**中国矿业大学计算机学院**

**2019级本科生课程作业**

|  |  |
| --- | --- |
| 课程名称 | Linux操作系统 |
| 作业次数 | 作业二 |
| 作业时间 | 2022年4月7日 |
| 姓名 | 胡钧耀 |
| 学号 | 06192081 |
| 专业 | 计算机科学与技术 |
| 任课教师 | 姜秀柱 |

目 录

[一、 作业 1](#_Toc100265438)

[1.1 目录 1](#_Toc100265439)

[1.1.1 根目录 1](#_Toc100265440)

[1.1.2 根目录的一级目录 1](#_Toc100265441)

[1.1.3 ls命令 3](#_Toc100265442)

[1.1.4 mkdir命令 5](#_Toc100265443)

[1.1.5 cd命令 5](#_Toc100265444)

[1.1.6 rmdir命令 5](#_Toc100265445)

[1.1.7 cp命令 6](#_Toc100265446)

[1.1.8 mv命令 6](#_Toc100265447)

[1.1.9 rm命令 7](#_Toc100265448)

[1.1.10 pwd命令 7](#_Toc100265449)

[1.2 帮助 8](#_Toc100265450)

[1.2.1 man命令 8](#_Toc100265451)

[1.2.2 --help命令 9](#_Toc100265452)

[1.2.3 help命令 9](#_Toc100265453)

[1.2.4 info命令 10](#_Toc100265454)

[1.2.5 whatis命令 11](#_Toc100265455)

[1.2.6 apropos命令 12](#_Toc100265456)

[1.3 文件 14](#_Toc100265457)

[1.3.1 文件类型 14](#_Toc100265458)

[1.3.2 文件属性 14](#_Toc100265459)

[1.3.3 chgrp命令 15](#_Toc100265460)

[1.3.4 chown命令 15](#_Toc100265461)

[1.3.5 chmod命令 16](#_Toc100265462)

[1.4 磁盘 17](#_Toc100265463)

[1.4.1 CHS、LBA及其特点 17](#_Toc100265464)

[1.4.2 MBR、GPT及其特点 17](#_Toc100265465)

[1.4.3 df命令 18](#_Toc100265466)

[1.4.4 du命令 19](#_Toc100265467)

[1.4.5 fdisk命令 19](#_Toc100265468)

[1.5 文件系统 21](#_Toc100265469)

[1.5.1 ext4文件系统 21](#_Toc100265470)

[1.5.2 mkfs命令 21](#_Toc100265471)

[1.5.3 fsck命令 21](#_Toc100265472)

[1.5.4 dumpe2fs命令 22](#_Toc100265473)

[1.6 挂载 23](#_Toc100265474)

[1.6.1 定义与作用 23](#_Toc100265475)

[1.6.2 mount命令 23](#_Toc100265476)

[1.6.3 umount命令 24](#_Toc100265477)

[1.6.4 lsblk命令 25](#_Toc100265478)

[1.6.5 findmnt命令 25](#_Toc100265479)

[1.7 用户、组与权限 26](#_Toc100265480)

[1.7.1 用户及其属性信息 26](#_Toc100265481)

[1.7.2 用户密码及其相关属性 26](#_Toc100265482)

[1.7.3 组及其属性信息 27](#_Toc100265483)

[1.7.4 组密码及其相关属性 27](#_Toc100265484)

[1.8 时间 28](#_Toc100265485)

[1.8.1 硬件时间和系统时间 28](#_Toc100265486)

[1.8.2 date命令 28](#_Toc100265487)

[1.8.3 tzselect命令 28](#_Toc100265488)

[1.8.4 hwclock命令 29](#_Toc100265489)

[1.8.5 cal命令 30](#_Toc100265490)

[二、 感悟 30](#_Toc100265491)

[参考文献 30](#_Toc100265492)

# 作业

## 目录

题目：给出Ubuntu安装后根目录下的一级目录的名称及存放的信息类别。常用目录命令有哪些（举例给出）？（10分）

### 根目录

Linux管理文件主要是通过目录管理，而不是C盘、D盘等盘符进行管理，所以根目录十分重要，它是整个系统最重要的一个目录，一是因为所有的目录都是在根目录之中衍生创建出来的，每一个文件和目录都从这里开始，二是根目录也与系统的开机/还原/系统修复等动作有关系。只有root用户具有该目录下的写权限。

查看根目录下的一级目录的方法如下，代码如下所示，使用ls命令显示当前目录下的文件，/表示根目录。

ls /

输入命令后，结果如下图所示。主要包括*/bin*，*/boot*，*/dev*，*/etc*，*/home*，*/root*，*/run*，*/sbin*，*/tmp*，*/usr*，*/var*等。

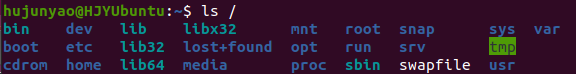


图 1.1‑1 查看根目录下的一级目录

### 根目录的一级目录

***/bin*目录**：即二进制（Binaries）的缩写。这里包含很重要的二进制可执行文件，存放着最经常使用的命令，例如cat，chmod，mkdir等。在*/bin*底下的指令可以被root与一般账号所使用。下图中的cat就是一个显示文件内容的命令，后面将详细介绍。

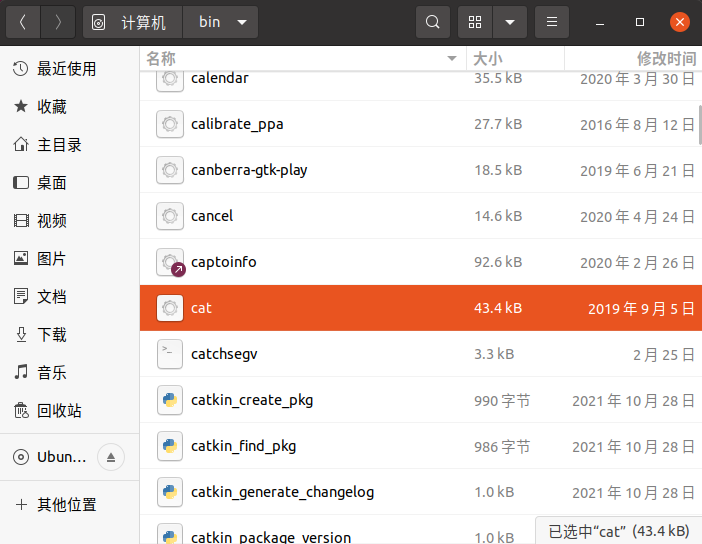


图 1.1‑2 在*/bin*文件夹下存放的命令

***/boot*目录**：存放引导加载程序的相关文件，也就是开机会使用到的文件，包括Linux核心文件等等。

***/dev*目录**：即设备（Device）的缩写。Linux系统中，任何装置与接口设备都是通过文件的形态存在于这个目录当中，这些包括终端设备、USB或连接到系统的任何设备。

***/etc*目录**：包含了所有程序所需的配置文件，系统主要的配置文件几乎都放置在此。一般来说，这个目录下的各文件一般使用者具有阅读权限的，但是只有root有修改权限。

***/home*目录**：系统默认的用户家目录。所有用户用home目录来存储他们的个人档案。如当前情况下，只有hujunyao一个用户。

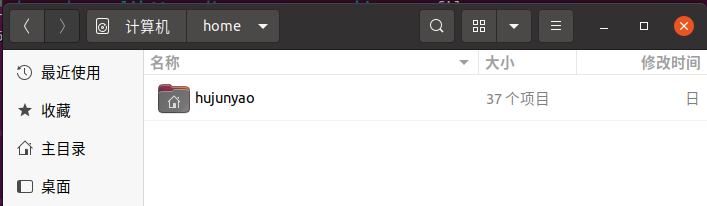


图 1.1‑3在*/home*文件夹的不同用户

***/lib*目录**：包含支持位于*/bin*和*/sbin*下的二进制文件的库文件，库文件名为*ld\**或*lib\*.so.\**。尤其重要的是*/lib/modules/*这个目录，该目录会放置核心相关的驱动程序。

***/media*目录**：即是媒体（Media）的英文，放置的就是可移除的装置，包括软盘、光盘、DVD等等装置都暂时挂载于此，由于自己使用的虚拟机，虽然电脑是没有光盘驱动的，但是VMware是使用的ISO映像文件，所以可以在该目录下查看到当前ubuntu操作系统的*.iso*文件。

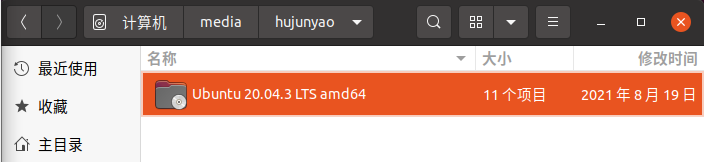


图 1.1‑4 在*/media*文件夹下的光盘映像文件

***/mnt*目录**：挂载（Mount）目录，是临时安装目录，系统管理员可以挂载文件系统。

***/opt*目录**：可选（Optional）目录，包含从个别厂商的附加应用程序。这些第三方的软件的应用程序应该安装在*/opt*或者*/opt*的子目录下。例如以前学习过ROS，因此能在该文件夹下查看到noetic版本的ROS文件夹。

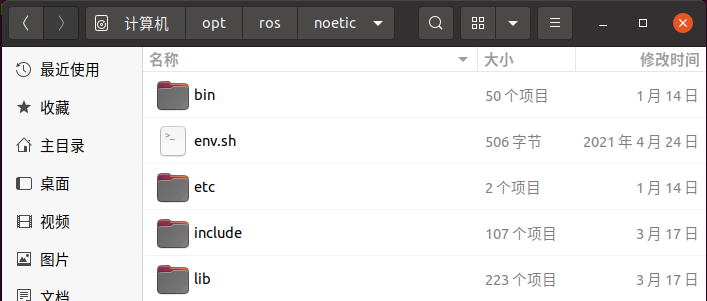


图 1.1‑5 在*/opt*文件夹下的ROS文件

***/root*目录**：系统管理员root的家目录。cd是Linux内建的命令，无法直接使用sudo cd方法进入该文件夹，需要使用sudo -i从用户hujunyao变成root才可以进入。

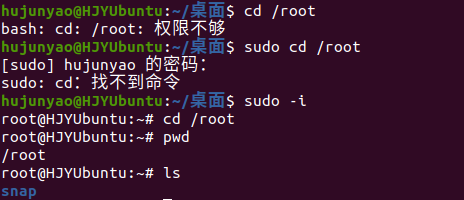


图 1.1‑6 无法直接进入*/root*文件夹

***/run*目录**：是一个临时文件系统，存储系统启动以来的信息。当系统重启时，这个目录下的文件应该被删掉或清除。

***/sbin*目录**：即是超级用户的二进制文件（Superuser Binaries）的缩写，这里存放的是系统管理员使用的系统管理程序。放在*/sbin*下的为开机过程中所需要的一些指令，里面包括了开机、修复、还原系统所需要的指令。常见指令包括fdisk，ifconfig等。

