

lehramt-informatik

Hermine Bschlangaul *

September 8, 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	23
2.6.1	Binärbaum	24
2.6.2	AVL-Baum	25
2.6.3	B-Baum	26
2.7	checkbox.sty	27
2.8	chomsky-normalform.sty	28
2.8.1	Makro-Kürzel	28
2.8.2	TeX-Markup-Grundgerüst	28
2.8.3	Konkretes TeX-Markup-Beispiel	28
2.9	cpm.sty	31
2.9.1	Makro-Kürzel	31
2.9.2	TeX-Markup-Beispiel: Graph	31
2.9.3	TeX-Markup-Beispiel: Ergebnistabelle	31
2.9.4	TeX-Markup-Beispiel: Nebenrechnungstabelle „Frühester Zeitpunkt“	31
2.9.5	TeX-Markup-Beispiel: Nebenrechnungstabelle „Spätester Zeitpunkt“	32
2.9.6	Hilfsmakros für Zeitpunkt-Tabelle	33
2.10	cyk-algorithmus.sty	35
2.10.1	Makro-Kürzel	35
2.10.2	TeX-Markup-Beispiel	35
2.11	entwurfsmuster.sty	36
2.11.1	Namensschema der Entwurfsmuster-Makros:	36
2.11.2	Reihenfolge	36
2.11.3	Abstrakte Fabrik (Abstract Factory)	36
2.11.4	Adapter	38
2.11.5	Beobachter (Observer)	39
2.11.6	Dekorierer (Decorator)	41
2.11.7	Einfache Fabrik (Simple Factory)	42
2.11.8	Einzelstück (Singleton)	43

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

2.11.9	Erbauer (Builder)	44
2.11.10	Fabrikmethode (Factory Method)	45
2.11.11	Kompositum (Composite)	47
2.11.12	Modell-Präsentation-Steuerung (Model-View-Controller)	47
2.11.13	Stellvertreter (Proxy)	48
2.11.14	Zustand (State)	49
2.12	er.sty	51
2.12.1	Tex-Markup-Beispiel: Komplettes Diagramm	51
2.12.2	Tex-Markup-Beispiel: EER Enhanced Entity-Relation-Modell nach Elmasri/Navante	52
2.12.3	Makro-Kürzel	52
2.13	formale-sprachen.sty	54
2.14	formatierung.sty	57
2.14.1	Schriftarten / Typographie	57
2.14.2	Farben	57
2.14.3	Überschriften	57
2.14.4	Listen	57
2.14.5	Kasten	57
2.14.6	Header	57
2.15	gantt.sty	58
2.16	grafik.sty	59
2.17	graph.sty	60
2.18	hanoi.sty	62
2.19	klassen-konfiguration-aufgabe.sty	63
2.20	klassen-konfiguration-examen.sty	64
2.21	komplexitaetstheorie.sty	66
2.21.1	Makro-Kürzel	66
2.22	kontrollflussgraph.sty	68
2.22.1	Makro-Kürzel	68
2.22.2	TeX-Markup-Beispiel	68
2.22.3	TikZ: pin	68
2.22.4	Umgebungen	69
2.22.5	Makros	70
2.23	kopf-fusszeilen.sty	71
2.24	literatur-dummy.sty	72
2.25	literatur.sty	73
2.26	makros.sty	74
2.27	master-theorem.sty	78
2.27.1	Makro-Kürzel	78
2.28	mathe.sty	82
2.29	minimierung.sty	83
2.30	normalformen.sty	86
2.30.1	Makro-Kürzel	86
2.31	o-notation.sty	89
2.31.1	Makro-Kürzel	89
2.32	petri.sty	90
2.32.1	Makro-Kürzel	90
2.33	potenzmengen-konstruktion.sty	92
2.34	pseudo.sty	94
2.35	pumping-lemma.sty	95
2.36	quicksort.sty	96
2.37	relationale-algebra.sty	99
2.38	rmodell.sty	100
2.38.1	Makro-Kürzel	100
2.39	sortieren.sty	101
2.40	spalten.sty	103
2.41	sql.sty	104
2.42	struktogramm.sty	105

2.43	syntax.sty	106
2.43.1	Makro-Kürzel	106
2.44	syntaxbaum.sty	108
2.45	synthese-algorithmus.sty	109
2.45.1	Makro-Kürzel	109
2.45.2	TeX-Markup Grundgerüst	109
2.45.3	TeX-Markup Linksreduktion	109
2.45.4	TeX-Markup Rechtsreduktion	109
2.45.5	TeX-Markup Relationen formen	110
2.46	tabelle.sty	113
2.47	typographie.sty	114
2.48	uml.sty	116
2.49	vollstaendige-induktion.sty	118
2.49.1	Makro-Kürzel	118
2.50	wasserfall.sty	120
2.51	wpkalkuel.sty	121
2.51.1	Makro-Kürzel	121
3	Index	122

1 Klassen

1.1 Vorlage Theorie-Teil

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

\begin{document}

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

\chapter{Thema des Theorie-Teils}

\literatur

\end{document}
```

1.2 Vorlage Aufgabensammlung

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

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

1.3 Vorlage Aufgabe

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

\end{document}
```

2 Pakete

2.1 abmessung.sty

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

2.2 aufgaben-einbinden.sty

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

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

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

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

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

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

31
```

2.3 aufgaben-metadaten.sty

```

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

35 \ExplSyntaxOn

\liMetaSetze Setze die Metadaten einer Aufgabe.

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

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

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

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

54 \ExplSyntaxOff

\liAufgabenTitel Momentan eine dummy Makro das die Thematik enthält.

55 \def\liAufgabenTitel#1{}

56

```

2.4 automaten.sty

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

2.4.1 Endlicher Automat

```
59 \liLadePakete{formale-sprachen}

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

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



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

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

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

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

```

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

```

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

```

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

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

```

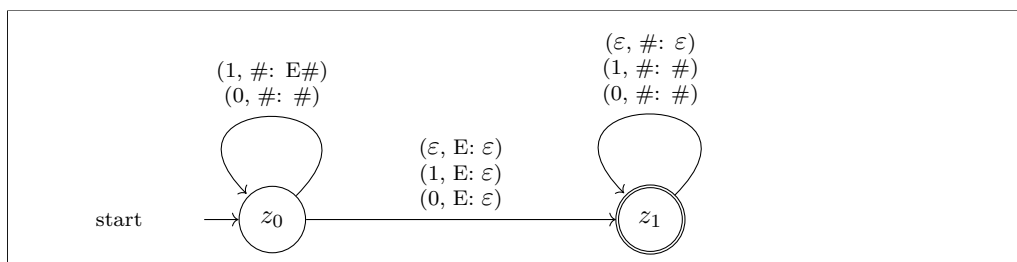
2.4.2 Kellerautomat

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

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

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

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



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

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

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

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

```

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

```

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

(a, #: A)
(b, #: ε)

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

```

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

```

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

```

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

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

```

2.4.3 Turingmaschine

```

163 \RequirePackage{amssymb}

```

\liTuringLeerzeichen

□

```

164 \def\liTuringLeerzeichen{\Box}

```

```

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

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

```

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

```

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

```

\liTuringUebergangZelle Formatiert einen Zustandsübergang für eine Übergangstabelle.

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

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

```

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

```

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

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

`(z1: \square , L)`

`(\square : \square , R)`


```

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

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

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

\liTuringUeberfuehrung

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

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

```

2.5 basis.sty

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

227 \RequirePackage{xparse}

228 \ExplSyntaxOn

\liLadePakete

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

\liLadeAllePakete

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

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

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

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

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

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

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

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

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

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

```

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

357 \cs_set:Nn \_trenner: {
358   \, / \,
359 }

360 \cs_gset:Npn \_gib_jahreszeit_durch_monat: #1 {
361   % \str_case funktioniert nicht mit den Tokenlist variablen.
362   \tl_case:Nn { #1 }
363   {
364     { 3 } { Frühjahr }
365     { 03 } { Frühjahr }
366     { 9 } { Herbst }
367     { 09 } { Herbst }
368   }
369 }

```

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

```

370 \cs_gset:Npn \_gib_examen_fach_durch_nummer: #1 {
371   \tl_case:Nn { #1 }
372   {
373     { 46110 } { Grundlagen der Informatik (nicht vertieft) }
374     { 46111 } { Programmentwicklung / Systemprogrammierung / Datenbanksysteme (nicht vertieft) }
375     { 46112 } { Grundlagen der Informatik (nicht vertieft) }
376     { 46113 } { Theoretische Informatik (nicht vertieft) }
377     { 46114 } { Algorithmen / Datenstrukturen / Programmiermethoden (nicht vertieft) }
378     { 46115 } { Theoretische Informatik / Algorithmen / Datenstrukturen (nicht vertieft) }
379     { 46116 } { Softwaretechnologie / Datenbanksysteme (nicht vertieft) }
380     { 46118 } { Fachdidaktik (Mittelschulen) }
381     { 46119 } { Fachdidaktik (Realschulen) }
382     { 46121 } { Fachdidaktik (berufliche Schulen) }
383     { 66110 } { Automatentheorie, Algorithmische Sprache (vertieft) }
384     { 66111 } { Betriebssysteme / Datenbanksysteme / Rechnerarchitektur (vertieft) }
385     { 66112 } { Automatentheorie / Komplexität / Algorithmen (vertieft) }
386     { 66113 } { Rechnerarchitektur / Datenbanken / Betriebssysteme (vertieft) }
387     { 66114 } { Datenbank- und Betriebssysteme (vertieft) }
388     { 66115 } { Theoretische Informatik / Algorithmen (vertieft) }
389     { 66116 } { Datenbanksysteme / Softwaretechnologie (vertieft) }
390     { 66118 } { Fachdidaktik (Gymnasium) }
391   }
392 }

```

Einzelprüfungsnummer / Jahr / Jahreszeit mit Trennzeichen

```

393 \cs_gset:Npn \_gib_einzelpreuefung_trenner: {
394   Staatsexamen ~
395   \g_auf_examen_nummer_tl
396
397   \_trenner:
398
399   \g_auf_examen_jahr_tl
400
401   \_trenner:
402
403   \gib_jahreszeit_durch_monat: \g_auf_examen_monat_tl
404 }

Thema Nr.1 / Teilaufgabe Nr. 2 / Aufgabe 3
405 \cs_gset:Npn \_gib_aufgaben_pfad_trenner: {
406   \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
407     Thema ~ \g_auf_examen_thema_nr_tl \_trenner:
408   }
409   \tl_if_empty:NTF \g_auf_examen_telaufgabe_nr_tl {} {
410     Teilaufgabe ~ \g_auf_examen_telaufgabe_nr_tl \_trenner:
411   }
412   \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} {
413     Aufgabe ~ \g_auf_examen_aufgabe_nr_tl
414   }
415 }

416 \cs_gset:Npn \_gib_examen_titel: {
417   \cs_set:Nn \_trenner: { \, / \, }
418   \bool_if:nTF
419   {
420     ! \tl_if_empty_p:N \g_auf_examen_nummer_tl &&
421     ! \tl_if_empty_p:N \g_auf_examen_jahr_tl &&
422     ! \tl_if_empty_p:N \g_auf_examen_monat_tl &&
423     ! \tl_if_empty_p:N \g_auf_examen_aufgabe_nr_tl
424   }
425   {
426     {
427       \footnotesize
428       \par
429       \noindent
430       Staatsexamen ~
431       \g_auf_examen_nummer_tl \_trenner:
432       \g_auf_examen_jahr_tl \_trenner:
433
434       \tl_case:Nn \g_auf_examen_monat_tl
435       {
436         { 03 } { Frühjahr }
437         { 09 } { Herbst }
438       } \_trenner:
439
440       \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
441         Thema ~ Nr. ~ \g_auf_examen_thema_nr_tl \_trenner:
442       }
443       \tl_if_empty:NTF \g_auf_examen_telaufgabe_nr_tl {} {
444         Teilaufgabe ~ Nr. ~ \g_auf_examen_telaufgabe_nr_tl \_trenner:
445       }
446       \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} {
447         Aufgabe ~ Nr. ~ \g_auf_examen_aufgabe_nr_tl
448       }
449       \par
450       \bigskip
451     }
452   }
453 }

```

```

454 \cs_new:Npn \_gib_github_url: {
455   \LehramtInformatikGithubDomain /
456   \LehramtInformatikGithubTexRepo /
457   blob /
458   \LehramtInformatikGitBranch /
459   \g_auf_relativer_pfad_tl
460 }

461 \cs_new:Npn \_gib_github_url_href: {
462   \tl_if_empty:NTF \g_auf_relativer_pfad_tl {} {
463     \url{ \_gib_github_url: }
464   }
465 }

466 \cs_new:Npn \_gib_aufgaben_titel: {
467   \g_auf_titel_tl
468
469   \tl_if_empty:NTF \g_auf_thematik_tl
470   {}
471   {
472     \, ~ [
473       \g_auf_thematik_tl
474     ]
475   }
476 }

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

\def\LehramtInformatikRepository{/pfad/zum/repository}
479 \input /etc/lehramt-informatik.config.tex

biblatex not working with lualatex and babel
480 % \RequirePackage{polyglossia}
481 % \setmainlanguage{german}
482

```

2.6 baum.sty

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

486 \RequirePackage{tikz}

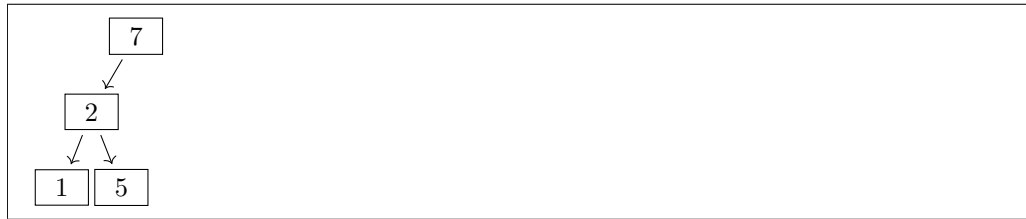
    für li binaer baum
487 \RequirePackage{tikz-qtrees}

    Für b baum
488 \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}
```



```
489 \tikzset{
490   li binaer baum/.style={
491     shorten <=2pt,
492     shorten >=2pt,
493     ->,
494     every tree node/.style={
495       minimum width=2em,
496       draw,
497       rectangle
498     },
499     blank/.style={
500       draw=none
501     },
502     edge from parent/.style={
503       draw,
504       edge from parent path={(\tikzparentnode) -- (\tikzchildnode)}
505     },
506     level distance=1cm,
507     every label/.style={
508       gray,
509       font=\footnotesize,
510       label position=0,
511       label distance=0cm,
512     }
513   },
514 }
```


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}
```

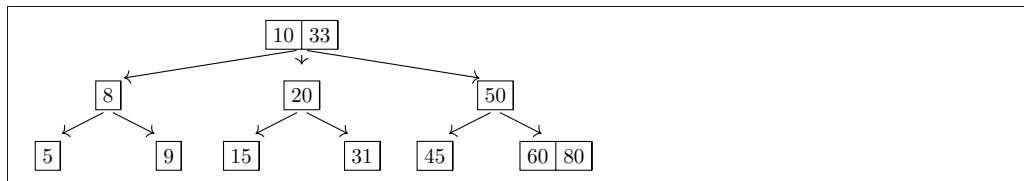


2.6.3 B-Baum

```

\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}

```



```

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

```

2.7 checkbox.sty

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

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

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

550
```

2.8 chomsky-normalform.sty

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

554 \ExplSyntaxOn

555 \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

```

556 \def\liChomskyUeberschrift#1{
557   {
558     \bfseries
559     \sffamily
560     \str_case:nn {#1} {
561       {1} {Elimination-der- $\varepsilon$ -Regeln}
562       {2} {Elimination-von-Kettenregeln}
563       {3} {Separation-von-Terminalzeichen}
564       {4} {Elimination-von-mehrelementigen-Nonterminalketten}
565     }
566   }
567 }

```

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

```

568 \def\liChomsky@erklaerung@texte#1{
569   \str_case:nn {#1} {
570     %
571     {1} {
572       Alle-Regeln-der-Form- $A \rightarrow \varepsilon$ -werden-eliminiert.~
573       Die-Ersetzung-von- $A$ -wird-durch- $\varepsilon$ -in-allen-anderen-
574       Regeln-vorweggenommen.
575     }
576     {2} {
577       Jede-Produktion-der-Form- $A \rightarrow B$ -mit- $A, B$ -in- $S$ -wird-
578       als-Kettenregel-bezeichnet.~Diese-tragen-nicht-zur-Produktion-
579       von-Terminalzeichen-bei-und-lassen-sich-ebenfalls-eliminieren.
580     }
581     {3} {
582       Jedes-Terminalzeichen- $\sigma$ ,~das-in-Kombination-mit-anderen-
583       Symbolen-auftaucht,~wird-durch-ein-neues-Nonterminal-
584        $S_{\sigma}$ -ersetzt-und-die-Menge-der-Produktionen-durch-die-
585       Regel- $S_{\sigma} \rightarrow \sigma$ -ergänzt.
586     }
587     {4} {
588       Alle-Produktionen-der-Form-
589        $A \rightarrow B_{\{1\}} B_{\{2\}} \dots B_{\{n\}}$ -
590       werden-in-die-Produktionen-
591        $A \rightarrow A_{\{n-1\}} B_{\{n\}}, A_{\{n-1\}} \rightarrow A_{\{n-2\}} B_{\{n-1\}}, \dots,$ 
592        $A_{\{2\}} \rightarrow B_{\{1\}} B_{\{2\}}$ -zerteilt.~
593       Nach-der-Ersetzung-sind-alle-längeren-Nonterminalketten-
594       vollständig-heruntergebrochen-und-die-Chomsky-Normalform-erreicht.
595     }
596   }
597 }
598 }

```

```

599 }
600 \def\liChomskyErklaerung#1{
601   {
602     \itshape
603     \footnotesize
604     \liParagraphMitLinien{\li@chomsky@erklaerung@texte{#1}}
605   }
606 }

```

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

```

607 \def\liChomskyUeberErklaerung#1{
608   \liChomskyUeberschrift{#1}\par
609   \liChomskyErklaerung{#1}
610 }

```

```

611 \ExplSyntaxOff
612

```

2.9 cpm.sty

```

613 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
614 \ProvidesPackage{lehramt-informatik-cpm}[2020/09/03]
615 \RequirePackage{tikz}

616 \liLadePakete{mathe,typographie}

```

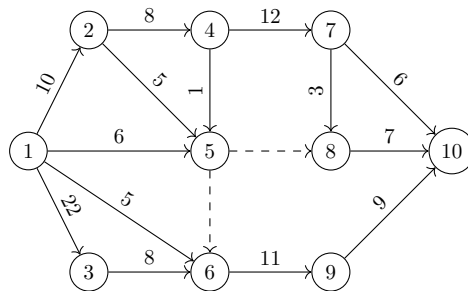
2.9.1 Makro-Kürzel

```

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

```

2.9.2 TeX-Markup-Beispiel: Graph



```

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

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

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

```

2.9.3 TeX-Markup-Beispiel: Ergebnistabelle

```

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

```

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

```

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

```

```

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

```

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

```

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

```

```

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

```

```

617 \ExplSyntaxOn
618 \NewDocumentCommand { \liCpmEreignis } { 0 } m m m } {
619   \tl_set:Nn \l_name_tl {}
620
621   \keys_define:nn { cpmEreignis } {
622     name .code:n = {\tl_set:Nn \l_name_tl {##1}},
623   }
624
625   \keys_set:nn { cpmEreignis } { #1 }
626
627   \tl_if_empty:NT \l_name_tl {
628     \tl_set:Nn \l_name_tl {#2}
629   }
630
631   \node[circle,draw] (\l_name_tl) at (#3,#4) {#2};
632 }
633 \ExplSyntaxOff

```

```

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

```

```

634 \ExplSyntaxOn
635 \NewDocumentCommand { \liCpmVorgang } { 0 } m m m } {
636   \tl_set:Nn \l_schein_tl {}
637   \tl_set:Nn \l_kritisch_tl {}
638
639   \keys_define:nn { cpmVorgang } {
640     schein .code:n = {\tl_set:Nn \l_schein_tl {dashed}},
641     kritisch .code:n = {\tl_set:Nn \l_kritisch_tl {thick}},
642   }
643
644   \keys_set:nn { cpmVorgang } { #1 }
645
646   \path[->,\l_schein_tl,\l_kritisch_tl] (#2) edge node[auto,sloped] {#4} (#3);
647 }
648 \ExplSyntaxOff

```


2.9.6 Hilfsmakros für Zeitpunkt-Tabelle

```

\begin{tabular}{|1||1|1|1|1|1|1|1|1|1|}
\hline
$i$ & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & \\ \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 & 0 & 2 & 0 & 0 & 2 & 0 \\ \hline
\end{tabular}

\liCpmVonZu Let-Abkürzung: \let\vz=\liCpmVonZu
\liCpmVonZu{1}(2-3):  $1_{(2 \rightarrow 3)}$ 
649 \def\liCpmVonZuOhneMathe#1(#2-#3){#1_{\scriptscriptstyle(#2\rightarrow#3)}}
650 \def\liCpmVonZu#1(#2-#3){%
651 \ifmmode%
652 \liCpmVonZuOhneMathe{#1}(\#2-\#3)%
653 \else%
654 $\liCpmVonZuOhneMathe{#1}(\#2-\#3)$%
655 \fi%
656 }

\liCpmVon Let-Abkürzung: \let\v=\liCpmVon
\liCpmVon{1}(2):  $1_{(\rightarrow 2)}$ 
657 \def\liCpmVonOhneMathe#1(#2){#1_{\scriptscriptstyle(\rightarrow#2)}}
658 \def\liCpmVon#1(#2){%
659 \ifmmode%
660 \liCpmVonOhneMathe{#1}(\#2)%
661 \else%
662 $\liCpmVonOhneMathe{#1}(\#2)$%
663 \fi%
664 }

