

作业二: shell 脚本程序设计报告

邵盛栋

信息与计算科学 3200103951

2022 年 6 月 28 日

shell 脚本文件代码

```
#!/bin/sh
```

```
salutation="Hello"
echo $salutation
echo "$salutation"
echo '$salutation'
echo \ $salutation
echo Hello
echo "The program $0 is now running"
echo "The second parameter was $2"
echo "The first parameter was $1"
echo "the parameter list was $*"
echo "The user's home directory is $HOME"
echo "Please enter a new greeting"
read salutation
echo $salutation
```

```
a="Hello"
b="hello"
if [ a = b ]; then
echo same
else
echo different
fi
```

```
for foo in hello world !
do
echo $foo
done
```

```
gender() {  
    echo "Are you a man?"  
    while true  
    do  
        echo -n "Enter yes or no:"  
        read x  
        case "$x" in  
            y | yes ) return 0;;  
            n | no )  return 1;;  
            * )      echo "Answer yes or no"  
        esac  
    done  
}  
if gender  
then  
    echo "Hello,sir."  
else  
    echo "Hello,madam."  
fi  
  
exit 0
```

输出与说明

- 1) 第一段程序主要学习的是变量的使用，当我需要为变量赋值时，我只需要使用变量名，该变量就会被自动创建。一种检查变量内容的方法是在变量名前加\$ 符号，再用echo 命令将它的内容输出到终端上。如果把一个\$ 变量表达式放在双引号中，程序执行到这一行时就会把变量替换为它的值；如果把它放在单引号中，就不会发生替换现象。通过在\$ 字符前面加上一个\字符以取消它的特殊含义。使用 read 可以修改变量的值。

对于环境变量和参数变量经过归纳后有如下说明：

环境变量	说明
\$HOME	当前用户的家目录
\$PATH	以冒号分隔的用来搜索命令的目录列表
\$PS1	命令提示符，通常是\$ 字符
\$PS2	二级提示符，用来提示后续的输入，通常是> 字符
\$IFS	输入域分隔符。当 shell 读取输入时，它给出用来分隔单词的一组字符，它们通常是空格、制表符和换行符
\$0	shell 脚本的名字
\$#	传递给脚本的参数个数
\$\$	shell 脚本的进程号

参数变量	说明
\$1,\$2,……	脚本程序的参数
\$*	在一个变量中列出所有的参数，各个参数之间用环境变量 IFS 中的第一个字符分隔开
\$@	\$* 精巧的变体，不使用 IFS 环境变量

赋予第一段程序相应参数，并且输入相应的字符，将得到如下的输出结果：

```

Hello
Hello
$salutation
$salutation
Hello
The program test.sh is now running
The second parameter was var
The first parameter was par
the parameter list was par var
The user's home directory is /home/student
Please enter a new greeting
right
right

```

- 2) 第二、三两段程序学习的是控制语句的使用，在这里我以 if 语句和 for 语句为例。对于这些语句可与其他语言找到共同点，易于理解，包括 while 语句、case 语句、列表等等，其中 for 语句的循环值通常是字符串。

运行第两段程序可得到如下结果：

```
different
hello
world
!
```

- 3) 第四段程序学习的是 shell 中的函数功能，我在这里举了一个判断性别的例子。脚本程序开始执行时，函数 gender 被定义，但先不会执行。在 if 语句中，脚本程序执行到函数 gender 时，会根据输入的内容向调用者返回一个值，if 结构再根据这个值去执行相应的语句。

输入 “yes” 后，第四段程序的运行结果如下：

```
Are you a man?
Enter yes or no:yes
Hello,sir.
```

除上述内容之外，在 shell 脚本程序中还有两类命令，分别为外部命令 (external command) 和内部命令 (internal command)，常用的命令有 break 命令、: 命令、continue 命令、. 命令、echo 命令、eval 命令等。在执行命令时，我们可以通过 \$(command) 语法来实现获取命令的输出。