**中国矿业大学计算机学院**

**2020 级本科生课程报告**

课程名称 操作系统课程设计

报告时间 2022年5月14日

学生姓名 杨再润

学 号 06203203

专 业 数据科学与大数据

任课教师 鲍 宇

目录

[实验三 Shell程序设计 3](#_Toc104728708)

[【实验目的】 3](#_Toc104728709)

[【实验内容】 3](#_Toc104728710)

[【实验步骤】 3](#_Toc104728711)

[【运行结果】 4](#_Toc104728712)

[【思考练习】 8](#_Toc104728713)

[【实验感受】 10](#_Toc104728714)

# 实验三 Shell程序设计

## 【实验目的】

* 了解Shell在Linux中的重要作用。
* 学会编写简单的Shell脚本程序。

## 【实验内容】

1. **准备实验**

使用浏览器平台领取本次实验对应的任务，从而在平台上创建个人项目（Linux 0.11 内核项目），然后使用VSCode将个人项目克隆到本地磁盘中并打开。

1. **Shell程序中的特殊字符**

了解7种在shell语句中起特殊作用的字符，包括转义字符等。

1. **回显**

学习回显语句的使用，回显语句是Shell脚本中使用最多的语句，用来向标准输出打印变量值或字符串。

1. **变量**

变量用于保存值或字符串，变量可以被创建、赋值和删除。通常所有变量都被看做字符串并以字符串来存储，即使它们被赋值为整数时也是如此。Shell和软件工具会在需要时把数值型字符串转换为对应的数值以对它们进行操作。注意，变量名称只能包含大、小写字母、数字（0~9）和下划线，且只能以字母或下划线开头。

1. **算数运算**

let 语句后可跟算术表达式执行整数算数运算，因而Shell含有各种算数运算符。

1. **函数**

Shell脚本程序中的函数与一般语言中函数的用法类似，必须先定义后使用，并且可以传递参数，并返回值。

1. **控制语句**

命令在脚本中的执行顺序称为脚本流。当编写一个较复杂的脚本时，常常需要根据不同的条件执行不同的脚本流，这就要求Shell提供各种流程控制语句。

1. **提交作业**

## 【实验步骤】

1. **准备实验**
2. 构建项目
3. Bochs 运行（不调试）
4. 使用vi编辑器编写Shell脚本程序
5. **Shell程序中的特殊字符**

**识记7中特殊字符的作用**

1. **回显**
2. 使用vi编辑器新建Shell脚本文件shell1.sh。
3. 编辑shell1.sh的内容如下（以‘#’开始的行为注释）：

#print hello world

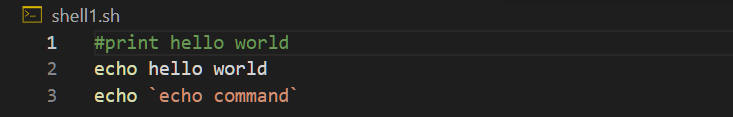
echo hello world

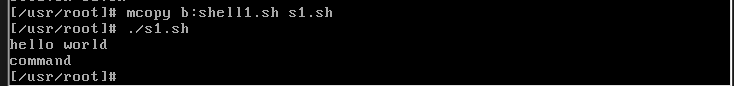
echo `echo command`

1. 退出vi编辑器并保存文件。
2. 在Shell中输入./shell1.sh后回车，观察运行结果。
3. 使用sync命令将新建的Shell脚本文件保存到硬盘。
4. **变量**
5. 编写shell2.sh，学习用户自定义变量
6. 编写shell3.sh，学习位置变量
7. 编写shell4.sh，学习Shell变量
8. 观察各个程序的执行结果。
9. **算数运算**
10. 编写如下Shell脚本文件shell5.sh，然后在命令行中输入“./shell5.sh 2 3”，并观察运行结果
11. **函数**
12. 编写如下Shell脚本文件shell6.sh，然后在命令行中输入“./shell6.sh 2 3”，并观察运行结果
13. **控制语句**
14. 学习测试语句的条件表达式
15. 学习条件语句、开关语句、循环语句、读入语句、等待语句、退出语句的写法
16. 为上述语句分别编写shell7-shell12.sh文件，观察运行结果
17. **提交作业**
18. 查看文件变更详情，并确保从shell1.sh到shell12.sh这十二个脚本文件在项目文件夹的根目录中然后再将本地项目提交到平台的个人项目中。

