

南京邮电大学

实验报告

(2025 / 2026 学年 第一学期)

课程名称	Linux 编程				
实验名称	Shell 脚本编程实验				
实验时间	2025	年	10	月	24
指导单位	计算机学院、软件学院、网络空间安全学院				
指导教师	王磊				

学生姓名	柳怡晨	班级学号	B23041307
学院(系)	计算机学院	专业	信息安全

实验报告

实验名称	Shell 脚本编程实验			指导教师	王磊
实验类型	上机	实验学时	4	实验时间	20

一、 实验目的和要求

掌握 Shell 脚本的基本语法和结构

熟悉 Linux 环境下 Shell 脚本的编写、执行和调试方法

掌握 Shell 脚本中变量、条件判断、循环结构、函数等的使用

能够使用 Shell 脚本解决实际的文件操作和系统管理问题

二、 实验环境(实验设备)

Ubuntu 环境、计算机

三、 实验原理及内容

任务 1：根据系统时间输出问候语

1.1 实验步骤：

创建脚本文件： nano task1.sh

输入代码：

```
#!/bin/bash
hour=$(date +\%H)
case $hour in
    0[0-9] | 1[01])
        echo "Good morning !"
        ;;
    1[2-7])
        echo "Good afternoon !"
        ;;
    *)
        echo "Good evening !"
        ;;
esac
```

保存退出: Ctrl+X → Y → Enter

添加执行权限: chmod +x task1.sh

运行脚本: ./task1.sh

1.2 实验结果:

```
.. ..
sshuser@y-virtual-machine:~/experiment$ nano task1.sh
sshuser@y-virtual-machine:~/experiment$ echo B20341307
B20341307
sshuser@y-virtual-machine:~/experiment$ echo B23041307
B23041307
sshuser@y-virtual-machine:~/experiment$ ./task1
bash: ./task1: 没有那个文件或目录
sshuser@y-virtual-machine:~/experiment$ chmod +x task1.sh
sshuser@y-virtual-machine:~/experiment$ ./task1
bash: ./task1: 没有那个文件或目录
sshuser@y-virtual-machine:~/experiment$ ./task1.sh
Good night!
```

任务 2: 比较两个整数大小

2.1 实验步骤:

创建脚本文档: nano task2.sh

保存退出

添加执行权限: chmod +x task2.sh

运行测试:

```
#!/bin/bash
```

任务 2: 比较两个整数

```
echo "==== 整数比较程序 ===="
```

```
# 获取第一个数字
```

```
echo -n "请输入第一个整数: "
```

```
read num1
```

```
# 获取第二个数字
```

```
echo -n "请输入第二个整数: "
```

```
read num2
```

```
# 比较小大
```

```
echo ""
```

```
echo "比较结果:"
```

```
if [ $num1 -gt $num2 ]; then
```

```
echo "$num1 大于 $num2"
elif [ $num1 -lt $num2 ]; then
    echo "$num1 小于 $num2"
else
    echo "$num1 等于 $num2"
fi
$ ./task2.sh
```

2 测试结果：

```
sshuser@y-virtual-machine:~/experiment$ nano task2
sshuser@y-virtual-machine:~/experiment$ nano task2.sh
sshuser@y-virtual-machine:~/experiment$ echo B23041307
B23041307
sshuser@y-virtual-machine:~/experiment$ chmod +x task2.sh
sshuser@y-virtual-machine:~/experiment$ ./task2.sh
--- 整数比较程序 ---
请输入第一个整数: 1
请输入第二个整数: 5

比较结果:
1 小于 5
```

