762 %   b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta
2763 %   g/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white,
2764
2765 % NOTE the b style was originally the same as the r(aised) style apart from
2766 % not being raised, but I find it nicer with a somewhat different
2767 % specification. I have not updated the images though.
2768
2769 % How the nodes are drawn depending on whether on the left of the pivot value
2770 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2771
2772 \def\DecoLEFT #1{%
2773   \xintFor* ##1 in {#1} \do
2774   {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2775 }
2776
2777 \def\DecoINERT #1{%
2778   \xintFor* ##1 in {#1} \do
2779   {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2780 }
2781
2782 \def\DecoRIGHT #1{%
2783   \xintFor* ##1 in {#1} \do
2784   {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2785 }
2786
2787 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2788   \xintFor* ##1 in {#1} \do
```



```

2789     {\stepcounter{cellcount}}%
2790     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2791 }
2792
2793 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2794     \xintFor* ##1 in {#1} \do
2795     {\stepcounter{cellcount}}%
2796     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2797 }
2798
2799 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2800     \xintFor* ##1 in {#1} \do
2801     {\stepcounter{cellcount}}%
2802     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2803 }
2804
2805 %-----
2806 % SECOND PART: the actual sorting routines.
2807
2808 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2809 \def\QS@sort@b #1{\ifcase #1
2810     \expandafter\QS@sort@empty
2811     \or\expandafter\QS@sort@single
2812     \else\expandafter\QS@sort@c
2813     \fi
2814 }%
2815 \def\QS@sort@empty #1{}
2816 \def\QS@sort@single #1{\QS@Ir {#1}}
2817
2818 % This step is to pick the last as pivot.
2819 \def\QS@sort@c #1%
2820     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2821
2822 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2823 % The trick with \xintApplyUnbraced is that for example when selecting
2824 % the elements smaller than pivot, if we had been using \xintApply we
2825 % would have had at the minimum an empty brace pair. Thus we use the
2826 % "unbraced" variant, but then the \QS@select@smaller has added in
2827 % anticipation a level of braces.
2828 \def\QS@sort@d #1#2{%
2829     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2830     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2831     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2832 }%
2833 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2834 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2835 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2836
2837 %
2838 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2839 %
2840 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2841 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2842 % latter must handle correctly an empty argument.
2843
2844 %-----
2845 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QS@initialize.
2846
2847 % This draws all with suitable highlighting for the newly chosen pivots
2848 % (which will be shown raised)

```

```

2849 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2850             \let\QSIr\DecoINERT
2851             \let\QSIrr\DecoINERT
2852             \let\QSRr\DecoRIGHTwithPivot
2853 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2854             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2855             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2856 }
2857
2858 % This sorts and then draws, showing where the pivot chosen in the previous
2859 % step go. Next time they will have become "inert". If pivotcount is still at
2860 % zero on exit from \QSpivotStep, then this is the signal to stop before
2861 % executing \QSSortStep.
2862 \def\QSSortStep {\def\QSLr {\noexpand\QS@sort@a}%
2863             \def\QSRr {\noexpand\QS@sort@a}%
2864             \def\QSIr {\noexpand\QSIrr}%
2865             \let\QSIrr\relax
2866             \edef\QS@list{\QS@list}%
2867             \let\QSLr\relax
2868             \let\QSRr\relax
2869             \let\QSIr\relax
2870             \edef\QS@list{\QS@list}%
2871             \let\QSLr\DecoLEFT
2872             \let\QSIr\DecoINERTwithPivot
2873             \let\QSIrr\DecoINERT
2874             \let\QSRr\DecoRIGHT
2875 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2876             \setcounter{cellcount}{0}%
2877             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2878 }
2879
2880 \def\QSinitialize #1{%
2881     % first, we convert the comma separated values into a list of braced items
2882     % we use an \edef, and anyhow many \edef's will be used later
2883     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2884     \let\QSRr\DecoRIGHT
2885     % The \QSRr marker mutated to draw the last element as
2886     % pivot and the earlier ones with the suitable style.
2887     %
2888     % The list of marked braced items \QS@list is used both for drawing
2889     % (as here) and for doing the exchange of elements during sort.
2890     \par\centerline{\rule[1.5mm]{0pt}{8mm}}\setcounter{cellcount}{0}%
2891     \begin{tikzpicture}\QS@list\end{tikzpicture}}
2892 }
2893

```

## 2.38 relationale-algebra.sty

```

2894 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2895 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]
2896 \RequirePackage{amsmath}
2897 \RequirePackage{amssymb}

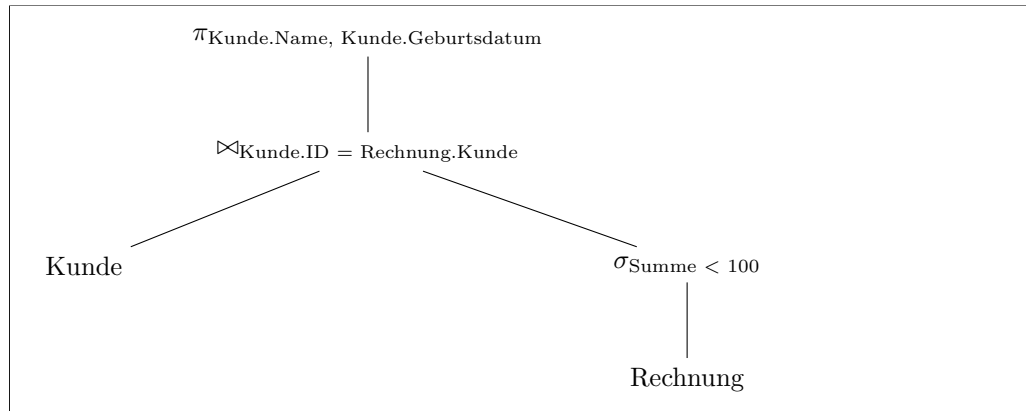
    Zum Zeichnen von Operatorenbäumen verwenden wir TikZ
% \begin{tikzpicture}
%   \node
%     (pi) {\pi_{\text{Kunde.Name, Kunde.Geburtsdatum}}};

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

%   \node[below left=of theta join]
%     {Kunde}
%   edge(theta join);

%   \node[below right=of theta join]
%     (sigma rechnerung) {\sigma_{\text{Summe < 100}}};
%   edge (theta join);

%   \node[below=of sigma rechnerung]
%     {Rechnerung}
%   edge(sigma rechnerung);
% \end{tikzpicture}
%
```



```

2898 \RequirePackage{tikz}
2899 \usetikzlibrary{positioning}

    Privates Makros, das zwei Querstriche erzeugt.
2900 \def\o@join{\setbox0=\hbox{\bowtie}}%
2901   \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2902 }

\leftouterjoin A \leftouterjoin B: A ⋈ B
2903 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}

\rightouterjoin A \rightouterjoin B: A ⋉ B
2904 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}

\fullouterjoin A \fullouterjoin B: A ⋈ B
2905 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
2906
```

## 2.39 rmodell.sty

```
2907 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2908 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2909 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2910 Datenbanken.]
2911 \RequirePackage{soul}
```

### 2.39.1 Makro-Kürzel

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

```
\liPrimaer \liPrimaer{text}: Unterstreichung für den Primärschlüssel
2912 \def\liPrimaer#1{\ul{#1}}
```

```
\liFremd \liFremd{text}: Überstreichung für den Fremdschlüssel
2913 \def\liFremd#1{{\setul{-0.9em}{}}\ul{#1}}
```

```
liRmodell \begin{liRmodell} \end{liRmodell}: Kleinere Schrift und Schreibmaschinenschrift.
2914 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2915 \ExplSyntaxOn
2916 \NewDocumentEnvironment { liRmodell }
2917 { +b }
2918 {
2919 \medskip
2920 {
2921 \linespread{2}
2922 \setlength{\parindent}{0pt}
2923 \li@Rmodell@Schrift#1
2924 }
2925 \medskip
2926 } {}
2927 \ExplSyntaxOff
```

```
\liRelationMenge Let-Abkürzung: \let\l=r=\liRelationMenge
\liRelationMenge{name}{attribut, attribut}: Umhüllen der Attribute mit geschweiften
und dann eckigen Klammern.
2928 \def\liRelationMenge#1#2{
2929 \noindent
2930 #1 : \{ [ #2 ] \}
2931 \par
2932 }
```

```
\liAttribut Let-Abkürzung: \let\l=a=\liAttribut
\liAttribut{text}: Gleiche Schrift wie Umgebung liRmodell
2933 \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}
%
```

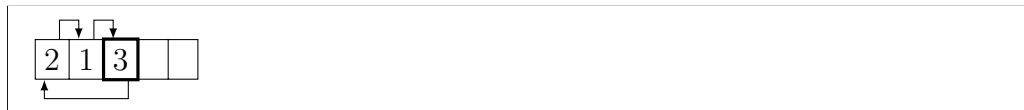
```
2934 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2935
```

## 2.40 sortieren.sty

```
2936 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2937 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
2938 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} \nodepart{five} }
%   \liSortierPfeil{one}{two}
%   \liSortierPfeil{two}{three}
%   \liSortierMarkierung{two split south}{three split north}
%   \liSortierPfeilUnten{three}{one}
% }
%
```



```
2939 \RequirePackage{tikz}
2940 \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.

```
2941 \def\liVertauschen#1{
2942   \directlua{
2943     local sortieren = require('lehramt-informatik-sortieren')
2944     sortieren('#1')
2945   }
2946 }
```

`\liSortierPfeil`

```
2947 \def\liSortierPfeil#1#2{
2948   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2949 }
```

`\liSortierPfeilUnten`

```
2950 \def\liSortierPfeilUnten#1#2{
2951   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2952 }
```