\liCpmZu Let-Abkürzung: \let\z=\liCpmZu
\liCpmZu{1}(2):  $1_{(\leftarrow 2)}$ 
665 \def\liCpmZuOhneMathe#1(#2){#1_{\scriptscriptstyle(\leftarrow#2)}}
666 \def\liCpmZu#1(#2){%
667 \ifmmode%
668 \liCpmZuOhneMathe{#1}(\#2)%
669 \else%
670 $\liCpmZuOhneMathe{#1}(\#2)$%
671 \fi%
672 }

673 \ExplSyntaxOn

\liCpmSpaetI Spätester Zeitpunkt, zu dem Ereignis  $i$  eintreten kann
Let-Abkürzung: \let\SZ=\liCpmSpaetI
674 \NewDocumentCommand{ \liCpmSpaetI } { 0{i} } {
675 \ifmmode
676 SZ\sb{#1}
677 \else
678 $\SZ\sb{#1}$
679 \fi
680 }

\liCpmFruehI Frühester Zeitpunkt, zu dem Ereignis  $i$  eintreten kann.
Let-Abkürzung: \let\FZ=\liCpmFruehI
681 \NewDocumentCommand{ \liCpmFruehI } { 0{i} } {
682 \ifmmode
683 FZ\sb{#1}
684 \else
685 $\FZ\sb{#1}$

```

```

686 \fi
687 }

```

\liCpmFruehErklaerung

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

```

688 \def\liCpmFruehErklaerung{
689   \liParagraphMitLinien{
690     Wir~führen~eine~Vorwärtsterminierung~durch~
691     und~addieren~die~Dauern.~
692
693     Kann~ein~Ereignis~über~mehrere~Vorgänge~erreicht~
694     werden,~wählen~wir~das~Maximum~aus.~
695
696     \textbf{Erläuterungen:}~
697
698     $\mathit{i}$:~
699     Ereignis~$\mathit{i}$;~,
700
701     \liCpmFruehI{}:~
702     Frühester~Zeitpunkt,~zu~dem~Ereignis~$\mathit{i}$~eintreten~kann
703     .
704   }
705 }

```

\liCpmSpaetErklaerung

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

```

706 \def\liCpmSpaetErklaerung{
707   \liParagraphMitLinien{
708     Wir~führen~eine~Rückwärtsterminierung~durch~
709     und~subtrahieren~die~Dauern~vom~letzten~Ereignis~aus.~
710
711     Kann~ein~Ereignis~über~mehrere~Vorgänge~erreicht~
712     werden,~wählen~wir~das~Minimum~aus.~
713
714     \textbf{Erläuterungen:}~
715
716     $\mathit{i}$:~
717     Ereignis~$\mathit{i}$;~,
718
719     \liCpmSpaetI{}:~
720     Spätester~Zeitpunkt,~zu~dem~Ereignis~$\mathit{i}$~eintreten~kann
721     .
722   }
723 }

724 \ExplSyntaxOff
725

```

2.10 cyk-algorithmus.sty

```
726 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
727 \ProvidesPackage{lehramt-informatik-cyk-algorithmus}[2021/06/18 Hilfsmakros
728 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
```

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

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

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

```
730 \NewDocumentCommand{ \liWortInSprache } { m O{L(G)} } {
731   \bigskip
732   \noindent
733   $\Rrightarrow$ #1 \in #2$
734 }
```

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

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

```
735 \NewDocumentCommand{ \liWortNichtInSprache } { m O{L(G)} } {
736   \bigskip
737   \noindent
738   $\Rrightarrow$ #1 \notin #2$
739 }
```

```
740
```

2.11 entwurfsmuster.sty

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

2.11.1 Namensschema der Entwurfsmuster-Makros:

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

2.11.2 Reihenfolge

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

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

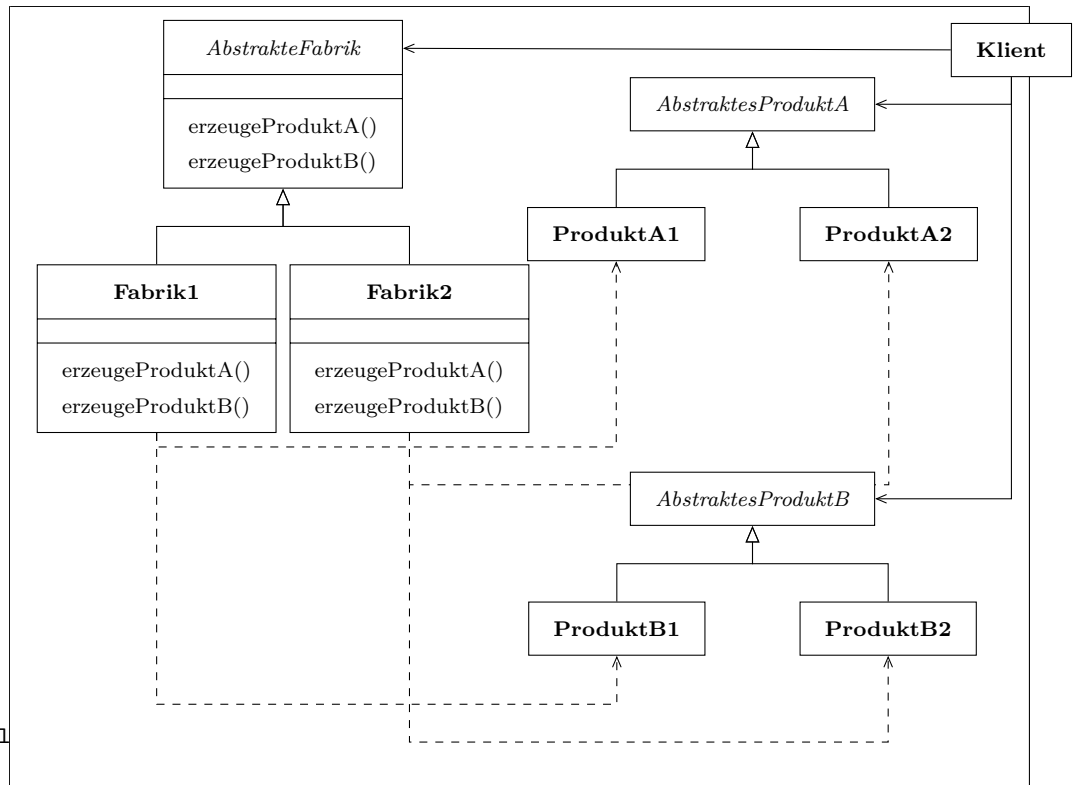
\li@EntwurfsCodeAllgemein Allgemeine Code-Beispiele zu den UML-Diagrammen und Stellvertretern

```
745 \RequirePackage{lehramt-informatik-syntax}
746 \def\li@EntwurfsCode#1#2{
747   \liJavaDatei{entwurfsmuster/#1/allgemein/#2}
748 }
```

2.11.3 Abstrakte Fabrik (Abstract Factory)

sAbstrakteFabrikBeschreibung

```
749 \def\liEntwurfsAbstrakteFabrikBeschreibung{
750   Es wird eine Schnittstelle bereitgestellt, um \emph{Familien
751     verbundener oder abhängiger Objekte} zu erstellen, ohne die konkreten
752     Klassen zu spezifizieren.\footcite[Seite 25]{eilebrecht}
753 }
```



```

754 \def\liEntwurfsAbstrakteFabrikUml{
755   \begin{tikzpicture}
756     \umlclass[type=abstract]{AbstrakteFabrik}{-}{-}{
757       erzeugeProduktA()\n
758       erzeugeProduktB()\n
759     }
760     \umlclass[below left=1cm and -1.5cm of AbstrakteFabrik]{Fabrik1}{-}{-}{
761       erzeugeProduktA()\n
762       erzeugeProduktB()\n
763     }
764     \umlclass[below right=1cm and -1.5cm of AbstrakteFabrik]{Fabrik2}{-}{-}{
765       erzeugeProduktA()\n
766       erzeugeProduktB()\n
767     }
768     \umlVHVinherit{Fabrik1}{AbstrakteFabrik}
769     \umlVHVinherit{Fabrik2}{AbstrakteFabrik}
770
771     \umlsimpleclass[right=3cm of AbstrakteFabrik,type=abstract]{AbstraktesProduktA}
772     \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktA]{ProduktA1}
773     \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktA]{ProduktA2}
774     \umlVHVinherit{ProduktA1}{AbstraktesProduktA}
775     \umlVHVinherit{ProduktA2}{AbstraktesProduktA}
776
777     \umlsimpleclass[above right=0cm and 1cm of AbstraktesProduktA]{Klient}
778
779     \umlsimpleclass[below=4.5cm of AbstraktesProduktA,type=abstract]{AbstraktesProduktB}
780     \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktB]{ProduktB1}
781     \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktB]{ProduktB2}
782     \umlVHVinherit{ProduktB1}{AbstraktesProduktB}
783     \umlVHVinherit{ProduktB2}{AbstraktesProduktB}
784
785     \umlVHVdep[arm1=-1.3cm,arm2=-1cm]{Fabrik1}{ProduktA1}
786     \umlVHVdep[arm2=-1cm]{Fabrik1}{ProduktB1}
787
788     \umlVHVdep[arm1=-1.8cm,arm2=-1.5cm]{Fabrik2}{ProduktA2}
789     \umlVHVdep[arm2=-1.5cm]{Fabrik2}{ProduktB2}
790
  
```

```

791 \umluniassoc[anchor2=25]{Klient}{AbstrakteFabrik}
792 \umlVHuniassoc[arm1=-1cm]{Klient}{AbstraktesProduktA}
793 \umlVHuniassoc{Klient}{AbstraktesProduktB}
794 \end{tikzpicture}
795 }

```

iEntwurfsAbstrakteFabrikCode

```

796 \def\liEntwurfsAbstrakteFabrikCode{
797 \li@EntwurfsCode{abstrakte_fabrik}{Produkte}
798 \li@EntwurfsCode{abstrakte_fabrik}{AbstrakteFabrik}
799 \li@EntwurfsCode{abstrakte_fabrik}{Klient}
800 }

```

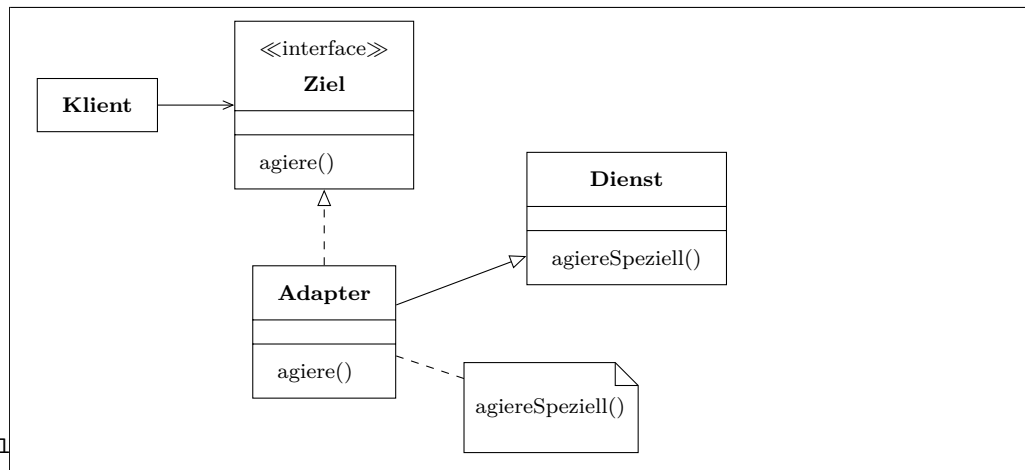
\liEntwurfsAbstrakteFabrik

```

801 \def\liEntwurfsAbstrakteFabrik{
802 \liEntwurfsAbstrakteFabrikBeschreibung
803
804 \liEntwurfsAbstrakteFabrikUml
805
806 \liEntwurfsAbstrakteFabrikCode
807 }

```

2.11.4 Adapter



\liEntwurfsAdapterUml

```

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

```

\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.

```

823 \def\liEntwurfsAdapterAkteure{
824   \begin{description}
825
826     \item[Ziel (Target)]
827
828     Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.
829
830     \item[Klient (Client)]
831
832     Der Klient nutzt Dienste über inkompatible Schnittstellen und greift
833     dabei auf adaptierte Schnittstellen zurück.
834
835     \item[Dienst (Adaptee)]
836
837     Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest
838     definierter Schnittstelle an.
839
840     \item[Adapter]
841
842     Der Adapter adaptiert die Schnittstelle des Dienstes auf die
843     Schnittstelle zum Klienten.\footcite{wiki:adapter}
844
845   \end{description}
846 }

```

\liEntwurfsAdapterCode

```

847 \def\liEntwurfsAdapterCode{
848   \li@EntwurfsCode{adapter}{Dienst}
849   \li@EntwurfsCode{adapter}{Ziel}
850   \li@EntwurfsCode{adapter}{Adapter}
851   \li@EntwurfsCode{adapter}{Klient}
852 }

```

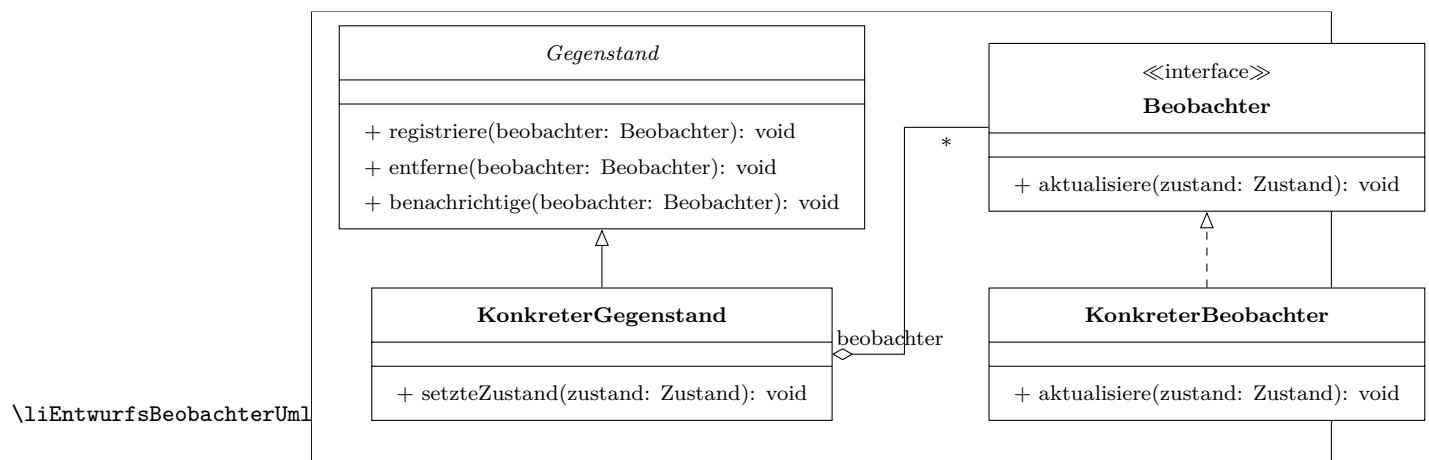
\liEntwurfsAdapter

```

853 \def\liEntwurfsAdapter{
854   \liEntwurfsAdapterUml
855   \liEntwurfsAdapterAkteure
856   \liEntwurfsAdapterCode
857 }

```

2.11.5 Beobachter (Observer)



```

858 \def\liEntwurfsBeobachterUml{
859   \begin{tikzpicture}
860     \umlclass[x=0,y=0,type=abstract]{Gegenstand}{}{

```

```

861     + registriere(beobachter: Beobachter): void\\
862     + entferne(beobachter: Beobachter): void\\
863     + benachrichtige(beobachter: Beobachter): void\\
864 }
865 \umlclass[x=0,y=-3]{KonkreterGegenstand}{-}{
866     + setzeZustand(zustand: Zustand): void
867 }
868 \umlinherit{KonkreterGegenstand}{Gegenstand}
869
870 \umlclass[x=8,y=0,type=interface]{Beobachter}{-}{
871     + aktualisiere(zustand: Zustand): void
872 }
873 \umlclass[x=8,y=-3]{KonkreterBeobachter}{-}{
874     + aktualisiere(zustand: Zustand): void
875 }
876 \umlreal{KonkreterBeobachter}{Beobachter}
877
878 \umlHVHaggreg[arg1=beobachter,pos1=0.8,mult2=*,pos2=2.5]
879 {KonkreterGegenstand}{Beobachter}
880 \end{tikzpicture}
881 }

```

\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.

```

882 \def\liEntwurfsBeobachterAkteure{
883   \begin{description}
884     \item[Gegenstand / Subjekt (Subject / Observable)]
885
886     Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also
887     „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne
888     deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An-
889     und Abmeldung von Beobachtern und eine Schnittstelle zur
890     Benachrichtigung von Beobachtern über Änderungen an.\footcite[Seite
891     251]{gof}
892
893     \item[Beobachter (Observer)]
894
895     Die Beobachter (auf Englisch auch subscriber, also „Abonnent“,
896     genannt) definieren eine Aktualisierungsschnittstelle.
897
898     \item[konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)]
899
900     Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert
901     den relevanten Zustand und benachrichtigt alle Beobachter bei

```



```

902 Zustandsänderungen über deren Aktualisierungsschnittstelle. Es
903 verfügt über eine Schnittstelle zur Erfragung des aktuellen
904 Zustands.
905
906 \item[Konkrete Beobachter (ConcreteObserver)]
907
908 Konkrete Beobachter verwalten die Referenz auf ein konkretes
909 Subjekt, dessen Zustand sie beobachten und speichern und dessen
910 Zustand konsistent ist. Sie implementieren eine
911 Aktualisierungsschnittstelle unter Verwendung der
912 Abfrageschnittstelle des konkreten Subjekts.
913 \footcite{wiki:beobachter}
914 \end{description}
915 }

```

\liEntwurfsBeobachterCode

```

916 \def\liEntwurfsBeobachterCode{
917 \li@EntwurfsCode{beobachter}{Gegenstand}
918 \li@EntwurfsCode{beobachter}{KonkreterGegenstand}
919 \li@EntwurfsCode{beobachter}{Beobachter}
920 \li@EntwurfsCode{beobachter}{KonkreterBeobachterA}
921 \li@EntwurfsCode{beobachter}{KonkreterBeobachterB}
922 \li@EntwurfsCode{beobachter}{Klient}
923 }

```

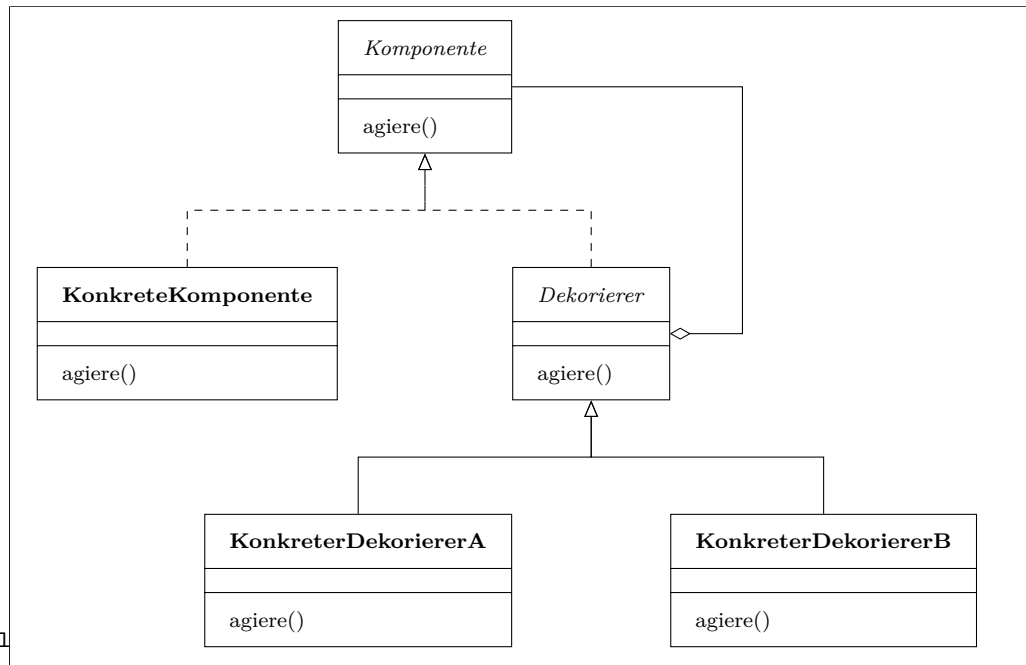
\liEntwurfsBeobachter

```

924 \def\liEntwurfsBeobachter{
925 \liEntwurfsBeobachterUml
926 \liEntwurfsBeobachterAkteure
927 \liEntwurfsBeobachterCode
928 }

```

2.11.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

929 \def\liEntwurfsDekoriererUml{
930 \begin{tikzpicture}
931 \umlclass[type=abstract]{Komponente}{}{agiere()}
932 \umlclass[below left=1.5cm and 0cm of Komponente]{KonkreteKomponente}{}{agiere()}
933 \umlclass[below right=1.5cm and 0cm of Komponente,type=abstract]{Dekorierer}{}{agiere()}
934

```

```

935 \umlVHVreal{KonkreteKomponente}{Komponente}
936 \umlVHVreal{Dekorierer}{Komponente}
937
938 \umlclass[below left=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererA}{-}{agiere()}
939 \umlclass[below right=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererB}{-}{agiere()}
940
941 \umlVHVinherit{KonkreterDekoriererA}{Dekorierer}
942 \umlVHVinherit{KonkreterDekoriererB}{Dekorierer}
943
944 \umlHVHaggreg[arm1=2cm]{Dekorierer}{Komponente}
945 \footcite{wiki:dekorierer}
946 \end{tikzpicture}
947 }

```

\liEntwurfsDekoriererCode

```

948 \def\liEntwurfsDekoriererCode{
949 \li@EntwurfsCode{dekorierer}{Komponente}
950 \li@EntwurfsCode{dekorierer}{KonkreteKomponente}
951 \li@EntwurfsCode{dekorierer}{Dekorierer}
952 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererA}
953 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererB}
954 \li@EntwurfsCode{dekorierer}{Klient}
955 }

