ECON 6009
Graduate Seminar
Memorial University of Newfoundland

Lecture 2-*Introduction to Latex (1)*

The objectives

- To strengthen the knowledge the student gained in previous Economics modules about how to conduct academic research and **present it to an academic audience**

- To develop students' capacity for self-motivated learning and problem solving during the practical process of conducting academic research

The tools

STATA available at Graduate
Resource Room and

Also see GradPlan as a purchasing
option

The tools

MiKTeX, which is free-source together with an editing interface of your choice (some of which are also free)

The tools

- LATEX (originally developed by Leslie Lamport) is a typesetting system very suitable for producing scientific and mathematical documents of high typographical quality

- and for producing all sorts of other documents, from simple letters to complete books

- LATEX uses TEX (originally developed by Donald Knuth) as its formatting engine
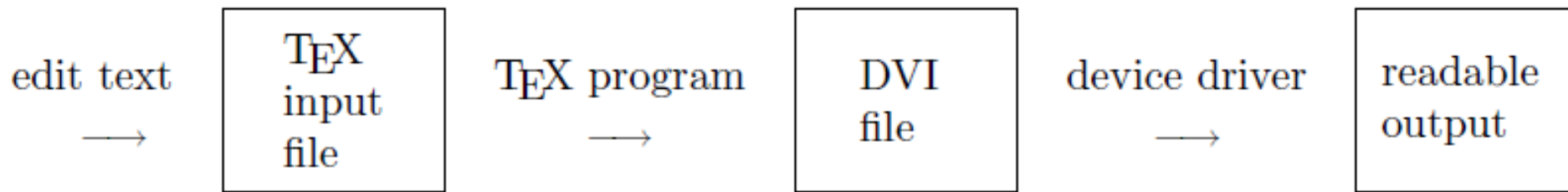
- [Notes based mostly on Oetiker et al. 2002]

How LATEX works

- LATEX needs you to provide additional information that describes the logical structure of your work: it needs "LATEX commands"

- This is VERY different from the WYSIWYG ("what you see if what you get") approach that most modern word processors such as MS Word or Corel WordPerfect use

How LATEX works

- With these, authors specify the document layout interactively while typing text into the computer and they can see on the screen how the final work will look when printed

- With LATEX you cannot see the final output while typing the text

- It can be previewed on the screen after processing the file with LATEX

The process may be thought of as proceeding in the following way:

edit text $\longrightarrow$ | T$_{\text{E}}$X input file | T$_{\text{E}}$X program $\longrightarrow$ | DVI file | device driver $\longrightarrow$ | readable output |

Advantages

- Professionally crafted layouts that make a document really look as if "printed"

- The typesetting of mathematical formulae is supported in a safe and convenient way

- The user only needs to learn a few easy-to-understand commands which specify the logical structure of a document, without having to tinker with the actual layout of the document

Advantages

- Footnotes, references, table of contents, equation numbers, pagination, and bibliographies can be generated easily

- Free (mostly everything is free with LATEX!) add-on packages exist for many typographical tasks not directly supported by basic LATEX

- For example, packages to typeset bibliographies conforming to exact standards

Advantages

- LATEX encourages you to write well-structured texts, by asking you to specify structure

(no more papers with missing author names, titles, dates, abstracts, etc.!!! Yay!!!)

- TEX, the formatting engine of LATEX, is highly portable and FREE (no more "your ∞ character looks like \%$3sj in my computer!!!")

- All these documents use very little computer memory too

Disadvantages

- Although some parameters can be adjusted within a predefined document layout, the design of a whole new layout is difficult and takes a lot of time

- It is very hard to write unstructured and disorganized documents

- It takes a bit of work to get used to it

- And initial frustration while you learn to debug your code

Disadvantages

- The snazziest (but expensive ☹) LATEX editors almost look like WORD and have extra bells and whistles like spellcheckers etc…

They also have more "pointers" to LATEX commands and environments, so you do not have to remember what to type

While the free ones tend to be more no frills, so they rely on your knowing exactly what to do

Some basics

It does not matter whether you enter one or several spaces after a word

An empty line starts a new paragraph

Some basics

Characters requiring special input

Character	Purpose	Input for literal output
\	Special symbols and instructions	`$\backslash$`
{	Open group	`$\{$`
}	Close group	`$\}$`
%	Comments	`\%`
&	Tabs and table alignments	`\&`
~	Unbreakable space	`\~{}`
$	Starting or ending math text	`\$`
^	Math superscripts	`\^{}`
_	Math subscripts	`\_{}`
#	Defining replacement symbols	`\#`

The backslash character \ cannot be entered by adding another backslash in front of it (\\), this sequence is used for linebreaking!!!

Some basics

LATEX commands are **case sensitive**

Two formats:

• a backslash \ and then have a name consisting of letters only

• a backslash and exactly one non-letter

Some basics

```
\section{My first fancy LATEX
   stuff.\label{Intro}}

I read that Knuth divides the
   people working with \TeX{} into
   \TeX{}nicians and \TeX perts.\\

Today is \today.  My fancy La\TeX\
   commands start in Section
   \ref{Intro}
```

Some basics

- Some commands use a parameter given in curly braces { } after the command name
- Some commands support optional parameters which are added after the command name in square brackets [ ]

Some basics

- `\begin{document}`

- `\begin{tabular}`

- `\begin{spacing}[1.7]`

- **Try:**

- `Our prof is quite funny \textsl{peculiar}, not so much funny \emph{haha}`

Some basics

- The symbol % will leave text out of your output, so it is great for comments for you and your coauthors

- Try:

- ```
  % This shabby table should never
  %get anywhere near the final
  %output!!!
  ```

Some basics

- The symbol $ is for mathematical text

- Try:

- I like my \alpha and \beta

- I like my α and β in \TeX

File structure

- Every input file must start with the command `\documentclass{...}`

- which specifies the kind of document you intend to write (article, letter, report, book thesis?).

- After that, commands that affect the style of the whole document

- you can load packages to add new features to the LATEX system: `\usepackage{...}`

File structure

start the body of the text with the command

`\begin{document}`

At the end of the document you add the

\end{document}

Anything which follows this

command will be ignored by LATEX

Try

```
\documentclass[a4paper,11pt]{article}
% define the title
\author{A. U. Thor}
\title{Minimalism}
\begin{document}
% generates the title
\maketitle
% insert the table of contents
\tableofcontents
\section{Start}
Well, and here begins my lovely article.
\section{End}
\ldots{} and here it ends.
\end{document}
```

Questions?

Any questions?

Any suggestions?

Any complaints?