## 【运行结果】

1. **回显**





**图 shell1.sh内容与运行结果**

运行结果共两行，分别打印出“hello world”和“command”。

1. **变量**



图 shell2.sh内容

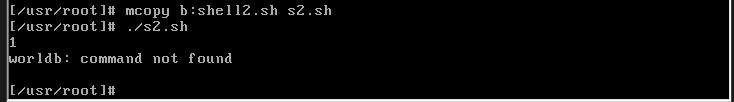


图 shell2.sh运行结果

结果仍为两行，第一行显示变量a的值，第二行预期为变量b的值，但命令解析出现了问题。

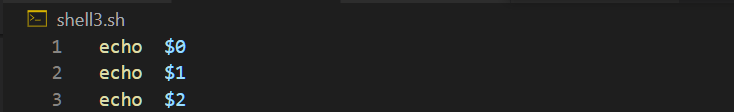


图 shell3.sh内容

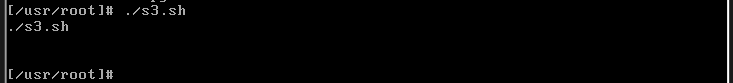


图 shell3.sh运行结果

运行结果为3行，第一行对应的$0为特殊变量，他表示当前文件名，后两行输出为空。

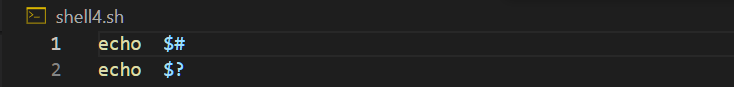


图 shell4.sh内容

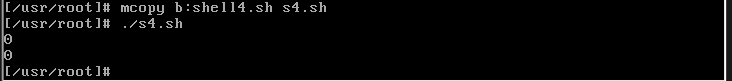


图 shell4.sh运行结果

Shell变量的名字和含义是固定的。$?用于得到上一条命令的十进制返回码，$$用于得到当前Shell进程的进程号，$!用于得到上个后台进程的进程号，$#用于得到传递给Shell程序的参数个数（不限于9个，但统计数量时不包含作为第一个参数的Shell文件名），$-用于得到当前Shell（用set）设置的执行标识名组成的字符串，$\*用于得到命令行中实际给出的实参字符串。Shell变量是由Shell程序本身设置的特殊变量，根据实际情况设置，不允许用户重新设置。

