

lehramt-informatik

Hermine Bschlangaul *

September 7, 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
2.6	baum.sty	22
2.6.1	Binärbaum	23
2.6.2	AVL-Baum	24
2.6.3	B-Baum	25
2.7	checkbox.sty	26
2.8	chomsky-normalform.sty	27
2.8.1	Makro-Kürzel	27
2.8.2	TeX-Markup-Grundgerüst	27
2.8.3	Konkretes TeX-Markup-Beispiel	27
2.9	cpm.sty	30
2.9.1	Makro-Kürzel	30
2.9.2	TeX-Markup-Beispiel: Graph	30
2.9.3	TeX-Markup-Beispiel: Ergebnistabelle	30
2.9.4	TeX-Markup-Beispiel: Nebenrechnungstabelle „Frühester Zeitpunkt“	30
2.9.5	TeX-Markup-Beispiel: Nebenrechnungstabelle „Spätester Zeitpunkt“	31
2.9.6	Hilfsmakros für Zeitpunkt-Tabelle	31
2.10	cyk-algorithmus.sty	34
2.10.1	Makro-Kürzel	34
2.10.2	TeX-Markup-Beispiel	34
2.11	entwurfsmuster.sty	35
2.11.1	Namensschema der Entwurfsmuster-Makros:	35
2.11.2	Reihenfolge	35
2.11.3	Abstrakte Fabrik (Abstract Factory)	35
2.11.4	Adapter	37
2.11.5	Beobachter (Observer)	38
2.11.6	Dekorierer (Decorator)	40
2.11.7	Einfache Fabrik (Simple Factory)	41
2.11.8	Einzelstück (Singleton)	42

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

2.11.9	Erbauer (Builder)	43
2.11.10	Fabrikmethode (Factory Method)	44
2.11.11	Kompositum (Composite)	46
2.11.12	Modell-Präsentation-Steuerung (Model-View-Controller)	46
2.11.13	Stellvertreter (Proxy)	47
2.11.14	Zustand (State)	48
2.12	er.sty	50
2.12.1	Tex-Markup-Beispiel: Komplettes Diagramm	50
2.12.2	Tex-Markup-Beispiel: EER Enhanced Entity-Relation-Modell nach Elmasri/Navante	51
2.12.3	Makro-Kürzel	51
2.13	formale-sprachen.sty	53
2.14	formatierung.sty	56
2.14.1	Schriftarten / Typographie	56
2.14.2	Farben	56
2.14.3	Überschriften	56
2.14.4	Listen	56
2.14.5	Kasten	56
2.14.6	Header	56
2.15	gantt.sty	57
2.16	grafik.sty	58
2.17	graph.sty	59
2.18	hanoi.sty	61
2.19	klassen-konfiguration-aufgabe.sty	62
2.20	klassen-konfiguration-examen.sty	63
2.21	komplexitaetstheorie.sty	65
2.21.1	Makro-Kürzel	65
2.22	kontrollflussgraph.sty	67
2.22.1	Makro-Kürzel	67
2.22.2	TeX-Markup-Beispiel	67
2.22.3	TikZ: pin	67
2.22.4	Umgebungen	68
2.22.5	Makros	69
2.23	kopf-fusszeilen.sty	70
2.24	literatur-dummy.sty	71
2.25	literatur.sty	72
2.26	makros.sty	73
2.27	master-theorem.sty	77
2.27.1	Makro-Kürzel	77
2.28	mathe.sty	81
2.29	minimierung.sty	82
2.30	normalformen.sty	85
2.30.1	Makro-Kürzel	85
2.31	petri.sty	88
2.31.1	Makro-Kürzel	88
2.32	potenzmengen-konstruktion.sty	90
2.33	pseudo.sty	92
2.34	pumping-lemma.sty	93
2.35	quicksort.sty	94
2.36	relationale-algebra.sty	97
2.37	rmodell.sty	98
2.37.1	Makro-Kürzel	98
2.38	sortieren.sty	99
2.39	spalten.sty	101
2.40	sql.sty	102
2.41	struktogramm.sty	103
2.42	syntax.sty	104
2.42.1	Makro-Kürzel	104

2.43	syntaxbaum.sty	106
2.44	synthese-algorithmus.sty	107
2.44.1	Makro-Kürzel	107
2.44.2	TeX-Markup Grundgerüst	107
2.44.3	TeX-Markup Linksreduktion	107
2.44.4	TeX-Markup Rechtsreduktion	107
2.44.5	TeX-Markup Relationen formen	108
2.45	tabelle.sty	111
2.46	typographie.sty	112
2.47	uml.sty	114
2.48	vollstaendige-induktion.sty	116
2.48.1	Makro-Kürzel	116
2.49	wasserfall.sty	118
2.50	wpkalkuel.sty	119
2.50.1	Makro-Kürzel	119
3	Index	120

1 Klassen

1.1 Vorlage Theorie-Teil

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

\begin{document}

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

\chapter{Thema des Theorie-Teils}

\literatur

\end{document}
```

1.2 Vorlage Aufgabensammlung

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

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

1.3 Vorlage Aufgabe

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

\end{document}
```

2 Pakete

2.1 abmessung.sty

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

2.2 aufgaben-einbinden.sty

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

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

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

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

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

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

31
```

2.3 aufgaben-metadaten.sty

```

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

35 \ExplSyntaxOn

\liMetaSetze Setze die Metadaten einer Aufgabe.

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

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

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

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

54 \ExplSyntaxOff

\liAufgabenTitel Momentan eine dummy Makro das die Thematik enthält.

55 \def\liAufgabenTitel#1{}

56

```

2.4 automaten.sty

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

2.4.1 Endlicher Automat

```
59 \liLadePakete{formale-sprachen}

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

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



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

`\liAutomat` `\liAutomat[$\langle automaten-name \rangle$]{ $\langle zustaende=Z, alphabet=\Sigma, delta=\delta, ende=E, start=z_0 \rangle$ }`

- `\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 }

```

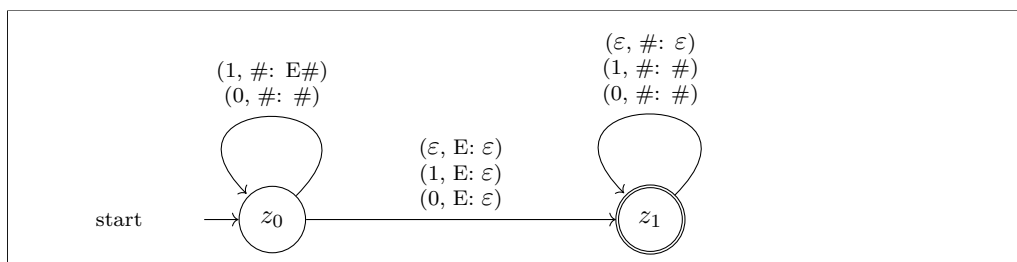
2.4.2 Kellerautomat

```
\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]
{\zustaende=Z,alphabet=\Sigma,kelleralphabet=\Gamma,delta=\delta,start=z_0,kellerboden=\#,ende=E}

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

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

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

```

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

```

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

(a, #: A)
(b, #: 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

\liLadePakete

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

\liLadeAllePakete

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

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

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

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

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

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

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

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

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

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

```

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

357 \cs_gset:Npn \_gib_examen_titel: {
358   \cs_set:Nn \_trenner: { \, / \, }
359   \bool_if:nTF
360   {
361     ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
362     ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
363     ! \tl_if_empty_p:N \g_auf_examen_monat_tl &&
364     ! \tl_if_empty_p:N \g_auf_examen_aufgabe_nr_tl
365   }
366   {
367     {
368       \footnotesize
369       \par
370       \noindent
371       Staatsexamen ~
372       \g_auf_examen_nummer_tl \_trenner:
373       \g_auf_examen_jahr_tl \_trenner:
374
375       \tl_case:Nn \g_auf_examen_monat_tl
376       {
377         { 03 } { Frühjahr }
378         { 09 } { Herbst }
379       } \_trenner:
380
381       \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
382         Thema ~ Nr. ~ \g_auf_examen_thema_nr_tl \_trenner:
383       }
384       \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
385         Teilaufgabe ~ Nr. ~ \g_auf_examen_teilaufgabe_nr_tl \_trenner:
386       }
387       \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} {
388         Aufgabe ~ Nr. ~ \g_auf_examen_aufgabe_nr_tl
389       }
390       \par
391       \bigskip
392     }
393   }
394 }

395 \cs_new:Npn \_gib_github_url: {

```

```

396 \tl_if_empty:NTF \g_auf_relativer_pfad_tl {} {
397   Github :~\href{
398     \LehramtInformatikGithubRawDomain /
399     \LehramtInformatikGithubTexRepo /
400     \LehramtInformatikGitBranch /
401     \g_auf_relativer_pfad_tl
402   }{
403     \nolinkurl{\g_auf_relativer_pfad_tl}
404   }
405 }
406 }

407 \cs_new:Npn \_gib_aufgaben_titel: {
408   \g_auf_titel_tl
409
410   \tl_if_empty:NTF \g_auf_thematik_tl
411   {}
412   {
413     \, ~ [
414       \g_auf_thematik_tl
415     ]
416   }
417 }

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

\def\LehramtInformatikRepository{/pfad/zum/repository}
420 \input /etc/lehramt-informatik.config.tex
      biblatex not working with lualatex and babel
421 % \RequirePackage{polyglossia}
422 % \setmainlanguage{german}
423

```

2.6 baum.sty

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

427 \RequirePackage{tikz}

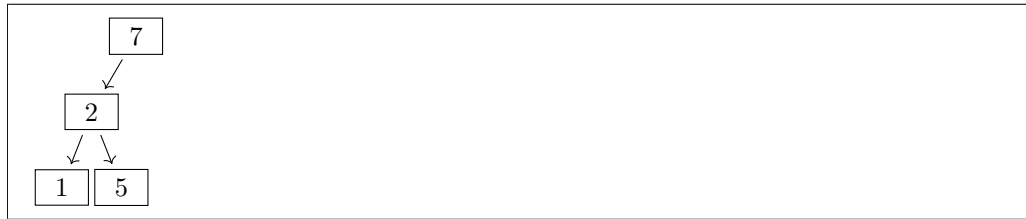
    für li binaer baum
428 \RequirePackage{tikz-qtree}

    Für b baum
429 \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}
```



```
430 \tikzset{
431   li binaer baum/.style={
432     shorten <=2pt,
433     shorten >=2pt,
434     ->,
435     every tree node/.style={
436       minimum width=2em,
437       draw,
438       rectangle
439     },
440     blank/.style={
441       draw=none
442     },
443     edge from parent/.style={
444       draw,
445       edge from parent path={(\tikzparentnode) -- (\tikzchildnode)}
446     },
447     level distance=1cm,
448     every label/.style={
449       gray,
450       font=\footnotesize,
451       label position=0,
452       label distance=0cm,
453     }
454   },
455 }
```

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}
```

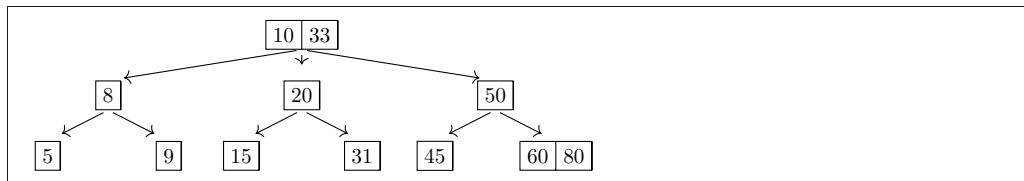


2.6.3 B-Baum

```

\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}

```



```

456 \tikzset{
457   li bbaum knoten/.style={
458     rectangle split parts=10,
459     rectangle split,
460     rectangle split horizontal,
461     rectangle split ignore empty parts,
462     draw,
463     fill=white
464   },
465   li bbaum/.style={
466     every node/.style={
467       li bbaum knoten
468     },
469     level 1/.style={
470       level distance=12mm,
471       sibling distance=25mm,
472     },
473     every child/.style={
474       shorten <= 2pt,
475       shorten >= 6pt,
476       ->,
477     },
478     level 2/.style={
479       level distance=9mm,
480       sibling distance=15mm,
481     },
482   }
483 }
484

```

2.7 checkbox.sty

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

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

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

491
```

2.8 chomsky-normalform.sty

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

495 \ExplSyntaxOn

496 \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

```

497 \def\liChomskyUeberschrift#1{
498 {
499   \bfseries
500   \sffamily
501   \str_case:nn {#1} {
502     {1} {Elimination-der- $\varepsilon$ -Regeln}
503     {2} {Elimination-von-Kettenregeln}
504     {3} {Separation-von-Terminalzeichen}
505     {4} {Elimination-von-mehrelementigen-Nonterminalketten}
506   }
507 }
508 }

```

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

```

509 \def\li@chomsky@erklaerung@texte#1{
510   \str_case:nn {#1} {
511     %
512     {1} {
513       Alle-Regeln-der-Form- $A \rightarrow \varepsilon$ -werden-eliminiert.~
514       Die-Ersetzung-von- $A$ -wird-durch- $\varepsilon$ -in-allen-anderen-
515       Regeln-vorweggenommen.
516     }
517     {2} {
518       Jede-Produktion-der-Form- $A \rightarrow B$ -mit- $A, B$ -in- $S$ -wird-
519       als-Kettenregel-bezeichnet.~Diese-tragen-nicht-zur-Produktion-
520       von-Terminalzeichen-bei-und-lassen-sich-ebenfalls-eliminieren.
521     }
522     {3} {
523       Jedes-Terminalzeichen- $\sigma$ ,~das-in-Kombination-mit-anderen-
524       Symbolen-auftaucht,~wird-durch-ein-neues-Nonterminal-
525        $S_{\sigma}$ -ersetzt-und-die-Menge-der-Produktionen-durch-die-
526       Regel- $S_{\sigma} \rightarrow \sigma$ -ergänzt.
527     }
528     {4} {
529       Alle-Produktionen-der-Form-
530        $A \rightarrow B_{\{1\}} B_{\{2\}} \dots B_{\{n\}}$ -
531       werden-in-die-Produktionen-
532        $A \rightarrow$ 
533        $A_{\{n-1\}} B_{\{n\}}, A_{\{n-1\}} \rightarrow$ 
534        $A_{\{n-2\}} B_{\{n-1\}}, \dots,$ 
535        $A_{\{2\}} \rightarrow B_{\{1\}} B_{\{2\}}$ -zerteilt.~
536       Nach-der-Ersetzung-sind-alle-längeren-Nonterminalketten-
537       vollständig-heruntergebrochen-und-die-Chomsky-Normalform-erreicht.
538     }
539   }

```

```

540 }
541 \def\liChomskyErklaerung#1{
542   {
543     \itshape
544     \footnotesize
545     \liParagraphMitLinien{\li@chomsky@erklaerung@texte{#1}}
546   }
547 }

```

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

```

548 \def\liChomskyUeberErklaerung#1{
549   \liChomskyUeberschrift{#1}\par
550   \liChomskyErklaerung{#1}
551 }

```

```

552 \ExplSyntaxOff
553

```

2.9 cpm.sty

```

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

557 \liLadePakete{mathe,typographie}

```

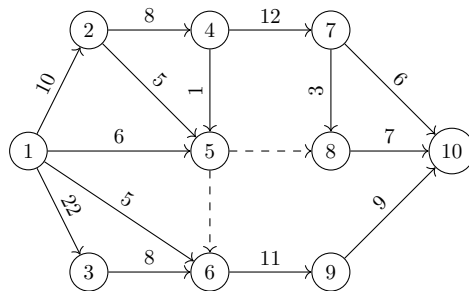
2.9.1 Makro-Kürzel

```

\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}{3}
\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
\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}

558 \ExplSyntaxOn
559 \NewDocumentCommand { \liCpmEreignis } { 0{} m m m } {
560   \tl_set:Nn \l_name_tl {}
561
562   \keys_define:nn { cpmEreignis } {
563     name .code:n = {\tl_set:Nn \l_name_tl {##1}},
564   }
565
566   \keys_set:nn { cpmEreignis } { #1 }
567
568   \tl_if_empty:NT \l_name_tl {
569     \tl_set:Nn \l_name_tl {#2}
570   }
571
572   \node[circle,draw] (\l_name_tl) at (#3,#4) {#2};
573 }
574 \ExplSyntaxOff

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

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

```

2.9.6 Hilfsmakros für Zeitpunkt-Tabelle

```

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

\liCpmVonZu Let-Abkürzung: \let\vz=\liCpmVonZu
\liCpmVonZu{1}(2-3):  $1_{(2 \rightarrow 3)}$ 
590 \def\liCpmVonZuOhneMathe#1(#2-#3){#1_{\scriptscriptstyle(#2\rightarrow#3)}}
591 \def\liCpmVonZu#1(#2-#3){%
592 \ifmmode%
593 \liCpmVonZuOhneMathe{#1}(\#2-\#3)%
594 \else%
595 $\liCpmVonZuOhneMathe{#1}(\#2-\#3)$%
596 \fi%
597 }

\liCpmVon Let-Abkürzung: \let\v=\liCpmVon
\liCpmVon{1}(2):  $1_{(\rightarrow 2)}$ 
598 \def\liCpmVonOhneMathe#1(#2){#1_{\scriptscriptstyle(\rightarrow#2)}}
599 \def\liCpmVon#1(#2){%
600 \ifmmode%
601 \liCpmVonOhneMathe{#1}(\#2)%
602 \else%
603 $\liCpmVonOhneMathe{#1}(\#2)$%
604 \fi%
605 }

\liCpmZu Let-Abkürzung: \let\z=\liCpmZu
\liCpmZu{1}(2):  $1_{(\leftarrow 2)}$ 
606 \def\liCpmZuOhneMathe#1(#2){#1_{\scriptscriptstyle(\leftarrow#2)}}
607 \def\liCpmZu#1(#2){%
608 \ifmmode%
609 \liCpmZuOhneMathe{#1}(\#2)%
610 \else%
611 $\liCpmZuOhneMathe{#1}(\#2)$%
612 \fi%
613 }

614 \ExplSyntaxOn

\liCpmSpaetI Spätester Zeitpunkt, zu dem Ereignis i eintreten kann
Let-Abkürzung: \let\SZ=\liCpmSpaetI
615 \NewDocumentCommand{ \liCpmSpaetI } { 0{i} } {
616 \ifmmode
617 SZ\sb{#1}
618 \else
619 $SZ\sb{#1}$
620 \fi
621 }

\liCpmFruehI Frühester Zeitpunkt, zu dem Ereignis i eintreten kann.
Let-Abkürzung: \let\FZ=\liCpmFruehI
622 \NewDocumentCommand{ \liCpmFruehI } { 0{i} } {
623 \ifmmode
624 FZ\sb{#1}
625 \else
626 $FZ\sb{#1}$
627 \fi
628 }

```


`\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. _____

```

629 \def\liCpmFruehErklaerung{
630   \liParagraphMitLinien{
631     Wir~führen~eine~Vorwärtsterminierung~durch~
632     und~addieren~die~Dauern.~
633
634     Kann~ein~Ereignis~über~mehrere~Vorgänge~erreicht~
635     werden,~wählen~wir~das~Maximum~aus.~
636
637     \textbf{Erläuterungen:}~
638
639     $i$:~
640     Ereignis~$i$;~\,
641
642     \liCpmFruehI{}:~
643     Frühester~Zeitpunkt,~zu~dem~Ereignis~$i$~eintreten~kann
644     .
645   }
646 }
```

`\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. _____

```

647 \def\liCpmSpaetErklaerung{
648   \liParagraphMitLinien{
649     Wir~führen~eine~Rückwärtsterminierung~durch~
650     und~subtrahieren~die~Dauern~vom~letzten~Ereignis~aus.~
651
652     Kann~ein~Ereignis~über~mehrere~Vorgänge~erreicht~
653     werden,~wählen~wir~das~Minimum~aus.~
654
655     \textbf{Erläuterungen:}~
656
657     $i$:~
658     Ereignis~$i$;~\,
659
660     \liCpmSpaetI{}:~
661     Spätester~Zeitpunkt,~zu~dem~Ereignis~$i$~eintreten~kann
662     .
663   }
664 }
```

```

665 \ExplSyntaxOff
666
```

2.10 cyk-algorithmus.sty

```
667 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
668 \ProvidesPackage{lehramt-informatik-cyk-algorithmus}[2021/06/18 Hilfsmakros
669 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`

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

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

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

```
671 \NewDocumentCommand{ \liWortInSprache } { m O{L(G)} } {
672   \bigskip
673   \noindent
674   $\Rrightarrow #1 \in #2$
675 }
```

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

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

```
676 \NewDocumentCommand{ \liWortNichtInSprache } { m O{L(G)} } {
677   \bigskip
678   \noindent
679   $\Rrightarrow #1 \notin #2$
680 }
```

```
681
```

2.11 entwurfsmuster.sty

```
682 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
683 \ProvidesPackage{lehramt-informatik-entwurfsmuster}[2021/05/06
684 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

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

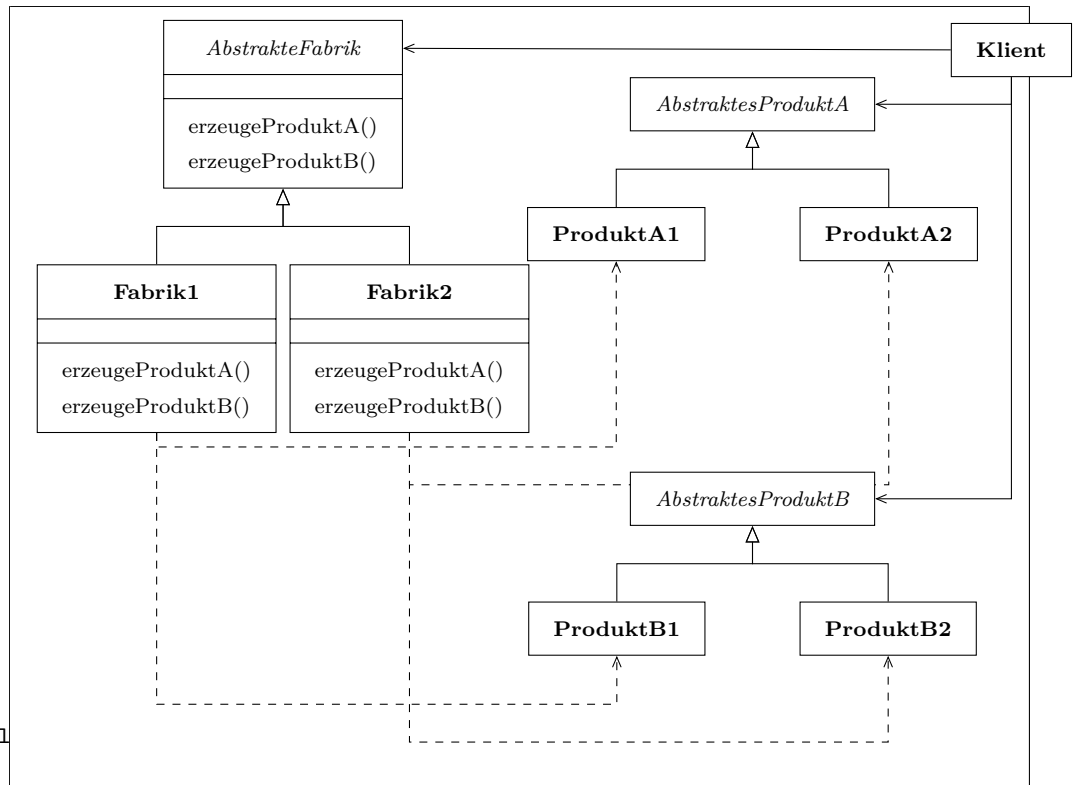
\li@EntwurfsCodeAllgemein Allgemeine Code-Beispiele zu den UML-Diagrammen und Stellvertretern

