LaTeX

PHSG SEM 3 AND 5

PAPER : SEC A1

SUNETRA DAS

Presentation #1 12-09-2023

- LATEX (PRONOUNCED "*LAY*-TEK" OR "*LAH*-TEK") IS A TOOL FOR TYPESETTING PROFESSIONAL-LOOKING DOCUMENTS.
- LATEX IS MOST OFTEN USED TO PRODUCE TECHNICAL OR SCIENTIFIC DOCUMENTS, BUT IT CAN BE USED FOR ALMOST ANY FORM OF PUBLISHING.
- TOOLS LIKE MS-WORD PROVIDE USERS WITH AN INTERACTIVE PAGE INTO WHICH THEY TYPE AND EDIT THEIR TEXT AND APPLY VARIOUS FORMS OF STYLING.
- LATEX WORKS VERY DIFFERENTLY: INSTEAD, YOUR DOCUMENT IS A PLAIN TEXT FILE INTERSPERSED WITH LATEX *COMMANDS* USED TO EXPRESS THE DESIRED (TYPESET) RESULTS. TO PRODUCE A VISIBLE, TYPESET DOCUMENT, YOUR LATEX FILE IS PROCESSED BY A PIECE OF SOFTWARE CALLED A *TEX ENGINE* WHICH USES THE COMMANDS EMBEDDED IN YOUR TEXT FILE TO GUIDE AND CONTROL THE TYPESETTING PROCESS, CONVERTING THE LATEX COMMANDS AND DOCUMENT TEXT INTO A PROFESSIONALLY TYPESET PDF FILE.

- SINCE LATEX COMPRISES A GROUP OF TEX COMMANDS, LATEX DOCUMENT PROCESSING IS ESSENTIALLY PROGRAMMING. YOU CREATE A TEXT FILE IN LATEX MARKUP, WHICH LATEX READS TO PRODUCE THE FINAL DOCUMENT.
- THIS APPROACH IS KNOWN AS WYSIWYM (WHAT YOU SEE IS WHAT YOU MEAN), HAS SOME DISADVANTAGES IN COMPARISON WITH A WYSIWYG (WHAT YOU SEE IS WHAT YOU GET) PROGRAM SUCH AS MICROSOFT WORD.
- VARIOUS ARGUMENTS CAN BE PROPOSED FOR, OR AGAINST, LEARNING TO USE LATEX INSTEAD OF OTHER DOCUMENT-AUTHORING APPLICATIONS; BUT, ULTIMATELY, IT IS A PERSONAL CHOICE BASED ON PREFERENCES, AFFINITIES, AND DOCUMENTATION REQUIREMENTS.

LATEX SYNTAX

- A MINIMAL EXAMPLE OF A LATEX SYNTAX LOOKS LIKE THE FOLLOWING:

```
\documentclass{article}

\begin{document}

Hello world!

\end{document}
```

- THE FIRST LINE OF CODE, \documentclass{article}, DECLARES THE DOCUMENT TYPE KNOWN AS ITS *CLASS*, WHICH CONTROLS THE OVERALL APPEARANCE OF THE DOCUMENT.

- DIFFERENT TYPES OF DOCUMENTS REQUIRE DIFFERENT CLASSES.
- OTHER TYPES OF DOCUMENTS YOU MAY BE WORKING ON MAY REQUIRE DIFFERENT CLASSES SUCH AS BOOK OR REPORT.
- HAVING SET THE DOCUMENT CLASS, THE CONTENT, KNOWN AS THE BODY OF THE DOCUMENT, IS WRITTEN BETWEEN \begin{document} AND \end{document} TAGS

OUTPUT

Hello world!

SPACES

- THE LATEX COMPILER NORMALISES WHITESPACE SO THAT WHITESPACE CHARACTERS, SUCH AS [SPACE] OR [TAB], ARE TREATED UNIFORMLY AS ”SPACE”: SEVERAL CONSECUTIVE ”SPACES” ARE TREATED AS ONE, ”SPACE” OPENING A LINE IS GENERALLY IGNORED, AND A SINGLE LINE BREAK ALSO YIELDS “SPACE”. A DOUBLE LINE BREAK (AN EMPTY LINE), HOWEVER, DEFINES THE END OF A PARAGRAPH; MULTIPLE EMPTY LINES ARE ALSO TREATED AS THE END OF A PARAGRAPH.