1. **算数运算**

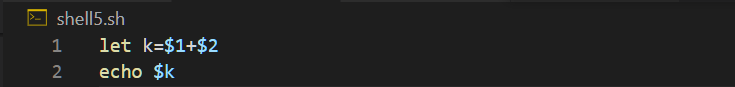


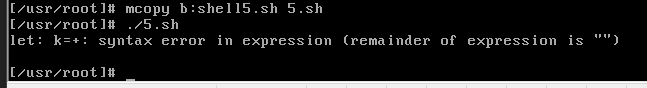
图 shell5.sh内容 

图 shell5.sh运行结果

let 语句后可跟算术表达式执行整数算数运算，因而Shell含有各种算数运算符。

1. **函数**

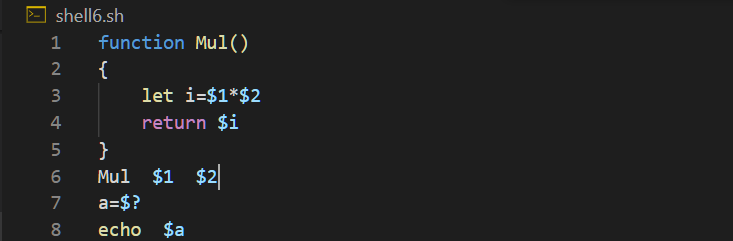


图 shell6.sh内容

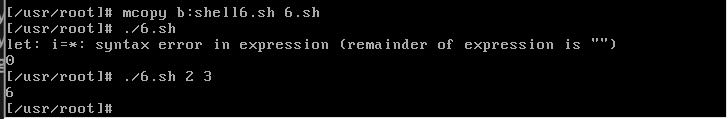


图 shell6.sh运行结果

在Shell脚本中调用函数与执行命令的格式一样：函数名 参数列表；

在第一次运行时，为指定参数，因而出错。指定参数后，参数表按顺序排列传入。

1. **控制语句**

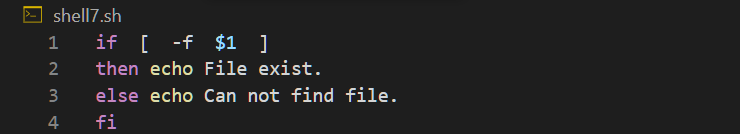


图 shell7.sh内容



图 shell7.sh运行结果

该命令用于判断某个文件是否存在，是个判断语句。

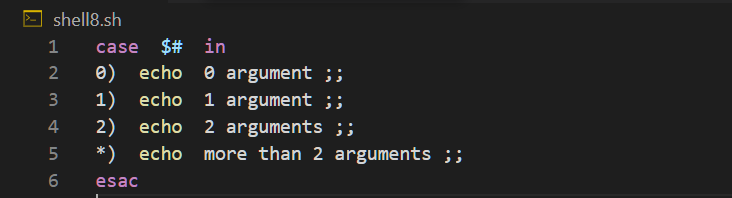


图 shell8.sh内容



图 shell8.sh 1 2运行结果

shell8.sh，用于判断传递给Shell脚本的参数个数

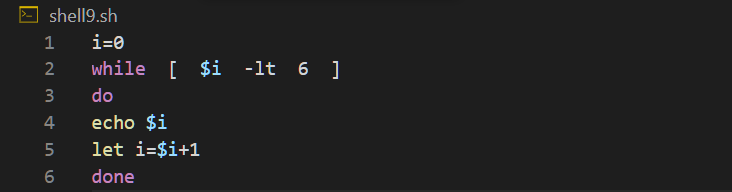


图 shell9.sh内容



图 shell9.sh运行结果

若测试语句返回真，则进入循环体执行命令表，然后再执行测试语句，直至其返回假为止。该循环体一共执行了6次。

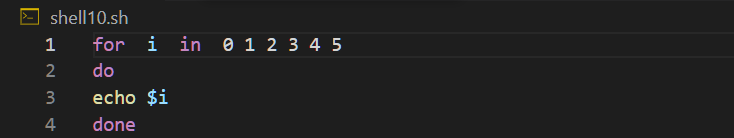


图 shell10.sh内容

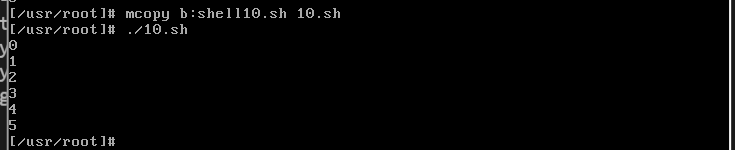


图 shell10.sh运行结果

变量依次取值表中的各个值，然后进入循环体并执行命令表中的命令，变量变为空时结束for循环。该循环一共执行了6次。

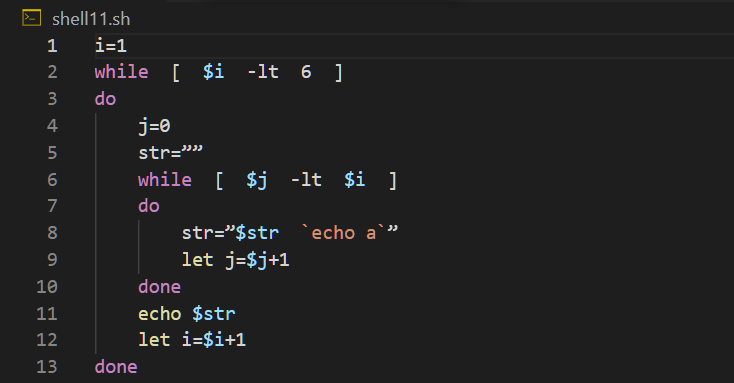


图 shell11.sh内容

由于打印字符串的问题，命令无法正常解析。

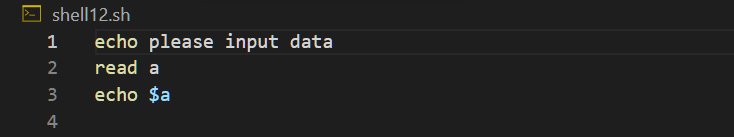


图 shell12.sh内容

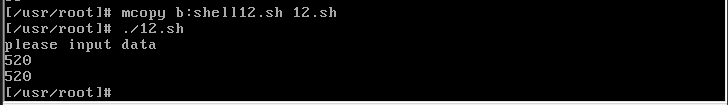


图 shell12.sh运行结果

从键盘读取数据并赋给变量a，并打印在屏幕上。

## 【思考练习】

1. 编写一个Shell脚本文件calculator.sh，要求从命令行输入简单的算术表达式（加减乘除），根据不同的运算符调用不同的函数计算并返回结果，最后打印结果。例如输入命令：