```
686 \RequirePackage{lehramt-informatik-syntax}
687 \def\li@EntwurfsCode#1#2{
688   \liJavaDatei{entwurfsmuster/#1/allgemein/#2}
689 }
```

2.11.3 Abstrakte Fabrik (Abstract Factory)

sAbstrakteFabrikBeschreibung

```
690 \def\liEntwurfsAbstrakteFabrikBeschreibung{
691   Es wird eine Schnittstelle bereitgestellt, um \emph{Familien
692   verbundener oder abhängiger Objekte} zu erstellen, ohne die konkreten
693   Klassen zu spezifizieren.\footcite[Seite 25]{eilebrecht}
694 }
```



```

695 \def\liEntwurfsAbstrakteFabrikUml{
696   \begin{tikzpicture}
697     \umlclass[type=abstract]{AbstrakteFabrik}{-}{-}{
698       erzeugeProduktA()\n
699       erzeugeProduktB()\n
700     }
701     \umlclass[below left=1cm and -1.5cm of AbstrakteFabrik]{Fabrik1}{-}{-}{
702       erzeugeProduktA()\n
703       erzeugeProduktB()\n
704     }
705     \umlclass[below right=1cm and -1.5cm of AbstrakteFabrik]{Fabrik2}{-}{-}{
706       erzeugeProduktA()\n
707       erzeugeProduktB()\n
708     }
709     \umlVHVinherit{Fabrik1}{AbstrakteFabrik}
710     \umlVHVinherit{Fabrik2}{AbstrakteFabrik}
711
712     \umlsimpleclass[right=3cm of AbstrakteFabrik,type=abstract]{AbstraktesProduktA}
713     \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktA]{ProduktA1}
714     \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktA]{ProduktA2}
715     \umlVHVinherit{ProduktA1}{AbstraktesProduktA}
716     \umlVHVinherit{ProduktA2}{AbstraktesProduktA}
717
718     \umlsimpleclass[above right=0cm and 1cm of AbstraktesProduktA]{Klient}
719
720     \umlsimpleclass[below=4.5cm of AbstraktesProduktA,type=abstract]{AbstraktesProduktB}
721     \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktB]{ProduktB1}
722     \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktB]{ProduktB2}
723     \umlVHVinherit{ProduktB1}{AbstraktesProduktB}
724     \umlVHVinherit{ProduktB2}{AbstraktesProduktB}
725
726     \umlVHVdep[arm1=-1.3cm,arm2=-1cm]{Fabrik1}{ProduktA1}
727     \umlVHVdep[arm2=-1cm]{Fabrik1}{ProduktB1}
728
729     \umlVHVdep[arm1=-1.8cm,arm2=-1.5cm]{Fabrik2}{ProduktA2}
730     \umlVHVdep[arm2=-1.5cm]{Fabrik2}{ProduktB2}
731

```

```

732 \umluniassoc[anchor2=25]{Klient}{AbstrakteFabrik}
733 \umlVHuniassoc[arm1=-1cm]{Klient}{AbstraktesProduktA}
734 \umlVHuniassoc{Klient}{AbstraktesProduktB}
735 \end{tikzpicture}
736 }

```

iEntwurfsAbstrakteFabrikCode

```

737 \def\liEntwurfsAbstrakteFabrikCode{
738 \li@EntwurfsCode{abstrakte_fabrik}{Produkte}
739 \li@EntwurfsCode{abstrakte_fabrik}{AbstrakteFabrik}
740 \li@EntwurfsCode{abstrakte_fabrik}{Klient}
741 }

```

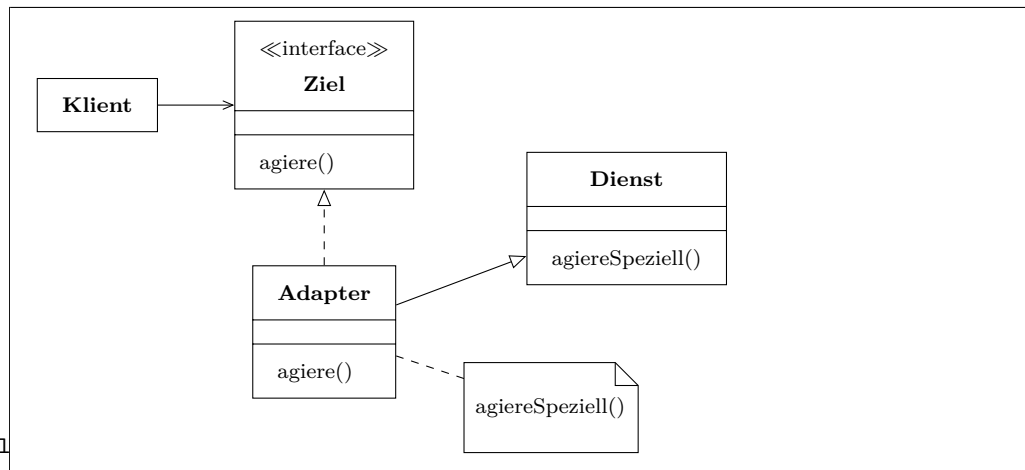
\liEntwurfsAbstrakteFabrik

```

742 \def\liEntwurfsAbstrakteFabrik{
743 \liEntwurfsAbstrakteFabrikBeschreibung
744
745 \liEntwurfsAbstrakteFabrikUml
746
747 \liEntwurfsAbstrakteFabrikCode
748 }

```

2.11.4 Adapter



\liEntwurfsAdapterUml

```

749 \def\liEntwurfsAdapterUml{
750 \begin{tikzpicture}
751 \umlsimpleclass[x=1,y=3]{Klient}{}{}
752 \umlclass[x=4,y=3,type=interface]{Ziel}{}{agiere()}
753 \umlclass[x=4,y=0]{Adapter}{}{agiere()}
754 \umlclass[x=8,y=1.5]{Dienst}{}{agiereSpeziell()}
755
756 \umlreal{Adapter}{Ziel}
757 \umluniassoc{Klient}{Ziel}
758 \umlinherit{Adapter}{Dienst}
759
760 \umlnote[x=7,y=-1,width=2cm]{Adapter}{agiereSpeziell()}
761 \end{tikzpicture}
762 \footcite[so ähnlich wie GoF]{wiki:adapter}
763 }

```

\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.

```

764 \def\liEntwurfsAdapterAkteure{
765   \begin{description}
766
767     \item[Ziel (Target)]
768
769     Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.
770
771     \item[Klient (Client)]
772
773     Der Klient nutzt Dienste über inkompatible Schnittstellen und greift
774     dabei auf adaptierte Schnittstellen zurück.
775
776     \item[Dienst (Adaptee)]
777
778     Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest
779     definierter Schnittstelle an.
780
781     \item[Adapter]
782
783     Der Adapter adaptiert die Schnittstelle des Dienstes auf die
784     Schnittstelle zum Klienten.\footcite{wiki:adapter}
785
786   \end{description}
787 }

```

\liEntwurfsAdapterCode

```

788 \def\liEntwurfsAdapterCode{
789   \li@EntwurfsCode{adapter}{Dienst}
790   \li@EntwurfsCode{adapter}{Ziel}
791   \li@EntwurfsCode{adapter}{Adapter}
792   \li@EntwurfsCode{adapter}{Klient}
793 }

```

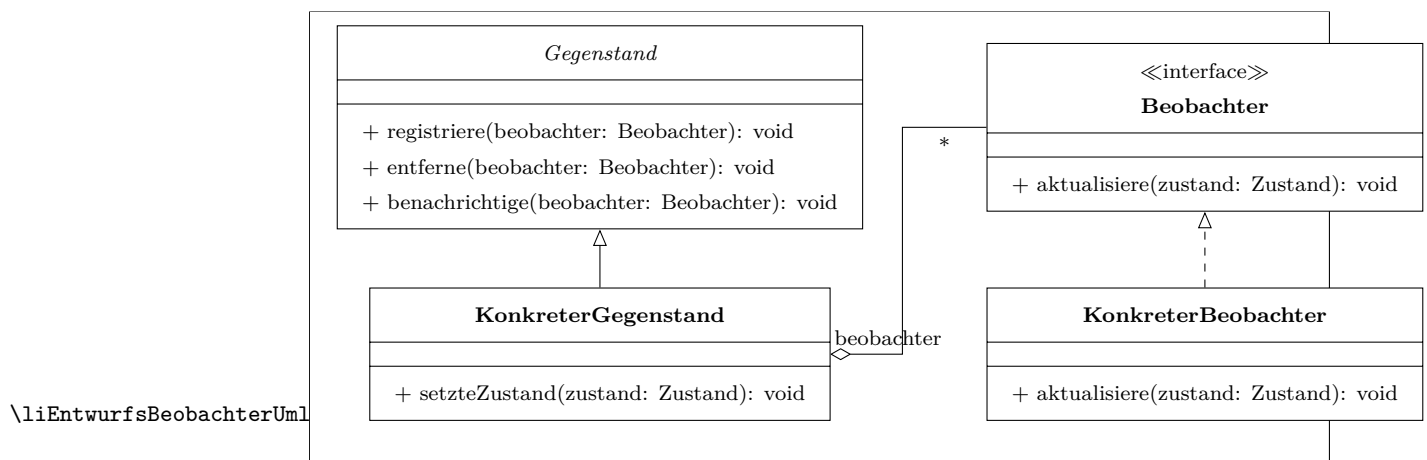
\liEntwurfsAdapter

```

794 \def\liEntwurfsAdapter{
795   \liEntwurfsAdapterUml
796   \liEntwurfsAdapterAkteure
797   \liEntwurfsAdapterCode
798 }

```

2.11.5 Beobachter (Observer)



```

799 \def\liEntwurfsBeobachterUml{
800   \begin{tikzpicture}
801     \umlclass[x=0,y=0,type=abstract]{Gegenstand}{}{

```

```

802     + registriere(beobachter: Beobachter): void\\
803     + entferne(beobachter: Beobachter): void\\
804     + benachrichtige(beobachter: Beobachter): void\\
805 }
806 \umlclass[x=0,y=-3]{KonkreterGegenstand}{-}{
807     + setzeZustand(zustand: Zustand): void
808 }
809 \umlinherit{KonkreterGegenstand}{Gegenstand}
810
811 \umlclass[x=8,y=0,type=interface]{Beobachter}{-}{
812     + aktualisiere(zustand: Zustand): void
813 }
814 \umlclass[x=8,y=-3]{KonkreterBeobachter}{-}{
815     + aktualisiere(zustand: Zustand): void
816 }
817 \umlreal{KonkreterBeobachter}{Beobachter}
818
819 \umlHVHaggreg[arg1=beobachter,pos1=0.8,mult2=*,pos2=2.5]
820 {KonkreterGegenstand}{Beobachter}
821 \end{tikzpicture}
822 }

```

\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.

```

823 \def\liEntwurfsBeobachterAkteure{
824   \begin{description}
825     \item[Gegenstand / Subjekt (Subject / Observable)]
826
827     Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also
828     „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne
829     deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An-
830     und Abmeldung von Beobachtern und eine Schnittstelle zur
831     Benachrichtigung von Beobachtern über Änderungen an.\footcite[Seite
832     251]{gof}
833
834     \item[Beobachter (Observer)]
835
836     Die Beobachter (auf Englisch auch subscriber, also „Abonnent“,
837     genannt) definieren eine Aktualisierungsschnittstelle.
838
839     \item[konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)]
840
841     Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert
842     den relevanten Zustand und benachrichtigt alle Beobachter bei

```

```

843 Zustandsänderungen über deren Aktualisierungsschnittstelle. Es
844 verfügt über eine Schnittstelle zur Erfragung des aktuellen
845 Zustands.
846
847 \item[Konkrete Beobachter (ConcreteObserver)]
848
849 Konkrete Beobachter verwalten die Referenz auf ein konkretes
850 Subjekt, dessen Zustand sie beobachten und speichern und dessen
851 Zustand konsistent ist. Sie implementieren eine
852 Aktualisierungsschnittstelle unter Verwendung der
853 Abfrageschnittstelle des konkreten Subjekts.
854 \footcite{wiki:beobachter}
855 \end{description}
856 }

```

\liEntwurfsBeobachterCode

```

857 \def\liEntwurfsBeobachterCode{
858 \li@EntwurfsCode{beobachter}{Gegenstand}
859 \li@EntwurfsCode{beobachter}{KonkreterGegenstand}
860 \li@EntwurfsCode{beobachter}{Beobachter}
861 \li@EntwurfsCode{beobachter}{KonkreterBeobachterA}
862 \li@EntwurfsCode{beobachter}{KonkreterBeobachterB}
863 \li@EntwurfsCode{beobachter}{Klient}
864 }

```

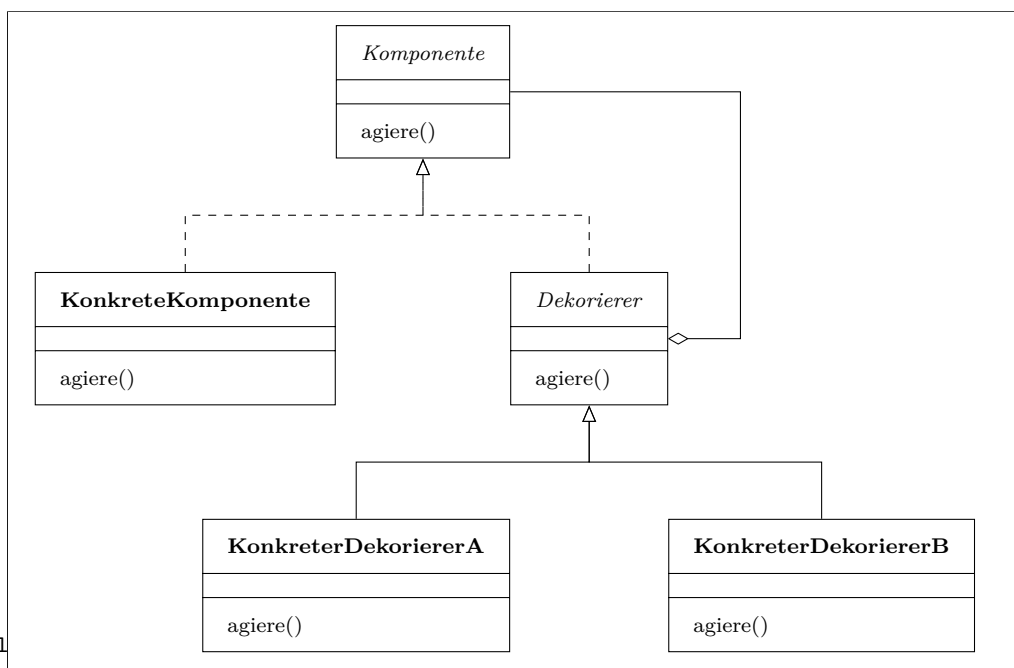
\liEntwurfsBeobachter

```

865 \def\liEntwurfsBeobachter{
866 \liEntwurfsBeobachterUml
867 \liEntwurfsBeobachterAkteure
868 \liEntwurfsBeobachterCode
869 }

```

2.11.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

870 \def\liEntwurfsDekoriererUml{
871 \begin{tikzpicture}
872 \umlclass[type=abstract]{Komponente}{}{agiere()}
873 \umlclass[below left=1.5cm and 0cm of Komponente]{KonkreteKomponente}{}{agiere()}
874 \umlclass[below right=1.5cm and 0cm of Komponente,type=abstract]{Dekorierer}{}{agiere()}
875

```



```

876 \umlVHVreal{KonkreteKomponente}{Komponente}
877 \umlVHVreal{Dekorierer}{Komponente}
878
879 \umlclass[below left=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererA}{\{agiere()\}}
880 \umlclass[below right=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererB}{\{agiere()\}}
881
882 \umlVHVinherit{KonkreterDekoriererA}{Dekorierer}
883 \umlVHVinherit{KonkreterDekoriererB}{Dekorierer}
884
885 \umlHVHaggreg[arm1=2cm]{Dekorierer}{Komponente}
886 \footcite{wiki:dekorierer}
887 \end{tikzpicture}
888 }

```

\liEntwurfsDekoriererCode

```

889 \def\liEntwurfsDekoriererCode{
890 \li@EntwurfsCode{dekorierer}{Komponente}
891 \li@EntwurfsCode{dekorierer}{KonkreteKomponente}
892 \li@EntwurfsCode{dekorierer}{Dekorierer}
893 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererA}
894 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererB}
895 \li@EntwurfsCode{dekorierer}{Klient}
896 }

```

\liEntwurfsDekorierer

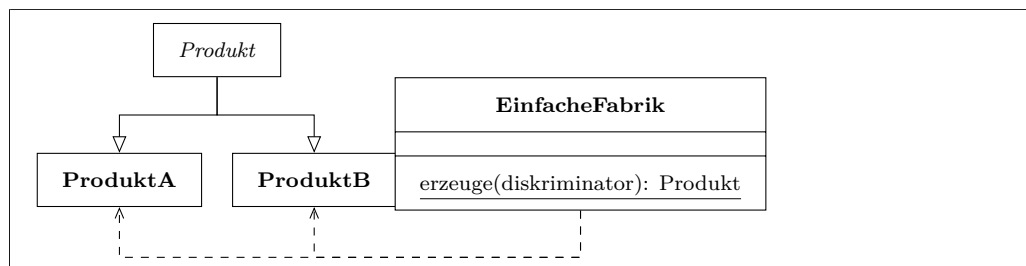
```

897 \def\liEntwurfsDekorierer{
898 \liEntwurfsDekoriererUml
899 \liEntwurfsDekoriererAkteure
900 \liEntwurfsDekoriererCode
901 }

```

2.11.7 Einfache Fabrik (Simple Factory)

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



```

902 \def\liEntwurfsEinfacheFabrikUml{
903 \begin{tikzpicture}
904 \umlsimpleclass[type=abstract]{Produkt}
905 \umlsimpleclass[below left=1cm and -0.65cm of Produkt]{ProduktA}
906 \umlsimpleclass[below right=1cm and -0.65cm of Produkt]{ProduktB}
907 \umlVHVinherit{Produkt}{ProduktA}
908 \umlVHVinherit{Produkt}{ProduktB}
909 \umlclass[below right=0cm and 1.5cm of Produkt]{EinfacheFabrik}{
910 }{
911 \umlstatic{erzeuge(diskriminator): Produkt}\
912 }
913 \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktA}
914 \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktB}
915 \end{tikzpicture}
916 }

```

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.

```

917 \def\liEntwurfsEinfacheFabrikAkteure{
918   \begin{description}
919     \item[EinfacheFabrik]
920
921     Eine Klasse mit einer Erzeugungsmethode, die über eine größere
922     Bedingung verschiedene Objekt instanziert.
923
924     \item[Produkt]
925
926     Eine abstrakte Klasse, die von den konkreten Produkten geerbt wird.
927
928     \item[KonkretesProdukt]
929
930     Ein konkretes Produkt, das von der einfachen Fabrik erzeugt wird.
931   \end{description}
932 }
```

\liEntwurfsEinfacheFabrik

```

933 \def\liEntwurfsEinfacheFabrik{
934   \liEntwurfsEinfacheFabrikUml
935   \liEntwurfsEinfacheFabrikAkteure
936 }
```

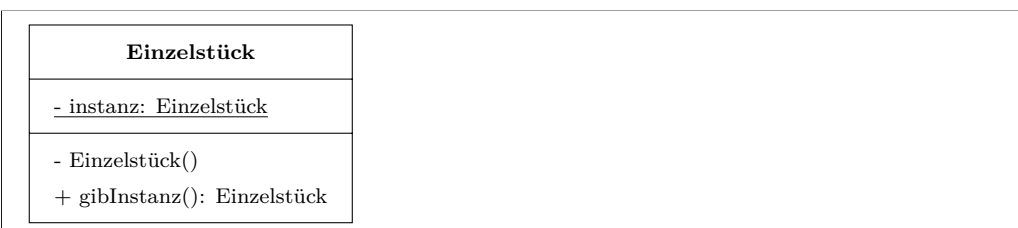
2.11.8 Einzelstück (Singleton)

urfsEinzelstueckBeschreibung

```

937 \def\liEntwurfsEinzelstueckBeschreibung{
938   Stellt sicher, dass nur \emph{genau eine Instanz einer Klasse} erzeugt
939   wird.\footcite[Seite 38]{eilebrecht}
940 }
941
```

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```

942 \def\liEntwurfsEinzelstueckUml{
943   \begin{tikzpicture}
944     \umlclass{Einzelstück}{
945       \umlstatic{- instanz: Einzelstück}\\
946     }{
947       - Einzelstück()\\
948       + gibInstanz(): Einzelstück
949     }
950   \end{tikzpicture}
951 }
```

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.