`\liSortierMarkierung`

```
2953 \def\liSortierMarkierung#1#2{\node[
2954   draw,
2955   very thick,
2956   fit=(reihe.#1) (reihe.#2),
2957   inner sep=0pt
2958 ] {};
```

```
2959 }

2960 \tikzset{
2961   li sortierung zahlenreihe/.style={
2962     draw,
2963     thin,
2964     font=\large,
2965     rectangle split horizontal,
2966     rectangle split,
2967   }
2968 }
```

```

2969 % https://tex.stackexchange.com/a/140895
2970 \RequirePackage{forest,xstring}
2971 \usetikzlibrary{calc}
2972
2973 \makeatletter
2974 \pgfmathdeclarefunction{strrepeat}{2}{%
2975   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2976   \let\pgfmathresult\pgfutil@empty
2977   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2978     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2979     \advance\pgfmath@count-1\relax
2980   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2981 \makeatother
2982
2983 \def\myNodes{}
2984
2985 \ExplSyntaxOn
2986 \newcommand*\sortList[1]{%
2987   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2988 \ExplSyntaxOff
2989
2990 \forestset{
2991   sort/.code={%
2992     \pgfmathparse{level()>\forestSortLevel}%
2993     \ifnum\pgfmathresult=0
2994       \StrSubstitute{\forestov{content}}{ }{,}[\myList]%
2995       \sortList\myList
2996       \StrSubstitute{\myList}{,}{ }[\myList]%
2997       \pgfmathparse{strrepeat("1",level())}%
2998       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-1!({\forestov{name}}$)
2999         (m\forestov{name}) {\myList}}%
3000       \pgfmathparse{level()=\forestSortLevel}%
3001       \ifnum\pgfmathresult=1
3002         \forestOget{\forestov{@first}}{name}\forestFirst
3003         \forestOget{\forestov{@last}}{name}\forestLast
3004         \xappto\myNodes{[<-]edge (\forestOv{\forestov{@first}}{name})
3005           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
3006       \fi
3007       \ifnum\forestov{@parent}=0\else
3008         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
3009       \fi
3010       \gappto\myNodes{;}%
3011     \fi}}
3012
3013 \forestset{sort level/.code=%
3014   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
3015   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
3016

```

## 2.41 spalten.sty

```
3017 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3018 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
3019 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
3020 realisiert werden kann.]
3021 \RequirePackage{multicol}
```

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

```
3022 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
3023
```

## 2.42 sql.sty

3024 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

3025 \ProvidesPackage{lehramt-informatik-tabelle}[2021/09/02 Zu Setzen von SQL]

### 2.42.1 Latex-Markup-Beispiel

```
% \begin{liAdditum}[Übungsdatenbank]
% % Datenbankname: Personalverwaltung
% \begin{minted}{sql}
% CREATE TABLE Abteilung(
%   AbteilungsID INTEGER PRIMARY KEY,
%   Bezeichnung VARCHAR(30)
% );
%
% CREATE TABLE Mitarbeiter(
%   MitarbeiterID INTEGER PRIMARY KEY,
%   Vorname VARCHAR(30),
%   Nachname VARCHAR(30),
%   Vorgesetzter INTEGER REFERENCES Mitarbeiter(MitarbeiterID),
%   AbteilungsID INTEGER REFERENCES Abteilung(AbteilungsID),
%   Telefonnummer VARCHAR(50),
%   Gehalt DOUBLE PRECISION
% );
%
% INSERT INTO Abteilung VALUES
%   (1, 'Buchhaltung');
%
% INSERT INTO Mitarbeiter
%   (MitarbeiterID, Vorname, Nachname, Vorgesetzter, AbteilungsID, Telefonnummer, Gehalt)
% VALUES
%   (1, 'Hans', 'Meier', 11, 4, '023/13432', 2335),
%   (2, 'Fred', 'Wolitz', 11, 2, '0233/413432', 1233);
% \end{minted}
% \index{SQL mit Übungsdatenbank}
% \end{liAdditum}
%

3026 \liLadePakete{syntax}

3027 \RequirePackage{fancyvrb}
3028 \DefineVerbatimEnvironment{liSqlErgebnis}{Verbatim}
3029 {fontsize=\footnotesize}

3030
```



## 2.43 struktogramm.sty

```
3031 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3032 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
3033 Paket struktex zum Setzen von Struktogrammen]
3034 \RequirePackage{struktex}
3035
```

## 2.44 syntax.sty

```

3036 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3037 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
3038 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
3039 \RequirePackage{xparse}

```

Um das Paket `minted` benutzen zu können, muss `pygmentize` installiert sein. Außerdem müssen die TeX-Dateien mit der Kommandozeilen-Option `--shell-escape` kompiliert werden.

### 2.44.1 Makro-Kürzel

```

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

```

```

3040 \ExplSyntaxOn
3041 \directlua{
3042   syntax = require('lehramt-informatik-syntax')
3043   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
3044   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
3045   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
3046   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
3047   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
3048   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
3049 }
3050 \RequirePackage{hyperref}
3051 \RequirePackage{minted}
3052 % pygmentize -L styles
3053 \usemintedstyle{colorful}
3054 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
3055 %\AfterEndEnvironment{minted}{\end{mdframed}}
3056 %\setminted{breaklines=true,linenos}
3057 \setminted{
3058   breaklines=true,
3059   linenos=false,
3060   fontsize=\footnotesize,
3061 }

```

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

```

3062 \newminted[liJavaAngabe]{java}{
3063   xleftmargin=1cm
3064 }

```

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

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

```

3065 \def\liJavaCode#1{
3066   \,
3067   \textcolor{blue}{
3068     \mintinline[
3069       fontsize=\normalsize,
3070       breakanywhere % https://github.com/gpoore/minted/issues/31#issuecomment-
458640242
3071     ]{java}|#1|
3072   }
3073   \,
3074 }

```

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

```

3075 \def\liLatexCode#1{\mintinline{latex}|#1|}

```

```

3076 \def\li@GithubLink#1#2{
3077   \begin{flushright}
3078     \tiny
3079     Code-Beispiel~auf~Github~ansehen:~
3080     \href{#1}{\nolinkurl{#2}}
3081   \end{flushright}
3082 }

\liJavaDatei Eine komplette Java-Datei einbinden, die Verzeichnis ./Code/src/main/java/org/bschlangaul
liegt.
3083 \NewDocumentCommand{\liJavaDatei}{ O{firstline=3} m }{
3084   \inputminted[#1]{java}{
3085     \directlua{
3086       syntax.drucke_absoluten_pfad('#2', false)
3087     }
3088   }
3089   \li@GithubLink
3090   {\directlua{syntax.drucke_github_url('#2', false)}}
3091   {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
3092 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
3093 \NewDocumentCommand{\liJavaTestDatei}{ O{firstline=3} m }{
3094   \inputminted[#1]{java}{
3095     \directlua{
3096       syntax.drucke_absoluten_pfad('#2', true)
3097     }
3098   }
3099   \li@GithubLink
3100   {\directlua{syntax.drucke_github_url('#2', true)}}
3101   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
3102 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([
\liJavaExamen$1{$2}{$3}{$4}{$5}
3103 \NewDocumentCommand{\liJavaExamen}{ O{firstline=3} m m m m }{
3104   \inputminted[#1]{java}{
3105     \directlua{
3106       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
3107     }
3108   }
3109   \li@GithubLink
3110   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
3111   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
3112 }
3113 }

\liAssemblerCode
3114 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
3115 \NewDocumentCommand{\liAssemblerDatei}{ m }{
3116   \inputminted{asm}{#1}
3117 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
3118 \NewDocumentCommand{\liMinispracheDatei}{ m }{
3119   \inputminted{componentpascal}{#1}
3120 }

```

```

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
3121 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
3122 \NewDocumentCommand{\liHaskellDatei}{ m }{
3123   \inputminted{haskell}{#1}
3124 }

3125 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
3126 \def\liSqlCode#1{\mintinline{sql}|#1|}

3127

```

## 2.45 syntaxbaum.sty

```
3128 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3129 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
3130 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
3131 \RequirePackage{tikz-qtrees}
3132
3133 \tikzset{li parsetree/.style={
3134     every internal node/.style={
3135         draw,circle
3136     },
3137     every leaf node/.style={
3138         draw,rectangle
3139     },
3140 }
3141 }
3142
```

## 2.46 synthese-algorithmus.sty

```
3143 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3144 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
3145 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
3146 Relation in die 3. Normalform]

3147 \liLadePakete{normalformen,mathe,typographie}
3148 \ExplSyntaxOn
```

### 2.46.1 Makro-Kürzel

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

### 2.46.2 TeX-Markup Grundgerüst

```
% \let\schrittE=\liSyntheseUeberErklaerung
% \begin{enumerate}
% \item \schrittE{1}
%
% \begin{enumerate}
% \item \schrittE{1-1}
% \item \schrittE{1-2}
% \item \schrittE{1-3}
% \item \schrittE{1-4}
% \end{enumerate}
%
% \item \schrittE{2}
% \item \schrittE{3}
% \item \schrittE{4}
% \end{enumerate}
%
```

### 2.46.3 TeX-Markup Linksreduktion

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

% \liPseudoUeberschrift{\fa{C, E -> D, X}}
%
%  $\mathbf{D, X} \in \mathbf{ahl\{C, E\} \{E\} \{A, C, B, \mathbf{D, X}\}}$ 
%  $\mathbf{D, X} \notin \mathbf{ahl\{C, E\} \{C\} \{E, F\}}$ 
%
% \liPseudoUeberschrift{\fa{C, E -> F}}
%
%  $\mathbf{F} \notin \mathbf{ahl\{C, E\} \{E\} \{A, C, B\}}$ 
%  $\mathbf{F} \in \mathbf{ahl\{C, E\} \{C\} \{E, \mathbf{F}\}}$ 
%
```

### 2.46.4 TeX-Markup Rechtsreduktion

```
% \let\ahr=\liRechtsReduktionInline
% \let\fa=\liFunktionaleAbhaengigkeit
%
% \liPseudoUeberschrift{F}
%
%  $\mathbf{F} \in \mathbf{ahr\{E -> F, X\} \{E -> F\} \{E, \mathbf{F}\}}$ 
%
% \liPseudoUeberschrift{A}
```

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

## 2.46.5 TeX-Markup Relationen formen

```
% \let\r=\liRelation
% \let\u=\underline
%
% \r[R1]{\u{A, D}, E}\
% \r[R2]{\u{B, C}, A, E}\
% \r[R3]{\u{D}, B}
%
```

### 1. Kanonische Überdeckung

— Die kanonische Überdeckung - also die kleinst mögliche noch äquivalente Menge von funktionalen Abhängigkeiten kann in vier Schritten erreicht werden. —

#### (a) Linksreduktion

— Führe für jede funktionale Abhängigkeit  $\alpha \rightarrow \beta \in F$  die Linksreduktion durch, überprüfe also für alle  $A \in \alpha$ , ob  $A$  überflüssig ist, d. h. ob  $\beta \subseteq \text{AttrHülle}(F, \alpha - A)$ .

#### (b) Rechtsreduktion

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

#### (c) Löschen leerer Klauseln

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

#### (d) Vereinigung

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

### 2. Relationsschemata formen

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

### 3. Schlüssel hinzufügen

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

### 4. Entfernung überflüssiger Teilschemata

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

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

