

Die Bschlangaul-Sammlung

Hermine Bschlangaul and Friends*

September 13, 2021

Contents

Klassen	4
aufgabe.cls	5
basis.cls	7
examen-scans.cls	8
examen.cls	10
haupt.cls	13
theorie.cls	14
Pakete	15
abmessung.sty	16
aufgaben-einbinden.sty	17
aufgaben-metadaten.sty	18
automaten.sty	19
Endlicher Automat	19
Kellerautomat	21
Turingmaschine	22
basis.sty	25
IFs	25
baum.sty	31
Binärbaum	32
AVL-Baum	33
B-Baum	34
checkbox.sty	36
chomsky-normalform.sty	37
Faulenzer	37
TeX-Markup-Grundgerüst	37
Konkretes TeX-Markup-Beispiel	37
cpm.sty	40
Faulenzer	40
TeX-Markup-Beispiel: Graph	40
TeX-Markup-Beispiel: Ergebnistabelle	40
TeX-Markup-Beispiel: Nebenrechnungstabelle „Frühester Zeitpunkt“	40
TeX-Markup-Beispiel: Nebenrechnungstabelle „Spätester Zeitpunkt“	41
Hilfsmakros für Zeitpunkt-Tabelle	42
cyk-algorithmus.sty	44
Faulenzer	44
TeX-Markup-Beispiel	44
entwurfsmuster.sty	45
Namensschema der Entwurfsmuster-Makros:	45
Reihenfolge	45

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

Abstrakte Fabrik (Abstract Factory)	45
Adapter	47
Beobachter (Observer)	48
Dekorierer (Decorator)	50
Einfache Fabrik (Simple Factory)	51
Einzelstück (Singleton)	52
Erbauer (Builder)	53
Fabrikmethode (Factory Method)	54
Kompositum (Composite)	56
Modell-Präsentation-Steuerung (Model-View-Controller)	56
Stellvertreter (Proxy)	57
Zustand (State)	58
er.sty	60
Tex-Markup-Beispiel: Komplettes Diagramm	60
Tex-Markup-Beispiel: EER Enhanced Entity-Relation-Modell nach Elmasri/Navante	61
Faulenzer	61
formale-sprachen.sty	63
formatierung.sty	66
Schriftarten / Typographie	66
Farben	66
Überschriften	66
Listen	66
Kasten	66
Header	66
Zeilenabstände	66
gantt.sty	68
grafik.sty	69
graph.sty	70
hanoi.sty	72
index.sty	73
komplexitaetstheorie.sty	74
Faulenzer	74
kontrollflussgraph.sty	76
Faulenzer	76
TeX-Markup-Beispiel	76
TikZ: pin	76
Umgebungen	77
Makros	78
kopfzeile.sty	79
literatur-dummy.sty	80
literatur.sty	81
makros.sty	82
Umgebungen, die Inhalte aus- und einblenden können	82
master-theorem.sty	87
Faulenzer	87
mathe.sty	91
meta.sty	92
Einfache Makros (Low level)	92
Zusammengesetzte Makros (High level)	93
minimierung.sty	95
normalformen.sty	98
Faulenzer	98
o-notation.sty	101
Faulenzer	101
TeX-Markup-Beispiel: Funktionsgraphen mit pgfplots	101
petri.sty	102
Faulenzer	102

potenzmengen-konstruktion.sty	104
pseudo.sty	106
pumping-lemma.sty	107
relationale-algebra.sty	108
rmodell.sty	109
Faulenzer	109
sortieren.sty	110
spalten.sty	112
sql.sty	113
Latex-Markup-Beispiel	113
struktogramm.sty	114
syntax.sty	115
Faulenzer	115
syntaxbaum.sty	118
TeX-Markup-Beispiel	119
synthese-algorithmus.sty	120
Faulenzer	120
TeX-Markup Grundgerüst	120
TeX-Markup Linksreduktion	120
TeX-Markup Rechtsreduktion	120
TeX-Markup Relationen formen	121
tabelle.sty	124
tex-dokumentation.sty	125
typographie.sty	126
uml.sty	127
vollstaendige-induktion.sty	129
Faulenzer	129
wasserfall.sty	131
Latex-Markup-Beispiel	131
wpkalkuel.sty	132
Faulenzer	132

Index	133
--------------	------------

Klassen

aufgabe.cls

```
1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesClass{bschlangaul-aufgabe}[2019/10/27 Minimale Klasse zu
3 Setzen einer Aufgabe]
4 \LoadClass{bschlangaul-basis}

Formatierung muss vor literatur sein, sonst Option clash
5 \bLadePakete{
6   formatierung,
7   abmessung,
8   literatur-dummy,
9   makros,
10  aufgaben-metadaten,
11  mathe,
12  grafik,
13  meta
14 }
```

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

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

In Aufgaben wollen wir das Additum sehen.
16 \ADDITUMtrue
17 \ExplSyntaxOn

18 \cs_gset:Npn \stichwoerter_formatiert: {
19   \tl_if_empty:NTF \g_stichwoerter_tl {} {
20     \textbf{Stichwörter:} ~
21     \g_stichwoerter_tl
22     \par
23   }
24 }

25 \cs_gset:Npn \horizontale_linie: {
26   \par
27   \noindent
28   \rule{\textwidth}{0.8pt}
29   \par
30 }

31 \cs_gset:Npn \thematik_formatiert: {
32   \tl_if_empty:NTF \g_thematik_tl {} {
33     \textit{
34       ( \g_thematik_tl )
35     }
36   }
37 }
```

\bAufgabenMetadaten

```
38 \def\bAufgabenMetadaten #1
39 {
40   \bMetaSetze{#1}
41
42   \tl_if_empty:NTF \g_einzelpruefungs_nr_tl {}
43   {
44     {
45       \noindent
46       \large
47       \_gib_einzelpruefung_trenner:
48       \par\medskip
49     }
50   }
51
52   {
53     \noindent
54     \bfseries
```

```

55 \Large
56 \tl_if_empty:NTF \g_einzelpruefungs_nr_tl
57 {
58 \g_titel_tl
59 }
60 {
61 \_gib_aufgaben_pfad_trenner:
62 }
63 }
64 \hfill \thematik_formatiert:
65 \par
66
67 \medskip
68
69 \noindent
70 {\footnotesize\stichwoerter_formatiert:}
71
72 \horizontale_linie:
73
74 \setze_kopfzeile_oben_rechts:n { \gib_aufgaben_pfad_lang_sicher: }
75
76 \bigskip
77
78 \keine_einrueckung:
79 }

80 \setze_kopf_fusszeilen:nn {} {}

81 \AddToHook { begindocument }
82 {
83 \repariere_kopfzeile_breite:
84 }

85 \AddToHook { enddocument }
86 {
87 \vfill
88 {
89 \bLogoTextProjekt
90 \bigskip
91
92 \bLogoTextCCLizenz
93 \bigskip
94
95 \begin{spacing}{1}
96 \tiny
97 \noindent
98 \bMetaHilfMit
99
100 \bMetaQuelltext
101 \_gib_github_url_href:
102 \end{spacing}
103 }
104 }

105 \ExplSyntaxOff
106

```

basis.cls

```
107 \NeedsTeXFormat{LaTeX2e}
108 \ProvidesClass{bschlangaul-basis}[2021/09/12 Basis-Klasse, die nur als
109 Unterklasse benutzt werden soll.]
```

```
110 \LoadClass[a4paper,oneside,12pt]{book}
```

Damit wir Klassen-Optionen mit \LaTeX -3 verwalten könnten.

```
111 \RequirePackage{l3keys2e}
```

```
112 \RequirePackage{bschlangaul-basis}
```

Lade Pakete die von allen Unterklassen benötigt werden.

```
113 \bLadePakete{
```

```
114   kopfzeile
```

```
115 }
```

```
116 \ExplSyntaxOn
```

```
117 \keys_define:nn { klassen-setup }
```

```
118 {
```

```
119   pakete .code:n = {
```

```
120     \bLadePakete{#1}
```

```
121   }
```

```
122 }
```

<https://tex.stackexchange.com/questions/544374/passing-options-to-a-class-and-evaluate-using-latex3-interfaces>

```
123 \ProcessKeysOptions { klassen-setup }
```

```
124 \ExplSyntaxOff
```

```
125
```

examen-scans.cls

```
126 \NeedsTeXFormat{LaTeX2e}
127 \ProvidesClass{bschlangaul-examen-scans}[2021/01/08 Zum
128 Zusammenfügen mehrerer Staatsexamen-Scans zu einer PDF-Datei.]
129 \LoadClass[a4paper,oneside]{book}

130 \LoadClass{bschlangaul-basis}

131 \RequirePackage{pdfpages}

    Formatierung muss vor literatur sein, sonst Option clash
132 \bLadePakete{formatierung}

    Komischer Option-Clash deshalb ganz am Ende, für die Silbentrennung
133 \RequirePackage[ngerman]{babel}
134 \ExplSyntaxOn

\bPruefungsNummer

135 \def\bPruefungsNummer#1{
136   \str_set:Nn \g_pruefungs_nummer {#1}
137 }

\bPruefungsTitel

138 \def\bPruefungsTitel#1{
139   \str_set:Nn \g_pruefungs_titel {#1}
140 }

141 \def\li@SansFett#1#2{
142   {
143     #1
144     \bfseries
145     \rmfamily
146     #2
147   }
148 }

\bTrennSeite

149 \def\bTrennSeite#1{
150   \clearpage
151   \strut
152   \vfill
153   \begin{center}
154   {
155     \li@SansFett{\LARGE}{\str_use:N \g_pruefungs_nummer} \\
156     \vspace{2cm}
157     \li@SansFett{\large}{\str_use:N \g_pruefungs_titel} \\
158     \vspace{5cm}
159     \li@SansFett{\Huge}{#1}
160   }
161   \end{center}
162   \vfill
163   \strut
164   \clearpage
165 }

\bTitelSeite

166 \def\bTitelSeite#1{
167   \clearpage
168   \strut
169   \vfill
170   \begin{center}
171   #1
172   \end{center}
173   \vfill
174   \strut
```



```

175 \clearpage
176 }

\BindePdfEin
177 \def\BindePdfEin#1{\includepdf[pages={1-}]{\LehramtInformatikRepository/#1}}

178 \AtBeginDocument{
179   \thispagestyle{empty}
180   \bTitelSeite{
181     \li@SansFett{\Large}{Sammlung~aller~Staatsexamensaufgaben~der~Prüfungsnummer} \\
182
183     \vspace{4cm}
184
185     \li@SansFett{\Huge}{\str_use:N \g_pruefungs_nummer} \\
186
187     \vspace{4cm}
188
189     \li@SansFett{\LARGE}{\str_use:N \g_pruefungs_titel} \\
190   }
191 }

192 \ExplSyntaxOff
193

```

examen.cls

```
194 \NeedsTeXFormat{LaTeX2e}
195 \ProvidesClass{bschlangaul-examen}[2021/06/10 Zum Einbinden von
196 mehreren Aufgaben zu einem Examen.]
```

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

```
197 \LoadClass[pakete={
198   formatierung,
199   literatur-dummy,
200   makros,
201   aufgaben-einbinden,
202   aufgaben-metadaten,
203   abmessung,
204   typographie,
205   grafik,
206   meta
207 }]{bschlangaul-basis}
```

Formatierung für die Überschriften setzen.

```
208 \RequirePackage{titlesec}
209 \titleformat{\section}{\huge\filcenter\bfseries}{\thesection}{1em}{}
210 \newcommand{\sectionbreak}{\clearpage}
211 \setcounter{secnumdepth}{0}
212 \bLadeAllePakete
```

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

```
213 \RequirePackage[ngerman]{babel}
214 \ExplSyntaxOn
```

\bSetzeExamenThemaNr

```
215 \def\bSetzeExamenThemaNr#1{
216   \tl_gset:Nn \g_thema_nr_tl { #1 }
217   \section{Thema~Nr.~#1}
218 }
```

\bSetzeExamenTeilaufgabeNr

```
219 \def\bSetzeExamenTeilaufgabeNr#1{
220   \tl_gset:Nn \g_teilaufgabe_nr_tl { #1 }
221   \subsection{Teilaufgabe~Nr.~#1}
222 }
```

\bBindeAufgabeEin

```
223 \def\bBindeAufgabeEin#1{
224   \tl_gset:Nn \g_aufgabe_nr_tl { #1 }
225   \input{
226     \LehramtInformatikRepository /
227     Staatsexamen /
228     \g_einzelpruefungs_nr_tl /
229     \g_jahr_tl /
230     \g_monat_tl /
231     \tl_if_empty:NTF \g_thema_nr_tl {} {
232       Thema - \g_thema_nr_tl /
233     }
234     \tl_if_empty:NTF \g_teilaufgabe_nr_tl {} {
235       Teilaufgabe - \g_teilaufgabe_nr_tl /
236     }
237     Aufgabe - \g_aufgabe_nr_tl .tex
238   }
239 }
```

\bAufgabenMetadaten Das Metadaten-Makro überschreiben

```
240 \def\bAufgabenMetadaten#1{
241   \bMetaSetze{#1}
```

```

242 \subsubsection{\_gib_aufgaben_titel:}
243 }

244 \cs_new:Npn \titel_seite:
245 {
246   \titel_seite:nn
247   {
248     {
249       \bfseries\Huge
250
251       \g_einzelpruefungs_nr_tl
252       \par
253
254       \g_jahreszeit_tl
255       \par
256
257       \g_jahr_tl
258       \par
259     }
260   }
261   {
262     \g_examen_fach_tl
263     \par
264
265     \vspace{0.5cm}
266
267     Aufgabenstellungen~mit~Lösungsvorschlägen
268     \par
269   }
270 }
271 }

272 \cs_new:Npn \inhalts_verzeichnis: {
273   % für den Abstand vor den section im Inhaltsverzeichnis
274   % https://tex.stackexchange.com/questions/241445/how-to-control-spacing-in-toc-for-different-sections
275   \setlength{\cftbeforesecskip}{1.5cm}
276   \setlength{\cftbeforesubsecskip}{0.5cm}
277   \renewcommand{\cftsubsecafterpnum}{\vspace{\cftbeforesubsecskip}}
278   \setlength{\cftbeforesubsubsecskip}{0.1cm}
279   \renewcommand{\contentsname}{Aufgabenübersicht}
280   \tableofcontents
281 }

282 \setcounter{tocdepth}{4}
283 \RequirePackage[titles]{tocloft}
284 \AddToHook { begin:document }
285 {
286   \titel_seite:
287
288   \clearpage
289
290   \inhalts_verzeichnis:
291
292   \vfill
293
294   \bLogoTextProjekt
295   \bigskip
296
297   \bLogoTextCCLizenz
298   \bigskip
299
300   \clearpage
301 }

302 \ExplSyntaxOff

```


haupt.cls

```
304 \NeedsTeXFormat{LaTeX2e}
305 \ProvidesClass{bschlangaul-haupt}[2019/10/27 Klasse zum
306 Setzen der Haupt-Datei, Bschlangaul-Sammlung.tex d.h. des wichtigsten
307 Dokuments, das alles zusammenfasst.]
```

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

clash

```
308 \LoadClass[pakete={
309   formatierung,
310   literatur-dummy,
311   makros,
312   aufgaben-einbinden,
313   aufgaben-metadaten,
314   abmessung,
315   typographie,
316   grafik,
317   meta,
318   index
319 }]{bschlangaul-basis}
320 \bLadeAllePakete
```

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

```
321 \RequirePackage[ngerman]{babel}
322 \ExplSyntaxOn
323 \setze_kopf_fusszeilen:nn {} {}
324 \AddToHook { begindocument }
325 {
326   \repariere_kopfzeile_breite:
327   \titel_seite:nn
328   {
329     {
330       \Huge
331       Die~komplette~Sammlung
332     }
333
334   }
335   {
336     Alle~Aufgaben
337   }
338
339   \tableofcontents
340   \clearpage
341
342   \pagestyle{fancy}
343 }
```

`\bAufgabenMetadaten` Das Metadaten-Makro überschreiben

```
344 \def\bAufgabenMetadaten#1{
345   \bMetaSetze{#1}
346   \subsection{\gib_aufgaben_pfad_lang_sicher:}
347 }

348 \ExplSyntaxOff
349
```

theorie.cls

```
350 \NeedsTeXFormat{LaTeX2e}
351 \ProvidesClass{bschlangaul-theorie}[2021/09/12 Klasse zum
352 Setzen von Theorie-Zusammenfassungen]
353 \LoadClass{bschlangaul-basis}

    Paket „formatierung“ muss vor „literatur“ sein, sonst Option clash
354 \bLadePakete{
355   formatierung,
356   literatur,
357   makros,
358   aufgaben-metadaten
359 }

    Komischer Option-Clash deshalb ganz am Ende, für die Silbentrennung
360 \RequirePackage[ngerman]{babel}
361
```

Pakete

abmessung.sty

```
362 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
363 \ProvidesPackage{bschlangaul-baum}[2021/09/04 Einstellung der
364 Seitenabmessung mit Hilfe des geometry-Pakets.]

365 \RequirePackage{geometry}

366 \geometry{
367   a4paper,
368   margin=2cm,
369   includeheadfoot,
370   % showframe,
371   % showcrop,
372   % verbose=true,
373 }

374
```


aufgaben-einbinden.sty

```
375 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
376 \ProvidesPackage{bschlangaul-aufgaben-einbinden}[2020/06/13
377 Binde Aufgaben in ein größeres Dokument ein.]
```

Mit Hilfe des standalone-Pakets können eigenständige Dokumente eingebunden werden, die auch uneingebunden kompiliert werden können, wie zum Beispiel Aufgaben

```
378 \RequirePackage{standalone}
```

`\bAufgabe` Eine Aufgaben mit den Pfad relativ zu Wurzelverzeichnis dieses Repository einbinden.

```
379 \def\bAufgabe#1{
380   \input{\LehramtInformatikRepository/#1.tex}
381 }
```

`\bExamensAufgabe` Eine Examensaufgaben mit den Pfad relativ zu Staatsexamen einbinden, z. B. `\bExamensAufgabe{46116/2/Teilaufgabe-1/Aufgabe-3}`

```
382 \def\bExamensAufgabe#1{
383   \input{\LehramtInformatikRepository/Staatsexamen/#1.tex}
384 }
```

`\bExamensAufgabeTTA`

```
385 \def\bExamensAufgabeTTA #1 / #2 / #3 : Thema #4 Teilaufgabe #5 Aufgabe #6 {
386   \input{\LehramtInformatikRepository/Staatsexamen/#1/#2/#3/Thema-#4/Teilaufgabe-
      #5/Aufgabe-#6.tex}
387 }
```

`\bExamensAufgabeTA`

```
388 \def\bExamensAufgabeTA #1 / #2 / #3 : Thema #4 Aufgabe #5 {
389   \input{\LehramtInformatikRepository/Staatsexamen/#1/#2/#3/Thema-#4/Aufgabe-#5.tex}
390 }
```

`\bExamensAufgabeA`

```
391 \def\bExamensAufgabeA #1 / #2 / #3 : Aufgabe #4 {
392   \input{\LehramtInformatikRepository/Staatsexamen/#1/#2/#3/Aufgabe-#4.tex}
393 }
```

```
394
```

aufgaben-metadaten.sty

```
395 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
396 \ProvidesPackage{bschlangaul-aufgaben-metadaten}[2020/07/07 Makros zum
397 Setzen der Aufgaben-Metadaten.]

398 \ExplSyntaxOn

\MetaSetze Setze die Metadaten einer Aufgabe.

399 \def\MetaSetze#1{
400   \_setze_variablen_zurueck:
401
402   \tl_clear:N \g_teilaufgabe_nr_tl
403
404   \keys_set:nn { aufgabenmetadaten } {
405     #1
406   }
407
408   \_setze_relativen_pfad:
409 }

\AufgabenMetadaten 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.

\AufgabenMetadaten{
  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,
}

410 \def\AufgabenMetadaten#1{
411   \MetaSetze{#1}
412
413   \_gib_examen_titel: {}
414
415   \section{\_gib_aufgaben_titel:}
416 }

417 \ExplSyntaxOff

\AufgabenTitel Momentan eine dummy Makro das die Thematik enthält.

418 \def\AufgabenTitel#1{}

419
```

automaten.sty

420 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
 421 \ProvidesPackage{bschlangaul-automaten}[2021/02/14 Zum Setzen von Automaten]

Endlicher Automat

```
422 \bLadePakete{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}
```



```
423 \RequirePackage{tikz}
424 \usetikzlibrary{arrows,automata,positioning}
425 \bLadePakete{mathe}
426 \directlua{
427   automaten = require('bschlangaul-automaten')
428 }
```

\bAutomat \bAutomat[⟨*automaten-name*⟩]{⟨*zustaende*=*Z*,*alphabet*=*Σ*,*delta*=*δ*,*ende*=*E*,*start*=*z*₀⟩}

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

```
429 \ExplSyntaxOn
430 \NewDocumentCommand {\bAutomat} { O{A} m } {
431   \tl_set:Nn \l_zustaende_tl {Z}
432   \tl_set:Nn \l_alphabet_tl {\Sigma}
433   \tl_set:Nn \l_delta_tl {\delta}
434   \tl_set:Nn \l_ende_tl {E}
435   \tl_set:Nn \l_start_tl {z\sb{0}}
436   \tl_set:Nn \l_typ_tl {}
437
438   \keys_define:nn { automat } {
439     zustaeende .code:n = {\tl_set:Nn \l_zustaende_tl {\bMenge{##1}}},
440     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\bMenge{##1}}},
441     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
442     ende .code:n = {\tl_set:Nn \l_ende_tl {\bMenge{##1}}},
```

```

443   start .code:n = {\tl_set:Nn \l_start_tl {##1}},
444   dea .value_forbidden:n = true,
445   dea .code:n = {\tl_set:Nn \l_typ_tl {\sb{\text{DEA}}}},
446   nea .value_forbidden:n = true,
447   nea .code:n = {\tl_set:Nn \l_typ_tl {\sb{\text{NEA}}}},
448 }
449
450 \keys_set:nn { automat } { #2 }
451
452 $#1 \l_typ_tl = (
453   \l_zustaende_tl,
454   \l_alphabet_tl,
455   \l_delta_tl,
456   \l_ende_tl,
457   \l_start_tl
458 )$
459 }
460 \ExplSyntaxOff

```

`\bAutomatenKante` **Makro-Faulenzer:** `\let\k=\liAutomatenKante`

```

461 \def\bAutomatenKante#1#2#3#4{
462   \path (#1) edge[#4] node{#3} (#2);
463 }
464 \tikzset{
465   li automat/.style={
466     ->,
467     node distance=2cm
468   },
469 }

```

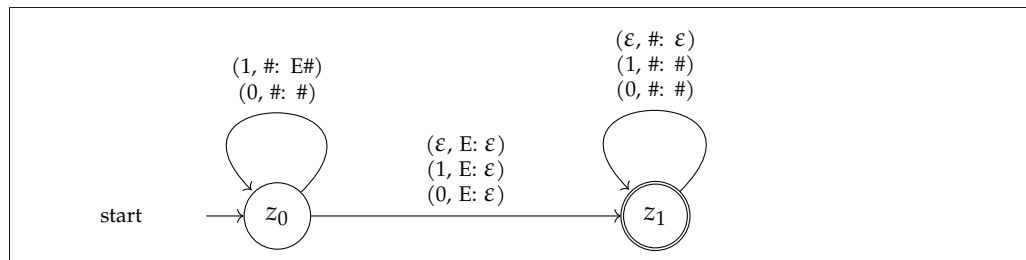
Kellerautomat

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

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

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

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



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

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

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

```
470 \ExplSyntaxOn
471 \NewDocumentCommand {\bkellerAutomat} { O{K} m } {
472   \tl_set:Nn \l_zustaeende_tl {Z}
473   \tl_set:Nn \l_alphabet_tl {\Sigma}
474   \tl_set:Nn \l_kelleralphabet_tl {\Gamma}
475   \tl_set:Nn \l_delta_tl {\delta}
476   \tl_set:Nn \l_start_tl {z\sb{0}}
477   \tl_set:Nn \l_kellerboden_tl {\#}
478   \tl_set:Nn \l_ende_tl {E}
479
480   \keys_define:nn { kellerautomat } {
481     zustaeende .code:n = {\tl_set:Nn \l_zustaeende_tl {\bMenge{##1}}},
482     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\bMenge{##1}}},
483     kelleralphabet .code:n = {\tl_set:Nn \l_kelleralphabet_tl {\bMenge{##1}}},
484     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
485     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
```

```

486   kellerboden .code:n = {\tl_set:Nn \l_kellerboden_tl {##1}},
487   ende .code:n = {\tl_set:Nn \l_ende_tl {\bMenge{##1}}},
488 }
489
490 \keys_set:nn { kellerautomat } { #2 }
491
492 $#1 = (
493   \l_zustaende_tl,
494   \l_alphabet_tl,
495   \l_kelleralphabet_tl,
496   \l_delta_tl,
497   \l_start_tl,
498   \l_kellerboden_tl,
499   \l_ende_tl
500 )$
501 }
502 \ExplSyntaxOff

```

`\bKellerUebergang` **Makro-Faulenzer:** `\let\u=\liKellerUebergang`
`\bKellerUebergang{a, KELLERBODEN, A; b KELLERBODEN, EPSILON}`

(a, #: A)
(b, #: ϵ)

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

```

503 \ExplSyntaxOn
504 \def\bKellerUebergang#1{
505   \directlua{automaten.drucke_keller_uebergaenge('#1')}
506 }
507 \ExplSyntaxOff

```

`\bKellerKante` `\bKellerKante[\langle tikz-optionen \rangle]{ \langle von \rangle }{ \langle zu \rangle }{ \langle übergänge \rangle }`
Makro-Faulenzer: `\let\k=\liKellerKante`

```

508 \NewDocumentCommand{\bKellerKante} { 0{above} m m m } {
509   \path (#2) edge[#1] node{\bKellerUebergang{#4}} (#3);
510 }
511 \tikzset{
512   li keller knoten/.style={
513     text width=2cm,
514     align=center,
515     font=\footnotesize,
516   },
517   li kellerautomat/.style={
518     li automat,
519     every edge/.append style={
520       every node/.style={
521         li keller knoten
522       }
523     }
524   }
525 }

```

Turingmaschine

```
526 \RequirePackage{amssymb}
```

`\bTuringLeerzeichen`

□

```
527 \def\bTuringLeerzeichen{\Box}
```

```

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

\beginTuringMaschine{
  zustaeende = \{ z_0, z_1, z_2 \},
  alphabet = \{ a, b, c \},
  bandalphabet = \{ \beginTuringLeerzeichen, A \},
  ende = \{ z_2 \},
}

```

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

```

528 \ExplSyntaxOn
529 \NewDocumentCommand \beginTuringMaschine { O{TM} m } {
530   \tl_set:Nn \l_zustaeende_tl {Z}
531   \tl_set:Nn \l_alphabet_tl {\Sigma}
532   \tl_set:Nn \l_bandalphabet_tl {\Gamma}
533   \tl_set:Nn \l_delta_tl {\delta}
534   \tl_set:Nn \l_start_tl {z\sb{0}}
535   \tl_set:Nn \l_leerzeichen_tl {\beginTuringLeerzeichen}
536   \tl_set:Nn \l_ende_tl {E}
537
538   \keys_define:nn { kellerautomat } {
539     zustaeende .code:n = {\tl_set:Nn \l_zustaeende_tl {\bMenge{##1}}},
540     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\bMenge{##1}}},
541     bandalphabet .code:n = {\tl_set:Nn \l_bandalphabet_tl {\bMenge{##1}}},
542     delta .code:n = {\tl_set:Nn \l_delta_tl {##1}},
543     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
544     leerzeichen .code:n = {\tl_set:Nn \l_leerzeichen_tl {##1}},
545     ende .code:n = {\tl_set:Nn \l_ende_tl {\bMenge{##1}}},
546   }
547
548   \keys_set:nn { kellerautomat } { #2 }
549
550   $\text{#1} = (
551     \l_zustaeende_tl,
552     \l_alphabet_tl,
553     \l_bandalphabet_tl,
554     \l_delta_tl,
555     \l_start_tl,
556     \l_leerzeichen_tl,
557     \l_ende_tl
558   )$
559 }
560 \ExplSyntaxOff

```

`\beginTuringUebergangZelle` Formatiert einen Zustandsübergang für eine Übergangstabelle.
Makro-Faulenzer: `\let\t=\liTuringUebergangZelle`
`\beginTuringUebergangZelle{z_1, LEER, R}:(z_1: \square, R) \beginTuringUebergangZelle{z_1, leer, l}:`
`(z_1: \square, L)`

```

561 \ExplSyntaxOn
562 \def\beginTuringUebergangZelle#1{
563   \directlua{tex.print(automaten.gib_einen_turing_uebergang('#1'))}
564 }
565 \ExplSyntaxOff

```

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

Makro-Faulenzer: `\let\t=\liTuringUebergaenge`
`(z_1: \square, L)`
`(\square: \square, R)`

```

566 \ExplSyntaxOn
567 \def\bTuringUebergaenge#1{
568   \directlua{automaten.drucke_turing_uebergaenge('#1')}
569 }
570 \ExplSyntaxOff

\bTuringKante \bTuringKante[⟨tikz-optionen⟩]{⟨zustand-oder-lese⟩}{⟨schreibe⟩}{⟨richtung⟩}
Makro-Faulenzer: \let\t=\liTuringKante

571 \NewDocumentCommand{\bTuringKante} { 0{above} m m m } {
572   \path (#2) edge[#1] node{\bTuringUebergaenge{#4}} (#3);
573 }

\bTuringUeberfuehrung

574 \def\bTuringUeberfuehrung{
575    $\delta : Z \times \Gamma \rightarrow Z \times \Gamma \times \{L, R, N\}$ 
576 }

577 \tikzset{
578   li turingmaschine/.style={
579     li automat,
580     every edge/.append style={
581       every node/.style={
582         li keller knoten
583       }
584     }
585   }
586 }

587

```


basis.sty

```
588 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
589 \ProvidesPackage{bschlangaul-basis}[2020/11/27]

590 \ExplSyntaxOn
```

IFs

Wir weichen von dem üblichen Namensschema ab und schreiben das Wort komplett in Großbuchstaben, damit die if-Befehle schöner lesbar sind, z. B. `\ifADDITUM`, `\ADDITUMtrue` und `\ADDITUMfalse`.

```
\ifADDITUM
\ADDITUMtrue 591 \newif\ifADDITUM
\ADDITUMfalse 592 \ADDITUMfalse

\ifEXKURS
\EXKURStrue 593 \newif\ifEXKURS
\EXKURStue 594 \EXKURStrue

\ifANTWORT
\ANTWORTtrue 595 \newif\ifANTWORT
\ANTWORTfalse 596 \ANTWORTtrue

\bLadePakete

597 \NewDocumentCommand{\bLadePakete}{ m }
598 {
599   \clist_map_inline:nn { #1 } { \RequirePackage{bschlangaul-##1} }
600 }

\bLadeAllePakete

601 \def\bLadeAllePakete{
602   \bLadePakete{
603     aufgaben-einbinden,
604     automaten,
605     checkbox,
606     chomsky-normalform,
607     cpm,
608     cyk-algorithmus,
609     entwurfsmuster,
610     er,
611     formale-sprachen,
612     gantt,
613     grafik,
614     graph,
615     hanoi,
616     kontrollflussgraph,
617     komplexitaetstheorie,
618     makros,
619     master-theorem,
620     mathe,
621     minimierung,
622     normalformen,
623     petri,
624     potenzmengen-konstruktion,
625     pumping-lemma,
626     pseudo,
627     quicksort,
628     relationale-algebra,
629     rmodell,
630     sortieren,
631     spalten,
632     struktogramm,
```

```

633     sql,
634     syntax,
635     syntaxbaum,
636     synthese-algorithmus,
637     tabelle,
638     typographie,
639     uml,
640     vollstaendige-induktion,
641     wasserfall,
642     wpkalkuel,
643     %
644     baum, % am Schluss sonst Fehler: undefined command \edge
645 }
646 }

```

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.

