

lehramt-informatik

Hermine Bschlangaul *

September 5, 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	111
2.48	vollstaendige-induktion.sty	113
2.48.1	Makro-Kürzel	113
2.49	wasserfall.sty	115
2.50	wpkalkuel.sty	116
2.50.1	Makro-Kürzel	116
3	Index	117

1 Klassen

1.1 Vorlage Theorie-Teil

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

\begin{document}

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

\chapter{Thema des Theorie-Teils}

\literatur

\end{document}
```

1.2 Vorlage Aufgabensammlung

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

1.3 Vorlage Aufgabe

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

\end{document}
```

2 Pakete

2.1 abmessung.sty

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

2.2 aufgaben-einbinden.sty

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

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

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

\liExamensAufgabeTTA
22 \def\liExamensAufgabeTTA #1 / #2 / #3 : Thema #4 Teilaufgabe #5 Aufgabe #6 {
23   \input{\LehramtInformatikRepository/Staatsexamen/#1/#2/#3/Thema-#4/Teilaufgabe-
24     #5/Aufgabe-#6.tex}
25 }

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

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

31
```

2.3 aufgaben-metadaten.sty

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

```
35 \ExplSyntaxOn
```

`\liMetaSetze` Setze die Metadaten einer Aufgabe.

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

```

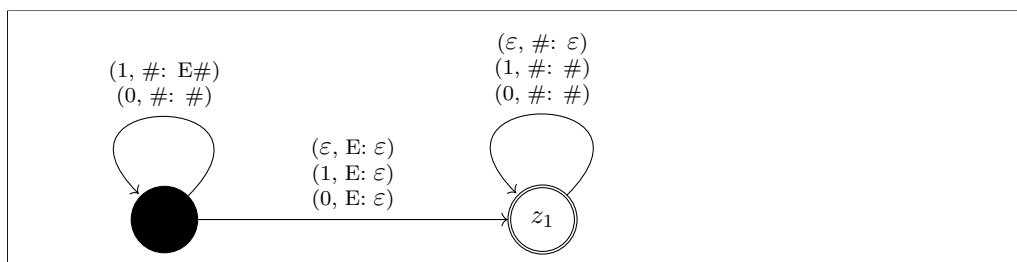
2.4.2 Kellerautomat

```
\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]{\von}{\zu}{\übergange}`
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]
{\zustaende=Z,alphabet= $\Sigma$ ,bandalphabet= $\Gamma$ ,delta= $\delta$ ,start= $z_0$ ,leerzeichen= $\square$ ,ende=E)}

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

```

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

```

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}:`
`(z1: \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`

`(z1: \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     baum,
242     checkbox,
243     chomsky-normalform,
244     cpm,
245     cyk-algorithmus,
246     entwurfsmuster,
247     er,
248     formale-sprachen,
249     gantt,
250     grafik,
251     graph,
252     hanoi,
253     kontrollflussgraph,
254     makros,
255     master-theorem,
256     mathe,
257     minimierung,
258     normalformen,
259     petri,
260     potenzmengen-konstruktion,
261     pseudo,
262     quicksort,
263     relationale-algebra,
264     rmodell,
265     sortieren,
266     spalten,
267     struktogramm,
268     syntax,
269     syntaxbaum,
270     synthese-algorithmus,
271     tabelle,
272     typographie,
273     uml,
274     vollstaendige-induktion,
275     wasserfall,
276     wpkalkuel,
277   }
278 }
```

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

```
279 \clist_new:N \g_auf_schluesssel_clist
280 \clist_set:Nn \g_auf_schluesssel_clist {
```

```

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

```

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

```

300 \clist_map_inline:Nn \g_auf_schluessel_clist {
301   \tl_new:c {g_auf_#1_tl}
302 }

```

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

```

303 \cs_new:Npn \setze_variablen_zurueck: {
304   \clist_map_inline:Nn \g_auf_schluessel_clist {
305     \tl_clear:c {g_auf_##1_tl}
306   }
307 }

```

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

```

308 \keys_define:nn { aufgabenmetadaten }
309 {
310   Titel .tl_gset:N = \g_auf_titel_tl,
311   Thematik .tl_gset:N = \g_auf_thematik_tl,
312   Stichwoerter .tl_gset:N = \g_auf_stichwoerter_tl,
313   ZitatSchluessel .tl_gset:N = \g_auf_zitat_beschreibung_tl,
314   ZitatBeschreibung .tl_gset:N = \g_auf_zitat_schluessel_tl,
315   %
316   BearbeitungsStand .tl_gset:N = \g_auf_bearbeitungs_stand_tl,
317   Korrektheit .tl_gset:N = \g_auf_korrekttheit_tl,
318   %
319   RelativerPfad .tl_gset:N = \g_auf_relativer_pfad_tl,
320   IdentischeAufgabe .tl_gset:N = \g_auf_identische_aufgabe_tl,
321   %
322   ExamenNummer .tl_gset:N = \g_auf_examen_nummer_tl,
323   ExamenJahr .tl_gset:N = \g_auf_examen_jahr_tl,
324   ExamenMonat .tl_gset:N = \g_auf_examen_monat_tl,
325   ExamenThemaNr .tl_gset:N = \g_auf_examen_thema_nr_tl,
326   ExamenTeilaufgabeNr .tl_gset:N = \g_auf_examen_telaufgabe_nr_tl,
327   ExamenAufgabeNr .tl_gset:N = \g_auf_examen_aufgabe_nr_tl,
328 }
329 \cs_gset:Npn \setze_relativen_pfad: {
330   \tl_if_empty:NTF \g_auf_relativer_pfad_tl
331   {
332     \bool_if:nTF
333     {
334       ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
335       ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
336       ! \tl_if_empty_p:N \g_auf_examen_monat_tl

```

```

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

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

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

```

```

400     }
401   }
402 }

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

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

\def\LehramtInformatikRepository{/pfad/zum/repository}
416 \input /etc/lehramt-informatik.config.tex
    biblatex not working with lualatex and babel
417 % \RequirePackage{polyglossia}
418 % \setmainlanguage{german}
419

```

2.6 baum.sty

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

423 \RequirePackage{tikz}

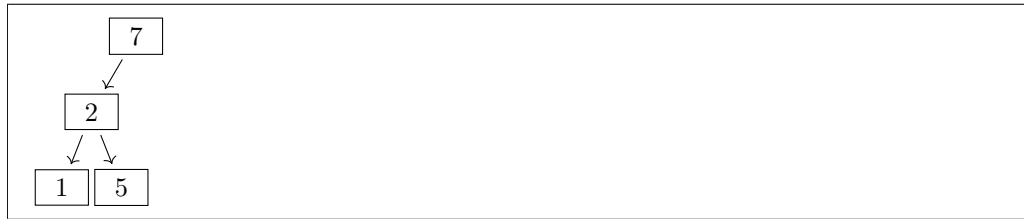
    für li binaer baum
424 \RequirePackage{tikz-qtrees}

    Für b baum
425 \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}
```



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

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}
```

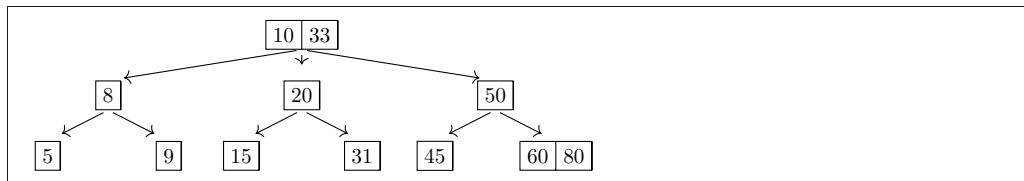


2.6.3 B-Baum

```

\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}

```



```

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

```

2.7 checkbox.sty

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

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

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

487
```

2.8 chomsky-normalform.sty

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

491 \ExplSyntaxOn

492 \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

```

493 \def\liChomskyUeberschrift#1{
494 {
495   \bfseries
496   \sffamily
497   \str_case:nn {#1} {
498     {1} {Elimination-der- $\varepsilon$ -Regeln}
499     {2} {Elimination-von-Kettenregeln}
500     {3} {Separation-von-Terminalzeichen}
501     {4} {Elimination-von-mehrelementigen-Nonterminalketten}
502   }
503 }
504 }

```

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

```

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

```

```

536 }
537 \def\liChomskyErklaerung#1{
538   {
539     \itshape
540     \footnotesize
541     \liParagraphMitLinien{\li@chomsky@erklaerung@texte{#1}}
542   }
543 }

```

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

```

544 \def\liChomskyUeberErklaerung#1{
545   \liChomskyUeberschrift{#1}\par
546   \liChomskyErklaerung{#1}
547 }

```

```

548 \ExplSyntaxOff
549

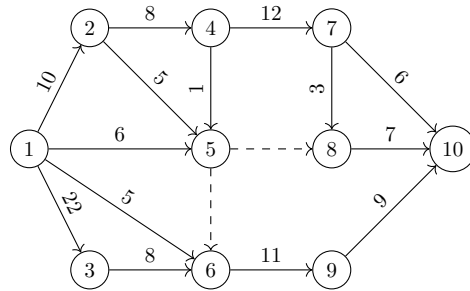
```

2.9 cpm.sty

```
550 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
551 \ProvidesPackage{lehramt-informatik-cpm}[2020/09/03]
552 \RequirePackage{tikz}
553 \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}
```

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

```

567
568 \node[circle,draw] (\l_name_tl) at (#3,#4) {#2};
569 }
570 \ExplSyntaxOff

\liCpmVorgang liCpmVORGANG\((.*)>(.*))\{(.*)\}
571 \ExplSyntaxOn
572 \NewDocumentCommand { \liCpmVorgang } { 0{ } m m m } {
573   \tl_set:Nn \l_schein_tl {}
574   \tl_set:Nn \l_kritisch_tl {}
575
576   \keys_define:nn { cpmVorgang } {
577     schein .code:n = {\tl_set:Nn \l_schein_tl {dashed}},
578     kritisch .code:n = {\tl_set:Nn \l_kritisch_tl {thick}},
579   }
580
581   \keys_set:nn { cpmVorgang } { #1 }
582
583   \path[->,\l_schein_tl,\l_kritisch_tl] (#2) edge node[auto,sloped] {#4} (#3);
584 }
585 \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):  $l_{(2 \rightarrow 3)}$ 
586 \def\liCpmVonZuOhneMathe#1(#2-#3){#1_{\scriptscriptstyle(#2\rightarrow#3)}}
587 \def\liCpmVonZu#1(#2-#3){%
588   \ifmmode%
589     \liCpmVonZuOhneMathe{#1}(\#2-\#3)%
590   \else%
591     $\liCpmVonZuOhneMathe{#1}(\#2-\#3)$%
592   \fi%
593 }

```

```

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

```

```

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

```

```

608 \fi%
609 }

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

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

612

```


2.10 cyk-algorithmus.sty

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

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

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

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

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

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

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

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

```
627
```

2.11 entwurfsmuster.sty

```
628 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
629 \ProvidesPackage{lehramt-informatik-entwurfsmuster}[2021/05/06
630 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

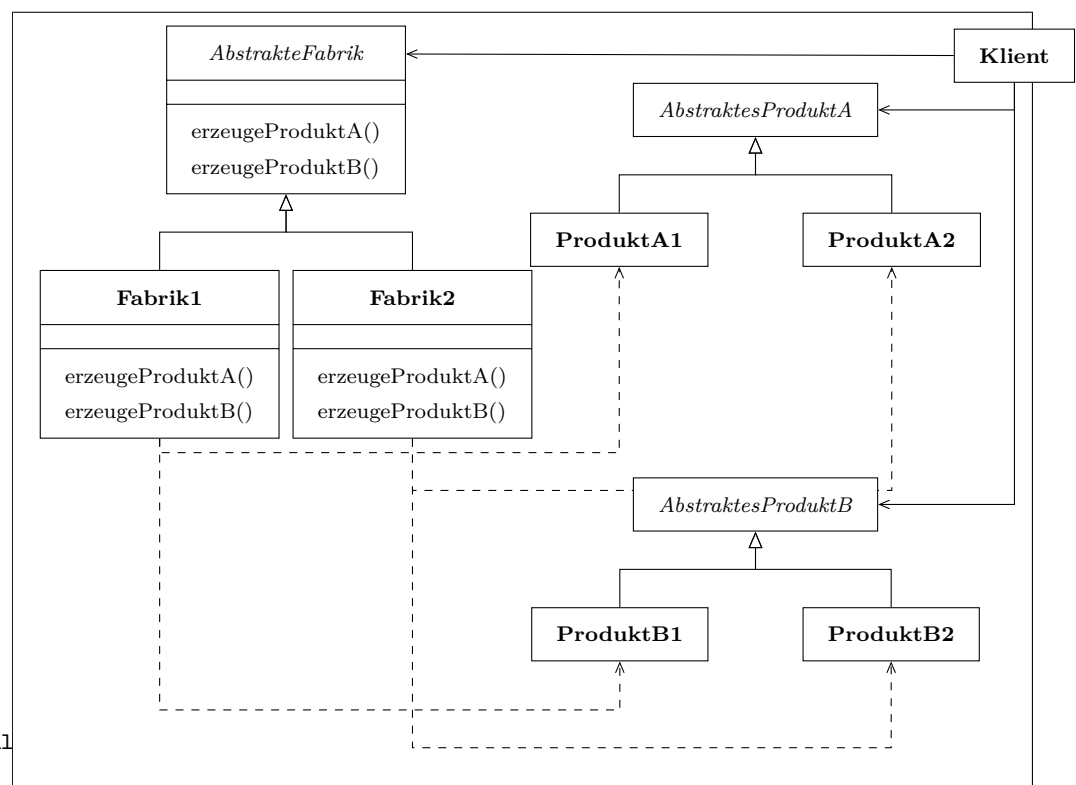
1. **Uml:** Uml-Klassendiagramm \liEntwurfsEinzelstueckUml
2. **Akteure:** Akteure, beteiligte Klassen \liEntwurfsEinzelstueckAkteure
3. **Code:** Allgemeines Code-Beispiel \liEntwurfsEinzelstueckCode
4. **ohne:** Ohne Suffix, Bündelung der einzelnen Makros eines Entwurfsmusters \liEntwurfsEinzelstueckAkteure

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

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

```
632 \RequirePackage{lehramt-informatik-syntax}
633 \def\li@EntwurfsCode#1#2{
634   \liJavaDatei{entwurfsmuster/#1/allgemein/#2}
635 }
```

2.11.3 Abstrakte Fabrik (Abstract Factory)



```

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

```

```

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

```

iEntwurfsAbstrakteFabrikCode

```

678 \def\liEntwurfsAbstrakteFabrikCode{
679     \li@EntwurfsCode{abstrakte_fabrik}{Produkte}
680     \li@EntwurfsCode{abstrakte_fabrik}{AbstrakteFabrik}
681     \li@EntwurfsCode{abstrakte_fabrik}{Klient}
682 }

```

\liEntwurfsAbstrakteFabrik

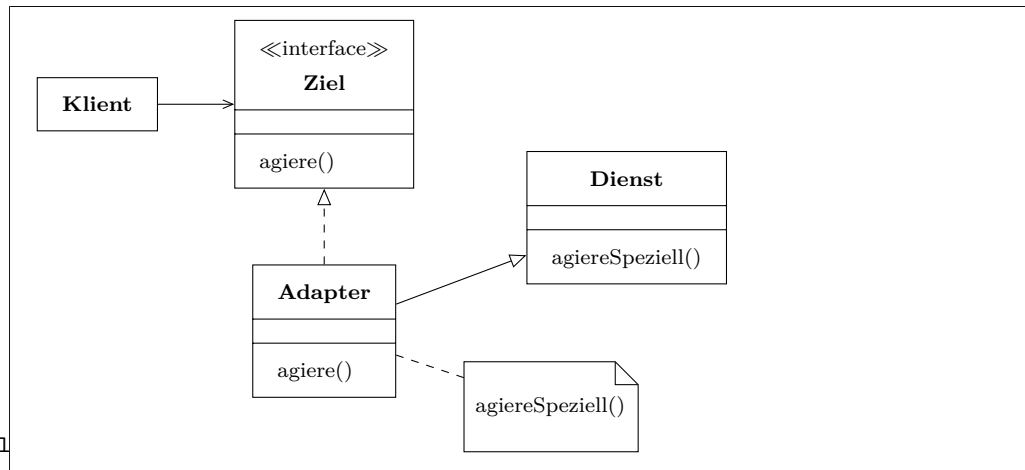
```

683 \def\liEntwurfsAbstrakteFabrik{
684     \liEntwurfsAbstrakteFabrikUml
685     \liEntwurfsAbstrakteFabrikCode
686 }

```

2.11.4 Adapter

\liEntwurfsAdapterUml



```

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

```

\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.

```

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

```

```

723
724 \end{description}
725 }

```

\liEntwurfsAdapterCode

```

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

```

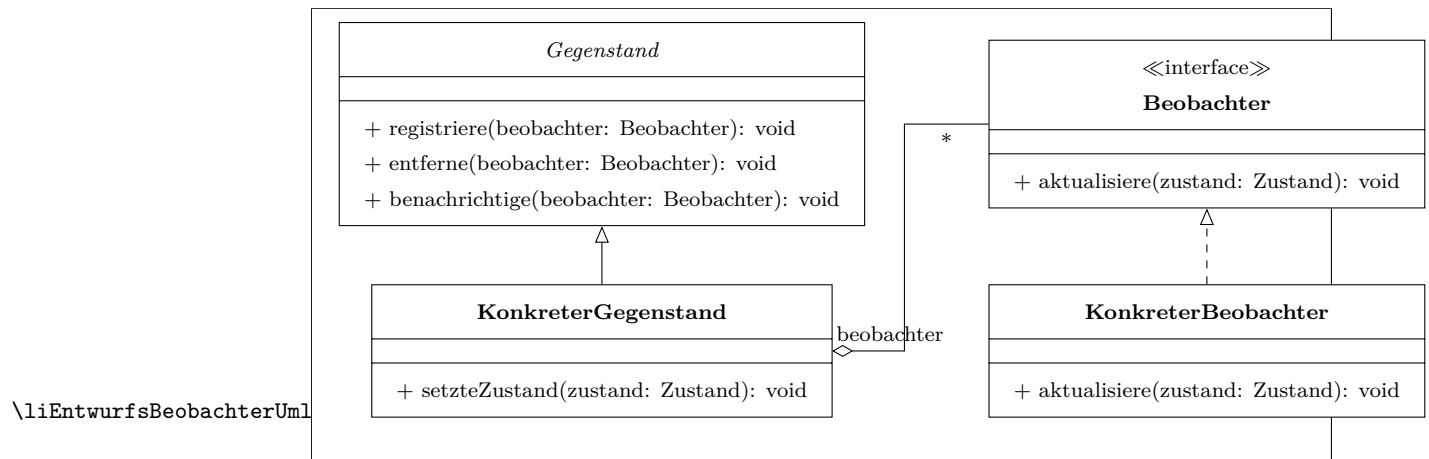
\liEntwurfsAdapter

```

732 \def\liEntwurfsAdapter{
733 \liEntwurfsAdapterUml
734 \liEntwurfsAdapterAkteure
735 \liEntwurfsAdapterCode
736 }

```

2.11.5 Beobachter (Observer)



```

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

```

\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.

