

lehramt-informatik

Hermine Bschlangaul *

September 4, 2021

Contents

1	Klassen	4
1.1	Vorlage Theorie-Teil	5
1.2	Vorlage Aufgabensammlung	6
1.3	Vorlage Aufgabe	7
2	Pakete	8
2.1	abmessung.sty	9
2.2	aufgaben-einbinden.sty	10
2.3	aufgaben-metadaten.sty	11
2.4	automaten.sty	12
2.4.1	Endlicher Automat	12
2.4.2	Kellerautomat	14
2.4.3	Turingmaschine	15
2.5	basis.sty	18
2.6	baum.sty	22
2.6.1	Binärbaum	23
2.6.2	AVL-Baum	24
2.6.3	B-Baum	25
2.7	checkbox.sty	26
2.8	chomsky-normalform.sty	27
2.9	cpm.sty	29
2.9.1	Makro-Kürzel	29
2.9.2	Hilfsmakros für Zeitpunkt-Tabelle	30
2.10	cyk-algorithmus.sty	32
2.10.1	Makro-Kürzel	32
2.10.2	TeX-Markup-Beispiel	32
2.11	entwurfsmuster.sty	33
2.11.1	Namensschema der Entwurfsmuster-Makros:	33
2.11.2	Reihenfolge	33
2.11.3	Abstrakte Fabrik (Abstract Factory)	33
2.11.4	Adapter	34
2.11.5	Beobachter (Observer)	36
2.11.6	Dekorierer (Decorator)	38
2.11.7	Einfache Fabrik (Simple Factory)	39
2.11.8	Einzelstück (Singleton)	40
2.11.9	Erbauer (Builder)	40
2.11.10	Fabrikmethode (Factory Method)	42
2.11.11	Kompositum (Composite)	43
2.11.12	Modell-Präsentation-Steuerung (Model-View-Controller)	44
2.11.13	Zustand (State)	45
2.12	er.sty	47
2.12.1	Tex-Markup-Beispiel: Komplettes Diagramm	47

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

2.12.2	Tex-Markup-Beispiel: EER Enhanced Entity-Relation-Modell nach Elmasri/Navante	48
2.12.3	Makro-Kürzel	48
2.13	formale-sprachen.sty	50
2.14	formatierung.sty	53
2.14.1	Schriftarten / Typographie	53
2.14.2	Farben	53
2.14.3	Überschriften	53
2.14.4	Listen	53
2.14.5	Kasten	53
2.14.6	Header	53
2.15	gantt.sty	54
2.16	grafik.sty	55
2.17	graph.sty	56
2.18	hanoi.sty	58
2.19	klassen-konfiguration-aufgabe.sty	59
2.20	klassen-konfiguration-examen.sty	60
2.21	komplexitaetstheorie.sty	61
2.21.1	Makro-Kürzel	61
2.22	kontrollflussgraph.sty	63
2.22.1	Makro-Kürzel	63
2.22.2	TeX-Markup-Beispiel	63
2.22.3	TikZ: pin	63
2.22.4	Umgebungen	64
2.22.5	Makros	65
2.23	kopf-fusszeilen.sty	66
2.24	literatur-dummy.sty	67
2.25	literatur.sty	68
2.26	makros.sty	69
2.27	master-theorem.sty	73
2.27.1	Makro-Kürzel	73
2.28	mathe.sty	77
2.29	minimierung.sty	78
2.30	normalformen.sty	81
2.30.1	Makro-Kürzel	81
2.31	petri.sty	84
2.31.1	Makro-Kürzel	84
2.32	potenzmengen-konstruktion.sty	86
2.33	pseudo.sty	88
2.34	pumping-lemma.sty	89
2.35	quicksort.sty	90
2.36	relationale-algebra.sty	93
2.37	rmodell.sty	94
2.37.1	Makro-Kürzel	94
2.38	sortieren.sty	95
2.39	spalten.sty	97
2.40	sql.sty	98
2.41	struktogramm.sty	99
2.42	syntax.sty	100
2.42.1	Makro-Kürzel	100
2.43	syntaxbaum.sty	102
2.44	synthese-algorithmus.sty	103
2.44.1	Makro-Kürzel	103
2.44.2	TeX-Markup Grundgerüst	103
2.44.3	TeX-Markup Linksreduktion	103
2.44.4	TeX-Markup Rechtsreduktion	103
2.44.5	TeX-Markup Relationen formen	104
2.45	tabelle.sty	107

2.46	typographie.sty	108
2.47	uml.sty	109
2.48	vollstaendige-induktion.sty	111
2.48.1	Makro-Kürzel	111
2.49	wasserfall.sty	113
2.50	wpkalkuel.sty	114
2.50.1	Makro-Kürzel	114
3	Index	115

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

```

`\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,
}

```

```

36 \def\liAufgabenMetadaten#1{
37
38   \_setze_variablen_zurueck:
39
40   \tl_clear:N \g_auf_examen_telaufgabe_nr_tl
41
42   \keys_set:nn { aufgabenmetadaten } {
43     #1
44   }
45
46   \_setze_relativen_pfad:
47
48   \tl_if_empty:NTF \g_auf_titel_tl
49   {
50     \msg_fatal:nn { aufgabenmetadaten } { kein-titel }
51   }
52   {
53   }
54
55   \_gib_examen_titel: {}
56
57   \section{\_gib_aufgaben_titel:}
58
59   \bigskip
60 }

```

```

61 \ExplSyntaxOff

```

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

```

62 \def\liAufgabenTitel#1{}

```

```

63

```

2.4 automaten.sty

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

2.4.1 Endlicher Automat

```
66 \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}
```



```
67 \RequirePackage{tikz}
68 \usetikzlibrary{arrows,automata,positioning}
69 \liLadePakete{mathe}
70 \directlua{
71   automaten = require('lehramt-informatik-automaten')
72 }
```

`\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}`: $A_{NEA} = (Z, \Sigma, \delta, E, z_0)$

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

```

87     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
88     dea .value_forbidden:n = true,
89     dea .code:n = {\tl_set:Nn \l_typ_tl {\sb{\text{DEA}}}},
90     nea .value_forbidden:n = true,
91     nea .code:n = {\tl_set:Nn \l_typ_tl {\sb{\text{NEA}}}},
92   }
93
94   \keys_set:nn { automat } { #2 }
95
96   $ #1 \l_typ_tl = (
97     \l_zustaende_tl,
98     \l_alphabet_tl,
99     \l_delta_tl,
100    \l_ende_tl,
101    \l_start_tl
102  )$
103 }
104 \ExplSyntaxOff

```

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

```

105 \def\liAutomatenKante#1#2#3#4{
106   \path (#1) edge[#4] node{#3} (#2);
107 }

108 \tikzset{
109   li automat/.style={
110     ->,
111     node distance=2cm
112   },
113 }

```

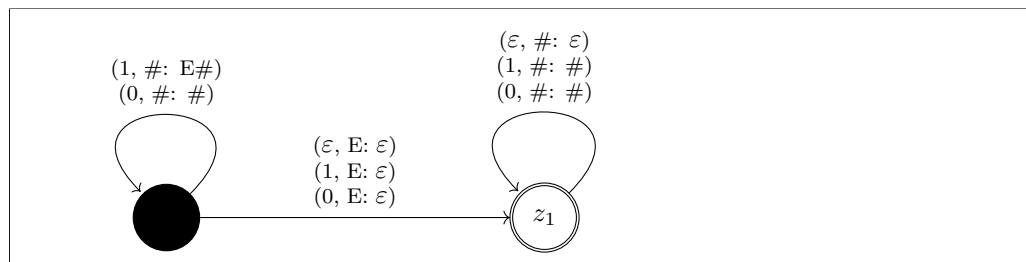
2.4.2 Kellerautomat

```
\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},
}
```

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

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

```

130   kellerboden .code:n = {\tl_set:Nn \l_kellerboden_tl {##1}},
131   ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
132 }
133
134 \keys_set:nn { kellerautomat } { #2 }
135
136 $#1 = (
137   \l_zustaende_tl,
138   \l_alphabet_tl,
139   \l_kelleralphabet_tl,
140   \l_delta_tl,
141   \l_start_tl,
142   \l_kellerboden_tl,
143   \l_ende_tl
144 )$
145 }
146 \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}`

```

147 \ExplSyntaxOn
148 \def\liKellerUebergang#1{
149   \directlua{automaten.drucke_keller_uebergaenge('#1')}
150 }
151 \ExplSyntaxOff

```

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

```

152 \NewDocumentCommand{\liKellerKante} { 0{above} m m m } {
153   \path (#2) edge[#1] node{\liKellerUebergang{#4}} (#3);
154 }

155 \tikzset{
156   li keller knoten/.style={
157     text width=2cm,
158     align=center,
159     font=\footnotesize,
160   },
161   li kellerautomat/.style={
162     li automat,
163     every edge/.append style={
164       every node/.style={
165         li keller knoten
166       }
167     }
168   }
169 }

```

2.4.3 Turingmaschine

```

170 \RequirePackage{amssymb}

```

\liTuringLeerzeichen

□

```

171 \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},
}

```

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

```

172 \ExplSyntaxOn
173 \NewDocumentCommand {\liTuringMaschine} { O{TM} m } {
174   \tl_set:Nn \l_zustaende_tl {Z}
175   \tl_set:Nn \l_alphabet_tl {\Sigma}
176   \tl_set:Nn \l_bandalphabet_tl {\Gamma}
177   \tl_set:Nn \l_delta_tl {\delta}
178   \tl_set:Nn \l_start_tl {z\sb{0}}
179   \tl_set:Nn \l_leerzeichen_tl {\liTuringLeerzeichen}
180   \tl_set:Nn \l_ende_tl {E}
181
182   \keys_define:nn { kellerautomat } {
183     zustaende .code:n = {\tl_set:Nn \l_zustaende_tl {\liMenge{##1}}},
184     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
185     bandalphabet .code:n = {\tl_set:Nn \l_bandalphabet_tl {\liMenge{##1}}},
186     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
187     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
188     leerzeichen .code:n = {\tl_set:Nn \l_leerzeichen_tl {##1}},
189     ende .code:n = {\tl_set:Nn \l_ende_tl {\liMenge{##1}}},
190   }
191
192   \keys_set:nn { kellerautomat } { #2 }
193
194   $\text{#1} = (
195     \l_zustaende_tl,
196     \l_alphabet_tl,
197     \l_bandalphabet_tl,
198     \l_delta_tl,
199     \l_start_tl,
200     \l_leerzeichen_tl,
201     \l_ende_tl
202   )$
203 }
204 \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)`

```

205 \ExplSyntaxOn
206 \def\liTuringUebergangZelle#1{
207   \directlua{tex.print(automaten.gib_einen_turing_uebergang('#1'))}
208 }
209 \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)`


```

210 \ExplSyntaxOn
211 \def\liTuringUebergaenge#1{
212   \directlua{automaten.drucke_turing_uebergaenge('#1')}
213 }
214 \ExplSyntaxOff

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

215 \NewDocumentCommand{\liTuringKante} { 0{above} m m m } {
216   \path (#2) edge[#1] node{\liTuringUebergaenge{#4}} (#3);
217 }

\liTuringUeberfuehrung

218 \def\liTuringUeberfuehrung{
219    $\delta : Z \times \Gamma \rightarrow Z \times \Gamma \times \{ L, R, N \}$ 
220 }

221 \tikzset{
222   li turingmaschine/.style={
223     li automat,
224     every edge/.append style={
225       every node/.style={
226         li keller knoten
227       }
228     }
229   }
230 }
231

```

2.5 basis.sty

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

234 \RequirePackage{xparse}

235 \ExplSyntaxOn

\liLadePakete

236 \NewDocumentCommand{\liLadePakete}{ m }
237 {
238   \clist_map_inline:nn { #1 } { \RequirePackage{lehramt-informatik-##1} }
239 }

\liLadeAllePakete

240 \def\liLadeAllePakete{
241   \liLadePakete{
242     aufgaben-einbinden,
243     automaten,
244     baum,
245     checkbox,
246     chomsky-normalform,
247     cpm,
248     cyk-algorithmus,
249     entwurfsmuster,
250     er,
251     formale-sprachen,
252     gantt,
253     grafik,
254     graph,
255     hanoi,
256     kontrollflussgraph,
257     makros,
258     master-theorem,
259     mathe,
260     minimierung,
261     normalformen,
262     petri,
263     potenzmengen-konstruktion,
264     pseudo,
265     quicksort,
266     relationale-algebra,
267     rmodell,
268     sortieren,
269     spalten,
270     struktogramm,
271     syntax,
272     syntaxbaum,
273     synthese-algorithmus,
274     tabelle,
275     typographie,
276     uml,
277     vollstaendige-induktion,
278     wasserfall,
279     wpkalkuel,
280   }
281 }
```

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.

```
282 \clist_new:N \g_auf_schluesssel_clist
283 \clist_set:Nn \g_auf_schluesssel_clist {
```

```

284  titel,
285  thematik,
286  stichwoerter,
287  zitat_schluessel,
288  zitat_beschreibung,
289  %
290  bearbeitungs_stand,
291  korrektheit,
292  %
293  relativer_pfad,
294  identische_aufgabe,
295  %
296  examen_nummer,
297  examen_jahr,
298  examen_monat,
299  examen_thema_nr,
300  examen_teilaufgabe_nr,
301  examen_aufgabe_nr,
302 }

```

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

```

303 \clist_map_inline:Nn \g_auf_schluessel_clist {
304   \tl_new:c {g_auf_#1_tl}
305 }

```

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

```

306 \cs_new:Npn \setze_variablen_zurueck: {
307   \clist_map_inline:Nn \g_auf_schluessel_clist {
308     \tl_clear:c {g_auf_##1_tl}
309   }
310 }

```

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

```

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

```

```

340     }
341     {
342         \tl_gset:Nn \g_auf_relativer_pfad_tl {
343             Staatsexamen /
344             \g_auf_examen_nummer_tl /
345             \g_auf_examen_jahr_tl /
346             \g_auf_examen_monat_tl /
347             \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} { Thema - \g_auf_examen_thema_nr_tl / }
348             \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} { Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl / }
349             \tl_if_empty:NTF \g_auf_examen_aufgabe_nr_tl {} { Aufgabe - \g_auf_examen_aufgabe_nr_tl / }
350         }
351     }
352     {}
353 }
354 {}
355 }

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

394 \cs_new:Npn \_gib_github_url: {
395     \tl_if_empty:NTF \g_auf_relativer_pfad_tl {} {
396         Github :~\href{
397             \LehramtInformatikGithubRawDomain /
398             \LehramtInformatikGithubTexRepo /
399             \LehramtInformatikGitBranch /
400             \g_auf_relativer_pfad_tl
401         }{
402             \nolinkurl{\g_auf_relativer_pfad_tl}

```

```

403     }
404   }
405 }

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

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

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

```

2.6 baum.sty

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

426 \RequirePackage{tikz}

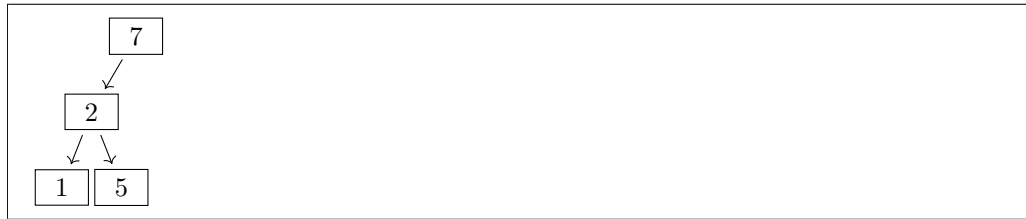
    für li binaer baum
427 \RequirePackage{tikz-qtrees}

    Für b baum
428 \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}
```



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

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}
```

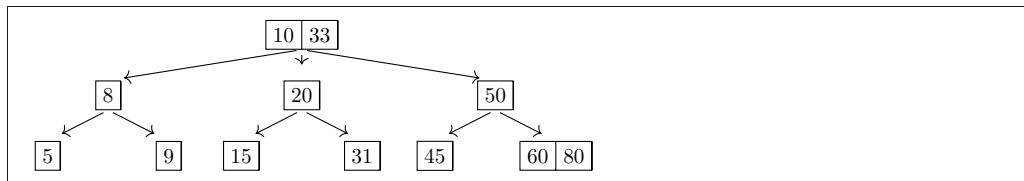


2.6.3 B-Baum

```

\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}

```



```

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

```

2.7 checkbox.sty

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

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

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

490
```

2.8 chomsky-normalform.sty

```

491 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
492 \ProvidesPackage{lehramt-informatik-chomsky-normalform}[2021/03/26
493 Hilfsmakros zum Setzen des Algorithmus zum Erreichen der Chomsky-Normalform]
494
495 \ExplSyntaxOn
496
497 \liLadePakete{typographie}

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

498 \def\liChomskyUeberschrift#1{
499   {
500     \bfseries
501     \sffamily
502     \str_case:nn {#1} {
503       {1} {Elimination~der~$\varepsilon$-Regeln}
504       {2} {Elimination~von~Kettenregeln}
505       {3} {Separation~von~Terminalzeichen}
506       {4} {Elimination~von~mehrelementigen~Nonterminalketten}
507     }
508   }
509 }

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

510 \def\li@chomsky@erklaerung@texte#1{
511   \str_case:nn {#1} {
512     %
513     {1} {
514       Alle~Regeln~der~Form~$A\rightarrow\varepsilon$~werden~eliminiert.~
515       Die~Ersetzung~von~$A$~wird~durch~$\varepsilon$~in~allen~anderen~
516       Regeln~vorweggenommen.
517     }
518     {2} {
519       Jede~Produktion~der~Form~$A\rightarrow B$~mit~$A, B\in S$~wird~
520       als~Kettenregel~bezeichnet.~Diese~tragen~nicht~zur~Produktion~
521       von~Terminalzeichen~bei~und~lassen~sich~ebenfalls~eliminieren.
522     }
523     {3} {
524       Jedes~Terminalzeichen~$\sigma$,~das~in~Kombination~mit~anderen~
525       Symbolen~auftaucht,~wird~durch~ein~neues~Nonterminal~
526       $S_{\sigma}$~ersetzt~und~die~Menge~der~Produktionen~durch~die~
527       Regel~$S_{\sigma}\rightarrow\sigma$~ergänzt.
528     }
529     {4} {
530       Alle~Produktionen~der~Form~
531       $A\rightarrow B_{\{1\}}B_{\{2\}}\dots B_{\{n\}}$~
532       werden~in~die~Produktionen~
533       $A\rightarrow$
534       $A_{\{n-1\}}B_{\{n\}}, A_{\{n-1\}}\rightarrow$
535       $A_{\{n-2\}}B_{\{n-1\}}, \dots, $
536       $A_{\{2\}}\rightarrow B_{\{1\}}B_{\{2\}}$~zerteilt.~
537       Nach~der~Ersetzung~sind~alle~längeren~Nonterminalketten~
538       vollständig~heruntergebrochen~und~die~Chomsky-Normalform~erreicht.
539     }
540   }
541 }

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

```

548 }

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

549 \def\liChomskyUeberErklaerung#1{

550 \liChomskyUeberschrift{#1}\par

551 \liChomskyErklaerung{#1}

552 }

553 \ExplSyntaxOff

554

2.9 cpm.sty

```

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

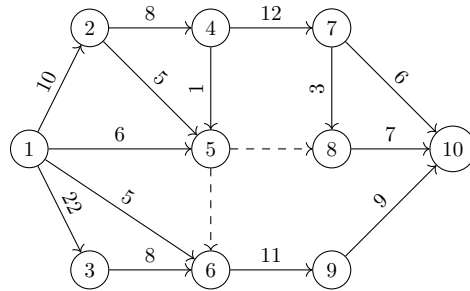
```

2.9.1 Makro-Kürzel

```

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

```



```

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

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

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

```

```

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

```

```

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

```

```

559 \ExplSyntaxOn
560 \NewDocumentCommand { \liCpmEreignis } { 0{} m m m } {
561   \tl_set:Nn \l_name_tl {}
562
563   \keys_define:nn { cpmEreignis } {
564     name .code:n = {\tl_set:Nn \l_name_tl {##1}},
565   }
566
567   \keys_set:nn { cpmEreignis } { #1 }
568
569   \tl_if_empty:NT \l_name_tl {
570     \tl_set:Nn \l_name_tl {#2}
571   }

```

```

572
573 \node[circle,draw] (\l_name_tl) at (#3,#4) {#2};
574 }
575 \ExplSyntaxOff

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

```

2.9.2 Hilfsmakros für Zeitpunkt-Tabelle

```

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

```

```

\liCpmVonZu Let-Abkürzung: \let\ vz=\liCpmVonZu
\liCpmVonZu{1}(2-3):  $l_{(2 \rightarrow 3)}$ 
591 \def\liCpmVonZuOhneMathe#1(#2-#3){#1_{\scriptscriptstyle(#2\rightarrow#3)}}
592 \def\liCpmVonZu#1(#2-#3){%
593   \ifmmode%
594     \liCpmVonZuOhneMathe{#1}(#2-#3)%
595   \else%
596     $\liCpmVonZuOhneMathe{#1}(#2-#3)$%
597   \fi%
598 }

```

```

\liCpmVon Let-Abkürzung: \let\ v=\liCpmVon
\liCpmVon{1}(2):  $l_{(\rightarrow 2)}$ 
599 \def\liCpmVonOhneMathe#1(#2){#1_{\scriptscriptstyle(\rightarrow#2)}}
600 \def\liCpmVon#1(#2){%
601   \ifmmode%
602     \liCpmVonOhneMathe{#1}(#2)%
603   \else%
604     $\liCpmVonOhneMathe{#1}(#2)$%
605   \fi%
606 }

```

```

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

```

```

613   \fi%
614 }

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

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

617

```

2.10 cyk-algorithmus.sty

```
618 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
619 \ProvidesPackage{lehramt-informatik-cyk-algorithmus}[2021/06/18 Hilfsmakros
620 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`

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

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

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

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

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

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

```
627 \NewDocumentCommand{ \liWortNichtInSprache } { m O{L(G)} } {
628   \bigskip
629   \noindent
630   $\Rrightarrow #1 \notin #2$
631 }
```

```
632
```