```

647 \clist_new:N \g_schluessel_clist
648 \clist_set:Nn \g_schluessel_clist {
649   titel,
650   thematik,
651   stichwoerter,
652   zitat_schluessel,
653   zitat_beschreibung,
654   %
655   bearbeitungs_stand,
656   korrektheit,
657   %
658   relativer_pfad,
659   identische_aufgabe,
660   %
661   einzelpruefungs_nr,
662   examen_fach,
663   jahr,
664   monat,
665   jahreszeit,
666   thema_nr,
667   teilaufgabe_nr,
668   aufgabe_nr,
669 }

```

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

```

670 \clist_map_inline:Nn \g_schluessel_clist {
671   \tl_new:c {g_auf_#1_tl}
672 }

```

Funktion über alle globalen Token-List-Variablen zurückzusetzen.

```

673 \cs_new:Npn \setze_variablen_zurueck: {
674   \clist_map_inline:Nn \g_schluessel_clist {
675     \tl_clear:c {g_auf_##1_tl}
676   }
677 }

```

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

```

678 \keys_define:nn { aufgabenmetadaten }
679 {
680   Titel .tl_gset:N = \g_titel_tl,
681   Thematik .tl_gset:N = \g_thematik_tl,
682   Stichwoerter .tl_gset:N = \g_stichwoerter_tl,
683   ZitatSchluessel .tl_gset:N = \g_zitat_beschreibung_tl,
684   ZitatBeschreibung .tl_gset:N = \g_zitat_schluessel_tl,

```

```

685 %
686 BearbeitungsStand .tl_gset:N = \g_bearbeitungs_stand_tl,
687 Korrektheit .tl_gset:N = \g_korrekttheit_tl,
688 Ueberprueft .tl_gset:N = \g_ueberprueft_tl,
689 %
690 RelativerPfad .tl_gset:N = \g_relativer_pfad_tl,
691 IdentischeAufgabe .tl_gset:N = \g_identische_aufgabe_tl,
692 %
693 ExamenNummer .tl_gset:N = \g_einzelpruefungs_nr_tl,
694 ExamenFach .tl_gset:N = \g_examen_fach_tl,
695 ExamenJahr .tl_gset:N = \g_jahr_tl,
696 ExamenMonat .tl_gset:N = \g_monat_tl,
697 ExamenJahreszeit .tl_gset:N = \g_jahreszeit_tl,
698 ExamenThemaNr .tl_gset:N = \g_thema_nr_tl,
699 ExamenTeilaufgabeNr .tl_gset:N = \g_teilaufgabe_nr_tl,
700 ExamenAufgabeNr .tl_gset:N = \g_aufgabe_nr_tl,
701 }

702 \cs_gset:Npn \_setze_relativen_pfad: {
703   \tl_if_empty:NTF \g_relativer_pfad_tl
704   {
705     \bool_if:nTF
706     {
707       ! \tl_if_empty_p:N \g_einzelpruefungs_nr_tl &&
708       ! \tl_if_empty_p:N \g_jahr_tl &&
709       ! \tl_if_empty_p:N \g_monat_tl
710     }
711     {
712       \tl_gset:Nn \g_relativer_pfad_tl {
713         Staatsexamen /
714         \g_einzelpruefungs_nr_tl /
715         \g_jahr_tl /
716         \g_monat_tl /
717         \tl_if_empty:NTF \g_thema_nr_tl {} { Thema - \g_thema_nr_tl / }
718         \tl_if_empty:NTF \g_teilaufgabe_nr_tl {} { Teilaufgabe - \g_teilaufgabe_nr_tl / }
719         \tl_if_empty:NTF \g_aufgabe_nr_tl {} { Aufgabe - \g_aufgabe_nr_tl .tex }
720       }
721     }
722   }{}
723 }
724 {}
725 }

726 \cs_set:Nn \_trenner: {
727   \, / \,
728 }

729 \cs_gset:Npn \_gib_jahreszeit_durch_monat: #1 {
730   % \str_case funktioniert nicht mit den Tokenlist variablen.
731   \tl_case:Nn { #1 }
732   {
733     { 3 } { Frühjahr }
734     { 03 } { Frühjahr }
735     { 9 } { Herbst }
736     { 09 } { Herbst }
737   }
738 }

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

739 \cs_gset:Npn \_gib_examen_fach_durch_nummer: #1 {
740   \tl_case:Nn { #1 }
741   {
742     { 46110 } { Grundlagen der Informatik (nicht vertieft) }
743     { 46111 } { Programmentwicklung / Systemprogrammierung / Datenbanksysteme (nicht vertieft) }
744     { 46112 } { Grundlagen der Informatik (nicht vertieft) }
745     { 46113 } { Theoretische Informatik (nicht vertieft) }

```

```

746 { 46114 } { Algorithmen / Datenstrukturen / Programmiermethoden (nicht vertieft) }
747 { 46115 } { Theoretische Informatik / Algorithmen / Datenstrukturen (nicht vertieft) }
748 { 46116 } { Softwaretechnologie / Datenbanksysteme (nicht vertieft) }
749 { 46118 } { Fachdidaktik (Mittelschulen) }
750 { 46119 } { Fachdidaktik (Realschulen) }
751 { 46121 } { Fachdidaktik (berufliche Schulen) }
752 { 66110 } { Automatentheorie, Algorithmische Sprache (vertieft) }
753 { 66111 } { Betriebssysteme / Datenbanksysteme / Rechnerarchitektur (vertieft) }
754 { 66112 } { Automatentheorie / Komplexität / Algorithmen (vertieft) }
755 { 66113 } { Rechnerarchitektur / Datenbanken / Betriebssysteme (vertieft) }
756 { 66114 } { Datenbank- und Betriebssysteme (vertieft) }
757 { 66115 } { Theoretische Informatik / Algorithmen (vertieft) }
758 { 66116 } { Datenbanksysteme / Softwaretechnologie (vertieft) }
759 { 66118 } { Fachdidaktik (Gymnasium) }
760 }
761 }

```

„Einzelprüfungsnummer / Jahr / Jahreszeit“ mit Trennzeichen

```

762 \cs_gset:Npn \_gib_einzelpruefung_trenner: {
763   \g_einzelpruefungs_nr_tl
764
765   \_trenner:
766
767   \g_jahr_tl
768
769   \_trenner:
770
771   \_gib_jahreszeit_durch_monat: \g_monat_tl
772 }

```

Thema 1 / Teilaufgabe 2 / Aufgabe 3

```

773 \cs_gset:Npn \_gib_aufgaben_pfad_trenner: {
774   \tl_if_empty:NTF \g_thema_nr_tl {} {
775     Thema ~ \g_thema_nr_tl \_trenner:
776   }
777   \tl_if_empty:NTF \g_teilaufgabe_nr_tl {} {
778     Teilaufgabe ~ \g_teilaufgabe_nr_tl \_trenner:
779   }
780   \tl_if_empty:NTF \g_aufgabe_nr_tl {} {
781     Aufgabe ~ \g_aufgabe_nr_tl
782   }
783 }

```

Gib den langen Titelpfad einer Aufgabe. Ist die Aufgabe keine Examensaufgabe, wird auch eine Titel zurückgegeben. Für die Kopfzeile gedacht.

```

784 \cs_new:Npn \gib_aufgaben_pfad_lang_sicher: {
785   \tl_if_empty:NTF \g_einzelpruefungs_nr_tl
786   {
787     \g_titel_tl
788   }
789   {
790     \_gib_einzelpruefung_trenner:
791     \_trenner:
792     \_gib_aufgaben_pfad_trenner:
793   }
794 }

795 \cs_gset:Npn \_gib_examen_titel:
796 {
797   \cs_set:Nn \_trenner: { \, / \, }
798   \bool_if:nTF
799   {
800     ! \tl_if_empty_p:N \g_einzelpruefungs_nr_tl &&
801     ! \tl_if_empty_p:N \g_jahr_tl &&
802     ! \tl_if_empty_p:N \g_monat_tl &&

```

```

803 ! \tl_if_empty_p:N \g_aufgabe_nr_tl
804 }
805 {
806 {
807 \footnotesize
808 \par
809 \noindent
810 Staatsexamen ~
811 \g_einzelpuefungs_nr_tl \_trenner:
812 \g_jahr_tl \_trenner:
813
814 \tl_case:Nn \g_monat_tl
815 {
816 { 03 } { Frühjahr }
817 { 09 } { Herbst }
818 } \_trenner:
819
820 \tl_if_empty:NTF \g_thema_nr_tl {} {
821 Thema ~ Nr. ~ \g_thema_nr_tl \_trenner:
822 }
823 \tl_if_empty:NTF \g_telaufgabe_nr_tl {} {
824 Teilaufgabe ~ Nr. ~ \g_telaufgabe_nr_tl \_trenner:
825 }
826 \tl_if_empty:NTF \g_aufgabe_nr_tl {} {
827 Aufgabe ~ Nr. ~ \g_aufgabe_nr_tl
828 }
829 \par
830 \bigskip
831 }
832 }
833 }

834 \cs_new:Npn \_gib_github_url: {
835 \LehramtInformatikGithubDomain /
836 \LehramtInformatikGithubTexRepo /
837 blob /
838 \LehramtInformatikGitBranch /
839 \g_relativer_pfad_tl
840 }

841 \cs_new:Npn \_gib_github_url_href: {
842 \tl_if_empty:NTF \g_relativer_pfad_tl {} {
843 \url{ \_gib_github_url: }
844 }
845 }

846 \cs_new:Npn \_gib_aufgaben_titel: {
847 \g_titel_tl
848
849 \tl_if_empty:NTF \g_thematik_tl
850 {}
851 {
852 \, ~ [
853 \g_thematik_tl
854 ]
855 }
856 }

857 \cs_new:Npn \titel_seite:nn #1 #2
858 {
859 \pagestyle{empty}
860
861 \strut
862 \vspace{1cm}
863
864 \begin{center}

```

```

865     \bfseries
866     \Large
867     #1
868     \par
869 \end{center}
870
871 \vfill
872
873 \begin{center}
874     \large
875     #2
876     \par
877 \end{center}
878
879 \vfill
880
881 \begin{center}
882     \bGrafikLogo[width=8cm]
883     \par
884
885     \vspace{4cm}
886
887     {
888         \bfseries
889         \Large
890         \bMetaBschlangaulSammlung
891     }
892     \par
893
894     \medskip
895
896     {
897         \large
898         \bMetaHermineFriends
899     }
900     \par
901 \end{center}
902
903 \vspace{2cm}
904
905 \strut
906
907 \clearpage
908 }
909 \cs_new:Npn \keine_einrueckung: {
910     \par
911     \@afterindentfalse
912     \@afterheading
913 }
914 \msg_new:nnn { aufgabenmetadaten } { kein-titel }
915 { Der~Schlüssel~ist~zwingend~notwendig }
916 \def\LehramtInformatikRepository{/pfad/zum/repository}
917 % \RequirePackage{polyglossia}
918 % \setmainlanguage{german}
919

```

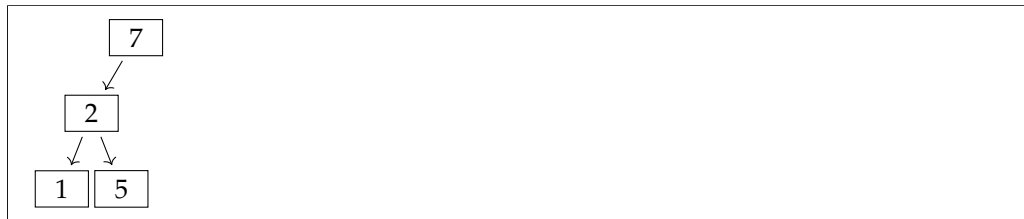
baum.sty

```
920 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
921 \ProvidesPackage{bschlangaul-baum}[2020/06/13 Zum Setzen von
922 Binär- und AVL-Bäumen. Hüll-Paket um TikZ and tikz-qtree.]
923 \RequirePackage{tikz}
    für li binaer baum
924 \RequirePackage{tikz-qtree}
    Für b baum
925 \usetikzlibrary{shapes.multipart}
```

Binärbaum

TikZ-Stil: li binaer baum: Knoten als Kreise, Kanten als Pfeile.

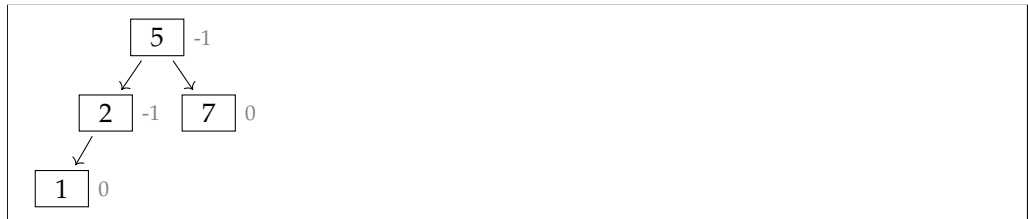
```
\begin{tikzpicture}[li binaer baum]
\Tree
[.7
  [.2
    [.1 ]
    [.5 ]
  ]
\edge[blank]; \node[blank]{};
]
\end{tikzpicture}
```



```
926 \tikzset{
927   li binaer baum/.style={
928     shorten <=2pt,
929     shorten >=2pt,
930     ->,
931     every tree node/.style={
932       minimum width=2em,
933       draw,
934       rectangle
935     },
936     blank/.style={
937       draw=none
938     },
939     edge from parent/.style={
940       draw,
941       edge from parent path={(\tikzparentnode) -- (\tikzchildnode)}
942     },
943     level distance=1cm,
944     every label/.style={
945       gray,
946       font=\footnotesize,
947       label position=0,
948       label distance=0cm,
949     }
950   },
951 }
```


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}
```

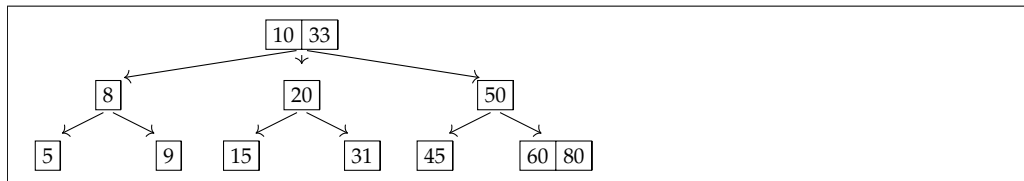


B-Baum

```

\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}

```



```

952 \tikzset{
953   li bbaum knoten/.style={
954     rectangle split parts=10,
955     rectangle split,
956     rectangle split horizontal,
957     rectangle split ignore empty parts,
958     draw,
959     fill=white
960   },
961   li bbaum/.style={
962     every node/.style={
963       li bbaum knoten
964     },
965     level 1/.style={
966       level distance=12mm,
967       sibling distance=25mm,
968     },
969     every child/.style={
970       shorten <= 2pt,
971       shorten >= 6pt,
972       ->,
973     },
974     level 2/.style={
975       level distance=9mm,
976       sibling distance=15mm,
977     },
978   }
979 }

```

`bBaum` `\begin{bBaum}{beschriftung}\end{bBaum}`: Zum Einbetten eines Baums.

```

980 \NewDocumentEnvironment { bBaum }{ m +b }
981 {
982   {
983     \small
984     \noindent
985     \textit{#1}:
986   }
987   \begin{center}
988     #2
989     \medskip
990     \end{center}
991 } {}

992

```

checkbox.sty

```
993 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
994 \ProvidesPackage{bschlangaul-checkbox}[2020/12/14 Zum Setzen von
995 Multiple-Choice-Fragen. Simulation von Kästchen zum ankreuzen]
996 \RequirePackage{amssymb}

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

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

999
```

chomsky-normalform.sty

```
1000 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1001 \ProvidesPackage{bschlangaul-chomsky-normalform}[2021/03/26
1002 Hilfsmakros zum Setzen des Algorithmus zum Erreichen der Chomsky-Normalform]

1003 \ExplSyntaxOn

1004 \bLadePakete{typographie}
```

Faulenzer

```
\let\erklaerung=\bChomskyErklaerung
\let\schritt=\bChomskyUeberschrift
\let\schrittE=\bChomskyUeberErklaerung
```

TeX-Markup-Grundgerüst

```
\let\schrittE=\bChomskyUeberErklaerung
\begin{enumerate}
\item \schrittE{1}
\item \schrittE{2}
\item \schrittE{3}
\item \schrittE{4}
\end{enumerate}
```

Konkretes TeX-Markup-Beispiel

```
\begin{enumerate}
\item \schrittE{1}

\bNichtsZuTun

\item \schrittE{2}

\begin{bProduktionsRegeln}
S -> d S e | a | U c T | S b U,
T -> d S e | a,
U -> d S e | a | U c T,
\end{bProduktionsRegeln}

\item \schrittE{3}

\begin{bProduktionsRegeln}
S -> D S E | a | U C T | S B U,
T -> D S E | a,
U -> D S E | a | U C T,
B -> b,
C -> c,
D -> d,
E -> e,
\end{bProduktionsRegeln}

\item \schrittE{4}
% S -> S S.1 | T2 S.2 | a | U S.3
% T -> T2 S.2 | a
% U -> T2 S.2 | a | U S.3
% T1 -> b
% T2 -> d
% T3 -> e
% T4 -> c
% S.1 -> T1 U
% S.2 -> S T3
```

```

% S.3 -> T4 T

\begin{bProduktionsRegeln}
S -> D S_E | a | U C_T | S B_U, % S -> S S.1 | T2 S.2 | a | U S.3
T -> D S_E | a, % T -> T2 S.2 | a
U -> D S_E | a | U C_T, % U -> T2 S.2 | a | U S.3
B -> b, % T1 -> b
C -> c, % T4 -> c
D -> d, % T2 -> d
E -> e, % T3 -> e
S_E -> S E, % S.2 -> S T3
C_T -> C T, % S.3 -> T4 T
B_U -> B U, % S.1 -> T1 U
\end{bProduktionsRegeln}
\end{enumerate}

```

\bChomskyUeberschrift Makro-Faulenzer: \let\schritt=\liChomskyUeberschrift

```

1005 \def\bChomskyUeberschrift#1{
1006   {
1007     \bfseries
1008     \rmfamily
1009     \str_case:nn {#1} {
1010       {1} {Elimination~der~$\varepsilon$-Regeln}
1011       {2} {Elimination~von~Kettenregeln}
1012       {3} {Separation~von~Terminalzeichen}
1013       {4} {Elimination~von~mehrelementigen~Nonterminalketten}
1014     }
1015   }
1016 }

```

\bChomskyErklaerung Makro-Faulenzer: \let\erklaerung=\liChomskyErklaerung
Hoffmann Seite 180

```

1017 \def\li@chomsky@erklaerung@texte#1{
1018   \str_case:nn {#1} {
1019     %
1020     {1} {
1021       Alle~Regeln~der~Form~$A\rightarrow\varepsilon$~werden~eliminiert.~
1022       Die~Ersetzung~von~$A$~wird~durch~$\varepsilon$~in~allen~anderen~
1023       Regeln~vorweggenommen.
1024     }
1025     {2} {
1026       Jede~Produktion~der~Form~$A\rightarrow B$~mit~$A, B$~in~$S$~wird~
1027       als~Kettenregel~bezeichnet.~Diese~tragen~nicht~zur~Produktion~
1028       von~Terminalzeichen~bei~und~lassen~sich~ebenfalls~eliminieren.
1029     }
1030     {3} {
1031       Jedes~Terminalzeichen~$\sigma$,~das~in~Kombination~mit~anderen~
1032       Symbolen~auftaucht,~wird~durch~ein~neues~Nonterminal~
1033       $S_{\sigma}$~ersetzt~und~die~Menge~der~Produktionen~durch~die~
1034       Regel~$S_{\sigma}\rightarrow\sigma$~ergänzt.
1035     }
1036     {4} {
1037       Alle~Produktionen~der~Form~
1038       $A\rightarrow B_{\{1\}}B_{\{2\}}\dots B_{\{n\}}$~
1039       werden~in~die~Produktionen~
1040       $A\rightarrow$
1041       $A_{\{n-1\}}B_{\{n\}}, A_{\{n-1\}}\rightarrow$
1042       $A_{\{n-2\}}B_{\{n-1\}}, \dots, $
1043       $A_{\{2\}}\rightarrow B_{\{1\}}B_{\{2\}}$~zerteilt.~
1044       Nach~der~Ersetzung~sind~alle~längeren~Nonterminalketten~
1045       vollständig~heruntergebrochen~und~die~Chomsky-Normalform~erreicht.
1046   }

```

```

1047 }
1048 }
1049 \def\bChomskyErklaerung#1{
1050 {
1051   \itshape
1052   \footnotesize
1053   \bParagraphMitLinien{\li@chomsky@erklaerung@texte{#1}}
1054 }
1055 }

```

\bChomskyUeberErklaerung **Makro-Faulenzer:** \let\schritteE=\liChomskyUeberErklaerung

```

1056 \def\bChomskyUeberErklaerung#1{
1057   \bChomskyUeberschrift{#1}\par
1058   \bChomskyErklaerung{#1}
1059 }

```

```

1060 \ExplSyntaxOff
1061

```

cpm.sty

```

1062 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1063 \ProvidesPackage{bschlangaul-cpm}[2020/09/03]
1064 \RequirePackage{tikz}

1065 \bLadePakete{mathe,typographie}

```

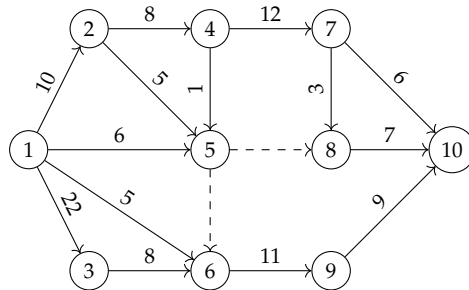
Faulenzer

```

\let\f=\footnotesize
\let\FZ=\bCpmFruehI
\let\SZ=\bCpmSpaetI
\let\v=\bCpmVon
\let\ vz=\bCpmVonZu
\let\z=\bCpmZu

```

TeX-Markup-Beispiel: Graph



```

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

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

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

```

TeX-Markup-Beispiel: Ergebnistabelle

```

\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}

```

TeX-Markup-Beispiel: Nebenrechnungstabelle „Frühester Zeitpunkt“

```

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

```



```

3      &                                     & 18   \\
4      &                                     & 7    \\
5      &                                     & 19   \\
6      &                                     & 26   \\
7      & $\max(19_3, 22_4)$                  & 22   \\
8      & $\max(30_5, 30_6, 28_7)$ & 30   \\ \\hline
\end{tabular}

```

TeX-Markup-Beispiel: Nebenrechnungstabelle „Spätester Zeitpunkt“

```

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

```

```

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

```

```

1066 \ExplSyntaxOn
1067 \NewDocumentCommand { \bCpmEreignis } { 0{} m m m } {
1068   \tl_set:Nn \l_name_tl {}
1069
1070   \keys_define:nn { cpmEreignis } {
1071     name .code:n = {\tl_set:Nn \l_name_tl {##1}},
1072   }
1073
1074   \keys_set:nn { cpmEreignis } { #1 }
1075
1076   \tl_if_empty:NT \l_name_tl {
1077     \tl_set:Nn \l_name_tl {#2}
1078   }
1079
1080   \node[circle,draw] (\l_name_tl) at (#3,#4) {#2};
1081 }
1082 \ExplSyntaxOff

```

```

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

```

```

1083 \ExplSyntaxOn
1084 \NewDocumentCommand { \bCpmVorgang } { 0{} m m m } {
1085   \tl_set:Nn \l_schein_tl {}
1086   \tl_set:Nn \l_kritisch_tl {}
1087
1088   \keys_define:nn { cpmVorgang } {
1089     schein .code:n = {\tl_set:Nn \l_schein_tl {dashed}},
1090     kritisch .code:n = {\tl_set:Nn \l_kritisch_tl {very~thick}},
1091   }
1092
1093   \keys_set:nn { cpmVorgang } { #1 }
1094
1095   \path[->,\l_schein_tl,\l_kritisch_tl] (#2) edge node[auto,sloped] {#4} (#3);
1096 }
1097 \ExplSyntaxOff

```

Hilfsmakros für Zeitpunkt-Tabelle

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

```
\bCpmVonZu Makro-Faulenzer: \let\vz=\liCpmVonZu
\bCpmVonZu{1}(2-3): 1(2→3)
1098 \def\bCpmVonZuOhneMathe#1(#2-#3){#1_{\scriptscriptstyle(#2\rightarrow#3)}}
1099 \def\bCpmVonZu#1(#2-#3){%
1100 \ifmmode%
1101 \bCpmVonZuOhneMathe{#1}(#2-#3)%
1102 \else%
1103 $\bCpmVonZuOhneMathe{#1}(#2-#3)$%
1104 \fi%
1105 }
```

```
\bCpmVon Makro-Faulenzer: \let\v=\liCpmVon
\bCpmVon{1}(2): 1(→2)
1106 \def\bCpmVonOhneMathe#1(#2){#1_{\scriptscriptstyle(\rightarrow#2)}}
1107 \def\bCpmVon#1(#2){%
1108 \ifmmode%
1109 \bCpmVonOhneMathe{#1}(#2)%
1110 \else%
1111 $\bCpmVonOhneMathe{#1}(#2)$%
1112 \fi%
1113 }
```

```
\bCpmZu Makro-Faulenzer: \let\z=\liCpmZu
\bCpmZu{1}(2): 1(←2)
1114 \def\bCpmZuOhneMathe#1(#2){#1_{\scriptscriptstyle(\leftarrow#2)}}
1115 \def\bCpmZu#1(#2){%
1116 \ifmmode%
1117 \bCpmZuOhneMathe{#1}(#2)%
1118 \else%
1119 $\bCpmZuOhneMathe{#1}(#2)$%
1120 \fi%
1121 }

1122 \ExplSyntaxOn
```

```
\bCpmSpaetI Spätester Zeitpunkt, zu dem Ereignis i eintreten kann
Makro-Faulenzer: \let\SZ=\liCpmSpaetI
1123 \NewDocumentCommand{ \bCpmSpaetI } { 0{i} } {
1124 \ifmmode
1125 SZ\sb{#1}
1126 \else
1127 $SZ\sb{#1}$
1128 \fi
1129 }
```

```
\bCpmFruehI Frühester Zeitpunkt, zu dem Ereignis i eintreten kann.
Makro-Faulenzer: \let\FZ=\liCpmFruehI
1130 \NewDocumentCommand{ \bCpmFruehI } { 0{i} } {
1131 \ifmmode
1132 FZ\sb{#1}
1133 \else
1134 $FZ\sb{#1}$
1135 }
```

```

1135 \fi
1136 }

```

\bCpmFruehErklaerung

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

```

1137 \def\bCpmFruehErklaerung{
1138   \bParagraphMitLinien{
1139     Wir~führen~eine~Vorwärtsterminierung~durch~
1140     und~addieren~die~Dauern.~
1141
1142     Kann~ein~Ereignis~über~mehrere~Vorgänge~erreicht~
1143     werden,~wählen~wir~das~Maximum~aus.~
1144
1145     \textbf{Erläuterungen:}~
1146
1147      $i$ :~
1148     Ereignis~ $i$ ;~\,
1149
1150     \bCpmFruehI{}:~
1151     Frühester~Zeitpunkt,~zu~dem~Ereignis~ $i$ ~eintreten~kann
1152     .
1153   }
1154 }

```

\bCpmSpaetErklaerung

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

```

1155 \def\bCpmSpaetErklaerung{
1156   \bParagraphMitLinien{
1157     Wir~führen~eine~Rückwärtsterminierung~durch~
1158     und~subtrahieren~die~Dauern~vom~letzten~Ereignis~aus.~
1159
1160     Kann~ein~Ereignis~über~mehrere~Vorgänge~erreicht~
1161     werden,~wählen~wir~das~Minimum~aus.~
1162
1163     \textbf{Erläuterungen:}~
1164
1165      $i$ :~
1166     Ereignis~ $i$ ;~\,
1167
1168     \bCpmSpaetI{}:~
1169     Spätester~Zeitpunkt,~zu~dem~Ereignis~ $i$ ~eintreten~kann
1170     .
1171   }
1172 }

```

```

1173 \ExplSyntaxOff

```

```

1174

```

cyk-algorithmus.sty

```
1175 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1176 \ProvidesPackage{bschlangaul-cyk-algorithmus}[2021/06/18 Hilfsmakros
1177 zum Setzen des CYK-Algorithmus]
```

Faulenzer

```
\let\l=\bKurzeTabellenLinie
```

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}
\bWortInSprache{acbcab}
```

```
\bKurzeTabellenLinie Makro-Faulenzer: \let\l=\liKurzeTabellenLinie
```

```
1178 \def\bKurzeTabellenLinie#1{\\cline{1-#1}}
```

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

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

```
1179 \NewDocumentCommand{ \bWortInSprache } { m O{L(G)} } {
1180   \bigskip
1181   \noindent
1182    $\Rightarrow$  #1 \in #2$
1183 }
```

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

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

```
1184 \NewDocumentCommand{ \bWortNichtInSprache } { m O{L(G)} } {
1185   \bigskip
1186   \noindent
1187    $\Rightarrow$  #1 \notin #2$
1188 }
```

```
1189
```

entwurfsmuster.sty

```
1190 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1191 \ProvidesPackage{bschlangaul-entwurfsmuster}[2021/05/06
1192 Hilfsmakros zum Setzen von Entwurfsmustern / Design Patterns]
```

Namensschema der Entwurfsmuster-Makros:

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

Reihenfolge

- (a) Beschreibung Kurze Beschreibung des Entwurfsmusters, z. B \bEntwurfsEinzelstueckBeschreibung
- (b) Uml: Uml-Klassendiagramm, z. B \bEntwurfsEinzelstueckUml
- (c) Akteure: Akteure, beteiligte Klassen, z. B \bEntwurfsEinzelstueckAkteure
- (d) Code: Allgemeines Code-Beispiel, z. B \bEntwurfsEinzelstueckCode
- (e) ohne: Ohne Suffix, Bündelung der einzelnen Makros eines Entwurfsmusters
 \bEntwurfsEinzelstueckAkteure

```
1193 \RequirePackage{bschlangaul-uml}
```

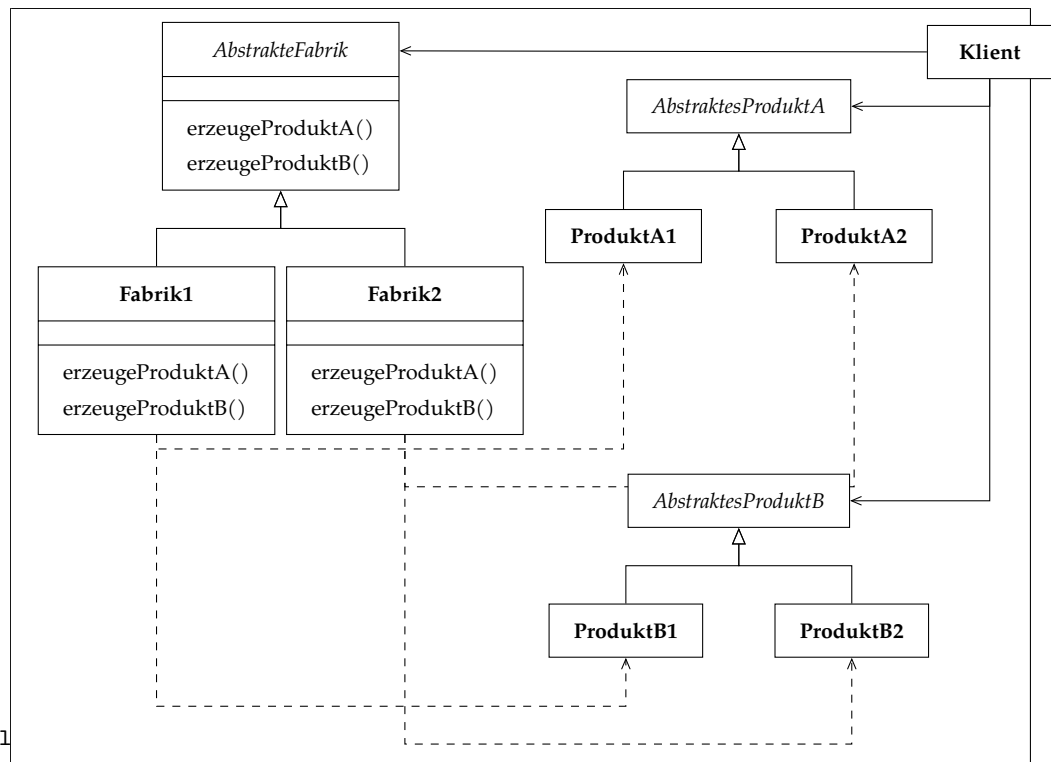
\li@EntwurfsCodeAllgemein Allgemeine Code-Beispiele zu den UML-Diagrammen und Stellvertretern

