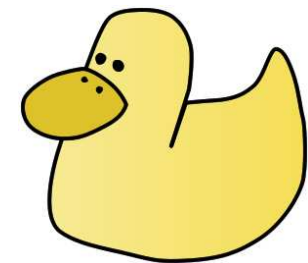


Professional Typesetting using \LaTeX

© 2003, 2004 Edsko de Vries

Hosted by **d u c s s**
Dublin University Computer Science Society



Introduction

- How (not why)
- Starting at the beginning
- Covering advanced topics
- Practical guidelines
- Sufficient for FYP documentation
- Slides available for reference

<http://www.edsko.net/latextalk.pdf>

Overview

1. Basics
2. Document Structure
3. Lists, tables and floats
4. Footnotes and references
5. Layout and fonts
6. Mathematics
7. Bibliography
8. Index
9. Code listings
10. Accents and special characters
11. Creating custom commands (macros)
12. Graphics
13. Diagrams, Trees and Graphs
14. Miscellaneous remarks

Hello, World

This is probably the simplest L^AT_EX file possible.

```
1 \documentclass{article}
2
3 \begin{document}
4 Hello, world
5 \end{document}
```

To compile:

- Save as `helloworld.tex`
- Run `latex helloworld.tex` to get `helloworld.dvi`
- Preview with `xdvi helloworld.dvi`

If You Are Running Windows

- Most popular T_EX distribution for Windows is MikT_EX (www.miktex.org).
- Alternatively, if you have access to a UNIX server
 1. Start an X-server (for example, Exceed)
 2. Login to the UNIX server (e.g. using PuTTY)
 3. Set your DISPLAY variable (e.g. `export DISPLAY=host:0.0`)
(`finger username` will tell you your host)
 4. Use `latex` and `xdvi` as normal
- If you are running some *NIX, but there is no T_EX installed, download teT_EX from www.tug.org/teTeX. Not difficult to install.

Printing and Generating PDFs

- To print, convert `.dvi` file to `.ps` and send to the printer queue.

```
1 dvips -o filename.ps filename.dvi
```

```
2 lpr filename.ps
```

- To generate a PDF (or to print on Windows), you have two options.

```
1 pdflatex filename.tex
```

(Will not work if you use `pstricks`). Alternatively:

```
1 dvips -Ppdf -o filename.ps filename.dvi
```

```
2 ps2pdf filename.ps
```

But make sure you have an up-to-date GhostScript (version 7 or higher).

Also, use PostScript Type 1 fonts (see section Fonts later on)

Paragraphs and Linebreaks

- New paragraphs are started by a linebreak
- A ‘hard return’ can be requested using `\\`
- More than one whitespace token is ignored

```
1 This is a bit of text \\
```

```
2 And another bit
```

```
3
```

```
4 Next paragraph
```

This is a bit of text

And another bit

Next paragraph

Document Structure (`article`)

An `article` is made up of sections and subsections. Available commands

- `\section{Section title}`
- `\subsection{Subsection title}`
- `\subsubsection{Subsubsection title}`

Or, if you want unnumbered ((sub)sub)section titles

- `\section*{Section title}`
- `\subsection*{Subsection title}`
- `\subsubsection*{Subsubsection title}`

If you must, subsubsections can be divided using paragraphs

- `\paragraph{Paragraph title}`
- `\subparagraph{Subparagraph title}`

Document Structure (`report`, `book`)

A `report` or a `book` is made up of chapters

- `\chapter{Chapter title}` or
- `\chapter*{Chapter title}`

(The individual chapters are made of sections.)

For appendices, use the command

- `\appendix`

All chapters defined after this command will be treated as appendices.

Finally, for large books consisting of multiple parts

- `\part{Part title}`

Example

```
1 \documentclass{report}
2
3 \begin{document}
4 \chapter{Introduction}
5 This is an introduction to \LaTeX.
6 \section{Why use \LaTeX}
7 Because it is better than the alternatives.
8 \section{Really?}
9 Yes
10 \chapter{Getting Started}
11 Download latextalk.pdf
12 \end{document}
```

Output (page 1)

Chapter 1

Introduction

This is an introduction to \LaTeX

1.1 Why use \LaTeX

Because it is better than the alternatives.

1.2 Really?

Yes

Output (page 2)

Chapter 2

Getting Started

[Download latextalk.pdf](#)

Table of Contents

To generate a table of contents, simply use

```
\tableofcontents
```

For the previous example, this yields

1	Introduction	2
1.1	Why use \LaTeX	2
1.2	Really?	2
2	Getting Started	3

Lists (1)

```
1 \begin{enumerate}
2 \item This is an example of a long item in a list
3 (notice how \LaTeX{} formats this paragraph)
4 \item Item 2
5 \item ...
6 \end{enumerate}
```

1. This is an example of a long item in a list (notice how L^AT_EX formats this paragraph)
2. Item 2
3. ...

Lists (2)

```
1 \begin{itemize}
2 \item This is an example of a long item in a list
3 (notice how \LaTeX{} formats this paragraph)
4 \item Item 2
5 \item ...
6 \end{itemize}
```

- This is an example of a long item in a list (notice how L^AT_EX formats this paragraph)
- Item 2
- ...

Lists (3)

```
1 \begin{description}
2 \item[Header A] This is an example of a long item in
3 a list (notice how \LaTeX{} formats this paragraph)
4 \item[Header B] Item 2
5 \item[Header C] ...
6 \end{description}
```

Header A This is an example of a long item in a list (notice how \LaTeX formats this paragraph)

Header B Item 2

Header C ...

More Compact Lists (1)

To get more compact lists, define your own (copy & paste):

```
1 \newcounter{cenum}
2 \newenvironment{cenum}%
3   {\begin{list}{\arabic{cenum}.}{%
4     \usecounter{cenum}%
5     \setlength{\itemsep}{0pt}%
6     \setlength{\parsep}{0pt}}}
7   {\end{list}}
8
9 \newenvironment{citem}%
10  {\begin{list}{$\bullet$}{%
11    \setlength{\itemsep}{0pt}%
12    \setlength{\parsep}{0pt}}}
13  {\end{list}}
```

Then use `\begin{cenum}` or `\begin{citem}` to get a compact list.

