

lehramt-informatik

Hermine Bschlangaul *

September 10, 2021

Contents

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

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

2.11.8 Einzelstück (Singleton)	43
2.11.9 Erbauer (Builder)	44
2.11.10 Fabrikmethode (Factory Method)	45
2.11.11 Kompositum (Composite)	47
2.11.12 Modell-Präsentation-Steuerung (Model-View-Controller)	47
2.11.13 Stellvertreter (Proxy)	48
2.11.14 Zustand (State)	49
2.12 er.sty	51
2.12.1 Tex-Markup-Beispiel: Komplettes Diagramm	51
2.12.2 Tex-Markup-Beispiel: EER Enhanced Entity-Relation-Modell nach Elmasri/Navante	52
2.12.3 Makro-Kürzel	52
2.13 formale-sprachen.sty	54
2.14 formatierung.sty	57
2.14.1 Schriftarten / Typographie	57
2.14.2 Farben	57
2.14.3 Überschriften	57
2.14.4 Listen	57
2.14.5 Kasten	57
2.14.6 Header	57
2.14.7 Zeilenabstände	57
2.15 gantt.sty	59
2.16 grafik.sty	60
2.17 graph.sty	61
2.18 hanoi.sty	63
2.19 klassen-konfiguration-aufgabe.sty	64
2.20 klassen-konfiguration-examen.sty	67
2.21 komplexitaetstheorie.sty	69
2.21.1 Makro-Kürzel	69
2.22 kontrollflussgraph.sty	71
2.22.1 Makro-Kürzel	71
2.22.2 TeX-Markup-Beispiel	71
2.22.3 TikZ: pin	71
2.22.4 Umgebungen	72
2.22.5 Makros	73
2.23 kopf-fusszeilen.sty	74
2.24 literatur-dummy.sty	75
2.25 literatur.sty	76
2.26 makros.sty	77
2.27 master-theorem.sty	81
2.27.1 Makro-Kürzel	81
2.28 mathe.sty	85
2.29 meta.sty	86
2.30 minimierung.sty	88
2.31 normalformen.sty	91
2.31.1 Makro-Kürzel	91
2.32 o-notation.sty	94
2.32.1 Makro-Kürzel	94
2.32.2 TeX-Markup-Beispiel: Funktionsgraphen mit pgfplots	94
2.33 petri.sty	95
2.33.1 Makro-Kürzel	95
2.34 potenzmengen-konstruktion.sty	97
2.35 pseudo.sty	99
2.36 pumping-lemma.sty	100
2.37 quicksort.sty	101
2.38 relationale-algebra.sty	104
2.39 rmodell.sty	105
2.39.1 Makro-Kürzel	105

2.40	sortieren.sty	106
2.41	spalten.sty	108
2.42	sql.sty	109
2.43	struktogramm.sty	110
2.44	syntax.sty	111
2.44.1	Makro-Kürzel	111
2.45	syntaxbaum.sty	114
2.46	synthese-algorithmus.sty	115
2.46.1	Makro-Kürzel	115
2.46.2	TeX-Markup Grundgerüst	115
2.46.3	TeX-Markup Linksreduktion	115
2.46.4	TeX-Markup Rechtsreduktion	115
2.46.5	TeX-Markup Relationen formen	116
2.47	tabelle.sty	119
2.48	typographie.sty	120
2.49	uml.sty	122
2.50	vollstaendige-induktion.sty	124
2.50.1	Makro-Kürzel	124
2.51	wasserfall.sty	126
2.52	wpkalkuel.sty	127
2.52.1	Makro-Kürzel	127

3 Index 128

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-
24     #5/Aufgabe-#6.tex}
25 }

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

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

31
```

2.3 aufgaben-metadaten.sty

```

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

35 \ExplSyntaxOn

\liMetaSetze Setze die Metadaten einer Aufgabe.

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

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

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

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

54 \ExplSyntaxOff

\liAufgabenTitel Momentan eine dummy Makro das die Thematik enthält.

55 \def\liAufgabenTitel#1{}

56

```

2.4 automaten.sty

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

2.4.1 Endlicher Automat

```
59 \liLadePakete{formale-sprachen}

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

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



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

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

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

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

```

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

```

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

```

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

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

```

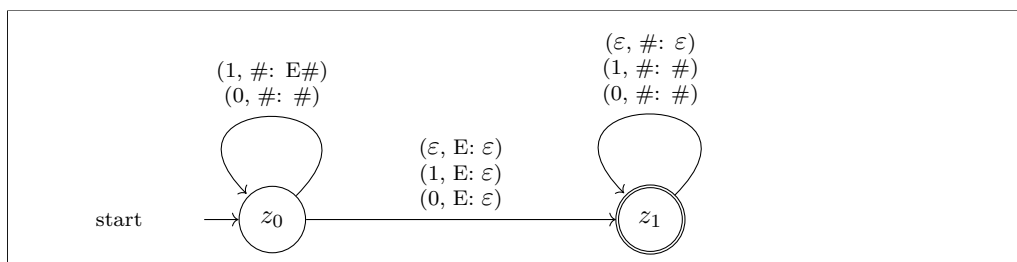
2.4.2 Kellerautomat

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

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

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

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



```
\liKellerAutomat \liKellerAutomat[\langle automaten-name \rangle]
\{ \langle zustaeende=Z, alphabet=\Sigma, kelleralphabet=\Gamma, delta=\delta, start=z_0, kellerboden=#, ende=E \rangle \}

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

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

```
107 \ExplSyntaxOn
108 \NewDocumentCommand {\liKellerAutomat} { O{K} m } {
109   \tl_set:Nn \l_zustaeende_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     zustaeende .code:n = {\tl_set:Nn \l_zustaeende_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, #: ε)

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

```

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

```

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

```

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

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

```

2.4.3 Turingmaschine

```

163 \RequirePackage{amssymb}

```

\liTuringLeerzeichen

□

```

164 \def\liTuringLeerzeichen{\Box}

```

```

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

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

```

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

```

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

```

\liTuringUebergangZelle Formatiert einen Zustandsübergang für eine Übergangstabelle.

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

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

```

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

```

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

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

`(z1: \square , L)`

`(\square : \square , R)`


```

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

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

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

\liTuringUeberfuehrung

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

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

```

2.5 basis.sty

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

227 \RequirePackage{xparse}

228 \ExplSyntaxOn
```

2.5.1 IFs

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

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

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

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

\liLadePakete

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

\liLadeAllePakete

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

```

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

```

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

```

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

```

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

```

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

```

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

```

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

```

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

```

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

```

```

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

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

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

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

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

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

```

```

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

```

Einzelprüfungsnummer / Jahr / Jahreszeit mit Trennzeichen

```

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

```

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

```

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

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

```

```

442     { 03 } { Frühjahr }
443     { 09 } { Herbst }
444 } \trenner:
445
446 \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
447     Thema ~ Nr. ~ \g_auf_examen_thema_nr_tl \trenner:
448 }
449 \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
450     Teilaufgabe ~ Nr. ~ \g_auf_examen_teilaufgabe_nr_tl \trenner:
451 }
452 \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} {
453     Aufgabe ~ Nr. ~ \g_auf_examen_aufgabe_nr_tl
454 }
455 \par
456 \bigskip
457 }
458 }
459 }
460 \cs_new:Npn \_gib_github_url: {
461     \LehramtInformatikGithubDomain /
462     \LehramtInformatikGithubTexRepo /
463     blob /
464     \LehramtInformatikGitBranch /
465     \g_auf_relativer_pfad_tl
466 }
467 \cs_new:Npn \_gib_github_url_href: {
468     \tl_if_empty:NTF \g_auf_relativer_pfad_tl {} {
469         \url{ \_gib_github_url: }
470     }
471 }
472 \cs_new:Npn \_gib_aufgaben_titel: {
473     \g_auf_titel_tl
474
475     \tl_if_empty:NTF \g_auf_thematik_tl
476     {}
477     {
478         \, ~ [
479             \g_auf_thematik_tl
480         ]
481     }
482 }
483 \msg_new:nnn { aufgabenmetadaten } { kein-titel }
484 { Der~Schlüssel~ist~zwingend~notwendig }
485
486 \def\LehramtInformatikRepository{/pfad/zum/repository}
487 \input /etc/lehramt-informatik.config.tex
488
489 biblatex not working with lualatex and babel
490 % \RequirePackage{polyglossia}
491 % \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-qtree.]

492 \RequirePackage{tikz}

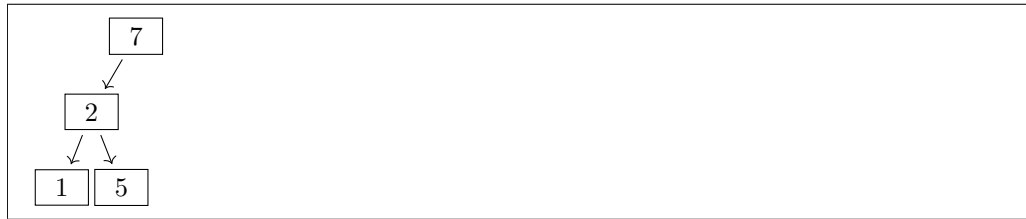
    für li binaer baum
493 \RequirePackage{tikz-qtree}

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

2.6.1 Binärbaum

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

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



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


2.6.2 AVL-Baum

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

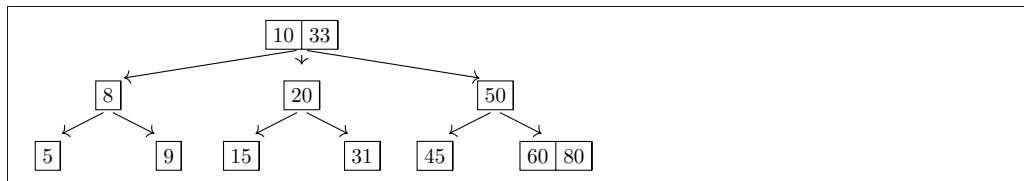


2.6.3 B-Baum

```

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

```



```

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

```

2.7 checkbox.sty

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

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

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

556
```

2.8 chomsky-normalform.sty

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

560 \ExplSyntaxOn

561 \liLadePakete{typographie}
```

2.8.1 Makro-Kürzel

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

2.8.2 TeX-Markup-Grundgerüst

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

2.8.3 Konkretes TeX-Markup-Beispiel

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

\liNichtsZuTun

\item \schrittE{2}

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

\item \schrittE{3}

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

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

```

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

```

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

```

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

```

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

```

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

```

```

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

```

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

```

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

```

```

617 \ExplSyntaxOff
618

```

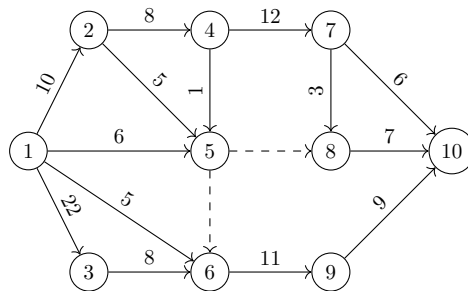
2.9 cpm.sty

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

2.9.1 Makro-Kürzel

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

2.9.2 TeX-Markup-Beispiel: Graph



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

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

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

2.9.3 TeX-Markup-Beispiel: Ergebnistabelle

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

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

```
\liCpmFruehErklaerung
\begin{tabular}{|l|l|r|}
\hline
i & Nebenrechnung & \FZ \\
1 & & 0 \\
2 & & 5 \\
3 & & 18
\end{tabular}
```

```

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

```

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

```

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

```

```

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

```

```

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

```

```

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

```

```

640 \ExplSyntaxOn
641 \NewDocumentCommand { \liCpmVorgang } { 0 } m m m } {
642   \tl_set:Nn \l_schein_tl {}
643   \tl_set:Nn \l_kritisch_tl {}
644
645   \keys_define:nn { cpmVorgang } {
646     schein .code:n = {\tl_set:Nn \l_schein_tl {dashed}},
647     kritisch .code:n = {\tl_set:Nn \l_kritisch_tl {thick}},
648   }
649
650   \keys_set:nn { cpmVorgang } { #1 }
651
652   \path[->,\l_schein_tl,\l_kritisch_tl] (#2) edge node[auto,sloped] {#4} (#3);
653 }
654 \ExplSyntaxOff

```


2.9.6 Hilfsmakros für Zeitpunkt-Tabelle

```

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

\liCpmVonZu Let-Abkürzung: \let\vz=\liCpmVonZu
\liCpmVonZu{1}(2-3):  $1_{(2 \rightarrow 3)}$ 
655 \def\liCpmVonZuOhneMathe#1(#2-#3){#1_{\scriptscriptstyle(#2\rightarrow#3)}}
656 \def\liCpmVonZu#1(#2-#3){%
657 \ifmmode%
658 \liCpmVonZuOhneMathe{#1}(\#2-\#3)%
659 \else%
660 $\liCpmVonZuOhneMathe{#1}(\#2-\#3)$%
661 \fi%
662 }

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

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

679 \ExplSyntaxOn

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

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

```

```

692 \fi
693 }

```

\liCpmFruehErklaerung

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

```

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

```

\liCpmSpaetErklaerung

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

```

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

730 \ExplSyntaxOff
731

```

2.10 cyk-algorithmus.sty

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

2.10.1 Makro-Kürzel

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

2.10.2 TeX-Markup-Beispiel

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

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

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

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

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

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

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

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

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

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

```
746
```

2.11 entwurfsmuster.sty

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

2.11.1 Namensschema der Entwurfsmuster-Makros:

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

2.11.2 Reihenfolge

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

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

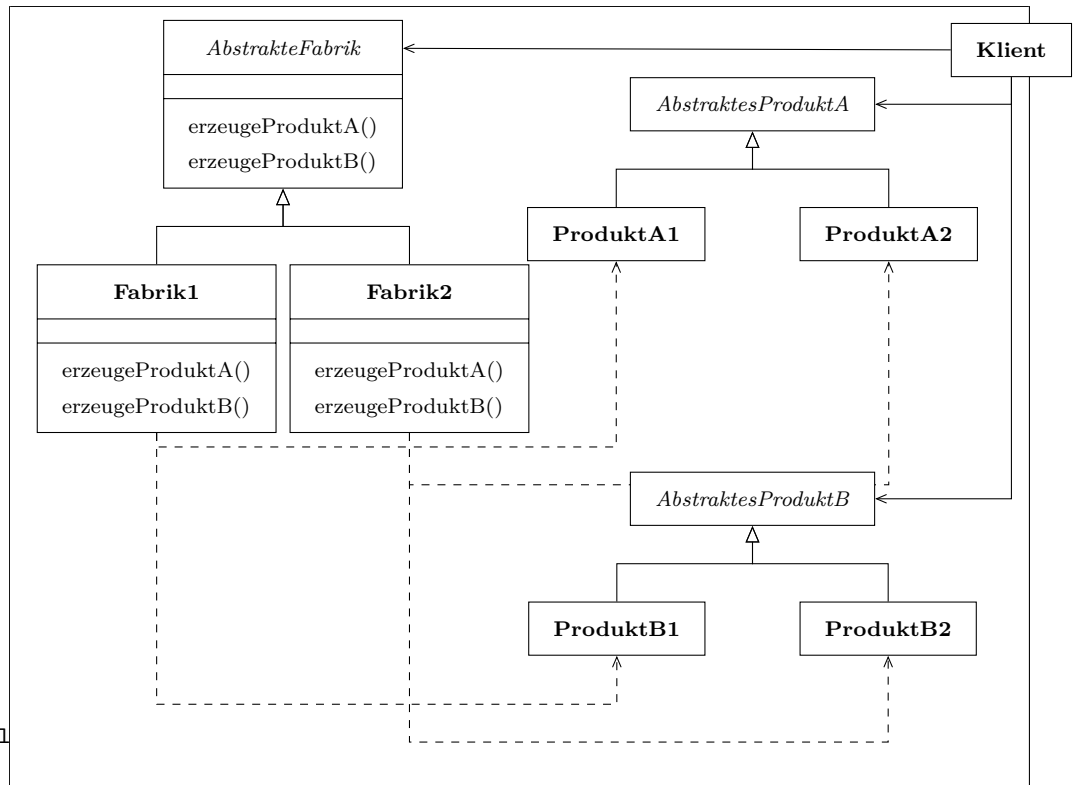
\li@EntwurfsCodeAllgemein Allgemeine Code-Beispiele zu den UML-Diagrammen und Stellvertretern

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

2.11.3 Abstrakte Fabrik (Abstract Factory)

sAbstrakteFabrikBeschreibung

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



```

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

```

```

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

```

iEntwurfsAbstrakteFabrikCode

```

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

```

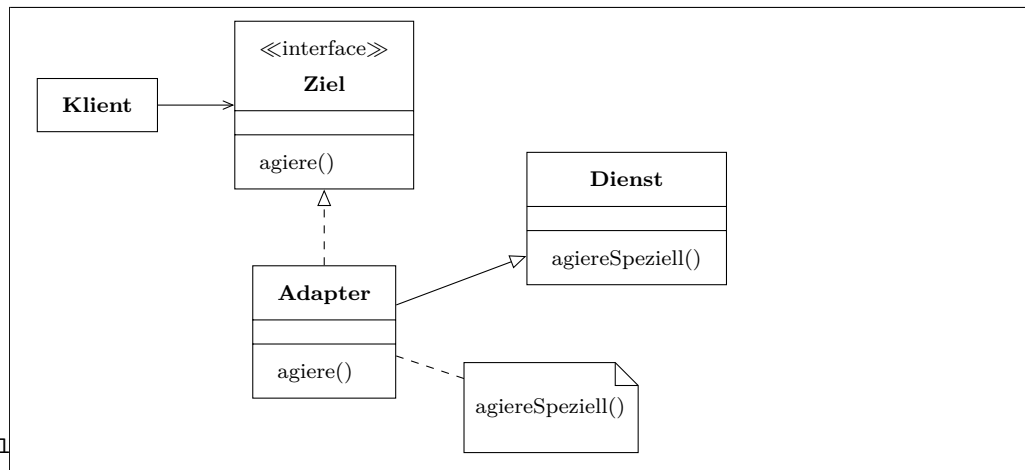
\liEntwurfsAbstrakteFabrik

```

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

```

2.11.4 Adapter



\liEntwurfsAdapterUml

```

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

```

\liEntwurfsAdapterAkteure

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

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

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

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

```

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

```

\liEntwurfsAdapterCode

```

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

