LaTeX– let's roll!

Radu Răzvan Slăvescu
România

October 2, 2017

Abstract

We will present how to write a `.tex` file and how to generate a `.pdf` from it. This includes writting text, equations, tables, algorithms and handling references. You need little more than one hour to understand the basics of LaTeX and one life to master it :)

1 Introduction

You should look in the `.tex` file (which is just a text file containing your text plus some formatting macros) as well as in the `.pdf` file generated from it. You will see plain content and some macros. A macro always starts with a backslash. For example, the macro `\section{...}` marks the beginning of a document section (of course, you should replace the dots with the section's actual name).

The comments are introduced by the % sign and will help you understand the meaning of the macros; the comment continues up to the end of line.

The commands needed for generating the `.pdf` are written in the file called `Makefile`; make sure the variables in its first lines are properly set. Then, all you need to do is to type `make` at the shell prompt. Alternatively, you can use LaTeX editors (see below) which provide the compiling commands.

1.1 Some basics

First, Romanian letters: ă î ș ț Ă Î Ș Ț â Â ü (even if the latter one does not seem to be in Romanian). In order to write them properly (use a comma instead of a cedilla for ș), you need to add `\usepackage{combelow}` to your document's preamble. Or you can add `\usepackage[utf8x]{inputenc}` to your document's preamble, do a `setxkbmap ro` in a terminal to change the keyboard layout to Romanian, then you can enter Romanian diacritics using `AltGr` (e.g., `AltGr+t` for ț).

We can use bullets:

- number one

- number two

Or we can use numbered items:

1. un (aka 1)

2. dos (aka 2)

To start a new paragraph, you just enter a blank line in the `.tex` file. The first paragraph will have no indentation. This is normal; do not attempt to alter this.

Some characters, like $, &, %, #, _, {, }, have special meanings in LATEX; if you want to escape them, place a backslash in front of them. In order to write a backslash in your `.pdf`, put `\backslash` in your `.tex` file. To write $<$ or $>$, you should enclose them between $ signs (`$<$` and `$>$` respectively).

When we cite a paper (e.g., paper [1]), we take its bibtex reference from the web, store it in the `.bib` file and label it there (e.g., `gigel`; this label must be unique in the `.bib` file). Then use `\cite{gigel}` in the `.tex` file and let LATEX handle ordering and cross-referencing. Each time you use an idea, text etc. borrowed from some other author, you must properly cite the source. It is a good time now to take a look in the `.bib` file and see how a bibtex entry of an article looks like. Then, you may want to search the web for the bibtex of an article (e.g., search bibtex "The Anatomy of a Large-Scale Hypertextual Web Search Engine"), grab it and add it to the `.bib` file.

The rest of this paper is structured as follows. Section 2 describes the implementation details, etc.

2 Implementation Details

If we have algorithms used/modified/implemented/introduced, we can describe them. For instance, algorithm 1. Describing algorithms requires studying the documentation of the `algorithmic` and `algorithm` packages.

Algorithm 1 An algorithm looks like this
---
if $Committed(G_1, GR, \alpha)$ **then**
  $BRT_\alpha = PredictBRT(G_1, GR, \alpha, C_{GR})$
  $C_\beta = ContextUpdate(C_\beta, o)$
end if
if $utility \geq CommunicationCost(G_2)$ **then**
  $Int.To(G_1, Communicate(G_1, G_2, o))$
end if

Table 1: Multinomial opinion multiplication

	belief			atomicity		
	poor	*avg*	*good*	*poor*	*avg*	*good*
success	0.1	0.2	0.3	0.4	0.5	0.6
failure	0.1	0.2	0.3	0.4	0.5	0.6

We can write equations:

$$\frac{S : a_m \wedge r_b \wedge d}{B : d_c \wedge ((d_e \wedge \neg d_{t_2} \wedge d_{t_{15}}) \vee (\neg d_e \wedge d_{t_2}))} \tag{1}$$

And big curly brackets:

$$\begin{cases} b = \frac{r}{r+s+2} \\ d = \frac{s}{r+s+2} \\ u = \frac{2}{r+s+2} \end{cases}$$

The notations for "n choose k" and sum look like C_n^k and $\sum_{k=0}^{n} C_n^k$ respectively.

3 Results

3.1 Tables

We summarize the results in a table, e.g., Table 1. Typically, the table is placed on top of the page where it is reffered for the first time, or on one of the consecutive pages. It is the LaTeX compiler decision.

3.2 Figures

We can also have figures, like Figure 1. Some sources claim they tell 1000 words/figure.

4 Conclusions

Now we know how to write text, tables, references etc.

Acknowledgments

Figure 1: Saying bye bye

A   How to run the example

The `v1.tex` file contains the source; edit it with

```
kile v1.tex
```

To generate the `.pdf` file, type

```
make
```

And no, we don't attempt to launch the `Makefile`. Just type `make` at the shell prompt and enjoy.

Alternatively, you may press the QuickBuild button in the Kile editor.

References

[1] C. Zhou, L.-T. Chia, and B.-S. Lee, "Daml-qos ontology for web services," in *ICWS '04: Proceeding's of the IEEE International Conference on Web Services.*   Washington, DC, USA: IEEE Computer Society, 2004, p. 472.