More Compact Lists (2)

Using `cenum`:

1. Item 1
2. Item 2
3. ...

Using `citem`:

- Item 1
- Item 2
- ...

Tables (1)

```
1 \documentclass{report}
2 \usepackage{array, longtable}
3
4 \begin{document}
5 \begin{longtable}{rcl}           % Right, center, left
6 item 1 & item 2 & item 3 \\
7 4      & 5      & 6      \\
8 \end{longtable}
9 \end{document}
```

item 1	item 2	item 3
4	5	6

Tables (2)

Adding rules.

```
1 \begin{longtable}{|r|c|l|}  
2 \hline  
3 item 1 & item 2 & item 3 \\  
4 \hline  
5 4 & 5 & 6 \\  
6 \hline  
7 \end{longtable}
```

item 1	item 2	item 3
4	5	6

Tables (3)

Column layout.

```
1 \begin{longtable}{|>{\bfseries}r|c|>{\sffamily\smaller}l|}  
2 \hline  
3 item 1 & item 2 & item 3 \\  
4 \hline  
5 4 & 5 & 6 \\  
6 \hline  
7 \end{longtable}
```

item 1	item 2	item 3
4	5	6

Simplification:

```
1 \newcolumntype{B}{>{\bfseries}r}  
2 \newcolumntype{S}{>{\sffamily\smaller}l}
```

Tables (4)

Long tables (more than one page).

```
1 \begin{longtable}{lll}
2 Head          & ... & ... \endfirsthead
3 Head (\textit{ctd.}) & ... & ... \endhead
4 Foot          & ... & ... \endfoot
5 Foot (\textit{end})  & ... & ... \endlastfoot
6 % Table contents
7 \end{longtable}
```

(All of these four lines are optional.)

Tables (5)

Multi-column and multi-row entries (needs `multirow` package).

Long Item		Item 2
Item 3	Item 4	Item 5
	Item 6	Item 7

```
1 \begin{longtable}{|1|1|1|} \hline
2 \multicolumn{2}{|c|}{Long Item} & Item 2 \\ \hline
3 \multirow{2}{2cm}{Item 3} & Item 4 & Item 5 \\ \cline{2-3}
4 & Item 6 & Item 7 \\ \hline
5 \end{longtable}
```

Note the use of `|c|` on line 2 instead of just `c`.

Tables (6)

Large table entries (more than one line of text)

```
1 \begin{longtable}{lp{0.8\linewidth}}
2 Item 1 & This is an example of a rather large cell. It
3   will only be formatted correctly with the
4   {\color{red} p} column type. \\ % color package
5 Item 2 & Example.
6 \end{longtable}
```

Item 1 This is an example of a rather long table cell. It will only be formatted correctly with the p column type.

Item 2 Example.

Floats (1)

- Objects that should not be split across pages (figures, tables)
- Position determined by \LaTeX using complex layout rules
- Four possible placements
 1. Inset in the text (**h**)
 2. Top of the page (**t**)
 3. Bottom of the page (**b**)
 4. Page of floats (**p**)
- User specifies allowable placements.
 1. `\begin{table}[h]` Try to place here. Not always possible.
 2. `\begin{figure}[tp]` Try to place at the top of the page, otherwise place on a page of floats
- `\listoftables` and `\listoffigures` will give a list of tables and figures respectively

Floats (2)

```
1 \begin{table}[htbp] % begin float for a table
2 \begin{longtable}{lll} \hline
3 a & b & c \\
4 d & e & f \\ \hline
5 \end{longtable}
6 \caption{Example floating table}
7 \end{table}
```

a	b	c
d	e	f

Table 1: Example floating table

Footnotes and Marginal Notes

To create a footnote^a, do

```
1 To create a footnote\footnote{Example}, do
```

A marginal note can be created with

```
1 \marginpar{text}
```

The note will be typeset in the outer margin.

Note	This is the normal text. The line separating the text and the margin is only shown here for clarity.
------	--

NOTE: Do not mix footnotes and margin notes.

(You will get a `floats lost` error message.)

^aExample

References

```
1 \section{Section Title}
2 \label{sec:SectionTitle}
3
4 ...
5
6 In section \ref{sec:SectionTitle} (page
7 \pageref{sec:SectionTitle}) we discussed...
```

Convention, prepend code to the label.

<code>sec</code>	Sections
<code>fig</code>	Figures
<code>tbl</code>	Tables
<code>eqn</code>	Equations
<code>lst</code>	Listings

References (2)

```
1 \usepackage{varioref}
2
3 As you can see in figure \vref{fig:LabelText}, ...
4
5 \begin{figure}[htbp]
6 ...
7 \caption{\label{fig:LabelText}LabelText}
8 \end{figure}
```

The `\vref` command generates references such as

- ‘3.5 on the next page’
- ‘6.2 on the facing page’
- ‘3.2 on page 64’

Layout (1)

- Default text alignment is justified. Change globally with

`\flushleft` Left aligned

`\flushright` Right aligned

`\center` Centered

For a local change, use `\begin{center} .. \end{center}`

- Default page alignment is portrait. Change (locally) with

```
1 \usepackage{lscape}
```

```
2
```

```
3 \begin{landscape}
```

```
4 ..
```

```
5 \end{landscape}
```

Useful for wide tables (`xdvi` will display this incorrectly)

Layout (2)

- To set headers and footers

```
1 \usepackage{fancyhdr}
2 \pagestyle{fancy}
3
4 \lhead{..}
5 \chead{..}
6 \rhead{..}
7 \lfoot{..}
8 \cfoot{..}
9 \rfoot{..}
```

- Useful commands to use are `\thepage`, `\thechapter`, `\chaptertitle`, `\thesection`, and `\sectiontitle`
- To get an empty page, use `\thispagestyle{empty}`

Layout (3)