```
1194 \RequirePackage{bschlangaul-syntax}
1195 \def\li@EntwurfsCode#1#2{
1196   \bJavaDatei{entwurfsmuster/#1/allgemein/#2}
1197 }
```

Abstrakte Fabrik (Abstract Factory)

sAbstrakteFabrikBeschreibung

```
1198 \def\bEntwurfsAbstrakteFabrikBeschreibung{
1199   Es wird eine Schnittstelle bereitgestellt, um \emph{Familien
1200   verbundener oder abhängiger Objekte} zu erstellen, ohne die konkreten
1201   Klassen zu spezifizieren.\footcite[Seite 25]{eilebrecht}
1202 }
```



```

1203 \def\bentwurfsAbstrakteFabrikUml{
1204   \begin{tikzpicture}
1205     \umlclass[type=abstract]{AbstrakteFabrik}{}{
1206       erzeugeProduktA()\n
1207       erzeugeProduktB()\n
1208     }
1209     \umlclass[below left=1cm and -1.5cm of AbstrakteFabrik]{Fabrik1}{}{
1210       erzeugeProduktA()\n
1211       erzeugeProduktB()\n
1212     }
1213     \umlclass[below right=1cm and -1.5cm of AbstrakteFabrik]{Fabrik2}{}{
1214       erzeugeProduktA()\n
1215       erzeugeProduktB()\n
1216     }
1217     \umlVHVinherit{Fabrik1}{AbstrakteFabrik}
1218     \umlVHVinherit{Fabrik2}{AbstrakteFabrik}
1219
1220     \umlsimpleclass[right=3cm of AbstrakteFabrik,type=abstract]{AbstraktesProduktA}
1221     \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktA]{ProduktA1}
1222     \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktA]{ProduktA2}
1223     \umlVHVinherit{ProduktA1}{AbstraktesProduktA}
1224     \umlVHVinherit{ProduktA2}{AbstraktesProduktA}
1225
1226     \umlsimpleclass[above right=0cm and 1cm of AbstraktesProduktA]{Klient}
1227
1228     \umlsimpleclass[below=4.5cm of AbstraktesProduktA,type=abstract]{AbstraktesProduktB}
1229     \umlsimpleclass[below left=1cm and -1cm of AbstraktesProduktB]{ProduktB1}
1230     \umlsimpleclass[below right=1cm and -1cm of AbstraktesProduktB]{ProduktB2}
1231     \umlVHVinherit{ProduktB1}{AbstraktesProduktB}
1232     \umlVHVinherit{ProduktB2}{AbstraktesProduktB}
1233
1234     \umlVHVdep[arm1=-1.3cm,arm2=-1cm]{Fabrik1}{ProduktA1}
1235     \umlVHVdep[arm2=-1cm]{Fabrik1}{ProduktB1}
1236
1237     \umlVHVdep[arm1=-1.8cm,arm2=-1.5cm]{Fabrik2}{ProduktA2}
1238     \umlVHVdep[arm2=-1.5cm]{Fabrik2}{ProduktB2}
1239

```

```

1240 \umluniassoc[anchor2=25]{Klient}{AbstrakteFabrik}
1241 \umlVHuniassoc[arm1=-1cm]{Klient}{AbstraktesProduktA}
1242 \umlVHuniassoc{Klient}{AbstraktesProduktB}
1243 \end{tikzpicture}
1244 }

```

bEntwurfsAbstrakteFabrikCode

```

1245 \def\bEntwurfsAbstrakteFabrikCode{
1246 \li@EntwurfsCode{abstrakte_fabrik}{Produkte}
1247 \li@EntwurfsCode{abstrakte_fabrik}{AbstrakteFabrik}
1248 \li@EntwurfsCode{abstrakte_fabrik}{Klient}
1249 }

```

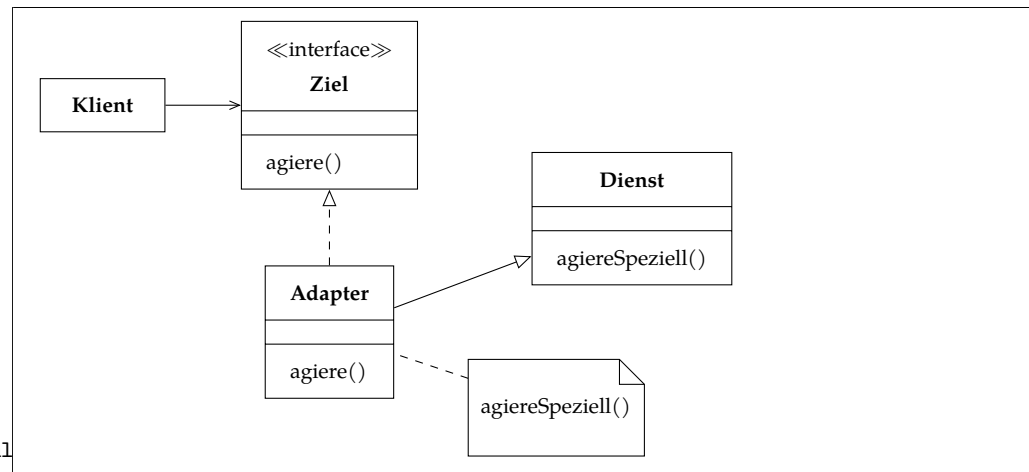
\bEntwurfsAbstrakteFabrik

```

1250 \def\bEntwurfsAbstrakteFabrik{
1251 \bEntwurfsAbstrakteFabrikBeschreibung
1252
1253 \bEntwurfsAbstrakteFabrikUml
1254
1255 \bEntwurfsAbstrakteFabrikCode
1256 }

```

Adapter



\bEntwurfsAdapterUml

```

1257 \def\bEntwurfsAdapterUml{
1258 \begin{tikzpicture}
1259 \umlsimpleclass[x=1,y=3]{Klient}{}{}
1260 \umlclass[x=4,y=3,type=interface]{Ziel}{}{agiere()}
1261 \umlclass[x=4,y=0]{Adapter}{}{agiere()}
1262 \umlclass[x=8,y=1.5]{Dienst}{}{agiereSpeziell()}
1263
1264 \umlreal{Adapter}{Ziel}
1265 \umluniassoc{Klient}{Ziel}
1266 \umlinherit{Adapter}{Dienst}
1267
1268 \umlnote[x=7,y=-1,width=2cm]{Adapter}{agiereSpeziell()}
1269 \end{tikzpicture}
1270 \footcite[so ähnlich wie GoF]{wiki:adapter}
1271 }

```

\bEntwurfsAdapterAkteure

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.

```

1272 \def\bEntwurfsAdapterAkteure{
1273   \begin{description}
1274
1275     \item[Ziel (Target)]
1276
1277     Das Ziel definiert die Schnittstelle, die der Klient nutzen kann.
1278
1279     \item[Klient (Client)]
1280
1281     Der Klient nutzt Dienste über inkompatible Schnittstellen und greift
1282     dabei auf adaptierte Schnittstellen zurück.
1283
1284     \item[Dienst (Adaptee)]
1285
1286     Der Dienst bietet wiederzuverwendende Dienstleistungen mit fest
1287     definierter Schnittstelle an.
1288
1289     \item[Adapter]
1290
1291     Der Adapter adaptiert die Schnittstelle des Dienstes auf die
1292     Schnittstelle zum Klienten.\footcite{wiki:adapter}
1293
1294   \end{description}
1295 }
```

\bEntwurfsAdapterCode

```

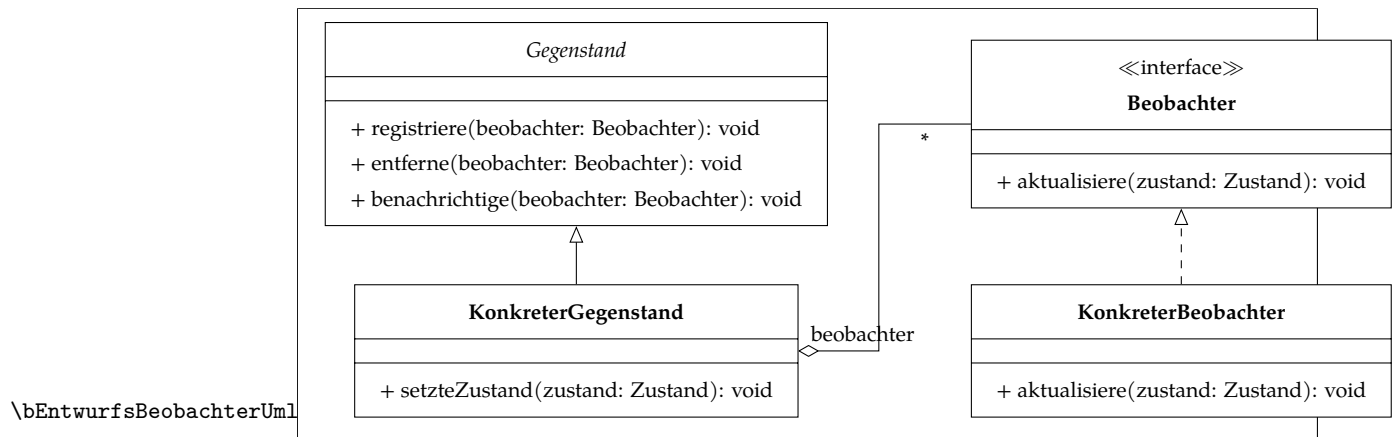
1296 \def\bEntwurfsAdapterCode{
1297   \li@EntwurfsCode{adapter}{Dienst}
1298   \li@EntwurfsCode{adapter}{Ziel}
1299   \li@EntwurfsCode{adapter}{Adapter}
1300   \li@EntwurfsCode{adapter}{Klient}
1301 }
```

\bEntwurfsAdapter

```

1302 \def\bEntwurfsAdapter{
1303   \bEntwurfsAdapterUml
1304   \bEntwurfsAdapterAkteure
1305   \bEntwurfsAdapterCode
1306 }
```

Beobachter (Observer)



```

1307 \def\bEntwurfsBeobachterUml{
```



```

1308 \begin{tikzpicture}
1309   \umlclass[x=0,y=0,type=abstract]{Gegenstand}{-}{-}{
1310     + registriere(beobachter: Beobachter): void\\
1311     + entferne(beobachter: Beobachter): void\\
1312     + benachrichtige(beobachter: Beobachter): void\\
1313   }
1314   \umlclass[x=0,y=-3]{KonkreterGegenstand}{-}{-}{
1315     + setzeZustand(zustand: Zustand): void
1316   }
1317   \umlinherit{KonkreterGegenstand}{Gegenstand}
1318
1319   \umlclass[x=8,y=0,type=interface]{Beobachter}{-}{-}{
1320     + aktualisiere(zustand: Zustand): void
1321   }
1322   \umlclass[x=8,y=-3]{KonkreterBeobachter}{-}{-}{
1323     + aktualisiere(zustand: Zustand): void
1324   }
1325   \umlreal{KonkreterBeobachter}{Beobachter}
1326
1327   \umlHVGgreg[arg1=beobachter,pos1=0.8,mult2=*,pos2=2.5]
1328   {KonkreterGegenstand}{Beobachter}
1329 \end{tikzpicture}
1330 }

```

\bEntwurfsBeobachterAkteure

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.

```

1331 \def\bEntwurfsBeobachterAkteure{
1332   \begin{description}
1333     \item[Gegenstand / Subjekt (Subject / Observable)]
1334
1335     Ein Subjekt (beobachtbares Objekt, auf Englisch publisher, also
1336     „Veröffentlicher“, genannt) hat eine Liste von Beobachtern, ohne
1337     deren konkrete Typen zu kennen. Es bietet eine Schnittstelle zur An-
1338     und Abmeldung von Beobachtern und eine Schnittstelle zur
1339     Benachrichtigung von Beobachtern über Änderungen an.\footcite[Seite
1340     251]{gof}
1341
1342     \item[Beobachter (Observer)]
1343
1344     Die Beobachter (auf Englisch auch subscriber, also „Abonnent“,
1345     genannt) definieren eine Aktualisierungsschnittstelle.
1346
1347     \item[konkreter/s Gegenstand / Subjekt (ConcreteSubject / ConcreteObservable)]
1348

```

```

1349 Ein konkretes Subjekt (konkretes, beobachtbares Objekt) speichert
1350 den relevanten Zustand und benachrichtigt alle Beobachter bei
1351 Zustandsänderungen über deren Aktualisierungsschnittstelle. Es
1352 verfügt über eine Schnittstelle zur Erfragung des aktuellen
1353 Zustands.
1354
1355 \item[Konkrete Beobachter (ConcreteObserver)]
1356
1357 Konkrete Beobachter verwalten die Referenz auf ein konkretes
1358 Subjekt, dessen Zustand sie beobachten und speichern und dessen
1359 Zustand konsistent ist. Sie implementieren eine
1360 Aktualisierungsschnittstelle unter Verwendung der
1361 Abfrageschnittstelle des konkreten Subjekts.
1362 \footcite{wiki:beobachter}
1363 \end{description}
1364 }

```

\bEntwurfsBeobachterCode

```

1365 \def\bEntwurfsBeobachterCode{
1366   \li@EntwurfsCode{beobachter}{Gegenstand}
1367   \li@EntwurfsCode{beobachter}{KonkreterGegenstand}
1368   \li@EntwurfsCode{beobachter}{Beobachter}
1369   \li@EntwurfsCode{beobachter}{KonkreterBeobachterA}
1370   \li@EntwurfsCode{beobachter}{KonkreterBeobachterB}
1371   \li@EntwurfsCode{beobachter}{Klient}
1372 }

```

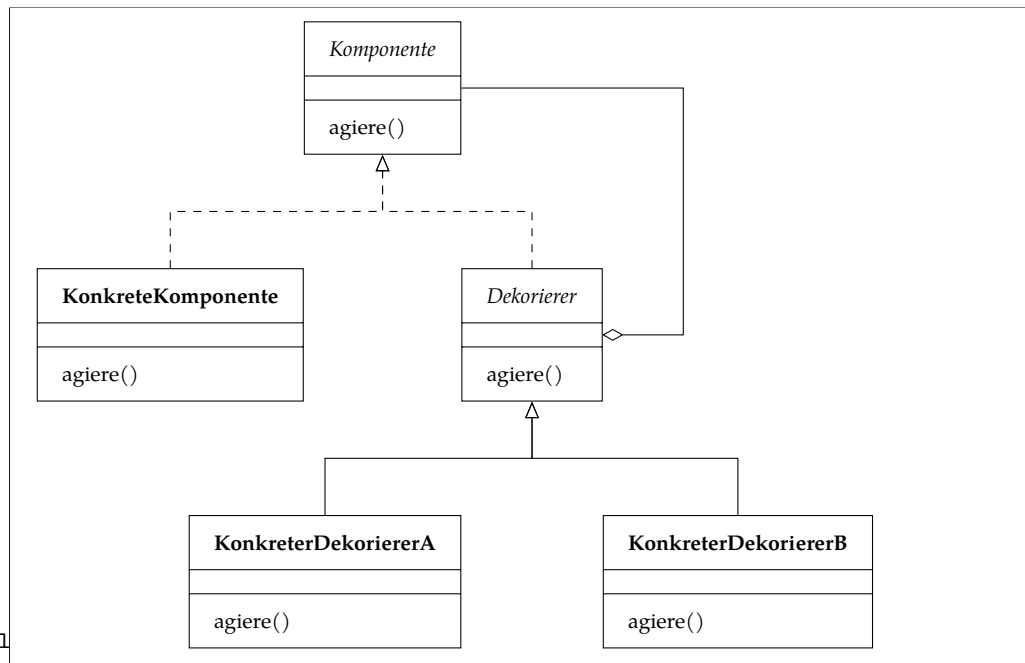
\bEntwurfsBeobachter

```

1373 \def\bEntwurfsBeobachter{
1374   \bEntwurfsBeobachterUml
1375   \bEntwurfsBeobachterAkteure
1376   \bEntwurfsBeobachterCode
1377 }

```

Dekorierer (Decorator)



\bEntwurfsDekoriererUml

```

1378 \def\bEntwurfsDekoriererUml{
1379   \begin{tikzpicture}
1380     \umlclass[type=abstract]{Komponente}{{}{agiere()}}
1381     \umlclass[below left=1.5cm and 0cm of Komponente]{KonkreteKomponente}{{}{agiere()}}

```

```

1382 \umlclass[below right=1.5cm and 0cm of Komponente,type=abstract]{Dekorierer}{-}{agiere()}
1383
1384 \umlVHVreal{KonkreteKomponente}{Komponente}
1385 \umlVHVreal{Dekorierer}{Komponente}
1386
1387 \umlclass[below left=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererA}{-}{agiere()}
1388 \umlclass[below right=1.5cm and 0cm of Dekorierer]{KonkreterDekoriererB}{-}{agiere()}
1389
1390 \umlVHVinherit{KonkreterDekoriererA}{Dekorierer}
1391 \umlVHVinherit{KonkreterDekoriererB}{Dekorierer}
1392
1393 \umlHVHaggreg[arm1=2cm]{Dekorierer}{Komponente}
1394 \footcite{wiki:dekorierer}
1395 \end{tikzpicture}
1396 }

```

\bEntwurfsDekoriererCode

```

1397 \def\bEntwurfsDekoriererCode{
1398 \li@EntwurfsCode{dekorierer}{Komponente}
1399 \li@EntwurfsCode{dekorierer}{KonkreteKomponente}
1400 \li@EntwurfsCode{dekorierer}{Dekorierer}
1401 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererA}
1402 \li@EntwurfsCode{dekorierer}{KonkreterDekoriererB}
1403 \li@EntwurfsCode{dekorierer}{Klient}
1404 }

```

\bEntwurfsDekorierer

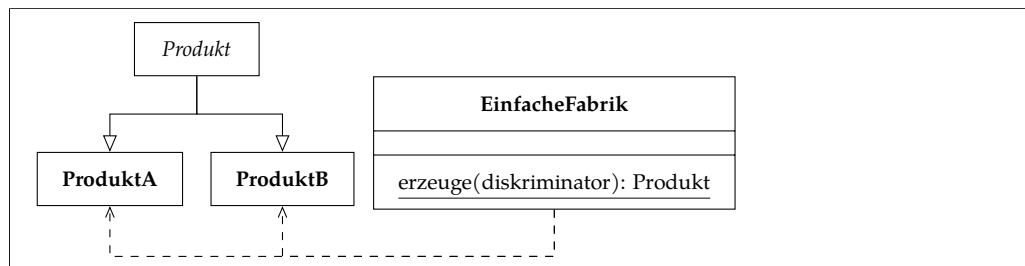
```

1405 \def\bEntwurfsDekorierer{
1406 \bEntwurfsDekoriererUml
1407 \bEntwurfsDekoriererAkteure
1408 \bEntwurfsDekoriererCode
1409 }

```

Einfache Fabrik (Simple Factory)

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



```

1410 \def\bEntwurfsEinfacheFabrikUml{
1411 \begin{tikzpicture}
1412 \umlsimpleclass[type=abstract]{Produkt}
1413 \umlsimpleclass[below left=1cm and -0.65cm of Produkt]{ProduktA}
1414 \umlsimpleclass[below right=1cm and -0.65cm of Produkt]{ProduktB}
1415 \umlVHVinherit{Produkt}{ProduktA}
1416 \umlVHVinherit{Produkt}{ProduktB}
1417 \umlclass[below right=0cm and 1.5cm of Produkt]{EinfacheFabrik}{
1418 }{
1419 \umlstatic{erzeuge(diskriminator): Produkt}\\
1420 }
1421 \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktA}
1422 \umlVHVdep[arm1=-1.5cm]{EinfacheFabrik}{ProduktB}
1423 \end{tikzpicture}
1424 }

```

\bEntwurfsEinfacheFabrikAkteure Quelle: frei überstetzt aus GoF

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

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

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

```

1425 \def\bEntwurfsEinfacheFabrikAkteure{
1426   \begin{description}
1427     \item[EinfacheFabrik]
1428
1429     Eine Klasse mit einer Erzeugungsmethode, die über eine größere
1430     Bedingung verschiedene Objekt instanziert.
1431
1432     \item[Produkt]
1433
1434     Eine abstrakte Klasse, die von den konkreten Produkten geerbt wird.
1435
1436     \item[KonkretesProdukt]
1437
1438     Ein konkretes Produkt, das von der einfachen Fabrik erzeugt wird.
1439   \end{description}
1440 }
```

\bEntwurfsEinfacheFabrik

```

1441 \def\bEntwurfsEinfacheFabrik{
1442   \bEntwurfsEinfacheFabrikUml
1443   \bEntwurfsEinfacheFabrikAkteure
1444 }
```

Einzelstück (Singleton)

urfsEinzelstueckBeschreibung

```

1445 \def\bEntwurfsEinzelstueckBeschreibung{
1446   Stellt sicher, dass nur \emph{genau eine Instanz einer Klasse} erzeugt
1447   wird.\footcite[Seite 38]{eilebrecht}
1448 }
1449
```

\bEntwurfsEinzelstueckUml Quelle nach der deutschen Wikipedia

Einzelstück
- instanz: Einzelstück
- Einzelstück() + gibInstanz(): Einzelstück

```

1450 \def\bEntwurfsEinzelstueckUml{
1451   \begin{tikzpicture}
1452     \umlclass{Einzelstück}{
1453       \umlstatic{- instanz: Einzelstück}\\
1454     }{
1455       - Einzelstück()\\
1456       + gibInstanz(): Einzelstück
1457     }
1458   \end{tikzpicture}
1459 }
```

bEntwurfsEinzelstueckAkteure 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.

```

1460 \def\bEntwurfsEinzelstueckAkteure{
1461   \begin{description}
1462     \item[Einzelstück (Singleton)]
1463
1464     stellt eine statische Methode bereit, mit deren Hilfe die Klienten
1465     nur auf eine einzige Instanz der Klasse zugreifen können.
1466   \end{description}
1467 }

```

\bEntwurfsEinzelstueckCode

```

1468 \def\bEntwurfsEinzelstueckCode{
1469   \li@EntwurfsCode{einzelstueck}{Einzelstueck}
1470 }

```

\bEntwurfsEinzelstueck

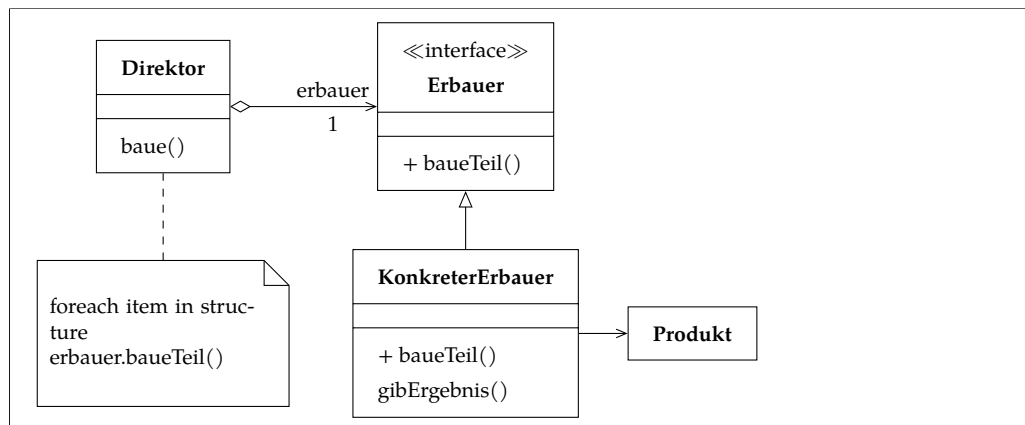
```

1471 \def\bEntwurfsEinzelstueck{
1472   \bEntwurfsEinzelstueckBeschreibung
1473
1474   \bEntwurfsEinzelstueckUml
1475
1476   \bEntwurfsEinzelstueckAkteure
1477
1478   \bEntwurfsEinzelstueckCode
1479 }

```

Erbauer (Builder)

\bEntwurfsErbauerUml Quelle nach der deutschen Wikipedia



```

1480 \def\bEntwurfsErbauerUml{
1481   \begin{tikzpicture}
1482     \umlsimpleclass[x=7,y=0]{Produkt}{}{}
1483     \umlclass[x=4,y=3,type=interface]{Erbauer}{}{+ baueTeil()}
1484     \umlclass[x=4,y=0]{KonkreterErbauer}{}{
1485       + baueTeil()\n
1486       gibErgebnis()}
1487     \umlclass[x=0,y=3]{Direktor}{}{baue()}
1488
1489     \umluniaggred[arg2=erbauer,pos2=0.7,mult2=1]{Direktor}{Erbauer}
1490     \umluniassoc{KonkreterErbauer}{Produkt}
1491     \umlinherit{KonkreterErbauer}{Erbauer}
1492
1493     \umlnote[x=0,y=0,width=3cm]{Direktor}{
1494       foreach item in structure\n
1495       erbauer.baueTeil()

```

```

1496 }
1497 \end{tikzpicture}
1498 \footcite{wiki:erbauer}
1499 }

```

\bEntwurfsErbauerAkteure 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.

```

1500 \def\bEntwurfsErbauerAkteure{
1501   \begin{description}
1502     \item[Erbauer]
1503
1504     Der Erbauer spezifiziert eine abstrakte Schnittstelle zur Erzeugung der
1505     Teile eines komplexen Objektes.
1506
1507     \item[KonkreterErbauer]
1508
1509     Der konkrete Erbauer erzeugt die Teile des komplexen Objekts durch
1510     Implementierung der Schnittstelle. Außerdem definiert und verwaltet er
1511     die von ihm erzeugte Repräsentation des Produkts. Er bietet auch eine
1512     Schnittstelle zum Auslesen des Produkts.
1513
1514     \item[Direktor]
1515
1516     Der Direktor konstruiert ein komplexes Objekt unter Verwendung der
1517     Schnittstelle des Erbauers. Der Direktor arbeitet eng mit dem Erbauer
1518     zusammen: Er weiß, welche Baureihenfolge der Erbauer verträgt oder
1519     benötigt. Der Direktor entkoppelt somit den Konstruktionsablauf vom
1520     Klienten.
1521
1522     \item[Produkt]
1523
1524     Das Produkt repräsentiert das zu konstruierende komplexe Objekt.
1525     \footcite{wiki:erbauer}
1526   \end{description}
1527 }

```

\bEntwurfsErbauer

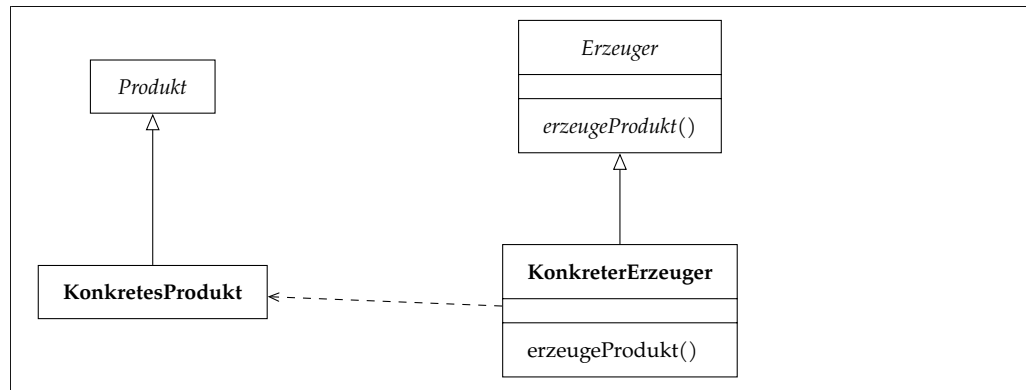
```

1528 \def\bEntwurfsErbauer{
1529   \bEntwurfsErbauerUml
1530   \bEntwurfsErbauerAkteure
1531 }

```

Fabrikmethode (Factory Method)

\bEntwurfsFabrikmethodeUml Quelle nach der deutschen Wikipedia



```

1532 \def\bEntwurfsFabrikmethodeUml{
1533   \begin{tikzpicture}
1534     \umlsimpleclass[type=abstract]{Produkt}
1535     \umlsimpleclass[below=2cm of Produkt]{KonkretesProdukt}
1536     \umlinherit{KonkretesProdukt}{Produkt}
1537
1538     \umlclass[type=abstract,right=4cm of Produkt]{Erzeuger}{\{
1539       \textit{erzeugeProdukt()}\}
1540   }
1541     \umlclass[below=1.2cm of Erzeuger]{KonkreterErzeuger}{\{
1542       erzeugeProdukt()
1543     \}
1544     \umlinherit{KonkreterErzeuger}{Erzeuger}
1545
1546     \umldep{KonkreterErzeuger}{KonkretesProdukt}
1547   \end{tikzpicture}
1548 }

```

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).