```

952 \def\liEntwurfsEinzelstueckAkteure{
953   \begin{description}
954     \item[Einzelstück (Singleton)]
955
956     stellt eine statische Methode bereit, mit deren Hilfe die Klienten
957     nur auf eine einzige Instanz der Klasse zugreifen können.
958   \end{description}
959 }

```

\liEntwurfsEinzelstueckCode

```

960 \def\liEntwurfsEinzelstueckCode{
961   \li@EntwurfsCode{einzelstueck}{Einzelstueck}
962 }

```

\liEntwurfsEinzelstueck

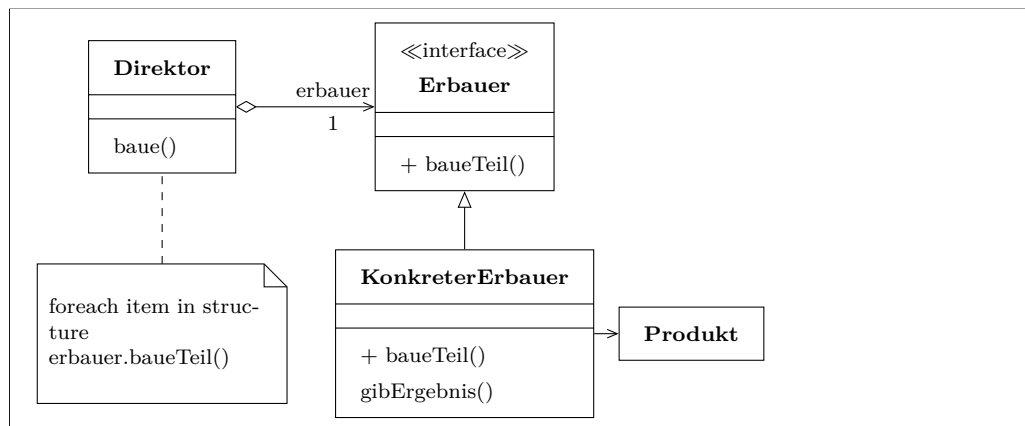
```

963 \def\liEntwurfsEinzelstueck{
964   \liEntwurfsEinzelstueckBeschreibung
965
966   \liEntwurfsEinzelstueckUml
967
968   \liEntwurfsEinzelstueckAkteure
969
970   \liEntwurfsEinzelstueckCode
971 }

```

2.11.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

972 \def\liEntwurfsErbauerUml{
973   \begin{tikzpicture}
974     \umlsimpleclass[x=7,y=0]{Produkt}{}{}
975     \umlclass[x=4,y=3,type=interface]{Erbauer}{}{+ baueTeil()}
976     \umlclass[x=4,y=0]{KonkreterErbauer}{}{
977       + baueTeil()\n
978       gibErgebnis()}
979     \umlclass[x=0,y=3]{Direktor}{}{baue()}
980
981     \umluniagg[arg2=erbauer,pos2=0.7,mult2=1]{Direktor}{Erbauer}
982     \umluniassoc{KonkreterErbauer}{Produkt}
983     \umlinherit{KonkreterErbauer}{Erbauer}
984
985     \umlnote[x=0,y=0,width=3cm]{Direktor}{
986       foreach item in structure\n
987       erbauer.baueTeil()

```

```

988 }
989 \end{tikzpicture}
990 \footcite{wiki:erbauer}
991 }

```

\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.

```

992 \def\liEntwurfsErbauerAkteure{
993   \begin{description}
994     \item[Erbauer]
995
996     Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der
997     Teile eines komplexen Objektes.
998
999     \item[KonkreterErbauer]
1000
1001     Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch
1002     Implementierung der Schnittstelle. Außerdem definiert und verwaltet er
1003     die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine
1004     Schnittstelle zum Auslesen des Produkts.
1005
1006     \item[Direktor]
1007
1008     Der Direktor konstruiert ein komplexes Objekt unter Verwendung der
1009     Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer
1010     zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder
1011     benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom
1012     Klienten.
1013
1014     \item[Produkt]
1015
1016     Das Produkt repräsentiert das zu konstruierende komplexe Objekt.
1017     \footcite{wiki:erbauer}
1018   \end{description}
1019 }

```

\liEntwurfsErbauer

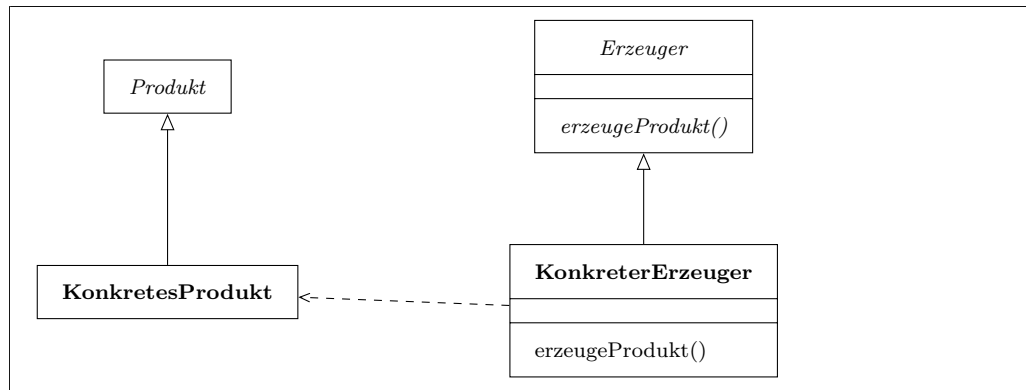
```

1020 \def\liEntwurfsErbauer{
1021   \liEntwurfsErbauerUml
1022   \liEntwurfsErbauerAkteure
1023 }

```

2.11.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

1024 \def\liEntwurfsFabrikmethodeUml{
1025   \begin{tikzpicture}
1026     \umlsimpleclass[type=abstract]{Produkt}
1027     \umlsimpleclass[below=2cm of Produkt]{KonkretesProdukt}
1028     \umlinherit{KonkretesProdukt}{Produkt}
1029
1030     \umlclass[type=abstract,right=4cm of Produkt]{Erzeuger}{\{
1031       \textit{erzeugeProdukt()}\}
1032   }
1033   \umlclass[below=1.2cm of Erzeuger]{KonkreterErzeuger}{\{
1034     erzeugeProdukt()
1035   }
1036   \umlinherit{KonkreterErzeuger}{Erzeuger}
1037
1038   \umldep{KonkreterErzeuger}{KonkretesProdukt}
1039 \end{tikzpicture}
1040 }

```

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).

```

1041 \def\liEntwurfsFabrikmethodeAkteure{
1042   \begin{description}
1043     \item[Produkt]
1044
1045     Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das
1046     zu erzeugende Produkt.
1047
1048     \item[KonkretesProdukt]
1049
1050     KonkretesProdukt implementiert die Produkt-Schnittstelle.
1051
1052     \item[Erzeuger]
1053
1054     Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt
1055     zu erzeugen und kann eine Default-Implementierung beinhalten.
1056
1057     \item[KonkreterErzeuger]
1058

```

```

1059     KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm
1060     entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den
1061     Konstruktor einer konkreten Produkt-Klasse aufruft).
1062
1063     \footcite{wiki:fabrikmethode}
1064 \end{description}
1065 }

```

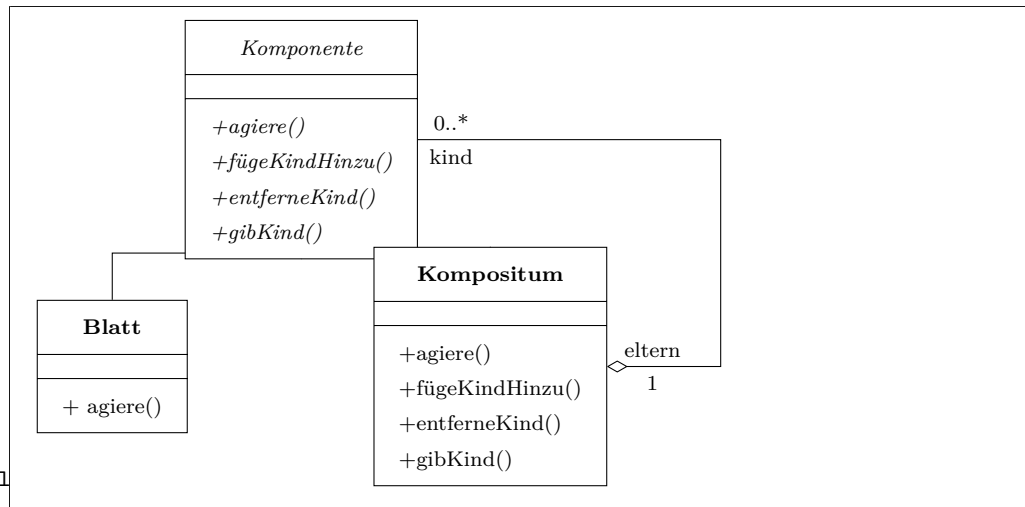
\liEntwurfsFabrikmethode

```

1066 \def\liEntwurfsFabrikmethode{
1067   \liEntwurfsFabrikmethodeUml
1068   \liEntwurfsFabrikmethodeAkteure
1069 }

```

2.11.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

1070 \def\liEntwurfsKompositumUml{
1071   \begin{tikzpicture}
1072     \umlclass[x=2.5,y=3,type=abstract]{Komponente}{-}{
1073       \textit{+agiere()}\
1074       \textit{+fügeKindHinzu()}\
1075       \textit{+entferneKind()}\
1076       \textit{+gibKind()}
1077     }
1078     \umlclass[x=0]{Blatt}{-}{+ agiere()}
1079     \umlclass[x=5]{Kompositum}{-}{
1080       +agiere()\
1081       +fügeKindHinzu()\
1082       +entferneKind()\
1083       +gibKind()
1084     }
1085
1086     \umlVHVinherit{Kompositum}{Komponente}
1087     \umlVHVinherit{Blatt}{Komponente}
1088     \umlHVVHaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,]
1089   \end{tikzpicture}
1090 }

```

\liEntwurfsFabrikmethode

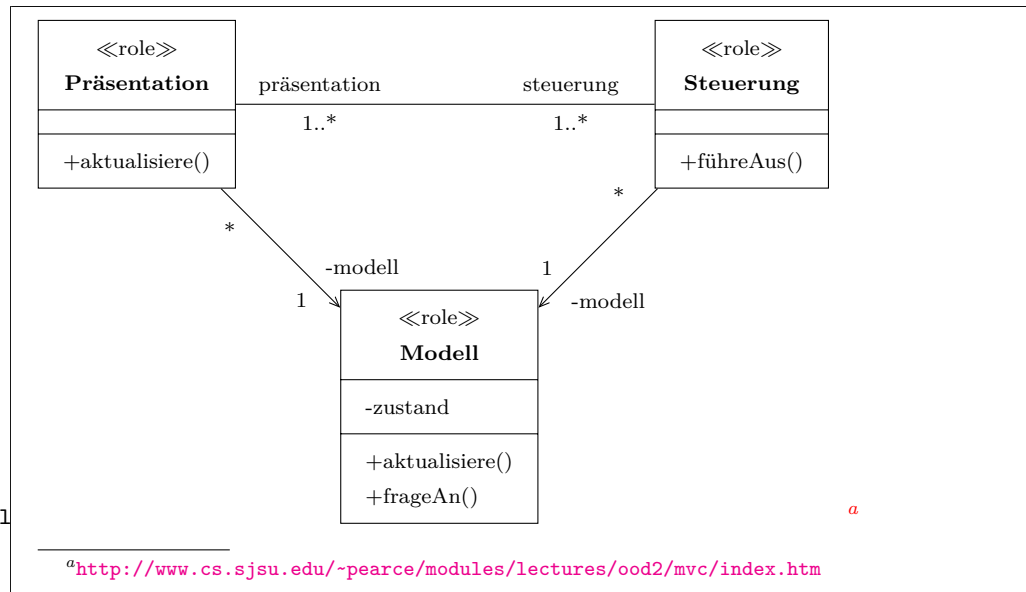
```

1091 \def\liEntwurfsKompositum{
1092   \liEntwurfsKompositumUml
1093   \liEntwurfsKompositumAkteure
1094 }

```

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

ModellPraesentationSteuerungUml



```

1095 \def\liEntwurfsModellPraesentationSteuerungUml{
1096   \begin{tikzpicture}
1097     \umlclass[x=-4cm,type=role]{Präsentation}{+aktualisiere()}
1098     \umlclass[x=4cm,type=role]{Steuerung}{+führeAus()}
1099     \umlclass[y=-4cm,type=role]{Modell}{
1100       -zustand
1101     }{
1102       +aktualisiere()\
1103       +frageAn()
1104     }
1105
1106     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
1107     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}
1108     \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
1109   \end{tikzpicture}
1110   \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
1111 }

```

ModellPraesentationSteuerung

```

1112 \def\liEntwurfs{
1113   \liEntwurfsModellPraesentationSteuerungUml
1114   \liEntwurfsModellPraesentationSteuerungAkteure
1115 }

```

2.11.13 Stellvertreter (Proxy)

\liEntwurfsStellvertreterUml

```

1116 \def\liEntwurfsStellvertreterUml{
1117   \begin{tikzpicture}
1118     \umlsimpleclass[x=-1,y=2]{Klient}
1119
1120     \umlclass[x=2,y=2]{Subjekt}{+ agiere()}
1121     \umlclass[x=0,y=-1]{KonkretesSubjekt}{+ agiere()}
1122     \umlclass[x=4,y=-1]{Stellvertreter}{+ agiere()}
1123
1124     \umlVHVinherit{KonkretesSubjekt}{Subjekt}
1125     \umlVHVinherit{Stellvertreter}{Subjekt}
1126     \umluniassoc{Stellvertreter}{KonkretesSubjekt}
1127     \umluniassoc{Klient}{Subjekt}
1128   \end{tikzpicture}
1129 }

```

\liEntwurfsStellvertreterCode

```

1130 \def\liEntwurfsStellvertreterCode{
1131   \li@EntwurfsCode{stellvertreter}{Subjekt}
1132   \li@EntwurfsCode{stellvertreter}{KonkretesSubjekt}
1133   \li@EntwurfsCode{stellvertreter}{Stellvertreter}
1134   \li@EntwurfsCode{stellvertreter}{Klient}
1135 }

```

\liEntwurfsStellvertreter

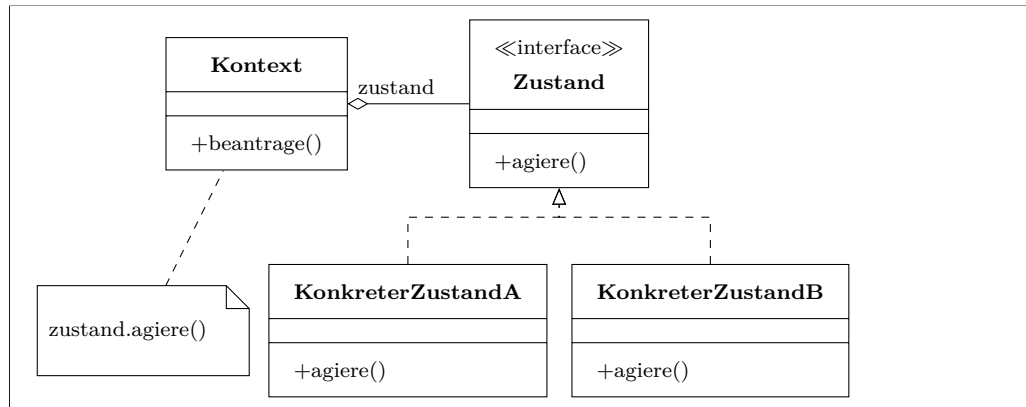
```

1136 \def\liEntwurfsStellvertreter{
1137   \liEntwurfsStellvertreterUml
1138   \liEntwurfsStellvertreterCode
1139 }

```

2.11.14 Zustand (State)

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



```

1140 \def\liEntwurfsZustandUml{
1141   \begin{tikzpicture}
1142     \umlclass[x=-1,y=3]{Kontext}{+beantrage()}
1143     \umlclass[x=3,y=3,type=interface]{Zustand}{+agiere()}
1144     \umlclass[x=1,y=0]{KonkreterZustandA}{+agiere()}
1145     \umlclass[x=5,y=0]{KonkreterZustandB}{+agiere()}
1146
1147     \umlVHVreal{KonkreterZustandA}{Zustand}
1148     \umlVHVreal{KonkreterZustandB}{Zustand}
1149
1150     \umlagg[arg=zustand,pos=0.4]{Kontext}{Zustand}
1151
1152     \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
1153   \end{tikzpicture}
1154 }

```

\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.

```

1155 \def\liEntwurfsZustandAkteure{
1156   \begin{description}
1157     \item[Kontext (Context)]
1158
1159     definiert die clientseitige Schnittstelle und verwaltet die separaten

```



```

1160     Zustandsklassen.
1161
1162     \item[State (Zustand)]
1163
1164     definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
1165     implementiert gegebenenfalls ein Standardverhalten.
1166
1167     \item[KontreterZustand (ConcreteState)]
1168
1169     implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
1170     verbunden ist.
1171 \end{description}
1172 }

\liEntwurfsZustandCode

1173 \def\liEntwurfsZustandCode{
1174     \li@EntwurfsCode{zustand}{Kontext}
1175     \li@EntwurfsCode{zustand}{Zustand}
1176 }

\liEntwurfsZustand

1177 \def\liEntwurfsZustand{
1178     \liEntwurfsZustandUml
1179     \liEntwurfsZustandAkteure
1180     \liEntwurfsZustandCode
1181 }

1182

```

2.12 er.sty

```
1183 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1184 \ProvidesPackage{lehramt-informatik-er}[2020/06/13 Zum Setzen von
1185 ER-Diagrammen]
```

```
1186 \RequirePackage{tikz-er2}
1187 \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);

```

```
1188 \RequirePackage{soul}
```

```
1189 \RequirePackage{fontawesome}
```

2.12.3 Makro-Kürzel

```

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

```

```
1190 \ExplSyntaxOn
```

```
\liErEntity
```

```
1191 \def\liErEntity#1{\textbf{#1}}
```

```
\liErRelationship
```

```
1192 \def\liErRelationship#1{\ul{#1}}
```

```
\liErAttribute
```

```
1193 \def\liErAttribute#1{\emph{#1}}
```

```
\liErMpEntity
```

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

```

1194 \def\liErMpEntity#1{
1195   \liErEntity{#1}
1196   \marginpar{
1197     \liErEntity{\tiny\faSquareO{~E:~#1}
1198   }
1199 }

```

□

```
\liErMpRelationship
```

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

```

1200 \def\liErMpRelationship#1{
1201   \liErRelationship{#1}
1202   \marginpar{
1203     \liErRelationship{\tiny\faGg{~R:~#1}
1204   }
1205 }

```

```
\liErMpAttribute
```

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

```

1206 \def\liErMpAttribute#1{
1207   \liErAttribute{#1}
1208   \marginpar{
1209     \liErAttribute{\tiny\faCircleThin{~A:~#1}
1210   }
1211 }

```

```

\liErDatenbankName Let-Abkürzung: \let\d=\liErDatenbankName
                    datenbank name
1212 \def\liErDatenbankName#1{
1213   {
1214     \footnotesize\texttt{(#1)}
1215   }
1216 }

1217 \ExplSyntaxOff
1218

```

2.13 formale-sprachen.sty

```

1219 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1220 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
1221 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]

1222 \directlua{
1223   formale_sprachen = require('lehramt-informatik-formale-sprachen')
1224 }

1225 \RequirePackage{hyperref}

1226 \liLadePakete{mathe,typographie}

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

1227 \def\liMengeOhneMathe#1{\{ #1 \}}
1228 \def\liMenge#1{%
1229   \ifmmode%
1230     \liMengeOhneMathe{#1}%
1231   \else%
1232     $\liMengeOhneMathe{#1}$%
1233   \fi%
1234 }

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

1235 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

1236 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1237 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1238 \def\liPotenzmenge#1{$\liPotenzmengeOhneMathe{#1}$}

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

1239 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
1240 \let\liZustandsmenge=\liPotenzmenge

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

1241 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1242 \def\liUeberfuehrungsFunktion#1{
1243   \ifmmode
1244     \liUeberfuehrungsFunktionOhneMathe{#1}
1245   \else
1246     $\liUeberfuehrungsFunktionOhneMathe{#1}$
1247   \fi
1248 }

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

1249 \def\liAlphabet#1{$\Sigma = \{ #1 \}$}

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

1250 \def\liBandAlphabet#1{$\Gamma = \Sigma \cup \{ #1 \}$}

\liZustandsBuchstabe

1251 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1252 \def\liZustandsBuchstabeGross{Z}

```

```

\liZustandsmengeNr
1253 \def\zustandsnamens@liste#1#2{
1254 $
1255 \{
1256 \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1257 \}
1258 $
1259 }
1260 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

\liZustandsmengeNrGross
1261 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

\liZustandsname \liZustandsname{1}: $z_1$
1262 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

\liZustandsnameGross \liZustandsnameGross{1}: $Z_1$
1263 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

\liAbleitung \liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1264 \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}

1265 \NewDocumentEnvironment { liProduktionsRegeln }
1266 { 0{P} +b }
1267 {
1268 \liGeschweifteKlammern{#1}
1269 {
1270 \begin{align*}
1271 \directlua{formale_sprachen.produktions_regeln('#2')}
1272 \end{align*}
1273 }{-0.2cm}{-1.5cm}
1274 } {}

\liProduktionen \liProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
1275 \def\liProduktionen#1{
1276 \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
1277 }

\liZustandsnameTiefgestellt Automatisch tiefgestellte Nummerierung \z1
Let-Abkürzung: \let\z=\liZustandsnameTiefgestellt
1278 \def\liZustandsnameTiefgestellt#1{
1279 \ifmmode
1280 \liZustandsBuchstabe\sb{#1}
1281 \else
1282 $\liZustandsBuchstabe\sb{#1}$
1283 \fi
1284 }

1285 \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}

```

```

1286 \NewDocumentCommand{ \liAusdruck } { O{L} m m } {
1287   $
1288   \tl_if_empty:nTF {#1} {} {#1 =}
1289   \{
1290     \, #2 \,
1291     |
1292     \, #3 \,
1293   \}$
1294 }
1295 \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

1296 \def\liFlaci#1{%
1297   \par
1298   {%
1299     \scriptsize
1300     Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1301     Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1302     Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1303     \href{https://flaci.com/#1}{flaci.com/#1}%
1304   }%
1305   \par
1306 }

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

1307 \ExplSyntaxOn
1308 \NewDocumentCommand {\liGrammatik} { O{G} m } {
1309   \tl_set:Nn \l_variablen_tl {V}
1310   \tl_set:Nn \l_alphabet_tl {\Sigma}
1311   \tl_set:Nn \l_produktionen_tl {P}
1312   \tl_set:Nn \l_start_tl {S}
1313
1314   \keys_define:nn { grammatik } {
1315     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1316     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1317     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1318     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1319   }
1320
1321   \keys_set:nn { grammatik } { #2 }
1322
1323   $#1 = (
1324     \l_variablen_tl,
1325     \l_alphabet_tl,
1326     \l_produktionen_tl,
1327     \l_start_tl
1328   )$
1329 }
1330 \ExplSyntaxOff
1331

```

2.14 formatierung.sty

```
1332 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1333 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

2.14.1 Schriftarten / Typographie

```
1334 \RequirePackage{mathpazo}
1335 \RequirePackage[no-math]{fontspec}
1336 \setmainfont{texgyrepagella}
1337 \setsansfont{QTAncientOlive}
1338 \RequirePackage{sectsty}
1339 \allsectionsfont{\fontspec{QTAncientOlive}}
```

2.14.2 Farben

```
1340 \RequirePackage{xcolor}
1341 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