***/tmp*目录**：即是临时（Temporary）的缩写，用来存放一些临时文件。

***/usr*目录**：即是共享资源（Unix Shared Resources）的缩写，一定分清楚这个不是用户User的意思，但是用户的很多应用程序和文件都放在这个目录下，类似于Windows系统下的Program Files目录。

***/var*目录**：即是变量（Variable）的缩写，这个目录中存放着在不断扩充着的东西，经常被修改的目录将被放在这个目录下，这包括各种日志文件。

***/lost+found*目录**：这个目录是使用标准的ext2/ext3文件系统格式才会产生的一个目录，目的在于当文件系统发生错误时， 将一些遗失的片段放置到这个目录下。

***/proc*目录**：进程（Process）目录，是一个虚拟文件系统，放置的数据都是在内存当中，包含有关正在运行的进程的信息。因为这个目录下的数据都是在内存当中，所以本身不占任何硬盘空间。

***/sys*目录**：系统（System）目录，也是一个虚拟的文件系统，主要也是记录与核心相关的信息。 包括目前已加载的核心模块与核心侦测到的硬件装置信息等等。这个目录同样不占硬盘容量。

***/snap*目录**：snap是一种全新的软件包管理方式，它类似一个容器拥有一个应用程序所有的文件和库，各个应用程序之间完全独立，snap软件包一般安装在*/snap*目录下。

***/srv*目录**：服务（Service）目录，包含服务器特定服务相关的数据，存放服务启动后需要提取的数据，不用服务器就是空。

### ls命令

含义为list，作用是显示目录文件。

ls [-aAdhl1] 目录名称

选项与参数：

-a ：全部的文件，连同隐藏档( 开头为 . 的文件) 一起列出来

-A ：全部的文件，连同隐藏档，但不包括 . 与 .. 这两个目录

-d ：仅列出目录本身，而不是列出目录内的文件数据

-h ：将文件容量以人类较易读的方式(例如 GB, KB 等等)列出来

-l ：长数据串列出，包含文件的属性与权限等等数据

下图实现了显示当前目录下非隐藏文件和目录。

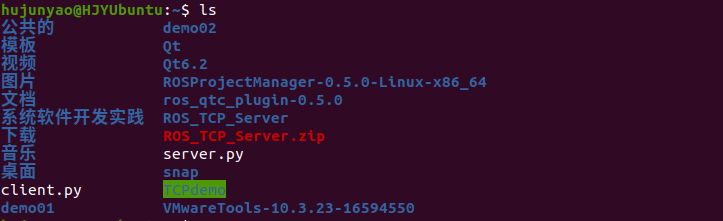


图 1.1‑7

下图实现了显示当前目录隐藏文件在内的所有文件列表（-a）。

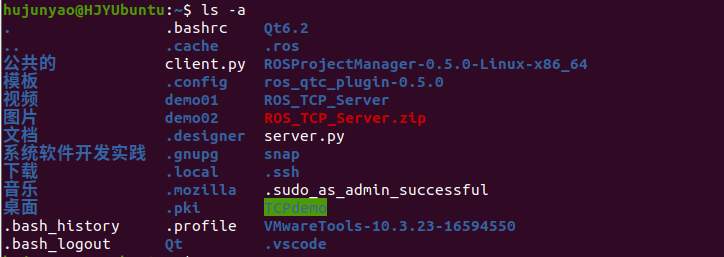


图 1.1‑8

下图实现了显示当前目录除了“.”和“..”的隐藏文件在内的所有文件列表（-A）。

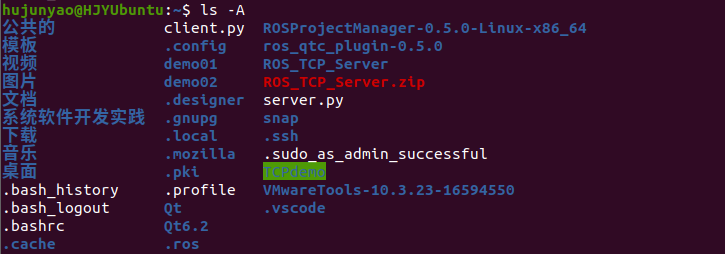


图 1.1‑9

下图实现了显示当前目录本身的详细信息（-dl）。



图 1.1‑10

下图实现了列出人性化显示的文件和文件夹的详细信息（-lh）。

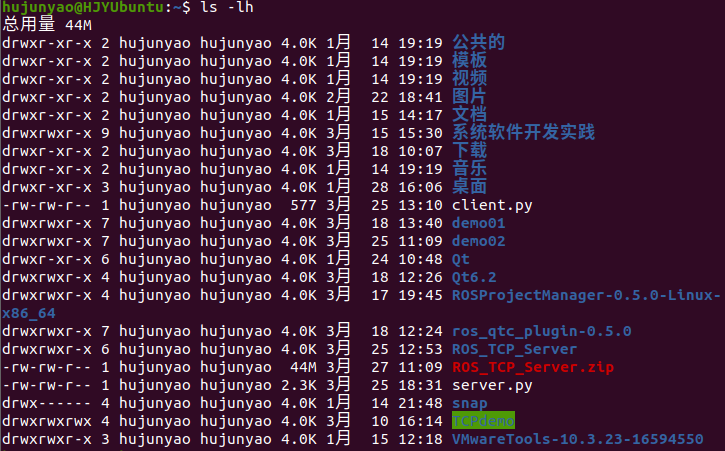


图 1.1‑11

### mkdir命令

含义为make directories，作用是创建新目录。

mkdir [-p] 目录名称

选项与参数：

-p ：递归创建

下图实现了在当前文件夹创建新文件夹和递归创建的文件夹（-p）。



图 1.1‑12

### cd命令

含义为change directory，作用是切换目录。

cd 目录名称

选项与参数：

~ ：home目录

.. ：目前目录位置的上一层目录

下图实现了进入home目录和某一文件夹。

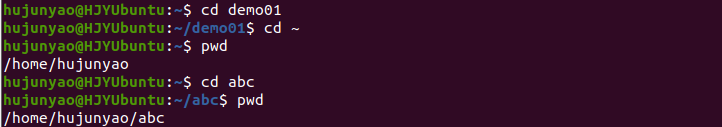


图 1.1‑13

### rmdir命令

含义为remove directories，作用是删除空目录（如果目录下存在文件则不能删除）。

rmdir [-p] 目录名称

选项与参数：

-p ：是当子目录被删除后使它也成为空目录的话，则顺便一并删除。

下图实现了删除100（含有101和102两个文件夹）和200（含有201一个文件夹）两个文件夹，100文件夹不能使用递归删除101，提示100非空（还有102文件夹）。

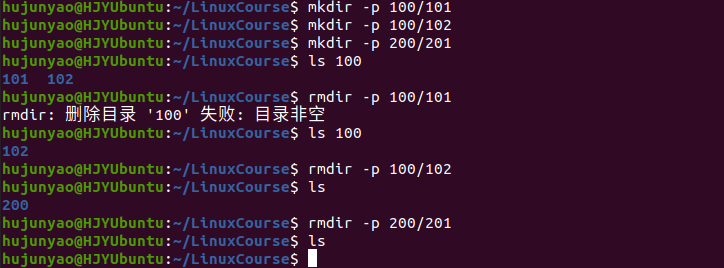


图 1.1‑14

### cp命令

含义为copy，作用是复制文件或目录。

cp [-adfIprl] 源文件… 目标目录

选项与参数：

-a ：此选项通常在复制目录时使用，它保留链接、文件属性，并复制目录下的所有内容。其作用等于dpr参数组合。

-d ：复制时保留链接。这里所说的链接相当于Windows系统中的快捷方式。

-f ：覆盖已经存在的目标文件而不给出提示。

-i ：与 -f 选项相反，在覆盖目标文件之前给出提示，要求用户确认是否覆盖，回答 y 时目标文件将被覆盖。

-p ：除复制文件的内容外，还把修改时间和访问权限也复制到新文件中。

-r ：若给出的源文件是一个目录文件，此时将复制该目录下所有的子目录和文件。

-l ：不复制文件，只是生成链接文件。

下图实现了将100文件夹中的*test.txt*复制到200文件夹中。

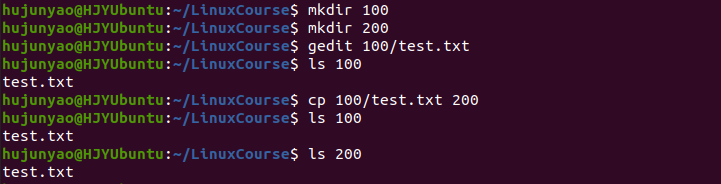


图 1.1‑15

下图实现了将101文件夹复制到201文件夹中，并且把*200/201/demo1.txt*覆盖为*100/101/demo1.txt*。

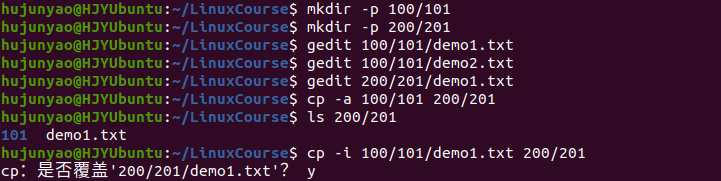


图 1.1‑16

### mv命令

含义为move，作用是剪切文件、改名。

mv [options] source dest

mv [options] source... directory

选项与参数：

-b ：当目标文件或目录存在时，在执行覆盖前，会为其创建一个备份。

-i ：如果指定移动的源目录或文件与目标的目录或文件同名，则会先询问是否覆盖旧文件，输入 y 表示直接覆盖，输入 n 表示取消该操作。

-f ：如果指定移动的源目录或文件与目标的目录或文件同名，不会询问，直接覆盖旧文件。

-n ：不要覆盖任何已存在的文件或目录。

-u ：当源文件比目标文件新或者目标文件不存在时，才执行移动操作。

下图实现了创建*100/demo.txt*和*200/demo1.txt*，把100文件夹移动到200文件夹内，把200文件夹重命名为300，再把*300/100/demo.txt*重命名为*300/100/demo2.txt*，然后把*300/100/demo2.txt*移动到300文件夹内。

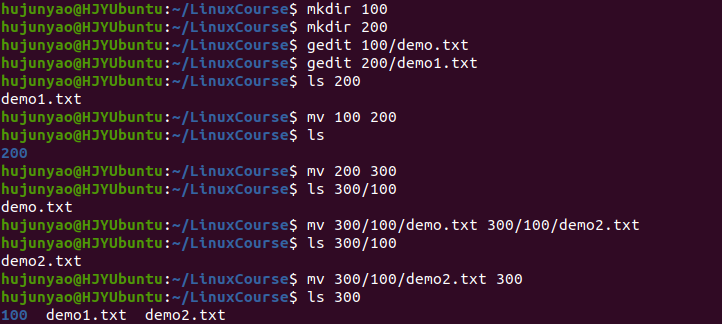
****

图 1.1‑17

### rm命令

含义为remove，作用是删除文件或目录。

rm [options] name...

