

Git 和 LaTeX 课程实例报告

姓名：孔维凯 学号：23020007051

2024 年 9 月 5 日

目录

1 Git 5

1.1 实例 1 5

1.1.1 练习内容 5

1.1.2 结果 5

1.1.3 解题感悟 5

1.2 实例 2 5

1.2.1 练习内容 5

1.2.2 结果 5

1.2.3 解题感悟 5

1.3 实例 3 5

1.3.1 练习内容 5

1.3.2 结果 5

1.3.3 解题感悟 6

1.4 实例 4 6

1.4.1 练习内容 6

1.4.2 结果 6

1.4.3 解题感悟 6

1.5 实例 5 6

1.5.1 练习内容 6

1.5.2 结果 6

1.5.3 解题感悟 6

1.6 实例 6 6

1.6.1 练习内容 6

1.6.2 结果 7

1.6.3 解题感悟 7

1.7 实例 7 7

1.7.1 练习内容 7

1.7.2 结果 7

1.7.3 解题感悟 7

1.8 实例 8 7

1.8.1 练习内容 7

1.8.2 结果 7

1.8.3 解题感悟 7

1.9 实例 9 8

1.9.1	练习内容	8
1.9.2	结果	8
1.9.3	解题感悟	8
1.10	实例 10	8
1.10.1	练习内容	8
1.10.2	结果	8
1.10.3	解题感悟	8
2	LaTeX	8
2.1	实例 1	8
2.1.1	练习内容	8
2.1.2	结果	8
2.1.3	解题感悟	9
2.2	实例 2	9
2.2.1	练习内容	9
2.2.2	结果	9
2.2.3	解题感悟	9
2.3	实例 3	9
2.3.1	练习内容	9
2.3.2	结果	9
2.3.3	解题感悟	9
2.4	实例 4	9
2.4.1	练习内容	9
2.4.2	结果	10
2.4.3	解题感悟	10
2.5	实例 5	10
2.5.1	练习内容	10
2.5.2	结果	10
2.5.3	解题感悟	10
2.6	实例 6	10
2.6.1	练习内容	10
2.6.2	结果	10
2.6.3	解题感悟	10
2.7	实例 7	10
2.7.1	练习内容	10
2.7.2	结果	11

2.7.3 解题感悟 11

2.8 实例 8 11

2.8.1 练习内容 11

2.8.2 结果 11

2.8.3 解题感悟 11

2.9 实例 9 11

2.9.1 练习内容 11

2.9.2 结果 11

2.9.3 解题感悟 11

2.10 实例 10 12

2.10.1 练习内容 12

2.10.2 结果 12

2.10.3 解题感悟 12

1 Git

1.1 实例 1

1.1.1 练习内容

下载并使用他人的代码

1.1.2 结果

首先运用 `git init bare` 命令初始化一个空仓库，仓库名为 `bare`，再输入 `cd+bare` 文件夹地址，然后使用 `git clone+ 代码地址` 将代码文件下载到 `bare` 文件夹里，就可以使用了。

1.1.3 解题感悟

通过这种方式就可以使用他人的代码，省时省力，但是使用前注意要查看协议，不能违反协议使用代码。

1.2 实例 2

1.2.1 练习内容

将 `git` 和 `github` 账户连接到一起

1.2.2 结果

输入 `ssh-keygen -t rsa -b 4096 -C "your email@example.com"` 其中 `your email@example.com` 改为你的电子邮箱地址，然后输入密钥存储地址，设置密码，然后打开密钥文本复制密钥在 `github` 账户上添加 SSH 密钥后就可以了。