2.11 entwurfsmuster.sty

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

2.11.1 Namensschema der Entwurfsmuster-Makros:

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

2.11.2 Reihenfolge

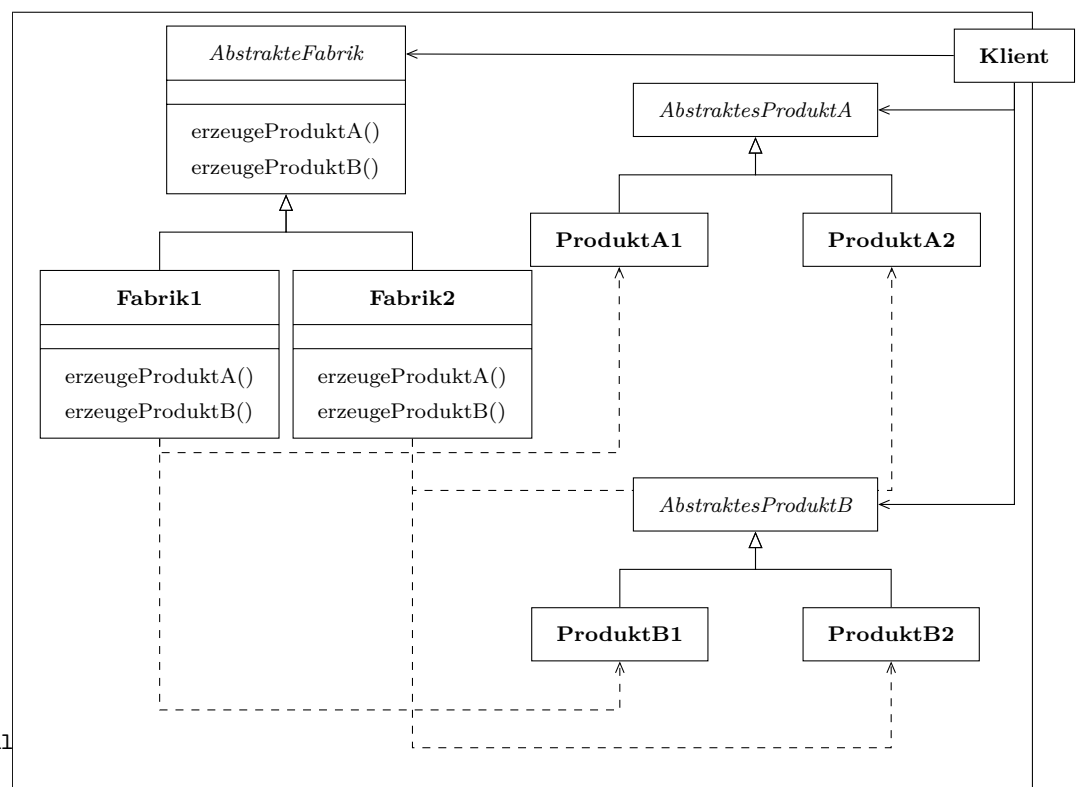
1. Uml: Uml-Klassendiagramm \liEntwurfsEinzelstueckUml
2. Akteure: Akteure, beteiligte Klassen \liEntwurfsEinzelstueckAkteure
3. Code: Allgemeines Code-Beispiel \liEntwurfsEinzelstueckCode
4. ohne: Ohne Suffix, Bündelung der einzelnen Makros eines Entwurfsmusters \liEntwurfsEinzelstueckAkteure

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

\liEntwurfsCodeAllgemein Allgemeine Code-Beispiele zu den UML-Diagrammen und Stellvertretern

```
637 \RequirePackage{lehramt-informatik-syntax}
638 \def\liEntwurfsCode#1#2{
639   \liJavaDatei{entwurfsmuster/#1/allgemein/#2}
640 }
```

2.11.3 Abstrakte Fabrik (Abstract Factory)



```
641 \def\liEntwurfsAbstrakteFabrikUml{
642   \begin{tikzpicture}
643     \umlclass[type=abstract]{AbstrakteFabrik}{-}{-}{
644       erzeugeProduktA()\
645       erzeugeProduktB()\
646     }
```

```

647 \umlclass[below left=1cm and -1.5cm of AbstrakteFabrik]{Fabrik1}{-}{
648     erzeugeProduktA()\
649     erzeugeProduktB()\
650 }
651 \umlclass[below right=1cm and -1.5cm of AbstrakteFabrik]{Fabrik2}{-}{
652     erzeugeProduktA()\
653     erzeugeProduktB()\
654 }
655 \umlVHVinherit{Fabrik1}{AbstrakteFabrik}
656 \umlVHVinherit{Fabrik2}{AbstrakteFabrik}
657
658 \umlsimpleclass[right=3cm of AbstrakteFabrik,type=abstract]{AbstraktesProduktA}
659 \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktA]{ProduktA1}
660 \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktA]{ProduktA2}
661 \umlVHVinherit{ProduktA1}{AbstraktesProduktA}
662 \umlVHVinherit{ProduktA2}{AbstraktesProduktA}
663
664 \umlsimpleclass[above right=0cm and 1cm of AbstraktesProduktA]{Klient}
665
666 \umlsimpleclass[below=4.5cm of AbstraktesProduktA,type=abstract]{AbstraktesProduktB}
667 \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktB]{ProduktB1}
668 \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktB]{ProduktB2}
669 \umlVHVinherit{ProduktB1}{AbstraktesProduktB}
670 \umlVHVinherit{ProduktB2}{AbstraktesProduktB}
671
672 \umlVHVdep[arm1=-1.3cm,arm2=-1cm]{Fabrik1}{ProduktA1}
673 \umlVHVdep[arm2=-1cm]{Fabrik1}{ProduktB1}
674
675 \umlVHVdep[arm1=-1.8cm,arm2=-1.5cm]{Fabrik2}{ProduktA2}
676 \umlVHVdep[arm2=-1.5cm]{Fabrik2}{ProduktB2}
677
678 \umluniassoc[anchor2=25]{Klient}{AbstrakteFabrik}
679 \umlVHuniassoc[arm1=-1cm]{Klient}{AbstraktesProduktA}
680 \umlVHuniassoc{Klient}{AbstraktesProduktB}
681 \end{tikzpicture}
682 }

```

iEntwurfsAbstrakteFabrikCode

```

683 \def\liEntwurfsAbstrakteFabrikCode{
684     \li@EntwurfsCode{abstrakte_fabrik}{Produkte}
685     \li@EntwurfsCode{abstrakte_fabrik}{AbstrakteFabrik}
686     \li@EntwurfsCode{abstrakte_fabrik}{Klient}
687 }

```

\liEntwurfsAbstrakteFabrik

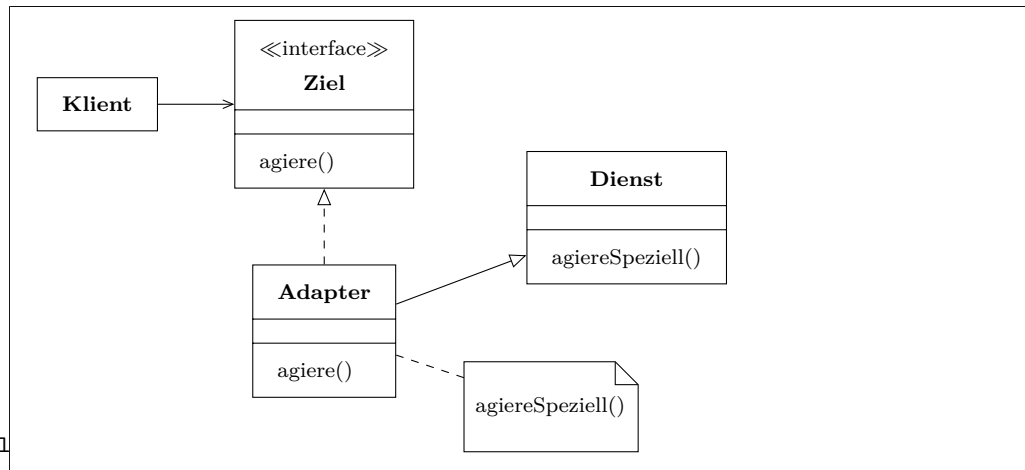
```

688 \def\liEntwurfsAbstrakteFabrik{
689     \liEntwurfsAbstrakteFabrikUml
690     \liEntwurfsAbstrakteFabrikCode
691 }

```

2.11.4 Adapter

\liEntwurfsAdapterUml



```

692 \def\liEntwurfsAdapterUml{
693   \begin{tikzpicture}
694     \umlsimpleclass[x=1,y=3]{Klient}{}{}
695     \umlclass[x=4,y=3,type=interface]{Ziel}{}{agiere()}
696     \umlclass[x=4,y=0]{Adapter}{}{agiere()}
697     \umlclass[x=8,y=1.5]{Dienst}{}{agiereSpeziell()}
698
699     \umlreal{Adapter}{Ziel}
700     \umluniassoc{Klient}{Ziel}
701     \umlinherit{Adapter}{Dienst}
702
703     \umlnote[x=7,y=-1,width=2cm]{Adapter}{agiereSpeziell()}
704   \end{tikzpicture}
705   \footcite[so ähnlich wie GoF]{wiki:adapter}
706 }

```

\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.

```

707 \def\liEntwurfsAdapterAkteure{
708   \begin{description}
709
710     \item[Ziel (Target)]
711
712     Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.
713
714     \item[Klient (Client)]
715
716     Der Klient nutzt Dienste über inkompatible Schnittstellen und greift
717     dabei auf adaptierte Schnittstellen zurück.
718
719     \item[Dienst (Adaptee)]
720
721     Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest
722     definierter Schnittstelle an.
723
724     \item[Adapter]
725
726     Der Adapter adaptiert die Schnittstelle des Dienstes auf die
727     Schnittstelle zum Klienten.\footcite{wiki:adapter}

```

```

728
729 \end{description}
730 }

```

\liEntwurfsAdapterCode

```

731 \def\liEntwurfsAdapterCode{
732 \li@EntwurfsCode{adapter}{Dienst}
733 \li@EntwurfsCode{adapter}{Ziel}
734 \li@EntwurfsCode{adapter}{Adapter}
735 \li@EntwurfsCode{adapter}{Klient}
736 }

```

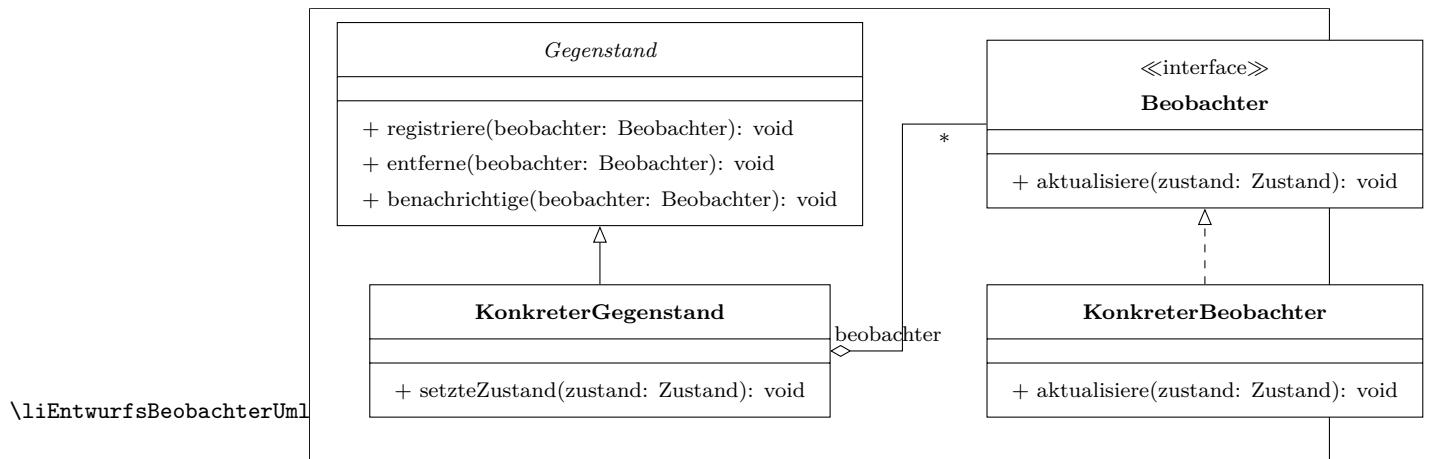
\liEntwurfsAdapter

```

737 \def\liEntwurfsAdapter{
738 \liEntwurfsAdapterUml
739 \liEntwurfsAdapterAkteure
740 \liEntwurfsAdapterCode
741 }

```

2.11.5 Beobachter (Observer)



```

742 \def\liEntwurfsBeobachterUml{
743 \begin{tikzpicture}
744 \umlclass[x=0,y=0,type=abstract]{Gegenstand}{-}{
745 + registriere(beobachter: Beobachter): void\\
746 + entferne(beobachter: Beobachter): void\\
747 + benachrichtige(beobachter: Beobachter): void\\
748 }
749 \umlclass[x=0,y=-3]{KonkreterGegenstand}{-}{
750 + setzteZustand(zustand: Zustand): void
751 }
752 \umlinherit{KonkreterGegenstand}{Gegenstand}
753
754 \umlclass[x=8,y=0,type=interface]{Beobachter}{-}{
755 + aktualisiere(zustand: Zustand): void
756 }
757 \umlclass[x=8,y=-3]{KonkreterBeobachter}{-}{
758 + aktualisiere(zustand: Zustand): void
759 }
760 \umlreal{KonkreterBeobachter}{Beobachter}
761
762 \umlHVHaggreg[arg1=beobachter,pos1=0.8,mult2=*,pos2=2.5]
763 {KonkreterGegenstand}{Beobachter}
764 \end{tikzpicture}
765 }

```

\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.

```

766 \def\liEntwurfsBeobachterAkteure{
767   \begin{description}
768     \item[Gegenstand / Subjekt (Subject / Observable)]
769
770     Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also
771     „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne
772     deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An-
773     und Abmeldung von Beobachtern und eine Schnittstelle zur
774     Benachrichtigung von Beobachtern über Änderungen an.\footcite[Seite
775     251]{gof}
776
777     \item[Beobachter (Observer)]
778
779     Die Beobachter (auf Englisch auch subscriber, also „Abonnent“,
780     genannt) definieren eine Aktualisierungsschnittstelle.
781
782     \item[konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)]
783
784     Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert
785     den relevanten Zustand und benachrichtigt alle Beobachter bei
786     Zustandsänderungen über deren Aktualisierungsschnittstelle. Es
787     verfügt über eine Schnittstelle zur Erfragung des aktuellen
788     Zustands.
789
790     \item[Konkrete Beobachter (ConcreteObserver)]
791
792     Konkrete Beobachter verwalten die Referenz auf ein konkretes
793     Subjekt, dessen Zustand sie beobachten und speichern und dessen
794     Zustand konsistent ist. Sie implementieren eine
795     Aktualisierungsschnittstelle unter Verwendung der
796     Abfrageschnittstelle des konkreten Subjekts.
797     \footcite{wiki:beobachter}
798   \end{description}
799 }
```

\liEntwurfsBeobachterCode

```

800 \def\liEntwurfsBeobachterCode{
801   \li@EntwurfsCode{beobachter}{Gegenstand}
802   \li@EntwurfsCode{beobachter}{KonkreterGegenstand}
803   \li@EntwurfsCode{beobachter}{Beobachter}
804   \li@EntwurfsCode{beobachter}{KonkreterBeobachterA}
805   \li@EntwurfsCode{beobachter}{KonkreterBeobachterB}
```

```

806 \li@EntwurfsCode{beobachter}{Klient}
807 }

```

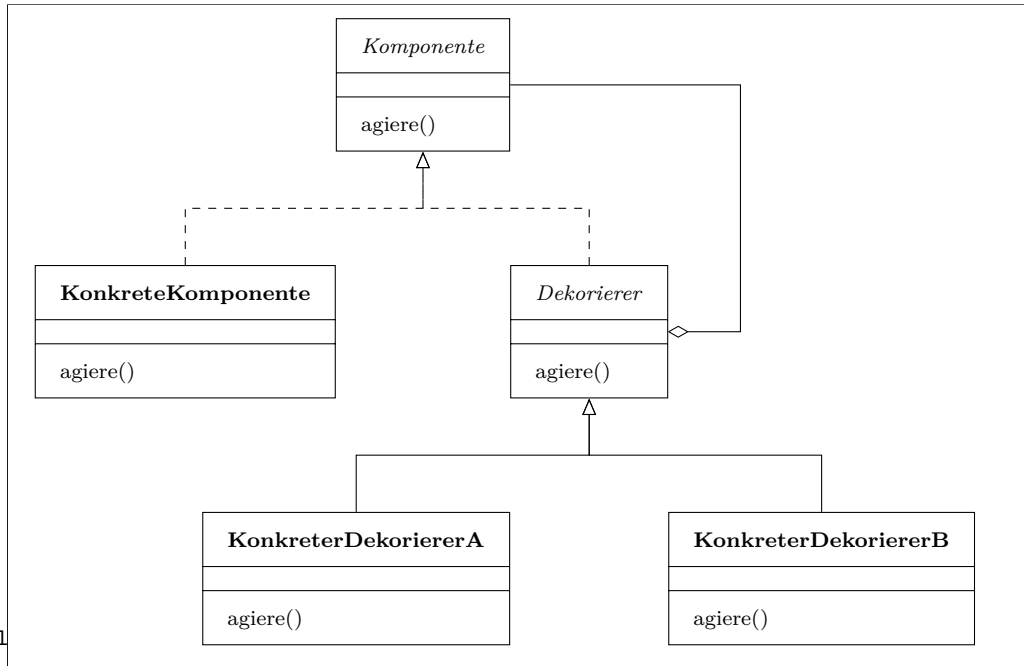
\liEntwurfsBeobachter

```

808 \def\liEntwurfsBeobachter{
809 \liEntwurfsBeobachterUml
810 \liEntwurfsBeobachterAkteure
811 \liEntwurfsBeobachterCode
812 }

```

2.11.6 Dekorierer (Decorator)



\liEntwurfsDekoriererUml

```

813 \def\liEntwurfsDekoriererUml{
814 \begin{tikzpicture}
815 \umlclass[type=abstract]{Komponente}{-}{agiere()}
816 \umlclass[below left=1.5cm and 0cm of Komponente]{KonkreteKomponente}{-}{agiere()}
817 \umlclass[below right=1.5cm and 0cm of Komponente,type=abstract]{Dekorierer}{-}{agiere()}
818
819 \umlVHVreal{KonkreteKomponente}{Komponente}
820 \umlVHVreal{Dekorierer}{Komponente}
821
822 \umlclass[below left=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererA}{-}{agiere()}
823 \umlclass[below right=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererB}{-}{agiere()}
824
825 \umlVHVinherit{KonkreterDekoriererA}{Dekorierer}
826 \umlVHVinherit{KonkreterDekoriererB}{Dekorierer}
827
828 \umlHVVaggreg[arm1=2cm]{Dekorierer}{Komponente}
829 \footcite{wiki:dekorierer}
830 \end{tikzpicture}
831 }

```

\liEntwurfsDekoriererCode

```

832 \def\liEntwurfsDekoriererCode{
833 \li@EntwurfsCode{dekorierer}{Komponente}
834 \li@EntwurfsCode{dekorierer}{KonkreteKomponente}
835 \li@EntwurfsCode{dekorierer}{Dekorierer}
836 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererA}
837 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererB}
838 \li@EntwurfsCode{dekorierer}{Klient}
839 }

```

\liEntwurfsDekorierer

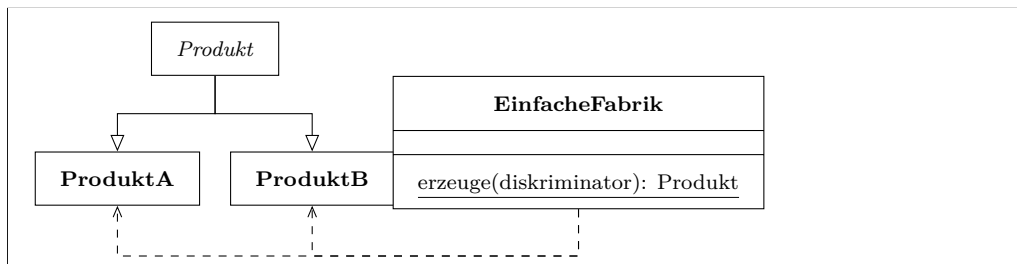
```

840 \def\liEntwurfsDekorierer{
841   \liEntwurfsDekoriererUml
842   \liEntwurfsDekoriererAkteure
843   \liEntwurfsDekoriererCode
844 }

```

2.11.7 Einfache Fabrik (Simple Factory)

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



```

845 \def\liEntwurfsEinfacheFabrikUml{
846   \begin{tikzpicture}
847     \umlsimpleclass[type=abstract]{Produkt}
848     \umlsimpleclass[below left=1cm and -0.65cm of Produkt]{ProduktA}
849     \umlsimpleclass[below right=1cm and -0.65cm of Produkt]{ProduktB}
850     \umlVHVinherit{Produkt}{ProduktA}
851     \umlVHVinherit{Produkt}{ProduktB}
852     \umlclass[below right=0cm and 1.5cm of Produkt]{EinfacheFabrik}{
853     }{
854       \umlstatic{erzeuge(diskriminator): Produkt}\\
855     }
856     \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktA}
857     \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktB}
858   \end{tikzpicture}
859 }

```

\liEntwurfsEinfacheFabrikAkteure Quelle: frei überstetzt aus GoF

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

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

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

```

860 \def\liEntwurfsEinfacheFabrikAkteure{
861   \begin{description}
862     \item[EinfacheFabrik]
863
864     Eine Klasse mit einer Erzeugungsmethode, die über eine größere
865     Bedingung verschiedene Objekt instanziiert.
866
867     \item[Produkt]
868
869     Eine abstrakte Klasse, die von den konkreten Produkten geerbt wird.
870
871     \item[KonkretesProdukt]
872
873     Ein konkretes Produkt, das von der einfachen Fabrik erzeugt wird.
874   \end{description}
875 }

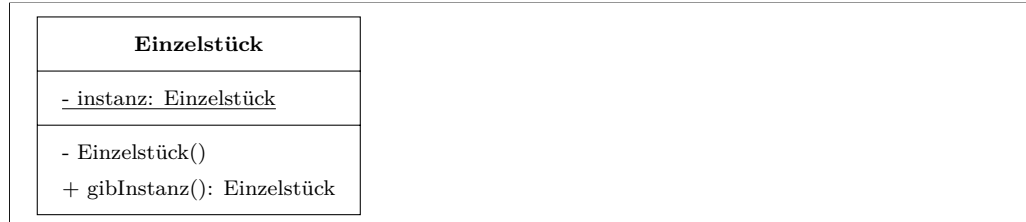
```

\liEntwurfsEinfacheFabrik

```
876 \def\liEntwurfsEinfacheFabrik{
877   \liEntwurfsEinfacheFabrikUml
878   \liEntwurfsEinfacheFabrikAkteure
879 }
```

2.11.8 Einzelstück (Singleton)

\liEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia



```
880 \def\liEntwurfsEinzelstueckUml{
881   \begin{tikzpicture}
882     \umlclass{Einzelstück}{
883       \umlstatic{- instanz: Einzelstück}\\
884     }{
885       - Einzelstück()\\
886       + gibInstanz(): Einzelstück
887     }
888   \end{tikzpicture}
889 }
```

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.

```
890 \def\liEntwurfsEinzelstueckAkteure{
891   \begin{description}
892     \item[Einzelstück (Singleton)]
893
894     stellt eine statische Methode bereit, mit deren Hilfe die Klienten
895     nur auf eine einzige Instanz der Klasse zugreifen können.
896   \end{description}
897 }
```

\liEntwurfsEinzelstueckCode

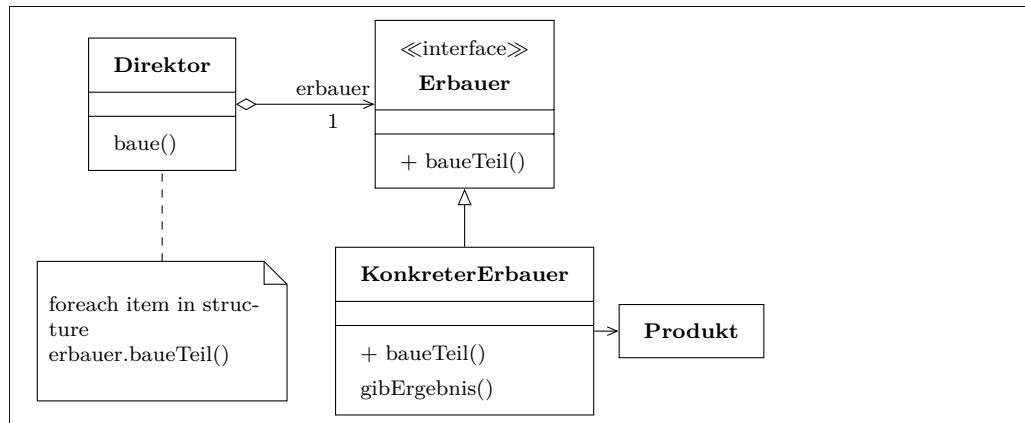
```
898 \def\liEntwurfsEinzelstueckCode{
899   \li@EntwurfsCode{einzelstueck}{Einzelstueck}
900 }
```

\liEntwurfsEinzelstueck

```
901 \def\liEntwurfsEinzelstueck{
902   \liEntwurfsEinzelstueckUml
903   \liEntwurfsEinzelstueckAkteure
904   \liEntwurfsEinzelstueckCode
905 }
```

2.11.9 Erbauer (Builder)

\liEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

906 \def\liEntwurfsErbauerUml{
907   \begin{tikzpicture}
908     \umlsimpleclass[x=7,y=0]{Produkt}{}{}
909     \umlclass[x=4,y=3,type=interface]{Erbauer}{}{+ baueTeil()}
910     \umlclass[x=4,y=0]{KonkreterErbauer}{}{
911       + baueTeil()\
912       gibErgebnis()}
913     \umlclass[x=0,y=3]{Direktor}{}{baue()}
914
915     \umluniagg[arg2=erbauer,pos2=0.7,mult2=1]{Direktor}{Erbauer}
916     \umluniassoc{KonkreterErbauer}{Produkt}
917     \umlinherit{KonkreterErbauer}{Erbauer}
918
919     \umlnote[x=0,y=0,width=3cm]{Direktor}{
920       foreach item in structure\
921       erbauer.baueTeil()
922     }
923   \end{tikzpicture}
924   \footcite{wiki:erbauer}
925 }

```

\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.

