



中国海洋大学
OCEAN UNIVERSITY OF CHINA

第一周

实 验 报 告

学 院： 信息科学与工学部

班 号： 计科一班

姓 名： 顾晓宁

学 号： 24020007036

实验编号： 第一周实验报告

指导教师： 周小伟

目录

1 实验目的 3

2 练习内容与结果 3

2.1 LaTeX 练习实例 3

2.1.1 LaTeX的基本区域、以及最基础的框架是? 3

2.1.2 常用的设置文字的设置有? 3

2.1.3 常用的章节命令有? 4

2.1.4 LaTeX特殊符号有? 且如何显示在文本中? 4

2.1.5 如何换行、换段、换页、首行缩进? 4

2.1.6 LaTeX的box如何使用? 5

2.1.7 如何插入图片? 5

2.1.8 如何插入列表? 6

2.1.9 如何插入数学公式? 6

2.1.10 如何插入表格? 7

2.1.11 如何引用参考文献? 8

2.1.12 除文献外, 如何引用表格、图片等内容? 9

2.1.13 如何插入代码? 9

2.1.14 如何插入链接? 10

2.1.15 LaTeX模板和页边距等如何设置? 10

2.2 Git 练习实例 11

2.2.1 从历史中删除文件 11

2.2.2 从 GitHub 上克隆某个仓库, 修改一些文件。当您使用 git stash 会发生什么? 当您执行 git log -all --oneline 时会显示什么? 通过 git stash pop 命令来撤销 git stash 操作, 什么时候会用到这一技巧? 12

2.2.3 如何在git中创建别名? 12

2.2.4 git如何忽略文件? 13

2.2.5 git使用流程? 13

3 实验感悟 13

4 参考文献示例, 仅用于演示 13

5 个人github账号 14

6 相关资料 14

第一周实验：Git、LaTeX

1 实验目的

练习使用Git进行版本控制，掌握LaTeX的基本使用方法

2 练习内容与结果

注意，以下内容均为练习实例，一般不含解释；语法等具体知识点看参照文末相关资料
同时该练习实例较为简陋，一般用于根据模板编写LaTeX，倘若对格式有更高或者自定义需求，建议系统、完整地参阅文档学习

2.1 LaTeX 练习实例

2.1.1 LaTeX的基本区域、以及最基础的框架是？

在LaTeX中，文档的基本结构包括导言区和正文区。导言区用于设置文档的类型、加载宏包和定义自定义命令，而正文区则包含实际的内容。

```
1 \documentclass[UTF8]{ctexart}
2 \title{title}
3 \author{author}
4 \date{\today}
5 % above is preamble that sets up the document
6 \begin{document}
7 \maketitle
8 This is the content of the document.
9 \end{document}
```

2.1.2 常用的设置文字的设置有？

```
1 \textbf{Bold text}
2 \textit{Italic text}
3 \underline{Underlined text}
```

展示：

Bold text

Italic text

Underlined text

2.1.3 常用的章节命令有？

part>chapter>section>subsection>subsubsection

```
1 \section{Level 1 Heading}
2 \subsection{Level 2 Heading}
3 \subsubsection{Level 3 Heading}
4 \chapter{Chapter Heading} % Only in book/report class
5 \part{Part Heading} % Only in book/report class
```

2.1.4 LaTeX特殊符号有？且如何显示在文本中？

特殊字符如下：

%	注释符号
&	表格对齐符号
\$	数学模式标记
~	非断行空格
^ 和 _	上下角标
{ 和 }	分组符号
#	宏定义符号
\	转义符号
\\	换行符

表 1: 特殊字符

```
1 \verb|char|
```

该命令逐字打印，用于显示LaTeX代码、特殊字符等，而不被LaTeX解释。

|是分隔符，可以用任何字符，如'（但不能是要显示的文本中包含的字符）

分隔符内的所有内容都会原样输出。

2.1.5 如何换行、换段、换页、首行缩进？

```
1 \\ % to newline
2 \newline \linebreak % also to newline
3 \par % to new paragraph
4 \newpage % to new page
5 \setlength{\parindent}{length} %first line indentation
```

展示：

这是第一段文本。

这是第二段文本，通过 \par 命令分隔。

这是第三段文本，使用空行分段（效果与 \par 相同）。

这是第四段文本。

这是新的一页内容。分页后从这里开始。
这段没有首行缩进。

这段有更大的缩进。
此外，常用的缩进命令有：

```
1 \hspace{1cm} % 1cm Horizontal indentation
2 \quad % 1 space, which equal to 1em
3 \qquad % 2 space
4 \vspace{10pt} downward vertical indentation
5 \vspace{-10pt} above vertical indentation
```

