桂林电子科技大学

**实验四 shell编程**  实验报告

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 实验名称 | **实验4 shell编程** | | | | | | | |  | 辅导员意见：  成绩 辅导员  签 名 |
| 院 系 | 计算机与信息安全学院 | | | 专业 | | 计算机科学与技术 | | |
| 学 号 | 1600300128 | | | 姓名 | | 王涛 | | |
| 实验日期 | 2018 | 年 | 12 | | 月 | | 8 | 日 |
|  |  | | | | | | | |

## 实验目的

1. 掌握Bash shell的建立和执行方式；
2. 掌握Bash shell的基本语法，并能够编写shell脚本。

## 实验要求

1. 理解shell中的特殊字符、变量、控制结构；
2. 能够针对具体问题编写shell脚本；
3. 要求使用vim/vi编写shell。

## 实验步骤

### 一、编辑、理解、运行shell脚本

1、 利用vim编辑器编写一个shell脚本程序filemenu：

说明：要求输入q|Q|5三个中的任何一个时，输出"good bye"，并退出菜单。

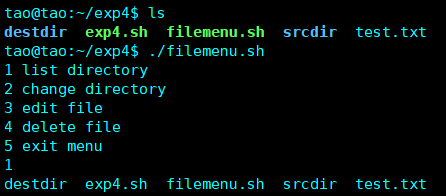
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27 | until  echo 1 list directory  echo 2 change directory  echo 3 edit file  echo 4 delete file  echo 5 exit menu  read choice  test $choice = 5  do  case $choice in  1) ls  ;;  2) read dir  cd $dir  pwd  ;;  3) read file  vim $file  ;;  4) read file  rm $file  ;;  q|Q|5) echo "good bye"  ;;  \*) echo "illegal option";;  esac  done |

2、 调试该程序使其能正常工作，运行之。

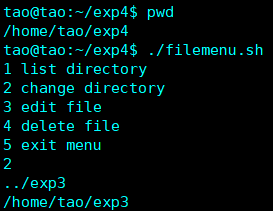
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29 | until  echo 1 list directory  echo 2 change directory  echo 3 edit file  echo 4 delete file  echo 5 exit menu  read choice  false  do  case $choice in  1) ls  ;;  2) read dir  cd $dir  pwd  ;;  3) read file  vim $file  ;;  4) read file  rm $file  ;;  q|Q|5) echo "good bye"  exit  ;;  \*) echo "illegal option"  ;;  esac  done |

1. 解释这个程序的功能。

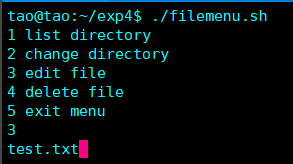
1.列出当前目录文件



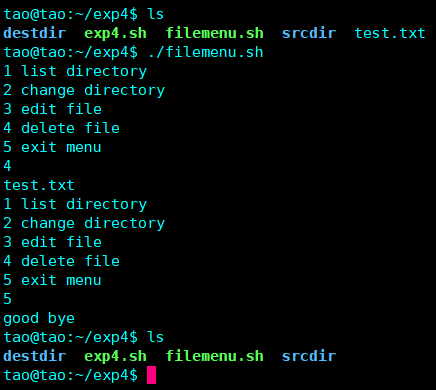
2.切换到目标目录后列出所在目录的绝对路径



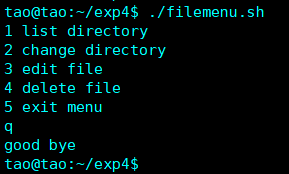
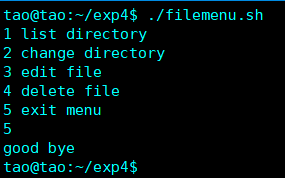
3.vim打开指定文件

