实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 实验名称 | 实验一 Linux常用命令（一） | | |
| 实验教室 | 丹青922 | 实验日期 | 2021年5月12日 |
| 学 号 | 2019224278 | 姓 名 | 张泽晨 |
| 专业班级 | 计算机科学与技术07班 | | |
| 指导教师 | 卢洋 | | |

东北林业大学

信息与计算机科学技术实验中心

|  |
| --- |
| 1. 实验目的   1、掌握Linux下文件和目录操作命令：cd、ls、mkdir、rmdir、rm  2、掌握Linux下文件信息显示命令：cat、more、head、tail  3、掌握Linux下文件复制、删除及移动命令：cp、mv  4、掌握Linux的文件排序命令：sort |
| 1. 实验环境   （1）计算机的硬件配置PC系列微机。  （2）计算机的软件配置VMware虚拟机软件及Ubuntu虚拟机。 |

|  |
| --- |
| 1. 实验内容及结果   1.使用命令切换到/etc目录，并显示当前工作目录路径    2、使用命令显示/home/lyj目录下所有文件目录的详细信息，包括隐藏文件。    3、使用命令创建目录/home/lyj/linux，然后删除该目录。    4、使用命令cat用输出重定向在/home/lyj目录下创建文件abc，文件内容为“Hello, Linux!”，并查看该文件的内容    5、使用命令创建目录/home/lyj/ak，然后将/home/lyj/abc文件复制到该目录下，最后将该目录及其目录下的文件一起删除。    6、查看文件/etc/adduser.conf的前3行内容，查看文件/etc/adduser.conf的最后5行内容。      分屏查看文件/etc/adduser.conf的内容。    8、使用命令cat用输出重定向在/home/lyj目录下创建文件facebook.txt，文件内容为：  google 110 5000  baidu 100 5000  guge 50 3000  sohu 100 4500    9．第一列为公司名称，第2列为公司人数，第3列为员工平均工资。  利用sort命令完成下列排序：     1. 按公司字母顺序排序      1. 按公司人数排序      1. 按公司人数排序，人数相同的按照员工平均工资升序排序      1. 按员工工资降序排序，如工资相同，则按公司人数升序排序     （5）从公司英文名称的第2个字母开始进行排序。 |

|  |
| --- |
| 1. 实验过程分析与讨论   Sort命令   * 功能说明：将文本文件内容加以排序,sort可针对文本文件的内容，以行为单位来排序。 * 格式：sort [选项] filename * -m 将已排序的输入文件，合并为一个排序后的输出数据流。 * -n 以整数类型比较字段 * -o outfile 将输入写到指定文件，而非标准输出。如果该文件为输入文件之一，则sort 在进行排序写到输入文件之前，会先将它复制到一个临时文件 * -r 倒置排序的顺序为 由大至小（descending）,而非默认的由小至大（ascending） * -t char 使用单个字符char作为默认的字段分割字符，取代默认的空白字符。 * -u 只有唯一的记录，丢弃所有具有相同键值的记录，只留其中的第一条。只有键值字段是重要的，也就是说：被丢弃的记录其他部分可能是不同值。 * 行为模式：sort 会读取指定的文件，如果未给定文件，则读取标准输入，在将排序好的数据写至标准输出。 * -b 忽略开头的空白 * -c 检查输入是否已正确排序，如输入未经排序，但退出码(exit code)为非零值，则不会有任何输出 * -d 字典顺序：仅文字数字与空白才有意义 * -g 一般数值：以浮点数字类型比较字段。这个选项的运作有点类似 -n.差别仅在于这个选项的数字可能有小数点及指数。（仅GNU版本提供此功能） * -f 以不管字母大小写的方式排序 * -i  忽略无法打印的字符 |
| 五、指导教师意见  指导教师签字：卢洋 |

实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 实验名称 | **Linux常用命令（二）** | | |
| 实验教室 |  | 实验日期 | 2021年5月19日 |
| 学 号 | 2019224278 | 姓 名 | 张泽晨 |
| 专业班级 | 计算机科学与技术7班 | | |
| 指导教师 | 卢洋 | | |