```

1549 \def\bEntwurfsFabrikmethodeAkteure{
1550   \begin{description}
1551     \item[Produkt]
1552
1553     Das Produkt ist der Basistyp (Klasse oder Schnittstelle) für das
1554     zu erzeugende Produkt.
1555
1556     \item[KonkretesProdukt]
1557
1558     KonkretesProdukt implementiert die Produkt-Schnittstelle.
1559
1560     \item[Erzeuger]
1561
1562     Der Erzeuger deklariert die Fabrikmethode, um ein solches Produkt
1563     zu erzeugen und kann eine Default-Implementierung beinhalten.
1564
1565     \item[KonkreterErzeuger]
1566

```

```

1567     KonkreterErzeuger überschreibt die Fabrikmethode, um die ihm
1568     entsprechenden konkreten Produkte zu erzeugen (z. B. indem er den
1569     Konstruktor einer konkreten Produkt-Klasse aufruft).
1570
1571     \footcite{wiki:fabrikmethode}
1572 \end{description}
1573 }

```

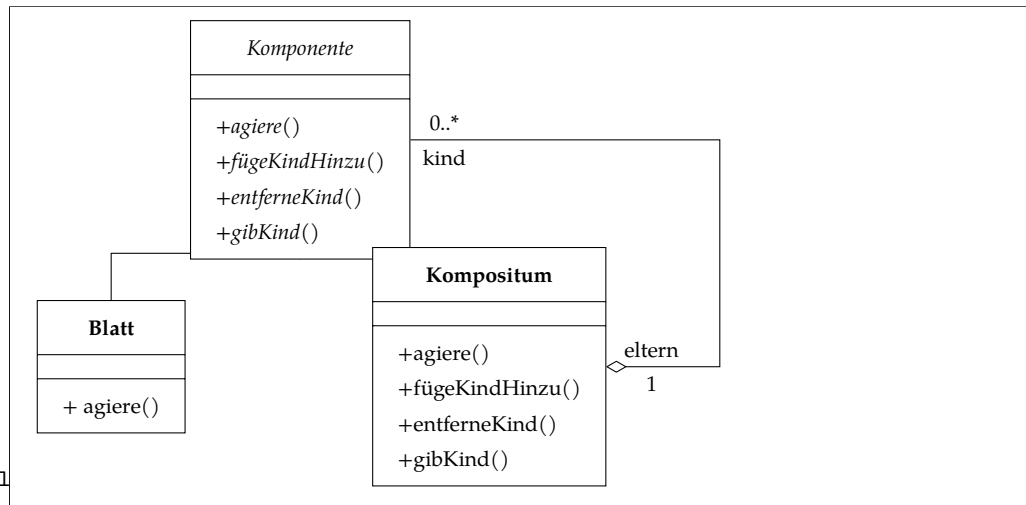
\bEntwurfsFabrikmethode

```

1574 \def\bEntwurfsFabrikmethode{
1575   \bEntwurfsFabrikmethodeUml
1576   \bEntwurfsFabrikmethodeAkteure
1577 }

```

Kompositum (Composite)



\bEntwurfsKompositumUml

```

1578 \def\bEntwurfsKompositumUml{
1579   \begin{tikzpicture}
1580     \umlclass[x=2.5,y=3,type=abstract]{Komponente}{}{
1581       \textit{+agiere()}\
1582       \textit{+fügeKindHinzu()}\
1583       \textit{+entferneKind()}\
1584       \textit{+gibKind()}
1585     }
1586     \umlclass[x=0]{Blatt}{}{+ agiere()}
1587     \umlclass[x=5]{Kompositum}{}{
1588       +agiere()\
1589       +fügeKindHinzu()\
1590       +entferneKind()\
1591       +gibKind()
1592     }
1593
1594     \umlVHVinherit{Kompositum}{Komponente}
1595     \umlVHVinherit{Blatt}{Komponente}
1596     \umlHVVHagg[anchor1=east,arm1=1.5cm,arg1=eltern,mult1=1,arg2=kind,mult2=0..*,pos2=2.9,p
1597   \end{tikzpicture}
1598 }

```

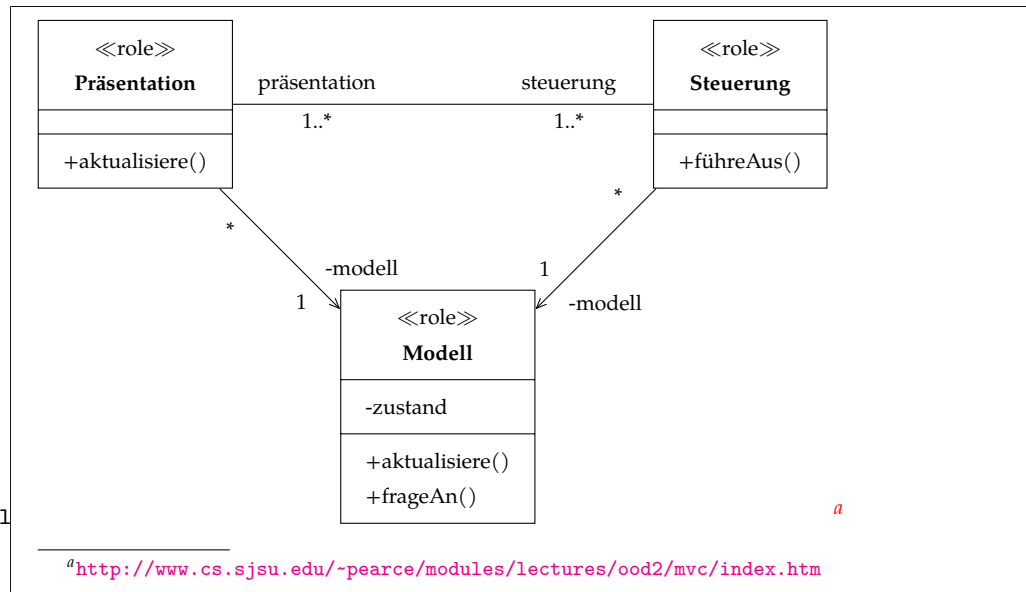
\bEntwurfsFabrikmethode

```

1599 \def\bEntwurfsKompositum{
1600   \bEntwurfsKompositumUml
1601   \bEntwurfsKompositumAkteure
1602 }

```

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



```

1603 \def\bEntwurfsModellPraesentationSteuerungUml{
1604   \begin{tikzpicture}
1605     \umlclass[x=-4cm,type=role]{Präsentation}{{+aktualisiere()}}
1606     \umlclass[x=4cm,type=role]{Steuerung}{{+führeAus()}}
1607     \umlclass[y=-4cm,type=role]{Modell}{
1608       -zustand
1609     }{
1610       +aktualisiere()\\
1611       +frageAn()
1612     }
1613
1614     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Präsentation}{Modell}
1615     \umluniassoc[arg2=-modell,mult2=1,mult1=*]{Steuerung}{Modell}
1616     \umlassoc[arg1=präsentation,mult1=1..*,arg2=steuerung,mult2=1..*]{Präsentation}{Steuerung}
1617   \end{tikzpicture}
1618   \bFussnoteUrl{http://www.cs.sjsu.edu/~pearce/modules/lectures/ood2/mvc/index.htm}
1619 }

```

ModellPraesentationSteuerung

```

1620 \def\bEntwurfs{
1621   \bEntwurfsModellPraesentationSteuerungUml
1622   \bEntwurfsModellPraesentationSteuerungAkteure
1623 }

```

Stellvertreter (Proxy)

\bEntwurfsStellvertreterUml

```

1624 \def\bEntwurfsStellvertreterUml{
1625   \begin{tikzpicture}
1626     \umlsimpleclass[x=-1,y=2]{Klient}
1627
1628     \umlclass[x=2,y=2]{Subjekt}{{+ agiere()}}
1629     \umlclass[x=0,y=-1]{KonkretesSubjekt}{{+ agiere()}}
1630     \umlclass[x=4,y=-1]{Stellvertreter}{{+ agiere()}}
1631
1632     \umlVHvinherit{KonkretesSubjekt}{Subjekt}
1633     \umlVHvinherit{Stellvertreter}{Subjekt}
1634     \umluniassoc{Stellvertreter}{KonkretesSubjekt}
1635     \umluniassoc{Klient}{Subjekt}
1636   \end{tikzpicture}
1637 }

```

\bEntwurfsStellvertreterCode

```

1638 \def\bEntwurfsStellvertreterCode{
1639   \li@EntwurfsCode{stellvertreter}{Subjekt}
1640   \li@EntwurfsCode{stellvertreter}{KonkretesSubjekt}
1641   \li@EntwurfsCode{stellvertreter}{Stellvertreter}
1642   \li@EntwurfsCode{stellvertreter}{Klient}
1643 }

```

\bEntwurfsStellvertreter

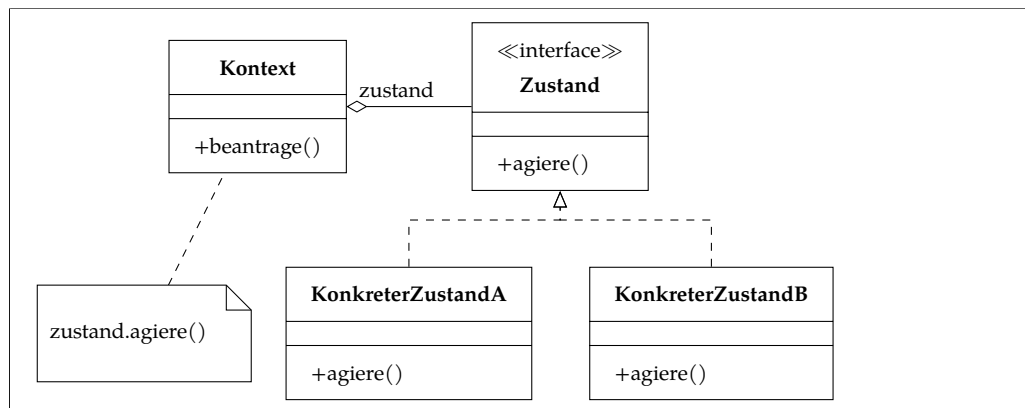
```

1644 \def\bEntwurfsStellvertreter{
1645   \bEntwurfsStellvertreterUml
1646   \bEntwurfsStellvertreterCode
1647 }

```

Zustand (State)

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



```

1648 \def\bEntwurfsZustandUml{
1649   \begin{tikzpicture}
1650     \umlclass[x=-1,y=3]{Kontext}{+beantrage()}
1651     \umlclass[x=3,y=3,type=interface]{Zustand}{+agiere()}
1652     \umlclass[x=1,y=0]{KonkreterZustandA}{+agiere()}
1653     \umlclass[x=5,y=0]{KonkreterZustandB}{+agiere()}
1654
1655     \umlVHVreal{KonkreterZustandA}{Zustand}
1656     \umlVHVreal{KonkreterZustandB}{Zustand}
1657
1658     \umlagg[arg=zustand,pos=0.4]{Kontext}{Zustand}
1659
1660     \umlnote[x=-2.5,y=0,width=2.5cm]{Kontext}{zustand.agiere()}
1661   \end{tikzpicture}
1662 }

```

\bEntwurfsZustandAkteure 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.

```

1663 \def\bEntwurfsZustandAkteure{
1664   \begin{description}
1665     \item[Kontext (Context)]
1666
1667     definiert die clientseitige Schnittstelle und verwaltet die separaten

```

```

1668 Zustandsklassen.
1669
1670 \item[State (Zustand)]
1671
1672 definiert eine einheitliche Schnittstelle aller Zustandsobjekte und
1673 implementiert gegebenenfalls ein Standardverhalten.
1674
1675 \item[KontreterZustand (ConcreteState)]
1676
1677 implementiert das Verhalten, das mit dem Zustand des Kontextobjektes
1678 verbunden ist.
1679 \end{description}
1680 }

```

\bEntwurfsZustandCode

```

1681 \def\bEntwurfsZustandCode{
1682   \li@EntwurfsCode{zustand}-{Kontext}
1683   \li@EntwurfsCode{zustand}-{Zustand}
1684 }

```

\bEntwurfsZustand

```

1685 \def\bEntwurfsZustand{
1686   \bEntwurfsZustandUml
1687   \bEntwurfsZustandAkteure
1688   \bEntwurfsZustandCode
1689 }

```

```

1690

```

er.sty

```
1691 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1692 \ProvidesPackage{bschlangaul-er}[2020/06/13 Zum Setzen von
1693 ER-Diagrammen]
1694 \RequirePackage{tikz-er2}
1695 \usetikzlibrary{positioning}
```

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}
```

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);
```

```
1696 \RequirePackage{soul}
```

```
1697 \RequirePackage{fontawesome}
```

Faulenzer

```
\let\a=\bErMpAttribute
\let\d=\bErDatenbankName
\let\e=\bErMpEntity
\let\r=\bErMpRelationship
```

```
1698 \ExplSyntaxOn
```

```
\bErEntity
```

```
1699 \def\bErEntity#1{\textbf{#1}}
```

```
\bErRelationship
```

```
1700 \def\bErRelationship#1{\ul{#1}}
```

```
\bErAttribute
```

```
1701 \def\bErAttribute#1{\emph{#1}}
```

```
\bErMpEntity
```

```
mp = marginpar
Makro-Faulenzer: \let\e=\liErMpEntity
```

```
1702 \def\bErMpEntity#1{
1703   \bErEntity{#1}
1704   \marginpar{
1705     \bErEntity{\tiny\faSquareO{}}~E:~#1}
1706   }
1707 }
```

```
□
```

```
\bErMpRelationship
```

```
Makro-Faulenzer: \let\r=\liErMpRelationship
```

```
1708 \def\bErMpRelationship#1{
1709   \bErRelationship{#1}
1710   \marginpar{
1711     \bErRelationship{\tiny\faGg{}}~R:~#1}
1712   }
1713 }
```

```
\bErMpAttribute
```

```
Makro-Faulenzer: \let\a=\liErMpAttribute
```

```
1714 \def\bErMpAttribute#1{
1715   \bErAttribute{#1}
1716   \marginpar{
1717     \bErAttribute{\tiny\faCircleThin{}}~A:~#1}
1718   }
1719 }
```

```
\bErDatenbankName Makro-Faulenzer: \let\d=\liErDatenbankName
                  datenbank name
1720 \def\bErDatenbankName#1{
1721   {
1722     \footnotesize\texttt{(#1)}
1723   }
1724 }

1725 \ExplSyntaxOff
1726
```

formale-sprachen.sty

```
1727 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1728 \ProvidesPackage{bschlangaul-formale-sprachen}[2021/02/21 Hilfsmakros
1729 zum Setzen von mathematischen Formeln bei dem Thema Formale Sprachen]
```

```
1730 \directlua{
1731   formale_sprachen = require('bschlangaul-formale-sprachen')
1732 }
```

```
1733 \RequirePackage{hyperref}
```

```
1734 \bLadePakete{mathe,typographie}
```

```
\bMenge   $\bMenge{a, b, c}$:  $\{a,b,c\}$ 
Makro-Faulenzer: \let\m=\liMenge
```

```
1735 \def\bMengeOhneMathe#1{\{ #1 \}}
1736 \def\bMenge#1{%
1737   \ifmmode%
1738     \bMengeOhneMathe{#1}%
1739   \else%
1740     $\bMengeOhneMathe{#1}$%
1741   \fi%
1742 }
```

```
\bEpsilon \bEpsilon:  $\varepsilon$ 
Makro-Faulenzer: \let\e=\liEpsilon
```

```
1743 \def\bEpsilon{$\varepsilon$}
```

```
\bPotenzmenge Umgeben mit geschweiften Klammern in einer Mathematik-Umgebung
```

```
1744 \def\erzeuge@tiefgestellt#1{\directlua{formale_sprachen.erzeuge_tiefgestellt('#1')}}
1745 \def\bPotenzmengeOhneMathe#1{\{ \erzeuge@tiefgestellt{#1} \}}
1746 \def\bPotenzmenge#1{$\bPotenzmengeOhneMathe{#1}$}
```

```
\bZustandsmenge \bZustandsmenge{z1, z2}:  $\{z_1, z_2\}$ 
```

```
1747 \let\bZustandsmengeOhneMathe=\bPotenzmengeOhneMathe
1748 \let\bZustandsmenge=\bPotenzmenge
```

```
\bUeberfuehrungsFunktion \bUeberfuehrungsFunktion{z0, a}:  $\delta(z_0, a)$ 
Makro-Faulenzer: \let\d=\liUeberfuehrungsFunktion
```

```
1749 \def\bUeberfuehrungsFunktionOhneMathe#1{\delta(\erzeuge@tiefgestellt{#1})}
1750 \def\bUeberfuehrungsFunktion#1{
1751   \ifmmode
1752     \bUeberfuehrungsFunktionOhneMathe{#1}
1753   \else
1754     $\bUeberfuehrungsFunktionOhneMathe{#1}$
1755   \fi
1756 }
```

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

```
1757 \def\bAlphabet#1{$\Sigma = \{ #1 \}$}
```

```
\bBandAlphabet \bBandAlphabet{\bTuringLeerzeichen}:  $\Gamma = \Sigma \cup \{\square\}$ 
```

```
1758 \def\bBandAlphabet#1{$\Gamma = \Sigma \cup \{ #1 \}$}
```

```
\bZustandsBuchstabe
```

```
1759 \def\bZustandsBuchstabe{z}
```

```
\bZustandsBuchstabeGross
```

```
1760 \def\bZustandsBuchstabeGross{Z}
```

```

\zustandsmengeNr
1761 \def\zustandsnamens@liste#1#2{
1762   $
1763   \{
1764     \directlua{formale_sprachen.erzeuge_zustandsnamens_liste('#1', '#2')}
1765   \}
1766   $
1767 }
1768 \def\bZustandsmengeNr#1{\zustandsnamens@liste{\bZustandsBuchstabe}{#1}}

\bZustandsmengeNrGross
1769 \def\bZustandsmengeNrGross#1{\zustandsnamens@liste{\bZustandsBuchstabeGross}{#1}}

\bZustandsname \bZustandsname{1}: $z_1$
1770 \def\bZustandsname#1{\$ \bZustandsBuchstabe_#1$}

\bZustandsnameGross \bZustandsnameGross{1}: $Z_1$
1771 \def\bZustandsnameGross#1{\$ \bZustandsBuchstabeGross_#1$}

\bAbleitung \bAbleitung{S -> aB -> ab}:  $S \vdash aB \vdash ab$ 
1772 \def\bAbleitung#1{\$ \directlua{formale_sprachen.formatiere_ableitung('#1')}$}

bProduktionsRegeln \begin{bProduktionsRegeln}[P_1]
    S -> S A B | EPSILON,
    B A -> A B,
    A A -> a a,
    B B -> b b
\end{bProduktionsRegeln}

1773 \NewDocumentEnvironment { bProduktionsRegeln }
1774 { 0{P} +b }
1775 {
1776   \bGeschweifteKlammern{#1}
1777   {
1778     \begin{align*}
1779       \directlua{formale_sprachen.produktions_regeln('#2')}
1780     \end{align*}
1781   }{-0.2cm}{-1.5cm}
1782 } {}

\bProduktionen \bProduktionen{S -> A, A -> a}:  $\{S \rightarrow A, A \rightarrow a\}$ 
1783 \def\bProduktionen#1{
1784   \bMenge{\directlua{formale_sprachen.produktions_regeln('#1', true)}}
1785 }

\bZustandsnameTiefgestellt Automatisch tiefgestellte Nummerierung \z1
Makro-Faulenzer: \let\z=\liZustandsnameTiefgestellt
1786 \def\bZustandsnameTiefgestellt#1{
1787   \ifmmode
1788     \bZustandsBuchstabe\sb{#1}
1789   \else
1790     \$\bZustandsBuchstabe\sb{#1}$
1791   \fi
1792 }

1793 \ExplSyntaxOn

\bAusdruck \bAusdruck[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 „=:“: \bAusdruck[] {x}{y}:  $\{x \mid y\}$ 
    Regulärer Ausdruck zum Konvertieren:
    \$(.*) += +\\{ *(.*) ( *\\, *)?\\| ( *\\, *)?(.*) *\\}\\$
    \\bAusdruck[$1]{$2}{$5}

```



```

1794 \NewDocumentCommand{ \bAusdruck } { O{L} m m } {
1795   $
1796   \tl_if_empty:nTF {#1} {} {#1 =}
1797   \{
1798     \, #2 \,
1799     |
1800     \, #3 \,
1801   \}$
1802 }
1803 \ExplSyntaxOff

\bFlaci Link zur flaci.com Website: \bFlaci{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

1804 \def\bFlaci#1{%
1805   \par
1806   {%
1807     \scriptsize
1808     Der Automat auf flaci.com (FLACI: Formale Sprachen, abstrakte
1809     Automaten, Compiler und Interpreter) Ein Projekt der Hochschule
1810     Zittau/Görlitz und der Pädagogischen Hochschule Schwyz:
1811     \href{https://flaci.com/#1}{flaci.com/#1}%
1812   }%
1813   \par
1814 }

\bGrammatik \bGrammatik[\langle grammatik-name \rangle]{\langle variablen=V,alphabet=\Sigma,produktionen=P,start=S \rangle}
    \bGrammatik{variablen={}, alphabet={}}

    - \bGrammatik{}:  $G = (V, \Sigma, P, S)$ 
    - \bGrammatik[G_1]{}:  $G_1 = (V, \Sigma, P, S)$ 
    - \bGrammatik{variablen={S,A,B}}:  $G = (\{S, A, B\}, \Sigma, P, S)$ 
    - \bGrammatik{alphabet={a,b}}:  $G = (V, \{a, b\}, P, S)$ 
    - \bGrammatik{start=X}:  $G = (V, \Sigma, P, X)$ 

1815 \ExplSyntaxOn
1816 \NewDocumentCommand {\bGrammatik} { O{G} m } {
1817   \tl_set:Nn \l_variablen_tl {V}
1818   \tl_set:Nn \l_alphabet_tl {\Sigma}
1819   \tl_set:Nn \l_produktionen_tl {P}
1820   \tl_set:Nn \l_start_tl {S}
1821
1822   \keys_define:nn { grammatik } {
1823     variablen .code:n = {\tl_set:Nn \l_variablen_tl {\bMenge{##1}}},
1824     alphabet .code:n = {\tl_set:Nn \l_alphabet_tl {\bMenge{##1}}},
1825     produktionen .code:n = {\tl_set:Nn \l_produktionen_tl {\bProduktionen{##1}}},
1826     start .code:n = {\tl_set:Nn \l_start_tl {##1}},
1827   }
1828
1829   \keys_set:nn { grammatik } { #2 }
1830
1831   $#1 = (
1832     \l_variablen_tl,
1833     \l_alphabet_tl,
1834     \l_produktionen_tl,
1835     \l_start_tl
1836   )$
1837 }
1838 \ExplSyntaxOff
1839

```

formatierung.sty

```
1840 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1841 \ProvidesPackage{bschlangaul-formatierung}[2020/11/27]
```

Schriftarten / Typographie

The package mathpazo Loading this package changes the default roman font family to Adobe Palatino, and the virtual ‘mathpazo’ fonts will be used for math. These virtual fonts are made up basically from Palatino Italic, with the missing math symbols coming from the CM and Pazo math fonts.

```
1842 \RequirePackage{mathpazo}
1843 \RequirePackage[no-math]{fontspec}
1844 \setmainfont{texgyrepagella}
```

Farben

```
1845 \RequirePackage{xcolor}
1846 \definecolor{infogray}{rgb}{0.97,0.97,0.97}
```

Überschriften

```
1847 \RequirePackage{titlesec}
1848 \titleformat{\chapter}[display]{\bfseries}{0pt}{\LARGE}
1849 \titlespacing{\chapter}{0pt}{0pt}{*1}
1850 \titleformat{\paragraph}[hang]{\normalsize\bfseries}{\theparagraph}{1em}{}
1851 \setcounter{secnumdepth}{0}
```

Listen

```
1852 \RequirePackage{paralist}
1853 \renewcommand\labelitemi{-}
1854 \renewcommand\labelitemii{-}
1855 \renewcommand\labelitemiii{-}
1856 \renewcommand\labelitemiv{-}
1857 % Counter: enumi enumii enumiii enumiv
1858 % Styles: \arabic{counter} \alph{counter} \Alph{counter} \roman{counter} \Roman{counter}
1859 \renewcommand{\labelenumi}{(\alph{enumi})}
1860 \renewcommand{\labelenumii}{(\roman{enumii})}
```

Kasten

```
1861 \RequirePackage{mdframed}
```

liKasten

```
1862 \NewDocumentEnvironment { liKasten }{ } {
1863   \begin{mdframed}[backgroundcolor=white!0]
1864 } {
1865   \end{mdframed}
1866 }
```

Header

```
1867 \RequirePackage{fancyhdr}
1868 \fancyhead[L,C,R]{}
1869 \fancyfoot[L]{}
1870 \fancyfoot[C]{}
1871 \fancyfoot[R]{\thepage}
1872 \pagestyle{fancy}
1873 \renewcommand{\headrulewidth}{0pt}
1874 \renewcommand{\footrulewidth}{0pt}
```

Zeilenabstände

Werden kleinere Schriften verwendet, passt sich der Zeilenabstand nicht entsprechend an. Mit der Umgebung spacing funktioniert es dann.

```
1875 \RequirePackage{setspace}
```


gantt.sty

```

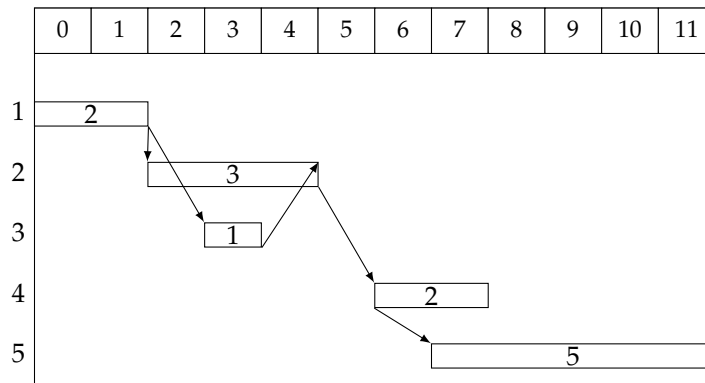
1877 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1878 \ProvidesPackage{bschlangaul-gantt}[2020/09/05]

\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}

```



```

1879 \RequirePackage{tikz-uml}
1880 \RequirePackage{pgfgantt}
1881 \setganttlinklabel{f-s}{}
1882 \setganttlinklabel{s-s}{}
1883 \setganttlinklabel{f-f}{}
1884 \setganttlinklabel{s-f}{}
1885

```

grafik.sty

```
1886 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1887 \ProvidesPackage{bschlangaul-grafik}[2020/12/27 Lädt das Paket
1888 TikZ, um Grafiken zeichnen zu können und graphicx um Bilder laden zu können.]

1889 \ExplSyntaxOn

1890 \RequirePackage{tikz}

1891 \RequirePackage{graphicx}

\bGrafikLogoPfad

1892 \def\bGrafikLogoPfad#1{
1893   \LehramtInformatikRepository / .tex / Logo / #1
1894 }

\bGrafikCCLizenz

1895 \NewDocumentCommand{ \bGrafikCCLizenz } { 0{} } {
1896   \includegraphics[#1]{
1897     \bGrafikLogoPfad{CC-by-nc-sa.eps}
1898   }
1899 }

\bGrafikLogo

1900 \NewDocumentCommand{ \bGrafikLogo } { 0{} } {
1901   \includegraphics[#1]{
1902     \bGrafikLogoPfad{Logo_nur-Pfade.eps}
1903   }
1904 }

1905 \ExplSyntaxOff

1906
```

graph.sty

```
1907 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1908 \ProvidesPackage{bschlangaul-graph}[2020/06/09]
1909 \RequirePackage{tikz}
```

Für die die Adjazenzliste (xrightarrow)

```
1910 \RequirePackage{amsmath}
```

Für Adjazenz-Matrix

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

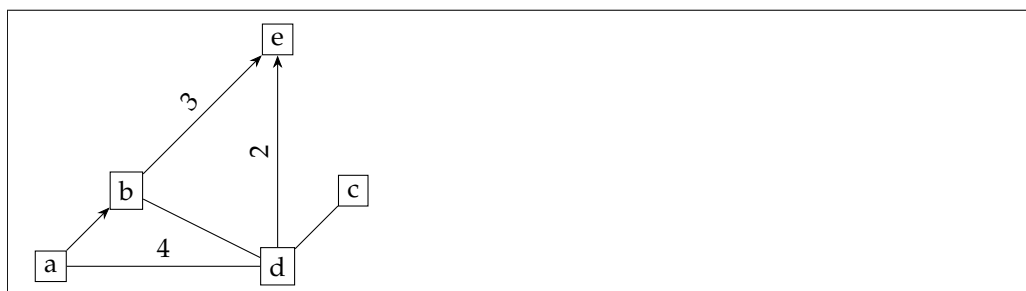
$$\begin{array}{c}
 a \quad b \quad c \quad d \quad e \\
 \begin{array}{c}
 a \\ b \\ c \\ d \\ e
 \end{array}
 \begin{pmatrix}
 0 & 1 & 0 & 4 & 0 \\
 0 & 0 & 0 & 1 & 3 \\
 0 & 0 & 0 & 1 & 0 \\
 4 & 1 & 1 & 0 & 2 \\
 0 & 0 & 0 & 0 & 0
 \end{pmatrix}
 \end{array}$$

```
1911 \RequirePackage{blkarray}
```

```
1912 \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}
```



```

1913 \tikzset{
1914   li graph/.style={
1915     every node/.style={
1916       rectangle,
1917       draw,
1918     },
1919     every edge/.style={
1920       >={Stealth[black]},
1921       draw,
1922     },
1923     every edge/.append style={
1924       every node/.style={
1925         sloped,
1926         auto,
1927       }
1928     }
1929   },
1930   li markierung/.style={
1931     ultra thick,
1932   }
1933 }

```

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}

```

```

1934 \NewDocumentEnvironment { liGraphenFormat }{ +b } {} {}

```

```

1935

```

hanoi.sty

```
1936 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1937 \ProvidesPackage{bschlangaul-hanoi}[2020/12/19 Paket zum Setzen
1938 von Hanoi-Grafiken]

    Quelle: https://kleinco.de/latex-snippets/7/tikz-towers-of-hanoi-illustration-for-lat
1939 \RequirePackage{tikz}
1940 \RequirePackage{xcolor}

\bHanoi \bHanoi{anzahl-scheiben}{gewicht-scheibe/turm-nr,gewicht-scheibe/turm-nr},
z. B.: \bHanoi{4}{4/1,3/1,2/3,1/2}

1941 \def\li@mset #1[#2]=#3{%
1942 \expandafter\xdef\csname #1#2\endcsname{#3}
1943 }
1944 \def\li@mget #1[#2]{%
1945 \csname #1#2\endcsname
1946 }
1947 \def\li@minc #1[#2]+=#3{%
1948 \pgfmathparse{\li@mget #1[#2]+#3}%
1949 \li@mset #1[#2]=\pgfmathresult
1950 }
1951
1952 \def\bHanoi#1#2{
1953   \edef\li@numdiscs{#1}
1954   \def\li@sequence{#2}
1955   \begin{tikzpicture}[line width=4mm,brown!40,line cap=round,xscale=3]
1956     % init colors
1957     \foreach[count=\j] \c in {red,green!80!black,blue,orange,violet,gray,yellow!80!black,purp
1958     \li@mset col[\j]={\c};
1959     % draw poles and init pole counters
1960     \foreach \j in {1,2,3}{
1961       \li@mset pos[\j]=0
1962       \draw (\j,-.5) -- +(0, .5 + .5*\li@numdiscs);
1963     }
1964     % draw base
1965     \draw (.5,-.5) -- +(3,0);
1966     % draw discs
1967     \foreach[count=\k] \i/\j in \li@sequence{
1968       \draw[draw={\li@mget col[\i]}] (\j,\li@mget pos[\j]) +(-.4*\i/\li@numdiscs,0) -- +(.4*
1969       \li@minc pos[\j]+=.5}
1970     }
1971   \end{tikzpicture}
1972 }

1973
```


index.sty

1974 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

1975 \ProvidesPackage{bschlangaul-index}[2021/09/12 Den Index anzeigen
1976 lassen]

1977 \ExplSyntaxOn

1978 \RequirePackage{makeidx}

1979 \makeindex

1980 \ExplSyntaxOff

1981

komplexitaetstheorie.sty

```
1982 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
1983 \ProvidesPackage{bschlangaul-komplexitaetstheorie}[2021/07/08 Zum
1984 Setzen von Karp's NP-vollständigen Problemen, Hilfsmakros für die
1985 Polynomialzeitreduktion.]
```

Faulenzer

```
\let\n=\bProblemName
\let\r=\bPolynomiellReduzierbar
\let\b=\bProblemBeschreibung
```

```
1986 \bLadePakete{mathe}
```

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

```
1987 \RequirePackage{mdframed}
```

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

```
1988 \def\bStrich#1{#1^\prime}
```

`\bProblemName` Zu Setzen von Problemnamen wie zum Beispiel SAT, COL, VERTEX COVER

Makro-Faulenzer: `\let\n=\liProblemName`

`\bProblemName: SAT VERTEX COVER`

```
1989 \def\bProblemName#1{\texttt{\textsc{#1}}}
```

`\bProblemBeschreibung` Zu setzen von Problem-Beschreibungen:

```
\bProblemBeschreibung
```

```
{}
```

```
{}
```

```
{}
```

<p style="text-align: center;">CLIQUE</p> <p>Gegeben: Ein ungerichteter Graph $G = (V, E)$, eine Zahl $k \in \mathcal{N}$</p> <p>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?</p>
--

