ETEX advanced

Walter Stieben 4stieben@inf Hauke Stieler 4stieler@inf Ruben Felgenhauer 4felgenh@inf

18. Dezember 2017



Danke Henning (8pridoeh) dass wir deine Folien aus dem WS14/15 benutzen dürfen :D

Referenzieren

Referenzieren (Abschnitte)

LATEX-Code:

```
\subsection{Cliquen in bipartiten Graphen} \label{sec:cliques}
```

% Irgendwo anders

Im Abschnitt \ref{sec:cliques} auf Seite \pageref{sec:cliques} wurde das Finden von Cliquen in bipartiten Graphen beschrieben.

Ergebnis: Im Abschnitt 3.2 auf Seite 7 wurde das Finden von Cliquen in bipartiten Graphen beschrieben.



Referenzieren

Referenzieren (Figures)

```
LEX-Code:
```

Ergebnis: Der Lichtstrahl wird gebrochen, wie Abbildung 3 zeigt.

BibTEX

- Man verwaltet eine BibTEX-Datei (*.bib) mit Literaturangaben
- Mit \cite[Seite X] {Referenz} referenziert man eine solche Angabe, mit optionaler Seitenangabe.
- Vor pdflatex wirft man bibtex an

BibT_EX

LATEX-Code:

% Im Header
\bibliographystyle{alpha}

% Beim Zitat

Für die Lösung des Travelling-Salesman-Problems wurde ein heuristischer Algorithmus \cite{lin19973} gewählt.

% An der Stelle des Literaturverzeichnis
\bibliography{literatur}



BibT_EX-Eintrag

BibT_EX-Ergebnis

Ergebnis:

Für die Lösung des Travelling-Salesman-Problems wurde ein heuristischer Algorithmus [LK73] gewählt.

Literatur

[LK73] Shen Lin and Brian W. Kernighan. An effective heuristic algorithm for the travelling-salesman problem. *Operations Research*, 21:498–516, 1973.



Mit minted

- minted arbeitet mit Pygments (python-library).
- Benötigt -shell-escape als Parameter von pdflatex.

LATEX-Code:

```
\usepackage{minted}
% ...
\begin{minted}{java}
class MeineKlasse{
   private int meineVariable; // Deklaration

   public void meineMethode(){
       meineVariable = 42; // Initialisierung
}
```

Mit minted

Ergebnis:

```
class MeineKlasse{
    private int meineVariable; // Deklaration

public void meineMethode(){
    meineVariable = 42; // Initialisierung
}
```

Mit 1stlisting

LATEX-Code:

```
\usepackage{listings}
\lstset{\ldots} % style-einstellungen
% ...
\begin{lstlisting}[caption=Variablen]
class MeineKlasse{
    private int meineVariable; // Deklaration
    public void meineMethode(){
        meineVariable = 42; // Initialisierung
    }
\end{lstlisting}
```

Mit 1stlisting

Ergebnis:

Listing 1: Variablen

Makros

Eigene Befehle

```
LATEX-Code:
```

```
% TeX-style
\def\heute{Heute ist der \today.}
% LaTeX-style (besser)
\newcommand{\heute}{Heute ist der \today.}
\newcommand{\TikZ}{Ti\textit{k}Z}
% Verwendung
\heute
\TikZ
```

Ergebnis: Heute ist der 18. Dezember 2017.

TikZ



Makros

Eigene Befehle

LATEX-Code:

```
% LaTeX-style
```

```
\newcommand{\bus}[4]{Ein Bus der Linie #1 Richtung #2 fährt von der Haltestelle #3 um #4 ab.}
```

% Verwendung

\bus{181}{Sternschanze}{Informatikum}{13:37}

Ergebnis: Ein Bus der Linie 181 Richtung Sternschanze fährt von Informatikum um 13:37 ab.