东北林业大学

信息与计算机科学技术实验中心

|  |
| --- |
| 1. 实验目的   1、掌握Linux下查找文件和统计文件行数、字数和字节数命令：find、locate 、wc  2、掌握Linux下文件打包、压缩命令：tar gzip  3、掌握Linux下符号链接命令和文件比较命令：ln、comm、diff  4、掌握Linux的文件权限管理命令：chmod chown |
| 1. 实验环境   （1）计算机的硬件配置PC系列微机。  （2）计算机的软件配置VMware虚拟机软件及Ubuntu虚拟机。 |
| 1. 实验内容及结果   1、查找指定文件  （1）在用户主目录下新建目录locate，在locate下新建文件newfile，内容随意写几行。    （2）在用户主目录下查找文件newfile，并显示该文件位置信息。    （3）统计newfile文件中所包含的行数、字数和字节数。    （4）创建文件newfile1，在用户主目录下查找比文件newfile更新的文件。    （4）在用户主目录下查找文件newfile，并删除该文件。    （5）查看文件夹locate内容，看一下是否删除了文件newfile。    2、文件打包  （1）在用户主目录下新建文件夹m1，在m1下新建文件f1和f2。    （2）在用户主目录下新建文件夹m2，在m2下新建文件f3。    （3）在用户主目录下新建文件f4。    （4）在用户主目录下对文件夹m1和f4进行打包，生成文件bao1.tar。    （5）查看包bao1.tar的内容。    （6）向包bao1.tar里添加文件夹m2的内容。    （7）将包bao1.tar复制到用户主目录下的新建文件夹m3中。    （8）进入m3文件夹，并还原包bao1.tar的内容。    3、符号链接内容  （1）新建文件a.txt,内容为12345。    （2）建立a.txt得硬链接文件b.txt，并比较b.txt的内容和a.txt是否相同，要求用comm或diff命令。    （3）查看a.txt和b.txt的i节点号(inode)是否相同。    （4）修改b.txt的内容为123456，然后通过命令判断a.txt与b.txt是否相同。    （5）删除a.txt文件，然后查看b.txt文件的inode及内容。    （6）建立文件b.txt的符号链接文件c.txt，然后查看b.txt和c.txt的inode号，观察两者是否相同，比较b.txt和c.txt的文件内容是否相同。      （7）删除b.txt后查看c.txt，观察系统给出什么提示信息。    4、权限管理  （1）新建文件tt.txt ，是否能创建。    （2）增加写权限，创建文件tt.txt并为该文件增加执行权限（所有用户都可以执行）。    （4）为文件tt.txt去除组和其它用户的执行权限。    （5） 更改文件的所有者。 |

|  |
| --- |
| 1. 实验过程分析与讨论   运行updatedb命令时，显示数据库被锁  Updatedb  updatedb命令用来创建或更新locate/slocate命令所必需的数据库文件。updatedb命令的执行过程较长，因为在执行时它会遍历整个系统的目录树，并将所有的文件信息写入locate/slocate数据库文件中。  *# 更新指定命令的slocate数据库* updatedb -U /usr/local/ |
| 五、指导教师意见  指导教师签字： 卢洋 |

实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 实验名称 | **实验三 vi编辑器及gcc编译器的使用** | | |
| 实验教室 |  | 实验日期 | 2021年5月17日 |
| 学 号 | 2019224278 | 姓 名 | 张泽晨 |
| 专业班级 | 计算机科学与技术7班 | | |
| 指导教师 | 卢洋 | | |

东北林业大学

信息与计算机科学技术实验中心

|  |
| --- |
| 1. 实验目的   掌握vi编辑器及gcc编译器的使用方法 |
| 1. 实验环境   （1）计算机的硬件配置PC系列微机。  （2）计算机的软件配置VMware虚拟机软件及Ubuntu虚拟机。 |
| 1. 实验内容及结果   1、vi编辑器和gcc编译器的简单使用  （1）在用户主目录下新建一个目录，命名为vifile    （2）进入目录vifile    （3）在vifile下用vi编辑器新建一个c语言程序文件，文件名为test.c  test.c文件内容为：  int main( )  {  printf(“hello world!\n”);  }      （4）保存test.c的内容，并退出    （5）编译test.c文件，生成可执行文件test，并执行test，查看执行结果。    2、vi编辑器的详细使用  （1）在用户主目录下建一个名为vi的目录。    （2）进入vi目录。    （3）将文件/etc/gai.conf复制到当前目录下，并用命令sudo修改gai.conf的属性为所有用户可以读写。    （4）使用vi编辑当前目录下的gai.conf。    （5）显示行号。    （6）将光标移到第18行。  gg+17+↓    （7）复制该行内容。  yy  （8）将光标移到最后一行行首。  G  （9）粘贴复制行的内容。  p    （10）存盘但不退出。  :w    （11）将光标移到首行。  gg  （12）插入模式下输入“Hello, this is vi world!”。    （13）删除字符串“this“。  光标指向this首字母+dw    （14）强制退出vi，不存盘。  :q! |