```
3149 \def\liSyntheseUeberschrift#1{
3150   {
3151     \bfseries
3152     \rmfamily
3153     \str_case:nn {#1} {
3154       {1} {Kanonische-Überdeckung}
3155       {1-1} {Linksreduktion}
3156       {1-2} {Rechtsreduktion}
```

```

3157 {1-3} {Löschen-leerer-Klauseln}
3158 {1-4} {Vereinigung}
3159 {2} {Relationsschemata~formen}
3160 {3} {Schlüssel-hinzufügen}
3161 {4} {Entfernung-überflüssiger-Teilschemata}
3162 }
3163 }
3164 }

```

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

```

3165 \def\li@synthese@erklaerung@texte#1{
3166   \str_case:nn {#1} {
3167     {1} {
3168       Die~kanonische~Überdeckung~---also~die~kleinst~mögliche~noch~
3169       äquivalente~Menge~von~funktionalen~Abhängigkeiten~kann~in~vier~
3170       Schritten~erreicht~werden.
3171     }
3172     {1-1} {
3173       Führe~für~jede~funktionale~Anhängigkeit~
3174        $\alpha \rightarrow \beta$ ~in~F~die~Linksreduktion~durch,~
3175       überprüfe~also~für~alle~
3176        $A \in \alpha$ ,~ob~ $A$ ~überflüssig~ist,~d.h.~ob~
3177        $\beta \subseteqq \text{AttributHuelle}(F, \alpha \cup A)$ .
3178     }
3179     {1-2} {
3180       Führe~für~jede~(verbliebene)~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~die~Rechtsreduktion~durch,~überprüfe~also~für~
3181       alle~ $B \in \beta$ ,~ob~ $B \in \text{AttributHuelle}(F, \alpha \rightarrow \beta \cup B)$ ,~
3182        $\alpha \rightarrow \beta$ ~gilt.~In~diesem~Fall~ist~B~auf~der~rechten~Seite~
3183       überflüssig~und~kann~eliminiert~werden,~dh~ $\alpha \rightarrow \beta$ ~wird~durch~ $\alpha \rightarrow \beta \cup B$ ~
3184       ersetzt.
3185     }
3186     {1-3} {
3187       Entferne~die~funktionalen~Abhängigkeiten~der~Form~ $\alpha \rightarrow \emptyset$ ,~die~im~2.~Schritt~möglicherweise~
3188       entstanden~sind.
3189     }
3190     {1-4} {
3191       Fasse~mittels~der~Vereinigungsregel~funktionale~Abhängigkeiten~
3192       der~Form~ $\alpha \rightarrow \beta \cup \alpha \rightarrow \beta \cup \dots \cup \alpha \rightarrow \beta \cup \alpha \rightarrow \beta$ ~so~dass~ $\alpha \rightarrow \beta \cup \alpha \rightarrow \beta \cup \dots \cup \alpha \rightarrow \beta$ ~
3193       verbleibt.
3194     }
3195     % Kemper Seite 197
3196     {2} {
3197       Erzeuge~für~jede~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~ein~Relationenschema~ $R \in \mathcal{R}$ ~so~dass~ $R \in \mathcal{R}$ ~
3198        $\alpha \rightarrow \beta$ .
3199     }
3200     {3} {
3201       Falls~eines~der~in~Schritt~2.~erzeugten~Schemata~ $R \in \mathcal{R}$ ~
3202       einen~Schlüsselkandidaten~von~ $\mathcal{R}$ ~bezüglich~ $F \in \mathcal{F}$ ~
3203       enthält,~sind~wir~fertig,~sonst~wähle~einen~Schlüsselkandidaten~
3204        $K \subseteqq \mathcal{R}$ ~aus~und~definiere~folgendes~
3205       zusätzliche~Schema:~ $R \in \mathcal{R}$ ~so~dass~ $R \in \mathcal{R}$ ~
3206       und~ $R \in \mathcal{F}$ ~
3207     }
3208     {4} {
3209       Eliminiere~diejenigen~Schemata~ $R \in \mathcal{R}$ ~die~in~einem~
3210       anderen~Relationenschema~ $R' \in \mathcal{R}$ ~enthalten~sind,~d.h.~
3211        $R \subseteqq R'$ .
3212     }
3213   }
3214 }

```



```

3219 }
3220 }
3221 \def\liSyntheseErklaerung#1{
3222 {
3223     \itshape
3224     \footnotesize
3225     \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
3226 }
3227 }

```

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

```

3228 \def\liSyntheseUeberErklaerung#1{
3229     \liSyntheseUeberschrift{#1}\par
3230     \liSyntheseErklaerung{#1}
3231 }

```

```

3232 \ExplSyntaxOff
3233

```

## 2.47 tabelle.sty

```
3234 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3235 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]
3236 \RequirePackage{tabularx}
3237
```

## 2.48 typography.sty

```
3238 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3239 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
3240 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
3241 formatierung.sty definiert.]
```

```
3242 \ExplSyntaxOn
```

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

```
3243 \RequirePackage{fontawesome}
```

`\liErledigt` `\liErledigt:` ☑

```
3244 \let\liErledigt=\faCheckSquareO
```

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

```
3245 \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 —

```
3246 \def\liParagraphMitLinien#1{
```

```
3247 \noindent
```

```
3248 \vrule height 2pt depth -1.6pt width 0.4cm
```

```
3249 \enspace
```

```
3250 #1
```

```
3251 \enspace
```

```
3252 \leaders\vrule height 2pt depth -1.6pt \hfill \null
```

```
3253 \par
```

```
3254 \medskip
```

```
3255 }
```

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

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

```
3256 \def\liGeschweifteKlammern#1#2#3#4{
```

```
3257 \par
```

```
3258 \medskip
```

```
3259 \noindent
```

```
3260 #1 \, $= \Bigl\{
```

```
3261 \vspace{#3}
```

```
3262 #2
```

```
3263 \vspace{#4}
```

```
3264 \begin{flushright}$\Bigr\}$\end{flushright}
```

```
3265 \par
```

```
3266 }
```

`\liTypoUeberschrift`

```
3267 \def\liTypoUeberschrift#1{
```

```
3268 {
```

```
3269 \bfseries\rmfamily
```

```
3270 #1
```

```
3271 }
```

```
3272 }
```

\liTypoUeberGross

```
3273 \def\liTypoUeberGross#1{
3274   {
3275     \huge
3276     \liTypoUeberschrift{#1}
3277   }
3278 }
```

\liTypoUeberGross

```
3279 \def\liTypoUeberGROSS#1{
3280   {
3281     \Huge
3282     \liTypoUeberschrift{#1}
3283   }
3284 }
```

```
3285 \ExplSyntaxOff
```

```
3286
```

## 2.49 uml.sty

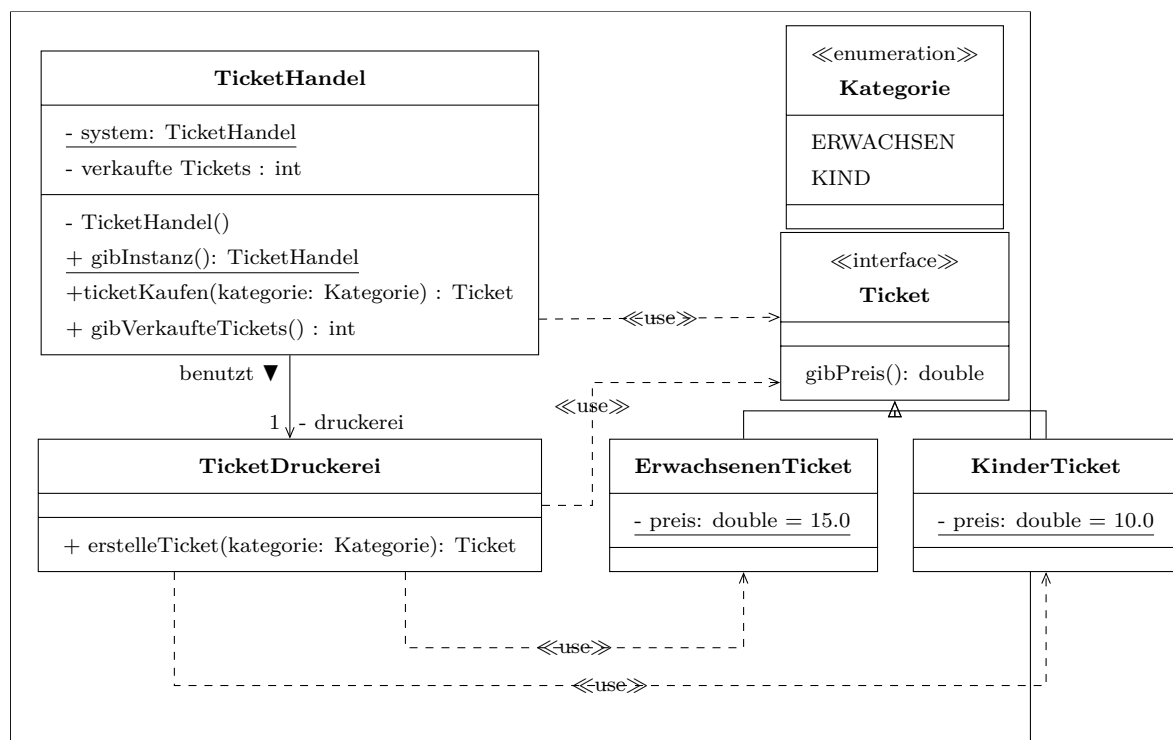
```

3287 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3288 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
3289 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
3290 Erweiterung bereitstellt]

3291 \RequirePackage{tikz-uml}
3292 \RequirePackage{tikz-uml-activity}
3293 % Not compatible with wasysym
3294 %\RequirePackage{mathabx}
3295 \RequirePackage{wasysym}
3296 \usetikzlibrary{positioning}

3297 \tikzumlset{
3298   fill class=white!0,
3299   font=\footnotesize,
3300   fill object=white!0,
3301   fill note=white!0,
3302   fill state=white!0,
3303   % Use case
3304   fill usecase=white!0,
3305   fill system=white!0,
3306 }

