

lehramt-informatik

Hermine Bschlangaul *

September 6, 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	Hilfsmakros für Zeitpunkt-Tabelle	31
2.10	cyk-algorithmus.sty	33
2.10.1	Makro-Kürzel	33
2.10.2	TeX-Markup-Beispiel	33
2.11	entwurfsmuster.sty	34
2.11.1	Namensschema der Entwurfsmuster-Makros:	34
2.11.2	Reihenfolge	34
2.11.3	Abstrakte Fabrik (Abstract Factory)	34
2.11.4	Adapter	35
2.11.5	Beobachter (Observer)	37
2.11.6	Dekorierer (Decorator)	39
2.11.7	Einfache Fabrik (Simple Factory)	40
2.11.8	Einzelstück (Singleton)	41
2.11.9	Erbauer (Builder)	41
2.11.10	Fabrikmethode (Factory Method)	43
2.11.11	Kompositum (Composite)	44
2.11.12	Modell-Präsentation-Steuerung (Model-View-Controller)	45

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

2.11.13 Zustand (State)	46
2.12 er.sty	48
2.12.1 Tex-Markup-Beispiel: Komplettes Diagramm	48
2.12.2 Tex-Markup-Beispiel: EER Enhanced Entity-Relation-Modell nach Elmasri/Navante	49
2.12.3 Makro-Kürzel	49
2.13 formale-sprachen.sty	51
2.14 formatierung.sty	54
2.14.1 Schriftarten / Typographie	54
2.14.2 Farben	54
2.14.3 Überschriften	54
2.14.4 Listen	54
2.14.5 Kasten	54
2.14.6 Header	54
2.15 gantt.sty	55
2.16 grafik.sty	56
2.17 graph.sty	57
2.18 hanoi.sty	59
2.19 klassen-konfiguration-aufgabe.sty	60
2.20 klassen-konfiguration-examen.sty	61
2.21 komplexitaetstheorie.sty	63
2.21.1 Makro-Kürzel	63
2.22 kontrollflussgraph.sty	65
2.22.1 Makro-Kürzel	65
2.22.2 TeX-Markup-Beispiel	65
2.22.3 TikZ: pin	65
2.22.4 Umgebungen	66
2.22.5 Makros	67
2.23 kopf-fusszeilen.sty	68
2.24 literatur-dummy.sty	69
2.25 literatur.sty	70
2.26 makros.sty	71
2.27 master-theorem.sty	75
2.27.1 Makro-Kürzel	75
2.28 mathe.sty	79
2.29 minimierung.sty	80
2.30 normalformen.sty	83
2.30.1 Makro-Kürzel	83
2.31 petri.sty	86
2.31.1 Makro-Kürzel	86
2.32 potenzmengen-konstruktion.sty	88
2.33 pseudo.sty	90
2.34 pumping-lemma.sty	91
2.35 quicksort.sty	92
2.36 relationale-algebra.sty	95
2.37 rmodell.sty	96
2.37.1 Makro-Kürzel	96
2.38 sortieren.sty	97
2.39 spalten.sty	99
2.40 sql.sty	100
2.41 struktogramm.sty	101
2.42 syntax.sty	102
2.42.1 Makro-Kürzel	102
2.43 syntaxbaum.sty	104
2.44 synthese-algorithmus.sty	105
2.44.1 Makro-Kürzel	105
2.44.2 TeX-Markup Grundgerüst	105
2.44.3 TeX-Markup Linksreduktion	105

2.44.4	TeX-Markup Rechtreduktion	105
2.44.5	TeX-Markup Relationen formen	106
2.45	tabelle.sty	109
2.46	typographie.sty	110
2.47	uml.sty	112
2.48	vollstaendige-induktion.sty	114
2.48.1	Makro-Kürzel	114
2.49	wasserfall.sty	116
2.50	wpkalkuel.sty	117
2.50.1	Makro-Kürzel	117
3	Index	118

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
47   \tl_if_empty:NTF \g_auf_titel_tl {
48     \msg_fatal:nn { aufgabenmetadaten } { kein-titel }
49   } { }
50 }
```

`\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,
}
```

```
51 \def\liAufgabenMetadaten#1{
52   \liMetaSetze{#1}
53
54   \_gib_examen_titel: {}
55
56   \section{\_gib_aufgaben_titel:}
57 }
```

```
58 \ExplSyntaxOff
```

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

```
59 \def\liAufgabenTitel#1{}
```

```
60
```

2.4 automaten.sty

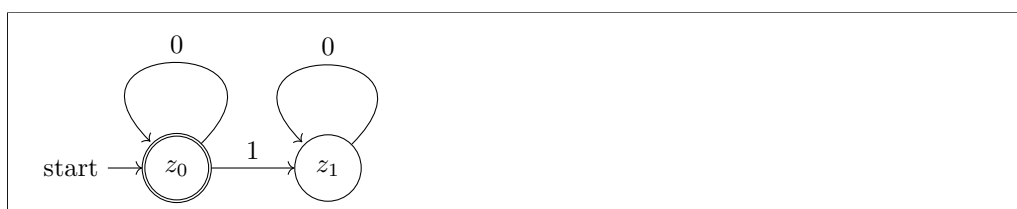
```
61 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
62 \ProvidesPackage{lehramt-informatik-automaten}[2021/02/14 Zum Setzen von Automaten]
```

2.4.1 Endlicher Automat

```
63 \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}
```



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

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

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

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

```

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

```

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

```

102 \def\liAutomatenKante#1#2#3#4{
103   \path (#1) edge[#4] node{#3} (#2);
104 }

105 \tikzset{
106   li automat/.style={
107     ->,
108     node distance=2cm
109   },
110 }

```

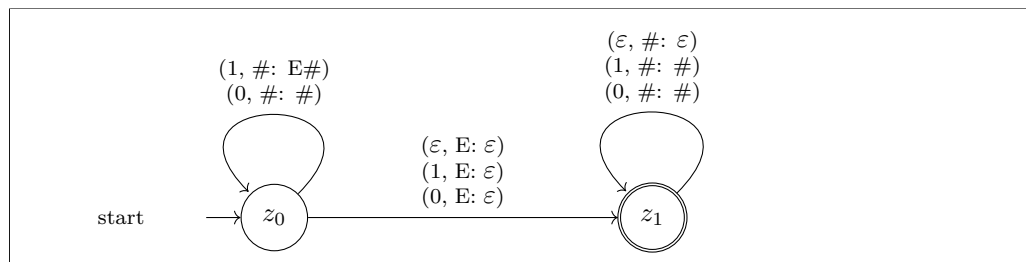
2.4.2 Kellerautomat

```
\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},
}
```

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

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

```

127   kellerboden .code:n = {\tl_set:Nn \l_kellerboden_tl {##1}},
128   ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
129 }
130
131 \keys_set:nn { kellerautomat } { #2 }
132
133 $#1 = (
134   \l_zustaende_tl,
135   \l_alphabet_tl,
136   \l_kelleralphabet_tl,
137   \l_delta_tl,
138   \l_start_tl,
139   \l_kellerboden_tl,
140   \l_ende_tl
141 )$
142 }
143 \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}`

```

144 \ExplSyntaxOn
145 \def\liKellerUebergang#1{
146   \directlua{automaten.drucke_keller_uebergaenge('#1')}
147 }
148 \ExplSyntaxOff

```

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

```

149 \NewDocumentCommand{\liKellerKante} { 0{above} m m m } {
150   \path (#2) edge[#1] node{\liKellerUebergang{#4}} (#3);
151 }

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

```

2.4.3 Turingmaschine

```

167 \RequirePackage{amssymb}

```

\liTuringLeerzeichen

□

```

168 \def\liTuringLeerzeichen{\Box}

```

```

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

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

```

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

```

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

```

\liTuringUebergangZelle Formatiert einen Zustandsübergang für eine Übergangstabelle.

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

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

```

202 \ExplSyntaxOn
203 \def\liTuringUebergangZelle#1{
204   \directlua{tex.print(automaten.gib_einen_turing_uebergang('#1'))}
205 }
206 \ExplSyntaxOff

```

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

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

`(z_1: \square, L)`

`(\square: \square, R)`


```

207 \ExplSyntaxOn
208 \def\liTuringUebergaenge#1{
209   \directlua{automaten.drucke_turing_uebergaenge('#1')}
210 }
211 \ExplSyntaxOff

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

212 \NewDocumentCommand{\liTuringKante} { 0{above} m m m } {
213   \path (#2) edge[#1] node{\liTuringUebergaenge{#4}} (#3);
214 }

\liTuringUeberfuehrung

215 \def\liTuringUeberfuehrung{
216    $\delta : Z \times \Gamma \rightarrow Z \times \Gamma \times \{ L, R, N \}$ 
217 }

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

```

2.5 basis.sty

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

231 \RequirePackage{xparse}

232 \ExplSyntaxOn

\liLadePakete

233 \NewDocumentCommand{\liLadePakete}{ m }
234 {
235   \clist_map_inline:nn { #1 } { \RequirePackage{lehramt-informatik-##1} }
236 }

\liLadeAllePakete

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

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.

```

281 \clist_new:N \g_auf_schluessel_clist
282 \clist_set:Nn \g_auf_schluessel_clist {
283   titel,
284   thematik,
285   stichwoerter,
286   zitat_schluessel,
287   zitat_beschreibung,
288   %
289   bearbeitungs_stand,
290   korrektheit,
291   %
292   relativer_pfad,
293   identische_aufgabe,
294   %
295   examen_nummer,
296   examen_jahr,
297   examen_monat,
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   ExamenJahr .tl_gset:N = \g_auf_examen_jahr_tl,
326   ExamenMonat .tl_gset:N = \g_auf_examen_monat_tl,
327   ExamenThemaNr .tl_gset:N = \g_auf_examen_thema_nr_tl,
328   ExamenTeilaufgabeNr .tl_gset:N = \g_auf_examen_teilaufgabe_nr_tl,
329   ExamenAufgabeNr .tl_gset:N = \g_auf_examen_aufgabe_nr_tl,
330 }
331 \cs_gset:Npn \setze_relativen_pfad: {
332   \tl_if_empty:NTF \g_auf_relativer_pfad_tl
333   {
334     \bool_if:nTF
335     {
336       ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&

```

```

337     ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
338     ! \tl_if_empty_p:N \g_auf_examen_monat_tl
339   }
340   {
341     \tl_gset:Nn \g_auf_relativer_pfad_tl {
342       Staatsexamen /
343       \g_auf_examen_nummer_tl /
344       \g_auf_examen_jahr_tl /
345       \g_auf_examen_monat_tl /
346       \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} { Thema - \g_auf_examen_thema_nr_tl / }
347       \tl_if_empty:NTF \g_auf_examen_telaufgabe_nr_tl {} { Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl / }
348       \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} { Aufgabe - \g_auf_examen_aufgabe_nr_tl / }
349     }
350   }
351   {}
352 }
353 {}
354 }

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

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

```

```

400     }{
401         \nolinkurl{\g_auf_relativer_pfad_tl}
402     }
403 }
404 }

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

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

\def\LehramtInformatikRepository{/pfad/zum/repository}

418 \input /etc/lehramt-informatik.config.tex
    biblatex not working with lualatex and babel

419 % \RequirePackage{polyglossia}
420 % \setmainlanguage{german}

421

```

2.6 baum.sty

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

425 \RequirePackage{tikz}

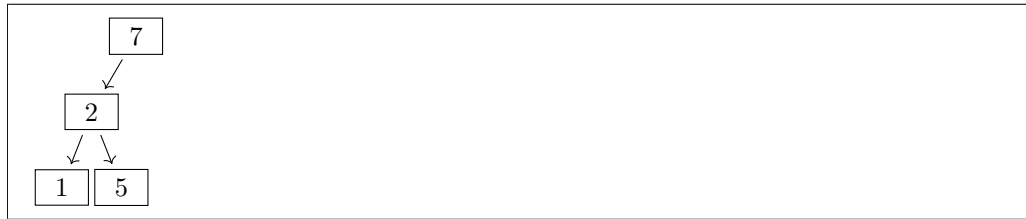
    für li binaer baum
426 \RequirePackage{tikz-qtrees}

    Für b baum
427 \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}
```



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

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}
```

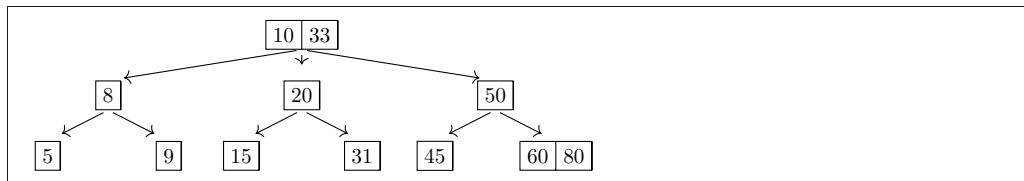


2.6.3 B-Baum

```

\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}

```



```

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

```

2.7 checkbox.sty

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

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

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

489
```

2.8 chomsky-normalform.sty

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

493 \ExplSyntaxOn

494 \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

```

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

```

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

```

507 \def\liChomsky@erklaerung@texte#1{
508   \str_case:nn {#1} {
509     %
510     {1} {
511       Alle-Regeln-der-Form- $A \rightarrow \varepsilon$ -werden-eliminiert.~
512       Die-Ersetzung-von- $A$ -wird-durch- $\varepsilon$ -in-allen-anderen-
513       Regeln-vorweggenommen.
514     }
515     {2} {
516       Jede-Produktion-der-Form- $A \rightarrow B$ -mit- $A, B$ -in- $S$ -wird-
517       als-Kettenregel-bezeichnet.~Diese-tragen-nicht-zur-Produktion-
518       von-Terminalzeichen-bei-und-lassen-sich-ebenfalls-eliminieren.
519     }
520     {3} {
521       Jedes-Terminalzeichen- $\sigma$ ,~das-in-Kombination-mit-anderen-
522       Symbolen-auftaucht,~wird-durch-ein-neues-Nonterminal-
523        $S_{\sigma}$ -ersetzt-und-die-Menge-der-Produktionen-durch-die-
524       Regel- $S_{\sigma} \rightarrow \sigma$ -ergänzt.
525     }
526     {4} {
527       Alle-Produktionen-der-Form-
528        $A \rightarrow B_1 B_2 \dots B_n$ -
529       werden-in-die-Produktionen-
530        $A \rightarrow$ 
531        $A_{n-1} B_n, A_{n-1} \rightarrow$ 
532        $A_{n-2} B_{n-1}, \dots,$ 
533        $A_2 \rightarrow B_1 B_2$ -zerteilt.~
534       Nach-der-Ersetzung-sind-alle-längeren-Nonterminalketten-
535       vollständig-heruntergebrochen-und-die-Chomsky-Normalform-erreicht.
536     }
537   }

```

```

538 }
539 \def\liChomskyErklaerung#1{
540   {
541     \itshape
542     \footnotesize
543     \liParagraphMitLinien{\li@chomsky@erklaerung@texte{#1}}
544   }
545 }

```

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

```

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

```

```

550 \ExplSyntaxOff
551

```

2.9 cpm.sty

```

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

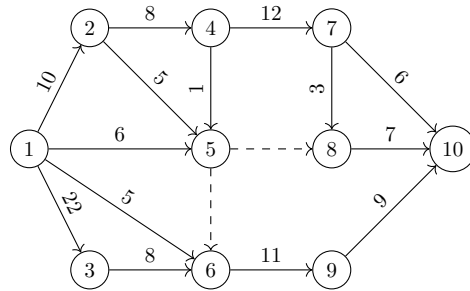
```

2.9.1 Makro-Kürzel

```

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

```



```

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

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

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

```

```

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

```

```

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

```

```

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

```

```

569
570 \node[circle,draw] (\l_name_tl) at (#3,#4) {#2};
571 }
572 \ExplSyntaxOff

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

```

2.9.2 Hilfsmakros für Zeitpunkt-Tabelle

```

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

```

```

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

```

```

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

```

```

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

```

```

610 \fi%
611 }

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

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

614

```


2.10 cyk-algorithmus.sty

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

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

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

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

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

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

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

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

```
629
```

2.11 entwurfsmuster.sty

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

2.11.1 Namensschema der Entwurfsmuster-Makros:

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

2.11.2 Reihenfolge

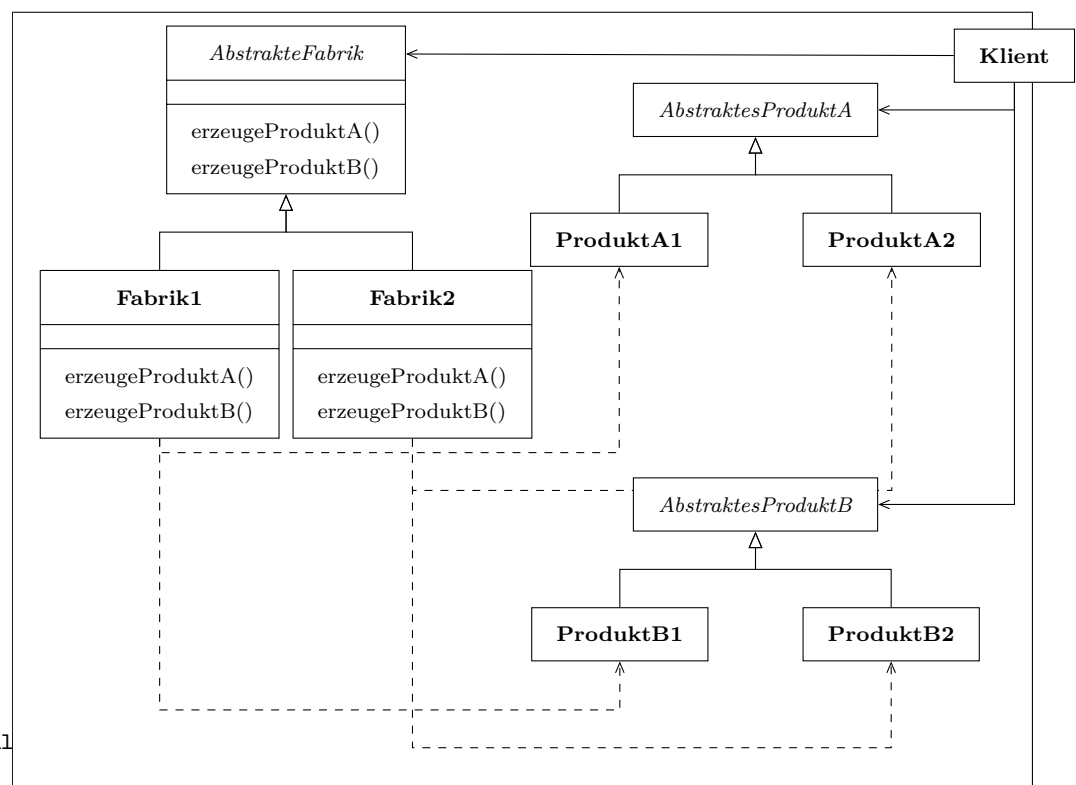
1. **Uml:** Uml-Klassendiagramm \liEntwurfsEinzelstueckUml
2. **Akteure:** Akteure, beteiligte Klassen \liEntwurfsEinzelstueckAkteure
3. **Code:** Allgemeines Code-Beispiel \liEntwurfsEinzelstueckCode
4. **ohne:** Ohne Suffix, Bündelung der einzelnen Makros eines Entwurfsmusters \liEntwurfsEinzelstueckAkteure

