T_EX

Nguyễn Quản Bá Hồng*

December 22, 2022

Abstract

This text contains some personal notes on TEX. The latest version of this text & its TEX file have been stored & downloadable at the following link: GitHub/NQBH/hobby/TEX.pdf[1] & GitHub/NQBH/hobby/TEX.tex[2].

Contents

---

*Independent Researcher, Ben Tre City, Vietnam
e-mail: nguyenquanbahong@gmail.com; website: https://nqbh.github.io.

[1]URL: https://github.com/NQBH/hobby/blob/master/TeX/NQBH_TeX.pdf.

[2]URL: https://github.com/NQBH/hobby/blob/master/TeX/NQBH_TeX.tex.

Resources

1. The Comprehensive TEX Archive Network (CTAN) is the central place for all kinds of material around TEX. Most of the packages are free & can be downloaded & used immediately.

2. Overleaf – LATEX, Evolved: The easy to use, online, collaborative LATEX editor.

3. TEX-LATEX StackExchange is a question & answer site for users of TEX, LATEX, ConTeXt, & related typesetting systems.

1   `babel` Package

"This package manages culturally-determined typographical (& other) rules for a wide range of languages. A document may select a single language to be supported, or it may select several, in which case the document may switch from 1 language to another in a variety of ways. `babel` uses contributed configuration files that provide the detail of what has to be done for each language. Included is also a set of ini files for about 250 languages. Many language styles work with pdfLATEX, as well as with XeLATEX & LuaLATEX, out of the box. A few even work with plain formats." – CTAN/`babel` – Multilingual support for LATEX, LuaLATEX, XeLATEX, & Plain TEX

2   Overleaf/Chemistry Formulae

"There are a few LATEX packages to create chemistry formulae: chemfig, ochem, streetex, & xymtex. The most intuitive is probably the `chemfig` package. This article explains how to use the `chemfig` package to create chemical formulae in LATEX."

2.1   Introduction

"Drawing a molecule consists mainly of connecting groups of atoms with lines. Simple linear formulae can be easily drawn using the `chemfig` package, as shown in the following example:

```
\documentclass{article}
\usepackage{chemfig}
\begin{document}

\section{Introduction}
Writing chemical formulae with chemfig is straightforward.

\chemfig{O=H}
\end{document}
```

The package is imported by `\usepackage{chemfig}` in the preamble. The command `\chemfig{O=H}` the draws the molecule. The symbol = determines the type of bond."

2.2   Angles

"There are several ways to define angles between bonds in molecules.

```
To define chemical formulae you can use units that define the angles
```

```
\chemfig{A-[1]B-[7]C}
```

```
Absolute angles
```

```
\chemfig{A-[:50]B-[:-25]C}
```

```
Relative angles
```

```
\chemfig{A-[::50]B-[::-25]C}
```

Each 1 of the 3 commands in the example above uses a different method to determine the angle between bonds. • `default units`: In the command `\chemfig{A-[1]B-[7]C}` the parameters inside brackets set the angle in special units, each unit equals 45°. Hence in the example the angles are 45° & 315°. • `absolute units`: The angles can be set in absolute units, in the command `\chemfig{A-[:50]B-[:-25]C}` the parameter inside the brackets represent the angle, in degrees, measured from the horizontal baseline. Negative angles are allowed. • `relative angles`: In the 3rd example `\chemfig{A-[::50]B-[::-25]C}` the angles are measured from the previous bond, instead of the baseline."

2.3   Rings

2.3.1   Connected rings

2.4   Branches

"Chemical formulae are not always linear, branched formulae are actually the most common type. Below an example on how to create them.

```
Branched molecule \vspace{.5cm}
```

```
\chemfig{H-C(-[2]H)(-[6]H)-C(=[1]O)-[7]H}
```

Branches in each node are created by adding formulas inside parentheses. E.g., the code `C(-[2]H)(-[6]H)` creates 2 branches in `C`, one with a 2 units angle (90°) & other with a 6 units angle (270°). Branches are also be added to rings:

```
Branched ring
\vspace{.5cm}
```

```
\chemfig{A*6(-B=C(-CH_3)-D-E-F(=G)=)}
```

The syntax is similar, using parentheses a branch can be attached to a node (atom). E.g., `F(=G)` attaches a branch to the node `F`. More complex examples can be created using nested branches & even attaching rings as branches."