Some variables governing layout

<code>\parskip</code>	l	Distance between paragraphs
<code>\parindent</code>	l	Indentation of the first line of a paragraph
<code>\marginparwidth</code>	l	Width of marginal notes
<hr/>		
<code>topnumber</code>	c	Max number of floats (top the page)
<code>bottomnumber</code>	c	Max number of floats (bottom)
<code>totalnumber</code>	c	Max number of floats (total)
<hr/>		
<code>\topfraction</code>	r	Max fraction at the top of the page used for floats
<code>\bottomfraction</code>	r	Likewise, bottom of the page
<code>\textfraction</code>	r	Min fraction of the page used for text

Set using

```
1 \setlength{\parskip}{1em}
2 \setcounter{topnumber}{2}
3 \renewcommand{\topfraction}{0.3}
```

(NB: `1ex` is average line height, and `1em` is average character width)

Layout (4)

Local layout changes (methods of last resort).

<code>\pagebreak</code>	Insert a page break
<code>\nopagebreak</code>	Prohibit a page break
<code>\clearpage</code>	Insert a page break, output all floats
<code>\enlargethispage{x}</code>	Enlarge the current page by x (may be negative)
<code>\enlargethispage*{x}</code>	Likewise, and squeeze page contents
<code>\indent</code>	Indent the next paragraph
<code>\noindent</code>	Do not indent the next paragraph
<code>\hspace{x}</code>	Insert x horizontal space
<code>\vspace{x}</code>	Insert x vertical space
<code>\hfill</code>	Insert a horizontal ‘stretch’ length
<code>\vfill</code>	Insert a vertical ‘stretch’ length
<code>\fussy</code>	Be fussy about linebreaking
<code>\sloppy</code>	Be sloppy about linebreaking

Fonts (family, series, shape)

Four properties (family, series, shape, size) can be set independently.

<code>\textnormal{...}</code>	<code>{\normalfont ...}</code>	Default font
<code>\textrm{...}</code>	<code>{\rmfamily ...}</code>	Roman family
<code>\textsf{...}</code>	<code>{\sffamily ...}</code>	Sans serif family
<code>\texttt{...}</code>	<code>{\ttfamily ...}</code>	Typewriter family
<code>\textmd{...}</code>	<code>{\mdseries ...}</code>	Medium series
<code>\textbf{...}</code>	<code>{\bfseries ...}</code>	Bold series
<code>\textup{...}</code>	<code>{\upshape ...}</code>	Upright shape
<code>\textit{...}</code>	<code>{\itshape ...}</code>	<i>Italic shape</i>
<code>\textsl{...}</code>	<code>{\slshape ...}</code>	<i>Slanted shape</i>
<code>\textsc{...}</code>	<code>{\scshape ...}</code>	SMALL CAPS SHAPE

Fonts (size)

`{\tiny ...}`

Example

`{\scriptsize ...}`

Example

`{\footnotesize ...}`

Example

`{\small ...}`

Example

`{\normalsize ...}`

Example

`{\large ...}`

Example

`{\Large ...}`

Example

`{\LARGE ...}`

Example

`{\huge ...}`

Example

`{\Huge ...}`

Example

Fonts (type face)

Changing type face: manually (difficult) or using package.

Package	Sans font	Roman font	Math Package
(default)	ComputerModernSans	ComputerModernRoman	(default)
times	Helvetica	Times	mathptm or mathpazo
palatino	Helvetica	Palatino	
helvet	Helvetica	–	
avant	AvantGarde	–	
newcent	AvantGarde	NewCenturySchoolBook	
bookman	AvantGarde	Bookman	

I highly recommend using one of these packages, because they change the fonts to scalable PS Type 1 fonts, which yield high quality PDF files.

Fonts (examples)

Changing to Times type face, and setting overall font size to 11pt.

```
1 \documentclass[11pt]{article}
2 \usepackage{mathptm, times}
```

Changing to Helvetica, using mathpazo for mathematics.

```
1 \documentclass{article}
2 \usepackage{mathpazo, helvet}
3
4 \begin{document}
5 \sfseries
6
7 ..
8 \end{document}
```

Fonts (other remarks)

Arbitrary font size.

```
1 { \fontsize{100}{105} \selectfont ENORMOUS. }
```

First argument is the font size in points (72pt = 1in = 2.54cm), second is the baseline skip (distance between two lines of text).

Fonts do not always mix very well. Solution:

```
1 \begin{document }
```

```
2 \usepackage{relsize}
```

```
3
```

```
4 \begin{document }
```

```
5 Fonts do not \textsf{\textsmaller{always}} mix.
```

```
6 \end{document }
```

Typesetting Mathematics

Two possibilities: inset in the text \dots , or typeset as a separate paragraph.

```
1 \begin{equation}
2 \label{eqn:EquationLabel}
3 \dots
4 \end{equation}
```

(Or `equation*` for an unnumbered equation)

Example

```
1 Side effects make  $n^2 \neq n \times n$ 
```

Side effects make $n^2 \neq n \times n$

Setup

```
1 \usepackage[intlimits]{amsmath}
2 \usepackage{amstext, amssymb}
3 \usepackage[mathscr]{euscript}
4 \usepackage{eufrak}
5 \usepackage[mathscr]{euscript}
```

Some useful definitions (copy & paste).

```
1 \newcommand{\tsum}{\textstyle\sum}
2 \newcommand{\dsum}{\displaystyle\sum}
3 \newcommand{\tint}{\textstyle\int}
4 \newcommand{\dint}{\displaystyle\int}
5 \newcommand{\toint}{\textstyle\oint}
6 \newcommand{\dooint}{\displaystyle\oint}
7 \newcommand{\tprod}{\textstyle\prod}
8 \newcommand{\dprod}{\displaystyle\prod}
9 \newcommand{\tcup}{\textstyle\bigcup}
10 \newcommand{\dcup}{\displaystyle\bigcup}
11 \newcommand{\tcap}{\textstyle\bigcap}
12 \newcommand{\dcap}{\displaystyle\bigcap}
```