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

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

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

2.11.3 Abstrakte Fabrik (Abstract Factory)



```

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

```

```

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

```

iEntwurfsAbstrakteFabrikCode

```

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

```

\liEntwurfsAbstrakteFabrik

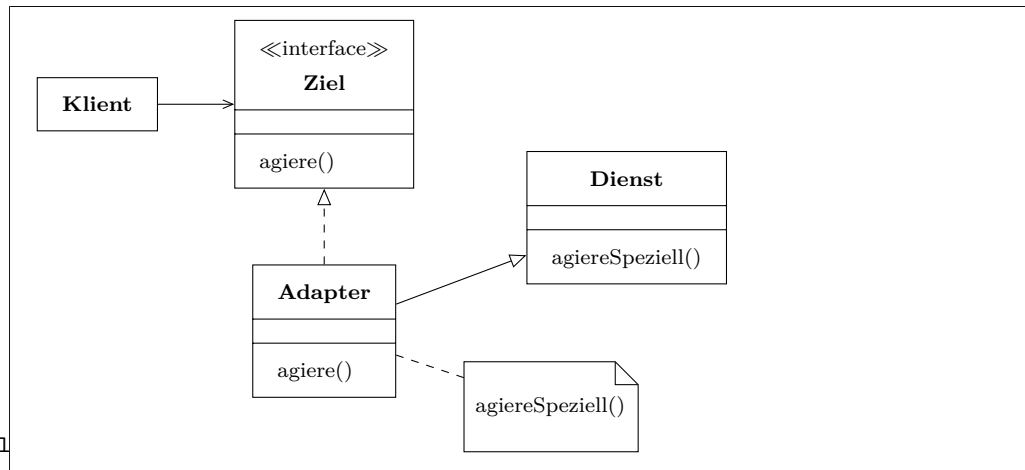
```

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

```

2.11.4 Adapter

\liEntwurfsAdapterUml



```

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

```

\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.

```

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

```

```

725
726 \end{description}
727 }

```

\liEntwurfsAdapterCode

```

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

```

\liEntwurfsAdapter

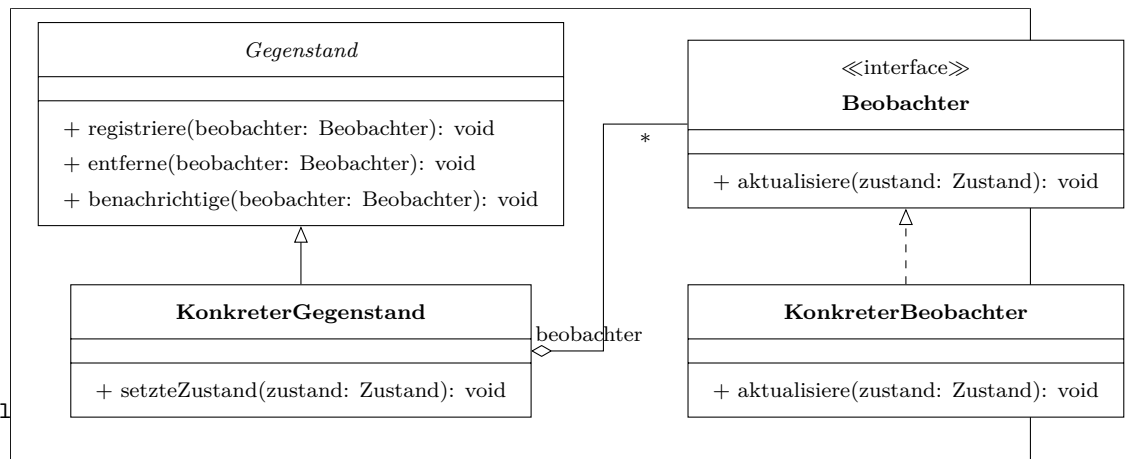
```

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

```

2.11.5 Beobachter (Observer)

\liEntwurfsBeobachterUml



```

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

```

\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.

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

\liEntwurfsBeobachterCode

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

```

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

```

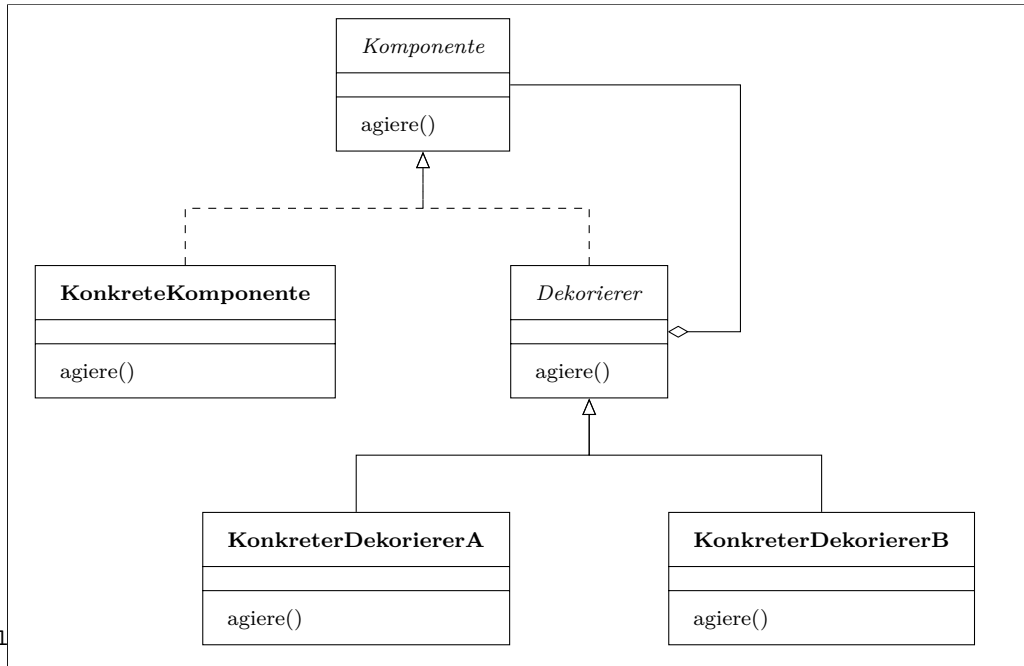
\liEntwurfsBeobachter

```

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

```

2.11.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

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

```

\liEntwurfsDekoriererCode

```

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

```

\liEntwurfsDekorierer

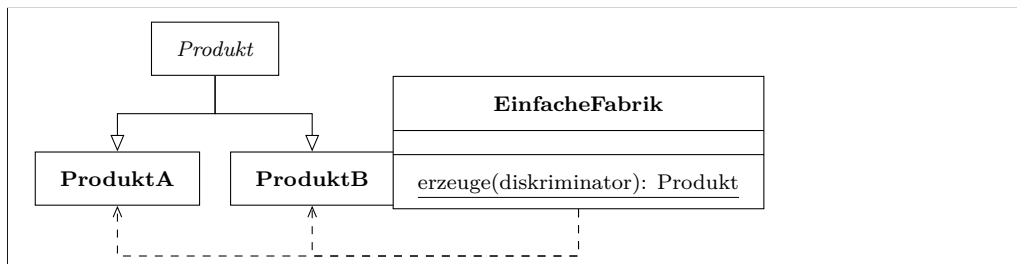
```

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

```

2.11.7 Einfache Fabrik (Simple Factory)

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



```

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

```

ntwurfsEinfacheFabrikAkteure Quelle: frei überstetzt aus GoF

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

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

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

```

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

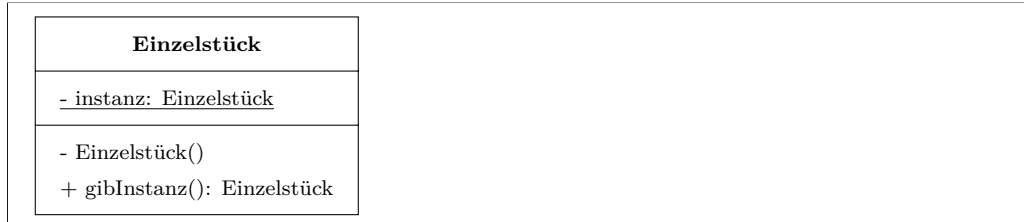
```


\liEntwurfsEinfacheFabrik

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

2.11.8 Einzelstück (Singleton)

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



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

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.

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

\liEntwurfsEinzelstueckCode

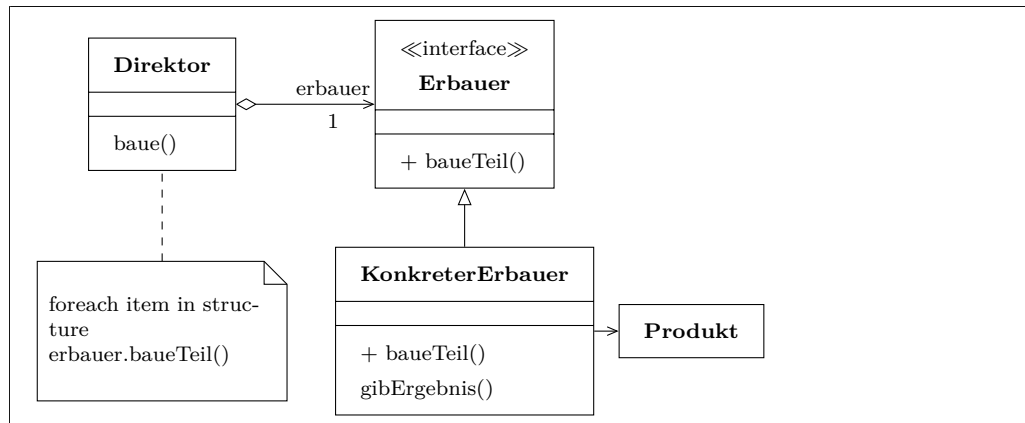
```
895 \def\liEntwurfsEinzelstueckCode{
896   \li@EntwurfsCode{einzelstueck}{Einzelstueck}
897 }
```

\liEntwurfsEinzelstueck

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

2.11.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

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

```

\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.

```

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

```

```

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

```

\liEntwurfsErbauer

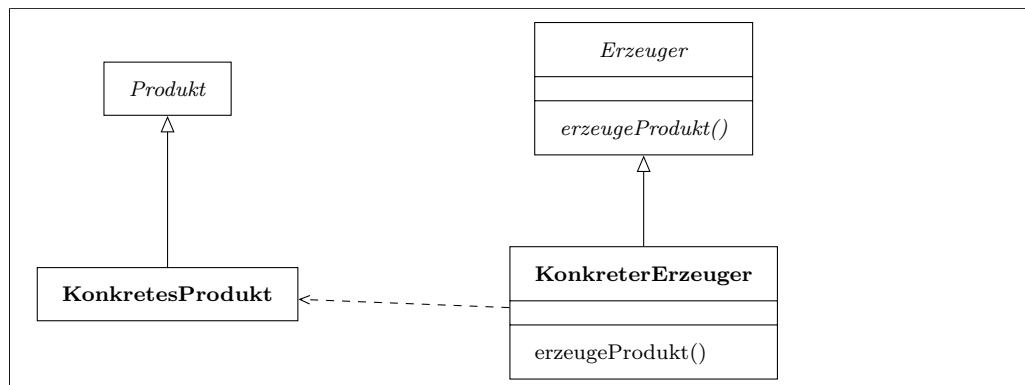
```

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

```

2.11.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

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

```

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).

```

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

```

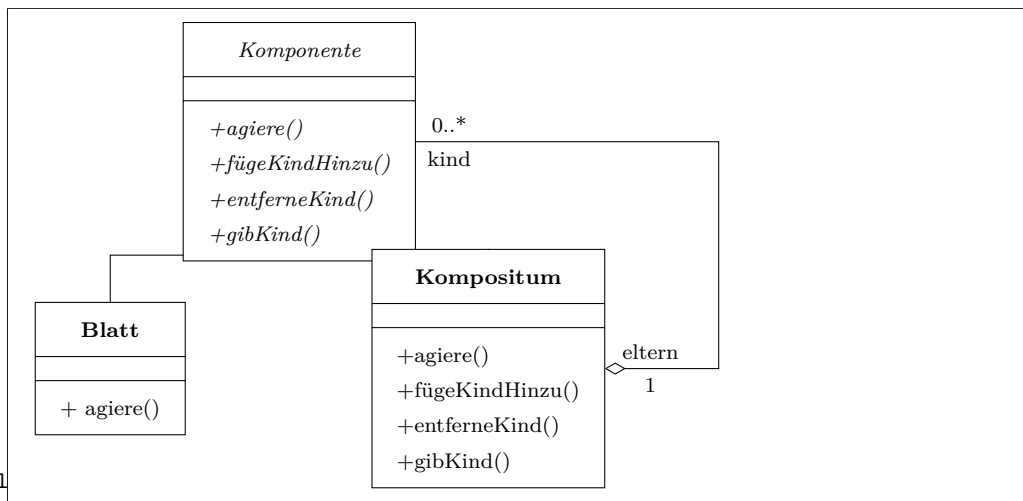
\liEntwurfsFabrikmethode

```

997 \def\liEntwurfsFabrikmethode{
998   \liEntwurfsFabrikmethodeUml
999   \liEntwurfsFabrikmethodeAkteure
1000 }

```

2.11.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

1001 \def\liEntwurfsKompositumUml{
1002   \begin{tikzpicture}
1003     \umlclass[x=2.5,y=3,type=abstract]{Komponente}{-}{
1004       \textit{+agiere()}\
1005       \textit{+fügeKindHinzu()}\
1006       \textit{+entferneKind()}\
1007       \textit{+gibKind()}
1008     }
1009     \umlclass[x=0]{Blatt}{-}{+ agiere()}
1010     \umlclass[x=5]{Kompositum}{-}{
1011       +agiere()\
1012       +fügeKindHinzu()\
1013       +entferneKind()\
1014       +gibKind()
1015     }
1016
1017     \umlVHVinherit{Kompositum}{Komponente}
1018     \umlVHVinherit{Blatt}{Komponente}
1019     \umlHVHaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,]
1020   \end{tikzpicture}
1021 }

```

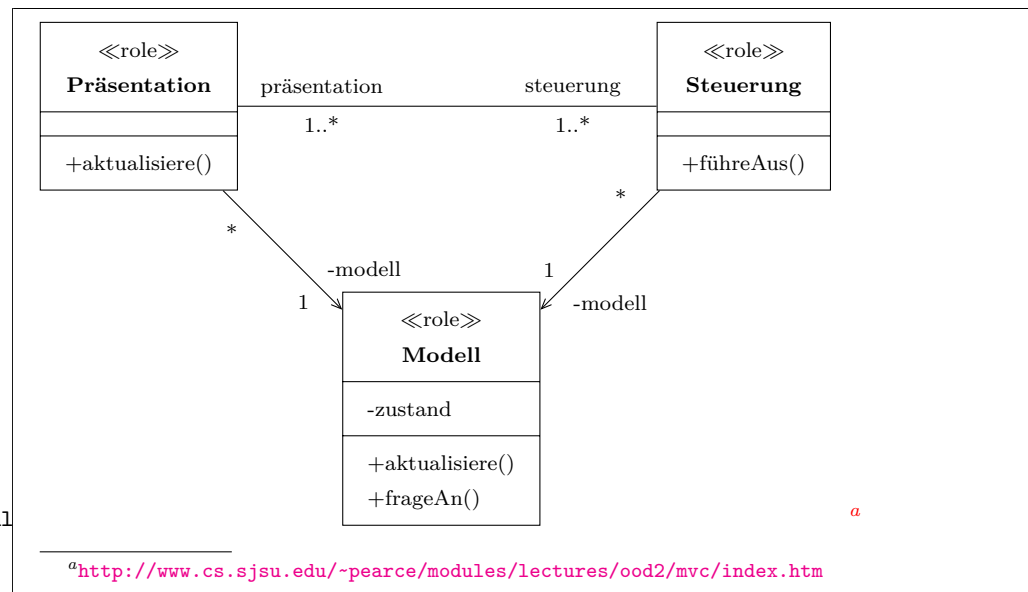
\liEntwurfsFabrikmethode

```

1022 \def\liEntwurfsKompositum{
1023   \liEntwurfsKompositumUml
1024   \liEntwurfsKompositumAkteure
1025 }

```

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



```

1026 \def\liEntwurfsModellPraesentationSteuerungUml{
1027   \begin{tikzpicture}
1028     \umlclass[x=-4cm,type=role]{Präsentation}{-}{+aktualisiere()}
1029     \umlclass[x=4cm,type=role]{Steuerung}{-}{+führeAus()}
1030     \umlclass[y=-4cm,type=role]{Modell}{-}{
1031       -zustand
1032     }{
1033       +aktualisiere()\
1034       +frageAn()
1035     }
1036
1037     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
1038     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}

```

```

1039 \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
1040 \end{tikzpicture}
1041 \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
1042 }

```

ModellPraesentationSteuerung

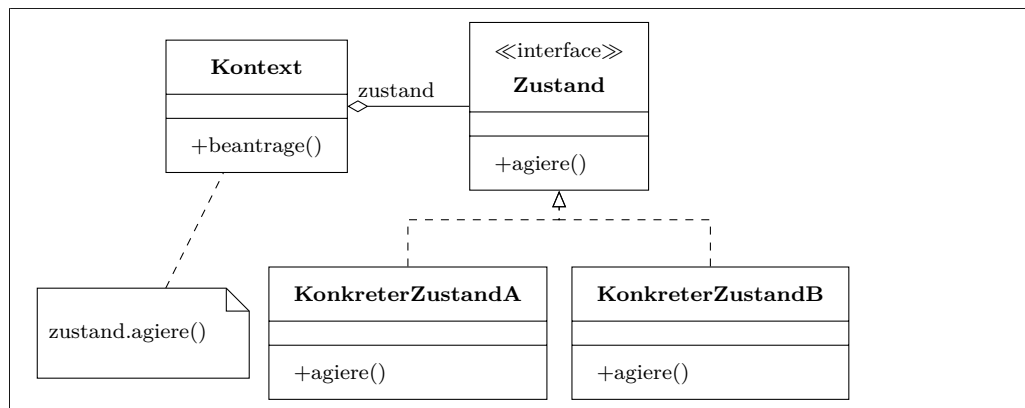
```

1043 \def\liEntwurfs{
1044 \liEntwurfsModellPraesentationSteuerungUml
1045 \liEntwurfsModellPraesentationSteuerungAkteure
1046 }

```

2.11.13 Zustand (State)

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



```

1047 \def\liEntwurfsZustandUml{
1048 \begin{tikzpicture}
1049 \umlcclass[x=-1,y=3]{Kontext}{+beantrage()}
1050 \umlcclass[x=3,y=3,type=interface]{Zustand}{+agiere()}
1051 \umlcclass[x=1,y=0]{KonkreterZustandA}{+agiere()}
1052 \umlcclass[x=5,y=0]{KonkreterZustandB}{+agiere()}
1053
1054 \umlVHVreal{KonkreterZustandA}{Zustand}
1055 \umlVHVreal{KonkreterZustandB}{Zustand}
1056
1057 \umlagg[reg[arg=zustand,pos=0.4]{Kontext}{Zustand}
1058
1059 \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
1060 \end{tikzpicture}
1061 }