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

\liEntwurfsBeobachterCode

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

```

801 \li@EntwurfsCode{beobachter}{Klient}
802 }

```

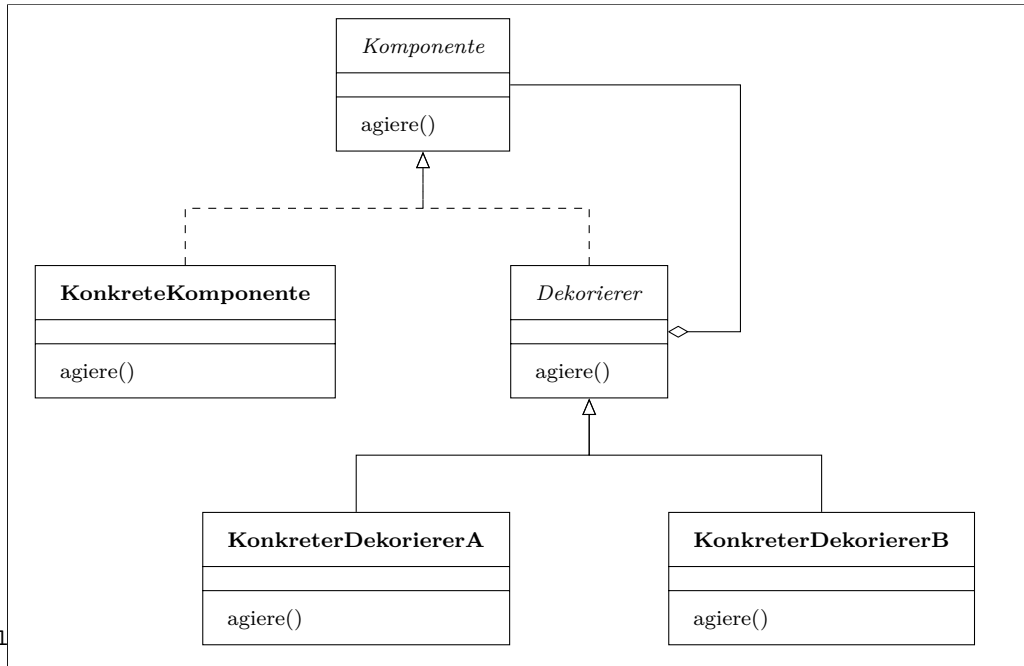
\liEntwurfsBeobachter

```

803 \def\liEntwurfsBeobachter{
804 \liEntwurfsBeobachterUml
805 \liEntwurfsBeobachterAkteure
806 \liEntwurfsBeobachterCode
807 }

```

2.11.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

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

```

\liEntwurfsDekoriererCode

```

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

```

\liEntwurfsDekorierer

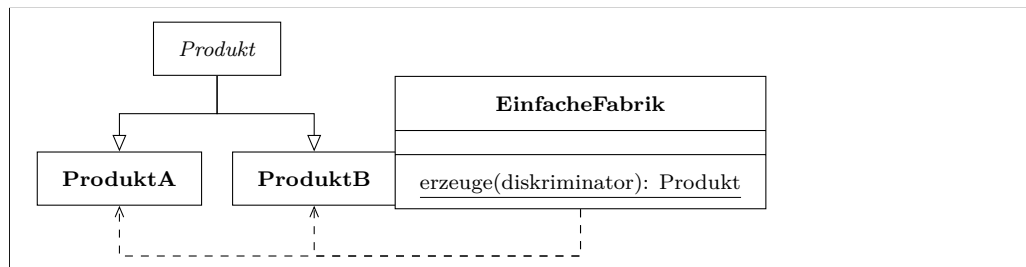
```

835 \def\liEntwurfsDekorierer{
836   \liEntwurfsDekoriererUml
837   \liEntwurfsDekoriererAkteure
838   \liEntwurfsDekoriererCode
839 }

```

2.11.7 Einfache Fabrik (Simple Factory)

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



```

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

```

\liEntwurfsEinfacheFabrikAkteure 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.

```

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

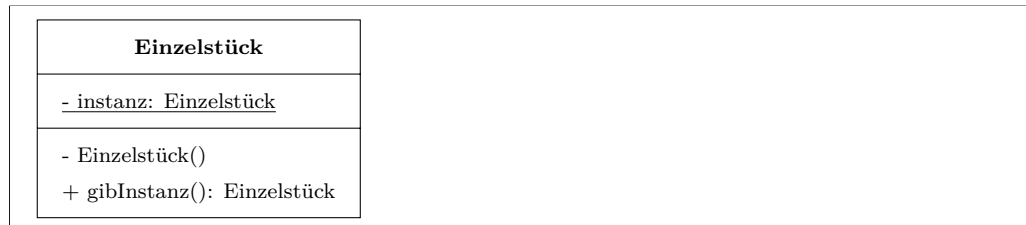
```


\liEntwurfsEinfacheFabrik

```
871 \def\liEntwurfsEinfacheFabrik{
872   \liEntwurfsEinfacheFabrikUml
873   \liEntwurfsEinfacheFabrikAkteure
874 }
```

2.11.8 Einzelstück (Singleton)

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



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

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.

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

\liEntwurfsEinzelstueckCode

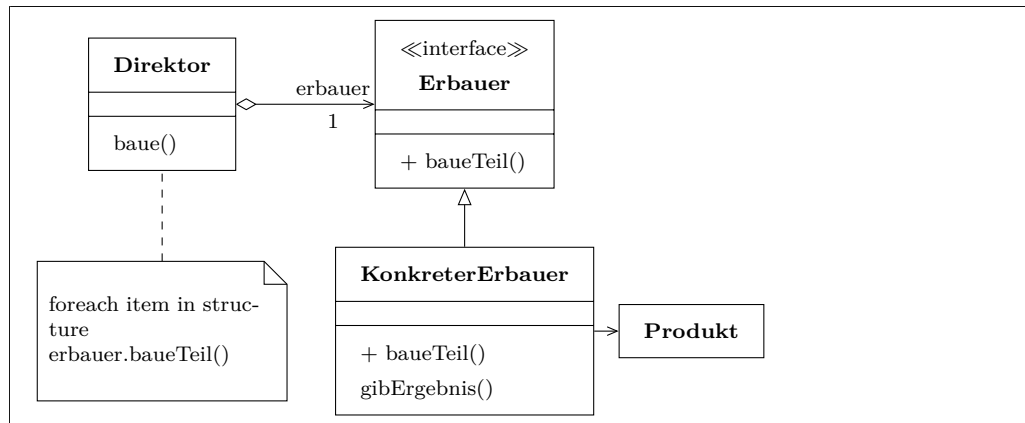
```
893 \def\liEntwurfsEinzelstueckCode{
894   \li@EntwurfsCode{einzelstueck}{Einzelstueck}
895 }
```

\liEntwurfsEinzelstueck

```
896 \def\liEntwurfsEinzelstueck{
897   \liEntwurfsEinzelstueckUml
898   \liEntwurfsEinzelstueckAkteure
899   \liEntwurfsEinzelstueckCode
900 }
```

2.11.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

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

```

\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.

```

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

```

```

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

```

\liEntwurfsErbauer

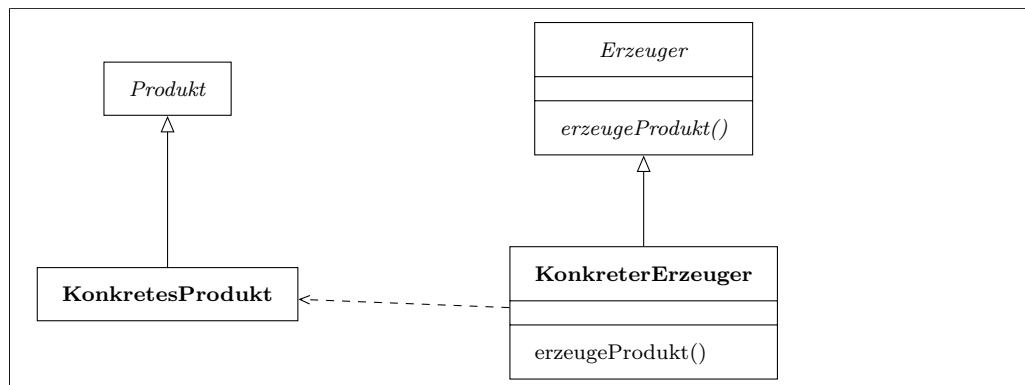
```

949 \def\liEntwurfsErbauer{
950   \liEntwurfsErbauerUml
951   \liEntwurfsErbauerAkteure
952 }

```

2.11.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

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

```

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).

```

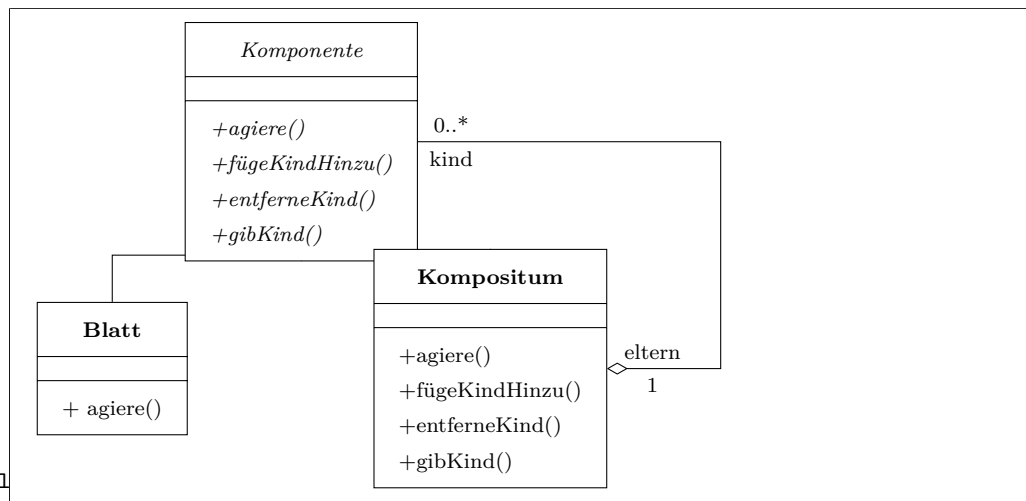
970 \def\liEntwurfsFabrikmethodeAkteure{
971   \begin{description}
972     \item[Produkt]
973
974     Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das
975     zu erzeugende Produkt.
976
977     \item[KonkretesProdukt]
978
979     KonkretesProdukt implementiert die Produkt-Schnittstelle.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

\liEntwurfsFabrikmethode

```

995 \def\liEntwurfsFabrikmethode{
996   \liEntwurfsFabrikmethodeUml
997   \liEntwurfsFabrikmethodeAkteure
998 }
```

2.11.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

999 \def\liEntwurfsKompositumUml{
1000 \begin{tikzpicture}
1001 \umlclass[x=2.5,y=3,type=abstract]{Komponente}{-}{-}{
1002 \textit{+agiere()}\
1003 \textit{+fügeKindHinzu()}\
1004 \textit{+entferneKind()}\
1005 \textit{+gibKind()}}
1006 }
1007 \umlclass[x=0]{Blatt}{-}{+ agiere()}
1008 \umlclass[x=5]{Kompositum}{-}{-}{
1009 +agiere()\
1010 +fügeKindHinzu()\
1011 +entferneKind()\
1012 +gibKind()}
1013 }
1014
1015 \umlVHVinherit{Kompositum}{Komponente}
1016 \umlVHVinherit{Blatt}{Komponente}
1017 \umlHVHaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,]
1018 \end{tikzpicture}
1019 }

```

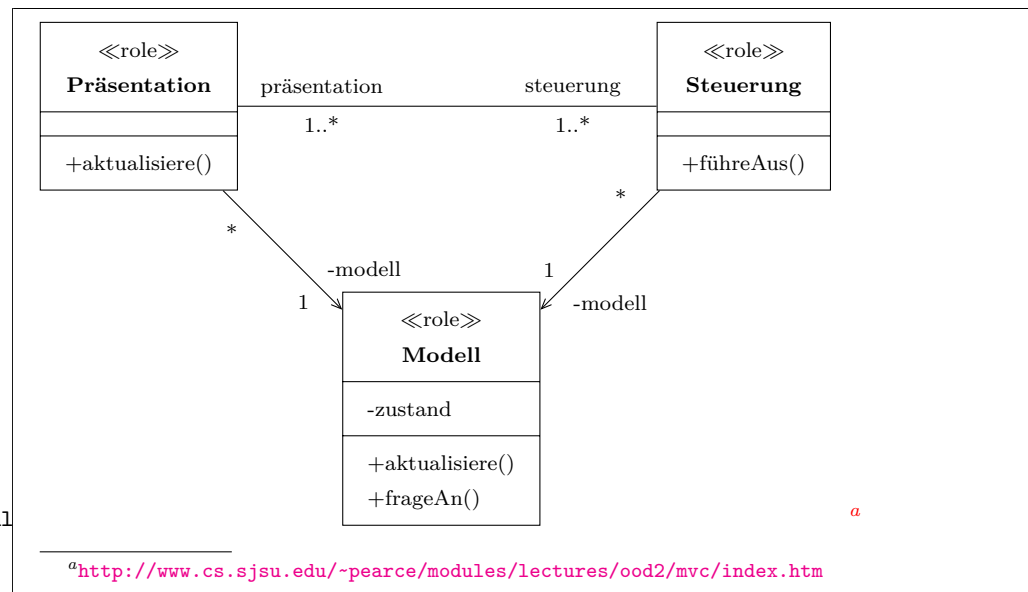
\liEntwurfsFabrikmethode

```

1020 \def\liEntwurfsKompositum{
1021 \liEntwurfsKompositumUml
1022 \liEntwurfsKompositumAkteure
1023 }

```

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



```

1024 \def\liEntwurfsModellPraesentationSteuerungUml{
1025 \begin{tikzpicture}
1026 \umlclass[x=-4cm,type=role]{Präsentation}{-}{+aktualisiere()}
1027 \umlclass[x=4cm,type=role]{Steuerung}{-}{+führeAus()}
1028 \umlclass[y=-4cm,type=role]{Modell}{-}{
1029 -zustand
1030 }{
1031 +aktualisiere()\
1032 +frageAn()
1033 }
1034
1035 \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
1036 \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}

```

```

1037 \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
1038 \end{tikzpicture}
1039 \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
1040 }

```

ModellPraesentationSteuerung

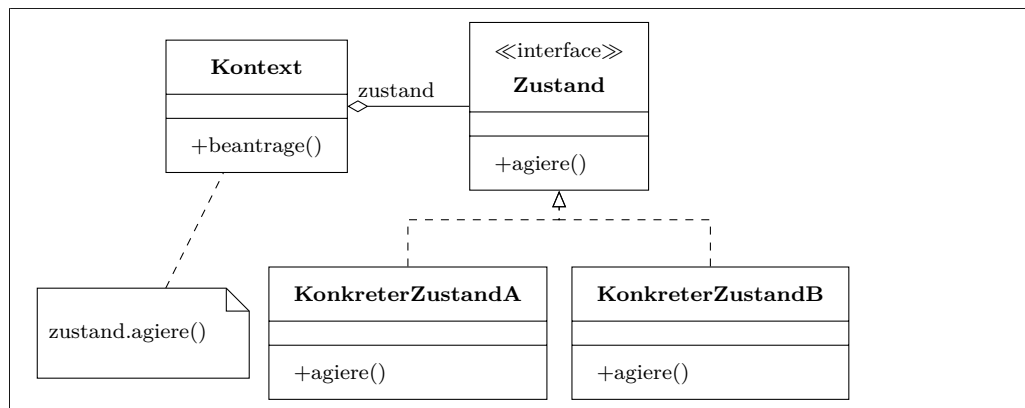
```

1041 \def\liEntwurfs{
1042 \liEntwurfsModellPraesentationSteuerungUml
1043 \liEntwurfsModellPraesentationSteuerungAkteure
1044 }

```

2.11.13 Zustand (State)

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



```

1045 \def\liEntwurfsZustandUml{
1046 \begin{tikzpicture}
1047 \umlcclass[x=-1,y=3]{Kontext}{+beantrage()}
1048 \umlcclass[x=3,y=3,type=interface]{Zustand}{+agiere()}
1049 \umlcclass[x=1,y=0]{KonkreterZustandA}{+agiere()}
1050 \umlcclass[x=5,y=0]{KonkreterZustandB}{+agiere()}
1051
1052 \umlVHVreal{KonkreterZustandA}{Zustand}
1053 \umlVHVreal{KonkreterZustandB}{Zustand}
1054
1055 \umlagg[arg=zustand,pos=0.4]{Kontext}{Zustand}
1056
1057 \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
1058 \end{tikzpicture}
1059 }

```

\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.

