Lab03 - Assignment

1. 基础

1.1 权限

对一个文本文件 `file.txt` 执行命令： `sudo chmod 777 file.txt` 。请解释该命令的含义并写出执行该命令后该文件的权限信息。

> 赋予权限 777 给文件 file.txt
>
> 777：rwxrwxrwx（对于所有用户，runnable、writable、executable）

1.2 管道、重定向

现有文件 `a.txt` ，文件内容为

```
Welcome to linux!
```

和文件 `f1` ，文件内容为

```
Hello World!
```

解释以下命令 `cat a.txt | cat >>f1 2>&1` 的原理,同时给出执行完命令后f1文件中的内容

> 原理：
>
> - cat a.txt 打印文件内容到管道中
> - 管道作为右侧命令的输入，再次打印，并将标准输出追加重定向至 f1；
>
> 文件内容：
>
> ```
> Hello World!Welcome to linux!
> ```

1.3 环境变量

自己添加一个环境变量，名称是 `STUDENT_ID` ，值为你的学号，并编写一个C程序来获取该环境变量，并打印出来。**请详细叙述你的操作过程以及操作过程的截图，并给出C程序的代码。**

> ```
> longin@LAPTOP-U67RF8M6:/mnt/d/Work/2022/Course/H2/系统编程/test$ export STUDENT_ID
> longin@LAPTOP-U67RF8M6:/mnt/d/Work/2022/Course/H2/系统编程/test$ ./a.out
> 19377167
> ```

```c
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
#include<sys/types.h>
#include<sys/stat.h>
#include<fcntl.h>

int main() {
system("echo $STUDENT_ID");
}
```

2. Shell 编程

2.1 脚本解释器

假如在脚本的第一行放入 `#!/bin/rm` 或者在普通文本文件中第一行放置 `#!/bin/more`，然后将文件设为可执行权限执行，看看会发生什么，并解释为什么。

> 以 #! 开头的第一行是该文件的默认脚本解释器，即如果改文件是 executable 的，并且直接执行（没有指定解释器）时，会用该解释器执行。
>
> 所以 rm 直接删除该文件，more 展示该文件。

2.2 任何位置都可以运行的 Bash

编写一个 bash 脚本，执行该脚本文件将得到两行输出，第一行是你的学号，第二行是当前的日期（考虑使用 `date` 命令）。

要求：

- 用户可以**在任意位置只需要输入文件名**就可以执行该脚本文件
- **不破坏除用户家目录之外的任何目录结构**，即不要在家目录之外的任何地方增删改任何文件

你的 bash 脚本文件内容：

```bash
echo 19377167
echo `date`
```

除编写该文件之外你进行了哪些操作？

> 额外操作：
>
> ```bash
> PATH="${PATH}:`pwd`"
> ```

2.3 Bash 实战 (1)

完成 LeetCode: 193. Valid Phone Numbers，并提供你的 AC 代码与截图。

提示：使用 `grep -e` 及正则表达式。

AC 代码：

```
grep -e "^[0-9]\{3\}\-[0-9]\{3\}\-[0-9]\{4\}$" -e "^([0-9]\{3\}) [0-9]\{3\}\-[0-9]\
{4\}$" file.txt
```

2.4 Bash 实战 (2)

完成 LeetCode: 195. Tenth Line，并提供你的 AC 代码与截图。

AC 代码：

```
tail -n+10 file.txt | head -n1
```

2.5 Bash 实战 (3)

编写一个文件 `manage.sh`，其用法如下：

```
manage.sh OPERATION
```

其中：

- 参数 OPERATION 有以下选项：
  - `mine`：输出当前路径下（不包括子目录）所有者为当前用户的所有文件名
  - `largest`：输出当前路径下（不包括子目录）最大文件的文件名
  - `expand` **（选做）**：递归地将当前目录中所有文件（无论在哪个子文件夹中）全部移动至当前目录下，并删除所有子文件夹。

对于 `expand` 操作，举个例子：

假设 `/root/lab03` 结构如下：

```
/root/lab03/
    |------ a.txt
    |------ b.txt
    |------ c/
            |------ d.txt
            |------ e.txt
    |------ f/
            |------ g/
                    |------ h.txt
```

运行 `./manage.sh expand` 后，其结构将变为：

```
/root/lab03/
    |------ a.txt
    |------ b.txt
    |------ d.txt
    |------ e.txt
    |------ h.txt
```

你的代码：

```
#!/bin/bash

case $1 in
mine)
    i_am=`who | head -n1 | awk '{print $1}'`;
    for file_name in `ls`; do
        file_owner=`ls -l "$1/$file_name" | awk '{print $3}'`;
```

```
        [ "$file_owner" = "$i_am" ] && echo "$file_name";
    done
    ;;
largest)
    largest_size=0
    largest_file=""
    for file_name in `ls`; do
        file_size=`ls -l "$file_name" | awk '{print $5}'`;
        if [ $file_size -gt $largest_size ]; then
            largest_size=$file_size;
            largest_file=$file_name;
        fi;
    done
    echo $largest_file;
    ;;
esac
```