```

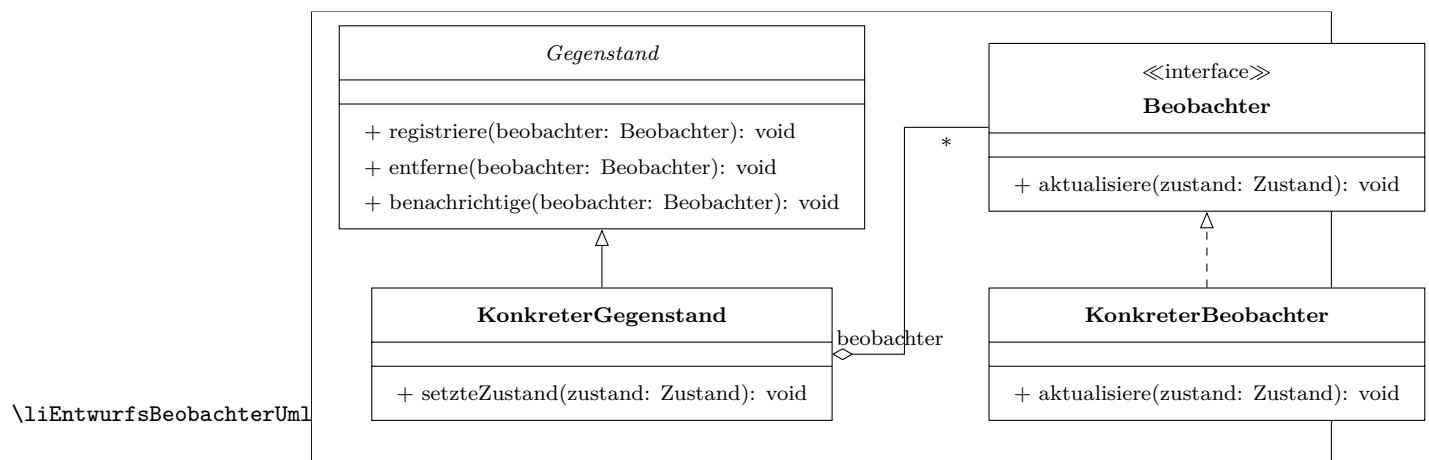
\liEntwurfsAdapter

```

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

```

2.11.5 Beobachter (Observer)



```

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

```

```

867     + registriere(beobachter: Beobachter): void\\
868     + entferne(beobachter: Beobachter): void\\
869     + benachrichtige(beobachter: Beobachter): void\\
870 }
871 \umlclass[x=0,y=-3]{KonkreterGegenstand}{-}{
872     + setzeZustand(zustand: Zustand): void
873 }
874 \umlinherit{KonkreterGegenstand}{Gegenstand}
875
876 \umlclass[x=8,y=0,type=interface]{Beobachter}{-}{
877     + aktualisiere(zustand: Zustand): void
878 }
879 \umlclass[x=8,y=-3]{KonkreterBeobachter}{-}{
880     + aktualisiere(zustand: Zustand): void
881 }
882 \umlreal{KonkreterBeobachter}{Beobachter}
883
884 \umlHVHagg[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 }

```

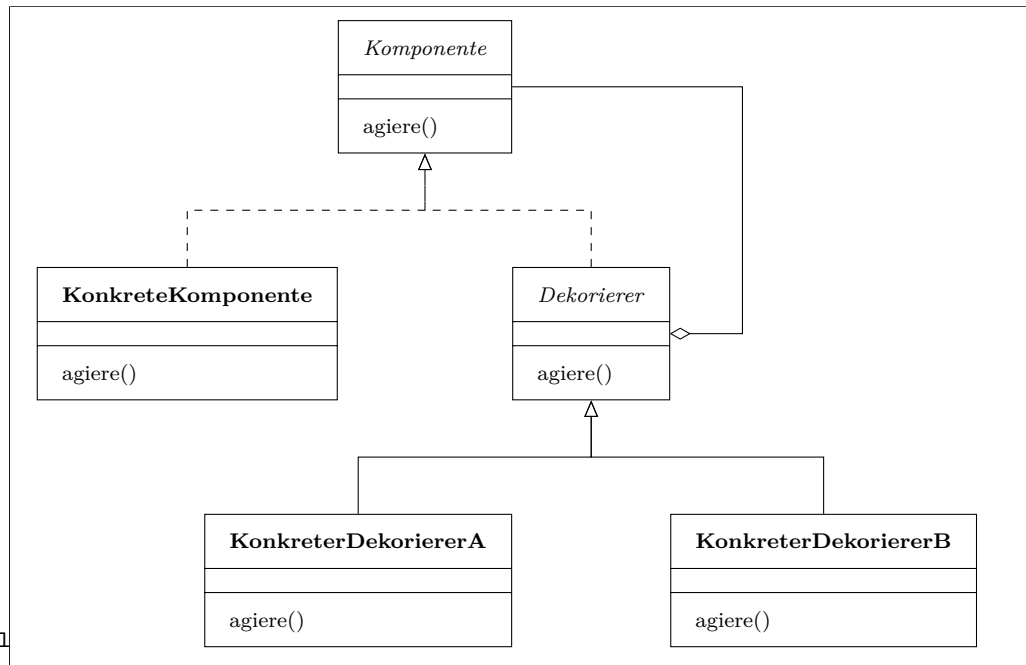
\liEntwurfsBeobachter

```

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

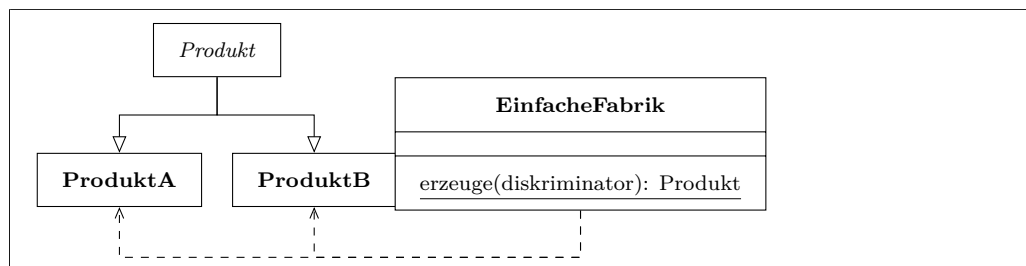
```

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)

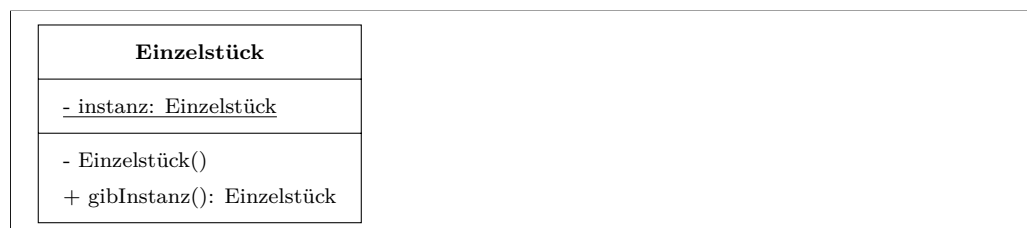
urfsEinzelstueckBeschreibung

```

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

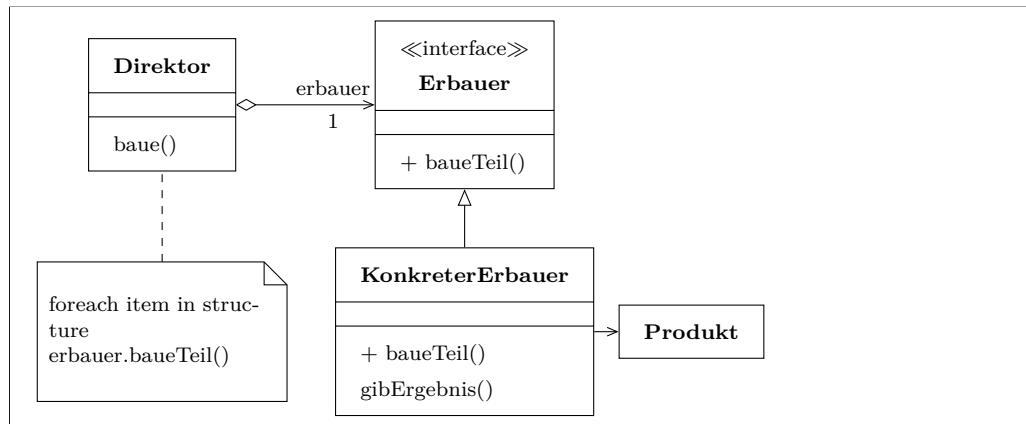
```

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

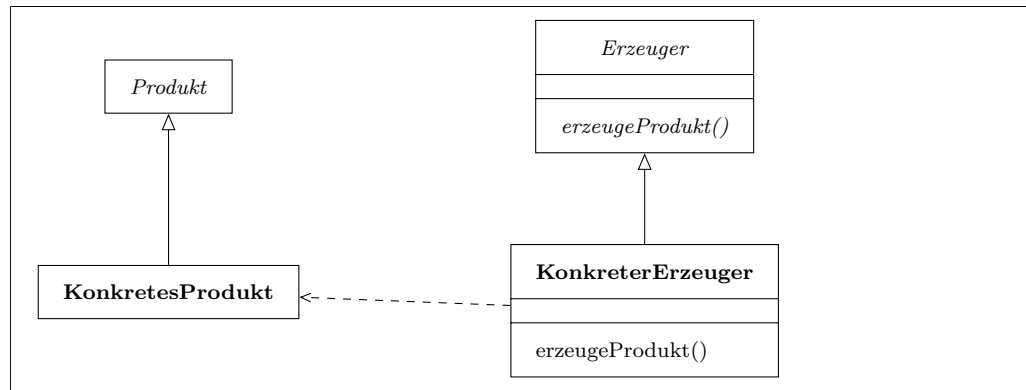
```

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 }

```

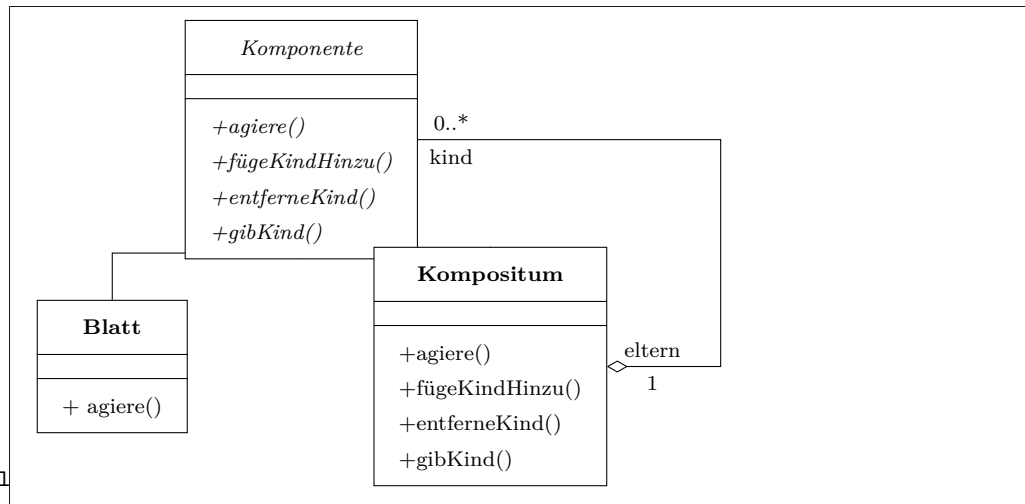
\liEntwurfsFabrikmethode

```

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

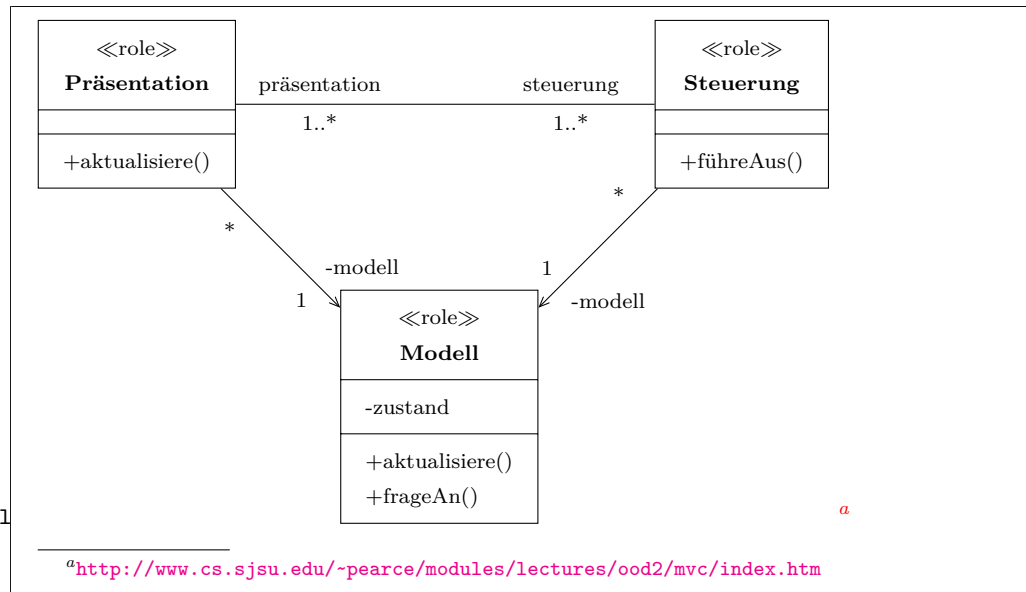
```

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

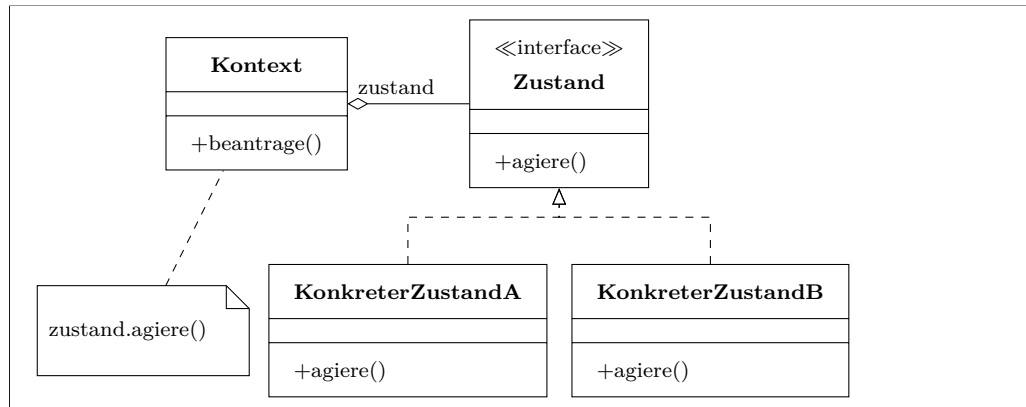
```

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.

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

```

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

```

```

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

\liEntwurfsZustandCode

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

\liEntwurfsZustand

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

1247

```

2.12 er.sty

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

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

2.12.1 Tex-Markup-Beispiel: Komplettes Diagramm

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

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

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

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

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

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

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

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

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

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

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

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

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

```

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

```

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

```

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

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

```

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

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

2.12.3 Makro-Kürzel

```

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

```

```
1255 \ExplSyntaxOn
```

```
\liErEntity
```

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

```
\liErRelationship
```

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

```
\liErAttribute
```

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

```
\liErMpEntity mp = marginpar
```

Let-Abkürzung: \let\e=\liErMpEntity

```

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

```

□

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

```

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

```

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

```

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

```

```

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

1282 \ExplSyntaxOff
1283

```

2.13 formale-sprachen.sty

```

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

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

1290 \RequirePackage{hyperref}

1291 \liLadePakete{mathe,typographie}

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

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

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

1300 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

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

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

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

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

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

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

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

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

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

\liZustandsBuchstabe

1316 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1317 \def\liZustandsBuchstabeGross{Z}

```

```

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

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

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

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

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

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

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

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

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

1350 \ExplSyntaxOn

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

```

```

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

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

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

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

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

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

```


2.14 formatierung.sty

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

2.14.1 Schriftarten / Typographie

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

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

2.14.2 Farben

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

2.14.3 Überschriften

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

2.14.4 Listen

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

2.14.5 Kasten

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

liKasten

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

2.14.6 Header

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

2.14.7 Zeilenabstände

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

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


2.15 gantt.sty

```

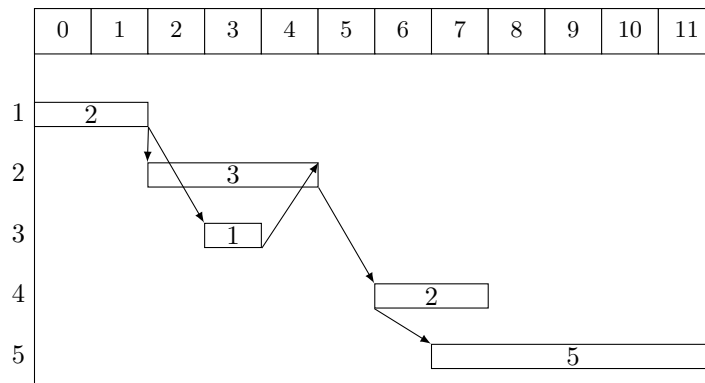
1435 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1436 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\begin{ganttchart}[x unit=0.75cm, y unit chart=0.8cm]{0}{11}
\gantttitlelist{0,...,11}{1} \\
\ganttbar[name=1]{1}{0}{1} \\
\ganttbar[name=2]{2}{2}{4} \\
\ganttbar[name=3]{3}{3}{3} \\
\ganttbar[name=4]{4}{6}{7} \\
\ganttbar[name=5]{5}{7}{11}

\node at (1) {2};
\node at (2) {3};
\node at (3) {1};
\node at (4) {2};
\node at (5) {5};

\ganttlink[link type=f-f]{3}{2}
\ganttlink[link type=f-s]{1}{2}
\ganttlink[link type=f-s]{1}{3}
\ganttlink[link type=f-s]{2}{4}
\ganttlink[link type=s-s]{4}{5}
\end{ganttchart}

```



```

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

1443

```

2.16 grafik.sty

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

\liGrafikLogoPfad

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

\liGrafikCCLizenz

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

\liGrafikLogo

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

1463 \ExplSyntaxOff
1464
```

2.17 graph.sty

```

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

```

Für die die Adjazenzliste (`xrightrightarrow`)

```

1468 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

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

```

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

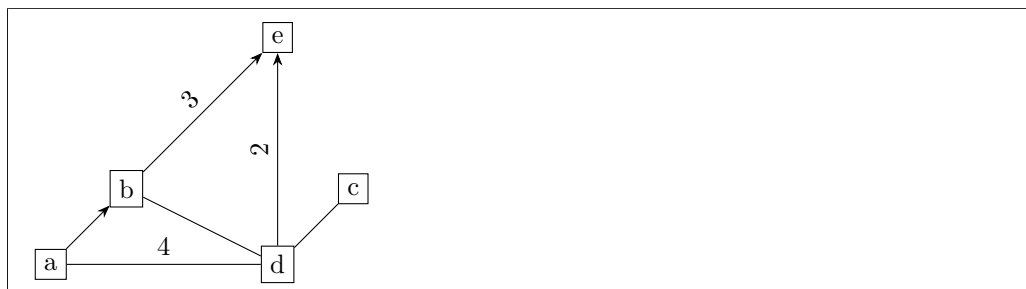
```

```

\begin{tikzpicture}[li graph]
\node (a) at (0,0) {a};
\node (b) at (1,1) {b};
\node (c) at (4,1) {c};
\node (d) at (3,0) {d};
\node (e) at (3,3) {e};

\path[->] (a) edge (b);
\path (b) edge (d);
\path[->] (b) edge node {3} (e);
\path (c) edge (d);
\path (d) edge node {4} (a);
\path[->] (d) edge node {2} (e);
\end{tikzpicture}

```



```

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

```

liGraphenFormat Dummy-Umgebung, zum Parsen durch Java gedacht.