Common Constructs

`x`, `x^n`, `x_m`, `x^n_m`, `x_{n+1}`, `x_{n_m}`

`\frac{x}{y}`, `\dfrac{x}{y}`

`\sum_x^y`, `\dsum_x^y`

`\prod_x^y`, `\dprod_x^y`

`\int_x^y`, `\dint_x^y`, `\oint_x^y`, `\doint_x^y`

`\cup_x^y`, `\dcup_x^y`, `\cap_x^y`, `\dcap_x^y`

`\sqrt{x}`, `\sqrt[y]{x}`

`\xrightarrow{xyz}`, `\xleftarrow[f]{xyz}`

x , x^n , x_m , x_m^n , x_{n+1} , x_{n_m}

$\frac{x}{y}$, $\frac{x}{y}$

\sum_x^y , \sum_x^y

\prod_x^y , \prod_x^y

\int_x^y , \int_x^y , \oint_x^y , \oint_x^y

\cup_x^y , \cup_x^y , \cap_x^y , \cap_x^y

\sqrt{x} , $\sqrt[y]{x}$

\xrightarrow{xyz} , $\xleftarrow[f]{xyz}$

Cases

```
1 \begin{equation}
2 \text{\texttt{\text{fac } x =}}
3   \begin{cases}
4     1 & \text{\texttt{\text{if } x = 0 \\}}
5     x \cdot \text{\texttt{\text{fac } (x-1)}} & \text{\texttt{\text{if } x > 0}}
6   \end{cases}
7 \end{equation}
```

$$\text{fac } x = \begin{cases} 1 & \text{if } x = 0 \\ x \cdot \text{fac } (x - 1) & \text{if } x > 0 \end{cases} \quad (1)$$

Matrices

```
1 \begin{equation}
2 \begin{pmatrix}
3 a & b & \cdots & c \\
4 d & e & & \\
5 \vdots & & \ddots & \\
6 f & & & z \\
7 \end{pmatrix}
8 \end{equation}
```

$$\begin{pmatrix} a & b & \cdots & c \\ d & e & & \\ \vdots & & \ddots & \\ f & & & z \end{pmatrix} \quad (2)$$

Also: `bmatrix` [..], `Bmatrix` {..}, `vmatrix` |..| and `Vmatrix` || .. ||

Delimiters

```
1 y = \left( \int_{-\infty}^{\infty} f(x) \right) ^{x}
```

$$y = \left(\int_{-\infty}^{\infty} f(x) \right)^x \quad (3)$$

```
1 \Biggl(\biggl(\Bigl(\bigl(x\bigr)\Bigr)\biggr)\Biggr)
```

$$\left(\left(\left(x\right)\right)\right) \quad (4)$$

If you want only one delimiter: `\left(x \right.` (Notice the dot)

Fonts in math mode

<code>\mathrm{cos}</code>	cos
<code>\mathbf{w}</code>	w
<code>\mathit{effect}</code>	<i>effect</i> (cf. <code>\$effect\$</code> , <i>effect</i>)
<code>\mathcal{C}</code> , <code>\mathcal{S}</code>	\mathcal{C} , \mathcal{S}
<code>\mathscr{L}</code>	\mathscr{L}
<code>\mathfrak{R}</code> , <code>\mathfrak{I}</code>	\mathfrak{R} , \mathfrak{I}
<code>\mathbb{R}</code> , <code>\mathbb{C}</code>	\mathbb{R} , \mathbb{C}

Also useful

<code>\text{if } \$a\$ is even}</code>	if a is even (cf. <i>ifaiseven</i>)
<code>x \quad x</code>	x x (Insert small space)
<code>x \qquad x</code>	x x (Insert large space)

Symbols available in math mode

1. Abstract mathematics: \circ (`circ`), \bullet (`bullet`), \sim (`sim`), \simeq (`simeq`), \propto (`propto`), \mapsto (`mapsto`), \forall (`forall`), \exists (`exists`), \nexists (`nexists`)
2. Arrows: \leftarrow (`leftarrow`), \Leftrightarrow (`Leftarrow`), \rightarrow (`rightarrow`), \Rightarrow (`Rightarrow`), \Uparrow (`uparrow`), \circlearrowleft (`circulararrowleft`), \Downarrow (`downdownarrows`)
3. Arithmetic: \times (`times`), \cdot (`cdot`), \pm (`pm`), \neq (`neq`), \leq (`leq`), \geq (`geq`), \ll (`ll`), \gg (`gg`), \equiv (`equiv`), \approx (`approx`), \Re (`Re`), \Im (`Im`), ∂ (`partial`), ∞ (`infty`)
4. Set theory: \in (`in`), \notin (`notin`), \subset (`subset`), \sqsubset (`sqsubset`), \subseteq (`subseteq`), $\not\subseteq$ (`nsubset`), \hookrightarrow (`hookrightarrow`), \setminus (`setminus`), \cap (`cap`), \cup (`cup`), \emptyset (`varnothing`), \wp (`wp`)
5. Logic: \neg (`neg`), \vee (`vee`), \wedge (`wedge`), \oplus (`oplus`), \vdash (`vdash`), \nvdash (`nvdash`), \models (`models`), $\not\models$ (`nvDash`)
6. Etc.: \dots (`ldots`), \cdots (`cdots`), \vdots (`vdots`), \ddots (`ddots`)
7. Delimiters: \lfloor (`lfloor`), \rfloor (`rfloor`), \langle (`langle`), \rangle (`rangle`), \lceil (`lceil`), \rceil (`rceil`)

Example: the set of graphs is denoted $\text{Set}^{\downarrow\downarrow \cdot}$

```
 $\text{\text{\text{Set}}^{\overset{\cdot}{\Downarrow}}}$ 
```