```



```

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

```

```

3307 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
3308   \def\@liDirLeft{}
3309   \def\@liDirRight{ \RIGHTarrow}
3310   \pgfkeys{/lese/dir/.is choice}
3311   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
3312   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
3313   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{ \LEFTarrow }}}
3314   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
3315
3316   \def\@liPos{above}
3317   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}

```

```

3318
3319 \def\@liDistance{0cm}
3320 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
3321
3322 \pgfkeys{/lese/.cd,#1}
3323
3324 \node[\@liPos = \@liDistance of #2-middle] {
3325   \@liDirLeft{\footnotesize#2}\@liDirRight
3326 };
3327 }
3328

```

## 2.50 vollstaendige-induktion.sty

```
3329 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3330 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01]
3331 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
3332 Überschriften für die einzelnen Schritte]
```

### 2.50.1 Makro-Kürzel

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

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

Lade häufig benötigte Pakete

```
3333 \RequirePackage{lehramt-informatik-typographie}
3334 \RequirePackage{lehramt-informatik-mathe}
3335 \RequirePackage{lehramt-informatik-syntax}
3336 \ExplSyntaxOn
```

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

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

```
3337 \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\le=\liInduktionErklaerung`

```
3338 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

`\liInduktionAnfang`

```
3339 \def\liInduktionAnfang{
```

```
3340   \liPseudoUeberschrift{Induktionsanfang}
```

```
3341
```

```
3342   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
3343   \liParagraphMitLinien{
```

```
3344     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
```

```
3345   }
```

```
3346 }
```

`\liInduktionVoraussetzung`

```
3347 \def\liInduktionVoraussetzung{
```

```
3348   \liPseudoUeberschrift{Induktionsvoraussetzung}
```

```
3349
```

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

```
3351   \liParagraphMitLinien{
```

```
3352     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ ~.
```

```
3353   }
```

```
3354 }
```

`\liInduktionSchritt`

```
3355 \def\liInduktionSchritt{
```

```
3356   \liPseudoUeberschrift{Induktionsschritt}
```

```
3357
```

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

```
3359   \liParagraphMitLinien{
```

```
3360     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
```

```
3361     auch~ $A(n=k+1)$ ~wahr~sein~muss.
```

```
3362   }
```

```
3363 }
```

```
3364 \ExplSyntaxOff
```

```
3365
```



## 2.51 wasserfall.sty

```
3366 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3367 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
3368 \RequirePackage{tikz}
3369 \tikzset{wasserfall/.style={
3370   >=stealth,
3371   node distance = 2mm and -8mm,
3372   start chain = A going below right,
3373   every node/.style = {
3374     draw,
3375     text width=24mm,
3376     minimum height=12mm,
3377     align=center,
3378     inner sep=1mm,
3379     fill=white,
3380     drop shadow={fill=black},
3381     on chain=A
3382   },
3383 }}
3384 \usetikzlibrary{chains,positioning,shadows}
3385
```

## 2.52 wpkalkuel.sty

```
3386 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3387 \ProvidesPackage{lehramt-informatik-wpkalkuel}[2020/06/13]
```

### 2.52.1 Makro-Kürzel

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

```
3388 \RequirePackage{amsmath}
3389 \ExplSyntaxOn
```

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

```
3390 \def\liWpKalkuelOhneMathe#1#2{
3391   \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)
3392 }
3393 \def\liWpKalkuel#1#2{
3394   \ifmmode
3395     \liWpKalkuelOhneMathe{#1}{#2}
3396   \else
3397     $\liWpKalkuelOhneMathe{#1}{#2}$
3398   \fi
3399 }
```

**\MatheEnv**

```
3400 \def\MatheEnv#1{
3401   \medskip
3402
3403   \hspace{1em}#1
3404
3405   \medskip
3406 }
```

**\Mathe**

```
3407 \def\Mathe#1{
3408   \MatheEnv{#1$}
3409 }
```

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

```
3410 \def\liWpEquivalent#1{
3411   \MatheEnv{$\equiv$\hspace{1em}$#1$}
3412 }
```

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

```
3413 \newlength{\@Skip@Erklaerung@Reset}
3414 \def\liWpErklaerung#1{
3415   \setlength{\@Skip@Erklaerung@Reset}{\leftskip}
3416   \setlength{\leftskip}{0.5cm}
3417
3418   \par
3419   \noindent
3420   {
3421     \scriptsize
3422     #1
3423   }
3424   \par
3425
3426   \setlength{\leftskip}{\@Skip@Erklaerung@Reset}
3427 }
```

```

3428 \def\liWpErklaerungVerzweigung{
3429   $\liWpKalkuel0hneMathe{if~\{{b~}\}~then~\{{a1~}\}~else~\{{a2~}\}}~{Q}}
3430   \equiv
3431   (b \land \liWpKalkuel0hneMathe{a1}~{Q})
3432   \lor
3433   (\neg b \land \liWpKalkuel0hneMathe{a2}~{Q})$
3434 }

3435 \ExplSyntaxOff

3436

```

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.

<code>\clist</code> . . . . .	237, 285, 286, 308, 312, 2987	<code>\empty</code> . . . . .	2019, 2537	2927, 2988, 3125, 3232, 3285, 3364, 3435
<code>\columnbreak</code> . . . . .	3022	<code>\emptyset</code> . . . . .	2447, 3191, 3212, 3245	<code>\ExplSyntaxOn</code> . . . . .
<code>\comment</code> . . . . .	2037	<code>\end</code> . . . . .	800, 826, 851, 886, 920, 952, 980, 996, 1015, 1023, 1054, 1083, 1104, 1129, 1154, 1174, 1193, 1218, 1236, 1337, 1423, 1529, 1625, 1720, 1776, 1777, 1784, 1893, 1969, 2029, 2048, 2072, 2090, 2091, 2105, 2116, 2117, 2198, 2214, 2239, 2262, 2277, 2287, 2360, 2364, 2365, 2388, 2401, 2462, 2463, 2511, 2556, 2704, 2724, 2855, 2877, 2891, 3055, 3081, 3264	<code>\ExplSyntaxOn</code> . . . . .
<code>\contentsname</code> . . . . .	1731	<code>\endcomment</code> . . . . .	2050	. . . . . 35, 66, 107, 140, 165, 198, 203, 228, 560, 623, 640, 679, 1255, 1350, 1372, 1447, 1548, 1650, 1901, 1913, 1962, 2133, 2310, 2465, 2570, 2658, 2915, 2985, 3040, 3148, 3242, 3336, 3389
<code>\cs</code> . . . . .	311, 339, 363, 366, 376, 399, 411, 422, 423, 460, 467, 472, 1549, 1556, 1562, 1680, 1724, 2077, 2356, 2577	<code>\endcsname</code> . . . . .	1500, 1503	<code>\faCheckSquare0</code> . . . . . 3244
<code>\csname</code> . . . . .	1500, 1503	<code>\endgroup</code> . . . . .	1968, 2512, 2980	<code>\faCircleThin</code> . . . . . 1274
<code>\cup</code> . . . . .	1315, 2539, 3183, 3197, 3204	<code>\enspace</code> . . . . .	3249, 3251	<code>\faGg</code> . . . . . 1268
<b>D</b>		environments:		<code>\fancyfoot</code> . . . . .
<code>\DeclareMathSymbol</code> . .	. . . . . 2304, 2305	<code>liAdditum</code> . . . . .	2033	. . . . . 1427, 1428, 1429, 1917, 1918, 1919, 1920
<code>\DecoINERT</code> . . . . .	2777, 2850, 2851, 2873	<code>liAHuelle</code> . . . . .	2506	<code>\fancyhead</code> . . . . .
<code>\DecoINERTwithPivot</code> .	. . . . . 2793, 2872	<code>liAntwort</code> . . . . .	1998	. . . . . 1426, 1914, 1915, 1916
<code>\DecoLEFT</code> . . . . .	2772, 2871	<code>liDiagramm</code> . . . . .	2107	<code>\faSquare0</code> . . . . . 1262
<code>\DecoLEFTwithPivot</code> . .	. . . . . 2787, 2849	<code>liEinbettung</code> . . . . .	1997	<code>\fi</code> . . . . . 661, 669, 677, 685, 692, 1298, 1312, 1348, 2022, 2025, 2031, 2038, 2051, 2075, 2146, 2156, 2166, 2180, 2503, 2540, 2585, 2646, 2813, 3005, 3006, 3009, 3011, 3398
<code>\DecoRIGHT</code> . . . . .	2782, 2874, 2884	<code>liExkurs</code> . . . . .	2053	<code>\filcenter</code> . . . . . 1644
<code>\DecoRIGHTwithPivot</code> .	. . . . . 2799, 2852	<code>liGraphenFormat</code> . .	1492	<code>\footcite</code> . . . . .
<code>\definecolor</code> . . . . .	1403	<code>liJavaAngabe</code> . . . .	3062	. . . . . 758, 827, 849, 896, 919, 951, 1004, 1055, 1082, 1128, 1789, 1792, 1799, 1804, 1809, 1813, 1819, 1824, 1929, 2215, 2216, 2470, 2707
<code>\DefineVerbatimEnvironment</code>	. . . . . 3028	<code>liKasten</code> . . . . .	1420	<code>\footnote</code> . . . . . 2120, 2124
<code>\delta</code> . . . . .	70, 112, 170, 212, 1306	<code>liKontrollflussgraph</code>	. . . . . 1887	<code>\footnotesize</code> . . . . . 152, 433, 515, 609, 1279, 1597, 1696, 1995, 2067, 2084, 2100, 2386, 2414, 2508, 2525, 2533, 2662, 2671, 2914, 3029, 3060, 3224, 3299, 3325
<code>\dh</code> . . . . .	2128, 3185	<code>liLernkartei</code> . . . .	2097	<code>\foreach</code> . . . . . 1515, 1518, 1525
<code>\directlua</code> . . . . .	63, 142, 200, 205, 1287, 1301, 1321, 1329, 1336, 1341, 2490, 2495, 2548, 2555, 2562, 2942, 3041, 3085, 3090, 3091, 3095, 3100, 3101, 3105, 3111, 3112	<code>liProduktionsRegeln</code>	. . . . . 1330	<code>\forestFirst</code> . . . . . 3002, 3005
<code>\do</code> . . . . .	2773, 2778, 2783, 2788, 2794, 2800	<code>liProjektSprache</code> . .	1996	<code>\forestLast</code> . . . . . 3003, 3005
<code>\dots</code> . . . . .	595, 599, 1810, 2701, 3196, 3197	<code>liQuellen</code> . . . . .	2077	<code>\forestOget</code> . . . . . 3002, 3003
<code>\DOWNarrow</code> . . . . .	3312	<code>liRelationenSchemaFormat</code>	. . . . . 2934	<code>\forestOnes</code> . . . . . 3015
<code>\draw</code> . . . . .	1520, 1523, 1526, 2649, 2948, 2951	<code>liRmodell</code> . . . . .	2914	<code>\forestOv</code> . . . . . 3004, 3005, 3008
<b>E</b>		<code>liUebergangsTabelle</code>	. . . . . 2455	<code>\forestov</code> . . . . . 2994, 2998, 2999, 3002, 3003, 3004, 3005, 3007, 3008
<code>\edef</code> . . . . .	1511, 2866, 2870, 2882, 2883	<code>\equiv</code> . . . . .	3411, 3430	<code>\forestset</code> . . . . . 2990, 3013
<code>\edge</code> . . . . .	282	<code>\erzeuge@tiefgestellt</code>	. . . . . 1301, 1302, 1306	<code>\forestSortLevel</code> . . . .
<code>\else</code> . . . . .	659, 667, 675, 683, 690, 1296, 1310, 1346, 2019, 2024, 2030, 2035, 2049, 2074, 2144, 2154, 2164, 2178, 2501, 2538, 2583, 2644, 2812, 3005, 3007, 3396	<code>\expandafter</code> . . . .	1500, 2037, 2050, 2808, 2810, 2811, 2812, 2820, 2978	. . . . . 2992, 3000, 3014, 3015
<code>\emph</code> . . . . .	756, 1003, 1258, 1791, 1820, 1822, 1971	<code>\ExplSyntaxOff</code> . . .	54, 97, 139, 144, 197, 202, 207, 617, 639, 654, 730, 1282, 1360, 1395, 1463, 1628, 1753, 1908, 1924, 2129, 2194, 2404, 2483, 2677,	