```

\liEntwurfsZustandAkteure Quelle: Deutsche Wikipedia

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

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

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

```

1062 \def\liEntwurfsZustandAkteure{
1063 \begin{description}
1064 \item[Kontext (Context)]
1065
1066 definiert die clientseitige Schnittstelle und verwaltet die separaten
1067 Zustandsklassen.
1068

```

```

1069 \item[State (Zustand)]
1070
1071 definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
1072 implementiert gegebenenfalls ein Standardverhalten.
1073
1074 \item[KontreterZustand (ConcreteState)]
1075
1076 implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
1077 verbunden ist.
1078 \end{description}
1079 }

```

\liEntwurfsZustand

```

1080 \def\liEntwurfsZustand{
1081 \liEntwurfsZustandUml
1082 \liEntwurfsZustandAkteure
1083 }
1084

```

2.12 er.sty

```
1085 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1086 \ProvidesPackage{lehramt-informatik-er}[2020/06/13 Zum Setzen von
1087 ER-Diagrammen]
```

```
1088 \RequirePackage{tikz-er2}
1089 \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);

```

```
1090 \RequirePackage{soul}
```

```
1091 \RequirePackage{fontawesome}
```

2.12.3 Makro-Kürzel

```

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

```

```
1092 \ExplSyntaxOn
```

```
\liErEntity
```

```
1093 \def\liErEntity#1{\textbf{#1}}
```

```
\liErRelationship
```

```
1094 \def\liErRelationship#1{\ul{#1}}
```

```
\liErAttribute
```

```
1095 \def\liErAttribute#1{\emph{#1}}
```

```
\liErMpEntity
```

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

```

1096 \def\liErMpEntity#1{
1097   \liErEntity{#1}
1098   \marginpar{
1099     \liErEntity{\tiny\faSquareO{~E:~#1}
1100   }
1101 }

```

□

```
\liErMpRelationship
```

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

```

1102 \def\liErMpRelationship#1{
1103   \liErRelationship{#1}
1104   \marginpar{
1105     \liErRelationship{\tiny\faGg{~R:~#1}
1106   }
1107 }

```

```
\liErMpAttribute
```

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

```

1108 \def\liErMpAttribute#1{
1109   \liErAttribute{#1}
1110   \marginpar{
1111     \liErAttribute{\tiny\faCircleThin{~A:~#1}
1112   }
1113 }

```

```

\liErDatenbankName Let-Abkürzung: \let\d=\liErDatenbankName
                    datenbank name
1114 \def\liErDatenbankName#1{
1115   {
1116     \footnotesize\texttt{(#1)}
1117   }
1118 }

1119 \ExplSyntaxOff
1120

```

2.13 formale-sprachen.sty

```

1121 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1122 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
1123 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]

1124 \directlua{
1125   formale_sprachen = require('lehramt-informatik-formale-sprachen')
1126 }

1127 \RequirePackage{hyperref}

1128 \liLadePakete{mathe,typographie}

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

1129 \def\liMengeOhneMathe#1{\{ #1 \}}
1130 \def\liMenge#1{%
1131   \ifmode%
1132     \liMengeOhneMathe{#1}%
1133   \else%
1134     $\liMengeOhneMathe{#1}$%
1135   \fi%
1136 }

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

1137 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

1138 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1139 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1140 \def\liPotenzmenge#1{$\liPotenzmengeOhneMathe{#1}$}

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

1141 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
1142 \let\liZustandsmenge=\liPotenzmenge

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

1143 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1144 \def\liUeberfuehrungsFunktion#1{
1145   \ifmode
1146     \liUeberfuehrungsFunktionOhneMathe{#1}
1147   \else
1148     $\liUeberfuehrungsFunktionOhneMathe{#1}$
1149   \fi
1150 }

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

1151 \def\liAlphabet#1{$\Sigma = \{ #1 \}$}

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

1152 \def\liBandAlphabet#1{$\Gamma = \Sigma \cup \{ #1 \}$}

\liZustandsBuchstabe

1153 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1154 \def\liZustandsBuchstabeGross{Z}

```

\liZustandsmengeNr

```

1155 \def\zustandsnamens@liste#1#2{
1156   $
1157   \{
1158     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1159   \}
1160   $
1161 }
1162 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

```

\liZustandsmengeNrGross

```

1163 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

```

\liZustandsname

```

\liZustandsname{1}: $z_1$
1164 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

```

\liZustandsnameGross

```

\liZustandsnameGross{1}: $Z_1$
1165 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

```

\liAbleitung

```

\liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1166 \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}

1167 \NewDocumentEnvironment { liProduktionsRegeln }
1168 { 0{P} +b }
1169 {
1170   \liGeschweifteKlammern{#1}
1171   {
1172     \begin{align*}
1173       \directlua{formale_sprachen.produktions_regeln('#2')}
1174     \end{align*}
1175   }{-0.2cm}{-1.5cm}
1176 } {}

```

\liProduktionen

```

\liProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
1177 \def\liProduktionen#1{
1178   \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
1179 }

```

\liZustandsnameTiefgestellt

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

```

1180 \def\liZustandsnameTiefgestellt#1{
1181   \ifmmode
1182     \liZustandsBuchstabe\sb{#1}
1183   \else
1184     $\liZustandsBuchstabe\sb{#1}$
1185   \fi
1186 }

```

```

1187 \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}

```

```

1188 \NewDocumentCommand{ \liAusdruck } { O{L} m m } {
1189   $
1190   \tl_if_empty:nTF {#1} {} {#1 =}
1191   \{
1192     \, #2 \,
1193     |
1194     \, #3 \,
1195   \}$
1196 }
1197 \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

1198 \def\liFlaci#1{%
1199   \par
1200   {%
1201     \scriptsize
1202     Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1203     Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1204     Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1205     \href{https://flaci.com/#1}{flaci.com/#1}%
1206   }%
1207   \par
1208 }

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

1209 \ExplSyntaxOn
1210 \NewDocumentCommand {\liGrammatik} { O{G} m } {
1211   \tl_set:Nn \l_variablen_tl {V}
1212   \tl_set:Nn \l_alphabet_tl {\Sigma}
1213   \tl_set:Nn \l_produktionen_tl {P}
1214   \tl_set:Nn \l_start_tl {S}
1215
1216   \keys_define:nn { grammatik } {
1217     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1218     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1219     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1220     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1221   }
1222
1223   \keys_set:nn { grammatik } { #2 }
1224
1225   $#1 = (
1226     \l_variablen_tl,
1227     \l_alphabet_tl,
1228     \l_produktionen_tl,
1229     \l_start_tl
1230   )$
1231 }
1232 \ExplSyntaxOff
1233

```

2.14 formatierung.sty

```
1234 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1235 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

2.14.1 Schriftarten / Typographie

```
1236 \RequirePackage{mathpazo}
1237 \RequirePackage[no-math]{fontspec}
1238 \setmainfont{texgyrepagella}
1239 \setsansfont{QTAncientOlive}
1240 \RequirePackage{sectsty}
1241 \allsectionsfont{\fontspec{QTAncientOlive}}
```

2.14.2 Farben

```
1242 \RequirePackage{xcolor}
1243 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

2.14.3 Überschriften

```
1244 \RequirePackage{titlesec}
1245 \titleformat{\chapter}[display]{\sffamily\bfseries}{\Opt}{\LARGE}
1246 \titlespacing{\chapter}{0pt}{0pt}{*1}
1247 \titleformat{\paragraph}[hang]{\sffamily\normalsize\bfseries}{\theparagraph}{1em}{}
1248 \setcounter{secnumdepth}{0}
```

2.14.4 Listen

```
1249 \RequirePackage{paralist}
1250 \renewcommand\labelitemi{-}
1251 \renewcommand\labelitemii{-}
1252 \renewcommand\labelitemiii{-}
1253 \renewcommand\labelitemiv{-}
1254 % Counter: enumi enumii enumiii enumiv
1255 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1256 \renewcommand{\labelenumi}{(\alph{enumi})}
1257 \renewcommand{\labelenumii}{(\roman{enumii})}
```

2.14.5 Kasten

```
1258 \RequirePackage{mdframed}
1259 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1260 \NewDocumentEnvironment { liKasten }{ } {
1261   \begin{mdframed}[backgroundcolor=white!0]
1262 } {
1263   \end{mdframed}
1264 }
```

2.14.6 Header

```
1265 \RequirePackage{fancyhdr}
1266 \fancyhead[L,C,R]{}
1267 \fancyfoot[L]{}
1268 \fancyfoot[C]{}
1269 \fancyfoot[R]{\thepage}
1270 \pagestyle{fancy}
1271 \renewcommand{\headrulewidth}{0pt}
1272 \renewcommand{\footrulewidth}{0pt}
1273
```

2.15 gantt.sty

```

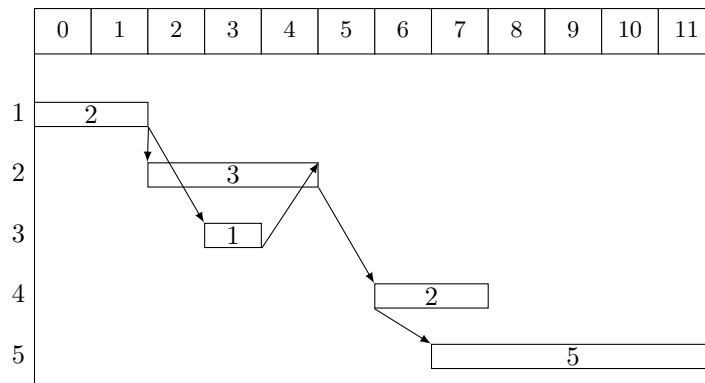
1274 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1275 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\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}

```



```

1276 \RequirePackage{tikz-uml}
1277 \RequirePackage{pgfgantt}
1278 \setganttlinklabel{f-s}{}
1279 \setganttlinklabel{s-s}{}
1280 \setganttlinklabel{f-f}{}
1281 \setganttlinklabel{s-f}{}
1282

```

2.16 grafik.sty

```
1283 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1284 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1285 TikZ, um Grafiken zeichnen zu können.]
1286 \RequirePackage{tikz}
1287
```


2.17 graph.sty

```

1288 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1289 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1290 \RequirePackage{tikz}

```

Für die die Adjazenzliste (`xrightrightarrow`)

```

1291 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

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

```

1292 \RequirePackage{blkarray}
1293 \usetikzlibrary{arrows.meta}

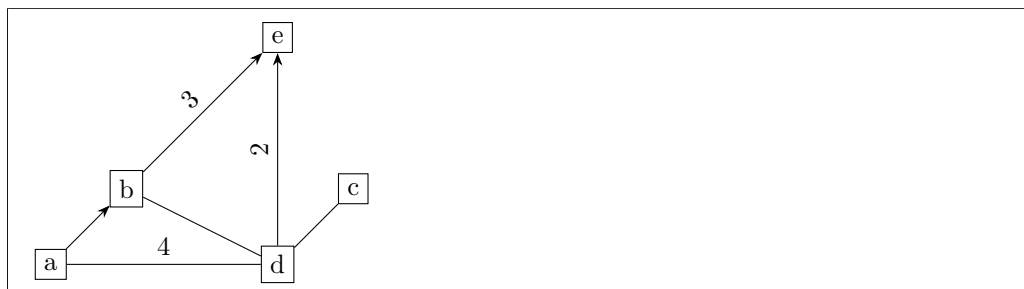
```

```

\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}

```



```

1294 \tikzset{
1295   li graph/.style={
1296     every node/.style={
1297       rectangle,
1298       draw,
1299     },
1300     every edge/.style={
1301       >={Stealth[black]},
1302       draw,
1303     },
1304     every edge/.append style={
1305       every node/.style={
1306         sloped,
1307         auto,
1308       }
1309     }
1310   },
1311   li markierung/.style={
1312     ultra thick,
1313   }
1314 }

```

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}

```

```

1315 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1316

```

2.18 hanoi.sty

```
1317 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1318 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
1319 von Hanoi-Grafiken]
```

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

```
1320 \RequirePackage{tikz}
1321 \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}

1322 \def\li@mset #1[#2]=#3{%
1323 \expandafter\xdef\csname #1#2\endcsname{#3}
1324 }
1325 \def\li@mget #1[#2]{%
1326 \csname #1#2\endcsname
1327 }
1328 \def\li@minc #1[#2] += #3{%
1329 \pgfmthparse{\li@mget #1[#2] + #3}%
1330 \li@mset #1[#2] = \pgfmthresult
1331 }
1332
1333 \def\liHanoi#1#2{
1334   \edef\li@numdiscs{#1}
1335   \def\li@sequence{#2}
1336   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1337     % init colors
1338     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purple!80!black}{
1339       \li@mset col[\j]={\c};
1340     }
1341     % draw poles and init pole counters
1342     \foreach \j in {1,2,3}{
1343       \li@mset pos[\j]=0
1344       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1345     }
1346     % draw base
1347     \draw (.5,-.5) -- +(3,0);
1348     % draw discs
1349     \foreach[count=\k] \i/\j in \li@sequence{
1350       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*\i/\li@numdiscs,0);
1351       \li@minc pos[\j] += {.5}
1352     }
1353   \end{tikzpicture}
1354 }
```

2.19 klassen-konfiguration-aufgabe.sty

```
1355 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1356 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-aufgabe}[2021/09/01 Die
1357 Klasse Aufgabe konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]

    Formatierung muss vor literatur sein, sonst Option clash
1358 \liLadePakete{
1359   formatierung,
1360   abmessung,
1361   literatur-dummy,
1362   makros,
1363   aufgaben-metadaten,
1364   kopf-fusszeilen,
1365   mathe
1366 }
1367
1368 % Komischer Option-Clash deshalb ganz am Ende, für die Silbentrennung
1369 \RequirePackage[ngerman]{babel}
1370 \ExplSyntaxOn
1371 \AddToHook{enddocument}{
1372   \_gib_github_url:
1373 }
1374 \ExplSyntaxOff
1375
```

2.20 klassen-konfiguration-examen.sty

```
1376 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1377 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-examen}[2021/09/04 Die
1378 Klasse liSetzeExamen konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

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

```
1379 \liLadePakete{
1380   formatierung,
1381   literatur-dummy,
1382   makros,
1383   aufgaben-metadaten,
1384   abmessung,
1385   typographie
1386 }
```

Formatierung für die Überschriften setzen.

```
1387 \RequirePackage{titlesec}
1388 \titleformat{\section}{\sffamily\huge\filcenter\bfseries}{\thesection}{1em}{}
1389 \newcommand{\sectionbreak}{\clearpage}
1390 \setcounter{secnumdepth}{0}
1391 \liLadeAllePakete
```

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

```
1392 \RequirePackage[ngerman]{babel}
1393 \RequirePackage{standalone}
1394 \ExplSyntaxOn
```

`\liSetzeExamen`

```
1395 \def\liSetzeExamen#1#2#3{
1396   \tl_gset:Nn \g_auf_examen_nummer_tl { #1 }
1397   \tl_gset:Nn \g_auf_examen_jahr_tl { #2 }
1398   \tl_gset:Nn \g_auf_examen_monat_tl { #3 }
1399 }
```

`\liSetzeExamenThemaNr`

```
1400 \def\liSetzeExamenThemaNr#1{
1401   \tl_gset:Nn \g_auf_examen_thema_nr_tl { #1 }
1402   \section{Thema-Nr.~#1}
1403 }
```

`\liSetzeExamenTeilaufgabeNr`

```
1404 \def\liSetzeExamenTeilaufgabeNr#1{
1405   \tl_gset:Nn \g_auf_examen_teilaufgabe_nr_tl { #1 }
1406   \subsection{Teilaufgabe-Nr.~#1}
1407 }
```

`\liBindeAufgabeEin`

```
1408 \def\liBindeAufgabeEin#1{
1409   \tl_gset:Nn \g_auf_examen_aufgabe_nr_tl { #1 }
1410   \input{
1411     \LehramtInformatikRepository /
1412     Staatsexamen /
1413     \g_auf_examen_nummer_tl /
1414     \g_auf_examen_jahr_tl /
1415     \g_auf_examen_monat_tl /
1416     \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
1417       Thema - \g_auf_examen_thema_nr_tl /
1418     }
1419     \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
1420       Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl /
1421     }
1422     Aufgabe - \g_auf_examen_aufgabe_nr_tl .tex
1423   }
1424 }
```

```

\liAufgabenMetadaten Das Metadaten-Makro überschreiben
1425 \def\liAufgabenMetadaten#1{
1426   \liMetaSetze{#1}
1427   \subsubsection{\_gib_aufgaben_titel:}
1428 }

1429 \setcounter{tocdepth}{4}
1430 \AddToHook{begindocument}{
1431   \begin{center}
1432     \liTypoUeberGROSS{Erste-Staatsprüfung-für-ein-Lehramt-an-öffentlichen-Schulen}
1433   \end{center}
1434
1435   \vfill
1436
1437   \begin{center}
1438     \liTypoUeberGROSS{\g_auf_examen_nummer_tl}
1439   \end{center}
1440
1441   \begin{center}
1442     \liTypoUeberGROSS{\g_auf_examen_jahr_tl}
1443   \end{center}
1444
1445   \begin{center}
1446     \liTypoUeberGROSS{\g_auf_examen_monat_tl}
1447   \end{center}
1448
1449   \vfill
1450   \tableofcontents
1451   \clearpage
1452 }

1453 \ExplSyntaxOff
1454

```

2.21 komplexitaetstheorie.sty

```
1455 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1456 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1457 Setzen von Karp's NP-vollständigen Problemen, Hilfsmakros für die
1458 Polynomialzeitreduktion.]
```

2.21.1 Makro-Kürzel

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

```
1459 \liLadePakete{mathe}
```

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

```
1460 \RequirePackage{mdframed}
```

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

```
1461 \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`

```
1462 \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`

```
1463 \def\liProblemBeschreibung#1#2#3{
1464   \begin{mdframed}[
1465     userdefinedwidth=9cm,
1466     align=center,
1467     backgroundcolor=white!0,
1468   ]
1469     \centerline{\large\liProblemName{#1}}
1470
1471     \medskip
1472
1473     \begin{description}
1474       \item[Gegeben:] #2
1475       \item[Frage:] #3
1476     \end{description}
1477   \end{mdframed}
1478 }
```

```

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

1479 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1480 \begin{displaymath}
1481   \liProblemName{#1}
1482   \preceq_{#2}
1483   \liProblemName{#3}
1484 \end{displaymath}
1485 }

\liProblemVertexCover

1486 \def\liProblemClique{%
1487 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1488 Mindestgröße  $n$  in einem gegebenen Graphen.
1489 \footcite[Seite 76]{theo:fs:4}
1490 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1491 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1492 \footcite{wiki:cliquenproblem}
1493 }

\liProblemVertexCover

1494 \def\liProblemVertexCover{%
1495 %
1496 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1497 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1498 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1499 \footcite{wiki:knotenueberdeckung}
1500
1501 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1502 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1503 mindestens einem Knoten aus  $U$  verbunden ist.
1504 \footcite[Seite 78]{theo:fs:4}%
1505 }