```

\liEntwurfsDekorierer

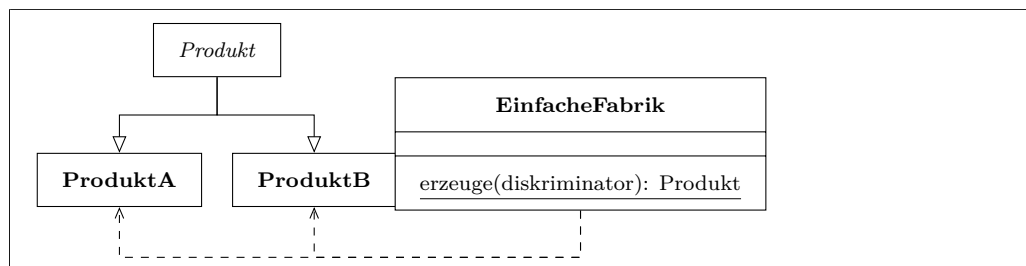
```

956 \def\liEntwurfsDekorierer{
957 \liEntwurfsDekoriererUml
958 \liEntwurfsDekoriererAkteure
959 \liEntwurfsDekoriererCode
960 }

```

2.11.7 Einfache Fabrik (Simple Factory)

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



```

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

```

ntwurfsEinfacheFabrikAkteure Quelle: frei überstetzt aus GoF

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

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

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

```

976 \def\liEntwurfsEinfacheFabrikAkteure{
977   \begin{description}
978     \item[EinfacheFabrik]
979
980     Eine Klasse mit einer Erzeugungsmethode, die über eine größere
981     Bedingung verschiedene Objekt instanziert.
982
983     \item[Produkt]
984
985     Eine abstrakte Klasse, die von den konkreten Produkten geerbt wird.
986
987     \item[KonkretesProdukt]
988
989     Ein konkretes Produkt, das von der einfachen Fabrik erzeugt wird.
990   \end{description}
991 }
```

\liEntwurfsEinfacheFabrik

```

992 \def\liEntwurfsEinfacheFabrik{
993   \liEntwurfsEinfacheFabrikUml
994   \liEntwurfsEinfacheFabrikAkteure
995 }
```

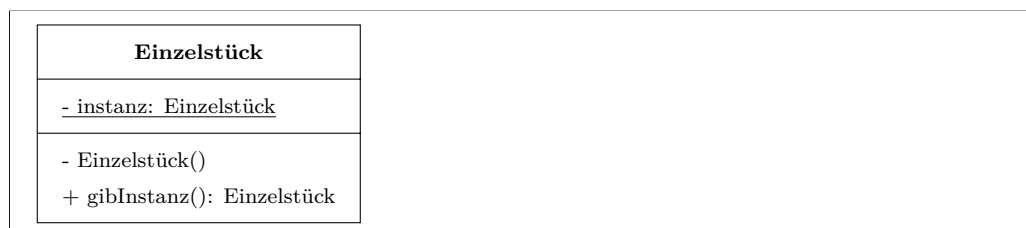
2.11.8 Einzelstück (Singleton)

urfsEinzelstueckBeschreibung

```

996 \def\liEntwurfsEinzelstueckBeschreibung{
997   Stellt sicher, dass nur \emph{genau eine Instanz einer Klasse} erzeugt
998   wird.\footcite[Seite 38]{eilebrecht}
999 }
1000
```

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```

1001 \def\liEntwurfsEinzelstueckUml{
1002   \begin{tikzpicture}
1003     \umlclass{Einzelstück}{
1004       \umlstatic{- instanz: Einzelstück}\\
1005     }{
1006       - Einzelstück()\\
1007       + gibInstanz(): Einzelstück
1008     }
1009   \end{tikzpicture}
1010 }
```

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.

```

1011 \def\liEntwurfsEinzelstueckAkteure{
1012   \begin{description}
1013     \item[Einzelstück (Singleton)]
1014
1015     stellt eine statische Methode bereit, mit deren Hilfe die Klienten
1016     nur auf eine einzige Instanz der Klasse zugreifen können.
1017   \end{description}
1018 }

```

\liEntwurfsEinzelstueckCode

```

1019 \def\liEntwurfsEinzelstueckCode{
1020   \li@EntwurfsCode{einzelstueck}{Einzelstueck}
1021 }

```

\liEntwurfsEinzelstueck

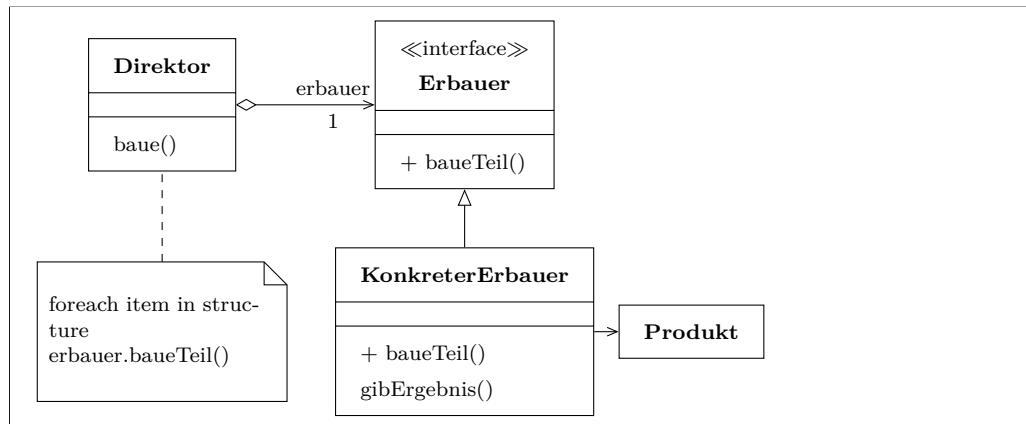
```

1022 \def\liEntwurfsEinzelstueck{
1023   \liEntwurfsEinzelstueckBeschreibung
1024
1025   \liEntwurfsEinzelstueckUml
1026
1027   \liEntwurfsEinzelstueckAkteure
1028
1029   \liEntwurfsEinzelstueckCode
1030 }

```

2.11.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

1031 \def\liEntwurfsErbauerUml{
1032   \begin{tikzpicture}
1033     \umlsimpleclass[x=7,y=0]{Produkt}{}{}
1034     \umlclass[x=4,y=3,type=interface]{Erbauer}{}{+ baueTeil()}
1035     \umlclass[x=4,y=0]{KonkreterErbauer}{}{
1036       + baueTeil()\n
1037       gibErgebnis()}
1038     \umlclass[x=0,y=3]{Direktor}{}{baue()}
1039
1040     \umluniagg[arg2=erbauer,pos2=0.7,mult2=1]{Direktor}{Erbauer}
1041     \umluniassoc{KonkreterErbauer}{Produkt}
1042     \umlinherit{KonkreterErbauer}{Erbauer}
1043
1044     \umlnote[x=0,y=0,width=3cm]{Direktor}{
1045       foreach item in structure\n
1046       erbauer.baueTeil()

```

```

1047 }
1048 \end{tikzpicture}
1049 \footcite{wiki:erbauer}
1050 }

```

\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.

```

1051 \def\liEntwurfsErbauerAkteure{
1052   \begin{description}
1053     \item[Erbauer]
1054
1055     Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der
1056     Teile eines komplexen Objektes.
1057
1058     \item[KonkreterErbauer]
1059
1060     Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch
1061     Implementierung der Schnittstelle. Außerdem definiert und verwaltet er
1062     die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine
1063     Schnittstelle zum Auslesen des Produkts.
1064
1065     \item[Direktor]
1066
1067     Der Direktor konstruiert ein komplexes Objekt unter Verwendung der
1068     Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer
1069     zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder
1070     benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom
1071     Klienten.
1072
1073     \item[Produkt]
1074
1075     Das Produkt repräsentiert das zu konstruierende komplexe Objekt.
1076     \footcite{wiki:erbauer}
1077   \end{description}
1078 }

```

\liEntwurfsErbauer

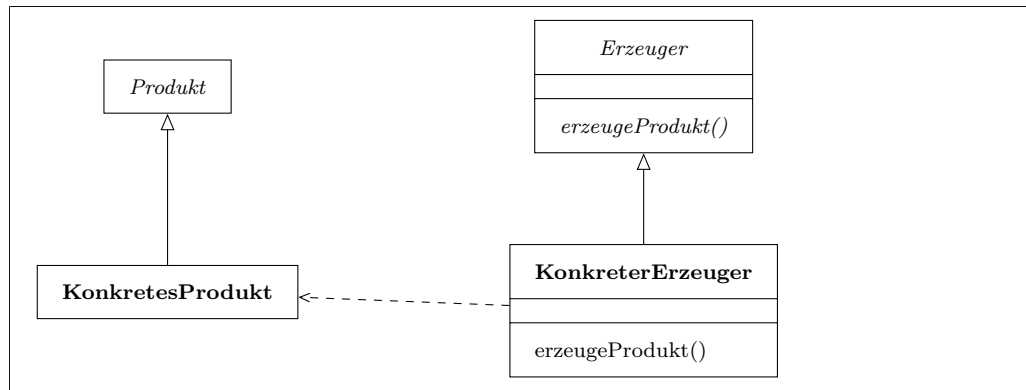
```

1079 \def\liEntwurfsErbauer{
1080   \liEntwurfsErbauerUml
1081   \liEntwurfsErbauerAkteure
1082 }

```

2.11.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

1083 \def\liEntwurfsFabrikmethodeUml{
1084   \begin{tikzpicture}
1085     \umlsimpleclass[type=abstract]{Produkt}
1086     \umlsimpleclass[below=2cm of Produkt]{KonkretesProdukt}
1087     \umlinherit{KonkretesProdukt}{Produkt}
1088
1089     \umlclass[type=abstract,right=4cm of Produkt]{Erzeuger}{\{
1090       \textit{erzeugeProdukt()}\}
1091   }
1092   \umlclass[below=1.2cm of Erzeuger]{KonkreterErzeuger}{\{
1093     erzeugeProdukt()
1094   }
1095   \umlinherit{KonkreterErzeuger}{Erzeuger}
1096
1097   \umldep{KonkreterErzeuger}{KonkretesProdukt}
1098 \end{tikzpicture}
1099 }

```

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).

```

1100 \def\liEntwurfsFabrikmethodeAkteure{
1101   \begin{description}
1102     \item[Produkt]
1103
1104     Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das
1105     zu erzeugende Produkt.
1106
1107     \item[KonkretesProdukt]
1108
1109     KonkretesProdukt implementiert die Produkt-Schnittstelle.
1110
1111     \item[Erzeuger]
1112
1113     Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt
1114     zu erzeugen und kann eine Default-Implementierung beinhalten.
1115
1116     \item[KonkreterErzeuger]
1117

```

```

1118     KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm
1119     entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den
1120     Konstruktor einer konkreten Produkt-Klasse aufruft).
1121
1122     \footcite{wiki:fabrikmethode}
1123 \end{description}
1124 }

```

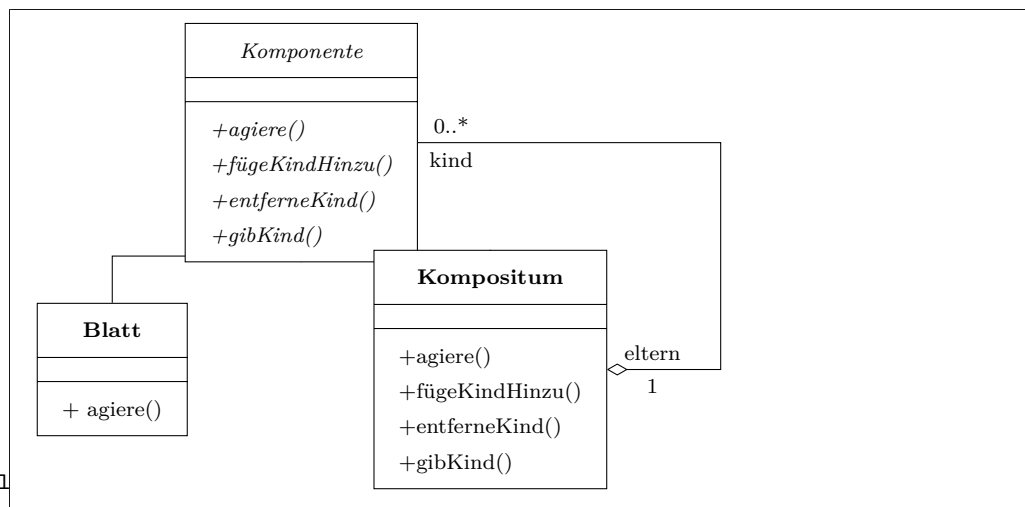
\liEntwurfsFabrikmethode

```

1125 \def\liEntwurfsFabrikmethode{
1126   \liEntwurfsFabrikmethodeUml
1127   \liEntwurfsFabrikmethodeAkteure
1128 }

```

2.11.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

1129 \def\liEntwurfsKompositumUml{
1130   \begin{tikzpicture}
1131     \umlclass[x=2.5,y=3,type=abstract]{Komponente}{-}{
1132       \textit{+agiere()}\
1133       \textit{+fügeKindHinzu()}\
1134       \textit{+entferneKind()}\
1135       \textit{+gibKind()}
1136     }
1137     \umlclass[x=0]{Blatt}{-}{+ agiere()}
1138     \umlclass[x=5]{Kompositum}{-}{
1139       +agiere()\
1140       +fügeKindHinzu()\
1141       +entferneKind()\
1142       +gibKind()
1143     }
1144
1145     \umlVHVinherit{Kompositum}{Komponente}
1146     \umlVHVinherit{Blatt}{Komponente}
1147     \umlHVVHaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,]
1148   \end{tikzpicture}
1149 }

```

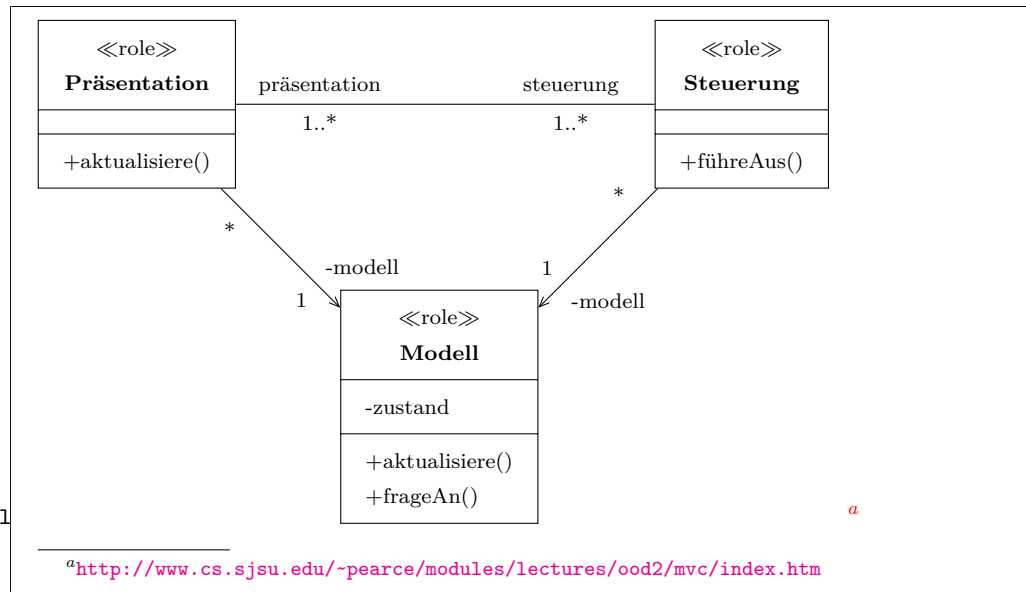
\liEntwurfsFabrikmethode

```

1150 \def\liEntwurfsKompositum{
1151   \liEntwurfsKompositumUml
1152   \liEntwurfsKompositumAkteure
1153 }

```

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



```

1154 \def\liEntwurfsModellPraesentationSteuerungUml{
1155   \begin{tikzpicture}
1156     \umlclass[x=-4cm,type=role]{Präsentation}{+aktualisiere()}
1157     \umlclass[x=4cm,type=role]{Steuerung}{+führeAus()}
1158     \umlclass[y=-4cm,type=role]{Modell}{
1159       -zustand
1160     }{
1161       +aktualisiere()\
1162       +frageAn()
1163     }
1164
1165     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
1166     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}
1167     \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
1168   \end{tikzpicture}
1169   \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
1170 }

```

ModellPraesentationSteuerung

```

1171 \def\liEntwurfs{
1172   \liEntwurfsModellPraesentationSteuerungUml
1173   \liEntwurfsModellPraesentationSteuerungAkteure
1174 }

```

2.11.13 Stellvertreter (Proxy)

\liEntwurfsStellvertreterUml

```

1175 \def\liEntwurfsStellvertreterUml{
1176   \begin{tikzpicture}
1177     \umlsimpleclass[x=-1,y=2]{Klient}
1178
1179     \umlclass[x=2,y=2]{Subjekt}{+ agiere()}
1180     \umlclass[x=0,y=-1]{KonkretesSubjekt}{+ agiere()}
1181     \umlclass[x=4,y=-1]{Stellvertreter}{+ agiere()}
1182
1183     \umlVHVinherit{KonkretesSubjekt}{Subjekt}
1184     \umlVHVinherit{Stellvertreter}{Subjekt}
1185     \umluniassoc{Stellvertreter}{KonkretesSubjekt}
1186     \umluniassoc{Klient}{Subjekt}
1187   \end{tikzpicture}
1188 }

```

\liEntwurfsStellvertreterCode


```

1189 \def\liEntwurfsStellvertreterCode{
1190   \li@EntwurfsCode{stellvertreter}{Subjekt}
1191   \li@EntwurfsCode{stellvertreter}{KonkretesSubjekt}
1192   \li@EntwurfsCode{stellvertreter}{Stellvertreter}
1193   \li@EntwurfsCode{stellvertreter}{Klient}
1194 }

```

\liEntwurfsStellvertreter

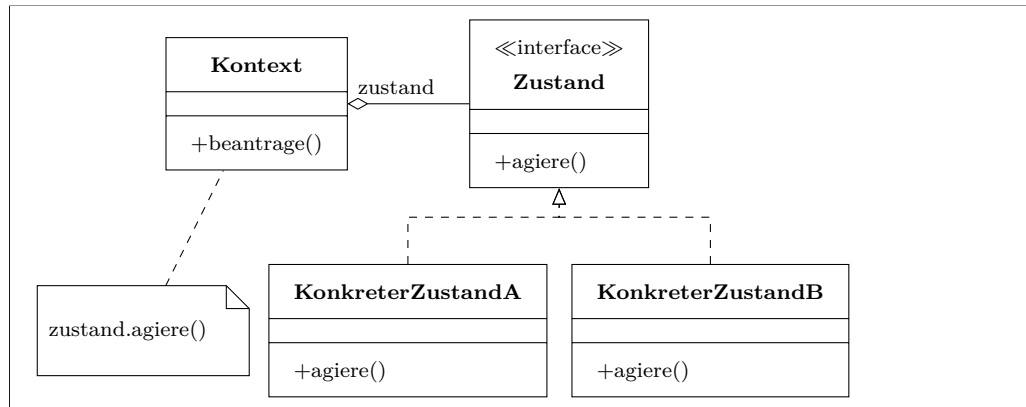
```

1195 \def\liEntwurfsStellvertreter{
1196   \liEntwurfsStellvertreterUml
1197   \liEntwurfsStellvertreterCode
1198 }

```

2.11.14 Zustand (State)

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



```

1199 \def\liEntwurfsZustandUml{
1200   \begin{tikzpicture}
1201     \umlclass[x=-1,y=3]{Kontext}{+beantrage()}
1202     \umlclass[x=3,y=3,type=interface]{Zustand}{+agiere()}
1203     \umlclass[x=1,y=0]{KonkreterZustandA}{+agiere()}
1204     \umlclass[x=5,y=0]{KonkreterZustandB}{+agiere()}
1205
1206     \umlVHVreal{KonkreterZustandA}{Zustand}
1207     \umlVHVreal{KonkreterZustandB}{Zustand}
1208
1209     \umlagg[arg=zustand,pos=0.4]{Kontext}{Zustand}
1210
1211     \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
1212   \end{tikzpicture}
1213 }

```

\liEntwurfsZustandAkteure Quelle: Deutsche Wikipedia

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

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

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

```

1214 \def\liEntwurfsZustandAkteure{
1215   \begin{description}
1216     \item[Kontext (Context)]
1217
1218     definiert die clientseitige Schnittstelle und verwaltet die separaten

```

```

1219     Zustandsklassen.
1220
1221     \item[State (Zustand)]
1222
1223     definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
1224     implementiert gegebenenfalls ein Standardverhalten.
1225
1226     \item[KontreterZustand (ConcreteState)]
1227
1228     implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
1229     verbunden ist.
1230 \end{description}
1231 }

```

\liEntwurfsZustandCode

```

1232 \def\liEntwurfsZustandCode{
1233   \li@EntwurfsCode{zustand}{Kontext}
1234   \li@EntwurfsCode{zustand}{Zustand}
1235 }

```

\liEntwurfsZustand

```

1236 \def\liEntwurfsZustand{
1237   \liEntwurfsZustandUml
1238   \liEntwurfsZustandAkteure
1239   \liEntwurfsZustandCode
1240 }

```

```

1241

```

2.12 er.sty

```
1242 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1243 \ProvidesPackage{lehramt-informatik-er}[2020/06/13 Zum Setzen von
1244 ER-Diagrammen]
```

```
1245 \RequirePackage{tikz-er2}
1246 \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);

```

```
1247 \RequirePackage{soul}
```

```
1248 \RequirePackage{fontawesome}
```