```

1060 \def\liEntwurfsZustandAkteure{
1061 \begin{description}
1062 \item[Kontext (Context)]
1063
1064 definiert die clientseitige Schnittstelle und verwaltet die separaten
1065 Zustandsklassen.
1066

```

```

1067 \item[State (Zustand)]
1068
1069 definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
1070 implementiert gegebenenfalls ein Standardverhalten.
1071
1072 \item[KontreterZustand (ConcreteState)]
1073
1074 implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
1075 verbunden ist.
1076 \end{description}
1077 }

```

\liEntwurfsZustand

```

1078 \def\liEntwurfsZustand{
1079 \liEntwurfsZustandUml
1080 \liEntwurfsZustandAkteure
1081 }
1082

```

2.12 er.sty

```
1083 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1084 \ProvidesPackage{lehramt-informatik-er}[2020/06/13 Zum Setzen von
1085 ER-Diagrammen]
```

```
1086 \RequirePackage{tikz-er2}
1087 \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);

```

```
1088 \RequirePackage{soul}
```

```
1089 \RequirePackage{fontawesome}
```

2.12.3 Makro-Kürzel

```

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

```

```
1090 \ExplSyntaxOn
```

```
\liErEntity
```

```
1091 \def\liErEntity#1{\textbf{#1}}
```

```
\liErRelationship
```

```
1092 \def\liErRelationship#1{\ul{#1}}
```

```
\liErAttribute
```

```
1093 \def\liErAttribute#1{\emph{#1}}
```

```
\liErMpEntity
```

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

```

1094 \def\liErMpEntity#1{
1095   \liErEntity{#1}
1096   \marginpar{
1097     \liErEntity{\tiny\faSquareO{~E:~#1}
1098   }
1099 }

```

□

```
\liErMpRelationship
```

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

```

1100 \def\liErMpRelationship#1{
1101   \liErRelationship{#1}
1102   \marginpar{
1103     \liErRelationship{\tiny\faGg{~R:~#1}
1104   }
1105 }

```

```
\liErMpAttribute
```

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

```

1106 \def\liErMpAttribute#1{
1107   \liErAttribute{#1}
1108   \marginpar{
1109     \liErAttribute{\tiny\faCircleThin{~A:~#1}
1110   }
1111 }

```

```

\liErDatenbankName Let-Abkürzung: \let\d=\liErDatenbankName
                    datenbank name
1112 \def\liErDatenbankName#1{
1113   {
1114     \footnotesize\texttt{(#1)}
1115   }
1116 }

1117 \ExplSyntaxOff
1118

```

2.13 formale-sprachen.sty

```

1119 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1120 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
1121 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]

1122 \directlua{
1123   formale_sprachen = require('lehramt-informatik-formale-sprachen')
1124 }

1125 \RequirePackage{hyperref}

1126 \liLadePakete{mathe,typographie}

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

1127 \def\liMengeOhneMathe#1{\{ #1 \}}
1128 \def\liMenge#1{%
1129   \ifmmode%
1130     \liMengeOhneMathe{#1}%
1131   \else%
1132     $\liMengeOhneMathe{#1}$%
1133   \fi%
1134 }

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

1135 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

1136 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1137 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1138 \def\liPotenzmenge#1{$\liPotenzmengeOhneMathe{#1}$}

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

1139 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
1140 \let\liZustandsmenge=\liPotenzmenge

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

1141 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1142 \def\liUeberfuehrungsFunktion#1{
1143   \ifmmode
1144     \liUeberfuehrungsFunktionOhneMathe{#1}
1145   \else
1146     $\liUeberfuehrungsFunktionOhneMathe{#1}$
1147   \fi
1148 }

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

1149 \def\liAlphabet#1{$\Sigma = \{ #1 \}$}

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

1150 \def\liBandAlphabet#1{$\Gamma = \Sigma \cup \{ #1 \}$}

\liZustandsBuchstabe

1151 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1152 \def\liZustandsBuchstabeGross{Z}

```

```

\liZustandsmengeNr
1153 \def\zustandsnamens@liste#1#2{
1154   $
1155   \{
1156     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1157   \}
1158   $
1159 }
1160 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

\liZustandsmengeNrGross
1161 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

\liZustandsname \liZustandsname{1}: $z_1$
1162 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

\liZustandsnameGross \liZustandsnameGross{1}: $Z_1$
1163 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

\liAbleitung \liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1164 \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}

1165 \NewDocumentEnvironment { liProduktionsRegeln }
1166 { 0{P} +b }
1167 {
1168   \liGeschweifteKlammern{#1}
1169   {
1170     \begin{align*}
1171       \directlua{formale_sprachen.produktions_regeln('#2')}
1172     \end{align*}
1173   }{-0.2cm}{-1.5cm}
1174 } {}

\liProduktionen \liProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
1175 \def\liProduktionen#1{
1176   \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
1177 }

\liZustandsnameTiefgestelltt Automatisch tiefgestellte Nummerierung \z1
Let-Abkürzung: \let\z=\liZustandsnameTiefgestelltt
1178 \def\liZustandsnameTiefgestelltt#1{
1179   \ifmmode
1180     \liZustandsBuchstabe\sb{#1}
1181   \else
1182     $\liZustandsBuchstabe\sb{#1}$
1183   \fi
1184 }

1185 \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}

```

```

1186 \NewDocumentCommand{ \liAusdruck } { O{L} m m } {
1187   $
1188   \tl_if_empty:nTF {#1} {} {#1 =}
1189   \{
1190     \, #2 \,
1191     |
1192     \, #3 \,
1193   \}$
1194 }
1195 \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

1196 \def\liFlaci#1{%
1197   \par
1198   {%
1199     \scriptsize
1200     Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1201     Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1202     Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1203     \href{https://flaci.com/#1}{flaci.com/#1}%
1204   }%
1205   \par
1206 }

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

1207 \ExplSyntaxOn
1208 \NewDocumentCommand {\liGrammatik} { O{G} m } {
1209   \tl_set:Nn \l_variablen_tl {V}
1210   \tl_set:Nn \l_alphabet_tl {\Sigma}
1211   \tl_set:Nn \l_produktionen_tl {P}
1212   \tl_set:Nn \l_start_tl {S}
1213
1214   \keys_define:nn { grammatik } {
1215     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1216     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1217     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1218     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1219   }
1220
1221   \keys_set:nn { grammatik } { #2 }
1222
1223   $#1 = (
1224     \l_variablen_tl,
1225     \l_alphabet_tl,
1226     \l_produktionen_tl,
1227     \l_start_tl
1228   )$
1229 }
1230 \ExplSyntaxOff

1231

```

2.14 formatierung.sty

```
1232 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1233 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

2.14.1 Schriftarten / Typographie

```
1234 \RequirePackage{mathpazo}
1235 \RequirePackage[no-math]{fontspec}
1236 \setmainfont{texgyrepagella}
1237 \setsansfont{QTAncientOlive}
1238 \RequirePackage{sectsty}
1239 \allsectionsfont{\fontspec{QTAncientOlive}}
```

2.14.2 Farben

```
1240 \RequirePackage{xcolor}
1241 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

2.14.3 Überschriften

```
1242 \RequirePackage{titlesec}
1243 \titleformat{\chapter}[display]{\sffamily\bfseries}{\Opt}{\LARGE}
1244 \titlespacing{\chapter}{0pt}{0pt}{*1}
1245 \titleformat{\paragraph}[hang]{\sffamily\normalsize\bfseries}{\theparagraph}{1em}{}
1246 \setcounter{secnumdepth}{0}
```

2.14.4 Listen

```
1247 \RequirePackage{paralist}
1248 \renewcommand\labelitemi{-}
1249 \renewcommand\labelitemii{-}
1250 \renewcommand\labelitemiii{-}
1251 \renewcommand\labelitemiv{-}
1252 % Counter: enumi enumii enumiii enumiv
1253 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1254 \renewcommand{\labelenumi}{(\alph{enumi})}
1255 \renewcommand{\labelenumii}{(\roman{enumii})}
```

2.14.5 Kasten

```
1256 \RequirePackage{mdframed}
1257 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1258 \NewDocumentEnvironment { liKasten }{ } {
1259   \begin{mdframed}[backgroundcolor=white!0]
1260 } {
1261   \end{mdframed}
1262 }
```

2.14.6 Header

```
1263 \RequirePackage{fancyhdr}
1264 \fancyhead[L,C,R]{}
1265 \fancyfoot[L]{}
1266 \fancyfoot[C]{}
1267 \fancyfoot[R]{\thepage}
1268 \pagestyle{fancy}
1269 \renewcommand{\headrulewidth}{0pt}
1270 \renewcommand{\footrulewidth}{0pt}
1271
```

2.15 gantt.sty

```

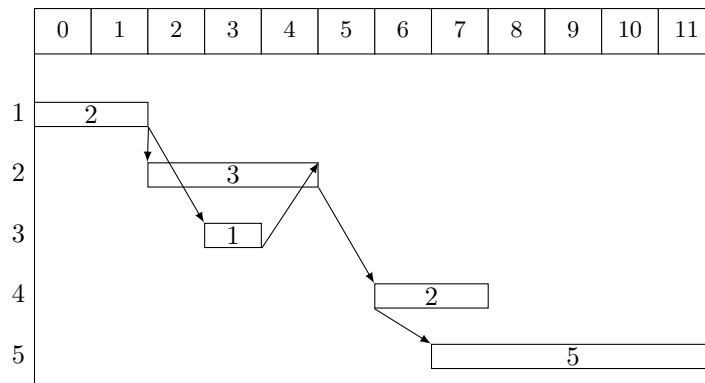
1272 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1273 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\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}

```



```

1274 \RequirePackage{tikz-uml}
1275 \RequirePackage{pgfgantt}
1276 \setganttlinklabel{f-s}{}
1277 \setganttlinklabel{s-s}{}
1278 \setganttlinklabel{f-f}{}
1279 \setganttlinklabel{s-f}{}

1280

```

2.16 grafik.sty

```
1281 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1282 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1283 TikZ, um Grafiken zeichnen zu können.]
1284 \RequirePackage{tikz}
1285
```


2.17 graph.sty

```

1286 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1287 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1288 \RequirePackage{tikz}

```

Für die die Adjazenzliste (`xrightharpoon`)

```

1289 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

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

```

1290 \RequirePackage{blkarray}
1291 \usetikzlibrary{arrows.meta}

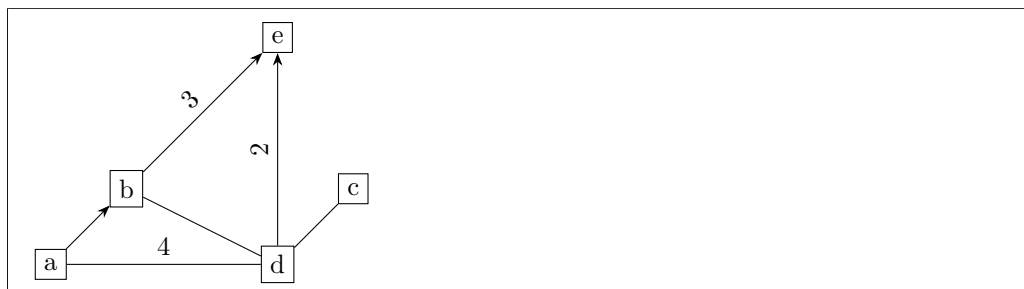
```

```

\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}

```



```

1292 \tikzset{
1293   li graph/.style={
1294     every node/.style={
1295       rectangle,
1296       draw,
1297     },
1298     every edge/.style={
1299       >={Stealth[black]},
1300       draw,
1301     },
1302     every edge/.append style={
1303       every node/.style={
1304         sloped,
1305         auto,
1306       }
1307     }
1308   },
1309   li markierung/.style={
1310     ultra thick,
1311   }
1312 }

```

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}

```

```

1313 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1314

```

2.18 hanoi.sty

1315 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 1316 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
 1317 von Hanoi-Grafiken]

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

1318 \RequirePackage{tikz}
 1319 \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}

1320 \def\li@mset #1[#2]=#3{%
1321 \expandafter\xdef\csname #1#2\endcsname{#3}
1322 }
1323 \def\li@mget #1[#2]{%
1324 \csname #1#2\endcsname
1325 }
1326 \def\li@minc #1[#2] += #3{%
1327 \pgfmathparse{\li@mget #1[#2] + #3}%
1328 \li@mset #1[#2] = \pgfmathresult
1329 }
1330
1331 \def\liHanoi#1#2{
1332   \edef\li@numdiscs{#1}
1333   \def\li@sequence{#2}
1334   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1335     % init colors
1336     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purple!80!black}{
1337       \li@mset col[\j]=\c;
1338     }
1339     % draw poles and init pole counters
1340     \foreach \j in {1,2,3}{
1341       \li@mset pos[\j]=0
1342       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1343     }
1344     % draw base
1345     \draw (.5,-.5) -- +(3,0);
1346     % draw discs
1347     \foreach[count=\k] \i/\j in \li@sequence{
1348       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*\i/\li@numdiscs,0);
1349       \li@minc pos[\j] += {.5}
1350     }
1351   \end{tikzpicture}
1352 }
```

2.19 klassen-konfiguration-aufgabe.sty

```
1353 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1354 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-aufgabe}[2021/09/01 Die
1355 Klasse Aufgabe konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]

    Formatierung muss vor literatur sein, sonst Option clash
1356 \liLadePakete{
1357   formatierung,
1358   abmessung,
1359   literatur-dummy,
1360   makros,
1361   aufgaben-metadaten,
1362   kopf-fusszeilen,
1363   mathe
1364 }
1365
1366 % Komischer Option-Clash deshalb ganz am Ende, für die Silbentrennung
1367 \RequirePackage[ngerman]{babel}
1368 \ExplSyntaxOn
1369 \AddToHook{enddocument}{
1370   \_gib_github_url:
1371 }
1372 \ExplSyntaxOff
1373
```

2.20 klassen-konfiguration-examen.sty

```
1374 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1375 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-examen}[2021/09/04 Die
1376 Klasse liSetzeExamen konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

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

```
1377 \liLadePakete{
1378   formatierung,
1379   literatur-dummy,
1380   makros,
1381   aufgaben-metadaten,
1382   abmessung
1383 }
```

Formatierung für die Überschriften setzen.