任务 3：查找列表中的最小值

3.1 实验步骤：

创建脚本文件：

```
#!/bin/bash
# 任务 3：查找最小值
# 定义数字列表
numbers=(8 2 18 0 -3 87)
# 初始化最小值
min=${numbers[0]}
# 遍历列表
echo "数字列表: ${numbers[@]}"
echo ""
for num in "${numbers[@]}"; do
    echo "检查数字: $num"
    # 比较并更新最小值
    if [ $num -lt $min ]; then
        echo "发现新的最小值: $num"
        min=$num
    fi
done
echo ""
```

```
echo "最小值是: $min"
保存退出
添加执行权限: chmod +x task3.sh
运行脚本: ./task3.sh
3.2 实验结果:
输入列表: 8, 2, 18, 0, -3, 87
算法验证: 手动验证列表中的最小值确实是-3
```

```
sshuser@y-virtual-machine:~/experiment$ nano task3.sh
sshuser@y-virtual-machine:~/experiment$ chmod +x task3.sh
sshuser@y-virtual-machine:~/experiment$ echo B23041307
B23041307
sshuser@y-virtual-machine:~/experiment$ ./task3.sh
数字列表: 8 2 18 0 -3 87

检查数字: 8
检查数字: 2
    发现新的最小值: 2
检查数字: 18
检查数字: 0
    发现新的最小值: 0
检查数字: -3
    发现新的最小值: -3
检查数字: 87

最小值是: -3
```

任务 4: 统计当前目录可执行文件数量

4.1 实验步骤:

创建脚本文档: nano task4.sh
添加执行权限: chmod +x task4.sh
运行脚本: ./task4.sh

```
#!/bin/bash
# 任务 4: 统计可执行文件
```

```
count=0
echo "当前目录: $(pwd)"
echo "开始扫描可执行文件..."
echo ""
# 遍历当前目录所有文件
for file in *; do
    if [ -f "$file" ] && [ -x "$file" ]; then
        echo "✓ $file"
        count=$((count + 1))
    fi
done
echo ""
echo "可执行文件总数: $count"
```

```
最小值是: -3
sshuser@y-virtual-machine:~/experiment$      nano task4.sh
sshuser@y-virtual-machine:~/experiment$ echo B23041307
B23041307
sshuser@y-virtual-machine:~/experiment$ chmod +x task4.sh
sshuser@y-virtual-machine:~/experiment$ ./task4.sh
当前目录: /home/sshuser/experiment
开始扫描可执行文件...

✓ task1.sh
✓ task2.sh
✓ task3.sh
✓ task4.sh
```

任务 5：判断素数

5.1 实验步骤：

创建脚本文件： nano task5.sh

添加执行权限： chmod +x task5.sh

```
# 任务5. 削弱杀奴

# 素数判断函数
is_prime() {
    local num=$1

    # 处理特殊情况
    if [ $num -le 1 ]; then
        return 1 # 非素数
    fi

    # 处理特殊情况
    if [ $num -eq 2 ]; then
        return 0 # 是素数
    fi

    # 检查偶数
    if [ $((num % 2)) -eq 0 ]; then
        return 1 # 非素数
    fi
}
```

```
可执行文件总数: 4
sshuser@y-virtual-machine:~/experiment$ nano task5.sh
sshuser@y-virtual-machine:~/experiment$ chmod +x task5.sh
sshuser@y-virtual-machine:~/experiment$ ./task5.sh 1
==== 素数判断程序 ====
1 不是素数
sshuser@y-virtual-machine:~/experiment$ ./task5.sh 97
==== 素数判断程序 ====
97 是素数
sshuser@y-virtual-machine:~/experiment$
```



实验报告

四、实验小结（包括总结上机调试过程中所遇到的问题和解决方法、感想与建议等）

在本次 Shell 脚本编程实验中，我通过系统地完成五个由浅入深的编程任务，全面掌握了在 Linux 环境下进行 Shell 脚本编写与调试的实践技能。从最初简单的问候脚本到实现素数判断的函数封装，我逐步理解了 Shell 编程的核心结构，包括变量的灵活使用、多种条件判断语句的应用、循环控制的实现方法以及函数的定义与调用机制。

实验过程中，我遇到了诸如脚本执行权限不足、时间格式获取方式不当、条件判断中空格处理错误、函数返回值逻辑混淆等实际问题。通过查阅手册、反复调试和主动思考，我不仅解决了这些问题，还加深了对 Shell 脚本严谨性和灵活性的认识。尤其是在任务五中编写素数判断函数时，我不仅优化了算法效率，还学会了处理用户输入和边界情况，这使我对 Shell 脚本的健壮性和实用性有了更深的体会。此次实验让我切身体会到 Shell 脚本在系统管理、文件操作和自动化任务中不可替代的作用，同时也锻炼了我分析问题、调试代码和编写文档的综合能力，为今后从事 Linux 系统运维和信息安全相关的工作打下了坚实的实践基础。

五、指导教师评语

成 绩		批阅人		日 期	
-----	--	-----	--	-----	--