选项与参数：

-i ：删除前逐一询问确认。

-f ：即使原档案属性设为唯读，亦直接删除，无需逐一确认。

-r ：将目录及以下之档案亦逐一删除。

下图实现了在100文件夹中创建3个txt文件，然后删除其中一个，最后删除该100文件夹。



图 1.1‑18

### pwd命令

**pwd命令**：含义为print working directory，作用是显示当前目录。如下图所示。

pwd

选项与参数：

-L, --logical ：目录连接链接时，输出连接路径，不考虑软、硬连接路径

-P, --physical ：查看软、硬链接最终指向文件的全部路径



图 1.1‑19

## 帮助

题目：Linux的命令帮助方式有哪些？各有什么特点？常用帮助命令有哪些（举例给出）？（15分）

主要有man命令、--help命令、help命令、info命令、whatis命令和apropos命令，下面分开论述这几个命令的特点和使用。

### man命令

man特点是适用于一般命令，非内置命令。它是包含了Linux中全部命令的手册（Manual），手册非常详细，这个手册分为八个大章节，使用man时也可以指定不同的章节来阅读。八个章节是：标准命令（Standard commands）、系统调用（System call）、库函数（Library function）、设备文件（Special devices）、文件格式（File formats）、游戏和娱乐（Games and toys）、杂项、惯例与协定（Miscellaneous）、管理员命令（Adminstrative commands）。

man [man options] [[section] page ...] ...

man -k [apropos options] regexp ...

man -f [whatis 选项] 页 ...

选项与参数：

-a, --all：

默认情况下，man在显示它找到的最符合的手册页后会退出。此选项强制 man 显示名字匹配搜索条件的所有的手册页。

-f, --whatis：

等同于whatis。显示来自手册页的简短说明，每个手册页中都有一个概述。whatis 搜索手册页名称并显示任何与名称匹配的手册页描述。

-k, --apropos：

等同于apropos。搜索关键词对应的手册概述并显示所有匹配结果。详见apropos。

下图实现了查询man命令的所有的手册页，查找到在第一章节和第七章节都有说明，但是该命令最常用的是第一章节，所以不加-a参数时返回的结果等同于man.7。

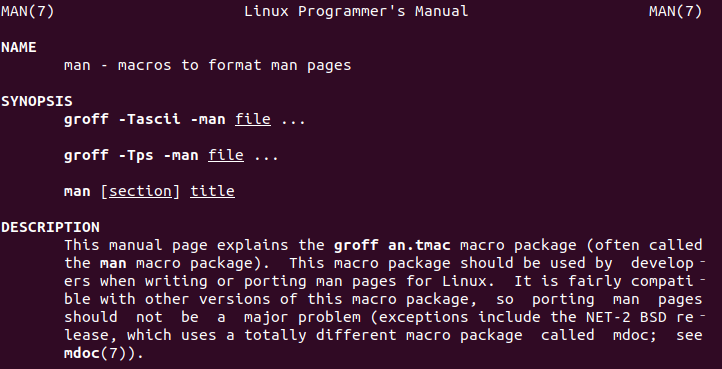
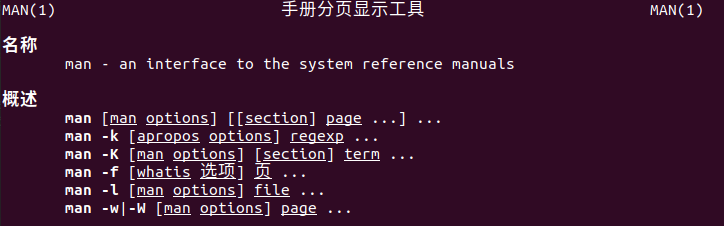
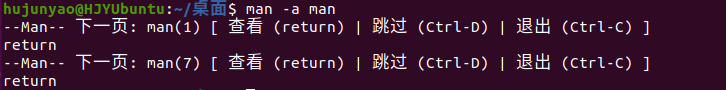


图 1.2‑1

下图实现了查询adjtime命令来自手册页的简短说明，可知有两个手册页被查询出来。每个手册页中都有一个概述。



图 1.2‑2

下图实现了查询和关键词time有关的手册概述并显示所有匹配结果，这些都是一些和时间配置相关的命令。

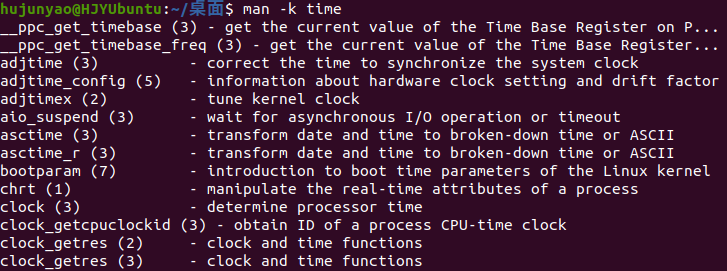


图 1.2‑3

### --help命令

获取一些最基础的帮助。在通常输入错误命令的时候就会提示【请尝试执行“xxx --help”】来获取更多信息。适用于一般命令，非内置命令。它自己本身就是个参数。

命令名 --help

下图实现了查询ls命令的基本的使用方法和参数的介绍。

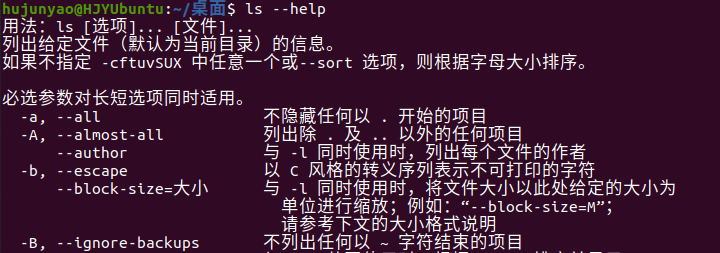


图 1.2‑4

### help命令

该命令用于获取bash内置命令（如cd、echo、help）帮助。

help [-dms] [模式 ...]

选项与参数：

-d ：输出每个主题的简短描述

-m ：以伪 man 手册的格式显示使用方法

-s ：为每一个匹配 PATTERN 模式的主题仅显示一个用法简介

PATTERN ：指定帮助主题的模式

对于某一些内部命令，使用该方法可以查询帮助，但是--help命令不可以，如echo命令，但是不是内建函数的肯定是不能查询到的，例如ls，如下图所示。

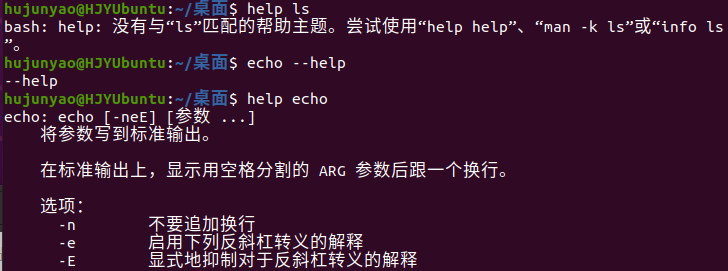


图 1.2‑5

下图实现了三种选项输出结果的比较。

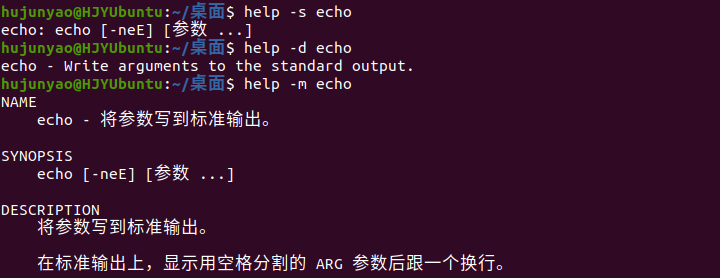


图 1.2‑6

### info命令

适用于一般命令，比man命令稍微友好一点。它不是像info page一口气输出一堆信息，info page则是将文件数据拆成一个一个的段落，每个段落用自己的页面来撰写，是分页显示，用百分比表示。

info [OPTION]... [MENU-ITEM...]

选项与参数：

-k, --apropos=STRING

在所有手册的所有索引中查找 STRING

--index-search=STRING

转到由索引项 STRING 指向的节点

-o, --output=FILE

输出被选择的节点内容到指定的文件

-w, --where, --location

查看命令的 info 格式的帮助文档地址。

下图实现了输入info进入顶级目录。

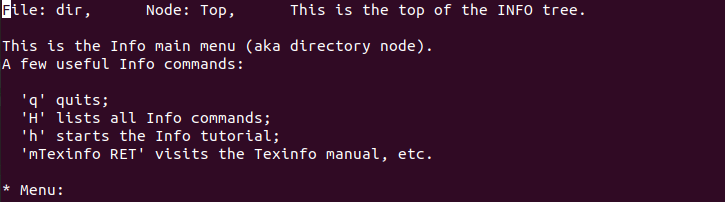


图 1.2‑7

下图实现了在手册的所有索引中查找与time有关的内容。

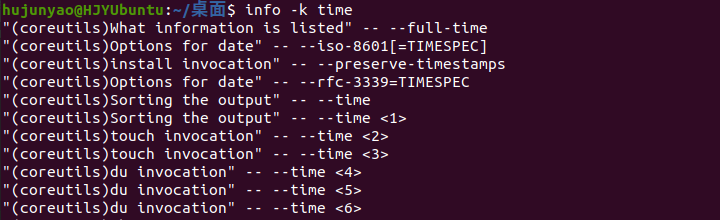


图 1.2‑8

下图实现了将echo命令的信息输出到*out.txt*文件中。



图 1.2‑9

下图实现了在info命令的文档中查找和index有关的节点，输入“，”可以尝试寻找下一个序列节点是index的节点。

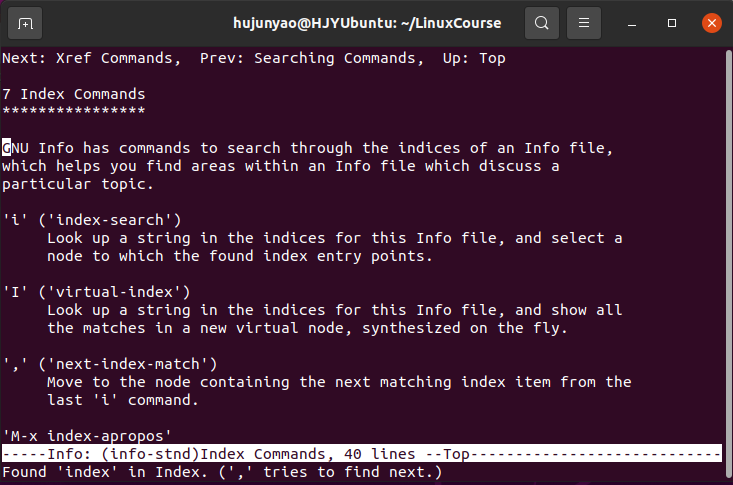


图 1.2‑10

### whatis命令

相当于man -f命令。将会查找一个存储所有命令相关信息的数据库，根据命令名返回相关结果。

whatis [-dlv?V] [-r|-w] [-s 列表] 名称 ...