```
1384 \RequirePackage{titlesec}
1385 %\titleformat*{\section}{center}
1386 \titleformat{\section}
1387 {\normalfont\Large\filcenter\bfseries}{\thesection}{1em}{}
1388 \newcommand{\sectionbreak}{\clearpage}
1389 \setcounter{secnumdepth}{0}
1390 \liLadeAllePakete
```

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

```
1391 \RequirePackage[ngerman]{babel}
1392 \RequirePackage{standalone}
1393 \ExplSyntaxOn
```

`\liSetzeExamen`

```
1394 \def\liSetzeExamen#1#2#3{
1395   \tl_gset:Nn \g_auf_examen_nummer_tl { #1 }
1396   \tl_gset:Nn \g_auf_examen_jahr_tl { #2 }
1397   \tl_gset:Nn \g_auf_examen_monat_tl { #3 }
1398 }
```

`\liSetzeExamenThemaNr`

```
1399 \def\liSetzeExamenThemaNr#1{
1400   \tl_gset:Nn \g_auf_examen_thema_nr_tl { #1 }
1401   \section{Thema-Nr.~#1}
1402 }
```

`\liSetzeExamenTeilaufgabeNr`

```
1403 \def\liSetzeExamenTeilaufgabeNr#1{
1404   \tl_gset:Nn \g_auf_examen_teilaufgabe_nr_tl { #1 }
1405   \subsection{Teilaufgabe-Nr.~#1}
1406 }
```

`\liBindeAufgabeEin`

```
1407 \def\liBindeAufgabeEin#1{
1408   \tl_gset:Nn \g_auf_examen_aufgabe_nr_tl { #1 }
1409   \input{
1410     \LehramtInformatikRepository /
1411     Staatsexamen /
1412     \g_auf_examen_nummer_tl /
1413     \g_auf_examen_jahr_tl /
1414     \g_auf_examen_monat_tl /
1415     \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
1416       Thema - \g_auf_examen_thema_nr_tl /
1417     }
1418     \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
1419       Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl /
1420     }
1421     Aufgabe - \g_auf_examen_aufgabe_nr_tl .tex
1422   }
1423 }
```

```
\liAufgabenMetadaten  Das Metadaten-Makro überschreiben
1424 \def\liAufgabenMetadaten#1{
1425   \liMetaSetze{#1}
1426   \subsubsection{\_gib_aufgaben_titel:}
1427 }

1428 \ExplSyntaxOff
1429
```

2.21 komplexitaetstheorie.sty

```
1430 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1431 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1432 Setzen von Karp's NP-vollständigen Problemen, Hilfsmakros für die
1433 Polynomialzeitreduktion.]
```

2.21.1 Makro-Kürzel

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

```
1434 \liLadePakete{mathe}
```

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

```
1435 \RequirePackage{mdframed}
```

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

```
1436 \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`

```
1437 \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`

```
1438 \def\liProblemBeschreibung#1#2#3{
1439   \begin{mdframed}[
1440     userdefinedwidth=9cm,
1441     align=center,
1442     backgroundcolor=white!0,
1443   ]
1444     \centerline{\large\liProblemName{#1}}
1445
1446     \medskip
1447
1448     \begin{description}
1449       \item[Gegeben:] #2
1450       \item[Frage:] #3
1451     \end{description}
1452   \end{mdframed}
1453 }
```

```

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

1454 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1455 \begin{displaymath}
1456 \liProblemName{#1}
1457 \preceq_{#2}
1458 \liProblemName{#3}
1459 \end{displaymath}
1460 }

\liProblemVertexCover

1461 \def\liProblemClique{%
1462 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1463 Mindestgröße  $n$  in einem gegebenen Graphen.
1464 \footcite[Seite 76]{theo:fs:4}
1465 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1466 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1467 \footcite{wiki:cliquenproblem}
1468 }

\liProblemVertexCover

1469 \def\liProblemVertexCover{%
1470 %
1471 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1472 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1473 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1474 \footcite{wiki:knotenueberdeckung}
1475
1476 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1477 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1478 mindestens einem Knoten aus  $U$  verbunden ist.
1479 \footcite[Seite 78]{theo:fs:4}%
1480 }

\liProblemSubsetSum Kein Karp-21-Problem

1481 \def\liProblemSubsetSum{%
1482 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1483 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1484 \footcite{wiki:teilsommenproblem}
1485 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1486 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1487 größer als eine gegebene obere Schranke  $c$  ist.
1488 \footcite[Seite 74]{theo:fs:4}
1489 }

\liProblemSubsetSum Kein Karp-21-Problem

1490 \def\liProblemSat{%
1491 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1492 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1493 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1494 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1495 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1496 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1497 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1498 aufgestellt werden.
1499 \footcite[Seite 71]{theo:fs:4}
1500 }

1501

```


2.22 kontrollflussgraph.sty

1502 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1503 \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 = ...}]
```

```

1504 \RequirePackage{tikz}
1505 \usetikzlibrary{positioning}
1506 \tikzset{
1507   li kontrollfluss/.style={
1508     knoten/.style={
1509       circle,
1510       draw
1511     },
1512     usebox/.style={
1513       draw,
1514       rectangle,
1515       font=\scriptsize,
1516       anchor=west,
1517       align=left,
1518     },
1519     bedingung/.style={
1520       midway,
1521       draw=none,
1522       font=\scriptsize
1523     },
1524     knotenbeschriftung/.style={
1525       draw,
1526       rectangle,
1527       midway,
1528       font=\scriptsize
1529     },
1530     wahr/.style={
1531       thick
1532     },
1533     falsch/.style={
1534       dashed
1535     },
1536     every node/.style={
1537       circle,
1538       draw,
1539     },
1540     every edge/.append style={
1541       every node/.style={
1542         draw=none,
1543         bedingung,
1544       }
1545     },
1546     every path/.style={
1547       draw,
1548       ->,
1549     },
1550     every pin/.style={
1551       draw,
1552       dotted,
1553       rectangle,
1554       pin position=right
1555     },
1556     every pin edge/.style={
1557       dotted,
1558       arrows=-,
1559     }
1560   }
1561 }

```

2.22.4 Umgebungen

liKontrollflussgraph

```

1562 \NewDocumentEnvironment { liKontrollflussgraph } { 0{} } {

```

```

1563 \begin{tikzpicture}[
1564     li kontrollfluss,
1565     #1
1566 ]
1567 } {
1568 \end{tikzpicture}
1569 }

```

2.22.5 Makros

`\liAnweisung`

```
1570 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}
```

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

```
1571 \def\liBedingung#1#2{\node[bedingung,#1]{\texttt{#2}}}
```

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

```
1572 \def\liBedingungWahr#1{\node[bedingung,#1]{\texttt{[true]}}}
```

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

```
1573 \def\liBedingungFalsch#1{\node[bedingung,#1]{\texttt{[false]}}}
```

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

```
1574 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}
```

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

```
1575 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw
```

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

```
1576 \ExplSyntaxOn
```

```
1577 \NewDocumentCommand { \liKontrollKnotenPfad } { m }
```

```
1578 {
```

```
1579   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
```

```
1580   \seq_set_map:Nnn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
```

```
1581   \seq_use:Nn \l_tmpa_seq { ~~~~ }
```

```
1582 }
```

```
1583 \ExplSyntaxOff
```

```
1584
```

2.23 kopf-fusszeilen.sty

```
1585 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1586 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1587 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1588 \ExplSyntaxOn

1589 \fancyhead{}
1590 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1591 \fancyhead[L0,RE]{\scriptsize\today}
1592 \fancyfoot{}
1593 \fancyfoot[LE,R0]{\thepage}
1594 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1595 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1596 \renewcommand{\headrulewidth}{0.4pt}
1597 \renewcommand{\footrulewidth}{0.4pt}
1598 \setlength{\headwidth}{\textwidth}

1599 \ExplSyntaxOff

1600
```

2.24 literatur-dummy.sty

```
1601 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1602 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1603 \def\literatur{}

\footcite

1604 % \RequirePackage[stable,multiple]{footmisc}
1605 \NewDocumentCommand{ \footcite } { o m }{}

1606
```

2.25 literatur.sty

```
1607 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1608 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1609 \RequirePackage{csquotes}
1610 \RequirePackage[
1611   bibencoding=utf8,
1612   citestyle=authortitle,
1613   backend=biber,
1614 ]{biblatex}
1615 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1616 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1617 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1618 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1619 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1620 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1621 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1622 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1623 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1624 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1625 % To allow footnotes in the heading
1626 \RequirePackage[stable,multiple]{footmisc}

\literatur

1627 \def\literatur{\printbibliography[heading=subbibliography]}

1628
```

2.26 makros.sty

```

1629 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1630 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1631 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1632 anderen Paket passen]
1633 \RequirePackage{hyperref}
1634 \RequirePackage{graphicx}

Für die Umgebung liQuellen benötigt.
1635 \RequirePackage{paralist}

\inhaltsverzeichnis
1636 \def\inhaltsverzeichnis {
1637   \begin{mdframed}
1638     \begin{group}
1639       \let\clearpage\relax
1640       \tableofcontents
1641     \end{group}
1642   \end{mdframed}
1643 }

\memph \mephm (\marginpar and \emph)
1644 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1645 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1646 \newcommand{\liPseudoUeberschrift}[1]{
1647   \bigskip
1648   \noindent
1649   \textsf{\textbf{#1}}
1650   \noindent
1651 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1652 \newcommand{\liBeschriftung}[1]{
1653   \par
1654   \noindent
1655   \medskip
1656   \textbf{#1}:
1657   \medskip
1658   \noindent
1659 }

\hinweis
1660 \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.
1661 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

liEinbettung
1662 \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.
1663 \RequirePackage{xparse}
1664 \ExplSyntaxOn

```

```

1665 \NewDocumentEnvironment{ liAntwort } { 0{standard} }
1666 {
1667   \str_case:nn {#1} {
1668     {standard} {
1669       \def\beschriftung{}
1670       \mdfsetup{backgroundcolor=gray!1,linecolor=gray}
1671     }
1672     {richtig} {
1673       \def\beschriftung{richtig}
1674       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1675     }
1676     {falsch} {
1677       \def\beschriftung{falsch}
1678       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1679     }
1680     {muster} {
1681       \def\beschriftung{Musterlösung}
1682       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1683     }
1684   }
1685   \ifx\beschriftung\empty\else
1686     \noindent
1687     \textbf{\beschriftung{:}}
1688     \fi
1689     \begin{mdframed}
1690   }
1691 {\end{mdframed}}

```

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

```

1692 \NewDocumentEnvironment{ liAdditum } { o }
1693 {
1694   \begin{mdframed}[backgroundcolor=yellow!5]
1695     \IfNoValueTF {#1}
1696     { \liPseudoUeberschrift{Additum} }
1697     { \liPseudoUeberschrift{Additum:~#1} }
1698   }
1699 {\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.

```

1700 \NewDocumentEnvironment{ liExkurs }{o +b}{
1701   \vspace{0.2cm}%
1702   \begin{mdframed}[
1703     backgroundcolor=white,
1704     bottomline=false,
1705     innermargin=1cm,
1706     leftline=true,
1707     linecolor=black,
1708     linewidth=0.1cm,
1709     outermargin=1cm,
1710     rightline=false,
1711     topline=false,
1712   ]

```



```

1713 \footnotesize
1714 \noindent%
1715 \textbf{Exkurs:~#1}\par%
1716 \noindent%
1717 #2
1718 \end{mdframed}
1719 \vspace{0.2cm}
1720 }{}

```

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

```

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

```

Weiterführende Literatur:

- Quelle 1
- Quelle 2

```

1721 \cs_new:Npn \listen@punkt #1 {\item #1}
1722 \NewDocumentEnvironment { liQuellen }{ +b }
1723 {
1724   \seq_clear_new:N \l_quellen
1725   \seq_set_split:Nnn \l_quellen {\item} {#1}
1726   \seq_remove_all:Nn \l_quellen {}
1727   \begin{mdframed}[backgroundcolor=white!0]
1728     \footnotesize
1729     \noindent
1730     \textsf{\textbf{Weiterführende~Literatur:}}
1731     \medskip
1732     \begin{compactitem}
1733       \seq_map_function:NN \l_quellen {\listen@punkt}
1734     \end{compactitem}
1735   \end{mdframed}
1736   %
1737   \makeatletter
1738   \par\@afterindentfalse\@afterheading
1739   \makeatother
1740 } {}

```

liLernkartei

```

1741 \NewDocumentEnvironment { liLernkartei }{ m +b }
1742 {
1743   \begin{mdframed}
1744     \footnotesize
1745     \noindent%
1746     \textbf{Lernkarteikarte:~#1}\par%
1747     \noindent%
1748     #2
1749   \end{mdframed}
1750 } {}

```

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

```

1751 \NewDocumentEnvironment { liDiagramm }{ m +b }
1752 {
1753   \begin{mdframed}[backgroundcolor=white!0]
1754     \small
1755     \noindent%
1756     \textit{#1}:
1757     \begin{center}

```

```

1758 #2
1759 \medskip
1760 \end{center}
1761 \end{mdframed}
1762 } {}
1763 \ExplSyntaxOff

\liFussnoteUrl \liFussnoteUrl[\langle zusätzlicher-text \rangle]{\langle url \rangle} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
1764 \NewDocumentCommand{\liFussnoteUrl} { o m } {
1765 \footnote{\url{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1766 }
1767

\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.
1768 \NewDocumentCommand{\liFussnoteLink} { o m m } {
1769 \footnote{\href{#3}{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1770 }

\zB
1771 \def\zB{z.\,B. }

\ZB
1772 \def\ZB{Z.\,B. }

\dh
1773 \def\dh{d.\,h. }
1774

```

2.27 master-theorem.sty

1775 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1776 \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}

1777 \ExplSyntaxOn

1778 \RequirePackage{amsmath}

\liRundeKlammer

1779 \def\liRundeKlammer#1{

1780 \negthinspace \left(#1 \right)

1781 }

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

1782 \def\liThetaOhneMathe#1{

1783 \Theta \liRundeKlammer{#1}

1784 }

1785 \def\liTheta#1{

1786 \ifmmode

1787 \liThetaOhneMathe{#1}

1788 \else

1789 \$\liThetaOhneMathe{#1}\$

1790 \fi

1791 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
1792 \def\liOmegaOhneMathe#1{
1793 \Omega \liRundeKlammer{#1}
1794 }
1795 \def\liOmega#1{
1796 \ifmmode
1797 \liOmegaOhneMathe{#1}
1798 \else
1799 $\liOmegaOhneMathe{#1}$
1800 \fi
1801 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
1802 \def\liOOhneMathe#1{
1803 \mathcal{O} \liRundeKlammer{#1}
1804 }
1805 \def\liO#1{
1806 \ifmmode
1807 \liOOhneMathe{#1}
1808 \else
1809 $\liOOhneMathe{#1}$
1810 \fi
1811 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
1812 \def\liTOhneMathe#1#2{
1813 \tl_if_blank:nTF {#1}
1814 {}
1815 {#1 \cdot }
1816 T
1817 \liRundeKlammer{\frac{n}{#2}}
1818 }
1819 \def\liT#1#2{
1820 \ifmmode
1821 \liTOhneMathe{#1}{#2}
1822 \else
1823 $\liTOhneMathe{#1}{#2}$
1824 \fi
1825 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
1826 \def\liRekursionsGleichung{
1827 $T(n) = \liT{a}{b} + f(n)$
1828 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
1829 \def\liBedingungEins{
1830 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
1831 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
1832 \def\liBedingungZwei{
1833 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
1834 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
1835 \def\liBedingungDrei{
1836 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
1837 }

1838 \ExplSyntaxOff

```

\liMasterVariablen

```

1839 \def\liMasterVariablen{
1840   \begin{displaymath}
1841     T(n) = \liT{a}{b} + f(n)
1842   \end{displaymath}
1843
1844   \begin{itemize}
1845     \item[$a = $]
1846       Anzahl der rekursiven Aufrufe, Anzahl der Unterprobleme in der
1847       Rekursion
1848       ($a \geq 1$).
1849
1850     \item[$\textstyle{\frac{1}{b}} = $]
1851       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
1852       repräsentiert wird, Anteil an der Verkleinerung des Problems ($b > 1$).
1853
1854     \item[$f(n) = $]
1855       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
1856       die Kombination der Teillösungen entstehen. Eine von $T(n)$
1857       unabhängige und nicht negative Funktion.
1858   \end{itemize}
1859   \footcite{wiki:master-theorem}
1860   \footcite[Seite 19-35]{aud:fs:2}
1861 }
```

\liMasterFaelle

```

1862 \def\liMasterFaelle{
1863   \begin{description}
1864     \item[1. Fall:]
1865       $T(n) \in \liTheta{n^{\log\sb{b}a}}$
1866
1867       \hfill falls \liBedingungEins
1868       für $\varepsilon > 0$
1869
1870     \item[2. Fall:]
1871       $T(n) \in \liTheta{n^{\log\sb{b}a} \cdot \log n}$
1872
1873       \hfill falls \liBedingungZwei
1874
1875     \item[3. Fall:]
1876       $T(n) \in \liTheta{f(n)}$
1877
1878       \hfill falls \liBedingungDrei
1879       für $\varepsilon > 0$
1880       und ebenfalls für ein $c$ mit $0 < c < 1$ und alle hinreichend großen $n$
1881       gilt:
1882       $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$
1883   \end{description}
1884 }
```

\liMasterVariablenDeklaration

```

1885 \def\liMasterVariablenDeklaration#1#2#3{
1886   \begin{description}
1887     \item[Allgemeine Rekursionsgleichung:] \strut
1888
1889     \liRekursionsGleichung
1890
1891     \item[Anzahl der rekursiven Aufrufe ($a$):] \strut
1892
1893     #1
1894
1895     \item[Anteil Verkleinerung des Problems ($b$):] \strut
1896
1897     #2
1898
1899     #3
1900   \end{description}
1901 }
```

```

1897     um  $\frac{1}{b}$  also  $b = \frac{1}{a}$ 
1898
1899     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut
1900
1901      $\frac{1}{b}$ 
1902
1903     \item[Ergibt folgende Rekursionsgleichung:] \strut
1904
1905      $T(n) = T(\frac{n}{b}) + \frac{1}{b}$ 
1906 \end{description}
1907 }

\liMasterFallRechnung
1908 \def\liMasterFallRechnung#1#2#3{
1909     \begin{description}
1910     \item[1. Fall:] \liBedingungEins:
1911
1912     #1
1913
1914     \item[2. Fall:] \liBedingungZwei:
1915
1916     #2
1917
1918     \item[3. Fall:] \liBedingungDrei:
1919
1920     #3
1921     \end{description}
1922 }

\liMasterExkurs
1923 \def\liMasterExkurs{
1924     \begin{liExkurs}[Master-Theorem]
1925     \liMasterVariablen
1926
1927     \noindent
1928     Dann gilt:
1929
1930     \liMasterFaelle
1931     \end{liExkurs}
1932 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
1933 \def\liMasterWolframLink#1{
1934     Berechne die Rekursionsgleichung auf WolframAlpha:
1935     \href{https://www.wolframalpha.com/input/?i=#1}{WolframAlpha}
1936 }

1937

```

2.28 mathe.sty

```
1938 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1939 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
1940
1941 % for example \ltimes \rtimes
1942 %\RequirePackage{amssymb}
1943 \RequirePackage{amsmath}
1944
1945 %%
1946 % \mlq \mrq
1947 %%
1948 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
1949 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
1950
```