RESERVED CHARACTERS

- THE FOLLOWING SYMBOLS ARE RESERVED CHARACTERS THAT EITHER HAVE A SPECIAL MEANING UNDER LATEX,
- # % ^ & _ { } ~ \
- THESE CHARACTERS CAN BE USED IN DOCUMENTS BY ADDING A PREFIX BACKSLASH:
- \# \$ \% \^{} \& _ \{ \} \~{} \textbackslash{}
- THE BACKSLASH CHARACTER \ CANNOT BE ENTERED BY ADDING ANOTHER BACKSLASH IN FRONT OF IT (\\); THIS SEQUENCE IS USED FOR LINE BREAKING. FOR INTRODUCING A BACKSLASH IN MATH MODE, YOU CAN USE \backslash INSTEAD.
- THE COMMANDS \~ AND \^ PRODUCE RESPECTIVELY A TILDE AND A HAT WHICH IS PLACED OVER THE NEXT LETTER. FOR EXAMPLE \~N GIVES Ñ. THAT'S WHY YOU NEED BRACES TO SPECIFY THERE IS NO LETTER AS ARGUMENT.

LATEX COMMANDS

- LATEX COMMANDS ARE CASE SENSITIVE, AND TAKE ONE OF THE FOLLOWING TWO FORMATS:

THEY START WITH A BACKSLASH \ AND THEN HAVE A NAME CONSISTING OF LETTERS ONLY. COMMAND NAMES ARE TERMINATED BY A SPACE, A NUMBER OR ANY OTHER ”NON-LETTER”.

- THEY CONSIST OF A BACKSLASH \ AND EXACTLY ONE NON-LETTER.
- SOME COMMANDS NEED AN ARGUMENT, WHICH HAS TO BE GIVEN BETWEEN CURLY BRACES { } AFTER THE COMMAND NAME. SOME COMMANDS SUPPORT OPTIONAL PARAMETERS, WHICH ARE ADDED AFTER THE COMMAND NAME IN SQUARE BRACKETS [ ]. THE GENERAL SYNTAX IS:

\commandname[option1,option2,...]{argument1}{argument2}...

COMMENTS

- WHEN LATEX ENCOUNTERS A % CHARACTER WHILE PROCESSING AN INPUT FILE, IT IGNORES THE REST OF THE CURRENT LINE, THE LINE BREAK, AND ALL WHITESPACE AT THE BEGINNING OF THE NEXT LINE.

```
\documentclass{article} %this line mentions the class.

\begin{document} %beginning of body of the file.

Hello world! %this is the content of the body.

\end{document} %end of body of the file.
```

example: Hello world!

- NOTE THAT THE % CHARACTER CAN BE USED TO SPLIT LONG INPUT LINES THAT DO NOT ALLOW WHITESPACE OR LINE BREAKS, AS WITH SUPERCALIFRAGILISTICEXPIALIDOCIOUS ABOVE. THE CORE LATEX LANGUAGE DOES NOT HAVE A PREDEFINED SYNTAX FOR COMMENTING OUT REGIONS SPANNING MULTIPLE LINES. REFER TO MULTILINE COMMENTS FOR SIMPLE WORKAROUNDS.

OUR FIRST DOCUMENT

- OPEN YOUR FAVORITE TEXT-EDITOR. VIM, EMACS, NOTEPAD++, AND OTHER TEXT EDITORS WILL HAVE SYNTAX HIGHLIGHTING THAT WILL HELP TO WRITE YOUR FILES.
- REPRODUCE THE FOLLOWING TEXT IN YOUR EDITOR. THIS IS THE LATEX SOURCE.

```
% hello.tex - Our first LaTeX example!

\documentclass{article}

\begin{document}

Hello World!

\end{document}
```

- SAVE YOUR FILE AS hello.tex.
- WHEN PICKING A NAME FOR YOUR FILE, MAKE SURE IT BEARS A .tex EXTENSION.
- COMPILE.