```

926 \def\liEntwurfsErbauerAkteure{
927   \begin{description}
928     \item[Erbauer]
929
930     Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der
931     Teile eines komplexen Objektes.
932
933     \item[KonkreterErbauer]
934
935     Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch
936     Implementierung der Schnittstelle. Außerdem definiert und verwaltet er

```

```

937     die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine
938     Schnittstelle zum Auslesen des Produkts.
939
940     \item[Direktor]
941
942     Der Direktor konstruiert ein komplexes Objekt unter Verwendung der
943     Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer
944     zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder
945     benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom
946     Klienten.
947
948     \item[Produkt]
949
950     Das Produkt repräsentiert das zu konstruierende komplexe Objekt.
951     \footcite{wiki:erbauer}
952 \end{description}
953 }

```

\liEntwurfsErbauer

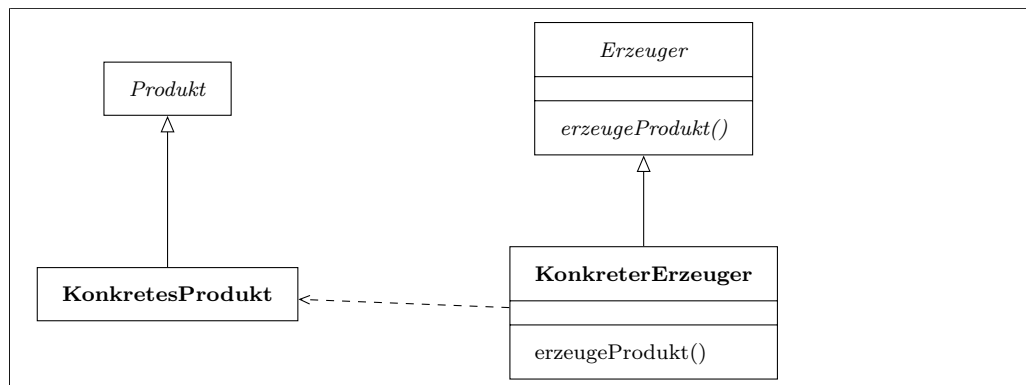
```

954 \def\liEntwurfsErbauer{
955   \liEntwurfsErbauerUml
956   \liEntwurfsErbauerAkteure
957 }

```

2.11.10 Fabrikmethode (Factory Method)

\liEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

958 \def\liEntwurfsFabrikmethodeUml{
959   \begin{tikzpicture}
960     \umlsimpleclass[type=abstract]{Produkt}
961     \umlsimpleclass[below=2cm of Produkt]{KonkretesProdukt}
962     \umlinherit{KonkretesProdukt}{Produkt}
963
964     \umlclass[type=abstract,right=4cm of Produkt]{Erzeuger}{\{
965       \textit{erzeugeProdukt()}\}
966     }
967     \umlclass[below=1.2cm of Erzeuger]{KonkreterErzeuger}{\{
968       erzeugeProdukt()
969     }
970     \umlinherit{KonkreterErzeuger}{Erzeuger}
971
972     \umldep{KonkreterErzeuger}{KonkretesProdukt}
973   \end{tikzpicture}
974 }

```

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).

```

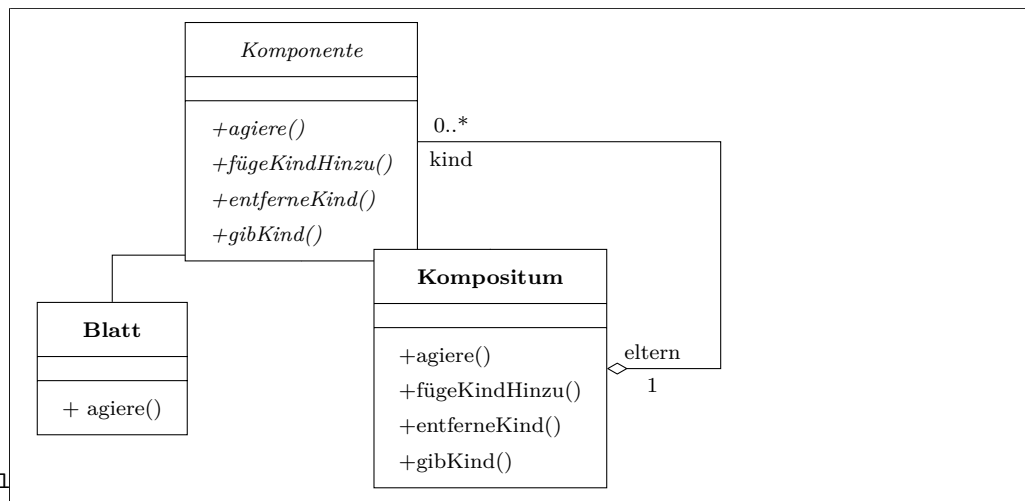
975 \def\liEntwurfsFabrikmethodeAkteure{
976   \begin{description}
977     \item[Produkt]
978
979     Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das
980     zu erzeugende Produkt.
981
982     \item[KonkretesProdukt]
983
984     KonkretesProdukt implementiert die Produkt-Schnittstelle.
985
986     \item[Erzeuger]
987
988     Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt
989     zu erzeugen und kann eine Default-Implementierung beinhalten.
990
991     \item[KonkreterErzeuger]
992
993     KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm
994     entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den
995     Konstruktor einer konkreten Produkt-Klasse aufruft).
996
997     \footcite{wiki:fabrikmethode}
998   \end{description}
999 }
```

\liEntwurfsFabrikmethode

```

1000 \def\liEntwurfsFabrikmethode{
1001   \liEntwurfsFabrikmethodeUml
1002   \liEntwurfsFabrikmethodeAkteure
1003 }
```

2.11.11 Kompositum (Composite)



\liEntwurfsKompositumUml

```

1004 \def\liEntwurfsKompositumUml{
1005   \begin{tikzpicture}
1006     \umlclass[x=2.5,y=3,type=abstract]{Komponente}{-}{-}{
1007       \textit{+agiere()}\
1008       \textit{+fügeKindHinzu()}\
1009       \textit{+entferneKind()}\
1010       \textit{+gibKind()}
1011     }
1012     \umlclass[x=0]{Blatt}{-}{+ agiere()}
1013     \umlclass[x=5]{Kompositum}{-}{-}{
1014       +agiere()\
1015       +fügeKindHinzu()\
1016       +entferneKind()\
1017       +gibKind()
1018     }
1019
1020     \umlVHVinherit{Kompositum}{Komponente}
1021     \umlVHVinherit{Blatt}{Komponente}
1022     \umlHVHaggreg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,]
1023   \end{tikzpicture}
1024 }

```

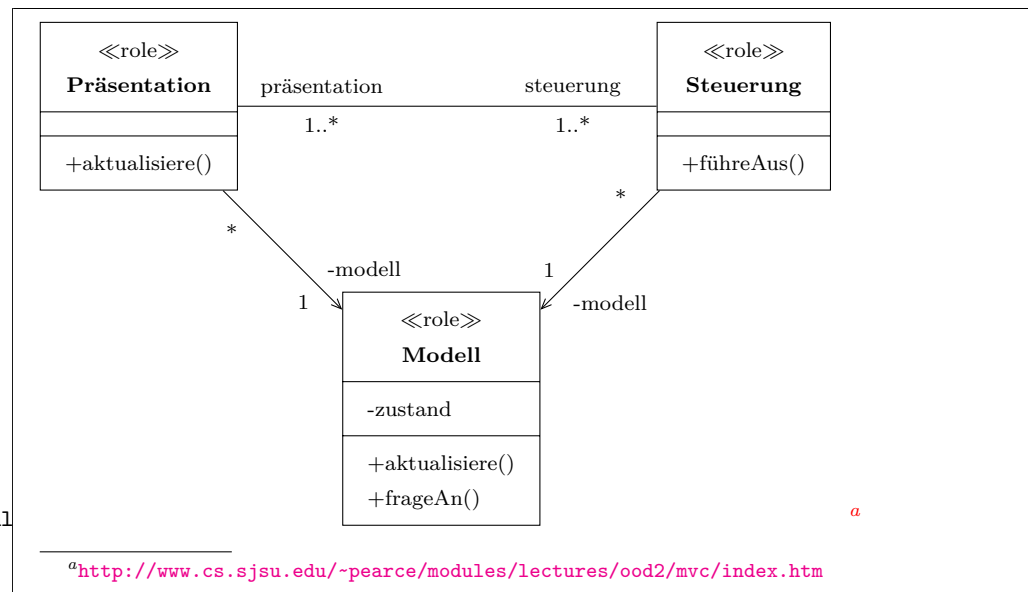
\liEntwurfsFabrikmethode

```

1025 \def\liEntwurfsKompositum{
1026   \liEntwurfsKompositumUml
1027   \liEntwurfsKompositumAkteure
1028 }

```

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



```

1029 \def\liEntwurfsModellPraesentationSteuerungUml{
1030   \begin{tikzpicture}
1031     \umlclass[x=-4cm,type=role]{Präsentation}{-}{+aktualisiere()}
1032     \umlclass[x=4cm,type=role]{Steuerung}{-}{+führeAus()}
1033     \umlclass[y=-4cm,type=role]{Modell}{-}{
1034       -zustand
1035     }{
1036       +aktualisiere()\
1037       +frageAn()
1038     }
1039
1040     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
1041     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}

```

```

1042 \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
1043 \end{tikzpicture}
1044 \liFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
1045 }

```

ModellPraesentationSteuerung

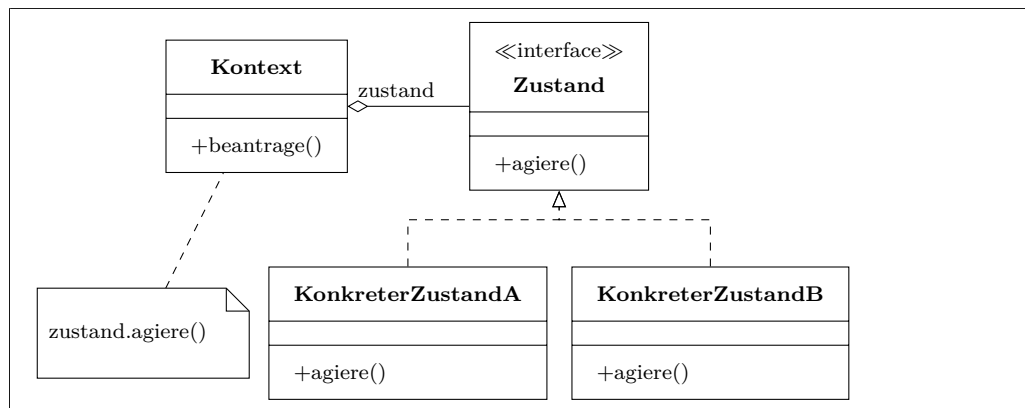
```

1046 \def\liEntwurfs{
1047 \liEntwurfsModellPraesentationSteuerungUml
1048 \liEntwurfsModellPraesentationSteuerungAkteure
1049 }

```

2.11.13 Zustand (State)

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



```

1050 \def\liEntwurfsZustandUml{
1051 \begin{tikzpicture}
1052 \umlcclass[x=-1,y=3]{Kontext}{+beantrage()}
1053 \umlcclass[x=3,y=3,type=interface]{Zustand}{+agiere()}
1054 \umlcclass[x=1,y=0]{KonkreterZustandA}{+agiere()}
1055 \umlcclass[x=5,y=0]{KonkreterZustandB}{+agiere()}
1056
1057 \umlVHVreal{KonkreterZustandA}{Zustand}
1058 \umlVHVreal{KonkreterZustandB}{Zustand}
1059
1060 \umlagg[reg[arg=zustand,pos=0.4]{Kontext}{Zustand}
1061
1062 \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
1063 \end{tikzpicture}
1064 }

```

\liEntwurfsZustandAkteure Quelle: Deutsche Wikipedia

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

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

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

```

1065 \def\liEntwurfsZustandAkteure{
1066 \begin{description}
1067 \item[Kontext (Context)]
1068
1069 definiert die clientseitige Schnittstelle und verwaltet die separaten
1070 Zustandsklassen.
1071

```

```

1072 \item[State (Zustand)]
1073
1074 definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
1075 implementiert gegebenenfalls ein Standardverhalten.
1076
1077 \item[KontreterZustand (ConcreteState)]
1078
1079 implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
1080 verbunden ist.
1081 \end{description}
1082 }

\liEntwurfsZustand
1083 \def\liEntwurfsZustand{
1084 \liEntwurfsZustandUml
1085 \liEntwurfsZustandAkteure
1086 }

1087

```

2.12 er.sty

```
1088 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1089 \ProvidesPackage{lehramt-informatik-er}[2020/06/13 Zum Setzen von
1090 ER-Diagrammen]
```

```
1091 \RequirePackage{tikz-er2}
1092 \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);

```

```
1093 \RequirePackage{soul}
```

```
1094 \RequirePackage{fontawesome}
```

2.12.3 Makro-Kürzel

```

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

```

```
1095 \ExplSyntaxOn
```

```
\liErEntity
```

```
1096 \def\liErEntity#1{\textbf{#1}}
```

```
\liErRelationship
```

```
1097 \def\liErRelationship#1{\ul{#1}}
```

```
\liErAttribute
```

```
1098 \def\liErAttribute#1{\emph{#1}}
```

```
\liErMpEntity mp = marginpar
```

Let-Abkürzung: \let\e=\liErMpEntity

```

1099 \def\liErMpEntity#1{
1100   \liErEntity{#1}
1101   \marginpar{
1102     \liErEntity{\tiny\faSquareO{~E:~#1}
1103   }
1104 }

```

□

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

```

1105 \def\liErMpRelationship#1{
1106   \liErRelationship{#1}
1107   \marginpar{
1108     \liErRelationship{\tiny\faGg{~R:~#1}
1109   }
1110 }

```

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

```

1111 \def\liErMpAttribute#1{
1112   \liErAttribute{#1}
1113   \marginpar{
1114     \liErAttribute{\tiny\faCircleThin{~A:~#1}
1115   }
1116 }

```



```

\liErDatenbankName Let-Abkürzung: \let\d=\liErDatenbankName
                    datenbank name
1117 \def\liErDatenbankName#1{
1118   {
1119     \footnotesize\texttt{(#1)}
1120   }
1121 }

1122 \ExplSyntaxOff
1123

```

2.13 formale-sprachen.sty

```

1124 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1125 \ProvidesPackage{lehramt-informatik-formale-sprachen}[2021/02/21 Hilfsmakros
1126 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]

1127 \directlua{
1128   formale_sprachen = require('lehramt-informatik-formale-sprachen')
1129 }

1130 \RequirePackage{hyperref}

1131 \liLadePakete{mathe,typographie}

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

1132 \def\liMengeOhneMathe#1{\{ #1 \}}
1133 \def\liMenge#1{%
1134   \ifmode%
1135     \liMengeOhneMathe{#1}%
1136   \else%
1137     $\liMengeOhneMathe{#1}$%
1138   \fi%
1139 }

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

1140 \def\liEpsilon{$\varepsilon$}

\liPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung

1141 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1142 \def\liPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1143 \def\liPotenzmenge#1{$\liPotenzmengeOhneMathe{#1}$}

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

1144 \let\liZustandsmengeOhneMathe=\liPotenzmengeOhneMathe
1145 \let\liZustandsmenge=\liPotenzmenge

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

1146 \def\liUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1147 \def\liUeberfuehrungsFunktion#1{
1148   \ifmode
1149     \liUeberfuehrungsFunktionOhneMathe{#1}
1150   \else
1151     $\liUeberfuehrungsFunktionOhneMathe{#1}$
1152   \fi
1153 }

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

1154 \def\liAlphabet#1{$\Sigma = \{ #1 \}$}

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

1155 \def\liBandAlphabet#1{$\Gamma = \Sigma \cup \{ #1 \}$}

\liZustandsBuchstabe

1156 \def\liZustandsBuchstabe{z}

\liZustandsBuchstabeGross

1157 \def\liZustandsBuchstabeGross{Z}

```

```

\liZustandsmengeNr
1158 \def\zustandsnamens@liste#1#2{
1159   $
1160   \{
1161     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1162   \}
1163   $
1164 }
1165 \def\liZustandsmengeNr#1{\zustandsnamens@liste{\liZustandsBuchstabe}{#1}}

\liZustandsmengeNrGross
1166 \def\liZustandsmengeNrGross#1{\zustandsnamens@liste{\liZustandsBuchstabeGross}{#1}}

\liZustandsname \liZustandsname{1}: $z_1$
1167 \def\liZustandsname#1{\liZustandsBuchstabe_#1$}

\liZustandsnameGross \liZustandsnameGross{1}: $Z_1$
1168 \def\liZustandsnameGross#1{\liZustandsBuchstabeGross_#1$}

\liAbleitung \liAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1169 \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}

1170 \NewDocumentEnvironment { liProduktionsRegeln }
1171 { 0{P} +b }
1172 {
1173   \liGeschweifteKlammern{#1}
1174   {
1175     \begin{align*}
1176       \directlua{formale_sprachen.produktions_regeln('#2')}
1177     \end{align*}
1178   }{-0.2cm}{-1.5cm}
1179 } {}

\liProduktionen \liProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
1180 \def\liProduktionen#1{
1181   \liMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
1182 }

\liZustandsnameTiefgestelltt Automatisch tiefgestellte Nummerierung \z1
Let-Abkürzung: \let\z=\liZustandsnameTiefgestelltt
1183 \def\liZustandsnameTiefgestelltt#1{
1184   \ifmmode
1185     \liZustandsBuchstabe\sb{#1}
1186   \else
1187     $\liZustandsBuchstabe\sb{#1}$
1188   \fi
1189 }

1190 \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}

```

```

1191 \NewDocumentCommand{ \liAusdruck } { O{L} m m } {
1192   $
1193   \tl_if_empty:nTF {#1} {} {#1 =}
1194   \{
1195     \, #2 \,
1196     |
1197     \, #3 \,
1198   \}$
1199 }
1200 \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

1201 \def\liFlaci#1{%
1202   \par
1203   {%
1204     \scriptsize
1205     Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1206     Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1207     Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1208     \href{https://flaci.com/#1}{flaci.com/#1}%
1209   }%
1210   \par
1211 }

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

1212 \ExplSyntaxOn
1213 \NewDocumentCommand {\liGrammatik} { O{G} m } {
1214   \tl_set:Nn \l_variablen_tl {V}
1215   \tl_set:Nn \l_alphabet_tl {\Sigma}
1216   \tl_set:Nn \l_produktionen_tl {P}
1217   \tl_set:Nn \l_start_tl {S}
1218
1219   \keys_define:nn { grammatik } {
1220     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\liMenge{##1}}},
1221     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\liMenge{##1}}},
1222     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\liProduktionen{##1}}},
1223     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1224   }
1225
1226   \keys_set:nn { grammatik } { #2 }
1227
1228   $#1 = (
1229     \l_variablen_tl,
1230     \l_alphabet_tl,
1231     \l_produktionen_tl,
1232     \l_start_tl
1233   )$
1234 }
1235 \ExplSyntaxOff

```

2.14 formatierung.sty

```
1237 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1238 \ProvidesPackage{lehramt-informatik-formatierung}[2020/11/27]
```

2.14.1 Schriftarten / Typographie

```
1239 \RequirePackage{mathpazo}
1240 \RequirePackage[no-math]{fontspec}
1241 \setmainfont{texgyrepagella}
1242 \setsansfont{QTAncientOlive}
1243 \RequirePackage{sectsty}
1244 \allsectionsfont{\fontspec{QTAncientOlive}}
```