2.29 minimierung.sty

```

1951 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1952 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
1953 Minimierungsalgorithmus von einem NEA zu einem DEA]

1954 \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

1955 \def\liFussnote#1{\$x_{#1}\$}

1956 \def\li@fussnote@text#1#2{
1957 \liFussnote{#1}
1958 \quad
1959 {\footnotesize #2}
1960 }

\liFussnoteEinsText

1961 \def\liFussnoteEinsText{
1962 \li@fussnote@text{1}
1963 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
1964 }

\liFussnoteZweiText

1965 \def\liFussnoteZweiText{
1966 \li@fussnote@text{2}
1967 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
1968 }

\liFussnoteDreiText

1969 \def\liFussnoteDreiText{
1970 \li@fussnote@text{3}

```



```

1971 {In weiteren Iterationen markierte Zustände.}
1972 }

\liFussnoteVierText
1973 \def\liFussnoteVierText{
1974   \li@fussnote@text{4}
1975   {...}
1976 }

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



1977 \def\liFussnoten{
1978   \bigskip
1979
1980   \noindent
1981   \liFussnoteEinsText
1982
1983   \noindent
1984   \liFussnoteZweiText
1985
1986   \noindent
1987   \liFussnoteDreiText
1988
1989   \noindent
1990   \liFussnoteVierText
1991 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
1992 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
1993 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
1994 \def\liZustandsPaar#1#2{
1995   $(
1996     \liZustandsPaarVariablenName_#1,
1997     \liZustandsPaarVariablenName_#2
1998   )$
1999 }

liUebergangsTabelle
2000 \renewcommand{\arraystretch}{1.4}
2001 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
2002   \liPseudoUeberschrift{Übergangstabelle}
2003   \begin{center}
2004     \begin{tabular}{r|l|l}
2005       \textbf{Zustandspaar} & \textbf{\#1} & \textbf{\#2} \\ \hline
2006     \end{tabular}
2007   \end{center}
2008 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle: Minimierungstabelle (Table filling)

2010 \ExplSyntaxOn
2011 \def\liUeberschriftDreiecksTabelle{
2012   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
2013 }

```

\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. —

```

2014 \def\liMinimierungErklaerung{
2015   %\footcite[Seite~19]{koenig}
2016   \liParagraphMitLinien{
2017     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
2018     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
2019     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
2020     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
2021      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
2022     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
2023     somit~zueinander~( $k - 1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
2024     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
2025     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
2026   }
2027 }
2028 \ExplSyntaxOff
2029

```

2.30 normalformen.sty

```
2030 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2031 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10]
2032 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
2033 Attributhülle]
```

Mathe für textit tex etc, typographie für geschweifte Klammer

```
2034 \liLadePakete{mathe,typographie}
2035 \directlua{
2036   helfer = require('lehramt-informatik-helfer')
2037   normalformen = require('lehramt-informatik-normalformen')
2038 }
```

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
```

```
2039 \def\liTeilen#1{
2040   \directlua{tex.print(normalformen.teilen('#1'))}
2041 }
```

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

```
2042 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(\#1)}
2043 \def\liAttributHuelle#1{
2044   \ifmmode
2045     \liAttributHuelleOhneMathe{\#1}
2046   \else
2047     $\liAttributHuelleOhneMathe{\#1}$
2048   \fi
2049 }
```

\liAttributMenge Let-Abkürzung: $\text{let}\ \text{m}=\text{liAttributMenge}$
2050 $\text{def}\ \text{liAttributMenge}\#1\{\backslash\{\ \text{textit}\{\#1\}\ \backslash\}$

liAHuelle

```
2051 \NewDocumentEnvironment{ liAHuelle } { +b } {
2052   \begingroup
2053   \footnotesize
2054   \begin{multline*}
2055     \#1
2056   \end{multline*}
2057   \endgroup
2058 } { }
```

\liLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multline

Let-Abkürzung: $\text{let}\ \text{ahL}=\text{liLinksReduktion}$
 $\text{ahL}\{\text{ursprüngliche linke Attributmenge}\{\text{ohne dieses Attribut}\}\{\text{Ergebnis}\}$
2059 $\text{def}\ \text{liLinksReduktion}\#1\#2\#3\{$
2060 $\text{shoveleft}\{$
2061 $\text{liAttributHuelleOhneMathe}\{\text{FA},$
2062 $\text{liAttributMenge}\{\#1\ \text{string}\ \#2\}\} =$
2063 $\}\ \backslash\backslash$

```

2064 \shoveright{
2065   \liAttributMenge{#3}
2066 } \\\
2067 }

```

```

\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( $F, \{A, E \setminus E\}$ ) =  $\{A, E, F, B, D\}$ 
2068 \def\liLinksReduktionInline#1#2#3{%
2069   {%
2070     \footnotesize%
2071     $\liAttributHuelleOhneMathe{F,
2072       \liAttributMenge{#1} \string\ #2}} =
2073     \liAttributMenge{#3}$
2074   }
2075 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahr=\liLinksReduktionInline
    \ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
2076 \def\liRechtsReduktionInline#1#2#3#4{%
2077   {%
2078     \footnotesize%
2079     $\liAttributHuelleOhneMathe{
2080       F \setminus
2081       \liFunktionaleAbhaengigkeit{#1}
2082       \def\tmp{#2}\ifx\tmp\empty
2083       \else
2084         \cup \liFunktionaleAbhaengigkeit{#2}
2085       \fi
2086       ,
2087       \liAttributMenge{#3}
2088     } =
2089     \liAttributMenge{#4}$
2090   }
2091 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
    \fa{A, B -> C, D}:  $\{A, B\} \rightarrow \{C, D\}$  Regulärer Ausdruck zum Konvertieren:
    \$(.*?) \\\rightarrow (.*)\$ \fa{$1 -> $2}
2092 \def\liFunktionaleAbhaengigkeit#1{%
2093   \directlua{normalformen.drucke_funk_abhaengigkeit('#1', false)}%
2094 }

```

```

\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;

```

```

2095 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
2096   \liGeschweifteKlammern
2097   {#1}
2098   {
2099     \begin{align*}
2100       \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
2101     \end{align*}
2102   }
2103   {-0.5cm}
2104   {-1.7cm}
2105 }

\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}

2106 \NewDocumentCommand {\liRelation} { O{R} m } {
2107   $\directlua{
2108     local name = helper.konvertiere_tiefgestellt('#1')
2109     tex.print(name)
2110   }$(\textit{\, #2\,})
2111 }

2112

```

2.31 petri.sty

2113 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2114 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]

2.31.1 Makro-Kürzel

\let\t=\liPetriTransitionsName

\let\tp=\liPetriTransPfeile

\let\k=\liPetriErreichKnotenDrei

2115 \RequirePackage{tikz}

2116 \usetikzlibrary{petri,arrows.meta}

Für die Darstellungsmatrix

2117 \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

```
2118 \def\liPetriSetzeSchluessel{%
2119   \def\TmpTransitionOne{}%
2120   \def\TmpTransitionTwo{}%
2121   \def\TmpTransitionThree{}%
2122   \def\TmpTransitionFour{}%
2123   \def\TmpTransitionFive{}%
2124   \def\TmpTransitionSix{}%
2125   \def\TmpTransitionSeven{}%
2126   \def\TmpTransitionEight{}%
2127   \def\TmpTransitionNine{}%
2128   \def\TmpTransitionTen{}%
2129   \pgfkeys{/petri/.cd,
2130     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2131     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2132     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2133     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
```

```

2134 p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
2135 p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2136 p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2137 p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2138 p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2139 p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2140 t1/.store in=\TmpTransitionOne,t1/.default=activated,
2141 t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2142 t3/.store in=\TmpTransitionThree,t3/.default=activated,
2143 t4/.store in=\TmpTransitionFour,t4/.default=activated,
2144 t5/.store in=\TmpTransitionFive,t5/.default=activated,
2145 t6/.store in=\TmpTransitionSix,t6/.default=activated,
2146 t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2147 t8/.store in=\TmpTransitionEight,t8/.default=activated,
2148 t9/.store in=\TmpTransitionNine,t9/.default=activated,
2149 t10/.store in=\TmpTransitionTen,t10/.default=activated,
2150 scale/.store in=\TmpScale,scale/.default=0.5,
2151 x/.store in=\TmpX,x/.default=5,
2152 y/.store in=\TmpY,y/.default=5,
2153 }%
2154 }

```

```

2155 \tikzset{
2156   li petri/.style={
2157     activated/.style={
2158       very thick
2159     },
2160     inhibitor/.style={
2161       {Circle[open,length=2mm,fill=white]}-
2162     }
2163   }
2164 }

```

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

```

2165 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2166 \def\liPetriTransitionsName#1{
2167   \ifmmode
2168     \liPetriTransitionsNameOhneMathe{#1}
2169   \else
2170     $\liPetriTransitionsNameOhneMathe{#1}$
2171   \fi
2172 }

```

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

```

2173 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2174   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2175 }

```

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

```

2176 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2177 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2178 }

```

2.32 potenzmengen-konstruktion.sty

```
2179 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2180 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2181 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2182 \liLadePakete{formale-sprachen}
2183 \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}
    }
  }
}
```

```
2184 \def\liZustandsMengenSammlung#1#2{
2185   \liZustandsnameGross{#1}
2186   {
2187     \footnotesize
2188     \liPotenzmenge{
2189       \str_case:nn {#1} {#2
2190       }
2191     }
2192 }
```

```
\liZustandsMengenSammlungNr
```

```
2193 \def\liZustandsMengenSammlungNr#1#2{
2194   \liZustandsnameGross{#1}
2195   {
```



```
2196     \footnotesize
2197     \liZustandsmengeNr{
2198         \str_case:nn {#1} #2
2199     }
2200 }
2201 }

2202 \ExplSyntaxOff
2203
```

2.33 pseudo.sty

2204 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2205 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen
 2206 von Pseudo-Code, Hüll-Paket um algorithm2e]

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

Algorithmus 1: Minimaler Spannbaum nach Kruskal

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

2207 \RequirePackage[german,boxruled]{algorithm2e}

2208

2.34 pumping-lemma.sty

2209 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2210 \ProvidesPackage{lehramt-informatik-pumping-lemma}[2021/08/11 Enthält die
 2211 Definitionen für das Pumping-Lemma in der Regulären Sprache und
 2212 in der Kontextfreien Sprache]

\liPumpingRegulaer

```

2213 \def\liPumpingRegulaer{%
2214   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2215   alle Wörter  $\omega \in L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2216    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2217   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2218
2219   \begin{enumerate}
2220     \item  $|v| \geq 1$ 
2221     (Das Wort  $v$  ist nicht leer.)
2222
2223     \item  $|uv| \leq j$ 
2224     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2225
2226     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w \in L$ 
2227     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2228     Sprache  $L$ )
2229   \end{enumerate}
2230
2231   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2232   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2233 }
```

\liPumpingKontextfrei

```

2234 \def\liPumpingKontextfrei{%
2235   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2236   sich alle Wörter  $\omega \in L$  mit  $|\omega| \geq j$  zerlegen lassen in
2237    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2238
2239   \begin{enumerate}
2240     \item  $|vx| \geq 1$ 
2241     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2242
2243     \item  $|vwx| \leq j$ 
2244     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2245
2246     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy \in L$  (Für jede
2247     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2248     Sprache  $L$ )
2249   \end{enumerate}
2250 }
```

2251

2.35 quicksort.sty

```

2252 % https://tex.stackexchange.com/a/142634
2253 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2254 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2255
2256 %-----
2257 % USAGE:
2258 % \QSinitialize{comma, separated, numerical, values}
2259 % \loop
2260 % \QSpivotStep
2261 % \ifnum\value{pivotcount}>0
2262 %   \QSSortStep
2263 % \repeat
2264 %-----
2265
2266 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2267 \RequirePackage{xintfrac, xinttools}
2268
2269 \RequirePackage{tikz}
2270
2271 %-----
2272 % FIRST PART: TikZ styles and macros for the actual drawing
2273 \newcounter{cellcount}% used for coordinates of the node
2274 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2275
2276 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2277 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2278 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2279 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2280 % by police of LaTeX good conduct ? )
2281 \tikzset{ll/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2282          oo/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2283          rr/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2284 % this is the "b" style as used in the image below
2285          bb/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2286 % nicer:
2287          bb/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2288          gg/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2289
2290 % NOTE the b style was originally the same as the r(aised) style apart from
2291 % not being raised, but I find it nicer with a somewhat different
2292 % specification. I have not updated the images though.
2293
2294 % How the nodes are drawn depending on whether on the left of the pivot value
2295 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2296
2297 \def\DecoLEFT #1{%
2298   \xintFor* ##1 in {#1} \do
2299     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2300 }
2301
2302 \def\DecoINERT #1{%
2303   \xintFor* ##1 in {#1} \do
2304     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2305 }
2306
2307 \def\DecoRIGHT #1{%
2308   \xintFor* ##1 in {#1} \do
2309     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2310 }
2311
2312 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2313   \xintFor* ##1 in {#1} \do

```

```

2314     {\stepcounter{cellcount}}%
2315     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2316 }
2317
2318 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2319     \xintFor* ##1 in {#1} \do
2320     {\stepcounter{cellcount}}%
2321     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2322 }
2323
2324 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2325     \xintFor* ##1 in {#1} \do
2326     {\stepcounter{cellcount}}%
2327     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2328 }
2329
2330 %-----
2331 % SECOND PART: the actual sorting routines.
2332
2333 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2334 \def\QS@sort@b #1{\ifcase #1
2335     \expandafter\QS@sort@empty
2336     \or\expandafter\QS@sort@single
2337     \else\expandafter\QS@sort@c
2338     \fi
2339 }%
2340 \def\QS@sort@empty #1{}
2341 \def\QS@sort@single #1{\QS@Ir {#1}}
2342
2343 % This step is to pick the last as pivot.
2344 \def\QS@sort@c #1%
2345     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2346
2347 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2348 % The trick with \xintApplyUnbraced is that for example when selecting
2349 % the elements smaller than pivot, if we had been using \xintApply we
2350 % would have had at the minimum an empty brace pair. Thus we use the
2351 % "unbraced" variant, but then the \QS@select@smaller has added in
2352 % anticipation a level of braces.
2353 \def\QS@sort@d #1#2{%
2354     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2355     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2356     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2357 }%
2358 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2359 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2360 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2361
2362 %
2363 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2364 %
2365 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2366 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2367 % latter must handle correctly an empty argument.
2368
2369 %-----
2370 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QS@initialize.
2371
2372 % This draws all with suitable highlighting for the newly chosen pivots
2373 % (which will be shown raised)

```

```

2374 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2375             \let\QSIr\DecoINERT
2376             \let\QSIrr\DecoINERT
2377             \let\QSRr\DecoRIGHTwithPivot
2378 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2379             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2380             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2381 }
2382
2383 % This sorts and then draws, showing where the pivot chosen in the previous
2384 % step go. Next time they will have become "inert". If pivotcount is still at
2385 % zero on exit from \QSpivotStep, then this is the signal to stop before
2386 % executing \QSsortStep.
2387 \def\QSsortStep {\def\QSLr {\noexpand\QS@sort@a}%
2388             \def\QSRr {\noexpand\QS@sort@a}%
2389             \def\QSIr {\noexpand\QSIrr}%
2390             \let\QSIrr\relax
2391             \edef\QS@list{\QS@list}%
2392             \let\QSLr\relax
2393             \let\QSRr\relax
2394             \let\QSIr\relax
2395             \edef\QS@list{\QS@list}%
2396             \let\QSLr\DecoLEFT
2397             \let\QSIr\DecoINERTwithPivot
2398             \let\QSIrr\DecoINERT
2399             \let\QSRr\DecoRIGHT
2400 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2401             \setcounter{cellcount}{0}%
2402             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2403 }
2404
2405 \def\QSinitialize #1{%
2406     % first, we convert the comma separated values into a list of braced items
2407     % we use an \edef, and anyhow many \edef's will be used later
2408     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2409     \let\QSRr\DecoRIGHT
2410     % The \QSRr marker mutated to draw the last element as
2411     % pivot and the earlier ones with the suitable style.
2412     %
2413     % The list of marked braced items \QS@list is used both for drawing
2414     % (as here) and for doing the exchange of elements during sort.
2415     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2416             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2417 }
2418

```

2.36 relationale-algebra.sty

```

2419 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2420 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]
2421 \RequirePackage{amsmath}
2422 \RequirePackage{amssymb}

```

```

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

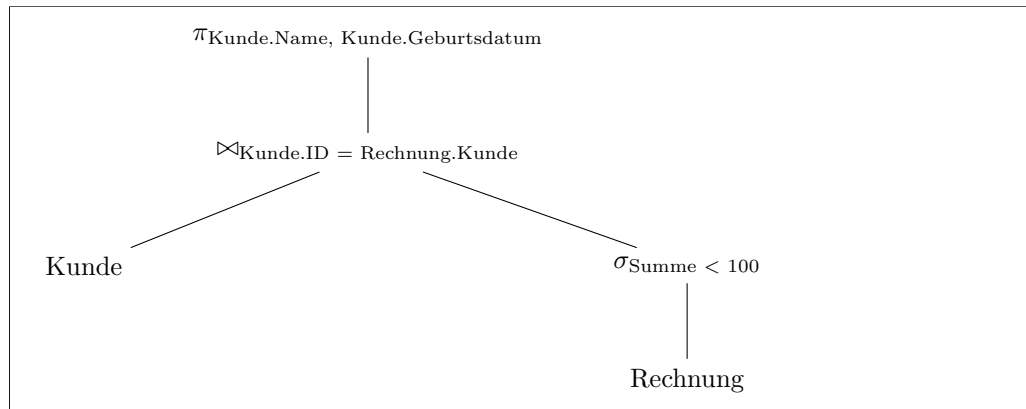
  \node[below=of pi]
    (theta join) {\bowtie_{\text{Kunde.ID = Rechnung.Kunde}}};
    edge (pi);

  \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}

```



```

2423 \RequirePackage{tikz}
2424 \usetikzlibrary{positioning}

    Privates Makros, das zwei Querstriche erzeugt.
2425 \def\o@join{\setbox0=\hbox{\bowtie}}%
2426   \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2427 }

\leftouterjoin A \leftouterjoin B: A ⋈ B
2428 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}

\rightouterjoin A \rightouterjoin B: A ⋉ B
2429 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}

\fullouterjoin A \fullouterjoin B: A ⋈⋉ B
2430 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
2431

```

2.37 rmodell.sty

```
2432 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2433 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2434 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2435 Datenbanken.]
2436 \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

```
2437 \def\liPrimaer#1{\ul{#1}}
```

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

```
2438 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

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

```
2439 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2440 \ExplSyntaxOn
2441 \NewDocumentEnvironment { liRmodell }
2442 { +b }
2443 {
2444   \medskip
2445   {
2446     \linespread{2}
2447     \setlength{\parindent}{0pt}
2448     \li@Rmodell@Schrift#1
2449   }
2450   \medskip
2451 } {}
2452 \ExplSyntaxOff
```

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

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

```
2453 \def\liRelationMenge#1#2{
2454   \noindent
2455   #1 : \{ [ #2 ] \}
2456   \par
2457 }
```

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

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

```
2458 \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}
```

```
2459 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2460
```


2.38 sortieren.sty

2461 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2462 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
 2463 für händisches Sortieren bzw. einen Schreibtischlauf setzen]

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



2464 \RequirePackage{tikz}
 2465 \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.