```

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

```

```

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

```

```

1493

```

2.18 hanoi.sty

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

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

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

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

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

1531
```

2.19 klassen-konfiguration-aufgabe.sty

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

Formatierung muss vor literatur sein, sonst Option clash

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

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

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

In Aufgaben wollen wir das Additum sehen.

```
1547 \LiADDITUMtrue
1548 \ExplSyntaxOn

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

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

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

`\liAufgabenMetadaten`

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



```

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

1608 \AddToHook{enddocument}{
1609   \vfill
1610   {
1611
1612     \begin{minipage}{5.5cm}
1613       \liGrafikLogo[width=5cm]
1614     \end{minipage}
1615     \begin{minipage}{10cm}
1616       {
1617         \bfseries
1618         \liMetaBschlangaulSammlung
1619       }\par
1620
1621       \liMetaHermineBschlangaulAndFriends\par
1622       \medskip\par
1623
1624       \begin{spacing}{1}
1625         \footnotesize
1626         \liMetaUeberDasProjekt
1627       \end{spacing}
1628     \end{minipage}
1629
1630   }\par
1631
1632   \bigskip
1633
1634   \begin{minipage}{5.5cm}
1635     \centerline{\liGrafikCCLLizenz[width=3cm]}
1636   \end{minipage}
1637   \begin{minipage}{10cm}
1638     \begin{spacing}{1}
1639       \scriptsize
1640       \liMetaCCLink
1641     \end{spacing}
1642   \end{minipage}
1643
1644   \bigskip
1645
1646   \begin{spacing}{1}
1647     \tiny
1648     \noindent

```

```
1649      \liMetaHilfMit
1650
1651      \liMetaQuelltext
1652      \_gib_github_url_href:
1653      \end{spacing}
1654  }
1655 }
1656 \ExplSyntaxOff
1657
```

2.20 klassen-konfiguration-examen.sty

```
1658 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1659 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-examen}[2021/09/04 Die
1660 Klasse liSetzeExamen konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

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

```
1661 \liLadePakete{
1662   formatierung,
1663   literatur-dummy,
1664   makros,
1665   aufgaben-metadaten,
1666   abmessung,
1667   typographie,
1668   grafik,
1669   meta
1670 }
```

Formatierung für die Überschriften setzen.

```
1671 \RequirePackage{titlesec}
1672 \titleformat{\section}{\huge\filcenter\bfseries}{\thesection}{1em}{}
1673 \newcommand{\sectionbreak}{\clearpage}
1674 \setcounter{secnumdepth}{0}
1675 \liLadeAllePakete
```

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

```
1676 \RequirePackage[ngerman]{babel}
1677 \RequirePackage{standalone}
1678 \ExplSyntaxOn
```

`\liSetzeExamenThemaNr`

```
1679 \def\liSetzeExamenThemaNr#1{
1680   \tl_gset:Nn \g_auf_examen_thema_nr_tl { #1 }
1681   \section{Thema-Nr.~#1}
1682 }
```

`\liSetzeExamenTeilaufgabeNr`

```
1683 \def\liSetzeExamenTeilaufgabeNr#1{
1684   \tl_gset:Nn \g_auf_examen_teilaufgabe_nr_tl { #1 }
1685   \subsection{Teilaufgabe-Nr.~#1}
1686 }
```

`\liBindeAufgabeEin`

```
1687 \def\liBindeAufgabeEin#1{
1688   \tl_gset:Nn \g_auf_examen_aufgabe_nr_tl { #1 }
1689   \input{
1690     \LehramtInformatikRepository /
1691     Staatsexamen /
1692     \g_auf_examen_nummer_tl /
1693     \g_auf_examen_jahr_tl /
1694     \g_auf_examen_monat_tl /
1695     \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
1696       Thema - \g_auf_examen_thema_nr_tl /
1697     }
1698     \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
1699       Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl /
1700     }
1701     Aufgabe - \g_auf_examen_aufgabe_nr_tl .tex
1702   }
1703 }
```

`\liAufgabenMetadaten` Das Metadaten-Makro überschreiben

```
1704 \def\liAufgabenMetadaten#1{
1705   \liMetaSetze{#1}
1706   \subsubsection{\_gib_aufgaben_titel:}
1707 }
```

```

1708 \setcounter{tocdepth}{4}
1709 \RequirePackage[titles]{tocloft}
1710 \AddToHook{begindocument}{
1711   \pagestyle{empty}
1712   \begin{center}
1713     \large
1714     Erste-Staatsprüfung-für-ein-Lehramt-an-öffentlichen-Schulen \par
1715
1716     \vspace{0.5cm}
1717
1718     Fach-Informatik \par
1719
1720     \vfill
1721
1722     \liGrafikLogo[width=8cm]\par
1723     \bigskip
1724     Die-Bschlangaul-Sammlung \par
1725     {\footnotesize \liMetaHermineBschlangaulAndFriends} \par
1726
1727     \vfill
1728
1729     {
1730       \bfseries\Huge
1731
1732       \g_auf_examen_jahreszeit_tl \par
1733
1734       \g_auf_examen_jahr_tl \par
1735     }
1736
1737     \vspace{2cm}
1738
1739     {\LARGE \g_auf_examen_nummer_tl \par}
1740
1741     \vspace{0.5cm}
1742
1743     \g_auf_examen_fach_tl \par
1744
1745     \vspace{3cm}
1746
1747     Aufgabenstellungen-mit-Lösungsvorschlägen \par
1748
1749     \end{center}
1750
1751     % für den Abstand vor den section im Inhaltsverzeichnis
1752     % https://tex.stackexchange.com/questions/241445/how-to-control-spacing-in-toc-
1753     % for-different-sections
1754     \setlength{\cftbeforesecskip}{1.5cm}
1755     \setlength{\cftbeforesubsecskip}{0.5cm}
1756     \renewcommand{\cftsubsecafterpnum}{\vspace{\cftbeforesubsecskip}}
1757     \setlength{\cftbeforesubsubsecskip}{0.1cm}
1758
1759     \vfill
1760     \renewcommand{\contentsname}{Aufgabenübersicht}
1761     \tableofcontents
1762   }
1763 \ExplSyntaxOff
1764

```

2.21 komplexitaetstheorie.sty

```
1765 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1766 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1767 Setzen von Karp's NP-vollständigen Problemen, Hilfsmakros für die
1768 Polynomialzeitreduktion.]
```

2.21.1 Makro-Kürzel

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

```
1769 \liLadePakete{mathe}
```

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

```
1770 \RequirePackage{mdframed}
```

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

```
1771 \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`

```
1772 \def\liProblemName#1{\texttt{\textsc{#1}}}
```

`\liProblemBeschreibung` Zu setzen von Problem-Beschreibungen:

```
\liProblemBeschreibung
```

```
{ }
```

```
{ }
```

```
{ }
```

CLIQUE

Gegeben: Ein ungerichteter Graph $G = (V, E)$, eine Zahl $k \in \mathcal{N}$

Frage: Gibt es eine Menge $S \subseteq V$ mit $|S| = k$, sodass für alle Knoten $u \neq v \in V$ gilt, dass $\{u, v\}$ eine Kante in E ist?

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

```
1773 \def\liProblemBeschreibung#1#2#3{
1774   \begin{mdframed}[
1775     userdefinedwidth=9cm,
1776     align=center,
1777     backgroundcolor=white!0,
1778   ]
1779   \centerline{\large\liProblemName{#1}}
1780
1781   \medskip
1782
1783   \begin{description}
1784     \item[Gegeben:] #2
1785     \item[Frage:] #3
1786   \end{description}
1787 \end{mdframed}
1788 }
```

```

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

1789 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1790 \begin{displaymath}
1791   \liProblemName{#1}
1792   \preceq_{#2}
1793   \liProblemName{#3}
1794 \end{displaymath}
1795 }

\liProblemVertexCover

1796 \def\liProblemClique{%
1797 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1798 Mindestgröße  $n$  in einem gegebenen Graphen.
1799 \footcite[Seite 76]{theo:fs:4}
1800 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1801 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1802 \footcite{wiki:cliquenproblem}
1803 }

\liProblemVertexCover

1804 \def\liProblemVertexCover{%
1805 %
1806 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1807 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1808 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1809 \footcite{wiki:knotenueberdeckung}
1810
1811 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1812 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1813 mindestens einem Knoten aus  $U$  verbunden ist.
1814 \footcite[Seite 78]{theo:fs:4}%
1815 }

\liProblemSubsetSum Kein Karp-21-Problem

1816 \def\liProblemSubsetSum{%
1817 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1818 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1819 \footcite{wiki:teilsommenproblem}
1820 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1821 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1822 größer als eine gegebene obere Schranke  $c$  ist.
1823 \footcite[Seite 74]{theo:fs:4}
1824 }

\liProblemSubsetSum Kein Karp-21-Problem

1825 \def\liProblemSat{%
1826 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1827 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1828 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1829 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1830 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1831 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1832 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1833 aufgestellt werden.
1834 \footcite[Seite 71]{theo:fs:4}
1835 }

1836

```

2.22 kontrollflussgraph.sty

1837 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1838 \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 = ...}]
```

```

1839 \RequirePackage{tikz}
1840 \usetikzlibrary{positioning}
1841 \tikzset{
1842   li kontrollfluss/.style={
1843     knoten/.style={
1844       circle,
1845       draw
1846     },
1847     usebox/.style={
1848       draw,
1849       rectangle,
1850       font=\scriptsize,
1851       anchor=west,
1852       align=left,
1853     },
1854     bedingung/.style={
1855       midway,
1856       draw=none,
1857       font=\scriptsize
1858     },
1859     knotenbeschriftung/.style={
1860       draw,
1861       rectangle,
1862       midway,
1863       font=\scriptsize
1864     },
1865     wahr/.style={
1866       thick
1867     },
1868     falsch/.style={
1869       dashed
1870     },
1871     every node/.style={
1872       circle,
1873       draw,
1874     },
1875     every edge/.append style={
1876       every node/.style={
1877         draw=none,
1878         bedingung,
1879       }
1880     },
1881     every path/.style={
1882       draw,
1883       ->,
1884     },
1885     every pin/.style={
1886       draw,
1887       dotted,
1888       rectangle,
1889       pin position=right
1890     },
1891     every pin edge/.style={
1892       dotted,
1893       arrows=-,
1894     }
1895   }
1896 }

```

2.22.4 Umgebungen

liKontrollflussgraph

```

1897 \NewDocumentEnvironment { liKontrollflussgraph } { 0{} } {

```



```

1898 \begin{tikzpicture}[
1899     li kontrollfluss,
1900     #1
1901 ]
1902 } {
1903 \end{tikzpicture}
1904 }

```

2.22.5 Makros

\liAnweisung

```
1905 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}
```

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

```
1906 \def\liBedingung#1#2{\node[bedingung,#1]{\texttt{#2}}}
```

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

```
1907 \def\liBedingungWahr#1{\node[bedingung,#1]{\texttt{[true]}}}
```

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

```
1908 \def\liBedingungFalsch#1{\node[bedingung,#1]{\texttt{[false]}}}
```

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

```
1909 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}
```

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

```
1910 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw
```

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

```
1911 \ExplSyntaxOn
```

```
1912 \NewDocumentCommand { \liKontrollKnotenPfad } { m }
```

```
1913 {
```

```
1914   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
```

```
1915   \seq_set_map:NNn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
```

```
1916   \seq_use:Nn \l_tmpa_seq { ~~~~ }
```

```
1917 }
```

```
1918 \ExplSyntaxOff
```

```
1919
```

2.23 kopf-fusszeilen.sty

```
1920 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1921 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1922 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1923 \ExplSyntaxOn

1924 \fancyhead{}
1925 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1926 \fancyhead[L0,RE]{\scriptsize\today}
1927 \fancyfoot{}
1928 \fancyfoot[LE,R0]{\thepage}
1929 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1930 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1931 \renewcommand{\headrulewidth}{0.4pt}
1932 \renewcommand{\footrulewidth}{0.4pt}
1933 \setlength{\headwidth}{\textwidth}

1934 \ExplSyntaxOff

1935
```

2.24 literatur-dummy.sty

```
1936 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1937 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1938 \def\literatur{}

\footcite

1939 % \RequirePackage[stable,multiple]{footmisc}
1940 \NewDocumentCommand{ \footcite } { o m }{}

1941
```

2.25 literatur.sty

```
1942 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1943 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1944 \RequirePackage{csquotes}
1945 \RequirePackage[
1946   bibencoding=utf8,
1947   citestyle=authortitle,
1948   backend=biber,
1949 ]{biblatex}
1950 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1951 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1952 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1953 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1954 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1955 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1956 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1957 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1958 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1959 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1960 % To allow footnotes in the heading
1961 \RequirePackage[stable,multiple]{footmisc}

\literatur

1962 \def\literatur{\printbibliography[heading=subbibliography]}

1963
```

2.26 makros.sty

```
1964 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1965 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1966 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1967 anderen Paket passen]
1968 \RequirePackage{hyperref}
1969 \RequirePackage{graphicx}
    Für die Umgebung liQuellen benötigt.
1970 \RequirePackage{paralist}
1971 \ExplSyntaxOn

\inhaltsverzeichnis
1972 \def\inhaltsverzeichnis {
1973   \begin{mdframed}
1974     \begin{group}
1975       \let\clearpage\relax
1976       \tableofcontents
1977     \end{group}
1978   \end{mdframed}
1979 }

\memph \mephm (\marginpar and \emph)
1980 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1981 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1982 \newcommand{\liPseudoUeberschrift}[1]{
1983   \bigskip
1984
1985   \par
1986   \noindent
1987   \textbf{#1}
1988
1989   \medskip
1990
1991   \par
1992   % Keine Einrückung
1993   \@afterindentfalse
1994   \@afterheading
1995 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1996 \newcommand{\liBeschriftung}[1]{
1997   \par
1998   \noindent
1999   \medskip
2000   \textbf{#1}:
2001   \medskip
2002   \noindent
2003 }

\hinweis
2004 \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.
2005 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}
```

liEinbettung

```
2006 \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.

```
2007 \NewDocumentEnvironment{ liAntwort } { O{standard} }
2008 {
2009   \ifLiANTWORT
2010     \str_case:nn {#1} {
2011       {standard} {
2012         \def\beschriftung{}
2013         \mdfsetup{backgroundcolor=gray!1,linecolor=gray}
2014       }
2015       {richtig} {
2016         \def\beschriftung{richtig}
2017         \mdfsetup{backgroundcolor=green!5,linecolor=green}
2018       }
2019       {falsch} {
2020         \def\beschriftung{falsch}
2021         \mdfsetup{backgroundcolor=red!3,linecolor=red}
2022       }
2023       {muster} {
2024         \def\beschriftung{Musterlösung}
2025         \mdfsetup{backgroundcolor=green!12,linecolor=green}
2026       }
2027     }
2028     \ifx\beschriftung\empty\else
2029       \noindent
2030       \textbf{\beschriftung{}}:}
2031     \fi
2032     \begin{mdframed}
2033   \else
2034     \fi
2035 }
2036 {
2037   \ifLiANTWORT
2038     \end{mdframed}
2039   \else
2040     \fi
2041 }
```

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

```
2042 \NewDocumentEnvironment{ liAdditum } { o +b}
2043 {
2044   \ifLiADDITUM
2045     \begin{mdframed}[backgroundcolor=yellow!5]
2046     \IfNoValueTF {#1}
2047     {
2048       \liPseudoUeberschrift{Additum}
2049     }
2050     {
2051       \liPseudoUeberschrift{Additum:~#1}
2052     }
2053     \end{mdframed}
2054   \else
2055     \fi
2056 } { }
```

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.