2.14.3 Überschriften

```
1342 \RequirePackage{titlesec}
1343 \titleformat{\chapter}[display]{\sffamily\bfseries}{\Opt}{\LARGE}
1344 \titlespacing{\chapter}{0pt}{0pt}{*1}
1345 \titleformat{\paragraph}[hang]{\sffamily\normalsize\bfseries}{\theparagraph}{1em}{}
1346 \setcounter{secnumdepth}{0}
```

2.14.4 Listen

```
1347 \RequirePackage{paralist}
1348 \renewcommand\labelitemi{-}
1349 \renewcommand\labelitemii{-}
1350 \renewcommand\labelitemiii{-}
1351 \renewcommand\labelitemiv{-}
1352 % Counter: enumi enumii enumiii enumiv
1353 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1354 \renewcommand{\labelenumi}{(\alph{enumi})}
1355 \renewcommand{\labelenumii}{(\roman{enumii})}
```

2.14.5 Kasten

```
1356 \RequirePackage{mdframed}
1357 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1358 \NewDocumentEnvironment { liKasten }{ } {
1359   \begin{mdframed}[backgroundcolor=white!0]
1360 } {
1361   \end{mdframed}
1362 }
```

2.14.6 Header

```
1363 \RequirePackage{fancyhdr}
1364 \fancyhead[L,C,R]{}
1365 \fancyfoot[L]{}
1366 \fancyfoot[C]{}
1367 \fancyfoot[R]{\thepage}
1368 \pagestyle{fancy}
1369 \renewcommand{\headrulewidth}{0pt}
1370 \renewcommand{\footrulewidth}{0pt}
1371
```


2.15 gantt.sty

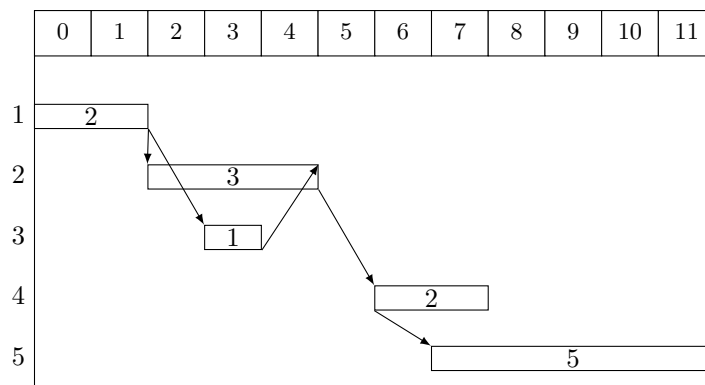
1372 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1373 \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}
```



```
1374 \RequirePackage{tikz-uml}
1375 \RequirePackage{pgfgantt}
1376 \setganttlinklabel{f-s}{}
1377 \setganttlinklabel{s-s}{}
1378 \setganttlinklabel{f-f}{}
1379 \setganttlinklabel{s-f}{}
1380
```

2.16 grafik.sty

```
1381 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1382 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1383 TikZ, um Grafiken zeichnen zu können.]
1384 \RequirePackage{tikz}
1385
```

2.17 graph.sty

```

1386 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1387 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1388 \RequirePackage{tikz}

```

Für die die Adjazenzliste (xrightarrow)

```

1389 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

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

```

1390 \RequirePackage{blkarray}
1391 \usetikzlibrary{arrows.meta}

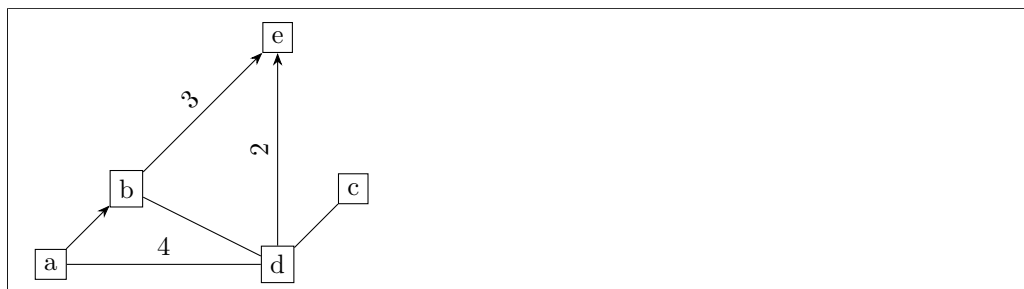
```

```

\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}

```



```

1392 \tikzset{
1393   li graph/.style={
1394     every node/.style={
1395       rectangle,
1396       draw,
1397     },
1398     every edge/.style={
1399       >={Stealth[black]},
1400       draw,
1401     },
1402     every edge/.append style={
1403       every node/.style={
1404         sloped,
1405         auto,
1406       }
1407     }
1408   },
1409   li markierung/.style={
1410     ultra thick,
1411   }
1412 }

```

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}

```

```

1413 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1414

```

2.18 hanoi.sty

```
1415 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1416 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
1417 von Hanoi-Grafiken]
```

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

```
1418 \RequirePackage{tikz}
1419 \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}

1420 \def\li@mset #1[#2]=#3{%
1421 \expandafter\xdef\csname #1#2\endcsname{#3}
1422 }
1423 \def\li@mget #1[#2]{%
1424 \csname #1#2\endcsname
1425 }
1426 \def\li@minc #1[#2] += #3{%
1427 \pgfmathparse{\li@mget #1[#2] + #3}%
1428 \li@mset #1[#2] = \pgfmathresult
1429 }
1430
1431 \def\liHanoi#1#2{
1432   \edef\li@numdiscs{#1}
1433   \def\li@sequence{#2}
1434   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1435     % init colors
1436     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purple!80!black}{
1437       \li@mset col[\j]=\c;
1438     }
1439     % draw poles and init pole counters
1440     \foreach \j in {1,2,3}{
1441       \li@mset pos[\j]=0
1442       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1443     }
1444     % draw base
1445     \draw (.5,-.5) -- +(3,0);
1446     % draw discs
1447     \foreach[count=\k] \i/\j in \li@sequence{
1448       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*\i/\li@numdiscs,0);
1449       \li@minc pos[\j] += {.5}
1450     }
1451   \end{tikzpicture}
1452 }
```

2.19 klassen-konfiguration-aufgabe.sty

```
1453 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1454 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-aufgabe}[2021/09/01 Die
1455 Klasse Aufgabe konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]

    Formatierung muss vor literatur sein, sonst Option clash
1456 \liLadePakete{
1457   formatierung,
1458   abmessung,
1459   literatur-dummy,
1460   makros,
1461   aufgaben-metadaten,
1462   kopf-fusszeilen,
1463   mathe
1464 }
1465
1466 % Komischer Option-Clash deshalb ganz am Ende, für die Silbentrennung
1467 \RequirePackage[ngerman]{babel}
1468 \ExplSyntaxOn
1469 \AddToHook{enddocument}{
1470   \_gib_github_url:
1471 }
1472 \ExplSyntaxOff
1473
```

2.20 klassen-konfiguration-examen.sty

```
1474 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1475 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-examen}[2021/09/04 Die
1476 Klasse liSetzeExamen konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

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

```
1477 \liLadePakete{
1478   formatierung,
1479   literatur-dummy,
1480   makros,
1481   aufgaben-metadaten,
1482   abmessung,
1483   typographie
1484 }
```

Formatierung für die Überschriften setzen.

```
1485 \RequirePackage{titlesec}
1486 \titleformat{\section}{\sffamily\huge\filcenter\bfseries}{\thesection}{1em}{}
1487 \newcommand{\sectionbreak}{\clearpage}
1488 \setcounter{secnumdepth}{0}
1489 \liLadeAllePakete
```

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

```
1490 \RequirePackage[ngerman]{babel}
1491 \RequirePackage{standalone}
1492 \ExplSyntaxOn
```

`\liSetzeExamenThemaNr`

```
1493 \def\liSetzeExamenThemaNr#1{
1494   \tl_gset:Nn \g_auf_examen_thema_nr_tl { #1 }
1495   \section{Thema-Nr.~#1}
1496 }
```

`\liSetzeExamenTeilaufgabeNr`

```
1497 \def\liSetzeExamenTeilaufgabeNr#1{
1498   \tl_gset:Nn \g_auf_examen_teilaufgabe_nr_tl { #1 }
1499   \subsection{Teilaufgabe-Nr.~#1}
1500 }
```

`\liBindeAufgabeEin`

```
1501 \def\liBindeAufgabeEin#1{
1502   \tl_gset:Nn \g_auf_examen_aufgabe_nr_tl { #1 }
1503   \input{
1504     \LehramtInformatikRepository /
1505     Staatsexamen /
1506     \g_auf_examen_nummer_tl /
1507     \g_auf_examen_jahr_tl /
1508     \g_auf_examen_monat_tl /
1509     \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
1510       Thema - \g_auf_examen_thema_nr_tl /
1511     }
1512     \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
1513       Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl /
1514     }
1515     Aufgabe - \g_auf_examen_aufgabe_nr_tl .tex
1516   }
1517 }
```

`\liAufgabenMetadaten` Das Metadaten-Makro überschreiben

```
1518 \def\liAufgabenMetadaten#1{
1519   \liMetaSetze{#1}
1520   \subsubsection{\_gib_aufgaben_titel:}
1521 }
```

```

1522 \setcounter{tocdepth}{4}
1523 \RequirePackage[titles]{tocloft}
1524 \AddToHook{begindocument}{
1525   \pagestyle{empty}
1526   \begin{center}
1527     \large
1528     Erste-Staatsprüfung-für-ein-Lehramt-an-öffentlichen-Schulen \par
1529
1530     \vspace{0.5cm}
1531
1532     Fach-Informatik \par
1533
1534     \vfill
1535
1536     {
1537       \sffamily\bfseries\Huge
1538
1539       \g_auf_examen_jahreszeit_tl \par
1540
1541       \g_auf_examen_jahr_tl \par
1542     }
1543
1544     \vspace{2cm}
1545
1546     {\LARGE \g_auf_examen_nummer_tl \par}
1547
1548     \vspace{0.5cm}
1549
1550     \g_auf_examen_fach_tl \par
1551
1552     \vspace{3cm}
1553
1554     Aufgabenstellungen-mit-Lösungsvorschlägen \par
1555
1556     \end{center}
1557
1558     % für den Abstand vor den section im Inhaltsverzeichnis
1559     % https://tex.stackexchange.com/questions/241445/how-to-control-spacing-in-toc-for-different-sections
1560     \setlength{\cftbeforesecskip}{1.5cm}
1561     \setlength{\cftbeforesubsecskip}{0.5cm}
1562     \renewcommand{\cftsubsecafterpnum}{\vspace{\cftbeforesubsecskip}}
1563     \setlength{\cftbeforesubsubsecskip}{0.1cm}
1564
1565     \vfill
1566     \renewcommand{\contentsname}{Aufgabenübersicht}
1567     \tableofcontents
1568     \clearpage
1569 }
1570 \ExplSyntaxOff
1571

```


2.21 komplexitaetstheorie.sty

```
1572 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1573 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1574 Setzen von Karp's NP-vollständigen Problemen, Hilfsmakros für die
1575 Polynomialzeitreduktion.]
```

2.21.1 Makro-Kürzel

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

```
1576 \liLadePakete{mathe}
```

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

```
1577 \RequirePackage{mdframed}
```

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

```
1578 \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`

```
1579 \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`

```
1580 \def\liProblemBeschreibung#1#2#3{
1581   \begin{mdframed}[
1582     userdefinedwidth=9cm,
1583     align=center,
1584     backgroundcolor=white!0,
1585   ]
1586   \centerline{\large\liProblemName{#1}}
1587
1588   \medskip
1589
1590   \begin{description}
1591     \item[Gegeben:] #2
1592     \item[Frage:] #3
1593   \end{description}
1594 \end{mdframed}
1595 }
```

```

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

1596 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1597 \begin{displaymath}
1598 \liProblemName{#1}
1599 \preceq_{#2}
1600 \liProblemName{#3}
1601 \end{displaymath}
1602 }

\liProblemVertexCover

1603 \def\liProblemClique{%
1604 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1605 Mindestgröße  $n$  in einem gegebenen Graphen.
1606 \footcite[Seite 76]{theo:fs:4}
1607 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1608 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1609 \footcite{wiki:cliquenproblem}
1610 }

\liProblemVertexCover

1611 \def\liProblemVertexCover{%
1612 %
1613 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1614 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1615 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1616 \footcite{wiki:knotenueberdeckung}
1617
1618 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1619 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1620 mindestens einem Knoten aus  $U$  verbunden ist.
1621 \footcite[Seite 78]{theo:fs:4}%
1622 }

\liProblemSubsetSum Kein Karp-21-Problem

1623 \def\liProblemSubsetSum{%
1624 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1625 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1626 \footcite{wiki:teilsommenproblem}
1627 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1628 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1629 größer als eine gegebene obere Schranke  $c$  ist.
1630 \footcite[Seite 74]{theo:fs:4}
1631 }

\liProblemSubsetSum Kein Karp-21-Problem

1632 \def\liProblemSat{%
1633 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1634 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1635 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1636 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1637 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1638 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1639 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1640 aufgestellt werden.
1641 \footcite[Seite 71]{theo:fs:4}
1642 }

1643

```

2.22 kontrollflussgraph.sty

1644 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1645 \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 = ...}]
```

```

1646 \RequirePackage{tikz}
1647 \usetikzlibrary{positioning}
1648 \tikzset{
1649   li kontrollfluss/.style={
1650     knoten/.style={
1651       circle,
1652       draw
1653     },
1654     usebox/.style={
1655       draw,
1656       rectangle,
1657       font=\scriptsize,
1658       anchor=west,
1659       align=left,
1660     },
1661     bedingung/.style={
1662       midway,
1663       draw=none,
1664       font=\scriptsize
1665     },
1666     knotenbeschriftung/.style={
1667       draw,
1668       rectangle,
1669       midway,
1670       font=\scriptsize
1671     },
1672     wahr/.style={
1673       thick
1674     },
1675     falsch/.style={
1676       dashed
1677     },
1678     every node/.style={
1679       circle,
1680       draw,
1681     },
1682     every edge/.append style={
1683       every node/.style={
1684         draw=none,
1685         bedingung,
1686       }
1687     },
1688     every path/.style={
1689       draw,
1690       ->,
1691     },
1692     every pin/.style={
1693       draw,
1694       dotted,
1695       rectangle,
1696       pin position=right
1697     },
1698     every pin edge/.style={
1699       dotted,
1700       arrows=-,
1701     }
1702   }
1703 }

```

2.22.4 Umgebungen

liKontrollflussgraph

```

1704 \NewDocumentEnvironment { liKontrollflussgraph } { 0{} } {

```

```

1705 \begin{tikzpicture}[
1706     li kontrollfluss,
1707     #1
1708 ]
1709 } {
1710 \end{tikzpicture}
1711 }

```

2.22.5 Makros

`\liAnweisung`

```
1712 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}
```

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

```
1713 \def\liBedingung#1#2{\node[bedingung,#1]{\texttt{#2}}}
```

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

```
1714 \def\liBedingungWahr#1{\node[bedingung,#1]{\texttt{[true]}}}
```

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

```
1715 \def\liBedingungFalsch#1{\node[bedingung,#1]{\texttt{[false]}}}
```

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

```
1716 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}
```

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

```
1717 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw
```

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

```
1718 \ExplSyntaxOn
```

```
1719 \NewDocumentCommand { \liKontrollKnotenPfad } { m }
```

```
1720 {
```

```
1721   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
```

```
1722   \seq_set_map:NNn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
```

```
1723   \seq_use:Nn \l_tmpa_seq { ~~~~ }
```

```
1724 }
```

```
1725 \ExplSyntaxOff
```

```
1726
```

2.23 kopf-fusszeilen.sty

```
1727 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1728 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1729 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1730 \ExplSyntaxOn

1731 \fancyhead{}
1732 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1733 \fancyhead[L0,RE]{\scriptsize\today}
1734 \fancyfoot{}
1735 \fancyfoot[LE,R0]{\thepage}
1736 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1737 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1738 \renewcommand{\headrulewidth}{0.4pt}
1739 \renewcommand{\footrulewidth}{0.4pt}
1740 \setlength{\headwidth}{\textwidth}

1741 \ExplSyntaxOff

1742
```

2.24 literatur-dummy.sty

```
1743 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1744 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1745 \def\literatur{}

\footcite

1746 % \RequirePackage[stable,multiple]{footmisc}
1747 \NewDocumentCommand{ \footcite } { o m }{}

1748
```

2.25 literatur.sty

```
1749 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1750 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1751 \RequirePackage{csquotes}
1752 \RequirePackage[
1753   bibencoding=utf8,
1754   citestyle=authortitle,
1755   backend=biber,
1756 ]{biblatex}
1757 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1758 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1759 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1760 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1761 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1762 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1763 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1764 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1765 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1766 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1767 % To allow footnotes in the heading
1768 \RequirePackage[stable,multiple]{footmisc}

\literatur

1769 \def\literatur{\printbibliography[heading=subbibliography]}

1770
```


2.26 makros.sty

```

1771 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1772 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1773 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1774 anderen Paket passen]

1775 \RequirePackage{hyperref}
1776 \RequirePackage{graphicx}

    Für die Umgebung liQuellen benötigt.
1777 \RequirePackage{paralist}

\inhaltsverzeichnis
1778 \def\inhaltsverzeichnis {
1779     \begin{mdframed}
1780         \begin{group}
1781             \let\clearpage\relax
1782             \tableofcontents
1783         \end{group}
1784     \end{mdframed}
1785 }

\memph \mephm (\marginpar and \emph)
1786 \newcommand\memph{1}{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1787 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1788 \newcommand\liPseudoUeberschrift{1}{
1789     \bigskip
1790     \noindent
1791     \textsf{\textbf{#1}}
1792     \noindent
1793 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1794 \newcommand\liBeschriftung{1}{
1795     \par
1796     \noindent
1797     \medskip
1798     \textbf{#1}:
1799     \medskip
1800     \noindent
1801 }

\hinweis
1802 \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.
1803 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

liEinbettung
1804 \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.
1805 \RequirePackage{xparse}
1806 \ExplSyntaxOn

```

```

1807 \NewDocumentEnvironment{ liAntwort } { 0{standard} }
1808 {
1809   \str_case:nn {#1} {
1810     {standard} {
1811       \def\beschriftung{}
1812       \mdfsetup{backgroundcolor=gray!1,linecolor=gray}
1813     }
1814     {richtig} {
1815       \def\beschriftung{richtig}
1816       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1817     }
1818     {falsch} {
1819       \def\beschriftung{falsch}
1820       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1821     }
1822     {muster} {
1823       \def\beschriftung{Musterlösung}
1824       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1825     }
1826   }
1827   \ifx\beschriftung\empty\else
1828     \noindent
1829     \textbf{\beschriftung{:}}
1830   \fi
1831   \begin{mdframed}
1832 }
1833 {\end{mdframed}}

```

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

```

1834 \NewDocumentEnvironment{ liAdditum } { o }
1835 {
1836   \begin{mdframed}[backgroundcolor=yellow!5]
1837     \IfNoValueTF {#1}
1838     { \liPseudoUeberschrift{Additum} }
1839     { \liPseudoUeberschrift{Additum:~#1} }
1840   }
1841 {\end{mdframed}}

```

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

Exkurs: Linear rekursiv

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

```

1842 \NewDocumentEnvironment{ liExkurs }{o +b}{
1843   \vspace{0.2cm}%
1844   \begin{mdframed}[
1845     backgroundcolor=white,
1846     bottomline=false,
1847     innermargin=1cm,
1848     leftline=true,
1849     linecolor=black,
1850     linewidth=0.1cm,
1851     outermargin=1cm,
1852     rightline=false,
1853     topline=false,
1854   ]

```

```

1855 \footnotesize
1856 \noindent%
1857 \textbf{Exkurs:~#1}\par%
1858 \noindent%
1859 #2
1860 \end{mdframed}
1861 \vspace{0.2cm}
1862 }{}

```

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

```

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

```

Weiterführende Literatur:

- Quelle 1
- Quelle 2

```

1863 \cs_new:Npn \listen@punkt #1 {\item #1}
1864 \NewDocumentEnvironment { liQuellen }{ +b }
1865 {
1866   \seq_clear_new:N \l_quellen
1867   \seq_set_split:Nnn \l_quellen {\item} {#1}
1868   \seq_remove_all:Nn \l_quellen {}
1869   \begin{mdframed}[backgroundcolor=white!0]
1870     \footnotesize
1871     \noindent
1872     \textsf{\textbf{Weiterführende~Literatur:}}
1873     \medskip
1874     \begin{compactitem}
1875       \seq_map_function:NN \l_quellen {\listen@punkt}
1876     \end{compactitem}
1877   \end{mdframed}
1878   %
1879   \makeatletter
1880   \par\@afterindentfalse\@afterheading
1881   \makeatother
1882 } {}

```

liLernkartei

```

1883 \NewDocumentEnvironment { liLernkartei }{ m +b }
1884 {
1885   \begin{mdframed}
1886     \footnotesize
1887     \noindent%
1888     \textbf{Lernkarteikarte:~#1}\par%
1889     \noindent%
1890     #2
1891   \end{mdframed}
1892 } {}

```

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

```

1893 \NewDocumentEnvironment { liDiagramm }{ m +b }
1894 {
1895   \begin{mdframed}[backgroundcolor=white!0]
1896     \small
1897     \noindent%
1898     \textit{#1}:
1899     \begin{center}

```

```

1900 #2
1901 \medskip
1902 \end{center}
1903 \end{mdframed}
1904 } {}
1905 \ExplSyntaxOff

\liFussnoteUrl \liFussnoteUrl[\langle zusätzlicher-text \rangle]{\langle url \rangle} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
1906 \NewDocumentCommand{\liFussnoteUrl} { o m } {
1907 \footnote{\url{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1908 }
1909

\liFussnoteLink \liFussnoteLink[\langle zusätzlicher-text \rangle]{\langle link-text \rangle}{\langle url \rangle} \liFussnoteLink[zusätzlicher Text]{tex
Einen Link, d. h. einen Link-Text und eine URL als Fußnote setzen.
1910 \NewDocumentCommand{\liFussnoteLink} { o m m } {
1911 \footnote{\href{#3}{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1912 }