选项与参数：

-r, --regex

将每个名称解释为正则表达式。如果名称匹配页面名称的任何部分，将视为一个匹配。该选项会使whatis略微变慢，这是由数据库搜索的原理导致的。

-w, --wildcard

将每个名称解释为包含shell风格通配符的模式。要匹配，展开的名称要与整个页面名称匹配。该选项会使whatis略微变慢，这是由数据库搜索的原理导致的。

-l, --long

不按终端宽度截断输出。

--usage

打印简短的帮助消息并退出。

下图实现了使用正则表达式查找以reset结尾的命令。

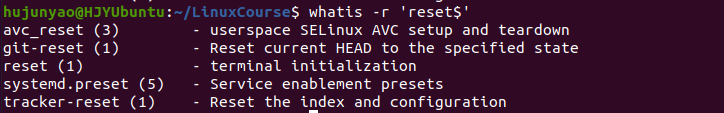


图 1.2‑11

下图实现了使用通配符查找含有time的命令。

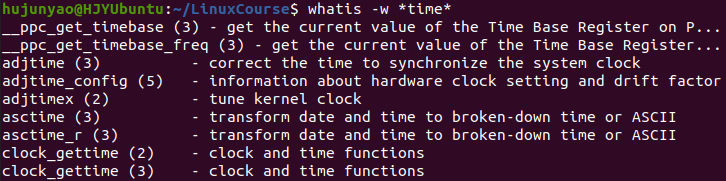


图 1.2‑12

下图实现了检索的vcstime命令的概述完整输出。

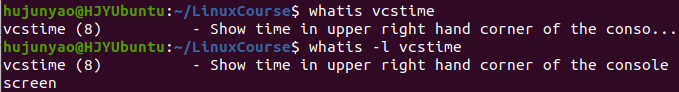


图 1.2‑13

下图实现了显示有关whatis命令的简短信息。它显示简短说明并退出。

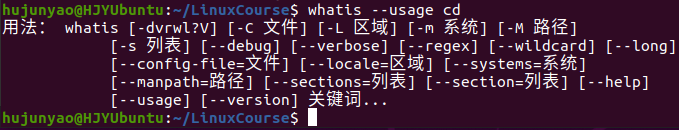


图 1.2‑14

### apropos命令

用来通过关键字查找定位手册页的名字和描述。它相当于使用带有-k选项的man命令。每个手册页里都有一个简短的描述，本命令会搜索包括关键词的描述。

apropos [-e|-w|-r] [-s 列表] 关键词 ...

选项与参数：

-r, --regex

将每个关键词解释为正则表达式。这是默认行为。每个关键词将与手册页名称和描述分别匹配。匹配不限于单词边界。

-w, --wildcard

将每个关键词解释为shell风格的通配符。关键词将与手册页名称和描述分别匹配。

-l, --long

不按终端宽度截断输出。

-s list, --sections=list, --section=list

只搜索指定的手册章节。列表是由冒号或逗号分隔的章节列表。如果列表中的指定的是简单的章节号，如“3”，则显示的描述列表将只包含“3”、“3perl”、“3x”等章节中的页面；如果列表中的章节号带扩展，如“3perl”，则显示的描述列表将只包含手册章节中对应部分的页面。

下图实现了使用正则表达式查找以reset结尾的命令。

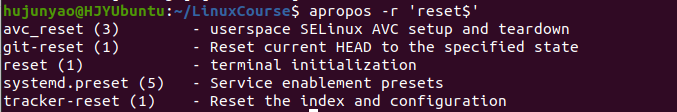


图 1.2‑15

下图实现了使用通配符查找含有time的命令。注意到这里bootparam命令不含time但是其简介中有time出现。

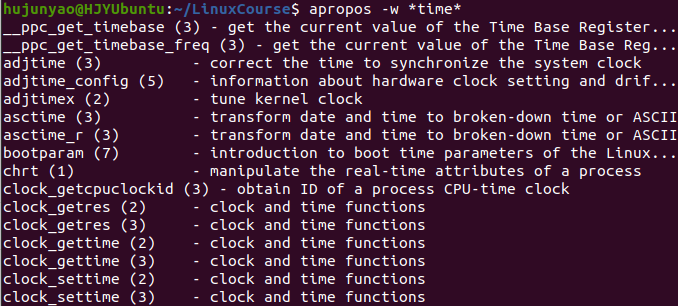


图 1.2‑16

下图实现了检索的vcstime命令的概述完整输出。

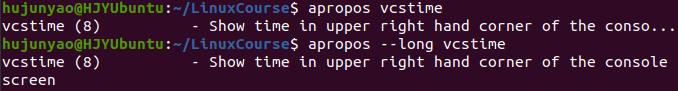


图 1.2‑17

下图实现了检索第二章和第五章中以time结尾的命令，并将概述完整输出。

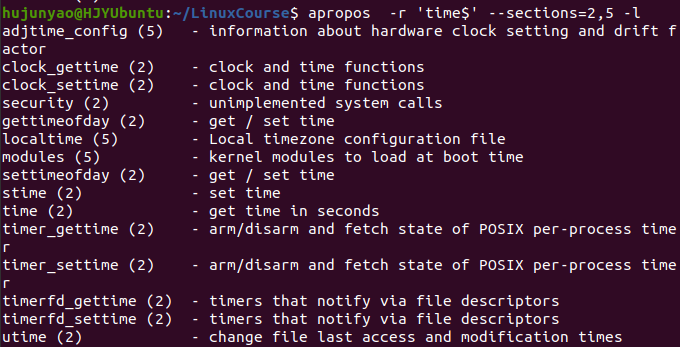


图 1.2‑18

## 文件

题目：Linux的文件类型有哪些？一个文件的属性有哪些？修改文件的属性的命令有哪些（举例给出）？（10分）

### 文件类型

Linux的文件类型主要有：普通文件、目录、特殊文件（符号链接文件、包括字符设备文件、套接口文件、命名管道文件、块设备文件）。

表 1.3‑1 Linux文件类型及其符号

|  |  |
| --- | --- |
| 符号 | 含义 |
| - | 普通文件 |
| d | 目录文件（Directory） |
| l | 链接文件（Link） |
| c | 字符设备文件（Character） |
| s | 套接字文件（Socket） |
| p | 命名管道文件（Pipeline） |
| b | 块文件（Block） |

使用前面提到的ls -lih命令可以具体查看某一文件夹内文件的类型和inode数据，文件大小同时人性化显示，如下图所示，在根目录下，查看最前面的十位字符串的第一位，可以了解到有普通文件（swapfile）、目录文件（boot等）和链接文件（bin等）。

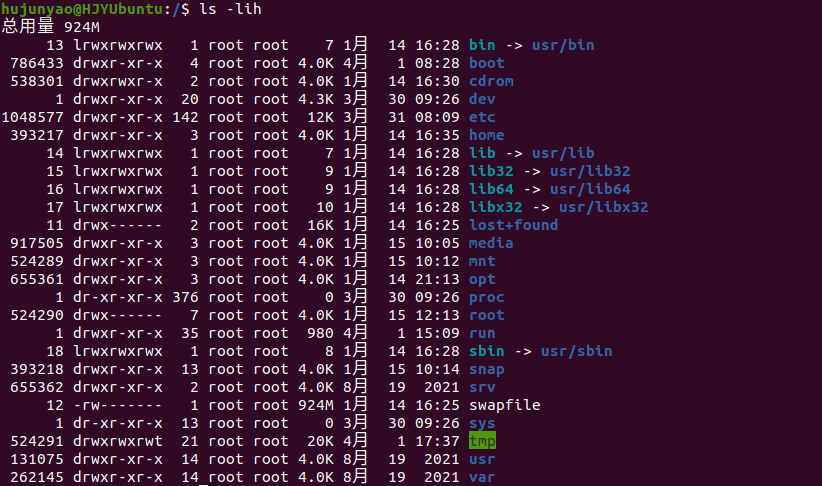


图 1.3‑1 根目录下文件的不同类型

### 文件属性

观察上图可以知道文件的基本属性有八个，从左往右依次有：inode索引节点、文件类型及权限、硬链接个数、文件所属用户、文件所属用户组、文件大小（目录通常是4KB大小）、文件修改时间、文件名。

**inode索引节点**：Linux系统的文件数据存储在块中，还需要一个地方存储文件的元信息（文件的创建者、创建日期等等），这种存储文件元信息的区域就叫做inode索引节点。因此，一个文件必须占用一个inode，但至少占用一个block。使用inode节点来产生的新的文件名，而不是产生新的文件，称为硬链接。再创建一个独立的文件，而这个文件会让数据的读取指向它连接的那个文件的文件名，inode不一样，但文件的内容是另一文件的路径（有些类似于重定向，或者说是快捷方式），称为软链接。

**文件类型及权限**：文件类型已经在上述小节介绍，文件的权限主要是体现在十位标识的后面九位，三个为一组，都是rwx三个参数进行组合，r表示可读（readable），w表示可写（writable），x表示可执行（executable），如果没有这一项权限，以-号代替该位，三组参数从左到右是属主权限、属组权限、其他用户权限，所以文件类型及权限一定是十位标识。

### chgrp命令

即改变群组（Change group），用于变更文件或目录的所属群组，它允许普通用户改变文件所属的组，只要该用户是该组的一员。

chgrp [-cfhRv][--help][--version][所属群组][文件或目录...]

选项与参数：

-c --changes

效果类似-v参数，但仅显示更改的部分。

-h 或 --no-dereference

只对符号连接的文件作修改，而不改动其他任何相关文件

-R --recursive

递归处理，将指定目录下的所有文件及子目录一并处理

-v --verbose

显示指令执行过程

下图实现了查看LinuxCourse文件夹下的*out.txt*文件的权限，并查看了可以修改为哪些属组（如adm、dip等），并将属组修改为cdrom，可以查看到属组成功修改。

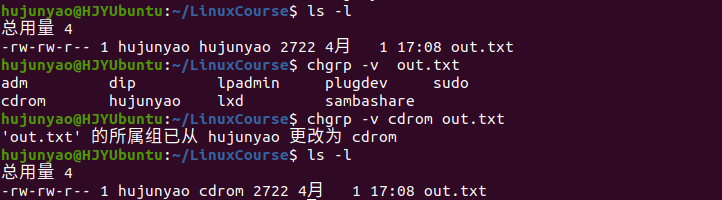


图 1.3‑2

### chown命令

即改变所属用户和组（Change owner），用于设置文件所有者和文件关联组的命令。该命令将指定文件的拥有者改为指定的用户或组，用户可以是用户名或者用户ID，组可以是组名或者组ID，文件是以空格分开的要改变权限的文件列表，支持通配符。该命令需要超级用户root的权限。

chown [-cfhvR] user[:group] file...