Makro-Faulenzer: `\let\b=\liProblemBeschreibung`

```
1990 \def\bProblemBeschreibung#1#2#3{
1991   \begin{mdframed}[
1992     userdefinedwidth=9cm,
1993     align=center,
1994     backgroundcolor=white!0,
1995   ]
1996   \centerline{\large\bProblemName{#1}}
1997
1998   \medskip
1999
2000   \begin{description}
2001     \item[Gegeben:] #2
2002     \item[Frage:] #3
2003   \end{description}
2004 \end{mdframed}
2005 }
```

```

\bPolynomiellReduzierbar Makro-Faulenzer: \let\r=\liPolynomiellReduzierbar

2006 \NewDocumentCommand{ \bPolynomiellReduzierbar } { m O{p} m } {
2007 \begin{displaymath}
2008   \bProblemName{#1}
2009   \preceq_{#2}
2010   \bProblemName{#3}
2011 \end{displaymath}
2012 }

\bProblemVertexCover

2013 \def\bProblemClique{%
2014 Das \textbf{Cliquesproblem} fragt nach der Existenz einer Clique der
2015 Mindestgröße  $n$  in einem gegebenen Graphen.
2016 \footcite[Seite 76]{theo:fs:4}
2017 Eine Clique ist eine Teilmenge von Knoten in einem ungerichteten
2018 Graphen, bei der \emph{jedes Knotenpaar durch eine Kante} verbunden ist.
2019 \footcite{wiki:cliquesproblem}
2020 }

\bProblemVertexCover

2021 \def\bProblemVertexCover{%
2022 %
2023 Das \textbf{Knotenüberdeckungsproblem} (\bProblemName{Vertex Cover})
2024 fragt, ob zu einem gegebenen einfachen Graphen und einer natürlichen
2025 Zahl  $k$  eine Knotenüberdeckung der Größe von höchstens  $k$  existiert.
2026 \footcite{wiki:knotenueberdeckung}
2027
2028 Das heißt, ob es eine aus maximal  $k$  Knoten bestehende Teilmenge  $U$ 
2029 der Knotenmenge gibt, so dass jede Kante des Graphen mit
2030 mindestens einem Knoten aus  $U$  verbunden ist.
2031 \footcite[Seite 78]{theo:fs:4}%
2032 }

\bProblemSubsetSum Kein Karp-21-Problem

2033 \def\bProblemSubsetSum{%
2034 Das \textbf{Teilsommenproblem} (\bProblemName{Subset Sum} oder
2035 \bProblemName{SSP}) ist ein spezielles Rucksackproblem.
2036 \footcite{wiki:teilsommenproblem}
2037 Gegeben sei eine Menge von ganzen Zahlen  $I = \{w_1, w_2, \dots, w_n\}$ .
2038 Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht
2039 größer als eine gegebene obere Schranke  $c$  ist.
2040 \footcite[Seite 74]{theo:fs:4}
2041 }

\bProblemSubsetSum Kein Karp-21-Problem

2042 \def\bProblemSat{%
2043 Das \textbf{Erfüllbarkeitsproblem der Aussagenlogik} \bProblemName{Sat}
2044 und \bProblemName{k-SAT} mit  $k \geq 3$ ,  $k \in \mathbb{N}$  (Satz von
2045 Cook) fragt, ob eine aussagenlogische Formel erfüllbar
2046 ist. \footcite{wiki:sat} Das Erfüllbarkeitsproblem der
2047 \emph{Aussagenlogik} ist in exponentieller Zeit in Abhängigkeit der
2048 Anzahl der Variablen mit Hilfe einer Wahrheitstabelle entscheidbar.
2049 Diese \emph{Wahrheitstabelle} kann nicht in polynomieller Zeit
2050 aufgestellt werden.
2051 \footcite[Seite 71]{theo:fs:4}
2052 }

2053

```

kontrollflussgraph.sty

2054 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2055 \ProvidesPackage{bschlangaul-kontrollflussgraph}[2020/11/07]

Faulenzer

```
\let\b=\bBedingung
\let\c=\bKontrollCode
\let\f=\bBedingungFalsch
\let\k=\bKontrollTextzeileKnoten
\let\p=\bKontrollKnotenPfad
\let\w=\bBedingungWahr
```

TeX-Markup-Beispiel

```
\begin{bKontrollflussgraph}[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{bKontrollflussgraph}
```

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 = ...}]
```

```

2056 \RequirePackage{tikz}
2057 \usetikzlibrary{positioning}
2058 \tikzset{
2059   li kontrollfluss/.style={
2060     knoten/.style={
2061       circle,
2062       draw
2063     },
2064     usebox/.style={
2065       draw,
2066       rectangle,
2067       font=\scriptsize,
2068       anchor=west,
2069       align=left,
2070     },
2071     bedingung/.style={
2072       midway,
2073       draw=none,
2074       font=\scriptsize
2075     },
2076     knotenbeschriftung/.style={
2077       draw,
2078       rectangle,
2079       midway,
2080       font=\scriptsize
2081     },
2082     wahr/.style={
2083       thick
2084     },
2085     falsch/.style={
2086       dashed
2087     },
2088     every node/.style={
2089       circle,
2090       draw,
2091     },
2092     every edge/.append style={
2093       every node/.style={
2094         draw=none,
2095         bedingung,
2096       }
2097     },
2098     every path/.style={
2099       draw,
2100       ->,
2101     },
2102     every pin/.style={
2103       draw,
2104       dotted,
2105       rectangle,
2106       pin position=right
2107     },
2108     every pin edge/.style={
2109       dotted,
2110       arrows=-,
2111     }
2112   }
2113 }

```

Umgebungen

bKontrollflussgraph

```

2114 \NewDocumentEnvironment { bKontrollflussgraph } { 0{ } } {

```

```

2115 \begin{tikzpicture}[
2116   li kontrollfluss,
2117   #1
2118 ]
2119 } {
2120 \end{tikzpicture}
2121 }

```

Makros

\bAnweisung

```
2122 \def\bAnweisung#1(#2,#3){\node[knoten] at (#2,#3) (#1) {#1};}
```

\bBedingung **Makro-Faulenzer:** \let\b=\liBedingung

```
2123 \def\bBedingung#1#2{node[bedingung,#1]{\texttt{#2}}}
```

\bBedingungWahr **Makro-Faulenzer:** \let\w=\liBedingungWahr

```
2124 \def\bBedingungWahr#1{node[bedingung,#1]{\texttt{[true]}}}
```

\bBedingungFalsch **Makro-Faulenzer:** \let\f=\liBedingungFalsch

```
2125 \def\bBedingungFalsch#1{node[bedingung,#1]{\texttt{[false]}}}
```

\bKontrollCode **Makro-Faulenzer:** \let\c=\liKontrollCode

```
2126 \def\bKontrollCode#1{{\tiny\texttt{\textcolor{gray}{#1}}}}
```

\bKontrollTextzeileKnoten **Makro-Faulenzer:** \let\k=\liKontrollTextzeileKnoten

```
2127 \def\bKontrollTextzeileKnoten#1{\raisebox{-2pt}{\tikz[scale=0.5,transform shape] \node[draw,c
```

\bKontrollKnotenPfad **Makro-Faulenzer:** \let\p=\liKontrollKnotenPfad

```
2128 \ExplSyntaxOn
```

```
2129 \NewDocumentCommand { \bKontrollKnotenPfad } { m }
```

```
2130 {
```

```
2131   \seq_set_split:Nnn \l_tmpa_seq { - } { #1 }
```

```
2132   \seq_set_map:Nnn \l_tmpa_seq \l_tmpa_seq {\bKontrollTextzeileKnoten{##1}}
```

```
2133   \seq_use:Nn \l_tmpa_seq { ~~~~ }
```

```
2134 }
```

```
2135 \ExplSyntaxOff
```

```
2136
```

kopfzeile.sty

```
2137 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2138 \ProvidesPackage{bschlangaul-kopfzeile}[2021/08/20 Kopf-
2139 und Fußzeilen mit Hilfe des Pakets fancyhdr.]

2140 \ExplSyntaxOn

2141 \RequirePackage{bschlangaul-meta}

2142 \cs_new:Npn \kopfzeile_schrift:n #1
2143 {
2144   {
2145     \scriptsize
2146     #1
2147   }
2148 }

2149 \cs_new:Npn \setze_kopf_fusszeilen:nn #1 #2
2150 {
2151   \fancyhead{}
2152   \fancyhead[L]{\kopfzeile_schrift:n{\bMetaBschlangaulSammlung}}
2153   \fancyhead[C]{\kopfzeile_schrift:n{#1}}
2154   \fancyhead[R]{\kopfzeile_schrift:n{#2}}
2155
2156   \fancyfoot{}
2157   \fancyfoot[L]{\kopfzeile_schrift:n{\bMetaHermineFriends}}
2158   \fancyfoot[C]{\kopfzeile_schrift:n{\bMetaEmail}}
2159   \fancyfoot[R]{\kopfzeile_schrift:n{\thepage}}
2160
2161   \renewcommand{\headrulewidth}{0.4pt}
2162   \renewcommand{\footrulewidth}{0.4pt}
2163 }

2164 \cs_new:Npn \setze_kopfzeile_oben_rechts:n #1
2165 {
2166   \fancyhead[R] {
2167     \kopfzeile_schrift:n { #1 }
2168   }
2169 }

    Das Makro darf nicht in der Präambel aufgerufen werden, da es die Textbreite
braucht.

2170 \cs_new:Npn \repariere_kopfzeile_breite:
2171 {
2172   \setlength{\headwidth}{\textwidth}
2173 }

2174 \ExplSyntaxOff

2175
```

literatur-dummy.sty

2176 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2177 \ProvidesPackage{bschlangaul-literatur-dummy}[2020/11/27]

\literatur

2178 \def\literatur{}

\footcite

2179 % \RequirePackage[stable,multiple]{footmisc}

2180 \NewDocumentCommand{ \footcite } { o m }{}

2181

literatur.sty

```
2182 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2183 \ProvidesPackage{bschlangaul-literatur}[2020/11/27]

2184 \RequirePackage{csquotes}
2185 \RequirePackage[
2186   bibencoding=utf8,
2187   citestyle=authortitle,
2188   backend=biber,
2189 ]{biblatex}
2190 \addbibresource{\LehramtInformatikRepository/Quellen/10_DB.bib}
2191 \addbibresource{\LehramtInformatikRepository/Quellen/20_OOMUP.bib}
2192 \addbibresource{\LehramtInformatikRepository/Quellen/30_AUD.bib}
2193 \addbibresource{\LehramtInformatikRepository/Quellen/40_SOSY.bib}
2194 \addbibresource{\LehramtInformatikRepository/Quellen/60_FUMUP.bib}
2195 \addbibresource{\LehramtInformatikRepository/Quellen/70_THEO.bib}
2196 \addbibresource{\LehramtInformatikRepository/Quellen/80_DDI.bib}
2197 \addbibresource{\LehramtInformatikRepository/Quellen/Allgemein.bib}
2198 \addbibresource{\LehramtInformatikRepository/Quellen/Examen.bib}
2199 \addbibresource{\LehramtInformatikRepository/Quellen/Wikipedia.bib}
2200 % To allow footnotes in the heading
2201 \RequirePackage[stable,multiple]{footmisc}

\literatur

2202 \def\literatur{\printbibliography[heading=subbibliography]}

2203
```

makros.sty

```
2204 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2205 \ProvidesPackage{bschlangaul-makros}[2020/11/27 Eine Sammlung von
2206 häufig verwendeten Makros und Umgebungen, die thematisch zu keinem
2207 anderen Paket passen]
```

```
2208 \RequirePackage{hyperref}
```

```
2209 \RequirePackage{graphicx}
```

Für die Umgebung bQuellen benötigt.

```
2210 \RequirePackage{paralist}
```

```
2211 \ExplSyntaxOn
```

`\inhaltsverzeichnis`

```
2212 \def\inhaltsverzeichnis {
2213   \begin{mdframed}
2214     \begin{group}
2215       \let\clearpage\relax
2216       \tableofcontents
2217     \end{group}
2218   \end{mdframed}
2219 }
```

`\bEmph` `\bEmph` (`\marginpar` and `\emph`)

```
2220 \def\bEmph#1
2221 {
2222   \emph{#1}
2223   \marginpar{
2224     \tiny#1
2225   }
2226 }
```

`\SLASH`

```
2227 \newcommand\SLASH{\char`\\}
```

`\bPseudoUeberschrift` Text, der sich wie eine Überschrift verhält.

```
2228 \newcommand{\bPseudoUeberschrift}[1]{
2229   \bigskip
2230
2231   \par
2232   \noindent
2233   \textbf{#1}
2234
2235   \medskip
2236
2237   \keine_einrueckung:
2238 }
```

`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.

```
2239 \NewDocumentEnvironment { liProjektSprache }{ o +b } {} {}
```

`liEinbettung`

```
2240 \NewDocumentEnvironment { liEinbettung }{ o +b } {#2} {}
```

Umgebungen, die Inhalte aus- und einblenden können

Die einfachste Möglichkeit, um den kompletten Inhalt einer Umgebung auszublenden ist die Verwendung `+b` in einer `xparse` erzeugten Umgebung. Manchmal funktioniert

diese Methode nicht. In der Dokumentation von xparse steht, dass dieses Feature etwas experimentell ist.

Eine andere Methode verwendet das exam-Package. Die Inhalt wird in eine Box verschoben, die dann einfach ignoriert wird.

```
\NewDocumentEnvironment{ bAdditum } { o }
{
  \ifADDITUM
  \else
    % Alles in eine Box verschieben und die dann ignorieren.
    \setbox 0 \vbox
    \bgroup
    \fi

    \begin{frame}
  } {
    \end{frame}

    \ifADDITUM
    \else
      \egroup
    \fi
}
```

bAntwort Zum Setzen von Antworten. Sie werden mit einem Kasten umgeben. Könnten eventuell ausgeblendet werden, wenn man nur die Lösung sehen will.

```
2241 \NewDocumentEnvironment{ bAntwort } { 0{standard} }
2242 {
2243   \ifANTWORT
2244   \else
2245     \setbox 0 \vbox
2246     \bgroup
2247     \fi
2248
2249     \str_case:nn {#1} {
2250       {standard} {
2251         \def\beschriftung{}
2252         \mdfsetup{backgroundcolor=gray!1}
2253       }
2254       {richtig} {
2255         \def\beschriftung{richtig}
2256         \mdfsetup{backgroundcolor=green!5,linecolor=green}
2257       }
2258       {falsch} {
2259         \def\beschriftung{falsch}
2260         \mdfsetup{backgroundcolor=red!3,linecolor=red}
2261       }
2262       {muster} {
2263         \def\beschriftung{Musterlösung}
2264         \mdfsetup{backgroundcolor=green!12,linecolor=green}
2265       }
2266     }
2267     \ifx\beschriftung\empty\else
2268       \noindent
2269       \textbf{\beschriftung{:}}
2270     \fi
2271     \begin{mdframed}[
2272       frametitle={\scriptsize\mdseries Lösungsvorschlag},
2273       innertopmargin=6pt,
2274       frametitleaboveskip=-10pt,
2275       frametitlealignment=\raggedleft
2276     ]
```

```

2277 }
2278 {
2279   \end{mdframed}
2280   \ifANTWORT
2281   \else
2282     \egroup
2283   \fi
2284 }

```

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

```

2285 \NewDocumentEnvironment{ bAdditum } { o }
2286 {
2287   \ifADDITUM
2288   \else
2289     \setbox 0 \vbox
2290     \bgroup
2291   \fi
2292
2293   \begin{mdframed}[backgroundcolor=yellow!5]
2294     \IfNoValueTF {#1}
2295     {
2296       \bPseudoUeberschrift{Additum}
2297     }
2298     {
2299       \bPseudoUeberschrift{Additum:~#1}
2300     }
2301   }
2302 {
2303   \end{mdframed}
2304
2305   \ifADDITUM
2306   \else
2307     \egroup
2308   \fi
2309 }

```

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

Exkurs: Linear rekursiv

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

```

2310 \NewDocumentEnvironment{ bExkurs }{ o +b }
2311 {
2312   \ifEXKURS
2313     \vspace{0.2cm}%
2314     \begin{mdframed}[
2315       backgroundcolor=white,
2316       bottomline=false,
2317       innermargin=1cm,
2318       leftline=true,
2319       linecolor=black,
2320       linewidth=0.1cm,
2321       outermargin=1cm,
2322       rightline=false,
2323       topline=false,
2324     ]

```

```

2325      \footnotesize
2326      \noindent%
2327      \textbf{Exkurs:~#1}\par%
2328      \noindent%
2329      #2
2330      \end{mdframed}
2331      \vspace{0.2cm}
2332 \else
2333 \fi
2334 }
2335 {}

```

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

```

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

```

Weiterführende Literatur:

- Quelle 1
- Quelle 2

```

2336 \cs_new:Npn \listen_punkt:n #1
2337 {
2338   \item #1
2339 }
2340 \NewDocumentEnvironment { bQuellen }{+b }
2341 {
2342   \seq_clear_new:N \l_quellen
2343   \seq_set_split:Nnn \l_quellen {\item} {#1}
2344   \seq_remove_all:Nn \l_quellen {}
2345   \begin{mdframed}[backgroundcolor=white!0]
2346     \footnotesize
2347     \noindent
2348     \textsf{\textbf{Weiterführende-Literatur:}}
2349     \medskip
2350     \begin{compactitem}
2351       \seq_map_function:NN \l_quellen {\listen_punkt:n}
2352     \end{compactitem}
2353   \end{mdframed}
2354
2355   \keine_einrueckung:
2356 } {}

```

`\bFussnoteUrl` `\bFussnoteUrl[<zusätzlicher-text>]{<url>}` `\bFussnoteUrl[zusätzlicher Text]{url}`:
Eine HTTP-URL als Fußnote setzen.

```

2357 \NewDocumentCommand { \bFussnoteUrl } { o m }
2358 {
2359   \footnote{
2360     \url{#2}
2361     \IfNoValueTF{#1}
2362     {}
2363     {
2364       ~(#1)
2365     }
2366   }
2367 }
2368

```

`\bFussnoteLink` `\bFussnoteLink[<zusätzlicher-text>]{<link-text>}{<url>}` `\bFussnoteLink[zusätzlicher Text]{text}`:
Einen Link, d. h. einen Link-Text und eine URL als Fußnote setzen.

```

2369 \NewDocumentCommand{ \bFussnoteLink } { o m m }
2370 {
2371   \footnote{
2372     \href{#3}{#2}
2373     \IfNoValueTF{#1}
2374     {}
2375     {
2376       ~(#1)
2377     }
2378   }
2379 }

\zB
2380 \def\zB{z.\,B. }

\ZB
2381 \def\ZB{Z.\,B. }

\dh
2382 \def\dh{d.\,h. }

2383 \ExplSyntaxOff
2384

```

master-theorem.sty

2385 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2386 \ProvidesPackage{bschlangaul-master-theorem}[2021/04/13]

Faulenzer

\let\O=\bO

\let\o=\bOomega

\let\T=\bT

\let\t=\bTheta

\bMasterVariablenDeklaration

{3} % a

{3} % b

{\mathcal{O}(1)} % f(n)

\bMasterFallRechnung

% 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

$f(n) = 5n^2 \notin \mathcal{O}(n^{\log_2 8 + \varepsilon})$

\bMasterWolframLink{T[n]=9T[n/3]\%2B5n^2}

\bMasterVariablenDeklaration

{ } % a

{ } % b

{ } % f(n) ohne \$mathe\$

\bMasterFallRechnung

% 1. Fall

{ }

% 2. Fall

{ }

% 3. Fall

{ }

$\rightarrow T(n) \in \mathcal{T}(n^2 \cdot \log n)$

\bMasterWolframLink{T[n]=9T[n/3]\%2B5n^2}

2387 \ExplSyntaxOn

2388 \RequirePackage{amsmath}

\bRundeKlammer

2389 \def\bRundeKlammer#1{

2390 \negthinspace \left(#1 \right)

2391 }

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

2392 \def\bThetaOhneMathe#1{

2393 \Theta \bRundeKlammer{#1}

2394 }

2395 \def\bTheta#1{

2396 \ifmmode

2397 \bThetaOhneMathe{#1}

2398 \else

2399 $\bThetaOhneMathe{#1}$

2400 \fi

2401 }

```

\Omega \Omega{n^2}: \Omega(n^2)
2402 \def\OmegaOhneMathe#1{
2403   \Omega \RundeKlammer{#1}
2404 }
2405 \def\Omega#1{
2406   \ifmmode
2407     \OmegaOhneMathe{#1}
2408   \else
2409     $\OmegaOhneMathe{#1}$
2410   \fi
2411 }

\O \O{n^2}: \mathcal{O}(n^2)
2412 \def\OOhneMathe#1{
2413   \mathcal{O} \RundeKlammer{#1}
2414 }
2415 \def\O#1{
2416   \ifmmode
2417     \OOhneMathe{#1}
2418   \else
2419     $\OOhneMathe{#1}$
2420   \fi
2421 }

\T Makro-Faulenzer: \let\T=\liT
      \T{16}{2}: 16 \cdot T(\frac{n}{2}) \T{}{2}: T(\frac{n}{2})
2422 \def\TOhneMathe#1#2{
2423   \tl_if_blank:nTF {#1}
2424   {}
2425   {#1 \cdot }
2426   T
2427   \RundeKlammer{\frac{n}{#2}}
2428 }
2429 \def\T#1#2{
2430   \ifmmode
2431     \TOhneMathe{#1}{#2}
2432   \else
2433     $\TOhneMathe{#1}{#2}$
2434   \fi
2435 }

\bRekursionsGleichung \bRekursionsGleichung: T(n) = a \cdot T(\frac{n}{b}) + f(n)
2436 \def\bRekursionsGleichung{
2437   $T(n) = \T{a}{b} + f(n)$
2438 }

\bBedingungEins \bBedingungEins: f(n) \in \mathcal{O}(n^{\log_b a - \varepsilon})
2439 \def\bBedingungEins{
2440   $f(n) \in \O{n^{\log\sb{b}a - \varepsilon}}$
2441 }

\bBedingungZwei \bBedingungZwei: f(n) \in \Theta(n^{\log_b a})
2442 \def\bBedingungZwei{
2443   $f(n) \in \Theta{n^{\log\sb{b}a}}$
2444 }

\bBedingungDrei \bBedingungDrei: f(n) \in \Omega(n^{\log_b a + \varepsilon})
2445 \def\bBedingungDrei{
2446   $f(n) \in \Omega{n^{\log\sb{b}a + \varepsilon}}$
2447 }

2448 \ExplSyntaxOff

```


\bMasterVariablen

```

2449 \def\bMasterVariablen{
2450   \begin{displaymath}
2451     T(n) = \bT{a}{b} + f(n)
2452   \end{displaymath}
2453
2454   \begin{itemize}
2455     \item[$a = $]
2456       Anzahl der rekursiven Aufrufe, Anzahl der Unterprobleme in der
2457       Rekursion
2458       ($a \geq 1$).
2459
2460     \item[$\textstyle{\frac{1}{b}} = $]
2461       Teil des Originalproblems, welches wiederum durch alle Unterprobleme
2462       repräsentiert wird, Anteil an der Verkleinerung des Problems ($b > 1$).
2463
2464     \item[$f(n) = $]
2465       Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und
2466       die Kombination der Teillösungen entstehen. Eine von  $T(n)$ 
2467       unabhängige und nicht negative Funktion.
2468   \end{itemize}
2469   \footcite{wiki:master-theorem}
2470   \footcite[Seite 19-35]{aud:fs:2}
2471 }
```

\bMasterFaelle

```

2472 \def\bMasterFaelle{
2473   \begin{description}
2474     \item[1. Fall:]
2475        $T(n) \in \bTheta{n^{\log_{\sb{b}}a}}$ 
2476
2477       \hfill falls \bBedingungEins
2478       für  $\varepsilon > 0$ 
2479
2480     \item[2. Fall:]
2481        $T(n) \in \bTheta{n^{\log_{\sb{b}}a} \cdot \log n}$ 
2482
2483       \hfill falls \bBedingungZwei
2484
2485     \item[3. Fall:]
2486        $T(n) \in \bTheta{f(n)}$ 
2487
2488       \hfill falls \bBedingungDrei
2489       für  $\varepsilon > 0$ 
2490       und ebenfalls für ein  $c$  mit  $0 < c < 1$  und alle hinreichend großen  $n$ 
2491       gilt:
2492        $a \cdot \textstyle{\frac{n}{b}} \leq c \cdot n$ 
2493   \end{description}
2494 }
```

\bMasterVariablenDeklaration

```

2495 \def\bMasterVariablenDeklaration#1#2#3{
2496   \begin{description}
2497     \item[Allgemeine Rekursionsgleichung:] \strut
2498
2499     \bRekursionsGleichung
2500
2501     \item[Anzahl der rekursiven Aufrufe ($a$):] \strut
2502
2503     #1
2504
2505     \item[Anteil Verkleinerung des Problems ($b$):] \strut
2506
2507     #2
2508
2509     #3
2510   \end{description}
2511 }
```

```

2507   um  $\frac{1}{\#2}$  also  $b = \#2$ 
2508
2509   \item[Laufzeit der rekursiven Funktion ( $f(n)$ ):] \strut
2510
2511    $\#3$ 
2512
2513   \item[Ergibt folgende Rekursionsgleichung:] \strut
2514
2515    $T(n) = T(\#1\{\#2\} + \#3$ 
2516 \end{description}
2517 }

```

\bMasterFallRechnung

```

2518 \def\bMasterFallRechnung#1#2#3{
2519   \begin{description}
2520   \item[1. Fall:] \bBedingungEins:
2521
2522     #1
2523
2524   \item[2. Fall:] \bBedingungZwei:
2525
2526     #2
2527
2528   \item[3. Fall:] \bBedingungDrei:
2529
2530     #3
2531   \end{description}
2532 }

```

\bMasterExkurs

```

2533 \def\bMasterExkurs{
2534   \begin{bExkurs}[Master-Theorem]
2535   \bMasterVariablen
2536
2537   \noindent
2538   Dann gilt:
2539
2540   \bMasterFaelle
2541   \end{bExkurs}
2542 }

```

\bMasterWolframLink Link zu Wolframalpha (+ durch Prozent 2B ersetzen)

```

2543 \def\bMasterWolframLink#1{
2544   Berechne die Rekursionsgleichung auf WolframAlpha:
2545   \href{https://www.wolframalpha.com/input/?i=\#1\{WolframAlpha\}
2546 }

```

```

2547

```

mathe.sty

```
2548 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2549 \ProvidesPackage{bschlangaul-mathe}[2020/06/10]
2550
2551 % for example \ltimes \rtimes
2552 %\RequirePackage{amssymb}
2553 \RequirePackage{amsmath}
2554
2555 %%
2556 % \mlq \mrq
2557 %%
2558 \DeclareMathSymbol{\mlq}{\mathord}{operators}{`}`}
2559 \DeclareMathSymbol{\mrq}{\mathord}{operators}{`'}`}
2560
```

meta.sty

```
2561 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2562 \ProvidesPackage{bschlangaul-klassen-meta}[2021/09/10 Sammlung
2563 von Textschnipseln, die das Projekt beschreiben]
```

```
2564 \ExplSyntaxOn
```

```
2565 \bLadePakete{grafik}
```

Einfache Makros (Low level)

`\bMetaBschlangaulSammlung`

```
2566 \def\bMetaBschlangaulSammlung
2567 {
2568   Die~Bschlangaul~Sammlung
2569 }
```

`\bMetaHermineFriends`

```
2570 \def\bMetaHermineFriends
2571 {
2572   Hermine~Bschlangaul~and~Friends
2573 }
```

`\bMetaUeberDasProjekt`

```
2574 \def\bMetaUeberDasProjekt
2575 {
2576   Eine~freie~Aufgabensammlung~mit~Lösungen~
2577   von~Studierenden~für~Studierende~
2578   zur~Vorbereitung~auf~die~1.~Staatsexamensprüfungen~
2579   des~Lehramts~Informatik~in~Bayern.
2580 }
```

`\bMetaCCLink`

```
2581 \def\bMetaCCLink
2582 {
2583   Diese~Materialsammlung~unterliegt~den~Bestimmungen~der~
2584   \href
2585   {
2586     https://creativecommons.org/licenses/by-nc-sa/4.0/deed.de
2587   }
2588   {
2589     Creative~Commons~Namensnennung~Nicht~kommerziell~Share~Alike~4.0~
2590     International~Lizenz
2591   }.
2592 }
```

`\bMetaEmail`

```
2593 \def\bMetaEmail
2594 {
2595   hermine.bschlangaul@gmx.net
2596 }
```

`\bMetaEmailLink`

```
2597 \def\bMetaEmailLink
2598 {
2599   \href
2600   {
2601     mailto:\bMetaEmail
2602   }{
2603     \bMetaEmail
2604   }
2605 }
```

`\bMetaHilfMit`

```
2606 \def\bMetaHilfMit
2607 {
2608   Hilf~mit!~
2609
2610   Die~Hermine~schafft~das~nicht~alleine!~
2611
2612   Das~ist~ein~Community~Projekt.~
2613
2614   Verbesserungsvorschläge,~Fehlerkorrekturen,~weitere~Lösungen~sind~
2615   herzlich~willkommen~--egal~wie~~per~Pull-Request~oder~per~E-Mail~an~
2616   \bMetaEmailLink.
2617 }
```

`\bMetaHilfMit`

```
2618 \def\bMetaQuelltext
2619 {
2620   Der~\TeX-Quelltext~dieses~Dokuments~kann~unter~folgender~
2621   URL~aufgerufen~werden:~
2622 }
```

Zusammengesetzte Makros (High level)

Plaziert zwei Minipages nebeneinander. Die erste Umgebung ist für eine Logo gedacht, die zweite für einen Text

```
2623 \cs_new:Npn \logo_dann_text:nn #1 #2
2624 {
2625   \begin{center}
2626     \begin{minipage}[c]{5.5cm}
2627       #1
2628     \end{minipage}
2629
2630     \begin{minipage}[c]{10cm}
2631       #2
2632     \end{minipage}
2633   \end{center}
2634 }
```

`\bLogoTextProjekt`

```
2635 \def\bLogoTextProjekt
2636 {
2637   \logo_dann_text:nn
2638   {
2639     \bGrafikLogo[width=5cm]
2640   }
2641   {
2642     {
2643       \bfseries
2644       \bMetaBschlangaulSammlung
2645     }
2646     \par
2647
2648     \bMetaHermineFriends
2649     \par
2650
2651     \medskip
2652
2653     \begin{spacing}{1}
2654       \footnotesize
2655       \bMetaUeberDasProjekt
2656     \end{spacing}
2657   }
2658 }
```

`\bLogoTextCCLizenz`

```
2659 \def\bLogoTextCCLizenz
2660 {
2661   \logo_dann_text:nn
2662   {
2663     \centerline{\bGrafikCCLizenz[width=3cm]}
2664   }
2665   {
2666     \begin{spacing}{1}
2667       \scriptsize
2668       \bMetaCCLink
2669     \end{spacing}
2670   }
2671 }

2672 \ExplSyntaxOff
2673
```

minimierung.sty

```
2674 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2675 \ProvidesPackage{bschlangaul-minimierung}[2021/03/13 Für den
2676 Minimierungsalgorithmus von einem NEA zu einem DEA]

2677 \bLadePakete{typographie}

\let\z=\bZustandsnameTiefgestellt
\let\f=\bFussnote
\let\l=\bLeereZelle
\let\Z=\bZustandsPaar
\let\erklaerung=\bMinimierungErklaerung

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

\bFussnoten

\begin{liUebergangsTabelle}{0}{1}
\Z01 & \Z10 & \Z23 & \l \\
\Z05 & \Z15 & \Z25 & \f2 \l \\
\Z15 & \Z05 & \Z35 & \f2 \l \\
\Z23 & \Z44 & \Z55 & \l \\
\Z24 & \Z44 & \Z55 & \l \\
\Z34 & \Z44 & \Z55 & \l \\
\end{liUebergangsTabelle}

\bFussnote

2678 \def\bFussnote#1{\$x_{#1}\$}

2679 \def\li@fussnote@text#1#2{
2680 \bFussnote{#1}
2681 \quad
2682 {\footnotesize #2}
2683 }

\bFussnoteEinsText
2684 \def\bFussnoteEinsText{
2685 \li@fussnote@text{1}
2686 {Paar aus End-/ Nicht-Endzustand kann nicht äquivalent sein.}
2687 }

\bFussnoteZweiText
2688 \def\bFussnoteZweiText{
2689 \li@fussnote@text{2}
2690 {Test, ob man mit der Eingabe zu einem bereits markiertem Paar kommt.}
2691 }

\bFussnoteDreiText
2692 \def\bFussnoteDreiText{
2693 \li@fussnote@text{3}
```

```

2694 {In weiteren Iterationen markierte Zustände.}
2695 }

\bFussnoteVierText

2696 \def\bFussnoteVierText{
2697   \li@fussnote@text{4}
2698   {...}
2699 }

\bFussnoten
 $x_1$    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.
 $x_4$    ...