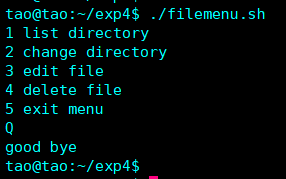


4.删除指定文件



5、Q、q.退出





### 二、根据要求设计编写shell脚本

（1）利用for循环将指定源目录srcdir下的.c文件移动到指定目的目录destdir下；

（2）将目录destdir下的文件，按照文件大小排序，显示移动后指定目录内容；

（3）将指定目录中文件大小小于指定大小的文件按照原目录结构进行打包，压缩成指定文件名.tar.gz文件。

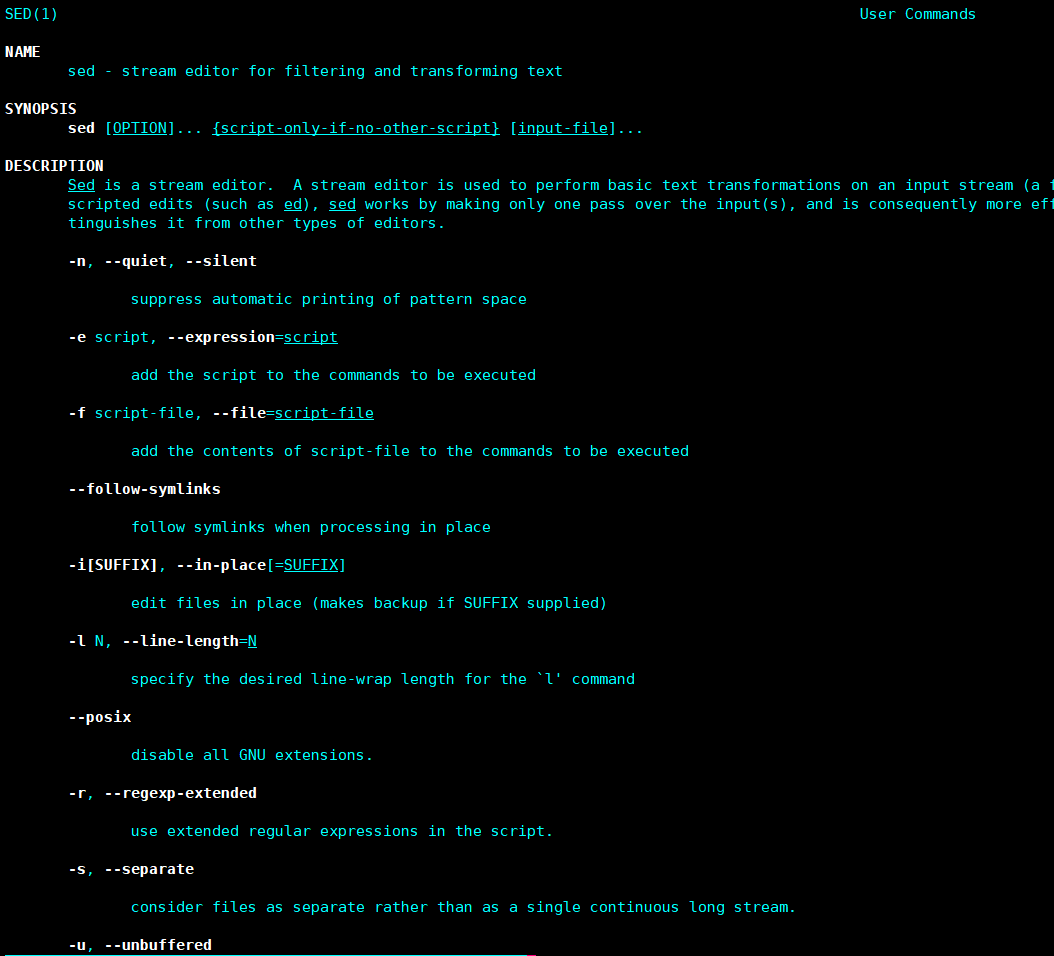
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40 | while [ ! -n "$srcdir" -o ! -d "$srcdir" ]  do  read -p "Please input the srcdir:" srcdir  done  while [ ! -n "$destdir" ]  do  read -p "Please input the destdir:" destdir  done  if [ ! -d "$destdir" ]; then  mkdir $destdir  fi  for f in $(ls $srcdir | grep .c$)  do  cp $srcdir/$f destdir  done  ls -S $destdir  read -p "Please input tarname(Enter to skip the step):" tarname  if [ -n "$tarname" ]; then  while [ ! -n "$size" ]  do  read -p "Please input the most min size:[b/k/m]" size  case ${size:0-1} in  b|B) size=${size%?};;  k|K) size=$[${size%?}\*1024];;  m|M) size=$[${size%?}\*1024\*1024];;  \*) size="" ;;  esac  done  mkdir $destdir/$tarname  for f in $(ls $destdir)  do  if [ $size -gt $(du -b $destdir/$f | awk '{print $1}') ];  then  cp $destdir/$f $destdir/$tarname  fi  done  tar -zcf $destdir/$tarname.tar.gz $destdir/$tarname  rm -rf $destdir/$tarname  fi  echo "Program has finished!" |