2.14.2 Farben

```
1245 \RequirePackage{xcolor}
1246 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

2.14.3 Überschriften

```
1247 \RequirePackage{titlesec}
1248 \titleformat{\chapter}[display]{\sffamily\bfseries}{\Opt}{\LARGE}
1249 \titlespacing{\chapter}{Opt}{Opt}{*1}
1250 \titleformat{\paragraph}[hang]{\sffamily\normalsize\bfseries}{\theparagraph}{1em}{}
1251 \setcounter{secnumdepth}{0}
```

2.14.4 Listen

```
1252 \RequirePackage{paralist}
1253 \renewcommand\labelitemi{-}
1254 \renewcommand\labelitemii{-}
1255 \renewcommand\labelitemiii{-}
1256 \renewcommand\labelitemiv{-}
1257 % Counter: enumi enumii enumiii enumiv
1258 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1259 \renewcommand{\labelenumi}{(\alph{enumi})}
1260 \renewcommand{\labelenumii}{(\roman{enumii})}
```

2.14.5 Kasten

```
1261 \RequirePackage{mdframed}
1262 \mdfsetup{backgroundcolor=infogray}
```

liKasten

```
1263 \NewDocumentEnvironment { liKasten }{ } {
1264   \begin{mdframed}[backgroundcolor=white!0]
1265 } {
1266   \end{mdframed}
1267 }
```

2.14.6 Header

```
1268 \RequirePackage{fancyhdr}
1269 \fancyhead[L,C,R]{}
1270 \fancyfoot[L]{}
1271 \fancyfoot[C]{}
1272 \fancyfoot[R]{\thepage}
1273 \pagestyle{fancy}
1274 \renewcommand{\headrulewidth}{Opt}
1275 \renewcommand{\footrulewidth}{Opt}

1276
```

2.15 gantt.sty

```

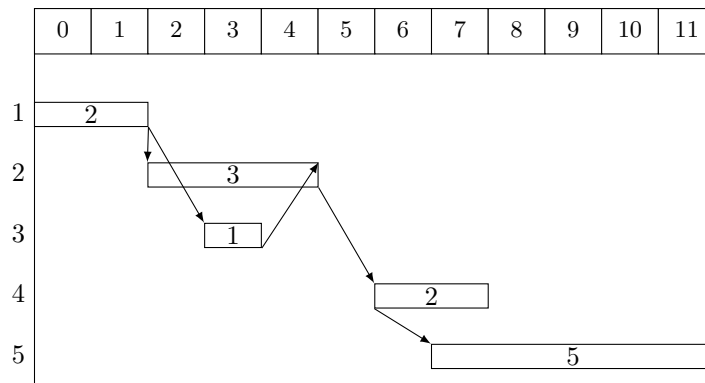
1277 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1278 \ProvidesPackage{lehramt-informatik-gantt}[2020/09/05]

\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}

```



```

1279 \RequirePackage{tikz-uml}
1280 \RequirePackage{pgfgantt}
1281 \setganttlinklabel{f-s}{}
1282 \setganttlinklabel{s-s}{}
1283 \setganttlinklabel{f-f}{}
1284 \setganttlinklabel{s-f}{}

1285

```

2.16 grafik.sty

```
1286 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1287 \ProvidesPackage{lehramt-informatik-grafik}[2020/12/27 Lädt das Paket
1288 TikZ, um Grafiken zeichnen zu können.]
1289 \RequirePackage{tikz}
1290
```

2.17 graph.sty

```

1291 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1292 \ProvidesPackage{lehramt-informatik-graph}[2020/06/09]
1293 \RequirePackage{tikz}

```

Für die die Adjazenzliste (xrightarrow)

```

1294 \RequirePackage{amsmath}

```

Für Adjazenz-Matrix

```

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

```

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

```

1295 \RequirePackage{blkarray}
1296 \usetikzlibrary{arrows.meta}

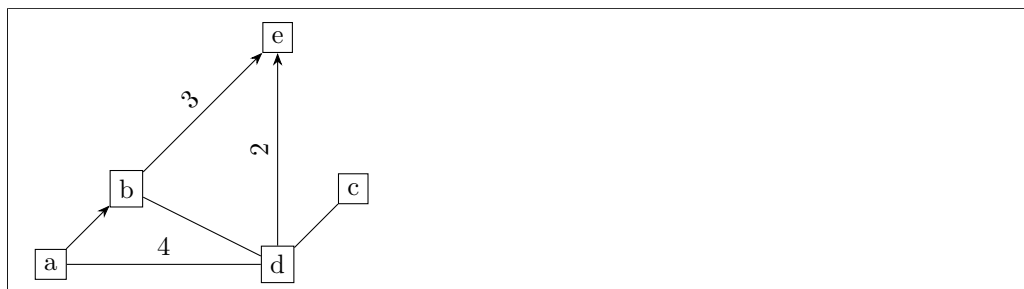
```

```

\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}

```




```

1297 \tikzset{
1298   li graph/.style={
1299     every node/.style={
1300       rectangle,
1301       draw,
1302     },
1303     every edge/.style={
1304       >={Stealth[black]},
1305       draw,
1306     },
1307     every edge/.append style={
1308       every node/.style={
1309         sloped,
1310         auto,
1311       }
1312     }
1313   },
1314   li markierung/.style={
1315     ultra thick,
1316   }
1317 }

```

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}

```

```

1318 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1319

```

2.18 hanoi.sty

```

1320 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1321 \ProvidesPackage{lehramt-informatik-hanoi}[2020/12/19 Paket zum Setzen
1322 von Hanoi-Grafiken]

    Quelle: https://kleinco.de/latex-snippets/7/tikz-towers-of-hanoi-illustration-for-lat
1323 \RequirePackage{tikz}
1324 \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}

1325 \def\li@mset #1[#2]=#3{%
1326 \expandafter\xdef\csname #1#2\endcsname{#3}
1327 }
1328 \def\li@mget #1[#2]{%
1329 \csname #1#2\endcsname
1330 }
1331 \def\li@minc #1[#2]+=#3{%
1332 \pgfmathparse{\li@mget #1[#2]+#3}%
1333 \li@mset #1[#2]=\pgfmathresult
1334 }
1335
1336 \def\liHanoi#1#2{
1337   \edef\li@numdiscs{#1}
1338   \def\li@sequence{#2}
1339   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1340     % init colors
1341     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purp
1342     \li@mset col[\j]={\c};
1343     % draw poles and init pole counters
1344     \foreach \j in {1,2,3}{
1345       \li@mset pos[\j]=0
1346       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1347     }
1348     % draw base
1349     \draw (.5,-.5) -- +(3,0);
1350     % draw discs
1351     \foreach[count=\k] \i/\j in \li@sequence{
1352       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*
1353       \li@minc pos[\j]+=.5}
1354     }
1355   \end{tikzpicture}
1356 }

1357

```

2.19 klassen-konfiguration-aufgabe.sty

```
1358 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1359 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-aufgabe}[2021/09/01 Die
1360 Klasse Aufgabe konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]

    Formatierung muss vor literatur sein, sonst Option clash
1361 \liLadePakete{
1362   formatierung,
1363   abmessung,
1364   literatur-dummy,
1365   makros,
1366   aufgaben-metadaten,
1367   kopf-fusszeilen,
1368   mathe
1369 }
1370
1371 % Komischer Option-Clash deshalb ganz am Ende, für die Silbentrennung
1372 \RequirePackage[ngerman]{babel}
1373 \ExplSyntaxOn
1374 \AddToHook{enddocument}{
1375   \_gib_github_url:
1376 }
1377 \ExplSyntaxOff
1378
```

2.20 klassen-konfiguration-examen.sty

```
1379 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1380 \ProvidesPackage{lehramt-informatik-klassen-konfiguration-examen}[2021/09/04 Die
1381 Klasse liSetzeExamen konfigurieren, d. h. Laden von einigen Paketen, Registrierung von Hooks]
```

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

```
1382 \liLadePakete{
1383   formatierung,
1384   literatur-dummy,
1385   makros,
1386   aufgaben-metadaten,
1387   abmessung
1388 }
```

```
1389 \liLadeAllePakete
```

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

```
1390 \RequirePackage[ngerman]{babel}
```

```
1391 \RequirePackage{standalone}
```

```
1392 \ExplSyntaxOn
```

```
\liSetzeExamen
```

```
1393 \def\liSetzeExamen#1#2#3{
1394   \tl_gset:Nn \g_auf_examen_nummer_tl { #1 }
1395   \tl_gset:Nn \g_auf_examen_jahr_tl { #2 }
1396   \tl_gset:Nn \g_auf_examen_monat_tl { #3 }
1397 }
```

```
\liSetzeExamenThemaNr
```

```
1398 \def\liSetzeExamenThemaNr#1{
1399   \tl_gset:Nn \g_auf_examen_thema_nr_tl { #1 }
1400   \section{Thema-Nr.~#1}
1401 }
```

```
\liSetzeExamenTeilaufgabeNr
```

```
1402 \def\liSetzeExamenTeilaufgabeNr#1{
1403   \tl_gset:Nn \g_auf_examen_teilaufgabe_nr_tl { #1 }
1404   \subsection{Teilaufgabe-Nr.~#1}
1405 }
```

```
\liBindeAufgabeEin
```

```
1406 \def\liBindeAufgabeEin#1{
1407   \tl_gset:Nn \g_auf_examen_aufgabe_nr_tl { #1 }
1408   \input{
1409     \LehramtInformatikRepository /
1410     Staatsexamen /
1411     \g_auf_examen_nummer_tl /
1412     \g_auf_examen_jahr_tl /
1413     \g_auf_examen_monat_tl /
1414     \tl_if_empty:NTF \g_auf_examen_thema_nr_tl {} {
1415       Thema - \g_auf_examen_thema_nr_tl /
1416     }
1417     \tl_if_empty:NTF \g_auf_examen_teilaufgabe_nr_tl {} {
1418       Teilaufgabe - \g_auf_examen_teilaufgabe_nr_tl /
1419     }
1420     Aufgabe - \g_auf_examen_aufgabe_nr_tl .tex
1421   }
1422 }
```

```
1423 \ExplSyntaxOff
```

```
1424
```

2.21 komplexitaetstheorie.sty

```
1425 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1426 \ProvidesPackage{lehramt-informatik-komplexitaetstheorie}[2021/07/08 Zum
1427 Setzen von Karp's NP-vollständigen Problemen, Hilfsmakros für die
1428 Polynomialzeitreduktion.]
```

2.21.1 Makro-Kürzel

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

```
1429 \liLadePakete{mathe}
```

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

```
1430 \RequirePackage{mdframed}
```

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

```
1431 \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`

```
1432 \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`

```
1433 \def\liProblemBeschreibung#1#2#3{
1434   \begin{mdframed}[
1435     userdefinedwidth=9cm,
1436     align=center,
1437     backgroundcolor=white!0,
1438   ]
1439   \centerline{\large\liProblemName{#1}}
1440
1441   \medskip
1442
1443   \begin{description}
1444     \item[Gegeben:] #2
1445     \item[Frage:] #3
1446   \end{description}
1447 \end{mdframed}
1448 }
```

```

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

1449 \NewDocumentCommand{ \liPolynomiellReduzierbar } { m O{p} m } {
1450 \begin{displaymath}
1451   \liProblemName{#1}
1452   \preceq_{#2}
1453   \liProblemName{#3}
1454 \end{displaymath}
1455 }

\liProblemVertexCover

1456 \def\liProblemClique{%
1457 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
1458 Mindestgröße  $n$  in einem gegebenen Graphen.
1459 \footcite[Seite 76]{theo:fs:4}
1460 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
1461 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
1462 \footcite{wiki:cliquenproblem}
1463 }

\liProblemVertexCover

1464 \def\liProblemVertexCover{%
1465 %
1466 Das \textbf{Knotenüberdeckungsproblem} (\liProblemName{Vertex Cover})
1467 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
1468 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
1469 \footcite{wiki:knotenueberdeckung}
1470
1471 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
1472 der Knotenmenge gibt, so dass jede Kante des Graphen mit
1473 mindestens einem Knoten aus  $U$  verbunden ist.
1474 \footcite[Seite 78]{theo:fs:4}%
1475 }

\liProblemSubsetSum Kein Karp-21-Problem

1476 \def\liProblemSubsetSum{%
1477 Das \textbf{Teilsommenproblem} (\liProblemName{Subset Sum} oder
1478 \liProblemName{SSP}) ist ein spezielles Rucksackproblem.
1479 \footcite{wiki:teilsommenproblem}
1480 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
1481 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
1482 größer als eine gegebene obere Schranke  $c$  ist.
1483 \footcite[Seite 74]{theo:fs:4}
1484 }

\liProblemSubsetSum Kein Karp-21-Problem

1485 \def\liProblemSat{%
1486 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \liProblemName{Sat}
1487 und \liProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
1488 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
1489 ist.\footcite{wiki:sat} Das Erfüllbarkeitsproblem der
1490 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
1491 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
1492 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
1493 aufgestellt werden.
1494 \footcite[Seite 71]{theo:fs:4}
1495 }

1496

```

2.22 kontrollflussgraph.sty

1497 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1498 \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 = ...}]
```

```

1499 \RequirePackage{tikz}
1500 \usetikzlibrary{positioning}
1501 \tikzset{
1502   li kontrollfluss/.style={
1503     knoten/.style={
1504       circle,
1505       draw
1506     },
1507     usebox/.style={
1508       draw,
1509       rectangle,
1510       font=\scriptsize,
1511       anchor=west,
1512       align=left,
1513     },
1514     bedingung/.style={
1515       midway,
1516       draw=none,
1517       font=\scriptsize
1518     },
1519     knotenbeschriftung/.style={
1520       draw,
1521       rectangle,
1522       midway,
1523       font=\scriptsize
1524     },
1525     wahr/.style={
1526       thick
1527     },
1528     falsch/.style={
1529       dashed
1530     },
1531     every node/.style={
1532       circle,
1533       draw,
1534     },
1535     every edge/.append style={
1536       every node/.style={
1537         draw=none,
1538         bedingung,
1539       }
1540     },
1541     every path/.style={
1542       draw,
1543       ->,
1544     },
1545     every pin/.style={
1546       draw,
1547       dotted,
1548       rectangle,
1549       pin position=right
1550     },
1551     every pin edge/.style={
1552       dotted,
1553       arrows=-,
1554     }
1555   }
1556 }

```

2.22.4 Umgebungen

liKontrollflussgraph

```

1557 \NewDocumentEnvironment { liKontrollflussgraph } { 0{} } {

```



```

1558 \begin{tikzpicture}[
1559     li kontrollfluss,
1560     #1
1561 ]
1562 } {
1563 \end{tikzpicture}
1564 }

```

2.22.5 Makros

`\liAnweisung`

```
1565 \def\liAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}
```

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

```
1566 \def\liBedingung#1#2{\node[bedingung,#1]{\texttt{#2}}}
```

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

```
1567 \def\liBedingungWahr#1{\node[bedingung,#1]{\texttt{[true]}}}
```

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

```
1568 \def\liBedingungFalsch#1{\node[bedingung,#1]{\texttt{[false]}}}
```

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

```
1569 \def\liKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}
```

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

```
1570 \def\liKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw
```

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

```
1571 \ExplSyntaxOn
```

```
1572 \NewDocumentCommand { \liKontrollKnotenPfad } { m }
```

```
1573 {
```

```
1574   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
```

```
1575   \seq_set_map:NNn \l_tmpa_seq \l_tmpa_seq {\liKontrollTextzeileKnoten{##1}}
```

```
1576   \seq_use:Nn \l_tmpa_seq { ~~~~ }
```

```
1577 }
```

```
1578 \ExplSyntaxOff
```

```
1579
```

2.23 kopf-fusszeilen.sty

```
1580 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1581 \ProvidesPackage{lehramt-informatik-kopf-fusszeilen}[2021/08/20 Kopf-
1582 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

1583 \ExplSyntaxOn

1584 \fancyhead{}
1585 \fancyhead[R0,LE]{\scriptsize\LehramtInformatikTitel}
1586 \fancyhead[L0,RE]{\scriptsize\today}
1587 \fancyfoot{}
1588 \fancyfoot[LE,R0]{\thepage}
1589 \fancyfoot[L0,CE]{\scriptsize\LehramtInformatikAutorName}
1590 \fancyfoot[CO,RE]{\scriptsize\LehramtInformatikAutorEmail}
1591 \renewcommand{\headrulewidth}{0.4pt}
1592 \renewcommand{\footrulewidth}{0.4pt}
1593 \setlength{\headwidth}{\textwidth}

1594 \ExplSyntaxOff

1595
```

2.24 literatur-dummy.sty

```
1596 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1597 \ProvidesPackage{lehramt-informatik-literatur-dummy}[2020/11/27]

\literatur

1598 \def\literatur{}

\footcite

1599 % \RequirePackage[stable,multiple]{footmisc}
1600 \NewDocumentCommand{ \footcite } { o m }{}

1601
```

2.25 literatur.sty

```
1602 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1603 \ProvidesPackage{lehramt-informatik-literatur}[2020/11/27]

1604 \RequirePackage{csquotes}
1605 \RequirePackage[
1606   bibencoding=utf8,
1607   citestyle=authortitle,
1608   backend=biber,
1609 ]{biblatex}
1610 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
1611 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
1612 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
1613 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
1614 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
1615 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
1616 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
1617 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
1618 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
1619 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
1620 % To allow footnotes in the heading
1621 \RequirePackage[stable,multiple]{footmisc}

\literatur

1622 \def\literatur{\printbibliography[heading=subbibliography]}

1623
```

2.26 makros.sty

```

1624 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1625 \ProvidesPackage{lehramt-informatik-makros}[2020/11/27 Eine Sammlung von
1626 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
1627 anderen Paket passen]
1628 \RequirePackage{hyperref}
1629 \RequirePackage{graphicx}

Für die Umgebung liQuellen benötigt.
1630 \RequirePackage{paralist}

\inhaltsverzeichnis
1631 \def\inhaltsverzeichnis {
1632   \begin{mdframed}
1633     \begin{group}
1634       \let\clearpage\relax
1635       \tableofcontents
1636     \end{group}
1637   \end{mdframed}
1638 }

\memph \mephm (\marginpar and \emph)
1639 \newcommand{\memph}[1]{\emph{#1}\marginpar{\tiny#1}}

\SLASH
1640 \newcommand\SLASH{\char`\\}

\liPseudoUeberschrift Text, der sich wie eine Überschrift verhält.
1641 \newcommand{\liPseudoUeberschrift}[1]{
1642   \bigskip
1643   \noindent
1644   \textsf{\textbf{#1}}
1645   \noindent
1646 }

\liBeschriftung Ähnlich dem Makro \liPseudoUeberschrift{}. Am Ende des Textes wird ein Doppelpunktzeichen angehängt.
1647 \newcommand{\liBeschriftung}[1]{
1648   \par
1649   \noindent
1650   \medskip
1651   \textbf{#1}:
1652   \medskip
1653   \noindent
1654 }

\hinweis
1655 \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.
1656 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}

liEinbettung
1657 \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.
1658 \RequirePackage{xparse}
1659 \ExplSyntaxOn

```

```

1660 \NewDocumentEnvironment{ liAntwort } { 0{standard} }
1661 {
1662   \str_case:nn {#1} {
1663     {standard} {
1664       \def\beschriftung{}
1665       \mdfsetup{backgroundcolor=gray!1,linecolor=gray}
1666     }
1667     {richtig} {
1668       \def\beschriftung{richtig}
1669       \mdfsetup{backgroundcolor=green!5,linecolor=green}
1670     }
1671     {falsch} {
1672       \def\beschriftung{falsch}
1673       \mdfsetup{backgroundcolor=red!3,linecolor=red}
1674     }
1675     {muster} {
1676       \def\beschriftung{Musterlösung}
1677       \mdfsetup{backgroundcolor=green!12,linecolor=green}
1678     }
1679   }
1680   \ifx\beschriftung\empty\else
1681     \noindent
1682     \textbf{\beschriftung{:}}
1683     \fi
1684     \begin{mdframed}
1685   }
1686 {\end{mdframed}}

```

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

```

1687 \NewDocumentEnvironment{ liAdditum } { o }
1688 {
1689   \begin{mdframed}[backgroundcolor=yellow!5]
1690     \IfNoValueTF {#1}
1691     { \liPseudoUeberschrift{Additum} }
1692     { \liPseudoUeberschrift{Additum:~#1} }
1693   }
1694 {\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.

```

1695 \NewDocumentEnvironment{ liExkurs }{o +b}{
1696   \vspace{0.2cm}%
1697   \begin{mdframed}[
1698     backgroundcolor=white,
1699     bottomline=false,
1700     innermargin=1cm,
1701     leftline=true,
1702     linecolor=black,
1703     linewidth=0.1cm,
1704     outermargin=1cm,
1705     rightline=false,
1706     topline=false,
1707   ]

```

```

1708 \footnotesize
1709 \noindent%
1710 \textbf{Exkurs:~#1}\par%
1711 \noindent%
1712 #2
1713 \end{mdframed}
1714 \vspace{0.2cm}
1715 }{}

```

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

```

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

```

Weiterführende Literatur:

- Quelle 1
- Quelle 2

```

1716 \cs_new:Npn \listen@punkt #1 {\item #1}
1717 \NewDocumentEnvironment { liQuellen }{ +b }
1718 {
1719   \seq_clear_new:N \l_quellen
1720   \seq_set_split:Nnn \l_quellen {\item} {#1}
1721   \seq_remove_all:Nn \l_quellen {}
1722   \begin{mdframed}[backgroundcolor=white!0]
1723     \footnotesize
1724     \noindent
1725     \textsf{\textbf{Weiterführende~Literatur:}}
1726     \medskip
1727     \begin{compactitem}
1728       \seq_map_function:NN \l_quellen {\listen@punkt}
1729     \end{compactitem}
1730   \end{mdframed}
1731   %
1732   \makeatletter
1733   \par\@afterindentfalse\@afterheading
1734   \makeatother
1735 } {}

```

liLernkartei

```

1736 \NewDocumentEnvironment { liLernkartei }{ m +b }
1737 {
1738   \begin{mdframed}
1739     \footnotesize
1740     \noindent%
1741     \textbf{Lernkarteikarte:~#1}\par%
1742     \noindent%
1743     #2
1744   \end{mdframed}
1745 } {}

```

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

```

1746 \NewDocumentEnvironment { liDiagramm }{ m +b }
1747 {
1748   \begin{mdframed}[backgroundcolor=white!0]
1749     \small
1750     \noindent%
1751     \textit{#1}:
1752     \begin{center}

