

Linux 第二次实验实验报告

1,是用调用系统的时间判断现在处于早上中午晚上三个时间段中的一个，打出问候语的文本。

文本代码部分：

```
#!/bin/bash
hour = $(date +%H)
case $hour in
0[1-9] | 1[01] )
echo "Good morining !!"
;;
1[234567] )
echo "Good afternoon !!"
;;
* )
echo "Good evening !! "
;;
esac
```

运行结果如下：



```
pi@pi-VMware-Virtual-Platform: ~/B22040612 YZ Linux/实验二
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ nano check_time.sh
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ chmod 777 check_time.sh
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ ./check_time.sh
./check_time.sh: 行 2: hour: 未找到命令
Good evening !!
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$
```

2,是按照提示输入两个整数，脚本将输出哪个数更大、更小或相等的一个判段文本。

文本代码部分：

```
#!/bin/sh
echo "Enter the first integer:"
read first
echo "Enter the second integer:"
read second
if [ "$first" -gt "$second" ]; then
    echo "$first is greater than $second"
elif [ "$first" -lt "$second" ]; then
    echo "$first is less than $second"
else
    echo "$first is equal to $second"
fi
```

运行结果如下：

```
pi@pi-VMware-Virtual-Platform: ~/B22040612 YZ Linux/实验二
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ nano compare_numbers.sh
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ chmod 777 compare_numbers.sh
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ ./compare_numbers.sh
Enter the first integer:
1
Enter the second integer:
2
1 is less than 2
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$
```

3,是当给定列表时,判断并筛选出列表中的最小值,让后打印出来展示在屏幕上面的一个寻找最小值的文本代码。

文本代码部分:

```
#!/bin/bash
# 初始化一个很大的数作为最小值
smallest=10000
# 循环遍历给定的数字列表
for i in 8 2 18 0 -3 87; do
    # 如果当前数字小于 smallest, 则更新 smallest
    if [ $i -lt $smallest ]; then
        smallest=$i
    fi
done
# 输出最小值
echo $smallest
```

运行结果如下:

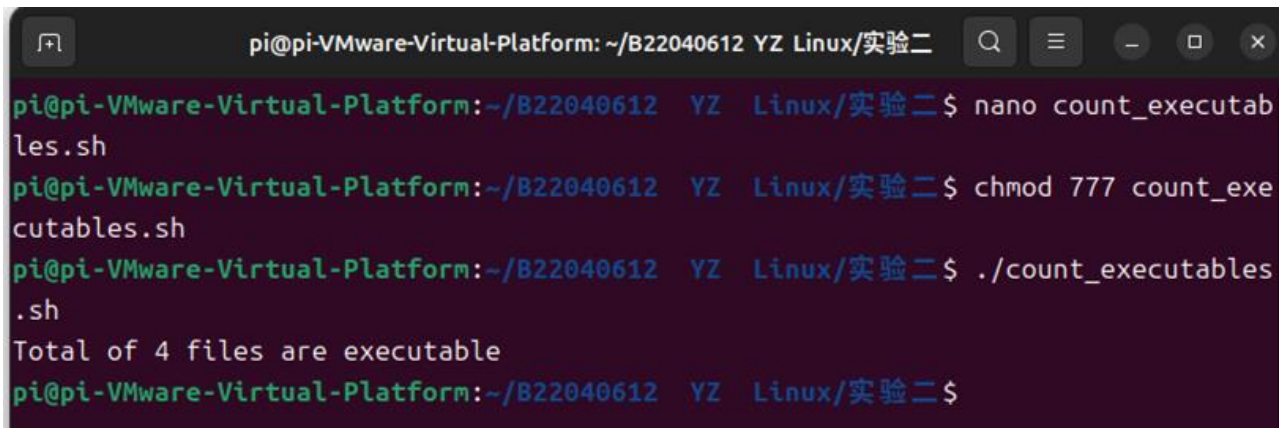
```
pi@pi-VMware-Virtual-Platform: ~/B22040612 YZ Linux/实验二
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ nano find_min.sh
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ chmod find_min.sh
chmod: "find_min.sh" 后缺少操作对象
请尝试执行 "chmod --help" 来获取更多信息。
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ chmod 777 find_min.sh
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ ./find_min.sh
-3
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$
```

4,脚本将计算并输出当前目录中的可执行文件数量,实现此功能的脚本代码。

文本代码如下:

```
#!/bin/bash
count=0
for i in *; do
    if [ -x "$i" ]; then
        count=$((count + 1))
    fi
done
echo "Total of $count files are executable"
```

运行结果如下：



```
pi@pi-VMware-Virtual-Platform: ~/B22040612 YZ Linux/实验二
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ nano count_executables.sh
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ chmod 777 count_executables.sh
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ ./count_executables.sh
Total of 4 files are executable
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$
```

5,定义了一个名为 **prime** 的函数来检查给定的数字是否为质数，并尝试调用该函数，并给出是否为质数的答案并且显示在屏幕上。

文本代码如下：

```
#!/bin/bash
# 定义检查质数的函数
prime() {
    local number=$1 # 获取传入的参数
    local flag=1    # 初始化标志为1，假设是质数
    local j=2       # 从2开始检查
    # 使用 while 循环检查到 number/2
    while [ $j -le $(( $number / 2 )) ]; do
        if [ $(( $number % $j )) -eq 0 ]; then # 如果能被 j 整除
            flag=0 # 不是质数
            break # 跳出循环
        fi
        j=$((j + 1)) # j 递增
    done

    # 根据 flag 的值返回结果
    if [ $flag -eq 1 ]; then
        return 1 # 是质数
    else
        return 0 # 不是质数
    fi
}
```

```
fi
}

# 从命令行参数获取数字
if [ -z "$1" ]; then
    echo "Usage: $0 <number>"
    exit 1
fi
# 调用 prime 函数并检查返回值
prime $1
if [ $? -eq 1 ]; then
    echo "$1 is a prime!"
else
    echo "$1 is not a prime!"
fi
```

运行结果如下：

A terminal window titled 'pi@pi-VMware-Virtual-Platform: ~/B22040612 YZ Linux/实验二' with search, menu, and window control icons. The terminal shows the following commands and output:

```
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ nano check_prime.sh
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ chmod 777 check_prime.sh
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ ./check_prime 15
bash: ./check_prime: 没有那个文件或目录
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$ ./check_prime.sh 17
17 is a prime!
pi@pi-VMware-Virtual-Platform:~/B22040612 YZ Linux/实验二$
```