```
2466 \def\liVertauschen#1{
2467   \directlua{
2468     local sortieren = require('lehramt-informatik-sortieren')
2469     sortieren('#1')
2470   }
2471 }
```

\liSortierPfeil

```
2472 \def\liSortierPfeil#1#2{
2473   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2474 }
```

\liSortierPfeilUnten

```
2475 \def\liSortierPfeilUnten#1#2{
2476   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2477 }
```

\liSortierMarkierung

```
2478 \def\liSortierMarkierung#1#2{\node[
2479   draw,
2480   very thick,
2481   fit=(reihe.#1) (reihe.#2),
2482   inner sep=0pt
2483 ] {}
2484 }

2485 \tikzset{
2486   li sortierung zahlenreihe/.style={
2487     draw,
2488     thin,
2489     font=\large,
2490     rectangle split horizontal,
2491     rectangle split,
2492   }
2493 }
```

```

2494 % https://tex.stackexchange.com/a/140895
2495 \RequirePackage{forest,xstring}
2496 \usetikzlibrary{calc}
2497
2498 \makeatletter
2499 \pgfmathdeclarefunction{strrepeat}{2}{%
2500   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2501   \let\pgfmathresult\pgfutil@empty
2502   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2503     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2504     \advance\pgfmath@count-1\relax
2505   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2506 \makeatother
2507
2508 \def\myNodes{}
2509
2510 \ExplSyntaxOn
2511 \newcommand*\sortList[1]{%
2512   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2513 \ExplSyntaxOff
2514
2515 \forestset{
2516   sort/.code={%
2517     \pgfmathparse{level()}>\forestSortLevel}%
2518     \ifnum\pgfmathresult=0
2519       \StrSubstitute{\forestov{content}}{ }{,}{\myList}%
2520       \sortList\myList
2521       \StrSubstitute{\myList}{,}{ }{\myList}%
2522       \pgfmathparse{strrepeat("1",level())}%
2523       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-!({\forestov{name}}$)
2524         (m\forestov{name}) {\myList}}%
2525       \pgfmathparse{level()}==\forestSortLevel}%
2526       \ifnum\pgfmathresult=1
2527         \forestOget{\forestov{@first}}{name}\forestFirst
2528         \forestOget{\forestov{@last}}{name}\forestLast
2529         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2530           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2531       \fi
2532       \ifnum\forestov{@parent}=0\else
2533         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2534       \fi
2535       \gappto\myNodes{;}%
2536     \fi}}
2537
2538 \forestset{sort level/.code=%
2539   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2540   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2541

```

2.39 spalten.sty

```
2542 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2543 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2544 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2545 realisiert werden kann.]
2546 \RequirePackage{multicol}
```

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

```
2547 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2548
```

2.40 sql.sty

```
2549 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2550 \ProvidesPackage{lehramt-informatik-tabelle}[2021/09/02 Zu Setzen von SQL]

2551 \liLadePakete{syntax}

2552 \RequirePackage{fancyvrb}
2553 \DefineVerbatimEnvironment{liSqlErgebnis}{Verbatim}
2554 {fontsize=\footnotesize}

2555
```

2.41 struktogramm.sty

```
2556 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2557 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2558 Paket struktex zum Setzen von Struktogrammen]
2559 \RequirePackage{struktex}
2560
```

2.42 syntax.sty

```
2561 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2562 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2563 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2564 \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
```

```
2565 \ExplSyntaxOn
2566 \directlua{
2567   syntax = require('lehramt-informatik-syntax')
2568   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2569   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2570   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2571   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2572   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2573   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2574 }
2575 \RequirePackage{hyperref}
2576 \RequirePackage{minted}
2577 % pygmentize -L styles
2578 \usemintedstyle{colorful}
2579 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2580 %\AfterEndEnvironment{minted}{\end{mdframed}}
2581 %\setminted{breaklines=true,linenos}
2582 \setminted{
2583   breaklines=true,
2584   linenos,
2585   fontsize=\footnotesize,
2586 }
```

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

Let-Abkürzung: `\let\j=\liJavaCode`

```
2587 \def\liJavaCode#1{\,\mintinline{java}|#1|\,}
```

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

```
2588 \def\liLatexCode#1{\mintinline{latex}|#1|}
```

```
2589 \def\li@GithubLink#1#2{
2590   \begin{flushright}
2591     \tiny
2592     Code-Beispiel~auf~Github~ansehen:~
2593     \href{#1}{\nolinkurl{#2}}
2594   \end{flushright}
2595 }
```

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

```
2596 \NewDocumentCommand{\liJavaDatei}{0{firstline=3} m }{
2597   \inputminted[#1]{java}{
2598     \directlua{
2599       syntax.drucke_absoluten_pfad('#2', false)
2600     }
2601   }
2602   \li@GithubLink
```

```

2603     {\directlua{syntax.drucke_github_url('#2', false)}}
2604     {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2605 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2606 \NewDocumentCommand{\liJavaTestDatei}{ 0{firstline=3} m }{
2607   \inputminted[#1]{java}{
2608     \directlua{
2609       syntax.drucke_absoluten_pfad('#2', true)
2610     }
2611   }
2612   \li@GithubLink
2613   {\directlua{syntax.drucke_github_url('#2', true)}}
2614   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2615 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei(\[.
\liJavaExamen$1{$2}{$3}{$4}{$5}
2616 \NewDocumentCommand{\liJavaExamen}{ 0{firstline=3} m m m m }{
2617   \inputminted[#1]{java}{
2618     \directlua{
2619       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2620     }
2621   }
2622
2623   \li@GithubLink
2624   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2625   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2626 }

\liAssemblerCode
2627 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2628 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2629   \inputminted{asm}{#1}
2630 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2631 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2632   \inputminted{componentpascal}{#1}
2633 }

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2634 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2635 \NewDocumentCommand{\liHaskellDatei}{ m }{
2636   \inputminted{haskell}{#1}
2637 }

2638 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2639 \def\liSqlCode#1{\mintinline{sql}|#1|}

2640

```

2.43 syntaxbaum.sty

```
2641 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2642 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2643 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
2644 \RequirePackage{tikz-qtrees}
2645
2646 \tikzset{li parsetree/.style={
2647     every internal node/.style={
2648         draw,circle
2649     },
2650     every leaf node/.style={
2651         draw,rectangle
2652     },
2653 }
2654 }
2655
```


2.44 synthese-algorithmus.sty

```
2656 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2657 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2658 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2659 Relation in die 3. Normalform]

2660 \liLadePakete{normalformen,mathe,typographie}
2661 \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}

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

\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

```
2662 \def\liSyntheseUeberschrift#1{
2663   {
2664     \bfseries
2665     \sffamily
2666     \str_case:nn {#1} {
2667       {1} {Kanonische-Überdeckung}
2668       {1-1} {Linksreduktion}
2669       {1-2} {Rechtsreduktion}
2670       {1-3} {Löschen-leerer-Klauseln}
2671       {1-4} {Vereinigung}
2672       {2} {Relationsschemata-formen}
2673       {3} {Schlüssel-hinzufügen}
2674       {4} {Entfernung-überflüssiger-Teilschemata}
2675     }
```

```

2676   }
2677 }

\liSyntheseErklaerung Let-Abkürzung: \let\erklaerung=\liSyntheseErklaerung
2678 \def\li@synthese@erklaerung@texte#1{
2679   \str_case:nn {#1} {
2680     {1} {
2681       Die~kanonische~Überdeckung~---also~die~kleinst~mögliche~noch~
2682       äquivalente~Menge~von~funktionalen~Abhängigkeiten~kann~in~vier~
2683       Schritten~erreicht~werden.
2684     }
2685     {1-1} {
2686       Führe~für~jede~funktionale~Anhängigkeit~
2687        $\alpha \rightarrow \beta$ ~die~Linksreduktion~durch,~
2688       überprüfe~also~für~alle~
2689        $A \in \alpha$ ,~ob~ $A$ ~überflüssig~ist,~d.h.~ob~
2690        $\beta \subseteqq \text{\liAttributHuelle}\{F, \alpha \cup A\}$ .
2691     }
2692     {1-2} {
2693       Führe~für~jede~(verbliebene)~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~die~Rechtsreduktion~durch,~überprüfe~also~für~
2694       alle~ $B \in \beta$ ,~ob~ $B \in \text{\liAttributHuelle}\{F, (\alpha \rightarrow \beta) \cup (\alpha \rightarrow B)\}$ ,~
2695        $\alpha$ ~gilt.~In~diesem~Fall~ist~ $B$ ~auf~der~rechten~Seite~
2696       überflüssig~und~kann~eliminiert~werden,~d.h.~ $\alpha \rightarrow \beta$ ~wird~durch~ $\alpha \rightarrow (\beta \cup B)$ ~
2697       ersetzt.
2698     }
2699     {1-3} {
2700       Entferne~die~funktionalen~Abhängigkeiten~der~Form~ $\alpha \rightarrow \emptyset$ ,~die~im~2.~Schritt~möglicherweise~
2701       entstanden~sind.
2702     }
2703     {1-4} {
2704       Fasse~mittels~der~Vereinigungsregel~funktionale~Abhängigkeiten~
2705       der~Form~ $\alpha \rightarrow \beta_{\{1\}}, \dots, \alpha \rightarrow \beta_{\{n\}}$ ,~so~dass~ $\alpha \rightarrow \beta_{\{1\}} \cup \dots \cup \beta_{\{n\}}$ ~
2706       verbleibt.
2707     }
2708     % Kemper Seite 197
2709     {2} {
2710       Erzeuge~für~jede~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~in~ $F$ ~ein~Relationenschema~ $\text{\mathcal{R}}_{\alpha}$ ~
2711        $:= \alpha \cup \beta$ .
2712     }
2713     {3} {
2714       Falls~eines~der~in~Schritt~2.~erzeugten~Schemata~ $\text{\mathcal{R}}_{\alpha}$ ~
2715       einen~Schlüsselkandidaten~von~ $\text{\mathcal{R}}$ ~bezüglich~ $F$ ~
2716       enthält,~sind~wir~fertig,~sonst~wähle~einen~Schlüsselkandidaten~
2717        $\text{\mathcal{K}} \subseteqq \text{\mathcal{R}}$ ~aus~und~definiere~folgendes~
2718       zusätzliche~Schema:~ $\text{\mathcal{R}}_{\mathcal{K}} := \text{\mathcal{K}} \cup \text{\mathcal{F}}$ ~
2719       und~ $\text{\mathcal{R}}_{\mathcal{K}} := \emptyset$ 
2720     }
2721     {4} {
2722       Eliminiere~diejenigen~Schemata~ $\text{\mathcal{R}}_{\alpha}$ ,~die~in~einem~
2723       anderen~Relationenschema~ $\text{\mathcal{R}}_{\alpha'}$ ~enthalten~sind,~d.h.~
2724        $\text{\mathcal{R}}_{\alpha} \subseteqq \text{\mathcal{R}}_{\alpha'}$ .
2725     }
2726   }
2727 }
2728 }
2729 \def\liSyntheseErklaerung#1{
2730 {
2731   \itshape
2732   \footnotesize

```

```

2738     \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
2739   }
2740 }

```

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

```

2741 \def\liSyntheseUeberErklaerung#1{
2742   \liSyntheseUeberschrift{#1}\par
2743   \liSyntheseErklaerung{#1}
2744 }

```

```

2745 \ExplSyntaxOff
2746

```

2.45 tabelle.sty

2747 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2748 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]

2749 \RequirePackage{tabularx}

2750

2.46 typographie.sty

```
2751 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2752 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
2753 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
2754 formatierung.sty definiert.]
```

```
2755 \ExplSyntaxOn
```

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

```
2756 \RequirePackage{fontawesome}
```

`\liErledigt` `\liErledigt:` ☑

```
2757 \let\liErledigt=\faCheckSquareO
```

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

```
2758 \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 —

```
2759 \def\liParagraphMitLinien#1{
2760   \noindent
2761   \vrule height 2pt depth -1.6pt width 0.4cm
2762   \enspace
2763   #1
2764   \enspace
2765   \leaders\vrule height 2pt depth -1.6pt \hfill \null
2766   \par
2767   \medskip
2768 }
```

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

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

```
2769 \def\liGeschweifteKlammern#1#2#3#4{
2770   \par
2771   \medskip
2772   \noindent
2773   #1 \, $= \Bigl\{ $
2774   \vspace{#3}
2775   #2
2776   \vspace{#4}
2777   \begin{flushright} $\Bigr\} $\end{flushright}
2778   \par
2779 }
```

```
2780 \ExplSyntaxOff
```

```
2781
```

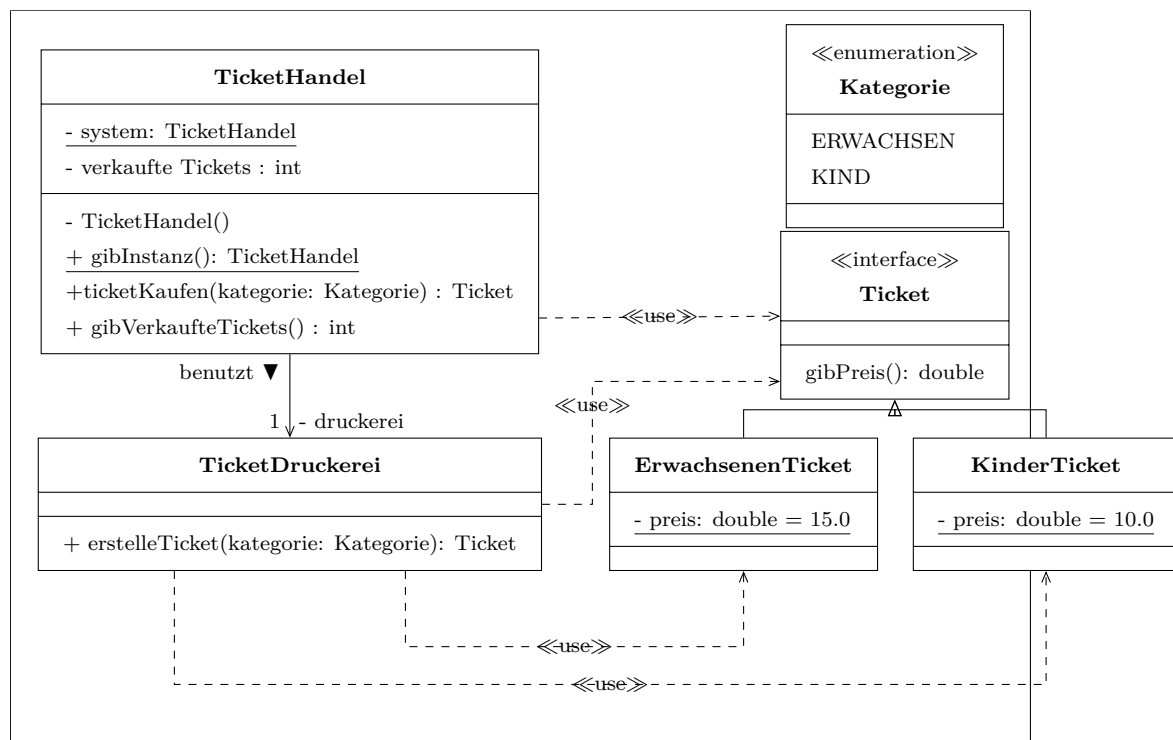
```

2782 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2783 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
2784 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
2785 Erweiterung bereitstellt]

