思考题部分：

**思考题1.**简单分析 CLI Shell，GUI Shell 在你使用过程中的各自优劣（100字以内）

答：GUI shell优点：图形化界面直观易懂，对于普通用户来说操作和控制方便；缺点：不知道操作或控制的原理和机制。CLI shell的优点：通过命令执行操作和控制，并且能理解其原理；缺点：不直观，操作不方便，普通用户无法理解。

**思考题2**.使用你知道的方法（包括重定向）创建下图内容的文件（文件命名为test），将创建该文件的命令序列保存在command文件中，并将test文件作为批处理文件运行，将运行结果输出至result文件中。给出command文件和result文件的内容，并对最后的结果进行解释说明（可以从test文件的内容入手）

答：创建test:

command文件：

**touch** test

**echo** "echo Shell Start..." **>**test

**echo** "echo set a = 1" **>>** test

**echo** "a=1" **>>** test

**echo** "echo set b = 2" **>>** test

**echo** "b=2" **>>** test

**echo** "echo set c = a+b" **>>** test

**echo** "c=\$[\$a+\$b]" **>>** test

**echo** "echo c = \$c" **>>** test

**echo** "echo save c to ./file1" **>>** test

**echo** "echo \$c>file1" **>>** test

**echo** "echo save b to ./file2" **>>** test

**echo** "echo \$b>file2" **>>** test

**echo** "echo save a to ./file3" **>>** test

**echo** "echo \$a>file3" **>>** test

**echo** "echo save file1 file2 file3 to file4" **>>** test

**echo** "cat file1>file4" **>>** test

**echo** "cat file2>>file4" **>>** test

**echo** "cat file3>>file4" **>>** test

**echo** "echo save file4 to ./result" **>>** test

**echo** "cat file4>>result" **>>** test

Result 文件：

3

2

1

结果分析：首先计算c，然后分别将a,b,c的值保存在3个文件中，再将三个文件的内容依次输出到第四个文件，“>>”这个符号是在文件末尾新增内容。所以结果有三行。

**思考题3**.仔细看看这张图，思考一下箭头中的 add the file 、stage the file 和commit 分别对应的是 Git 里的哪些命令呢？

答：

**git** add

**git** add

**git** commit

**思考题4**.深夜，小明在做操作系统实验。困意一阵阵袭来，小明睡倒在了键盘上。等到小明早上醒来的时候，他惊恐地发现，他把一个重要的代码文件printf.c删除掉了。苦恼的小明向你求助，你该怎样帮他把代码文件恢复呢？

正在小明苦恼的时候，小红主动请缨帮小明解决问题。小红很爽快地在键盘上敲下了git rm printf.c，这下事情更复杂了，现在你又该如何处理才能弥补小红的过错呢？

处理完代码文件，你正打算去找小明说他的文件已经恢复了，但突然发现小明的仓库里有一个叫Tucao.txt，你好奇地打开一看，发现是吐槽操作系统实验的，且该文件已经被添加到暂存区了，面对这样的情况，你该如何设置才能使Tucao.txt在不从工作区删除的情况下不会被git commit指令提交到版本库？

答：（1）**git** checkout **--** printf.c从工作区中删除文件，可以从缓存区、版本库中找回来。

（2）**git** checkout **--** printf.c从缓存区删除文件依然可以从版本库中找回来。

（3）**git** rm **--**cached Tucao.txt删除缓存区的文件。

**思考题5**.思考下面四个描述，你觉得哪些正确，哪些错误，请给出你参考的资料或实验证据。

1. 克隆时所有分支均被克隆，但只有HEAD指向的分支被检出。
2. 克隆出的工作区中执行 git log、git status、git checkout、git commit等操作不会去访问远程版本库。
3. 克隆时只有远程版本库HEAD指向的分支被克隆。
4. 克隆后工作区的默认分支处于master分支。

答：（1）克隆时分支并没有被检出，而需要使用git checkout <file>才能将分支检出，克隆时只克隆远程库的master分支；

（2）正确，只有通过git push之后才会将本地版本库的文件与远程版本库更新；

1. 正确；
2. 正确。

实验难点图示：

1. 第一个难点是Makefile文件的编写。

程序编写Fibo.c

编写Mkefile

执行使得fibo可执行文件生成

生成fibo

1. 第二个难点是shell脚本文件的编写。

hello\_os.sh脚本编写

利用bash命令执行

生成hello\_os.c文件（包含file文件特定行内容）

LOREM IPSUM

体会与感想：

本次lab0作业主要是熟悉Linux命令和git的操作，刚开始学不是很有头绪，查阅了大量的资料，渐渐地，掌握了基本的Linux命令，shell脚本的简单编写，也试着更改vim编辑器的操作界面（但这么课程不允许使用）。

下次作业不知道是什么难度了，只是目前操作系统课程才刚开始，对进程管理还没有思路，所以还正在学习中。