2.5   Customizing the formulae

"Several parameters such as colors & the node separation can be changed, also additional text to describe the formula can be added.

```
{\huge
\setchemfig{atom sep=2em,bond style={line width=1pt,red,dash pattern=on 2pt off 2pt}}
\chemname
{\chemfig{H-C(-[2]H)(-[6]H)-C(=[1]O)-[7]H}}
{Ethanal}
}
```

There are 3 new commands here: • `\setbondstyle{ }`: Inside the braces several style-related commands can be passed using the `<code<tikz` syntax. • `\setatomstep{2em}`: The separation between atoms (nodes) in the formula is set to `2em`. Other LATEX units can be used. • `\chemname{}{}`: The 1st parameter in this command is a `chemfig` formula, the 2nd one is some text that will be printed below the formula. In the example the text is "Ethanal".

Notice also that the font used is `huge`. You can use any other font size & the formula will be scaled accordingly."

2.6   Reference guide

3   Overleaf/International Language Support

"LATEX supports many worldwide languages by means of some special packages."

3.1 Introduction

"If you are a non-English speaker, LaTeX can be configured to typeset in your language." [...] "The package that makes possible to display special characters is `babel`, this package also changes the language of the elements in the document. In the example instead of "abstract" & "Contents" the Spanish words "resumen" & "Índice" are used."

3.2 Input encoding

"Modern computer systems allow you to input letters of national alphabets directly from the keyboard. In order to handle a variety of input encodings used for different groups of languages &/or on different computer platforms LaTeX employs the `inputenc` package to set up input encoding. To use this package, add the next line to the *preamble* of your document:

`\usepackage[encoding]{inputenc}`

The recommended input encoding is `utf8`, which supports a lot of national alphabets letter (inside the brackets, instead of the word "encoding" you must put the name of the encoding you are using). If you want, you can also use other encodings connected with different groups of languages &/or on different computer platforms."

OS	Western European Latin encoding	Central European Latin encoding	Cyrillic encoding
Windows	`cp1252`	`cp1250`	`cp1251`
GNU/Linux & Unix-like (*BSD, Mac OS X)	`latin1`	`latin2`	`koi8-ru`
Recommended for all systems	`utf8`	`utf8`	`utf8`

Remark 3.1. *"If you can't input some letters of national alphabets directly from the keyboard, you can use LaTeX alternative commands for accents & special characters."*

3.3 Font encoding

"To proper LaTeX document generation you must also choose a font which has to support specific characters for a given language by using `fontenc` package:

`\usepackage[encoding]{fontenc}`

The default LaTeX font encoding is `OT1`, but it contains only 128 characters. The `T1` encoding contains letters & punctuation characters for most of the European languages using Latin script. For languages using Cyrillic script you can use `T2A`, `T2B`, `T2C`, or `X2` font encodings."

3.4 `babel`

"The `babel` package allows to use special characters & also translates some elements within the document. This package also automatically activates the appropriate hyphenation rules for the language you choose. You can activate the babel package by adding the next command to the preamble:

`\usepackage[language]{babel}`

Change the `language` to the name of the language you need. You can see list of the languages available in the babel package documentation, under Sect. 1.26 "Languages supported by `babel` with ldf files"."

3.5 Using ≥ 2 language in a document

"`babel` command can be called with multiple languages" e.g.,

```
\documentclass{article}
\usepackage[utf8]{inputenc}
\usepackage[english, russian]{babel}
\usepackage[T1, T2A]{fontenc}
...
```

"Notice at the preamble that 2 encodings & 2 languages are passed as parameters to the `fontenc` & `babel` packages respectively. When using this syntax the last language in the option list will be active (i.e. Russian), & you can use the command `\selectlanguage{english}` at any point to change the active language."