<code>\frac</code>	2173, 2206, 2238, 2253	<code>\ifnum</code>	2736, 1374, 1375, 1376,
<code>\fullouterjoin</code>	2905		1377, 1380, 1381,
<b>G</b>			
<code>\g</code>	39, 285, 286, 308,	<code>\ifx</code>	2019, 2537, 3005
	312, 318, 319, 320,	<code>\in</code>	583, 739,
	321, 322, 324, 325,		1817, 2186, 2189,
	327, 328, 330, 331,		2192, 2221, 2227,
	332, 333, 334, 335,		2232, 2690, 2701,
	336, 337, 340, 344,		2711, 2721, 3174,
	345, 346, 349, 351,		3176, 3182, 3203, 3352
	352, 353, 354, 355,	<code>\includegraphics</code>	1454, 1459
	356, 401, 405, 409,		1724, 1741
	412, 413, 415, 416,	<code>\inhalts</code>	1724, 1741
	418, 419, 426, 427,	<code>\inhaltsverzeichnis</code>	1963
	428, 429, 437, 438,	<code>\input</code>	17, 20,
	440, 446, 447, 449,		23, 26, 29, 485, 1661
	450, 452, 453, 465,	<code>\inputminted</code>	3084, 3094,
	468, 473, 475, 479,		3104, 3116, 3119, 3123
	1550, 1552, 1563,	<code>\int</code>	2987
	1565, 1572, 1585,	<code>\item</code>	554, 555, 832, 836,
	1586, 1652, 1656,		841, 846, 890, 899,
	1660, 1664, 1665,		904, 912, 984, 989,
	1666, 1667, 1668,		993, 1019, 1059,
	1670, 1671, 1673,		1064, 1071, 1079,
	1703, 1705, 1710, 1714		1108, 1113, 1117,
<code>\Gamma</code>	111, 169, 212, 1315		1122, 1222, 1227,
<code>\gappto</code>	3010		1232, 1774, 1775,
<code>\geometry</code>	5		2077, 2081, 2201,
<code>\geq</code>	1817, 2204,		2206, 2210, 2220,
	2690, 2695, 2711, 2715		2226, 2231, 2243,
<b>H</b>			
<code>\hbox</code>	2900		2247, 2251, 2255,
<code>\headrulewidth</code>	1431, 1921		2259, 2266, 2270,
<code>\headwidth</code>	1923		2274, 2695, 2698,
<code>\hfill</code>	1591,		2701, 2715, 2718, 2721
	2223, 2229, 2234, 3252	<code>\itshape</code>	608, 3223
<code>\hinweis</code>	1995	<b>J</b>	
<code>\hline</code>	2460	<code>\j</code>	1515, 1516, 1518, 1519,
<code>\horizontale</code>	1556, 1599		1520, 1525, 1526, 1527
<code>\href</code>	1368, 2124,	<b>K</b>	
	2291, 2326, 2335, 3080	<code>\k</code>	1525
<code>\hspace</code>	2652, 3403, 3411	<code>\keys</code>	41, 75,
<code>\ht</code>	2901		87, 117, 127, 175,
<code>\Huge</code>	1701, 3281		185, 316, 627, 631,
<code>\huge</code>	1644, 3275		645, 650, 1379, 1386
<b>I</b>			
<code>\i</code>	1525, 1526	<b>L</b>	
<code>\ifcase</code>	2809	<code>\l</code>	68, 69, 70, 71,
<code>\ifLiADDITUM</code>	229, 2035, 2049		72, 73, 76, 77, 78,
<code>\ifLiANTWORT</code>	233, 2000, 2028		79, 80, 82, 84, 89,
<code>\ifLiEXKURS</code>	231, 2054		90, 91, 92, 93, 94,
<code>\ifmmode</code>	657, 665, 673,		109, 110, 111, 112,
	681, 688, 1294,		113, 114, 115, 118,
	1308, 1344, 2142,		119, 120, 121, 122,
	2152, 2162, 2176,		123, 124, 130, 131,
	2499, 2581, 2642, 3394		132, 133, 134, 135,
<code>\IfNoValueTF</code>	2040, 2120, 2124		136, 167, 168, 169,
			170, 171, 172, 173,
			176, 177, 178, 179,
			180, 181, 182, 188,
			189, 190, 191, 192,
			193, 194, 625, 628,
			633, 634, 637, 642,
			643, 646, 647, 652,
			1374, 1375, 1376,
			1382, 1383, 1389,
			1390, 1391, 1392,
			1904, 1905, 1906,
			2080, 2081, 2082, 2089
		<code>\labelenumi</code>	1416
		<code>\labelenumii</code>	1417
		<code>\labelitemi</code>	1410
		<code>\labelitemii</code>	1411
		<code>\labelitemiii</code>	1412
		<code>\labelitemiv</code>	1413
		<code>\land</code>	3431, 3433
		<code>\LARGE</code>	1405, 1710
		<code>\Large</code>	1584
		<code>\large</code>	1575, 1684, 1769, 2964
		<code>\leaders</code>	3252
		<code>\left</code>	2136, 2575
		<code>\LEFTTarrow</code>	3313
		<code>\leftarrow</code>	671
		<code>\leftouterjoin</code>	2903
		<code>\leftskip</code>	3415, 3416, 3426
		<code>\LehramtInformatikAutorEmail</code>	1920
		<code>\LehramtInformatikAutorName</code>	1919
		<code>\LehramtInformatikGitBranch</code>	464, 3048
		<code>\LehramtInformatikGithubCodeRepo</code>	3047
		<code>\LehramtInformatikGithubDomain</code>	461, 3044
		<code>\LehramtInformatikGithubRawDomain</code>	3045
		<code>\LehramtInformatikGithubTexRepo</code>	462, 3046
		<code>\LehramtInformatikRepository</code>	17, 20, 23, 26, 29,
			1451, 1662, 1940,
			1941, 1942, 1943,
			1944, 1945, 1946,
			1947, 1948, 1949, 3043
		<code>\LehramtInformatikTitel</code>	1915
		<code>\leq</code>	2238, 2698, 2718
		<code>\let</code>	1304,
			1305, 1966, 2849,
			2850, 2851, 2852,
			2865, 2867, 2868,
			2869, 2871, 2872,
			2873, 2874, 2884,
			2976, 3014, 3015, 3244
		<code>\li@chomsky@erklaerung@texte</code>	574, 610
		<code>\li@EntwurfsCode</code>	752, 803, 804, 805,
			854, 855, 856, 857,
			923, 924, 925, 926,
			927, 928, 955, 956,
			957, 958, 959, 960,
			1026, 1196, 1197,
			1198, 1199, 1239, 1240