2.12.3 Makro-Kürzel

```

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

```

```
1249 \ExplSyntaxOn
```

```
\liErEntity
```

```
1250 \def\liErEntity#1{\textbf{#1}}
```

```
\liErRelationship
```

```
1251 \def\liErRelationship#1{\ul{#1}}
```

```
\liErAttribute
```

```
1252 \def\liErAttribute#1{\emph{#1}}
```

```
\liErMpEntity mp = marginpar
```

Let-Abkürzung: \let\e=\liErMpEntity

```

1253 \def\liErMpEntity#1{
1254   \liErEntity{#1}
1255   \marginpar{
1256     \liErEntity{\tiny\faSquareO{~E:~#1}
1257   }
1258 }

```

□

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

```

1259 \def\liErMpRelationship#1{
1260   \liErRelationship{#1}
1261   \marginpar{
1262     \liErRelationship{\tiny\faGg{~R:~#1}
1263   }
1264 }

```

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

```

1265 \def\liErMpAttribute#1{
1266   \liErAttribute{#1}
1267   \marginpar{
1268     \liErAttribute{\tiny\faCircleThin{~A:~#1}
1269   }
1270 }

```

```

\liErDatenbankName Let-Abkürzung: \let\d=\liErDatenbankName
                    datenbank name
1271 \def\liErDatenbankName#1{
1272   {
1273     \footnotesize\texttt{(#1)}
1274   }
1275 }

1276 \ExplSyntaxOff
1277

```

2.13 formale-sprachen.sty

```

1278 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1279 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
1280 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]

1281 \directlua{
1282   formale_sprachen = require('lehramt-informatik-formale-sprachen')
1283 }

1284 \RequirePackage{hyperref}

1285 \liLadePakete{mathe,typographie}

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

1286 \def\liMengeOhneMathe#1{\{ #1 \}}
1287 \def\liMenge#1{%
1288   \ifmode%
1289     \liMengeOhneMathe{#1}%
1290   \else%
1291     $\liMengeOhneMathe{#1}$%
1292   \fi%
1293 }

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

1294 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

1295 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1296 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1297 \def\liPotenzmenge#1{$\liPotenzmengeOhneMathe{#1}$}

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

1298 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
1299 \let\liZustandsmenge=\liPotenzmenge

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

1300 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1301 \def\liUeberfuehrungsFunktion#1{
1302   \ifmode
1303     \liUeberfuehrungsFunktionOhneMathe{#1}
1304   \else
1305     $\liUeberfuehrungsFunktionOhneMathe{#1}$
1306   \fi
1307 }

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

1308 \def\liAlphabet#1{$\Sigma = \{ #1 \}$}

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

1309 \def\liBandAlphabet#1{$\Gamma = \Sigma \cup \{ #1 \}$}

\liZustandsBuchstabe

1310 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1311 \def\liZustandsBuchstabeGross{Z}

```

\liZustandsmengeNr

```

1312 \def\zustandsnamens@liste#1#2{
1313   $
1314   \{
1315     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1316   \}
1317   $
1318 }
1319 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

```

\liZustandsmengeNrGross

```

1320 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

```

\liZustandsname \liZustandsname{1}: \$z_1\$

```

1321 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

```

\liZustandsnameGross \liZustandsnameGross{1}: \$Z_1\$

```

1322 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

```

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

```

1323 \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}

```

1324 \NewDocumentEnvironment { liProduktionsRegeln }

```

```

1325 { 0{P} +b }

```

```

1326 {

```

```

1327   \liGeschweifteKlammern{#1}

```

```

1328   {

```

```

1329     \begin{align*}

```

```

1330       \directlua{formale_sprachen.produktions_regeln('#2')}

```

```

1331     \end{align*}

```

```

1332   }{-0.2cm}{-1.5cm}

```

```

1333 } {}

```

\liProduktionen \liProduktionen{S -> A, A -> a}: $\{S \rightarrow A, A \rightarrow a\}$

```

1334 \def\liProduktionen#1{

```

```

1335   \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}

```

```

1336 }

```

\liZustandsnameTiefgestellt Automatisch tiefgestellte Nummerierung \z1

Let-Abkürzung: \let\z=\liZustandsnameTiefgestellt

```

1337 \def\liZustandsnameTiefgestellt#1{

```

```

1338   \ifmmode

```

```

1339     \liZustandsBuchstabe\sb{#1}

```

```

1340   \else

```

```

1341     $\liZustandsBuchstabe\sb{#1}$

```

```

1342   \fi

```

```

1343 }

```

```

1344 \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}

```

```

1345 \NewDocumentCommand{ \liAusdruck } { O{L} m m } {
1346   $
1347   \tl_if_empty:nTF {#1} {} {#1 =}
1348   \{
1349     \, #2 \,
1350     |
1351     \, #3 \,
1352   \}$
1353 }
1354 \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

1355 \def\liFlaci#1{%
1356   \par
1357   {%
1358     \scriptsize
1359     Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1360     Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1361     Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1362     \href{https://flaci.com/#1}{flaci.com/#1}%
1363   }%
1364   \par
1365 }

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



1366 \ExplSyntaxOn
1367 \NewDocumentCommand {\liGrammatik} { O{G} m } {
1368   \tl_set:Nn \l_variablen_tl {V}
1369   \tl_set:Nn \l_alphabet_tl {\Sigma}
1370   \tl_set:Nn \l_produktionen_tl {P}
1371   \tl_set:Nn \l_start_tl {S}
1372
1373   \keys_define:nn { grammatik } {
1374     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1375     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1376     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1377     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1378   }
1379
1380   \keys_set:nn { grammatik } { #2 }
1381
1382   $#1 = (
1383     \l_variablen_tl,
1384     \l_alphabet_tl,
1385     \l_produktionen_tl,
1386     \l_start_tl
1387   )$
1388 }
1389 \ExplSyntaxOff
1390

```


2.14 formatierung.sty

```
1391 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1392 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

2.14.1 Schriftarten / Typographie

```
1393 \RequirePackage{mathpazo}
1394 \RequirePackage[no-math]{fontspec}
1395 \setmainfont{texgyrepagella}
1396 \setsansfont{QTAncientOlive}
1397 \RequirePackage{sectsty}
1398 \allsectionsfont{\fontspec{QTAncientOlive}}
```

2.14.2 Farben

```
1399 \RequirePackage{xcolor}
1400 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

2.14.3 Überschriften

```
1401 \RequirePackage{titlesec}
1402 \titleformat{\chapter}[display]{\sffamily\bfseries}{\Opt}{\LARGE}
1403 \titlespacing{\chapter}{0pt}{0pt}{*1}
1404 \titleformat{\paragraph}[hang]{\sffamily\normalsize\bfseries}{\theparagraph}{1em}{}
1405 \setcounter{secnumdepth}{0}
```

2.14.4 Listen

```
1406 \RequirePackage{paralist}
1407 \renewcommand\labelitemi{-}
1408 \renewcommand\labelitemii{-}
1409 \renewcommand\labelitemiii{-}
1410 \renewcommand\labelitemiv{-}
1411 % Counter: enumi enumii enumiii enumiv
1412 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1413 \renewcommand{\labelenumi}{(\alph{enumi})}
1414 \renewcommand{\labelenumii}{(\roman{enumii})}
```

2.14.5 Kasten

```
1415 \RequirePackage{mdframed}
1416 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1417 \NewDocumentEnvironment { liKasten }{ } {
1418   \begin{mdframed}[backgroundcolor=white!0]
1419 } {
1420   \end{mdframed}
1421 }
```

2.14.6 Header

```
1422 \RequirePackage{fancyhdr}
1423 \fancyhead[L,C,R]{}
1424 \fancyfoot[L]{}
1425 \fancyfoot[C]{}
1426 \fancyfoot[R]{\thepage}
1427 \pagestyle{fancy}
1428 \renewcommand{\headrulewidth}{0pt}
1429 \renewcommand{\footrulewidth}{0pt}
1430
```

2.15 gantt.sty

```

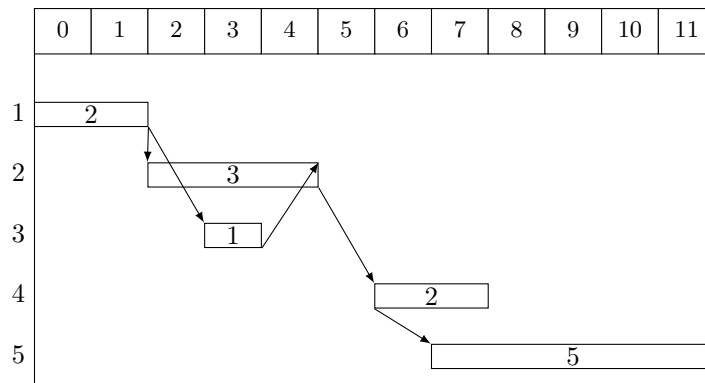
1431 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1432 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\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}

```



```

1433 \RequirePackage{tikz-uml}
1434 \RequirePackage{pgfgantt}
1435 \setganttlinklabel{f-s}{}
1436 \setganttlinklabel{s-s}{}
1437 \setganttlinklabel{f-f}{}
1438 \setganttlinklabel{s-f}{}

1439

```

2.16 grafik.sty

```
1440 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1441 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1442 TikZ, um Grafiken zeichnen zu können.]
1443 \RequirePackage{tikz}
1444
```

2.17 graph.sty

```

1445 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1446 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1447 \RequirePackage{tikz}

```

Für die die Adjazenzliste (xrightarrow)

```

1448 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

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

```

1449 \RequirePackage{blkarray}
1450 \usetikzlibrary{arrows.meta}

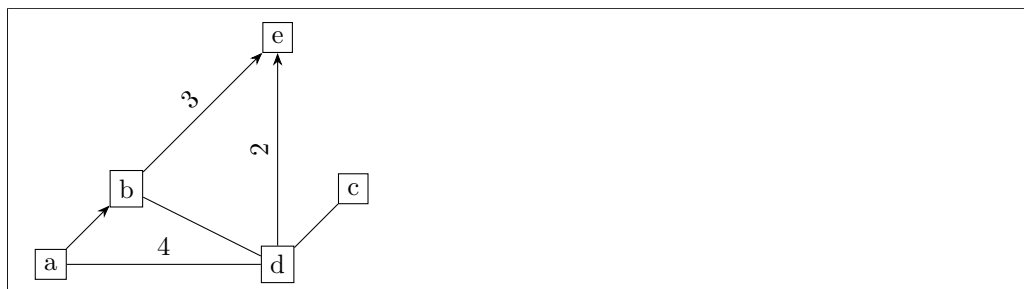
```

```

\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}

```



```

1451 \tikzset{
1452   li graph/.style={
1453     every node/.style={
1454       rectangle,
1455       draw,
1456     },
1457     every edge/.style={
1458       >={Stealth[black]},
1459       draw,
1460     },
1461     every edge/.append style={
1462       every node/.style={
1463         sloped,
1464         auto,
1465       }
1466     }
1467   },
1468   li markierung/.style={
1469     ultra thick,
1470   }
1471 }

```

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}

```

```

1472 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1473

```

2.18 hanoi.sty

1474 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 1475 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
 1476 von Hanoi-Grafiken]

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

```
1477 \RequirePackage{tikz}
1478 \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}

1479 \def\li@mset #1[#2]=#3{%
1480 \expandafter\xdef\csname #1#2\endcsname{#3}
1481 }
1482 \def\li@mget #1[#2]{%
1483 \csname #1#2\endcsname
1484 }
1485 \def\li@minc #1[#2]+=#3{%
1486 \pgfmathparse{\li@mget #1[#2]+#3}%
1487 \li@mset #1[#2]=\pgfmathresult
1488 }
1489
1490 \def\liHanoi#1#2{
1491   \edef\li@numdiscs{#1}
1492   \def\li@sequence{#2}
1493   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1494     % init colors
1495     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purple!80!black}{
1496       \li@mset col[\j]={\c};
1497     }
1498     % draw poles and init pole counters
1499     \foreach \j in {1,2,3}{
1500       \li@mset pos[\j]=0
1501       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1502     }
1503     % draw base
1504     \draw (.5,-.5) -- +(3,0);
1505     % draw discs
1506     \foreach[count=\k] \i/\j in \li@sequence{
1507       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*\i/\li@numdiscs,0);
1508       \li@minc pos[\j]+=.5
1509     }
1510   \end{tikzpicture}
1511 }
```

2.19 klassen-konfiguration-aufgabe.sty

```
1512 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1513 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-aufgabe}[2021/09/01 Die
1514 Klasse Aufgabe konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

Formatierung muss vor literatur sein, sonst Option clash

```
1515 \liLadePakete{
1516   formatierung,
1517   abmessung,
1518   literatur-dummy,
1519   makros,
1520   aufgaben-metadaten,
1521   kopf-fusszeilen,
1522   mathe
1523 }
1524
1525 % Komischer Option-Clash deshalb ganz am Ende, für die Silbentrennung
1526 \RequirePackage[ngerman]{babel}
1527 \ExplSyntaxOn
```

\liAufgabenMetadaten

```
1528 \def\liAufgabenMetadaten#1{
1529   \liMetaSetze{#1}
1530
1531   \noindent
1532   {\large \_gib_einzelpruefung_trenner: \par}
1533
1534   \medskip
1535
1536   \noindent
1537   {\bfseries\Large\_gib_aufgaben_pfad_trenner:}
1538 }
1539 \AddToHook{enddocument}{
1540   \vfill
1541   {
1542     \tiny
1543     \noindent
1544     Hilf~mit!~
1545     Das~ist~ein~Community~Projekt.~
1546     Verbesserungsvorschläge,~Fehlerkorrekturen,~weitere~Lösungen~sind~
1547     sehr~willkommen~~~egal~wie~~~per~Pull~Request~oder~per~E-Mail~an~
1548     \href{mailto:hermine.bsclangaul@gmx.net}{hermine.bsclangaul@gmx.net}~
1549
1550     \par\noindent
1551     Der~\TeX-Quelltext~dieses~PDFs~kann~unter~folgender~URL~aufgerufen~werden:~
1552
1553     \par\noindent\hfill
1554     \_gib_github_url_href:}
1555 }
1556 \ExplSyntaxOff
1557
```

2.20 klassen-konfiguration-examen.sty

```
1558 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1559 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-examen}[2021/09/04 Die
1560 Klasse liSetzeExamen konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

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

```
1561 \liLadePakete{
1562   formatierung,
1563   literatur-dummy,
1564   makros,
1565   aufgaben-metadaten,
1566   abmessung,
1567   typographie
1568 }
```

Formatierung für die Überschriften setzen.

```
1569 \RequirePackage{titlesec}
1570 \titleformat{\section}{\sffamily\huge\filcenter\bfseries}{\thesection}{1em}{}
1571 \newcommand{\sectionbreak}{\clearpage}
1572 \setcounter{secnumdepth}{0}
1573 \liLadeAllePakete
```

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

```
1574 \RequirePackage[ngerman]{babel}
1575 \RequirePackage{standalone}
1576 \ExplSyntaxOn
```

`\liSetzeExamenThemaNr`

```
1577 \def\liSetzeExamenThemaNr#1{
1578   \tl_gset:Nn \g_auf_examen_thema_nr_tl { #1 }
1579   \section{Thema-Nr.~#1}
1580 }
```

`\liSetzeExamenTeilaufgabeNr`

```
1581 \def\liSetzeExamenTeilaufgabeNr#1{
1582   \tl_gset:Nn \g_auf_examen_teilaufgabe_nr_tl { #1 }
1583   \subsection{Teilaufgabe-Nr.~#1}
1584 }
```

`\liBindeAufgabeEin`

```
1585 \def\liBindeAufgabeEin#1{
1586   \tl_gset:Nn \g_auf_examen_aufgabe_nr_tl { #1 }
1587   \input{
1588     \LehramtInformatikRepository /
1589     Staatsexamen /
1590     \g_auf_examen_nummer_tl /
1591     \g_auf_examen_jahr_tl /
1592     \g_auf_examen_monat_tl /
1593     \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
1594       Thema - \g_auf_examen_thema_nr_tl /
1595     }
1596     \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
1597       Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl /
1598     }
1599     Aufgabe - \g_auf_examen_aufgabe_nr_tl .tex
1600   }
1601 }
```

`\liAufgabenMetadaten` Das Metadaten-Makro überschreiben

```
1602 \def\liAufgabenMetadaten#1{
1603   \liMetaSetze{#1}
1604   \subsubsection{\_gib_aufgaben_titel:}
1605 }
```



```

1606 \setcounter{tocdepth}{4}
1607 \RequirePackage[titles]{tocloft}
1608 \AddToHook{begindocument}{
1609   \pagestyle{empty}
1610   \begin{center}
1611     \large
1612     Erste-Staatsprüfung-für-ein-Lehramt-an-öffentlichen-Schulen \par
1613
1614     \vspace{0.5cm}
1615
1616     Fach-Informatik \par
1617
1618     \vfill
1619
1620     {
1621       \sffamily\bfseries\Huge
1622
1623       \g_auf_examen_jahreszeit_tl \par
1624
1625       \g_auf_examen_jahr_tl \par
1626     }
1627
1628     \vspace{2cm}
1629
1630     {\LARGE \g_auf_examen_nummer_tl \par}
1631
1632     \vspace{0.5cm}
1633
1634     \g_auf_examen_fach_tl \par
1635
1636     \vspace{3cm}
1637
1638     Aufgabenstellungen-mit-Lösungsvorschlägen \par
1639
1640     \end{center}
1641
1642     % für den Abstand vor den section im Inhaltsverzeichnis
1643     % https://tex.stackexchange.com/questions/241445/how-to-control-spacing-in-toc-for-different-sections
1644     \setlength{\cftbeforesecskip}{1.5cm}
1645     \setlength{\cftbeforesubsecskip}{0.5cm}
1646     \renewcommand{\cftsubsecafterpnum}{\vspace{\cftbeforesubsecskip}}
1647     \setlength{\cftbeforesubsubsecskip}{0.1cm}
1648
1649     \vfill
1650     \renewcommand{\contentsname}{Aufgabenübersicht}
1651     \tableofcontents
1652     \clearpage
1653 }
1654 \ExplSyntaxOff
1655

```

2.21 komplexitaetstheorie.sty

```
1656 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1657 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1658 Setzen von Karp's NP-vollständigen Problemen, Hilfsmakros für die
1659 Polynomialzeitreduktion.]
```

2.21.1 Makro-Kürzel

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

```
1660 \liLadePakete{mathe}
```

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

```
1661 \RequirePackage{mdframed}
```

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

```
1662 \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`

```
1663 \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`

```
1664 \def\liProblemBeschreibung#1#2#3{
1665   \begin{mdframed}[
1666     userdefinedwidth=9cm,
1667     align=center,
1668     backgroundcolor=white!0,
1669   ]
1670   \centerline{\large\liProblemName{#1}}
1671
1672   \medskip
1673
1674   \begin{description}
1675     \item[Gegeben:] #2
1676     \item[Frage:] #3
1677   \end{description}
1678 \end{mdframed}
1679 }
```

```

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

1680 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1681 \begin{displaymath}
1682 \liProblemName{#1}
1683 \preceq_{#2}
1684 \liProblemName{#3}
1685 \end{displaymath}
1686 }

\liProblemVertexCover

1687 \def\liProblemClique{%
1688 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1689 Mindestgröße  $n$  in einem gegebenen Graphen.
1690 \footcite[Seite 76]{theo:fs:4}
1691 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1692 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1693 \footcite{wiki:cliquenproblem}
1694 }

\liProblemVertexCover

1695 \def\liProblemVertexCover{%
1696 %
1697 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1698 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1699 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1700 \footcite{wiki:knotenueberdeckung}
1701
1702 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1703 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1704 mindestens einem Knoten aus  $U$  verbunden ist.
1705 \footcite[Seite 78]{theo:fs:4}%
1706 }

\liProblemSubsetSum Kein Karp-21-Problem

1707 \def\liProblemSubsetSum{%
1708 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1709 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1710 \footcite{wiki:teilsommenproblem}
1711 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1712 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1713 größer als eine gegebene obere Schranke  $c$  ist.
1714 \footcite[Seite 74]{theo:fs:4}
1715 }

\liProblemSubsetSum Kein Karp-21-Problem

1716 \def\liProblemSat{%
1717 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1718 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1719 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1720 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1721 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1722 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1723 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1724 aufgestellt werden.
1725 \footcite[Seite 71]{theo:fs:4}
1726 }

1727

```

2.22 kontrollflussgraph.sty

1728 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1729 \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 = ...}]
```

```

1730 \RequirePackage{tikz}
1731 \usetikzlibrary{positioning}
1732 \tikzset{
1733   li kontrollfluss/.style={
1734     knoten/.style={
1735       circle,
1736       draw
1737     },
1738     usebox/.style={
1739       draw,
1740       rectangle,
1741       font=\scriptsize,
1742       anchor=west,
1743       align=left,
1744     },
1745     bedingung/.style={
1746       midway,
1747       draw=none,
1748       font=\scriptsize
1749     },
1750     knotenbeschriftung/.style={
1751       draw,
1752       rectangle,
1753       midway,
1754       font=\scriptsize
1755     },
1756     wahr/.style={
1757       thick
1758     },
1759     falsch/.style={
1760       dashed
1761     },
1762     every node/.style={
1763       circle,
1764       draw,
1765     },
1766     every edge/.append style={
1767       every node/.style={
1768         draw=none,
1769         bedingung,
1770       }
1771     },
1772     every path/.style={
1773       draw,
1774       ->,
1775     },
1776     every pin/.style={
1777       draw,
1778       dotted,
1779       rectangle,
1780       pin position=right
1781     },
1782     every pin edge/.style={
1783       dotted,
1784       arrows=-,
1785     }
1786   }
1787 }

```

2.22.4 Umgebungen

liKontrollflussgraph

```

1788 \NewDocumentEnvironment { liKontrollflussgraph } { 0{} } {

```

```

1789 \begin{tikzpicture}[
1790     li kontrollfluss,
1791     #1
1792 ]
1793 } {
1794 \end{tikzpicture}
1795 }

```

2.22.5 Makros

`\liAnweisung`

```
1796 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}
```