3.6 Right-to-left writing

3.6.1 Arabic language

"The `arabic` package provides the Right-to-Left scripts support for LaTeX without the need of any external preprocessor. You can include the `arabtex` package for extended capabilities when working with documents in Arabic or Hebrew. If you need to insert latin text inside the arabic text use `\textLR{Latin text}`."

```
\documentclass[11pt,a4paper]{report}
\usepackage{arabtex}
\usepackage[utf8]{inputenc}
\usepackage[LFE,LAE]{fontenc}
\usepackage[arabic]{babel}
...
```

3.7 Examples of Supported Languages

Arabic, Chinese, French, German, Greek, Italian, Japanese, Korean, Portuguese, Russian, Spanish (with links & examples).

3.8 Reference guide

3.8.1 Accents & special characters

"If you can't input some letters of national alphabets directly from the keyboard, you can use LaTeX commands for accents & special characters." See Overleaf/international language support/reference guide for a list.

4 Overleaf/multiple columns

4.1 Introduction

"2-column documents can be easily created by passing the parameter `\twocolumn` to the document class statement. If you need more flexibility in the column layout, or to create a document with multiple columns, the package `multicol` provides a set of commands for that. This article explains how use the `multicol` package, starting with this basic example:

```
\documentclass{article}
\usepackage{blindtext}
\usepackage{multicol}
\title{Multicols Demo}
\author{Overleaf}
\date{April 2021}

\begin{document}
\maketitle

\begin{multicols}{3}
[
\section{First Section}
All human things are subject to decay. And when fate summons, Monarchs must obey.
]
\blindtext\blindtext
\end{multicols}

\end{document}
```

To import the package, the line

```
\usepackage{multicol}
```

is added to the preamble. Once the package is imported, the environment `multicols` can be used. The environment takes 2 parameters:

- Number of columns. This parameter must be passed inside braces, & its value is 3 in the example.

- "Header text", which is inserted in between square brackets. This is optional & will be displayed on top of the multicolumn text. Any LaTeX command can be used here, except for floating elements e.g. figures & tables. In the example, the section title & a small paragraph are set here.

The text enclosed inside the tags `\begin{multicols}` & `\end{multicols}` is printed in multicolumn format."

4.2   Column separation

"The column separation is determined by `\columnsep`. See the example below:

```
\documentclass{article}
\usepackage{blindtext}
\usepackage{multicol}
\setlength{\columnsep}{1cm}
\title{Second multicols Demo}
\author{Overleaf}
\date{April 2021}

\begin{document}
\maketitle

\begin{multicols}{2}
[
\section{First Section}
All human things are subject to decay. And when fate summons, Monarchs must obey.
]
\blindtext\blindtext
\end{multicols}

\end{document}
```

Here, the command `\setlength{\columnsep}{1cm}` sets the column separation to 1cm. See Lengths in LaTeX for a list of available units."

4.3   Unbalanced columns

"In the default `multicols` environment the columns are balanced so each one contains the same amount of text. This default format can be changed by the stared environment `multicols*`:

```
\documentclass{article}
\usepackage{blindtext}
\usepackage{multicol}
\setlength{\columnsep}{1cm}
\title{Second multicols Demo}
\author{Overleaf}
\date{April 2021}
\begin{document}
\maketitle
\begin{multicols*}{3}
[
\section{First Section}
All human things are subject to decay. And when fate summons, Monarchs must obey.
]
\blindtext\blindtext
\end{multicols*}

\end{document}
```

If you open this example on Overleaf you'll see that the text is printed in a column till the end of the page is reached, then the in continues in the next column, & so on."

4.4   Inserting floating elements

"Floating elements (tables & figures) can be inserted in a multicolumn document with `wrapfig` & `wraptable`.

```
\begin{multicols}{2}
[
\section{First Section}
All human things are subject to decay. And when fate summons, Monarchs must obey.
]


Hello, here is some text without a meaning.  This text should show what
a printed text will look like at this place.
If you read this text, you will get no information.  Really?  Is there
no information?  Is there.

\vfill

\begin{wrapfigure}{l}{0.7\linewidth}
\includegraphics[width=\linewidth]{overleaf-logo}
\caption{This is the Overleaf logo}
\end{wrapfigure}

A blind text like this gives you information about the selected font, how
the letters are written \& an impression of the look.  This text should
contain all...

\begin{wraptable}{l}{0.7\linewidth}
\centering
\begin{tabular}{|c|c|}
\hline
Name & ISO \\
\hline
Afghanistan & AF \\
Aland Islands & AX \\
Albania     &AL  \\
Algeria    &DZ \\
American Samoa & AS \\
Andorra & AD    \\
Angola & AO \\
\hline
\end{tabular}
\caption{Table, floating element}
\label{table:ta}
\end{wraptable}

\end{multicols}

\end{document}
```

