****

**综 合 实 验 报 告**

学年学期： 2022 - 2023 学年第 二 学期

课程名称： 脚本编程技术

专 业： 21软件工程

班 级：

姓 名：

学 号：

指导教师： 黄子纯

工学院

二O二三 年 六月 二十日

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **实验项目名称** | | **Shell编程综合实验** | | | | |
| **时间** | **2023.6.20** | | **地点** | **实I-A203** | **学时** | **16** |
| **一、实验目的**  考核学生综合应用技能：1.熟练掌握Shell的常用命令及编程基础语法2.掌握Bash程序结构(顺序结构/分支结构/循环结构)3.掌握Shell函数定义、传参及调用4.熟悉Shell正则表达式5.掌握Shell脚本输入处理。 | | | | | | |
| **二、实验内容和要求**  **实验内容：**  综合运用所学的Shell脚本编程知识，设计和实现一个综合应用脚本程序。具体要求如下：  1.执行脚本，输出提示用户选择的菜单信息如下：  1)Shell常用命令  2)石头剪刀布游戏  3)加减乘除计算器  4）学生成绩管理系统  2.以上提示信息包含的4个功能，通过函数实现。4个函数可分别取名：shell\_commands、shell\_stonegame、shell\_calculator、student\_score  3.用户输入对应数字，具体实现要求如下：  用户输入1，则调用函数shell\_commands函数，分类打印常用的shell命令。  用户输入2，则调用函数shell\_stonegame函数，进入石头剪刀布游戏。  用户输入3，则调用shell\_calculator函数，实现一个加减乘除计算器。  用户输入4，则调用student\_score函数，实现一个学生成绩管理系统。  **实验要求：**  1、完成报告，报告结构合理，图文并茂，逻辑思路清晰，没有明显的表达错误和雷同观点和结果，前后描述一致，总结到位。  2、报告命名方式：：**班级\_学号\_姓名.doc**；  **（例：21软件工程B1班\_202110610101\_XXX）**  3、实验方法和步骤必须要有文字描述（例：介绍设计背景，描述设计思路，介绍相关技术），**字数要求大于800字**。  4、实验结果可截图，但需有文字解释截图属于哪部分功能。  **5、文字格式要求：首行缩进2个字符，1.5倍行距，字体为宋体小四。** | | | | | | |
| **三、实验环境**  **Linux虚拟机Centos8系统** | | | | | | |
| **四、实验方法和步骤**  1. Shell常用命令  1.1 常用命令介绍：  该功能主要是打印出一些常用的Shell命令，并提供相应的示例。  1.2 设计思路：  通过在函数`shell\_commands()`中使用`echo`语句输出常用命令和示例。  1.3 具体实现步骤和代码：  # 定义函数：打印Shell常用命令  shell\_commands() {  echo "常用的Shell命令如下："  echo "1) ls：列出当前目录下的文件和文件夹"  echo " 示例：ls"  echo  echo "2) cd：切换当前目录"  echo " 示例：cd /path/to/directory"  echo  echo "3) cp：复制文件或目录"  echo " 示例：cp source\_file target\_file"  echo  echo "4) rm：删除文件或目录"  echo " 示例：rm file\_or\_directory"  echo  echo "5) mv：移动文件或目录"  echo " 示例：mv source\_file target\_file"  }  2. 石头剪刀布游戏  2.1 相关背景介绍：  这个功能是一个石头剪刀布游戏，用户根据提示选择手势，并与电脑进行对战。  2.2 设计思路：  通过定义函数`shell\_stonegame()`来实现游戏的逻辑。  首先，输出欢迎信息和选择提示。  然后，使用`read`命令获取用户输入的手势，并通过`case`语句判断用户输入的手势。  接下来，生成电脑的手势，并判断胜负关系，输出结果。  2.3 具体实现步骤和代码：  # 定义函数：石头剪刀布游戏  shell\_stonegame() {  echo "欢迎进入石头剪刀布游戏！"  echo "请根据提示选择你的手势："  echo "1) 石头"  echo "2) 剪刀"  echo "3) 布"    read -p "请输入数字选择手势：" gesture  case $gesture in  1)  user\_gesture="石头"  ;;  2)  user\_gesture="剪刀"  ;;  3)  user\_gesture="布"  ;;  \*)  echo "无效的输入，请重新运行游戏。"  return  ;;  esac  computer\_gesture=$((RANDOM % 3 + 1))  case $computer\_gesture in  1)  computer\_gesture="石头"  ;;  2)  computer\_gesture="剪刀"  ;;  3)  computer\_gesture="布"  ;;  esac  echo "你选择了：$user\_gesture"  echo "电脑选择了：$computer\_gesture"  if [[ $user\_gesture == $computer\_gesture ]]; then  echo "平局！"  elif [[ ($user\_gesture == "石头" && $computer\_gesture == "剪刀") || ($user\_gesture == "剪刀" && $computer\_gesture == "布") || ($user\_gesture == "布" && $computer\_gesture == "石头") ]]; then  echo "你赢了！"  else  echo "你输了！"  fi  }  3. 加减乘除计算器  3.1 相关背景介绍：  该功能是一个简单的加减乘除计算器，用户输入两个数和运算操作，输出结果。  3.2 设计思路：  通过定义函数`shell\_calculator()`来实现计算器功能。  首先，使用`read`命令获取用户输入的两个数字和运算操作。  然后，通过`case`语句根据不同的运算操作进行相应的计算，并输出结果。  3.3 具体实现步骤和代码：  # 定义函数：加减乘除计算器  shell\_calculator() {  read -p "请输入第一个数字：" num1  read -p "请输入第二个数字：" num2  echo "请选择运算操作："  echo "1) 加法"  echo "2) 减法"  echo "3) 乘法"  echo "4) 除法"  read -p "请输入数字选择运算操作：" operation  case $operation in  1)  result=$(($num1 + $num2))  operator="+"  ;;  2)  result=$(($num1 - $num2))  operator="-"  ;;  3)  result=$(($num1 \* $num2))  operator="\*"  ;;  4)  result=$(($num1 / $num2))  operator="/"  ;;  \*)  echo "无效的输入，请重新运行计算器。"  return  ;;  esac  echo "$num1 $operator $num2 = $result"  }  4. 学生成绩管理系统  4.1 相关背景介绍：  该功能是一个简单的学生成绩管理系统，用户可以录入学生信息、从文件读取学生信息、打印学生信息和保存学生信息到文件。  4.2 设计思路：  通过定义函数`student\_score()`来实现学生成绩管理系统。  首先，在一个无限循环中，输出选择操作的提示，并使用`read`命令获取用户输入的操作选项。  然后，使用`case`语句根据不同的选项执行相应的操作。  4.3 具体实现步骤和代码：  # 定义函数：学生成绩管理系统  student\_score() {  while true; do  echo "请选择操作："  echo "1) 录入学生信息"  echo "2) 从文件读取学生信息"  echo "3) 打印学生信息"  echo "4) 保存学生信息到文件"  echo "5) 返回主菜单"  read -p "请输入数字选择操作：" choice  case $choice in  1)  read -p "请输入学生姓名：" name  read -p "请输入学生成绩：" score  echo "$name $score" >> students.txt  echo "学生信息已录入。"  ;;  2)  if [[ -f "students.txt" ]]; then  echo "从文件中读取的学生信息如下："  cat students.txt  else  echo "文件不存在或为空。"  fi  ;;  3)  if [[ -f "students.txt" ]]; then  echo "学生信息如下："  cat students.txt  else  echo "文件不存在或为空。"  fi  ;;  4)  if [[ -f "students.txt" ]]; then  cp students.txt students\_backup.txt  echo "学生信息已保存到文件students\_backup.txt。"  else  echo "文件不存在或为空。"  fi  ;;  5)  return  ;;  \*)  echo "无效的输入，请重新选择操作。"  ;;  esac  done  } | | | | | | |