|  |
| --- |
| 1. 实验过程分析与讨论   在运行gcc test.txt -o test时遇到无法编译的情况，通过使用sudo apt-get install build-essential命令解决了该问题 |
| 五、指导教师意见  指导教师签字： 卢洋 |

实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 实验名称 | **实验四 用户和用户组管理** | | |
| 实验教室 |  | 实验日期 | 2021年5月18日 |
| 学 号 | 2019224278 | 姓 名 | 张泽晨 |
| 专业班级 | 计算机科学与技术7班 | | |
| 指导教师 | 卢洋 | | |

东北林业大学

信息与计算机科学技术实验中心

|  |
| --- |
| 1. 实验目的   1、掌握用户管理命令，包括命令useradd，usermod，userdel，newusers  2、掌握用户组管理命令，包括命令groupadd，groupdel  3、掌握用户和用户组维护命令，包括命令passwd，su，sudo |
| 1. 实验环境   （1）计算机的硬件配置PC系列微机。  （2）计算机的软件配置VMware虚拟机软件及Ubuntu虚拟机。 |
| 1. 实验内容及结果   1、建立一个用户名为jone，描述信息为jone，登录shell为/bin/sh，登录主目录为/home/jone的用户，并设置口令为123456。    2、使用命令从用户root切换到用户jone，修改jone的UID为2000，其shell类型为/bin/csh  。    3、使用命令从用户jone切换到root。    4、使用命令删除jone用户，并且在删除该用户的同时一起删除其主目录。    5、使用命令newusers批量创建用户，并使用命令chpasswd为这个批量用户创建密码（密码也是批量创建的），查看/etc/passwd文件确认是否创建成功。        6、使用命令创建用户组group1，并在创建时设置其GID为3000。    7、在用户组group1中添加两个之前批量创建的用户。      8、切换到group1组中的某个用户，在该用户下使用sudo命令查看/etc/shadow文件，看一下是否可以执行。若不能执行，修改sudoers文件使得该用户可以查看/etc/shadow文件内容（尝试两种方法）。 |

|  |
| --- |
| 1. 实验过程分析与讨论   　示例：  　　创建组群china  　　[root@localhost ~]# groupadd china  　　创建组群ou，并且设置该组群GID为800  　　[root@localhost ~]# grouadd -g 800 ou  　　创建系统组群chinese  　　[root@localhost ~]# groupadd -r chinese  主要概念：  1、基本上，一个组就是一个整数组ID（gid）       lzgonline:x: 500: 2、每个在系统上运行的进程都是属于一个组的集合（gids） 3、/etc/group 文件把组ID映射到组名称和组成员身上            /etc/group文件存储格式（组名称：组密码：组ID：组成员）       root:x:0:root       lzgonline:x:500:       字段解释：       组名称：每个组都有一个组名称       组密码：可以给组提供一个密码，一般很少这么做       组ID：像用户ID一样，linux内核使用ID来识别       组成员：定义组成员用户名列表，用半角逗号隔开  4、文件系统中的每个文件有唯一的组ID，就像拥有唯一的所有者ID一样      drwxrwxr-x. 2 lzgonline  lzgonline 4096  6月 23 23:47 coding       drwxr-xr-x. 2 lzgonline  lzgonline  4096  6月 23 22:03 公共的 5、用户有一个在/etc/passwd 文件中定义的主要组（第4个字段定义）       root:x:0:0:root:/root:/bin/bash 6、用户可以在/etc/group 文件中定义多个次要组（例从下面可以看到root用户属于多个组）       root:x:0:root       bin:x:1:root,bin,daemon       daemon:x:2:root,bin,daemon       sys:x:3:root,bin,adm       adm:x:4:root,adm,daemon       disk:x:6:root       wheel:x:10:root   7、在redhat企业版中，用户的主要组几乎总是与用户名相同            /etc/passwd 文件： lzgonline:x:500: 500:liuzhigong:/home/lzgonline:/bin/bash       /etc/group文件：     lzgonline:x: 500:  8、文件系统上的每个文件有一个用户所有者和一个组所有者     如何在linux中查询一个组有哪些用户？ 执行cat /etc/group | less  命令，寻找相应的组名称，查看其最后一个字段即可   如何在linux中查询一个用户属于哪些组？ 执行cat /etc/group | grep username  即可（将username替换为查找的用户名）。 |
| 五、指导教师意见  指导教师签字： 卢洋 |