Floats in the `multicol` package are poorly supported in the current version. Elements inserted with the conventional `figure*` & `table*` environments will show up only at the top or bottom of the next page after they are inserted, & will break the layout. The example presented here is a workaround, but you may expect some rough edges. E.g., if the float width is set to `\linewidth` it causes a weird text overlapping. This said, below is a brief description of the commands:

- `\usepackage{wrapfig}`. Put this line in the preamble to import the package `wrapfig`

- The environment `wrapfigure` will insert a figure wrapped in the text. For more information & further examples about this environment see Positioning images & tables.

- The environment `wraptable` is the equivalent to `wrapfigure` but for tables. See Positioning images & tables for more information."

4.5   Inserting vertical rulers

"A vertical ruler can be inserted as column separator to may improve readability in some documents:

```
\documentclass{article}
\usepackage{blindtext}
\usepackage{multicol}
\usepackage{color}
\setlength{\columnseprule}{1pt}
\def\columnseprulecolor{\color{blue}}

\begin{document}

\begin{multicols}{3}
[
\section{First Section}
All human things are subject to decay. And when fate summons, Monarchs must obey.
]
Hello, here is some text without a meaning.  This text should show what
a printed text will look like at this place.

If you read this text, you will get no information.  Really?  Is there
no information?  Is there.

\columnbreak
\blindtext
This will be in a new column, here is some text without a meaning.  This text
should show what a printed text will look like at this place.

If you read this text, you will get no information.  Really?  Is there
no information?  Is there...
\end{multicols}

\blindtext

\end{document}
```

If you open this example on Overleaf you will see the column separator can be set to a specific color also. Below a description of each command:

```
\usepackage{color}
```

This line is inserted in the preamble to enable the use of several colors within the document.

```
\setlength{\columnseprule}{1pt}
```

This determines the width of the ruler to be used as column separator, it's set to 0 by default. In the example a column whose width is 1pt is printed.

```
\def\columnseprulecolor{\color{blue}}
```

The color of the separator ruler is set to blue. See the article about using colors in LaTeX for more information on color manipulation.

```
\columnbreak
```

This command inserts a column breakpoint. In this case, the behaviour of the text is different from what you may expect. The column break is inserted, then the paragraphs before the breakpoint are evenly distributed to fill all available space. In the example, the 2nd paragraph is at the bottom of the column & a blank space is inserted in between the 2nd & the 1st paragraphs."

5 Overleaf/TikZ Package

5.1 Introduction

"TikZ is probably the most complex & powerful tool to create graphic elements in LaTeX. Starting with a simple example, this article introduces some basic concepts: drawing lines, dots, curves, circles, rectangles, etc. 1stly, load the `tikz` package by including `\usepackage{tikz}` in the preamble of your document, then draw a graphic using the `tikzpicture` environment.

```
\documentclass{article}
\usepackage{tikz}
\begin{document}
\begin{tikzpicture}
\draw[gray, thick] (-1,2) -- (2,-4);
\draw[gray, thick] (-1,-1) -- (2,2);
\filldraw[black] (0,0) circle (2pt) node[anchor=west]{Intersection point};
\end{tikzpicture}
\end{document}
```

This example produces the following output:
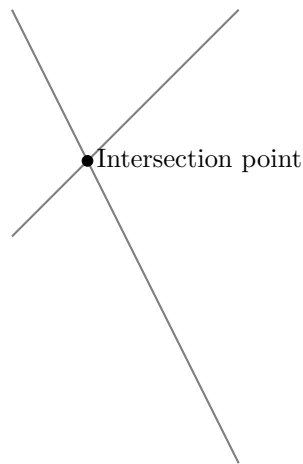
In this example 2 lines & 1 point are drawn. To add a line the command `\draw[gray, thick]` defines a graphic element whose *color* is gray & with a `thick` stroke. The line is actually defined by it's 2 endpoints, `(-1,2)` & `(2,-4)`, joined by `--`.