\liProblemSubsetSum Kein Karp-21-Problem

1506 \def\liProblemSubsetSum{%
1507 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1508 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1509 \footcite{wiki:teilsommenproblem}
1510 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1511 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1512 größer als eine gegebene obere Schranke  $c$  ist.
1513 \footcite[Seite 74]{theo:fs:4}
1514 }

\liProblemSubsetSum Kein Karp-21-Problem

1515 \def\liProblemSat{%
1516 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1517 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1518 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1519 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1520 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1521 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1522 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1523 aufgestellt werden.
1524 \footcite[Seite 71]{theo:fs:4}
1525 }

1526

```


2.22 kontrollflussgraph.sty

1527 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1528 \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 = ...}]
```

```

1529 \RequirePackage{tikz}
1530 \usetikzlibrary{positioning}
1531 \tikzset{
1532   li kontrollfluss/.style={
1533     knoten/.style={
1534       circle,
1535       draw
1536     },
1537     usebox/.style={
1538       draw,
1539       rectangle,
1540       font=\scriptsize,
1541       anchor=west,
1542       align=left,
1543     },
1544     bedingung/.style={
1545       midway,
1546       draw=none,
1547       font=\scriptsize
1548     },
1549     knotenbeschriftung/.style={
1550       draw,
1551       rectangle,
1552       midway,
1553       font=\scriptsize
1554     },
1555     wahr/.style={
1556       thick
1557     },
1558     falsch/.style={
1559       dashed
1560     },
1561     every node/.style={
1562       circle,
1563       draw,
1564     },
1565     every edge/.append style={
1566       every node/.style={
1567         draw=none,
1568         bedingung,
1569       }
1570     },
1571     every path/.style={
1572       draw,
1573       ->,
1574     },
1575     every pin/.style={
1576       draw,
1577       dotted,
1578       rectangle,
1579       pin position=right
1580     },
1581     every pin edge/.style={
1582       dotted,
1583       arrows=-,
1584     }
1585   }
1586 }

```

2.22.4 Umgebungen

liKontrollflussgraph

```

1587 \NewDocumentEnvironment { liKontrollflussgraph } { 0{} } {

```

```

1588 \begin{tikzpicture}[
1589     li kontrollfluss,
1590     #1
1591 ]
1592 } {
1593 \end{tikzpicture}
1594 }

```

2.22.5 Makros

`\liAnweisung`

```
1595 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}
```

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

```
1596 \def\liBedingung#1#2{\node[bedingung,#1]{\texttt{#2}}}
```

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

```
1597 \def\liBedingungWahr#1{\node[bedingung,#1]{\texttt{[true]}}}
```

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

```
1598 \def\liBedingungFalsch#1{\node[bedingung,#1]{\texttt{[false]}}}
```

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

```
1599 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}
```

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

```
1600 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw
```

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

```
1601 \ExplSyntaxOn
```

```
1602 \NewDocumentCommand { \liKontrollKnotenPfad } { m }
```

```
1603 {
```

```
1604   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
```

```
1605   \seq_set_map:Nnn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
```

```
1606   \seq_use:Nn \l_tmpa_seq { ~~~~ }
```

```
1607 }
```

```
1608 \ExplSyntaxOff
```

```
1609
```

2.23 kopf-fusszeilen.sty

```
1610 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1611 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1612 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1613 \ExplSyntaxOn

1614 \fancyhead{}
1615 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1616 \fancyhead[L0,RE]{\scriptsize\today}
1617 \fancyfoot{}
1618 \fancyfoot[LE,R0]{\thepage}
1619 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1620 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1621 \renewcommand{\headrulewidth}{0.4pt}
1622 \renewcommand{\footrulewidth}{0.4pt}
1623 \setlength{\headwidth}{\textwidth}

1624 \ExplSyntaxOff

1625
```

2.24 literatur-dummy.sty

```
1626 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1627 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1628 \def\literatur{}

\footcite

1629 % \RequirePackage[stable,multiple]{footmisc}
1630 \NewDocumentCommand{ \footcite } { o m }{}

1631
```

2.25 literatur.sty

```
1632 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1633 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1634 \RequirePackage{csquotes}
1635 \RequirePackage[
1636   bibencoding=utf8,
1637   citestyle=authortitle,
1638   backend=biber,
1639 ]{biblatex}
1640 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1641 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1642 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1643 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1644 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1645 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1646 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1647 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1648 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1649 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1650 % To allow footnotes in the heading
1651 \RequirePackage[stable,multiple]{footmisc}

\literatur

1652 \def\literatur{\printbibliography[heading=subbibliography]}

1653
```

2.26 makros.sty

```

1654 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1655 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1656 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1657 anderen Paket passen]
1658 \RequirePackage{hyperref}
1659 \RequirePackage{graphicx}

Für die Umgebung liQuellen benötigt.
1660 \RequirePackage{paralist}

\inhaltsverzeichnis
1661 \def\inhaltsverzeichnis {
1662   \begin{mdframed}
1663     \begin{group}
1664       \let\clearpage\relax
1665       \tableofcontents
1666     \end{group}
1667   \end{mdframed}
1668 }

\memph \mephm (\marginpar and \emph)
1669 \newcommand\memph{1}{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1670 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1671 \newcommand\liPseudoUeberschrift{1}{
1672   \bigskip
1673   \noindent
1674   \textsf{\textbf{#1}}
1675   \noindent
1676 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1677 \newcommand\liBeschriftung{1}{
1678   \par
1679   \noindent
1680   \medskip
1681   \textbf{#1}:
1682   \medskip
1683   \noindent
1684 }

\hinweis
1685 \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.
1686 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

liEinbettung
1687 \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.
1688 \RequirePackage{xparse}
1689 \ExplSyntaxOn

```

```

1690 \NewDocumentEnvironment{ liAntwort } { 0{standard} }
1691 {
1692   \str_case:nn {#1} {
1693     {standard} {
1694       \def\beschriftung{}
1695       \mdfsetup{backgroundcolor=gray!1,linecolor=gray}
1696     }
1697     {richtig} {
1698       \def\beschriftung{richtig}
1699       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1700     }
1701     {falsch} {
1702       \def\beschriftung{falsch}
1703       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1704     }
1705     {muster} {
1706       \def\beschriftung{Musterlösung}
1707       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1708     }
1709   }
1710   \ifx\beschriftung\empty\else
1711     \noindent
1712     \textbf{\beschriftung{:}}
1713   \fi
1714   \begin{mdframed}
1715 }
1716 {\end{mdframed}}

```

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

```

1717 \NewDocumentEnvironment{ liAdditum } { o }
1718 {
1719   \begin{mdframed}[backgroundcolor=yellow!5]
1720     \IfNoValueTF {#1}
1721     { \liPseudoUeberschrift{Additum} }
1722     { \liPseudoUeberschrift{Additum:~#1} }
1723   }
1724 {\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.

```

1725 \NewDocumentEnvironment{ liExkurs }{o +b}{
1726   \vspace{0.2cm}%
1727   \begin{mdframed}[
1728     backgroundcolor=white,
1729     bottomline=false,
1730     innermargin=1cm,
1731     leftline=true,
1732     linecolor=black,
1733     linewidth=0.1cm,
1734     outermargin=1cm,
1735     rightline=false,
1736     topline=false,
1737   ]

```



```

1738 \footnotesize
1739 \noindent%
1740 \textbf{Exkurs:~#1}\par%
1741 \noindent%
1742 #2
1743 \end{mdframed}
1744 \vspace{0.2cm}
1745 }{}

```

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

```

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

```

Weiterführende Literatur:

- Quelle 1
- Quelle 2

```

1746 \cs_new:Npn \listen@punkt #1 {\item #1}
1747 \NewDocumentEnvironment { liQuellen }{ +b }
1748 {
1749   \seq_clear_new:N \l_quellen
1750   \seq_set_split:Nnn \l_quellen {\item} {#1}
1751   \seq_remove_all:Nn \l_quellen {}
1752   \begin{mdframed}[backgroundcolor=white!0]
1753     \footnotesize
1754     \noindent
1755     \textsf{\textbf{Weiterführende~Literatur:}}
1756     \medskip
1757     \begin{compactitem}
1758       \seq_map_function:NN \l_quellen {\listen@punkt}
1759     \end{compactitem}
1760   \end{mdframed}
1761   %
1762   \makeatletter
1763   \par\@afterindentfalse\@afterheading
1764   \makeatother
1765 } {}

```

liLernkartei

```

1766 \NewDocumentEnvironment { liLernkartei }{ m +b }
1767 {
1768   \begin{mdframed}
1769     \footnotesize
1770     \noindent%
1771     \textbf{Lernkarteikarte:~#1}\par%
1772     \noindent%
1773     #2
1774   \end{mdframed}
1775 } {}

```

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

```

1776 \NewDocumentEnvironment { liDiagramm }{ m +b }
1777 {
1778   \begin{mdframed}[backgroundcolor=white!0]
1779     \small
1780     \noindent%
1781     \textit{#1}:
1782     \begin{center}

```

```

1783 #2
1784 \medskip
1785 \end{center}
1786 \end{mdframed}
1787 } {}
1788 \ExplSyntaxOff

\liFussnoteUrl \liFussnoteUrl[\langle zusätzlicher-text \rangle]{\langle url \rangle} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
1789 \NewDocumentCommand{\liFussnoteUrl} { o m } {
1790 \footnote{\url{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1791 }
1792

\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.
1793 \NewDocumentCommand{\liFussnoteLink} { o m m } {
1794 \footnote{\href{#3}{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1795 }

\zB
1796 \def\zB{z.\,B. }

\ZB
1797 \def\ZB{Z.\,B. }

\dh
1798 \def\dh{d.\,h. }

1799

```

2.27 master-theorem.sty

1800 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1801 \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}

1802 \ExplSyntaxOn

1803 \RequirePackage{amsmath}

\liRundeKlammer

1804 \def\liRundeKlammer#1{

1805 \negthinspace \left(#1 \right)

1806 }

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

1807 \def\liThetaOhneMathe#1{

1808 \Theta \liRundeKlammer{#1}

1809 }

1810 \def\liTheta#1{

1811 \ifmmode

1812 \liThetaOhneMathe{#1}

1813 \else

1814 \$\liThetaOhneMathe{#1}\$

1815 \fi

1816 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
1817 \def\liOmegaOhneMathe#1{
1818 \Omega \liRundeKlammer{#1}
1819 }
1820 \def\liOmega#1{
1821 \ifmmode
1822 \liOmegaOhneMathe{#1}
1823 \else
1824 $\liOmegaOhneMathe{#1}$
1825 \fi
1826 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
1827 \def\liOOhneMathe#1{
1828 \mathcal{O} \liRundeKlammer{#1}
1829 }
1830 \def\liO#1{
1831 \ifmmode
1832 \liOOhneMathe{#1}
1833 \else
1834 $\liOOhneMathe{#1}$
1835 \fi
1836 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
1837 \def\liTOhneMathe#1#2{
1838 \tl_if_blank:nTF {#1}
1839 {}
1840 {#1 \cdot }
1841 T
1842 \liRundeKlammer{\frac{n}{#2}}
1843 }
1844 \def\liT#1#2{
1845 \ifmmode
1846 \liTOhneMathe{#1}{#2}
1847 \else
1848 $\liTOhneMathe{#1}{#2}$
1849 \fi
1850 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
1851 \def\liRekursionsGleichung{
1852 $T(n) = \liT{a}{b} + f(n)$
1853 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
1854 \def\liBedingungEins{
1855 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
1856 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
1857 \def\liBedingungZwei{
1858 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
1859 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
1860 \def\liBedingungDrei{
1861 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
1862 }

1863 \ExplSyntaxOff

```

\liMasterVariablen

```

1864 \def\liMasterVariablen{
1865   \begin{displaymath}
1866     T(n) = \liT{a}{b} + f(n)
1867   \end{displaymath}
1868
1869   \begin{itemize}
1870     \item[$a = $]
1871       Anzahl der rekursiven Aufrufe, Anzahl der Unterprobleme in der
1872       Rekursion
1873       ($a \geq 1$).
1874
1875     \item[$\textstyle{\frac{1}{b}} = $]
1876       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
1877       repräsentiert wird, Anteil an der Verkleinerung des Problems ($b > 1$).
1878
1879     \item[$f(n) = $]
1880       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
1881       die Kombination der Teillösungen entstehen. Eine von $T(n)$
1882       unabhängige und nicht negative Funktion.
1883   \end{itemize}
1884   \footcite{wiki:master-theorem}
1885   \footcite[Seite 19-35]{aud:fs:2}
1886 }

```

\liMasterFaelle

```

1887 \def\liMasterFaelle{
1888   \begin{description}
1889     \item[1. Fall:]
1890       $T(n) \in \liTheta{n^{\log\sb{b}a}}$
1891
1892       \hfill falls \liBedingungEins
1893       für $\varepsilon > 0$
1894
1895     \item[2. Fall:]
1896       $T(n) \in \liTheta{n^{\log\sb{b}a} \cdot \log n}$
1897
1898       \hfill falls \liBedingungZwei
1899
1900     \item[3. Fall:]
1901       $T(n) \in \liTheta{f(n)}$
1902
1903       \hfill falls \liBedingungDrei
1904       für $\varepsilon > 0$
1905       und ebenfalls für ein $c$ mit $0 < c < 1$ und alle hinreichend großen $n$
1906       gilt:
1907       $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$
1908   \end{description}
1909 }

```

\liMasterVariablenDeklaration

```

1910 \def\liMasterVariablenDeklaration#1#2#3{
1911   \begin{description}
1912     \item[Allgemeine Rekursionsgleichung:] \strut
1913
1914     \liRekursionsGleichung
1915
1916     \item[Anzahl der rekursiven Aufrufe ($a$):] \strut
1917
1918     #1
1919
1920     \item[Anteil Verkleinerung des Problems ($b$):] \strut
1921

```

```

1922     um  $\frac{1}{\#2}$  also  $b = \#2$ 
1923
1924     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut
1925
1926      $\#3$ 
1927
1928     \item[Ergibt folgende Rekursionsgleichung:] \strut
1929
1930      $T(n) = \text{\liT{\#1}{\#2}} + \#3$ 
1931 \end{description}
1932 }

\liMasterFallRechnung
1933 \def\liMasterFallRechnung#1#2#3{
1934     \begin{description}
1935     \item[1. Fall:] \liBedingungEins:
1936
1937     \#1
1938
1939     \item[2. Fall:] \liBedingungZwei:
1940
1941     \#2
1942
1943     \item[3. Fall:] \liBedingungDrei:
1944
1945     \#3
1946     \end{description}
1947 }

\liMasterExkurs
1948 \def\liMasterExkurs{
1949     \begin{liExkurs}[Master-Theorem]
1950     \liMasterVariablen
1951
1952     \noindent
1953     Dann gilt:
1954
1955     \liMasterFaelle
1956     \end{liExkurs}
1957 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
1958 \def\liMasterWolframLink#1{
1959     Berechne die Rekursionsgleichung auf WolframAlpha:
1960     \href{https://www.wolframalpha.com/input/?i=\#1}{WolframAlpha}
1961 }

1962

```

2.28 mathe.sty

```
1963 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1964 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
1965
1966 % for example \ltimes \rtimes
1967 %\RequirePackage{amssymb}
1968 \RequirePackage{amsmath}
1969
1970 %%
1971 % \mlq \mrq
1972 %%
1973 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
1974 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
1975
```

2.29 minimierung.sty

```

1976 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1977 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
1978 Minimierungsalgorithmus von einem NEA zu einem DEA]

1979 \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

1980 \def\liFussnote#1{\$x_{#1}\$}

1981 \def\li@fussnote@text#1#2{
1982 \liFussnote{#1}
1983 \quad
1984 {\footnotesize #2}
1985 }

\liFussnoteEinsText

1986 \def\liFussnoteEinsText{
1987 \li@fussnote@text{1}
1988 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
1989 }

\liFussnoteZweiText

1990 \def\liFussnoteZweiText{
1991 \li@fussnote@text{2}
1992 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
1993 }

\liFussnoteDreiText

1994 \def\liFussnoteDreiText{
1995 \li@fussnote@text{3}

```



```

1996 {In weiteren Iterationen markierte Zustände.}
1997 }

\liFussnoteVierText
1998 \def\liFussnoteVierText{
1999 \li@fussnote@text{4}
2000 {...}
2001 }

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



2002 \def\liFussnoten{
2003 \bigskip
2004
2005 \noindent
2006 \liFussnoteEinsText
2007
2008 \noindent
2009 \liFussnoteZweiText
2010
2011 \noindent
2012 \liFussnoteDreiText
2013
2014 \noindent
2015 \liFussnoteVierText
2016 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
2017 \def\liLeereZelle{\emptyset}

\liZustandsPaarVariablenName
2018 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
2019 \def\liZustandsPaar#1#2{
2020 $(
2021 \liZustandsPaarVariablenName_#1,
2022 \liZustandsPaarVariablenName_#2
2023 )$
2024 }

liUebergangsTabelle
2025 \renewcommand{\arraystretch}{1.4}
2026 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
2027 \liPseudoUeberschrift{Übergangstabelle}
2028 \begin{center}
2029 \begin{tabular}{r|l|l}
2030 \textbf{Zustandspaar} & \textbf{#1} & \textbf{#2} \\ \hline
2031 } {
2032 \end{tabular}
2033 \end{center}
2034 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle: Minimierungstabelle (Table filling)

2035 \ExplSyntaxOn
2036 \def\liUeberschriftDreiecksTabelle{
2037 \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
2038 }

```

\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. —

```

2039 \def\liMinimierungErklaerung{
2040   %\footcite[Seite~19]{koenig}
2041   \liParagraphMitLinien{
2042     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
2043     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
2044     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
2045     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
2046      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
2047     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
2048     somit~zueinander~( $k - 1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
2049     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
2050     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
2051   }
2052 }
2053 \ExplSyntaxOff
2054

```

2.30 normalformen.sty

```
2055 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2056 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10
2057 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
2058 Attributhülle]
```

Mathe für textit tex etc, typographie für geschweifte Klammer

```
2059 \liLadePakete{mathe,typographie}
2060 \directlua{
2061   helper = require('lehramt-informatik-helfer')
2062   normalformen = require('lehramt-informatik-normalformen')
2063 }
```

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
```

```
2064 \def\liTeilen#1{
2065   \directlua{tex.print(normalformen.teilen('#1'))}
2066 }
```

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

```
2067 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(\#1)}
2068 \def\liAttributHuelle#1{
2069   \ifmmode
2070     \liAttributHuelleOhneMathe{\#1}
2071   \else
2072     $\liAttributHuelleOhneMathe{\#1}$
2073   \fi
2074 }
```

\liAttributMenge Let-Abkürzung: $\text{let}\ \text{m}=\text{liAttributMenge}$
2075 $\text{def}\ \text{liAttributMenge}\#1\{\backslash\{\ \text{textit}\{\#1\}\ \backslash\}$

liAHuelle