2786 \RequirePackage{tikz-uml}
2787 \RequirePackage{tikz-uml-activity}
2788 % Not compatible with wasysym
2789 %\RequirePackage{mathabx}
2790 \RequirePackage{wasysym}
2791 \usetikzlibrary{positioning}

2792 \tikzumlset{
2793   fill class=white!0,
2794   font=\footnotesize,
2795   fill object=white!0,
2796   fill note=white!0,
2797   fill state=white!0,
2798   % Use case
2799   fill usecase=white!0,
2800   fill system=white!0,
2801 }

```



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

111

```

2814 \def\@liDistance{0cm}
2815 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
2816
2817 \pgfkeys{/lese/.cd,#1}
2818
2819 \node[\@liPos = \@liDistance of #2-middle] {
2820   \@liDirLeft{\footnotesize#2}\@liDirRight
2821 };
2822 }
2823

```


2.48 vollstaendige-induktion.sty

2824 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2825 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01
 2826 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
 2827 Überschriften für die einzelnen Schritte]

2.48.1 Makro-Kürzel

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

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

Lade häufig benötigte Pakete

2828 \RequirePackage{lehramt-informatik-typographie}
 2829 \RequirePackage{lehramt-informatik-mathe}
 2830 \RequirePackage{lehramt-informatik-syntax}
 2831 \ExplSyntaxOn

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

Let-Abkürzung: \let\m=\liInduktionMarkierung

2832 \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

```
2833 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
2834 \def\liInduktionAnfang{
```

```
2835   \liPseudoUeberschrift{Induktionsanfang}
```

```
2836
```

```
2837   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
2838   \liParagraphMitLinien{
```

```
2839     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
```

```
2840   }
```

```
2841 }
```

\liInduktionVoraussetzung

```
2842 \def\liInduktionVoraussetzung{
```

```
2843   \liPseudoUeberschrift{Induktionsvoraussetzung}
```

```
2844
```

```
2845   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
2846   \liParagraphMitLinien{
```

```
2847     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ ~.
```

```
2848   }
```

```
2849 }
```

\liInduktionSchritt

```
2850 \def\liInduktionSchritt{
```

```
2851   \liPseudoUeberschrift{Induktionsschritt}
```

```
2852
```

```
2853   % https://de.wikibooks.org/wiki/Mathe\_für\_Nicht-Freaks:\_Vollständige\_Induktion
```

```
2854   \liParagraphMitLinien{
```

```
2855     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
```

```
2856     auch~ $A(n=k+1)$ ~wahr~sein~muss.
```

```
2857   }
```

```
2858 }
```

```
2859 \ExplSyntaxOff
```

```
2860
```

2.49 wasserfall.sty

```
2861 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2862 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
2863 \RequirePackage{tikz}
2864 \tikzset{wasserfall/.style={
2865   >=stealth,
2866   node distance = 2mm and -8mm,
2867   start chain = A going below right,
2868   every node/.style = {
2869     draw,
2870     text width=24mm,
2871     minimum height=12mm,
2872     align=center,
2873     inner sep=1mm,
2874     fill=white,
2875     drop shadow={fill=black},
2876     on chain=A
2877   },
2878 }}
2879 \usetikzlibrary{chains,positioning,shadows}
2880
```

2.50 wpkalkuel.sty

```
2881 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2882 \ProvidesPackage{lehrant-informatik-wpkalkuel}[2020/06/13]
```

2.50.1 Makro-Kürzel

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

```
2883 \RequirePackage{amsmath}
```

```
2884 \ExplSyntaxOn
```

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

```
2885 \def\liWpKalkuelOhneMathe#1#2{
2886   \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)
2887 }
2888 \def\liWpKalkuel#1#2{
2889   \ifmmode
2890     \liWpKalkuelOhneMathe{#1}{#2}
2891   \else
2892     $\liWpKalkuelOhneMathe{#1}{#2}$
2893   \fi
2894 }
```

```
\MatheEnv
```

```
2895 \def\MatheEnv#1{
2896   \medskip
2897
2898   \hspace{1em}#1
2899
2900   \medskip
2901 }
```

```
\Mathe
```

```
2902 \def\Mathe#1{
2903   \MatheEnv{#1$}
2904 }
```

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

```
2905 \def\liWpEquivalent#1{
2906   \MatheEnv{$\equiv$\hspace{1em}$#1$}
2907 }
```

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

```
2908 \newlength{\@Skip@Erklaerung@Reset}
2909 \def\liWpErklaerung#1{
2910   \setlength{\@Skip@Erklaerung@Reset}{\leftskip}
2911   \setlength{\leftskip}{0.5cm}
2912
2913   \par
2914   \noindent
2915   {
2916     \scriptsize
2917     #1
2918   }
2919   \par
2920
2921   \setlength{\leftskip}{\@Skip@Erklaerung@Reset}
2922 }
```

`\liWpErklaerungVerzweigung`

```

2923 \def\liWpErklaerungVerzweigung{
2924   $\liWpKalkuelOhneMathe{if~\{-b~\}~then~\{-a1~\}~else~\{-a2~\}}{Q}
2925   \equiv
2926   (b \land \liWpKalkuelOhneMathe{a1}{Q})
2927   \lor
2928   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
2929 }