```
2057 \NewDocumentEnvironment{ liExkurs }{o +b}{
2058   \ifLiEXKURS
2059     \vspace{0.2cm}%
2060     \begin{mdframed}[
2061       backgroundcolor=white,
2062       bottomline=false,
2063       innermargin=1cm,
2064       leftline=true,
2065       linecolor=black,
2066       linewidth=0.1cm,
2067       outermargin=1cm,
2068       rightline=false,
2069       topline=false,
2070     ]
2071     \footnotesize
2072     \noindent%
2073     \textbf{Exkurs:~\#1}\par%
2074     \noindent%
2075     \#2
2076     \end{mdframed}
2077     \vspace{0.2cm}
2078   \else
2079     \fi
2080 }{}
```

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

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

Weiterführende Literatur:

- Quelle 1
- Quelle 2

```
2081 \cs_new:Npn \listen@punkt #1 {\item #1}
2082 \NewDocumentEnvironment { liQuellen }{ +b }
2083 {
2084   \seq_clear_new:N \l_quellen
2085   \seq_set_split:Nnn \l_quellen {\item} {\#1}
2086   \seq_remove_all:Nn \l_quellen {}
2087   \begin{mdframed}[backgroundcolor=white!0]
2088     \footnotesize
2089     \noindent
2090     \textsf{\textbf{Weiterführende~Literatur:}}
2091     \medskip
2092     \begin{compactitem}
2093       \seq_map_function:NN \l_quellen {\listen@punkt}
2094     \end{compactitem}
2095   \end{mdframed}
2096   %
2097   \par
2098   \@afterindentfalse
2099   \@afterheading
```

```

2100 } {}

liLernkartei
2101 \NewDocumentEnvironment { liLernkartei }{ m +b }
2102 {
2103   \begin{mdframed}
2104     \footnotesize
2105     \noindent%
2106     \textbf{Lernkarteikarte:~\#1}\par%
2107     \noindent%
2108     #2
2109   \end{mdframed}
2110 } {}

liDiagramm \begin{liDiagramm}{beschriftung}\end{liDiagramm}: Zu setzen einer Graphik bzw
eines Diagramms.
2111 \NewDocumentEnvironment { liDiagramm }{ m +b }
2112 {
2113   \begin{mdframed}[backgroundcolor=white!0]
2114     \small
2115     \noindent%
2116     \textit{\#1}:
2117     \begin{center}
2118       #2
2119     \medskip
2120     \end{center}
2121   \end{mdframed}
2122 } {}

\liFussnoteUrl \liFussnoteUrl[\langlezusätzlicher-text\rangle]{\langleurl\rangle} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
2123 \NewDocumentCommand{\liFussnoteUrl} { o m } {
2124   \footnote{\url{\#2}\IfNoValueTF{\#1}{\{ \} }{ ( \#1 ) }}
2125 }
2126

\liFussnoteLink \liFussnoteLink[\langlezusätzlicher-text\rangle]{\langlelink-text\rangle}{\langleurl\rangle} \liFussnoteLink[zusätzlicher Text]{tex
Einen Link, d. h. einen Link-Text und eine URL als Fußnote setzen.
2127 \NewDocumentCommand{\liFussnoteLink} { o m m } {
2128   \footnote{\href{\#3}{\#2}\IfNoValueTF{\#1}{\{ \} }{ ( \#1 ) }}
2129 }

\zB
2130 \def\zB{z.\,B. }

\ZB
2131 \def\ZB{Z.\,B. }

\dh
2132 \def\dh{d.\,h. }

2133 \ExplSyntaxOff
2134

```


2.27 master-theorem.sty

2135 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2136 \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}

2137 \ExplSyntaxOn

2138 \RequirePackage{amsmath}

\liRundeKlammer

2139 \def\liRundeKlammer#1{

2140 \negthinspace \left(#1 \right)

2141 }

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

2142 \def\liThetaOhneMathe#1{

2143 \Theta \liRundeKlammer{#1}

2144 }

2145 \def\liTheta#1{

2146 \ifmmode

2147 \liThetaOhneMathe{#1}

2148 \else

2149 \$\liThetaOhneMathe{#1}\$

2150 \fi

2151 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
2152 \def\liOmegaOhneMathe#1{
2153   \Omega \liRundeKlammer{#1}
2154 }
2155 \def\liOmega#1{
2156   \ifmmode
2157     \liOmegaOhneMathe{#1}
2158   \else
2159     $\liOmegaOhneMathe{#1}$
2160   \fi
2161 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
2162 \def\liOOhneMathe#1{
2163   \mathcal{O} \liRundeKlammer{#1}
2164 }
2165 \def\liO#1{
2166   \ifmmode
2167     \liOOhneMathe{#1}
2168   \else
2169     $\liOOhneMathe{#1}$
2170   \fi
2171 }

\liT Let-Abkürzung: \let\T=\liT
      \liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
2172 \def\liTOhneMathe#1#2{
2173   \tl_if_blank:nTF {#1}
2174   {}
2175   {#1 \cdot }
2176   T
2177   \liRundeKlammer{\frac{n}{#2}}
2178 }
2179 \def\liT#1#2{
2180   \ifmmode
2181     \liTOhneMathe{#1}{#2}
2182   \else
2183     $\liTOhneMathe{#1}{#2}$
2184   \fi
2185 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
2186 \def\liRekursionsGleichung{
2187   $T(n) = \liT{a}{b} + f(n)$
2188 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
2189 \def\liBedingungEins{
2190   $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
2191 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
2192 \def\liBedingungZwei{
2193   $f(n) \in \liTheta{n^{\log\sb{b}a}}$
2194 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
2195 \def\liBedingungDrei{
2196   $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
2197 }

2198 \ExplSyntaxOff

```

\liMasterVariablen

```

2199 \def\liMasterVariablen{
2200   \begin{displaymath}
2201     T(n) = \liT{a}{b} + f(n)
2202   \end{displaymath}
2203
2204   \begin{itemize}
2205     \item[$a = $]
2206       Anzahl der rekursiven Aufrufe, Anzahl der Unterprobleme in der
2207       Rekursion
2208       ($a \geq 1$).
2209
2210     \item[$\textstyle{\frac{1}{b}} = $]
2211       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
2212       repräsentiert wird, Anteil an der Verkleinerung des Problems ($b > 1$).
2213
2214     \item[$f(n) = $]
2215       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
2216       die Kombination der Teillösungen entstehen. Eine von $T(n)$
2217       unabhängige und nicht negative Funktion.
2218   \end{itemize}
2219   \footcite{wiki:master-theorem}
2220   \footcite[Seite 19-35]{aud:fs:2}
2221 }
```

\liMasterFaelle

```

2222 \def\liMasterFaelle{
2223   \begin{description}
2224     \item[1. Fall:]
2225       $T(n) \in \liTheta{n^{\log\sb{b}a}}$
2226
2227       \hfill falls \liBedingungEins
2228       für $\varepsilon > 0$
2229
2230     \item[2. Fall:]
2231       $T(n) \in \liTheta{n^{\log\sb{b}a} \cdot \log n}$
2232
2233       \hfill falls \liBedingungZwei
2234
2235     \item[3. Fall:]
2236       $T(n) \in \liTheta{f(n)}$
2237
2238       \hfill falls \liBedingungDrei
2239       für $\varepsilon > 0$
2240       und ebenfalls für ein $c$ mit $0 < c < 1$ und alle hinreichend großen $n$
2241       gilt:
2242       $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$
2243   \end{description}
2244 }
```

\liMasterVariablenDeklaration

```

2245 \def\liMasterVariablenDeklaration#1#2#3{
2246   \begin{description}
2247     \item[Allgemeine Rekursionsgleichung:] \strut
2248
2249     \liRekursionsGleichung
2250
2251     \item[Anzahl der rekursiven Aufrufe ($a$):] \strut
2252
2253     #1
2254
2255     \item[Anteil Verkleinerung des Problems ($b$):] \strut
2256
2257     #2
2258
2259     #3
2260   \end{description}
2261 }
```

```

2257     um  $\frac{1}{b}$  also  $b = \frac{1}{a}$ 
2258
2259     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut
2260
2261      $\frac{1}{b}$ 
2262
2263     \item[Ergibt folgende Rekursionsgleichung:] \strut
2264
2265      $T(n) = T(\frac{n}{b}) + \frac{1}{b}$ 
2266 \end{description}
2267 }

\liMasterFallRechnung
2268 \def\liMasterFallRechnung#1#2#3{
2269   \begin{description}
2270     \item[1. Fall:] \liBedingungEins:
2271
2272     #1
2273
2274     \item[2. Fall:] \liBedingungZwei:
2275
2276     #2
2277
2278     \item[3. Fall:] \liBedingungDrei:
2279
2280     #3
2281   \end{description}
2282 }

\liMasterExkurs
2283 \def\liMasterExkurs{
2284   \begin{liExkurs}[Master-Theorem]
2285     \liMasterVariablen
2286
2287     \noindent
2288     Dann gilt:
2289
2290     \liMasterFaelle
2291   \end{liExkurs}
2292 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
2293 \def\liMasterWolframLink#1{
2294   Berechne die Rekursionsgleichung auf WolframAlpha:
2295   \href{https://www.wolframalpha.com/input/?i=#1}{WolframAlpha}
2296 }

2297

```

2.28 mathe.sty

```
2298 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2299 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
2300
2301 % for example \ltimes \rtimes
2302 %\RequirePackage{amssymb}
2303 \RequirePackage{amsmath}
2304
2305 %%
2306 % \mlq \mrq
2307 %%
2308 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
2309 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
2310
```

2.29 meta.sty

```
2311 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2312 \ProvidesPackage{lehramt-informatik-klassen-meta}[2021/09/10 Sammlung
2313 von Textschnipseln, die das Projekt beschreiben]
```

```
2314 \ExplSyntaxOn
```

```
\liMetaBschlangaulSammlung
```

```
2315 \def\liMetaBschlangaulSammlung{
2316   Die~Bschlangaul-Sammlung
2317 }
```

```
HermineBschlangaulAndFriends
```

```
2318 \def\liMetaHermineBschlangaulAndFriends{
2319   Hermine~Bschlangaul-and-Friends
2320 }
```

```
\liMetaUeberDasProjekt
```

```
2321 \def\liMetaUeberDasProjekt{
2322   Eine~freie~Aufgabensammlung~mit~Lösungen~
2323   von~Studierenden~für~Studierende~
2324   zur~Vorbereitung~auf~die~1.~Staatsexamensprüfungen~
2325   des~Lehramts~Informatik~in~Bayern.
2326 }
```

```
\liMetaCCLink
```

```
2327 \def\liMetaCCLink{
2328   Diese~Materialsammlung~unterliegt~den~Bestimmungen~der~
2329   \href{
2330     https://creativecommons.org/licenses/by-nc-sa/4.0/deed.de
2331   }
2332   {
2333     Creative~Commons~Namensnennung~Nicht~kommerziell~Share~Alike~4.0~
2334     International~Lizenz
2335   }.
2336 }
```

```
\liMetaEmailLink
```

```
2337 \def\liMetaEmailLink{
2338   \href{
2339     mailto:hermine.bschlangaul@gmx.net
2340   }{
2341     hermine.bschlangaul@gmx.net
2342   }
2343 }
```

```
\liMetaHilfMit
```

```
2344 \def\liMetaHilfMit{
2345   Hilf~mit!~
2346
2347   Die~Hermine~schafft~das~nicht~alleine!~
2348
2349   Das~ist~ein~Community-Projekt.~
2350
2351   Verbesserungsvorschläge,~Fehlerkorrekturen,~weitere~Lösungen~sind~
2352   herzlich~willkommen~~~egal~wie~~~per~Pull-Request~oder~per~E-Mail~an~
2353   \liMetaEmailLink.
2354 }
```

```
\liMetaHilfMit
```

```
2355 \def\liMetaQuelltext{
2356   Der~\TeX-Quelltext~dieses~Dokuments~kann~unter~folgender~
2357   URL~aufgerufen~werden:~
2358 }
```

2359 \ExplSyntaxOff

2360

2.30 minimierung.sty

```

2361 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2362 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
2363 Minimierungsalgorithmus von einem NEA zu einem DEA]

2364 \liLadePakete{typographie}

\let\z=\liZustandsnameTiefgestellt
\let\f=\liFussnote
\let\l=\liLeereZelle
\let\Z=\liZustandsPaar
\let\erklaerung=\liMinimierungErklaerung

\begin{tabular}{|c||c|c|c|c|c|c|c|c|c|}
\hline
\z0 & \l & \l & \l & \l & \l & \l & \l & \l & \l & \l \\
\z1 & & \l & \l & \l & \l & \l & \l & \l & \l & \l \\
\z2 & & & \l & \l & \l & \l & \l & \l & \l & \l \\
\z3 & & & & \l & \l & \l & \l & \l & \l & \l \\
\z4 & & & & & \l & \l & \l & \l & \l & \l \\
\z5 & & & & & & \l & \l & \l & \l & \l \\
\z6 & & & & & & & \l & \l & \l & \l \\
\z7 & & & & & & & & \l & \l & \l \\
\z8 & & & & & & & & & \l & \l \\
& \z0 & \z1 & \z2 & \z3 & \z4 & \z5 & \z6 & \z7 & \z8 & \l \\
\end{tabular}

\liFussnoten

\begin{liUebergangsTabelle}{0}{1}
\Z01 & \Z10 & \Z23 & \\
\Z05 & \Z15 & \Z25 & \f2 \\
\Z15 & \Z05 & \Z35 & \f2 \\
\Z23 & \Z44 & \Z55 & \\
\Z24 & \Z44 & \Z55 & \\
\Z34 & \Z44 & \Z55 & \\
\end{liUebergangsTabelle}

\liFussnote

2365 \def\liFussnote#1{\$x_{#1}\$}

2366 \def\li@fussnote@text#1#2{
2367 \liFussnote{#1}
2368 \quad
2369 {\footnotesize #2}
2370 }

\liFussnoteEinsText
2371 \def\liFussnoteEinsText{
2372 \li@fussnote@text{1}
2373 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
2374 }

\liFussnoteZweiText
2375 \def\liFussnoteZweiText{
2376 \li@fussnote@text{2}
2377 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
2378 }

\liFussnoteDreiText
2379 \def\liFussnoteDreiText{
2380 \li@fussnote@text{3}

```



```

2381 {In weiteren Iterationen markierte Zustände.}
2382 }

\liFussnoteVierText
2383 \def\liFussnoteVierText{
2384   \li@fussnote@text{4}
2385   {...}
2386 }

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



2387 \def\liFussnoten{
2388   \bigskip
2389
2390   \noindent
2391   \liFussnoteEinsText
2392
2393   \noindent
2394   \liFussnoteZweiText
2395
2396   \noindent
2397   \liFussnoteDreiText
2398
2399   \noindent
2400   \liFussnoteVierText
2401 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
2402 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
2403 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
2404 \def\liZustandsPaar#1#2{
2405   $(
2406     \liZustandsPaarVariablenName_#1,
2407     \liZustandsPaarVariablenName_#2
2408   )$
2409 }

liUebergangsTabelle
2410 \renewcommand{\arraystretch}{1.4}
2411 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
2412   \liPseudoUeberschrift{Übergangstabelle}
2413   \begin{center}
2414     \begin{tabular}{r|l|l}
2415       \textbf{Zustandspaar} & \textbf{\#1} & \textbf{\#2} \\ \hline
2416     \end{tabular}
2417   \end{center}
2418 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle:

Minimierungstabelle (Table filling)
2420 \ExplSyntaxOn

```

```

2421 \def\liUeberschriftDreiecksTabelle{
2422   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
2423 }

```

\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. —

```

2424 \def\liMinimierungErklaerung{
2425   %\footcite[Seite~19]{koenig}
2426   \liParagraphMitLinien{
2427     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
2428     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
2429     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
2430     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
2431      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
2432     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
2433     somit~zueinander~( $k - 1$ )-äquivalent,~aber~nicht~ $k$ -äquivalent~und~
2434     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
2435     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
2436   }
2437 }
2438 \ExplSyntaxOff

```

2439

2.31 normalformen.sty

```
2440 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2441 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10
2442 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
2443 Attributhülle]
```

Mathe für textit tex etc, typographie für geschweifte Klammer

```
2444 \liLadePakete{mathe,typographie}
```

```
2445 \directlua{
2446   helfer = require('lehramt-informatik-helfer')
2447   normalformen = require('lehramt-informatik-normalformen')
2448 }
```

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
```

```
2449 \def\liTeilen#1{
2450   \directlua{tex.print(normalformen.teilen('#1'))}
2451 }
```

```
\liAttributHuelle Let-Abkürzung: \let\ah=\liAttributHuelle
                  \ah{F, \m{A, B}} AttrHülle( $F, \{A, B\}$ ) Regulärer Ausdruck zum Konvertieren
                  AttrHülle((.*)\)\ \ah{$1}
2452 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(#1)}
2453 \def\liAttributHuelle#1{
2454   \ifmmode
2455     \liAttributHuelleOhneMathe{#1}
2456   \else
2457     $\liAttributHuelleOhneMathe{#1}$
2458   \fi
2459 }
```

```
\liAttributMenge Let-Abkürzung: \let\m=\liAttributMenge
2460 \def\liAttributMenge#1{\{ \textit{#1} \}}
```

liAHuelle

```
2461 \NewDocumentEnvironment{ liAHuelle } { +b } {
2462   \begingroup
2463   \footnotesize
2464   \begin{multline*}
2465     #1
2466   \end{multline*}
2467   \endgroup
2468 } { }
```

```
\liLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multline
Let-Abkürzung: \let\ahL=\liLinksReduktion
                  \ahL{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}
2469 \def\liLinksReduktion#1#2#3{
2470   \shoveleft{
2471     \liAttributHuelleOhneMathe{FA,
2472       \liAttributMenge{#1 \string\ #2}} =
2473   } \\\
```

```

2474 \shoveright{
2475   \liAttributMenge{#3}
2476 } \\\
2477 }

```

```

\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( $F, \{A, E \setminus E\}$ ) =  $\{A, E, F, B, D\}$ 
2478 \def\liLinksReduktionInline#1#2#3{%
2479   {%
2480     \footnotesize%
2481     $\liAttributHuelleOhneMathe{F,
2482       \liAttributMenge{#1} \string\ #2}} =
2483     \liAttributMenge{#3}$
2484   }
2485 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahr=\liLinksReduktionInline
    \ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
2486 \def\liRechtsReduktionInline#1#2#3#4{%
2487   {%
2488     \footnotesize%
2489     $\liAttributHuelleOhneMathe{
2490       F \setminus
2491       \liFunktionaleAbhaengigkeit{#1}
2492       \def\tmp{#2}\ifx\tmp\empty
2493       \else
2494         \cup \liFunktionaleAbhaengigkeit{#2}
2495       \fi
2496     ,
2497     \liAttributMenge{#3}
2498   } =
2499   \liAttributMenge{#4}$
2500 }
2501 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
    \fa{A, B -> C, D}:  $\{A, B\} \rightarrow \{C, D\}$  Regulärer Ausdruck zum Konvertieren:
    \$(.*?) \\\rightarrow (.*)\$ \\\fa{$1 -> $2}
2502 \def\liFunktionaleAbhaengigkeit#1{%
2503   \directlua{normalformen.drucke_funk_abhaengigkeit('#1', false)}%
2504 }

```

```

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

```

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

```

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

```

```

2505 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
2506   \liGeschweifteKlammern
2507   {#1}
2508   {
2509     \begin{align*}
2510       \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
2511     \end{align*}
2512   }
2513   {-0.5cm}
2514   {-1.7cm}
2515 }

