

L^AT_EX Experiments

AeAeA

January 24, 2020

1 TeX distributions

1.1 MacTeX

The best for Mac.

```
$ brew cask install mactex
```

1.2 Visual Studio Code LaTeX Workshop Extension

LaTeX Workshop is an extension for Visual Studio Code, aiming to provide core features for LaTeX typesetting with Visual Studio Code.

- <https://github.com/James-Yu/LaTeX-Workshop>
- <https://github.com/James-Yu/LaTeX-Workshop/wiki/Compile>

Build LaTeX file by calling the command **Build LaTeX project** from the Command Palette or from the TeX badge. This command is bound to **Cmd+Ctrl+b**

You can change VS Code settings by opening Settings tab:

Cmd+, -> Extensions -> LaTeX

or, alternatively, by directly editing settings.json file:

`~/Library/Application\ Support/Code/User/settings.json`

Recommended settings for LaTeX Workshop:

```
{
  "latex-workshop.view.pdf.viewer": "tab",
  "latex-workshop.latex.outDir": "%DIR%/texout",
  "latex-workshop.latex.autoBuild.run": "never",
  "latex-workshop.latex.autoClean.run": "onBuilt"
}
```

1.3 MiKTeX

Not for Mac. Old MiKTeX installation:

`/usr/local/bin/`

`/Applications/MiKTeX\ Console.app/`

1.4 TinyTeX

TinyTeX - a lightweight, cross-platform, portable, and easy-to-maintain L^AT_EX distribution based on TeX Live.

Currently TinyTeX works best for R users. Installing and maintaining TinyTeX is easy for R users, since the R package `tinytex` has provided wrapper functions.

For other (non-R) users:

- TinyTeX docs: <https://yihui.org/tinytex/>
- In the directory
~/Library/TinyTeX/texmf-dist/tex/latex/
you can find all L^AT_EX packages installed for TinyTeX.
- If you compile a LaTeX document and run into an error message like this:
! LaTeX Error: File 'times.sty' not found.
It basically indicates a missing LaTeX package.
Use the command `tlmgr search` to find the name of the missing package:

```
$ tlmgr search --global --file "/times.sty"
psnfss: texmf-dist/tex/latex/psnfss/times.sty
```


In this case, the missing package is `psnfss`, and we can install a package via `tlmgr install`, e.g.,

```
$ tlmgr install psnfss
```


If you still see error messages that you don't understand, you may need to update everything:

```
$ tlmgr update --self --all
$ tlmgr path add
$ fmtutil-sys --all
```
- To uninstall TinyTeX use command line:

```
$ tlmgr path remove
$ rm -r "~/Library/TinyTeX"
```

2 Epigraph

In doing what we ought we deserve no praise, because it is our duty.

— Saint Augustine

2.1 Online docs

- <https://en.wikibooks.org/wiki/LaTeX> L^AT_EXwiki (very informative).
- <http://texdoc.net/> TeXdoc is a T_EX and L^AT_EX documentation lookup system.

2.2 Units and page layout

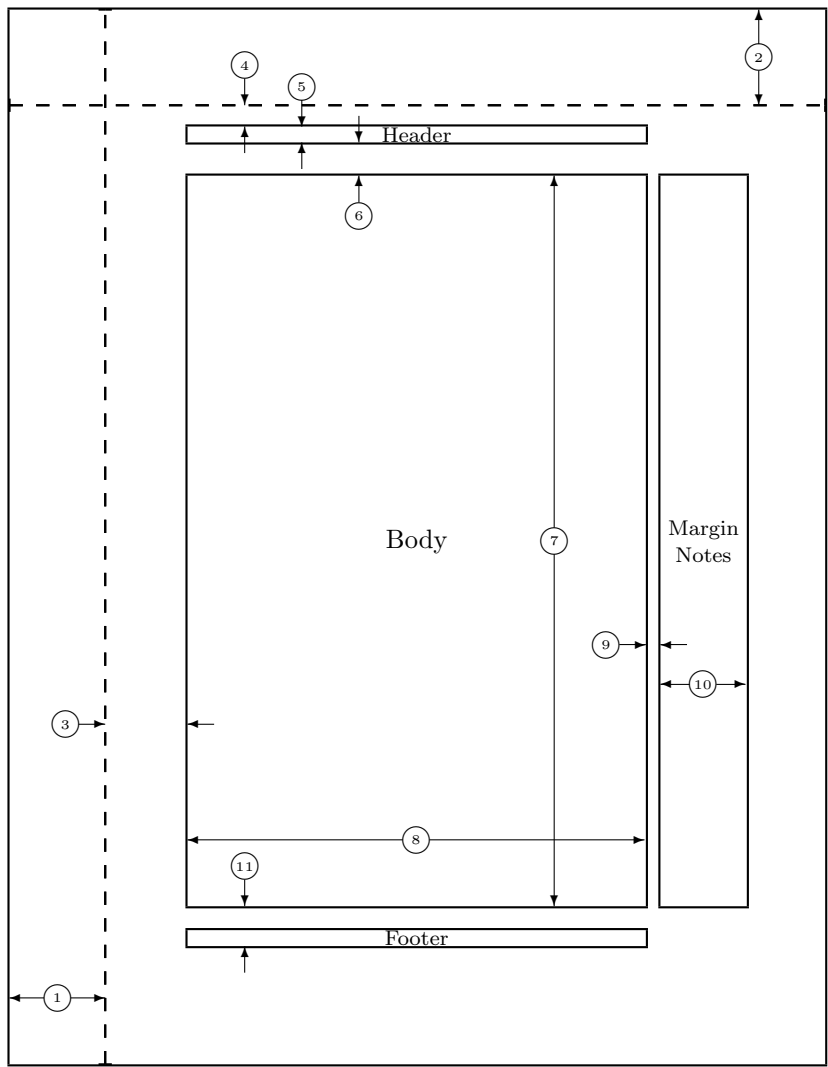
- https://en.wikibooks.org/wiki/LaTeX/Page_Layout