\zB
1913 \def\zB{z.\,B. }

\ZB
1914 \def\ZB{Z.\,B. }

\dh
1915 \def\dh{d.\,h. }

1916

```

2.27 master-theorem.sty

1917 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1918 \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}

1919 \ExplSyntaxOn

1920 \RequirePackage{amsmath}

\liRundeKlammer

1921 \def\liRundeKlammer#1{

1922 \negthinspace \left(#1 \right)

1923 }

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

1924 \def\liThetaOhneMathe#1{

1925 \Theta \liRundeKlammer{#1}

1926 }

1927 \def\liTheta#1{

1928 \ifmmode

1929 \liThetaOhneMathe{#1}

1930 \else

1931 \$\liThetaOhneMathe{#1}\$

1932 \fi

1933 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
1934 \def\liOmegaOhneMathe#1{
1935 \Omega \liRundeKlammer{#1}
1936 }
1937 \def\liOmega#1{
1938 \ifmmode
1939 \liOmegaOhneMathe{#1}
1940 \else
1941 $\liOmegaOhneMathe{#1}$
1942 \fi
1943 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
1944 \def\liOOhneMathe#1{
1945 \mathcal{O} \liRundeKlammer{#1}
1946 }
1947 \def\liO#1{
1948 \ifmmode
1949 \liOOhneMathe{#1}
1950 \else
1951 $\liOOhneMathe{#1}$
1952 \fi
1953 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
1954 \def\liTOhneMathe#1#2{
1955 \tl_if_blank:nTF {#1}
1956 {}
1957 {#1 \cdot }
1958 T
1959 \liRundeKlammer{\frac{n}{#2}}
1960 }
1961 \def\liT#1#2{
1962 \ifmmode
1963 \liTOhneMathe{#1}{#2}
1964 \else
1965 $\liTOhneMathe{#1}{#2}$
1966 \fi
1967 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
1968 \def\liRekursionsGleichung{
1969 $T(n) = \liT{a}{b} + f(n)$
1970 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
1971 \def\liBedingungEins{
1972 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
1973 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
1974 \def\liBedingungZwei{
1975 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
1976 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
1977 \def\liBedingungDrei{
1978 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
1979 }

1980 \ExplSyntaxOff

```

\liMasterVariablen

```

1981 \def\liMasterVariablen{
1982   \begin{displaymath}
1983     T(n) = \liT{a}{b} + f(n)
1984   \end{displaymath}
1985
1986   \begin{itemize}
1987     \item[$a = $]
1988       Anzahl der rekursiven Aufrufe, Anzahl der Unterprobleme in der
1989       Rekursion
1990       ($a \geq 1$).
1991
1992     \item[$\textstyle{\frac{1}{b}} = $]
1993       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
1994       repräsentiert wird, Anteil an der Verkleinerung des Problems ($b > 1$).
1995
1996     \item[$f(n) = $]
1997       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
1998       die Kombination der Teillösungen entstehen. Eine von $T(n)$
1999       unabhängige und nicht negative Funktion.
2000   \end{itemize}
2001   \footcite{wiki:master-theorem}
2002   \footcite[Seite 19-35]{aud:fs:2}
2003 }
```

\liMasterFaelle

```

2004 \def\liMasterFaelle{
2005   \begin{description}
2006     \item[1. Fall:]
2007       $T(n) \in \liTheta{n^{\log\sb{b}a}}$
2008
2009       \hfill falls \liBedingungEins
2010       für $\varepsilon > 0$
2011
2012     \item[2. Fall:]
2013       $T(n) \in \liTheta{n^{\log\sb{b}a} \cdot \log n}$
2014
2015       \hfill falls \liBedingungZwei
2016
2017     \item[3. Fall:]
2018       $T(n) \in \liTheta{f(n)}$
2019
2020       \hfill falls \liBedingungDrei
2021       für $\varepsilon > 0$
2022       und ebenfalls für ein $c$ mit $0 < c < 1$ und alle hinreichend großen $n$
2023       gilt:
2024       $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$
2025   \end{description}
2026 }
```

liMasterVariablenDeklaration

```

2027 \def\liMasterVariablenDeklaration#1#2#3{
2028   \begin{description}
2029     \item[Allgemeine Rekursionsgleichung:] \strut
2030
2031     \liRekursionsGleichung
2032
2033     \item[Anzahl der rekursiven Aufrufe ($a$):] \strut
2034
2035     #1
2036
2037     \item[Anteil Verkleinerung des Problems ($b$):] \strut
2038   \end{description}
```

```

2039     um  $\frac{1}{b}$  also  $b = \frac{1}{a}$ 
2040
2041     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut
2042
2043      $T(n) = \frac{1}{b} + T(\frac{n}{b})$ 
2044
2045     \item[Ergibt folgende Rekursionsgleichung:] \strut
2046
2047      $T(n) = \frac{1}{b} + T(\frac{n}{b})$ 
2048 \end{description}
2049 }

\liMasterFallRechnung
2050 \def\liMasterFallRechnung#1#2#3{
2051   \begin{description}
2052     \item[1. Fall:] \liBedingungEins:
2053
2054     #1
2055
2056     \item[2. Fall:] \liBedingungZwei:
2057
2058     #2
2059
2060     \item[3. Fall:] \liBedingungDrei:
2061
2062     #3
2063   \end{description}
2064 }

\liMasterExkurs
2065 \def\liMasterExkurs{
2066   \begin{liExkurs}[Master-Theorem]
2067     \liMasterVariablen
2068
2069     \noindent
2070     Dann gilt:
2071
2072     \liMasterFaelle
2073   \end{liExkurs}
2074 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
2075 \def\liMasterWolframLink#1{
2076   Berechne die Rekursionsgleichung auf WolframAlpha:
2077   \href{https://www.wolframalpha.com/input/?i=#1}{WolframAlpha}
2078 }

2079

```


2.28 mathe.sty

```
2080 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2081 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
2082
2083 % for example \ltimes \rtimes
2084 %\RequirePackage{amssymb}
2085 \RequirePackage{amsmath}
2086
2087 %%
2088 % \mlq \mrq
2089 %%
2090 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
2091 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
2092
```

2.29 minimierung.sty

```

2093 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2094 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
2095 Minimierungsalgorithmus von einem NEA zu einem DEA]
2096 \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

2097 \def\liFussnote#1{\$x_{#1}\$}

2098 \def\li@fussnote@text#1#2{
2099 \liFussnote{#1}
2100 \quad
2101 {\footnotesize #2}
2102 }

\liFussnoteEinsText
2103 \def\liFussnoteEinsText{
2104 \li@fussnote@text{1}
2105 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
2106 }

\liFussnoteZweiText
2107 \def\liFussnoteZweiText{
2108 \li@fussnote@text{2}
2109 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
2110 }

\liFussnoteDreiText
2111 \def\liFussnoteDreiText{
2112 \li@fussnote@text{3}

```

```

2113 {In weiteren Iterationen markierte Zustände.}
2114 }

\liFussnoteVierText
2115 \def\liFussnoteVierText{
2116   \li@fussnote@text{4}
2117   {...}
2118 }

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



2119 \def\liFussnoten{
2120   \bigskip
2121
2122   \noindent
2123   \liFussnoteEinsText
2124
2125   \noindent
2126   \liFussnoteZweiText
2127
2128   \noindent
2129   \liFussnoteDreiText
2130
2131   \noindent
2132   \liFussnoteVierText
2133 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
2134 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
2135 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
2136 \def\liZustandsPaar#1#2{
2137   $(
2138     \liZustandsPaarVariablenName_#1,
2139     \liZustandsPaarVariablenName_#2
2140   )$
2141 }

liUebergangsTabelle
2142 \renewcommand{\arraystretch}{1.4}
2143 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
2144   \liPseudoUeberschrift{Übergangstabelle}
2145   \begin{center}
2146     \begin{tabular}{r|l|l}
2147       \textbf{Zustandspaar} & \textbf{#1} & \textbf{#2} \\ \hline
2148     \end{tabular}
2149   \end{center}
2150 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle: Minimierungstabelle (Table filling)

2152 \ExplSyntaxOn
2153 \def\liUeberschriftDreiecksTabelle{
2154   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
2155 }

```

\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. —————

```

2156 \def\liMinimierungErklaerung{
2157   %\footcite[Seite~19]{koenig}
2158   \liParagraphMitLinien{
2159     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
2160     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
2161     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
2162     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
2163      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
2164     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
2165     somit~zueinander~( $k - 1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
2166     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
2167     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
2168   }
2169 }
2170 \ExplSyntaxOff
2171

```

2.30 normalformen.sty

```
2172 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2173 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10
2174 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
2175 Attributhülle]
```

Mathe für textit tex etc, typographie für geschweifte Klammer

```
2176 \liLadePakete{mathe,typographie}
2177 \directlua{
2178   helfer = require('lehramt-informatik-helfer')
2179   normalformen = require('lehramt-informatik-normalformen')
2180 }
```

2.30.1 Makro-Kürzel

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

```
2181 \def\liTeilen#1{
2182   \directlua{tex.print(normalformen.teilen('#1'))}
2183 }
```

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

```
2184 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(\#1)}
2185 \def\liAttributHuelle#1{
2186   \ifmmode
2187     \liAttributHuelleOhneMathe{\#1}
2188   \else
2189     $\liAttributHuelleOhneMathe{\#1}$
2190   \fi
2191 }
```

\liAttributMenge Let-Abkürzung: $\text{let}\ \text{m}=\text{liAttributMenge}$
 2192 \def\liAttributMenge#1{\{ \textit{\text{#1}} \}}

liAHuelle

```
2193 \NewDocumentEnvironment{ liAHuelle } { +b } {
2194   \begingroup
2195   \footnotesize
2196   \begin{multline*}
2197     \#1
2198   \end{multline*}
2199   \endgroup
2200 } { }
```

\liLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multline

Let-Abkürzung: $\text{let}\ \text{ahL}=\text{liLinksReduktion}$
 $\text{ahL}\{\text{ursprüngliche linke Attributmenge}\{\text{ohne dieses Attribut}\}\{\text{Ergebnis}\}$
 2201 \def\liLinksReduktion#1#2#3{
 2202 \shoveleft{
 2203 \liAttributHuelleOhneMathe{FA,
 2204 \liAttributMenge{\#1 \string\ #2}} =
 2205 } \}

```

2206 \shoveright{
2207 \liAttributMenge{#3}
2208 } \\\
2209 }

```

```

\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( $F, \{A, E \setminus E\}$ ) =  $\{A, E, F, B, D\}$ 
2210 \def\liLinksReduktionInline#1#2#3{%
2211 {%
2212 \footnotesize%
2213 $\liAttributHuelleOhneMathe{F,
2214 \liAttributMenge{#1 \string\ #2}} =
2215 \liAttributMenge{#3}$
2216 }
2217 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahr=\liLinksReduktionInline
\ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
2218 \def\liRechtsReduktionInline#1#2#3#4{%
2219 {%
2220 \footnotesize%
2221 $\liAttributHuelleOhneMathe{
2222 F \setminus
2223 \liFunktionaleAbhaengigkeit{#1}
2224 \def\tmp{#2}\ifx\tmp\empty
2225 \else
2226 \cup \liFunktionaleAbhaengigkeit{#2}
2227 \fi
2228 ,
2229 \liAttributMenge{#3}
2230 } =
2231 \liAttributMenge{#4}$
2232 }
2233 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
\fa{A, B -> C, D}:  $\{A, B\} \rightarrow \{C, D\}$  Regulärer Ausdruck zum Konvertieren:
\$(.*?) \\\rightarrow (.*)\$ \fa{$1 -> $2}
2234 \def\liFunktionaleAbhaengigkeit#1{%
2235 \directlua{normalformen.drucke_funk_abhaengigkeit('#1', false)}%
2236 }

```

```

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

```

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

```

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

```

```

2237 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
2238   \liGeschweifteKlammern
2239   {#1}
2240   {
2241     \begin{align*}
2242       \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
2243     \end{align*}
2244   }
2245   {-0.5cm}
2246   {-1.7cm}
2247 }

\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}

2248 \NewDocumentCommand {\liRelation} { O{R} m } {
2249   $\directlua{
2250     local name = helper.konvertiere_tiefgestellt('#1')
2251     tex.print(name)
2252   }$(\textit{\, #2\,})
2253 }

2254

```

2.31 petri.sty

```
2255 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2256 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]
```

2.31.1 Makro-Kürzel

```
\let\t=\liPetriTransitionsName
\let\tp=\liPetriTransPfeile
\let\k=\liPetriErreichKnotenDrei
```

```
2257 \RequirePackage{tikz}
2258 \usetikzlibrary{petri,arrows.meta}
```

Für die Darstellungsmatrix

```
2259 \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

```
2260 \def\liPetriSetzeSchluessel{%
2261   \def\TmpTransitionOne{}%
2262   \def\TmpTransitionTwo{}%
2263   \def\TmpTransitionThree{}%
2264   \def\TmpTransitionFour{}%
2265   \def\TmpTransitionFive{}%
2266   \def\TmpTransitionSix{}%
2267   \def\TmpTransitionSeven{}%
2268   \def\TmpTransitionEight{}%
2269   \def\TmpTransitionNine{}%
2270   \def\TmpTransitionTen{}%
2271   \pgfkeys{/petri/.cd,
2272     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2273     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2274     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2275     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
```



```

2276 p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
2277 p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2278 p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2279 p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2280 p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2281 p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2282 t1/.store in=\TmpTransitionOne,t1/.default=activated,
2283 t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2284 t3/.store in=\TmpTransitionThree,t3/.default=activated,
2285 t4/.store in=\TmpTransitionFour,t4/.default=activated,
2286 t5/.store in=\TmpTransitionFive,t5/.default=activated,
2287 t6/.store in=\TmpTransitionSix,t6/.default=activated,
2288 t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2289 t8/.store in=\TmpTransitionEight,t8/.default=activated,
2290 t9/.store in=\TmpTransitionNine,t9/.default=activated,
2291 t10/.store in=\TmpTransitionTen,t10/.default=activated,
2292 scale/.store in=\TmpScale,scale/.default=0.5,
2293 x/.store in=\TmpX,x/.default=5,
2294 y/.store in=\TmpY,y/.default=5,
2295 }%
2296 }

```

```

2297 \tikzset{
2298   li petri/.style={
2299     activated/.style={
2300       very thick
2301     },
2302     inhibitor/.style={
2303       {Circle[open,length=2mm,fill=white]}-
2304     }
2305   }
2306 }

```

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

```

2307 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2308 \def\liPetriTransitionsName#1{
2309   \ifmmode
2310     \liPetriTransitionsNameOhneMathe{#1}
2311   \else
2312     $\liPetriTransitionsNameOhneMathe{#1}$
2313   \fi
2314 }

```

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

```

2315 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2316   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2317 }

```

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

```

2318 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2319 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2320 }

```

2.32 potenzmengen-konstruktion.sty

```
2321 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2322 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2323 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2324 \liLadePakete{formale-sprachen}
2325 \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}
    }
  }
}
```

```
2326 \def\liZustandsMengenSammlung#1#2{
2327   \liZustandsnameGross{#1}
2328   {
2329     \footnotesize
2330     \liPotenzmenge{
2331       \str_case:nn {#1} {#2
2332       }
2333     }
2334 }
```

```
\liZustandsMengenSammlungNr
```

```
2335 \def\liZustandsMengenSammlungNr#1#2{
2336   \liZustandsnameGross{#1}
2337   {
```

```
2338     \footnotesize
2339     \liZustandsmengeNr{
2340         \str_case:nn {#1} #2
2341     }
2342 }
2343 }

2344 \ExplSyntaxOff
2345
```

2.33 pseudo.sty

```

2346 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2347 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen
2348 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>$E' \leftarrow \emptyset$; $L \leftarrow E$; Sortiere die Kanten in L aufsteigend nach ihrem Kantengewicht.;</p> <p>while $L \neq \emptyset$ do</p> <div style="margin-left: 20px;"> wähle eine Kante $e \in L$ mit kleinstem Kantengewicht; entferne die Kante e aus L; if der Graph $(V, E' \cup \{e\})$ keinen Kreis enthält then $E' \leftarrow E' \cup \{e\}$; end </div> <p>end</p> <p>Result: $M = (V, E')$ ist ein minimaler Spannbaum von G.</p>

```

2349 \RequirePackage[german,boxruled]{algorithm2e}

```

```

2350

```

2.34 pumping-lemma.sty

2351 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2352 \ProvidesPackage{lehrant-informatik-pumping-lemma}[2021/08/11 Enthält die
 2353 Definitionen für das Pumping-Lemma in der Regulären Sprache und
 2354 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2355 \def\liPumpingRegulaer{%
2356   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2357   alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2358    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2359   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2360
2361   \begin{enumerate}
2362     \item  $|v| \geq 1$ 
2363     (Das Wort  $v$  ist nicht leer.)
2364
2365     \item  $|uv| \leq j$ 
2366     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2367
2368     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w$  in  $L$ 
2369     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2370     Sprache  $L$ )
2371   \end{enumerate}
2372
2373   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2374   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2375 }
```

\liPumpingKontextfrei

```
2376 \def\liPumpingKontextfrei{%
2377   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2378   sich alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  zerlegen lassen in
2379    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2380
2381   \begin{enumerate}
2382     \item  $|vx| \geq 1$ 
2383     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2384
2385     \item  $|vwx| \leq j$ 
2386     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2387
2388     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy$  in  $L$  (Für jede
2389     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2390     Sprache  $L$ )
2391   \end{enumerate}
2392 }
2393
```

2.35 quicksort.sty

```

2394 % https://tex.stackexchange.com/a/142634
2395 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2396 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2397
2398 %-----
2399 % USAGE:
2400 % \QSinitialize{comma, separated, numerical, values}
2401 % \loop
2402 % \QSpivotStep
2403 % \ifnum\value{pivotcount}>0
2404 %   \QSSortStep
2405 % \repeat
2406 %-----
2407
2408 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2409 \RequirePackage{xintfrac, xinttools}
2410
2411 \RequirePackage{tikz}
2412
2413 %-----
2414 % FIRST PART: TikZ styles and macros for the actual drawing
2415 \newcounter{cellcount}% used for coordinates of the node
2416 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2417
2418 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2419 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2420 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2421 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2422 % by police of LaTeX good conduct ? )
2423 \tikzset{l/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2424          o/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2425          r/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2426 % this is the "b" style as used in the image below
2427          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2428 % nicer:
2429          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2430          g/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2431
2432 % NOTE the b style was originally the same as the r(aised) style apart from
2433 % not being raised, but I find it nicer with a somewhat different
2434 % specification. I have not updated the images though.
2435
2436 % How the nodes are drawn depending on whether on the left of the pivot value
2437 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2438
2439 \def\DecoLEFT #1{%
2440   \xintFor* ##1 in {#1} \do
2441     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2442 }
2443
2444 \def\DecoINERT #1{%
2445   \xintFor* ##1 in {#1} \do
2446     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2447 }
2448
2449 \def\DecoRIGHT #1{%
2450   \xintFor* ##1 in {#1} \do
2451     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2452 }
2453
2454 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2455   \xintFor* ##1 in {#1} \do

```

```

2456     {\stepcounter{cellcount}}%
2457     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2458 }
2459
2460 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2461     \xintFor* ##1 in {#1} \do
2462     {\stepcounter{cellcount}}%
2463     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2464 }
2465
2466 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2467     \xintFor* ##1 in {#1} \do
2468     {\stepcounter{cellcount}}%
2469     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2470 }
2471
2472 %-----
2473 % SECOND PART: the actual sorting routines.
2474
2475 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2476 \def\QS@sort@b #1{\ifcase #1
2477     \expandafter\QS@sort@empty
2478     \or\expandafter\QS@sort@single
2479     \else\expandafter\QS@sort@c
2480     \fi
2481 }%
2482 \def\QS@sort@empty #1{}
2483 \def\QS@sort@single #1{\QSIr {#1}}
2484
2485 % This step is to pick the last as pivot.
2486 \def\QS@sort@c #1%
2487     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2488
2489 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2490 % The trick with \xintApplyUnbraced is that for example when selecting
2491 % the elements smaller than pivot, if we had been using \xintApply we
2492 % would have had at the minimum an empty brace pair. Thus we use the
2493 % "unbraced" variant, but then the \QS@select@smaller has added in
2494 % anticipation a level of braces.
2495 \def\QS@sort@d #1#2{%
2496     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2497     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2498     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2499 }%
2500 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{#{#2}}{ }}% space will stop a f-
    expansion
2501 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{#{#2}}{ }}% space will stop a f-
    expansion
2502 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{#{#2}}{ }}% space will stop a f-
    expansion
2503
2504 %
2505 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2506 %
2507 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2508 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2509 % latter must handle correctly an empty argument.
2510
2511 %-----
2512 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QSinitialize.
2513
2514 % This draws all with suitable highlighting for the newly chosen pivots
2515 % (which will be shown raised)

```

```

2516 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2517             \let\QSIr\DecoINERT
2518             \let\QSIrr\DecoINERT
2519             \let\QSRr\DecoRIGHTwithPivot
2520 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2521             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2522             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2523 }
2524
2525 % This sorts and then draws, showing where the pivot chosen in the previous
2526 % step go. Next time they will have become "inert". If pivotcount is still at
2527 % zero on exit from \QSpivotStep, then this is the signal to stop before
2528 % executing \QSSortStep.
2529 \def\QSSortStep {\def\QSLr {\noexpand\QS@sort@a}%
2530             \def\QSRr {\noexpand\QS@sort@a}%
2531             \def\QSIr {\noexpand\QSIrr}%
2532             \let\QSIrr\relax
2533             \edef\QS@list{\QS@list}%
2534             \let\QSLr\relax
2535             \let\QSRr\relax
2536             \let\QSIr\relax
2537             \edef\QS@list{\QS@list}%
2538             \let\QSLr\DecoLEFT
2539             \let\QSIr\DecoINERTwithPivot
2540             \let\QSIrr\DecoINERT
2541             \let\QSRr\DecoRIGHT
2542 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2543             \setcounter{cellcount}{0}%
2544             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2545 }
2546
2547 \def\QSinitialize #1{%
2548     % first, we convert the comma separated values into a list of braced items
2549     % we use an \edef, and anyhow many \edef's will be used later
2550     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2551     \let\QSRr\DecoRIGHT
2552     % The \QSRr marker mutated to draw the last element as
2553     % pivot and the earlier ones with the suitable style.
2554     %
2555     % The list of marked braced items \QS@list is used both for drawing
2556     % (as here) and for doing the exchange of elements during sort.
2557     \par\centerline{\rule[1.5mm]{0pt}{8mm}}\setcounter{cellcount}{0}%
2558     \begin{tikzpicture}\QS@list\end{tikzpicture}}
2559 }
2560

```


2.36 relationale-algebra.sty

```

2561 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2562 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]
2563 \RequirePackage{amsmath}
2564 \RequirePackage{amssymb}

```