Creating a Bibliography

First, create a database.

```
1 @book{bib:Jones:1993,  
2   author      ={N.D. Jones and C.K. Gomard and P. Sestoft},  
3   title       ={{Partial Evaluation and Automatic Program Generation}},  
4   publisher   ={Prentice Hall International},  
5   year        ={1993},  
6   note        ={Currently out of print.}  
7 }  
8  
9 @manual{bib:Paxson:1995,  
10  author      ={Vern Paxson},  
11  title       ={{The GNU \texttt{flex} Manual}},  
12  year        ={1995}  
13 }
```

- Double accolades around title preserve capitalisation
- Use ‘x and y and z’ style for authors

Creating a Bibliography (2)

```
1 @misc{bib:Power:2002,  
2   author      = {James Power},  
3   title       = {{Notes on Formal Language Theory and Parsing}},  
4   note        = {http://\-www.\-cs.\-may.\-ie/\-\~{}jpower/\-parsing},  
5   year        = {2002}  
6 }  
7  
8 @article{bib:Aycock:2002,  
9   author      = {John Aycock and R. Nigel Horspool},  
10  title       = {{Practical Earley Parsing}},  
11  journal     = {The Computer Journal},  
12  year        = {2002},  
13  volume     = {45},  
14  number     = {6},  
15  pages      = {620--630},  
16  month      = {April}  
17 }
```

Creating a Bibliography (3)

If your L^AT_EX file is `foo.tex`, save the bibliography database as `foo.bib`. Then add citations to your document:

```
1 \cite[note]{label}
```

For example

```
1 As stated in \cite[pp. 45--48]{bib:A1},  
2 article \cite{bib:A2} is rubbish.
```

And at the end of your document

```
1 \bibliographystyle{acm}  
2 \bibliography{foo}
```

(Many other styles available; search for `.bst` files.)

Creating a Bibliography (4)

Run `latex`. You will get warnings

```
1 LaTeX Warning: Citation 'a1' on page 1 undefined on input line 5
2 LaTeX Warning: Citation 'a2' on page 1 undefined on input line 5
```

But a file `foo.aux` has been created with a list of the required references.

Then run `bibtex`

```
1 # bibtex foo
```

Run `LATEX`

```
1 latex
```

If you see the following warning, run `LATEX` once more.

```
1 LaTeX Warning: Label(s) may have changed.
2     Rerun to get cross-references right.
```

Bibliography Example (1)

```
1 @article{a1,  
2   author   ={First Author},  
3   title    ={This is the first test},  
4   journal  ={Communications of the ACM},  
5   year     ={2000}  
6 }  
7  
8 @article{a2,  
9   author   ={Another Author},  
10  title    ={This is the second test},  
11  journal  ={Artificial Life},  
12  year     ={2001}  
13 }
```

```
1 As stated in \cite[pp. 45--48]{a1},  
2 article \cite{a2} is a load of rubbish.  
3  
4 \bibliographystyle{acm}  
5 \bibliography{t}
```

Bibliography Example (2)

In the document

As stated in [2, pp. 45–48], article [1] is a load of rubbish.

The bibliography itself

References

[1] AUTHOR, A. This is the second test. *Artificial Life* (2001)

[2] AUTHOR, F. This is the first test. *Communications of the ACM* (2000)

Using a different style (`ieeetr`)

As stated in [1, pp. 45–48], article [2] is a load of rubbish.

and

References

[1] F. Author, “This is the first test,” *Communications of the ACM*, 2000

[2] A. Author, “This is the second test,” *Artificial Life*, 2001

Creating an Index (1)

Setup

```
1 \usepackage{makeidx}
2 \makeindex
```

In the document

```
1 \index{a}
2 \index{b|see{a}}
3 text
4
5 \pagebreak
6 \index{1@{ $\lambda$ }}
7 text
```

At the end of the document

```
1 \printindex
```

Creating an Index (2)

Run

```
1 latex foo.tex  
2 makeindex foo  
3 latex foo.tex
```

The index will look like

Index

a, 1

b, *see* a

λ , 2

Typesetting Listings

```
1 \usepackage{lstlistings}
2
3 \begin{lstlisting}[language=c]
4 #include <stdio.h>
5
6 void main()
7 {
8     printf( "Hello, world" );
9 }
10 \end{lstlisting}
```

```
#include <stdio.h>

void main()
{
    printf( "Hello, world" );
}
```

Typesetting Listings (2)

Configuration

```
1 \lstset{var1=val1,var2=val2,...}
```

fbat	htbp
basicstyle	\ttfamily
tabsize	3
showstringspaces	false
numbers	left
numberstyle	\tiny
name	(Like-named listings use continuous numbering)
caption	(Listing caption)
label	(Used in \ref and \pageref)
breaklines	true
frame	tblr
mathescape	true
escapechar	\%
boxpos	t (Use when typesetting a listing inside a table)

Typesetting Listings (3)

```
1 \lstset{
2     literate=
3         {<}{{\langle$}}{1}
4         {>}{{\rangle$}}{1}
5         {::=}{{\rightarrow$}}{1}
6         {\{\}}{{\epsilon$}}{1}
7     }
8
9 \begin{lstlisting}
10 <A> ::= a <B> <C>
11 <B> ::= {}
12 <C> ::= {}
13 \end{lstlisting}
```

Typesetting Listings (4)

This code yields

```
⟨A⟩ → a  ⟨B⟩  ⟨C⟩  
⟨B⟩ → ε  
⟨C⟩ → ε
```

Other useful commands

<code>\lstinputlisting{filename}</code>	Input listing from a file
<code>\lstlistoflistings</code>	Get a list of all listings
<code>\lstdefinestyle{name}{..}</code>	Define a style.