<code>\li@EntwurfsCodeAllgemein</code>	<code>\liChomskyUeberErklaerung</code>	<code>\liEntwurfsEinzelstueckBeschreibung</code>
..... <a href="#">751</a>	..... <a href="#">613</a>	..... <a href="#">1002</a> , <a href="#">1029</a>
<code>\li@fussnote@text</code> <a href="#">2411</a> ,	<code>\liChomskyUeberschrift</code>	<code>\liEntwurfsEinzelstueckCode</code>
<a href="#">2417</a> , <a href="#">2421</a> , <a href="#">2425</a> , <a href="#">2429</a>	..... <a href="#">562</a> , <a href="#">614</a>	..... <a href="#">1025</a> , <a href="#">1035</a>
<code>\li@GithubLink</code> .....	<code>\liCpmEreignis</code> .....	<code>\liEntwurfsEinzelstueckUml</code>
<a href="#">3076</a> , <a href="#">3089</a> , <a href="#">3099</a> , <a href="#">3110</a>	<a href="#">623</a>	..... <a href="#">1007</a> , <a href="#">1031</a>
<code>\li@mget</code> . <a href="#">1502</a> , <a href="#">1506</a> , <a href="#">1526</a>	<code>\liCpmFruehErklaerung</code> <a href="#">694</a>	<code>\liEntwurfsErbauer</code> . <a href="#">1085</a>
<code>\li@minc</code> .....	<code>\liCpmFruehI</code> ... <a href="#">687</a> , <a href="#">707</a>	<code>\liEntwurfsErbauerAkteure</code>
<a href="#">1505</a> , <a href="#">1527</a>	<code>\liCpmSpaetErklaerung</code> <a href="#">712</a>	..... <a href="#">1057</a> , <a href="#">1087</a>
<code>\li@mset</code> .....	<code>\liCpmSpaetI</code> ... <a href="#">680</a> , <a href="#">725</a>	<code>\liEntwurfsErbauerUml</code>
<a href="#">1499</a> , <a href="#">1507</a> , <a href="#">1516</a> , <a href="#">1519</a>	<code>\liCpmVon</code> .....	..... <a href="#">1037</a> , <a href="#">1086</a>
<code>\li@numdiscs</code> .....	<code>\liCpmVonOhneMathe</code> ..	<code>\liEntwurfsFabrikmethode</code>
... <a href="#">1511</a> , <a href="#">1520</a> , <a href="#">1526</a>	..... <a href="#">663</a> , <a href="#">666</a> , <a href="#">668</a>	..... <a href="#">1131</a> , <a href="#">1156</a>
<code>\li@Rmodell@Schrift</code> .	<code>\liCpmVonZu</code> .....	<code>\liEntwurfsFabrikmethodeAkteure</code>
... <a href="#">2914</a> , <a href="#">2923</a> , <a href="#">2933</a>	<a href="#">655</a>	..... <a href="#">1106</a> , <a href="#">1133</a>
<code>\li@sequence</code> .. <a href="#">1512</a> , <a href="#">1525</a>	<code>\liCpmVonZuOhneMathe</code>	<code>\liEntwurfsFabrikmethodeUml</code>
<code>\li@synthese@erklaerung@text</code>	..... <a href="#">655</a> , <a href="#">658</a> , <a href="#">660</a>	..... <a href="#">1089</a> , <a href="#">1132</a>
..... <a href="#">3165</a> , <a href="#">3225</a>	<code>\liCpmVorgang</code> .....	<code>\liEntwurfsKompositum</code>
<code>\liAbleitung</code> .....	<code>\liCpmZu</code> .....	..... <a href="#">1156</a>
<a href="#">1329</a>	<a href="#">671</a>	<code>\liEntwurfsKompositumAkteure</code>
<code>liAdditum (environment)</code>	<code>\liCpmZuOhneMathe</code> ...	..... <a href="#">1158</a>
..... <a href="#">2033</a>	..... <a href="#">671</a> , <a href="#">674</a> , <a href="#">676</a>	<code>\liEntwurfsKompositumUml</code>
<code>\LiADDITUMfalse</code> .....	<code>liDiagramm (environ-</code>	..... <a href="#">1135</a> , <a href="#">1157</a>
<a href="#">229</a>	<code>ment)</code> .....	<code>\liEntwurfsModellPraesentationSteuerung</code>
<code>\LiADDITUMtrue</code> . <a href="#">229</a> , <a href="#">1547</a>	<a href="#">2107</a>	..... <a href="#">1177</a>
<code>liAHuelle (environment)</code>	<code>liEinbettung (environ-</code>	<code>\liEntwurfsModellPraesentationSteuerungAk</code>
..... <a href="#">2506</a>	<code>ment)</code> .....	<code>\liEntwurfsModellPraesentationSteuerungUm</code>
<code>\liAlphabet</code> .....	<a href="#">1997</a>	..... <a href="#">1160</a> , <a href="#">1178</a>
<a href="#">1314</a>	<code>\liEntwurfs</code> .....	<code>\liEntwurfsStellvertreter</code>
<code>liAntwort (environment)</code>	<code>\liEntwurfsAbstrakteFabrik</code>	..... <a href="#">1201</a>
..... <a href="#">1998</a>	..... <a href="#">807</a>	<code>\liEntwurfsStellvertreterCode</code>
<code>\LiANTWORTfalse</code> .....	<code>\liEntwurfsAbstrakteFabrikBeschreibung</code> .....	..... <a href="#">1195</a> , <a href="#">1203</a>
<a href="#">233</a>	..... <a href="#">755</a> , <a href="#">808</a>	<code>\liEntwurfsStellvertreterUml</code>
<code>\LiANTWORTtrue</code> .....	<code>\liEntwurfsAbstrakteFabrikCode</code>	..... <a href="#">1181</a> , <a href="#">1202</a>
<a href="#">233</a>	..... <a href="#">802</a> , <a href="#">812</a>	<code>\liEntwurfsZustand</code> . <a href="#">1242</a>
<code>\liAnweisung</code> .....	<code>\liEntwurfsAbstrakteFabrikUml</code>	<code>\liEntwurfsZustandAkteure</code>
<a href="#">1895</a>	..... <a href="#">760</a> , <a href="#">810</a>	..... <a href="#">1220</a> , <a href="#">1244</a>
<code>\liAssemblerCode</code> ... <a href="#">3114</a>	<code>\liEntwurfsAdapter</code> .. <a href="#">859</a>	<code>\liEntwurfsZustandCode</code>
<code>\liAssemblerDatei</code> .. <a href="#">3115</a>	<code>\liEntwurfsAdapterAkteure</code>	..... <a href="#">1238</a> , <a href="#">1245</a>
<code>\liAttribut</code> .....	..... <a href="#">829</a> , <a href="#">861</a>	<code>\liEntwurfsZustandUml</code>
<a href="#">2933</a>	<code>\liEntwurfsAdapterCode</code>	..... <a href="#">1205</a> , <a href="#">1243</a>
<code>\liAttributHuelle</code> ...	..... <a href="#">853</a> , <a href="#">862</a>	<code>\liEpsilon</code> .....
... <a href="#">2497</a> , <a href="#">3177</a> , <a href="#">3182</a>	<code>\liEntwurfsAdapterUml</code>	<code>\liErAttribute</code> .....
<code>\liAttributHuelleOhneMathe</code>	..... <a href="#">814</a> , <a href="#">860</a>	... <a href="#">1258</a> , <a href="#">1272</a> , <a href="#">1274</a>
..... <a href="#">2497</a> , <a href="#">2500</a> ,	<code>\liEntwurfsBeobachter</code> <a href="#">930</a>	<code>\liErDatenbankName</code> . <a href="#">1277</a>
<a href="#">2502</a> , <a href="#">2516</a> , <a href="#">2526</a> , <a href="#">2534</a>	<code>\liEntwurfsBeobachterAkteure</code>	<code>\liErEntity</code> <a href="#">1256</a> , <a href="#">1260</a> , <a href="#">1262</a>
<code>\liAttributMenge</code> ....	..... <a href="#">888</a> , <a href="#">932</a>	<code>\liErledigt</code> .....
<a href="#">2505</a> , <a href="#">2517</a> , <a href="#">2520</a> ,	<code>\liEntwurfsBeobachterCode</code>	<code>\liErMpAttribute</code> ... <a href="#">1271</a>
<a href="#">2527</a> , <a href="#">2528</a> , <a href="#">2542</a> , <a href="#">2544</a>	..... <a href="#">922</a> , <a href="#">933</a>	<code>\liErMpEntity</code> .....
<code>\liAufgabe</code> .....	<code>\liEntwurfsBeobachterUml</code>	<code>\liErMpRelationship</code> <a href="#">1265</a>
<a href="#">16</a>	..... <a href="#">864</a> , <a href="#">931</a>	... <a href="#">1257</a> , <a href="#">1266</a> , <a href="#">1268</a>
<code>\liAufgabenMetadaten</code>	<code>\liEntwurfsDekorierer</code> <a href="#">962</a>	<code>\liExamensAufgabe</code> .... <a href="#">19</a>
..... <a href="#">47</a> , <a href="#">1569</a> , <a href="#">1676</a>	<code>\liEntwurfsDekoriererAkteure</code>	<code>\liExamensAufgabeA</code> ... <a href="#">28</a>
<code>\liAufgabenTitel</code> .... <a href="#">55</a>	..... <a href="#">964</a>	<code>\liExamensAufgabeTA</code> .. <a href="#">25</a>
<code>\liAusdruck</code> .....	<code>\liEntwurfsDekoriererCode</code>	<code>\liExamensAufgabeTTA</code> . <a href="#">22</a>
<a href="#">1351</a>	..... <a href="#">954</a> , <a href="#">965</a>	<code>\liExkurs (environment)</code> <a href="#">2053</a>
<code>\liAutomat</code> .....	<code>\liEntwurfsDekoriererUml</code>	<code>\LiEXKURSfalse</code> .....
<a href="#">66</a>	..... <a href="#">935</a> , <a href="#">963</a>	<a href="#">231</a>
<code>\liAutomatenKante</code> .... <a href="#">98</a>	<code>\liEntwurfsEinfacheFabrik</code>	<code>\LiEXKURStrue</code> .....
<code>\liBandAlphabet</code> .... <a href="#">1315</a>	..... <a href="#">998</a>	<a href="#">231</a>
<code>\liBedingung</code> .....	<code>\liEntwurfsEinfacheFabrikAkteure</code>	<code>\liFalsch</code> .....
<a href="#">1896</a>	..... <a href="#">982</a> , <a href="#">1000</a>	<a href="#">555</a>
<code>\liBedingungDrei</code> ....	<code>\liEntwurfsEinfacheFabrikUml</code>	<code>\liFlaci</code> .....
... <a href="#">2191</a> , <a href="#">2234</a> , <a href="#">2274</a>	..... <a href="#">967</a> , <a href="#">999</a>	<a href="#">1361</a>
<code>\liBedingungEins</code> ....	<code>\liEntwurfsEinzelstueck</code>	<code>\liFremd</code> .....
... <a href="#">2185</a> , <a href="#">2223</a> , <a href="#">2266</a>	..... <a href="#">1028</a>	<a href="#">2913</a>
<code>\liBedingungFalsch</code> . <a href="#">1898</a>	<code>\liEntwurfsEinzelstueckAkteure</code>	
<code>\liBedingungWahr</code> ... <a href="#">1897</a>	..... <a href="#">1017</a> , <a href="#">1033</a>	
<code>\liBedingungZwei</code> ....		
... <a href="#">2188</a> , <a href="#">2229</a> , <a href="#">2270</a>		
<code>\liBeschriftung</code> .... <a href="#">1987</a>		
<code>\liBindeAufgabeEin</code> . <a href="#">1659</a>		
<code>\liChomskyErklaerung</code>		
..... <a href="#">574</a> , <a href="#">615</a>		