The *point* is actually a `circle` drawn by `\filldraw[black]`, this command will not only draw the circle but fill it using black. In this command the center point `(0,0)` & the radius `(2pt)` are declared. Next to the point is a node, which is actually a box containing the text `Intersection point`, & anchored at the `west` of the point. It's important to notice the semicolon `;` used at the end of each `draw` command.

Note 5.1. *The `tikzfigure` environment can be enclosed inside a figure or similar environment."*
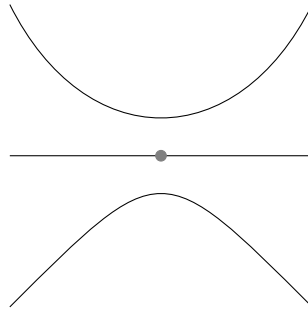
5.2 Basic Elements: Points, Lines, & Paths

Provide some examples showing how to create some basic graphic elements which can be combined to create more elaborate figures.

```
\documentclass{article}
\usepackage{tikz}
\begin{document}
\begin{tikzpicture}

\draw (-2,0) -- (2,0);
\filldraw [gray] (0,0) circle (2pt);
\draw (-2,-2) .. controls (0,0) .. (2,-2);
\draw (-2,2) .. controls (-1,0) and (1,0) .. (2,2);

\end{tikzpicture}
\end{document}
```

This example produces the following output:

There are 3 basic commands in this example: • \draw (-2,0) -- (2,0);: This defines a line whose endpoints are (-2,0) & (2,0). • \filldraw [gray] (0,0) circle (2pt);: The point is created as a very small gray circle centered at (0,0) & whose radius is (2pt). The \filldraw command is used to draw elements & fill them with a specific color. • \draw (-2,2) .. controls (-1,0) and (1,0) .. (2,2);: Draw a Bézier curve. There are 4 points defining it: (-2,2) & (2,2) are its endpoints, (-1,0) & (1,0) are *control points* that determine "how curved" it is. You can think of these 2 points as "attractor points"."

5.3   Basic Geometric Shapes: Circles, Ellipses, & Polygons

"Geometric figures can be constructed from simpler elements so let's start with circles, ellipses, & arcs.

```
\documentclass{article}
\usepackage{tikz}
\begin{document}
\begin{tikzpicture}
\filldraw[color=red!60, fill=red!5, very thick](-1,0) circle (1.5);
\fill[blue!50] (2.5,0) ellipse (1.5 and 0.5);
\draw[ultra thick, ->] (6.5,0) arc (0:220:1);
\end{tikzpicture}
\end{document}
```

This example produces the following output:
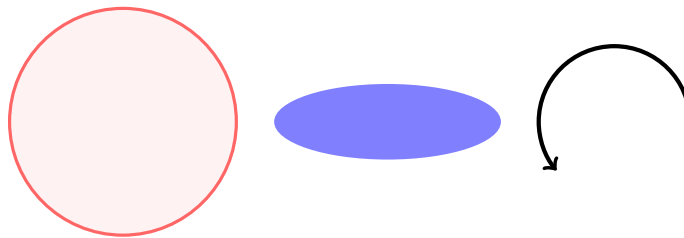
• \filldraw[color=red!60, fill=red!5, very thick](-1,0) circle (1.5);: This command was used in the previous section to draw a *point*, but in this instance there are some additional parameters inside the brackets. These are explained below:

5.4   Diagrams

6   Overleaf/typesetting quotations

6.1   Introduction

"When it comes to quotations & quotation marks, each language has its own symbols & rules.

6.2   `dirtytalk` **package**

6.3   `csquotes` **package**

6.4   `epigraph` **package**

6.5   `fancychapters` **package (obsolete)**

6.6   `quotchap` **package**

6.7   **Reference guide**

7   Vanilla TEXLive

UBUNTU does not pre-install Vanilla TEXLive, you need to install it manually & additionally.