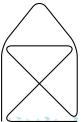
Makros

Eigene Umgebungen

```
LATEX-Code:
```

TikZ

- TikZ ist kein Zeichenprogramm.
- Abbildungen werden mit TikZ beschrieben und durch PGF gerendert.
- Sehr umfangreiches Paket (Dokumentation: >1000 Seiten), viele Möglichkeiten.
- Hat direkte Unterstützung für Petrinetze :-)
- Overkill: Animationen in einer PDF mittels JavaScript und TikZ:o



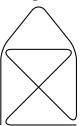


TikZ

LATEX-Code:

\begin{tikzpicture}
\draw[thick,rounded corners=8pt]
(0,0) -- (0,2) -- (1,3.25) -(2,2) -- (2,0) -- (0,2) -(2,2) -- (0,0) -- (2,0);
\end{tikzpicture}

Ergebnis:

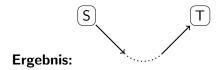


Nodes und Lines

TikZ-Code:

```
\begin{tikzpicture}
   \node[shape=rectangle,draw=black,rounded corners]
       (s) at (0, 0) {S}:
   \node[shape=rectangle,draw=black,rounded corners]
       (t) at (3, 0) {T}:
   \draw[thick, ->] (s) -- (1, -1):
   \draw[thick, dotted]
       (1, -1) to [bend right = 45] (2, -1);
   \draw[thick, ->] (2, -1) -- (t):
\end{tikzpicture}
```

Nodes und Lines



Hobby-Kurven

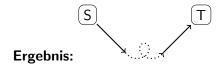
■ Hobby-Kurven mittels hobby-Paket

TikZ-Code:

```
\begin{tikzpicture}
  \node[shape=rectangle,draw=black,rounded corners]
    (s) at (0, 0) {S};
  \node[shape=rectangle,draw=black,rounded corners]
    (t) at (3, 0) {T};

  \draw[thick, ->]    (s)    -- (1, -1);
  \draw[thick, ->, dotted]
    (1, -1)
    to[curve through={(1.5, -1.1) .. (1.5, -0.75) .. (1.5, -1.1)}]
    (2, -1);
  \draw[thick, ->]    (2, -1) -- (t);
  \end{tikzpicture}
```

Hobby-Kurven

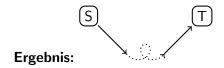


Styles für gesamtes TikZpicture

TikZ-Code:

```
\begin{tikzpicture}
        ->.
        thick.
        knoten/.style={shape=rectangle,draw=black,rounded corners}
    \node[knoten] (s) at (0, 0) {S}:
    \node[knoten] (t) at (3, 0) {T};
    \draw(s) -- (1, -1):
    \draw[dotted]
        (1, -1)
        to [curve through=\{(1.5, -1.1) .. (1.5, -0.75) .. (1.5, -1.1)\}]
        (2, -1);
    \draw (2, -1) -- (t):
\end{tikzpicture}
```

Styles für gesamtes TikZpicture



Einführung

- Automaten (state-machines) per automata-Paket
- Für Positionierung positioning-Paket
- Und für Pfeile arrows-Paket

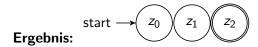
Mehr Informationen über Automaten, Pfeile, Positionierung, Optionen, etc. gibt es unter http://hauke-stieler.de/public/tikz-for-state-machines.pdf (im selben Ordner ist auch die *.tex Datei).



Zustände

```
TikZ-Code:
\usetikzlibrary{
    automata,
    arrows}
% ...
\begin{tikzpicture}[->,
    >=stealth',
    semithickl
                            (0)
                                              {$z_0$};
    \node[state,initial]
    \node[state]
                            (1)
                                 [right of=0] \{z_1\};
    \node[state,accepting]
                            (2)
                                 [right of=1] \{z_2\};
\end{tikzpicture}
```

Zustände



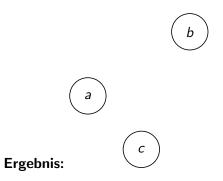


Positionierung

TikZ-Code:

```
\usetikzlibrary{
    automata.
    arrows,
    positioning}
% ...
\begin{tikzpicture}[->,
    >=stealth',
    semithick,
    node distance=2cml
    \node [state] (a)
                                                       {$a$};
    \node [state] (b) [above right=1cm and 2cm of a] {$b$};
    \node [state] (c) [below right of = a]
```

Positionierung

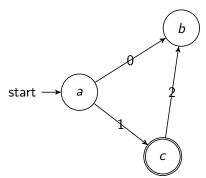


Pfeile

TikZ-Code:

```
\begin{tikzpicture}[->,
   >=stealth',
    semithick,
   node distance=2cml
    \node [state,initial]
                           (a)
                                            {$a$}:
    \node [state]
                            (b)
        [above right=1cm and 2cm of a]
                                            {$b$}:
    \node [state,accepting] (c)
        [below right = 1cm and 1.5cm of a] {$c$};
    \path (a) edge node {0} (b)
              edge node {1} (c)
          (c) edge node {2} (b);
\end{tikzpicture}
```