实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 实验名称 | **实验五 Shell程序的创建及条件判断语句** | | |
| 实验教室 |  | 实验日期 | 2021年5月19日 |
| 学 号 | 2019224278 | 姓 名 | 张泽晨 |
| 专业班级 | 计算机科学与技术7班 | | |
| 指导教师 | 卢洋 | | |

东北林业大学

信息与计算机科学技术实验中心

|  |
| --- |
| 1. 实验目的   1、掌握Shell程序的创建过程及Shell程序的执行方法。  2、掌握Shell变量的定义方法，及用户定义变量、参数位置等。  3、掌握变量表达式，包括字符串比较、数字比较、逻辑测试、文件测试。  4、掌握条件判断语句，如if语句、case语句。 |
| 1. 实验环境   （1）计算机的硬件配置PC系列微机。  （2）计算机的软件配置VMware虚拟机软件及Ubuntu虚拟机。 |
| 1. 实验内容及结果   1、定义变量AK的值为200，并将其显示在屏幕上。（终端上执行）    2、定义变量AM的值为100，并使用test命令比较其值是否大于150，并显示test命令的退出码。（终端上执行）    3、创建一个简单的Shell程序，其功能为显示计算机主机名（hostname）和系统时间（date）      4、创建一个简单的Shell程序，要求带一个参数，判断该参数是否是水仙花数。所谓水仙花数是指一个 3位数，它的每个位上的数字的 3次幂之和等于它本身。例如153=13+33+53，153是水仙花数。编写程序时要求首先进行参数个数判断，判断是否带了一个参数，如果没有参数则给出提示信息，否则给出该数是否是水仙花数。要求对153，124，370分别进行测试判断。      5、创建一个简单的Shell程序，实现输入目录名，查看当前文件夹下有没有这个目录。如果没有则创建该目录，若已存在则输出“exist”。      6、创建一个简单的shell程序，输入学生的成绩，给出该成绩对应的等级，90分以上为A，80-90为B，70-80为C，60-70为D，小于60分为E。要求使用if…elif….else fi实现。 |

|  |
| --- |
| 1. 实验过程分析与讨论   shell中的逻辑判断一般用if语句，if语句中通常用[]来表示条件测试，可以比较字符串、判断文件是否存等。备注：[ ] 中表达式两边与括号之间要有空格  if … else 语句常用基本的语法如下：  1.if [];then fi 语句  建一个测试脚本test.sh如下  #!/bin/bash  a=$1  b=$2  if [ $a == $b ];then  echo "a and b is equal"  fi  if [ $a != $b ];then  echo "a and b is not equal"  fi  执行命令sh test.sh 2 3给参数$1和$2赋值2和3，输出结果a and b is not equal  不加else的if语句表达式成立执行then后面的语句，表达式不成立则不执行任何命令。  2.if [];then else fi 语句  if [ expression ];then  executed Statement\_expression\_true  else  executed Statement\_expression\_false  fi  备注：expression表达式 和方括号[ ]之间必须有空格，否则会有语法错误。如果表达式成立，then后面的语句将会被执行；如果表达式不成立则执行else后面的语句。  3.if [];then elif []; then else fi 语句，哪个expression表达式成立则执行哪个then后面的语句，否则执行else后面的语句。  if [ expression1 ];then  executed Statement\_expression1\_true  elif [ expression2 ];then  executed Statement\_expression2\_true  else  executed Statement\_expression1\_2\_false  fi  #!/bin/bash  a=$1  b=$2  if [ $a == $b ];then  echo "a and b is equal"  elif [ $a -lt $b ];then  echo "a less than b"  else  echo "a bigger than b"  fi  例如建个测试脚本test.sh如上，执行命令sh test.sh 2 3给参数$1、$2赋值2、3，输出结果a less than b；执行sh test.sh 3 2 结果为a bigger then b  #!/bin/bash  a=$1  b=$2  if [ $a == $b ];then  echo "a and b is equal"  else  if [ $a -lt $b ];then  echo "a less than b"  else  echo "a bigger than b"  fi  fi  些  4.if … else 语句也经常与 test 命令结合使用，test 命令用于检查某个条件是否成立，与方括号[ ]功能类似  #!/bin/bash  a=$1  b=$2  if test $a == $b;then  echo "a and b is equal"  else  echo "a and b is not equal"  fi  例如上述脚本，其中if test $a == $b;与if [ $a == $b ];效果一样。  5.if语句常用命令选项有：  == or =： 等于  -eq ： 等于  -ne ：不等于  -gt ：大于  -ge ：大于等于  -lt ：小于  -le ：小于等于 |
| 五、指导教师意见  指导教师签字： 卢洋 |