```
2076 \NewDocumentEnvironment{ liAHuelle } { +b } {
2077   \begingroup
2078   \footnotesize
2079   \begin{multline*}
2080     \#1
2081   \end{multline*}
2082   \endgroup
2083 } { }
```

\liLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multline

Let-Abkürzung: $\text{let}\ \text{ahL}=\text{liLinksReduktion}$
 $\text{ahL}\{\text{ursprüngliche linke Attributmenge}\{\text{ohne dieses Attribut}\}\{\text{Ergebnis}\}$
2084 $\text{def}\ \text{liLinksReduktion}\#1\#2\#3\{$
2085 $\text{shoveleft}\{$
2086 $\text{liAttributHuelleOhneMathe}\{\text{FA},$
2087 $\text{liAttributMenge}\{\#1\ \text{string}\ \#2\}\} =$
2088 $\}\ \backslash\backslash$

```

2089 \shoveright{
2090 \liAttributMenge{#3}
2091 } \\\
2092 }

```

```

\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( $F, \{A, E \setminus E\}$ ) =  $\{A, E, F, B, D\}$ 
2093 \def\liLinksReduktionInline#1#2#3{%
2094 {%
2095 \footnotesize%
2096 $\liAttributHuelleOhneMathe{F,
2097 \liAttributMenge{#1 \string\ #2}} =
2098 \liAttributMenge{#3}$
2099 }
2100 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahr=\liLinksReduktionInline
\ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
2101 \def\liRechtsReduktionInline#1#2#3#4{%
2102 {%
2103 \footnotesize%
2104 $\liAttributHuelleOhneMathe{
2105 F \setminus
2106 \liFunktionaleAbhaengigkeit{#1}
2107 \def\tmp{#2}\ifx\tmp\empty
2108 \else
2109 \cup \liFunktionaleAbhaengigkeit{#2}
2110 \fi
2111 ,
2112 \liAttributMenge{#3}
2113 } =
2114 \liAttributMenge{#4}$
2115 }
2116 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
\fa{A, B -> C, D}:  $\{A, B\} \rightarrow \{C, D\}$  Regulärer Ausdruck zum Konvertieren:
\$(.*?) \\\rightarrow (.*)\$ \fa{$1 -> $2}
2117 \def\liFunktionaleAbhaengigkeit#1{%
2118 \directlua{normalformen.drucke_funk_abhaengigkeit('#1', false)}%
2119 }

```

```

\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;

```

```

2120 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
2121   \liGeschweifteKlammern
2122   {#1}
2123   {
2124     \begin{align*}
2125       \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
2126     \end{align*}
2127   }
2128   {-0.5cm}
2129   {-1.7cm}
2130 }

\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}

2131 \NewDocumentCommand {\liRelation} { O{R} m } {
2132   $\directlua{
2133     local name = helper.konvertiere_tiefgestellt('#1')
2134     tex.print(name)
2135   }$(\textit{\, #2\,})
2136 }

2137

```

2.31 petri.sty

2138 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2139 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]

2.31.1 Makro-Kürzel

\let\t=\liPetriTransitionsName

\let\tp=\liPetriTransPfeile

\let\k=\liPetriErreichKnotenDrei

2140 \RequirePackage{tikz}

2141 \usetikzlibrary{petri,arrows.meta}

Für die Darstellungsmatrix

2142 \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

```
2143 \def\liPetriSetzeSchluessel{%
2144   \def\TmpTransitionOne{}%
2145   \def\TmpTransitionTwo{}%
2146   \def\TmpTransitionThree{}%
2147   \def\TmpTransitionFour{}%
2148   \def\TmpTransitionFive{}%
2149   \def\TmpTransitionSix{}%
2150   \def\TmpTransitionSeven{}%
2151   \def\TmpTransitionEight{}%
2152   \def\TmpTransitionNine{}%
2153   \def\TmpTransitionTen{}%
2154   \pgfkeys{/petri/.cd,
2155     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2156     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2157     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2158     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
```

```

2159 p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
2160 p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2161 p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2162 p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2163 p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2164 p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2165 t1/.store in=\TmpTransitionOne,t1/.default=activated,
2166 t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2167 t3/.store in=\TmpTransitionThree,t3/.default=activated,
2168 t4/.store in=\TmpTransitionFour,t4/.default=activated,
2169 t5/.store in=\TmpTransitionFive,t5/.default=activated,
2170 t6/.store in=\TmpTransitionSix,t6/.default=activated,
2171 t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2172 t8/.store in=\TmpTransitionEight,t8/.default=activated,
2173 t9/.store in=\TmpTransitionNine,t9/.default=activated,
2174 t10/.store in=\TmpTransitionTen,t10/.default=activated,
2175 scale/.store in=\TmpScale,scale/.default=0.5,
2176 x/.store in=\TmpX,x/.default=5,
2177 y/.store in=\TmpY,y/.default=5,
2178 }%
2179 }

```

```

2180 \tikzset{
2181   li petri/.style={
2182     activated/.style={
2183       very thick
2184     },
2185     inhibitor/.style={
2186       {Circle[open,length=2mm,fill=white]}-
2187     }
2188   }
2189 }

```

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

```

2190 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2191 \def\liPetriTransitionsName#1{
2192   \ifmmode
2193     \liPetriTransitionsNameOhneMathe{#1}
2194   \else
2195     $\liPetriTransitionsNameOhneMathe{#1}$
2196   \fi
2197 }

```

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

```

2198 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2199   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2200 }

```

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

```

2201 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2202 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2203 }

```

2.32 potenzmengen-konstruktion.sty

```
2204 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2205 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2206 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2207 \liLadePakete{formale-sprachen}
2208 \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}
    }
  }
}
```

```
2209 \def\liZustandsMengenSammlung#1#2{
2210   \liZustandsnameGross{#1}
2211   {
2212     \footnotesize
2213     \liPotenzmenge{
2214       \str_case:nn {#1} {#2
2215       }
2216     }
2217 }
```

```
\liZustandsMengenSammlungNr
```

```
2218 \def\liZustandsMengenSammlungNr#1#2{
2219   \liZustandsnameGross{#1}
2220   {
```



```
2221     \footnotesize
2222     \liZustandsmengeNr{
2223         \str_case:nn {#1} #2
2224     }
2225 }
2226 }

2227 \ExplSyntaxOff
2228
```

2.33 pseudo.sty

```

2229 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2230 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen
2231 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>

```

2232 \RequirePackage[german,boxruled]{algorithm2e}

```

```

2233

```

2.34 pumping-lemma.sty

2234 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2235 \ProvidesPackage{lehramt-informatik-pumping-lemma}[2021/08/11 Enthält die
 2236 Definitionen für das Pumping-Lemma in der Regulären Sprache und
 2237 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2238 \def\liPumpingRegulaer{%
2239   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2240   alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2241    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2242   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2243
2244   \begin{enumerate}
2245     \item  $|v| \geq 1$ 
2246     (Das Wort  $v$  ist nicht leer.)
2247
2248     \item  $|uv| \leq j$ 
2249     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2250
2251     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w$  in  $L$ 
2252     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2253     Sprache  $L$ )
2254   \end{enumerate}
2255
2256   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2257   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2258 }
```

\liPumpingKontextfrei

```
2259 \def\liPumpingKontextfrei{%
2260   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2261   sich alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  zerlegen lassen in
2262    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2263
2264   \begin{enumerate}
2265     \item  $|vx| \geq 1$ 
2266     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2267
2268     \item  $|vwx| \leq j$ 
2269     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2270
2271     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy$  in  $L$  (Für jede
2272     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2273     Sprache  $L$ )
2274   \end{enumerate}
2275 }
2276
```

2.35 quicksort.sty

```

2277 % https://tex.stackexchange.com/a/142634
2278 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2279 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2280
2281 %-----
2282 % USAGE:
2283 % \QSinitialize{comma, separated, numerical, values}
2284 % \loop
2285 % \QSpivotStep
2286 % \ifnum\value{pivotcount}>0
2287 %   \QSSortStep
2288 % \repeat
2289 %-----
2290
2291 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2292 \RequirePackage{xintfrac, xinttools}
2293
2294 \RequirePackage{tikz}
2295
2296 %-----
2297 % FIRST PART: TikZ styles and macros for the actual drawing
2298 \newcounter{cellcount}% used for coordinates of the node
2299 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2300
2301 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2302 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2303 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2304 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2305 % by police of LaTeX good conduct ? )
2306 \tikzset{l/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2307          o/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2308          r/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, t
2309 % this is the "b" style as used in the image below
2310 %   b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black,
2311 % nicer:
2312 %   b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta
2313 %   g/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white,
2314
2315 % NOTE the b style was originally the same as the r(aised) style apart from
2316 % not being raised, but I find it nicer with a somewhat different
2317 % specification. I have not updated the images though.
2318
2319 % How the nodes are drawn depending on whether on the left of the pivot value
2320 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2321
2322 \def\DecoLEFT #1{%
2323   \xintFor* ##1 in {#1} \do
2324     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2325 }
2326
2327 \def\DecoINERT #1{%
2328   \xintFor* ##1 in {#1} \do
2329     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2330 }
2331
2332 \def\DecoRIGHT #1{%
2333   \xintFor* ##1 in {#1} \do
2334     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2335 }
2336
2337 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2338   \xintFor* ##1 in {#1} \do

```

```

2339     {\stepcounter{cellcount}}%
2340     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2341 }
2342
2343 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2344     \xintFor* ##1 in {#1} \do
2345     {\stepcounter{cellcount}}%
2346     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2347 }
2348
2349 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2350     \xintFor* ##1 in {#1} \do
2351     {\stepcounter{cellcount}}%
2352     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2353 }
2354
2355 %-----
2356 % SECOND PART: the actual sorting routines.
2357
2358 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2359 \def\QS@sort@b #1{\ifcase #1
2360     \expandafter\QS@sort@empty
2361     \or\expandafter\QS@sort@single
2362     \else\expandafter\QS@sort@c
2363     \fi
2364 }%
2365 \def\QS@sort@empty #1{}
2366 \def\QS@sort@single #1{\QSIr {#1}}
2367
2368 % This step is to pick the last as pivot.
2369 \def\QS@sort@c #1%
2370     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2371
2372 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2373 % The trick with \xintApplyUnbraced is that for example when selecting
2374 % the elements smaller than pivot, if we had been using \xintApply we
2375 % would have had at the minimum an empty brace pair. Thus we use the
2376 % "unbraced" variant, but then the \QS@select@smaller has added in
2377 % anticipation a level of braces.
2378 \def\QS@sort@d #1#2{%
2379     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2380     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2381     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2382 }%
2383 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2384 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2385 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2386
2387 %
2388 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2389 %
2390 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2391 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2392 % latter must handle correctly an empty argument.
2393
2394 %-----
2395 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QSinitialize.
2396
2397 % This draws all with suitable highlighting for the newly chosen pivots
2398 % (which will be shown raised)

```

```

2399 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2400             \let\QSIr\DecoINERT
2401             \let\QSIrr\DecoINERT
2402             \let\QSRr\DecoRIGHTwithPivot
2403 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2404             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2405             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2406 }
2407
2408 % This sorts and then draws, showing where the pivot chosen in the previous
2409 % step go. Next time they will have become "inert". If pivotcount is still at
2410 % zero on exit from \QSpivotStep, then this is the signal to stop before
2411 % executing \QSSortStep.
2412 \def\QSSortStep {\def\QSLr {\noexpand\QS@sort@a}%
2413             \def\QSRr {\noexpand\QS@sort@a}%
2414             \def\QSIr {\noexpand\QSIrr}%
2415             \let\QSIrr\relax
2416             \edef\QS@list{\QS@list}%
2417             \let\QSLr\relax
2418             \let\QSRr\relax
2419             \let\QSIr\relax
2420             \edef\QS@list{\QS@list}%
2421             \let\QSLr\DecoLEFT
2422             \let\QSIr\DecoINERTwithPivot
2423             \let\QSIrr\DecoINERT
2424             \let\QSRr\DecoRIGHT
2425 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2426             \setcounter{cellcount}{0}%
2427             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2428 }
2429
2430 \def\QSinitialize #1{%
2431     % first, we convert the comma separated values into a list of braced items
2432     % we use an \edef, and anyhow many \edef's will be used later
2433     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2434     \let\QSRr\DecoRIGHT
2435     % The \QSRr marker mutated to draw the last element as
2436     % pivot and the earlier ones with the suitable style.
2437     %
2438     % The list of marked braced items \QS@list is used both for drawing
2439     % (as here) and for doing the exchange of elements during sort.
2440     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2441             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2442 }
2443

```

2.36 relationale-algebra.sty

```

2444 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2445 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]
2446 \RequirePackage{amsmath}
2447 \RequirePackage{amssymb}

```

```

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

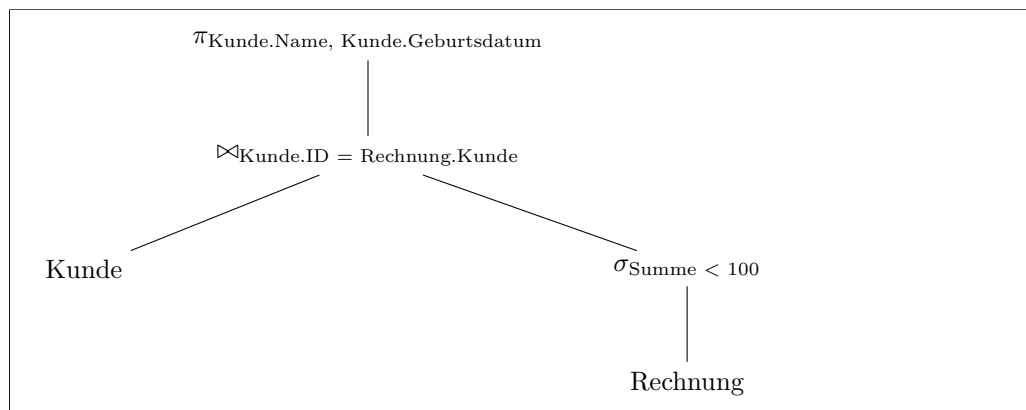
  \node[below=of pi]
    (theta join) {\bowtie_{\text{Kunde.ID = Rechnung.Kunde}}};
    edge (pi);

  \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}

```



```

2448 \RequirePackage{tikz}
2449 \usetikzlibrary{positioning}

    Privates Makros, das zwei Querstriche erzeugt.
2450 \def\o@join{\setbox0=\hbox{\bowtie}}%
2451   \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2452 }

\leftouterjoin A \leftouterjoin B: A ⋈ B
2453 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}

\rightouterjoin A \rightouterjoin B: A ⋉ B
2454 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}

\fullouterjoin A \fullouterjoin B: A ⋈⋉ B
2455 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
2456

```

2.37 rmodell.sty

```
2457 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2458 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2459 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2460 Datenbanken.]
2461 \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

```
2462 \def\liPrimaer#1{\ul{#1}}
```

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

```
2463 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

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

```
2464 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2465 \ExplSyntaxOn
2466 \NewDocumentEnvironment { liRmodell }
2467 { +b }
2468 {
2469   \medskip
2470   {
2471     \linespread{2}
2472     \setlength{\parindent}{0pt}
2473     \li@Rmodell@Schrift#1
2474   }
2475   \medskip
2476 } {}
2477 \ExplSyntaxOff
```

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

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

```
2478 \def\liRelationMenge#1#2{
2479 \noindent
2480 #1 : \[ #2 ]\}
2481 \par
2482 }
```

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

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

```
2483 \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}
```

```
2484 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2485
```


2.38 sortieren.sty

```
2486 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2487 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
2488 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}
}
```



```
2489 \RequirePackage{tikz}
2490 \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.

```
2491 \def\liVertauschen#1{
2492   \directlua{
2493     local sortieren = require('lehramt-informatik-sortieren')
2494     sortieren('#1')
2495   }
2496 }
```

`\liSortierPfeil`

```
2497 \def\liSortierPfeil#1#2{
2498   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2499 }
```

`\liSortierPfeilUnten`

```
2500 \def\liSortierPfeilUnten#1#2{
2501   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2502 }
```

`\liSortierMarkierung`

```
2503 \def\liSortierMarkierung#1#2{\node[
2504   draw,
2505   very thick,
2506   fit=(reihe.#1) (reihe.#2),
2507   inner sep=0pt
2508 ] {}
2509 }

2510 \tikzset{
2511   li sortierung zahlenreihe/.style={
2512     draw,
2513     thin,
2514     font=\large,
2515     rectangle split horizontal,
2516     rectangle split,
2517   }
2518 }
```

```

2519 % https://tex.stackexchange.com/a/140895
2520 \RequirePackage{forest,xstring}
2521 \usetikzlibrary{calc}
2522
2523 \makeatletter
2524 \pgfmathdeclarefunction{strrepeat}{2}{%
2525   \begingroup\pgfmathint{#2}\pgfmathcount\pgfmathresult
2526   \let\pgfmathresult\pgfutil@empty
2527   \pgfutil@loop\ifnum\pgfmathcount>0\relax
2528     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2529     \advance\pgfmathcount-1\relax
2530   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2531 \makeatother
2532
2533 \def\myNodes{}
2534
2535 \ExplSyntaxOn
2536 \newcommand*\sortList[1]{%
2537   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2538 \ExplSyntaxOff
2539
2540 \forestset{
2541   sort/.code={%
2542     \pgfmathparse{level()>\forestSortLevel}%
2543     \ifnum\pgfmathresult=0
2544       \StrSubstitute{\forestov{content}}{ }{,}[\myList]%
2545       \sortList\myList
2546       \StrSubstitute{\myList}{,}{ }[\myList]%
2547       \pgfmathparse{strrepeat("1",level())}%
2548       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-1!({\forestov{name}}$)
2549         (m\forestov{name}) {\myList}}%
2550       \pgfmathparse{level()=\forestSortLevel}%
2551       \ifnum\pgfmathresult=1
2552         \forestOget{\forestov{@first}}{name}\forestFirst
2553         \forestOget{\forestov{@last}}{name}\forestLast
2554         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2555           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2556       \fi
2557       \ifnum\forestov{@parent}=0\else
2558         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2559       \fi
2560       \gappto\myNodes{;}%
2561     \fi}}
2562
2563 \forestset{sort level/.code=%
2564   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2565   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2566

```

2.39 spalten.sty

```
2567 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2568 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2569 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2570 realisiert werden kann.]
2571 \RequirePackage{multicol}
```

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

```
2572 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2573
```

2.40 sql.sty

```
2574 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2575 \ProvidesPackage{lehramt-informatik-tabelle}[2021/09/02 Zu Setzen von SQL]
2576 \liLadePakete{syntax}
2577 \RequirePackage{fancyvrb}
2578 \DefineVerbatimEnvironment{liSqlErgebnis}{Verbatim}
2579 {fontsize=\footnotesize}
2580
```

2.41 struktogramm.sty

```
2581 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2582 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2583 Paket struktex zum Setzen von Struktogrammen]
2584 \RequirePackage{struktex}
2585
```

2.42 syntax.sty

```
2586 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2587 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2588 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2589 \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
```

```
2590 \ExplSyntaxOn
2591 \directlua{
2592   syntax = require('lehramt-informatik-syntax')
2593   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2594   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2595   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2596   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2597   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2598   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2599 }
2600 \RequirePackage{hyperref}
2601 \RequirePackage{minted}
2602 % pygmentize -L styles
2603 \usemintedstyle{colorful}
2604 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2605 %\AfterEndEnvironment{minted}{\end{mdframed}}
2606 %\setminted{breaklines=true,linenos}
2607 \setminted{
2608   breaklines=true,
2609   linenos,
2610   fontsize=\footnotesize,
2611 }
```