\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}

2516 \NewDocumentCommand {\liRelation} { O{R} m } {
2517   $\directlua{
2518     local name = helper.konvertiere_tiefgestellt('#1')
2519     tex.print(name)
2520   }$(\textit{\, #2\,})
2521 }

2522

```

2.32 o-notation.sty

2523 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2524 \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}
```

2525 \ExplSyntaxOn

2526 \RequirePackage{amssymb}

2527 \RequirePackage{pgfplots}

Für echte Teilmenge \subsetneq: \subsetneq

2528 \RequirePackage{amssymb}

\liRundeKlammer

```
2529 \def\liRundeKlammer#1{
2530   \negthinspace \left( #1 \right)
2531 }
```

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

```
2532 \cs_new:Npn \o_notation_O:n #1 {
2533   \mathcal{O} \liRundeKlammer{#1}
2534 }
2535 \def\liONotationO#1{
2536   \ifmmode
2537     \o_notation_O:n { #1 }
2538   \else
2539     $ \o_notation_O:n { #1 } $
2540   \fi
2541 }
```

2542

2.33 petri.sty

2543 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2544 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]

2.33.1 Makro-Kürzel

\let\t=\liPetriTransitionsName

\let\tp=\liPetriTransPfeile

\let\k=\liPetriErreichKnotenDrei

2545 \RequirePackage{tikz}

2546 \usetikzlibrary{petri,arrows.meta}

Für die Darstellungsmatrix

2547 \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

```
2548 \def\liPetriSetzeSchluessel{%
2549   \def\TmpTransitionOne{}%
2550   \def\TmpTransitionTwo{}%
2551   \def\TmpTransitionThree{}%
2552   \def\TmpTransitionFour{}%
2553   \def\TmpTransitionFive{}%
2554   \def\TmpTransitionSix{}%
2555   \def\TmpTransitionSeven{}%
2556   \def\TmpTransitionEight{}%
2557   \def\TmpTransitionNine{}%
2558   \def\TmpTransitionTen{}%
2559   \pgfkeys{/petri/.cd,
2560     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2561     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2562     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2563     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
```

```

2564     p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
2565     p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2566     p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2567     p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2568     p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2569     p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2570     t1/.store in=\TmpTransitionOne,t1/.default=activated,
2571     t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2572     t3/.store in=\TmpTransitionThree,t3/.default=activated,
2573     t4/.store in=\TmpTransitionFour,t4/.default=activated,
2574     t5/.store in=\TmpTransitionFive,t5/.default=activated,
2575     t6/.store in=\TmpTransitionSix,t6/.default=activated,
2576     t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2577     t8/.store in=\TmpTransitionEight,t8/.default=activated,
2578     t9/.store in=\TmpTransitionNine,t9/.default=activated,
2579     t10/.store in=\TmpTransitionTen,t10/.default=activated,
2580     scale/.store in=\TmpScale,scale/.default=0.5,
2581     x/.store in=\TmpX,x/.default=5,
2582     y/.store in=\TmpY,y/.default=5,
2583   }%
2584 }

```

```

2585 \tikzset{
2586   li petri/.style={
2587     activated/.style={
2588       very thick
2589     },
2590     inhibitor/.style={
2591       {Circle[open,length=2mm,fill=white]}-
2592     }
2593   }
2594 }

```

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

```

2595 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2596 \def\liPetriTransitionsName#1{
2597   \ifmmode
2598     \liPetriTransitionsNameOhneMathe{#1}
2599   \else
2600     $\liPetriTransitionsNameOhneMathe{#1}$
2601   \fi
2602 }

```

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

```

2603 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2604   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2605 }

```

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

```

2606 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2607 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2608 }

```


2.34 potenzmengen-konstruktion.sty

```

2609 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2610 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2611 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]

2612 \liLadePakete{formale-sprachen}
2613 \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}
    }
  }
}

2614 \def\liZustandsMengenSammlung#1#2{
2615   \liZustandsnameGross{#1}
2616   {
2617     \footnotesize
2618     \liPotenzmenge{
2619       \str_case:nn {#1} #2
2620     }
2621   }
2622 }

\liZustandsMengenSammlungNr

2623 \def\liZustandsMengenSammlungNr#1#2{
2624   \liZustandsnameGross{#1}
2625   {

```

```
2626     \footnotesize
2627     \liZustandsmengeNr{
2628         \str_case:nn {#1} #2
2629     }
2630 }
2631 }

2632 \ExplSyntaxOff
2633
```

2.35 pseudo.sty

```
2634 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2635 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen
2636 von Pseudo-Code, Hüll-Paket um algorithm2e]
```

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

Algorithmus 1: Minimaler Spannbaum nach Kruskal

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

```
2637 \RequirePackage[german,boxruled]{algorithm2e}
```

```
2638
```

2.36 pumping-lemma.sty

2639 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2640 \ProvidesPackage{lehramt-informatik-pumping-lemma}[2021/08/11 Enthält die
 2641 Definitionen für das Pumping-Lemma in der Regulären Sprache und
 2642 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2643 \def\liPumpingRegulaer{%
2644   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2645   alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2646    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2647   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2648
2649   \begin{enumerate}
2650     \item  $|v| \geq 1$ 
2651     (Das Wort  $v$  ist nicht leer.)
2652
2653     \item  $|uv| \leq j$ 
2654     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2655
2656     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w$  in  $L$ 
2657     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2658     Sprache  $L$ )
2659   \end{enumerate}
2660
2661   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2662   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2663 }
```

\liPumpingKontextfrei

```
2664 \def\liPumpingKontextfrei{%
2665   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2666   sich alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  zerlegen lassen in
2667    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2668
2669   \begin{enumerate}
2670     \item  $|vx| \geq 1$ 
2671     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2672
2673     \item  $|vwx| \leq j$ 
2674     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2675
2676     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy$  in  $L$  (Für jede
2677     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2678     Sprache  $L$ )
2679   \end{enumerate}
2680 }
```

2681

2.37 quicksort.sty

```

2682 % https://tex.stackexchange.com/a/142634
2683 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2684 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2685
2686 %-----
2687 % USAGE:
2688 % \QSinitialize{comma, separated, numerical, values}
2689 % \loop
2690 % \QSpivotStep
2691 % \ifnum\value{pivotcount}>0
2692 %   \QSSortStep
2693 % \repeat
2694 %-----
2695
2696 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2697 \RequirePackage{xintfrac, xinttools}
2698
2699 \RequirePackage{tikz}
2700
2701 %-----
2702 % FIRST PART: TikZ styles and macros for the actual drawing
2703 \newcounter{cellcount}% used for coordinates of the node
2704 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2705
2706 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2707 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2708 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2709 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2710 % by police of LaTeX good conduct ? )
2711 \tikzset{
2712   l/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2713   o/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2714   r/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2715   b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2716   n/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2717   g/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2718 }
2719
2720 % NOTE the b style was originally the same as the r(aised) style apart from
2721 % not being raised, but I find it nicer with a somewhat different
2722 % specification. I have not updated the images though.
2723
2724 % How the nodes are drawn depending on whether on the left of the pivot value
2725 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2726
2727 \def\DecoLEFT #1{%
2728   \xintFor* ##1 in {#1} \do
2729   {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2730 }
2731
2732 \def\DecoINERT #1{%
2733   \xintFor* ##1 in {#1} \do
2734   {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2735 }
2736
2737 \def\DecoRIGHT #1{%
2738   \xintFor* ##1 in {#1} \do
2739   {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2740 }
2741
2742 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2743   \xintFor* ##1 in {#1} \do

```

```

2744     {\stepcounter{cellcount}}%
2745     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2746 }
2747
2748 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2749     \xintFor* ##1 in {#1} \do
2750     {\stepcounter{cellcount}}%
2751     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2752 }
2753
2754 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2755     \xintFor* ##1 in {#1} \do
2756     {\stepcounter{cellcount}}%
2757     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2758 }
2759
2760 %-----
2761 % SECOND PART: the actual sorting routines.
2762
2763 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2764 \def\QS@sort@b #1{\ifcase #1
2765     \expandafter\QS@sort@empty
2766     \or\expandafter\QS@sort@single
2767     \else\expandafter\QS@sort@c
2768     \fi
2769 }%
2770 \def\QS@sort@empty #1{}
2771 \def\QS@sort@single #1{\QS@Ir {#1}}
2772
2773 % This step is to pick the last as pivot.
2774 \def\QS@sort@c #1%
2775     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2776
2777 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2778 % The trick with \xintApplyUnbraced is that for example when selecting
2779 % the elements smaller than pivot, if we had been using \xintApply we
2780 % would have had at the minimum an empty brace pair. Thus we use the
2781 % "unbraced" variant, but then the \QS@select@smaller has added in
2782 % anticipation a level of braces.
2783 \def\QS@sort@d #1#2{%
2784     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2785     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2786     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2787 }%
2788 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2789 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2790 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2791
2792 %
2793 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2794 %
2795 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2796 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2797 % latter must handle correctly an empty argument.
2798
2799 %-----
2800 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QSinitialize.
2801
2802 % This draws all with suitable highlighting for the newly chosen pivots
2803 % (which will be shown raised)

```

```

2804 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2805             \let\QSIr\DecoINERT
2806             \let\QSIrr\DecoINERT
2807             \let\QSRr\DecoRIGHTwithPivot
2808 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2809             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2810             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2811 }
2812
2813 % This sorts and then draws, showing where the pivot chosen in the previous
2814 % step go. Next time they will have become "inert". If pivotcount is still at
2815 % zero on exit from \QSpivotStep, then this is the signal to stop before
2816 % executing \QSsortStep.
2817 \def\QSsortStep {\def\QSLr {\noexpand\QS@sort@a}%
2818             \def\QSRr {\noexpand\QS@sort@a}%
2819             \def\QSIr {\noexpand\QSIrr}%
2820             \let\QSIrr\relax
2821             \edef\QS@list{\QS@list}%
2822             \let\QSLr\relax
2823             \let\QSRr\relax
2824             \let\QSIr\relax
2825             \edef\QS@list{\QS@list}%
2826             \let\QSLr\DecoLEFT
2827             \let\QSIr\DecoINERTwithPivot
2828             \let\QSIrr\DecoINERT
2829             \let\QSRr\DecoRIGHT
2830 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2831             \setcounter{cellcount}{0}%
2832             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2833 }
2834
2835 \def\QSinitialize #1{%
2836     % first, we convert the comma separated values into a list of braced items
2837     % we use an \edef, and anyhow many \edef's will be used later
2838     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2839     \let\QSRr\DecoRIGHT
2840     % The \QSRr marker mutated to draw the last element as
2841     % pivot and the earlier ones with the suitable style.
2842     %
2843     % The list of marked braced items \QS@list is used both for drawing
2844     % (as here) and for doing the exchange of elements during sort.
2845     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2846             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2847 }
2848

```

2.38 relationale-algebra.sty

2849 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2850 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]

2851 \RequirePackage{amsmath}

2852 \RequirePackage{amssymb}

Zum Zeichnen von Operatorenbäumen verwenden wir TikZ

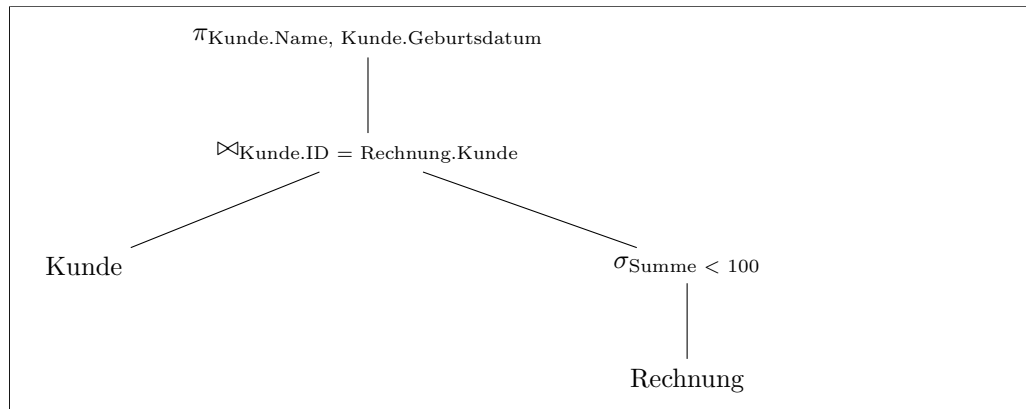
```
\begin{tikzpicture}
  \node
    (pi) {\pi_{\text{Kunde.Name, Kunde.Geburtsdatum}}};

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

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

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

  \node[below=of sigma rechnung]
    {Rechnung}
    edge(sigma rechnung);
\end{tikzpicture}
```



2853 \RequirePackage{tikz}

2854 \usetikzlibrary{positioning}

Privates Makros, das zwei Querstriche erzeugt.

```
2855 \def\o@join{\setbox0=\hbox{\bowtie}}%
2856 \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2857 }
```

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

2858 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}

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

2859 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}

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

2860 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}

2861

2.39 rmodell.sty

```
2862 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2863 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2864 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2865 Datenbanken.]
2866 \RequirePackage{soul}
```

2.39.1 Makro-Kürzel

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

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

```
2867 \def\liPrimaer#1{\ul{#1}}
```

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

```
2868 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

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

```
2869 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2870 \ExplSyntaxOn
2871 \NewDocumentEnvironment { liRmodell }
2872 { +b }
2873 {
2874   \medskip
2875   {
2876     \linespread{2}
2877     \setlength{\parindent}{0pt}
2878     \li@Rmodell@Schrift#1
2879   }
2880   \medskip
2881 } {}
2882 \ExplSyntaxOff
```

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

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

```
2883 \def\liRelationMenge#1#2{
2884 \noindent
2885 #1 : \{ [ #2 ] \}
2886 \par
2887 }
```

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

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

```
2888 \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}
```

```
2889 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2890
```

2.40 sortieren.sty

```
2891 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2892 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
2893 für händisches Sortieren bzw. einen Schreibtischlauf setzen]
```

```
\tikz[
  rectangle split parts=5,
]{
  \node[li sortierung zahlenreihe] (reihe) {\nodepart{one} 2 \nodepart{two} 1 \nodepart{three} 3 \nodepart{four} 4 \nodepart{five} 5}
  \liSortierPfeil{one}{two}
  \liSortierPfeil{two}{three}
  \liSortierMarkierung{two split south}{three split north}
  \liSortierPfeilUnten{three}{one}
}
```



```
2894 \RequirePackage{tikz}
2895 \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.

```
2896 \def\liVertauschen#1{
2897   \directlua{
2898     local sortieren = require('lehramt-informatik-sortieren')
2899     sortieren('#1')
2900   }
2901 }
```

`\liSortierPfeil`

```
2902 \def\liSortierPfeil#1#2{
2903   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2904 }
```

`\liSortierPfeilUnten`

```
2905 \def\liSortierPfeilUnten#1#2{
2906   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2907 }
```

`\liSortierMarkierung`

```
2908 \def\liSortierMarkierung#1#2{\node[
2909   draw,
2910   very thick,
2911   fit=(reihe.#1) (reihe.#2),
2912   inner sep=0pt
2913 ] {}
2914 }

2915 \tikzset{
2916   li sortierung zahlenreihe/.style={
2917     draw,
2918     thin,
2919     font=\large,
2920     rectangle split horizontal,
2921     rectangle split,
2922   }
2923 }
```

```

2924 % https://tex.stackexchange.com/a/140895
2925 \RequirePackage{forest,xstring}
2926 \usetikzlibrary{calc}
2927
2928 \makeatletter
2929 \pgfmathdeclarefunction{strrepeat}{2}{%
2930   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2931   \let\pgfmathresult\pgfutil@empty
2932   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2933     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2934     \advance\pgfmath@count-1\relax
2935   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2936 \makeatother
2937
2938 \def\myNodes{}
2939
2940 \ExplSyntaxOn
2941 \newcommand*\sortList[1]{%
2942   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2943 \ExplSyntaxOff
2944
2945 \forestset{
2946   sort/.code={%
2947     \pgfmathparse{level()}>\forestSortLevel}%
2948     \ifnum\pgfmathresult=0
2949       \StrSubstitute{\forestov{content}}{ }{,}{\myList}%
2950       \sortList\myList
2951       \StrSubstitute{\myList}{,}{ }{\myList}%
2952       \pgfmathparse{strrepeat("1",level())}%
2953       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-1!({\forestov{name}}$)
2954         (m\forestov{name}) {\myList}}%
2955       \pgfmathparse{level()}==\forestSortLevel}%
2956       \ifnum\pgfmathresult=1
2957         \forestOget{\forestov{@first}}{name}\forestFirst
2958         \forestOget{\forestov{@last}}{name}\forestLast
2959         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2960           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2961       \fi
2962       \ifnum\forestov{@parent}=0\else
2963         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2964       \fi
2965       \gappto\myNodes{;}%
2966     \fi}}
2967
2968 \forestset{sort level/.code=%
2969   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2970   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2971

```

2.41 spalten.sty

```
2972 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2973 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2974 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2975 realisiert werden kann.]
2976 \RequirePackage{multicol}
```

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

```
2977 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2978
```

2.42 sql.sty

```
2979 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2980 \ProvidesPackage{lehramt-informatik-tabelle}[2021/09/02 Zu Setzen von SQL]
2981 \liLadePakete{syntax}
2982 \RequirePackage{fancyvrb}
2983 \DefineVerbatimEnvironment{liSqlErgebnis}{Verbatim}
2984 {fontsize=\footnotesize}
2985
```

2.43 struktogramm.sty

```
2986 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2987 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2988 Paket struktex zum Setzen von Struktogrammen]
2989 \RequirePackage{struktex}
2990
```

2.44 syntax.sty

```

2991 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2992 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2993 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2994 \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

```

```

2995 \ExplSyntaxOn
2996 \directlua{
2997   syntax = require('lehramt-informatik-syntax')
2998   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2999   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
3000   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
3001   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
3002   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
3003   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
3004 }
3005 \RequirePackage{hyperref}
3006 \RequirePackage{minted}
3007 % pygmentize -L styles
3008 \usemintedstyle{colorful}
3009 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
3010 %\AfterEndEnvironment{minted}{\end{mdframed}}
3011 %\setminted{breaklines=true,linenos}
3012 \setminted{
3013   breaklines=true,
3014   linenos=false,
3015   fontsize=\footnotesize,
3016 }

```

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

```

3017 \newminted[liJavaAngabe]{java}{
3018   xleftmargin=1cm
3019 }

```

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

Let-Abkürzung: `\let\j=\liJavaCode`

```

3020 \def\liJavaCode#1{
3021   \,
3022   \textcolor{blue}{
3023     \mintinline[
3024       fontsize=\normalsize,
3025       breakanywhere % https://github.com/gpoore/minted/issues/31#issuecomment-
458640242
3026     ]{java}|#1|
3027   }
3028   \,
3029 }

```

\liLatexCode Im Zeilenfluss einen kurzen L^AT_EX-Code-Ausschnitt setzen.