```

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

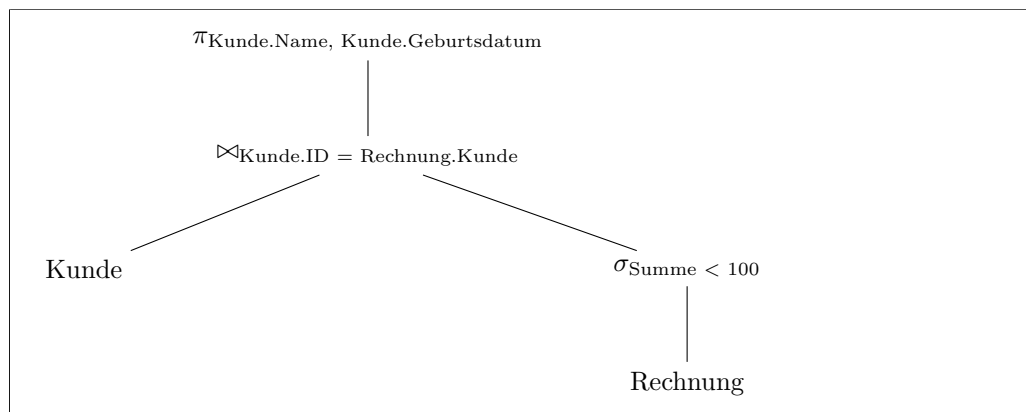
  \node[below=of pi]
    (theta join) {\bowtie_{\text{Kunde.ID = Rechnung.Kunde}}};
    edge (pi);

  \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}

```



```

2565 \RequirePackage{tikz}
2566 \usetikzlibrary{positioning}

    Privates Makros, das zwei Querstriche erzeugt.
2567 \def\o@join{\setbox0=\hbox{\bowtie}}%
2568 \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2569 }

```

```
\leftouterjoin A \leftouterjoin B: A ⋈ B
```

```
2570 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}
```

```
\rightouterjoin A \rightouterjoin B: A ⋉ B
```

```
2571 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}
```

```
\fullouterjoin A \fullouterjoin B: A ⋈⋉ B
```

```
2572 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
```

```
2573
```

2.37 rmodell.sty

```
2574 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2575 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01
2576 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2577 Datenbanken.]
2578 \RequirePackage{soul}
```

2.37.1 Makro-Kürzel

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

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

```
2579 \def\liPrimaer#1{\ul{#1}}
```

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

```
2580 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

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

```
2581 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2582 \ExplSyntaxOn
2583 \NewDocumentEnvironment { liRmodell }
2584 { +b }
2585 {
2586   \medskip
2587   {
2588     \linespread{2}
2589     \setlength{\parindent}{0pt}
2590     \li@Rmodell@Schrift#1
2591   }
2592   \medskip
2593 } {}
2594 \ExplSyntaxOff
```

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

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

```
2595 \def\liRelationMenge#1#2{
2596 \noindent
2597 #1 : \[ #2 ]\}
2598 \par
2599 }
```

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

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

```
2600 \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}
```

```
2601 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2602
```

2.38 sortieren.sty

2603 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2604 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
 2605 für händisches Sortieren bzw. einen Schreibtischlauf setzen]

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



2606 \RequirePackage{tikz}
 2607 \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.

```
2608 \def\liVertauschen#1{
2609   \directlua{
2610     local sortieren = require('lehramt-informatik-sortieren')
2611     sortieren('#1')
2612   }
2613 }
```

\liSortierPfeil

```
2614 \def\liSortierPfeil#1#2{
2615   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2616 }
```

\liSortierPfeilUnten

```
2617 \def\liSortierPfeilUnten#1#2{
2618   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2619 }
```

\liSortierMarkierung

```
2620 \def\liSortierMarkierung#1#2{\node[
2621   draw,
2622   very thick,
2623   fit=(reihe.#1) (reihe.#2),
2624   inner sep=0pt
2625 ] {}
2626 }

2627 \tikzset{
2628   li sortierung zahlenreihe/.style={
2629     draw,
2630     thin,
2631     font=\large,
2632     rectangle split horizontal,
2633     rectangle split,
2634   }
2635 }
```

```

2636 % https://tex.stackexchange.com/a/140895
2637 \RequirePackage{forest,xstring}
2638 \usetikzlibrary{calc}
2639
2640 \makeatletter
2641 \pgfmathdeclarefunction{strrepeat}{2}{%
2642   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2643   \let\pgfmathresult\pgfutil@empty
2644   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2645     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2646     \advance\pgfmath@count-1\relax
2647   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2648 \makeatother
2649
2650 \def\myNodes{}
2651
2652 \ExplSyntaxOn
2653 \newcommand*\sortList[1]{%
2654   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2655 \ExplSyntaxOff
2656
2657 \forestset{
2658   sort/.code={%
2659     \pgfmathparse{level()>\forestSortLevel}%
2660     \ifnum\pgfmathresult=0
2661       \StrSubstitute{\forestov{content}}{ }{,}{\myList}%
2662       \sortList\myList
2663       \StrSubstitute{\myList}{,}{ }{\myList}%
2664       \pgfmathparse{strrepeat("1",level())}%
2665       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-1!({\forestov{name}}$)
2666         (m\forestov{name}) {\myList}}%
2667       \pgfmathparse{level()=\forestSortLevel}%
2668       \ifnum\pgfmathresult=1
2669         \forestOget{\forestov{@first}}{name}\forestFirst
2670         \forestOget{\forestov{@last}}{name}\forestLast
2671         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2672           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2673       \fi
2674       \ifnum\forestov{@parent}=0\else
2675         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2676       \fi
2677       \gappto\myNodes{;}%
2678     \fi}}
2679
2680 \forestset{sort level/.code=%
2681   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2682   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2683

```

2.39 spalten.sty

```
2684 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2685 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2686 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2687 realisiert werden kann.]
2688 \RequirePackage{multicol}
```

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

```
2689 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2690
```

2.40 sql.sty

```
2691 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2692 \ProvidesPackage{lehramt-informatik-tabelle}[2021/09/02 Zu Setzen von SQL]
2693 \liLadePakete{syntax}
2694 \RequirePackage{fancyvrb}
2695 \DefineVerbatimEnvironment{liSqlErgebnis}{Verbatim}
2696 {fontsize=\footnotesize}
2697
```

2.41 struktogramm.sty

```
2698 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2699 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2700 Paket struktex zum Setzen von Struktogrammen]
2701 \RequirePackage{struktex}
2702
```

2.42 syntax.sty

```
2703 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2704 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2705 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2706 \RequirePackage{xparse}
```

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

2.42.1 Makro-Kürzel

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

```
2707 \ExplSyntaxOn
2708 \directlua{
2709   syntax = require('lehramt-informatik-syntax')
2710   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2711   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2712   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2713   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2714   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2715   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2716 }
2717 \RequirePackage{hyperref}
2718 \RequirePackage{minted}
2719 % pygmentize -L styles
2720 \usemintedstyle{colorful}
2721 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2722 %\AfterEndEnvironment{minted}{\end{mdframed}}
2723 %\setminted{breaklines=true,linenos}
2724 \setminted{
2725   breaklines=true,
2726   linenos,
2727   fontsize=\footnotesize,
2728 }
```

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

Let-Abkürzung: `\let\j=\liJavaCode`

```
2729 \def\liJavaCode#1{\,\mintinline{java}|#1|\,}
```

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

```
2730 \def\liLatexCode#1{\mintinline{latex}|#1|}
```

```
2731 \def\li@GithubLink#1#2{
2732   \begin{flushright}
2733     \tiny
2734     Code-Beispiel~auf~Github~ansehen:~
2735     \href{#1}{\nolinkurl{#2}}
2736   \end{flushright}
2737 }
```

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

```
2738 \NewDocumentCommand{\liJavaDatei}{ O{firstline=3} m }{
2739   \inputminted[#1]{java}{
2740     \directlua{
2741       syntax.drucke_absoluten_pfad('#2', false)
2742     }
2743   }
2744   \li@GithubLink
```



```

2745     {\directlua{syntax.drucke_github_url('#2', false)}}
2746     {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2747 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2748 \NewDocumentCommand{\liJavaTestDatei}{ O{firstline=3} m }{
2749   \inputminted[#1]{java}{
2750     \directlua{
2751       syntax.drucke_absoluten_pfad('#2', true)
2752     }
2753   }
2754   \li@GithubLink
2755   {\directlua{syntax.drucke_github_url('#2', true)}}
2756   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2757 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([.
\liJavaExamen$1{$2}{$3}{$4}{$5}
2758 \NewDocumentCommand{\liJavaExamen}{ O{firstline=3} m m m m }{
2759   \inputminted[#1]{java}{
2760     \directlua{
2761       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2762     }
2763   }
2764
2765   \li@GithubLink
2766   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2767   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2768 }

\liAssemblerCode
2769 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2770 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2771   \inputminted{asm}{#1}
2772 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2773 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2774   \inputminted{componentpascal}{#1}
2775 }

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2776 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2777 \NewDocumentCommand{\liHaskellDatei}{ m }{
2778   \inputminted{haskell}{#1}
2779 }

2780 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2781 \def\liSqlCode#1{\mintinline{sql}|#1|}
2782

```

2.43 syntaxbaum.sty

```
2783 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2784 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2785 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
2786 \RequirePackage{tikz-qtrees}
2787
2788 \tikzset{li parsetree/.style={
2789     every internal node/.style={
2790         draw,circle
2791     },
2792     every leaf node/.style={
2793         draw,rectangle
2794     },
2795 }
2796 }
2797
```

2.44 synthese-algorithmus.sty

```
2798 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2799 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2800 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2801 Relation in die 3. Normalform]
2802 \liLadePakete{normalformen,mathe,typographie}
2803 \ExplSyntaxOn
```

2.44.1 Makro-Kürzel

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

2.44.2 TeX-Markup Grundgerüst

```
\let\schrittE=\liSyntheseUeberErklaerung
\begin{enumerate}
\item \schrittE{1}

\begin{enumerate}
\item \schrittE{1-1}
\item \schrittE{1-2}
\item \schrittE{1-3}
\item \schrittE{1-4}
\end{enumerate}

\item \schrittE{2}
\item \schrittE{3}
\item \schrittE{4}
\end{enumerate}
```

2.44.3 TeX-Markup Linksreduktion

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

\liPseudoUeberschrift{\fa{C, E -> D, X}}

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

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

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

2.44.4 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}

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

2.44.5 TeX-Markup Relationen formen

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

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

1. Kanonische Überdeckung

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

(a) Linksreduktion

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

(b) Rechtsreduktion

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

(c) Löschen leerer Klauseln

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

(d) Vereinigung

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

2. Relationsschemata formen

— Erzeuge für jede funktionale Abhängigkeit $\alpha \rightarrow \beta \in F_c$ ein Relationenschema $R_\alpha := \alpha \cup \beta$. —

3. Schlüssel hinzufügen

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

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

```
2804 \def\liSyntheseUeberschrift#1{
2805   {
2806     \bfseries
2807     \sffamily
2808     \str_case:nn {#1} {
2809       {1} {Kanonische-Überdeckung}
2810       {1-1} {Linksreduktion}
2811       {1-2} {Rechtsreduktion}
2812       {1-3} {Löschen-leerer-Klauseln}
2813       {1-4} {Vereinigung}
2814       {2} {Relationsschemata-formen}
2815       {3} {Schlüssel-hinzufügen}
2816       {4} {Entfernung-überflüssiger-Teilschemata}
2817     }
```

2818 }
2819 }

```

2820 \def\li@synthese@erklaerung@texte#1{
2821   \str_case:nn {#1} {
2822     {1} {
2823       Die-kanonische-Überdeckung---also-die-kleinst-mögliche-noch-
2824       äquivalente-Menge-von-funktionalen-Abhängigkeiten-kann-in-vier-
2825       Schritten-erreicht-werden.
2826     }
2827     {1-1} {
2828       Führe-für-jede-funktionale-Anhängigkeit~
2829        $\alpha \rightarrow \beta$ -die-Linksreduktion-durch,~
2830       überprüfe-also-für-alle~
2831        $A \rightarrow \alpha$ ,~ob- $A$ -überflüssig-ist,~d.h.-ob-
2832        $\beta \subseteq \text{AttributHuelle}\{F, \alpha\}$ .
2833     }
2834     {1-2} {
2835       Führe-für-jede-(verbliebene)-funktionale-Abhängigkeit- $\alpha \rightarrow \beta$ -die-Rechtsreduktion-durch,~überprüfe-also-für-
2836       alle- $B \rightarrow \beta$ ,~ob- $B \rightarrow \text{AttributHuelle}\{F, \alpha\}$ ,~
2837        $\alpha \rightarrow \beta$ - $\cup$ - $(\alpha \rightarrow \beta)$ ,~
2838        $\alpha$ -gilt.~In-diesem-Fall-ist- $B$ -auf-der-rechten-Seite-
2839       überflüssig-und-kann-eliminiert-werden,~dh- $\alpha \rightarrow \beta$ -wird-durch- $\alpha \rightarrow (\beta \cup B)$ -
2840       ersetzt.
2841     }
2842     {1-3} {
2843       Entferne-die-funktionalen-Abhängigkeiten-der-Form- $\alpha \rightarrow \emptyset$ ,~die-im-2.-Schritt-möglicherweise-
2844       entstanden-sind.
2845     }
2846     {1-4} {
2847       Fasse-mittels-der-Vereinigungsregel-funktionale-Abhängigkeiten-
2848       der-Form- $\alpha \rightarrow \beta_1, \dots, \alpha \rightarrow \beta_n$ ,~so-dass- $\alpha \rightarrow \beta_1 \cup \dots \cup \beta_n$ -
2849       verbleibt.
2850     }
2851     % Kemper Seite 197
2852     {2} {
2853       Erzeuge-für-jede-funktionale-Abhängigkeit- $\alpha \rightarrow \beta$ -ein-Relationenschema- $\mathcal{R} \subseteq \text{AttributHuelle}\{\alpha\}$ -
2854       := $\alpha \cup \beta$ .
2855     }
2856     {3} {
2857       Falls-eines-der-in-Schritt-2.-erzeugten-Schemata- $\mathcal{R} \subseteq \text{AttributHuelle}\{\alpha\}$ -
2858       einen-Schlüsselkandidaten-von- $\mathcal{R}$ -bezüglich- $F \subseteq C$ -
2859       enthält,~sind-wir-fertig,~sonst-wähle-einen-Schlüsselkandidaten-
2860        $K \subseteq C$ - $\subseteq \mathcal{R}$ -aus-und-definiere-folgendes-
2861       zusätzliche-Schema- $\mathcal{R} \subseteq \mathcal{K}$ -:= $\mathcal{K}$ -
2862       und- $\mathcal{F} \subseteq \mathcal{K}$ -:= $\emptyset$ 
2863     }
2864     {4} {
2865       Eliminiere-diejenigen-Schemata- $\mathcal{R} \subseteq \text{AttributHuelle}\{\alpha\}$ ,~die-in-einem-
2866       anderen-Relationenschema- $\mathcal{R} \subseteq \text{AttributHuelle}\{\alpha\}$ -enthalten-sind,~d.h.-
2867        $\mathcal{R} \subseteq \mathcal{R} \subseteq \text{AttributHuelle}\{\alpha\}$ .
2868     }
2869   }
2870 }
2871 \def\liSyntheseErklaerung#1{
2872   {
2873     \itshape
2874     \footnotesize
2875   }
2876 }

```

```

2880     \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
2881   }
2882 }

```

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

```

2883 \def\liSyntheseUeberErklaerung#1{
2884   \liSyntheseUeberschrift{#1}\par
2885   \liSyntheseErklaerung{#1}
2886 }

```

```

2887 \ExplSyntaxOff
2888

```

2.45 tabelle.sty

```
2889 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2890 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]
2891 \RequirePackage{tabularx}
2892
```

2.46 typographie.sty

```
2893 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2894 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
2895 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
2896 formatierung.sty definiert.]
```

```
2897 \ExplSyntaxOn
```

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

```
2898 \RequirePackage{fontawesome}
```

```
\liErledigt \liErledigt: ☑
```

```
2899 \let\liErledigt=\faCheckSquareO
```

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

```
2900 \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 —

```
2901 \def\liParagraphMitLinien#1{
```

```
2902 \noindent
```

```
2903 \vrule height 2pt depth -1.6pt width 0.4cm
```

```
2904 \enspace
```

```
2905 #1
```

```
2906 \enspace
```

```
2907 \leaders\vrule height 2pt depth -1.6pt \hfill \null
```

```
2908 \par
```

```
2909 \medskip
```

```
2910 }
```

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

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

```
2911 \def\liGeschweifteKlammern#1#2#3#4{
```

```
2912 \par
```

```
2913 \medskip
```

```
2914 \noindent
```

```
2915 #1 \, $= \Bigl\{ $\
```

```
2916 \vspace{#3}
```

```
2917 #2
```

```
2918 \vspace{#4}
```

```
2919 \begin{flushright} $\Bigr\} $\end{flushright}
```

```
2920 \par
```

```
2921 }
```

```
\liTypoUeberschrift
```

```
2922 \def\liTypoUeberschrift#1{
```

```
2923 {
```

```
2924 \bfseries\sffamily
```

```
2925 #1
```

```
2926 }
```

```
2927 }
```


\liTypoUeberGross

```
2928 \def\liTypoUeberGross#1{
2929   {
2930     \huge
2931     \liTypoUeberschrift{#1}
2932   }
2933 }
```

\liTypoUeberGross

```
2934 \def\liTypoUeberGROSS#1{
2935   {
2936     \Huge
2937     \liTypoUeberschrift{#1}
2938   }
2939 }
```

2940 \ExplSyntaxOff

2941

2.47 uml.sty

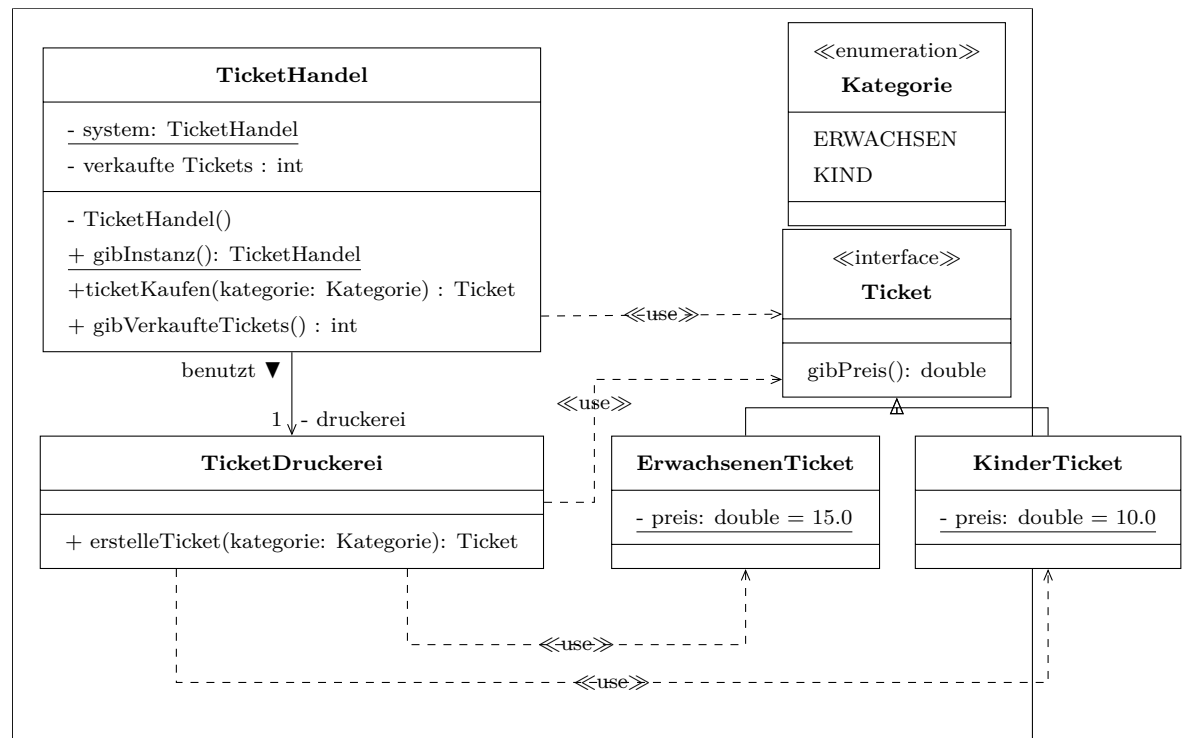
```

2942 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2943 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
2944 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
2945 Erweiterung bereitstellt]

2946 \RequirePackage{tikz-uml}
2947 \RequirePackage{tikz-uml-activity}
2948 % Not compatible with wasysym
2949 %\RequirePackage{mathabx}
2950 \RequirePackage{wasysym}
2951 \usetikzlibrary{positioning}

2952 \tikzumlset{
2953   fill class=white!0,
2954   font=\footnotesize,
2955   fill object=white!0,
2956   fill note=white!0,
2957   fill state=white!0,
2958   % Use case
2959   fill usecase=white!0,
2960   fill system=white!0,
2961 }

```



```

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

```

```

2962 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
2963   \def\@liDirLeft{}
2964   \def\@liDirRight{ \RIGHTarrow}
2965   \pgfkeys{/lese/dir/.is choice}
2966   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
2967   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
2968   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{\LEFTarrow }}}
2969   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
2970
2971   \def\@liPos{above}
2972   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
2973

```

```

2974 \def\@liDistance{0cm}
2975 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
2976
2977 \pgfkeys{/lese/.cd,#1}
2978
2979 \node[\@liPos = \@liDistance of #2-middle] {
2980   \@liDirLeft{\footnotesize#2}\@liDirRight
2981 };
2982 }
2983

```

2.48 vollstaendige-induktion.sty

```

2984 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2985 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01]
2986 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
2987 Überschriften für die einzelnen Schritte]

```

2.48.1 Makro-Kürzel

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

```

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

```

Lade häufig benötigte Pakete

```

2988 \RequirePackage{lehramt-informatik-typographie}
2989 \RequirePackage{lehramt-informatik-mathe}
2990 \RequirePackage{lehramt-informatik-syntax}
2991 \ExplSyntaxOn

```

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

Let-Abkürzung: \let\m=\liInduktionMarkierung

```
2992 \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

```
2993 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
2994 \def\liInduktionAnfang{
2995   \liPseudoUeberschrift{Induktionsanfang}
2996
2997   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2998   \liParagraphMitLinien{
2999     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
3000   }
3001 }
```

\liInduktionVoraussetzung

```
3002 \def\liInduktionVoraussetzung{
3003   \liPseudoUeberschrift{Induktionsvoraussetzung}
3004
3005   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
3006   \liParagraphMitLinien{
3007     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ $.
3008   }
3009 }
```

\liInduktionSchritt

```
3010 \def\liInduktionSchritt{
3011   \liPseudoUeberschrift{Induktionsschritt}
3012
3013   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
3014   \liParagraphMitLinien{
3015     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
3016     auch~ $A(n=k+1)$ ~wahr~sein~muss.
3017   }
3018 }

3019 \ExplSyntaxOff
3020
```