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

Let-Abkürzung: `\let\j=\liJavaCode`

```
2612 \def\liJavaCode#1{\,\mintinline{java}|#1|\,}
```

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

```
2613 \def\liLatexCode#1{\mintinline{latex}|#1|}
```

```
2614 \def\li@GithubLink#1#2{
2615   \begin{flushright}
2616     \tiny
2617     Code-Beispiel~auf~Github~ansehen:~
2618     \href{#1}{\nolinkurl{#2}}
2619   \end{flushright}
2620 }
```

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

```
2621 \NewDocumentCommand{\liJavaDatei}{0{firstline=3} m }{
2622   \inputminted[#1]{java}{
2623     \directlua{
2624       syntax.drucke_absoluten_pfad('#2', false)
2625     }
2626   }
2627   \li@GithubLink
```

```

2628     {\directlua{syntax.drucke_github_url('#2', false)}}
2629     {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2630 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2631 \NewDocumentCommand{\liJavaTestDatei}{ 0{firstline=3} m }{
2632   \inputminted[#1]{java}{
2633     \directlua{
2634       syntax.drucke_absoluten_pfad('#2', true)
2635     }
2636   }
2637   \li@GithubLink
2638   {\directlua{syntax.drucke_github_url('#2', true)}}
2639   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2640 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([.
\liJavaExamen$1{$2}{$3}{$4}{$5}
2641 \NewDocumentCommand{\liJavaExamen}{ 0{firstline=3} m m m m }{
2642   \inputminted[#1]{java}{
2643     \directlua{
2644       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2645     }
2646   }
2647
2648   \li@GithubLink
2649   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2650   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2651 }

\liAssemblerCode
2652 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2653 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2654   \inputminted{asm}{#1}
2655 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2656 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2657   \inputminted{componentpascal}{#1}
2658 }

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2659 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2660 \NewDocumentCommand{\liHaskellDatei}{ m }{
2661   \inputminted{haskell}{#1}
2662 }

2663 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2664 \def\liSqlCode#1{\mintinline{sql}|#1|}

2665

```

2.43 syntaxbaum.sty

```
2666 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2667 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2668 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
2669 \RequirePackage{tikz-qtrees}
2670
2671 \tikzset{li parsetree/.style={
2672     every internal node/.style={
2673         draw,circle
2674     },
2675     every leaf node/.style={
2676         draw,rectangle
2677     },
2678 }
2679 }
2680
```


2.44 synthese-algorithmus.sty

```
2681 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2682 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2683 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2684 Relation in die 3. Normalform]

2685 \liLadePakete{normalformen,mathe,typographie}
2686 \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}}

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

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

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

2.44.4 TeX-Markup Rechtsreduktion

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

\liPseudoUeberschrift{F}

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

\liPseudoUeberschrift{A}

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

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

```
2687 \def\liSyntheseUeberschrift#1{
2688   {
2689     \bfseries
2690     \sffamily
2691     \str_case:nn {#1} {
2692       {1} {Kanonische-Überdeckung}
2693       {1-1} {Linksreduktion}
2694       {1-2} {Rechtsreduktion}
2695       {1-3} {Löschen-leerer-Klauseln}
2696       {1-4} {Vereinigung}
2697       {2} {Relationsschemata-formen}
2698       {3} {Schlüssel-hinzufügen}
2699       {4} {Entfernung-überflüssiger-Teilschemata}
2700     }
```

```

2701 }
2702 }

\liSyntheseErklaerung Let-Abkürzung: \let\erklaerung=\liSyntheseErklaerung
2703 \def\li@synthese@erklaerung@texte#1{
2704   \str_case:nn {#1} {
2705     {1} {
2706       Die~kanonische~Überdeckung~---also~die~kleinst~mögliche~noch~
2707       äquivalente~Menge~von~funktionalen~Abhängigkeiten~kann~in~vier~
2708       Schritten~erreicht~werden.
2709     }
2710     {1-1} {
2711       Führe~für~jede~funktionale~Anhängigkeit~
2712        $\alpha \rightarrow \beta$ ~in~F~die~Linksreduktion~durch,~
2713       überprüfe~also~für~alle~
2714        $A \in \alpha$ ,~ob~ $A$ ~überflüssig~ist,~d.h.~ob~
2715        $\beta \subseteqq \text{\liAttributHuelle}\{F, \alpha \cup A\}$ .
2716     }
2717     {1-2} {
2718       Führe~für~jede~(verbliebene)~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~die~Rechtsreduktion~durch,~überprüfe~also~für~
2719       alle~ $B \in \beta$ ,~ob~ $B \in \text{\liAttributHuelle}\{F, (\alpha \rightarrow \beta) \cup (\alpha \rightarrow B)\}$ ,~
2720        $\alpha$ ~gilt.~In~diesem~Fall~ist~B~auf~der~rechten~Seite~
2721       überflüssig~und~kann~eliminiert~werden,~d.h.~ $\alpha \rightarrow \beta$ ~wird~durch~ $\alpha \rightarrow (\beta \cup B)$ ~
2722       ersetzt.
2723     }
2724     {1-3} {
2725       Entferne~die~funktionalen~Abhängigkeiten~der~Form~ $\alpha \rightarrow \emptyset$ ,~die~im~2.~Schritt~möglicherweise~
2726       entstanden~sind.
2727     }
2728     {1-4} {
2729       Fasse~mittels~der~Vereinigungsregel~funktionale~Abhängigkeiten~
2730       der~Form~ $\alpha \rightarrow \beta_{\{1\}}, \dots, \alpha \rightarrow \beta_{\{n\}}$ ,~so~dass~ $\alpha \rightarrow \beta_{\{1\}} \cup \dots \cup \beta_{\{n\}}$ ~
2731       verbleibt.
2732     }
2733     % Kemper Seite 197
2734     {2} {
2735       Erzeuge~für~jede~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~in~F~ein~Relationenschema~ $\mathcal{R}_{\alpha}$ ~
2736        $:= \alpha \cup \beta$ .
2737     }
2738     {3} {
2739       Falls~eines~der~in~Schritt~2.~erzeugten~Schemata~ $\mathcal{R}_{\alpha}$ ~einen~Schlüsselkandidaten~von~ $\mathcal{R}_{\alpha}$ ~bezüglich~ $F$ ~
2740       enthält,~sind~wir~fertig,~sonst~wähle~einen~Schlüsselkandidaten~
2741        $\mathcal{K} \subseteqq \mathcal{R}_{\alpha}$ ~aus~und~definiere~folgendes~
2742       zusätzliche~Schema:~ $\mathcal{R}_{\alpha} \setminus \mathcal{K} := \mathcal{K}$ ~
2743       und~ $\mathcal{F} \setminus \mathcal{K} := \emptyset$ 
2744     }
2745     {4} {
2746       Eliminiere~diejenigen~Schemata~ $\mathcal{R}_{\alpha}$ ,~die~in~einem~
2747       anderen~Relationenschema~ $\mathcal{R}_{\alpha'}$ ~enthalten~sind,~d.h.~
2748        $\mathcal{R}_{\alpha} \subseteqq \mathcal{R}_{\alpha'}$ .
2749     }
2750   }
2751 }
2752 \def\liSyntheseErklaerung#1{
2753 {
2754   \itshape
2755   \footnotesize

```

```

2763     \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
2764   }
2765 }

```

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

```

2766 \def\liSyntheseUeberErklaerung#1{
2767   \liSyntheseUeberschrift{#1}\par
2768   \liSyntheseErklaerung{#1}
2769 }

```

```

2770 \ExplSyntaxOff
2771

```

2.45 tabelle.sty

2772 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2773 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]

2774 \RequirePackage{tabularx}

2775

2.46 typographie.sty

```
2776 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2777 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
2778 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
2779 formatierung.sty definiert.]
```

```
2780 \ExplSyntaxOn
```

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

```
2781 \RequirePackage{fontawesome}
```

```
\liErledigt \liErledigt: ☑
```

```
2782 \let\liErledigt=\faCheckSquareO
```

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

```
2783 \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 —

```
2784 \def\liParagraphMitLinien#1{
```

```
2785 \noindent
```

```
2786 \vrule height 2pt depth -1.6pt width 0.4cm
```

```
2787 \enspace
```

```
2788 #1
```

```
2789 \enspace
```

```
2790 \leaders\vrule height 2pt depth -1.6pt \hfill \null
```

```
2791 \par
```

```
2792 \medskip
```

```
2793 }
```

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

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

```
2794 \def\liGeschweifteKlammern#1#2#3#4{
```

```
2795 \par
```

```
2796 \medskip
```

```
2797 \noindent
```

```
2798 #1 \, \$= \Bigl\{ \$
```

```
2799 \vspace{\#3}
```

```
2800 #2
```

```
2801 \vspace{\#4}
```

```
2802 \begin{flushright} \$\Bigr\} \$\end{flushright}
```

```
2803 \par
```

```
2804 }
```

```
\liTypoUeberschrift
```

```
2805 \def\liTypoUeberschrift#1{
```

```
2806 {
```

```
2807 \bfseries\sffamily
```

```
2808 #1
```

```
2809 }
```

```
2810 }
```

```

\liTypoUeberGross
2811 \def\liTypoUeberGross#1{
2812   {
2813     \huge
2814     \liTypoUeberschrift{#1}
2815   }
2816 }

```

```

\liTypoUeberGross
2817 \def\liTypoUeberGROSS#1{
2818   {
2819     \Huge
2820     \liTypoUeberschrift{#1}
2821   }
2822 }

2823 \ExplSyntaxOff
2824

```

2.47 uml.sty

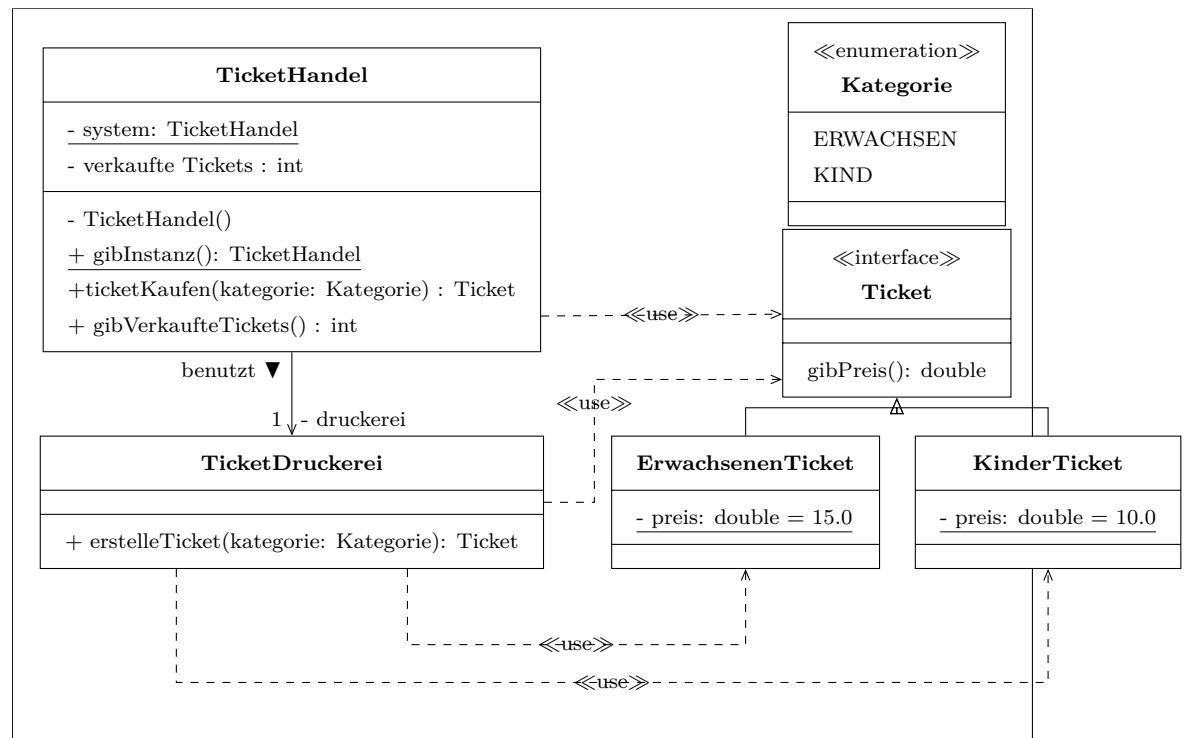
```

2825 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2826 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
2827 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
2828 Erweiterung bereitstellt]

2829 \RequirePackage{tikz-uml}
2830 \RequirePackage{tikz-uml-activity}
2831 % Not compatible with wasysym
2832 %\RequirePackage{mathabx}
2833 \RequirePackage{wasysym}
2834 \usetikzlibrary{positioning}

2835 \tikzumlset{
2836   fill class=white!0,
2837   font=\footnotesize,
2838   fill object=white!0,
2839   fill note=white!0,
2840   fill state=white!0,
2841   % Use case
2842   fill usecase=white!0,
2843   fill system=white!0,
2844 }

```



```

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

```

```

2845 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
2846   \def\@liDirLeft{}
2847   \def\@liDirRight{ \RIGHTarrow}
2848   \pgfkeys{/lese/dir/.is choice}
2849   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
2850   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
2851   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{\LEFTarrow }}}
2852   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
2853
2854   \def\@liPos{above}
2855   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
2856

```



```

2857 \def\@liDistance{0cm}
2858 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
2859
2860 \pgfkeys{/lese/.cd,#1}
2861
2862 \node[\@liPos = \@liDistance of #2-middle] {
2863   \@liDirLeft{\footnotesize#2}\@liDirRight
2864 };
2865 }
2866

```

2.48 vollstaendige-induktion.sty

2867 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2868 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01
 2869 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
 2870 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{\textcolor{violet}{cn}}(\{m\{n + 1\} - 1)\}
  \{m\{n + 1\} + 1\}
}{
  \{e\{Java nach Mathe}\} \\\
\%
&= \frac{
  \{(4\{m\{n\} + 2) \cdot \text{\textcolor{violet}{cn}}(\{m\{n\}\}
  \{m\{n + 2\}\}
}{
  \{e\{addiert, subtrahiert}\} \\\
\%
&= \frac{
  \{(4n + 2) \cdot \{m\{(2n)\}\}
  \{(n + 2) \cdot \{m\{(n + 1)\} \cdot n!\}
}{
  \{e\{für cn(n) Formel eingesetzt}\} \\\
\%
&= \frac{
  \{(4n + 2) \cdot (2n)! \cdot \{m\{\cdot (n + 1)\}\}
  \{(n + 2) \cdot (n + 1)! \cdot n! \cdot \{m\{\cdot (n + 1)\}\}
}{
  \{e\{(n + 1)\} multipliziert\} \\\
\%
&= \frac{
  \{(4n + 2) \cdot \{m\{(n + 1) \cdot (2n)\}\}
  \{(n + 2) \cdot (n + 1)! \cdot n! \cdot \{m\{(n + 1) \cdot n!\}\}
}{
  \{e\{umsortiert\} \\\
\%
&= \frac{
  \{\{m\{(2(n + 1))!\}
  \{\{m\{(n + 2)! \cdot (n + 1)!\}
}{
  \{e\{Hilfsgleichungen verwendet}\} \\\
\%
&= \frac{
  \{(2(\{m\{n + 1\}\}!\}
  \{((\{m\{n + 1\}\} + 1)! \cdot (\{m\{n + 1\}\}!\}
}{
  \{e\{(n + 1)\} verdeutlicht\} \\\
\end{align*}
```

Lade häufig benötigte Pakete

2871 \RequirePackage{lehramt-informatik-typographie}
 2872 \RequirePackage{lehramt-informatik-mathe}
 2873 \RequirePackage{lehramt-informatik-syntax}
 2874 \ExplSyntaxOn

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

Let-Abkürzung: \let\m=\liInduktionMarkierung

2875 \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

```
2876 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
2877 \def\liInduktionAnfang{
2878   \liPseudoUeberschrift{Induktionsanfang}
2879
2880   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2881   \liParagraphMitLinien{
2882     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
2883   }
2884 }
```

\liInduktionVoraussetzung

```
2885 \def\liInduktionVoraussetzung{
2886   \liPseudoUeberschrift{Induktionsvoraussetzung}
2887
2888   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2889   \liParagraphMitLinien{
2890     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ $.
2891   }
2892 }
```

\liInduktionSchritt

```
2893 \def\liInduktionSchritt{
2894   \liPseudoUeberschrift{Induktionsschritt}
2895
2896   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2897   \liParagraphMitLinien{
2898     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
2899     auch~ $A(n=k+1)$ ~wahr~sein~muss.
2900   }
2901 }

2902 \ExplSyntaxOff
2903
```

2.49 wasserfall.sty

```
2904 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2905 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
2906 \RequirePackage{tikz}
2907 \tikzset{wasserfall/.style={
2908   >=stealth,
2909   node distance = 2mm and -8mm,
2910   start chain = A going below right,
2911   every node/.style = {
2912     draw,
2913     text width=24mm,
2914     minimum height=12mm,
2915     align=center,
2916     inner sep=1mm,
2917     fill=white,
2918     drop shadow={fill=black},
2919     on chain=A
2920   },
2921 }}
2922 \usetikzlibrary{chains,positioning,shadows}
2923
```

2.50 wpkalkuel.sty

2924 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2925 \ProvidesPackage{lehrant-informatik-wpkalkuel}[2020/06/13]

2.50.1 Makro-Kürzel

\let\wp=\liWpKalkuel

\let\equivalent=\liWpEquivalent

\let\erklaerung=\liWpErklaerung

2926 \RequirePackage{amsmath}

2927 \ExplSyntaxOn

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

2928 \def\liWpKalkuelOhneMathe#1#2{

2929 \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)

2930 }

2931 \def\liWpKalkuel#1#2{

2932 \ifmmode

2933 \liWpKalkuelOhneMathe{#1}{#2}

2934 \else

2935 \$\liWpKalkuelOhneMathe{#1}{#2}\$

2936 \fi

2937 }

\MatheEnv

2938 \def\MatheEnv#1{

2939 \medskip

2940

2941 \hspace{1em}#1

2942

2943 \medskip

2944 }

\Mathe

2945 \def\Mathe#1{

2946 \MatheEnv{\${#1}\$}

2947 }

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

2948 \def\liWpEquivalent#1{

2949 \MatheEnv{\${\equiv}\hspace{1em}\${#1}\$}

2950 }

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

2951 \newlength{\@Skip@Erklaerung@Reset}

2952 \def\liWpErklaerung#1{

2953 \setlength{\@Skip@Erklaerung@Reset}{\leftskip}

2954 \setlength{\leftskip}{0.5cm}

2955

2956 \par

2957 \noindent

2958 {

2959 \scriptsize

2960 #1

2961 }

2962 \par

2963

2964 \setlength{\leftskip}{\@Skip@Erklaerung@Reset}

2965 }

\liWpErklaerungVerzweigung

```

2966 \def\liWpErklaerungVerzweigung{
2967   $\liWpKalkuelOhneMathe{if~\{-b~\}~then~\{-a1~\}~else~\{-a2~\}}{Q}
2968   \equiv
2969   (b \land \liWpKalkuelOhneMathe{a1}{Q})
2970   \lor
2971   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
2972 }