2.1.6 LaTeX的box如何使用？

```
1 \hbox{text}
2 \hbox to width{text}
3 \vbox{text}
4 \makebox[width][align]{text} % The most commonly used
5 \framebox[width][align]{text}
6 % align : l\c\r\s , s:scatter
```

`\parbox[基线对齐位置tcb][height][内部对齐]{宽度}{内容}`

展示，用四个makebox展示左中右分散对齐，1个framebox，1个parbox：
左对齐

居中

右对齐

分 散 对 齐

带框盒子

这是一个较长的段落文本，会在指定的5厘米宽度内自动换行。

2.1.7 如何插入图片？

```
1 \usepackage{graphicx} % in preamble
2 % then in document;and Omit some content
3 \begin{figure}[htbp]
4     \centering
5     \includegraphics[width=0.5\textwidth]{path/to/image}
6     \caption{Caption text}
7     \label{fig:label}
8 \end{figure}
```

展示：

```

图片：
在导言区
\usepackage{graphicx} //包含包，内含常用图片操作指令
正文
\includegraphics[可选参数]{图片名}

```

图 1: 图片示例

2.1.8 如何插入列表？

列表有无序和有序之分

```

1 % Unordered list
2 \begin{itemize}
3     \item First item
4     \item Second item
5     \item Third item
6 \end{itemize}
7 % Ordered list
8 \begin{enumerate}
9     \item First item
10    \item Second item
11    \item Third item
12 \end{enumerate}

```

展示：

无序

- First item
- Second item
- Third item

有序

1. First item
2. Second item
3. Third item

2.1.9 如何插入数学公式？

```

1 $E=mc^2$ % Inline math mode
2
3 \begin{equation}
4 E=mc^2 % Display math mode with number
5 \end{equation}
6

```

```

7 \[
8 a^2 + b^2 = c^2 % Display math mode without number
9 \]
10 % or use $$ math mode $$
11
12 % if too long
13 \begin{split}
14 a &= b + c + d + e + f + g + h + i + j + k + l + m + n + o + p \\
15 &\quad + q + r + s + t + u + v + w + x + y + z
16 \end{split}
17
18 % discuss case by case
19 f(x)=
20 \begin{cases}
21 1, & \text{if } x > 0 \\
22 0, & \text{if } x = 0 \\
23 -1, & \text{if } x < 0
24 \end{cases}

```

展示:

单行公式

$$E = mc^2$$

单行带编号

$$a^2 + b^2 = c^2 \tag{1}$$

单行不带编号

$$a^2 + b^2 = c^2$$

由于split和cases需要在数学环境中使用，此处选择将其包含在equation环境中。

多行

$$a = b + c + d + e + f + g + h + i + j + k + l + m + n + o + p + q + r + s + t + u + v + w + x + y + z \tag{2}$$

根据条件分类

$$f(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{if } x = 0 \\ -1, & \text{if } x < 0 \end{cases} \tag{3}$$

2.1.10 如何插入表格?

```

1 \begin{table}

```

```

2 \centering
3 \begin{tabular}{|c|c|c|}
4 \hline
5 Header 1 & Header 2 & Header 3 \\
6 \hline
7 Cell 1 & Cell 2 & Cell 3 \\
8 \hline
9 Cell 4 & Cell 5 & Cell 6 \\
10 \hline
11 \end{tabular}
12 \caption{Sample Table}
13 \label{tab:sample}
14 \end{table}

```

展示:

Header 1	Header 2	Header 3
Cell 1	Cell 2	Cell 3
Cell 4	Cell 5	Cell 6

表 2: Sample Table

2.1.11 如何引用参考文献?

注意, 你通常需要两次编译。LaTeX 中, 交叉引用 (如公式、章节编号)、参考文献的链接信息, 需要在第一次编译时记录位置, 第二次编译时才能正确插入。

简单的用法:

```

1 % references
2 \begin{thebibliography} {}
3 \bibitem[label 1] ...content...
4 \bibitem[label 2] ...content...
5 \end{thebibliography}
6 % how to cite them
7 \cite{label}

```

展示:

```

1 \section{references}
2 \begin{thebibliography}{9}
3     \bibitem[bib:one] ...content...
4 \end{thebibliography}
5 % use \cite{bib:one} to cite it

```


常用用法:

```
1 \cite{label} % to cite
2 \bibliographystyle{unsrt} %in main body
3 \bibliography{bibFileName} % add references
```

展示:

根据爱因斯坦的相对论[1], 时间和空间是相对的。

多项研究表明...[1, 2, 3], 这些工作对科学和出版领域产生了深远影响。

2.1.12 除文献外, 如何引用表格、图片等内容?

```
1 \usepackage{hyperref} % preamble
2 % after setting label
3 \autoref{label}
```

展示:

展示Latex的特殊字符的表格是: **Table 1**

2.1.13 如何插入代码?