选项与参数：

user : 新的文件拥有者的使用者

group : 新的文件拥有者的使用者组

-c : 显示更改的部分的信息

-f : 忽略错误信息

-h : 修复符号链接

-v : 显示详细的处理信息

-R : 处理指定目录以及其子目录下的所有文件

下图实现了*/opt/ros*文件夹的权限从root到hujunyao进行互换，可以发现如果以hujunyao的权限尝试运行chown命令会提示不允许操作，必须切换到root权限才可以进行修改。

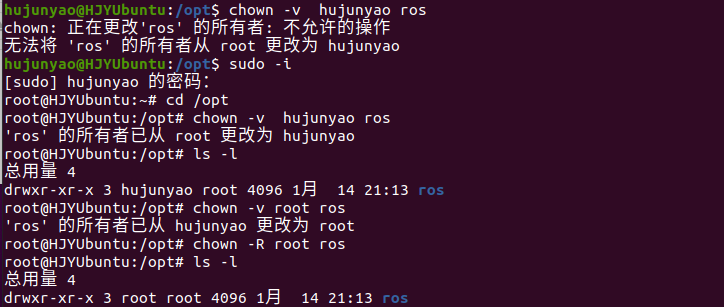


图 1.3‑3

### chmod命令

Linux文件属性有两种设置方法，一种是数字，一种是符号。Linux 文件的基本权限就有九个，分别是属主/属组/其他（owner/group/others）三种身份各有自己的可读、可写、可执行权限。r对应4，w对应2，x对应1。综上，-rwxrwx---就等价于770。还有一个改变权限的方法是直接用文字rwx进行描述等于、新增、减少权限。综上，-rwxrwx---就等价于chmod u=rwx,g=rwx,o=-文件名

chmod [-R] xyz 文件或目录

chmod u/g/o/a +/-/= r/w/x 文件或目录

选项与参数：

xyz :数字类型的权限属性，为rwx属性数值

-R : 进行递归(recursive)的持续变更，以及连同次目录下的所有文件都会变更

下图实现了对*out.txt*文件权限进行修改，首先设置为777（属主、属组、其他用户均具有读写执行权限），然后清空其他用户的所有权限，然后减少属组的执行权限，增加其他用户的读写权限。

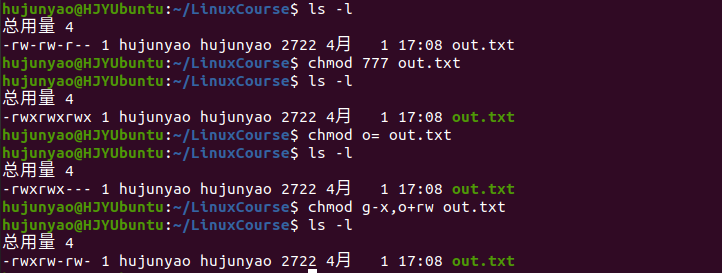


图 1.3‑4

## 磁盘

题目：什么是CHS、LBA？什么是MBR、GPT？它们各自特点是什么？Linux的常用磁盘管理命令有哪些（举例给出）？（15分）

### CHS、LBA及其特点

CHS和LBA都是指磁盘的寻址模式。CHS和LBA类似于真实地址和虚拟地址的区别，二者可以使用公式相互映射，相互转换，但是查找地址的效率略有不同。

**CHS**：是FDISK在分区期间所需的磁盘信息。这种寻址模式将硬盘划分为磁头（Heads）、柱面(Cylinder)、扇区(Sector)，模式的名字即磁头、柱面、扇区的首字母缩写。相当于三级坐标，类似我国的“省、县、镇（乡）”三级行政区划，用第几磁头就可以表示数据在哪个磁面，用第几柱面可以表示数据在哪个磁道，用第几扇区就可以表示数据在某一磁道中的位置，一般来说，每个扇区的容量为512B。知道磁头数、柱面数、扇区数，就可以确定数据在硬盘中的位置，也可以确定硬盘的容量，硬盘容量公式：硬盘容量=磁头数×柱面数×扇区数×512B。

**LBA**：是指逻辑块寻址模式（Logical Block Addressing），是描述计算机存储设备上数据所在区块的通用机制。在LBA中，地址不再表示实际硬盘的实际物理地址（CHS），它将CHS这种三维寻址方式映射为一维的线性寻址，从0开始编号来定位区块，第一区块LBA=0，第二区块LBA=1，依此类推，效率大大提高，避免了烦琐的磁头/柱面/扇区的寻址方式。如果想访问真实物理地址，在访问硬盘时由硬盘控制器将这辑地址转换为实际的物理地址，只是多了一步转换。逻辑扇区号LBA的公式：LBA=磁头数×每磁道扇区数×当前所在柱面号+每磁道扇区数×当前所在磁头号+当前所在扇区号–1。

### MBR、GPT及其特点

MBR和GPT都是指磁盘的分区技术。从技术改进和容量支持上看，相比于MBR，GPT的优势更为明显，也更合理一些。

**MBR**：是指主分区引导记录（Master Boot Record）。在LBA中，引导扇区是每个分区的第一扇区，而主引导扇区则是整个硬盘的第一扇区（主分区的第一个扇区）。MBR保存在主引导扇区中。在MBR中，分区表的大小是固定不变的，一共可容纳4个主分区信息。MBR分区表中支持的硬盘最大分区容量上限大约为2TB，对2TB以上容量的物理硬盘，不适合使用MBR分区方案。

**GPT**：即全局唯一标识磁盘分区表（Globally Unique Identifier Partition Table， GUID）的缩写，这是实体硬盘的分区表的结构布局的标准。在GTP磁盘的第一个数据块中同样有一个与MBR类似的标记，叫做PMBR。当不支持GPT的分区工具试图对硬盘进行操作时，它可以根据这份PMBR以传统方式启动，过程和MBR+BIOS完全一致，极大地提高了兼容性。而支持GPT的系统在检测PMBR后会直接跳到GPT表头读取分区表。GPT的分区方案相对MBR来说更为先进，主要是GPT分区表头中可自定义分区数量的最大值，GPT分区表的大小是可变的。相比MBR，GPT分区表则能够识别2TB以上的硬盘空间。除此之外，GPT分区方案在硬盘的末端还有一个备份分区表，保证了分区信息不容易丢失。

下图展示了使用MBR分区和GPT分区的主要分区结构，可以知道虽然GPT结构更加复杂，但是支持的磁盘容量是要大很多的。

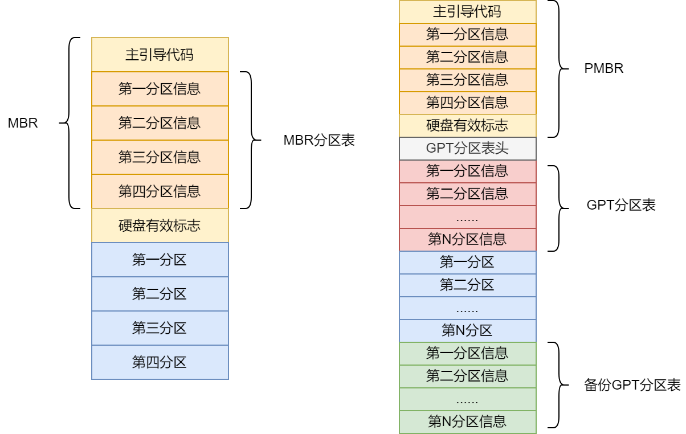


图 1.4‑1 MBR分区（左）与GPT分区（右）结构对比

### df命令

其作用是显示每个指定文件所在的文件系统的信息，默认是显示所有文件系统，也就是默认查看整个文件系统对于磁盘的使用情况（Disk full）。

df [选项]... [目录或文件]...

选项与参数：

-a ：列出所有的文件系统，包含虚拟、重复和无法访问的文件系统

-h ：以GB、MB、KB等格式显示大小

-T ：显示文件系统类型

-i ：不用硬盘容量，而以inode的数量来显示

下图实现了将系统内所有的文件系统列出，并显示文件系统类型。

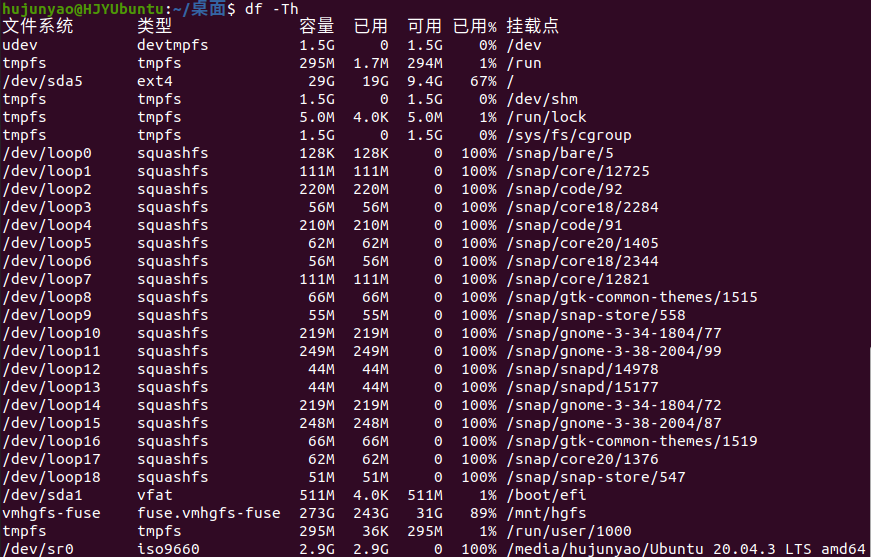


图 1.4‑2

下图实现了列出opt文件夹的文件系统的内存和节点使用情况。可以发现内存和节点的使用情况是不一样的，一般很少有这种情况，其实出现的根本原因是inode储存在一边，文件又存储在另一边，各自的内存大小的限制也不一样。

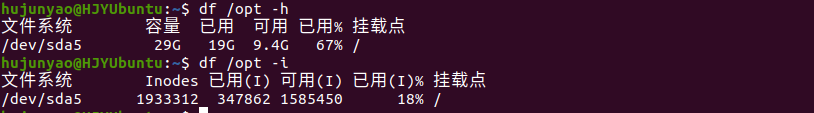


图 1.4‑3

### du命令

其作用是，查看文件和目录磁盘使用的空间，即计算每个文件的磁盘用量，如果是目录则取总用量（Disk used），其实也就是查看该文件夹下文件还有文件夹的大小。

du [选项]... [目录或文件]...