Once a style is defined

```
1 \begin{lstlisting}[style=name]
```

Typesetting Listings (5)

To define your own language, use

```
1 \lstdefinlanguage {name} { .. }
```

Summary of available commands

<code>keywords={a,b,..}</code>	Define the keywords of the language
<code>keywordsprefix={x}</code>	All identifiers starting with 'x' are keywords
<code>tag={x}{y}</code>	Keywords must be delimited by 'x' and 'y' (HTML)
<code>sensitive={x}</code>	Are keywords case sensitive?
<code>alsoletter={x}</code>	Treat 'x' as part of an identifier
<code>comment=[l]{x}</code>	Define a line comment (e.g. '//')
<code>comment=[n]{x}</code>	Define a nested comment (e.g. '/* .. */')
<code>string=[b]{x}</code>	Define strings (also <code>morestring</code>)

Typesetting Listings (6)

Example language definitions.

```
1 \lstdefinlanguage{java}{
2   keywords={abstract,const,final,public,static,...},
3   comment=[l]{//},
4   morecomment=[n]{/*}{*/},
5   string=[b]{"}
6 }
7
8 \lstdefinlanguage{m68k}{
9   keywords={equ,macro,subq,bgt,endm,dc.w,move.l,...},
10  comment=[l]{*},
11  sensitive=false,
12  alsochar={.},
13  string=[b]{'}
14 }
```

Accents and Symbols

- This overview shows what commands to use to get accented characters.

ò	<code>\`{o}</code>	ö	<code>\" {o}</code>	o	<code>\b{o}</code>	ô	<code>\t{oo}</code>
ó	<code>\' {o}</code>	õ	<code>\H{o}</code>	ô	<code>\. {o}</code>	ô	<code> \$\widehat{\text{oo}}\$</code>
ô	<code>\^ {o}</code>	õ	<code>\~ {o}</code>	ç	<code>\c{c}</code>	ô	<code> \$\widetilde{\text{oo}}\$</code>
ö	<code>\v{o}</code>	ō	<code>\={o}</code>	o	<code>\d{o}</code>		

Note: dot-less i and j: `\i` and `\j` (`\ " {\i}`). Not always necessary/available.

- Other symbols available

æ	<code>\ae</code>	å	<code>\aa</code>	ł	<code>\l</code>	†	<code>\dag</code>	§	<code>\S</code>
Æ	<code>\AE</code>	Å	<code>\AA</code>	Ł	<code>\L</code>	‡	<code>\ddag</code>	¶	<code>\P</code>
œ	<code>\oe</code>	ø	<code>\o</code>	ß	<code>\ss</code>	...	<code>\dots</code>	©	<code>\copyright</code>
Œ	<code>\OE</code>	Ø	<code>\O</code>						

Special Characters

The following list of characters have a special meaning. If you want to insert one of these characters into your document (literally), you must use a different notation.

Character	Use	Character	Use
{	\{	< (<, <)	\verb!<! (\$<\$, \$\langle\$)
}	\}	> (>, >)	\verb!>! (\$>\$, \$\rangle\$)
-	\-		\$ \$
%	\%	\	\$\$\backslash\$
&	\&	^	\$\$^\wedge\$
\$	\\$	~ (~, ~)	\$\$\sim\$ (\$^\sim\$, ~{ })
#	\#	“ ”	`` , '' (don't use ")

Custom Commands

For example, to simplify use of the tilde (\sim)

```
1 \newcommand{\tilde}{\sim}
```

This works, except when the command is used in math mode, for example

```
1 A is proportional to B, denoted $A \tilde B$
```

Substitution yields

```
1 A is proportional to B, denoted $A \sim B$
```

This is wrong. Why?

```
1 A is proportional to B, denoted $A \^{\sim} B$
```

Solution:

```
1 \newcommand{\tilde}{\ensuremath{\sim}}
```

Custom Commands (2)

To simplify the use of a wide hat ($\widehat{\circ\circ}$)

```
1 \newcommand{\wh}[1]{\ensuremath{\widehat{\text{\#1}}}}
```

Now we can use `\wh{\circ\circ}` to get $\widehat{\circ\circ}$.

Note. Consider

```
1 \newcommand{\projectname}{My Project}
```

Then

```
1 \projectname is a parser for general CFGs.
```

```
2 \projectname{} is a parser for general CFGs.
```

Yields

My Project is a parser for general CFGs.

My Project is a parser for general CFGs.

Including Graphics

Use `.jpg` or `.png` for `pdflatex`, use `.eps` (Encapsulated PostScript) for `latex` (many programs can convert to EPS, e.g. Gimp).

Then

```
1 \usepackage{graphicx}
2 ...
3 \includegraphics[width=x]{filename}
```

Do *not* include the extension in the filename. `latex` will automatically choose the `.eps`, `pdflatex` will automatically choose the `.jpg` or `.png` version (in that order).

To get a box only (no actual picture, speed up typesetting)

```
1 \usepackage[draft]{graphicx}
```

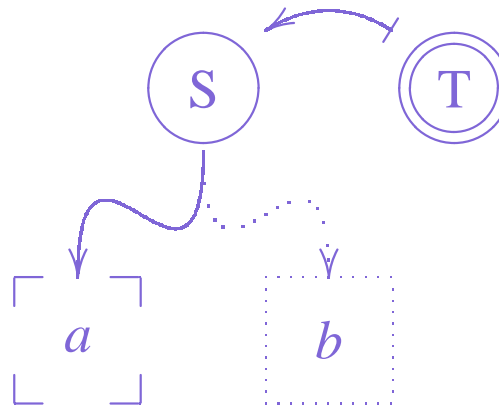
Typesetting Diagrams with Xy-pic

- Works with `pdflatex`
- Very powerful
- Not too difficult (if you follow the rules)



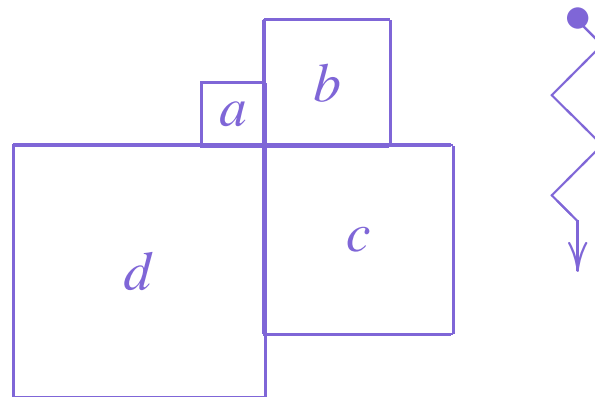
```
1 \usepackage[all]{xy}
2
3 \xy <1cm,0cm>:
4 \POS (0,0) *+{a} *\frm{o} ="a"
5 \POS (3,0) *+{b} *\frm{o} ="b"
6 \POS "a" , \ar @{->} "b"
7 \endxy
```