<code>\liFunktionaleAbhaengigkeit</code>	1759, 2311, 2409, ... 2536, 2539, 2547	<code>\liPetriSetzeSchluessel</code>	2593
<code>\liFunktionaleAbhaengigkeiten</code>	2550	<code>\liPetriTransitionsName</code>	2640, 2652
<code>\liFussnote</code>	2410, 2412	<code>\liPetriTransitionsNameOhneMathe</code>	2640, 2643, 2645
<code>\liFussnoteDreiText</code>	2424, 2442	<code>\liPetriTransPfeile</code>	2652
<code>\liFussnoteEinsText</code>	2416, 2436	<code>\liPolynomiellReduzierbar</code>	1779
<code>\liFussnoteLink</code>	2123	<code>\liPotenzmenge</code>	1301, 1305, 2663
<code>\liFussnoten</code>	2432	<code>\liPotenzmengeOhneMathe</code>	1302, 1303, 1304
<code>\liFussnoteUrl</code>	1175, 2119	<code>\liPrimaer</code>	2912
<code>\liFussnoteVierText</code>	2428, 2445	<code>\liProblemBeschreibung</code>	1763
<code>\liFussnoteZweiText</code>	2420, 2439	<code>\liProblemClique</code>	1786
<code>\liGeschweifteKlammern</code>	1333, 2551, 3256	<code>\liProblemName</code>	1762, 1769, 1781, 1783, 1796, 1807, 1808, 1816, 1817
<code>\liGrafikCCLizenz</code>	1453, 2395	<code>\liProblemSat</code>	1815
<code>\liGrafikLogo</code>	1458, 1693, 2371	<code>\liProblemSubsetSum</code>	1806, 1815
<code>\liGrafikLogoPfad</code>	1450, 1455, 1460	<code>\liProblemVertexCover</code>	1786, 1794
<code>\liGrammatik</code>	1372	<code>\liProduktionen</code>	1340, 1382
<code>liGraphenFormat</code> (environment)	1492	<code>liProduktionsRegeln</code> (environment)	1330
<code>\liHanoi</code>	1499	<code>liProjektSprache</code> (environment)	1996
<code>\liHaskellCode</code>	3121	<code>\liPseudoUeberschrift</code>	1973, 2042, 2045, 2457, 2467, 3340, 3348, 3356
<code>\liHaskellDatei</code>	3122	<code>\liPumpingKontextfrei</code>	2709
<code>\liInduktionAnfang</code>	3339	<code>\liPumpingRegulaer</code>	2688
<code>\liInduktionErklaerung</code>	3338	<code>liQuellen</code> (environment)	2077
<code>\liInduktionMarkierung</code>	3337	<code>\liRechtsReduktionInline</code>	2531
<code>\liInduktionSchritt</code>	3355	<code>\liRekursionsGleichung</code>	2182, 2245
<code>\liInduktionVoraussetzung</code>	3347	<code>\liRelation</code>	2561
<code>liJavaAngabe</code> (environment)	3062	<code>liRelationenSchemaFormat</code> (environment)	2934
<code>\liJavaCode</code>	3065	<code>\liRelationMenge</code>	2928
<code>\liJavaDatei</code>	753, 3083	<code>\liRichtig</code>	554
<code>\liJavaExamen</code>	3103	<code>liRmodell</code> (environment)	2914
<code>\liJavaTestDatei</code>	3093	<code>\liRundeKlammer</code>	2135, 2139, 2149, 2159, 2173, 2574, 2578
<code>liKasten</code> (environment)	1420	<code>\liSetzeExamenTeilaufgabeNr</code>	1655
<code>\liKellerAutomat</code>	107	<code>\liSetzeExamenThemaNr</code>	1651
<code>\liKellerKante</code>	145	<code>\liSortierMarkierung</code>	2953
<code>\liKellerUebergang</code>	140, 146	<code>\liSortierPfeil</code>	2947
<code>\liKontrollCode</code>	1899	<code>\liSortierPfeilUnten</code>	2950
<code>liKontrollflussgraph</code> (environment)	1887	<code>\liSpaltenUmbruch</code>	3022
<code>\liKontrollKnotenPfad</code>	1901	<code>\liSqlCode</code>	3126
<code>\liKontrollTextzeileKnoten</code>	1900, 1905		
<code>\liKurzeTabellenLinie</code>	735		
<code>\liLadeAllePakete</code>	239, 1647		
<code>\liLadePakete</code>	59, 62, 235, 240, 561, 622, 1291, 1535, 1633,		
<code>\liLatexCode</code>	3075		
<code>\liLeereZelle</code>	2447		
<code>liLernkartei</code> (environment)	2097		
<code>\liLinksReduktion</code>	2514		
<code>\liLinksReduktionInline</code>	2523, 2531		
<code>\liLogoTextCCLizenz</code>	1615, 1748, 2391		
<code>\liLogoTextProjekt</code>	1612, 1745, 2367		
<code>\liMasterExkurs</code>	2279		
<code>\liMasterFaelle</code>	2218, 2286		
<code>\liMasterFallRechnung</code>	2264		
<code>\liMasterVariablen</code>	2195, 2281		
<code>\liMasterVariablenDeklaration</code>	2241		
<code>\liMasterWolframLink</code>	2289		
<code>\liMenge</code>	76, 77, 79, 118, 119, 120, 124, 176, 177, 178, 182, 1292, 1341, 1380, 1381		
<code>\liMengeOhneMathe</code>	1292, 1295, 1297		
<code>\liMetaBsclangaulSammlung</code>	2312, 2376		
<code>\liMetaCCLink</code>	2324, 2400		
<code>\liMetaEmailLink</code>	2334, 2350		
<code>\liMetaHermineBsclangaulAndFriends</code>	1696, 2315, 2380		
<code>\liMetaHilfMit</code>	1621, 2341, 2352		
<code>\liMetaQuelltext</code>	1623, 2352		
<code>\liMetaSetze</code>	36, 48, 1570, 1677		
<code>\liMetaUeberDasProjekt</code>	2318, 2387		
<code>\liMinimierungErklaerung</code>	2469		
<code>\liMinispracheDatei</code>	3118		
<code>\linespread</code>	2921		
<code>\liNichtsZuTun</code>	3245		
<code>\liO</code>	2158, 2186		
<code>\liOmega</code>	2148, 2192		
<code>\liOmegaOhneMathe</code>	2148, 2153, 2155		
<code>\liONotationO</code>	2577		
<code>\liOOhneMathe</code>	2158, 2163, 2165		
<code>\liParagraphMitLinien</code>	610, 695, 713, 2471, 3225, 3246, 3343, 3351, 3359		
<code>\liPetriErreichKnotenDrei</code>	2651		
<code>\liPetriErreichTransition</code>	2648		







1685, 1689, 1693, 1695, 1696, 1703, 1705, 1710, 1714, 1718, 1976, 1982, 1988, 2069, 2093, 2102, 2378, 2381, 2853, 2875, 2890, 2931, 3229, 3253, 3257, 3265, 3418, 3424	\QS@sort@b . . . . . 2808, 2809 \QS@sort@c . . . . . 2812, 2819 \QS@sort@d . . . . . 2820, 2828 \QS@sort@empty . 2810, 2815 \QS@sort@single 2811, 2816 \QSinitialize . . . . . . . . 2733, 2845, 2880 \QSIr . 2816, 2822, 2830, 2850, 2864, 2869, 2872 \QSirr 2851, 2864, 2865, 2873 \QSLr . . . . . 2822, 2829, 2840, 2841, 2849, 2862, 2867, 2871 \QSpivotStep . . . . . 2735, 2845, 2849, 2860 \QSR . . . . . 2822 \QSRr . . . . . 2831, 2852, 2863, 2868, 2874, 2883, 2884, 2885 \QSSortStep . . . . . 2737, 2845, 2861, 2862 \quad . . . . . 2413	\rightarrow . . . . . 212, 578, 583, 591, 595, 597, 598, 600, 655, 663, 2652, 3174, 3181, 3183, 3186, 3191, 3196, 3197, 3202 \rightarrowouterjoin . . . . . 2904 \rmfamily . . . . . 565, 1526, 3152, 3269 \Roman . . . . . 1415 \roman . . . . . 1415, 1417 \romannumeral . . . . . 2820 \rtimes . . . . . 2297 \rule . . . . . 1559, 2853, 2875, 2890, 2901
\paragraph . . . . . 1407 \parindent . . . . . 2922 \path . . . . . 99, 146, 209, 652 \pgfkeys . . 2604, 3310, 3311, 3312, 3313, 3314, 3317, 3320, 3322 \pgfmath@count . . . . . . . . 2975, 2977, 2979 \pgfmath@smuggleone 2980 \pgfmathdeclarefunction . . . . . 2974 \pgfmathint . . . . . 2975 \pgfmathparse . . . . . . . . . . 1506, 2992, 2997, 3000, 3014, 3015 \pgfmathresult . . . . . . . . . . 1507, 2975, 2976, 2978, 2980, 2993, 3001, 3014, 3015 \pgfutil@empty . . . . . 2976 \pgfutil@loop . . . . . 2977 \pgfutil@repeat . . . . . 2980 \preceq . . . . . 1782 \prime . . . . . 1761 \printbibliography . 1952 \ProvidesPackage . . . . . 2, 15, 33, 58, 226, 490, 551, 558, 620, 733, 748, 1249, 1285, 1398, 1436, 1445, 1466, 1495, 1533, 1631, 1756, 1828, 1911, 1927, 1933, 1955, 2132, 2295, 2308, 2407, 2486, 2569, 2589, 2655, 2680, 2685, 2729, 2895, 2908, 2937, 3018, 3025, 3032, 3037, 3129, 3144, 3235, 3239, 3288, 3330, 3367, 3387	\R . . . . . \raisebox . . . . . 1900 \relax . . . . . 1966, 2822, 2865, 2867, 2868, 2869, 2977, 2979 \renewcommand . . . . . 1410, 1411, 1412, 1413, 1416, 1417, 1431, 1432, 1729, 1731, 1921, 1922, 2455 \repeat . . . . . 2738 \RequirePackage . 4, 60, 163, 227, 237, 486, 492, 493, 553, 621, 750, 751, 1251, 1253, 1254, 1290, 1399, 1400, 1402, 1404, 1409, 1418, 1425, 1433, 1437, 1438, 1448, 1449, 1467, 1468, 1469, 1497, 1498, 1546, 1643, 1648, 1649, 1735, 1760, 1829, 1929, 1934, 1935, 1951, 1958, 1959, 1960, 1961, 2134, 2298, 2299, 2571, 2572, 2573, 2590, 2592, 2682, 2742, 2744, 2896, 2897, 2898, 2911, 2939, 2970, 3021, 3027, 3034, 3039, 3050, 3051, 3131, 3236, 3243, 3291, 3292, 3294, 3295, 3333, 3334, 3335, 3368, 3388 \right . . . . . 2136, 2575 \RIGHTarrow . . . 3309, 3314 \rightarrow . . . . . 739, 744	S \sb . . . . . 72, 82, 84, 113, 171, 590, 591, 595, 598, 599, 600, 682, 684, 689, 691, 1345, 1347, 2186, 2189, 2192, 2221, 2227, 2475, 2640, 2649, 3196, 3197, 3198, 3203, 3207, 3208, 3211, 3212, 3215, 3216, 3217 \scriptscriptstyle . . . . . . . 655, 663, 671 \scriptsize 1364, 1840, 1847, 1853, 1915, 1916, 1919, 1920, 2399, 3338, 3391, 3421 \section . . . 52, 1644, 1653 \sectionbreak . . . . . 1645 \seq . 1904, 1905, 1906, 2080, 2081, 2082, 2089 \setbox . . . . . 2900 \setcounter 1408, 1646, 1734, 2854, 2876, 2890 \setganttlinklabel . . 1439, 1440, 1441, 1442 \setlength . . . . . 1727, 1728, 1730, 1923, 2922, 3415, 3416, 3426 \setmainfont . . . . . 1401 \setmainlanguage . . . . 487 \setminted . . . . . 3056, 3057 \setminus . . . . . 2535 \setul . . . . . 2913 \shoveleft . . . . . 2515 \shoveright . . . . . 2519 \Sigma . . . . . 69, 110, 168, 1314, 1315, 1375 \sigma . . . . . 588, 590, 591 \SLASH . . . . . 1972 \small . . . . . 2110 \sort . . . . . 2987 \sortList . . . . . 2986, 2995 \square . . . . . 555 \stepcounter 2774, 2779, 2784, 2787, 2789, 2793, 2795, 2799, 2801
Q \QS@list . . . . . 2855, 2866, 2870, 2877, 2883, 2888, 2891 \QS@select@equal . . . . . . . . . . 2830, 2834 \QS@select@greater . . . . . . . 2831, 2835 \QS@select@smaller . . . . . 2826, 2829, 2833 \QS@sort@a . . . . . 2808, 2841, 2862, 2863		