### 三、实验扩展：sed和awk学习

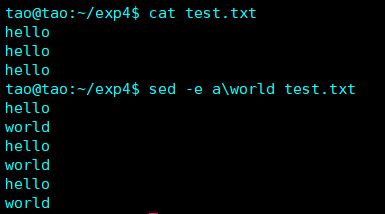
1. 学习流编辑器sed

使用man命令查看sed的帮助文档。

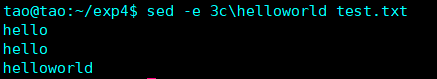


示例：

插入：



替换：

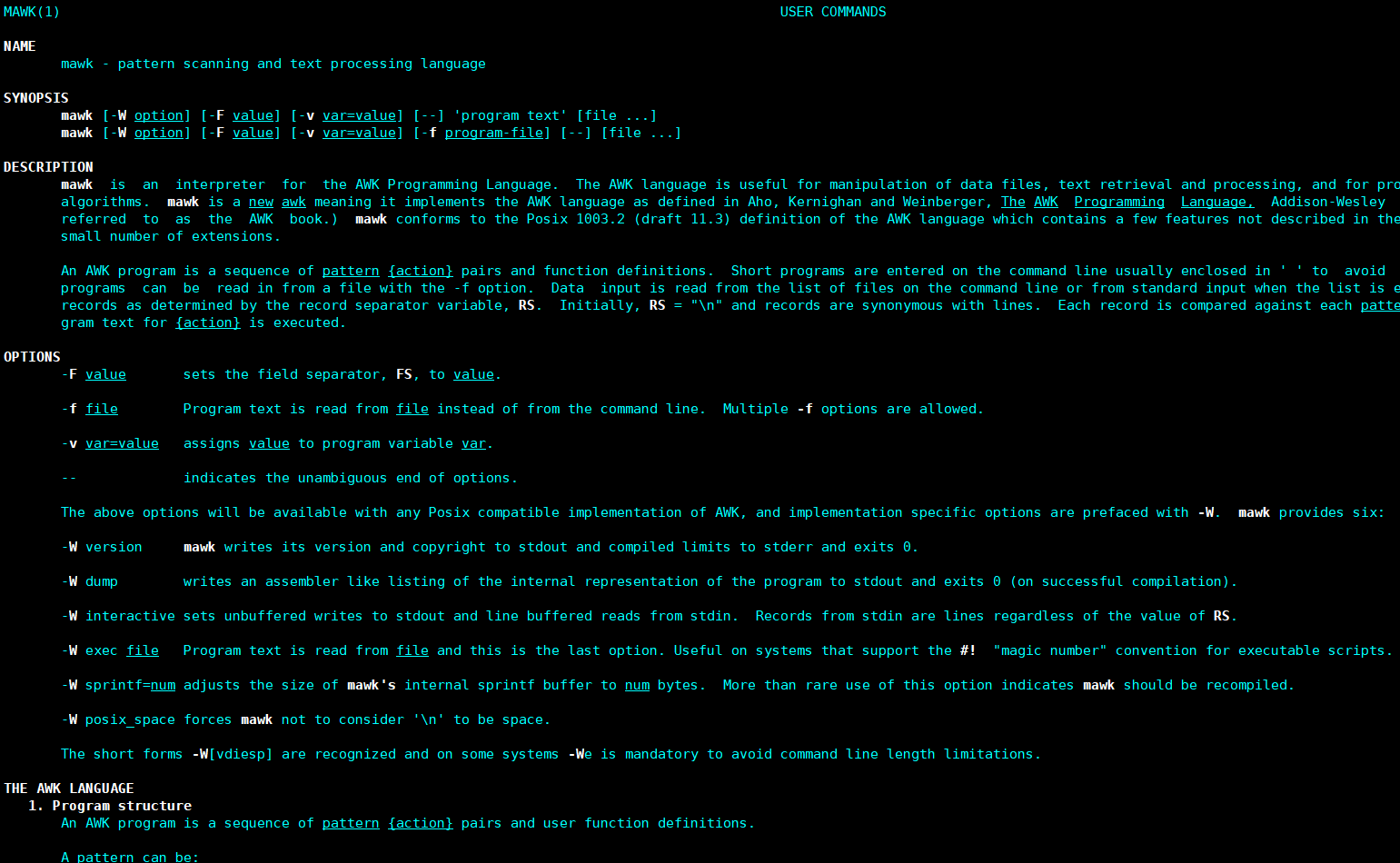


删除：

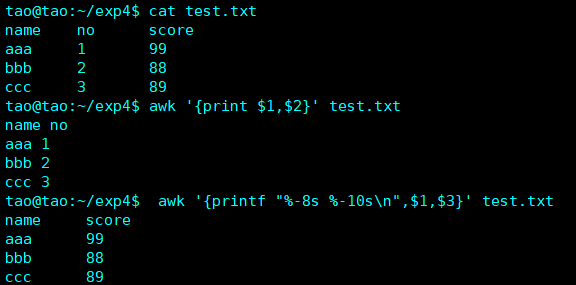


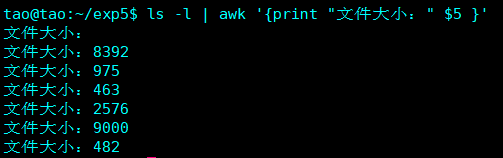
2学习格式化报表工具awk

使用man命令查看awk的帮助文档。



示例：





## 问题记录和实验总结

问题记录：

1.编写的shell脚本无法执行

解决：无法执行往往是因为文件没有执行权限，通过chmod命令给问价加上x权限，文件会变成可执行的颜色，便可执行。

2.shell常见语法错误

解决：shell脚本的语法比较严格，比如if后[]里面左右一定要有空格，同时写代码时先把语句结构的开始语句和结束语句写上再去中间去写逻辑代码，可以有效地降低代码错误，shell的与或及字符串截取等操作与之前学的语言有很大差距，需要慢慢熟练。

3.shell编程中的路径问题

解决：编写shell脚本时一定要注意程序所执行的路径，编写的程序可能只能在特定的路径下执行，在其他的路径执行就会出现错误，需考虑脚本的英语场景为了提高程序的可用性，要注意相对路径和绝对路径的使用。

通过本次实验，我学会了shell基本语法，类比之前所学的编程知识，在了解shell的特殊字符、变量、控制结构等语法规范后，可以很容易上手，但shell编程包括一些细节方面是和传统的编程语言有很多不同的，比如shell脚本中可以直接写Linux命令，使得shell编程的功能更加丰富。在shell编程中结合awk等命令的使用，可以更方便的实现想要的功能。使用vim进行shell编写，更加熟练了vim的使用，最终可以达到自己独立完成程序解决需要解决的问题，但需要经常查看shell语法文档，希望在之后的使用中慢慢记住，更加熟练的掌握。