```

```

1753   #2
1754   \medskip
1755   \end{center}
1756   \end{mdframed}
1757 } {}
1758 \ExplSyntaxOff

\liFussnoteUrl \liFussnoteUrl[\langle zusätzlicher-text \rangle]{\langle url \rangle} \liFussnoteUrl[zusätzlicher Text]{url}:
Eine HTTP-URL als Fußnote setzen.
1759 \NewDocumentCommand{\liFussnoteUrl} { o m } {
1760   \footnote{\url{#2}\IfNoValueTF{#1}{}{ ( #1 ) }}
1761 }
1762

\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.
1763 \NewDocumentCommand{\liFussnoteLink} { o m m } {
1764   \footnote{\href{#3}{#2}\IfNoValueTF{#1}{}{ ( #1 ) }}
1765 }

\zB
1766 \def\zB{z.\,B. }

\ZB
1767 \def\ZB{Z.\,B. }

\dh
1768 \def\dh{d.\,h. }

1769

```


2.27 master-theorem.sty

1770 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1771 \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}

1772 \ExplSyntaxOn

1773 \RequirePackage{amsmath}

\liRundeKlammer

1774 \def\liRundeKlammer#1{

1775 \negthinspace \left(#1 \right)

1776 }

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

1777 \def\liThetaOhneMathe#1{

1778 \Theta \liRundeKlammer{#1}

1779 }

1780 \def\liTheta#1{

1781 \ifmmode

1782 \liThetaOhneMathe{#1}

1783 \else

1784 \$\liThetaOhneMathe{#1}\$

1785 \fi

1786 }

```

\liOmega \liOmega{n^2}:  $\Omega(n^2)$ 
1787 \def\liOmegaOhneMathe#1{
1788 \Omega \liRundeKlammer{#1}
1789 }
1790 \def\liOmega#1{
1791 \ifmmode
1792 \liOmegaOhneMathe{#1}
1793 \else
1794 $\liOmegaOhneMathe{#1}$
1795 \fi
1796 }

\liO \liO{n^2}:  $\mathcal{O}(n^2)$ 
1797 \def\liOOhneMathe#1{
1798 \mathcal{O} \liRundeKlammer{#1}
1799 }
1800 \def\liO#1{
1801 \ifmmode
1802 \liOOhneMathe{#1}
1803 \else
1804 $\liOOhneMathe{#1}$
1805 \fi
1806 }

\liT Let-Abkürzung: \let\T=\liT
\liT{16}{2}:  $16 \cdot T\left(\frac{n}{2}\right)$  \liT{}{2}:  $T\left(\frac{n}{2}\right)$ 
1807 \def\liTOhneMathe#1#2{
1808 \tl_if_blank:nTF {#1}
1809 {}
1810 {#1 \cdot }
1811 T
1812 \liRundeKlammer{\frac{n}{#2}}
1813 }
1814 \def\liT#1#2{
1815 \ifmmode
1816 \liTOhneMathe{#1}{#2}
1817 \else
1818 $\liTOhneMathe{#1}{#2}$
1819 \fi
1820 }

\liRekursionsGleichung \liRekursionsGleichung:  $T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$ 
1821 \def\liRekursionsGleichung{
1822 $T(n) = \liT{a}{b} + f(n)$
1823 }

\liBedingungEins \liBedingungEins:  $f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})$ 
1824 \def\liBedingungEins{
1825 $f(n) \in \liO{n^{\log\sb{b}a - \varepsilon}}$
1826 }

\liBedingungZwei \liBedingungZwei:  $f(n) \in \Theta(n^{\log_b a})$ 
1827 \def\liBedingungZwei{
1828 $f(n) \in \liTheta{n^{\log\sb{b}a}}$
1829 }

\liBedingungDrei \liBedingungDrei:  $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ 
1830 \def\liBedingungDrei{
1831 $f(n) \in \liOmega{n^{\log\sb{b}a + \varepsilon}}$
1832 }

1833 \ExplSyntaxOff

```

\liMasterVariablen

```

1834 \def\liMasterVariablen{
1835   \begin{displaymath}
1836   T(n) = \liT{a}{b} + f(n)
1837   \end{displaymath}
1838
1839   \begin{itemize}
1840   \item[$a = $]
1841     Anzahl der Unterprobleme in der Rekursion
1842
1843   \item[$\textstyle{\frac{1}{b}} = $]
1844     Teil des Originalproblems, welches wiederum durch alle Unterprobleme
1845     repräsentiert wird
1846
1847   \item[$f(n) = $]
1848     Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
1849     die Kombination der Teillösungen entstehen
1850   \end{itemize}
1851   \footcite{wiki:master-theorem}
1852   \footcite[Seite 19-35 (PDF 11-24)]{aud:fs:2}
1853 }

```

\liMasterFaelle

```

1854 \def\liMasterFaelle{
1855   \begin{description}
1856   \item[1. Fall:]
1857      $T(n) \in \liTheta{n^{\log_{sb{b}}a}}$ 
1858
1859     \hfill falls \liBedingungEins
1860     für  $\varepsilon > 0$ 
1861
1862   \item[2. Fall:]
1863      $T(n) \in \liTheta{n^{\log_{sb{b}}a} \cdot \log n}$ 
1864
1865     \hfill falls \liBedingungZwei
1866
1867   \item[3. Fall:]
1868      $T(n) \in \liTheta{f(n)}$ 
1869
1870     \hfill falls \liBedingungDrei
1871     für  $\varepsilon > 0$ 
1872     und ebenfalls für ein  $c$  mit  $0 < c < 1$  und alle hinreichend großen  $n$ 
1873     gilt:
1874      $a \cdot f(\textstyle{\frac{n}{b}}) \leq c \cdot f(n)$ 
1875   \end{description}
1876 }

```

\liMasterVariablenDeklaration

```

1877 \def\liMasterVariablenDeklaration#1#2#3{
1878   \begin{description}
1879     \item[Allgemeine Rekursionsgleichung:] \strut
1880
1881     \liRekursionsGleichung
1882
1883     \item[Anzahl der rekursiven Aufrufe ( $a$ ):] \strut
1884
1885     #1
1886
1887     \item[Anteil Verkleinerung des Problems ( $b$ ):] \strut
1888
1889     um  $\frac{1}{#2}$  also  $b = #2$ 
1890
1891     \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut

```

```

1892
1893     $#3$
1894
1895     \item[Ergibt folgende Rekursionsgleichung:] \strut
1896
1897      $T(n) = T_{\#1}^{\#2} + \#3$ 
1898     \end{description}
1899 }

\liMasterFallRechnung
1900 \def\liMasterFallRechnung#1#2#3{
1901     \begin{description}
1902     \item[1. Fall:] \liBedingungEins:
1903
1904     #1
1905
1906     \item[2. Fall:] \liBedingungZwei:
1907
1908     #2
1909
1910     \item[3. Fall:] \liBedingungDrei:
1911
1912     #3
1913     \end{description}
1914 }

\liMasterExkurs
1915 \def\liMasterExkurs{
1916     \begin{liExkurs}[Master-Theorem]
1917     \liMasterVariablen
1918
1919     \noindent
1920     Dann gilt:
1921
1922     \liMasterFaelle
1923     \end{liExkurs}
1924 }

\liMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)
1925 \def\liMasterWolframLink#1{
1926     Berechne die Rekursionsgleichung auf WolframAlpha:
1927     \href{https://www.wolframalpha.com/input/?i=\#1}{WolframAlpha}
1928 }

1929

```

2.28 mathe.sty

```
1930 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1931 \ProvidesPackage{lehramt-informatik-mathe}[2020/06/10]
1932
1933 % for example \ltimes \rtimes
1934 %\RequirePackage{amssymb}
1935 \RequirePackage{amsmath}
1936
1937 %%
1938 % \mlq \mrq
1939 %%
1940 \DeclareMathSymbol{\mlq}{\mathord}{operators}{``}
1941 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`}
1942
```

2.29 minimierung.sty

```

1943 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1944 \ProvidesPackage{lehramt-informatik-mathe}[2021/03/13 Für den
1945 Minimierungsalgorithmus von einem NEA zu einem DEA]

1946 \liLadePakete{typographie}

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


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


1947 \def\liFussnote#1{$x_{\text{\scriptsize #1}}$}

1948 \def\li@fussnote@text#1#2{
1949   \liFussnote{\scriptsize #1}
1950   \quad
1951   {\footnotesize #2}
1952 }

1953 \def\liFussnoteEinsText{
1954   \li@fussnote@text{1}
1955   {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
1956 }

1957 \def\liFussnoteZweiText{
1958   \li@fussnote@text{2}
1959   {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
1960 }

1961 \def\liFussnoteDreiText{
1962   \li@fussnote@text{3}

```

```

1963 {In weiteren Iterationen markierte Zustände.}
1964 }

\liFussnoteVierText
1965 \def\liFussnoteVierText{
1966   \li@fussnote@text{4}
1967   {...}
1968 }

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



1969 \def\liFussnoten{
1970   \bigskip
1971
1972   \noindent
1973   \liFussnoteEinsText
1974
1975   \noindent
1976   \liFussnoteZweiText
1977
1978   \noindent
1979   \liFussnoteDreiText
1980
1981   \noindent
1982   \liFussnoteVierText
1983 }

\liLeereZelle \liLeereZelle:  $\emptyset$ 
Let-Abkürzung: \let\l=\liLeereZelle
1984 \def\liLeereZelle{\$\emptyset\$}

\liZustandsPaarVariablenName
1985 \def\liZustandsPaarVariablenName{z}

\liZustandsPaar
1986 \def\liZustandsPaar#1#2{
1987   $(
1988     \liZustandsPaarVariablenName_#1,
1989     \liZustandsPaarVariablenName_#2
1990   )$
1991 }

liUebergangsTabelle
1992 \renewcommand{\arraystretch}{1.4}
1993 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
1994   \liPseudoUeberschrift{Übergangstabelle}
1995   \begin{center}
1996     \begin{tabular}{r|l|l}
1997       \textbf{Zustandspaar} & \textbf{\#1} & \textbf{\#2} \\ \hline
1998     \end{tabular}
1999   \end{center}
2000 }
2001 }

liUeberschriftDreiecksTabelle \liUeberschriftDreiecksTabelle: Minimierungstabelle (Table filling)

2002 \ExplSyntaxOn
2003 \def\liUeberschriftDreiecksTabelle{
2004   \liPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
2005 }

```

\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. —

```

2006 \def\liMinimierungErklaerung{
2007   %\footcite[Seite~19]{koenig}
2008   \liParagraphMitLinien{
2009     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus-genannt)~
2010     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
2011     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
2012     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
2013      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
2014     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
2015     somit~zueinander~( $k - 1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
2016     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
2017     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
2018   }
2019 }
2020 \ExplSyntaxOff
2021

```


2.30 normalformen.sty

```
2022 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2023 \ProvidesPackage{lehramt-informatik-normalformen}[2020/12/10
2024 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
2025 Attributhülle]
```

```
Mathe für textit tex etc, typographie für geschweifte Klammer
2026 \liLadePakete{mathe,typographie}
2027 \directlua{
2028   helfer = require('lehramt-informatik-helfer')
2029   normalformen = require('lehramt-informatik-normalformen')
2030 }
```

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
```

```
2031 \def\liTeilen#1{
2032   \directlua{tex.print(normalformen.teilen('#1'))}
2033 }
```

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

```
2034 \def\liAttributHuelleOhneMathe#1{\text{AttrHülle}(\#1)}
2035 \def\liAttributHuelle#1{
2036   \ifmmode
2037     \liAttributHuelleOhneMathe{\#1}
2038   \else
2039     $\liAttributHuelleOhneMathe{\#1}$
2040   \fi
2041 }
```

\liAttributMenge Let-Abkürzung: $\text{let}\ \text{m}=\text{liAttributMenge}$
2042 $\backslash\text{def}\ \text{liAttributMenge}\#1\{\backslash\{\ \text{textit}\{\#1\}\ \backslash\}$

liAHuelle

```
2043 \NewDocumentEnvironment{ liAHuelle } { +b } {
2044   \begingroup
2045   \footnotesize
2046   \begin{multline*}
2047     \#1
2048   \end{multline*}
2049   \endgroup
2050 } { }
```

\liLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multline

Let-Abkürzung: $\text{let}\ \text{ahL}=\text{liLinksReduktion}$
 $\text{ahL}\{\text{ursprüngliche linke Attributmenge}\{\text{ohne dieses Attribut}\}\{\text{Ergebnis}\}$
2051 $\backslash\text{def}\ \text{liLinksReduktion}\#1\#2\#3\{$
2052 $\backslash\text{shoveleft}\{$
2053 $\text{liAttributHuelleOhneMathe}\{\text{FA},$
2054 $\text{liAttributMenge}\{\#1\ \text{string}\ \#2\}\} =$
2055 $\}\ \backslash\backslash$

```

2056 \shoveright{
2057   \liAttributMenge{#3}
2058 } \\\
2059 }

```

```

\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( $F, \{A, E \setminus E\}$ ) =  $\{A, E, F, B, D\}$ 
2060 \def\liLinksReduktionInline#1#2#3{%
2061   {%
2062     \footnotesize%
2063     $\liAttributHuelleOhneMathe{F,
2064       \liAttributMenge{#1} \string\ #2}} =
2065     \liAttributMenge{#3}$
2066   }
2067 }

```

```

\liLinksReduktionInline Let-Abkürzung: \let\ahr=\liLinksReduktionInline
    \ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
2068 \def\liRechtsReduktionInline#1#2#3#4{%
2069   {%
2070     \footnotesize%
2071     $\liAttributHuelleOhneMathe{
2072       F \setminus
2073       \liFunktionaleAbhaengigkeit{#1}
2074       \def\tmp{#2}\ifx\tmp\empty
2075       \else
2076         \cup \liFunktionaleAbhaengigkeit{#2}
2077       \fi
2078       ,
2079       \liAttributMenge{#3}
2080     } =
2081     \liAttributMenge{#4}$
2082   }
2083 }

```

```

\liFunktionaleAbhaengigkeit Let-Abkürzung: \let\fa=\liFunktionaleAbhaengigkeit
    \fa{A, B -> C, D}:  $\{A, B\} \rightarrow \{C, D\}$  Regulärer Ausdruck zum Konvertieren:
    \$(.*?) \\\rightarrow (.*)\$ \fa{$1 -> $2}
2084 \def\liFunktionaleAbhaengigkeit#1{%
2085   \directlua{normalformen.drucke_funk_abhaengigkeit('#1', false)}%
2086 }

```

```

\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;

```

```

2087 \NewDocumentCommand {\liFunktionaleAbhaengigkeiten} { O{FA} m } {
2088   \liGeschweifteKlammern
2089   {#1}
2090   {
2091     \begin{align*}
2092       \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
2093     \end{align*}
2094   }
2095   {-0.5cm}
2096   {-1.7cm}
2097 }

\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}

2098 \NewDocumentCommand {\liRelation} { O{R} m } {
2099   $\directlua{
2100     local name = helper.konvertiere_tiefgestellt('#1')
2101     tex.print(name)
2102   }$(\textit{\,#2\,})
2103 }

2104

```

2.31 petri.sty

2105 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2106 \ProvidesPackage{lehramt-informatik-petri}[2020/12/03]

2.31.1 Makro-Kürzel

\let\t=\liPetriTransitionsName

\let\tp=\liPetriTransPfeile

\let\k=\liPetriErreichKnotenDrei

2107 \RequirePackage{tikz}

2108 \usetikzlibrary{petri,arrows.meta}

Für die Darstellungsmatrix

2109 \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

```
2110 \def\liPetriSetzeSchluessel{%
2111   \def\TmpTransitionOne{}%
2112   \def\TmpTransitionTwo{}%
2113   \def\TmpTransitionThree{}%
2114   \def\TmpTransitionFour{}%
2115   \def\TmpTransitionFive{}%
2116   \def\TmpTransitionSix{}%
2117   \def\TmpTransitionSeven{}%
2118   \def\TmpTransitionEight{}%
2119   \def\TmpTransitionNine{}%
2120   \def\TmpTransitionTen{}%
2121   \pgfkeys{/petri/.cd,
2122     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2123     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2124     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2125     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
```

```

2126 p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
2127 p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2128 p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2129 p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2130 p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2131 p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2132 t1/.store in=\TmpTransitionOne,t1/.default=activated,
2133 t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2134 t3/.store in=\TmpTransitionThree,t3/.default=activated,
2135 t4/.store in=\TmpTransitionFour,t4/.default=activated,
2136 t5/.store in=\TmpTransitionFive,t5/.default=activated,
2137 t6/.store in=\TmpTransitionSix,t6/.default=activated,
2138 t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2139 t8/.store in=\TmpTransitionEight,t8/.default=activated,
2140 t9/.store in=\TmpTransitionNine,t9/.default=activated,
2141 t10/.store in=\TmpTransitionTen,t10/.default=activated,
2142 scale/.store in=\TmpScale,scale/.default=0.5,
2143 x/.store in=\TmpX,x/.default=5,
2144 y/.store in=\TmpY,y/.default=5,
2145 }%
2146 }

```

```

2147 \tikzset{
2148   li petri/.style={
2149     activated/.style={
2150       very thick
2151     },
2152     inhibitor/.style={
2153       {Circle[open,length=2mm,fill=white]}-
2154     }
2155   }
2156 }

```

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

```

2157 \def\liPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2158 \def\liPetriTransitionsName#1{
2159   \ifmmode
2160     \liPetriTransitionsNameOhneMathe{#1}
2161   \else
2162     $\liPetriTransitionsNameOhneMathe{#1}$
2163   \fi
2164 }

```

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

```

2165 \NewDocumentCommand{ \liPetriErreichTransition } { m m m O{} O{} } {
2166   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{t\sb{#3}$} (#2);
2167 }

```

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

```

2168 \def\liPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

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

```

2169 \def\liPetriTransPfeile#1{ $\rightarrow \hspace{0.4cm} \liPetriTransitionsName{#1} \hspace{0.3cm}
2170

```

2.32 potenzmengen-konstruktion.sty

```
2171 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2172 \ProvidesPackage{lehramt-informatik-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2173 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2174 \liLadePakete{formale-sprachen}
2175 \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}
    }
  }
}
```

```
2176 \def\liZustandsMengenSammlung#1#2{
2177   \liZustandsnameGross{#1}
2178   {
2179     \footnotesize
2180     \liPotenzmenge{
2181       \str_case:nn {#1} {#2
2182         }
2183     }
2184 }
```

```
\liZustandsMengenSammlungNr
```

```
2185 \def\liZustandsMengenSammlungNr#1#2{
2186   \liZustandsnameGross{#1}
2187   {
```

```
2188     \footnotesize
2189     \liZustandsmengeNr{
2190         \str_case:nn {#1} #2
2191     }
2192 }
2193 }

2194 \ExplSyntaxOff
2195
```

2.33 pseudo.sty

2196 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2197 \ProvidesPackage{lehramt-informatik-pseudo}[2020/12/30 Paket zum Setzen
 2198 von Pseudo-Code, Hüll-Paket um algorithm2e]

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

Algorithmus 1: Minimaler Spannbaum nach Kruskal

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

2199 \RequirePackage[german,boxruled]{algorithm2e}

2200

2.34 pumping-lemma.sty

2201 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2202 \ProvidesPackage{lehrant-informatik-pumping-lemma}[2021/08/11 Enthält die
 2203 Definitionen für das Pumping-Lemma in der Regulären Sprache und
 2204 in der Kontextfreien Sprache]

\liPumpingRegulaer

```
2205 \def\liPumpingRegulaer{%
2206   Es sei  $L$  eine reguläre Sprache. Dann gibt es eine Zahl  $j$ , sodass für
2207   alle Wörter  $\omega \in L$  mit  $|\omega| \geq j$  (jedes Wort  $\omega$  in
2208    $L$  mit Mindestlänge  $j$ ) jeweils eine Zerlegung  $\omega = uvw$ 
2209   existiert, sodass die folgenden Eigenschaften erfüllt sind:
2210
2211   \begin{enumerate}
2212     \item  $|v| \geq 1$ 
2213     (Das Wort  $v$  ist nicht leer.)
2214
2215     \item  $|uv| \leq j$ 
2216     (Die beiden Wörter  $u$  und  $v$  haben zusammen höchstens die Länge  $j$ .)
2217
2218     \item Für alle  $i = 0, 1, 2, \dots$  gilt  $uv^i w \in L$ 
2219     (Für jede natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^i w$  in der
2220     Sprache  $L$ )
2221   \end{enumerate}
2222
2223   Die kleinste Zahl  $j$ , die diese Eigenschaften erfüllt, wird
2224   Pumping-Zahl der Sprache  $L$  genannt.\footcite{wiki:pumping-lemma}
2225 }
```

\liPumpingKontextfrei

```
2226 \def\liPumpingKontextfrei{%
2227   Es sei  $L$  eine kontextfreie Sprache. Dann gibt es eine Zahl  $j$ , sodass
2228   sich alle Wörter  $\omega \in L$  mit  $|\omega| \geq j$  zerlegen lassen in
2229    $\omega = uvwxy$ , sodass die folgenden Eigenschaften erfüllt sind:
2230
2231   \begin{enumerate}
2232     \item  $|vx| \geq 1$ 
2233     (Die Wörter  $v$  und  $x$  sind nicht leer.)
2234
2235     \item  $|vwx| \leq j$ 
2236     (Die Wörter  $v$ ,  $w$  und  $x$  haben zusammen höchstens die Länge  $j$ .)
2237
2238     \item Für alle  $i \in \mathbb{N}_0$  gilt  $uv^iwx^iy \in L$  (Für jede
2239     natürliche Zahl (mit  $0$ )  $i$  ist das Wort  $uv^iwx^iy$  in der
2240     Sprache  $L$ )
2241   \end{enumerate}
2242 }
2243
```

2.35 quicksort.sty

```

2244 % https://tex.stackexchange.com/a/142634
2245 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2246 \ProvidesPackage{lehramt-informatik-quicksort}[2020/06/12]
2247
2248 %-----
2249 % USAGE:
2250 % \QSinitialize{comma, separated, numerical, values}
2251 % \loop
2252 % \QSpivotStep
2253 % \ifnum\value{pivotcount}>0
2254 %   \QSSortStep
2255 % \repeat
2256 %-----
2257
2258 % xintfrac does not load xinttools, this must be done explicitly if needed as here.
2259 \RequirePackage{xintfrac, xinttools}
2260
2261 \RequirePackage{tikz}
2262
2263 %-----
2264 % FIRST PART: TikZ styles and macros for the actual drawing
2265 \newcounter{cellcount}% used for coordinates of the node
2266 \newcounter{pivotcount}% when it will remain at zero, will signal the sort is finished.
2267
2268 % Styles defined by Tom Bombaldi. (modified: all share the same size)
2269 % (re-modified \bf -> \bfseries due to extremely annoying warnings from
2270 % KOMA-script which are truly a pain and do not make any sense regarding \bf:
2271 % if I want to use \bf, and know what I am doing, why should I get HARASSED
2272 % by police of LaTeX good conduct ? )
2273 \tikzset{l/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=lime!70!gray},
2274          o/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=olive!50},
2275          r/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2276 % this is the "b" style as used in the image below
2277          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=magenta!50!black, text=white},
2278 % nicer:
2279          b/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=white, text=magenta},
2280          g/.style={minimum width=6mm, minimum height=6mm, draw=black, fill=gray, text=white, text=white},
2281
2282 % NOTE the b style was originally the same as the r(aised) style apart from
2283 % not being raised, but I find it nicer with a somewhat different
2284 % specification. I have not updated the images though.
2285
2286 % How the nodes are drawn depending on whether on the left of the pivot value
2287 % or on the right, or is a pivot value, or a raised pivot during selection phase.
2288
2289 \def\DecoLEFT #1{%
2290   \xintFor* ##1 in {#1} \do
2291     {\stepcounter{cellcount}\node[o] at (\arabic{cellcount},0) {##1};}%
2292 }
2293
2294 \def\DecoINERT #1{%
2295   \xintFor* ##1 in {#1} \do
2296     {\stepcounter{cellcount}\node[g] at (\arabic{cellcount},0) {##1};}%
2297 }
2298
2299 \def\DecoRIGHT #1{%
2300   \xintFor* ##1 in {#1} \do
2301     {\stepcounter{cellcount}\node[l] at (\arabic{cellcount},0) {##1};}%
2302 }
2303
2304 \def\DecoLEFTwithPivot #1{\stepcounter{pivotcount}%
2305   \xintFor* ##1 in {#1} \do