注意, 也包括LaTeX代码; 对于参数也可以额外设置;

```
\usepackage{listings} % central package
\usepackage{xcolor} % optional, to set color
\begin{lstlisting}[language=Python]
...
\end{lstlisting}
% follow is style-settings
\lstset{
    language=[LaTeX]TeX, % 指定语言为 LaTeX
    basicstyle=\ttfamily\small, % 基本字体设置
    keywordstyle=\color{blue}, % 关键字颜色
    commentstyle=\color{gray}, % 注释颜色
    stringstyle=\color{red}, % 字符串颜色
    showstringspaces=false, % 不显示字符串中的空格标记
    numbers=left, % 行号位置
    numberstyle=\tiny\color{gray}, % 行号样式
    frame=single, % 边框样式
    breaklines=true, % 自动换行
    escapeinside={\%*}{*}% % 允许在代码中插入 LaTeX 命令
}
```

可以直接导入外部的代码文件:

```
\lstinputlisting[language=,numbers=]{demo.py}
```

展示:

```
1 def MyMain():
2     print("hello,world!")
3
4 MyMain()
```

也可以使用下面这个命令, 来导入代码:

```
1 \begin{verbatim}
2 ...
3 \end{verbatim}
```

2.1.14 如何插入链接?

```
1 \usepackage{hyperref}
2 \href{url}{description}
3 \url{url} % directly displat this url
4 \href{mailto:email_address}{text} % add email address
```

展示,其中邮箱example@outlook.com是任意设置的:

[LaTex官网](#)

<https://www.latex-project.org/>

注意: 部分浏览器无法解析mailto协议, 导致将其视为网址的一部分

[MyEmail](#)

值得一提的是pdfborder可以被打印出来, 可以手动将其取消,示例如下:

```
1 \usepackage[colorlinks=true,pdfborder={0 0 0},
2     linkcolor=red,
3     citecolor=green,
4     urlcolor=blue,
5 ]{hyperref}
```

2.1.15 LaTeX模板和页边距等如何设置?

具体可见本目录下template.tex文件

2.2 Git 练习实例

2.2.1 从历史中删除文件

使用 Git 时的一个常见错误是提交本不应该由 Git 管理的大文件，或是将含有敏感信息的文件提交给 Git。尝试向仓库中添加一个文件并添加提交信息，然后将其从历史中删除

答案解析中提供了一种强大且危险的命令，来删除所有历史中 *my_password* 文件。该操作会改变哈希值，建议仅在本地使用。

```
1 git filter-branch --force --index-filter\  
2 'git rm --cached --ignore-unmatch ./my_password' \  
3 --prune-empty --tag-name-filter cat -- --all
```

不过也有其他方法可用，适用于本地且该危险提交发生不久，优点是命令简单，例如：

```
1 git reset --hard <commit_hash>
```

```
commit 518b8c9bf3fc8bda1d5982b222e0f97002e1d8f4 (HEAD -> main)  
Author: sufwis <sufferingwish@outlook.com>  
Date: Sat Aug 30 17:09:15 2025 +0800  
  
    add my password  
  
commit 8780e73e82749aeff2e0082ecb65b2c67e2de293  
Author: sufwis <sufferingwish@outlook.com>  
Date: Sat Aug 30 17:06:45 2025 +0800  
  
    first_commit
```

图 2: 添加密码文件

```
D:\learn\Practice Cases for Development Tools\git\latex learn\gitlearn>git reset --hard 8780e73e82749aeff2e0082ecb65b2c67e2de293  
HEAD is now at 8780e73 first_commit  
  
D:\learn\Practice Cases for Development Tools\git\latex learn\gitlearn>git log  
commit 8780e73e82749aeff2e0082ecb65b2c67e2de293 (HEAD -> main)  
Author: sufwis <sufferingwish@outlook.com>  
Date: Sat Aug 30 17:06:45 2025 +0800  
  
    first_commit
```

图 3: 执行回滚

回滚后，如图可见log中并无第二次commit

```
D:\learn\Practice Cases for Development Tools\git\latex learn\gitlearn>git reflog  
8780e73 (HEAD -> main) HEAD@{0}: reset: moving to 8780e73e82749aeff2e0082ecb65b2c67e2de293  
518b8c9 HEAD@{1}: reset: moving to 518b8c9bf3fc8bda1d5982b222e0f97002e1d8f4  
518b8c9 HEAD@{2}: commit: add my password  
8780e73 (HEAD -> main) HEAD@{3}: commit (initial): first_commit
```