2930 \ExplSyntaxOff

2931

```

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		
<code>\#</code> 118	<code>\AddToHook</code> 1369	<code>\bfseries</code> 495, 1243, 1245,
<code>\,</code> 354, 409, 1190, 1192, 1771, 1772, 1773, 2110, 2587, 2773	<code>\advance</code> 2504	1387, 2277, 2283,
<code>\@Skip@Erklaerung@Reset</code> ... 2908, 2910, 2921	<code>\AfterEndEnvironment</code> 2580	2285, 2287, 2288, 2664
<code>\@afterheading</code> 1738	<code>\allsectionsfont</code> ... 1239	<code>\Bigl</code> 2773
<code>\@afterindentfalse</code> . 1738	<code>\Alph</code> 1253	<code>\Bigr</code> 2777
<code>\@liDirLeft</code> 2803, 2808, 2820	<code>\alph</code> 1253, 1254	<code>\bigskip</code> 387, 618, 623, 1647, 1978
<code>\@liDirRight</code> 2804, 2806, 2807, 2808, 2809, 2820	<code>\alpha</code> 2687, 2689, 2690, 2693, 2695, 2696, 2697, 2698, 2699,	<code>\bool</code> 332, 355
<code>\@liDistance</code> 2814, 2815, 2819	2703, 2709, 2710, 2715, 2716, 2717, 2720, 2728, 2729, 2730	<code>\bowtie</code> 2425, 2428, 2429, 2430
<code>\@liPos</code> .. 2811, 2812, 2819	<code>\arabic</code> 1253, 2299, 2304, 2309, 2315, 2321, 2327	<code>\Box</code> 168
<code>\</code> 616, 639, 640, 643, 644, 647, 648, 740, 741, 742, 849, 878, 880, 906, 915, 960, 1002, 1003, 1004, 1009, 1010, 1011, 1031, 1645, 2005, 2063, 2066	<code>\arraystretch</code> 2000	<code>\boxtimes</code> 485
<code>\{</code> 216, 1127, 1137, 1149, 1150, 1155, 1189, 1485, 2050, 2455, 2773, 2924	B	C
<code>\}</code> 216, 1127, 1137, 1149, 1150, 1157, 1193, 1486, 2050, 2455, 2777, 2924	<code>\BeforeBeginEnvironment</code> 2579	<code>\c</code> 1336, 1337
<code>_</code> .. 37, 45, 54, 56, 303, 329, 353, 354, 368, 369, 375, 378, 381, 391, 403, 1370, 1426	<code>\begin</code> 637, 688, 703, 738, 762, 809, 841, 856, 876, 886, 902, 922, 954, 971, 1000, 1025, 1046, 1061, 1170, 1259, 1334, 1439, 1448, 1455, 1563, 1637, 1689, 1694, 1702, 1727, 1732, 1743, 1753, 1757, 1840, 1844, 1863, 1886, 1909, 1924, 2003, 2004, 2054, 2099, 2219, 2239, 2380, 2402, 2416, 2579, 2590, 2777	<code>\cdot</code> 1815, 1871, 1882
<code>\sq</code> 2062, 2072	<code>\begingroup</code> 1638, 2052, 2500	<code>\centerline</code> 1444, 2378, 2400, 2415
A	<code>\beschriftung</code> 1669, 1673, 1677, 1681, 1685, 1687	<code>\chapter</code> 1243, 1244
<code>\addbibresource</code> 1615, 1616, 1617, 1618, 1619, 1620, 1621, 1622, 1623, 1624	<code>\beta</code> 2687, 2690, 2694, 2695, 2696, 2699, 2709, 2710, 2711, 2716, 2717	<code>\char</code> 1645
	<code>\bf</code> 2277, 2278, 2279	<code>\clearpage</code> 1388, 1639
		<code>\cline</code> 616
		<code>\clist</code> 235, 279, 280, 300, 304, 2512
		<code>\columnbreak</code> 2547
		<code>\cs</code> 303, 329, 353, 354, 391, 403, 1721
		<code>\csname</code> 1321, 1324
		<code>\cup</code> 1150, 2084, 2696, 2710, 2717
		D
		<code>\DeclareMathSymbol</code> 1948, 1949
		<code>\DecoINERT</code> 2302, 2375, 2376, 2398
		<code>\DecoINERTwithPivot</code> 2318, 2397
		<code>\DecoLEFT</code> 2297, 2396
		<code>\DecoLEFTwithPivot</code> 2312, 2374
		<code>\DecoRIGHT</code> 2307, 2399, 2409

<code>\ifmmode</code>	588, 596, 604, 1129, 1143, 1179, 1786, 1796, 1806, 1820, 2044, 2167, 2889	197, 198, 556, 559, 564, 565, 568, 573, 574, 577, 578, 583, 1209, 1210, 1211, 1212, 1215, 1216, 1217, 1218, 1224, 1225, 1226, 1227, 1579, 1580, 1581, 1724, 1725, 1726, 1733	<code>\li@EntwurfsCodeAllgemein</code> 632 <code>\li@fussnote@text</code> 1956, 1962, 1966, 1970, 1974 <code>\li@GithubLink</code> 2589, 2602, 2612, 2623 <code>\li@mget</code> . 1323, 1327, 1347 <code>\li@minc</code> 1326, 1348 <code>\li@mset</code> 1320, 1328, 1337, 1340 <code>\li@numdiscs</code> 1332, 1341, 1347 <code>\li@Rmodell@Schrift</code> 2439, 2448, 2458 <code>\li@sequence</code> .. 1333, 1346 <code>\li@synthese@erklaerung@texte</code> 2678, 2738 <code>\liAbleitung</code> 1164 <code>liAdditum</code> (environment) 1692 <code>liAHuelle</code> (environment) 2051 <code>\liAlphabet</code> 1149 <code>liAntwort</code> (environment) 1663 <code>\liAnweisung</code> 1570 <code>\liAssemblerCode</code> ... 2627 <code>\liAssemblerDatei</code> .. 2628 <code>\liAttribut</code> 2458 <code>\liAttributHuelle</code> 2042, 2690, 2695 <code>\liAttributHuelleOhneMathe</code> 2042, 2045, 2047, 2061, 2071, 2079 <code>\liAttributMenge</code> 2050, 2062, 2065, 2072, 2073, 2087, 2089 <code>\liAufgabe</code> 16 <code>\liAufgabenMetadaten</code> 51, 1424 <code>\liAufgabenTitel</code> 59 <code>\liAusdruck</code> 1186 <code>\liAutomat</code> 70 <code>\liAutomatenKante</code> ... 102 <code>\liBandAlphabet</code> 1150 <code>\liBedingung</code> 1571 <code>\liBedingungDrei</code> 1835, 1878, 1918 <code>\liBedingungEins</code> 1829, 1867, 1910 <code>\liBedingungFalsch</code> . 1573 <code>\liBedingungWahr</code> ... 1572 <code>\liBedingungZwei</code> 1832, 1873, 1914 <code>\liBeschriftung</code> 1652 <code>\liBindeAufgabeEin</code> . 1407 <code>\liChomsky@erklaerung@texte</code> 505, 541 <code>\li@EntwurfsCode</code> 633, 679, 680, 681, 727, 728, 729, 730, 796, 797, 798, 799, 800, 801, 828, 829, 830, 831, 832, 833, 894
<code>\IfNoValueTF</code>	1695, 1765, 1769		
<code>\ifnum</code>	2261, 2502, 2518, 2526, 2532		
<code>\ifx</code>	1685, 2082, 2530		
<code>\in</code>	514, 620, 1492, 1830, 1833, 1836, 1865, 1871, 1876, 2215, 2226, 2236, 2246, 2687, 2689, 2695, 2716, 2847	<code>\labelenumi</code> 1254 <code>\labelenumii</code> 1255 <code>\labelitemi</code> 1248 <code>\labelitemii</code> 1249 <code>\labelitemiii</code> 1250 <code>\labelitemiv</code> 1251 <code>\land</code> 2926, 2928 <code>\LARGE</code> 1243 <code>\Large</code> 1387 <code>\large</code> 1444, 2489 <code>\leaders</code> 2765 <code>\left</code> 1780 <code>\LEFTarrow</code> 2808 <code>\leftarrow</code> 602 <code>\leftouterjoin</code> 2428 <code>\leftskip</code> 2910, 2911, 2921 <code>\LehramtInformatikAutorEmail</code> 1595 <code>\LehramtInformatikAutorName</code> 1594 <code>\LehramtInformatikGitBranch</code> 396, 2573 <code>\LehramtInformatikGithubCodeRepo</code> 2572 <code>\LehramtInformatikGithubDomain</code> 2569 <code>\LehramtInformatikGithubRawDomain</code> 394, 2570 <code>\LehramtInformatikGithubTexRepo</code> 395, 2571 <code>\LehramtInformatikRepository</code> 17, 20, 23, 26, 29, 1410, 1615, 1616, 1617, 1618, 1619, 1620, 1621, 1622, 1623, 1624, 2568 <code>\LehramtInformatikTitel</code> 1590 <code>\leq</code> 1882, 2223, 2243 <code>\let</code> 1139, 1140, 1639, 2374, 2375, 2376, 2377, 2390, 2392, 2393, 2394, 2396, 2397, 2398, 2399, 2409, 2501, 2539, 2540, 2757	
<code>\inhaltsverzeichnis</code>	1636		
<code>\input</code>	17, 20, 23, 26, 29, 416, 1409		
<code>\inputminted</code>	2597, 2607, 2617, 2629, 2632, 2636		
<code>\int</code>	2512		
<code>\item</code>	485, 486, 705, 709, 714, 719, 763, 772, 777, 785, 857, 862, 866, 887, 923, 928, 935, 943, 972, 977, 981, 986, 1062, 1067, 1072, 1449, 1450, 1721, 1725, 1845, 1850, 1854, 1864, 1870, 1875, 1887, 1891, 1895, 1899, 1903, 1910, 1914, 1918, 2220, 2223, 2226, 2240, 2243, 2246		
<code>\itshape</code>	539, 2736		
J			
<code>\j</code>	1336, 1337, 1339, 1340, 1341, 1346, 1347, 1348		
K			
<code>\k</code>	1346		
<code>\keys</code>	41, 79, 91, 121, 131, 179, 189, 308, 558, 562, 576, 581, 1214, 1221		
L			
<code>\l</code>	72, 73, 74, 75, 76, 77, 80, 81, 82, 83, 84, 86, 88, 93, 94, 95, 96, 97, 98, 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,		

<code>\liO</code>	1802 , 1830	<code>liRmodell</code> (environment)	2439	<code>\liZustandsBuchstabe</code>	1151 ,
<code>\liOmega</code>	1792 , 1836	<code>\liRundeKlammer</code> .	1779 ,		1160 , 1162 , 1180 , 1182
<code>\liOmegaOhneMathe</code> . . .	1792 , 1797 , 1799		1783 , 1793 , 1803 , 1817	<code>\liZustandsBuchstabeGross</code>	1152 , 1161 , 1163
<code>\liOOhneMathe</code>	1802 , 1807 , 1809	<code>\liSetzeExamen</code>	1394	<code>\liZustandsmenge</code> . . .	1139
<code>\liParagraphMitLinien</code>	541 , 2016 , 2738 ,	<code>\liSetzeExamenTeilaufgabeNr</code>	1403	<code>\liZustandsmengeNr</code> . .	1153 , 2197
	2759 , 2838 , 2846 , 2854	<code>\liSetzeExamenThemaNr</code>	1399	<code>\liZustandsmengeNrGross</code>	1161
<code>\liPetriErreichKnotenDrei</code>	2176	<code>\liSortierMarkierung</code>	2478	<code>\liZustandsMengenSammlung</code>	2184
<code>\liPetriErreichTransition</code>	2173	<code>\liSortierPfeil</code>	2472	<code>\liZustandsMengenSammlungNr</code>	2193
<code>\liPetriSetzeSchluessel</code>	2118	<code>\liSortierPfeilUnten</code>	2475	<code>\liZustandsmengeOhneMathe</code>	1139
<code>\liPetriTransitionsName</code>	2165 , 2177	<code>\liSpaltenUmbruch</code> . .	2547	<code>\liZustandsname</code>	1162
<code>\liPetriTransitionsNameOhneMathe</code>	2165 , 2168 , 2170	<code>\liSqlCode</code>	2639	<code>\liZustandsnameGross</code>	1163 , 2185 , 2194
<code>\liPetriTransPfeile</code>	2177	<code>\listen@punkt</code> . .	1721 , 1733	<code>\liZustandsnameTiefgestellt</code>	1178
<code>\liPolynomiellReduzierbar</code>	1454	<code>\liStrich</code>	1436	<code>\liZustandsPaar</code>	1994
<code>\liPotenzmenge</code>	1136 , 1140 , 2188	<code>\liSyntheseErklaerung</code>	2678 , 2743	<code>\liZustandsPaarVariablenName</code>	1993 , 1996 , 1997
<code>\liPotenzmengeOhneMathe</code>	1137 , 1138 , 1139	<code>\liSyntheseUeberErklaerung</code>	2741		2426
<code>\liPrimaer</code>	2437	<code>\liSyntheseUeberschrift</code>	2662 , 2742	<code>\log</code>	1830 ,
<code>\liProblemBeschreibung</code>	1438	<code>\liT</code> 1812 , 1827 , 1841 , 1905			1833 , 1836 , 1865 , 1871
<code>\liProblemClique</code> . . .	1461	<code>\liTeilen</code>	2039	<code>\loop</code>	2259
<code>\liProblemName</code>	1437 , 1444 ,	<code>\liLiteratur</code>	1603 , 1627	<code>\lor</code>	2927
	1456 , 1458 , 1471 ,	<code>\liTheta</code>	1782 ,	<code>\ltimes</code>	1941
	1482 , 1483 , 1491 , 1492	1833 , 1865 , 1871 , 1876			
<code>\liProblemSat</code>	1490	<code>\liThetaOhneMathe</code> . . .	1782 , 1787 , 1789		
<code>\liProblemSubsetSum</code> .	1481 , 1490	<code>\liTOhneMathe</code>	1812 , 1821 , 1823		
<code>\liProblemVertexCover</code>	1461 , 1469	<code>\liTuringKante</code>	212		
<code>\liProduktionen</code> 1175 , 1217		<code>\liTuringLeerzeichen</code>	168 , 176		
<code>liProduktionsRegeln</code>	1165	<code>\liTuringMaschine</code> . . .	169		
(environment)		<code>\liTuringUeberfuehrung</code>	215		
<code>liProjektSprache</code> (envi-	1661	<code>\liTuringUebergaenge</code>	207 , 213		
ronment)		<code>\liTuringUebergangZelle</code>	202		
<code>\liPseudoUeberschrift</code>	1646 ,	<code>\liUeberfuehrungsFunktion</code>	1141		
	1696 , 1697 , 2002 ,	<code>\liUeberfuehrungsFunktionOhneMathe</code>	1141 , 1144 , 1146		
	2012 , 2835 , 2843 , 2851	<code>liUebergangsTabelle</code>	2000		
<code>\liPumpingKontextfrei</code>	2234	(environment)			
	2213	<code>\liUeberschriftDreiecksTabelle</code>	2010		
<code>liQuellen</code> (environment)	1721				
	2076	<code>\liUmlLeserichtung</code> .	2802		
<code>\liRechtsReduktionInline</code>	1826 , 1889	<code>\liVertauschen</code>	2466		
<code>\liRekursionsGleichung</code>	2106	<code>\liWortInSprache</code>	617		
<code>\liRelation</code>	2453	<code>\liWortNichtInSprache</code>	622		
<code>liRelationenSchemaFormat</code>	2459	<code>\liWpEquivalent</code>	2905		
(environment)		<code>\liWpErklaerung</code>	2908		
<code>\liRelationMenge</code> . . .	485	<code>\liWpErklaerungVerzweigung</code>	2923		
<code>\liRichtig</code>			2885		
		<code>\liWpKalkuel</code>	2885 , 2890 ,		
		<code>\liWpKalkuelOhneMathe</code>	2892 , 2924 , 2926 , 2928		

481, 488, 550, 613, 628, 1083, 1119, 1232, 1272, 1281, 1286, 1315, 1353, 1374, 1430, 1502, 1585, 1601, 1607, 1629, 1775, 1938, 1951, 2030, 2113, 2179, 2204, 2209, 2253, 2419, 2432, 2461, 2542, 2549, 2556, 2561, 2641, 2656, 2747, 2751, 2782, 2824, 2861, 2881	1715, 1738, 1746, 2378, 2400, 2415, 2456, 2742, 2766, 2770, 2778, 2913, 2919	\QSinitialize 2258, 2370, 2405
\neg 2928	\paragraph 1245	\QSIr . 2341, 2347, 2355, 2375, 2389, 2394, 2397
\negthinspace 1780	\parindent 2447	\QSIrr 2376, 2389, 2390, 2398
\newcounter ... 2273, 2274	\path ... 103, 150, 213, 583	\QSLr 2347, 2354, 2365, 2366, 2374, 2387, 2392, 2396
\NewDocumentCommand . 71, 112, 149, 170, 212, 233, 555, 572, 617, 622, 1186, 1208, 1454, 1577, 1605, 1764, 1768, 2095, 2106, 2173, 2596, 2606, 2616, 2628, 2631, 2635, 2802	\pgfkeys .. 2129, 2805, 2806, 2807, 2808, 2809, 2812, 2815, 2817	\QSpivotStep 2260, 2370, 2374, 2385
\NewDocumentEnvironment 1165, 1258, 1313, 1562, 1661, 1662, 1665, 1692, 1700, 1722, 1741, 1751, 2001, 2051, 2441, 2459	\pgfmath@count 2500, 2502, 2504	\QSRr 2347 2377, 2388, 2393, 2399, 2408, 2409, 2410
\newlength 2908	\pgfmath@smuggleone 2505	\QSSortStep 2262, 2370, 2386, 2387
\node 568, 1570, 1575, 2299, 2304, 2309, 2315, 2321, 2327, 2478, 2523, 2819	\pgfmathdeclarefunction 2499	\quad 1958
\noexpand 2387, 2388, 2389, 2408, 2523	\pgfmathhint 2500	
\noindent 366, 619, 624, 1648, 1650, 1654, 1658, 1686, 1714, 1716, 1729, 1745, 1747, 1755, 1927, 1980, 1983, 1986, 1989, 2454, 2760, 2772, 2914	\pgfmathparse 1327, 2517, 2522, 2525, 2539, 2540	R
\nolinkurl 399, 2593	\pgfmathresult 1328, 2500, 2501, 2503, 2505, 2518, 2526, 2539, 2540	\raisebox 1575
\normalfont 1387	\pgfutil@empty 2501	\relax 1639, 2347, 2390, 2392, 2393, 2394, 2502, 2504
\normalsize 1245	\pgfutil@loop 2502	\renewcommand ... 1248, 1249, 1250, 1251, 1254, 1255, 1269, 1270, 1596, 1597, 2000
\notin 625	\pgfutil@repeat 2505	\repeat 2263
\null 2765	\preceq 1457	\RequirePackage 4, 64, 167, 231, 235, 417, 423, 424, 484, 552, 631, 632, 1086, 1088, 1089, 1125, 1234, 1235, 1238, 1240, 1242, 1247, 1256, 1263, 1274, 1275, 1284, 1288, 1289, 1290, 1318, 1319, 1367, 1384, 1391, 1392, 1435, 1504, 1604, 1609, 1610, 1626, 1633, 1634, 1635, 1663, 1778, 1942, 1943, 2115, 2117, 2207, 2267, 2269, 2421, 2422, 2423, 2436, 2464, 2495, 2546, 2552, 2559, 2564, 2575, 2576, 2644, 2749, 2756, 2786, 2787, 2789, 2790, 2828, 2829, 2830, 2863, 2883
O	\prime 1436	\right 1780
\o@join 2425, 2428, 2429, 2430	\printbibliography . 1627	\RIGHTarrow ... 2804, 2809
\Omega 1793	\ProvidesPackage 2, 15, 33, 62, 230, 421, 482, 489, 551, 614, 629, 1084, 1120, 1233, 1273, 1282, 1287, 1316, 1354, 1375, 1431, 1503, 1586, 1602, 1608, 1630, 1776, 1939, 1952, 2031, 2114, 2180, 2205, 2210, 2254, 2420, 2433, 2462, 2543, 2550, 2557, 2562, 2642, 2657, 2748, 2752, 2783, 2825, 2862, 2882	\Rightarrow 620, 625
\omega 2215, 2216, 2236, 2237	Q	\rightarrow 216, 509, 514, 522, 526, 528, 529, 531, 586, 594, 2177, 2687, 2694, 2696, 2699, 2704, 2709, 2710, 2715
\or 2336	\QS@list 2380, 2391, 2395, 2402, 2408, 2413, 2416	\rightouterjoin 2429
P	\QS@select@equal 2355, 2359	
\pagestyle 1268	\QS@select@greater 2356, 2360	
\par ... 365, 386, 545, 1197, 1205, 1653,	\QS@select@smaller 2351, 2354, 2358	
	\QS@sort@a 2333, 2366, 2387, 2388	
	\QS@sort@b 2333, 2334	
	\QS@sort@c 2337, 2344	
	\QS@sort@d 2345, 2353	
	\QS@sort@empty . 2335, 2340	
	\QS@sort@single 2336, 2341	

<code>\Roman</code>	1253	T	<code>\TmpPlaceEight</code>	2137	
<code>\roman</code>	1253, 1255	<code>\tableofcontents</code> . . .	1640	<code>\TmpPlaceFive</code>	2134
<code>\romannumeral</code>	2345	<code>\text</code>	86, 88, 191, 2042, 2833, 2886	<code>\TmpPlaceFour</code>	2133
<code>\rtimes</code>	1941	<code>\textbf</code>	1091, 1462, 1471, 1482, 1491, 1649, 1656, 1687, 1715, 1730, 1746, 2005	<code>\TmpPlaceNine</code>	2138
<code>\rule</code> 2378, 2400, 2415, 2426		<code>\textcolor</code>	1574, 2832	<code>\TmpPlaceOne</code>	2130
S		<code>\textit</code>	960, 1002, 1003, 1004, 1005, 1756, 2050, 2110	<code>\TmpPlaceSeven</code>	2136
<code>\sb</code>	76, 86, 88, 117, 175, 521, 522, 526, 529, 530, 531, 1180, 1182, 1830, 1833, 1836, 1865, 1871, 2020, 2165, 2174, 2709, 2710, 2711, 2716, 2720, 2721, 2724, 2725, 2728, 2729, 2730	<code>\textsc</code>	1437	<code>\TmpPlaceSix</code>	2135
<code>\scriptscriptstyle</code> . .		<code>\textsf</code>	1649, 1730	<code>\TmpPlaceTen</code>	2139
<code>\scriptsize</code>	1199, 1515, 1522, 1528, 1590, 1591, 1594, 1595, 2833, 2886, 2916	<code>\textstyle</code>	1850, 1882	<code>\TmpPlaceThree</code>	2132
<code>\section</code> 56, 1385, 1386, 1401		<code>\texttt</code> 1114, 1437, 1571, 1572, 1573, 1574, 2886		<code>\TmpPlaceTwo</code>	2131
<code>\sectionbreak</code>	1388	<code>\textwidth</code>	1598	<code>\TmpScale</code>	2150
<code>\seq</code> . 1579, 1580, 1581, 1724, 1725, 1726, 1733		<code>\the page</code>	1267, 1593	<code>\TmpTransitionEight</code> .	2126, 2147
<code>\setbox</code>	2425	<code>\theparagraph</code>	1245	<code>\TmpTransitionFive</code> . .	2123, 2144
<code>\setcounter</code>	1246, 1389, 2379, 2401, 2415	<code>\thesection</code>	1387	<code>\TmpTransitionFour</code> . .	2122, 2143
<code>\setganttlinklabel</code> . .		<code>\Theta</code>	1783	<code>\TmpTransitionNine</code> . .	2127, 2148
<code>\setlength</code>	1598, 2447, 2910, 2911, 2921	<code>\thinspace</code>	2886	<code>\TmpTransitionOne</code> . . .	2119, 2140
<code>\setmainfont</code>	1236	<code>\tikz</code>	1575	<code>\TmpTransitionSeven</code> .	2125, 2146
<code>\setmainlanguage</code>	418	<code>tikz: bbaum</code>	25	<code>\TmpTransitionSix</code> . . .	2124, 2145
<code>\setminted</code>	2581, 2582	<code>tikz: li binaer baum</code>	23	<code>\TmpTransitionTen</code> . . .	2128, 2149
<code>\setminus</code>	2080	<code>\tikzchildnode</code>	441	<code>\TmpTransitionThree</code> .	2121, 2142
<code>\setsansfont</code>	1237	<code>\tikzparentnode</code>	441	<code>\TmpTransitionTwo</code> . . .	2120, 2141
<code>\setul</code>	2438	<code>\tikzset</code>	105, 152, 218, 426, 452, 1292, 1506, 2155, 2281, 2485, 2646, 2864	<code>\TmpX</code>	2151
<code>\sffamily</code>	496, 1243, 1245, 1347, 2665	<code>\tikzumlset</code>	2792	<code>\TmpY</code>	2152
<code>\shoveleft</code>	2060	<code>\times</code>	216	<code>\today</code>	1591
<code>\shoveright</code>	2064	<code>\tiny</code>	1097, 1103, 1109, 1574, 1644, 2591	<code>\ttfamily</code>	2439
<code>\Sigma</code>	73, 114, 172, 1149, 1150, 1210	<code>\titleformat</code>		U	
<code>\sigma</code>	519, 521, 522	<code>\titlespacing</code>	1243, 1245, 1385, 1386	<code>\ul</code>	1092, 2437, 2438
<code>\SLASH</code>	1645	<code>\titlespacing</code>	1244	<code>\umlaggreg</code>	1055
<code>\small</code>	1754	<code>\tl</code>	39, 47, 72, 73, 74, 75, 76, 77, 80, 81, 82, 83, 84, 86, 88, 113, 114, 115, 116, 117, 118, 119, 122, 123, 124, 125, 126, 127, 128, 171, 172, 173, 174, 175, 176, 177, 180, 181, 182, 183, 184, 185, 186, 301, 305, 330, 334, 335, 336, 339, 344, 345, 346, 357, 358, 359, 360, 371, 377, 380, 383, 392, 406, 556, 559, 564, 565, 573, 574, 577, 578, 1188, 1209, 1210, 1211, 1212, 1215, 1216, 1217, 1218, 1395, 1396, 1397, 1400, 1404, 1408, 1415, 1418, 1813	<code>\umlclass</code>	1037
<code>\sort</code>	2512			<code>\umlclass</code>	638, 642, 646, 690, 691, 692, 739, 744, 749, 752, 810, 811, 812, 817, 818, 847, 877, 904, 905, 908, 959, 962, 1001, 1007, 1008, 1026, 1027, 1028, 1047, 1048, 1049, 1050
<code>\sortList</code>	2511, 2520			<code>\umldep</code>	967
<code>\square</code>	486			<code>\umlHVHaggreg</code>	757, 823, 1017
<code>\stepcounter</code> 2299, 2304, 2309, 2312, 2314, 2318, 2320, 2324, 2326				<code>\umlinherit</code>	696, 747, 912, 957, 965
<code>\str</code>	497, 506, 1667, 2189, 2198, 2666, 2679			<code>\umlnote</code>	698, 914, 1057
<code>\string</code>	2062, 2072			<code>\umlreal</code>	694, 755
<code>\StrSubstitute</code> . 2519, 2521				<code>\umlsimpleclass</code> . . .	653, 654, 655, 659, 661, 662, 663, 689, 842, 843, 844, 903, 955, 956
<code>\strut</code>	1887, 1891, 1895, 1899, 1903, 2547			<code>\umlstatic</code>	849, 878
<code>\subsection</code>	1405			<code>\umluniaggreg</code>	910
<code>\subseq</code> 2690, 2723, 2730				<code>\umluniassoc</code>	673, 695, 911, 1035, 1036
<code>\subsubsection</code>	1426	<code>\tmp</code>	2082		

<code>\umlVHuniassoc</code> .. 674, 675	V	<code>\xintFor</code>
<code>\umlVHVdep</code> 667,	<code>\value</code> 2261	2298, 2303, 2308,
668, 670, 671, 851, 852	<code>\varepsilon</code> 498,	2313, 2319, 2325, 2366
<code>\umlVHVinherit</code>	509, 510, 1135,	<code>\xintIfEq</code> 2359
650, 651, 656, 657,	1830, 1836, 1868, 1879	<code>\xintIfForLast</code>
664, 665, 820, 821,	<code>\vfill</code> 2547	... 2315, 2321, 2327
845, 846, 1015, 1016	<code>\vrule</code> 2761, 2765	<code>\xintIfGt</code> 2360
<code>\umlVHVreal</code>	<code>\vspace</code>	<code>\xintIfLt</code> 2358
. 814, 815, 1052, 1053	1701, 1719, 2774, 2776	<code>\xintLength</code> 2333
<code>\UParrow</code> 2806	X	<code>\xintntheft</code> 2345
<code>\url</code> 1765	<code>\xappto</code> .. 2523, 2529, 2533	Z
<code>\usemintedstyle</code> 2578	<code>\xdef</code> 1321	<code>\ZB</code> <u>1772</u>
<code>\usetikzlibrary</code> .. 65,	<code>\xintApply</code> 2349	<code>\zB</code> <u>1771</u>
425, 1087, 1291,	<code>\xintApplyUnbraced</code> ..	<code>\zustandsnamens@liste</code>
1505, 2116, 2424,	2348, 2354, 2355, 2356	... 1153, 1160, 1161
2465, 2496, 2791, 2879	<code>\xintCSVtoList</code> 2408	