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

```
1797 \def\liBedingung#1#2{\node[bedingung,#1]{\texttt{#2}}}
```

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

```
1798 \def\liBedingungWahr#1{\node[bedingung,#1]{\texttt{[true]}}}
```

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

```
1799 \def\liBedingungFalsch#1{\node[bedingung,#1]{\texttt{[false]}}}
```

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

```
1800 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}
```

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

```
1801 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw
```

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

```
1802 \ExplSyntaxOn
```

```
1803 \NewDocumentCommand { \liKontrollKnotenPfad } { m }
```

```
1804 {
```

```
1805   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
```

```
1806   \seq_set_map:NNn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
```

```
1807   \seq_use:Nn \l_tmpa_seq { ~~~~ }
```

```
1808 }
```

```
1809 \ExplSyntaxOff
```

```
1810
```

2.23 kopf-fusszeilen.sty

```
1811 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1812 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1813 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1814 \ExplSyntaxOn

1815 \fancyhead{}
1816 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1817 \fancyhead[L0,RE]{\scriptsize\today}
1818 \fancyfoot{}
1819 \fancyfoot[LE,R0]{\thepage}
1820 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1821 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1822 \renewcommand{\headrulewidth}{0.4pt}
1823 \renewcommand{\footrulewidth}{0.4pt}
1824 \setlength{\headwidth}{\textwidth}

1825 \ExplSyntaxOff

1826
```

2.24 literatur-dummy.sty

```
1827 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1828 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1829 \def\literatur{}

\footcite

1830 % \RequirePackage[stable,multiple]{footmisc}
1831 \NewDocumentCommand{ \footcite } { o m }{}

1832
```


2.25 literatur.sty

```
1833 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1834 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1835 \RequirePackage{csquotes}
1836 \RequirePackage[
1837   bibencoding=utf8,
1838   citestyle=authortitle,
1839   backend=biber,
1840 ]{biblatex}
1841 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1842 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1843 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1844 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1845 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1846 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1847 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1848 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1849 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1850 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1851 % To allow footnotes in the heading
1852 \RequirePackage[stable,multiple]{footmisc}

\literatur

1853 \def\literatur{\printbibliography[heading=subbibliography]}

1854
```

2.26 makros.sty

```

1855 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1856 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1857 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1858 anderen Paket passen]
1859 \RequirePackage{hyperref}
1860 \RequirePackage{graphicx}

    Für die Umgebung liQuellen benötigt.
1861 \RequirePackage{paralist}

\inhaltsverzeichnis
1862 \def\inhaltsverzeichnis {
1863     \begin{mdframed}
1864         \begin{group}
1865             \let\clearpage\relax
1866             \tableofcontents
1867         \end{group}
1868     \end{mdframed}
1869 }

\memph \mephm (\marginpar and \emph)
1870 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1871 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1872 \newcommand{\liPseudoUeberschrift}[1]{
1873     \bigskip
1874     \noindent
1875     \textsf{\textbf{#1}}
1876     \noindent
1877 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1878 \newcommand{\liBeschriftung}[1]{
1879     \par
1880     \noindent
1881     \medskip
1882     \textbf{#1}:
1883     \medskip
1884     \noindent
1885 }

\hinweis
1886 \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.
1887 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

liEinbettung
1888 \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.
1889 \RequirePackage{xparse}
1890 \ExplSyntaxOn

```

```

1891 \NewDocumentEnvironment{ liAntwort } { 0{standard} }
1892 {
1893   \str_case:nn {#1} {
1894     {standard} {
1895       \def\beschriftung{}
1896       \mdfsetup{backgroundcolor=gray!1,linecolor=gray}
1897     }
1898     {richtig} {
1899       \def\beschriftung{richtig}
1900       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1901     }
1902     {falsch} {
1903       \def\beschriftung{falsch}
1904       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1905     }
1906     {muster} {
1907       \def\beschriftung{Musterlösung}
1908       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1909     }
1910   }
1911   \ifx\beschriftung\empty\else
1912     \noindent
1913     \textbf{\beschriftung{:}}
1914   \fi
1915   \begin{mdframed}
1916 }
1917 {\end{mdframed}}

```

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

```

1918 \NewDocumentEnvironment{ liAdditum } { o }
1919 {
1920   \begin{mdframed}[backgroundcolor=yellow!5]
1921     \IfNoValueTF {#1}
1922     { \liPseudoUeberschrift{Additum} }
1923     { \liPseudoUeberschrift{Additum:~#1} }
1924   }
1925 {\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.

```

1926 \NewDocumentEnvironment{ liExkurs }{o +b}{
1927   \vspace{0.2cm}%
1928   \begin{mdframed}[
1929     backgroundcolor=white,
1930     bottomline=false,
1931     innermargin=1cm,
1932     leftline=true,
1933     linecolor=black,
1934     linewidth=0.1cm,
1935     outermargin=1cm,
1936     rightline=false,
1937     topline=false,
1938   ]

```

```

1939 \footnotesize
1940 \noindent%
1941 \textbf{Exkurs:~#1}\par%
1942 \noindent%
1943 #2
1944 \end{mdframed}
1945 \vspace{0.2cm}
1946 }{}

```

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

```

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

```

Weiterführende Literatur:

- Quelle 1
- Quelle 2

```

1947 \cs_new:Npn \listen@punkt #1 {\item #1}
1948 \NewDocumentEnvironment { liQuellen }{ +b }
1949 {
1950 \seq_clear_new:N \l_quellen
1951 \seq_set_split:Nnn \l_quellen {\item} {#1}
1952 \seq_remove_all:Nn \l_quellen {}
1953 \begin{mdframed}[backgroundcolor=white!0]
1954 \footnotesize
1955 \noindent
1956 \textsf{\textbf{Weiterführende~Literatur:}}
1957 \medskip
1958 \begin{compactitem}
1959 \seq_map_function:NN \l_quellen {\listen@punkt}
1960 \end{compactitem}
1961 \end{mdframed}
1962 %
1963 \makeatletter
1964 \par\@afterindentfalse\@afterheading
1965 \makeatother
1966 } {}

```

liLernkartei

```

1967 \NewDocumentEnvironment { liLernkartei }{ m +b }
1968 {
1969 \begin{mdframed}
1970 \footnotesize
1971 \noindent%
1972 \textbf{Lernkarteikarte:~#1}\par%
1973 \noindent%
1974 #2
1975 \end{mdframed}
1976 } {}

```

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

```

1977 \NewDocumentEnvironment { liDiagramm }{ m +b }
1978 {
1979 \begin{mdframed}[backgroundcolor=white!0]
1980 \small
1981 \noindent%
1982 \textit{#1}:
1983 \begin{center}

```

```

1984 #2
1985 \medskip
1986 \end{center}
1987 \end{mdframed}
1988 } {}
1989 \ExplSyntaxOff

\liFussnoteUrl \liFussnoteUrl[\langle zusätzlicher-text \rangle]{\langle url \rangle} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
1990 \NewDocumentCommand{\liFussnoteUrl} { o m } {
1991 \footnote{\url{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1992 }
1993

\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.
1994 \NewDocumentCommand{\liFussnoteLink} { o m m } {
1995 \footnote{\href{#3}{#2}\IfNoValueTF{#1}{}{ ( #1 )}}
1996 }

\zB
1997 \def\zB{z.\,B. }

\ZB
1998 \def\ZB{Z.\,B. }

\dh
1999 \def\dh{d.\,h. }

2000

```

2.27 master-theorem.sty

2001 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2002 \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}

2003 \ExplSyntaxOn

2004 \RequirePackage{amsmath}

\liRundeKlammer

2005 \def\liRundeKlammer#1{

2006 \negthinspace \left(#1 \right)

2007 }

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

2008 \def\liThetaOhneMathe#1{

2009 \Theta \liRundeKlammer{#1}

2010 }

2011 \def\liTheta#1{

2012 \ifmmode

2013 \liThetaOhneMathe{#1}

2014 \else

2015 \$\liThetaOhneMathe{#1}\$

2016 \fi

2017 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
2018 \def\liOmegaOhneMathe#1{
2019 \Omega \liRundeKlammer{#1}
2020 }
2021 \def\liOmega#1{
2022 \ifmmode
2023 \liOmegaOhneMathe{#1}
2024 \else
2025 $\liOmegaOhneMathe{#1}$
2026 \fi
2027 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
2028 \def\liOOhneMathe#1{
2029 \mathcal{O} \liRundeKlammer{#1}
2030 }
2031 \def\liO#1{
2032 \ifmmode
2033 \liOOhneMathe{#1}
2034 \else
2035 $\liOOhneMathe{#1}$
2036 \fi
2037 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
2038 \def\liTOhneMathe#1#2{
2039 \tl_if_blank:nTF {#1}
2040 {}
2041 {#1 \cdot }
2042 T
2043 \liRundeKlammer{\frac{n}{#2}}
2044 }
2045 \def\liT#1#2{
2046 \ifmmode
2047 \liTOhneMathe{#1}{#2}
2048 \else
2049 $\liTOhneMathe{#1}{#2}$
2050 \fi
2051 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
2052 \def\liRekursionsGleichung{
2053 $T(n) = \liT{a}{b} + f(n)$
2054 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
2055 \def\liBedingungEins{
2056 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
2057 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
2058 \def\liBedingungZwei{
2059 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
2060 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
2061 \def\liBedingungDrei{
2062 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
2063 }

2064 \ExplSyntaxOff

```

\liMasterVariablen

```

2065 \def\liMasterVariablen{
2066   \begin{displaymath}
2067     T(n) = \liT{a}{b} + f(n)
2068   \end{displaymath}
2069
2070   \begin{itemize}
2071     \item[$a = $]
2072       Anzahl der rekursiven Aufrufe, Anzahl der Unterprobleme in der
2073       Rekursion
2074       ($a \geq 1$).
2075
2076     \item[$\textstyle{\frac{1}{b}} = $]
2077       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
2078       repräsentiert wird, Anteil an der Verkleinerung des Problems ($b > 1$).
2079
2080     \item[$f(n) = $]
2081       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
2082       die Kombination der Teillösungen entstehen. Eine von $T(n)$
2083       unabhängige und nicht negative Funktion.
2084   \end{itemize}
2085   \footcite{wiki:master-theorem}
2086   \footcite[Seite 19-35]{aud:fs:2}
2087 }
```

\liMasterFaelle

```

2088 \def\liMasterFaelle{
2089   \begin{description}
2090     \item[1. Fall:]
2091       $T(n) \in \liTheta{n^{\log\sb{b}a}}$
2092
2093       \hfill falls \liBedingungEins
2094       für $\varepsilon > 0$
2095
2096     \item[2. Fall:]
2097       $T(n) \in \liTheta{n^{\log\sb{b}a} \cdot \log n}$
2098
2099       \hfill falls \liBedingungZwei
2100
2101     \item[3. Fall:]
2102       $T(n) \in \liTheta{f(n)}$
2103
2104       \hfill falls \liBedingungDrei
2105       für $\varepsilon > 0$
2106       und ebenfalls für ein $c$ mit $0 < c < 1$ und alle hinreichend großen $n$
2107       gilt:
2108       $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$
2109   \end{description}
2110 }
```

\liMasterVariablenDeklaration

```

2111 \def\liMasterVariablenDeklaration#1#2#3{
2112   \begin{description}
2113     \item[Allgemeine Rekursionsgleichung:] \strut
2114
2115     \liRekursionsGleichung
2116
2117     \item[Anzahl der rekursiven Aufrufe ($a$):] \strut
2118
2119     #1
2120
2121     \item[Anteil Verkleinerung des Problems ($b$):] \strut
2122   \end{description}
}
```



```

2123     um  $\frac{1}{b}$  also  $b = \frac{1}{a}$ 
2124
2125     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut
2126
2127      $\frac{1}{b}$ 
2128
2129     \item[Ergibt folgende Rekursionsgleichung:] \strut
2130
2131      $T(n) = T(\frac{n}{b}) + \frac{1}{b}$ 
2132 \end{description}
2133 }

\liMasterFallRechnung
2134 \def\liMasterFallRechnung#1#2#3{
2135   \begin{description}
2136     \item[1. Fall:] \liBedingungEins:
2137
2138     #1
2139
2140     \item[2. Fall:] \liBedingungZwei:
2141
2142     #2
2143
2144     \item[3. Fall:] \liBedingungDrei:
2145
2146     #3
2147   \end{description}
2148 }

\liMasterExkurs
2149 \def\liMasterExkurs{
2150   \begin{liExkurs}[Master-Theorem]
2151     \liMasterVariablen
2152
2153     \noindent
2154     Dann gilt:
2155
2156     \liMasterFaelle
2157   \end{liExkurs}
2158 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
2159 \def\liMasterWolframLink#1{
2160   Berechne die Rekursionsgleichung auf WolframAlpha:
2161   \href{https://www.wolframalpha.com/input/?i=#1}{WolframAlpha}
2162 }

2163

```

2.28 mathe.sty

```
2164 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2165 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
2166
2167 % for example \ltimes \rtimes
2168 %\RequirePackage{amssymb}
2169 \RequirePackage{amsmath}
2170
2171 %%
2172 % \mlq \mrq
2173 %%
2174 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
2175 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
2176
```

2.29 minimierung.sty

```

2177 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2178 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
2179 Minimierungsalgorithmus von einem NEA zu einem DEA]

2180 \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

2181 \def\liFussnote#1{\$x_{#1}\$}

2182 \def\li@fussnote@text#1#2{
2183 \liFussnote{#1}
2184 \quad
2185 {\footnotesize #2}
2186 }

\liFussnoteEinsText

2187 \def\liFussnoteEinsText{
2188 \li@fussnote@text{1}
2189 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
2190 }

\liFussnoteZweiText

2191 \def\liFussnoteZweiText{
2192 \li@fussnote@text{2}
2193 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
2194 }

\liFussnoteDreiText

2195 \def\liFussnoteDreiText{
2196 \li@fussnote@text{3}

```

```

2197 {In weiteren Iterationen markierte Zustände.}
2198 }

\liFussnoteVierText
2199 \def\liFussnoteVierText{
2200   \li@fussnote@text{4}
2201   {...}
2202 }

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



2203 \def\liFussnoten{
2204   \bigskip
2205
2206   \noindent
2207   \liFussnoteEinsText
2208
2209   \noindent
2210   \liFussnoteZweiText
2211
2212   \noindent
2213   \liFussnoteDreiText
2214
2215   \noindent
2216   \liFussnoteVierText
2217 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
2218 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
2219 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
2220 \def\liZustandsPaar#1#2{
2221   $(
2222     \liZustandsPaarVariablenName_#1,
2223     \liZustandsPaarVariablenName_#2
2224   )$
2225 }

liUebergangsTabelle
2226 \renewcommand{\arraystretch}{1.4}
2227 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
2228   \liPseudoUeberschrift{Übergangstabelle}
2229   \begin{center}
2230     \begin{tabular}{r|l|l}
2231       \textbf{Zustandspaar} & \textbf{#1} & \textbf{#2} \\ \hline
2232     \end{tabular}
2233   \end{center}
2234 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle: Minimierungstabelle (Table filling)

2236 \ExplSyntaxOn
2237 \def\liUeberschriftDreiecksTabelle{
2238   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
2239 }

```

\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. —————

```

2240 \def\liMinimierungErklaerung{
2241   %\footcite[Seite~19]{koenig}
2242   \liParagraphMitLinien{
2243     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
2244     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
2245     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
2246     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
2247      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
2248     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
2249     somit~zueinander~( $k - 1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
2250     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
2251     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
2252   }
2253 }
2254 \ExplSyntaxOff
2255

```

2.30 normalformen.sty

```
2256 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2257 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10]
2258 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
2259 Attributhülle]
```

Mathe für textit tex etc, typographie für geschweifte Klammer

```
2260 \liLadePakete{mathe,typographie}
2261 \directlua{
2262   helper = require('lehramt-informatik-helfer')
2263   normalformen = require('lehramt-informatik-normalformen')
2264 }
```

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
```

```
2265 \def\liTeilen#1{
2266   \directlua{tex.print(normalformen.teilen('#1'))}
2267 }
```

```
\liAttributHuelle Let-Abkürzung: \let\ah=\liAttributHuelle
                  \ah{F, \m{A, B}} AttrHülle( $F, \{A, B\}$ ) Regulärer Ausdruck zum Konvertieren
                  AttrHülle((.*)) \ah{$1}
2268 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(#1)}
2269 \def\liAttributHuelle#1{
2270   \ifmmode
2271     \liAttributHuelleOhneMathe{#1}
2272   \else
2273     $\liAttributHuelleOhneMathe{#1}$
2274   \fi
2275 }
```

```
\liAttributMenge Let-Abkürzung: \let\m=\liAttributMenge
2276 \def\liAttributMenge#1{\{ \textit{#1} \}}
```

liAHuelle

```
2277 \NewDocumentEnvironment{ liAHuelle } { +b } {
2278   \begingroup
2279   \footnotesize
2280   \begin{multline*}
2281     #1
2282   \end{multline*}
2283   \endgroup
2284 } { }
```

```
\liLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multline
Let-Abkürzung: \let\ahL=\liLinksReduktion
                  \ahL{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}
2285 \def\liLinksReduktion#1#2#3{
2286   \shoveleft{
2287     \liAttributHuelleOhneMathe{FA,
2288       \liAttributMenge{#1 \string\ #2}} =
2289   } \}
```

```

2290 \shoveright{
2291   \liAttributMenge{#3}
2292 } \\\
2293 }

```

```

\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( $F, \{A, E \setminus E\}$ ) =  $\{A, E, F, B, D\}$ 
2294 \def\liLinksReduktionInline#1#2#3{%
2295   {%
2296     \footnotesize%
2297     $\liAttributHuelleOhneMathe{F,
2298       \liAttributMenge{#1 \string\ #2}} =
2299       \liAttributMenge{#3}$
2300   }
2301 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahr=\liLinksReduktionInline
    \ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
2302 \def\liRechtsReduktionInline#1#2#3#4{%
2303   {%
2304     \footnotesize%
2305     $\liAttributHuelleOhneMathe{
2306       F \setminus
2307       \liFunktionaleAbhaengigkeit{#1}
2308       \def\tmp{#2}\ifx\tmp\empty
2309       \else
2310         \cup \liFunktionaleAbhaengigkeit{#2}
2311       \fi
2312       ,
2313       \liAttributMenge{#3}
2314     } =
2315     \liAttributMenge{#4}$
2316   }
2317 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
    \fa{A, B -> C, D}:  $\{A, B\} \rightarrow \{C, D\}$  Regulärer Ausdruck zum Konvertieren:
    \$(.*?) \\\rightarrow (.*)\$ \fa{$1 -> $2}
2318 \def\liFunktionaleAbhaengigkeit#1{%
2319   \directlua{normalformen.drucke_funk_abhaengigkeit('#1', false)}%
2320 }

```

```

\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;

```

```

2321 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
2322   \liGeschweifteKlammern
2323   {#1}
2324   {
2325     \begin{align*}
2326       \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
2327     \end{align*}
2328   }
2329   {-0.5cm}
2330   {-1.7cm}
2331 }

\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}

2332 \NewDocumentCommand {\liRelation} { O{R} m } {
2333   $\directlua{
2334     local name = helper.konvertiere_tiefgestellt('#1')
2335     tex.print(name)
2336   }$(\textit{\,#2\,})
2337 }

2338

```


2.31 o-notation.sty

2339 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2340 \ProvidesPackage{lehramt-informatik-o-notation}[2021/09/08]

2.31.1 Makro-Kürzel

\let\O=\liONotationO

2341 \ExplSyntaxOn

2342 \RequirePackage{amssymb}

Für echte Teilmenge $\mathcal{O} \subsetneq$

2343 \RequirePackage{amssymb}

\liRundeKlammer

2344 \def\liRundeKlammer#1{

2345 \negthinspace \left(#1 \right)

2346 }

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

2347 \cs_new:Npn \o_notation_0:n #1 {

2348 \mathcal{O} \liRundeKlammer{#1}

2349 }

2350 \def\liONotationO#1{

2351 \ifmmode

2352 \o_notation_0:n { #1 }

2353 \else

2354 \$ \o_notation_0:n { #1 } \$

2355 \fi

2356 }

2357

2.32 petri.sty

2358 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2359 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]

2.32.1 Makro-Kürzel

\let\t=\liPetriTransitionsName

\let\tp=\liPetriTransPfeile

\let\k=\liPetriErreichKnotenDrei

2360 \RequirePackage{tikz}

2361 \usetikzlibrary{petri,arrows.meta}

Für die Darstellungsmatrix

2362 \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

```
2363 \def\liPetriSetzeSchluessel{%
2364   \def\TmpTransitionOne{}%
2365   \def\TmpTransitionTwo{}%
2366   \def\TmpTransitionThree{}%
2367   \def\TmpTransitionFour{}%
2368   \def\TmpTransitionFive{}%
2369   \def\TmpTransitionSix{}%
2370   \def\TmpTransitionSeven{}%
2371   \def\TmpTransitionEight{}%
2372   \def\TmpTransitionNine{}%
2373   \def\TmpTransitionTen{}%
2374   \pgfkeys{/petri/.cd,
2375     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2376     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2377     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2378     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
```

```

2379 p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
2380 p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2381 p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2382 p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2383 p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2384 p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2385 t1/.store in=\TmpTransitionOne,t1/.default=activated,
2386 t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2387 t3/.store in=\TmpTransitionThree,t3/.default=activated,
2388 t4/.store in=\TmpTransitionFour,t4/.default=activated,
2389 t5/.store in=\TmpTransitionFive,t5/.default=activated,
2390 t6/.store in=\TmpTransitionSix,t6/.default=activated,
2391 t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2392 t8/.store in=\TmpTransitionEight,t8/.default=activated,
2393 t9/.store in=\TmpTransitionNine,t9/.default=activated,
2394 t10/.store in=\TmpTransitionTen,t10/.default=activated,
2395 scale/.store in=\TmpScale,scale/.default=0.5,
2396 x/.store in=\TmpX,x/.default=5,
2397 y/.store in=\TmpY,y/.default=5,
2398 }%
2399 }

```