2700 \def\bFussnoten{
2701   \bigskip
2702
2703   \noindent
2704   \bFussnoteEinsText
2705
2706   \noindent
2707   \bFussnoteZweiText
2708
2709   \noindent
2710   \bFussnoteDreiText
2711
2712   \noindent
2713   \bFussnoteVierText
2714 }

\bLeereZelle \bLeereZelle:  $\emptyset$ 
Makro-Faulenzer: \let\l=\liLeereZelle

2715 \def\bLeereZelle{\$\emptyset\$}

\bZustandsPaarVariablenName

2716 \def\bZustandsPaarVariablenName{z}

\bZustandsPaar

2717 \def\bZustandsPaar#1#2{
2718   $(
2719     \bZustandsPaarVariablenName_#1,
2720     \bZustandsPaarVariablenName_#2
2721   )$
2722 }

liUebergangsTabelle

2723 \renewcommand{\arraystretch}{1.4}
2724 \NewDocumentEnvironment{ liUebergangsTabelle } { m m } {
2725   \bPseudoUeberschrift{Übergangstabelle}
2726   \begin{center}
2727     \begin{tabular}{r|l|l}
2728       \textbf{Zustandspaar} & \textbf{#1} & \textbf{#2} \\ \hline
2729     \end{tabular}
2730   \end{center}
2731 }
2732 }

bUeberschriftDreiecksTabelle \bUeberschriftDreiecksTabelle:

Minimierungstabelle (Table filling)

2733 \ExplSyntaxOn

```



```

2734 \def\bUeberschriftDreiecksTabelle{
2735   \bPseudoUeberschrift{Minimierungstabelle~(Table~filling)}
2736 }

```

\bMinimierungErklaerung **Makro-Faulenzer:** \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. —

```

2737 \def\bMinimierungErklaerung{
2738   %\footcite[Seite-19]{koenig}
2739   \bParagraphMitLinien{
2740     Der~Minimierungs-Algorithmus~(auch~Table-Filling-Algorithmus~genannt)~
2741     trägt~in~seinem~Verlauf~eine~Markierung~in~alle~
2742     diejenigen~Zellen~der~Tabelle~ein,~die~zueinander~nicht~äquivalente~
2743     Zustände~bezeichnen.~Die~Markierung~„ $x_{\sb{n}}$ “~in~einer~Tabellenzelle~( $i$ ,~
2744      $j$ )~bedeutet~dabei,~dass~das~Zustandspaar~( $i$ ,~ $j$ )~in~der~ $k$ -ten~
2745     Iteration~des~Algorithmus~markiert~wurde~und~die~Zustände~ $i$ ~und~ $j$ ~
2746     somit~zueinander~( $k-1$ )~äquivalent,~aber~nicht~ $k$ -äquivalent~und~
2747     somit~insbesondere~nicht~äquivalent~sind.~Bleibt~eine~Zelle~bis~zum~Ende~
2748     unmarkiert,~sind~die~entsprechenden~Zustände~zueinander~äquivalent.
2749   }
2750 }
2751 \ExplSyntaxOff

```

2752

normalformen.sty

```
2753 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2754 \ProvidesPackage{bschlangaul-normalformen}[2020/12/10]
2755 Hilfsmakros zum Setzen von Normalformen, Funktionalen Abhängigkeiten,
2756 Attributhülle]
```

Mathe für textit tex etc, typographie für geschweifte Klammer

```
2757 \bLadePakete{mathe,typographie}
2758 \directlua{
2759   helfer = require('bschlangaul-helfer')
2760   normalformen = require('bschlangaul-normalformen')
2761 }
```

Faulenzer

```
\let\ah=\bAttributHuelle
\let\ahL=\bLinksReduktion
\let\ahl=\bLinksReduktionInline
\let\ahr=\bRechtsReduktionInline
\let\fa=\bFunktionaleAbhaengigkeit
\let\FA=\bFunktionaleAbhaengigkeiten
\let\m=\bAttributMenge
\let\r=\bRelation
\let\u=\underline
```

```
2762 \def\bTeilen#1{
2763   \directlua{tex.print(normalformen.teilen('#1'))}
2764 }
```

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

```
2765 \def\bAttributHuelleOhneMathe#1{\text{AttrHülle}\{#1\}}
2766 \def\bAttributHuelle#1{
2767   \ifmmode
2768     \bAttributHuelleOhneMathe{#1}
2769   \else
2770     $\bAttributHuelleOhneMathe{#1}$
2771   \fi
2772 }
```

\bAttributMenge Makro-Faulenzer: $\text{let } \text{m} = \text{liAttributMenge}$

```
2773 \def\bAttributMenge#1{\{ \textit{#1} \}}
```

liAHuelle

```
2774 \NewDocumentEnvironment{ liAHuelle } { +b } {
2775   \begingroup
2776   \footnotesize
2777   \begin{multline*}
2778     #1
2779   \end{multline*}
2780   \endgroup
2781 } { }
```

\bLinksReduktion Nur innerhalb von liAHuelle zu verwenden bzw. multline

Makro-Faulenzer: $\text{let } \text{ahL} = \text{liLinksReduktion}$
 $\text{ahL}\{\text{ursprüngliche linke Attributmenge}\{\text{ohne dieses Attribut}\}\{\text{Ergebnis}\}$

```
2782 \def\bLinksReduktion#1#2#3{
2783   \shoveleft{
2784     \bAttributHuelleOhneMathe{FA,
2785       \bAttributMenge{#1 \string\ #2}} =
2786   } \}
```

```

2787 \shoveright{
2788     \bAttributMenge{#3}
2789 } \
2790 }

```

```

\bLinksReduktionInline Makro-Faulenzer: \let\ahL=\liLinksReduktionInline
    \ahl{ursprüngliche linke Attributmenge}{ohne dieses Attribut}{Ergebnis}
    \ahl{A, E}{E}{A, E, F, B, \textbf{D}}: AttrHülle( $F, \{A, E \setminus E\} = \{A, E, F, B, D\}$ )
2791 \def\bLinksReduktionInline#1#2#3{%
2792     {%
2793         \footnotesize%
2794         $\bAttributHuelleOhneMathe{F,
2795             \bAttributMenge{#1 \string\ #2}} =
2796             \bAttributMenge{#3}$
2797     }
2798 }

```

```

\bLinksReduktionInline Makro-Faulenzer: \let\ahr=\liLinksReduktionInline
    \ahr{gelöschte FA}{neue FA ohne rechts Attribut}{gegebene Attribute}{Ergebnis}
2799 \def\bRechtsReduktionInline#1#2#3#4{%
2800     {%
2801         \footnotesize%
2802         $\bAttributHuelleOhneMathe{
2803             F \setminus
2804             \bFunktionaleAbhaengigkeit{#1}
2805             \def\tmp{#2}\ifx\tmp\empty
2806             \else
2807                 \cup \bFunktionaleAbhaengigkeit{#2}
2808             \fi
2809             ,
2810             \bAttributMenge{#3}
2811         } =
2812         \bAttributMenge{#4}$
2813     }
2814 }

```

```

\bFunktionaleAbhaengigkeit Makro-Faulenzer: \let\fa=\liFunktionaleAbhaengigkeit
    \fa{A, B -> C, D}:  $\{A, B\} \rightarrow \{C, D\}$  Regulärer Ausdruck zum Konvertieren:  $\$(.*) \rightarrow$ 
    \fa{$1 -> $2}
2815 \def\bFunktionaleAbhaengigkeit#1{%
2816     \directlua{normalformen.drucke_funk_abhaengigkeit('#1', false)}%
2817 }

```

```

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

```

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

```

Makro-Faulenzer: \let\FA=\liFunktionaleAbhaengigkeiten
    Regulärer Ausdruck zum Konvertieren:  $\backslash\item \$(.*) \rightarrow (.)\$(.*) \rightarrow$ 
    $1 -> $2;

```

```

2818 \NewDocumentCommand {\bFunktionaleAbhaengigkeiten} { O{FA} m } {
2819   \bGeschweifteKlammern
2820   {#1}
2821   {
2822     \begin{align*}
2823       \directlua{normalformen.drucke_funk_abhaengigkeiten('#2')}
2824     \end{align*}
2825   }
2826   {-0.5cm}
2827   {-1.7cm}
2828 }

\bRelation Makro-Faulenzer: \let\r=\liRelation
           \r[R3]{\u{A}, B, C}:  $R_3(\underline{A}, B, C)$  Regulärer Ausdruck zum Konvertieren:  $\$(R.*)\left((.*)\right)\$$ 
           \\bRelation[$1]{$2}

2829 \NewDocumentCommand {\bRelation} { O{R} m } {
2830   $\directlua{
2831     local name = helper.konvertiere_tiefgestellt('#1')
2832     tex.print(name)
2833   }$(\textit{\,,#2\,,})
2834 }

2835

```

o-notation.sty

2836 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

2837 \ProvidesPackage{bschlangaul-o-notation}[2021/09/08]

Faulenzer

\let\O=\bONotationO

TeX-Markup-Beispiel: Funktionsgraphen mit pgfplots

```
\begin{tikzpicture}
  \begin{axis}[
    xlabel=$n$,
    legend entries={\f a, \f b, \f c, \f d, \f e},
    ymax=500,
    xmin=0,
    xmax=7,
    legend pos=north west,
    domain=0:7
  ]
    \addplot{sqrt(x^5) + (4 * x) - 5};
    \addplot{log2(log2(x))};
    \addplot{2^x};
    \addplot{x^2 * log10(x) + (2 * x)};
    \addplot{4^x / (log2(x))};
  \end{axis}
\end{tikzpicture}
```

2838 \ExplSyntaxOn

2839 \RequirePackage{amssymb}

2840 \RequirePackage{pgfplots}

Für echte Teilmenge \subsetneq: \subsetneq

2841 \RequirePackage{amssymb}

\bRundeKlammer

```
2842 \def\bRundeKlammer#1{
2843   \negthinspace \left( #1 \right)
2844 }
```

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

```
2845 \cs_new:Npn \o_notation_O:n #1 {
2846   \mathcal{O} \bRundeKlammer{#1}
2847 }
2848 \def\bONotationO#1{
2849   \ifmmode
2850     \o_notation_O:n { #1 }
2851   \else
2852     $ \o_notation_O:n { #1 } $
2853   \fi
2854 }
```

2855

petri.sty

```
2856 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2857 \ProvidesPackage{bschlangaul-petri}[2020/12/03]
```

Faulenzer

```
\let\t=\bPetriTransitionsName
\let\tp=\bPetriTransPfeile
\let\k=\bPetriErreichKnotenDrei
```

```
2858 \RequirePackage{tikz}
2859 \usetikzlibrary{petri,arrows.meta}
```

Für die Darstellungsmatrix

```
2860 \RequirePackage{blkarray}
```

```
\def\TmpA#1{
  \bPetriSetzeSchluessel%
  \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}
}
```

\bPetriSetzeSchluessel

```
2861 \def\bPetriSetzeSchluessel{%
2862   \def\TmpTransitionOne{}%
2863   \def\TmpTransitionTwo{}%
2864   \def\TmpTransitionThree{}%
2865   \def\TmpTransitionFour{}%
2866   \def\TmpTransitionFive{}%
2867   \def\TmpTransitionSix{}%
2868   \def\TmpTransitionSeven{}%
2869   \def\TmpTransitionEight{}%
2870   \def\TmpTransitionNine{}%
2871   \def\TmpTransitionTen{}%
2872   \pgfkeys{/petri/.cd,
2873     p1/.store in=\TmpPlaceOne,p1/.default=0,p1,
2874     p2/.store in=\TmpPlaceTwo,p2/.default=0,p2,
2875     p3/.store in=\TmpPlaceThree,p3/.default=0,p3,
2876     p4/.store in=\TmpPlaceFour,p4/.default=0,p4,
```

```

2877 p5/.store in=\TmpPlaceFive,p5/.default=0,p5,
2878 p6/.store in=\TmpPlaceSix,p6/.default=0,p6,
2879 p7/.store in=\TmpPlaceSeven,p7/.default=0,p7,
2880 p8/.store in=\TmpPlaceEight,p8/.default=0,p8,
2881 p9/.store in=\TmpPlaceNine,p9/.default=0,p9,
2882 p10/.store in=\TmpPlaceTen,p10/.default=0,p10,
2883 t1/.store in=\TmpTransitionOne,t1/.default=activated,
2884 t2/.store in=\TmpTransitionTwo,t2/.default=activated,
2885 t3/.store in=\TmpTransitionThree,t3/.default=activated,
2886 t4/.store in=\TmpTransitionFour,t4/.default=activated,
2887 t5/.store in=\TmpTransitionFive,t5/.default=activated,
2888 t6/.store in=\TmpTransitionSix,t6/.default=activated,
2889 t7/.store in=\TmpTransitionSeven,t7/.default=activated,
2890 t8/.store in=\TmpTransitionEight,t8/.default=activated,
2891 t9/.store in=\TmpTransitionNine,t9/.default=activated,
2892 t10/.store in=\TmpTransitionTen,t10/.default=activated,
2893 scale/.store in=\TmpScale,scale/.default=0.5,
2894 x/.store in=\TmpX,x/.default=5,
2895 y/.store in=\TmpY,y/.default=5,
2896 }%
2897 }

2898 \tikzset{
2899   li petri/.style={
2900     activated/.style={
2901       very thick
2902     },
2903     inhibitor/.style={
2904       {Circle[open,length=2mm,fill=white]}-
2905     }
2906   }
2907 }

```

```

\bPetriTransitionsName Makro-Faulenzer: \let\t=\liPetriTransitionsName
    \t_(\d+)\$ \t$1
2908 \def\bPetriTransitionsNameOhneMathe#1{t\sb{#1}}
2909 \def\bPetriTransitionsName#1{
2910   \ifmmode
2911     \bPetriTransitionsNameOhneMathe{#1}
2912   \else
2913     $\bPetriTransitionsNameOhneMathe{#1}$
2914   \fi
2915 }

```

```

\bPetriErreichTransition Makro-Faulenzer: \let\t=\liPetriErreichTransition
2916 \NewDocumentCommand{ \bPetriErreichTransition } { m m m 0{ } 0{ } } {
2917   \draw[->] (#1) edge[#4] node[pos=0.5,auto,sloped,#5]{$t\sb{#3}$} (#2);
2918 }

```

```

\bPetriErreichKnotenDrei Makro-Faulenzer: \let\k=\liPetriErreichKnotenDrei
2919 \def\bPetriErreichKnotenDrei#1#2#3{(#1,#2,#3)}

```

```

\bPetriTransPfeile Makro-Faulenzer: \let\tp=\liPetriTransPfeile
2920 \def\bPetriTransPfeile#1{$\rightarrow \hspace{0.4cm} \bPetriTransitionsName{#1} \hspace{0.35cm}
2921

```

potenzmengen-konstruktion.sty

```
2922 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2923 \ProvidesPackage{bschlangaul-potenzmengen-konstruktion}[2021/02/21 Hilfsmakros
2924 einzusetzen bei der Potenzmengen-Konstruktion (Potenzmengen-Algorithmus)]
```

```
2925 \bLadePakete{formale-sprachen}
```

```
2926 \ExplSyntaxOn
```

```
\def\z#1{
  \bZustandsMengenSammlungNr{#1}{
    {
      {0} {0}
      {1} {0,1}
      {2} {0,2}
      {3} {0,1,3}
      {4} {0,2,3}
      {5} {0,3}
    }
  }
}

\let\s=\bZustandsnameGross

\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}
```

```
\bZustandsMengenSammlung \bZustandsMengenSammlung{nummer}{latex3 str_case:nn}
```

```
\def\z#1{
  \bZustandsMengenSammlung{#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}
    }
  }
}
```

```
2927 \def\bZustandsMengenSammlung#1#2{
2928   \bZustandsnameGross{#1}
2929   {
2930     \footnotesize
2931     \bPotenzmenge{
2932       \str_case:nn {#1} {#2
2933       }
2934     }
2935 }
```

```
\bZustandsMengenSammlungNr
```

```
2936 \def\bZustandsMengenSammlungNr#1#2{
2937   \bZustandsnameGross{#1}
2938   {
```



```
2939     \footnotesize
2940     \bZustandsmengeNr{
2941         \str_case:nn {#1} #2
2942     }
2943 }
2944 }

2945 \ExplSyntaxOff
2946
```

pseudo.sty

```

2947 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2948 \ProvidesPackage{bschlangaul-pseudo}[2020/12/30 Paket zum Setzen
2949 von Pseudo-Code, Hüll-Paket um algorithm2e]

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

```

Algorithmus 1: Minimaler Spannbaum nach Kruskal

Data: $G = (V, E, w)$: ein zusammenhängender, ungerichteter, kantengewichteter Graph $\text{kruskal}(G)$

$E' \leftarrow \emptyset$;
 $L \leftarrow E$;
 Sortiere die Kanten in L aufsteigend nach ihrem Kantengewicht.;

while $L \neq \emptyset$ **do**
 wähle eine Kante $e \in L$ mit kleinstem Kantengewicht;
 entferne die Kante e aus L ;
 if der Graph $(V, E' \cup \{e\})$ keinen Kreis enthält **then**
 $E' \leftarrow E' \cup \{e\}$;
 end
end

Result: $M = (V, E')$ ist ein minimaler Spannbaum von G .

```

2950 \RequirePackage[german,boxruled]{algorithm2e}

```

```

2951

```

pumping-lemma.sty

2952 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2953 \ProvidesPackage{bschlangaul-pumping-lemma}[2021/08/11 Enthält die
2954 Definitionen für das Pumping-Lemma in der Regulären Sprache und
2955 in der Kontextfreien Sprache]

\bPumpingRegulaer

2956 \def\bPumpingRegulaer{%
2957 Es sei L eine reguläre Sprache. Dann gibt es eine Zahl j , sodass für
2958 alle Wörter ω in L mit $|\omega| \geq j$ (jedes Wort ω in
2959 L mit Mindestlänge j) jeweils eine Zerlegung $\omega = uvw$
2960 existiert, sodass die folgenden Eigenschaften erfüllt sind:
2961
2962 \begin{enumerate}
2963 \item $|v| \geq 1$
2964 (Das Wort v ist nicht leer.)
2965
2966 \item $|uv| \leq j$
2967 (Die beiden Wörter u und v haben zusammen höchstens die Länge j .)
2968
2969 \item Für alle $i = 0, 1, 2, \dots$ gilt $uv^i w$ in L
2970 (Für jede natürliche Zahl (mit 0) i ist das Wort $uv^i w$ in der
2971 Sprache L)
2972 \end{enumerate}
2973
2974 Die kleinste Zahl j , die diese Eigenschaften erfüllt, wird
2975 Pumping-Zahl der Sprache L genannt.\footcite{wiki:pumping-lemma}
2976 }

\bPumpingKontextfrei

2977 \def\bPumpingKontextfrei{%
2978 Es sei L eine kontextfreie Sprache. Dann gibt es eine Zahl j , sodass
2979 sich alle Wörter ω in L mit $|\omega| \geq j$ zerlegen lassen in
2980 $\omega = uvwxy$, sodass die folgenden Eigenschaften erfüllt sind:
2981
2982 \begin{enumerate}
2983 \item $|vx| \geq 1$
2984 (Die Wörter v und x sind nicht leer.)
2985
2986 \item $|vwx| \leq j$
2987 (Die Wörter v , w und x haben zusammen höchstens die Länge j .)
2988
2989 \item Für alle i in \mathbb{N}_0 gilt uv^iwx^iy in L (Für jede
2990 natürliche Zahl (mit 0) i ist das Wort uv^iwx^iy in der
2991 Sprache L)
2992 \end{enumerate}
2993 }

2994

relationale-algebra.sty

```

2995 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
2996 \ProvidesPackage{bschlangaul-relationale-algebra}[2020/12/11]
2997 \RequirePackage{amsmath}
2998 \RequirePackage{amssymb}

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

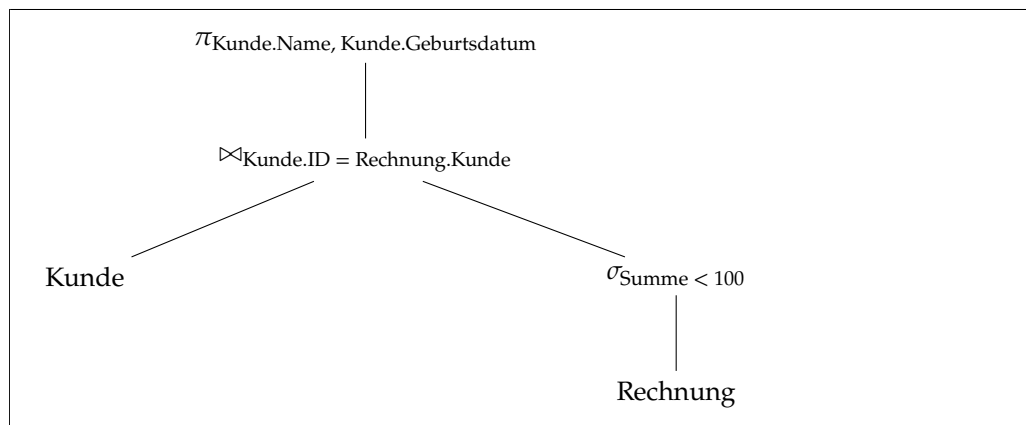
  \node[below=of pi]
    (theta join) {$\bowtie_{\text{Kunde.ID = Rechnung.Kunde}}$}
    edge (pi);

  \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}

```



```

2999 \RequirePackage{tikz}
3000 \usetikzlibrary{positioning}

    Privates Makros, das zwei Querstriche erzeugt.
3001 \def\o@join{\setbox0=\hbox{$\bowtie$}%
3002   \rule[-.02ex]{.25em}{.4pt}\llap{\rule[\ht0]{.25em}{.4pt}}%
3003 }

\leftouterjoin A \leftouterjoin B: A ⋈ B
3004 \def\leftouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie}}

\rightouterjoin A \rightouterjoin B: A ⋈ B
3005 \def\rightouterjoin{\mathbin{\bowtie\mkern-5.8mu\o@join}}

\fullouterjoin A \fullouterjoin B: A ⋈ B
3006 \def\fullouterjoin{\mathbin{\o@join\mkern-5.8mu\bowtie\mkern-5.8mu\o@join}}
3007

```

rmodell.sty

```
3008 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3009 \ProvidesPackage{bschlangaul-rmodell}[2020/09/01
3010 Makros und Umgebungen zum Setzen des Relationenmodells beim Thema
3011 Datenbanken.]
3012 \RequirePackage{soul}
```

Faulenzer

```
\let\ a=\bAttribut
\let\ f=\bFremd
\let\ p=\bPrimaer
\let\ r=\bRelationMenge
```

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

```
3013 \def\bPrimaer#1{\ul{#1}}
```

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

```
3014 \def\bFremd#1{{\setul{-0.9em}{}}\ul{#1}}
```

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

```
3015 \def\li@Rmodell@Schrift{\footnotesize\ttfamily}
3016 \ExplSyntaxOn
3017 \NewDocumentEnvironment { liRmodell }
3018 { +b }
3019 {
3020   \medskip
3021   {
3022     \linespread{2}
3023     \setlength{\parindent}{0pt}
3024     \li@Rmodell@Schrift#1
3025   }
3026   \medskip
3027 } {}
3028 \ExplSyntaxOff
```

`\bRelationMenge` **Makro-Faulenzer:** `\let\ r=\liRelationMenge`

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

```
3029 \def\bRelationMenge#1#2{
3030 \noindent
3031 #1 : \[ #2 ]\}
3032 \par
3033 }
```

`\bAttribut` **Makro-Faulenzer:** `\let\ a=\liAttribut`

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

```
3034 \def\bAttribut#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}
```

```
3035 \NewDocumentEnvironment { liRelationenSchemaFormat }{ +b } {} {}
```

```
3036
```

sortieren.sty

```
3037 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3038 \ProvidesPackage{bschlangaul-sortieren}[2020/06/10 Schaubilder
3039 für händisches Sortieren bzw. einen Schreibtischlauf setzen]
```

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



```
3040 \RequirePackage{tikz}
3041 \usetikzlibrary{shapes.multipart,positioning}
```

`\bVertauschen` `\bVertauschen{1 2 >4 <3 5}`: Setze ein Schaubild mit Hilfe von TikZ. < und > werden dazu verwendet, um den Vertauschprozess zu visualisieren.

```
3042 \def\bVertauschen#1{
3043   \directlua{
3044     local sortieren = require('bschlangaul-sortieren')
3045     sortieren('#1')
3046   }
3047 }
```

`\bSortierPfeil`

```
3048 \def\bSortierPfeil#1#2{
3049   \draw[-latex] ([xshift=1mm]reihe.#1 north) -- ++(0,0.25) -| ([xshift=-1mm]reihe.#2 north);
3050 }
```

`\bSortierPfeilUnten`

```
3051 \def\bSortierPfeilUnten#1#2{
3052   \draw[-latex] ([xshift=1mm]reihe.#1 south) -- ++(0,-0.25) -| ([xshift=-1mm]reihe.#2 south);
3053 }
```

`\bSortierMarkierung`

```
3054 \def\bSortierMarkierung#1#2{\node[
3055   draw,
3056   very thick,
3057   fit=(reihe.#1) (reihe.#2),
3058   inner sep=0pt
3059 ] {}
3060 }

3061 \tikzset{
3062   li sortierung zahlenreihe/.style={
3063     draw,
3064     thin,
3065     font=\large,
3066     rectangle split horizontal,
3067     rectangle split,
3068   }
3069 }
```

```

3070 % https://tex.stackexchange.com/a/140895
3071 \RequirePackage{forest,xstring}
3072 \usetikzlibrary{calc}
3073
3074 \makeatletter
3075 \pgfmathdeclarefunction{strrepeat}{2}{%
3076   \begingroup\pgfmathint{#2}\pgfmathcount\pgfmathresult
3077   \let\pgfmathresult\pgfutil@empty
3078   \pgfutil@loop\ifnum\pgfmathcount>0\relax
3079     \expandafter\def\expandafter\pgfmathresult\expandafter{\pgfmathresult#1}%
3080     \advance\pgfmathcount-1\relax
3081     \pgfutil@repeat\pgfmath@smuggleone\pgfmathresult\endgroup}
3082 \makeatother
3083
3084 \def\myNodes{}
3085
3086 \ExplSyntaxOn
3087 \newcommand*\sortList[1]{%
3088   \clist_sort:Nn#1{\int_compare:nNnTF{##1}>{##2}\sort_return_swapped:\sort_return_same:}}
3089 \ExplSyntaxOff
3090
3091 \forestset{
3092   sort/.code={%
3093     \pgfmathparse{level()>\forestSortLevel}%
3094     \ifnum\pgfmathresult=0
3095       \StrSubstitute{\forestov{content}}{ }{ }[\myList]%
3096       \sortList\myList
3097       \StrSubstitute{\myList}{ }{ }[\myList]%
3098       \pgfmathparse{strrepeat("1",level())}%
3099       \xappto\myNodes{\noexpand\node at ($(\forestov{name})|-m)!-1!(\forestov{name}))$}
3100       (m\forestov{name}) {[\myList]}%
3101       \pgfmathparse{level()=\forestSortLevel}%
3102       \ifnum\pgfmathresult=1
3103         \forestOget{\forestov{@first}}{name}\forestFirst
3104         \forestOget{\forestov{@last}}{name}\forestLast
3105         \xappto\myNodes{[<-]edge (\forestOv{\forestov{@first}}{name})
3106           \ifx\forestFirst\forestLast\else edge (\forestOv{\forestov{@last}}{name})\fi}}%
3107       \fi
3108       \ifnum\forestov{@parent}=0\else
3109         \xappto\myNodes{edge (m\forestOv{\forestov{@parent}}{name})}%
3110       \fi
3111       \gappto\myNodes{;}%
3112     \fi}}
3113
3114 \forestset{sort level/.code=%
3115   \pgfmathparse{#1}\let\forestSortLevel\pgfmathresult
3116   \pgfmathparse{strrepeat("1",\forestSortLevel+1)}\let\forestOnes\pgfmathresult}
3117

```

spalten.sty

```
3118 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3119 \ProvidesPackage{bschlangaul-spalten}[2020/12/07 Lädt das Paket
3120 „multicol“, damit mehrspaltiger Satz mit Hilfe der Umgebung „multicols“
3121 realisiert werden kann.]
3122 \RequirePackage{multicol}
```

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

```
3123 \def\bSpaltenUmbruch{\vfill\strut\columnbreak}
```

```
3124
```


sql.sty

3125 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

3126 \ProvidesPackage{bschlangaul-tabelle}[2021/09/02 Zu Setzen von SQL]

Latex-Markup-Beispiel

```
\begin{bAdditum}[Übungsdatenbank]
% Datenbankname: Personalverwaltung
\begin{minted}{sql}
CREATE TABLE Abteilung(
  AbteilungsID INTEGER PRIMARY KEY,
  Bezeichnung VARCHAR(30)
);

CREATE TABLE Mitarbeiter(
  MitarbeiterID INTEGER PRIMARY KEY,
  Vorname VARCHAR(30),
  Vorgesetzter INTEGER REFERENCES Mitarbeiter(MitarbeiterID),
  Telefonnummer VARCHAR(50),
  Gehalt DOUBLE PRECISION
);

INSERT INTO Abteilung VALUES
  (1, 'Buchhaltung');

INSERT INTO Mitarbeiter
  (MitarbeiterID, Vorname, Nachname, Vorgesetzter, AbteilungsID, Telefonnummer, Gehalt)
VALUES
  (1, 'Hans', 'Meier', 11, 4, '023/13432', 2335),
  (2, 'Fred', 'Wolitz', 11, 2, '0233/413432', 1233);
\end{minted}
\index{SQL mit Übungsdatenbank}
\end{bAdditum}
```

3127 \bLadePakete{syntax}

3128 \RequirePackage{fancyvrb}

3129 \DefineVerbatimEnvironment{liSqlErgebnis}{Verbatim}

3130 {fontsize=\footnotesize}

3131

struktogramm.sty

3132 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

3133 \ProvidesPackage{bschlangaul-struktogramm}[2021/01/31 Lädt das
3134 Paket struktex zum Setzen von Struktogrammen]

3135 \RequirePackage{struktex}

3136

syntax.sty

```
3137 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3138 \ProvidesPackage{bschlangaul-syntax}[2020/06/10 Ein Hüll-Paket um
3139 `minted`, das sich um die Syntax-Hervorhebung des Java-Codes kümmert.]
3140 \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.

Faulenzer

```
\let\j=\bJavaCode
\let\s=\bSqlCode
```

```
3141 \ExplSyntaxOn
3142 \directlua{
3143   syntax = require('bschlangaul-syntax')
3144   syntax.importiere_konfiguration('tex_repo_lokaler_pfad', '\LehramtInformatikRepository')
3145   syntax.importiere_konfiguration('github_domain', '\LehramtInformatikGithubDomain')
3146   syntax.importiere_konfiguration('github_raw_domain', '\LehramtInformatikGithubRawDomain')
3147   syntax.importiere_konfiguration('github_tex_repo', '\LehramtInformatikGithubTexRepo')
3148   syntax.importiere_konfiguration('github_code_repo', '\LehramtInformatikGithubCodeRepo')
3149   syntax.importiere_konfiguration('git_branch', '\LehramtInformatikGitBranch')
3150 }
3151 \RequirePackage{hyperref}
3152 \RequirePackage{minted}
3153 % pygmentize -L styles
3154 \usemintedstyle{colorful}
3155 %\BeforeBeginEnvironment{minted}{\begin{mdframed}}
3156 %\AfterEndEnvironment{minted}{\end{mdframed}}
3157 %\setminted{breaklines=true,linenos}
3158 \setminted
3159 {
3160   breaklines=true,
3161   linenos=false,
3162   fontsize=\footnotesize,
3163 }
```

bJavaAngabe Eine Umgebung für Java-Code, ohne Zeilennummer und etwas eingerückt um den Java-Code in Angaben / Aufgabestellungen zu setzen.

```
3164 \newminted[bJavaAngabe]{java}
3165 {
3166   xleftmargin=1cm
3167 }
```

\bJavaCode Im Zeilenfluss einen kurzen Java-Code-Ausschnitt setzen. Es werden automatische Zeilenumbrüche gemacht.