2973 \ExplSyntaxOff

2974

```

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			
\#	118	\AddToHook	1371, 1430, 2721, 2724, 2734, 2735, 2736, 2741, 2742
\,	356, 411, 1192, 1194, 1796, 1797, 1798, 2135, 2612, 2798	\advance	2529
\@Skip@Erklaerung@Reset	2951, 2953, 2964	\AfterEndEnvironment	2605
\@afterheading	1763	\allsectionsfont	1241
\@afterindentfalse	1763	\Alph	1255
\@liDirLeft	2846, 2851, 2863	\alph	1255, 1256
\@liDirRight	2847, 2849, 2850, 2851, 2852, 2863	\alpha	2712, 2714, 2715, 2718, 2720, 2721, 2722, 2723, 2724, 2728, 2734, 2735, 2740, 2741, 2742, 2745, 2753, 2754, 2755
\@liDistance	2857, 2858, 2862	\arabic	1255, 2324, 2329, 2334, 2340, 2346, 2352
\@liPos	2854, 2855, 2862	\arraystretch	2025
\\	618, 641, 642, 645, 646, 649, 650, 742, 743, 744, 851, 880, 882, 908, 917, 962, 1004, 1005, 1006, 1011, 1012, 1013, 1033, 1670, 2030, 2088, 2091	B	
\{	216, 1129, 1139, 1151, 1152, 1157, 1191, 1510, 2075, 2480, 2798, 2967	\BeforeBeginEnvironment	2604
\}	216, 1129, 1139, 1151, 1152, 1159, 1195, 1511, 2075, 2480, 2802, 2967	\begin	639, 690, 705, 740, 764, 811, 843, 858, 878, 888, 904, 924, 956, 973, 1002, 1027, 1048, 1063, 1172, 1261, 1336, 1431, 1437, 1441, 1445, 1464, 1473, 1480, 1588, 1662, 1714, 1719, 1727, 1752, 1757, 1768, 1778, 1782, 1865, 1869, 1888, 1911, 1934, 1949, 2028, 2029, 2079, 2124, 2244, 2264, 2405, 2427, 2441, 2604, 2615, 2802
_	37, 45, 54, 56, 305, 331, 355, 356, 370, 371, 377, 380, 383, 393, 405, 1372, 1427	\begingroup	1663, 2077, 2525
_	2087, 2097	\beschriftung	1694, 1698, 1702, 1706, 1710, 1712
A		\beta	2712, 2715, 2719, 2720,
\addbibresource	1640, 1641, 1642, 1643, 1644, 1645, 1646, 1647, 1648, 1649	C	
		\c	1338, 1339
		\cdot	1840, 1896, 1907
		\centerline	1469, 2403, 2425, 2440
		\chapter	1245, 1246
		\char	1670
		\clearpage	1389, 1451, 1664
		\cline	618
		\clist	235, 281, 282, 302, 306, 2537
		\columnbreak	2572
		\cs	305, 331, 355, 356, 393, 405, 1746
		\csname	1323, 1326
		\cup	1152, 2109, 2721, 2735, 2742
		D	
		\DeclareMathSymbol	1973, 1974
		\DecoINERT	2327, 2400, 2401, 2423
		\DecoINERTwithPivot	2343, 2422
		\DecoLEFT	2322, 2421

\DecoLEFTwithPivot ..		environments:		\footcite	
..... 2337, 2399		liAdditum	1717	702, 724, 771, 794,	
\DecoRIGHT 2332, 2424, 2434		liAHuelle	2076	826, 921, 948, 994,	
\DecoRIGHTwithPivot .		liAntwort	1688	1489, 1492, 1499,	
..... 2349, 2402		liDiagramm	1776	1504, 1509, 1513,	
\definecolor	1243	liEinbettung	1687	1519, 1524, 1629,	
\DefineVerbatimEnvironment		liExkurs	1725	1884, 1885, 2040, 2257	
..... 2578		liGraphenFormat .	1315	\footnote	1790, 1794
\delta 74, 116, 174, 216, 1143		liKasten	1260	\footnotesize	156, 366,
\dh	1798, 2723	liKontrollflussgraph		448, 542, 1116,	
\directlua 1587		1685, 1738, 1753,	
67, 146, 204, 209,		liLernkartei	1766	1769, 1984, 2078,	
1124, 1138, 1158,		liProduktionsRegeln		2095, 2103, 2212,	
1166, 1173, 1178,	 1167		2221, 2464, 2579,	
2060, 2065, 2118,		liProjektSprache	1686	2610, 2762, 2837, 2863	
2125, 2132, 2492,		liQuellen	1746	\footrulewidth .	1272, 1622
2591, 2623, 2628,		liRelationenSchemaFormat		\foreach .	1338, 1341, 1348
2629, 2633, 2638,	 2484		\forestFirst ..	2552, 2555
2639, 2643, 2649, 2650		liRmodell	2464	\forestLast ...	2553, 2555
\do	2323, 2328,	liUebergangsTabelle		\forestOget ...	2552, 2553
2333, 2338, 2344, 2350	 2025		\forestOnes	2565
\dots	528, 532,	\equiv	2949, 2968	\forestOv	2554, 2555, 2558
1510, 2251, 2734, 2735		\erzeuge@tiefgestellt		\forestov .	2544, 2548,
\DOWNarrow	2850	... 1138, 1139, 1143		2549, 2552, 2553,	
\draw	1343, 1346,	\expandafter		2554, 2555, 2557, 2558	
1349, 2199, 2498, 2501		1323, 2358, 2360,		\forestset	2540, 2563
E		2361, 2362, 2370, 2528		\forestSortLevel	
\edef	1334,	\ExplSyntaxOff ...	58,	2542, 2550, 2564, 2565	
2416, 2420, 2432, 2433		101, 143, 148, 201,		\frac	1842, 1875, 1907, 1922
\edge	278	206, 211, 550, 572,		\fullouterjoin	2455
\else	592, 600,	587, 1119, 1197,		G	
608, 1133, 1147,		1232, 1374, 1453,		\g	39, 47, 281, 282, 302,
1183, 1710, 1813,		1608, 1624, 1788,		306, 312, 313, 314,	
1823, 1833, 1847,		1863, 2053, 2227,		315, 316, 318, 319,	
2071, 2108, 2194,		2477, 2538, 2663,		321, 322, 324, 325,	
2362, 2555, 2557, 2934		2770, 2823, 2902, 2973		326, 327, 328, 329,	
\emph	1095,	\ExplSyntaxOn ..	35, 70,	332, 336, 337, 338,	
1491, 1520, 1522, 1669		111, 144, 169, 202,		341, 343, 344, 345,	
\empty	1710, 2107	207, 232, 493, 556,		346, 347, 348, 359,	
\emptyset		573, 1092, 1187,		360, 361, 362, 370,	
2017, 2729, 2750, 2783		1209, 1370, 1394,		371, 373, 379, 380,	
\end	678,	1601, 1613, 1689,		382, 383, 385, 386,	
701, 726, 761, 795,		1802, 2035, 2208,		394, 399, 401, 406,	
827, 855, 871, 885,		2465, 2535, 2590,		408, 412, 1396,	
893, 920, 949, 970,		2686, 2780, 2874, 2927		1397, 1398, 1401,	
995, 1020, 1040,		F		1405, 1409, 1413,	
1060, 1078, 1174,		\faCheckSquare0	2782	1414, 1415, 1416,	
1263, 1352, 1433,		\faCircleThin	1111	1417, 1419, 1420,	
1439, 1443, 1447,		\faGg	1105	1422, 1438, 1442, 1446	
1476, 1477, 1484,		\fancyfoot		\Gamma .	115, 173, 216, 1152
1593, 1667, 1716,		1267, 1268, 1269,		\gappto	2560
1724, 1743, 1759,		1617, 1618, 1619, 1620		\geometry	5
1760, 1774, 1785,		\fancyhead		\geq	1517, 1873,
1786, 1867, 1883,		1266, 1614, 1615, 1616		2240, 2245, 2261, 2265	
1908, 1931, 1946,		\faSquare0	1099	H	
1956, 2032, 2033,		\fi			

<code>\href</code>	395, 1205, 1794, 1960, 2618	113, 114, 115, 116, 117, 118, 119, 122, 123, 124, 125, 126, 127, 128, 134, 135, 136, 137, 138, 139, 140, 171, 172, 173, 174, 175, 176, 177, 180, 181, 182, 183, 184, 185, 186, 192, 193, 194, 195, 196, 197, 198, 558, 561, 566, 567, 570, 575, 576, 579, 580, 585, 1211, 1212, 1213, 1214, 1217, 1218, 1219, 1220, 1226, 1227, 1228, 1229, 1604, 1605, 1606, 1749, 1750, 1751, 1758	2423, 2424, 2434, 2526, 2564, 2565, 2782 <code>\li@chomsky@erklaerung@texte</code> 507, 543 <code>\li@EntwurfsCode</code> 635, 681, 682, 683, 729, 730, 731, 732, 798, 799, 800, 801, 802, 803, 830, 831, 832, 833, 834, 835, 896 <code>\li@EntwurfsCodeAllgemein</code> 634 <code>\li@fussnote@text</code> 1981, 1987, 1991, 1995, 1999 <code>\li@GithubLink</code> 2614, 2627, 2637, 2648 <code>\li@mget</code> . 1325, 1329, 1349 <code>\li@minc</code> 1328, 1350 <code>\li@mset</code> 1322, 1330, 1339, 1342 <code>\li@numdiscs</code> 1334, 1343, 1349 <code>\li@Rmodell@Schrift</code> 2464, 2473, 2483 <code>\li@sequence</code> . 1335, 1348 <code>\li@synthese@erklaerung@texte</code> 2703, 2763 <code>\liAbleitung</code> 1166 <code>liAdditum</code> (environment) 1717 <code>liAHuelle</code> (environment) 2076 <code>\liAlphabet</code> 1151 <code>liAntwort</code> (environment) 1688 <code>\liAnweisung</code> 1595 <code>\liAssemblerCode</code> . . . 2652 <code>\liAssemblerDatei</code> . . 2653 <code>\liAttribut</code> 2483 <code>\liAttributHuelle</code> 2067, 2715, 2720 <code>\liAttributHuelleOhneMathe</code> 2067, 2070, 2072, 2086, 2096, 2104 <code>\liAttributMenge</code> 2075, 2087, 2090, 2097, 2098, 2112, 2114 <code>\liAufgabe</code> 16 <code>\liAufgabenMetadaten</code> 51, 1425 <code>\liAufgabenTitel</code> 59 <code>\liAusdruck</code> 1188 <code>\liAutomat</code> 70 <code>\liAutomatenKante</code> . . . 102 <code>\liBandAlphabet</code> 1152 <code>\liBedingung</code> 1596 <code>\liBedingungDrei</code> 1860, 1903, 1943 <code>\liBedingungEins</code> 1854, 1892, 1935 <code>\liBedingungFalsch</code> . 1598 <code>\liBedingungWahr</code> . . . 1597	
I				
<code>\i</code>	1348, 1349			
<code>\ifcase</code>	2359			
<code>\ifmmode</code>	590, 598, 606, 1131, 1145, 1181, 1811, 1821, 1831, 1845, 2069, 2192, 2932			
<code>\IfNoValueTF</code>				
.	1720, 1790, 1794			
<code>\ifnum</code>	2286, 2527, 2543, 2551, 2557			
<code>\ifx</code>	1710, 2107, 2555			
<code>\in</code>	516, 622, 1517, 1855, 1858, 1861, 1890, 1896, 1901, 2240, 2251, 2261, 2271, 2712, 2714, 2720, 2741, 2890	<code>\labelenumi</code> 1256 <code>\labelenumii</code> 1257 <code>\labelitemi</code> 1250 <code>\labelitemii</code> 1251 <code>\labelitemiii</code> 1252 <code>\labelitemiv</code> 1253 <code>\land</code> 2969, 2971 <code>\LARGE</code> 1245 <code>\large</code> 1469, 2514 <code>\leaders</code> 2790 <code>\left</code> 1805 <code>\LEFTarrow</code> 2851 <code>\leftarrow</code> 604 <code>\leftouterjoin</code> 2453 <code>\leftskip</code> . 2953, 2954, 2964 <code>\LehramtInformatikAutorEmail</code> 1620 <code>\LehramtInformatikAutorName</code> 1619 <code>\LehramtInformatikGitBranch</code> 398, 2598 <code>\LehramtInformatikGithubCodeRepo</code> 2597 <code>\LehramtInformatikGithubDomain</code> 2594 <code>\LehramtInformatikGithubRawDomain</code> 396, 2595 <code>\LehramtInformatikGithubTexRepo</code> 397, 2596 <code>\LehramtInformatikRepository</code> 17, 20, 23, 26, 29, 1411, 1640, 1641, 1642, 1643, 1644, 1645, 1646, 1647, 1648, 1649, 2593 <code>\LehramtInformatikTitel</code> 1615 <code>\leq</code> 1907, 2248, 2268 <code>\let</code> 1141, 1142, 1664, 2399, 2400, 2401, 2402, 2415, 2417, 2418, 2419, 2421, 2422,		
<code>\inhaltsverzeichnis</code> .	1661			
<code>\input</code>	17, 20, 23, 26, 29, 418, 1410			
<code>\inputminted</code>	2622, 2632, 2642, 2654, 2657, 2661			
<code>\int</code>	2537			
<code>\item</code>	487, 488, 707, 711, 716, 721, 765, 774, 779, 787, 859, 864, 868, 889, 925, 930, 937, 945, 974, 979, 983, 988, 1064, 1069, 1074, 1474, 1475, 1746, 1750, 1870, 1875, 1879, 1889, 1895, 1900, 1912, 1916, 1920, 1924, 1928, 1935, 1939, 1943, 2245, 2248, 2251, 2265, 2268, 2271			
<code>\itshape</code>	541, 2761			
J				
<code>\j</code>	1338, 1339, 1341, 1342, 1343, 1348, 1349, 1350			
K				
<code>\k</code>	1348			
<code>\keys</code>	41, 79, 91, 121, 131, 179, 189, 310, 560, 564, 578, 583, 1216, 1223			
L				
<code>\l</code>	72, 73, 74, 75, 76, 77, 80, 81, 82, 83, 84, 86, 88, 93, 94, 95, 96, 97, 98,			

\liBedingungZwei	1857, 1898, 1939	\liEntwurfsEinzelstueckAkteur	887, 900	\liFussnoten	2002
\liBeschriftung	1677	\liEntwurfsEinzelstueckCode	895, 901	\liFussnoteUrl	1041, 1789
\liBindeAufgabeEin	1408	\liEntwurfsEinzelstueckUml	877, 899	\liFussnoteVierText	1998, 2015
\liChomskyErklaerung	507, 548	\liEntwurfsErbauer	951	\liFussnoteZweiText	1990, 2009
\liChomskyUeberErklaerung	546	\liEntwurfsErbauerAkteure	923, 953	\liGeschweifteKlammern	1170, 2121, 2794
\liChomskyUeberschrift	495, 547	\liEntwurfsErbauerUml	903, 952	\liGrammatik	1209
\liCpmEreignis	556	\liEntwurfsFabrikmethode	997, 1022	liGraphenFormat (environment)	1315
\liCpmFruehsterI	613	\liEntwurfsFabrikmethodeAkteure	972, 999	\liHanoi	1322
\liCpmSpaetesterI	612	\liEntwurfsFabrikmethodeUml	955, 998	\liHaskellCode	2659
\liCpmVon	596	\liEntwurfsKompositum	1022	\liHaskellDatei	2660
\liCpmVonOhneMathe	596, 599, 601	\liEntwurfsKompositumAkteure	1024	\liInduktionAnfang	2877
\liCpmVonZu	588	\liEntwurfsKompositumUml	1001, 1023	\liInduktionErklaerung	2876
\liCpmVonZuOhneMathe	588, 591, 593	\liEntwurfsModellPraesentation	1043	\liInduktionMarkierung	2875
\liCpmVorgang	573	\liEntwurfsModellPraesentationAkteure	1045	\liInduktionSchritt	2893
\liCpmZu	604	\liEntwurfsModellPraesentationSteuerung	1026, 1044	\liInduktionVoraussetzung	2885
\liCpmZuOhneMathe	604, 607, 609	\liEntwurfsModellPraesentationSteuerungUml	1080	\liJavaCode	2612
liDiagramm (environment)	1776	\liEntwurfsZustand	1062, 1082	\liJavaDatei	636, 2621
liEinbettung (environment)	1687	\liEntwurfsZustandAkteure	1047, 1081	\liJavaExamen	2641
\liEntwurfs	1043	\liEntwurfsZustandUml	1137	\liJavaTestDatei	2631
\liEntwurfsAbstrakteFabrik	685	\liEpsilon	1095, 1109, 1111	liKasten (environment)	1260
\liEntwurfsAbstrakteFabrikCode	680, 687	\liErAttribute	1114	\liKellerAutomat	111
\liEntwurfsAbstrakteFabrikUml	638, 686	\liErDatenbankName	1093, 1097, 1099	\liKellerKante	149
\liEntwurfsAdapter	734	\liErLedigt	2782	\liKellerUebergang	144, 150
\liEntwurfsAdapterAkteure	704, 736	\liErMpAttribute	1108	\liKontrollCode	1599
\liEntwurfsAdapterCode	728, 737	\liErMpEntity	1096	liKontrollflussgraph (environment)	1587
\liEntwurfsAdapterUml	689, 735	\liErMpRelationship	1102	\liKontrollKnotenPfad	1601
\liEntwurfsBeobachter	805	\liErRelationship	1094, 1103, 1105	\liKontrollTextzeileKnoten	1600, 1605
\liEntwurfsBeobachterAkteure	763, 807	\liExamensAufgabe	19	\liKurzeTabellenLinie	618
\liEntwurfsBeobachterCode	797, 808	\liExamensAufgabeA	28	\liLadeAllePakete	237, 1391
\liEntwurfsBeobachterUml	739, 806	\liExamensAufgabeTA	25	\liLadePakete	63, 66, 233, 238, 494, 555, 1128, 1358, 1379, 1459, 1979, 2059, 2207, 2576, 2685
\liEntwurfsDekorierer	837	\liExamensAufgabeTTA	22	\liLatexCode	2613
\liEntwurfsDekoriererAkteure	839	\liExkurs (environment)	1725	\liLeereZelle	2017
\liEntwurfsDekoriererCode	829, 840	\liFalsch	488	liLernkartei (environment)	1766
\liEntwurfsDekoriererUml	810, 838	\liFlaci	1198	\liLinksReduktion	2084
\liEntwurfsEinfacheFabrik	873	\liFremd	2463	\liLinksReduktionInline	2093, 2101
\liEntwurfsEinfacheFabrikAkteure	857, 875	\liFunktionaleAbhaengigkeit	2106, 2109, 2117	\liMasterExkurs	1948
\liEntwurfsEinfacheFabrikUml	842, 874	\liFunktionaleAbhaengigkeiten	2120	\liMasterFaelle	1887, 1955
\liEntwurfsEinzelstueck	898	\liFussnote	1980, 1982	\liMasterFallRechnung	1933
		\liFussnoteDreiText	1994, 2012	\liMasterVariablen	1864, 1950
