Introduction to LaTeX

Lecture 5: Advanced usages of LaTeX

Liu Yihao

SJTU-UMJI Technology Department

April 15, 2020

Newcommand

Sometimes you are building a huge project (like this lecture), and you may use certain type of syntax for many many times. Now it's time to define your own command with \newcommand in the beginning of the document (where the \usepackage commands appear).

Command

```
\newcommand{\yourcommand}[arg_num]{code}
```

- arg_num - number of arguments in your command
- code - the code of your command, use #1, #2, . . . , #n to represent the arguments

Example

```
\newcommand{\samplecommand}[1]{\textbackslash
#1}
```

It is defined to simply display the commands in red in

Renewcommand

Another times you need to redefine the commands, then \renewcommand can be used. It's very similar to \newcommand, the only difference is that you must use \newcommand when the command doesn't exists, while using \renewcommand when the command has been defined (by you or LaTeX packages) before.

Command

```
\renewcommand\definedcommand[arg_num]{code}
```

Example

```
\renewcommand\thesection{\Roman{section}}
\renewcommand\thesubsection{\Alph{subsection}}
```

By default, the number before the section titles of \section is 1, 2, 3, etc, this command will change them to a capital form of roman numbers, I, II, III, etc. And a subsection numbers become A, B, C, etc.

New/Renewenvironment

Environments can also be defined.

Command

```
\newenvironment{name}[arg_num]{begdef}{enddef}
\renewenvironment{name}[arg_num]{begdef}{enddef}
```

- name - the name of your environment
- arg_num - number of arguments in your environment
- begdef - the code to substitute the begin clause of your environment
- enddef - the code to substitute the end clause of your environment

Example

```
\newenvironment{command}{\begin{block}{Command}}{
```

Include and Input

When you are building a huge project, if you write all of the code in a single file, the compiling of the whole project will be very slow, and the length of the file will also confuse you. Then you can use \include and \input to avoid this.

Command

\include{file} - Include the file on a new page, the files are compiled separately.
\input{file} - Directly replace the command with the whole file, doesn't start a new page, but the compiling won't speed up.

If you are including a .tex file, then the extension name can be omitted. Another command \includeonly{list} can be added to the beginning of the document, so that only the include files in list are compiled and others are ignored, this is very useful in debugging huge projects.

Insert pdf

If you want to insert pdf files into your tex file, you can

Command

```
\usepackage{pdfpages}
\includepdf{file}
```

Chemical formulas

For typing formulas in LaTeX, besides equation environment, you can also use mhchem package. The documentation for mhchem can be referred at `http://ctan.mirrors.hoobly.com/macros/latex/contrib/mhchem/mhchem.pdf`

Hyperlink

Hyperlinks are supported in LaTeX, use the hyperref package.

Command

```
\usepackage{hyperref}
\hypersetup{options}
\url{url}
\href{url}{text}
```

Some common options are listed below:

- colorlinks - boolean (default false)
- urlcolor - color for linked URLs (default magenta)
- linkcolor - color for normal internal links (default red)