```

3030 \def\liLatexCode#1{\mintinline{latex}|#1|}

```

```

3031 \def\li@GithubLink#1#2{
3032   \begin{flushright}
3033     \tiny
3034     Code-Beispiel~auf~Github~ansehen:~
3035     \href{#1}{\nolinkurl{#2}}
3036   \end{flushright}
3037 }

\liJavaDatei Eine komplette Java-Datei einbinden, die Verzeichnis ./Code/src/main/java/org/bschlangaul
liegt.
3038 \NewDocumentCommand{\liJavaDatei}{ O{firstline=3} m }{
3039   \inputminted[#1]{java}{
3040     \directlua{
3041       syntax.drucke_absoluten_pfad('#2', false)
3042     }
3043   }
3044   \li@GithubLink
3045   {\directlua{syntax.drucke_github_url('#2', false)}}
3046   {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
3047 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
3048 \NewDocumentCommand{\liJavaTestDatei}{ O{firstline=3} m }{
3049   \inputminted[#1]{java}{
3050     \directlua{
3051       syntax.drucke_absoluten_pfad('#2', true)
3052     }
3053   }
3054   \li@GithubLink
3055   {\directlua{syntax.drucke_github_url('#2', true)}}
3056   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
3057 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([
\liJavaExamen$1{$2}{$3}{$4}{$5}
3058 \NewDocumentCommand{\liJavaExamen}{ O{firstline=3} m m m m }{
3059   \inputminted[#1]{java}{
3060     \directlua{
3061       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
3062     }
3063   }
3064   \li@GithubLink
3065   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
3066   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
3067 }

\liAssemblerCode
3069 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
3070 \NewDocumentCommand{\liAssemblerDatei}{ m }{
3071   \inputminted{asm}{#1}
3072 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
3073 \NewDocumentCommand{\liMinispracheDatei}{ m }{
3074   \inputminted{componentpascal}{#1}
3075 }

```



```

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
3076 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
3077 \NewDocumentCommand{\liHaskellDatei}{ m }{
3078   \inputminted{haskell}{#1}
3079 }

3080 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
3081 \def\liSqlCode#1{\mintinline{sql}|#1|}

3082

```

2.45 syntaxbaum.sty

```
3083 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3084 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
3085 Syntaxbäumen mit Hilfe des Pakets tikz-qtreet]
3086 \RequirePackage{tikz-qtreet}
3087
3088 \tikzset{li parsetree/.style={
3089     every internal node/.style={
3090         draw,circle
3091     },
3092     every leaf node/.style={
3093         draw,rectangle
3094     },
3095 }
3096 }
3097
```

2.46 synthese-algorithmus.sty

```
3098 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3099 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
3100 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
3101 Relation in die 3. Normalform]
3102 \liLadePakete{normalformen,mathe,typographie}
3103 \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}}

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

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

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

2.46.4 TeX-Markup Rechtsreduktion

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

\liPseudoUeberschrift{F}

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

\liPseudoUeberschrift{A}

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

2.46.5 TeX-Markup Relationen formen

```
\let\r=\liRelation
\let\u=\underline

\r[R1]{\u{A, D}, E}\
\r[R2]{\u{B, C}, A, E}\
\r[R3]{\u{D}, B}
```

1. Kanonische Überdeckung

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

(a) Linksreduktion

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

(b) Rechtsreduktion

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

(c) Löschen leerer Klauseln

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

(d) Vereinigung

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

2. Relationsschemata formen

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

3. Schlüssel hinzufügen

— Falls eines der in Schritt 2. erzeugten Schemata \mathcal{R}_α 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}_α , 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

```
3104 \def\liSyntheseUeberschrift#1{
3105   {
3106     \bfseries
3107     \rmfamily
3108     \str_case:nn {#1} {
3109       {1} {Kanonische-Überdeckung}
3110       {1-1} {Linksreduktion}
3111       {1-2} {Rechtsreduktion}
3112       {1-3} {Löschen-leerer-Klauseln}
3113       {1-4} {Vereinigung}
3114       {2} {Relationsschemata-formen}
3115       {3} {Schlüssel-hinzufügen}
3116       {4} {Entfernung-überflüssiger-Teilschemata}
3117     }
```

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

```

3120 \def\li@synthese@erklaerung@texte#1{
3121   \str_case:nn {#1} {
3122     {1} {
3123       Die-kanonische-Überdeckung---also-die-kleinst-mögliche-noch-
3124       äquivalente-Menge-von-funktionalen-Abhängigkeiten-kann-in-vier-
3125       Schritten-erreicht-werden.
3126     }
3127     {1-1} {
3128       Führe-für-jede-funktionale-Anhängigkeit~
3129        $\alpha \rightarrow \beta$ -die-Linksreduktion-durch,~
3130       überprüfe-also-für-alle~
3131        $A \in \alpha$ ,~ob- $A$ -überflüssig-ist,~d.h.-ob-
3132        $\beta \subseteq \text{AttributHuelle}\{F, \alpha\}$ .
3133     }
3134     {1-2} {
3135       Führe-für-jede-(verbliebene)-funktionale-Abhängigkeit- $\alpha \rightarrow \beta$ -die-Rechtsreduktion-durch,~überprüfe-also-für-
3136       alle- $B \in \beta$ ,~ob- $B \in \text{AttributHuelle}\{F, (\alpha \rightarrow \beta)\}$ ,~
3137        $\alpha \rightarrow \beta$ -gilt.~In-diesem-Fall-ist- $B$ -auf-der-rechten-Seite-
3138       überflüssig-und-kann-eliminiert-werden,~dh- $\alpha \rightarrow \beta$ -wird-durch- $\alpha \rightarrow (\beta \setminus B)$ -
3139       ersetzt.
3140     }
3141     {1-3} {
3142       Entferne-die-funktionalen-Abhängigkeiten-der-Form- $\alpha \rightarrow \emptyset$ ,~die-im-2.-Schritt-möglicherweise-
3143       entstanden-sind.
3144     }
3145     {1-4} {
3146       Fasse-mittels-der-Vereinigungsregel-funktionale-Abhängigkeiten-
3147       der-Form- $\alpha \rightarrow \beta_1, \dots, \alpha \rightarrow \beta_n$ ,~so-dass- $\alpha \rightarrow \beta_1 \cup \dots \cup \beta_n$ -
3148       verbleibt.
3149     }
3150     % Kemper Seite 197
3151     {2} {
3152       Erzeuge-für-jede-funktionale-Abhängigkeit- $\alpha \rightarrow \beta \in F$ -ein-Relationenschema- $\mathcal{R} \subseteq \mathcal{A}$ -
3153       := $\alpha \cup \beta$ .
3154     }
3155     {3} {
3156       Falls-eines-der-in-Schritt-2.-erzeugten-Schemata- $\mathcal{R} \subseteq \alpha$ -einen-Schlüsselkandidaten-von- $\mathcal{R}$ -bezüglich- $F \subseteq c$ -enthält,~sind-wir-fertig,~sonst-wähle-einen-Schlüsselkandidaten-
3157        $\mathcal{K} \subseteq \mathcal{R}$ - $\subseteq \mathcal{R}$ -aus-und-definiere-folgendes-zusätzliche-Schema:~ $\mathcal{R} \setminus \mathcal{K} := \mathcal{K}$ -und- $\mathcal{F} \setminus \mathcal{K} := \emptyset$ 
3158     }
3159     {4} {
3160       Eliminiere-diejenigen-Schemata- $\mathcal{R} \subseteq \alpha$ ,~die-in-einem-anderen-Relationenschema- $\mathcal{R} \subseteq \alpha'$ -enthalten-sind,~d.h.- $\mathcal{R} \subseteq \alpha \subseteq \alpha'$ .
3161     }
3162   }
3163 }
3164 \def\liSyntheseErklaerung#1{
3165 {
3166   \itshape
3167   \footnotesize

```

```

3180     \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
3181   }
3182 }

```

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

```

3183 \def\liSyntheseUeberErklaerung#1{
3184   \liSyntheseUeberschrift{#1}\par
3185   \liSyntheseErklaerung{#1}
3186 }

```

```

3187 \ExplSyntaxOff
3188

```

2.47 tabelle.sty

```
3189 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3190 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]
3191 \RequirePackage{tabularx}
3192
```

2.48 typographie.sty

```
3193 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3194 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
3195 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
3196 formatierung.sty definiert.]
```

```
3197 \ExplSyntaxOn
```

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

```
3198 \RequirePackage{fontawesome}
```

`\liErledigt` `\liErledigt:` ☑

```
3199 \let\liErledigt=\faCheckSquareO
```

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

```
3200 \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 —

```
3201 \def\liParagraphMitLinien#1{
3202   \noindent
3203   \vrule height 2pt depth -1.6pt width 0.4cm
3204   \enspace
3205   #1
3206   \enspace
3207   \leaders\vrule height 2pt depth -1.6pt \hfill \null
3208   \par
3209   \medskip
3210 }
```

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

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

```
3211 \def\liGeschweifteKlammern#1#2#3#4{
3212   \par
3213   \medskip
3214   \noindent
3215   #1 \, $= \Bigl\{ $
3216   \vspace{#3}
3217   #2
3218   \vspace{#4}
3219   \begin{flushright} $\Bigr\}$ \end{flushright}
3220   \par
3221 }
```

`\liTypoUeberschrift`

```
3222 \def\liTypoUeberschrift#1{
3223   {
3224     \bfseries\rmfamily
3225     #1
3226   }
3227 }
```


\liTypoUeberGross

```
3228 \def\liTypoUeberGross#1{
3229   {
3230     \huge
3231     \liTypoUeberschrift{#1}
3232   }
3233 }
```

\liTypoUeberGross

```
3234 \def\liTypoUeberGROSS#1{
3235   {
3236     \Huge
3237     \liTypoUeberschrift{#1}
3238   }
3239 }
```

3240 \ExplSyntaxOff

3241

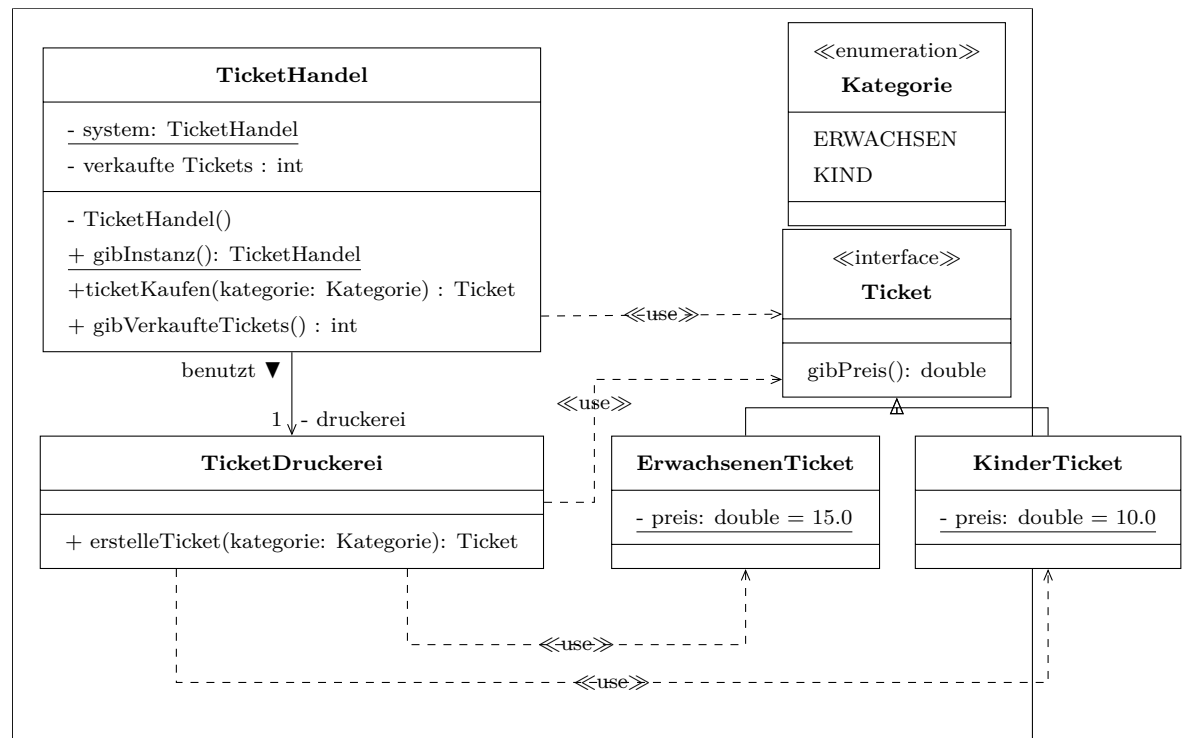
2.49 uml.sty

```

3242 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3243 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
3244 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
3245 Erweiterung bereitstellt]

3246 \RequirePackage{tikz-uml}
3247 \RequirePackage{tikz-uml-activity}
3248 % Not compatible with wasysym
3249 %\RequirePackage{mathabx}
3250 \RequirePackage{wasysym}
3251 \usetikzlibrary{positioning}

3252 \tikzumlset{
3253   fill class=white!0,
3254   font=\footnotesize,
3255   fill object=white!0,
3256   fill note=white!0,
3257   fill state=white!0,
3258   % Use case
3259   fill usecase=white!0,
3260   fill system=white!0,
3261 }
```



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

```

3262 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
3263   \def\@liDirLeft{}
3264   \def\@liDirRight{ \RIGHTarrow}
3265   \pgfkeys{/lese/dir/.is choice}
3266   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
3267   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
3268   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{ \LEFTarrow }}}
3269   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
3270
3271   \def\@liPos{above}
3272   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
3273

```

```

3274 \def\@liDistance{0cm}
3275 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
3276
3277 \pgfkeys{/lese/.cd,#1}
3278
3279 \node[\@liPos = \@liDistance of #2-middle] {
3280   \@liDirLeft{\footnotesize#2}\@liDirRight
3281 };
3282 }
3283

```

2.50 vollstaendige-induktion.sty

```
3284 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3285 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01]
3286 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
3287 Überschriften für die einzelnen Schritte]
```

2.50.1 Makro-Kürzel

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

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

Lade häufig benötigte Pakete

```
3288 \RequirePackage{lehramt-informatik-typographie}
3289 \RequirePackage{lehramt-informatik-mathe}
3290 \RequirePackage{lehramt-informatik-syntax}
3291 \ExplSyntaxOn
```

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

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

```
3292 \def\liInduktionMarkierung#1{\textcolor{violet}{#1}}
```

`\liInduktionErklaerung` Gedacht für die rechte Spalte in der align-Umgebung. Das text-Makro ist dann nicht mehr nötig.

Let-Abkürzung: \let\e=\liInduktionErklaerung

```
3293 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
3294 \def\liInduktionAnfang{
3295   \liPseudoUeberschrift{Induktionsanfang}
3296
3297   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
3298   \liParagraphMitLinien{
3299     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
3300   }
3301 }
```

\liInduktionVoraussetzung

```
3302 \def\liInduktionVoraussetzung{
3303   \liPseudoUeberschrift{Induktionsvoraussetzung}
3304
3305   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
3306   \liParagraphMitLinien{
3307     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ $.
3308   }
3309 }
```

\liInduktionSchritt

```
3310 \def\liInduktionSchritt{
3311   \liPseudoUeberschrift{Induktionsschritt}
3312
3313   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
3314   \liParagraphMitLinien{
3315     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
3316     auch~ $A(n=k+1)$ ~wahr~sein~muss.
3317   }
3318 }

3319 \ExplSyntaxOff
3320
```

2.51 wasserfall.sty

```
3321 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3322 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
3323 \RequirePackage{tikz}
3324 \tikzset{wasserfall/.style={
3325   >=stealth,
3326   node distance = 2mm and -8mm,
3327   start chain = A going below right,
3328   every node/.style = {
3329     draw,
3330     text width=24mm,
3331     minimum height=12mm,
3332     align=center,
3333     inner sep=1mm,
3334     fill=white,
3335     drop shadow={fill=black},
3336     on chain=A
3337   },
3338 }}
3339 \usetikzlibrary{chains,positioning,shadows}
3340
```

2.52 wpkalkuel.sty

```
3341 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3342 \ProvidesPackage{lehramt-informatik-wpkalkuel}[2020/06/13]
```

2.52.1 Makro-Kürzel

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

```
3343 \RequirePackage{amsmath}
3344 \ExplSyntaxOn
```

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

```
3345 \def\liWpKalkuelOhneMathe#1#2{
3346   \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)
3347 }
3348 \def\liWpKalkuel#1#2{
3349   \ifmmode
3350     \liWpKalkuelOhneMathe{#1}{#2}
3351   \else
3352     $\liWpKalkuelOhneMathe{#1}{#2}$
3353   \fi
3354 }
```

\MatheEnv

```
3355 \def\MatheEnv#1{
3356   \medskip
3357
3358   \hspace{1em}#1
3359
3360   \medskip
3361 }
```

\Mathe

```
3362 \def\Mathe#1{
3363   \MatheEnv{${#1}$}
3364 }
```

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

```
3365 \def\liWpEquivalent#1{
3366   \MatheEnv{${\equiv}\hspace{1em}${#1}$}
3367 }
```

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

```
3368 \newlength{\@Skip@Erklaerung@Reset}
3369 \def\liWpErklaerung#1{
3370   \setlength{\@Skip@Erklaerung@Reset}{\leftskip}
3371   \setlength{\leftskip}{0.5cm}
3372
3373   \par
3374   \noindent
3375   {
3376     \scriptsize
3377     #1
3378   }
3379   \par
3380
3381   \setlength{\leftskip}{\@Skip@Erklaerung@Reset}
3382 }
```

\liWpErklaerungVerzweigung