Makro-Faulenzer: `\let\j=\liJavaCode`

```
3168 \def\bJavaCode#1
3169 {
3170   \,
3171   \textcolor{blue}{
3172     \mintinline[
3173       fontsize=\normalsize,
3174       breakanywhere % https://github.com/gpoore/minted/issues/31#issuecomment-458640242
3175     ]{java}|#1|
3176   }
3177   \,
3178 }
```

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

```
3179 \def\bLatexCode#1
3180 {
3181   \mintinline{latex}|#1|
3182 }

3183 \def\li@GithubLink#1#2
3184 {
3185   \begin{flushright}
3186     \tiny
3187     Code-Beispiel~auf~Github~ansehen:~
3188     \href{#1}{\nolinkurl{#2}}
3189   \end{flushright}
3190 }
```

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

```
3191 \NewDocumentCommand{ \bJavaDatei }{ 0{firstline=3} m }
3192 {
3193   \inputminted[#1]{java}{
3194     \directlua{
3195       syntax.drucke_absoluten_pfad('#2', false)
3196     }
3197   }
3198   \li@GithubLink
3199   {\directlua{syntax.drucke_github_url('#2', false)}}
3200   {\directlua{syntax.drucke_relativen_repo_pfad('#2', false)}}
3201 }
```

`\bJavaTestDatei` Eine komplette Java-Test-Datei einbinden, die Verzeichnis `./Code/src/test/java/org/bschlangaul` liegt.

```
3202 \NewDocumentCommand{ \bJavaTestDatei }{ 0{firstline=3} m }
3203 {
3204   \inputminted[#1]{java}{
3205     \directlua{
3206       syntax.drucke_absoluten_pfad('#2', true)
3207     }
3208   }
3209   \li@GithubLink
3210   {\directlua{syntax.drucke_github_url('#2', true)}}
3211   {\directlua{syntax.drucke_relativen_repo_pfad('#2', true)}}
3212 }
```

`\bJavaExamen` `\bJavaExamen{66116}{2015}{03}{Kunde}` Regulärer Ausdruck zum Konvertieren: `\bJavaDatei([.*\bJavaExamen$1{$2}{$3}{$4}{$5}`

```
3213 \NewDocumentCommand{ \bJavaExamen }{ 0{firstline=3} m m m m }
3214 {
3215   \inputminted[#1]{java}{
3216     \directlua{
3217       syntax.drucke_absoluten_examens_pfad('#2', '#3', '#4', '#5')
3218     }
3219   }
3220
3221   \li@GithubLink
3222   {\directlua{syntax.drucke_github_examens_url('#2', '#3', '#4', '#5')}}
3223   {\directlua{syntax.drucke_relativen_examens_repo_pfad('#2', '#3', '#4', '#5')}}
3224 }
```

`\bAssemblerCode`

```
3225 \def\bAssemblerCode#1
3226 {
3227   \mintinline{asm}|#1|
3228 }
```

```

\bAssemblerDatei \bAssemblerDatei{relativer-pfad}: Relativer Pfad zu einer Assembler-Datei.
3229 \NewDocumentCommand { \bAssemblerDatei } { m }
3230 {
3231   \inputminted{asm}{#1}
3232 }

\bMinispracheDatei \bMinispracheDatei{relativer-pfad}: Relativer Pfad zu einer Minisprachen-Datei
(Hochsprache für die Minimaschine von Albert Wiedemann).
3233 \NewDocumentCommand { \bMinispracheDatei } { m }
3234 {
3235   \inputminted{componentpascal}{#1}
3236 }

\bHaskellCode \bHaskellCode{haskell}: Zum Setzen von Haskell-Code.
3237 \def\bHaskellCode#1
3238 {
3239   \mintinline{haskell}|#1|
3240 }

\bHaskellDatei \bHaskellDatei{relativer-pfad}: Relativer Pfad zu einer Haskell-Datei.
3241 \NewDocumentCommand { \bHaskellDatei } { m }
3242 {
3243   \inputminted{haskell}{#1}
3244 }

\bSqlCode \bHaskellCode{sql}: Zum Setzen von SQL-Code.
Makro-Faulenzer: \let\s=\liSqlCode
3245 \def\bSqlCode#1
3246 {
3247   \mintinline{sql}|#1|
3248 }

3249 \ExplSyntaxOff
3250

```

syntaxbaum.sty

3251 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

3252 \ProvidesPackage{bschlangaul-syntaxbaum}[2021/02/14 Zum Setzen von

3253 Syntaxbäumen mit Hilfe des Pakets tikz-qtree]

TeX-Markup-Beispiel

```
\begin{tikzpicture}[b syntaxbaum,level distance=1cm]
\Tree [.S 0 [ 0 [.S $\varepsilon$ ] 0 ] 0 ]
\end{tikzpicture}
```



```
3254 \RequirePackage{tikz-qtree}
3255 \tikzset{b syntaxbaum/.style={
3256     every internal node/.style={
3257         draw,circle
3258     },
3259     every leaf node/.style={
3260         draw,rectangle
3261     },
3262 }
3263 }
3264
```

synthese-algorithmus.sty

```
3265 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3266 \ProvidesPackage{bschlangaul-synthese-algorithmus}[2021/03/19
3267 Hilfsmakros zum Setzen des Synthese-Algorithmuses zur Umformung einer
3268 Relation in die 3. Normalform]

3269 \bLadePakete{
3270   normalformen,
3271   mathe,
3272   typographie
3273 }

3274 \ExplSyntaxOn
```

Faulenzer

```
\let\schrittE=\bSyntheseUeberErklaerung
```

TeX-Markup Grundgerüst

```
\let\schrittE=\bSyntheseUeberErklaerung
\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}
```

TeX-Markup Linksreduktion

```
\let\ahl=\bLinksReduktionInline
\let\fa=\bFunktionaleAbhaengigkeit
\let\m=\bAttributMenge
\let\b=\textbf

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

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

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

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

TeX-Markup Rechtsreduktion

```
\let\ahr=\bRechtsReduktionInline
\let\fa=\bFunktionaleAbhaengigkeit

\bPseudoUeberschrift{F}

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



```
\bPseudoUeberschrift{A}

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

TeX-Markup Relationen formen

```
\let\r=\bRelation
\let\u=\underline

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

(a) Kanonische Überdeckung

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

(i) 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 $\beta \subseteq \text{AttrHülle}(F, \alpha - A)$. —

(ii) 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. —

(iii) Löschen leerer Klauseln

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

(iv) 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. —

(b) Relationsschemata formen

— Erzeuge für jede funktionale Abhängigkeit $\alpha \rightarrow \beta \in F_c$ ein Relationenschema $\mathcal{R}_\alpha := \alpha \cup \beta$. —

(c) Schlüssel hinzufügen

— Falls eines der in Schritt 2. erzeugten Schemata \mathcal{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 $\mathcal{F}_\mathcal{K} := \emptyset$ —

(d) Entfernung überflüssiger Teilschemata

— Eliminiere diejenigen Schemata \mathcal{R}_α , die in einem anderen Relationenschema $\mathcal{R}_{\alpha'}$ enthalten sind, d. h. $\mathcal{R}_\alpha \subseteq \mathcal{R}_{\alpha'}$. —

\bSyntheseUeberschrift **Makro-Faulenzer:** \let\schritt=\liSyntheseUeberschrift

```
3275 \def\bSyntheseUeberschrift#1
3276 {
3277   {
3278     \bfseries
3279     \rmfamily
3280     \str_case:nn {#1} {
3281       {1} {Kanonische~Überdeckung}
3282       {1-1} {Linksreduktion}
3283       {1-2} {Rechtsreduktion}
```

```

3284      {1-3} {Löschen-leerer-Klauseln}
3285      {1-4} {Vereinigung}
3286      {2} {Relationsschemata-formen}
3287      {3} {Schlüssel-hinzufügen}
3288      {4} {Entfernung-überflüssiger-Teilschemata}
3289    }
3290  }
3291 }

```

\bSyntheseErklaerung **Makro-Faulenzer:** \let\erklaerung=\liSyntheseErklaerung

```

3292 \def\li@synthese@erklaerung@texte#1
3293 {
3294   \str_case:nn {#1} {
3295     {1} {
3296       Die-kanonische-Überdeckung---also-die-kleinst-mögliche-noch-
3297       äquivalente-Menge-von-funktionalen-Abhängigkeiten-kann-in-vier-
3298       Schritten-erreicht-werden.
3299     }
3300     {1-1} {
3301       Führe-für-jede-funktionale-Anhängigkeit-
3302        $\alpha \rightarrow \beta$ -in- $F$ -die-Linksreduktion-durch,-
3303       überprüfe-also-für-alle-
3304        $A \in \alpha$ , ob- $A$ -überflüssig-ist,-d.h.-ob-
3305        $\beta \subseteq \text{AttributHuelle}\{F, \alpha\}$ .
3306     }
3307     {1-2} {
3308       Führe-für-jede-(verbliebene)-funktionale-Abhängigkeit- $\alpha \rightarrow \beta$ -die-Rechtsreduktion-durch,-überprüfe-also-für-
3309       alle- $B \in \beta$ , ob- $B \in \text{AttributHuelle}\{F, (\alpha \rightarrow \beta)\}$ ,
3310        $\alpha \rightarrow \beta \cup (\alpha \rightarrow B)$ ,
3311        $\alpha$ -gilt.-In-diesem-Fall-ist- $B$ -auf-der-rechten-Seite-
3312       überflüssig-und-kann-eliminiert-werden,-dh- $\alpha \rightarrow \beta$ -wird-durch- $\alpha \rightarrow (\beta \cup B)$ -
3313       ersetzt.
3314     }
3315     {1-3} {
3316       Entferne-die-funktionalen-Abhängigkeiten-der-Form- $\alpha \rightarrow \emptyset$ ,
3317       die-im-2.-Schritt-möglicherweise-entstanden-sind.
3318     }
3319     {1-4} {
3320       Fasse-mittels-der-Vereinigungsregel-funktionale-Abhängigkeiten-
3321       der-Form- $\alpha \rightarrow \beta_{\{1\}}, \dots, \alpha \rightarrow \beta_{\{n\}}$ ,
3322       so-dass- $\alpha \rightarrow \beta_{\{1\}} \cup \dots \cup \beta_{\{n\}}$ -verbleibt.
3323     }
3324     % Kemper Seite 197
3325     {2} {
3326       Erzeuge-für-jede-funktionale-Abhängigkeit- $\alpha \rightarrow \beta$ -ein-Relationenschema- $\mathcal{R}_{\alpha}$ -
3327        $:= \alpha \cup \beta$ .
3328     }
3329     {3} {
3330       Falls-eines-der-in-Schritt-2.-erzeugten-Schemata- $\mathcal{R}_{\alpha}$ -
3331       einen-Schlüsselkandidaten-von- $\mathcal{R}$ -bezüglich- $F$ -
3332       enthält,-sind-wir-fertig,-sonst-wähle-einen-Schlüsselkandidaten-
3333        $\mathcal{K} \subseteq \mathcal{R}$ -aus-und-definiere-folgendes-
3334       zusätzliche-Schema:- $\mathcal{R}_{\mathcal{K}} := \mathcal{K}$ -
3335       und- $\mathcal{F}_{\mathcal{K}} := \emptyset$ 
3336     }
3337     {4} {
3338       Eliminiere-diejenigen-Schemata- $\mathcal{R}_{\alpha}$ ,
3339       die-in-einem-anderen-Relationenschema- $\mathcal{R}_{\alpha'}$ -enthalten-sind,-d.h.-
3340        $\mathcal{R}_{\alpha} \subseteq \mathcal{R}_{\alpha'}$ .
3341     }
3342   }
3343 }

```

```

3346     }
3347   }
3348 }
3349 \def\bSyntheseErklaerung#1
3350 {
3351   {
3352     \itshape
3353     \footnotesize
3354     \bParagraphMitLinien{\li@synthese@erklaerung@texte{#1}}
3355   }
3356 }

```

\bSyntheseUeberErklaerung **Makro-Faulenzer:** \let\schrittE=\liSyntheseUeberErklaerung

```

3357 \def\bSyntheseUeberErklaerung#1
3358 {
3359   \bSyntheseUeberschrift{#1}\par
3360   \bSyntheseErklaerung{#1}
3361 }

3362 \ExplSyntaxOff
3363

```

tabelle.sty

3364 \NeedsTeXFormat{LaTeX2e}[1995/12/01]

3365 \ProvidesPackage{bschlangaul-tabelle}[2020/12/05 Lädt das Paket tabluarx]

3366 \RequirePackage{tabularx}

3367

tex-dokumentation.sty

```
3368 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3369 \ProvidesPackage{bschlangaul-tex-dokumentation}[2021/09/12 Hilfsmakros
3370 für die DTX-Dokumentation]

3371 \ExplSyntaxOn

3372 \RequirePackage{mdframed}

3373 \newenvironment{bBeispiel}
3374 {
3375   \begin{mdframed}
3376 }
3377 {
3378   \end{mdframed}
3379 }

\bMakroFaulenzer \bMakroFaulenzer{⟨Abkürzung⟩}{⟨Makroname ohne Schrägstrich⟩}

3380 \cs_new:Npn \let_abkuerzung:nn #1 #2
3381 {
3382   \texttt{
3383     {
3384       \textbackslash let
3385       \textbackslash #1
3386       =
3387       \textbackslash #2
3388     }
3389 }
3390 \def\bMakroFaulenzer #1 #2
3391 {
3392   \par
3393   \noindent
3394   \textbf{Makro-Faulenzer:~}
3395   \let_abkuerzung:nn {#1}{#2}
3396   \par
3397 }

3398 \prop_new:N \l_faulenzer_prop

\bFaulenzer

3399 \def\bFaulenzer#1
3400 {
3401   \prop_clear:N \l_faulenzer_prop
3402   \prop_put_from_keyval:Nn \l_faulenzer_prop {#1}
3403   \subsubsection{Faulenzer}
3404   \prop_map_inline:Nn \l_faulenzer_prop
3405   {
3406     \noindent
3407     \let_abkuerzung:nn {##1} {##2}
3408     \par
3409   }
3410   \bigskip
3411 }

3412 \ExplSyntaxOff

3413
```

typographie.sty

```

3414 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3415 \ProvidesPackage{bschlangaul-typographie}[2021/03/20 Typographische Makros,
3416 die das Erscheinungsbild verändern. Die Schriftdefinition sind in
3417 formatierung.sty definiert.]

```

```

3418 \ExplSyntaxOn

```

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

```

3419 \RequirePackage{fontawesome}

```

`\bErledigt` `\bErledigt:` ☑

```

3420 \let\bErledigt=\faCheckSquareO

```

`\bNichtsZuTun` `\bNichtsZuTun:` ∅ Nichts zu tun

```

3421 \def\bNichtsZuTun{\emptyset~Nichts~zu~tun}

```

`\bParagraphMitLinien` `\bParagraphMitLinien{Lorem ipsum...}:`

— Lorem ipsum dolor sit, ipsum dolor sit, ipsum dolor sit, ipsum dolor sit, ipsum dolor sit, ipsum dolor sit —

```

3422 \def\bParagraphMitLinien#1
3423 {
3424   \noindent
3425   \vrule height 2pt depth -1.6pt width 0.4cm
3426   \enspace
3427   #1
3428   \enspace
3429   \leaders\vrule height 2pt depth -1.6pt \hfill \null
3430   \par
3431   \medskip
3432 }

```

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

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

```

3433 \def\bGeschweifteKlammern#1#2#3#4
3434 {
3435   \par
3436   \medskip
3437   \noindent
3438   #1 \, $= \Bigl\{ \$
3439   \vspace{#3}
3440   #2
3441   \vspace{#4}
3442   \begin{flushright} \$\Bigr\} \$\end{flushright}
3443   \par
3444 }

```

```

3445 \ExplSyntaxOff

```

```

3446

```

uml.sty

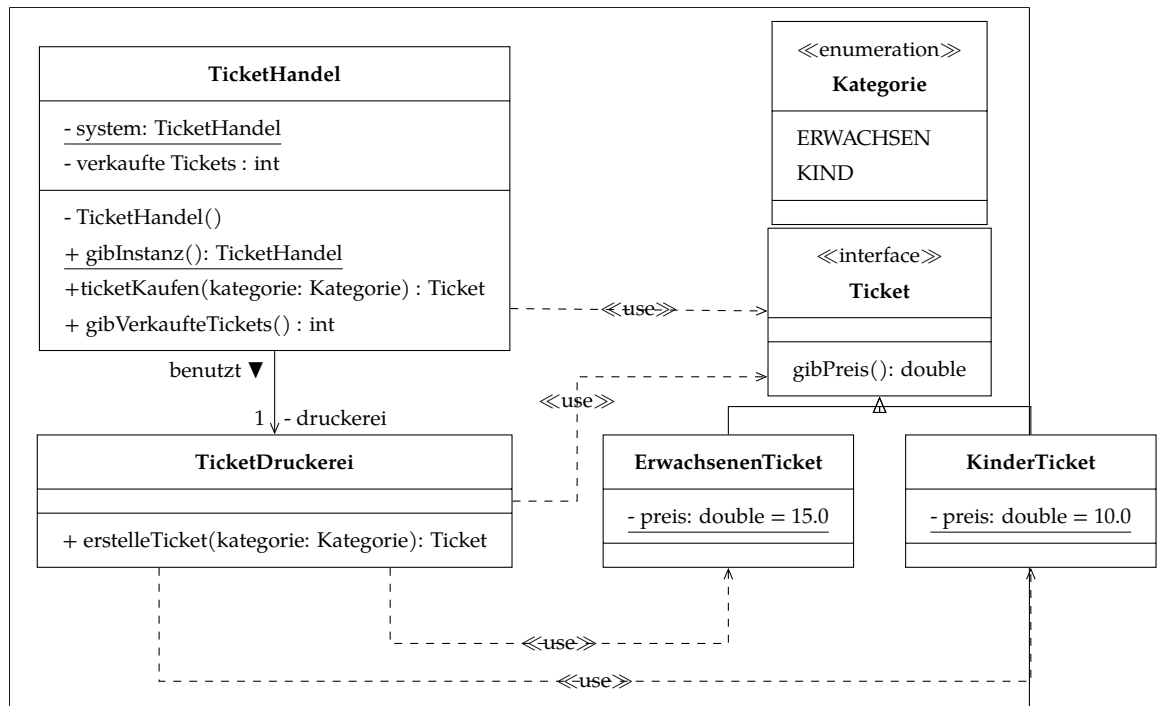
```

3447 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3448 \ProvidesPackage{bschlangaul-uml}[2020/06/13 Ein Hüll-Paket um
3449 `tikz-uml', das einige Design-Einstellungen vornimmt und manche
3450 Erweiterung bereitstellt]
3451 \RequirePackage{tikz-uml}
3452 \RequirePackage{tikz-uml-activity}

    Not compatible with wasysym
3453 %\RequirePackage{mathabx}
3454 \RequirePackage{wasysym}
3455 \usetikzlibrary{positioning}

3456 \tikzumlset{
3457   fill class=white!0,
3458   font=\footnotesize,
3459   fill object=white!0,
3460   fill note=white!0,
3461   fill state=white!0,
3462   % Use case
3463   fill usecase=white!0,
3464   fill system=white!0,
3465 }

```



```

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

```

```

3466 \NewDocumentCommand{ \bUmlLeserichtung } { 0{dir=right} m }
3467 {
3468   \def\@liDirLeft{
3469     \def\@liDirRight{ \RIGHTarrow}
3470     \pgfkeys{/lese/dir/.is choice}
3471     \pgfkeys{/lese/dir/up/.code={\def\@liDirRight{ \UParrow}}}
3472     \pgfkeys{/lese/dir/down/.code={\def\@liDirRight{ \DOWNarrow}}}
3473     \pgfkeys{/lese/dir/left/.code={\def\@liDirRight{}\def\@liDirLeft{ \LEFTarrow }}}
3474     \pgfkeys{/lese/dir/right/.code={\def\@liDirRight{ \RIGHTarrow}}}
3475
3476     \def\@liPos{above}

```

```

3477 \pgfkeys{/lese/pos/.code={\def\@liPos{##1}}}
3478
3479 \def\@liDistance{0cm}
3480 \pgfkeys{/lese/distance/.code={\def\@liDistance{##1}}}
3481
3482 \pgfkeys{/lese/.cd,#1}
3483
3484 \node[\@liPos = \@liDistance of #2-middle] {
3485   \@liDirLeft{\footnotesize#2}\@liDirRight
3486 };
3487 }
3488

```


vollstaendige-induktion.sty

```
3489 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3490 \ProvidesPackage{bschlangaul-vollstaendige-induktion}[2021/07/01
3491 Hilfsmakros zum Setzen der Vollständigen Induktion, vor allem die
3492 Überschriften für die einzelnen Schritte]
```

Faulenzer

```
\let\m=\bInduktionMarkierung
\let\e=\bInduktionErklaerung
```

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

Lade häufig benötigte Pakete

```
3493 \bLadePakete{
3494   typographie,
3495   mathe,
3496   syntax
3497 }
3498 \ExplSyntaxOn
```

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

Makro-Faulenzer: `\let\m=\liInduktionMarkierung`

```
3499 \def\bInduktionMarkierung#1
3500 {
```

```

3501 \textcolor{violet}{#1}
3502 }

```

`\bInduktionErklaerung` Gedacht für die rechte Spalte in der align-Umgebung. Das text-Makro ist dann nicht mehr nötig.

Makro-Faulenzer: `\let\le=\liInduktionErklaerung`

```

3503 \def\bInduktionErklaerung#1
3504 {
3505   \scriptsize\text{#1}
3506 }

```

`\bInduktionAnfang`

```

3507 \def\bInduktionAnfang{
3508   \bPseudoUeberschrift{Induktionsanfang}
3509
3510   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
3511   \bParagraphMitLinien{
3512     Beweise,~dass~$A(1)$~eine~wahre~Aussage~ist.
3513   }
3514 }

```

`\bInduktionVoraussetzung`

```

3515 \def\bInduktionVoraussetzung{
3516   \bPseudoUeberschrift{Induktionsvoraussetzung}
3517
3518   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
3519   \bParagraphMitLinien{
3520     Die~Aussage~$A(k)$~ist~wahr~für~ein~beliebiges~$k$ \in \mathbb{N}$.
3521   }
3522 }

```

`\bInduktionSchritt`

```

3523 \def\bInduktionSchritt{
3524   \bPseudoUeberschrift{Induktionsschritt}
3525
3526   % https://de.wikibooks.org/wiki/Mathe_für_Nicht-Freaks:_Vollständige_Induktion
3527   \bParagraphMitLinien{
3528     Beweise,~dass~wenn~$A(n=k)$~wahr~ist,~
3529     auch~$A(n=k+1)$~wahr~sein~muss.
3530   }
3531 }

```

```

3532 \ExplSyntaxOff

```

```

3533

```

wasserfall.sty

```
3534 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3535 \ProvidesPackage{bschlangaul-wasserfall}[2020/06/10 Zu Setzen des
3536 Diagramms zum Wasserfallmodell]
```

Latex-Markup-Beispiel

```
\begin{tikzpicture}[b wasserfall]
\node {Systemanforderung}; % A-1
\node {Softwareanforderung};
\node {Analyse};
\node {Entwurf};
\node {Implementierung};
\node {Test};
\node {Betrieb};

\foreach \i [count=\j] in {2,...,7}
{
  \draw[->, thick] (A-\i) -| (A-\j);
  \draw[->, thick] (A-\j) -| (A-\i);
}
\end{tikzpicture}

3537 \RequirePackage{tikz}
3538 \usetikzlibrary{chains,positioning,shadows}
3539 \tikzset{
3540   b wasserfall/.style={
3541     >=stealth,
3542     node distance = 2mm and -8mm,
3543     start chain = A going below right,
3544     every node/.style = {
3545       draw,
3546       text width=24mm,
3547       minimum height=12mm,
3548       align=center,
3549       inner sep=1mm,
3550       fill=white,
3551       drop shadow={fill=black},
3552       on chain=A
3553     },
3554   }
3555 }

3556
```

wpkalkuel.sty

```
3557 \NeedsTeXFormat{LaTeX2e}[1995/12/01]
3558 \ProvidesPackage{bschlangaul-wpkalkuel}[2020/06/13 Hilfsmakros zum
3559 Setzen des WP-Kalküls]
```

Faulenzer

```
\let\wp=\bWpKalkuel
\let\equivalent=\bWpEquivalent
\let\erklaerung=\bWpErklaerung
```

```
3560 \RequirePackage{amsmath}
```

```
3561 \ExplSyntaxOn
```

\bWpKalkuel Makro-Faulenzer: \let\wp=\liWpKalkuel

```
3562 \def\bWpKalkuelOhneMathe#1#2
3563 {
3564   \text{wp}(\texttt{\scriptsize"#1"},\thinspace #2)
3565 }
3566 \def\bWpKalkuel#1#2{
3567   \ifmmode
3568     \bWpKalkuelOhneMathe{#1}{#2}
3569   \else
3570     $\bWpKalkuelOhneMathe{#1}{#2}$
3571   \fi
3572 }

3573 \cs_new:Npn \mathe_umgebung:n #1
3574 {
3575   \medskip
3576
3577   \hspace{1em}#1
3578
3579   \medskip
3580 }
```

\bWpPseudoMatheUmgebung

```
3581 \def\bWpPseudoMatheUmgebung#1
3582 {
3583   \mathe_umgebung:n{#1$}
3584 }
```

\bWpEquivalent Makro-Faulenzer: \let\equivalent=\liWpEquivalent

```
3585 \def\bWpEquivalent#1
3586 {
3587   \mathe_umgebung:n {
3588     $\equiv$
3589     \hspace{1em}
3590     $#1$
3591   }
3592 }
```

\bWpErklaerung Makro-Faulenzer: \let\erklaerung=\liWpErklaerung

```
3593 \newlength{\@Skip@Erklaerung@Reset}
3594 \def\bWpErklaerung#1
3595 {
3596   \setlength{\@Skip@Erklaerung@Reset}{\leftskip}
3597   \setlength{\leftskip}{0.5cm}
3598
3599   \par
3600   \noindent
3601   {
```

```

3602 \scriptsize
3603 #1
3604 }
3605 \par
3606
3607 \setlength{\leftskip}{\@Skip@Erklaerung@Reset}
3608 }

```

\bWpErklaerungVerzweigung

```

3609 \def\bWpErklaerungVerzweigung
3610 {
3611 $ \bWpKalkuelOhneMathe{if~\{~b~\}~then~\{~a1~\}~else~\{~a2~\}}{Q}
3612 \equiv
3613 (b \land \bWpKalkuelOhneMathe{a1}{Q})
3614 \lor
3615 (\neg b \land \bWpKalkuelOhneMathe{a2}{Q})$
3616 }

3617 \ExplSyntaxOff
3618

```

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	_ . 47, 61, 101, 242, 400,	B
\# 477	408, 413, 415, 673,	\bAbleitung 1772
\, .. 727, 797, 852, 1148,	702, 726, 729, 739,	bAdditum (environment) 2285
1166, 1798, 1800,	762, 765, 769, 771,	\bAlphabet 1757
2380, 2381, 2382,	773, 775, 778, 790,	bAntwort (environment) 2241
2833, 3170, 3177, 3438	791, 792, 795, 797,	\bAnweisung 2122
\@Skip@Erklaerung@Reset	811, 812, 818, 821,	\bAssemblerCode 3225
.... 3593, 3596, 3607	824, 834, 841, 843, 846	\bAssemblerDatei ... 3229
\@afterheading 912		\bAttribut 3034
\@afterindentfalse .. 911	_ 2785, 2795	\bAttributHuelle
\@liDirLeft 3468, 3473, 3485	 2765, 3305, 3310
\@liDirRight 3469, 3471,	A	\bAttributHuelleOhneMathe
3472, 3473, 3474, 3485	\addbibresource 2765, 2768,
\@liDistance 2190, 2191, 2192,	2770, 2784, 2794, 2802
.... 3479, 3480, 3484	2193, 2194, 2195,	\bAttributMenge
\@liPos ... 3476, 3477, 3484	2196, 2197, 2198, 2199	. 2773, 2785, 2788,
\\ ... 155, 157, 181, 185,	\ADDITUMfalse 591	2795, 2796, 2810, 2812
189, 1178, 1206,	\ADDITUMtrue 16, 591	\bAufgabe 379
1207, 1210, 1211,	\AddToHook . 81, 85, 284, 324	\bAufgabenMetadaten .
1214, 1215, 1310,	\advance 3080 38, 240, 344, 410
1311, 1312, 1419,	\AfterEndEnvironment 3156	\bAufgabenTitel 418
1453, 1455, 1485,	\Alpha 1858	\bAusdruck 1794
1494, 1539, 1581,	\alpha 1858, 1859	\bAutomat 429
1582, 1583, 1588,	\alpha . 3302, 3304, 3305,	\bAutomatenKante 461
1589, 1590, 1610,	3308, 3310, 3311,	\bBandAlphabet 1758
2227, 2728, 2786, 2789	3312, 3313, 3314,	bBaum (environment) .. 980
\{ 575, 1735,	3318, 3324, 3325,	\bBedingung 2123
1745, 1757, 1758,	3330, 3331, 3332,	\bBedingungDrei
1763, 1797, 2037,	3335, 3343, 3344, 3345 2445, 2488, 2528
2773, 3031, 3438, 3611	\ANTWORTfalse 595	\bBedingungEins
\} 575, 1735,	\ANTWORTtrue 595 2439, 2477, 2520
1745, 1757, 1758,	\arabic 1858	\bBedingungFalsch .. 2125
1765, 1801, 2038,	\arraystretch 2723	\bBedingungWahr 2124
2773, 3031, 3442, 3611	\AtBeginDocument 178	

\bBedingungZwei	\bEntwurfsAdapterUml	\bEntwurfsZustandCode
. 2442, 2483, 2524 1257, 1303 1681, 1688
\bBindeAufgabeEin . . . 223	\bEntwurfsBeobachter 1373	\bEntwurfsZustandUml
\bBindePdfEin 177	\bEntwurfsBeobachterAkteure 1648, 1686
\bChomskyErklaerung 1331, 1375	\bEpsilon 1743
. 1017, 1058	\bEntwurfsBeobachterCode	\bErAttribute
\bChomskyUeberErklaerung 1365, 1376 1701, 1715, 1717
. 1056	\bEntwurfsBeobachterUml	\bErDatenbankName . . 1720
\bChomskyUeberschrift 1307, 1374	\bErEntity 1699, 1703, 1705
. 1005, 1057	\bEntwurfsDekorierer 1405	\bErledigt 3420
\bCpmEreignis 1066	\bEntwurfsDekoriererAkteure	\bErMpAttribute 1714
\bCpmFruehErklaerung 1137 1407	\bErMpEntity 1702
\bCpmFruehI . . . 1130, 1150	\bEntwurfsDekoriererCode	\bErMpRelationship . . 1708
\bCpmSpaetErklaerung 1155 1397, 1408	\bErRelationship
\bCpmSpaetI . . . 1123, 1168	\bEntwurfsDekoriererUml 1700, 1709, 1711
\bCpmVon 1106 1378, 1406	\bbeschreibung
\bCpmVonOhneMathe	\bEntwurfsEinfacheFabrik 2251, 2255,
. 1106, 1109, 1111 1441	2259, 2263, 2267, 2269
\bCpmVonZu 1098	\bEntwurfsEinfacheFabrikAkteure	\bEta 3302,
\bCpmVonZuOhneMathe 1425, 1443	3305, 3309, 3310,
. 1098, 1101, 1103	\bEntwurfsEinfacheFabrikUml	3311, 3314, 3324,
\bCpmVorgang 1083 1410, 1442	3325, 3326, 3331, 3332
\bCpmZu 1114	\bEntwurfsEinzelstueck	\bExamensAufgabe . . . 382
\bCpmZuOhneMathe 1471	\bExamensAufgabeA . . . 391
. 1114, 1117, 1119	\bEntwurfsEinzelstueckAkteure	\bExamensAufgabeTA . . 388
\bBeforeBeginEnvironment 1460, 1476	\bExamensAufgabeTTA . . 385
. 3155	\bEntwurfsEinzelstueckBeschreibung	\bExams (environment) 2310
\bbegin . 95, 153, 170, 864, 1445, 1472	\bFalsch 998
873, 881, 987, 1204,	\bEntwurfsEinzelstueckCode	\bFaulenzer 3399
1258, 1273, 1308, 1468, 1478	\bFlaci 1804
1332, 1379, 1411,	\bEntwurfsEinzelstueckUml	\bFremd 3014
1426, 1451, 1461, 1450, 1474	\bFseries . 54, 144, 209,
1481, 1501, 1533,	\bEntwurfsErbauer . . 1528	249, 865, 888, 1007,
1550, 1579, 1604,	\bEntwurfsErbauerAkteure	1848, 1850, 2643, 3278
1625, 1649, 1664, 1500, 1530	\bFunktionaleAbhaengigkeit
1778, 1863, 1955,	\bEntwurfsErbauerUml 2804, 2807, 2815
1991, 2000, 2007, 1480, 1529	\bFunktionaleAbhaengigkeiten
2115, 2213, 2271,	\bEntwurfsFabrikmethode 2818
2293, 2314, 2345, 1574, 1599	\bFussnote 2678, 2680
2350, 2450, 2454,	\bEntwurfsFabrikmethodeAkteure	\bFussnoteDreiText
2473, 2496, 2519, 1549, 1576 2692, 2710
2534, 2625, 2626,	\bEntwurfsFabrikmethodeUml	\bFussnoteEinsText
2630, 2653, 2666, 1532, 1575 2684, 2704
2726, 2727, 2777,	\bEntwurfsKompositum 1599	\bFussnoteLink 2369
2822, 2962, 2982,	\bEntwurfsKompositumAkteure	\bFussnoten 2700
3155, 3185, 3375, 3442 1601	\bFussnoteUrl . . 1618, 2357
\bbegingroup 2214, 2775, 3076	\bEntwurfsKompositumUml	\bFussnoteVierText
\bEmph 2220 1578, 1600 2696, 2713
\bEntwurfs 1620	\bEntwurfsModellPraesentation	\bFussnoteZweiText
\bEntwurfsAbstrakteFabrik 1620 2688, 2707
. 1250	\bEntwurfsModellPraesentationAkteure	\bFussnoteZweiText
\bEntwurfsAbstrakteFabrikBeschreibung 1622 1776, 2819, 3433
. 1198, 1251	\bEntwurfsModellPraesentationUml	\bFussnoteZweiText
\bEntwurfsAbstrakteFabrikCode 1603, 1621 1895, 2663
. 1245, 1255	\bEntwurfsStellvertreter	\bGrafikLogo 882, 1900, 2639
\bEntwurfsAbstrakteFabrikUml 1644	\bGrafikLogoPfad
. 1203, 1253	\bEntwurfsStellvertreterCode 1892, 1897, 1902
\bEntwurfsAdapter . . 1302 1638, 1646	\bGrammatik 1815
\bEntwurfsAdapterAkteure 1624, 1645	\bgroup 2246, 2290
. 1272, 1304	\bEntwurfsZustand . . 1685	\bHanoi 1941
\bEntwurfsAdapterCode	\bEntwurfsZustandAkteure	\bHaskellCode 3237
. 1296, 1305 1663, 1687	\bHaskellDatei 3241
		\bIgl 3438
		\bIgr 3442