`% hello.tex - Our first LaTeX example!`	The first line is a *comment*. This is because it begins with the percent symbol (%); when LaTeX sees this, it simply ignores the rest of the line. Comments are useful for people to annotate parts of the source file. For example, you could put information about the author and the date, or whatever you wish.
`\documentclass{article}`	This line is a command and tells LaTeX to use the `article` document class. A document class file defines the formatting, which in this case is a generic article format. The handy thing is that if you want to change the appearance of your document, substitute article for another class file that exists.
`\begin{document}`	This line is the beginning of the environment called document; it alerts LaTeX that content of the document is about to commence. Anything above this command is known generally to belong in the *preamble*.
`Hello World!`	This was the only actual line containing real content - the text that we wanted displayed on the page.
`\end{document}`	The document environment ends here. It tells LaTeX that the document source is complete, anything after this line will be ignored.

DOCUMENT STRUCTURE

- EVERY INPUT FILE MUST CONTAIN THE COMMANDS,

\documentclass{...}

\begin{document}

...

\end{document}

- THIS IS THE GLOBAL STRUCTURE OF A LaTeX FILE.

PREAMBLE

- THE AREA BETWEEN \documentclass{...} and \begin{document} IS CALLED THE PREAMBLE. IT NORMALLY CONTAINS COMMANDS THAT AFFECT THE ENTIRE DOCUMENT.
    1. DOCUMENT CLASSES: THE TYPE OF DOCUMENT IS SPECIFIED WITH THE *\documentclass* COMMAND: *\documentclass[options]{class}*
    2. *EXAMPLE: AN INPUT FILE FOR A LATEX DOCUMENT COULD START WITH THE LINE*

\documentclass[11pt,twoside,a4paper]{article}

WHICH INSTRUCTS LATEX TO TYPESET THE DOCUMENT AS AN ARTICLE WITH A BASE FONT SIZE OF 11 POINTS, AND TO PRODUCE A LAYOUT SUITABLE FOR DOUBLE SIDED PRINTING ON A4 PAPER.

Document Classes	
`article`	For articles in scientific journals, presentations, short reports, program documentation, invitations, ...
`IEEEtran`	For articles with the IEEE Transactions format.
`proc`	A class for proceedings based on the article class.
`report`	For longer reports containing several chapters, small books, thesis, ...
`book`	For real books.
`slides`	For slides. The class uses big sans serif letters.
`memoir`	For changing sensibly the output of the document. It is based on the `book` class, but you can create any kind of document with it `http://www.ctan.org/tex-archive/macros/latex/contrib/memoir/memman.pdf`
`letter`	For writing letters.
`beamer`	For writing presentations (see LaTeX/Presentations[1]).

DOCUMENT CLASS OPTIONS

Document Class Options	
`10pt, 11pt, 12pt`	Sets the size of the main font in the document. If no option is specified, 10pt is assumed.
`a4paper, letterpaper,...`	Defines the paper size. The default size is `letterpaper`; However, many European distributions of TeX now come pre-set for A4, not Letter, and this is also true of all distributions of pdfLaTeX. Besides that, `a5paper, b5paper, executivepaper`, and `legalpaper` can be specified.
`fleqn`	Typesets displayed formulas left-aligned instead of centered.
`leqno`	Places the numbering of formulas on the left hand side instead of the right.
`titlepage, notitlepage`	Specifies whether a new page should be started after the document title or not. The article class does not start a new page by default, while report and book do.
`twocolumn`	Instructs LaTeX to typeset the document in two columns instead of one.
`twoside, oneside`	Specifies whether double or single sided output should be generated. The classes `article` and `report` are single sided and the `book` class is double sided by default. Note that this option concerns the style of the document only. The option `twoside` does not tell the printer you use that it should actually make a two-sided printout.
`landscape`	Changes the layout of the document to print in landscape mode.
`openright, openany`	Makes chapters begin either only on right hand pages or on the next page available. This does not work with the `article` class, as it does not know about chapters. The `report` class by default starts chapters on the next page available and the `book` class starts them on right hand pages.

For example, if you want a report to be in 12pt type on A4, but printed one-sided:

\documentclass[12pt,a4paper,oneside]{report}