```

3383 \def\liWpErklaerungVerzweigung{
3384   $\liWpKalkuelOhneMathe{if~\{-b~\}~then~\{-a1~\}~else~\{-a2~\}}{Q}
3385   \equiv
3386   (b \land \liWpKalkuelOhneMathe{a1}{Q})
3387   \lor
3388   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
3389 }

3390 \ExplSyntaxOff

3391

```

3 Index

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

Symbols		
\#	114	2669, 2810, 2832, 2846, 3009, 3032, 3219
\,	364, 423, 478, 705, 723, 1355, 1357, 2130, 2131, 2132, 2520, 3021, 3028, 3215	\begingroup 1974, 2462, 2930
\@Skip@Erklaerung@Reset	3368, 3370, 3381	\beschriftung
\@afterheading	1606, 1994, 2099	2012, 2016, 2020, 2024, 2028, 2030
\@afterindentfalse	1605, 1993, 2098	\beta 3129, 3132, 3136, 3137, 3138, 3141, 3151, 3152, 3153, 3158, 3159
\@liDirLeft	3263, 3268, 3280	\bf 2707, 2708, 2709
\@liDirRight	3264, 3266, 3267, 3268, 3269, 3280	\bfseries 564, 1405, 1407, 1583, 1617, 1672, 1730, 2707, 2713, 2715, 2717, 2718, 3106, 3224
\@liDistance	3274, 3275, 3279	\Bigl 3215
\@liPos	3271, 3272, 3279	\Bigr 3219
\\	735, 763, 764, 767, 768, 771, 772, 867, 868, 869, 976, 1010, 1012, 1042, 1051, 1096, 1138, 1139, 1140, 1145, 1146, 1147, 1167, 1981, 2415, 2473, 2476	\bigskip 456, 737, 742, 1601, 1632, 1644, 1723, 1983, 2388
\{	212, 1292, 1302, 1314, 1315, 1320, 1354, 1820, 2460, 2885, 3215, 3384	\bool 342, 424
\}	212, 1292, 1302, 1314, 1315, 1322, 1358, 1821, 2460, 2885, 3219, 3384	\bowtie
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	2855, 2858, 2859, 2860
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	\Box 164
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	\boxtimes 554
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	C
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	\c 1515, 1516
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	\cdot 2175, 2231, 2242
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	\centerline 1635, 1779, 2808, 2830, 2845
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	\cftbeforesecskip 1753
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	\cftbeforesubsecskip 1754, 1755
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	\cftbeforesubsubsecskip 1756
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	\cftsubsecafterpnum 1755
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	\chapter 1405, 1406
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	\char 1981
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	\clearpage 1673, 1761, 1975
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	\cline 735
_	37, 45, 50, 52, 311, 339, 363, 366, 376, 399, 403, 407, 409, 411, 413, 416, 422, 423, 437, 438, 444, 447, 450, 460, 467, 469, 472, 1576, 1588, 1652, 1706	\clist 237, 285, 286, 308, 312, 2942

G		2196, 2225, 2231, 2236, 2645, 2656, 2666, 2676, 3129, 3131, 3137, 3158, 3307	<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> 3386, 3388 <code>\LARGE</code> 1405, 1739 <code>\Large</code> 1584 <code>\large</code> 1575, 1713, 1779, 2919 <code>\leaders</code> 3207 <code>\left</code> 2140, 2530 <code>\LEFTarrow</code> 3268 <code>\leftarrow</code> 671 <code>\leftouterjoin</code> 2858 <code>\leftskip</code> 3370, 3371, 3381 <code>\LehramtInformatikAutorEmail</code> 1930 <code>\LehramtInformatikAutorName</code> 1929 <code>\LehramtInformatikGitBranch</code> 464, 3003 <code>\LehramtInformatikGithubCodeRepo</code> 3002 <code>\LehramtInformatikGithubDomain</code> 461, 2999 <code>\LehramtInformatikGithubRawDomain</code> 3000 <code>\LehramtInformatikGithubTexRepo</code> 462, 3001 <code>\LehramtInformatikRepository</code> 17, 20, 23, 26, 29, 1451, 1690, 1950, 1951, 1952, 1953, 1954, 1955, 1956, 1957, 1958, 1959, 2998 <code>\LehramtInformatikTitel</code> 1925 <code>\leq</code> 2242, 2653, 2673 <code>\let</code> 1304, 1305, 1975, 2804, 2805, 2806, 2807, 2820, 2822, 2823, 2824, 2826, 2827, 2828, 2829, 2839, 2931, 2969, 2970, 3199 <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> 751 <code>\li@fussnote@text</code> 2366, 2372, 2376, 2380, 2384 <code>\li@GithubLink</code> 3031, 3044, 3054, 3065 <code>\li@mget</code> . 1502, 1506, 1526
<code>\g</code> . . . 39, 285, 286, 308, 312, 318, 319, 320, 321, 322, 324, 325, 327, 328, 330, 331, 332, 333, 334, 335, 336, 337, 340, 344, 345, 346, 349, 351, 352, 353, 354, 355, 356, 401, 405, 409, 412, 413, 415, 416, 418, 419, 426, 427, 428, 429, 437, 438, 440, 446, 447, 449, 450, 452, 453, 465, 468, 473, 475, 479, 1550, 1552, 1563, 1565, 1572, 1585, 1586, 1680, 1684, 1688, 1692, 1693, 1694, 1695, 1696, 1698, 1699, 1701, 1732, 1734, 1739, 1743	<code>\includegraphics</code> 1454, 1459 <code>\inhaltsverzeichnis</code> 1972 <code>\input</code> 17, 20, 23, 26, 29, 485, 1689 <code>\inputminted</code> 3039, 3049, 3059, 3071, 3074, 3078 <code>\int</code> 2942 <code>\item</code> 554, 555, 832, 836, 841, 846, 890, 899, 904, 912, 984, 989, 993, 1019, 1059, 1064, 1071, 1079, 1108, 1113, 1117, 1122, 1222, 1227, 1232, 1784, 1785, 2081, 2085, 2205, 2210, 2214, 2224, 2230, 2235, 2247, 2251, 2255, 2259, 2263, 2270, 2274, 2278, 2650, 2653, 2656, 2670, 2673, 2676 <code>\itshape</code> 608, 3178	<code>\gamma</code> . 111, 169, 212, 1315 <code>\gappto</code> 2965 <code>\geometry</code> 5 <code>\geq</code> 1827, 2208, 2645, 2650, 2666, 2670	
H		J	K
<code>\hbox</code> 2855 <code>\headrulewidth</code> . 1431, 1931 <code>\headwidth</code> 1933 <code>\hfill</code> 1591, 2227, 2233, 2238, 3207 <code>\hinweis</code> 2004 <code>\hline</code> 2415 <code>\horizontale</code> . . 1556, 1599 <code>\href</code> 1368, 2128, 2295, 2329, 2338, 3035 <code>\hspace</code> . . 2607, 3358, 3366 <code>\ht</code> 2856 <code>\Huge</code> 1730, 3236 <code>\huge</code> 1672, 3230	<code>\j</code> 1515, 1516, 1518, 1519, 1520, 1525, 1526, 1527	<code>\k</code> 1525 <code>\keys</code> 41, 75, 87, 117, 127, 175, 185, 316, 627, 631, 645, 650, 1379, 1386	<code>\l</code> 68, 69, 70, 71, 72, 73, 76, 77, 78, 79, 80, 82, 84, 89, 90, 91, 92, 93, 94, 109, 110, 111, 112, 113, 114, 115, 118, 119, 120, 121, 122, 123, 124, 130, 131, 132, 133, 134, 135, 136, 167, 168, 169, 170, 171, 172, 173, 176, 177, 178, 179, 180, 181, 182, 188, 189, 190, 191, 192, 193, 194, 625, 628, 633, 634, 637, 642, 643, 646, 647, 652, 1374, 1375, 1376, 1377, 1380, 1381, 1382, 1383, 1389, 1390, 1391, 1392, 1914, 1915, 1916, 2084, 2085, 2086, 2093
I		L	
<code>\i</code> 1525, 1526 <code>\ifcase</code> 2764 <code>\ifLiADDITUM</code> . . 229, 2044 <code>\ifLiANTWORT</code> 233, 2009, 2037 <code>\ifLiEXKURS</code> . . . 231, 2058 <code>\ifmmode</code> 657, 665, 673, 681, 688, 1294, 1308, 1344, 2146, 2156, 2166, 2180, 2454, 2536, 2597, 3349 <code>\IfNoValueTF</code> 2046, 2124, 2128 <code>\ifnum</code> 2691, 2932, 2948, 2956, 2962 <code>\ifx</code> 2028, 2492, 2960 <code>\in</code> 583, 739, 1827, 2190, 2193,	<code>\l</code> 68, 69, 70, 71, 72, 73, 76, 77, 78, 79, 80, 82, 84, 89, 90, 91, 92, 93, 94, 109, 110, 111, 112, 113, 114, 115, 118, 119, 120, 121, 122, 123, 124, 130, 131, 132, 133, 134, 135, 136, 167, 168, 169, 170, 171, 172, 173, 176, 177, 178, 179, 180, 181, 182, 188, 189, 190, 191, 192, 193, 194, 625, 628, 633, 634, 637, 642, 643, 646, 647, 652, 1374, 1375, 1376, 1377, 1380, 1381, 1382, 1383, 1389, 1390, 1391, 1392, 1914, 1915, 1916, 2084, 2085, 2086, 2093		

\li@minc	1505, 1527	\liCpmSpaetI . . .	680, 725	\liEntwurfsErbauerAkteure	1057, 1087
\li@mset	1499, 1507, 1516, 1519	\liCpmVon	663	\liEntwurfsErbauerUml	1037, 1086
\li@numdiscs	1511, 1520, 1526	\liCpmVonOhneMathe . .	663, 666, 668	\liEntwurfsFabrikmethode	1131, 1156
\li@Rmodell@Schrift .	2869, 2878, 2888	\liCpmVonZu	655	\liEntwurfsFabrikmethodeAkteure	1106, 1133
\li@sequence	1512, 1525	\liCpmVonZuOhneMathe	655, 658, 660	\liEntwurfsFabrikmethodeUml	1089, 1132
\li@synthese@erklaerung@text	3120, 3180	\liCpmVorgang	640	\liEntwurfsKompositum	1156
\liAbleitung	1329	\liCpmZu	671	\liEntwurfsKompositumAkteure	1158
liAdditum (environment)	2042	\liCpmZuOhneMathe . .	671, 674, 676	\liEntwurfsKompositumUml	1135, 1157
\LiADDITUMfalse	229	liDiagramm (environ-	2111	\liEntwurfsModellPraesentationSteuerung	1177
\LiADDITUMtrue	229, 1547	ment)	2111	\liEntwurfsModellPraesentationSteuerungAK	1179
liAHuelle (environment)	2461	liEinbettung (environ-	2006	\liEntwurfsModellPraesentationSteuerungUm	1160, 1178
\liAlphabet	1314	ment)	2006	\liEntwurfsStellvertreter	1201
liAntwort (environment)	2007	\liEntwurfs	1177	\liEntwurfsStellvertreterCode	1195, 1203
\LiANTWORTfalse	233	\liEntwurfsAbstrakteFabrik	807	\liEntwurfsStellvertreterUml	1181, 1202
\LiANTWORTtrue	233	\liEntwurfsAbstrakteFabrikBeschreibung	755, 808	\liEntwurfsZustand	1242
\liAnweisung	1905	\liEntwurfsAbstrakteFabrikCode	802, 812	\liEntwurfsZustandAkteure	1220, 1244
\liAssemblerCode	3069	\liEntwurfsAbstrakteFabrikUml	760, 810	\liEntwurfsZustandCode	1238, 1245
\liAssemblerDatei	3070	\liEntwurfsAdapter	859	\liEntwurfsZustandUml	1205, 1243
\liAttribut	2888	\liEntwurfsAdapterAkteure	829, 861	\liEpsilon	1300
\liAttributHuelle	2452, 3132, 3137	\liEntwurfsAdapterCode	853, 862	\liErAttribute	1258, 1272, 1274
\liAttributHuelleOhneMathe	2452, 2455,	\liEntwurfsAdapterUml	814, 860	\liErDatenbankName	1277
2457, 2471, 2481, 2489		\liEntwurfsBeobachter	930	\liErEntity 1256, 1260, 1262	
\liAttributMenge	2460, 2472, 2475,	\liEntwurfsBeobachterAkteure	888, 932	\liErErledigt	3199
2482, 2483, 2497, 2499		\liEntwurfsBeobachterCode	922, 933	\liErMpAttribute	1271
\liAufgabe	16	\liEntwurfsBeobachterUml	864, 931	\liErMpEntity	1259
\liAufgabenMetadaten	47, 1569, 1704	\liEntwurfsDekorierer	962	\liErMpRelationship 1265	
\liAufgabenTitel	55	\liEntwurfsDekoriererAkteure	964	\liErRelationship	1257, 1266, 1268
\liAusdruck	1351	\liEntwurfsDekoriererCode	954, 965	\liExamensAufgabe	19
\liAutomat	66	\liEntwurfsDekoriererUml	935, 963	\liExamensAufgabeA	28
\liAutomatenKante	98	\liEntwurfsEinfacheFabrik	998	\liExamensAufgabeTA	25
\liBandAlphabet	1315	\liEntwurfsEinfacheFabrikAkteure	982, 1000	\liExamensAufgabeTTA	22
\liBedingung	1906	\liEntwurfsEinfacheFabrikUml	967, 999	\liExkurs (environment) 2057	
\liBedingungDrei	2195, 2238, 2278	\liEntwurfsEinzelstueck	1028	\LiEXKURfalse	231
\liBedingungEins	2189, 2227, 2270	\liEntwurfsEinzelstueckAkteur	1017, 1033	\LiEXKURtrue	231
\liBedingungFalsch	1908	\liEntwurfsEinzelstueckBeschreibung	1002, 1029	\liFalsch	555
\liBedingungWahr	1907	\liEntwurfsEinzelstueckCode	1025, 1035	\LiFlaci	1361
\liBedingungZwei	2192, 2233, 2274	\liEntwurfsEinzelstueckUml	1007, 1031	\liFremd	2868
\liBeschriftung	1996	\liEntwurfsErbauer	1085	\liFunktionaleAbhaengigkeit	2491, 2494, 2502
\liBindeAufgabeEin	1687			\liFunktionaleAbhaengigkeiten	2505
\liChomskyErklaerung	574, 615			\liFussnote	2365, 2367
\liChomskyUeberErklaerung	613			\liFussnoteDreiText	2379, 2397
\liChomskyUeberschrift	562, 614				
\liCpmEreignis	623				
\liCpmFruehErklaerung	694				
\liCpmFruehI	687, 707				
\liCpmSpaetErklaerung	712				

<code>\liFussnoteEinsText</code> .	<code>\liLinksReduktionInline</code>	<code>\liPotenzmengeOhneMathe</code>
..... 2371 , 2391 2478 , 2486	... 1302 , 1303 , 1304
<code>\liFussnoteLink</code> 2127	<code>\liMasterExkurs</code> 2283	<code>\liPrimaer</code> 2867
<code>\liFussnoten</code> 2387	<code>\liMasterFaelle</code> 2222 , 2290	<code>\liProblemBeschreibung</code>
<code>\liFussnoteUrl</code> . 1175 , 2123	<code>\liMasterFallRechnung</code> 1773
<code>\liFussnoteVierText</code> 2268	<code>\liProblemClique</code> ... 1796
..... 2383 , 2400	<code>\liMasterVariablen</code> ..	<code>\liProblemName</code>
<code>\liFussnoteZweiText</code> 2199 , 2285 1772 , 1779 ,
..... 2375 , 2394	<code>\liMasterVariablenDeklaration</code>	1791 , 1793 , 1806 ,
<code>\liGeschweifteKlammern</code> 2245	1817 , 1818 , 1826 , 1827
... 1333 , 2506 , 3211	<code>\liMasterWolframLink</code> 2293	<code>\liProblemSat</code> 1825
<code>\liGrafikCCLizenz</code> ...	<code>\liMenge</code> ... 76 , 77 , 79 ,	<code>\liProblemSubsetSum</code> .
..... 1453 , 1635	118 , 119 , 120 , 124 , 1816 , 1825
<code>\liGrafikLogo</code>	176 , 177 , 178 , 182 ,	<code>\liProblemVertexCover</code>
... 1458 , 1613 , 1722	1292 , 1341 , 1380 , 1381 1796 , 1804
<code>\liGrafikLogoPfad</code> ...	<code>\liMengeOhneMathe</code> ...	<code>\liProduktionen</code> 1340 , 1382
... 1450 , 1455 , 1460	... 1292 , 1295 , 1297	<code>liProduktionsRegeln</code>
<code>\liGrammatik</code> 1372	<code>\liMetaBschlangaulSammlung</code>	(environment) 1330
<code>liGraphenFormat</code> (envi- 1618 , 2315	<code>liProjektSprache</code> (envi-
ronment) 1492	<code>\liMetaCCLink</code> .. 1640 , 2327	ronment) 2005
<code>\liHanoi</code> 1499	<code>\liMetaEmailLink</code>	<code>\liPseudoUeberschrift</code>
<code>\liHaskellCode</code> 3076 2337 , 2353 1982 ,
<code>\liHaskellDatei</code> 3077	<code>\liMetaHermineBschlangaulAndFriends</code>	2048 , 2051 , 2412 ,
<code>\liInduktionAnfang</code> . 3294	... 1621 , 1725 , 2318	2422 , 3295 , 3303 , 3311
<code>\liInduktionErklaerung</code>	<code>\liMetaHilfMit</code>	<code>\liPumpingKontextfrei</code>
..... 3293	... 1649 , 2344 , 2355 2664
<code>\liInduktionMarkierung</code>	<code>\liMetaQuelltext</code>	<code>\liPumpingRegulaer</code> . 2643
..... 3292 1651 , 2355	<code>liQuellen</code> (environment)
<code>\liInduktionSchritt</code> 3310	<code>\liMetaSetze</code> 2081