2.49 wasserfall.sty

```
3021 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3022 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
3023 \RequirePackage{tikz}
3024 \tikzset{wasserfall/.style={
3025   >=stealth,
3026   node distance = 2mm and -8mm,
3027   start chain = A going below right,
3028   every node/.style = {
3029     draw,
3030     text width=24mm,
3031     minimum height=12mm,
3032     align=center,
3033     inner sep=1mm,
3034     fill=white,
3035     drop shadow={fill=black},
3036     on chain=A
3037   },
3038 }}
3039 \usetikzlibrary{chains,positioning,shadows}
3040
```

2.50 wpkalkuel.sty

```
3041 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3042 \ProvidesPackage{lehramt-informatik-wpkalkuel}[2020/06/13]
```

2.50.1 Makro-Kürzel

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

```
3043 \RequirePackage{amsmath}
3044 \ExplSyntaxOn
```

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

```
3045 \def\liWpKalkuelOhneMathe#1#2{
3046   \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)
3047 }
3048 \def\liWpKalkuel#1#2{
3049   \ifmmode
3050     \liWpKalkuelOhneMathe{#1}{#2}
3051   \else
3052     $\liWpKalkuelOhneMathe{#1}{#2}$
3053   \fi
3054 }
```

\MatheEnv

```
3055 \def\MatheEnv#1{
3056   \medskip
3057
3058   \hspace{1em}#1
3059
3060   \medskip
3061 }
```

\Mathe

```
3062 \def\Mathe#1{
3063   \MatheEnv{${#1}$}
3064 }
```

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

```
3065 \def\liWpEquivalent#1{
3066   \MatheEnv{$\equiv$\hspace{1em}${#1}$}
3067 }
```

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

```
3068 \newlength{\@Skip@Erklaerung@Reset}
3069 \def\liWpErklaerung#1{
3070   \setlength{\@Skip@Erklaerung@Reset}{\leftskip}
3071   \setlength{\leftskip}{0.5cm}
3072
3073   \par
3074   \noindent
3075   {
3076     \scriptsize
3077     #1
3078   }
3079   \par
3080
3081   \setlength{\leftskip}{\@Skip@Erklaerung@Reset}
3082 }
```

\liWpErklaerungVerzweigung

```

3083 \def\liWpErklaerungVerzweigung{
3084   $\liWpKalkuelOhneMathe{if~\{-b~\}~then~\{-a1~\}~else~\{-a2~\}}{Q}
3085   \equiv
3086   (b \land \liWpKalkuelOhneMathe{a1}{Q})
3087   \lor
3088   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
3089 }

3090 \ExplSyntaxOff

3091

```

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		1760, 1761, 1762,	2838, 2841, 2851,
\#	114	1763, 1764, 1765, 1766	2852, 2853, 2858, 2859
\,	358, 413,	\AddToHook	2419, 2420, 2421
	640, 658, 1290,	\advance	2646
	1292, 1913, 1914,	\AfterEndEnvironment	2722
	1915, 2252, 2729, 2915	\allsectionsfont	1339
\@Skip@Erklaerung@Reset		\Alph	1353
	3068, 3070, 3081	\alph	1353, 1354
\@afterheading	1880	\alpha	2829, 2831, 2832,
\@afterindentfalse	1880		2835, 2837, 2838,
\@liDirLeft	2963, 2968, 2980		2839, 2840, 2841,
\@liDirRight	2964, 2966,		2845, 2851, 2852,
	2967, 2968, 2969, 2980		2857, 2858, 2859,
\@liDistance			2862, 2870, 2871, 2872
	2974, 2975, 2979	\arabic	1353, 2441, 2446,
\@liPos	2971, 2972, 2979		2451, 2457, 2463, 2469
\	670, 698,	\arraystretch	2142
	699, 702, 703, 706,	B	
	707, 802, 803, 804,	\BeforeBeginEnvironment	
	911, 945, 947, 977,		2721
	986, 1031, 1073,	\begin	696, 750, 765,
	1074, 1075, 1080,		800, 824, 871, 903,
	1081, 1082, 1102,		918, 943, 953, 973,
	1787, 2147, 2205, 2208		993, 1025, 1042,
\{	212, 1227,		1071, 1096, 1117,
	1237, 1249, 1250,		1141, 1156, 1270,
	1255, 1289, 1627,		1359, 1434, 1526,
	2192, 2597, 2915, 3084		1581, 1590, 1597,
\}	212, 1227,		1705, 1779, 1831,
	1237, 1249, 1250,		1836, 1844, 1869,
	1257, 1293, 1628,		1874, 1885, 1895,
	2192, 2597, 2919, 3084		1899, 1982, 1986,
_	37, 45, 50, 52, 305,		2005, 2028, 2051,
	333, 357, 358, 372,		2066, 2145, 2146,
	373, 379, 382, 385,		2196, 2241, 2361,
	395, 407, 1470, 1520		2381, 2522, 2544,
			2558, 2721, 2732, 2919
_	2204, 2214	\begingroup	1780, 2194, 2642
		\beschriftung	
			1811, 1815,
			1819, 1823, 1827, 1829
\addbibresource		\beta	2829,
	1757, 1758, 1759,		2832, 2836, 2837,
			2226, 2838, 2852, 2859
		C	
		\c	1436, 1437
		\cdot	1957, 2013, 2024
		\centerline	
			1586, 2520, 2542, 2557
		\cftbeforesecskip	1560
		\cftbeforesubsecskip	
			1561, 1562
		\cftbeforesubsubsecskip	
			1563
		\cftsubsecafterpnum	1562
		\chapter	1343, 1344
		\char	1787
		\clearpage	1487, 1568, 1781
		\cline	670
		\clist	231, 279,
			280, 302, 306, 2654
		\columnbreak	2689
		\contentsname	1566
		\cs	305, 333, 357,
			358, 395, 407, 1863
		\csname	1421, 1424
		\cup	1250,
			2226, 2838, 2852, 2859

D		2198, 2243, 2371,	<code>\fi</code> 596, 604, 612,
<code>\DeclareMathSymbol</code> ..	2391, 2522, 2544,		620, 627, 1233,
..... 2090, 2091	2558, 2722, 2736, 2919		1247, 1283, 1830,
<code>\DecoINERT</code>	<code>\endscname</code> 1421, 1424		1932, 1942, 1952,
2444, 2517, 2518, 2540	<code>\endgroup</code> 1783, 2199, 2647		1966, 2190, 2227,
<code>\DecoINERTwithPivot</code> .	<code>\enspace</code> 2904, 2906		2313, 2480, 2672,
..... 2460, 2539	environments:		2673, 2676, 2678, 3053
<code>\DecoLEFT</code> 2439, 2538	<code>liAdditum</code> <u>1834</u>	<code>\filcenter</code> 1486	
<code>\DecoLEFTwithPivot</code> ..	<code>liAHuelle</code> <u>2193</u>	<code>\fontspec</code> 1339	
..... 2454, 2516	<code>liAntwort</code> <u>1805</u>	<code>\footcite</code> 693, 762, 784,	
<code>\DecoRIGHT</code> 2449, 2541, 2551	<code>liDiagramm</code> <u>1893</u>	831, 854, 886, 939,	
<code>\DecoRIGHTwithPivot</code> .	<code>liEinbettung</code> <u>1804</u>	990, 1017, 1063,	
..... 2466, 2519	<code>liExkurs</code> <u>1842</u>	1606, 1609, 1616,	
<code>\definecolor</code> 1341	<code>liGraphenFormat</code> . <u>1413</u>	1621, 1626, 1630,	
<code>\DefineVerbatimEnvironment</code>	<code>liKasten</code> <u>1358</u>	1636, 1641, <u>1746</u> ,	
..... 2695	<code>liKontrollflussgraph</code>	2001, 2002, 2157, 2374	
<code>\delta</code> 70, 112, 170, 212, 1241 <u>1704</u>	<code>\footnote</code> 1907, 1911	
<code>\dh</code> <u>1915</u> , 2840	<code>liLernkartei</code> <u>1883</u>	<code>\footnotesize</code> 152, 368,	
<code>\directlua</code>	<code>liProduktionsRegeln</code>	450, 544, 1214,	
63, 142, 200, 205, <u>1265</u>	1802, 1855, 1870,	
1222, 1236, 1256,	<code>liProjektSprache</code> <u>1803</u>	1886, 2101, 2195,	
1264, 1271, 1276,	<code>liQuellen</code> <u>1863</u>	2212, 2220, 2329,	
2177, 2182, 2235,	<code>liRelationenSchemaFormat</code>	2338, 2581, 2696,	
2242, 2249, 2609, <u>2601</u>	2727, 2879, 2954, 2980	
2708, 2740, 2745,	<code>liRmodell</code> <u>2581</u>	<code>\footrulewidth</code> . 1370, 1739	
2746, 2750, 2755,	<code>liUebergangsTabelle</code>	<code>\foreach</code> . 1436, 1439, 1446	
2756, 2760, 2766, 2767 <u>2142</u>	<code>\forestFirst</code> .. 2669, 2672	
<code>\do</code> 2440, 2445,	<code>\equiv</code> 3066, 3085	<code>\forestLast</code> ... 2670, 2672	
2450, 2455, 2461, 2467	<code>\erzeuge@tiefgestellt</code>	<code>\forestOget</code> ... 2669, 2670	
<code>\dots</code> 530, 534,	... 1236, 1237, 1241	<code>\forestOnes</code> 2682	
1627, 2368, 2851, 2852	<code>\expandafter</code>	<code>\forestOv</code> 2671, 2672, 2675	
<code>\DOWNarrow</code> 2967	1421, 2475, 2477,	<code>\forestov</code> . 2661, 2665,	
<code>\draw</code> 1441, 1444,	2478, 2479, 2487, 2645	2666, 2669, 2670,	
1447, 2316, 2615, 2618	<code>\ExplSyntaxOff</code> . 54, 97,	2671, 2672, 2674, 2675	
E		<code>\forestset</code> 2657, 2680	
<code>\edef</code> 1432,	139, 144, 197, 202,	<code>\forestSortLevel</code>	
2533, 2537, 2549, 2550	207, 552, 574, 589,	2659, 2667, 2681, 2682	
<code>\edge</code> 276	665, 1217, 1295,	<code>\frac</code> 1959, 1992, 2024, 2039	
<code>\else</code> 594, 602, 610, 618,	1330, 1472, 1570,	<code>\fullouterjoin</code> <u>2572</u>	
625, 1231, 1245,	1725, 1741, 1905,		
1281, 1827, 1930,	1980, 2170, 2344,	G	
1940, 1950, 1964,	2594, 2655, 2780,	<code>\g</code> 39, 279, 280,	
2188, 2225, 2311,	2887, 2940, 3019, 3090	302, 306, 312, 313,	
2479, 2672, 2674, 3051	<code>\ExplSyntaxOn</code>	314, 315, 316, 318,	
<code>\emph</code> .. 691, 938, 1193, 35, 66, 107,	319, 321, 322, 324,	
1608, 1637, 1639, 1786	140, 165, 198, 203,	325, 326, 327, 328,	
<code>\empty</code> 1827, 2224	228, 495, 558, 575,	329, 330, 331, 334,	
<code>\emptyset</code>	614, 1190, 1285,	338, 339, 340, 343,	
2134, 2846, 2867, 2900	1307, 1468, 1492,	345, 346, 347, 348,	
<code>\end</code> ... 735, 761, 786,	1718, 1730, 1806,	349, 350, 361, 362,	
821, 855, 887, 915,	1919, 2152, 2325,	363, 364, 372, 373,	
931, 950, 958, 989,	2582, 2652, 2707,	375, 381, 382, 384,	
1018, 1039, 1064,	2803, 2897, 2991, 3044	385, 387, 388, 396,	
1089, 1109, 1128,	F		
1153, 1171, 1272,	<code>\faCheckSquareO</code> 2899	401, 403, 408, 410,	
1361, 1450, 1556,	<code>\faCircleThin</code> 1209	414, 1494, 1498,	
1593, 1594, 1601,	<code>\faGg</code> 1203	1502, 1506, 1507,	
1710, 1784, 1833,	<code>\fancyfoot</code>	1508, 1509, 1510,	
1841, 1860, 1876,	1365, 1366, 1367,	1512, 1513, 1515,	
1877, 1891, 1902,	1734, 1735, 1736, 1737	1539, 1541, 1546, 1550	
1903, 1984, 2000,	<code>\fancyhead</code>	<code>\Gamma</code> . 111, 169, 212, 1250	
2025, 2048, 2063,	1364, 1731, 1732, 1733	<code>\gappto</code> 2677	
2073, 2149, 2150,	<code>\faSquareO</code> 1197	<code>\geometry</code> 5	
		<code>\geq</code> 1634, 1990,	
		2357, 2362, 2378, 2382	

H			185, 310, 562, 566,	<code>\leq</code> 2024, 2365, 2385
<code>\hbox</code> 2567			580, 585, 1314, 1321	<code>\let</code> 1239,
<code>\headrulewidth</code> . 1369, 1738				1240, 1781, 2516,
<code>\headwidth</code> 1740		L		2517, 2518, 2519,
<code>\hfill</code> 2009, 2015, 2020, 2907		<code>\l</code> 68, 69, 70, 71,		2532, 2534, 2535,
<code>\hinweis</code> 1802		72, 73, 76, 77, 78,		2536, 2538, 2539,
<code>\hline</code> 2147		79, 80, 82, 84, 89,		2540, 2541, 2551,
<code>\href</code> 397,		90, 91, 92, 93, 94,		2643, 2681, 2682, 2899
1303, 1911, 2077, 2735		109, 110, 111, 112,		<code>\li@chomsky@erklaerung@texte</code>
<code>\hspace</code> .. 2319, 3058, 3066		113, 114, 115, 118,	 509, 545
<code>\ht</code> 2568		119, 120, 121, 122,		<code>\li@EntwurfsCode</code>
<code>\Huge</code> 1537, 2936		123, 124, 130, 131,		687, 738, 739, 740,
<code>\huge</code> 1486, 2930		132, 133, 134, 135,		789, 790, 791, 792,
		136, 167, 168, 169,		858, 859, 860, 861,
		170, 171, 172, 173,		862, 863, 890, 891,
		176, 177, 178, 179,		892, 893, 894, 895,
		180, 181, 182, 188,		961, 1131, 1132,
		189, 190, 191, 192,		1133, 1134, 1174, 1175
		193, 194, 560, 563,		<code>\li@EntwurfsCodeAllgemein</code>
		568, 569, 572, 577,	 686
		578, 581, 582, 587,		<code>\li@fussnote@text</code> 2098,
		1309, 1310, 1311,		2104, 2108, 2112, 2116
		1312, 1315, 1316,		<code>\li@GithubLink</code>
		1317, 1318, 1324,		2731, 2744, 2754, 2765
		1325, 1326, 1327,		<code>\li@mget</code> . 1423, 1427, 1447
		1721, 1722, 1723,		<code>\li@minc</code> 1426, 1448
		1866, 1867, 1868, 1875		<code>\li@mset</code>
	<code>\labelenumi</code> 1354			1420, 1428, 1437, 1440
	<code>\labelenumii</code> 1355			<code>\li@numdiscs</code>
	<code>\labelitemi</code> 1348			... 1432, 1441, 1447
	<code>\labelitemii</code> 1349			<code>\li@Rmodell@Schrift</code> .
	<code>\labelitemiii</code> 1350			... 2581, 2590, 2600
	<code>\labelitemiv</code> 1351			<code>\li@sequence</code> .. 1433, 1446
	<code>\land</code> 3086, 3088			<code>\li@synthese@erklaerung@texte</code>
	<code>\LARGE</code> 1343, 1546		 2820, 2880
	<code>\large</code> ... 1527, 1586, 2631			<code>\li@Ableitung</code> 1264
	<code>\leaders</code> 2907			<code>liAdditum</code> (environment)
	<code>\left</code> 1922		 1834
	<code>\LEFTarrow</code> 2968			<code>liAHuelle</code> (environment)
	<code>\leftarrow</code> 606		 2193
	<code>\leftouterjoin</code> 2570			<code>\li@Alphabet</code> 1249
	<code>\leftskip</code> 3070, 3071, 3081			<code>liAntwort</code> (environment)
	<code>\LehramtInformatikAutorEmail</code>		 1805
 1737			<code>\li@Anweisung</code> 1712
	<code>\LehramtInformatikAutorName</code>			<code>\li@AssemblerCode</code> ... 2769
 1736			<code>\li@AssemblerDatei</code> .. 2770
	<code>\LehramtInformatikGitBranch</code>			<code>\li@Attribut</code> 2600
 400, 2715			<code>\li@AttributHuelle</code> ...
	<code>\LehramtInformatikGithubCodeRepo</code>			... 2184, 2832, 2837
 2714			<code>\li@AttributHuelleOhneMathe</code>
	<code>\LehramtInformatikGithubDomain</code>		 2184, 2187,
 2711			<code>\li@AttributMenge</code>
	<code>\LehramtInformatikGithubRawDomain</code>			2189, 2203, 2213, 2221
 398, 2712			<code>\li@AttributMenge</code>
	<code>\LehramtInformatikGithubTexRepo</code>			2192, 2204, 2207,
 399, 2713			2214, 2215, 2229, 2231
	<code>\LehramtInformatikRepository</code>			<code>\li@Aufgabe</code> 16
 17, 20, 23,			<code>\li@AufgabenMetadaten</code>
	26, 29, 1504, 1757,		 47, 1518
	1758, 1759, 1760,			<code>\li@AufgabenTitel</code> 55
	1761, 1762, 1763,			<code>\li@Ausdruck</code> 1286
	1764, 1765, 1766, 2710			<code>\li@Automat</code> 66
	<code>\LehramtInformatikTitel</code>			<code>\li@AutomatenKante</code> 98
 1732			<code>\li@BandAlphabet</code> 1250
I				
<code>\i</code> 1446, 1447				
<code>\ifcase</code> 2476				
<code>\ifmode</code> 592,				
600, 608, 616, 623,				
1229, 1243, 1279,				
1928, 1938, 1948,				
1962, 2186, 2309, 3049				
<code>\IfNoValueTF</code>				
... 1837, 1907, 1911				
<code>\ifnum</code> 2403,				
2644, 2660, 2668, 2674				
<code>\ifx</code> 1827, 2224, 2672				
<code>\in</code> 518, 674,				
1634, 1972, 1975,				
1978, 2007, 2013,				
2018, 2357, 2368,				
2378, 2388, 2829,				
2831, 2837, 2858, 3007				
<code>\inhaltsverzeichnis</code> 1778				
<code>\input</code> 17, 20,				
23, 26, 29, 420, 1503				
<code>\inputminted</code> 2739, 2749,				
2759, 2771, 2774, 2778				
<code>\int</code> 2654				
<code>\item</code> ... 489, 490, 767,				
771, 776, 781, 825,				
834, 839, 847, 919,				
924, 928, 954, 994,				
999, 1006, 1014,				
1043, 1048, 1052,				
1057, 1157, 1162,				
1167, 1591, 1592,				
1863, 1867, 1987,				
1992, 1996, 2006,				
2012, 2017, 2029,				
2033, 2037, 2041,				
2045, 2052, 2056,				
2060, 2362, 2365,				
2368, 2382, 2385, 2388				
<code>\itshape</code> 543, 2878				
J				
<code>\j</code> 1436, 1437, 1439, 1440,				
1441, 1446, 1447, 1448				
K				
<code>\k</code> 1446				
<code>\keys</code> 41, 75,				
87, 117, 127, 175,				