选项与参数：

-a ：输出所有文件的磁盘用量，不仅仅是目录

-c, --total：显示总计信息

-h ：以GB、MB、KB等格式显示大小

-s ：列出总量而已，而不列出每个各别的目录占用容量；

-S ：不包括子目录的占用量

-k ：以 KBytes 列出容量显示

下图实现了查询*~/ROS\_TCP\_Server*文件夹的各文件夹内存大小（含子文件夹，不含文件，统计内存大小时加上子目录的占用量）、各文件夹内存大小（含子文件夹，统计内存大小时不加上子目录的占用量）、当前目录下各文件夹内存大小（不含子文件夹，加上子目录的占用量）、各文件夹和文件内存大小（含子文件夹，含文件）、各文件夹内存大小（含当前查询的文件夹的大小，即总用量）。

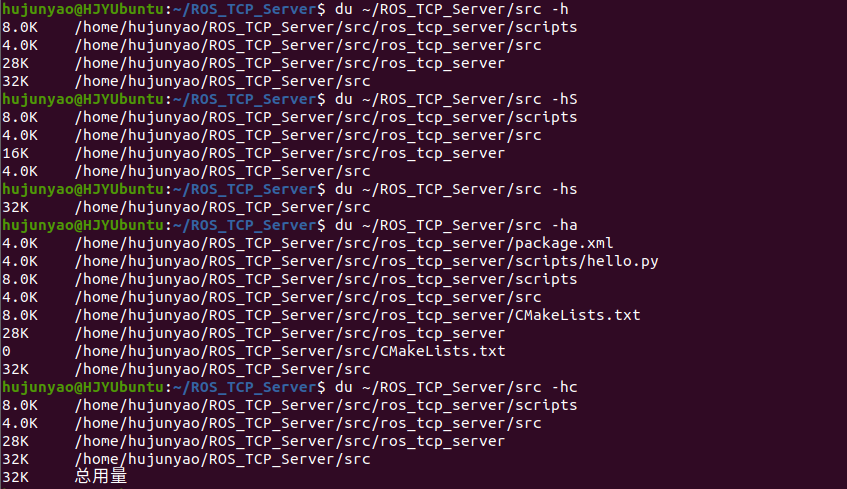


图 1.4‑4

### fdisk命令

用于显示或操作磁盘分区表。

fdisk [选项] <磁盘> 更改分区表

fdisk [选项] -l [<磁盘>] 列出分区表

下图实现了列出磁盘分区表的信息。

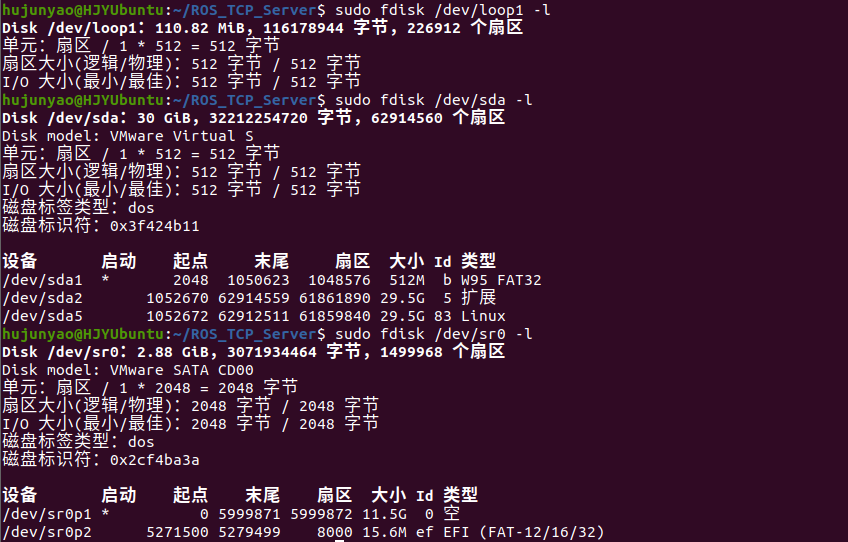


图 1.4‑5

下图实现了进入fdisk的交互界面，可根据提示完成获取磁盘信息、新增/删除一个分区等对磁盘的操作。对磁盘的操作首先位于内存中，只有通过w退出时才会将修改真正执行，相应的使用q退出则不会做修改。需要用户确定操作无误后再执行。



图 1.4‑6

下图实现了显示*/dev/sda*的分区表，一共有三个分区，其中*/dev/sda1*是启动分区。



图 1.4‑7

下图实现了删除*/dev/sda5*分区，再查看硬盘分区情况，发现已经删除，然后新建一个分区，从1054718开始，大小为2000000，也就是末尾到3054718，大小为976.6M，类型为Linux，分区为*/dev/sda5*。以上步骤在内存中进行，然后不保存退出。

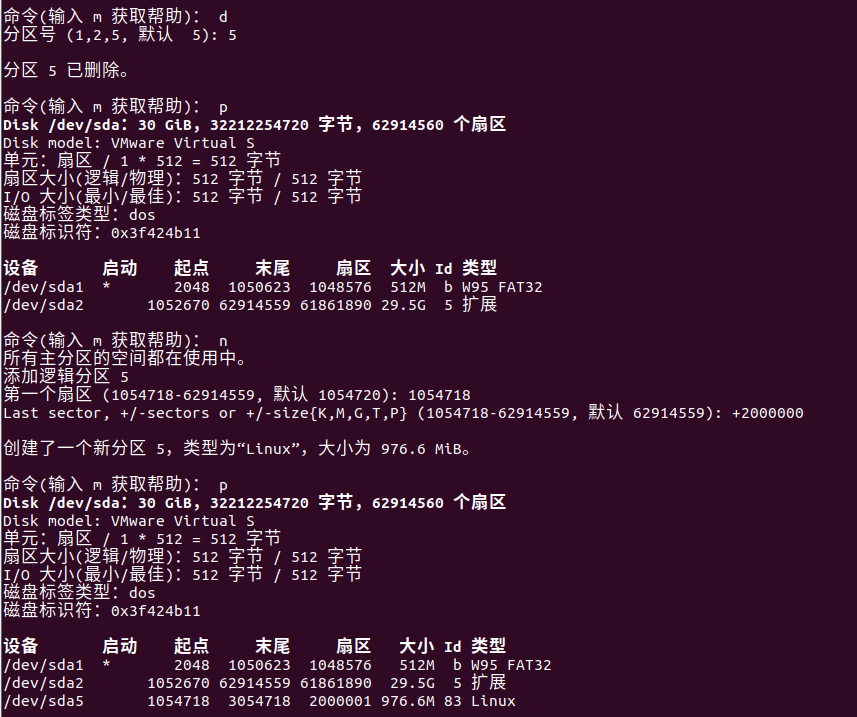


图 1.4‑8

## 文件系统

题目：什么是ext4文件系统？Linux的常用文件系统管理命令有哪些（举例给出）？（15分）

### ext4文件系统

文件系统是操作系统用来合理组织存储不同分区的文件的方法，使用一种数据结构来对文件进行管理，而ext4文件系统指的是Linux第四代扩展文件系统（Fourth extended filesystem），这是一种Linux系统下的日志文件系统，也是ext3文件系统的后继版本，目前的大部分Linux文件系统也都默认采用了ext4文件系统。

### mkfs命令

格式化也就是做一个新的文件系统（Make file system），其作用是选择一个分区后，在其上面新的建立Linux文件系统。

mkfs [选项] [-t <类型>] [文件系统选项] <设备> [<大小>]

选项：

-t, --type=<类型> ：文件系统类型；若不指定，将使用 ext2

下图实现了查看命令的版本和该命令支持的文件格式。

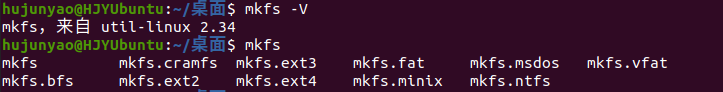


图 1.5‑1

下图实现了即将格式化*/dev/sda*分区，并指定为ext4类型。

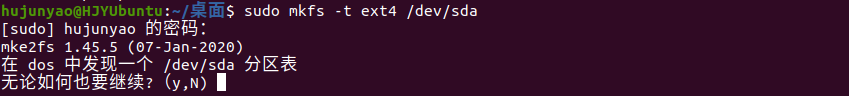


图 1.5‑2

### fsck命令

fsck用于检测并修复Linux文件系统中的错误，一般会在系统启动时运行。

fsck [选项] -- [文件系统选项] [<文件系统> ...]

参数与选项：

-A ：检查所有文件系统

-s ：串行检查序列。如果有多个文件系统需要检查，那么按照顺序来操作。

-t <类型> ：指定要检查的文件系统类型

-C [<fd>] ：显示进度条；文件描述符 fd 用于图形用户界面

-M ：不检查已挂臷的文件系统

-P ：并行检查文件系统，包括 root

-R ：跳过根文件系统；只在指定 '-A' 时有用

-r [<fd>] ：报告对每个已检查设备的统计；文件描述符用于 GUI

-V ：解释正在进行的操作

下图实现了对所有文件系统进行检测，检测到有文件系统已经挂载，系统提示可能出现问题，于是继续排除已经挂载的文件系统，然后输出过程中产生的信息。

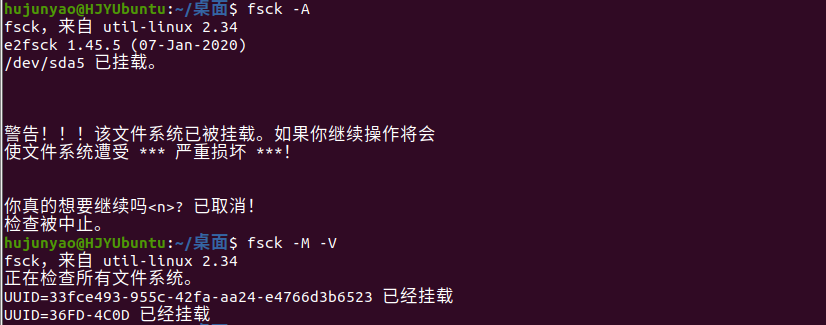


图 1.5‑3

### dumpe2fs命令

其作用是查看文件系统的详细信息，使用该命令会打印文件系统的超级块和块组信息，但是，当与挂载的文件系统一起使用时，打印的信息可能是旧的或不一致的。

dumpe2fs [选项] [设备名]

-b ：打印文件系统中预留的块信息

-h ：仅显示超级块信息

-i ：从指定的文件系统映像文件中读取文件系统信息

-x ：以16进制格式打印信息块成员

下图实现了对*/dev/sda5*分区实现显示预留的块信息、超级块信息。



图 1.5‑4

下图实现了对*/dev/sda5*分区以16进制打印信息块成员，因信息较多只截取了部分。



图 1.5‑5

## 挂载

题目：什么是Linux的挂载？挂载的作用是什么？常用的挂载命令有哪些（举例给出）？（10分）

### 定义与作用

Linux系统里面所有事物都是以文件的形式存在的，包括各种硬件设备也是如此，都有一个文件夹，这些所有文件都放置文件夹中，但是它们的文件系统是各是一套，文件目录结构不一样。如果Linux系统想使用这些硬件设备时，只有将Linux本身的文件目录与硬件设备的文件目录合二为一才能使用，分开是没有办法进行使用的。两套文件系统合并起来，访问这个目录就等同访问这个目录对应的设备文件，这个合二为一的过程就叫做挂载（Mount），而这个目录就叫做挂载点。如果不进行挂载，尽管通过Linux的GUI界面查看，找到硬件设备，但命令行的方式无法找到。

### mount命令

该命令用于挂载文件系统。

mount [-lhV]

mount -a [选项]

mount [选项] [--source] <源> | [--target] <目录>

mount [选项] <源> <目录>

mount <操作> <挂载点> [<目标>]

选项：

-a, --all 挂载fstab中的所有文件系统

-o, --options <列表> 挂载选项列表，以英文逗号分隔

-O, --test-opts <列表> 限制文件系统集合(和 -a 选项一起使用)