		\liFussnoteEinsText	1986, 2006	\liMasterVariablenDeklaration	1910
		\liFussnoteLink	1793	\liMasterWolframLink	1958

<code>\liMenge</code> ... 80, 81, 83, 122, 123, 124, 128, 180, 181, 182, 186, 1129, 1178, 1217, 1218	<code>liQuellen</code> (environment) 1746	<code>liUebergangsTabelle</code> (environment) 2025
<code>\liMengeOhneMathe</code> 1129, 1132, 1134	<code>\liRechtsReduktionInline</code> 2101	<code>\liUeberschriftDreiecksTabelle</code> 2035
<code>\liMetaSetze</code> 36, 52, 1426	<code>\liRekursionsGleichung</code> 1851, 1914	<code>\liUmlLeserichtung</code> . 2845
<code>\liMinimierungErklaerung</code> 2039	<code>\liRelation</code> 2131	<code>\liVertauschen</code> 2491
<code>\liMinispracheDatei</code> 2656	<code>liRelationenSchemaFormat</code> (environment) 2484	<code>\liWortInSprache</code> 619
<code>\linespread</code> 2471	<code>\liRelationMenge</code> ... 2478	<code>\liWortNichtInSprache</code> 624
<code>\liNichtsZuTun</code> 2783	<code>\liRichtig</code> 487	<code>\liWpEquivalent</code> 2948
<code>\liO</code> 1827, 1855	<code>liRmodell</code> (environment) 2464	<code>\liWpErklaerung</code> 2951
<code>\liOmega</code> 1817, 1861	<code>\liRundeKlammer</code> . 1804, 1808, 1818, 1828, 1842	<code>\liWpErklaerungVerzweigung</code> 2966
<code>\liOmegaOhneMathe</code> 1817, 1822, 1824	<code>\liSetzeExamen</code> 1395	<code>\liWpKalkuel</code> 2928
<code>\liOOhneMathe</code> 1827, 1832, 1834	<code>\liSetzeExamenTeilaufgabeNr</code> 1404	<code>\liWpKalkuelOhneMathe</code> 2928, 2933, 2935, 2967, 2969, 2971
<code>\liParagraphMitLinien</code> . 543, 2041, 2763, 2784, 2881, 2889, 2897	<code>\liSetzeExamenThemaNr</code> 1400	<code>\liZustandsBuchstabe</code> 1153, 1162, 1164, 1182, 1184
<code>\liPetriErreichKnotenDrei</code> 2201	<code>\liSortierMarkierung</code> 2503	<code>\liZustandsBuchstabeGross</code> ... 1154, 1163, 1165
<code>\liPetriErreichTransition</code> 2198	<code>\liSortierPfeil</code> 2497	<code>\liZustandsmenge</code> ... 1141
<code>\liPetriSetzeSchluessel</code> 2143	<code>\liSortierPfeilUnten</code> 2500	<code>\liZustandsmengeNr</code> 1155, 2222
<code>\liPetriTransitionsName</code> 2190, 2202	<code>\liSpaltenUmbruch</code> .. 2572	<code>\liZustandsmengeNrGross</code> 1163
<code>\liPetriTransitionsNameOhneMathe</code> ... 2190, 2193, 2195	<code>\liSqlCode</code> 2664	<code>\liZustandsMengenSammlung</code> 2209
<code>\liPetriTransPfeile</code> 2202	<code>\listen@punkt</code> .. 1746, 1758	<code>\liZustandsMengenSammlungNr</code> 2218
<code>\liPolynomiellReduzierbar</code> 1479	<code>\liStrich</code> 1461	<code>\liZustandsmengeOhneMathe</code> 1141
<code>\liPotenzmenge</code> 1138, 1142, 2213	<code>\liSyntheseErklaerung</code> 2703, 2768	<code>\liZustandsname</code> 1164
<code>\liPotenzmengeOhneMathe</code> ... 1139, 1140, 1141	<code>\liSyntheseUeberErklaerung</code> 2766	<code>\liZustandsnameGross</code> ... 1165, 2210, 2219
<code>\liPrimaer</code> 2462	<code>\liSyntheseUeberschrift</code> 2687, 2767	<code>\liZustandsnameTiefgestellt</code> 1180
<code>\liProblemBeschreibung</code> 1463	<code>\liT</code> 1837, 1852, 1866, 1930	<code>\liZustandsPaar</code> 2019
<code>\liProblemClique</code> ... 1486	<code>\liTeilen</code> 2064	<code>\liZustandsPaarVariablenName</code> ... 2018, 2021, 2022
<code>\liProblemName</code> 1462, 1469, 1481, 1483, 1496, 1507, 1508, 1516, 1517	<code>\liLiteratur</code> 1628, 1652	<code>\llap</code> 2451
<code>\liProblemSat</code> 1515	<code>\liTheta</code> 1807, 1858, 1890, 1896, 1901	<code>\log</code> 1855, 1858, 1861, 1890, 1896
<code>\liProblemSubsetSum</code> 1506, 1515	<code>\liThetaOhneMathe</code> 1807, 1812, 1814	<code>\loop</code> 2284
<code>\liProblemVertexCover</code> 1486, 1494	<code>\liTOhneMathe</code> 1837, 1846, 1848	<code>\lor</code> 2970
<code>\liProduktionen</code> 1177, 1219	<code>\liTuringKante</code> 212	<code>\ltimes</code> 1966
<code>liProduktionsRegeln</code> (environment) 1167	<code>\liTuringLeerzeichen</code> 168, 176	
<code>liProjektSprache</code> (envi- ronment) 1686	<code>\liTuringMaschine</code> ... 169	
<code>\liPseudoUeberschrift</code> 1671, 1721, 1722, 2027, 2037, 2878, 2886, 2894	<code>\liTuringUeberfuehrung</code> 215	
<code>\liPumpingKontextfrei</code> 2259	<code>\liTuringUebergaenge</code> 207, 213	
<code>\liPumpingRegulaer</code> . 2238	<code>\liTuringUebergangZelle</code> 202	
	<code>\liTypoUeberGROSS</code> ... 1438, 1442, 1446, 2817	
	<code>\liTypoUeberGross</code> 1432, 2811, 2817	
	<code>\liTypoUeberschrift</code> 2805, 2814, 2820	
	<code>\liUeberfuehrungsFunktion</code> 1143	
	<code>\liUeberfuehrungsFunktionOhneMathe</code> ... 1143, 1146, 1148	
		M
		<code>\makeatletter</code> .. 1762, 2523
		<code>\makeatother</code> .. 1764, 2531
		<code>\marginpar</code> 1098, 1104, 1110, 1669
		<code>\mathbb</code> .. 1517, 2271, 2890
		<code>\mathbin</code> . 2453, 2454, 2455
		<code>\mathcal</code> .. 1828, 2741, 2746, 2748, 2749, 2750
		<code>\Mathe</code> 2945
		<code>\MatheEnv</code> 2938, 2946, 2949
		<code>\mathord</code> 1973, 1974
		<code>\mdfsetup</code> 1259, 1695, 1699, 1703, 1707
		<code>\mathrel</code> 1471, 1680, 1682, 1756,

1784, 2469, 2475, 2792, 2796, 2939, 2943	\nolinkurl 401, 2618	Q
\memph 1669	\normalsize 1247	\QS@list
\mintinline 2612,	\notin 627	2405, 2416, 2420,
2613, 2652, 2659, 2664	\null 2790	2427, 2433, 2438, 2441
\mkern . . . 2453, 2454, 2455	O	\QS@select@equal
\mlq 1971, 1973	\o@join 2380, 2384
\mrq 1971, 1974	2450, 2453, 2454, 2455	\QS@select@greater . .
\msg 48, 416	\Omega 1818 2381, 2385
\myList	\omega 2240, 2241, 2261, 2262	\QS@select@smaller . .
2544, 2545, 2546, 2549	\or 2361	. . . 2376, 2379, 2383
\myNodes 2533,	P	\QS@sort@a
2548, 2554, 2558, 2560	\pagestyle 1270	2358, 2391, 2412, 2413
N	\par . . . 367, 388, 547,	\QS@sort@b 2358, 2359
\NeedsTeXFormat . 1, 14,	1199, 1207, 1678,	\QS@sort@c 2362, 2369
32, 61, 229, 422,	1740, 1763, 1771,	\QS@sort@d 2370, 2378
483, 490, 552, 615,	2403, 2425, 2440,	\QS@sort@empty . 2360, 2365
630, 1085, 1121,	2481, 2767, 2791,	\QS@sort@single 2361, 2366
1234, 1274, 1283,	2795, 2803, 2956, 2962	\QS@initialize
1288, 1317, 1355,	\paragraph 1247	. . . 2283, 2395, 2430
1376, 1455, 1527,	\parindent 2472	\QS@Ir . 2366, 2372, 2380,
1610, 1626, 1632,	\path . . . 103, 150, 213, 585	2400, 2414, 2419, 2422
1654, 1800, 1963,	\pgfkeys . . 2154, 2848,	\QS@Irr 2401, 2414, 2415, 2423
1976, 2055, 2138,	2849, 2850, 2851,	\QSLr 2372,
2204, 2229, 2234,	2852, 2855, 2858, 2860	2379, 2390, 2391,
2278, 2444, 2457,	\pgfmath@count	2399, 2412, 2417, 2421
2486, 2567, 2574,	. . . 2525, 2527, 2529	\QS@pivotStep
2581, 2586, 2666,	\pgfmath@smuggleone 2530	2285, 2395, 2399, 2410
2681, 2772, 2776,	\pgfmathdeclarefunction	\QSRr 2372
2825, 2867, 2904, 2924 2524	\QSRr 2381,
\neg 2971	\pgfmathhint 2525	2402, 2413, 2418,
\negthinspace 1805	\pgfmathparse	2424, 2433, 2434, 2435
\newcounter . . . 2298, 2299 1329, 2542,	\QS@sortStep
\NewDocumentCommand .	2547, 2550, 2564, 2565	2287, 2395, 2411, 2412
71, 112, 149, 170,	\pgfmathresult	\quad 1983
212, 233, 557, 574, 1330, 2525,	R
619, 624, 1188,	2526, 2528, 2530,	\raisebox 1600
1210, 1479, 1602,	2543, 2551, 2564, 2565	\relax 1664,
1630, 1789, 1793,	\pgfutil@empty 2526	2372, 2415, 2417,
2120, 2131, 2198,	\pgfutil@loop 2527	2418, 2419, 2527, 2529
2621, 2631, 2641,	\pgfutil@repeat 2530	\renewcommand . . . 1250,
2653, 2656, 2660, 2845	\preceq 1482	1251, 1252, 1253,
\NewDocumentEnvironment	\prime 1461	1256, 1257, 1271,
1167, 1260, 1315,	\printbibliography . 1652	1272, 1621, 1622, 2025
1587, 1686, 1687,	\ProvidesPackage 2, 15,	\repeat 2288
1690, 1717, 1725,	33, 62, 230, 423,	\RequirePackage
1747, 1766, 1776,	484, 491, 553, 616, 4, 64, 167,
2026, 2076, 2466, 2484	631, 1086, 1122,	231, 235, 419, 425,
\newlength 2951	1235, 1275, 1284,	426, 486, 554, 633,
\node 570, 1595,	1289, 1318, 1356,	634, 1088, 1090,
1600, 2324, 2329,	1377, 1456, 1528,	1091, 1127, 1236,
2334, 2340, 2346,	1611, 1627, 1633,	1237, 1240, 1242,
2352, 2503, 2548, 2862	1655, 1801, 1964,	1244, 1249, 1258,
\noexpand 2412,	1977, 2056, 2139,	1265, 1276, 1277,
2413, 2414, 2433, 2548	2205, 2230, 2235,	1286, 1290, 1291,
\noindent 368,	2279, 2445, 2458,	1292, 1320, 1321,
621, 626, 1673,	2487, 2568, 2575,	1369, 1387, 1392,
1675, 1679, 1683,	2582, 2587, 2667,	1393, 1460, 1529,
1711, 1739, 1741,	2682, 2773, 2777,	1629, 1634, 1635,
1754, 1770, 1772,	2826, 2868, 2905, 2925	1651, 1658, 1659,
1780, 1952, 2005,		1660, 1688, 1803,
2008, 2011, 2014,		1967, 1968, 2140,
2479, 2785, 2797, 2957		2142, 2232, 2292,
		2294, 2446, 2447,

2448, 2461, 2489, 2520, 2571, 2577, 2584, 2589, 2600, 2601, 2669, 2774, 2781, 2829, 2830, 2832, 2833, 2871, 2872, 2873, 2906, 2926	\right 1805 \RIGHArrow 2847, 2852 \Rightarrow 622, 627 \rightarrow 216, 511, 516, 524, 528, 530, 531, 533, 588, 596, 2202, 2712, 2719, 2721, 2724, 2729, 2734, 2735, 2740 \rightouterjoin 2454 \Roman 1255 \roman 1255, 1257 \romannumeral 2370 \rtimes 1966 \rule 2403, 2425, 2440, 2451	\sigma 521, 523, 524 \SLASH 1670 \small 1779 \sort 2537 \sortList 2536, 2545 \square 488 \stepcounter 2324, 2329, 2334, 2337, 2339, 2343, 2345, 2349, 2351 \str .. 499, 508, 1692, 2214, 2223, 2691, 2704 \string 2087, 2097 \StrSubstitute . 2544, 2546 \strut 1912, 1916, 1920, 1924, 1928, 2572 \subsection 1406 \subsepeq 2715, 2748, 2755 \subsubsection 1427	122, 123, 124, 125, 126, 127, 128, 171, 172, 173, 174, 175, 176, 177, 180, 181, 182, 183, 184, 185, 186, 303, 307, 332, 336, 337, 338, 341, 346, 347, 348, 359, 360, 361, 362, 373, 379, 382, 385, 394, 408, 558, 561, 566, 567, 575, 576, 579, 580, 1190, 1211, 1212, 1213, 1214, 1217, 1218, 1219, 1220, 1396, 1397, 1398, 1401, 1405, 1409, 1416, 1419, 1838 \tmp 2107 \TmpPlaceEight 2162 \TmpPlaceFive 2159 \TmpPlaceFour 2158 \TmpPlaceNine 2163 \TmpPlaceOne 2155 \TmpPlaceSeven 2161 \TmpPlaceSix 2160 \TmpPlaceTen 2164 \TmpPlaceThree 2157 \TmpPlaceTwo 2156 \TmpScale 2175 \TmpTransitionEight 2151, 2172 \TmpTransitionFive 2148, 2169 \TmpTransitionFour 2147, 2168 \TmpTransitionNine 2152, 2173 \TmpTransitionOne 2144, 2165 \TmpTransitionSeven 2150, 2171 \TmpTransitionSix 2149, 2170 \TmpTransitionTen 2153, 2174 \TmpTransitionThree 2146, 2167 \TmpTransitionTwo 2145, 2166 \TmpX 2176 \TmpY 2177 \today 1616 \ttfamily 2464
S	\sb 76, 86, 88, 117, 175, 523, 524, 528, 531, 532, 533, 1182, 1184, 1855, 1858, 1861, 1890, 1896, 2045, 2190, 2199, 2734, 2735, 2736, 2741, 2745, 2746, 2749, 2750, 2753, 2754, 2755 \scriptscriptstyle 588, 596, 604 \scriptsize 1201, 1540, 1547, 1553, 1615, 1616, 1619, 1620, 2876, 2929, 2959 \section ... 56, 1388, 1402 \sectionbreak 1389 \seq . 1604, 1605, 1606, 1749, 1750, 1751, 1758 \setbox 2450 \setcounter 1248, 1390, 1429, 2404, 2426, 2440 \setganttlinklabel .. 1278, 1279, 1280, 1281 \setlength 1623, 2472, 2953, 2954, 2964 \setmainfont 1238 \setmainlanguage 420 \setminted 2606, 2607 \setminus 2105 \setsansfont 1239 \setul 2463 \sffamily 498, 1245, 1247, 1349, 1388, 2690, 2807 \shoveleft 2085 \shoveright 2089 \Sigma 73, 114, 172, 1151, 1152, 1212	T \tableofcontents 1450, 1665 \text 86, 88, 191, 2067, 2876, 2929 \textbf ... 1093, 1487, 1496, 1507, 1516, 1674, 1681, 1712, 1740, 1755, 1771, 2030 \textcolor 1599, 2875 \textit 962, 1004, 1005, 1006, 1007, 1781, 2075, 2135 \textsc 1462 \textsf 1674, 1755 \textstyle 1875, 1907 \texttt 1116, 1462, 1596, 1597, 1598, 1599, 2929 \textwidth 1623 \thepage 1269, 1618 \theparagraph 1247 \thesection 1388 \Theta 1808 \thinspace 2929 \tikz 1600 tikz: bbaum 25 tikz: li binaer baum 23 \tikzchildnode 443 \tikzparentnode 443 \tikzset 105, 152, 218, 428, 454, 1294, 1531, 2180, 2306, 2510, 2671, 2907 \tikzumlset 2835 \times 216 \tiny 1099, 1105, 1111, 1599, 1669, 2616 \titleformat 1245, 1247, 1388 \titlespacing 1246 \tl 39, 47, 72, 73, 74, 75, 76, 77, 80, 81, 82, 83, 84, 86, 88, 113, 114, 115, 116, 117, 118, 119,	U \ul 1094, 2462, 2463 \umlaggreg 1057 \umlassoc 1039 \umlclass ... 640, 644, 648, 692, 693, 694, 741, 746, 751, 754, 812, 813, 814, 819, 820, 849, 879, 906,

907, 910, 961, 964, 1003, 1009, 1010, 1028, 1029, 1030, 1049, 1050, 1051, 1052	\umlVHVinherit	X
\umldep 969	652, 653, 658, 659, 666, 667, 822, 823, 847, 848, 1017, 1018	\xappto . . 2548, 2554, 2558
\umlHVVHaggreg	\umlVHVreal	\xdef 1323
. 759, 825, 1019	. 816, 817, 1054, 1055	\xintApply 2374
\umlinherit	\UParrow 2849	\xintApplyUnbraced . .
698, 749, 914, 959, 967	\url 1790	2373, 2379, 2380, 2381
\umlnote . . . 700, 916, 1059	\usemintedstyle 2603	\xintCSVtoList 2433
\umlreal 696, 757	\usetikzlibrary . . 65,	\xintFor
\umlsimpleclass . . 655,	427, 1089, 1293,	2323, 2328, 2333,
664, 665, 691, 844,	1530, 2141, 2449,	2338, 2344, 2350, 2391
845, 846, 905, 957, 958	2490, 2521, 2834, 2922	\xintifEq 2384
\umlstatic 851, 880	V	\xintifForLast
\umluniaggreg 912	\value 2286	. . . 2340, 2346, 2352
\umluniassoc 675,	\varepsilon 500,	\xintifGt 2385
697, 913, 1037, 1038	511, 512, 1137,	\xintifLt 2383
\umlVHuniassoc . . 676, 677	1855, 1861, 1893, 1904	\xintLength 2358
\umlVHVdep 669,	\vfill . . . 1435, 1449, 2572	\xintnthelt 2370
670, 672, 673, 853, 854	\vrule 2786, 2790	Z
	\vspace	\ZB 1797
	1726, 1744, 2799, 2801	\zB 1796
		\zustandsnamens@liste
		. . . 1155, 1162, 1163