实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 实验名称 | **实验六 Shell程序的创建及条件判断语句** | | |
| 实验教室 |  | 实验日期 | 2021年5月19日 |
| 学 号 | 2019224278 | 姓 名 | 张泽晨 |
| 专业班级 | 计算机科学与技术7班 | | |
| 指导教师 | 卢洋 | | |

东北林业大学

信息与计算机科学技术实验中心

|  |
| --- |
| 1. 实验目的   （1）熟练掌握Shell循环语句：for、while、until  （2）熟练掌握Shell循环控制语句：break、continue |
| 1. 实验环境   （1）计算机的硬件配置PC系列微机。  （2）计算机的软件配置VMware虚拟机软件及Ubuntu虚拟机。 |
| 1. 实验内容及结果 2. 编写一个shell脚本，利用for循环把当前目录下的所有\*.sh文件复制到指定的目录中，并为没有执行权限的文件添加执行权限。（可以在当前目录下先建立几个\*.sh文件，用来测试，复制到的指定目录可以自己建立一个）          1. 编写shell脚本，利用while循环求前10个偶数之和。        1. 编写shell脚本，利用until循环求1到10的平方和。         （4）运行下列程序，观察程序的运行结果。红色的语句分别为break，break 2，continue，continue2，观察四种情况下的实验结果。  #!/bin/sh  for i in a b c d   do  echo -n $i  　for j in 1 2 3 4 5 6 7 8 9 10      do        if [ $j -eq 5 ];then          break或continue        fi        echo –n “ $j”      done  echo $j   done  1.当为break时，外层for循环只有在内层for循环中的j==5时跳出当前循环，j只输出5         1. 当为break 2时，内层for循环中j==5时，直接跳出内外两层循环，因此只输出外层循环的第一个值a           3.当为continue时，内层for循环中的j==5时，直接跳过当前循环体内剩余的语句，直接进行下一次循环，因为只有当整个内层循环全部执行完之后才输出j，因为最后一次循环j=10，故输出j=10    4.当为continue 2时，内层for循环中j==5时，直接跳过两层循环体剩余的语句，因此不会执行第一层for循环内的 echo $j，而只会执行echo $i |