./calculator.sh 2 + 4

将调用计算加法的函数后返回结果6，并打印“2+4=6”。

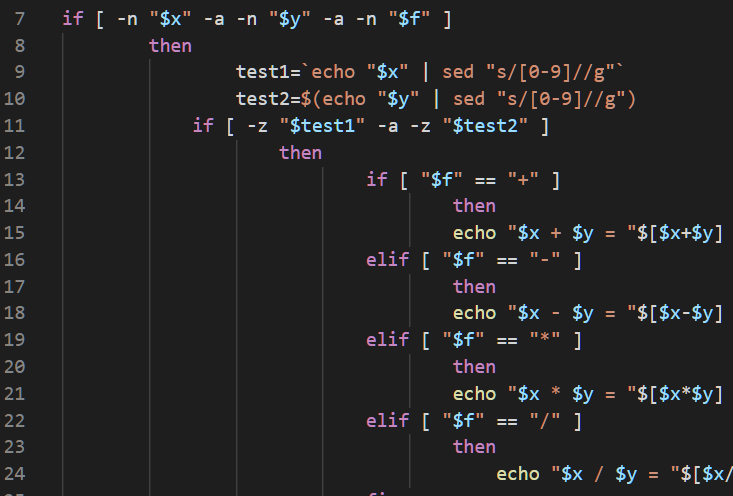


图 部分代码

1. 请分别使用for语句和until语句改写shell11.sh文件中的双重循环，打印用字母a填充的三角形。

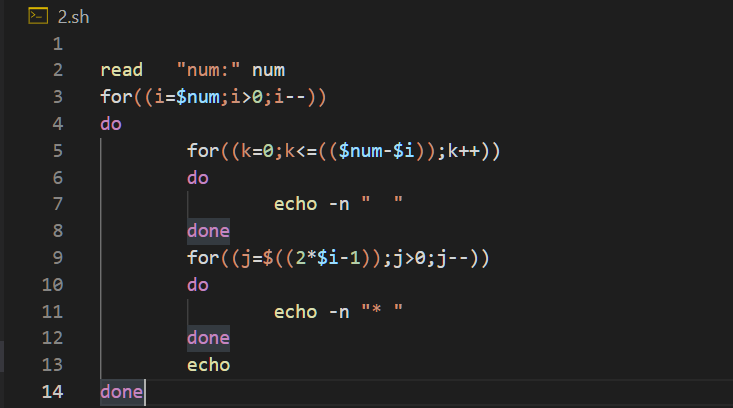


图 用for改写的完整代码

1. 请选择一种循环语句编写Shell脚本文件，实现打印九九乘法表功能。

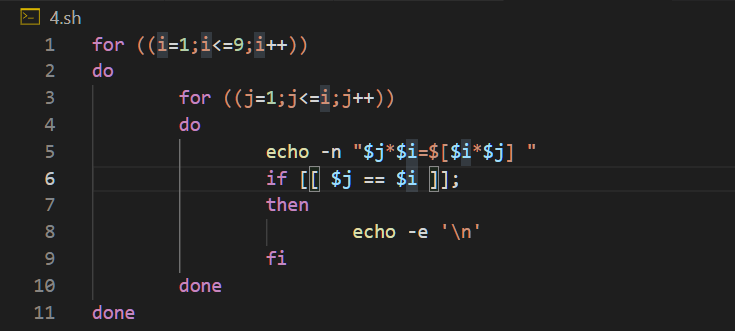


图 打印乘法表的完整代码

## 【实验感受】

* Shell脚本语法简单、使用便捷、上手比较快，但老版本的linux与现代的linux对shell的语法不同，常常带来意想不到的问题。
* 通过这次实验，也掌握了vi编辑文本的技能，还有一些其他的常用Linux命令
* 这次实验比较简单，做的也没有障碍。