```

2400 \tikzset{
2401   li petri/.style={
2402     activated/.style={
2403       very thick
2404     },
2405     inhibitor/.style={
2406       {Circle[open,length=2mm,fill=white]}-
2407     }
2408   }
2409 }

```

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

```

2410 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2411 \def\liPetriTransitionsName#1{
2412   \ifmmode
2413     \liPetriTransitionsNameOhneMathe{#1}
2414   \else
2415     $\liPetriTransitionsNameOhneMathe{#1}$
2416   \fi
2417 }

```

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

```

2418 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2419   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2420 }

```

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

```

2421 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2422 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2423 }

```

2.33 potenzmengen-konstruktion.sty

```
2424 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2425 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2426 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2427 \liLadePakete{formale-sprachen}
2428 \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}
    }
  }
}
```

```
2429 \def\liZustandsMengenSammlung#1#2{
2430   \liZustandsnameGross{#1}
2431   {
2432     \footnotesize
2433     \liPotenzmenge{
2434       \str_case:nn {#1} {#2
2435       }
2436     }
2437 }
```

```
\liZustandsMengenSammlungNr
```

```
2438 \def\liZustandsMengenSammlungNr#1#2{
2439   \liZustandsnameGross{#1}
2440   {
```

```
2441     \footnotesize
2442     \liZustandsmengeNr{
2443         \str_case:nn {#1} #2
2444     }
2445 }
2446 }

2447 \ExplSyntaxOff
2448
```

2.34 pseudo.sty

```

2449 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2450 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen
2451 von Pseudo-Code, Hüll-Paket um algorithm2e]

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

```

Algorithmus 1: Minimaler Spannbaum nach Kruskal

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

```

2452 \RequirePackage[german,boxruled]{algorithm2e}

```

```

2453

```

2.35 pumping-lemma.sty

2454 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2455 \ProvidesPackage{lehramt-informatik-pumping-lemma}[2021/08/11 Enthält die
 2456 Definitionen für das Pumping-Lemma in der Regulären Sprache und
 2457 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2458 \def\liPumpingRegulaer{%
2459   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2460   alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2461    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2462   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2463
2464   \begin{enumerate}
2465     \item  $|v| \geq 1$ 
2466     (Das Wort  $v$  ist nicht leer.)
2467
2468     \item  $|uv| \leq j$ 
2469     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2470
2471     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w$  in  $L$ 
2472     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2473     Sprache  $L$ )
2474   \end{enumerate}
2475
2476   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2477   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2478 }
```

\liPumpingKontextfrei

```
2479 \def\liPumpingKontextfrei{%
2480   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2481   sich alle Wörter  $\omega$  in  $L$  mit  $|\omega| \geq j$  zerlegen lassen in
2482    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2483
2484   \begin{enumerate}
2485     \item  $|vx| \geq 1$ 
2486     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2487
2488     \item  $|vwx| \leq j$ 
2489     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2490
2491     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy$  in  $L$  (Für jede
2492     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2493     Sprache  $L$ )
2494   \end{enumerate}
2495 }
```

2496

2.36 quicksort.sty

```

2497 % https://tex.stackexchange.com/a/142634
2498 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2499 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2500
2501 %-----
2502 % USAGE:
2503 % \QSinitialize{comma, separated, numerical, values}
2504 % \loop
2505 % \QSpivotStep
2506 % \ifnum\value{pivotcount}>0
2507 %   \QSSortStep
2508 % \repeat
2509 %-----
2510
2511 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2512 \RequirePackage{xintfrac, xinttools}
2513
2514 \RequirePackage{tikz}
2515
2516 %-----
2517 % FIRST PART: TikZ styles and macros for the actual drawing
2518 \newcounter{cellcount}% used for coordinates of the node
2519 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2520
2521 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2522 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2523 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2524 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2525 % by police of LaTeX good conduct ? )
2526 \tikzset{l/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2527          o/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2528          r/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2529 % this is the "b" style as used in the image below
2530          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2531 % nicer:
2532          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2533          g/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2534
2535 % NOTE the b style was originally the same as the r(aised) style apart from
2536 % not being raised, but I find it nicer with a somewhat different
2537 % specification. I have not updated the images though.
2538
2539 % How the nodes are drawn depending on whether on the left of the pivot value
2540 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2541
2542 \def\DecoLEFT #1{%
2543   \xintFor* ##1 in {#1} \do
2544     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2545 }
2546
2547 \def\DecoINERT #1{%
2548   \xintFor* ##1 in {#1} \do
2549     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2550 }
2551
2552 \def\DecoRIGHT #1{%
2553   \xintFor* ##1 in {#1} \do
2554     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2555 }
2556
2557 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2558   \xintFor* ##1 in {#1} \do

```



```

2559     {\stepcounter{cellcount}}%
2560     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2561 }
2562
2563 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2564     \xintFor* ##1 in {#1} \do
2565     {\stepcounter{cellcount}}%
2566     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2567 }
2568
2569 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2570     \xintFor* ##1 in {#1} \do
2571     {\stepcounter{cellcount}}%
2572     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2573 }
2574
2575 %-----
2576 % SECOND PART: the actual sorting routines.
2577
2578 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2579 \def\QS@sort@b #1{\ifcase #1
2580     \expandafter\QS@sort@empty
2581     \or\expandafter\QS@sort@single
2582     \else\expandafter\QS@sort@c
2583     \fi
2584 }%
2585 \def\QS@sort@empty #1{}
2586 \def\QS@sort@single #1{\QSIr {#1}}
2587
2588 % This step is to pick the last as pivot.
2589 \def\QS@sort@c #1%
2590     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2591
2592 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2593 % The trick with \xintApplyUnbraced is that for example when selecting
2594 % the elements smaller than pivot, if we had been using \xintApply we
2595 % would have had at the minimum an empty brace pair. Thus we use the
2596 % "unbraced" variant, but then the \QS@select@smaller has added in
2597 % anticipation a level of braces.
2598 \def\QS@sort@d #1#2{%
2599     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2600     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2601     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2602 }%
2603 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2604 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2605 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{#{2}}{ }}% space will stop a f-
    expansion
2606
2607 %
2608 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2609 %
2610 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2611 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2612 % latter must handle correctly an empty argument.
2613
2614 %-----
2615 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QSinitialize.
2616
2617 % This draws all with suitable highlighting for the newly chosen pivots
2618 % (which will be shown raised)

```

```

2619 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2620             \let\QSIr\DecoINERT
2621             \let\QSIrr\DecoINERT
2622             \let\QSRr\DecoRIGHTwithPivot
2623 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2624             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2625             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2626 }
2627
2628 % This sorts and then draws, showing where the pivot chosen in the previous
2629 % step go. Next time they will have become "inert". If pivotcount is still at
2630 % zero on exit from \QSpivotStep, then this is the signal to stop before
2631 % executing \QSsortStep.
2632 \def\QSsortStep {\def\QSLr {\noexpand\QS@sort@a}%
2633             \def\QSRr {\noexpand\QS@sort@a}%
2634             \def\QSIr {\noexpand\QSIrr}%
2635             \let\QSIrr\relax
2636             \edef\QS@list{\QS@list}%
2637             \let\QSLr\relax
2638             \let\QSRr\relax
2639             \let\QSIr\relax
2640             \edef\QS@list{\QS@list}%
2641             \let\QSLr\DecoLEFT
2642             \let\QSIr\DecoINERTwithPivot
2643             \let\QSIrr\DecoINERT
2644             \let\QSRr\DecoRIGHT
2645 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2646             \setcounter{cellcount}{0}%
2647             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2648 }
2649
2650 \def\QSinitialize #1{%
2651     % first, we convert the comma separated values into a list of braced items
2652     % we use an \edef, and anyhow many \edef's will be used later
2653     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2654     \let\QSRr\DecoRIGHT
2655     % The \QSRr marker mutated to draw the last element as
2656     % pivot and the earlier ones with the suitable style.
2657     %
2658     % The list of marked braced items \QS@list is used both for drawing
2659     % (as here) and for doing the exchange of elements during sort.
2660     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2661             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2662 }
2663

```

2.37 relationale-algebra.sty

```

2664 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2665 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]
2666 \RequirePackage{amsmath}
2667 \RequirePackage{amssymb}

```

```

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

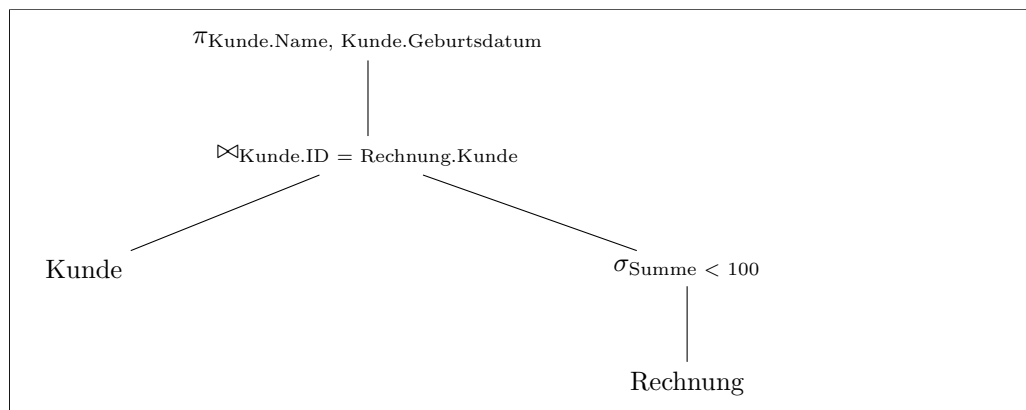
  \node[below=of pi]
    (theta join) {\bowtie_{\text{Kunde.ID = Rechnung.Kunde}}};
    edge (pi);

  \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}

```



```

2668 \RequirePackage{tikz}
2669 \usetikzlibrary{positioning}

    Privates Makros, das zwei Querstriche erzeugt.
2670 \def\o@join{\setbox0=\hbox{\bowtie}}%
2671   \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2672 }

\leftouterjoin A \leftouterjoin B: A ⋈ B
2673 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}

\rightouterjoin A \rightouterjoin B: A ⋉ B
2674 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}

\fullouterjoin A \fullouterjoin B: A ⋈⋉ B
2675 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
2676

```

2.38 rmodell.sty

```
2677 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2678 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2679 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2680 Datenbanken.]
2681 \RequirePackage{soul}
```

2.38.1 Makro-Kürzel

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

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

```
2682 \def\liPrimaer#1{\ul{#1}}
```

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

```
2683 \def\liFremd#1{{\setul{-0.9em}{}}\ul{#1}}
```

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

```
2684 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2685 \ExplSyntaxOn
2686 \NewDocumentEnvironment { liRmodell }
2687 { +b }
2688 {
2689   \medskip
2690   {
2691     \linespread{2}
2692     \setlength{\parindent}{0pt}
2693     \li@Rmodell@Schrift#1
2694   }
2695   \medskip
2696 } {}
2697 \ExplSyntaxOff
```

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

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

```
2698 \def\liRelationMenge#1#2{
2699 \noindent
2700 #1 : \[ #2 ]\}
2701 \par
2702 }
```

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

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

```
2703 \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}
```

```
2704 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2705
```

2.39 sortieren.sty

2706 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2707 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
 2708 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}
}
```



2709 \RequirePackage{tikz}
 2710 \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.

```
2711 \def\liVertauschen#1{
2712   \directlua{
2713     local sortieren = require('lehramt-informatik-sortieren')
2714     sortieren('#1')
2715   }
2716 }
```

\liSortierPfeil

```
2717 \def\liSortierPfeil#1#2{
2718   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2719 }
```

\liSortierPfeilUnten

```
2720 \def\liSortierPfeilUnten#1#2{
2721   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2722 }
```

\liSortierMarkierung

```
2723 \def\liSortierMarkierung#1#2{\node[
2724   draw,
2725   very thick,
2726   fit=(reihe.#1) (reihe.#2),
2727   inner sep=0pt
2728 ] {}
2729 }

2730 \tikzset{
2731   li sortierung zahlenreihe/.style={
2732     draw,
2733     thin,
2734     font=\large,
2735     rectangle split horizontal,
2736     rectangle split,
2737   }
2738 }
```

```

2739 % https://tex.stackexchange.com/a/140895
2740 \RequirePackage{forest,xstring}
2741 \usetikzlibrary{calc}
2742
2743 \makeatletter
2744 \pgfmathdeclarefunction{strrepeat}{2}{%
2745   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2746   \let\pgfmathresult\pgfutil@empty
2747   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2748     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2749     \advance\pgfmath@count-1\relax
2750   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2751 \makeatother
2752
2753 \def\myNodes{}
2754
2755 \ExplSyntaxOn
2756 \newcommand*\sortList[1]{%
2757   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2758 \ExplSyntaxOff
2759
2760 \forestset{
2761   sort/.code={%
2762     \pgfmathparse{level()>\forestSortLevel}%
2763     \ifnum\pgfmathresult=0
2764       \StrSubstitute{\forestov{content}}{ }{,}[\myList]%
2765       \sortList\myList
2766       \StrSubstitute{\myList}{,}{ }[\myList]%
2767       \pgfmathparse{strrepeat("1",level())}%
2768       \xappto\myNodes{\noexpand\node at ({\forestov{name}|-m})!-1!({\forestov{name}}$)
2769         (m\forestov{name}) {\myList}}%
2770       \pgfmathparse{level()=\forestSortLevel}%
2771       \ifnum\pgfmathresult=1
2772         \forestOget{\forestov{@first}}{name}\forestFirst
2773         \forestOget{\forestov{@last}}{name}\forestLast
2774         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2775           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2776       \fi
2777       \ifnum\forestov{@parent}=0\else
2778         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2779       \fi
2780       \gappto\myNodes{;}%
2781     \fi}}
2782
2783 \forestset{sort level/.code=%
2784   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2785   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2786

```

2.40 spalten.sty

```
2787 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2788 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2789 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2790 realisiert werden kann.]
2791 \RequirePackage{multicol}
```

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

```
2792 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2793
```

2.41 sql.sty

```
2794 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2795 \ProvidesPackage{lehramt-informatik-tabelle}[2021/09/02 Zu Setzen von SQL]

2796 \liLadePakete{syntax}

2797 \RequirePackage{fancyvrb}
2798 \DefineVerbatimEnvironment{liSqlErgebnis}{Verbatim}
2799 {fontsize=\footnotesize}

2800
```


2.42 struktogramm.sty

```
2801 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2802 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2803 Paket struktex zum Setzen von Struktogrammen]
2804 \RequirePackage{struktex}
2805
```

2.43 syntax.sty

```
2806 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2807 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2808 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2809 \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.

2.43.1 Makro-Kürzel

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

```
2810 \ExplSyntaxOn
2811 \directlua{
2812   syntax = require('lehramt-informatik-syntax')
2813   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2814   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2815   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2816   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2817   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2818   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2819 }
2820 \RequirePackage{hyperref}
2821 \RequirePackage{minted}
2822 % pygmentize -L styles
2823 \usemintedstyle{colorful}
2824 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2825 %\AfterEndEnvironment{minted}{\end{mdframed}}
2826 %\setminted{breaklines=true,linenos}
2827 \setminted{
2828   breaklines=true,
2829   linenos,
2830   fontsize=\footnotesize,
2831 }
```

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

Let-Abkürzung: `\let\j=\liJavaCode`

```
2832 \def\liJavaCode#1{\,\mintinline{java}|#1|\,}
```

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

```
2833 \def\liLatexCode#1{\mintinline{latex}|#1|}
```

```
2834 \def\li@GithubLink#1#2{
2835   \begin{flushright}
2836     \tiny
2837     Code-Beispiel~auf~Github~ansehen:~
2838     \href{#1}{\nolinkurl{#2}}
2839   \end{flushright}
2840 }
```

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

```
2841 \NewDocumentCommand{\liJavaDatei}{0{firstline=3} m }{
2842   \inputminted[#1]{java}{
2843     \directlua{
2844       syntax.drucke_absoluten_pfad('#2', false)
2845     }
2846   }
2847   \li@GithubLink
```

```

2848     {\directlua{syntax.drucke_github_url('#2', false)}}
2849     {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2850 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2851 \NewDocumentCommand{\liJavaTestDatei}{ O{firstline=3} m }{
2852   \inputminted[#1]{java}{
2853     \directlua{
2854       syntax.drucke_absoluten_pfad('#2', true)
2855     }
2856   }
2857   \li@GithubLink
2858   {\directlua{syntax.drucke_github_url('#2', true)}}
2859   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2860 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([.
\liJavaExamen$1{$2}{$3}{$4}{$5}
2861 \NewDocumentCommand{\liJavaExamen}{ O{firstline=3} m m m m }{
2862   \inputminted[#1]{java}{
2863     \directlua{
2864       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2865     }
2866   }
2867   \li@GithubLink
2868   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2869   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2870   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2871 }

\liAssemblerCode
2872 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2873 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2874   \inputminted{asm}{#1}
2875 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2876 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2877   \inputminted{componentpascal}{#1}
2878 }

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2879 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2880 \NewDocumentCommand{\liHaskellDatei}{ m }{
2881   \inputminted{haskell}{#1}
2882 }

2883 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2884 \def\liSqlCode#1{\mintinline{sql}|#1|}
2885

```

2.44 syntaxbaum.sty

```
2886 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2887 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2888 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
2889 \RequirePackage{tikz-qtrees}
2890
2891 \tikzset{li parsetree/.style={
2892     every internal node/.style={
2893         draw,circle
2894     },
2895     every leaf node/.style={
2896         draw,rectangle
2897     },
2898 }
2899 }
2900
```

2.45 synthese-algorithmus.sty

```
2901 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2902 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2903 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2904 Relation in die 3. Normalform]

2905 \liLadePakete{normalformen,mathe,typographie}
2906 \ExplSyntaxOn
```

2.45.1 Makro-Kürzel

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

2.45.2 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}
```

2.45.3 TeX-Markup Linksreduktion

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

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

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

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

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

2.45.4 TeX-Markup Rechtsreduktion

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

\liPseudoUeberschrift{F}

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

\liPseudoUeberschrift{A}

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

2.45.5 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

```
2907 \def\liSyntheseUeberschrift#1{
2908   {
2909     \bfseries
2910     \sffamily
2911     \str_case:nn {#1} {
2912       {1} {Kanonische-Überdeckung}
2913       {1-1} {Linksreduktion}
2914       {1-2} {Rechtsreduktion}
2915       {1-3} {Löschen-leerer-Klauseln}
2916       {1-4} {Vereinigung}
2917       {2} {Relationsschemata-formen}
2918       {3} {Schlüssel-hinzufügen}
2919       {4} {Entfernung-überflüssiger-Teilschemata}
2920     }
```

```

2921 }
2922 }

\liSyntheseErklaerung Let-Abkürzung: \let\erklaerung=\liSyntheseErklaerung
2923 \def\li@synthese@erklaerung@texte#1{
2924   \str_case:nn {#1} {
2925     {1} {
2926       Die~kanonische~Überdeckung~---also~die~kleinst~mögliche~noch~
2927       äquivalente~Menge~von~funktionalen~Abhängigkeiten~kann~in~vier~
2928       Schritten~erreicht~werden.
2929     }
2930     {1-1} {
2931       Führe~für~jede~funktionale~Anhängigkeit~
2932        $\alpha \rightarrow \beta$ ~die~Linksreduktion~durch,~
2933       überprüfe~also~für~alle~
2934        $A \in \alpha$ ,~ob~ $A$ ~überflüssig~ist,~d.h.~ob~
2935        $\beta \subseteqq \text{liAttributHuelle}\{F, \alpha \cup A\}$ .
2936     }
2937     {1-2} {
2938       Führe~für~jede~(verbliebene)~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~die~Rechtsreduktion~durch,~überprüfe~also~für~
2939       alle~ $B \in \beta$ ,~ob~ $B \in \text{liAttributHuelle}\{F \cup (\alpha \rightarrow \beta) \cup (\alpha \rightarrow B)\}$ ,~
2940        $\alpha$ ~gilt.~In~diesem~Fall~ist~B~auf~der~rechten~Seite~
2941       überflüssig~und~kann~eliminiert~werden,~d.h.~ $\alpha \rightarrow \beta$ ~wird~durch~ $\alpha \rightarrow (\beta \cup B)$ ~
2942       ersetzt.
2943     }
2944     {1-3} {
2945       Entferne~die~funktionalen~Abhängigkeiten~der~Form~ $\alpha \rightarrow \emptyset$ ,~die~im~2.~Schritt~möglicherweise~
2946       entstanden~sind.
2947     }
2948     {1-4} {
2949       Fasse~mittels~der~Vereinigungsregel~funktionale~Abhängigkeiten~
2950       der~Form~ $\alpha \rightarrow \beta_{\{1\}}, \dots, \alpha \rightarrow \beta_{\{n\}}$ ,~so~dass~ $\alpha \rightarrow \beta_{\{1\}} \cup \dots \cup \beta_{\{n\}}$ ~
2951       verbleibt.
2952     }
2953     % Kemper Seite 197
2954     {2} {
2955       Erzeuge~für~jede~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~in~ $F$ ~ein~Relationenschema~ $\mathcal{R}_{\alpha}$ ~
2956        $:= \alpha \cup \beta$ .
2957     }
2958     {3} {
2959       Falls~eines~der~in~Schritt~2.~erzeugten~Schemata~ $\mathcal{R}_{\alpha}$ ~einen~Schlüsselkandidaten~von~ $\mathcal{R}$ ~bezüglich~ $F$ ~enthält,~sind~wir~fertig,~sonst~wähle~einen~Schlüsselkandidaten~
2960        $\mathcal{K} \subseteqq \mathcal{R}$ ~aus~und~definiere~folgendes~zusätzliche~Schema:~ $\mathcal{R}_{\mathcal{K}} := \mathcal{K} \cup \mathcal{F} \setminus \mathcal{K}$ ~
2961       und~ $\mathcal{R}_{\mathcal{K}} \subseteqq \mathcal{R}_{\mathcal{K}'}$ .
2962     }
2963     {4} {
2964       Eliminiere~die~jenigen~Schemata~ $\mathcal{R}_{\alpha}$ ,~die~in~einem~anderen~Relationenschema~ $\mathcal{R}_{\alpha'}$ ~enthalten~sind,~d.h.~ $\mathcal{R}_{\alpha} \subseteqq \mathcal{R}_{\alpha'}$ .
2965     }
2966   }
2967 }
2968 }
2969 \def\liSyntheseErklaerung#1{
2970 {
2971   \itshape
2972   \footnotesize

```