<code>\liInduktionVoraussetzung</code>	.. 36 , 48 , 1570 , 1705	<code>\liRechtsReduktionInline</code>
..... 3302	<code>\liMetaUeberDasProjekt</code> 2486
<code>liJavaAngabe</code> (environ- 1626 , 2321	<code>\liRekursionsGleichung</code>
ment) 3017	<code>\liMinimierungErklaerung</code> 2186 , 2249
<code>\liJavaCode</code> 3020 2424	<code>\liRelation</code> 2516
<code>\liJavaDatei</code> .. 753 , 3038	<code>\liMinispracheDatei</code> 3073	<code>liRelationenSchemaFormat</code>
<code>\liJavaExamen</code> 3058	<code>\linespread</code> 2876	(environment) 2889
<code>\liJavaTestDatei</code> ... 3048	<code>\liNichtsZuTun</code> 3200	<code>\liRelationMenge</code> ... 2883
<code>liKasten</code> (environment) 1420	<code>\liO</code> 2162 , 2190	<code>\liRichtig</code> 554
<code>\liKellerAutomat</code> 107	<code>\liOmega</code> 2152 , 2196	<code>liRmodell</code> (environment)
<code>\liKellerKante</code> 145	<code>\liOmegaOhneMathe</code> 2869
<code>\liKellerUebergang</code> 2152 , 2157 , 2159	<code>\liRundeKlammer</code>
..... 140 , 146	<code>\liONotationO</code> 2532	2139 , 2143 , 2153 ,
<code>\liKontrollCode</code> 1909	<code>\liOOhneMathe</code>	2163 , 2177 , 2529 , 2533
<code>liKontrollflussgraph</code>	... 2162 , 2167 , 2169	<code>\liSetzeExamenTeilaufgabeNr</code>
(environment) 1897	<code>\liParagraphMitLinien</code> 1683
<code>\liKontrollKnotenPfad</code> 610 , 695 ,	<code>\liSetzeExamenThemaNr</code>
..... 1911	713 , 2426 , 3180 , 1679
<code>\liKontrollTextzeileKnoten</code>	3201 , 3298 , 3306 , 3314	<code>\liSortierMarkierung</code> 2908
..... 1910 , 1915	<code>\liPetriErreichKnotenDrei</code>	<code>\liSortierPfeil</code> 2902
<code>\liKurzeTabellenLinie</code> 735 2606	<code>\liSortierPfeilUnten</code> 2905
<code>\liLadeAllePakete</code> ...	<code>\liPetriErreichTransition</code>	<code>\liSpaltenUmbruch</code> .. 2977
..... 239 , 1675 2603	<code>\liSqlCode</code> 3081
<code>\liLadePakete</code> 59 ,	<code>\liPetriSetzeSchluessel</code>	<code>\listen@punkt</code> .. 2081 , 2093
62 , 235 , 240 , 561 , 2548	<code>\liStrich</code> 1771
622 , 1291 , 1535 ,	<code>\liPetriTransitionsName</code>	<code>\liSyntheseErklaerung</code>
1661 , 1769 , 2364 , 2595 , 2607 3120 , 3185
2444 , 2612 , 2981 , 3102	<code>\liPetriTransitionsNameOhneMathe</code>	<code>\liSyntheseUeberErklaerung</code>
<code>\liLatexCode</code> 3030	... 2595 , 2598 , 2600 3183
<code>\liLeereZelle</code> 2402	<code>\liPetriTransPfeile</code> 2607	<code>\liSyntheseUeberschrift</code>
<code>liLernkartei</code> (environ-	<code>\liPolynomiellReduzierbar</code> 3104 , 3184
ment) 2101 1789	<code>\liT</code> 2172 , 2187 , 2201 , 2265
<code>\liLinksReduktion</code> .. 2469	<code>\liPotenzmenge</code>	<code>\liTeilen</code> 2449
	... 1301 , 1305 , 2618	<code>\literatur</code> 1938 , 1962

<code>\liTheta</code>	2142, 2193, 2225, 2231, 2236	<code>\liZustandsPaarVariablenName</code>	208, 235, 624, 641, 680, 687, 736, 741, 1351, 1373, 1453, 1458, 1789, 1912, 1940, 2123, 2127, 2505, 2516, 2603, 3038, 3048, 3058, 3070, 3073, 3077, 3262
<code>\liThetaOhneMathe</code>	2142, 2147, 2149	<code>\llap</code>	2856
<code>\liTOhneMathe</code>	2172, 2181, 2183	<code>\log</code>	2190, 2193, 2196, 2225, 2231
<code>\liTuringKante</code>	208	<code>\loop</code>	2689
<code>\liTuringLeerzeichen</code>	164, 172	<code>\lor</code>	3387
<code>\liTuringMaschine</code>	165	<code>\ltimes</code>	2301
<code>\liTuringUeberfuehrung</code>	211	M	
<code>\liTuringUebergange</code>	203, 209		
<code>\liTuringUebergangZelle</code>	198	<code>\makeatletter</code>	2928
<code>\liTypoUeberGROSS</code>	3234	<code>\makeatother</code>	2936
<code>\liTypoUeberGross</code>	3228, 3234	<code>\marginpar</code>	1261, 1267, 1273, 1980
<code>\liTypoUeberschrift</code>	3222, 3231, 3237	<code>\mathbb</code>	1827, 2676, 3307
<code>\liUeberfuehrungsFunktion</code>	1306	<code>\mathbin</code>	2858, 2859, 2860
<code>\liUeberfuehrungsFunktionOhneMathe</code>	1306, 1309, 1311	<code>\mathcal</code>	2163, 2533, 3158, 3163, 3165, 3166, 3167
<code>liUebergangsTabelle</code> (environment)	2410	<code>\Mathe</code>	3362
<code>\liUeberschriftDreiecksTabelle</code>	2420	<code>\MatheEnv</code>	3355, 3363, 3366
<code>\liUmlLeserichtung</code>	3262	<code>\mathord</code>	2308, 2309
<code>\liVertauschen</code>	2896	<code>\mdfsetup</code>	1419, 2013, 2017, 2021, 2025
<code>\liWortInSprache</code>	736	<code>\medskip</code>	1577, 1594, 1622, 1781, 1989, 1999, 2001, 2091, 2119, 2874, 2880, 3209, 3213, 3356, 3360
<code>\liWortNichtInSprache</code>	741	<code>\memph</code>	1980
<code>\liWpEquivalent</code>	3365	<code>\mintinline</code>	3023, 3030, 3069, 3076, 3081
<code>\liWpErklaerung</code>	3368	<code>\mkern</code>	2858, 2859, 2860
<code>\liWpErklaerungVerzweigung</code>	3383	<code>\mlq</code>	2306, 2308
<code>\liWpKalkuel</code>	3345	<code>\mrq</code>	2306, 2309
<code>\liWpKalkuelOhneMathe</code>	3345, 3350, 3352, 3384, 3386, 3388	<code>\msg</code>	483
<code>\liZustandsBuchstabe</code>	1316, 1325, 1327, 1345, 1347	<code>\myList</code>	2949, 2950, 2951, 2954
<code>\liZustandsBuchstabeGross</code>	1317, 1326, 1328	<code>\myNodes</code>	2938, 2953, 2959, 2963, 2965
<code>\liZustandsmenge</code>	1304	N	
<code>\liZustandsmengeNr</code>	1318, 2627		
<code>\liZustandsmengeNrGross</code>	1326	<code>\NeedsTeXFormat</code>	1, 14, 32, 57, 225, 489, 550, 557, 619, 732, 747, 1248, 1284, 1397, 1435, 1444, 1465, 1494, 1532, 1658, 1765, 1837, 1920, 1936, 1942, 1964, 2135, 2298, 2311, 2361, 2440, 2523, 2543, 2609, 2634, 2639, 2683, 2849, 2862, 2891, 2972, 2979, 2986, 2991, 3083, 3098, 3189, 3193, 3242, 3284, 3321, 3341
<code>\liZustandsMengenSammlung</code>	2614	<code>\neg</code>	3388
<code>\liZustandsMengenSammlungNr</code>	2623	<code>\negthinspace</code>	2140, 2530
<code>\liZustandsmengeOhneMathe</code>	1304	<code>\newcounter</code>	2703, 2704
<code>\liZustandsname</code>	1327	<code>\NewDocumentCommand</code>	67, 108, 145, 166,
<code>\liZustandsnameGross</code>	1328, 2615, 2624	O	
<code>\liZustandsnameTiefgestellt</code>	1343		
<code>\liZustandsPaar</code>	2404	<code>\o</code>	2532, 2537, 2539
		<code>\o@join</code>	2855, 2858, 2859, 2860
		<code>\Omega</code>	2153
		<code>\omega</code>	2645, 2646, 2666, 2667
		<code>\or</code>	2766
		P	
		<code>\pagestyle</code>	1430, 1711
		<code>\par</code>	434, 455, 614, 1362, 1370, 1553, 1557, 1560, 1577, 1592, 1603, 1619, 1621, 1622, 1630, 1714, 1718, 1722, 1724, 1725, 1732, 1734, 1739, 1743, 1747, 1985, 1991, 1997, 2073, 2097, 2106, 2808, 2830, 2845, 2886, 3184, 3208, 3212, 3220, 3373, 3379
		<code>\paragraph</code>	1407
		<code>\parindent</code>	2877
		<code>\path</code>	99, 146, 209, 652

<code>\pgfkeys</code> .. 2559, 3265, 3266, 3267, 3268, 3269, 3272, 3275, 3277	<code>\QSLr</code> 2777, 2784, 2795, 2796, 2804, 2817, 2822, 2826	<code>\roman</code> 1415, 1417
<code>\pgfmath@count</code> 2930, 2932, 2934	<code>\QSPivotStep</code> 2690, 2800, 2804, 2815	<code>\romannumeral</code> 2775
<code>\pgfmath@smuggleone</code> 2935	<code>\QSR</code> 2777	<code>\rtimes</code> 2301
<code>\pgfmathdeclarefunction</code> 2929	<code>\QSRr</code> 2786, 2807, 2818, 2823, 2829, 2838, 2839, 2840	<code>\rule</code> 1559, 2808, 2830, 2845, 2856
<code>\pgfmathhint</code> 2930	<code>\QSSortStep</code> 2692, 2800, 2816, 2817	S
<code>\pgfmathparse</code> 1506, 2947, 2952, 2955, 2969, 2970	<code>\quad</code> 2368	<code>\sb</code> 72, 82, 84, 113, 171, 590, 591, 595, 598, 599, 600, 682, 684, 689, 691, 1345, 1347, 2190, 2193, 2196, 2225, 2231, 2430, 2595, 2604, 3151, 3152, 3153, 3158, 3162, 3163, 3166, 3167, 3170, 3171, 3172
<code>\pgfmathresult</code> 1507, 2930, 2931, 2933, 2935, 2948, 2956, 2969, 2970	R	<code>\scriptscriptstyle</code> 655, 663, 671
<code>\pgfutil@empty</code> 2931	<code>\raisebox</code> 1910	<code>\scriptsize</code> 1364, 1639, 1850, 1857, 1863, 1925, 1926, 1929, 1930, 3293, 3346, 3376
<code>\pgfutil@loop</code> 2932	<code>\relax</code> 1975, 2777, 2820, 2822, 2823, 2824, 2932, 2934	<code>\section</code> ... 52, 1672, 1681
<code>\pgfutil@repeat</code> 2935	<code>\renewcommand</code> 1410, 1411, 1412, 1413, 1416, 1417, 1431, 1432, 1755, 1759, 1931, 1932, 2410	<code>\sectionbreak</code> 1673
<code>\preceq</code> 1792	<code>\repeat</code> 2693	<code>\seq</code> . 1914, 1915, 1916, 2084, 2085, 2086, 2093
<code>\prime</code> 1771	<code>\RequirePackage</code> 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, 1671, 1676, 1677, 1709, 1770, 1839, 1939, 1944, 1945, 1961, 1968, 1969, 1970, 2138, 2302, 2303, 2526, 2527, 2528, 2545, 2547, 2637, 2697, 2699, 2851, 2852, 2853, 2866, 2894, 2925, 2976, 2982, 2989, 2994, 3005, 3006, 3086, 3191, 3198, 3246, 3247, 3249, 3250, 3288, 3289, 3290, 3323, 3343	<code>\setbox</code> 2855
<code>\printbibliography</code> . 1962	<code>\right</code> 2140, 2530	<code>\setcounter</code> 1408, 1674, 1708, 2809, 2831, 2845
<code>\ProvidesPackage</code> 2, 15, 33, 58, 226, 490, 551, 558, 620, 733, 748, 1249, 1285, 1398, 1436, 1445, 1466, 1495, 1533, 1659, 1766, 1838, 1921, 1937, 1943, 1965, 2136, 2299, 2312, 2362, 2441, 2524, 2544, 2610, 2635, 2640, 2684, 2850, 2863, 2892, 2973, 2980, 2987, 2992, 3084, 3099, 3190, 3194, 3243, 3285, 3322, 3342	<code>\RIGHTarrow</code> ... 3264, 3269	<code>\setganttlinklabel</code> .. 1439, 1440, 1441, 1442
Q	<code>\Rightarrow</code> 739, 744	<code>\setlength</code> 1753, 1754, 1756, 1933, 2877, 3370, 3371, 3381
<code>\QS@list</code> 2810, 2821, 2825, 2832, 2838, 2843, 2846	<code>\rightarrow</code> 212, 578, 583, 591, 595, 597, 598, 600, 655, 663, 2607, 3129, 3136, 3138, 3141, 3146, 3151, 3152, 3157	<code>\setmainfont</code> 1401
<code>\QS@select@equal</code> 2785, 2789	<code>\rightouterjoin</code> 2859	<code>\setmainlanguage</code> 487
<code>\QS@select@greater</code> 2786, 2790	<code>\rmfamily</code> 565, 1526, 3107, 3224	<code>\setminted</code> 3011, 3012
<code>\QS@select@smaller</code> 2781, 2784, 2788	<code>\Roman</code> 1415	<code>\setminus</code> 2490
<code>\QS@sort@a</code> 2763, 2796, 2817, 2818		<code>\setul</code> 2868
<code>\QS@sort@b</code> 2763, 2764		<code>\shoveleft</code> 2470
<code>\QS@sort@c</code> 2767, 2774		<code>\shoveright</code> 2474
<code>\QS@sort@d</code> 2775, 2783		<code>\Sigma</code> 69, 110, 168, 1314, 1315, 1375
<code>\QS@sort@empty</code> . 2765, 2770		<code>\sigma</code> 588, 590, 591
<code>\QS@sort@single</code> 2766, 2771		<code>\SLASH</code> 1981
<code>\QS@initialize</code> 2688, 2800, 2835		<code>\small</code> 2114
<code>\QSIr</code> . 2771, 2777, 2785, 2805, 2819, 2824, 2827		<code>\sort</code> 2942
<code>\QSIrr</code> 2806, 2819, 2820, 2828		<code>\sortList</code> 2941, 2950
		<code>\square</code> 555
		<code>\stepcounter</code> 2729, 2734, 2739, 2742, 2744, 2748, 2750, 2754, 2756
		<code>\stichwoerter</code> .. 1549, 1597
		<code>\str</code> 367, 566, 575, 2010, 2619, 2628, 3108, 3121
		<code>\string</code> 2472, 2482
		<code>\StrSubstitute</code> . 2949, 2951
		<code>\strut</code> 2247, 2251, 2255, 2259, 2263, 2977
		<code>\subsection</code> 1685
		<code>\subseq</code> 3132, 3165, 3172
		<code>\subsubsection</code> 1706

T		
\tableofcontents	1760, 1976	1376, 1377, 1380, 1381, 1382, 1383, 1550, 1563, 1572, 1585, 1680, 1684, 1688, 1695, 1698, 2173
\TeX	2356	
\text	82, 84, 187, 2452, 3293, 3346	\tmp
\textbf	702, 720, 1256, 1551, 1797, 1806, 1817, 1826, 1987, 2000, 2030, 2073, 2090, 2106, 2415	\TmpPlaceEight
\textcolor	1909, 3022, 3292	\TmpPlaceFive
\textit	1096, 1138, 1139, 1140, 1141, 1564, 2116, 2460, 2520	\TmpPlaceFour
\textsc	1772	\TmpPlaceNine
\textsf	2090	\TmpPlaceOne
\textstyle	2210, 2242	\TmpPlaceSeven
\texttt	1279, 1772, 1906, 1907, 1908, 1909, 3346	\TmpPlaceSix
\textwidth	1559, 1933	\TmpPlaceTen
\thematik	1562, 1591	\TmpPlaceThree
\thepage	1429, 1928	\TmpPlaceTwo
\theparagraph	1407	\TmpScale
\thesection	1672	\TmpTransitionEight
\Theta	2143	\TmpTransitionFive
\thinspace	3346	\TmpTransitionFour
\tikz	1910	\TmpTransitionNine
tikz: bbaum	26	\TmpTransitionOne
tikz: li binaer baum	24	\TmpTransitionSeven
\tikzchildnode	510	\TmpTransitionSix
\tikzparentnode	510	\TmpTransitionTen
\tikzset	101, 148, 214, 495, 521, 1471, 1841, 2585, 2711, 2915, 3088, 3324	\TmpTransitionThree
\tikzumlset	3252	\TmpTransitionTwo
\times	212	\TmpX
\tiny	1262, 1268, 1274, 1647, 1909, 1980, 3033	\TmpY
\titleformat	\today
\titlespacing	1406	\ttfamily
\tl	39, 68, 69, 70, 71, 72, 73, 76, 77, 78, 79, 80, 82, 84, 109, 110, 111, 112, 113, 114, 115, 118, 119, 120, 121, 122, 123, 124, 167, 168, 169, 170, 171, 172, 173, 176, 177, 178, 179, 180, 181, 182, 309, 313, 340, 344, 345, 346, 349, 354, 355, 356, 368, 377, 412, 415, 418, 426, 427, 428, 429, 440, 446, 449, 452, 468, 475, 625, 628, 633, 634, 642, 643, 646, 647, 1353, 1374, 1375,	U
		\ul
		\umlaggreg
		\umlassoc
		\umlclass
		\umldep
		\umlHVHaggreg
		\umlinherit
		\umlnote
		\umlreal
		\umlsimpleclass
		\umlstatic
		\umluniaggreg
		\umluniassoc
		\umlVHuniassoc
		\umlVHVdep
		\umlVHVinherit
		\umlVHVreal
		\UParrow
		\url
		\usemintedstyle
		\usetikzlibrary
		\value
		\varepsilon
		\vfill
		\vrule
		\vspace
		X
		\xappto
		\xdef
		\xintApply
		\xintApplyUnbraced
		\xintCSVtoList
		\xintFor
		\xintIfEq
		\xintIfForLast
		\xintIfGt
		\xintIfLt
		\xintLength
		\xintntheft
		Z
		\ZB
		\zB
		\zustandsnamens@liste