Xy-pic (2)



```
1 \xy <1cm,0cm>:
2 \POS (0,0) *=(1,1)\txt{S} *\frm{o} ="s"
3 \POS (2,0) *=(1,1)\txt{T} *\frm{oo} ="t"
4 \POS (-1,-2) *=(1,1){a} *\frm{--} ="a"
5 \POS (1,-2) *=(1,1){b} *\frm{.} ="b"
6 \POS "s" , \ar @/^0.5cm/ @{<.>} "t"
7 \POS "s" !DC , \ar @{->} @(d,u) "a" !UC
8 \POS "s" !DC , \ar @{.>} @(d,u) "b" !UC
9 \endxy
```

Xy-pic (3)

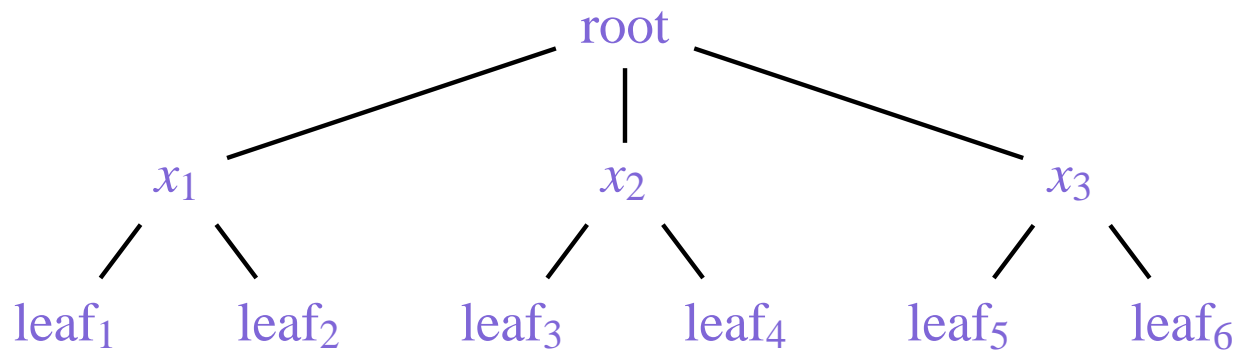


```
1 \xy <1cm,0cm>:
2 \POS (0,0) *=(0.5,0.5)!DR {a} *\frm{-}
3 \POS (0,0) *=(1.0,1.0)!DL {b} *\frm{-}
4 \POS (0,0) *=(1.5,1.5)!UL {c} *\frm{-}
5 \POS (0,0) *=(2.0,2.0)!UR {d} *\frm{-}
6 \POS (2.5,1) *{\bullet}
7 \POS (2.5,1) , \ar @{->} '+(0.2,-0.2) '-(0.4,0.4)
8     '+(0.4,-0.4) '-(0.4,0.4) '+(0.2,-0.2) '-(0,0.4)
9 \endxy
```

Tree Diagrams

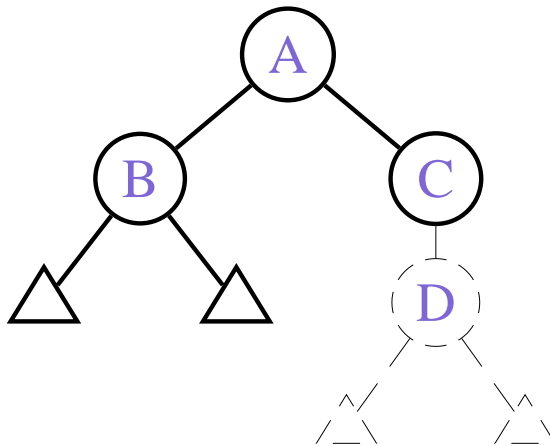
Uses `pstricks` package. Does not work with `pdflatex`!

```
1 \usepackage{pst-all}
2 \psset{nodesep=1ex,levelsep=6ex}
3 \pstree{\TR{root}} {
4   \pstree{\TR{$x_1$}} { \TR{leaf$_1$} \TR{leaf$_2$} }
5   \pstree{\TR{$x_2$}} { \TR{leaf$_3$} \TR{leaf$_4$} }
6   \pstree{\TR{$x_3$}} { \TR{leaf$_5$} \TR{leaf$_6$} }
7 }
```



Tree Diagrams (2)

```
1 \pstree[levelsep=5ex]{\Tcircle{A}} {  
2   \pstree{\Tcircle{B}} { \Ttri{} \Ttri{} }  
3   \pstree{\Tcircle{C}} {  
4     \psset{linestyle=dashed,linewidth=0.1pt}  
5     \pstree{\Tcircle{D}} { \Ttri{} \Ttri{} }  
6   }  
7 }
```



Drawing Graphs: PostScript Stack Machine

We can draw highly accurate graphs by encoding a function

$$y = f(x)$$

in the PostScript programming language. The PS engine is a *stack machine*, and takes a bit of getting used to. Example:

$$y = 2x^2$$

is encoded by

```
1 2 x dup mul mul
```

Another PS example

The function

$$4 \sin(x^2) - 10x$$

Is encoded by

```
1 4 x dup mul sin mul 10 x mul sub
```

To understand, imagine the PS stack.

