LaTeX, beamer, tikz und Co.

11. Quelltexte und Algorithmen

Thomas Worsch

Fakultät für Informatik
Karlsruher Institut für Technologie

Wintersemester 2016/2017

man unterscheide den Textsatz von

- Quelltexten in einer formalen Sprache
  - `(for i=0; i<n; i++)`
- Algorithmen in Pseudocode-Notation
  - **for** $i \leftarrow 0$ **to** $n - 1$ **do**

Wörtliches

Zwischen Quelltext und Pseudocode

Pakete (nur) für Pseudocode

Überblick

Eingebautes in LaTeX

- Kommando `verb`
- Umgebung `verbatim`
- Umgebung `alltt` aus gleichnamigem Paket

Kommando \verb

- \verb⟨*char*⟩⟨*text*⟩⟨*char*⟩
  \verb*⟨*char*⟩⟨*text*⟩⟨*char*⟩
- Begrenzung vorne und hinten durch *gleiches* Zeichen ⟨*char*⟩
  - nicht *
  - nicht Leerzeichen
- Beispiele:

```
\verb|$!@#{}^ _ \verb|
```

$!@#{}^ _ \verb

```
\verb*|$!@#{}^ _ \verb|
```

$!@#{}^␣_␣\verb

Umgebung `verbatim`

- analog `\verb` ohne und mit `*`:

```
\begin{verbatim}
    ⟨Rumpf⟩
\end{verbatim}
```

```
\begin{verbatim*}
    ⟨Rumpf⟩
\end{verbatim*}
```

- Beispiel

```
\begin{verbatim}
  @~#%&*!@~$%
  \verb|$%~|
\end{verbatim}
```

```
@~#%&*!@~$%
\verb|$%~|
```

Paket und Umgebung `alltt`

- analog Umgebung `verbatim`, aber
- Zeichen `\`, `{` und `}` behalten ihre Bedeutung
- Beispiel

```
\begin{alltt}
  das ist \emph{betont}
  das nicht math mode $x^3$
  aber das \(x\sp{3}\)
  und dies ist \color{red} rot
\end{alltt}
```

```
  das ist betont
  das nicht math mode $x^3$
  aber das x^3
  und dies ist  rot
```

Paket `fancyvrb`

- viele zusätzliche Möglichkeiten
  siehe Doku
- `\VerbatimInput{`⟨*Dateiname*⟩`}`
- Beispiel

```
\VerbatimInput{imin.c}
```

```
    int i, xmin, imin;
    imin = 0;
    xmin = A[imin];
    for (i=1; i<n; i++) {
      if (A[i]<xmin) {
        imin = i;
        xmin = A[imin];
      }
    }
```

Paket `newverbs`

- für Definition von Varianten von `\verb`, die automatisch vor und/oder hinter dem `verb`-Teil LaTeX-Code einfügen
- zwei vordefinierte Beispiele
  - `\qverb`⟨*char*⟩⟨*text*⟩⟨*char*⟩
    - `\qverb|a^{12}|` : ‘a^{12}’
  - `\fverb`⟨*char*⟩⟨*text*⟩⟨*char*⟩
    - `\fverb|a^{12}|` : a^{12}

`listings`: Algorithmen

- wie wirds genutzt:
  - Kommando `\lstinline`
  - Umgebung `lstlisting`
  - Kommando `\lstinputlisting`
- was wird gemacht:
  - der Eingabetext wird geparst und alles in eine der folgenden Schubladen gepackt:
    - `keyword` explizit definiert, mehrere Typen möglich
    - `string` explizit definiert
    - `comment` explizit definiert
    - `identifier` der Rest
- fertige Definitionen für viele Programmiersprachen
- kann aber auch für Pseudocode benutzt werden:
  `http://tex.stackexchange.com/questions/31328/how-to-typeset-data-structures`

`listings`: Grausamkeiten

listings: Grausamkeiten

- per default grausames Spacing:

```
\begin{lstlisting}
  float pi=3.14, f, r=25.4;
  f = pi*r*r;     /* die Fläche */
\end{lstlisting}
```

```
float pi=3.14, f, r=25.4;
f = pi*r*r; /* die Fl che */
```

- UTF-8 funktioniert nicht einfach so
  - Paket listingsutf8

`listings`: grausames Spacing beseitigen

```
\begin{lstlisting}[columns=flexible]
    float pi = 3.14, f, r = 25.4;
    f = pi * r * r;     /* die Fläche */
\end{lstlisting}
```

```
float pi = 3.14, f, r = 25.4;
f = pi * r * r; /* die Flche */
```

`listings`: grausames Spacing beseitigen (2)

```
\begin{lstlisting}[columns=fullflexible]
  float pi = 3.14, f, r = 25.4;
  f = pi * r * r;     /* die Fläche */
\end{lstlisting}
```

```
float pi = 3.14, f, r = 25.4;
f = pi * r * r; /* die Flche */
```

Überblick

Paket `minted`

- erfordert Installation des `pygments` Pakets für python
- LaTeX-Binary muss es erlaubt sein, einen weiteren Prozess zu starten
  - TeXLive: `pdflatex -shell-escape` …
  - Miktex: die Option heißt wohl `-enable-write18`
- in der Präambel `\usepackage{minted}`
- im Rumpf
  - Kommando `\mint`
  - Umgebung `minted`
  - Kommando `\inputminted`

Paket `minted`

- `\mint{[`⟨*options*⟩`]`⟨*language*⟩`}`⟨*char*⟩⟨*text*⟩⟨*char*⟩
- `\begin{minted}{[`⟨*options*⟩`]`⟨*language*⟩`}`
  ⟨*text*⟩
  `\end{minted}`
- `\inputminted[`⟨*options*⟩`]{`⟨*language*⟩`}{`⟨*filename*⟩`}`

- die Optionen kommen überwiegend von `fancyvrb`

Paket `minted` — ein Beispiel

C Quelle, mit `minted` gesetzt

```c
float pi = 3.14, f, r = 25.4;
f = pi * r * r ;     /* die Fläche */
```

Paket `minted` — ein Beispiel

C Quelle, mit `listings` gesetzt

```
float pi = 3.14, f, r = 25.4;
f = pi * r * r ;    /* die Flche */
```

Paket `minted` — ein Beispiel

C Quelle, mit `listingsutf8` gesetzt

```c
float pi = 3.14, f, r = 25.4;
f = pi * r * r ;     /* die Fläche */
```

Überblick

Pakete für Pseudocode gibt es viele

- `clrscode` bzw. `clrscode3e`
- `algorithm2e`
- weitere …

`clrscode` bzw. `clrscode3e`

- Algorithmennotation wie in *Introduction to Algorithms* von Cormen, Leiserson, Rivest und Stein

- `\usepackage{clrscode}`
  - wie in zweiter Auflage von CLRS
  - in texlive
- `\usepackage{clrscode3e}`
  - wie in dritter Auflage von CLRS
  - in texlive
- Unterschiede Geschmackssache

clrscode3e: Umgebung codebox

```
\begin{codebox}
  \Procname{$\proc{Foo-Sort}(A)$}
  \li \For $j \gets 2$ \To $\id{length}[A]$ \Do
  \li       $\id{key} \gets A[j]$
  \li       \Comment Insert $A[j]$ into .....
  \li           $i \gets j-1$
\end{codebox}
```

FOO-SORT(A)

```
1  for j = 2 to length[A]
2        key = A[j]
3        // Insert A[j] into .....
4        i = j − 1
```