图 4: 通过git reflog 查看HEAD移动记录

此处，虽然git保留了添加密码的移动记录，但是并不会被push到远端仓库。git reflog能够查看几乎所

有移动过的快照，同时具有本地性和时效性——一段时间后会自动清理。配合回滚操作，能够做到恢复大部分文件的效果，通常应用于误删。

2.2.2 从 GitHub 上克隆某个仓库，修改一些文件。当您使用 `git stash` 会发生什么？当您执行 `git log -all -oneline` 时会显示什么？通过 `git stash pop` 命令来撤销 `git stash` 操作，什么时候会用到这一技巧？

`git stash`: 暂存工作区和暂存区

`git log -all -oneline`: 显示所有本地和远端分支信息，且以oneline即一行的形式，仅保留哈希值、提交信息、引用信息

`git stash pop`: 恢复最近的一次`git stash`保存的修改，并将该 `stash` 从 `stash` 列表中移除。通常用于：

中断当前任务，处理紧急任务；
尝试不同的解决方案；
合并分支前暂存修改；

```
D:\learn\Practice Cases for Development Tools\git&latex learn\gitlearn>git stash
Saved working directory and index state WIP on main: 8780e73 first_commit
```

图 5: 使用`git stash`

```
D:\learn\Practice Cases for Development Tools\git&latex learn\gitlearn>git stash pop
On branch main
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   test.py

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (b75aa85e8b749b15ffb0e31a018725a2d904715b)

D:\learn\Practice Cases for Development Tools\git&latex learn\gitlearn>git add .
D:\learn\Practice Cases for Development Tools\git&latex learn\gitlearn>git commit -m "modify test.py"
[main 9f43d69] modify test.py
 1 file changed, 2 insertions(+)
```

图 6: 使用`git stash pop`

当我修改`test.py`后，并未暂存。而是直接使用`git stash`。之后新建分支，做一次提交后，签出到`main`分支，使用`git stash pop`。此时，`test.py`文件恢复到使用`stash`暂存时的状态。

2.2.3 如何在`git`中创建别名？

1. `git config --global alias.缩写` ;原命令
2. 设置局部别名（仅对当前仓库有效）：去掉 `--global` 参数即可
3. 直接编辑配置文件`.gitconfig`

结果如图

```
D:\learn\Practice Cases for Development Tools\git&latex learn\gitlearn>git config --global alias.graph "log --all --graph --decorate --oneline"
D:\learn\Practice Cases for Development Tools\git&latex learn\gitlearn>git graph
* 9f43d69 (HEAD -> main) modify test.py
| * 9abc509 (test1) add README.md
|/
* 8780e73 first_commit
```

图 7: 使用git config --global alias.缩写原命令

2.2.4 git如何忽略文件?

通常, 建议常见一个.gitignore文件, 在其中写入要删除的文件即可。可以使用**匹配任意目录。

2.2.5 git使用流程?

方法一:

从本地, 简述为: init 初始化 add. commit 提交, remote add origin,push -u origin main 推送

方法二:

从远端, git clone 下分支;

git常用指令可见附件



双击

如果浏览器无法解析,可见./gitlearn/Git.txt

3 实验感悟

从LaTeX的学习中可以了解到, 只有亲自动手练习才是增进熟练度最合适的方式。 LaTeX能够精确控制文档格式, 支持数学符号和公式, 常用于论文设置。尽管仅仅只是简单地学习其语法, 以及基础的使用方法, 也能感受到它的强大。 无疑, LaTeX是值得推荐的。 对于有更高需求的使用者, 深入学习LaTeX也是必要的。

Git不必多言,最常用且最受欢迎的版本控制工具之一。 不管是用于本地管理,还是远端与他人协同工作,git 常常能够提供有力的帮助。如果只是处于使用的考量,学习使用git指令,或者安装对应插件是不错的选择。同时也推荐了解其工作原理,这会对理解git的回滚很有帮助.倘若时间富余,也可深入研究。

4 参考文献示例, 仅用于演示

参考文献

- [1] Albert Einstein. On the electrodynamics of moving bodies. *Annalen der Physik*, 322(10):891–921, 1905.
- [2] Donald E. Knuth. *The TeXbook*. Addison-Wesley, 1984.

[3] Leslie Lamport. Latex: A document preparation system. In *Addison-Wesley Professional*, 1985.

5 个人github账号

[个人github账号](#)

[对应仓库](#)

6 相关资料

[LaTex模板改自此处](#)

[LaTex参考资料](#)

[LaTex官网](#)

[git相关参考资料](#)