```

2983     \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
2984   }
2985 }

```

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

```

2986 \def\liSyntheseUeberErklaerung#1{
2987   \liSyntheseUeberschrift{#1}\par
2988   \liSyntheseErklaerung{#1}
2989 }

```

```

2990 \ExplSyntaxOff
2991

```


2.46 tabelle.sty

2992 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2993 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]

2994 \RequirePackage{tabularx}

2995

2.47 typographie.sty

2996 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2997 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
 2998 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
 2999 formatierung.sty definiert.]

3000 \ExplSyntaxOn

Mit dem Packet wasysym gab es Unverträglichkeiten, deshalb verwenden wir fontawes-
 some.

3001 \RequirePackage{fontawesome}

\liErledigt \liErledigt: ☑

3002 \let\liErledigt=\faCheckSquareO

\liNichtsZuTun \liNichtsZuTun: ∅ Nichts zu tun

3003 \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 —

3004 \def\liParagraphMitLinien#1{

3005 \noindent

3006 \vrule height 2pt depth -1.6pt width 0.4cm

3007 \enspace

3008 #1

3009 \enspace

3010 \leaders\vrule height 2pt depth -1.6pt \hfill \null

3011 \par

3012 \medskip

3013 }

\liGeschweifteKlammern Große geschweifte Klammer mit Istgleich-Zeichen.

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

3014 \def\liGeschweifteKlammern#1#2#3#4{

3015 \par

3016 \medskip

3017 \noindent

3018 #1 \, \$\= \Bigl\{ \$\

3019 \vspace{#3}

3020 #2

3021 \vspace{#4}

3022 \begin{flushright} \$\Bigr\} \$\end{flushright}

3023 \par

3024 }

\liTypoUeberschrift

3025 \def\liTypoUeberschrift#1{

3026 {

3027 \bfseries\sffamily

3028 #1

3029 }

3030 }

\liTypoUeberGross

```
3031 \def\liTypoUeberGross#1{
3032   {
3033     \huge
3034     \liTypoUeberschrift{#1}
3035   }
3036 }
```

\liTypoUeberGross

```
3037 \def\liTypoUeberGROSS#1{
3038   {
3039     \Huge
3040     \liTypoUeberschrift{#1}
3041   }
3042 }
```

```
3043 \ExplSyntaxOff
```

```
3044
```

2.48 uml.sty

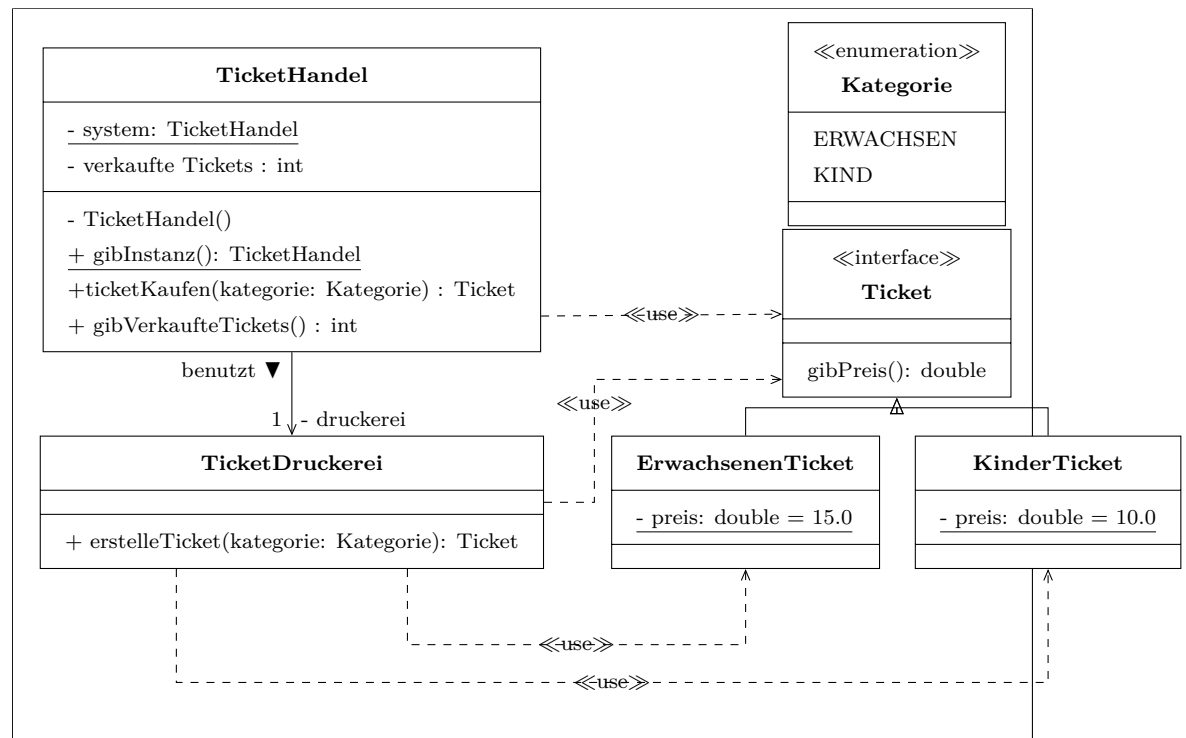
```

3045 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3046 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
3047 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
3048 Erweiterung bereitstellt]

3049 \RequirePackage{tikz-uml}
3050 \RequirePackage{tikz-uml-activity}
3051 % Not compatible with wasysym
3052 %\RequirePackage{mathabx}
3053 \RequirePackage{wasysym}
3054 \usetikzlibrary{positioning}

3055 \tikzumlset{
3056   fill class=white!0,
3057   font=\footnotesize,
3058   fill object=white!0,
3059   fill note=white!0,
3060   fill state=white!0,
3061   % Use case
3062   fill usecase=white!0,
3063   fill system=white!0,
3064 }

```



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

```

3065 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
3066   \def\@liDirLeft{}
3067   \def\@liDirRight{ \RIGHTarrow}
3068   \pgfkeys{/lese/dir/.is choice}
3069   \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
3070   \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
3071   \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{ \LEFTarrow }}}
3072   \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
3073
3074   \def\@liPos{above}
3075   \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
3076

```

```

3077 \def\@liDistance{0cm}
3078 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
3079
3080 \pgfkeys{/lese/.cd,#1}
3081
3082 \node[\@liPos = \@liDistance of #2-middle] {
3083   \@liDirLeft{\footnotesize#2}\@liDirRight
3084 };
3085 }
3086

```

2.49 vollstaendige-induktion.sty

```

3087 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3088 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01]
3089 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
3090 Überschriften für die einzelnen Schritte]

```

2.49.1 Makro-Kürzel

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

```

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

```

Lade häufig benötigte Pakete

```

3091 \RequirePackage{lehramt-informatik-typographie}
3092 \RequirePackage{lehramt-informatik-mathe}
3093 \RequirePackage{lehramt-informatik-syntax}
3094 \ExplSyntaxOn

```

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

Let-Abkürzung: \let\m=\liInduktionMarkierung

```
3095 \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

```
3096 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
3097 \def\liInduktionAnfang{
3098   \liPseudoUeberschrift{Induktionsanfang}
3099
3100   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
3101   \liParagraphMitLinien{
3102     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
3103   }
3104 }
```

\liInduktionVoraussetzung

```
3105 \def\liInduktionVoraussetzung{
3106   \liPseudoUeberschrift{Induktionsvoraussetzung}
3107
3108   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
3109   \liParagraphMitLinien{
3110     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ $.
3111   }
3112 }
```

\liInduktionSchritt

```
3113 \def\liInduktionSchritt{
3114   \liPseudoUeberschrift{Induktionsschritt}
3115
3116   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
3117   \liParagraphMitLinien{
3118     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
3119     auch~ $A(n=k+1)$ ~wahr~sein~muss.
3120   }
3121 }

3122 \ExplSyntaxOff
3123
```

2.50 wasserfall.sty

```
3124 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3125 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
3126 \RequirePackage{tikz}
3127 \tikzset{wasserfall/.style={
3128   >=stealth,
3129   node distance = 2mm and -8mm,
3130   start chain = A going below right,
3131   every node/.style = {
3132     draw,
3133     text width=24mm,
3134     minimum height=12mm,
3135     align=center,
3136     inner sep=1mm,
3137     fill=white,
3138     drop shadow={fill=black},
3139     on chain=A
3140   },
3141 }}
3142 \usetikzlibrary{chains,positioning,shadows}
3143
```


2.51 wpkalkuel.sty

```
3144 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3145 \ProvidesPackage{lehrant-informatik-wpkalkuel}[2020/06/13]
```

2.51.1 Makro-Kürzel

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

```
3146 \RequirePackage{amsmath}
```

```
3147 \ExplSyntaxOn
```

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

```
3148 \def\liWpKalkuelOhneMathe#1#2{
3149   \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)
3150 }
3151 \def\liWpKalkuel#1#2{
3152   \ifmmode
3153     \liWpKalkuelOhneMathe{#1}{#2}
3154   \else
3155     $\liWpKalkuelOhneMathe{#1}{#2}$
3156   \fi
3157 }
```

```
\MatheEnv
```

```
3158 \def\MatheEnv#1{
3159   \medskip
3160
3161   \hspace{1em}#1
3162
3163   \medskip
3164 }
```

```
\Mathe
```

```
3165 \def\Mathe#1{
3166   \MatheEnv{${#1}$}
3167 }
```

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

```
3168 \def\liWpEquivalent#1{
3169   \MatheEnv{$\equiv$\hspace{1em}${#1}$}
3170 }
```

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

```
3171 \newlength{\@Skip@Erklaerung@Reset}
3172 \def\liWpErklaerung#1{
3173   \setlength{\@Skip@Erklaerung@Reset}{\leftskip}
3174   \setlength{\leftskip}{0.5cm}
3175
3176   \par
3177   \noindent
3178   {
3179     \scriptsize
3180     #1
3181   }
3182   \par
3183
3184   \setlength{\leftskip}{\@Skip@Erklaerung@Reset}
3185 }
```


1592, 1593, 1594, 1596, 1597, 1599, 1623, 1625, 1630, 1634	\itshape 602, 2981	\LehramtInformatikGithubRawDomain 2815
\Gamma . 111, 169, 212, 1309	J	\LehramtInformatikGithubTexRepo 456, 2816
\gappto 2780	\j 1495, 1496, 1498, 1499, 1500, 1505, 1506, 1507	\LehramtInformatikRepository 17, 20, 23, 26, 29, 1588, 1841, 1842, 1843, 1844, 1845, 1846, 1847, 1848, 1849, 1850, 2813
\geometry 5	K	\LehramtInformatikTitel 1816
\geq 1718, 2074, 2460, 2465, 2481, 2485	\k 1505	\leq 2108, 2468, 2488
H	\keys 41, 75, 87, 117, 127, 175, 185, 310, 621, 625, 639, 644, 1373, 1380	\let 1298, 1299, 1865, 2619, 2620, 2621, 2622, 2635, 2637, 2638, 2639, 2641, 2642, 2643, 2644, 2654, 2746, 2784, 2785, 3002
\hbox 2670	L	\li@chomsky@erklaerung@texte 568, 604
\headrulewidth . 1428, 1822	\l 68, 69, 70, 71, 72, 73, 76, 77, 78, 79, 80, 82, 84, 89, 90, 91, 92, 93, 94, 109, 110, 111, 112, 113, 114, 115, 118, 119, 120, 121, 122, 123, 124, 130, 131, 132, 133, 134, 135, 136, 167, 168, 169, 170, 171, 172, 173, 176, 177, 178, 179, 180, 181, 182, 188, 189, 190, 191, 192, 193, 194, 619, 622, 627, 628, 631, 636, 637, 640, 641, 646, 1368, 1369, 1370, 1371, 1374, 1375, 1376, 1377, 1383, 1384, 1385, 1386, 1805, 1806, 1807, 1950, 1951, 1952, 1959	\li@EntwurfsCode 746, 797, 798, 799, 848, 849, 850, 851, 917, 918, 919, 920, 921, 922, 949, 950, 951, 952, 953, 954, 1020, 1190, 1191, 1192, 1193, 1233, 1234
\hfill 1553, 2093, 2099, 2104, 3010	\land 3189, 3191	\li@EntwurfsCodeAllgemein 745
\hinweis 1886	\LARGE 1402, 1630	\li@fussnote@text 2182, 2188, 2192, 2196, 2200
\hline 2231	\Large 1537	\li@GithubLink 2834, 2847, 2857, 2868
\href 1362, 1548, 1995, 2161, 2838	\large 1532, 1611, 1670, 2734	\li@mget . 1482, 1486, 1506
\hspace . . 2422, 3161, 3169	\leaders 3010	\li@minc 1485, 1507
\ht 2671	\left 2006, 2345	\li@mset 1479, 1487, 1496, 1499
\Huge 1621, 3039	\LEFTarrow 3071	\li@numdiscs 1491, 1500, 1506
\huge 1570, 3033	\leftarrow 665	\li@Rmodell@Schrift 2684, 2693, 2703
I	\leftouterjoin 2673	\li@sequence . . 1492, 1505
\i 1505, 1506	\leftskip 3173, 3174, 3184	\li@synthese@erklaerung@texte 2923, 2983
\ifcase 2579	\LehramtInformatikAutorEmail	\li@Ableitung 1323
\ifmmode 651, 659, 667, 675, 682, 1288, 1302, 1338, 2012, 2022, 2032, 2046, 2270, 2351, 2412, 3152	\LehramtInformatikAutorName	liAdditum (environment) 1918
\IfNoValueTF 1921, 1991, 1995 1821	liAHuelle (environment) 2277
\ifnum 2506, 2747, 2763, 2771, 2777	\LehramtInformatikGitBranch	\li@Alphabet 1308
\ifx 1911, 2308, 2775 458, 2818	\li@AssemblerDatei . . 2873
\in 577, 733, 1718, 2056, 2059, 2062, 2091, 2097, 2102, 2460, 2471, 2481, 2491, 2932, 2934, 2940, 2961, 3110	\LehramtInformatikGithubCodeRepo	\li@Attribut 2703
\inhaltsverzeichnis 1862 2817	liAttributHuelle 2268, 2935, 2940
\input 17, 20, 23, 26, 29, 479, 1587	\LehramtInformatikGithubDomain	\li@AttributHuelleOhneMathe 2268, 2271,
\inputminted 2842, 2852, 2862, 2874, 2877, 2881 455, 2814	
\int 2757		
\item 548, 549, 826, 830, 835, 840, 884, 893, 898, 906, 978, 983, 987, 1013, 1053, 1058, 1065, 1073, 1102, 1107, 1111, 1116, 1216, 1221, 1226, 1675, 1676, 1947, 1951, 2071, 2076, 2080, 2090, 2096, 2101, 2113, 2117, 2121, 2125, 2129, 2136, 2140, 2144, 2465, 2468, 2471, 2485, 2488, 2491		

2273, 2287, 2297, 2305	\liEntwurfsAdapterUml	\liEntwurfsZustandAkteure
\liAttributMenge 808, 854 1214, 1238
2276, 2288, 2291,	\liEntwurfsBeobachter 924	\liEntwurfsZustandCode
2298, 2299, 2313, 2315	\liEntwurfsBeobachterAkteure 1232, 1239
\liAufgabe 16 882, 926	\liEntwurfsZustandUml
\liAufgabenMetadaten	\liEntwurfsBeobachterCode 1199, 1237
. 47, 1528, 1602 916, 927	\liEpsilon 1294
\liAufgabenTitel 55	\liEntwurfsBeobachterUml	\liErAttribute
\liAusdruck 1345 858, 925 1252, 1266, 1268
\liAutomat 66	\liEntwurfsDekorierer 956	\liErDatenbankName . 1271
\liAutomatenKante 98	\liEntwurfsDekoriererAkteure	\liErEntity 1250, 1254, 1256
\liBandAlphabet 1309 958	\liErledigt 3002
\liBedingung 1797	\liEntwurfsDekoriererCode	\liErMpAttribute . . . 1265
\liBedingungDrei 948, 959	\liErMpEntity 1253
. 2061, 2104, 2144	\liEntwurfsDekoriererUml	\liErMpRelationship 1259
\liBedingungEins 929, 957	\liErRelationship . . .
. 2055, 2093, 2136	\liEntwurfsEinfacheFabrik 1251, 1260, 1262
\liBedingungFalsch . 1799 992	\liExamensAufgabe . . . 19
\liBedingungWahr . . . 1798	\liEntwurfsEinfacheFabrikAkteure	\liExamensAufgabeA . . 28
\liBedingungZwei 976, 994	\liExamensAufgabeTA . . 25
. 2058, 2099, 2140	\liEntwurfsEinfacheFabrikUml	\liExamensAufgabeTTA . 22
\liBeschriftung 1878 961, 993	\liExkurs (environment) 1926
\liBindeAufgabeEin . 1585	\liEntwurfsEinzelstueck	\liFalsch 549
\liChomskyErklaerung 1022	\liFlaci 1355
. 568, 609	\liEntwurfsEinzelstueckAkteure	\liFremd 2683
\liChomskyUeberErklaerung 1011, 1027	\liFunktionaleAbhaengigkeit
. 607	\liEntwurfsEinzelstueckBeschreibung . .	\liFunktionaleAbhaengigkeiten
\liChomskyUeberschrift 996, 1023 2321
. 556, 608	\liEntwurfsEinzelstueckCode	\liFussnote . . . 2181, 2183
\liCpmEreignis 617 1019, 1029	\liFussnoteDreiText .
\liCpmFruehErklaerung 688	\liEntwurfsEinzelstueckUml 2195, 2213
\liCpmFruehI 681, 701 1001, 1025	\liFussnoteEinsText .
\liCpmSpaetErklaerung 706	\liEntwurfsErbauer . 1079 2187, 2207
\liCpmSpaetI 674, 719	\liEntwurfsErbauerAkteure	\liFussnoteLink 1994
\liCpmVon 657 1051, 1081	\liFussnoten 2203
\liCpmVonOhneMathe . .	\liEntwurfsErbauerUml	\liFussnoteUrl . 1169, 1990
. 657, 660, 662 1031, 1080	\liFussnoteVierText .
\liCpmVonZu 649	\liEntwurfsFabrikmethode 2199, 2216
\liCpmVonZuOhneMathe 1125, 1150	\liFussnoteZweiText .
. 649, 652, 654	\liEntwurfsFabrikmethodeAkteure 2191, 2210
\liCpmVorgang 634 1100, 1127	\liGeschweifteKlammern
\liCpmZu 665	\liEntwurfsFabrikmethodeUml 1327, 2322, 3014
\liCpmZuOhneMathe 1083, 1126	\liGrammatik 1366
. 665, 668, 670	\liEntwurfsKompositum	\liGraphenFormat (envi-
\liDiagramm (environ- 1150	ronment) 1472
ment) 1977	\liEntwurfsKompositumAkteure	\liHanoi 1479
\liEinbettung (environ- 1152	\liHaskellCode 2879
ment) 1888	\liEntwurfsKompositumUml	\liHaskellDatei 2880
\liEntwurfs 1171 1129, 1151	\liInduktionAnfang . 3097
\liEntwurfsAbstrakteFabrik	\liEntwurfsModellPraesentation	\liInduktionErklaerung
. 801 1171 3096
\liEntwurfsAbstrakteFabrikBeschreibung	\liEntwurfsModellPraesentationAkteure	\liInduktionErklaerung
. 749, 802 1173 3095
\liEntwurfsAbstrakteFabrikCode	\liEntwurfsModellPraesentationBeschreibung	\liInduktionSchritt 3113
. 796, 806 1154, 1172	\liInduktionVoraussetzung
\liEntwurfsAbstrakteFabrikUml	\liEntwurfsStellvertreter 3105
. 754, 804 1195	\liJavaCode 2832
\liEntwurfsAdapter . . 853	\liEntwurfsStellvertreterCode	\liJavaDatei . . 747, 2841
\liEntwurfsAdapterAkteure 1189, 1197	\liJavaExamen 2861
. 823, 855	\liEntwurfsStellvertreterUml	\liJavaTestDatei . . . 2851
\liEntwurfsAdapterCode 1175, 1196	\liKasten (environment) 1417
. 847, 856	\liEntwurfsZustand . 1236	\liKellerAutomat 107