<code>\stichwoerter</code> .. 1549, 1597	180, 181, 182, 309,	1162, 1163, 1164,
<code>\str</code> 367, 566, 575, 2001,	313, 340, 344, 345,	1185, 1186, 1187,
2664, 2673, 3153, 3166	346, 349, 354, 355,	1207, 1208, 1209, 1210
<code>\string</code> ..... 2517, 2527	356, 368, 377, 412,	<code>\umldep</code> ..... 1103
<code>\StrSubstitute</code> . 2994, 2996	415, 418, 426, 427,	<code>\umlHVHaggreg</code> .....
<code>\strut</code> .... 2243, 2247,	428, 429, 440, 446,	..... 884, 950, 1153
2251, 2255, 2259, 3022	449, 452, 468, 475,	<code>\umlinherit</code> ..... 823,
<code>\subsection</code> ..... 1657	625, 628, 633, 634,	874, 1048, 1093, 1101
<code>\subseteq</code> 3177, 3210, 3217	642, 643, 646, 647,	<code>\umlnote</code> .. 825, 1050, 1217
<code>\subsubsection</code> ..... 1678	1353, 1374, 1375,	<code>\umlreal</code> ..... 821, 882
	1376, 1377, 1380,	<code>\umlsimpleclass</code> .....
	1381, 1382, 1383,	... 777, 778, 779,
<b>T</b>	1550, 1563, 1572,	783, 785, 786, 787,
<code>\tableofcontents</code> ....	1585, 1652, 1656,	816, 969, 970, 971,
..... 1732, 1967	1660, 1667, 1670, 2169	1039, 1091, 1092, 1183
<code>\TeX</code> ..... 2353	<code>\tmp</code> ..... 2537	<code>\umlstatic</code> .... 976, 1010
<code>\text</code> ..... 82, 84,	<code>\TmpPlaceEight</code> ..... 2612	<code>\umluniaggreg</code> ..... 1046
187, 2497, 3338, 3391	<code>\TmpPlaceFive</code> ..... 2609	<code>\umluniassoc</code> .....
<code>\textbf</code> ..... 702, 720,	<code>\TmpPlaceFour</code> ..... 2608	... 797, 822, 1047,
1256, 1551, 1787,	<code>\TmpPlaceNine</code> ..... 2613	1171, 1172, 1191, 1192
1796, 1807, 1816,	<code>\TmpPlaceOne</code> ..... 2605	<code>\umlVHuniassoc</code> .. 798, 799
1978, 1991, 2021,	<code>\TmpPlaceSeven</code> ..... 2611	<code>\umlVHVdep</code> ..... 791,
2069, 2086, 2102, 2460	<code>\TmpPlaceSix</code> ..... 2610	792, 794, 795, 978, 979
<code>\textcolor</code> 1899, 3067, 3337	<code>\TmpPlaceTen</code> ..... 2614	<code>\umlVHVinherit</code> 774, 775,
<code>\textit</code> ... 1096, 1138,	<code>\TmpPlaceThree</code> ..... 2607	780, 781, 788, 789,
1139, 1140, 1141,	<code>\TmpPlaceTwo</code> ..... 2606	947, 948, 972, 973,
1564, 2112, 2505, 2565	<code>\TmpScale</code> ..... 2625	1151, 1152, 1189, 1190
<code>\textsc</code> ..... 1762	<code>\TmpTransitionEight</code> .	<code>\umlVHVreal</code> .....
<code>\textsf</code> ..... 2086	..... 2601, 2622	... 941, 942, 1212, 1213
<code>\textstyle</code> .... 2206, 2238	<code>\TmpTransitionFive</code> ..	<code>\UParrow</code> ..... 3311
<code>\texttt</code> 1279, 1762, 1896,	..... 2598, 2619	<code>\url</code> ..... 469, 2120
1897, 1898, 1899, 3391	<code>\TmpTransitionFour</code> ..	<code>\usemintedstyle</code> .... 3053
<code>\textwidth</code> .... 1559, 1923	..... 2597, 2618	<code>\usetikzlibrary</code> .. 61,
<code>\thematik</code> ..... 1562, 1591	<code>\TmpTransitionNine</code> ..	494, 1252, 1470,
<code>\thepage</code> ..... 1429, 1918	..... 2602, 2623	1830, 2591, 2899,
<code>\theparagraph</code> ..... 1407	<code>\TmpTransitionOne</code> ...	2940, 2971, 3296, 3384
<code>\thesection</code> ..... 1644	..... 2594, 2615	
<code>\Theta</code> ..... 2139	<code>\TmpTransitionSeven</code> .	<b>V</b>
<code>\thinspace</code> ..... 3391	..... 2600, 2621	<code>\value</code> ..... 2736
<code>\tikz</code> ..... 1900	<code>\TmpTransitionSix</code> ...	<code>\varepsilon</code> ..... 567,
<code>tikz: bbaum</code> ..... 26	..... 2599, 2620	578, 579, 1300,
<code>tikz: li binaer baum</code> .... 24	<code>\TmpTransitionTen</code> ...	2186, 2192, 2224, 2235
<code>\tikzchildnode</code> ..... 510	..... 2603, 2624	<code>\vfill</code> .... 1609, 1691,
<code>\tikzparentnode</code> ..... 510	<code>\TmpTransitionThree</code> .	1698, 1722, 1743, 3022
<code>\tikzset</code> ..... 101,	..... 2596, 2617	<code>\vrule</code> ..... 3248, 3252
148, 214, 495, 521,	<code>\TmpTransitionTwo</code> ...	<code>\vspace</code> ... 1687, 1708,
1471, 1831, 2630,	..... 2595, 2616	1712, 1716, 1729,
2756, 2960, 3133, 3369	<code>\TmpX</code> ..... 2626	2055, 2073, 3261, 3263
<code>\tikzumlset</code> ..... 3297	<code>\TmpY</code> ..... 2627	
<code>\times</code> ..... 212	<code>\today</code> ..... 1916	<b>X</b>
<code>\tiny</code> . 1262, 1268, 1274,	<code>\ttfamily</code> ..... 2914	<code>\xappto</code> .. 2998, 3004, 3008
1619, 1899, 1971, 3078		<code>\xdef</code> ..... 1500
<code>\titel</code> ..... 1680, 1737		<code>\xintApply</code> ..... 2824
<code>\titleformat</code> .....		<code>\xintApplyUnbraced</code> ..
... 1405, 1407, 1644	<b>U</b>	2823, 2829, 2830, 2831
<code>\titlespacing</code> ..... 1406	<code>\ul</code> ..... 1257, 2912, 2913	<code>\xintCSVtoList</code> ..... 2883
<code>\tl</code> . 39, 68, 69, 70, 71,	<code>\umlaggreg</code> ..... 1215	<code>\xintFor</code> .....
72, 73, 76, 77, 78,	<code>\umlassoc</code> ..... 1173	2773, 2778, 2783,
79, 80, 82, 84, 109,	<code>\umlclass</code> .....	2788, 2794, 2800, 2841
110, 111, 112, 113,	762, 766, 770, 817,	<code>\xintifEq</code> ..... 2834
114, 115, 118, 119,	818, 819, 866, 871,	<code>\xintifForLast</code> .....
120, 121, 122, 123,	876, 879, 937, 938,	... 2790, 2796, 2802
124, 167, 168, 169,	939, 944, 945, 974,	<code>\xintifGt</code> ..... 2835
170, 171, 172, 173,	1009, 1040, 1041,	<code>\xintifLt</code> ..... 2833
176, 177, 178, 179,	1044, 1095, 1098,	
	1137, 1143, 1144,	

\xintLength	.....	2808		<b>Z</b>		\zustandsnamens@liste
			\ZB	.....	<u>2127</u>	... 1318, 1325, 1326
\xintnthelt	.....	2820	\zB	.....	<u>2126</u>	