|  |
| --- |
| 1. **实验结果（可截图）**   **1 菜单运行界面**    **2 Shell常用命令运行界面**    **3 石头剪刀布游戏运行界面**    **4 加减乘除计算器运行界面**    **5 学生成绩管理系统运行界面** |
| **六、实验总结**   1. 程序功能设计合理：程序通过主菜单给用户提供了多个选择，每个功能都被封装为一个函数，这样可以使程序结构清晰、易于维护和扩展。 2. 用户交互友好：程序通过打印提示信息和获取用户输入来与用户进行交互，用户可以根据自己的需求选择相应的功能。这种交互方式简单直观，便于用户理解和操作。 3. 功能实现正确性：在测试过程中，我逐个选择了各个功能，并进行了多次测试，结果显示所有功能均能正常运行并输出正确的结果。这表明程序在功能实现上是正确的。 4. 代码可读性良好：程序的代码结构清晰，函数命名规范，注释详细，这使得代码易于阅读和理解。此外，代码还有良好的缩进和格式化，提高了代码的可读性。 5. 可扩展性强：由于每个功能都被封装为一个函数，所以可以很方便地添加新的功能或修改现有功能。这种可扩展性使得程序具有更好的适应性和灵活性。 |
| **批阅教师评语：**  评定等级： 批阅教师签名： 年 月 日 |