|  |
| --- |
| 1. 实验过程分析与讨论   for循环使用  1、for循环  （1）for循环有三种结构：一种是列表for循环，第二种是不带列表for循环。第三种是类C风格的for循环。  （2）列表for循环  #!/bin/bash    for varible1 in {1..5}  #for varible1 in 1 2 3 4 5  do  echo "Hello, Welcome $varible1 times "  done  do和done之间的命令称为循环体，执行次数和list列表中常数或字符串的个数相同。for循环，首先将in后list列表的第一个常数或字符串赋值给循环变量，然后执行循环体，以此执行list，最后执行done命令后的命令序列。  Sheel支持列表for循环使用略写的计数方式，1～5的范围用{1…5}表示（大括号不能去掉，否则会当作一个字符串处理）。  Sheel中还支持按规定的步数进行跳跃的方式实现列表for循环，例如计算1～100内所有的奇数之和。  #!/bin/bash  sum=0    for i in {1..100..2}  do  let "sum+=i"  done    echo "sum=$sum"  通过i的按步数2不断递增，计算sum值为2500。同样可以使用seq命令实现按2递增来计算1～100内的所有奇数之和，for i in $(seq 1 2 100)，seq表示起始数为1，跳跃的步数为2，结束条件值为100。  for循环对字符串进行操作，例如通过for循环显示当前目录下所有的文件。  #!/bin/bash    for file in $( ls )  #for file in \*  do  echo "file: $file"  done  也可一使用for file in ，通配符产生文件名扩展，匹配当前目录下的所有文件。  for通过命令行来传递脚本中for循环列表参数  #!/bin/bash    echo "number of arguments is $#"    echo "What you input is: "    for argument in "$@"  do  echo "$argument"  done  $ #表示参数的个数， @ 表 示 参 数 列 表 而 @表示参数列表而 @表示参数列表而\*则把所有的参数当作一个字符串显示。  （3）不带列表for循环  由用户制定参数和参数的个数，与上述的for循环列表参数功能相同。  #!/bin/bash  echo "number of arguments is $#"    echo "What you input is: "    for argument  do  echo "$argument"  done  比上述代码少了$ @参数列表，$\*参数字符串。 |
| 五、指导教师意见  指导教师签字： 卢洋 |

实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 实验名称 | 实验七 Shell函数 | | |
| 实验教室 |  | 实验日期 | 2021年5月20日 |
| 学 号 | 2019224278 | 姓 名 | 张泽晨 |
| 专业班级 | 计算机科学与技术7班 | | |
| 指导教师 | 卢洋 | | |

东北林业大学

信息与计算机科学技术实验中心

|  |
| --- |
| 1. 实验目的   1、掌握Shell函数的定义方法  2、掌握shell函数的参数传递、调用和返回值  3、掌握shell函数的递归调用方法  4、理解shell函数的嵌套。 |
| 1. 实验环境   （1）计算机的硬件配置PC系列微机。  （2）计算机的软件配置VMware虚拟机软件及Ubuntu虚拟机。 |
| 1. 实验内容及结果   编写shell脚本，定义一个函数返回两个数的和。  @TLL5B_ER6L[4RH`RUIGDQ0   1. 编写shell脚本，该脚本中定义一个递归函数，求n的阶乘。       (3)已知shell脚本test.sh内容如下所示，试运行下列程序，观察程序运行结果，理解函数嵌套的含义。  #!/bin/bash  function first() {  function second() {  function third() {  echo "------this is third"  }  echo "this is the second"  third  }  echo "this is the first"  second  }    echo "start..."  first |

|  |
| --- |
| 1. 实验过程分析与讨论   函数调用的相关知识。  **Shell 函数定义的语法格式如下：**  function name() {     statements     [return value] }  对各个部分的说明：  function是 Shell 中的关键字，专门用来定义函数；  name是函数名；  statements是函数要执行的代码，也就是一组语句；  return value表示函数的返回值，其中 return 是 Shell 关键字，专门用在函数中返回一个值；这一部分可以写也可以不写。  由{ }包围的部分称为函数体，调用一个函数，实际上就是执行函数体中的代码。  函数定义的简化写法  如果你嫌麻烦，函数定义时也可以不写 function 关键字：  name() {     statements     [return value] }  如果写了 function 关键字，也可以省略函数名后面的小括号：  function name {     statements     [return value] }  **函数调用**  调用 Shell 函数时可以给它传递参数，也可以不传递。如果不传递参数，直接给出函数名字即可：  name  如果传递参数，那么多个参数之间以空格分隔：  name param1 param2 param3  不管是哪种形式，函数名字后面都不需要带括号。  和其它编程语言不同的是，Shell 函数在定义时不能指明参数，但是在调用时却可以传递参数，并且给它传递什么参数它就接收什么参数。  Shell 也不限制定义和调用的顺序，你可以将定义放在调用的前面，也可以反过来，将定义放在调用的后面。 |
| 五、指导教师意见  指导教师签字： 卢洋 |