-r, --read-only 以只读方式挂载文件系统(同 -o ro)

-t, --types <列表> 限制文件系统类型集合

--source <源> 指明源(路径、标签、uuid)

--target <目标> 指明挂载点

-v, --verbose 打印当前进行的操作

-w, --rw, --read-write 以读写方式挂载文件系统(默认)

源：

-L, --label <标签> ：同 LABEL=<label>

-U, --uuid <uuid> ：同 UUID=<uuid>

LABEL=<标签> ：按文件系统标签指定设备

UUID=<uuid> ：按文件系统 UUID 指定设备

PARTLABEL=<标签> ：按分区标签指定设备

PARTUUID=<uuid> ：按分区 UUID 指定设备

<设备> 按设备路径路径指定设备

<目录> 绑定式挂载的挂载点(参阅 --bind/rbind)

<文件> 用于设置回环设备的常规文件

下图实现了直接输入命令，获取*/etc/mtab*文件夹内的已挂载设备信息。

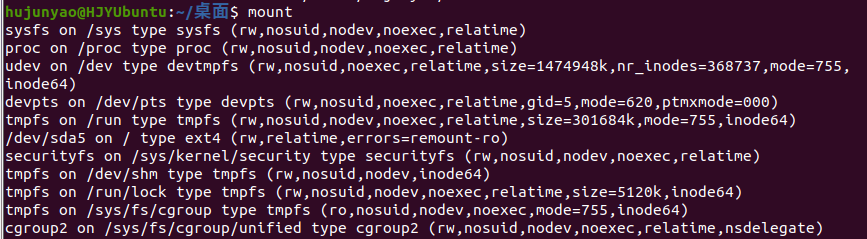


图 1.6‑1

下图实现了对于未挂载的新的磁盘设备*/dev/sda6*挂载到*/media/hujunyao/newdisk*文件夹下，同时使用了df命令对是否挂载上进行了检验。

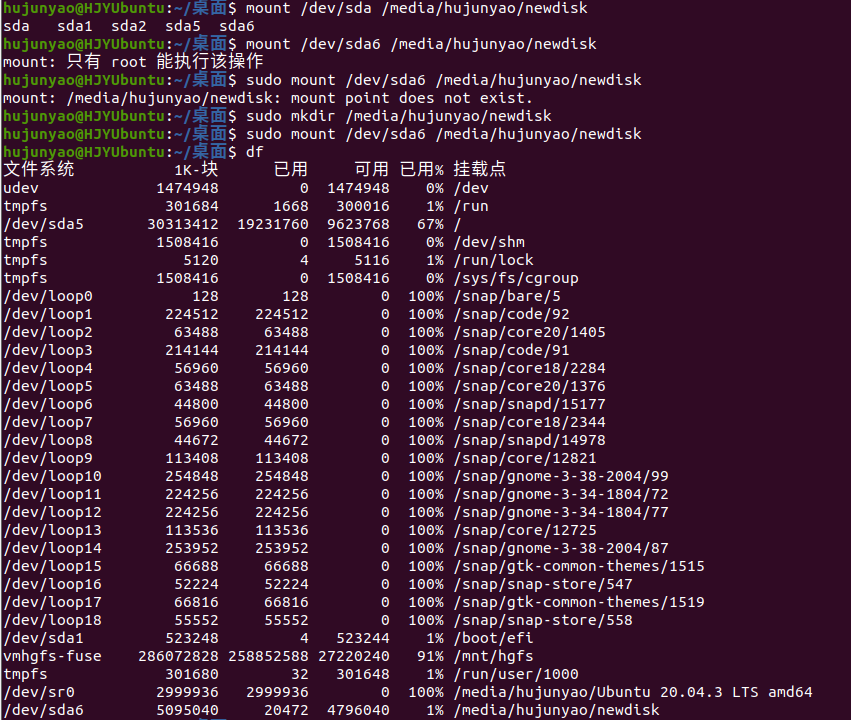


图 1.6‑2

### umount命令

和mount相反，这个命令刚好是为了进行卸载文件系统，和挂载相反。

umount [-hV]

umount -a [选项]

umount [选项] <源> | <目录>

选项：

-a, --all 卸载所有文件系统

-A, --all-targets 卸载当前名字空间内指定设备对应的所有挂臷点

-l, --lazy 立即断开文件系统，清理以后执行

-O, --test-opts <列表> 限制文件系统集合(和 -a 选项一起使用)

-R, --recursive 递归卸载目录及其子对象

-r, --read-only 若卸载失败，尝试以只读方式重新挂臷

-t, --types <列表> 限制文件系统集合

-v, --verbose 打印当前进行的操作

下图实现了对于已经挂载到*/media/hujunyao/newdisk*文件夹下的磁盘设备*/dev/sda6*进行卸载，显示卸载过程，同时使用了df命令对是否卸载进行了检验。

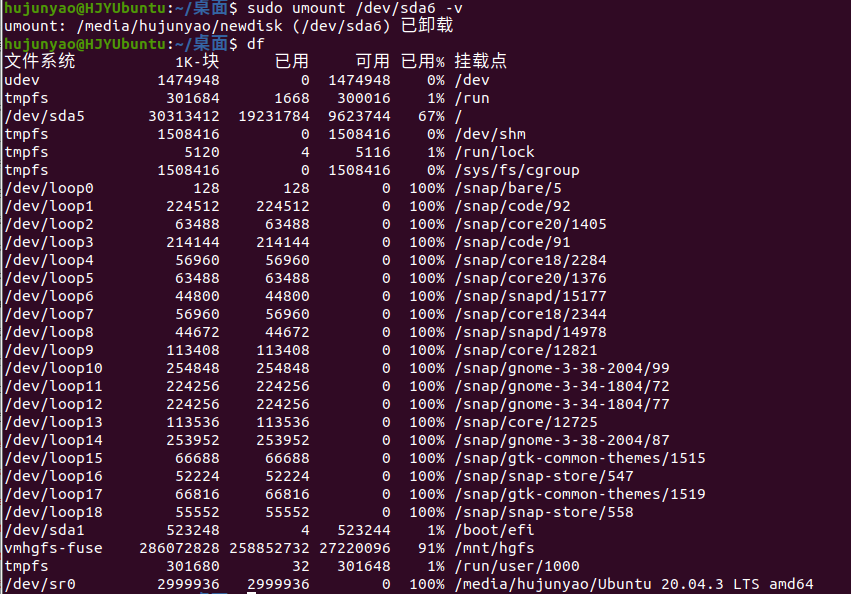


图 1.6‑3

### lsblk命令

主要是列出了块设备的信息，其中可以体现块设备的挂载点的位置。

下图实现了查看挂载和没有挂载的所有各种文件系统类型信息。

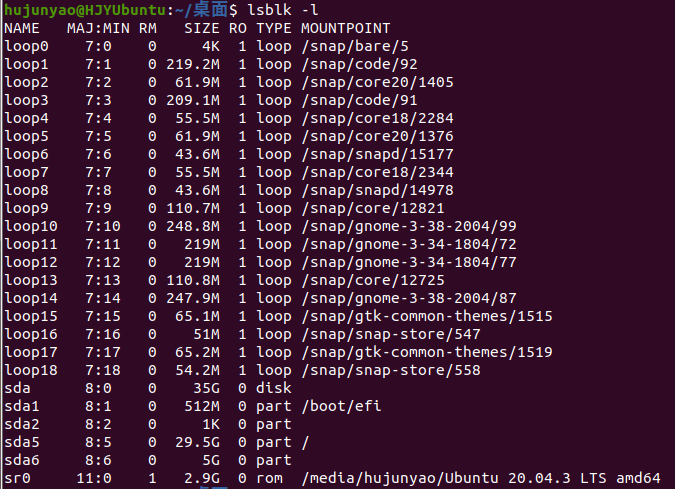


图 1.6‑4

### findmnt命令

该命令用于列出Linux中已挂载的文件系统。该命令将在*/etc/fstab*、*/etc/mtab*、*/proc/self/mountinfo*中查找特定的文件系统。

下图实现了查看部分分区对应的挂载点。

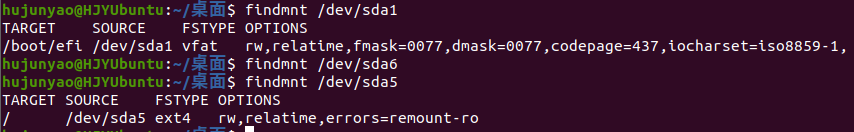


图 1.6‑5

## 用户、组与权限

题目：Linux用户、组及其权限的信息存放在什么文件里，这些文件存放在什么位置？，文件内的信息组织格式是什么？（15分）

在Linux中，万物皆文件，所以用户与组，以及他们的权限，都是以配置文件的形式保存在系统当中，主要有四个文件，这四个文件都存放在根目录下的*etc*文件夹，以下为用户和组的主要配置文件和放置位置以及信息组织格式的介绍。

### 用户及其属性信息

**存放的文件及位置：***/etc/passwd*文件。

**信息组织格式**：下图实现了使用cat命令读取*/etc/passwd*文件，可以查看用户及其属性信息的具体内容。

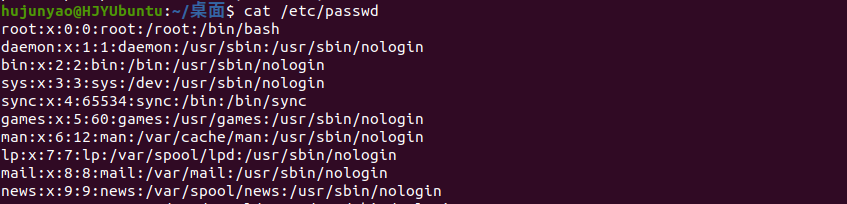


图 1.7‑1 1.7.1 用户及其属性信息文件

下面代码是读取的第一行内容，每一行内容类似。

root:x:0:0:root:/root:/bin/bash

每一项内容由冒号隔开，每一项分别是：登录名（root）、可选的加密后的密码（x）、数字用户ID（0）、数字组ID（0）、用户名和注释字段（root）、用户主目录（*/root*）、可选的用户命令解释器（*/bin/bash）*。

### 用户密码及其相关属性

**存放的文件及位置**：*/etc/shadow*文件。

**信息组织格式**：使用cat命令读取*/etc/shadow*文件，可以查看用户密码及其相关属性的具体内容。

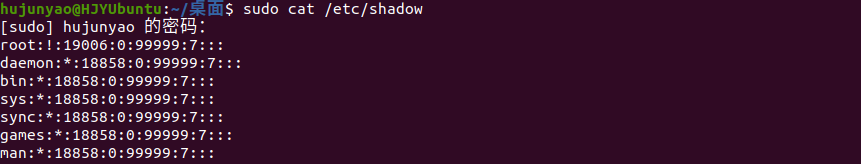


图 1.7‑2 1.7.2 用户密码及其相关属性文件

下面代码是读取的第一行内容，每一行内容类似。

root:!:19006:0:99999:7:::

