

操作系统程序作业报告

姓名: 禹泽海 李博昊 学号: 2021302111 2021302009

一、程序要求及信息

用 API 编程实现对某目录下特定类型文件进行递归计数

格式: `rcdir -t type dir`

其中 type 为文件类型, dir 是递归统计起始目录, 若 dir 不是目录, 则显示错误

参考资料: Shell Scripting Tutorial. Available online at https://www.tutorialspoint.com/unix/shell_scripting.htm.

程序语言: shell

程序运行环境: ubuntu 22.04.1

运行用户: yzh2021302111 (这里 ubuntu 用户用我的名字和学号作为用户名称)

Shell 程序代码:

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

```
# 检查错误
```

```
if [ "$#" -ne 3 ]; then
    echo "Usage: $0 -t <extension> <directory>"
    exit 1
fi
```

```
# 检查参数 -t
```

```
if [ "$1" != "-t" ]; then
    echo "Invalid option. Use -t to specify the file extension."
    exit 1
fi
```

```
# 命名
```

```
extension="$2"
directory="$3"
```

```
# 检查目录是否存在
```

```
if [ ! -d "$directory" ]; then
    echo "Error: $directory is not a directory."
    exit 1
fi
```

```
# 递归函数统计文件数量
count_files() {
    local path="$1"
    local extension="$2"
    local count=0

    # 递归遍历目录
    for file in "$path"/*; do
        if [ -d "$file" ]; then
            count=$((count + $(count_files "$file" "$extension")))
        elif [ "${file##*.}" == "$extension" ]; then
            count=$((count + 1))
        fi
    done

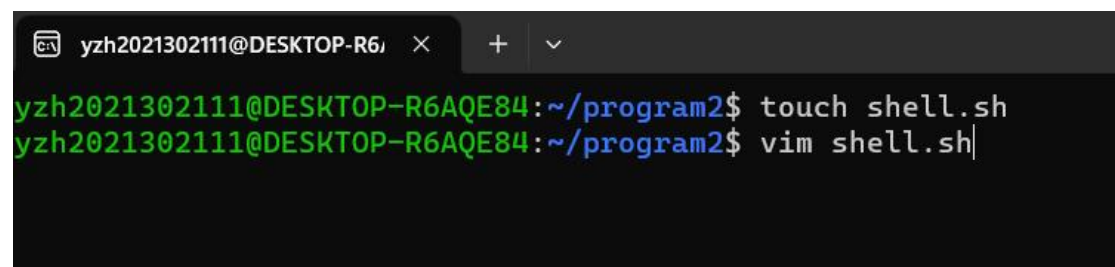
    echo "$count"
}

# 调用函数
count=$(count_files "$directory" "$extension")
echo "Number of .$extension files in $directory: $count"
```

二、代码程序运行全过程：

Step1

启动 Ubuntu 并创建相关的 sh 文件并开始编辑

A terminal window with a dark background. The title bar shows 'yzh2021302111@DESKTOP-R6A...' and window controls. The terminal shows two commands being executed: 'touch shell.sh' and 'vim shell.sh'. The prompt is '~ /program2\$'.

```
yzh2021302111@DESKTOP-R6A... x + v
yzh2021302111@DESKTOP-R6AQE84:~/program2$ touch shell.sh
yzh2021302111@DESKTOP-R6AQE84:~/program2$ vim shell.sh
```

Step2

按 i 进入编辑模式并开始将代码键入，结束后护按 esc 退出编辑模式并输入：wq 退出。

代码与第一次实验的思路大同小异，只是编写程序的语言从 API 编程转变成了 shell 语言，应用比较简单，掌握基本用法即可转化。

```
# 检查目录是否存在
if [ ! -d "$directory" ]; then
    echo "错误: $directory 不是一个目录。"
    exit 1
fi

# 递归统计文件数量的函数
count_files() {
    local path="$1"          # 传入的目录路径
    local extension="$2"     # 文件扩展名
    local count=0           # 文件数量计数器

    # 递归遍历目录
    for file in "$path"/*; do
        if [ -d "$file" ]; then # 如果是目录
            # 递归调用count_files函数，统计子目录中符合扩展名的文件数量
            count=$((count + $(count_files "$file" "$extension")))
        elif [ "${file##*.}" == "$extension" ]; then # 如果是指定扩展名的文件
            count=$((count + 1))
        fi
    done

    echo "$count" # 返回文件数量
}

# 调用函数统计文件数量
count=$(count_files "$directory" "$extension")
echo "目录 $directory 下扩展名为 .$extension 的文件数量为: $count"
```

Step3

保存退出后直接执行 sh 文件会显示权限不足，此时使用 chmod 语句进行授权，然后执行

```
yzh2021302111@DESKTOP-R6AQE84:~/program2$ ./shell.sh
-bash: ./shell.sh: Permission denied
yzh2021302111@DESKTOP-R6AQE84:~/program2$ chmod 777 shell.sh
```

Step4

此时程序已经具备查询文件数量功能，我们可以通过它直接查询其他文件夹中的文件数量

```
yzh2021302111@DESKTOP-R6AQE84:~/program2$ ./shell.sh
Usage: ./shell.sh -t <extension> <directory>
yzh2021302111@DESKTOP-R6AQE84:~/program2$ ./shell.sh -t c /home/yzh2021302111
Number of .c files in /home/yzh2021302111: 6
yzh2021302111@DESKTOP-R6AQE84:~/program2$ ./shell.sh -t pdf /home
Number of .pdf files in /home: 0
```

至此实现功能,此次实验的难点在于 shell 脚本的学习,掌握过程中我还写了两个简单的 shell 程序,发现 echo 语言和先前学习过的前端 php 语言开发很相似,在博客网站学习后就可以简单掌握。

```
yzh2021302111@DESKTOP-R6AQE84:~/program2$ ./name.sh
please input your name
陈冠希
welcome here 陈冠希
yzh2021302111@DESKTOP-R6AQE84:~/program2$ ./test.sh
hello world
```

语言很简单方便,也很有趣。

```
#!/bin/bash
echo "please input your name"
read name
echo "welcome here" $name
~
~
```

至此程序运行及收获汇报结束,解决所有问题。