Pfeile



Ergebnis:

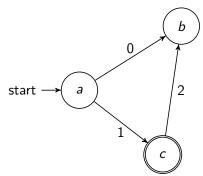
Pfeile

TikZ-Code:

```
\begin{tikzpicture}[->,
   >=stealth',
    semithick,
   node distance=2cml
    \node [state,initial]
                           (a)
                                            {$a$}:
    \node [state]
                            (b)
        [above right=1cm and 2cm of a]
                                            {$b$}:
    \node [state,accepting] (c)
        [below right = 1cm and 1.5cm of a] {$c$};
    \path (a) edge[above] node {0} (b)
              edge[below] node {1} (c)
          (c) edge[right] node {2} (b);
\end{tikzpicture}
```

31 / 41

Pfeile



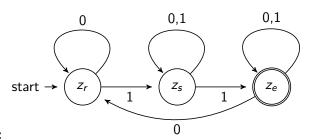
Ergebnis:

Pfeile

TikZ-Code:

```
\begin{tikzpicture}[->,>=stealth',
    shorten >=5pt,
   node distance=2.5cm,
    semithickl
    \node[initial,state]
                            (R.)
                                             \{ \z r \} :
    \node[state]
                            (S) [right of=R] {$z_s$};
    \node[state,accepting] (E) [right of=S] {$z_e$};
                                         node {0}
                                                     (R)
    \path
            (R) edge [loop,above]
                edge [below]
                                         node {1}
                                                     (S)
            (S) edge [loop,above]
                                         node {0,1} (S)
                edge [below]
                                         node {1}
                                                     (E)
            (E) edge [bend left,below] node {0}
                                                     (R)
                edge [loop,above]
                                         node \{0,1\} (E);
\end{tikzpicture}
```

Pfeile



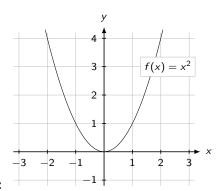
Ergebnis:



TikZ

```
\usepackage{pgf}
% ...
\begin{tikzpicture}[>=latex,semithick,font=\scriptsize,scale=0.75]
    \draw[very thin,color=lightgray] (-3.2,-1.2) grid (3.2,4.2);
    \draw[->] (-3.2,0) -- (3.4,0) node[right] {$x$};
    \draw[->] (0,-1.2) -- (0,4.4) node[above] {$v$};
    \foreach \x/\xtext in \{-3/-3, -2/-2, -1/-1, 1/1, 2/2, 3/3\}
    \displaystyle \frac{\sinh(x_0)}{1} = \frac{(x_0)}{1} = \frac{(x_0)}{1} = \frac{(x_0)}{1}
    \foreach \v/\\ytext in \{-1/-1, 1/1, 2/2, 3/3, 4/4\}
    \label{left} $$ \operatorname{shift}=\{(0,y)\}$ (2pt,0pt) -- (-2pt,0pt) node[left] {$\ytext$};
    \draw[thin,domain=-2.075:2.075,smooth,variable=\x,black]
        plot (\{\x\},\{\x*\x\});
    \draw[thin] node[inner sep=1mm,
                      fill=white,
                      draw=lightgray] at (2.25,3) \{ f(x) = x^2 \} ;
```

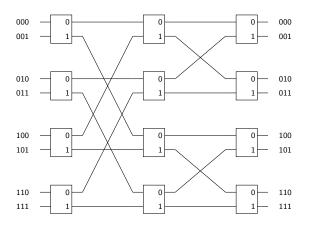
TikZ



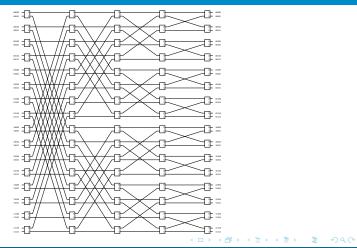
Ergebnis:



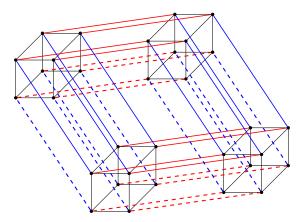
Banyan-Netz (3 Stufen)



Banyan-Netz (5 Stufen)



5-dimensionaler Hyperwürfel



... mehrere kaputte Kaffeemaschinen später ...



TikZat its best

