**山东大学 软件 学院**

**Linux应用 课程实验报告**

|  |  |  |  |
| --- | --- | --- | --- |
| 学号： **202100202072** | 姓名：**张志恒** | | 班级：**地空21.1/自动22.4 跨专业选修** |
| 实验题目：**（实验四）Shell的使用** | | | |
| 实验学时：4 | | 实验日期： **2022.12.7** | |
| 实验目的：  1. 掌握shell通配符的含义和使用  包括：\* ？[…] [!...]  2. 掌握重定向的使用  2. 掌握管道的使用  3. 掌握输入和输出  4. 掌握tee的使用  5. 掌握正则表达  6. 熟悉搜寻和排序  7. 掌握Shell程序的三种运行方式 | | | |
| 硬件环境：  MacBook Pro 13-inch M1, 2020  芯片 Apple M1  内存 16 GB | | | |
| 软件环境：  环境1：macOS Ventura 13.0.1  环境2：macOS Ventura 13.0.1，Parallel Desktop 18.1.0，ubuntu 22.04 ARM64 | | | |
| 实验过程报告：  **Part 1 << shell环境**   1. 命令和文件名补全功能   1. 用date命令查看系统当前时间，在输入da后，按tab键，让shell自动补齐命令的后半部分  2. 用mkdir命令创建新的目录newdir。首先输入第一个字母m，然后按tab键，由于以m开头的命令太多，shell会提示是否显示全部的可能命令，输入n  输入n后则只剩下之前输入的m  3. 再多输入一个字母k，按tab键，让shell列出以mk开头的所有命令的列表  4. 在列表中查找mkdir命令，看看还需要多输入几个字母才能确定mkdir这个命令，然后输入需要的字母，再按tab键，让shell补齐剩下的命令  3个字母**mkd**就能确定**mkdir**  5. 最后输入要创建的目录名newdir，按回车键执行命令  **$ mkdir newdir**  6. 用ls命令查看newdir目录下的内容，看newdir是否可以用文件补全功能来输入？  可以  二、命令别名功能  7. 输入alias命令，显示目前已经设置好的命令的别名  8. 设置别名ls为ls –l,以长格形式显示文件列表 **$ alias ls=”ls -l”**  9. 显示别名ls代表的命令，确认设置生效 **$ alias ls**  10. 使用别名ls显示当前目录中的文件列表 **$ ls**  11. 在使定义的别名不失效的情况下，使用系统的ls命令显示当前目录中的命令列表  **$ \ls**  12. 删除别名ls **$ unalias ls**  13. 显示别名ls，确认删除别名已经生效 **$ alias ls**  14. 最后再用命令ls 显示当前目录中的文件列表 **$ ls**  l5. 写出定义别名cde为：cd /etc的命令 **$ alias cde=”cd /etc”**  **Part 2 << 通配符的使用**  16. 以tux1帐号登入系统，用ls /bin/\*sh 命令显示系统中的shell种类？系统中共有哪些shell？该命令的作用是什么？  有7种shell，该命令的作用是列出在指定目录下以sh结尾的文件  17. 进入/etc目录并且列出该目录下的所有文件  **$ cd /etc; ls**  18. 使用带通配符的ls命令列出文件名  注意：通配符的匹配是由shell来做的。如果匹配的文件名是一个目录名，ls命令默认将列出那个目录的内容，而不是文件名本身。可以使用-d参数来阻止该情况的发生。  (1) 文件名以conf结尾的文件  **$ ls -d \*conf**  (2) 文件名以一个字母d或D开头的文件  **$ ls -d [dD]\***  (3) 文件名的第五位包含一个字母o的文件  **$ ls -d ????o\***  (4) 文件名包含文字tab且不区分大小写字母的文件  **$ ls -d \*[tT][aA][bB]\***  (5) 文件名以一位数字结尾的文件  **$ ls -d \*[0-9]**  (6) 文件名不以一位数字结尾的文件  **$ ls -d \*[!0-9]**  19. 执行命令ls -d ?[!y]\*[e-f]将会发生什么情况？匹配的最短的文件名会是什么？  **Part 3 << 重定向**  21.  (1) 使用cat命令重定向创建一个名为junk的文件，该文件包含了一些文本行。键入一些行以后，中止cat命令返回到shell提示符，随后查看你刚才创建的文件的内容  (2) 使用重定向在junk文件中再添加一些行，然后查看该文件的内容  22. 对于命令 $ cat junk junk2  (1) 将其标准输出重定向到文件cat.out，标准错误重定向到cat.err  **$ cat > cat.out 2> cat.err junk junk2**  (2) 如果显示输出信息，但不保留该命令的错误信息呢？  **$ cat 2> /dev/null junk junk2**  **Part 4 << 管道**  23. 使用管道统计你的当前目录里的文件数，不要手动去计数  **$ ls | wc -w**  24. 执行ls -l> tempfile ; wc -l tempfile ; rm tempfile 命令与第前一题作用一样吗？为什么一样或不一样？  不一样，因为是先创建tempfile后ls再输出的，统计行数时多算了total和tempfile两行  25. 使用ls命令并把输出结果保存在名为tempfile2的文件中，然后统计并显示文件数  **$ ls > tempfile2**  **$ temp=`cat tempfile2 | wc -l` ; temp=$((temp - 1)) ; echo $temp**  **Part 5 << 命令组**  26. 在同一个命令行上，显示当前的系统日期和登入的所有用户，包括一些说明注释，结果存入到一个文件中  **$ date > temp ; who >> temp**  **Part 6 << 进程环境**  27. 显示在你的当前进程环境中定义的所有变量。显示所有环境变量。  **$ set**  **$ env**  28. 用命令 $localvar=10 给本地变量localvar赋值。显示该变量的值，然后再显示当前所有的变量。  **$ localvar=10**  **$ echo $localvar**  **$ set**  29. 用 $bash 命令启动一个子shell。再显示变量localvar的值。这次和28中显示的为什么不同？  **$ bash**  **$ echo $localvar**  没有值，因为子进程没有继承父进程的本地变量  30. 用什么命令可以将localvar的赋值带到子shell中去？  **$ export localvar**  **Part 7 << 使用grep和正则表达式**  33. 复制/etc/passwd文件到你的主目录，并换名为lspasswd  **$ cp /etc/passwd /home/akisui/lspasswd**  34.  (1) lspasswd文件中所有首字母是s的行  **$ grep "^s.\*" lspasswd**  (2) 用wc, grep和管道组合成一条命令统计lspasswd文件中首字母是s的行数  **$ grep "^s.\*" lspasswd | wc -l**  35. 重复上面的操作，但这次只显示符合条件的行的计数  **$ grep -c "^s.\*" lspasswd**  **Part 8 << Shell程序及其三种运行方式**  37.使用vi编辑器,编辑一个名为setdir的脚本文件:  dir1=/tmp  dir2=/usr  echo $dir1 $dir2  38.分别用学过的三种不同的运行方式运行该脚本,每次运行一种方式后执行echo $dir1 $dir2  39.这三种运行方式有什么不同？分别得到什么样的运行结果？为什么？  第二种. 是调用当前Shell，运行结束后在当前Shell可以访问本地变量  其余两种都是在子Shell中运行，结束后退出，变量不再可访问 | | | |
| 结论分析与体会：  本次实验采用macOS自带的终端、PD虚拟机下的ubuntu来完成。  本次实验使我熟悉了Linux/Unix系统的Shell的使用方式，初步了解了Shell脚本的编程，让我对Shell编程更加感兴趣，同时也认识到学好Linux的Shell编程可以用脚本代替很多繁琐的操作。 | | | |