实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 实验名称 | 实验八 sed和awk | | |
| 实验教室 |  | 实验日期 | 2021年5月21日 |
| 学 号 | 2019224278 | 姓 名 | 张泽晨 |
| 专业班级 | 计算机科学与技术7班 | | |
| 指导教师 | 卢洋 | | |

东北林业大学

信息与计算机科学技术实验中心

|  |
| --- |
| 1. 实验目的   1、掌握sed基本编辑命令的使用方法  2、掌握sed与shel变量的交互方法  3、掌握awk命令的使用方法  4、掌握awk与shell变量的交互方法 |
| 1. 实验环境   （1）计算机的硬件配置PC系列微机。  （2）计算机的软件配置VMware虚拟机软件及Ubuntu虚拟机。 |
| 1. 实验内容及结果   1、已知quote.txt文件内容如下  The honeysuckle band played all night long for only $90.  It was an evening of splendid music and company.  Too bad the disco floor fell through at 23:10.  The local nurse Miss P.Neave was in attendance.  试编写sed命令实现如下功能：   1. 删除$符号      1. 显示包含music文字的行内容及行号      1. 在第4行后面追加文件“hello world！”      1. 将文本“The”修改为“Ok”      1. 将第3行内容修改为“This is the third line.”      1. 删除第2行内容。      1. 设置shell变量var的值为evening，用sed命令查找匹配var变量值的行。       2、已知文件aaa.txt内容如下‘’  one : two : three  four : five : six  (注：每个冒号前后都有空格)  试编写awk命令实现如下功能：分别以空格和冒号做分隔符，显示第2列的内容，观察两者的区别    如果以一个空格作为分隔符，则冒号会被视为单独的一列  如果以一个冒号作为分隔符，则则会将字段分为5组，且第一组的冒号: 会被保留，且对角线上的元素会被分为一列  3、已知文件b.txt里面都是数字，且每行包含3个数字，数字之前以空格作为分隔符，试将b.txt里的所有偶数输出，并输出偶数的个数。要求：判断每行的3个数字是否为偶数时用循环结果，即要求程序里包含循环和分支结构。  例如：b.txt内容为：  2 4 3  15 46 79  则输出为：  2  4  46    4、已知脚本t.sh的内容如下，试通过运行该脚本，理解该脚本实现的功能。  #!/bin/bash  read –p “enter search pattern: “ pattern  awk “/$pattern/”‘{ nmatches++; print } END { print nmatches “found.” }’ info.txt    awk中"/$pattern/"这一部分用双引号括起来，是为了允许引号内的Shell变量进行替换  此脚本的作用用于匹配字符串  首先输入你要匹配的字符串，脚本中指定的文件为info.txt  并在info.txt文件中查找相应的字符串，如果能匹配到，则nmatches变量就加一，并在最后输出要匹配字符串出现的位置，以及出现的次数 |

|  |
| --- |
| 1. 实验过程分析与讨论   sed和awk的用法：   1. sed 命令的作用是利用脚本来处理文本文件。使用方法：   sed [参数] [n1][n2]function   n1,n2不一定存在，一般表示进行动作的行。如果动作在10-20行进行，则为 10,20[function]  **参数说明**：   * -e或--expression= 以选项中指定的 script 来处理输入的文本文件，这个-e可以省略，直接写表达式。 * -f或--file=以选项中指定的 script 文件来处理输入的文本文件。 * -h或--help显示帮助。 * -n 或 --quiet 或 --silent 仅显示 script 处理后的结果。 * -V 或 --version 显示版本信息。 * -i 直接在源文件里修改内容   **动作说明**[function]：   * a：追加， a 的后面可以接字串，而这些字串会在目标行末尾追加～ * c：取代， c 的后面可以接字串，这些字串可以取代 n1,n2 之间的行！ * d：删除，因为是删除啊，所以 d 后面通常不接任何咚咚； * i：插入， i 的后面可以接字串，而这些字串会在新的一行出现(目前的上一行)； * p：打印，亦即将某个选择的数据印出。通常 p 会与参数 sed -n 一起运行～ * s：取代，通常这个 s 的动作可以搭配正规表示法，例如 1,20s/old/new/g |
| 五、指导教师意见  指导教师签字： 卢洋 |