1.2.3 解题感悟

将 `git` 和 `github` 账户链接在一起便于使用 `git` 从 `github` 开源网站上下载他人的代码，便于使用。

1.3 实例 3

1.3.1 练习内容

修改下载文件的地址

1.3.2 结果

使用 `cd+ 文件夹地址` 就可以将当前默认地址修改为预期地址

1.3.3 解题感悟

这样做方便使用者快捷查看下载的代码。

1.4 实例 4

1.4.1 练习内容

合并分支、发送合并文件以及删除分支

1.4.2 结果

`git checkout master`

`git merge + 分支名`可以合并分支

`git push+< 远程仓库名 > < 本地分支名 >:< 远程分支名 >` 就可以将合并后的更改到远程仓库

`git branch -d + 分支名`可以删除本地分支

1.4.3 解题感悟

`git` 的指令可以方便用户按照料想快捷完成操作。

1.5 实例 5

1.5.1 练习内容

如何查看基本操作指令和 `git` 版本

1.5.2 结果

输入 `git help` 指令即可查看一些基本操作指令，通过输入 `git version` 指令可以查看当前 `git` 版本

1.5.3 解题感悟

通过 `help` 指令可以快速学习一些基本操作指令，通过 `version` 查看版本情况时刻保持版本更新。

1.6 实例 6

1.6.1 练习内容

设置全局用户名和邮箱

1.6.2 结果

通过 `git config --global user.name+ 名字` 来设置全局用户名

通过 `git config --global user.email+ 电子邮件地址` 来设置全局邮箱

通过 `git config --list` 指令来显示参数

1.6.3 解题感悟

这些指令可以帮助识别代码的作者。

1.7 实例 7

1.7.1 练习内容

分支相关指令

1.7.2 结果

`git checkout+ 分支名` 用于切换分支

`git checkout -b+ 分支名` 用于创建并切换到新分支

`git branch` 用于查看 `git` 仓库的分支情况

1.7.3 解题感悟

分支让开发者可以在不影响主分支的情况下进行试验，有利于保护原有代码。

1.8 实例 8

1.8.1 练习内容

提交代码覆盖了别人的代码怎么挽回

1.8.2 结果

首先使用 `git log` 命令找到覆盖之前的提交，用 `git checkout` 语句检出那个安全的提交，然后创建一个新分支用于合并开发者的更改，再通过 `git push` 语句将开发者的修改发送到远程仓库。

1.8.3 解题感悟

在使用 `git` 的过程中注意不要覆盖别人的代码，要新建分支。

1.9 实例 9

1.9.1 练习内容

使用 git 下载指定版本代码

1.9.2 结果

首先使用 `git clone + 地址`，进入克隆的文件夹使用 `cd + 文件名`，然后查看仓库的历史提交记录，找到想要下载的版本提交哈希值，使用 `git checkout + 该值` 切换到指定版本，再使用 `git log` 指令确认切换成功。

1.9.3 解题感悟

通过这种操作用户可以随心选择以前的版本代码，便于用户选择，满足用户需求。

1.10 实例 10

1.10.1 练习内容

解决 git 冲突

1.10.2 结果

首先使用 `git status` 命令查看冲突的文件，然后取舍保留哪些更改，并删除冲突标记，再使用 `git add + 文件名` 命令以及 `git commit` 指令合并文件。

1.10.3 解题感悟

git 的这种冲突有利于团队合作，防止代码混乱变换。

2 LaTeX

2.1 实例 1

2.1.1 练习内容

将标题、姓名、学号、日期加粗并加下划线

2.1.2 结果

如果直接对 `maketitle` 语句进行操作会使首页变为正文页，而第二页变为原来的首页，不符合预期，经过尝试，发现应该在 `title`、`author`、`date` 语句中进行操作

2.1.3 解题感悟

经过这次练习,我明白了 maketitle 是功能性语句,而如果想要修改标题的内容和格式应该对 title、author、date 这些信息语句进行修改。

2.2 实例 2

2.2.1 练习内容

在代码下面添加图片并规定图片大小为 0.6 倍页面宽度,并使图片居中

2.2.2 结果

首先将图片添加到 main.tex 文件的下面,然后通过 begin figure 环境中的 includegraphics[] 语句中添加图片文件命名,在 [] 中输入 width=0.6textwidth 即可将图片调节为 0.6 倍页面宽度,在 begin figure 后加上 [h],就可以使图片出现在代码下面,centering 语句使图片居中。