`\bigskip` 76, 90, 93, 295, 298, 830, 1180, 1185, 2229, 2701, 3410
`\bInduktionAnfang` .. 3507
`\bInduktionErklaerung` 3503
`\bInduktionMarkierung` 3499
`\bInduktionSchritt` . 3523
`\bInduktionVoraussetzung` 3515
`bJavaAngabe` (environment) 3164
`\bJavaCode` 3168
`\bJavaDatei` ... 1196, 3191
`\bJavaExamen` 3213
`\bJavaTestDatei` 3202
`\bKellerAutomat` 470
`\bKellerKante` 508
`\bKellerUebergang` 503, 509
`\bKontrollCode` 2126
`bKontrollflussgraph` (environment) . 2114
`\bKontrollKnotenPfad` 2128
`\bKontrollTextzeileKnoten` 2127, 2132
`\bKurzeTabellenLinie` 1178
`\bLadeAllePakete` 212, 320, 601
`\bLadePakete` ... 5, 113, 120, 132, 354, 422, 425, 597, 602, 1004, 1065, 1734, 1986, 2565, 2677, 2757, 2925, 3127, 3269, 3493
`\bLatexCode` 3179
`\bLeereZelle` 2715
`\bLinksReduktion` ... 2782
`\bLinksReduktionInline` 2791, 2799
`\bLogoTextCCLizenz` .. 92, 297, 2659
`\bLogoTextProjekt` ... 89, 294, 2635
`\bMakroFaulenzer` ... 3380
`\bMasterExkurs` 2533
`\bMasterFaelle` . 2472, 2540
`\bMasterFallRechnung` 2518
`\bMasterVariablen` ... 2449, 2535
`\bMasterVariablenDeklaration` 2495
`\bMasterWolframLink` 2543
`\bMenge` .. 439, 440, 442, 481, 482, 483, 487, 539, 540, 541, 545, 1735, 1784, 1823, 1824
`\bMengeOhneMathe` 1735, 1738, 1740
`\bMetaBsclangaulSammlung` . 890, 2152, 2566, 2644
`\bMetaCCLink` ... 2581, 2668
`\bMetaEmail` 2158, 2593, 2601, 2603
`\bMetaEmailLink` 2597, 2616
`\bMetaHermineFriends` . 898, 2157, 2570, 2648
`\bMetaHilfMit` 98, 2606, 2618
`\bMetaQuelltext` . 100, 2618
`\bMetaSetze` 40, 241, 345, 399, 411
`\bMetaUeberDasProjekt` 2574, 2655
`\bMinimierungErklaerung` 2737
`\bMinispracheDatei` . 3233
`\bNichtsZuTun` 3421
`\b0` 2412, 2440
`\bOmega` 2402, 2446
`\bOmegaOhneMathe` 2402, 2407, 2409
`\b0Notation0` 2845
`\b0OhneMathe` 2412, 2417, 2419
`\bool` 705, 798
`\bowtie` 3001, 3004, 3005, 3006
`\Box` 527
`\boxtimes` 997
`\bParagraphMitLinien` . 1053, 1138, 1156, 2739, 3354, 3422, 3511, 3519, 3527
`\bPetriErreichKnotenDrei` 2919
`\bPetriErreichTransition` 2916
`\bPetriSetzeSchluessel` 2861
`\bPetriTransitionsName` 2908, 2920
`\bPetriTransitionsNameOhneMathe` 2908, 2911, 2913
`\bPetriTransPfeile` . 2920
`\bPolynomiellReduzierbar` 2006
`\bPotenzmenge` 1744, 1748, 2931
`\bPotenzmengeOhneMathe` 1745, 1746, 1747
`\bPrimaer` 3013
`\bProblemBeschreibung` 1990
`\bProblemClique` 2013
`\bProblemName` 1989, 1996, 2008, 2010, 2023, 2034, 2035, 2043, 2044
`\bProblemSat` 2042
`\bProblemSubsetSum` . 2033, 2042
`\bProblemVertexCover` . 2013, 2021
`\bProduktionen` . 1783, 1825
`bProduktionsRegeln` (environment) 1773
`\bPruefungsNummer` ... 135
`\bPruefungsTitel` 138
`\bPseudoUeberschrift` . 2228, 2296, 2299, 2725, 2735, 3508, 3516, 3524
`\bPumpingKontextfrei` 2977
`\bPumpingRegulaer` .. 2956
`bQuellen` (environment) 2336
`\bRechtsReduktionInline` 2799
`\bRekursionsGleichung` 2436, 2499
`\bRelation` 2829
`\bRelationMenge` 3029
`\bRichtig` 997
`\bRundeKlammer` 2389, 2393, 2403, 2413, 2427, 2842, 2846
`\bSetzeExamenTeilaufgabeNr` 219
`\bSetzeExamenThemaNr` . 215
`\bSortierMarkierung` 3054
`\bSortierPfeil` 3048
`\bSortierPfeilUnten` 3051
`\bSpaltenUmbruch` ... 3123
`\bSqlCode` 3245
`\bStrich` 1988
`\bSyntheseErklaerung` . 3292, 3360
`\bSyntheseUeberErklaerung` 3357
`\bSyntheseUeberschrift` 3275, 3359
`\bT` .. 2422, 2437, 2451, 2515
`\bTeilen` 2762
`\bTheta` 2392, 2443, 2475, 2481, 2486
`\bThetaOhneMathe` 2392, 2397, 2399
`\bTitelSeite` 166, 180
`\bTOhneMathe` 2422, 2431, 2433
`\bTrennSeite` 149
`\bTuringKante` 571
`\bTuringLeerzeichen` . 527, 535
`\bTuringMaschine` 528
`\bTuringUeberfuehrung` 574
`\bTuringUebergange` . 566, 572
`\bTuringUebergangZelle` 561
`\bUeberfuehrungsFunktion` 1749
`\bUeberfuehrungsFunktionOhneMathe` 1749, 1752, 1754
`\bUeberschriftDreiecksTabelle` 2733
`\bUmlLeserichtung` .. 3466
`\bVertauschen` 3042
`\bWortInSprache` 1179
`\bWortNichtInSprache` 1184
`\bWpEquivalent` 3585
`\bWpErklaerung` 3593
`\bWpErklaerungVerzweigung` 3609
`\bWpKalkuel` 3562

<code>\bWpKalkuelOhneMathe</code> 3562, 3568, 3570, 3611, 3613, 3615	D	<code>bAntwort</code> 2241
<code>\bWpPseudoMatheUmgebung</code> 3581	<code>\DeclareMathSymbol</code> 2558, 2559	<code>bBaum</code> 980
<code>\bZustandsBuchstabe</code> 1759, 1768, 1770, 1788, 1790	<code>\definecolor</code> 1846	<code>bExkurs</code> 2310
<code>\bZustandsBuchstabeGross</code> 1760, 1769, 1771	<code>\DefineVerbatimEnvironment</code> 3129	<code>bJavaAngabe</code> 3164
<code>\bZustandsmenge</code> 1747	<code>\delta</code> 433, 475, 533, 575, 1749	<code>bKontrollflussgraph</code> 2114
<code>\bZustandsmengeNr</code> 1761, 2940	<code>\dh</code> 2382, 3313	<code>bProduktionsRegeln</code> 1773
<code>\bZustandsmengeNrGross</code> 1769	<code>\directlua</code> 426, 505, 563, 568, 1730, 1744, 1764, 1772, 1779, 1784, 2758, 2763, 2816, 2823, 2830, 3043, 3142, 3194, 3199, 3200, 3205, 3210, 3211, 3216, 3222, 3223	<code>bQuellen</code> 2336
<code>\bZustandsMengenSammlung</code> 2927	<code>\dots</code> 1038, 1042, 2037, 2969, 3324, 3325	<code>liAHuelle</code> 2774
<code>\bZustandsMengenSammlungNr</code> 2936	<code>\DOWNarrow</code> 3472	<code>liEinbettung</code> 2240
<code>\bZustandsmengeOhneMathe</code> 1747	<code>\draw</code> 1962, 1965, 1968, 2917, 3049, 3052	<code>liGraphenFormat</code> . 1934
<code>\bZustandsname</code> 1770	E	<code>liKasten</code> 1862
<code>\bZustandsnameGross</code> 1771, 2928, 2937	<code>\edef</code> 1953	<code>liProjektSprache</code> 2239
<code>\bZustandsnameTiefgestellt</code> 1786	<code>\edge</code> 644	<code>liRelationenSchemaFormat</code> 3035
<code>\bZustandsPaar</code> 2717	<code>\egroup</code> 2282, 2307	<code>liRmodell</code> 3015
<code>\bZustandsPaarVariablenName</code> 2716, 2719, 2720	<code>\else</code> .. 1102, 1110, 1118, 1126, 1133, 1739, 1753, 1789, 2244, 2267, 2281, 2288, 2306, 2332, 2398, 2408, 2418, 2432, 2769, 2806, 2851, 2912, 3106, 3108, 3569	<code>liUebergangsTabelle</code> 2723
C	<code>\emph</code> .. 1199, 1446, 1701, 2018, 2047, 2049, 2222	<code>\equiv</code> 3588, 3612
<code>\c</code> 1957, 1958	<code>\empty</code> 2267, 2805	<code>\erzeuge@tiefgestellt</code> 1744, 1745, 1749
<code>\cdot</code> 2425, 2481, 2492	<code>\emptyset</code> 2715, 3319, 3340, 3421	<code>\EXKURStue</code> 593
<code>\centerline</code> ... 1996, 2663	<code>\end</code> .. 102, 161, 172, 869, 877, 901, 990, 1243, 1269, 1294, 1329, 1363, 1395, 1423, 1439, 1458, 1466, 1497, 1526, 1547, 1572, 1597, 1617, 1636, 1661, 1679, 1780, 1865, 1971, 2003, 2004, 2011, 2120, 2218, 2279, 2303, 2330, 2352, 2353, 2452, 2468, 2493, 2516, 2531, 2541, 2628, 2632, 2633, 2656, 2669, 2730, 2731, 2779, 2824, 2972, 2992, 3156, 3189, 3378, 3442	<code>\expandafter</code> ... 1942, 3079
<code>\cftbeforesecskip</code> ... 275	<code>\endcsname</code> 1942, 1945	<code>\ExplSyntaxOff</code> 105, 124, 192, 302, 348, 417, 460, 502, 507, 560, 565, 570, 1060, 1082, 1097, 1173, 1725, 1803, 1838, 1905, 1980, 2135, 2174, 2383, 2448, 2672, 2751, 2945, 3028, 3089, 3249, 3362, 3412, 3445, 3532, 3617
<code>\cftbeforesubsecskip</code> 276, 277	<code>\endgroup</code> . 2217, 2780, 3081	<code>\ExplSyntaxOn</code> 17, 116, 134, 214, 322, 398, 429, 470, 503, 528, 561, 566, 590, 1003, 1066, 1083, 1122, 1698, 1793, 1815, 1889, 1977, 2128, 2140, 2211, 2387, 2564, 2733, 2838, 2926, 3016, 3086, 3141, 3274, 3371, 3418, 3498, 3561
<code>\cftbeforesubsubsecskip</code> 278	<code>\enspace</code> 3426, 3428	F
<code>\cftsubsecafterpnum</code> . 277	<code>environments:</code>	<code>\faCheckSquare0</code> 3420
<code>\chapter</code> 1848, 1849	<code>bAdditum</code> 2285	<code>\faCircleThin</code> 1717
<code>\char</code> 2227		<code>\faGg</code> 1711
<code>\clearpage</code> 150, 164, 167, 175, 210, 288, 300, 340, 907, 2215		<code>\fancyfoot</code> 1869, 1870, 1871, 2156, 2157, 2158, 2159
<code>\cline</code> 1178		<code>\fancyhead</code> . 1868, 2151, 2152, 2153, 2154, 2166
<code>\clist</code> 599, 647, 648, 670, 674, 3088		<code>\faSquare0</code> 1705
<code>\columnbreak</code> 3123		<code>\fi</code> 1104, 1112, 1120, 1128, 1135, 1741, 1755, 1791, 2247, 2270, 2283, 2291, 2308, 2333,
<code>\contentsname</code> 279		
<code>\cs</code> 18, 25, 31, 244, 272, 673, 702, 726, 729, 739, 762, 773, 784, 795, 797, 834, 841, 846, 857, 909, 2142, 2149, 2164, 2170, 2336, 2623, 2845, 3380, 3573		
<code>\csname</code> 1942, 1945		
<code>\cup</code> 1758, 2807, 3311, 3325, 3332		

2400, 2410, 2420, 2434, 2771, 2808, 2853, 2914, 3106, 3107, 3110, 3112, 3571	\geq 2044, 2458, 2958, 2963, 2979, 2983	\itshape 1051, 3352
\filcenter 209	\gib 74, 346, 784	J
\footcite 1201, 1270, 1292, 1339, 1362, 1394, 1447, 1498, 1525, 1571, 2016, 2019, 2026, 2031, 2036, 2040, 2046, 2051, 2179, 2469, 2470, 2738, 2975	H	\j 1957, 1958, 1960, 1961, 1962, 1967, 1968, 1969
\footnote 2359, 2371	\hbox 3001	K
\footnotesize 70, 515, 807, 946, 1052, 1722, 2325, 2346, 2654, 2682, 2776, 2793, 2801, 2930, 2939, 3015, 3130, 3162, 3353, 3458, 3485	\headrulewidth . 1873, 2161	\k 1967
\footrulewidth . 1874, 2162	\headwidth 2172	\keine .. 78, 909, 2237, 2355
\foreach .. 1957, 1960, 1967	\hfill 64, 2477, 2483, 2488, 3429	\keys . 117, 404, 438, 450, 480, 490, 538, 548, 678, 1070, 1074, 1088, 1093, 1822, 1829
\forestFirst ... 3103, 3106	\hline 2728	\kopfzeile 2142, 2152, 2153, 2154, 2157, 2158, 2159, 2167
\forestLast ... 3104, 3106	\horizontale 25, 72	L
\forestOget ... 3103, 3104	\href 1811, 2372, 2545, 2584, 2599, 3188	\l ... 431, 432, 433, 434, 435, 436, 439, 440, 441, 442, 443, 445, 447, 452, 453, 454, 455, 456, 457, 472, 473, 474, 475, 476, 477, 478, 481, 482, 483, 484, 485, 486, 487, 493, 494, 495, 496, 497, 498, 499, 530, 531, 532, 533, 534, 535, 536, 539, 540, 541, 542, 543, 544, 545, 551, 552, 553, 554, 555, 556, 557, 1068, 1071, 1076, 1077, 1080, 1085, 1086, 1089, 1090, 1095, 1817, 1818, 1819, 1820, 1823, 1824, 1825, 1826, 1832, 1833, 1834, 1835, 2131, 2132, 2133, 2342, 2343, 2344, 2351, 3398, 3401, 3402, 3404
\forestOones 3116	\hspace ... 2920, 3577, 3589	\labelenumi 1859
\forestOv . 3105, 3106, 3109	\ht 3002	\labelenumii 1860
\forestov .. 3095, 3099, 3100, 3103, 3104, 3105, 3106, 3108, 3109	\huge 159, 185, 249, 330	\labelitemi 1853
\forestset 3091, 3114	\huge 209	\labelitemii 1854
\forestSortLevel 3093, 3101, 3115, 3116	I	\labelitemiii 1855
\frac . 2427, 2460, 2492, 2507	\i 1967, 1968	\labelitemiv 1856
\fullouterjoin 3006	\ifADDITUM . 591, 2287, 2305	\land 3613, 3615
G	\ifANTWORT . 595, 2243, 2280	\LARGE 155, 189, 1848
\g 19, 21, 32, 34, 42, 56, 58, 136, 139, 155, 157, 185, 189, 216, 220, 224, 228, 229, 230, 231, 232, 234, 235, 237, 251, 254, 257, 262, 402, 647, 648, 670, 674, 680, 681, 682, 683, 684, 686, 687, 688, 690, 691, 693, 694, 695, 696, 697, 698, 699, 700, 703, 707, 708, 709, 712, 714, 715, 716, 717, 718, 719, 763, 767, 771, 774, 775, 777, 778, 780, 781, 785, 787, 800, 801, 802, 803, 811, 812, 814, 820, 821, 823, 824, 826, 827, 839, 842, 847, 849, 853	\ifEXKURS 593, 2312	\Large 55, 181, 866, 889
\Gamma .. 474, 532, 575, 1758	\ifmmode 1100, 1108, 1116, 1124, 1131, 1737, 1751, 1787, 2396, 2406, 2416, 2430, 2767, 2849, 2910, 3567	\large 46, 157, 874, 897, 1996, 3065
\gappto 3111	\IfNoValueTF 2294, 2361, 2373	\leaders 3429
\geometry 366	\ifnum 3078, 3094, 3102, 3108	\left 2390, 2843
	\ifx 2267, 2805, 3106	\LEFTarrow 3473
	\in 1026, 1182, 2044, 2440, 2443, 2446, 2475, 2481, 2486, 2958, 2969, 2979, 2989, 3302, 3304, 3310, 3331, 3520	\leftarrow 1114
	\includegraphics 1896, 1901	\leftouterjoin 3004
	\includepdf 177	\leftskip . 3596, 3597, 3607
	\inhalts 272, 290	\LehramtInformatikGitBranch 838, 3149
	\inhaltsverzeichnis 2212	
	\input 225, 380, 383, 386, 389, 392, 916	
	\inputminted 3193, 3204, 3215, 3231, 3235, 3243	
	\int 3088	
	\item ... 997, 998, 1275, 1279, 1284, 1289, 1333, 1342, 1347, 1355, 1427, 1432, 1436, 1462, 1502, 1507, 1514, 1522, 1551, 1556, 1560, 1565, 1665, 1670, 1675, 2001, 2002, 2338, 2343, 2455, 2460, 2464, 2474, 2480, 2485, 2497, 2501, 2505, 2509, 2513, 2520, 2524, 2528, 2963, 2966, 2969, 2983, 2986, 2989	

<code>\LehramtInformatikGithubCodeRepositorium</code>	<code>\Rmodel (environment)</code>	<code>\negthinspace ..</code>	2390, 2843
..... 3148 3015	<code>\NewDocumentCommand ..</code>	430, 471,
<code>\LehramtInformatikGithubDomain</code>	<code>\listen</code>	508, 529, 571, 597,	
..... 835, 3145	<code>\literatur</code>	1067, 1084, 1123,	
<code>\LehramtInformatikGithubRawDomain</code>	<code>\lshergangstabelle</code>	1130, 1179, 1184,	
..... 3146	(environment) .	1794, 1816, 1895,	
<code>\LehramtInformatikGithubTexRepo</code>	<code>\llap</code>	1900, 2006, 2129,	
..... 836, 3147	<code>\LoadClass</code>	2180, 2357, 2369,	
<code>\LehramtInformatikRepository</code>	129, 130, 197, 308, 353	2818, 2829, 2916,	
..... 177, 226,	<code>\log</code>	3191, 3202, 3213,	
380, 383, 386, 389,	2440,	3229, 3233, 3241, 3466	
392, 1893, 2190,	2443, 2446, 2475, 2481		
2191, 2192, 2193,	<code>\logo</code>		
2194, 2195, 2196,	2623, 2637, 2661	<code>\NewDocumentEnvironment</code>	
2197, 2198, 2199, 3144	<code>\lor</code> 980, 1773,	
<code>\leq</code>	2551	1862, 1934, 2114,	
2492, 2966, 2986		2239, 2240, 2241,	
<code>\let ...</code>	M	2285, 2310, 2340,	
1747, 1748, 2215,	<code>\makeatletter</code>	2724, 2774, 3017, 3035	
3077, 3115, 3116,	<code>\makeatother</code>	<code>\newif</code>	591, 593, 595
3380, 3395, 3407, 3420	<code>\makeindex</code>	<code>\newlength</code>	3593
<code>\li@chomsky@erklaerung@texte</code>	<code>\marginpar</code>	<code>\newminted</code>	3164
..... 1017, 1053	1704, 1710, 1716, 2223	<code>\node</code>	1080, 2122,
<code>\li@EntwurfsCode</code>	<code>\mathbb</code>	2127, 3054, 3099, 3484	
..... 1195, 1246,	2044, 2989, 3520	<code>\noexpand</code>	3099
1247, 1248, 1297,	<code>\mathbin</code>	<code>\noindent</code>	27,
1298, 1299, 1300,	3004, 3005, 3006	45, 53, 69, 97, 809,	
1366, 1367, 1368,	<code>\mathcal</code>	984, 1181, 1186,	
1369, 1370, 1371,	2413, 2846, 3331,	2232, 2268, 2326,	
1398, 1399, 1400,	3336, 3338, 3339, 3340	2328, 2347, 2537,	
1401, 1402, 1403,	<code>\mathe</code>	2703, 2706, 2709,	
1469, 1639, 1640,	3573, 3583, 3587	2712, 3030, 3393,	
1641, 1642, 1682, 1683	<code>\mathord</code>	3406, 3424, 3437, 3600	
<code>\li@EntwurfsCodeAllgemein</code>	<code>\mdfsetup</code>	<code>\nolinkurl</code>	3188
..... 1194	2252, 2256, 2260, 2264	<code>\normalsize ...</code>	1850, 3173
<code>\li@fussnote@text</code>	<code>\mdseries</code>	<code>\notin</code>	1187
2679,	<code>\medskip</code>	<code>\null</code>	3429
2685, 2689, 2693, 2697	48, 67, 894, 989,		
<code>\li@GithubLink</code>	1998, 2235, 2349,	O	
3183, 3198, 3209, 3221	2651, 3020, 3026,	<code>\o</code>	2845, 2850, 2852
<code>\li@mget ..</code>	3431, 3436, 3575, 3579	<code>\o@join</code>	3001, 3004, 3005, 3006
1944, 1948, 1968	<code>\mintinline</code>	<code>\Omega</code>	2403
<code>\li@minc</code>	3172,	<code>\omega</code>	2958, 2959, 2979, 2980
1947, 1969	3181, 3227, 3239, 3247		
<code>\li@mset</code>	<code>\mkern</code>		
1941, 1949, 1958, 1961	3004, 3005, 3006		
<code>\li@numdiscs</code>	<code>\mlq</code>		
..... 1953, 1962, 1968	2556, 2558		
<code>\li@Rmodell@Schrift ..</code>	<code>\mrq</code>		
..... 3015, 3024, 3034	2556, 2559		
<code>\li@SansFett ..</code>	<code>\msg</code>		
141, 155,	914		
157, 159, 181, 185, 189	<code>\myList</code>		
<code>\li@sequence ...</code>	3095, 3096, 3097, 3100		
1954, 1967	<code>\myNodes</code>		
<code>\li@synthese@erklaerung@texte</code>	3084,		
..... 3292, 3354	3099, 3105, 3109, 3111		
<code>liAHuelle (environment)</code>			
..... 2774	N		
<code>liEinbettung (environ-</code>	<code>\NeedsTeXFormat</code>		
<code>ment)</code>	1, 107,		
2240	126, 194, 304, 350,		
<code>liGraphenFormat (envi-</code>	362, 375, 395, 420,		
<code>ronment)</code>	588, 920, 993, 1000,		
1934	1062, 1175, 1190,		
<code>liKasten (environment)</code>	1691, 1727, 1840,		
1862	1877, 1886, 1907,		
<code>\linespread</code>	1936, 1974, 1982,		
3022	2054, 2137, 2176,		
<code>liProjektSprache (envi-</code>	2182, 2204, 2385,		
<code>ronment)</code>	2548, 2561, 2674,		
2239	2753, 2836, 2856,		
<code>liRelationenSchemaFormat</code>	2922, 2947, 2952,		
(environment) .	2995, 3008, 3037,		
3035	3118, 3125, 3132,		
	3137, 3251, 3265,		
	3364, 3368, 3414,		
	3447, 3489, 3534, 3557		
	<code>\neg</code>		
	3615		
		<code>\pagestyle ..</code>	342, 859, 1872
		<code>\par</code>	22, 26,
		29, 48, 65, 252, 255,	
		258, 263, 268, 808,	
		829, 868, 876, 883,	
		892, 900, 910, 1057,	
		1805, 1813, 2231,	
		2327, 2646, 2649,	
		3032, 3359, 3392,	
		3396, 3408, 3430,	
		3435, 3443, 3599, 3605	
		<code>\paragraph</code>	1850
		<code>\parindent</code>	3023
		<code>\path ...</code>	462, 509, 572, 1095
		<code>\pgfkeys ...</code>	2872, 3470,
		3471, 3472, 3473,	
		3474, 3477, 3480, 3482	
		<code>\pgfmath@count</code>	
	 3076, 3078, 3080	
		<code>\pgfmath@smuggleone</code>	3081

<code>\tikzset</code>	464, 511, 577, 926, 952, 1913, 2058, 2898, 3061, 3255, 3539
<code>\tikzumlset</code>	3456
<code>\times</code>	575
<code>\tiny</code>	96, 1705, 1711, 1717, 2126, 2224, 3186
<code>\titel</code>	244, 246, 286, 327, 857
<code>\titleformat</code>	209, 1848, 1850
<code>\titlespacing</code>	1849
<code>\tl</code>	19, 32, 42, 56, 216, 220, 224, 231, 234, 402, 431, 432, 433, 434, 435, 436, 439, 440, 441, 442, 443, 445, 447, 472, 473, 474, 475, 476, 477, 478, 481, 482, 483, 484, 485, 486, 487, 530, 531, 532, 533, 534, 535, 536, 539, 540, 541, 542, 543, 544, 545, 671, 675, 703, 707, 708, 709, 712, 717, 718, 719, 731, 740, 774, 777, 780, 785, 800, 801, 802, 803, 814, 820, 823, 826, 842, 849, 1068, 1071, 1076, 1077, 1085, 1086, 1089, 1090, 1796, 1817, 1818, 1819, 1820, 1823, 1824, 1825, 1826, 2423
<code>\tmp</code>	2805
<code>\TmpPlaceEight</code>	2880
<code>\TmpPlaceFive</code>	2877
<code>\TmpPlaceFour</code>	2876
<code>\TmpPlaceNine</code>	2881
<code>\TmpPlaceOne</code>	2873
<code>\TmpPlaceSeven</code>	2879
<code>\TmpPlaceSix</code>	2878
<code>\TmpPlaceTen</code>	2882
<code>\TmpPlaceThree</code>	2875
<code>\TmpPlaceTwo</code>	2874
<code>\TmpScale</code>	2893
<code>\TmpTransitionEight</code>	2869, 2890
<code>\TmpTransitionFive</code>	2866, 2887
<code>\TmpTransitionFour</code>	2865, 2886
<code>\TmpTransitionNine</code>	2870, 2891
<code>\TmpTransitionOne</code>	2862, 2883
<code>\TmpTransitionSeven</code>	2868, 2889
<code>\TmpTransitionSix</code>	2867, 2888
<code>\TmpTransitionTen</code>	2871, 2892
<code>\TmpTransitionThree</code>	2864, 2885
<code>\TmpTransitionTwo</code>	2863, 2884
<code>\TmpX</code>	2894
<code>\TmpY</code>	2895
<code>\ttfamily</code>	3015
U	
<code>\ul</code>	1700, 3013, 3014
<code>\umlagggreg</code>	1658
<code>\umlassoc</code>	1616
<code>\umlclass</code>	1205, 1209, 1213, 1260, 1261, 1262, 1309, 1314, 1319, 1322, 1380, 1381, 1382, 1387, 1388, 1417, 1452, 1483, 1484, 1487, 1538, 1541, 1580, 1586, 1587, 1605, 1606, 1607, 1628, 1629, 1630, 1650, 1651, 1652, 1653
<code>\umldep</code>	1546
<code>\umlHVHagggreg</code>	1327, 1393, 1596
<code>\umlinherit</code>	1266, 1317, 1491, 1536, 1544
<code>\umlnote</code>	1268, 1493, 1660
<code>\umlreal</code>	1264, 1325
<code>\umlsimpleclass</code>	1220, 1221,
<code>\umlstatic</code>	1419, 1453
<code>\umluniagggreg</code>	1489
<code>\umluniassoc</code>	1240, 1265, 1490, 1614, 1615, 1634, 1635
<code>\umlVHuniassoc</code>	1241, 1242
<code>\umlVHVdep</code>	1234, 1235, 1237, 1238, 1421, 1422
<code>\umlVHVinherit</code>	1217, 1218, 1223, 1224, 1231, 1232, 1390, 1391, 1415, 1416, 1594, 1595, 1632, 1633
<code>\umlVHVreal</code>	1384, 1385, 1655, 1656
<code>\UParrow</code>	3471
<code>\url</code>	843, 2360
<code>\usemintedstyle</code>	3154
<code>\usetikzlibrary</code>	424, 925, 1695, 1912, 2057, 2859, 3000, 3041, 3072, 3455, 3538
V	
<code>\varepsilon</code>	1010, 1021, 1022, 1743, 2440, 2446, 2478, 2489
<code>\vbox</code>	2245, 2289
<code>\vfill</code>	87, 152, 162, 169, 173, 292, 871, 879, 3123
<code>\vrule</code>	3425, 3429
<code>\vspace</code>	156, 158, 183, 187, 265, 277, 862, 885, 903, 2313, 2331, 3439, 3441
X	
<code>\xappto</code>	3099, 3105, 3109
<code>\xdef</code>	1942
Z	
<code>\ZB</code>	2381
<code>\zB</code>	2380
<code>\zustandsnamens@liste</code>	1761, 1768, 1769