\liBedingung	1713	\liEntwurfsDekoriererCode	889, 900	\liErMpEntity	1194
\liBedingungDrei	1977, 2020, 2060	\liEntwurfsDekoriererUml	870, 898	\liErMpRelationship	1200
\liBedingungEins	1971, 2009, 2052	\liEntwurfsEinfacheFabrik	933	\liErRelationship	1192, 1201, 1203
\liBedingungFalsch	1715	\liEntwurfsEinfacheFabrikAkteure	917, 935	\liExamensAufgabe	19
\liBedingungWahr	1714	\liEntwurfsEinfacheFabrikUml	902, 934	\liExamensAufgabeA	28
\liBedingungZwei	1974, 2015, 2056	\liEntwurfsEinzelstueck	963	\liExamensAufgabeTA	25
\liBeschriftung	1794	\liEntwurfsEinzelstueckAkteure	952, 968	\liExamensAufgabeTTA	22
\liBindeAufgabeEin	1501	\liEntwurfsEinzelstueckBeschreibung	937, 964	\liExkurs (environment)	1842
\liChomskyErklaerung	509, 550	\liEntwurfsEinzelstueckCode	960, 970	\liFalsch	490
\liChomskyUeberErklaerung	548	\liEntwurfsEinzelstueckUml	942, 966	\liFlaci	1296
\liChomskyUeberschrift	497, 549	\liEntwurfsErbauer	1020	\liFremd	2580
\liCpmEreignis	558	\liEntwurfsErbauerAkteure	992, 1022	\liFunktionaleAbhaengigkeit	2223, 2226, 2234
\liCpmFruehErklaerung	629	\liEntwurfsErbauerUml	972, 1021	\liFunktionaleAbhaengigkeiten	2237
\liCpmFruehI	622, 642	\liEntwurfsFabrikmethode	1066, 1091	\liFussnote	2097, 2099
\liCpmSpaetErklaerung	647	\liEntwurfsFabrikmethodeAkteure	1041, 1068	\liFussnoteDreiText	2111, 2129
\liCpmSpaetI	615, 660	\liEntwurfsFabrikmethodeUml	1024, 1067	\liFussnoteEinsText	2103, 2123
\liCpmVon	598	\liEntwurfsKompositum	1091	\liFussnoteLink	1910
\liCpmVonOhneMathe	598, 601, 603	\liEntwurfsKompositumAkteure	1093	\liFussnoten	2119
\liCpmVonZu	590	\liEntwurfsKompositumUml	1070, 1092	\liFussnoteUrl	1110, 1906
\liCpmVonZuOhneMathe	590, 593, 595	\liEntwurfsModellPraesentation	1112	\liFussnoteVierText	2115, 2132
\liCpmVorgang	575	\liEntwurfsModellPraesentationAkteure	1114	\liFussnoteZweiText	2107, 2126
\liCpmZu	606	\liEntwurfsModellPraesentationBeschreibung	1095, 1113	\liGeschweifteKlammern	1268, 2238, 2911
\liCpmZuOhneMathe	606, 609, 611	\liEntwurfsModellPraesentationCode	737, 747	\liGrammatik	1307
liDiagramm (environment)	1893	\liEntwurfsModellPraesentationUml	695, 745	liGraphenFormat (environment)	1413
liEinbettung (environment)	1804	\liEntwurfsAdapter	794	\liHanoi	1420
\liEntwurfs	1112	\liEntwurfsAdapterAkteure	764, 796	\liHaskellCode	2776
\liEntwurfsAbstrakteFabrik	742	\liEntwurfsAdapterCode	788, 797	\liHaskellDatei	2777
\liEntwurfsAbstrakteFabrikBeschreibung	690, 743	\liEntwurfsAdapterUml	749, 795	\liInduktionAnfang	2994
\liEntwurfsAbstrakteFabrikCode	1095, 1113	\liEntwurfsBeobachter	865	\liInduktionErklaerung	2993
\liEntwurfsAbstrakteFabrikUml	1136	\liEntwurfsBeobachterAkteure	823, 867	\liInduktionMaschine	2992
\liEntwurfsAdapter	794	\liEntwurfsBeobachterCode	857, 868	\liInduktionSchritt	3010
\liEntwurfsAdapterAkteure	764, 796	\liEntwurfsBeobachterUml	799, 866	\liInduktionVoraussetzung	3002
\liEntwurfsAdapterCode	788, 797	\liEntwurfsDekorierer	897	\liJavaCode	2729
\liEntwurfsAdapterUml	749, 795	\liEntwurfsDekoriererAkteure	899	\liJavaDatei	688, 2738
\liEntwurfsBeobachter	865	\liEntwurfsDekoriererCode	889, 900	\liJavaExamen	2758
\liEntwurfsBeobachterAkteure	823, 867	\liEntwurfsDekoriererUml	870, 898	\liJavaTestDatei	2748
\liEntwurfsBeobachterCode	857, 868	\liEntwurfsEinfacheFabrik	933	liKasten (environment)	1358
\liEntwurfsBeobachterUml	799, 866	\liEntwurfsEinfacheFabrikAkteure	917, 935	\liKellerAutomat	107
\liEntwurfsDekorierer	897	\liEntwurfsEinfacheFabrikUml	902, 934	\liKellerKante	145
\liEntwurfsDekoriererAkteure	899	\liEntwurfsEinzelstueck	963	\liKellerUebergang	140, 146
\liEntwurfsDekoriererCode	889, 900	\liEntwurfsEinzelstueckAkteure	952, 968	\liKontrollCode	1716
\liEntwurfsDekoriererUml	870, 898	\liEntwurfsEinzelstueckBeschreibung	937, 964	liKontrollflussgraph (environment)	1704
\liEntwurfsEinfacheFabrik	933	\liEntwurfsEinzelstueckCode	960, 970	\liKontrollKnotenPfad	1718
\liEntwurfsEinfacheFabrikAkteure	917, 935	\liEntwurfsEinzelstueckUml	942, 966	\liKontrollTextzeileKnoten	1717, 1722
\liEntwurfsEinfacheFabrikUml	902, 934	\liEntwurfsErbauer	1020	\liKurzeTabellenLinie	670
\liEntwurfsEinzelstueck	963	\liEntwurfsErbauerAkteure	992, 1022	\liLadeAllePakete	233, 1489
\liEntwurfsEinzelstueckAkteure	952, 968	\liEntwurfsErbauerUml	972, 1021		
\liEntwurfsEinzelstueckBeschreibung	937, 964	\liEntwurfsFabrikmethode	1066, 1091		
\liEntwurfsEinzelstueckCode	960, 970	\liEntwurfsFabrikmethodeAkteure	1041, 1068		
\liEntwurfsEinzelstueckUml	942, 966	\liEntwurfsFabrikmethodeUml	1024, 1067		
\liEntwurfsErbauer	1020	\liEntwurfsKompositum	1091		
\liEntwurfsErbauerAkteure	992, 1022	\liEntwurfsKompositumAkteure	1093		
\liEntwurfsErbauerUml	972, 1021	\liEntwurfsKompositumUml	1070, 1092		
\liEntwurfsFabrikmethode	1066, 1091	\liEntwurfsModellPraesentation	1112		
\liEntwurfsFabrikmethodeAkteure	1041, 1068	\liEntwurfsModellPraesentationAkteure	1114		
\liEntwurfsFabrikmethodeUml	1024, 1067	\liEntwurfsModellPraesentationBeschreibung	1095, 1113		
\liEntwurfsKompositum	1091	\liEntwurfsModellPraesentationCode	737, 747		
\liEntwurfsKompositumAkteure	1093	\liEntwurfsModellPraesentationUml	695, 745		
\liEntwurfsKompositumUml	1070, 1092	\liEntwurfsAdapter	794		
\liEntwurfsModellPraesentation	1112	\liEntwurfsAdapterAkteure	764, 796		
\liEntwurfsModellPraesentationAkteure	1114	\liEntwurfsAdapterCode	788, 797		
\liEntwurfsModellPraesentationBeschreibung	1095, 1113	\liEntwurfsAdapterUml	749, 795		
\liEntwurfsModellPraesentationCode	737, 747	\liEntwurfsBeobachter	865		
\liEntwurfsModellPraesentationUml	695, 745	\liEntwurfsBeobachterAkteure	823, 867		
\liEntwurfsStellvertreter	1136	\liEntwurfsBeobachterCode	857, 868		
\liEntwurfsStellvertreterCode	1130, 1138	\liEntwurfsBeobachterUml	799, 866		
\liEntwurfsStellvertreterUml	1116, 1137	\liEntwurfsDekorierer	897		
\liEntwurfsZustand	1177	\liEntwurfsDekoriererAkteure	899		
\liEntwurfsZustandAkteure	1155, 1179	\liEntwurfsDekoriererCode	889, 900		
\liEntwurfsZustandCode	1173, 1180	\liEntwurfsDekoriererUml	870, 898		
\liEntwurfsZustandUml	1140, 1178	\liEntwurfsEinfacheFabrik	933		
\liEpsilon	1235	\liEntwurfsEinfacheFabrikAkteure	917, 935		
\liErAttribute	1193, 1207, 1209	\liEntwurfsEinfacheFabrikUml	902, 934		
\liErDatenbankName	1212	\liEntwurfsEinzelstueck	963		
\liErEntity	1191, 1195, 1197	\liEntwurfsEinzelstueckAkteure	952, 968		
\liErledigt	2899	\liEntwurfsEinzelstueckBeschreibung	937, 964		
\liErMpAttribute	1206	\liEntwurfsEinzelstueckCode	960, 970		

<code>\liLadePakete</code>	59, 62, 229, 234, 496, 557, 1226, 1456, 1477, 1576, 2096, 2176, 2324, 2693, 2802
<code>\liLatexCode</code>	2730
<code>\liLeereZelle</code>	2134
<code>liLernkartei</code> (environ- ment)	1883
<code>\liLinksReduktion</code> . .	2201
<code>\liLinksReduktionInline</code>	2210, 2218
<code>\liMasterExkurs</code>	2065
<code>\liMasterFaelle</code> 2004, 2072	
<code>\liMasterFallRechnung</code>	2050
<code>\liMasterVariablen</code> . .	1981, 2067
<code>\liMasterVariablenDeklaration</code>	2027
<code>\liMasterWolframLink</code> 2075	
<code>\liMenge</code> . . . 76, 77, 79, 118, 119, 120, 124, 176, 177, 178, 182, 1227, 1276, 1315, 1316	
<code>\liMengeOhneMathe</code>	1227, 1230, 1232
<code>\liMetaSetze</code> 36, 48, 1519	
<code>\liMinimierungErklaerung</code>	2156
<code>\liMinispracheDatei</code> 2773	
<code>\linespread</code>	2588
<code>\liNichtsZuTun</code>	2900
<code>\liO</code>	1944, 1972
<code>\liOmega</code>	1934, 1978
<code>\liOmegaOhneMathe</code>	1934, 1939, 1941
<code>\liOOhneMathe</code>	1944, 1949, 1951
<code>\liParagraphMitLinien</code>	545, 630, 648, 2158, 2880, 2901, 2998, 3006, 3014
<code>\liPetriErreichKnotenDrei</code>	2318
<code>\liPetriErreichTransition</code>	2315
<code>\liPetriSetzeSchluessel</code>	2260
<code>\liPetriTransitionsName</code>	2307, 2319
<code>\liPetriTransitionsNameOhneMathe</code>	2307, 2310, 2312
<code>\liPetriTransPfeile</code> 2319	
<code>\liPolynomiellReduzierbar</code>	1596
<code>\liPotenzmenge</code>	1236, 1240, 2330
<code>\liPotenzmengeOhneMathe</code> . . .	1237, 1238, 1239
<code>\liPrimaer</code>	2579
<code>\liProblemBeschreibung</code>	1580
<code>\liProblemClique</code> . . .	1603
<code>\liProblemName</code>	1579, 1586, 1598, 1600, 1613, 1624, 1625, 1633, 1634
<code>\liProblemSat</code>	1632
<code>\liProblemSubsetSum</code>	1623, 1632
<code>\liProblemVertexCover</code>	1603, 1611
<code>\liProduktionen</code> 1275, 1317	
<code>liProduktionsRegeln</code> (environment)	1265
<code>liProjektSprache</code> (envi- ronment)	1803
<code>\liPseudoUeberschrift</code>	1788, 1838, 1839, 2144, 2154, 2995, 3003, 3011
<code>\liPumpingKontextfrei</code>	2376
<code>\liPumpingRegulaer</code> .	2355
<code>liQuellen</code> (environment)	1863
<code>\liRechtsReduktionInline</code>	2218
<code>\liRekursionsGleichung</code>	1968, 2031
<code>\liRelation</code>	2248
<code>liRelationenSchemaFormat</code> (environment)	2601
<code>\liRelationMenge</code> . . .	2595
<code>\liRichtig</code>	489
<code>liRmodell</code> (environment)	2581
<code>\liRundeKlammer</code> .	1921, 1925, 1935, 1945, 1959
<code>\liSetzeExamenTeilaufgabeNr</code>	1497
<code>\liSetzeExamenThemaNr</code>	1493
<code>\liSortierMarkierung</code> 2620	
<code>\liSortierPfeil</code>	2614
<code>\liSortierPfeilUnten</code> 2617	
<code>\liSpaltenUmbruch</code> . .	2689
<code>\liSqlCode</code>	2781
<code>\listen@punkt</code> . .	1863, 1875
<code>\liStrich</code>	1578
<code>\liSyntheseErklaerung</code>	2820, 2885
<code>\liSyntheseUeberErklaerung</code>	2883
<code>\liSyntheseUeberschrift</code>	2804, 2884
<code>\liT</code> 1954, 1969, 1983, 2047	
<code>\liTeilen</code>	2181
<code>\liLiteratur</code>	1745, 1769
<code>\liTheta</code>	1924, 1975, 2007, 2013, 2018
<code>\liThetaOhneMathe</code>	1924, 1929, 1931
<code>\liTOhneMathe</code>	1954, 1963, 1965
<code>\liTuringKante</code>	208
<code>\liTuringLeerzeichen</code>	164, 172
<code>\liTuringMaschine</code> . . .	165
<code>\liTuringUeberfuehrung</code>	211
<code>\liTuringUebergange</code>	203, 209
<code>\liTuringUebergangZelle</code>	198
<code>\liTypoUeberGROSS</code> . .	2934
<code>\liTypoUeberGross</code>	2928, 2934
<code>\liTypoUeberschrift</code>	2922, 2931, 2937
<code>\liUeberfuehrungsFunktion</code>	1241
<code>\liUeberfuehrungsFunktionOhneMathe</code> . . .	1241, 1244, 1246
<code>liUebergangsTabelle</code> (environment)	2142
<code>\liUeberschriftDreiecksTabelle</code>	2152
<code>\liUmlLeserichtung</code> .	2962
<code>\liVertauschen</code>	2608
<code>\liWortInSprache</code>	671
<code>\liWortNichtInSprache</code> 676	
<code>\liWpEquivalent</code>	3065
<code>\liWpErklaerung</code>	3068
<code>\liWpErklaerungVerzweigung</code>	3083
<code>\liWpKalkuel</code>	3045
<code>\liWpKalkuelOhneMathe</code>	3045, 3050, 3052, 3084, 3086, 3088
<code>\liZustandsBuchstabe</code>	1251, 1260, 1262, 1280, 1282
<code>\liZustandsBuchstabeGross</code> . . .	1252, 1261, 1263
<code>\liZustandsmenge</code> . . .	1239
<code>\liZustandsmengeNr</code>	1253, 2339
<code>\liZustandsmengeNrGross</code>	1261
<code>\liZustandsMengenSammlung</code>	2326
<code>\liZustandsMengenSammlungNr</code>	2335
<code>\liZustandsmengeOhneMathe</code>	1239
<code>\liZustandsname</code>	1262
<code>\liZustandsnameGross</code> . . .	1263, 2327, 2336
<code>\liZustandsnameTiefgestellt</code>	1278
<code>\liZustandsPaar</code>	2136
<code>\liZustandsPaarVariablenName</code> . . .	2135, 2138, 2139
<code>\llap</code>	2568
<code>\log</code>	1972, 1975, 1978, 2007, 2013
<code>\loop</code>	2401

<code>\lor</code>	3087	<code>\NewDocumentEnvironment</code>		<code>\pgfutil@repeat</code>	2647	
<code>\ltimes</code>	2083		1265, 1358, 1413, 1704, 1803, 1804, 1807, 1834, 1842, 1864, 1883, 1893, 2143, 2193, 2583, 2601	<code>\preceq</code>	1599	
M				<code>\prime</code>	1578	
<code>\makeatletter</code> ..	1879, 2640			<code>\printbibliography</code> .	1769	
<code>\makeatother</code> ..	1881, 2648			<code>\ProvidesPackage</code> 2, 15, 33, 58, 226, 425, 486, 493, 555, 668, 683, 1184, 1220, 1333, 1373, 1382, 1387, 1416, 1454, 1475, 1573, 1645, 1728, 1744, 1750, 1772, 1918, 2081, 2094, 2173, 2256, 2322, 2347, 2352, 2396, 2562, 2575, 2604, 2685, 2692, 2699, 2704, 2784, 2799, 2890, 2894, 2943, 2985, 3022, 3042		
<code>\marginpar</code>	1196, 1202, 1208, 1786	<code>\newlength</code>	3068			
<code>\mathbb</code> ..	1634, 2388, 3007	<code>\node</code>	572, 1712, 1717, 2441, 2446, 2451, 2457, 2463, 2469, 2620, 2665, 2979			
<code>\mathbin</code> .	2570, 2571, 2572					
<code>\mathcal</code> ..	1945, 2858, 2863, 2865, 2866, 2867	<code>\noexpand</code>	2529, 2530, 2531, 2550, 2665			
<code>\Mathe</code>	3062	<code>\noindent</code>	370, 673, 678, 1790, 1792, 1796, 1800, 1828, 1856, 1858, 1871, 1887, 1889, 1897, 2069, 2122, 2125, 2128, 2131, 2596, 2902, 2914, 3074			
<code>\MatheEnv</code> <u>3055</u> , 3063, 3066						
<code>\mathord</code>	2090, 2091					
<code>\mdfsetup</code>	1357, 1812, 1816, 1820, 1824					
<code>\medskip</code>	1588, 1797, 1799, 1873, 1901, 2586, 2592, 2909, 2913, 3056, 3060	<code>\nolinkurl</code>	403, 2735			
<code>\memph</code>	<u>1786</u>	<code>\normalsize</code>	1345			
<code>\mintinline</code>	2729, 2730, 2769, 2776, 2781	<code>\notin</code>	679			
<code>\mkern</code> ...	2570, 2571, 2572	<code>\null</code>	2907			
<code>\mlq</code>	2088, 2090	O				
<code>\mrq</code>	2088, 2091	<code>\o@join</code>	2567, 2570, 2571, 2572			
<code>\msg</code>	418	<code>\Omega</code>	1935			
<code>\myList</code>	2661, 2662, 2663, 2666	<code>\omega</code> 2357, 2358, 2378, 2379				
<code>\myNodes</code>	2650, 2665, 2671, 2675, 2677	<code>\or</code>	2478			
N		P				
<code>\NeedsTeXFormat</code> .	1, 14, 32, 57, 225, 424, 485, 492, 554, 667, 682, 1183, 1219, 1332, 1372, 1381, 1386, 1415, 1453, 1474, 1572, 1644, 1727, 1743, 1749, 1771, 1917, 2080, 2093, 2172, 2255, 2321, 2346, 2351, 2395, 2561, 2574, 2603, 2684, 2691, 2698, 2703, 2783, 2798, 2889, 2893, 2942, 2984, 3021, 3041	<code>\pagestyle</code>	1368, 1525			
<code>\neg</code>	3088	<code>\par</code> 369, 390, 549, 1297, 1305, 1528, 1532, 1539, 1541, 1546, 1550, 1554, 1795, 1857, 1880, 1888, 2520, 2542, 2557, 2598, 2884, 2908, 2912, 2920, 3073, 3079				
<code>\negthinspace</code>	1922	<code>\paragraph</code>	1345			
<code>\newcounter</code> ...	2415, 2416	<code>\parindent</code>	2589			
<code>\NewDocumentCommand</code> .	67, 108, 145, 166, 208, 229, 559, 576, 615, 622, 671, 676, 1286, 1308, 1596, 1719, 1747, 1906, 1910, 2237, 2248, 2315, 2738, 2748, 2758, 2770, 2773, 2777, 2962	<code>\path</code>	99, 146, 209, 587			
		<code>\pgfkeys</code> ..	2271, 2965, 2966, 2967, 2968, 2969, 2972, 2975, 2977			
		<code>\pgfmath@count</code>	2642, 2644, 2646			
		<code>\pgfmath@smuggleone</code>	2647			
		<code>\pgfmathdeclarefunction</code>	2641			
		<code>\pgfmathhint</code>	2642			
		<code>\pgfmathparse</code>	1427, 2659, 2664, 2667, 2681, 2682			
		<code>\pgfmathresult</code>	1428, 2642, 2643, 2645, 2647, 2660, 2668, 2681, 2682			
		<code>\pgfutil@empty</code>	2643			
		<code>\pgfutil@loop</code>	2644			
		Q				
		<code>\QS@list</code>	2522, 2533, 2537, 2544, 2550, 2555, 2558			
		<code>\QS@select@equal</code>	2497, 2501			
		<code>\QS@select@greater</code> ..	2498, 2502			
		<code>\QS@select@smaller</code> ..	2493, 2496, 2500			
		<code>\QS@sort@a</code>	2475, 2508, 2529, 2530			
		<code>\QS@sort@b</code>	2475, 2476			
		<code>\QS@sort@c</code>	2479, 2486			
		<code>\QS@sort@d</code>	2487, 2495			
		<code>\QS@sort@empty</code> .	2477, 2482			
		<code>\QS@sort@single</code> 2478, 2483				
		<code>\QS@initialize</code>	2400, 2512, 2547			
		<code>\QSIr</code> .	2483, 2489, 2497, 2517, 2531, 2536, 2539			
		<code>\QSIrr</code> 2518, 2531, 2532, 2540				
		<code>\QSLr</code>	2489, 2496, 2507, 2508, 2516, 2529, 2534, 2538			
		<code>\QSPivotStep</code>	2402, 2512, 2516, 2527			
		<code>\QSR</code>	2489			
		<code>\QSRr</code>	2498, 2519, 2530, 2535, 2541, 2550, 2551, 2552			
		<code>\QSSortStep</code>	2404, 2512, 2528, 2529			
		<code>\quad</code>	2100			
		R				
		<code>\raisebox</code>	1717			
		<code>\relax</code>	1781, 2489, 2532, 2534, 2535, 2536, 2644, 2646			
		<code>\renewcommand</code>	1348, 1349, 1350, 1351, 1354, 1355,			

<code>\TmpTransitionOne</code> 2261, 2282	<code>\umlinherit</code> 758, 809, 983, 1028, 1036	<code>\varepsilon</code> 502, 513, 514, 1235, 1972, 1978, 2010, 2021
<code>\TmpTransitionSeven</code> 2267, 2288	<code>\umlnote</code> ... 760, 985, 1152	<code>\vfill</code> ... 1534, 1565, 2689
<code>\TmpTransitionSix</code> 2266, 2287	<code>\umlreal</code> 756, 817	<code>\vrule</code> 2903, 2907
<code>\TmpTransitionTen</code> 2270, 2291	<code>\umlsimpleclass</code> 712, 713, 714, 718, 720, 721, 722, 751, 904, 905, 906, 974, 1026, 1027, 1118	<code>\vspace</code> ... 1530, 1544, 1548, 1552, 1562, 1843, 1861, 2916, 2918
<code>\TmpTransitionThree</code> 2263, 2284	<code>\umlstatic</code> 911, 945	X
<code>\TmpTransitionTwo</code> 2262, 2283	<code>\umluniagg</code> 981	<code>\xappto</code> .. 2665, 2671, 2675
<code>\TmpX</code> 2293	<code>\umluniassoc</code> 732, 757, 982, 1106, 1107, 1126, 1127	<code>\xdef</code> 1421
<code>\TmpY</code> 2294	<code>\umlVHuniassoc</code> .. 733, 734	<code>\xintApply</code> 2491
<code>\today</code> 1733	<code>\umlVHVdep</code> 726, 727, 729, 730, 913, 914	<code>\xintApplyUnbraced</code> .. 2490, 2496, 2497, 2498
<code>\ttfamily</code> 2581	<code>\umlVHVinherit</code> 709, 710, 715, 716, 723, 724, 882, 883, 907, 908, 1086, 1087, 1124, 1125	<code>\xintCSVtoList</code> 2550
U	<code>\umlVHVreal</code> 876, 877, 1147, 1148	<code>\xintFor</code> 2440, 2445, 2450, 2455, 2461, 2467, 2508
<code>\ul</code> 1192, 2579, 2580	<code>\UParrow</code> 2966	<code>\xintifEq</code> 2501
<code>\umlagg</code> 1150	<code>\url</code> 1907	<code>\xintifForLast</code> 2457, 2463, 2469
<code>\umlassoc</code> 1108	<code>\usemintedstyle</code> 2720	<code>\xintifGt</code> 2502
<code>\umlclass</code> 697, 701, 705, 752, 753, 754, 801, 806, 811, 814, 872, 873, 874, 879, 880, 909, 944, 975, 976, 979, 1030, 1033, 1072, 1078, 1079, 1097, 1098, 1099, 1120, 1121, 1122, 1142, 1143, 1144, 1145	<code>\usetikzlibrary</code> .. 61, 429, 1187, 1391, 1647, 2258, 2566, 2607, 2638, 2951, 3039	<code>\xintifLt</code> 2500
<code>\umldep</code> 1038		<code>\xintLength</code> 2475
<code>\umlHVHagg</code> 819, 885, 1088	V	<code>\xintntheft</code> 2487
	<code>\value</code> 2403	Z
		<code>\ZB</code> 1914
		<code>\zB</code> 1913
		<code>\zustandsnamen@liste</code> ... 1253, 1260, 1261