Standard L^AT_EX units: mm, cm, pt, in, with $1\text{in} = 72.27\text{pt}$ and $1\text{pt} = 0.3515\text{mm}$ and $1\text{mm} = 2.8445\text{pt}$.

US Letter (letterpaper) is 8.5 x 11 in, 215.9 x 279.4 mm, 614.295 x 794.97 pt, aspect ratio 1.294.

A4 (a4paper) is 8.3 x 11.7 in, 210 x 297 mm, 597.44 x 844.95 pt, aspect ratio 1.414 ($\approx \sqrt{2}$).

The current page layout picture below is generated by calling command `\layout*` from `\usepackage{layout}`.¹



- | | |
|-------------------------|----------------------------------|
| 1 one inch + \hoffset | 2 one inch + \voffset |
| 3 \oddsidemargin = 62pt | 4 \topmargin = 16pt |
| 5 \headheight = 12pt | 6 \headsep = 25pt |
| 7 \textheight = 550pt | 8 \textwidth = 345pt |
| 9 \marginparsep = 11pt | 10 \marginparwidth = 65pt |
| 11 \footskip = 30pt | \marginparpush = 5pt (not shown) |
| \hoffset = 0pt | \voffset = 0pt |
| \paperwidth = 614pt | \paperheight = 794pt |

¹An example footnote.

Margin note:
 $1in = 72.27pt$
 $1pt = 0.35mm$
 $1mm = 2.84pt$

3 verbatim and listings

3.1 verbatim

Text enclosed inside

```
\begin{verbatim} ... \end{verbatim}
```

environment is printed directly
and all `\LaTeX{}` commands are ignored.

Text enclosed inside `\begin{verbatim*}` environment
is printed directly
and all `\LaTeX{}` commands are ignored,
and white spaces are emphasized with a special symbol.

Use `\verb+<inline verbatim text>+` like this:

The `\ldots` command produces ...

3.2 listings: Source code printing

- listings package documentation
- https://www.overleaf.com/learn/latex/Code_listing

3.2.1 minimal setup

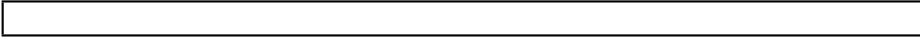
Example of using the `\begin{lstlisting}[language=Python]` environment
from the `\usepackage{listings}` package to highlight Python code:

```
import numpy as np

def incmatrix(genl1, genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m, 1), int) #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2), 1)

    for i in range(m-1):
        for j in range(i+1, m):
            [r, c] = np.where(M2 == M1[i, j])
            for k in range(len(r)):
                VT[(i)*n + r[k]] = 1;
                VT[(i)*n + c[k]] = 1;
                VT[(j)*n + r[k]] = 1;
                VT[(j)*n + c[k]] = 1;
```



```
        if M is None:
            M = np.copy(VT)
        else:
            M = np.concatenate((M, VT), 1)

    VT = np.zeros((n*m,1), int)

    return M
```

3.2.2 with code styles and colours

You need `\usepackage{xcolor}` package for the code colouring.

Just like in floats (tables and figures), captions can be added to a listing for a more clear presentation. This caption can be later used in the list of Listings `\lstlistoflistings`.

```
1 import numpy as np
2
3 def incmatrix(genl1,genl2):
4     m = len(genl1)
5     n = len(genl2)
6     M = None #to become the incidence matrix
7     VT = np.zeros((n*m,1), int) #dummy variable
8
9     s = "codepurple"
10
11     #compute the bitwise xor matrix
12     M1 = bitxormatrix(genl1)
13     M2 = np.triu(bitxormatrix(genl2),1)
14
15     for i in range(m-1):
16         for j in range(i+1, m):
17             [r,c] = np.where(M2 == M1[i,j])
18             for k in range(len(r)):
19                 VT[(i)*n + r[k]] = 1;
20                 VT[(i)*n + c[k]] = 1;
21                 VT[(j)*n + r[k]] = 1;
22                 VT[(j)*n + c[k]] = 1;
23
24             if M is None:
25                 M = np.copy(VT)
26             else:
27                 M = np.concatenate((M, VT), 1)
28
29             VT = np.zeros((n*m,1), int)
30
31     return M
```

Listing 1: Python example

Listings		
1	Python example	6