2.2.3 解题感悟

LaTeX 中添加图片很方便,同时还可以规范图片格式,数学论文中可以使用,比其他软件优越。

2.3 实例 3

2.3.1 练习内容

并排添加图片

2.3.2 结果

通过在 begin figure 环境中添加 begin minipage 环境在这个环境中就可以并排添加图片了,并且在 minipage 后添加 width=0.4textwidth 就可以将每个图片的页面大小调整到整个页面的 0.4 倍,然后图片的 width=0.6textwidth 并不会超出页面,而是在 0.4 基础上的 0.6 也就是 0.24 倍总页面宽度。

2.3.3 解题感悟

LaTeX 优于 Word 的地方之一就是可以并排添加图片,同时可以规范图片大小,极为方便

2.4 实例 4

2.4.1 练习内容

让页数自动清点总页数并展示出来

2.4.2 结果

通过 `pageref` 语句中输出 `LastPage` 页，但只是这样编译后会出现两个`?`，这时应该加入宏包 `lastpage` 就可以正常显示了。

2.4.3 解题感悟

LaTeX 通过代码来编译为文本使得软件非常智能，无需亲自清点页数，非常省力。

2.5 实例 5

2.5.1 练习内容

人工输入列表修改时特别繁琐，如何智能列表并在列表中添加内容时自动编号

2.5.2 结果

通过在 `begin enumerate` 环境中添加 `item` 语句，然后在 `item` 后面书写内容，要修改时直接在中间填入一行 `item` 语句，系统自动修改编号，无需手动修改。

2.5.3 解题感悟

LaTeX 的列表功能不仅对仗工整，每个序号都上下对齐而且自动编号，省时省力。

2.6 实例 6

2.6.1 练习内容

让标题独占一页

2.6.2 结果

通过在 `maketitle` 语句下方加入 `newpage` 语句就可以使标题独占一页

2.6.3 解题感悟

这个功能使得整个文本的格式清晰合理。

2.7 实例 7

2.7.1 练习内容

自定义页眉

2.7.2 结果

文本中有 6 个页眉区域分别是上左、上中、上右、下左、下中、下右，想要自定义页眉，首先引入 fancyhdr 宏包，然后使用 fancyhf, lhead 表示上左，chead 表示上中、rhead 表示上右、lfoot 表示下左、cfoot 表示下中、rfoot 表示下右，在这 6 个语句中添加自定义的页眉内容即可。

2.7.3 解题感悟

通过这种方式自定义的页眉工整，位置恰当，看起来不突兀，很美观。

2.8 实例 8

2.8.1 练习内容

为文本添加目录

2.8.2 结果

首先注意的一点，系统是以小标题来设计目录的，所以书写标题时要用 section、subsection、subsubsection 语句，然后通过 tableofcontents 语句，系统自动规划好目录并呈现给使用者。

2.8.3 解题感悟

如果人工输入目录，不仅繁琐而且修改复杂，而让系统自动生成就方便许多。

2.9 实例 9

2.9.1 练习内容

设置页面的布局即页面的参数

2.9.2 结果

首先引入 geometry 宏包，然后使用 geometry 语句，在语句中书写比如 left=2cm,right=2cm,top=3.5cm,bottom=3.5cm，参数可以随意更改。

2.9.3 解题感悟

通过设置页面参数可以使整个文本看起来更加美观，赏心悦目。

2.10 实例 10

2.10.1 练习内容

让系统自动输出当前页数及当前日期

2.10.2 结果

文章中常常在下中页眉处表明当前页数，但系统默认输出一个数字，但如果你想输出“第几页”，就需要运用 `cfoot` 语句的同时用到 `thepage` 语句，这个语句可以自动输出当前页数比如“第 `thepage` 页”，要注意的是 `thepage` 和旁边文字间必须有空格，否则会出错导致无法输出当前页数。

如果要输出当前日期，只需要用到 `today` 语句就可以了。

2.10.3 解题感悟

LaTeX 的这些操作减少了使用者的修改工作，大大方便了使用者使用。