```

```

2306     {\stepcounter{cellcount}}%
2307     \xintifForLast {\node[r]}\node[o]} at (\arabic{cellcount},0) {##1};}%
2308 }
2309
2310 \def\DecoINERTwithPivot #1{\stepcounter{pivotcount}%
2311     \xintFor* ##1 in {#1} \do
2312     {\stepcounter{cellcount}}%
2313     \xintifForLast {\node[b]}\node[g]} at (\arabic{cellcount},0) {##1};}%
2314 }
2315
2316 \def\DecoRIGHTwithPivot #1{\stepcounter{pivotcount}%
2317     \xintFor* ##1 in {#1} \do
2318     {\stepcounter{cellcount}}%
2319     \xintifForLast {\node[r]}\node[l]} at (\arabic{cellcount},0) {##1};}%
2320 }
2321
2322 %-----
2323 % SECOND PART: the actual sorting routines.
2324
2325 \def\QS@sort@a #1{\expandafter \QS@sort@b \expandafter {\xintLength {#1}}{#1}}
2326 \def\QS@sort@b #1{\ifcase #1
2327     \expandafter\QS@sort@empty
2328     \or\expandafter\QS@sort@single
2329     \else\expandafter\QS@sort@c
2330     \fi
2331 }%
2332 \def\QS@sort@empty #1{}
2333 \def\QS@sort@single #1{\QSIr {#1}}
2334
2335 % This step is to pick the last as pivot.
2336 \def\QS@sort@c #1%
2337     {\expandafter\QS@sort@d\expandafter {\romannumeral0\xintnthelt {-1}{#1}}{#1}}%
2338
2339 % Here \QSLr, \QSIr, \QSR have been let to \relax.
2340 % The trick with \xintApplyUnbraced is that for example when selecting
2341 % the elements smaller than pivot, if we had been using \xintApply we
2342 % would have had at the minimum an empty brace pair. Thus we use the
2343 % "unbraced" variant, but then the \QS@select@smaller has added in
2344 % anticipation a level of braces.
2345 \def\QS@sort@d #1#2{%
2346     \QSLr {\xintApplyUnbraced {\QS@select@smaller {#1}}{#2}}}%
2347     \QSIr {\xintApplyUnbraced {\QS@select@equal {#1}}{#2}}}%
2348     \QSRr {\xintApplyUnbraced {\QS@select@greater {#1}}{#2}}}%
2349 }%
2350 \def\QS@select@smaller #1#2{\xintifLt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2351 \def\QS@select@equal #1#2{\xintifEq {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2352 \def\QS@select@greater #1#2{\xintifGt {#2}{#1}{{#2}}{ }}% space will stop a f-
    expansion
2353
2354 %
2355 % NOTE 1: thus, each comparison with the pivot is done three (!) times.
2356 %
2357 % NOTE 2: we may well end up with \QSLr {<empty>} situations. This is handled
2358 % silently by the \xintFor loops, and also when \QSLr becomes \QS@sort@a, the
2359 % latter must handle correctly an empty argument.
2360
2361 %-----
2362 % THIRD PART: the main macros \QSpivotStep, \QSsortStep and \QSinitialize.
2363
2364 % This draws all with suitable highlighting for the newly chosen pivots
2365 % (which will be shown raised)

```

```

2366 \def\QSpivotStep {\let\QSLr\DecoLEFTwithPivot
2367             \let\QSIr\DecoINERT
2368             \let\QSIrr\DecoINERT
2369             \let\QSRr\DecoRIGHTwithPivot
2370 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2371             \setcounter{cellcount}{0}\setcounter{pivotcount}{0}%
2372             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2373 }
2374
2375 % This sorts and then draws, showing where the pivot chosen in the previous
2376 % step go. Next time they will have become "inert". If pivotcount is still at
2377 % zero on exit from \QSpivotStep, then this is the signal to stop before
2378 % executing \QSsortStep.
2379 \def\QSsortStep {\def\QSLr {\noexpand\QS@sort@a}%
2380             \def\QSRr {\noexpand\QS@sort@a}%
2381             \def\QSIr {\noexpand\QSIrr}%
2382             \let\QSIrr\relax
2383             \edef\QS@list{\QS@list}%
2384             \let\QSLr\relax
2385             \let\QSRr\relax
2386             \let\QSIr\relax
2387             \edef\QS@list{\QS@list}%
2388             \let\QSLr\DecoLEFT
2389             \let\QSIr\DecoINERTwithPivot
2390             \let\QSIrr\DecoINERT
2391             \let\QSRr\DecoRIGHT
2392 \par\centerline{\rule[1.5mm]{0pt}{8mm}}%
2393             \setcounter{cellcount}{0}%
2394             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2395 }
2396
2397 \def\QSinitialize #1{%
2398     % first, we convert the comma separated values into a list of braced items
2399     % we use an \edef, and anyhow many \edef's will be used later
2400     \edef\QS@list {\noexpand\QSRr {\xintCSVtoList {#1}}}%
2401     \let\QSRr\DecoRIGHT
2402     % The \QSRr marker mutated to draw the last element as
2403     % pivot and the earlier ones with the suitable style.
2404     %
2405     % The list of marked braced items \QS@list is used both for drawing
2406     % (as here) and for doing the exchange of elements during sort.
2407     \par\centerline{\rule[1.5mm]{0pt}{8mm}\setcounter{cellcount}{0}%
2408             \begin{tikzpicture}\QS@list\end{tikzpicture}}
2409 }
2410

```

2.36 relationale-algebra.sty

```

2411 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2412 \ProvidesPackage{lehramt-informatik-relationale-algebra}[2020/12/11]
2413 \RequirePackage{amsmath}
2414 \RequirePackage{amssymb}

```

```

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

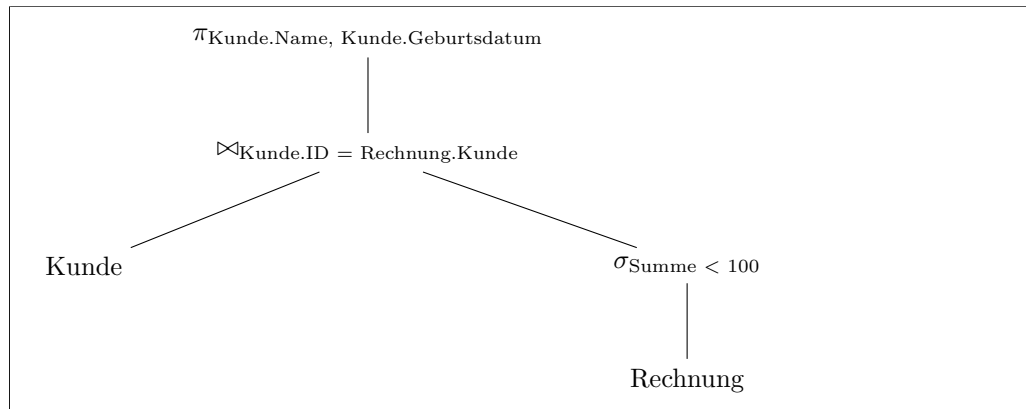
  \node[below=of pi]
    (theta join) {\bowtie_{\text{Kunde.ID = Rechnung.Kunde}}};
    edge (pi);

  \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}

```



```

2415 \RequirePackage{tikz}
2416 \usetikzlibrary{positioning}

    Privates Makros, das zwei Querstriche erzeugt.
2417 \def\o@join{\setbox0=\hbox{\bowtie}}%
2418 \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
2419 }

\leftouterjoin A \leftouterjoin B: A ⋈ B
2420 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}

\rightouterjoin A \rightouterjoin B: A ⋉ B
2421 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}

\fullouterjoin A \fullouterjoin B: A ⋈⋉ B
2422 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
2423

```

2.37 rmodell.sty

```
2424 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2425 \ProvidesPackage{lehramt-informatik-rmodell}[2020/09/01]
2426 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
2427 Datenbanken.]
2428 \RequirePackage{soul}
```

2.37.1 Makro-Kürzel

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

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

```
2429 \def\liPrimaer#1{\ul{#1}}
```

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

```
2430 \def\liFremd#1{{\setul{-0.9em}{}\ul{#1}}}
```

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

```
2431 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
2432 \ExplSyntaxOn
2433 \NewDocumentEnvironment { liRmodell }
2434 { +b }
2435 {
2436   \medskip
2437   {
2438     \linespread{2}
2439     \setlength{\parindent}{0pt}
2440     \li@Rmodell@Schrift#1
2441   }
2442   \medskip
2443 } {}
2444 \ExplSyntaxOff
```

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

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

```
2445 \def\liRelationMenge#1#2{
2446 \noindent
2447 #1 : \[ #2 ]\}
2448 \par
2449 }
```

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

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

```
2450 \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}
```

```
2451 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
2452
```

2.38 sortieren.sty

```
2453 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2454 \ProvidesPackage{lehramt-informatik-sortieren}[2020/06/10 Schaubilder
2455 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}
}
```



```
2456 \RequirePackage{tikz}
2457 \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.

```
2458 \def\liVertauschen#1{
2459   \directlua{
2460     local sortieren = require('lehramt-informatik-sortieren')
2461     sortieren('#1')
2462   }
2463 }
```

`\liSortierPfeil`

```
2464 \def\liSortierPfeil#1#2{
2465   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
2466 }
```

`\liSortierPfeilUnten`

```
2467 \def\liSortierPfeilUnten#1#2{
2468   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
2469 }
```

`\liSortierMarkierung`

```
2470 \def\liSortierMarkierung#1#2{\node[
2471   draw,
2472   very thick,
2473   fit=(reihe.#1) (reihe.#2),
2474   inner sep=0pt
2475 ] {}
2476 }

2477 \tikzset{
2478   li sortierung zahlenreihe/.style={
2479     draw,
2480     thin,
2481     font=\large,
2482     rectangle split horizontal,
2483     rectangle split,
2484   }
2485 }
```

```

2486 % https://tex.stackexchange.com/a/140895
2487 \RequirePackage{forest,xstring}
2488 \usetikzlibrary{calc}
2489
2490 \makeatletter
2491 \pgfmathdeclarefunction{strrepeat}{2}{%
2492   \begingroup\pgfmathint{#2}\pgfmath@count\pgfmathresult
2493   \let\pgfmathresult\pgfutil@empty
2494   \pgfutil@loop\ifnum\pgfmath@count>0\relax
2495     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
2496     \advance\pgfmath@count-1\relax
2497   \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
2498 \makeatother
2499
2500 \def\myNodes{}
2501
2502 \ExplSyntaxOn
2503 \newcommand*\sortList[1]{%
2504   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
2505 \ExplSyntaxOff
2506
2507 \forestset{
2508   sort/.code={%
2509     \pgfmathparse{level()>\forestSortLevel}%
2510     \ifnum\pgfmathresult=0
2511       \StrSubstitute{\forestov{content}}{ }{,}[\myList]%
2512       \sortList\myList
2513       \StrSubstitute{\myList}{,}{ }[\myList]%
2514       \pgfmathparse{strrepeat("1",level())}%
2515       \xappto\myNodes{\noexpand\node at ({\forestov{name}}|-m)!-1!({\forestov{name}}$)
2516         (m\forestov{name}) {\myList}}%
2517       \pgfmathparse{level()=\forestSortLevel}%
2518       \ifnum\pgfmathresult=1
2519         \forestOget{\forestov{@first}}{name}\forestFirst
2520         \forestOget{\forestov{@last}}{name}\forestLast
2521         \xappto\myNodes{{[<-]edge (\forestOv{\forestov{@first}}{name})
2522           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
2523       \fi
2524       \ifnum\forestov{@parent}=0\else
2525         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
2526       \fi
2527       \gappto\myNodes{;}%
2528     \fi}}
2529
2530 \forestset{sort level/.code=%
2531   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
2532   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
2533

```


2.39 spalten.sty

```
2534 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2535 \ProvidesPackage{lehramt-informatik-spalten}[2020/12/07 Lädt das Paket
2536 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
2537 realisiert werden kann.]
2538 \RequirePackage{multicol}
```

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

```
2539 \def\liSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
2540
```

2.40 sql.sty

```
2541 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2542 \ProvidesPackage{lehramt-informatik-tabelle}[2021/09/02 Zu Setzen von SQL]
2543 \liLadePakete{syntax}
2544 \RequirePackage{fancyvrb}
2545 \DefineVerbatimEnvironment{liSqlErgebnis}{Verbatim}
2546 {fontsize=\footnotesize}
2547
```

2.41 struktogramm.sty

```
2548 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2549 \ProvidesPackage{lehramt-informatik-struktogramm}[2021/01/31 Lädt das
2550 Paket struktex zum Setzen von Struktogrammen]
2551 \RequirePackage{struktex}
2552
```

2.42 syntax.sty

```
2553 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2554 \ProvidesPackage{lehramt-informatik-syntax}[2020/06/10 Ein Hüll-Paket um
2555 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
2556 \RequirePackage{xparse}
```

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

2.42.1 Makro-Kürzel

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

```
2557 \ExplSyntaxOn
2558 \directlua{
2559   syntax = require('lehramt-informatik-syntax')
2560   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
2561   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
2562   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
2563   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
2564   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
2565   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
2566 }
2567 \RequirePackage{hyperref}
2568 \RequirePackage{minted}
2569 % pygmentize -L styles
2570 \usemintedstyle{colorful}
2571 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
2572 %\AfterEndEnvironment{minted}{\end{mdframed}}
2573 %\setminted{breaklines=true,linenos}
2574 \setminted{
2575   breaklines=true,
2576   linenos,
2577   fontsize=\footnotesize,
2578 }
```

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

Let-Abkürzung: `\let\j=\liJavaCode`

```
2579 \def\liJavaCode#1{\,\mintinline{java}|#1|\,}
```

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

```
2580 \def\liLatexCode#1{\mintinline{latex}|#1|}
```

```
2581 \def\li@GithubLink#1#2{
2582   \begin{flushright}
2583     \tiny
2584     Code-Beispiel~auf~Github~ansehen:~
2585     \href{#1}{\nolinkurl{#2}}
2586   \end{flushright}
2587 }
```

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

```
2588 \NewDocumentCommand{\liJavaDatei}{0{firstline=3} m }{
2589   \inputminted[#1]{java}{
2590     \directlua{
2591       syntax.drucke_absoluten_pfad('#2', false)
2592     }
2593   }
2594   \li@GithubLink
```

```

2595     {\directlua{syntax.drucke_github_url('#2', false)}}
2596     {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
2597 }

\liJavaTestDatei Eine komplette Java-Test-Datei einbinden, die Verzeichnis ./Code/src/test/java/org/bschlangaul
liegt.
2598 \NewDocumentCommand{\liJavaTestDatei}{ O{firstline=3} m }{
2599   \inputminted[#1]{java}{
2600     \directlua{
2601       syntax.drucke_absoluten_pfad('#2', true)
2602     }
2603   }
2604   \li@GithubLink
2605   {\directlua{syntax.drucke_github_url('#2', true)}}
2606   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
2607 }

\liJavaExamen \liJavaExamen{66116}{2015}{03}{Kunde} Regulärer Ausdruck zum Konvertieren: \liJavaDatei([.
\liJavaExamen$1{$2}{$3}{$4}{$5}
2608 \NewDocumentCommand{\liJavaExamen}{ O{firstline=3} m m m m }{
2609   \inputminted[#1]{java}{
2610     \directlua{
2611       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
2612     }
2613   }
2614
2615   \li@GithubLink
2616   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
2617   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
2618 }

\liAssemblerCode
2619 \def\liAssemblerCode#1{\mintinline{asm}|#1|}

\liAssemblerDatei \liAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
2620 \NewDocumentCommand{\liAssemblerDatei}{ m }{
2621   \inputminted{asm}{#1}
2622 }

\liMinispracheDatei \liMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
2623 \NewDocumentCommand{\liMinispracheDatei}{ m }{
2624   \inputminted{componentpascal}{#1}
2625 }

\liHaskellCode \liHaskellCode{haskell}: Zum Setzen von Haskell-Code.
2626 \def\liHaskellCode#1{\mintinline{haskell}|#1|}

\liHaskellDatei \liHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
2627 \NewDocumentCommand{\liHaskellDatei}{ m }{
2628   \inputminted{haskell}{#1}
2629 }

2630 \ExplSyntaxOff

\liSqlCode \liHaskellCode{sql}: Zum Setzen von SQL-Code.
Let-Abkürzung: \let\s=\liSqlCode
2631 \def\liSqlCode#1{\mintinline{sql}|#1|}

2632

```

2.43 syntaxbaum.sty

```
2633 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2634 \ProvidesPackage{lehramt-informatik-syntaxbaum}[2021/02/14 Zum Setzen von
2635 Syntaxbäumen mit Hilfe des Pakets tikz-qtrees]
2636 \RequirePackage{tikz-qtrees}
2637
2638 \tikzset{li parsetree/.style={
2639     every internal node/.style={
2640         draw,circle
2641     },
2642     every leaf node/.style={
2643         draw,rectangle
2644     },
2645 }
2646 }
2647
```

2.44 synthese-algorithmus.sty

```
2648 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2649 \ProvidesPackage{lehramt-informatik-synthese-algorithmus}[2021/03/19
2650 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
2651 Relation in die 3. Normalform]
2652 \liLadePakete{normalformen,mathe,typographie}
2653 \ExplSyntaxOn
```

2.44.1 Makro-Kürzel

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

2.44.2 TeX-Markup Grundgerüst

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

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

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

2.44.3 TeX-Markup Linksreduktion

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

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

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

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

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

2.44.4 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}

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

2.44.5 TeX-Markup Relationen formen

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

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

1. Kanonische Überdeckung

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

(a) Linksreduktion

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

(b) Rechtsreduktion

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

(c) Löschen leerer Klauseln

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

(d) Vereinigung

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

2. Relationsschemata formen

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

3. Schlüssel hinzufügen

— Falls eines der in Schritt 2. erzeugten Schemata R_α einen Schlüsselkandidaten von \mathcal{R} bezüglich F_c enthält, sind wir fertig, sonst wähle einen Schlüsselkandidaten $\mathcal{K} \subseteq \mathcal{R}$ aus und definiere folgendes zusätzliche Schema: $\mathcal{R}_\mathcal{K} := \mathcal{K}$ und $\mathcal{F}_\mathcal{K} := \emptyset$ —

4. Entfernung überflüssiger Teilschemata

— Eliminiere diejenigen Schemata R_α , die in einem anderen Relationenschema $R_{\alpha'}$ enthalten sind, d. h. $R_\alpha \subseteq R_{\alpha'}$. —

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

```
2654 \def\liSyntheseUeberschrift#1{
2655   {
2656     \bfseries
2657     \sffamily
2658     \str_case:nn {#1} {
2659       {1} {Kanonische-Überdeckung}
2660       {1-1} {Linksreduktion}
2661       {1-2} {Rechtsreduktion}
2662       {1-3} {Löschen-leerer-Klauseln}
2663       {1-4} {Vereinigung}
2664       {2} {Relationsschemata-formen}
2665       {3} {Schlüssel-hinzufügen}
2666       {4} {Entfernung-überflüssiger-Teilschemata}
2667     }
```



```

2668 }
2669 }

\liSyntheseErklaerung Let-Abkürzung: \let\verklaerung=\liSyntheseErklaerung
2670 \def\li@synthese@erklaerung@texte#1{
2671   \str_case:nn {#1} {
2672     {1} {
2673       Die~kanonische~Überdeckung~---also~die~kleinst~mögliche~noch~
2674       äquivalente~Menge~von~funktionalen~Abhängigkeiten~kann~in~vier~
2675       Schritten~erreicht~werden.
2676     }
2677     {1-1} {
2678       Führe~für~jede~funktionale~Anhängigkeit~
2679        $\alpha \rightarrow \beta$ ~in~F~die~Linksreduktion~durch,~
2680       überprüfe~also~für~alle~
2681        $A \in \alpha$ ,~ob~ $A$ ~überflüssig~ist,~d.h.~ob~
2682        $\beta \subseteqq \text{liAttributHuelle}\{F, \alpha \cup A\}$ .
2683     }
2684     {1-2} {
2685       Führe~für~jede~(verbliebene)~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~die~Rechtsreduktion~durch,~überprüfe~also~für~
2686       alle~ $B \in \beta$ ,~ob~ $B \in \text{liAttributHuelle}\{F, (\alpha \rightarrow \beta) \cup (\alpha \rightarrow B)\}$ ,~
2687        $\alpha$ ~gilt.~In~diesem~Fall~ist~B~auf~der~rechten~Seite~
2688       überflüssig~und~kann~eliminiert~werden,~d.h.~ $\alpha \rightarrow \beta$ ~wird~durch~ $\alpha \rightarrow (\beta \cup B)$ ~
2689       ersetzt.
2690     }
2691     {1-3} {
2692       Entferne~die~funktionalen~Abhängigkeiten~der~Form~ $\alpha \rightarrow \emptyset$ ,~die~im~2.~Schritt~möglicherweise~
2693       entstanden~sind.
2694     }
2695     {1-4} {
2696       Fasse~mittels~der~Vereinigungsregel~funktionale~Abhängigkeiten~
2697       der~Form~ $\alpha \rightarrow \beta_{\{1\}}, \dots, \alpha \rightarrow \beta_{\{n\}}$ ,~so~dass~ $\alpha \rightarrow \beta_{\{1\}} \cup \dots \cup \beta_{\{n\}}$ ~
2698       verbleibt.
2699     }
2700     % Kemper Seite 197
2701     {2} {
2702       Erzeuge~für~jede~funktionale~Abhängigkeit~ $\alpha \rightarrow \beta$ ~in~F~ein~Relationenschema~ $\mathcal{R}_{\alpha}$ ~
2703        $:= \alpha \cup \beta$ .
2704     }
2705     {3} {
2706       Falls~eines~der~in~Schritt~2.~erzeugten~Schemata~ $\mathcal{R}_{\alpha}$ ~
2707       einen~Schlüsselkandidaten~von~ $\mathcal{R}_{\alpha}$ ~bezüglich~ $F$ ~
2708       enthält,~sind~wir~fertig,~sonst~wähle~einen~Schlüsselkandidaten~
2709        $\mathcal{K} \subseteqq \mathcal{R}_{\alpha}$ ~aus~und~definiere~folgendes~
2710       zusätzliche~Schema:~ $\mathcal{R}_{\alpha} \setminus \mathcal{K} := \mathcal{K}$ ~
2711       und~ $\mathcal{F} \setminus \mathcal{K} := \emptyset$ 
2712     }
2713     {4} {
2714       Eliminiere~diejenigen~Schemata~ $\mathcal{R}_{\alpha}$ ,~die~in~einem~
2715       anderen~Relationenschema~ $\mathcal{R}_{\alpha'}$ ~enthalten~sind,~d.h.~
2716        $\mathcal{R}_{\alpha} \subseteqq \mathcal{R}_{\alpha'}$ .
2717     }
2718   }
2719 }
2720 \def\liSyntheseErklaerung#1{
2721   {
2722     \itshape
2723     \footnotesize

```