4	x	dup	mul	sin	mul	10	x	mul	sub
4	4	4	4	4	$4 \sin(x^2)$	$4 \sin(x^2)$	$4 \sin(x^2)$	$4 \sin(x^2)$	$4 \sin(x^2) - 10x$
	x	x	x^2	$\sin(x^2)$		10	10	10x	
		x					x		

PostScript Operators

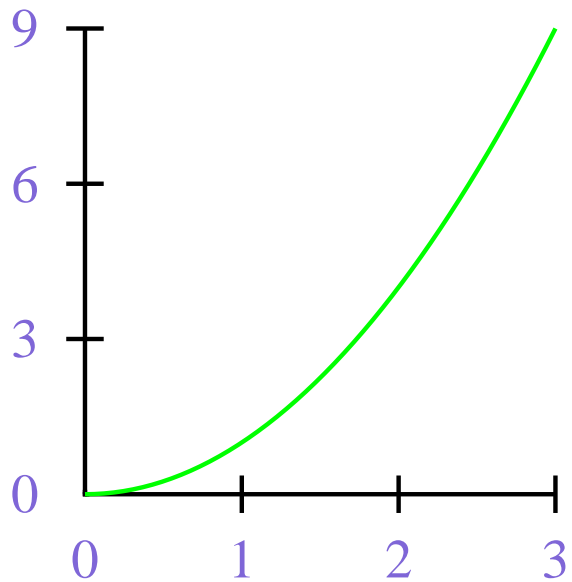
exch	$\text{pop}(x_0), \text{pop}(x_1), \text{push}(x_0), \text{push}(x_1)$
dup	$\text{pop}(x_0), \text{push}(x_0), \text{push}(x_0)$
neg	$\text{pop}(x_0), \text{push}(-x_0)$
add	$\text{pop}(x_0), \text{pop}(x_1), \text{push}(x_1 + x_0)$
sub	$\text{pop}(x_0), \text{pop}(x_1), \text{push}(x_1 - x_0)$
mul	$\text{pop}(x_0), \text{pop}(x_1), \text{push}(x_1 \times x_0)$
div	$\text{pop}(x_0), \text{pop}(x_1), \text{push}(x_1 / x_0)$
exp	$\text{pop}(x_0), \text{pop}(x_1), \text{push}(x_1^{x_0})$
sin	$\text{pop}(x_0), \text{push}(\sin(x_0))$ (degrees)
cos	$\text{pop}(x_0), \text{push}(\cos(x_0))$
atan	$\text{pop}(x_0), \text{push}(\arctan(x_0))$
sqrt	$\text{pop}(x_0), \text{push}(\sqrt{x_0})$
log	$\text{pop}(x_0), \text{push}(\log_{10}(x_0))$
ln	$\text{pop}(x_0), \text{push}(\log_e(x_0))$
round	$\text{pop}(x_0), \text{push}(\text{round}(x_0))$
fbor	$\text{pop}(x_0), \text{push}(\lfloor x_0 \rfloor)$
ceiling	$\text{pop}(x_0), \text{push}(\lceil x_0 \rceil)$



$$\log_y x = \frac{\log_{10} x}{\log_{10} y}$$

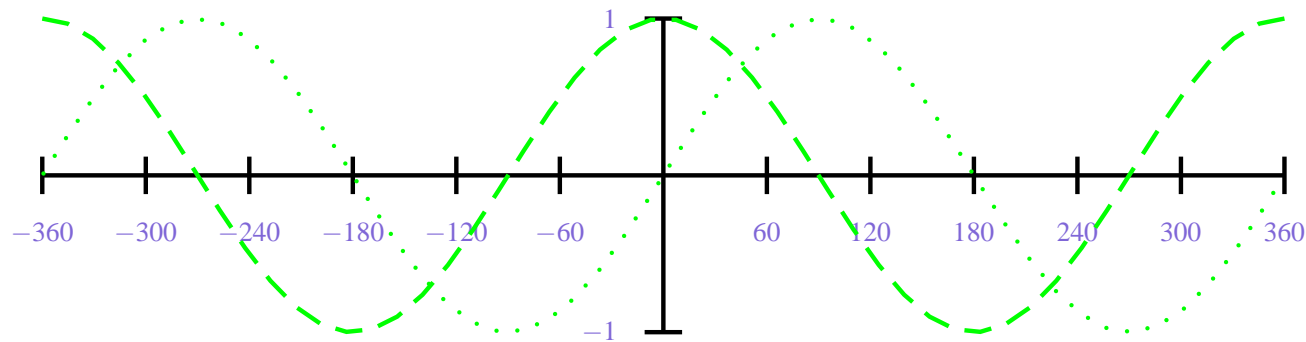
Drawing Graphs (2)

```
1 \begin{pspicture}(3,3)
2 \psset{xunit=1,yunit=0.33}
3 \psaxes[Dx=1,Dy=3](3,9)
4 \psplot{0}{3}{x dup mul}
5 \end{pspicture}
```



Drawing Graphs (3)

```
1 \begin{pspicture}(-4,-1)(4,1)
2 \tiny
3 \psset{xunit=0.011,yunit=1}
4 \psaxes[Dx=60,Dy=1](0,0)(-360,-1)(360,1)
5 \psplot[linestyle=dotted]{-360}{360}{x sin}
6 \psplot[linestyle=dashed]{-360}{360}{x cos}
7 \end{pspicture}
```



Some final remarks

- You can split up a \LaTeX file into different files. Then use

```
1 \input filename
```

to ‘include’ them

- To typeset a small bit of ‘verbatim’ code

```
1 Use \verb!\textbf! to select the bold font series
```

(Syntax: `\verb\diamond\diamond`)

- Sometimes `xdvi` gets confused with diagrams. Press `SHIFT+V` twice.
- To check the spelling of a \LaTeX document, use `ispell`
- Works well (of course) with CVS (add `.tex` and `.bib` files)

Other references

- Most L^AT_EX distributions come with excellent documentation. On UNIX systems, an index is usually found at

`/usr/share/texmf/doc/newhelpindex.html`

- Books

1. The L^AT_EX Companion. *Michel Goossens, Frank Mittelbach and Alexander Samarin*, Addison-Wesley
2. The L^AT_EX Graphics Companion. *Michael Goossens, Sebastian Rahtz, Frank Mittelbach*, Addison-Wesley
3. The T_EXbook. *Donald E. Knuth*, Addison-Wesley

- Me (edsko@edsko.net)

Questions & Answers