task1

通过 cd src 命令进入~/学号/src 目录（可以通过 cd ../ 命令返回父目录）

然后使用 vim palindrome.c 命令打开 palindrome.c 文件，按下 insert 键，看到左下角出插入 

的字样就可以进行编辑，方向键上下左右可以移动光标位置。

（然后就是正常写代码~）

写完后猛按 esc 然后输入:wq (记得有冒号)回车退出 vim 并保存修改

使用 cd ../ 命令返回~/学号/目录

输入 git add . （注意这里有个点！）

然后正常 git commit ，git push 即可进行检查（但是在完成 task4 之前评测机并看不懂宁的代码）

至此 task1 完成

task2

仿照 task1 中的步骤进入 src 目录下

使用 vim Makefile 打开 Makefile 文件

在里面输入

all : palindrome.c （所有涉及编译等操作的文件，后面可以继续列）

gcc -o palindrome palindrome.c （含义：gcc ：调用 gcc 编译器 -o 生成 object 文件（先不用管） palindrome ：生成的文件名 palindrome.c ： 使用到的文件）

（此处本来有一张图片）

（写完后大概长这样,注意一定要打 tab，否则会不解析）

写完后猛按 esc 然后输入:wq (记得有冒号)回车退出 vim 并保存修改

至此 task2 任务全部完成，可以 commit + push 了，记得退回根目录 git add . 防止漏文件

task3

本任务设计编写 shell 脚本

仿照 task1 的命令，进入 src/sh_test 目录下 ,打开 hello_os.sh 文件

这里我们要用到 sed 命令来提取目标文件对应行

语法为：sed -n ‘8p’ $1 > $2

和 sed -m ‘8p’ $1 > $2

含义为：-n ：结果不向命令行输出

‘8p’ 找到第八行，注意一定有 p

$1 :这个位置后放置查找源文件

> 代表将前面指令的输出重定向到 $2 为文件名的文件，并覆盖

>> 代表将前面指令的输出重定向并追加到原来内容后面，不覆盖

同时我们这里用到了 shell 脚本的函数功能，语法如下：

```
fn() {
//函数体
}
fn $1 $2 //函数调用
```

$1 、$2 分别代表调用脚本的时候，输入的第一个和第二个参数，以此类推

总体写出来大概是这个样子：

（此处本来有一张图片）

（千万不要忘了前面的 #!/bin/bash ,相当于头文件）

task4

这里就是复制文件的练习 使用 cp 指令 语法如下：

cp 目标文件 目标路径

值得一提的是，这里将路径调整到 src 下 直接写下 cp -r ./ ../dst 可以完成一键复制

task5

task5 与 task3 内容类似，只不过新加了循环内容（官方已经给咱写好了），然后循环里用 rm -r 命令移除目录，mv 命令给目录改名 ，具体写法如下：

（此处本来有一张图片）

需要注意的是 linux 字符串拼接变量的方法：

“字符串内容${变量}”

task6

task6 也与 task3 类似，只不过增加了 grep 命令和 awk 命令，具体写法如下：

（此处本来有一张图片）

grep 语法解析！：

-n 查找行号 -E 代表后面是正则 $2 一个捕获的字符串 $1 grep 操作的文件名

这里重点关注下管道符号 |

这里管道符号的意思是：将前一个命令的输出，作为后一个命令的输入

awk 语法解析！：

-F: 注意这里冒号是分隔符号 {print $1} 打印出由冒号分隔的第一组

至此 task6 完毕

task7

task7 也只是尝试一下 sed 命令的新功能

Sed -i “s/$2/$3/g” $1

这个函数意为，在头到尾之间将$2 替换为$3 的内容,相信聪明的你现在一定能看懂了吧~!

Task8

Task8 涉及到较为复杂的跨目录 Makefile 操作，具体写法如下，且等细细道来

（这里曾经有一张图片）

首先：变量的建立

这里设计到了 wildcard 函数的应用，这个函数可以在我们指定的

目录下，如(code/*) 下，查找所有的符合我们规定格式的文件名(比如.c 后缀文件),并返回一个字符串存储到对应变量里。

在 all 的时候，我们可以通过这种方法快速获得跨目录条件下的所有依赖项

Gcc -c 生成.o 文件

-I include 头文件！重要！

-o 输出到哪个目录下的哪个文件（没有会新创建）

直接 gcc (多个.c) -I 生成可执行文件

至此 lab0 终于结束了！