<code>\liKellerKante</code>	145	<code>\liPetriSetzeSchluessel</code>	2363	<code>\listen@punkt</code> . .	1947, 1959
<code>\liKellerUebergang</code> . .	140, 146	<code>\liPetriTransitionsName</code>	2410, 2422	<code>\liStrich</code>	1662
<code>\liKontrollCode</code>	1800	<code>\liPetriTransitionsNameOhneMathe</code>	2410, 2413, 2415	<code>\liSyntheseErklaerung</code>	2923, 2988
<code>liKontrollflussgraph</code> (environment)	1788	<code>\liPetriTransPfeile</code>	2422	<code>\liSyntheseUeberErklaerung</code>	2986
<code>\liKontrollKnotenPfad</code>	1802	<code>\liPolynomiellReduzierbar</code>	1680	<code>\liSyntheseUeberschrift</code>	2907, 2987
<code>\liKontrollTextzeileKnoten</code>	1801, 1806	<code>\liPotenzmenge</code>	1295, 1299, 2433	<code>\liT</code> 2038, 2053, 2067, 2131	
<code>\liKurzeTabellenLinie</code>	729	<code>\liPotenzmengeOhneMathe</code>	1296, 1297, 1298	<code>\liTeilen</code>	2265
<code>\liLadeAllePakete</code> . . .	233, 1573	<code>\liPrimaer</code>	2682	<code>\liLiteratur</code>	1829, 1853
<code>\liLadePakete</code>	59, 62, 229, 234, 555, 616, 1285, 1515, 1561, 1660, 2180, 2260, 2427, 2796, 2905	<code>\liProblemBeschreibung</code>	1664	<code>\liTheta</code>	2008, 2059, 2091, 2097, 2102
<code>\liLatexCode</code>	2833	<code>\liProblemClique</code> . .	1687	<code>\liThetaOhneMathe</code> . .	2008, 2013, 2015
<code>\liLeereZelle</code>	2218	<code>\liProblemName</code>	1663, 1670, 1682, 1684, 1697, 1708, 1709, 1717, 1718	<code>\liTOhneMathe</code>	2038, 2047, 2049
<code>liLernkartei</code> (environ- ment)	1967	<code>\liProblemSat</code>	1716	<code>\liTuringKante</code>	208
<code>\liLinksReduktion</code> . .	2285	<code>\liProblemSubsetSum</code> .	1707, 1716	<code>\liTuringLeerzeichen</code>	164, 172
<code>\liLinksReduktionInline</code>	2294, 2302	<code>\liProblemVertexCover</code>	1687, 1695	<code>\liTuringMaschine</code> . .	165
<code>\liMasterExkurs</code>	2149	<code>\liProduktionen</code> 1334, 1376		<code>\liTuringUeberfuehrung</code>	211
<code>\liMasterFaelle</code> 2088, 2156		<code>liProduktionsRegeln</code> (environment)	1324	<code>\liTuringUebergaenge</code>	203, 209
<code>\liMasterFallRechnung</code>	2134	<code>liProjektSprache</code> (envi- ronment)	1887	<code>\liTuringUebergangZelle</code>	198
<code>\liMasterVariablen</code> . .	2065, 2151	<code>\liPseudoUeberschrift</code>	1872, 1922, 1923, 2228, 2238, 3098, 3106, 3114	<code>\liTypoUeberGROSS</code> . .	3037
<code>\liMasterVariablenDeklaration</code>	2111	<code>\liPumpingKontextfrei</code>	2479	<code>\liTypoUeberGross</code> . . .	3031, 3037
<code>\liMasterWolframLink</code>	2159	<code>\liPumpingRegulaer</code> .	2458	<code>\liTypoUeberschrift</code> .	3025, 3034, 3040
<code>\liMenge</code>	76, 77, 79, 118, 119, 120, 124, 176, 177, 178, 182, 1286, 1335, 1374, 1375	<code>liQuellen</code> (environment)	1947	<code>\liUeberfuehrungsFunktion</code>	1300
<code>\liMengeOhneMathe</code> . .	1286, 1289, 1291	<code>\liRechtsReduktionInline</code>	2302	<code>\liUeberfuehrungsFunktionOhneMathe</code>	1300, 1303, 1305
<code>\liMetaSetze</code>	36, 48, 1529, 1603	<code>\liRekursionsGleichung</code>	2052, 2115	<code>liUebergangsTabelle</code> (environment)	2226
<code>\liMinimierungErklaerung</code>	2240	<code>\liRelation</code>	2332	<code>\liUeberschriftDreiecksTabelle</code>	2236
<code>\liMinispracheDatei</code>	2876	<code>liRelationenSchemaFormat</code> (environment)	2704	<code>\liUmlLeserichtung</code> .	3065
<code>\linespread</code>	2691	<code>\liRelationenMenge</code> . .	2698	<code>\liVertauschen</code>	2711
<code>\liNichtsZuTun</code>	3003	<code>\liRichtig</code>	548	<code>\liWortInSprache</code> . . .	730
<code>\liO</code>	2028, 2056	<code>liRmodell</code> (environment)	2684	<code>\liWortNichtInSprache</code>	735
<code>\liOmega</code>	2018, 2062	<code>\liRundeKlammer</code>	2005, 2009, 2019, 2029, 2043, 2344, 2348	<code>\liWpEquivalent</code> . . .	3168
<code>\liOmegaOhneMathe</code> . .	2018, 2023, 2025	<code>\liSetzeExamenTeilaufgabeNr</code>	1581	<code>\liWpErklaerung</code> . . .	3171
<code>\liONotationO</code>	2347	<code>\liSetzeExamenThemaNr</code>	1577	<code>\liWpErklaerungVerzweigung</code>	3186
<code>\liOOhneMathe</code>	2028, 2033, 2035	<code>\liSortierMarkierung</code>	2723	<code>\liWpKalkuel</code>	3148
<code>\liParagraphMitLinien</code>	604, 689, 707, 2242, 2983, 3004, 3101, 3109, 3117	<code>\liSortierPfeil</code>	2717	<code>\liWpKalkuelOhneMathe</code>	3148, 3153, 3155, 3187, 3189, 3191
<code>\liPetriErreichKnotenDrei</code>	2421	<code>\liSortierPfeilUnten</code>	2720	<code>\liZustandsBuchstabe</code>	1310, 1319, 1321, 1339, 1341
<code>\liPetriErreichTransition</code>	2418	<code>\liSpaltenUmbruch</code> . .	2792	<code>\liZustandsBuchstabeGross</code>	1311, 1320, 1322
		<code>\liSqlCode</code>	2884	<code>\liZustandsmenge</code> . .	1298
				<code>\liZustandsmengeNr</code> . .	1312, 2442
				<code>\liZustandsmengeNrGross</code>	1320

<code>\liZustandsMengenSammlung</code>	2424, 2449, 2454,	2701, 2987, 3011,
.....	2429	3015, 3023, 3176, 3182
<code>\liZustandsMengenSammlungNr</code>	2706, 2787, 2794,	<code>\paragraph</code> 1404
.....	2438	<code>\parindent</code> 2692
<code>\liZustandsmengeOhneMathe</code>	2901, 2992, 2996,	<code>\path</code> 99, 146, 209, 646
.....	1298	<code>\pgfkeys</code> .. 2374, 3068,
<code>\liZustandsname</code>	3045, 3087, 3124, 3144	3069, 3070, 3071,
<code>\liZustandsnameGross</code>	<code>\neg</code> 3191	3072, 3075, 3078, 3080
... 1322, 2430, 2439	<code>\negthinspace</code> .. 2006, 2345	<code>\pgfmather@count</code>
<code>\liZustandsnameTiefgestellt</code>	<code>\newcounter</code> ... 2518, 2519	... 2745, 2747, 2749
.....	<code>\NewDocumentCommand</code> .	<code>\pgfmather@smuggleone</code> 2750
<code>\liZustandsPaar</code> 67, 108,	<code>\pgfmather@declarefunction</code>
<code>\liZustandsPaarVariablenName</code>	145, 166, 208, 229, 2744
... 2219, 2222, 2223	618, 635, 674, 681,	<code>\pgfmather@hint</code> 2745
<code>\llap</code> 2671	730, 735, 1345,	<code>\pgfmather@parse</code>
<code>\log</code> 2056,	1367, 1680, 1803, 1486, 2762,
2059, 2062, 2091, 2097	1831, 1990, 1994,	2767, 2770, 2784, 2785
<code>\loop</code> 2504	2321, 2332, 2418,	<code>\pgfmather@result</code>
<code>\lor</code> 3190	2841, 2851, 2861, 1487, 2745,
<code>\ltimes</code> 2167	2873, 2876, 2880, 3065	2746, 2748, 2750,
	<code>\NewDocumentEnvironment</code>	2763, 2771, 2784, 2785
	1324, 1417, 1472,	<code>\pgfutil@empty</code> 2746
M	1788, 1887, 1888,	<code>\pgfutil@loop</code> 2747
<code>\makeatletter</code> .. 1963, 2743	1891, 1918, 1926,	<code>\pgfutil@repeat</code> 2750
<code>\makeatother</code> .. 1965, 2751	1948, 1967, 1977,	<code>\preceq</code> 1683
<code>\marginpar</code>	2227, 2277, 2686, 2704	<code>\prime</code> 1662
1255, 1261, 1267, 1870	<code>\newlength</code> 3171	<code>\printbibliography</code> . 1853
<code>\mathbb</code> .. 1718, 2491, 3110	<code>\node</code> 631, 1796,	<code>\ProvidesPackage</code>
<code>\mathbin</code> . 2673, 2674, 2675	1801, 2544, 2549, 2, 15, 33,
<code>\mathcal</code>	2554, 2560, 2566,	58, 226, 484, 545,
2029, 2348, 2961,	2572, 2723, 2768, 3082	552, 614, 727, 742,
2966, 2968, 2969, 2970	<code>\noexpand</code> 2632,	1243, 1279, 1392,
<code>\Mathe</code> 3165	2633, 2634, 2653, 2768	1432, 1441, 1446,
<code>\MatheEnv</code> 3158, 3166, 3169	<code>\noindent</code> 429, 732, 737,	1475, 1513, 1559,
<code>\mathord</code> 2174, 2175	1531, 1536, 1543,	1657, 1729, 1812,
<code>\mdfsetup</code> 1416,	1550, 1553, 1874,	1828, 1834, 1856,
1896, 1900, 1904, 1908	1876, 1880, 1884,	2002, 2165, 2178,
<code>\medskip</code> .. 1534, 1672,	1912, 1940, 1942,	2257, 2340, 2359,
1881, 1883, 1957,	1955, 1971, 1973,	2425, 2450, 2455,
1985, 2689, 2695,	1981, 2153, 2206,	2499, 2665, 2678,
3012, 3016, 3159, 3163	2209, 2212, 2215,	2707, 2788, 2795,
<code>\memph</code> 1870	2699, 3005, 3017, 3177	2802, 2807, 2887,
<code>\mintinline</code> 2832,	<code>\nolinkurl</code> 2838	2902, 2993, 2997,
2833, 2872, 2879, 2884	<code>\normalsize</code> 1404	3046, 3088, 3125, 3145
<code>\mkern</code> ... 2673, 2674, 2675	<code>\notin</code> 738	
<code>\mlq</code> 2172, 2174	<code>\null</code> 3010	
<code>\mrq</code> 2172, 2175		Q
<code>\msg</code> 477		<code>\QS@list</code>
<code>\myList</code>	O	2625, 2636, 2640,
2764, 2765, 2766, 2769	<code>\o</code> 2347, 2352, 2354	2647, 2653, 2658, 2661
<code>\myNodes</code> 2753,	<code>\o@join</code>	<code>\QS@select@equal</code>
2768, 2774, 2778, 2780	2670, 2673, 2674, 2675 2600, 2604
	<code>\Omega</code> 2019	<code>\QS@select@greater</code> ..
N	<code>\omega</code> 2460, 2461, 2481, 2482 2601, 2605
<code>\NeedsTeXFormat</code>	<code>\or</code> 2581	<code>\QS@select@smaller</code> ..
..... 1, 14, 32,		... 2596, 2599, 2603
57, 225, 483, 544,	P	<code>\QS@sort@a</code>
551, 613, 726, 741,	<code>\pagestyle</code> 1427, 1609	2578, 2611, 2632, 2633
1242, 1278, 1391,	<code>\par</code> 428, 449, 608, 1356,	<code>\QS@sort@b</code> 2578, 2579
1431, 1440, 1445,	1364, 1532, 1550,	<code>\QS@sort@c</code> 2582, 2589
1474, 1512, 1558,	1553, 1612, 1616,	<code>\QS@sort@d</code> 2590, 2598
1656, 1728, 1811,	1623, 1625, 1630,	<code>\QS@sort@empty</code> . 2580, 2585
1827, 1833, 1855,	1634, 1638, 1879,	<code>\QS@sort@single</code> 2581, 2586
2001, 2164, 2177,	1941, 1964, 1972,	<code>\QS@initialize</code>
2256, 2339, 2358,	2623, 2645, 2660,	... 2503, 2615, 2650

<code>\QSIr</code> . 2586, 2592, 2600, 2620, 2634, 2639, 2642	<code>\Roman</code> 1412	<code>\subsection</code> 1583
<code>\QSIrr</code> 2621, 2634, 2635, 2643	<code>\roman</code> 1412, 1414	<code>\subseteq</code> 2935, 2968, 2975
<code>\QSLr</code> 2592, 2599, 2610, 2611, 2619, 2632, 2637, 2641	<code>\romannumeral</code> 2590	<code>\subsubsection</code> 1604
<code>\QSpivotStep</code> 2505, 2615, 2619, 2630	<code>\rtimes</code> 2167	
<code>\QSR</code> 2592	<code>\rule</code> 2623, 2645, 2660, 2671	T
<code>\QSRr</code> 2601, 2622, 2633, 2638, 2644, 2653, 2654, 2655	S	<code>\tableofcontents</code> 1651, 1866
<code>\QSSortStep</code> 2507, 2615, 2631, 2632	<code>\sb</code> 72, 82, 84, 113, 171, 584, 585, 589, 592, 593, 594, 676, 678, 683, 685, 1339, 1341, 2056, 2059, 2062, 2091, 2097, 2246, 2410, 2419, 2954, 2955, 2956, 2961, 2965, 2966, 2969, 2970, 2973, 2974, 2975	<code>\TeX</code> 1551
<code>\quad</code> 2184	<code>\scriptscriptstyle</code> 649, 657, 665	<code>\text</code> 82, 84, 187, 2268, 3096, 3149
R	<code>\scriptsize</code> 1358, 1741, 1748, 1754, 1816, 1817, 1820, 1821, 3096, 3149, 3179	<code>\textbf</code> 696, 714, 1250, 1688, 1697, 1708, 1717, 1875, 1882, 1913, 1941, 1956, 1972, 2231
<code>\raisebox</code> 1801	<code>\section</code> ... 52, 1570, 1579	<code>\textcolor</code> 1800, 3095
<code>\relax</code> 1865, 2592, 2635, 2637, 2638, 2639, 2747, 2749	<code>\sectionbreak</code> 1571	<code>\textit</code> 1090, 1132, 1133, 1134, 1135, 1982, 2276, 2336
<code>\renewcommand</code> 1407, 1408, 1409, 1410, 1413, 1414, 1428, 1429, 1646, 1650, 1822, 1823, 2226	<code>\seq</code> . 1805, 1806, 1807, 1950, 1951, 1952, 1959	<code>\textsc</code> 1663
<code>\repeat</code> 2508	<code>\setbox</code> 2670	<code>\textsf</code> 1875, 1956
<code>\RequirePackage</code> 4, 60, 163, 227, 231, 480, 486, 487, 547, 615, 744, 745, 1245, 1247, 1248, 1284, 1393, 1394, 1397, 1399, 1401, 1406, 1415, 1422, 1433, 1434, 1443, 1447, 1448, 1449, 1477, 1478, 1526, 1569, 1574, 1575, 1607, 1661, 1730, 1830, 1835, 1836, 1852, 1859, 1860, 1861, 1889, 2004, 2168, 2169, 2342, 2343, 2360, 2362, 2452, 2512, 2514, 2666, 2667, 2668, 2681, 2709, 2740, 2791, 2797, 2804, 2809, 2820, 2821, 2889, 2994, 3001, 3049, 3050, 3052, 3053, 3091, 3092, 3093, 3126, 3146	<code>\setcounter</code> 1405, 1572, 1606, 2624, 2646, 2660	<code>\textstyle</code> 2076, 2108
<code>\right</code> 2006, 2345	<code>\setganttlinklabel</code> .. 1435, 1436, 1437, 1438	<code>\texttt</code> 1273, 1663, 1797, 1798, 1799, 1800, 3149
<code>\RIGHTarrow</code> ... 3067, 3072	<code>\setlength</code> 1644, 1645, 1647, 1824, 2692, 3173, 3174, 3184	<code>\textwidth</code> 1824
<code>\Rightarrow</code> 733, 738	<code>\setmainfont</code> 1395	<code>\thepage</code> 1426, 1819
<code>\rightarrow</code> 212, 572, 577, 585, 589, 591, 592, 594, 649, 657, 2422, 2932, 2939, 2941, 2944, 2949, 2954, 2955, 2960	<code>\setmainlanguage</code> 481	<code>\theparagraph</code> 1404
<code>\rightouterjoin</code> 2674	<code>\setminted</code> 2826, 2827	<code>\thesection</code> 1570
	<code>\setminus</code> 2306	<code>\Theta</code> 2009
	<code>\setsansfont</code> 1396	<code>\thinspace</code> 3149
	<code>\setul</code> 2683	<code>\tikz</code> 1801
	<code>\sffamily</code> 559, 1402, 1404, 1506, 1570, 1621, 2910, 3027	<code>tikz: bbaum</code> 26
	<code>\shoveleft</code> 2286	<code>tikz: li binaer baum</code> 24
	<code>\shoveright</code> 2290	<code>\tikzchildnode</code> 504
	<code>\Sigma</code> 69, 110, 168, 1308, 1309, 1369	<code>\tikzparentnode</code> 504
	<code>\sigma</code> 582, 584, 585	<code>\tikzset</code> 101, 148, 214, 489, 515, 1451, 1732, 2400, 2526, 2730, 2891, 3127
	<code>\SLASH</code> 1871	<code>\tikzumset</code> 3055
	<code>\small</code> 1980	<code>\times</code> 212
	<code>\sort</code> 2757	<code>\tiny</code> . 1256, 1262, 1268, 1542, 1800, 1870, 2836
	<code>\sortList</code> 2756, 2765	<code>\titleformat</code> 1402, 1404, 1570
	<code>\square</code> 549	<code>\titlespacing</code> 1403
	<code>\stepcounter</code> 2544, 2549, 2554, 2557, 2559, 2563, 2565, 2569, 2571	<code>\tl</code> 39, 68, 69, 70, 71, 72, 73, 76, 77, 78, 79, 80, 82, 84, 109, 110, 111, 112, 113, 114, 115, 118, 119, 120, 121, 122, 123, 124, 167, 168, 169, 170, 171, 172, 173, 176, 177, 178, 179, 180, 181, 182, 303, 307, 334, 338, 339, 340, 343, 348, 349, 350, 362, 371, 406, 409, 412, 420, 421, 422, 423, 434, 440, 443, 446, 462, 469, 619, 622, 627, 628, 636, 637, 640,
	<code>\str</code> 361, 560, 569, 1893, 2434, 2443, 2911, 2924	
	<code>\string</code> 2288, 2298	
	<code>\StrSubstitute</code> . 2764, 2766	
	<code>\strut</code> 2113, 2117, 2121, 2125, 2129, 2792	

641, 1347, 1368, 1369, 1370, 1371, 1374, 1375, 1376, 1377, 1578, 1582, 1586, 1593, 1596, 2039			
\tmp	2308		
\TmpPlaceEight	2382		
\TmpPlaceFive	2379		
\TmpPlaceFour	2378		
\TmpPlaceNine	2383		
\TmpPlaceOne	2375		
\TmpPlaceSeven	2381		
\TmpPlaceSix	2380		
\TmpPlaceTen	2384		
\TmpPlaceThree	2377		
\TmpPlaceTwo	2376		
\TmpScale	2395		
\TmpTransitionEight	2371, 2392		
\TmpTransitionFive	2368, 2389		
\TmpTransitionFour	2367, 2388		
\TmpTransitionNine	2372, 2393		
\TmpTransitionOne	2364, 2385		
\TmpTransitionSeven	2370, 2391		
\TmpTransitionSix	2369, 2390		
\TmpTransitionTen	2373, 2394		
\TmpTransitionThree	2366, 2387		
\TmpTransitionTwo	2365, 2386		
\TmpX	2396		
\TmpY	2397		
\today	1817		
\ttfamily	2684		
		U	
		\ul	1251, 2682, 2683
		\umlagggreg	1209
		\umlassoc	1167
		\umlclass	
			756, 760, 764, 811, 812, 813, 860, 865, 870, 873, 931, 932, 933, 938, 939, 968, 1003, 1034, 1035, 1038, 1089, 1092, 1131, 1137, 1138, 1156, 1157, 1158, 1179, 1180, 1181, 1201, 1202, 1203, 1204
		\umldep	1097
		\umlHVVHagggreg	
			878, 944, 1147
		\umlinherit	817, 868, 1042, 1087, 1095
		\umlnote ..	819, 1044, 1211
		\umlreal	815, 876
		\umlsimpleclass	
			771, 772, 773, 777, 779, 780, 781, 810, 963, 964, 965, 1033, 1085, 1086, 1177
		\umlstatic	970, 1004
		\umluniagggreg	1040
		\umluniassoc	
			791, 816, 1041, 1165, 1166, 1185, 1186
		\umlVHuniassoc ..	792, 793
		\umlVHVdep	785, 786, 788, 789, 972, 973
		\umlVHVinherit 768, 769, 774, 775, 782, 783, 941, 942, 966, 967, 1145, 1146, 1183, 1184	
		\umlVHVreal	
			935, 936, 1206, 1207
		\UParrow	3069
		\url	463, 1991
		\usemintedstyle	2823
		\usetikzlibrary ..	61, 488, 1246, 1450, 1731, 2361, 2669, 2710, 2741, 3054, 3142
		V	
		\value	2506
		\varepsilon	561, 572, 573, 1294, 2056, 2062, 2094, 2105
		\vfill 1540, 1618, 1649, 2792	
		\vrule	3006, 3010
		\vspace ...	1614, 1628, 1632, 1636, 1646, 1927, 1945, 3019, 3021
		X	
		\xappto ..	2768, 2774, 2778
		\xdef	1480
		\xintApply	2594
		\xintApplyUnbraced	2593, 2599, 2600, 2601
		\xintCSVtoList	2653
		\xintFor	
			2543, 2548, 2553, 2558, 2564, 2570, 2611
		\xintifEq	2604
		\xintifForLast	
			2560, 2566, 2572
		\xintifGt	2605
		\xintifLt	2603
		\xintLength	2578
		\xintnthelt	2590
		Z	
		\ZB	1998
		\zB	1997
		\zustandsnamens@liste ...	1312, 1319, 1320