```

2730     \liParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
2731   }
2732 }

```

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

```

2733 \def\liSyntheseUeberErklaerung#1{
2734   \liSyntheseUeberschrift{#1}\par
2735   \liSyntheseErklaerung{#1}
2736 }

```

```

2737 \ExplSyntaxOff
2738

```

2.45 tabelle.sty

2739 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2740 \ProvidesPackage{lehramt-informatik-tabelle}[2020/12/05 Lädt das Paket tabluarx]

2741 \RequirePackage{tabularx}

2742

2.46 typographie.sty

```

2743 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2744 \ProvidesPackage{lehramt-informatik-typographie}[2021/03/20 Typographische Makros,
2745 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
2746 formatierung.sty definiert.]
2747 \ExplSyntaxOn

```

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

```
2748 \RequirePackage{fontawesome}
```

`\liErledigt` `\liErledigt:` ☑

```
2749 \let\liErledigt=\faCheckSquareO
```

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

```
2750 \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 —

```

2751 \def\liParagraphMitLinien#1{
2752   \noindent
2753   \vrule height 2pt depth -1.6pt width 0.4cm
2754   \enspace
2755   #1
2756   \enspace
2757   \leaders\vrule height 2pt depth -1.6pt \hfill \null
2758   \par
2759   \medskip
2760 }

```

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

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

```

2761 \def\liGeschweifteKlammern#1#2#3#4{
2762   \par
2763   \medskip
2764   \noindent
2765   #1 \, $= \Bigl\{ \$
2766   \vspace{#3}
2767   #2
2768   \vspace{#4}
2769   \begin{flushright} \$\Bigr\} \$\end{flushright}
2770   \par
2771 }

```

```
2772 \ExplSyntaxOff
```

```
2773
```

2.47 uml.sty

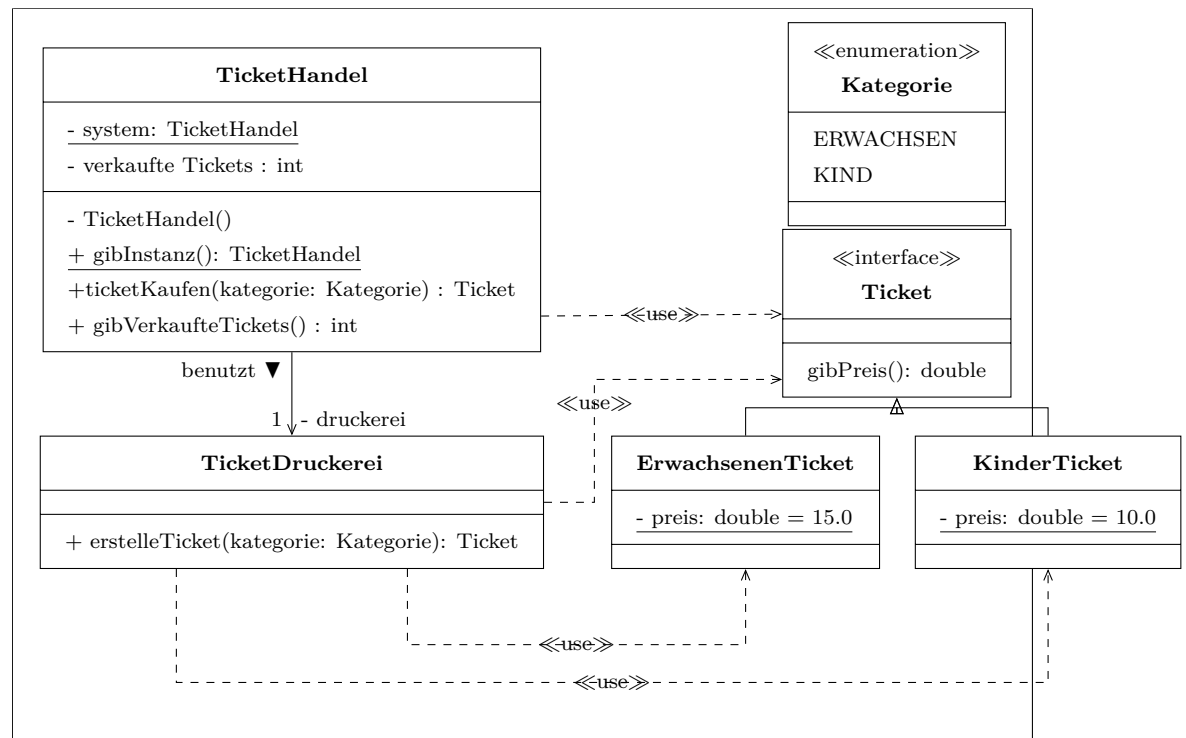
```

2774 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2775 \ProvidesPackage{lehramt-informatik-uml}[2020/06/13 Ein Hüll-Paket um
2776 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
2777 Erweiterung bereitstellt]

2778 \RequirePackage{tikz-uml}
2779 \RequirePackage{tikz-uml-activity}
2780 % Not compatible with wasysym
2781 %\RequirePackage{mathabx}
2782 \RequirePackage{wasysym}
2783 \usetikzlibrary{positioning}

2784 \tikzumlset{
2785   fill class=white!0,
2786   font=\footnotesize,
2787   fill object=white!0,
2788   fill note=white!0,
2789   fill state=white!0,
2790   % Use case
2791   fill usecase=white!0,
2792   fill system=white!0,
2793 }

```



```

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

```

```

2794 \NewDocumentCommand{ \liUmlLeserichtung } { 0{dir=right} m } {
2795   \def\@liDirLeft{
2796     \def\@liDirRight{ \RIGHTarrow}
2797     \pgfkeys{/lese/dir/.is choice}
2798     \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
2799     \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
2800     \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{\def\@liDirLeft{\LEFTarrow }}}
2801     \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
2802
2803     \def\@liPos{above}
2804     \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
2805

```

```

2806 \def\@liDistance{0cm}
2807 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}}
2808
2809 \pgfkeys{/lese/.cd,#1}
2810
2811 \node[\@liPos = \@liDistance of #2-middle] {
2812   \@liDirLeft{\footnotesize#2}\@liDirRight
2813 };
2814 }
2815

```

2.48 vollstaendige-induktion.sty

2816 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 2817 \ProvidesPackage{lehramt-informatik-vollstaendige-induktion}[2021/07/01
 2818 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
 2819 Überschriften für die einzelnen Schritte]

2.48.1 Makro-Kürzel

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

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

Lade häufig benötigte Pakete

2820 \RequirePackage{lehramt-informatik-typographie}
 2821 \RequirePackage{lehramt-informatik-mathe}
 2822 \RequirePackage{lehramt-informatik-syntax}
 2823 \ExplSyntaxOn

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

Let-Abkürzung: \let\m=\liInduktionMarkierung

2824 \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

```
2825 \def\liInduktionErklaerung#1{\scriptsize\text{#1}}
```

\liInduktionAnfang

```
2826 \def\liInduktionAnfang{
2827   \liPseudoUeberschrift{Induktionsanfang}
2828
2829   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2830   \liParagraphMitLinien{
2831     Beweise,~dass~ $A(1)$ ~eine~wahre~Aussage~ist.
2832   }
2833 }
```

\liInduktionVoraussetzung

```
2834 \def\liInduktionVoraussetzung{
2835   \liPseudoUeberschrift{Induktionsvoraussetzung}
2836
2837   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2838   \liParagraphMitLinien{
2839     Die~Aussage~ $A(k)$ ~ist~wahr~für~ein~beliebiges~ $k \in \mathbb{N}$ $.
2840   }
2841 }
```

\liInduktionSchritt

```
2842 \def\liInduktionSchritt{
2843   \liPseudoUeberschrift{Induktionsschritt}
2844
2845   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
2846   \liParagraphMitLinien{
2847     Beweise,~dass~wenn~ $A(n=k)$ ~wahr~ist,~
2848     auch~ $A(n=k+1)$ ~wahr~sein~muss.
2849   }
2850 }
```

```
2851 \ExplSyntaxOff
```

```
2852
```


2.49 wasserfall.sty

```
2853 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2854 \ProvidesPackage{lehramt-informatik-wasserfall}[2020/06/10]
2855 \RequirePackage{tikz}
2856 \tikzset{wasserfall/.style={
2857   >=stealth,
2858   node distance = 2mm and -8mm,
2859   start chain = A going below right,
2860   every node/.style = {
2861     draw,
2862     text width=24mm,
2863     minimum height=12mm,
2864     align=center,
2865     inner sep=1mm,
2866     fill=white,
2867     drop shadow={fill=black},
2868     on chain=A
2869   },
2870 }}
2871 \usetikzlibrary{chains,positioning,shadows}
2872
```

2.50 wpkalkuel.sty

2873 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2874 \ProvidesPackage{lehrant-informatik-wpkalkuel}[2020/06/13]

2.50.1 Makro-Kürzel

\let\wp=\liWpKalkuel

\let\equivalent=\liWpEquivalent

\let\erklaerung=\liWpErklaerung

2875 \RequirePackage{amsmath}

2876 \ExplSyntaxOn

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

2877 \def\liWpKalkuelOhneMathe#1#2{

2878 \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)

2879 }

2880 \def\liWpKalkuel#1#2{

2881 \ifmmode

2882 \liWpKalkuelOhneMathe{#1}{#2}

2883 \else

2884 \$\liWpKalkuelOhneMathe{#1}{#2}\$

2885 \fi

2886 }

\MatheEnv

2887 \def\MatheEnv#1{

2888 \medskip

2889

2890 \hspace{1em}#1

2891

2892 \medskip

2893 }

\Mathe

2894 \def\Mathe#1{

2895 \MatheEnv{\${#1}\$}

2896 }

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

2897 \def\liWpEquivalent#1{

2898 \MatheEnv{\$\equiv\$\hspace{1em}\${#1}\$}

2899 }

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

2900 \newlength{\@Skip@Erklaerung@Reset}

2901 \def\liWpErklaerung#1{

2902 \setlength{\@Skip@Erklaerung@Reset}{\leftskip}

2903 \setlength{\leftskip}{0.5cm}

2904

2905 \par

2906 \noindent

2907 {

2908 \scriptsize

2909 #1

2910 }

2911 \par

2912

2913 \setlength{\leftskip}{\@Skip@Erklaerung@Reset}

2914 }

\liWpErklaerungVerzweigung

```

2915 \def\liWpErklaerungVerzweigung{
2916   $\liWpKalkuelOhneMathe{if~\{-b~\}~then~\{-a1~\}~else~\{-a2~\}}{Q}
2917   \equiv
2918   (b \land \liWpKalkuelOhneMathe{a1}{Q})
2919   \lor
2920   (\neg b \land \liWpKalkuelOhneMathe{a2}{Q})$
2921 }

2922 \ExplSyntaxOff

2923

```

3 Index

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

Symbols		
\# 121	\AddToHook 1374	\bfseries .. 500, 1248,
\, 357, 412, 1195,	\advance 2496	1250, 2269, 2275,
1197, 1766, 1767,	\AfterEndEnvironment 2572	2277, 2279, 2280, 2656
1768, 2102, 2579, 2765	\allsectionsfont ... 1244	\Bigl 2765
\@Skip@Erklaerung@Reset	\Alph 1258	\Bigr 2769
... 2900, 2902, 2913	\alph 1258, 1259	\bigskip 59, 390,
\@afterheading 1733	\alpha 2679, 2681, 2682,	623, 628, 1642, 1970
\@afterindentfalse . 1733	2685, 2687, 2688,	\bool 335, 358
\@liDirLeft 2795, 2800, 2812	2689, 2690, 2691,	\bowtie
\@liDirRight 2796, 2798,	2695, 2701, 2702,	2417, 2420, 2421, 2422
2799, 2800, 2801, 2812	2707, 2708, 2709,	\Box 171
\@liDistance	2712, 2720, 2721, 2722	\boxtimes 488
... 2806, 2807, 2811	\arabic 1258, 2291, 2296,	
\@liPos .. 2803, 2804, 2811	2301, 2307, 2313, 2319	C
\\ 621, 644,	\arraystretch 1992	\c 1341, 1342
645, 648, 649, 652,		\cdot 1810, 1863, 1874
653, 745, 746, 747,	B	\centerline
854, 883, 885, 911,	\BeforeBeginEnvironment	1439, 2370, 2392, 2407
920, 965, 1007, 2571	\chapter 1248, 1249
1008, 1009, 1014,	\begin 642, 693, 708, 743,	\char 1640
1015, 1016, 1036,	767, 814, 846, 861,	\clearpage 1634
1640, 1997, 2055, 2058	881, 891, 907, 927,	\cline 621
\{ 219, 1132,	959, 976, 1005,	\clist 238, 282,
1142, 1154, 1155,	1030, 1051, 1066,	283, 303, 307, 2504
1160, 1194, 1480,	1175, 1264, 1339,	\columnbreak 2539
2042, 2447, 2765, 2916	1434, 1443, 1450,	\cs 306, 332, 356,
\} 219, 1132,	1558, 1632, 1684,	357, 394, 406, 1716
1142, 1154, 1155,	1689, 1697, 1722,	\csname 1326, 1329
1162, 1198, 1481,	1727, 1738, 1748,	\cup 1155,
2042, 2447, 2769, 2916	1752, 1835, 1839,	2076, 2688, 2702, 2709
_ 38, 46, 55, 57,	1855, 1878, 1901,	
306, 332, 356, 357,	1916, 1995, 1996,	D
371, 372, 378, 381,	2046, 2091, 2211,	\DeclareMathSymbol ..
384, 394, 406, 1375	2231, 2372, 2394, 1940, 1941
	2408, 2571, 2582, 2769	\DecoINERT
	\begingroup 1633, 2044, 2492	2294, 2367, 2368, 2390
	\beschriftung	\DecoINERTwithPivot .
 1664, 1668, 2310, 2389
	1672, 1676, 1680, 1682	\DecoLEFT 2289, 2388
A	\beta 2679,	\DecoLEFTwithPivot ..
\addbibresource	2682, 2686, 2687, 2304, 2366
1610, 1611, 1612,	2688, 2691, 2701,	\DecoRIGHT 2299, 2391, 2401
1613, 1614, 1615,	2702, 2703, 2708, 2709	\DecoRIGHTwithPivot .
1616, 1617, 1618, 1619	\bf 2269, 2270, 2271 2316, 2369

<code>\definecolor</code>	1246	<code>liKontrollflussgraph</code>	1557	1655, 1708, 1723, 1739, 1951, 2045, 2062, 2070, 2179, 2188, 2431, 2546, 2577, 2729, 2786, 2812
<code>\DefineVerbatimEnvironment</code>	2545	<code>liLernkartei</code>	1736	
<code>\delta</code> 77, 119, 177, 219, 1146		<code>liProduktionsRegeln</code>	1170	
<code>\dh</code>	1768, 2690	<code>liProjektSprache</code>	1656	<code>\footrulewidth</code> . 1275, 1592
<code>\directlua</code>		<code>liQuellen</code>	1716	<code>\foreach</code> . 1341, 1344, 1351
70, 149, 207, 212, 1127, 1141, 1161, 1169, 1176, 1181, 2027, 2032, 2085, 2092, 2099, 2459, 2558, 2590, 2595, 2596, 2600, 2605, 2606, 2610, 2616, 2617		<code>liRelationenSchemaFormat</code>	2451	<code>\forestFirst</code> . . 2519, 2522
<code>\do</code>	2290, 2295, 2300, 2305, 2311, 2317	<code>liRmodell</code>	2431	<code>\forestLast</code> . . . 2520, 2522
<code>\dots</code>	531, 535, 1480, 2218, 2701, 2702	<code>liUebergangsTabelle</code>	1992	<code>\forestOget</code> . . . 2519, 2520
<code>\DOWNarrow</code>	2799	<code>\equiv</code>	2898, 2917	<code>\forestOnes</code> 2532
<code>\draw</code>	1346, 1349, 1352, 2166, 2465, 2468	<code>\erzeuge@tiefgestellt</code>	1141, 1142, 1146	<code>\forestOv</code> 2521, 2522, 2525
		<code>\expandafter</code>		<code>\forestov</code> . 2511, 2515, 2516, 2519, 2520, 2521, 2522, 2524, 2525
		1326, 2325, 2327, 2328, 2329, 2337, 2495		<code>\forestset</code> 2507, 2530
		<code>\ExplSyntaxOff</code> . . . 61, 104, 146, 151, 204, 209, 214, 553, 575, 590, 1122, 1200, 1235, 1377, 1423, 1578, 1594, 1758, 1833, 2020, 2194, 2444, 2505, 2630, 2737, 2772, 2851, 2922		<code>\forestSortLevel</code> 2509, 2517, 2531, 2532
		<code>\ExplSyntaxOn</code> . . 35, 73, 114, 147, 172, 205, 210, 235, 495, 559, 576, 1095, 1190, 1212, 1373, 1392, 1571, 1583, 1659, 1772, 2002, 2175, 2432, 2502, 2557, 2653, 2747, 2823, 2876		<code>\frac</code> 1812, 1843, 1874, 1889
				<code>\fullouterjoin</code> 2422
				G
E				<code>\g</code> 40, 48, 282, 283, 303, 307, 313, 314, 315, 316, 317, 319, 320, 322, 323, 325, 326, 327, 328, 329, 330, 333, 337, 338, 339, 342, 344, 345, 346, 347, 348, 349, 360, 361, 362, 363, 371, 372, 374, 380, 381, 383, 384, 386, 387, 395, 400, 402, 407, 409, 413, 1394, 1395, 1396, 1399, 1403, 1407, 1411, 1412, 1413, 1414, 1415, 1417, 1418, 1420
<code>\edef</code>	1337, 2383, 2387, 2399, 2400			<code>\Gamma</code> . 118, 176, 219, 1155
<code>\else</code>	595, 603, 611, 1136, 1150, 1186, 1680, 1783, 1793, 1803, 1817, 2038, 2075, 2161, 2329, 2522, 2524, 2883			<code>\gappto</code> 2527
<code>\emph</code>	1098, 1461, 1490, 1492, 1639			<code>\geometry</code> 5
<code>\empty</code>	1680, 2074			<code>\geq</code> 1487, 2207, 2212, 2228, 2232
<code>\emptyset</code>	1984, 2696, 2717, 2750			
<code>\end</code> 681, 704, 729, 764, 798, 830, 858, 874, 888, 896, 923, 952, 973, 998, 1023, 1043, 1063, 1081, 1177, 1266, 1355, 1446, 1447, 1454, 1563, 1637, 1686, 1694, 1713, 1729, 1730, 1744, 1755, 1756, 1837, 1850, 1875, 1898, 1913, 1923, 1999, 2000, 2048, 2093, 2221, 2241, 2372, 2394, 2408, 2572, 2586, 2769		F		
<code>\endcsname</code>	1326, 1329	<code>\faCheckSquareO</code>	2749	
<code>\endgroup</code> 1636, 2049, 2497		<code>\faCircleThin</code>	1114	
<code>\enspace</code>	2754, 2756	<code>\faGg</code>	1108	
environments:		<code>\fancyfoot</code>		
<code>liAdditum</code>	1687	1270, 1271, 1272, 1587, 1588, 1589, 1590		
<code>liAHuelle</code>	2043	<code>\fancyhead</code>		
<code>liAntwort</code>	1658	1269, 1584, 1585, 1586		
<code>liDiagramm</code>	1746	<code>\faSquareO</code>	1102	H
<code>liEinbettung</code>	1657	<code>\fi</code> 597, 605, 613, 1138, 1152, 1188, 1683, 1785, 1795, 1805, 1819, 2040, 2077, 2163, 2330, 2522, 2523, 2526, 2528, 2885		<code>\hbox</code> 2417
<code>liExkurs</code>	1695			<code>\headrulewidth</code> . 1274, 1591
<code>liGraphenFormat</code>	1318	<code>\fontspec</code>	1244	<code>\headwidth</code> 1593
<code>liKasten</code>	1263	<code>\footcite</code>		<code>\hfill</code> 1859, 1865, 1870, 2757
		705, 727, 774, 797, 829, 924, 951, 997, 1459, 1462, 1469, 1474, 1479, 1483, 1489, 1494, 1599, 1851, 1852, 2007, 2224		<code>\hinweis</code> 1655
		<code>\footnote</code>	1760, 1764	<code>\hline</code> 1997
		<code>\footnotesize</code> 159, 367, 449, 545, 1119,		<code>\href</code> 396, 1208, 1764, 1927, 2585
				<code>\hspace</code> . . 2169, 2890, 2898
				<code>\ht</code> 2418
				I
				<code>\i</code> 1351, 1352
				<code>\ifcase</code> 2326
				<code>\ifmmode</code> 593, 601, 609, 1134, 1148, 1184, 1781, 1791, 1801, 1815, 2036, 2159, 2881