每一项内容由冒号隔开，每一项分别是：登录名（root）、可选的加密后的密码（！）、最后一次更改密码的日期（最近一次更改密码的时间，表示从1970年1月1日开始的天数，19006）、密码的最小年龄（用户一次更改密码之后，要等多长时间才再次被允许更改密码，这里是0）、最大密码年龄（这写天之后，用户必须更改密码，这里是99999）、密码警告时间段（密码过期之前，提前警告用户的的天数，这里是7，也就是提前一周）、密码禁用期（密码过期后，仍然接受此密码的天数，空字段表示没有强制密码过期）、账户过期日期（空字段表示账户永不过期）、保留字段。

### 组及其属性信息

**存放的文件及位置**：*/etc/group*文件。

**信息组织格式**：使用cat命令读取*/etc/group*文件，可以查看组及其属性信息的具体内容。

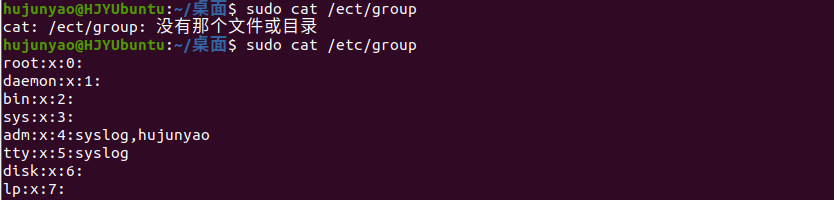


图 1.7‑3 1.7.3 组及其属性信息文件

下面代码是读取的其中一行的内容，每一行内容类似。

sudo:x:27:hujunyao

每一项内容由冒号隔开，每一项分别是：属组名（sudo）、可选的加密后的密码（x）、数字组ID（27）、用户列表（这里是这个组中的一系列用户的列表，使用逗号分隔，这里只有用户hujunyao）。

### 组密码及其相关属性

**存放的文件及位置**：*/etc/gshadow*文件。

**信息组织格式**：使用cat命令读取*/etc/gshadow*文件，可以查看组密码及其相关属性信息的具体内容。

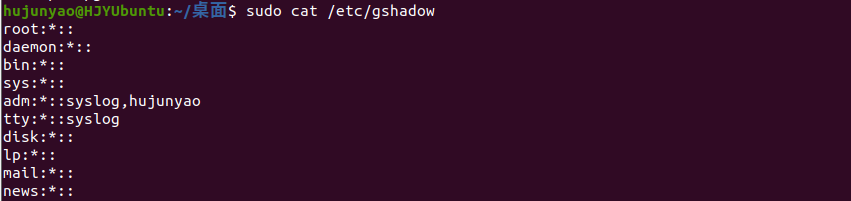


图 1.7‑4 1.7.4 组密码及其相关属性文件

下面代码是读取的其中一行的内容，每一行内容类似。

sudo:\*::hujunyao

每一项内容由冒号隔开，每一项分别是：属组名（sudo）、可选的加密后的密码（\*）、组管理员列表（这里没有）、用户列表（这里只有用户hujunyao）。

## 时间

题目：Linux的硬件时间和系统时间有什么区别？Linux的日期、时间命令有哪些（举例给出）？（10分）

### 硬件时间和系统时间

**硬件时间**：指的是主板上BIOS的时间，在主板上有一个给BIOS供电的电池，在系统开机的时候需要读取该时间，并根据它来设定系统时间，正是因为有这个单独的电池才能在持续断电、关机的情况下，电脑仍然可以进行计时，刚开机联网前的时间和实际时间基本一致的原因也在于此。

**系统时间**：这个时间就是执行date命令看到的时间，Linux系统下所有的时间调用都基于这个时间，除了那些直接调用硬件时间的命令使用硬件时间，如hwclock等。

### date命令

以给定格式的字符串的形式显示当前时间，或者设置系统日期，也就是和系统时间有关的一系列操作在这完成。

date [选项]... [+格式]

date [-u|--utc|--universal] [MMDDhhmm[[CC]YY][.ss]]

选项与参数：

-d, --date=字符串 ：显示给定<字符串>描述的时间，而非“当前时间”

%c ：当前locale 的日期和时间

%X ：显示24小时的格式

-R ：查看时区

下图实现了首先查询了当前时间，然后并将一个时间戳以本地日期格式进行输出，然后以管理员权限设置当前时间为凌晨零点五分，并输出当前系统时间，可知已被修改，经过一分钟左右之后，系统时间自动更新为正确的时间。

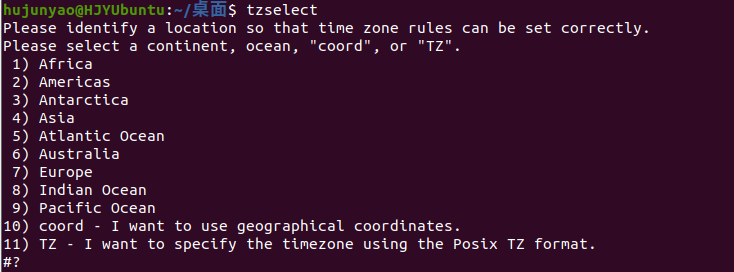


图 1.8‑1

### tzselect命令

该命令用于时区的选择（Timezone Select），使用该命令选择自己所处国家或地区，使用本地的时区。

下图实现了时区的选择，选择中国上海作为本地时区的参考。可能还需要手动执行一步TZ='Asia/Shanghai'; export TZ之后才能更新时区。



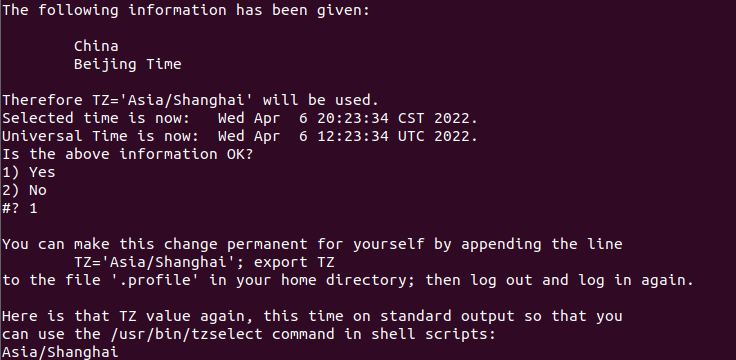


图 1.8‑2

### hwclock命令

该命令用于获取和设定硬件时钟（Hardware Clock）。

hwclock [function] [option...]

选项与参数：

-a ：根据先前的记录来估算硬件时钟的偏差，并用来校正目前的硬件时钟。

--set --date=<日期与时间> ：定硬件时钟

--show ：显示硬件时钟的时间与日期

-w, --systohc ：将硬件时钟调整为与目前的系统时钟一致

-s, --hctosys ：将系统时钟调整为与目前的硬件时钟一致

下图实现了获取硬件时间，将硬件时钟调整为与目前的系统时钟一致，将系统时钟调整为与目前的硬件时钟一致（在系统时钟自动更新之前实现），以及根据先前的记录来估算硬件时钟的偏差。

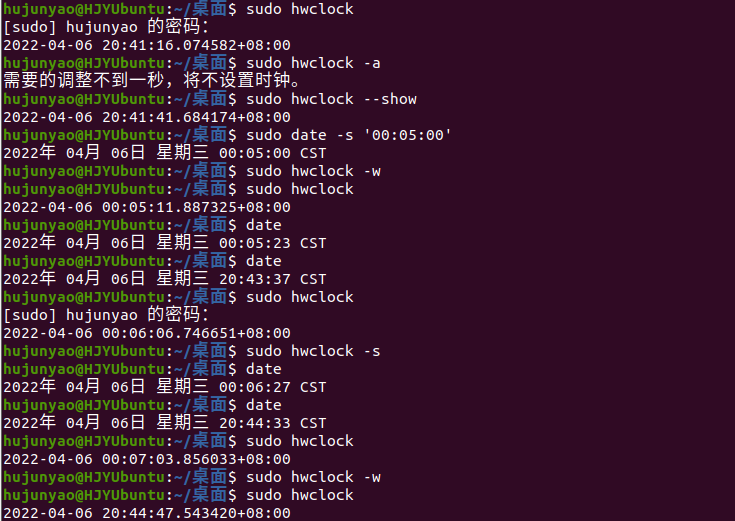


图 1.8‑3

### cal命令

这个命令可以用来在命令行显示月历、年历。

cal [参数][月份][年份]

参数和选项：

-m ：显示指定的月份

-j ：显示在当年中的第几天

-3 ：上个月、这个月、下个月这三个月的月历

下图实现了显示今年12月月历、显示今天在今年是第几天第几周、显示当前时间前中后三个月。

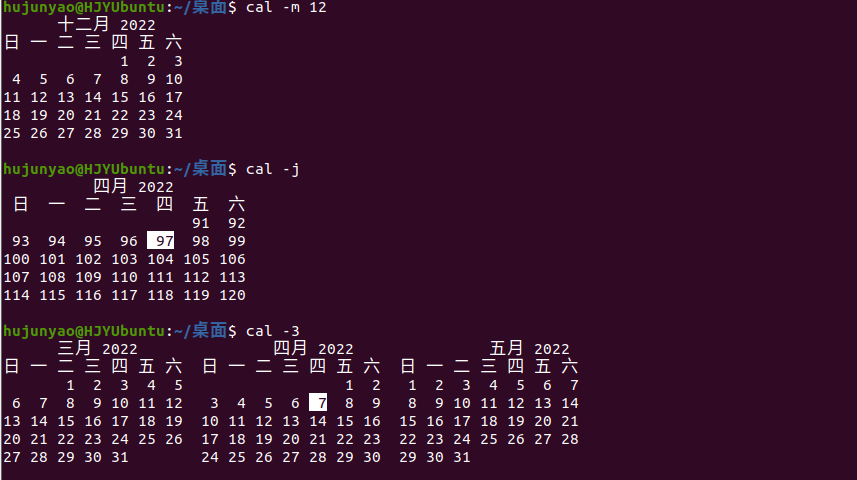


图 1.8‑4

# 感悟

Linux操作系统有很多的命令，课程的学习也是主要围绕命令展开，所以光在课上听理论肯定是不够的，要想熟练肯定得不断进行实践，往往实践出真知，特别是课上的CentOS和Ubuntu操作系统在执行的具体细节上还有一些差别，需要自行在网上查询更多的资料去了解，但是大体的东西都是一致的，这也体现了Linux操作系统的便利。

在学习Linux的过程中，需要理论和实践相结合，做到知行合一，才能学好，主要还是得自己多去尝试，以前只是使用该系统进行ROS的一些相关操作，只知道一些最基本的命令，通过这次作业，巩固了对各种命令的操作和作用的理解，也加深了对操作系统、计算机组成原理等其他课程的一些认识。

# 参考文献

1. 鸟哥. 鸟哥的Linux私房菜（第四版）[M].人民邮电出版社,2019.
2. 刘遄. Linux 就该这么学[M]. 人民邮电出版社, 2017.
3. 菜鸟教程. Linux 教程[OL]. https://www.runoob.com/linux/linux-tutorial.html.