<code>\IfNoValueTF</code>	1217, 1220, 1221, 1222, 1223, 1229,	<code>\li@GithubLink</code>
. . . 1690, 1760, 1764	1230, 1231, 1232,	2581, 2594, 2604, 2615
<code>\ifnum</code> 2253,	1574, 1575, 1576,	<code>\li@mget</code> . 1328, 1332, 1352
2494, 2510, 2518, 2524	1719, 1720, 1721, 1728	<code>\li@minc</code> 1331, 1353
<code>\ifx</code> 1680, 2074, 2522		<code>\li@mset</code>
<code>\in</code> 519, 625,	<code>\labelenumi</code> 1259	1325, 1333, 1342, 1345
1487, 1825, 1828,	<code>\labelenumii</code> 1260	<code>\li@numdiscs</code>
1831, 1857, 1863,	<code>\labelitemi</code> 1253	. . . 1337, 1346, 1352
1868, 2207, 2218,	<code>\labelitemii</code> 1254	<code>\li@Rmodell@Schrift</code> .
2228, 2238, 2679,	<code>\labelitemiii</code> 1255	. . . 2431, 2440, 2450
2681, 2687, 2708, 2839	<code>\labelitemiv</code> 1256	<code>\li@sequence</code> . . 1338, 1351
<code>\inhaltsverzeichnis</code> <u>1631</u>	<code>\land</code> 2918, 2920	<code>\li@synthese@erklaerung@texte</code>
<code>\input</code> 17, 20,	<code>\LARGE</code> 1248 2670, 2730
23, 26, 29, 419, 1408	<code>\large</code> 1439, 2481	<code>\liAbleitung</code> <u>1169</u>
<code>\inputminted</code> 2589, 2599,	<code>\leaders</code> 2757	<code>liAdditum</code> (environment)
2609, 2621, 2624, 2628	<code>\left</code> 1775 <u>1687</u>
<code>\int</code> 2504	<code>\LEFTarrow</code> 2800	<code>liAHuelle</code> (environment)
<code>\item</code> 488,	<code>\leftarrow</code> 607 <u>2043</u>
489, 710, 714, 719,	<code>\leftouterjoin</code> 2420	<code>\liAlphabet</code> <u>1154</u>
724, 768, 777, 782,	<code>\leftskip</code> 2902, 2903, 2913	<code>liAntwort</code> (environment)
790, 862, 867, 871,	<code>\LehramtInformatikAutorEmail</code> <u>1658</u>
892, 928, 933, 940, 1590	<code>\liAnweisung</code> <u>1565</u>
948, 977, 982, 986,	<code>\LehramtInformatikAutorName</code>	<code>\liAssemblerCode</code> . . . 2619
991, 1067, 1072, 1589	<code>\liAssemblerDatei</code> . . 2620
1077, 1444, 1445,	<code>\LehramtInformatikGitBranch</code>	<code>\liAttribut</code> <u>2450</u>
1716, 1720, 1840, 399, 2565	<code>\liAttributHuelle</code> . . .
1843, 1847, 1856,	<code>\LehramtInformatikGithubCodeRepo</code>	. . . 2034, 2682, 2687
1862, 1867, 1879, 2564	<code>\liAttributHuelleOhneMathe</code>
1883, 1887, 1891,	<code>\LehramtInformatikGithubDomain</code> 2034, 2037,
1895, 1902, 1906, 2561	<code>\liAttributMenge</code>
1910, 2212, 2215,	<code>\LehramtInformatikGithubRawDomain</code> 2042, 2054, 2057,
2218, 2232, 2235, 2238 397, 2562 2064, 2065, 2079, 2081
<code>\itshape</code> 544, 2728	<code>\LehramtInformatikGithubTexRepo</code>	<code>\liAufgabe</code> <u>16</u>
 398, 2563	<code>\liAufgabenMetadaten</code> . 36
	<code>\LehramtInformatikRepository</code>	<code>\liAufgabenTitel</code> <u>62</u>
 17, 20, 23,	<code>\liAusdruck</code> <u>1191</u>
	26, 29, 1409, 1610,	<code>\liAutomat</code> <u>73</u>
	1611, 1612, 1613,	<code>\liAutomatenKante</code> . . . <u>105</u>
	1614, 1615, 1616,	<code>\liBandAlphabet</code> <u>1155</u>
	1617, 1618, 1619, 2560	<code>\liBedingung</code> <u>1566</u>
	<code>\LehramtInformatikTitel</code>	<code>\liBedingungDrei</code>
 1585	. . . 1830, 1870, 1910
	<code>\leq</code> 1874, 2215, 2235	<code>\liBedingungEins</code>
	<code>\let</code> 1144,	. . . 1824, 1859, 1902
	1145, 1634, 2366,	<code>\liBedingungFalsch</code> . <u>1568</u>
	2367, 2368, 2369,	<code>\liBedingungWahr</code> . . . <u>1567</u>
	2382, 2384, 2385,	<code>\liBedingungZwei</code>
	2386, 2388, 2389,	. . . 1827, 1865, 1906
	2390, 2391, 2401,	<code>\liBeschriftung</code> <u>1647</u>
	2493, 2531, 2532, 2749	<code>\liBindeAufgabeEin</code> . <u>1406</u>
	<code>\li@chomsky@erklaerung@texte</code>	<code>\liChomskyErklaerung</code>
 510, 546 510, 551
	<code>\li@EntwurfsCode</code>	<code>\liChomskyUeberErklaerung</code>
	638, 684, 685, 686, 549
	732, 733, 734, 735,	<code>\liChomskyUeberschrift</code>
	801, 802, 803, 804, 498, 550
	805, 806, 833, 834,	<code>\liCpmEreignis</code> <u>559</u>
	835, 836, 837, 838, 899	<code>\liCpmFruehesterI</code> . . . <u>616</u>
	<code>\li@EntwurfsCodeAllgemein</code>	<code>\liCpmSpaetesterI</code> . . . <u>615</u>
 637	<code>\liCpmVon</code> <u>599</u>
	<code>\li@fussnote@text</code> 1948,	<code>\liCpmVonOhneMathe</code> . .
	1954, 1958, 1962, 1966 599, 602, 604

\liCpmVonZu	591	\liEntwurfsFabrikmethodeUml	958, 1001	\liInduktionErklaerung	2825
\liCpmVonZuOhneMathe	591, 594, 596	\liEntwurfsKompositum	1025	\liInduktionMarkierung	2824
\liCpmVorgang	576	\liEntwurfsKompositumAkteure	1027	\liInduktionSchritt	2842
\liCpmZu	607	\liEntwurfsKompositumUml	1004, 1026	\liInduktionVoraussetzung	2834
\liCpmZuOhneMathe	607, 610, 612	\liEntwurfsModellPraesentation	1046	\liJavaCode	2579
liDiagramm (environment)	1746	\liEntwurfsModellPraesentationAkteure	1048	\liJavaDatei	639, 2588
liEinbettung (environment)	1657	\liEntwurfsModellPraesentationSteuerungAkteure	1029, 1047	\liJavaExamen	2608
\liEntwurfs	1046	\liEntwurfsModellPraesentationSteuerungUml	1083	\liJavarescorder	2598
\liEntwurfsAbstrakteFabrik	688	\liEntwurfsZustand	1065, 1085	liKasten (environment)	1263
\liEntwurfsAbstrakteFabrikCode	683, 690	\liEntwurfsZustandAkteure	1050, 1084	liKellerAutomat	114
\liEntwurfsAbstrakteFabrikUml	641, 689	\liEntwurfsZustandUml	1140	liKellerKante	152
\liEntwurfsAdapter	737	\liEpsilon	1098, 1112, 1114	liKellerUebergang	147, 153
\liEntwurfsAdapterAkteure	707, 739	\liErAttribute	1117	\liKontrollCode	1569
\liEntwurfsAdapterCode	731, 740	\liErDatenbankName	1096, 1100, 1102	liKontrollflussgraph (environment)	1557
\liEntwurfsAdapterUml	692, 738	\liErEntity	2749	\liKontrollKnotenPfad	1571
\liEntwurfsBeobachter	808	\liErledigt	1111	\liKontrollTextzeileKnoten	1570, 1575
\liEntwurfsBeobachterAkteure	766, 810	\liErMpAttribute	1099	\liKurzeTabellenLinie	621
\liEntwurfsBeobachterCode	800, 811	\liErMpEntity	1105	\liLadeAllePakete	240, 1389
\liEntwurfsBeobachterUml	742, 809	\liErMpRelationship	1097, 1106, 1108	\liLadePakete	66, 69, 236, 241, 497, 558, 1131, 1361, 1382, 1429, 1946, 2026, 2174, 2543, 2652
\liEntwurfsDekorierer	840	\liErRelationship	19	\liLatexCode	2580
\liEntwurfsDekoriererAkteure	842	\liExamensAufgabe	28	\liLeereZelle	1984
\liEntwurfsDekoriererCode	832, 843	\liExamensAufgabeA	25	liLernkartei (environment)	1736
\liEntwurfsDekoriererUml	813, 841	\liExamensAufgabeTA	22	\liLinksReduktion	2051
\liEntwurfsEinfacheFabrik	876	\liExamensAufgabeTTA	1695	\liLinksReduktionInline	2060, 2068
\liEntwurfsEinfacheFabrikAkteure	860, 878	liExkurs (environment)	489	\liMasterExkurs	1915
\liEntwurfsEinfacheFabrikUml	845, 877	\liFalsch	1201	\liMasterFaelle	1854, 1922
\liEntwurfsEinzelstueck	901	\liFlaci	2430	\liMasterFallRechnung	1900
\liEntwurfsEinzelstueckAkteure	890, 903	\liFremd	2073, 2076, 2084	\liMasterVariablen	1834, 1917
\liEntwurfsEinzelstueckCode	898, 904	\liFunktionaleAbhaengigkeit	2087	\liMasterVariablenDeklaration	1877
\liEntwurfsEinzelstueckUml	880, 902	\liFunktionaleAbhaengigkeiten	1947, 1949	\liMasterWolframLink	1925
\liEntwurfsErbauer	954	\liFussnote	1961, 1979	\liMenge	83, 84, 86, 125, 126, 127, 131, 183, 184, 185, 189, 1132, 1181, 1220, 1221
\liEntwurfsErbauerAkteure	926, 956	\liFussnoteDreiText	1953, 1973	\liMengeOhneMathe	1132, 1135, 1137
\liEntwurfsErbauerUml	906, 955	\liFussnoteEinsText	1763	\liMinimierungErklaerung	2006
\liEntwurfsFabrikmethode	1000, 1025	\liFussnoteLink	1969	\liMinispracheDatei	2623
\liEntwurfsFabrikmethodeAkteure	975, 1002	\liFussnoten	1044, 1759	\liLinespread	2438
		\liFussnoteUrl	1965, 1982	\liNichtsZuTun	2750
		\liFussnoteVierText	1957, 1976	liO	1797, 1825
		\liFussnoteZweiText	1173, 2088, 2761	liOmega	1787, 1831
		\liGeschweifteKlammern	1212	\liOmegaOhneMathe	1787, 1792, 1794
		\liGrammatik	1318		
		liGraphenFormat (environment)	1325		
		\liHanoi	2626		
		\liHaskellCode	2627		
		\liHaskellDatei	2826		
		\liInduktionAnfang			

1624, 1770, 1930, 1943, 2022, 2105, 2171, 2196, 2201, 2245, 2411, 2424, 2453, 2534, 2541, 2548, 2553, 2633, 2648, 2739, 2743, 2774, 2816, 2853, 2873	\neg 2920	\pgfkeys .. 2121, 2797, 2798, 2799, 2800, 2801, 2804, 2807, 2809	\QSpivotStep 2252, 2362, 2366, 2377
\negthinspace 1775	\newcounter ... 2265, 2266	\pgfm@count 2492, 2494, 2496	\QSr 2339
\NewDocumentCommand . 74, 115, 152, 173, 215, 236, 560, 577, 622, 627, 1191, 1213, 1449, 1572, 1600, 1759, 1763, 2087, 2098, 2165, 2588, 2598, 2608, 2620, 2623, 2627, 2794	\NewDocumentEnvironment 1170, 1263, 1318, 1557, 1656, 1657, 1660, 1687, 1695, 1717, 1736, 1746, 1993, 2043, 2433, 2451	\pgfm@declarefunction 2491	\QSSortStep 2254, 2362, 2378, 2379
\newlength 2900	\node 573, 1565, 1570, 2291, 2296, 2301, 2307, 2313, 2319, 2470, 2515, 2811	\pgfm@hint 2492	\quad 1950
\noexpand 2379, 2380, 2381, 2400, 2515	\noindent 369, 624, 629, 1643, 1645, 1649, 1653, 1681, 1709, 1711, 1724, 1740, 1742, 1750, 1919, 1972, 1975, 1978, 1981, 2446, 2752, 2764, 2906	\pgfm@parse 1332, 2509, 2514, 2517, 2531, 2532	R
\nolinkurl 402, 2585	\normalsize 1250	\pgfm@result 1333, 2492, 2493, 2495, 2497, 2510, 2518, 2531, 2532	\raisebox 1570
\notin 630	\null 2757	\pgfutil@empty 2493	\relax 1634, 2339, 2382, 2384, 2385, 2386, 2494, 2496
O	\o@join 2417, 2420, 2421, 2422	\pgfutil@loop 2494	\renewcommand ... 1253, 1254, 1255, 1256, 1259, 1260, 1274, 1275, 1591, 1592, 1992
\Omega 1788	\omega 2207, 2208, 2228, 2229	\pgfutil@repeat 2497	\repeat 2255
\or 2328	P	\prime 1431	\RequirePackage . 4, 67, 170, 234, 238, 420, 426, 427, 487, 557, 636, 637, 1091, 1093, 1094, 1130, 1239, 1240, 1243, 1245, 1247, 1252, 1261, 1268, 1279, 1280, 1289, 1293, 1294, 1295, 1323, 1324, 1372, 1390, 1391, 1430, 1499, 1599, 1604, 1605, 1621, 1628, 1629, 1630, 1658, 1773, 1934, 1935, 2107, 2109, 2199, 2259, 2261, 2413, 2414, 2415, 2428, 2456, 2487, 2538, 2544, 2551, 2556, 2567, 2568, 2636, 2741, 2748, 2778, 2779, 2781, 2782, 2820, 2821, 2822, 2855, 2875
\pagestyle 1273	\par ... 368, 389, 550, 1202, 1210, 1648, 1710, 1733, 1741, 2370, 2392, 2407, 2448, 2734, 2758, 2762, 2770, 2905, 2911	\preceq 1452	\right 1775
\paragraph 1250	\parindent 2439	\printbibliography . 1622	\RIGHTTarrow ... 2796, 2801
\path ... 106, 153, 216, 588		\ProvidesPackage 2, 15, 33, 65, 233, 424, 485, 492, 556, 619, 634, 1089, 1125, 1238, 1278, 1287, 1292, 1321, 1359, 1380, 1426, 1498, 1581, 1597, 1603, 1625, 1771, 1931, 1944, 2023, 2106, 2172, 2197, 2202, 2246, 2412, 2425, 2454, 2535, 2542, 2549, 2554, 2634, 2649, 2740, 2744, 2775, 2817, 2854, 2874	\Rightarrow 625, 630
		Q	\rightarrow 219, 514, 519, 527, 531, 533, 534, 536, 591, 599, 2169, 2679, 2686, 2688, 2691, 2696, 2701, 2702, 2707
		\QS@list 2372, 2383, 2387, 2394, 2400, 2405, 2408	\rightouterjoin ... 2421
		\QS@select@equal 2347, 2351	\Roman 1258
		\QS@select@greater 2348, 2352	\roman 1258, 1260
		\QS@select@smaller 2343, 2346, 2350	\romannumeral 2337
		\QS@sort@a 2325, 2358, 2379, 2380	\rtimes 1933
		\QS@sort@b 2325, 2326	\rule 2370, 2392, 2407, 2418
		\QS@sort@c 2329, 2336	
		\QS@sort@d 2337, 2345	S
		\QS@sort@empty . 2327, 2332	\sb 79, 89, 91, 120, 178, 526,
		\QS@sort@single 2328, 2333	
		\QS@initialize 2250, 2362, 2397	
		\QSIr . 2333, 2339, 2347, 2367, 2381, 2386, 2389	
		\QSIrr 2368, 2381, 2382, 2390	
		\QSLr 2339, 2346, 2357, 2358, 2366, 2379, 2384, 2388	

527, 531, 534, 535, 536, 1185, 1187, 1825, 1828, 1831, 1857, 1863, 2012, 2157, 2166, 2701, 2702, 2703, 2708, 2712, 2713, 2716, 2717, 2720, 2721, 2722	\scriptscriptstyle 591, 599, 607	\textit 965, 1007, 1008, 1009, 1010, 1751, 2042, 2102	\TmpTransitionEight 2118, 2139
\scriptsize 1204, 1510, 1517, 1523, 1585, 1586, 1589, 1590, 2825, 2878, 2908	\the page 1272, 1588	\textsc 1432	\TmpTransitionFive 2115, 2136
\section 57, 1400	\theparagraph 1250	\textsf 1644, 1725	\TmpTransitionFour 2114, 2135
\seq . 1574, 1575, 1576, 1719, 1720, 1721, 1728	\Theta 1778	\textstyle 1843, 1874	\TmpTransitionNine 2119, 2140
\setbox 2417	\thinspace 2878	\texttt 1119, 1432, 1566, 1567, 1568, 1569, 2878	\TmpTransitionOne 2111, 2132
\setcounter 1251, 2371, 2393, 2407	\tikz 1570	\textwidth 1593	\TmpTransitionSeven 2117, 2138
\setganttlinklabel .. 1281, 1282, 1283, 1284	tikz: bbaum 25	\the paragraph 1250	\TmpTransitionSix 2116, 2137
\setlength 1593, 2439, 2902, 2903, 2913	tikz: li binaer baum 23	\Theta 1778	\TmpTransitionTen 2120, 2141
\setmainfont 1241	\tikzchildnode 444	\thinspace 2878	\TmpTransitionThree 2113, 2134
\setmainlanguage 421	\tikzparentnode 444	\tikz 1570	\TmpTransitionTwo 2112, 2133
\setminted 2573, 2574	\tikzset 108, 155, 221, 429, 455, 1297, 1501, 2147, 2273, 2477, 2638, 2856	\tikz 1570	\TmpX 2143
\setminus 2072	\tikzumlset 2784	\tiny 1102, 1108, 1114, 1569, 1639, 2583	\TmpY 2144
\setsansfont 1242	\times 219	\titleformat .. 1248, 1250	\today 1586
\setul 2430	\tiny 1102, 1108, 1114, 1569, 1639, 2583	\titlespacing 1249	\ttfamily 2431
\sffamily 501, 1248, 1250, 1352, 2657	\tl 40, 48, 75, 76, 77, 78, 79, 80, 83, 84, 85, 86, 87, 89, 91, 116, 117, 118, 119, 120, 121, 122, 125, 126, 127, 128, 129, 130, 131, 174, 175, 176, 177, 178, 179, 180, 183, 184, 185, 186, 187, 188, 189, 304, 308, 333, 337, 338, 339, 342, 347, 348, 349, 360, 361, 362, 363, 374, 380, 383, 386, 395, 409, 561, 564, 569, 570, 578, 579, 582, 583, 1193, 1214, 1215, 1216, 1217, 1220, 1221, 1222, 1223, 1394, 1395, 1396, 1399, 1403, 1407, 1414, 1417, 1808	\ul 1097, 2429, 2430	\U 2431
\shoveleft 2052	\tmp 2074	\uml 1097, 2429, 2430	\U 2431
\shoveright 2056	\TmpPlaceEight 2129	\umlagg 1060	\U 2431
\Sigma 76, 117, 175, 1154, 1155, 1215	\TmpPlaceFive 2126	\umlassoc 1042	\U 2431
\sigma 524, 526, 527	\TmpPlaceFour 2125	\umlclass ... 643, 647, 651, 695, 696, 697, 744, 749, 754, 757, 815, 816, 817, 822, 823, 852, 882, 909, 910, 913, 964, 967, 1006, 1012, 1013, 1031, 1032, 1033, 1052, 1053, 1054, 1055	\U 2431
\SLASH 1640	\TmpPlaceNine 2130	\umldep 972	\U 2431
\small 1749	\TmpPlaceOne 2122	\umlHVHaggreg 762, 828, 1022	\U 2431
\sort 2504	\TmpPlaceSeven 2128	\umlinherit 701, 752, 917, 962, 970	\U 2431
\sortList 2503, 2512	\TmpPlaceSix 2127	\umlnote ... 703, 919, 1062	\U 2431
\square 489	\TmpPlaceTen 2131	\umlreal 699, 760	\U 2431
\stepcounter 2291, 2296, 2301, 2304, 2306, 2310, 2312, 2316, 2318	\TmpPlaceThree 2124	\umlsimpleclass .. 658, 659, 660, 664, 666, 667, 668, 694, 847, 848, 849, 908, 960, 961	\U 2431
\str .. 502, 511, 1662, 2181, 2190, 2658, 2671	\TmpPlaceTwo 2123	\umlstatic 854, 883	\U 2431
\string 2054, 2064	\TmpScale 2142	\umluniaggreg 915	\U 2431
\StrSubstitute . 2511, 2513		\umluniassoc 678, 700, 916, 1040, 1041	\U 2431
\strut 1879, 1883, 1887, 1891, 1895, 2539		\umlVHuniasoc .. 679, 680	\U 2431
\subsection 1404		\umlVHVdep 672, 673, 675, 676, 856, 857	\U 2431
\subseql 2682, 2715, 2722		\umlVHVinherit 655, 656, 661, 662, 669, 670, 825, 826, 850, 851, 1020, 1021	\U 2431
		\umlVHVreal 819, 820, 1057, 1058	\U 2431
T		\UParrow 2798	\U 2431
\tableofcontents ... 1635		\url 1760	\U 2431
\text 89, 91, 194, 2034, 2825, 2878			\U 2431
\textbf ... 1096, 1457, 1466, 1477, 1486, 1644, 1651, 1682, 1710, 1725, 1741, 1997			\U 2431
\textcolor 1569, 2824			\U 2431

<code>\usemintedstyle</code>	2570	<code>\vspace</code>		<code>\xintifEq</code>	2351
<code>\usetikzlibrary</code> . .	68,		1696, 1714, 2766, 2768	<code>\xintifForLast</code>	
	428, 1092, 1296,				2307, 2313, 2319
	1500, 2108, 2416,			<code>\xintifGt</code>	2352
	2457, 2488, 2783, 2871			<code>\xintifLt</code>	2350
		X		<code>\xintLength</code>	2325
		<code>\xappto</code> . .	2515, 2521, 2525	<code>\xintnthelt</code>	2337
		<code>\xdef</code>	1326		
		<code>\xintApply</code>	2341		
	V	<code>\xintApplyUnbraced</code> . .			
<code>\value</code>	2253		2340, 2346, 2347, 2348	Z	
<code>\varepsilon</code>	503,	<code>\xintCSVtoList</code>	2400	<code>\ZB</code>	1767
	514, 515, 1140,	<code>\xintFor</code>		<code>\zB</code>	1766
	1825, 1831, 1860, 1871		2290, 2295, 2300,	<code>\zustandsnamens@liste</code>	
<code>\vfill</code>	2539		2305, 2311, 2317, 2358		1158, 1165, 1166
<code>\vrule</code>	2753, 2757				