

Ubuntu Linux复习（仅供参考）

试卷格式

- 选择 30`
- 填空 5×2`
- 简答 3×5`
- 综合题 3×10`
- 应用 1×15`

Linux系统组成、特点

Linux系统由以下几个核心组件组成：

- **Linux内核（Linux Kernel）：**
Linux内核是Linux操作系统的核心部分，它管理系统的资源并提供硬件驱动程序。Linux内核负责处理系统的底层操作，如进程管理、内存管理、设备管理和文件系统等。
- **GNU工具集（GNU Tools）：**
GNU工具集是一套与Linux系统紧密结合的工具集合，它包含了各种常用的命令行工具和实用程序。这些工具包括shell（如Bash）、文本编辑器（如Vim和Emacs）、编译器（如GCC）、调试器（如GDB）等。
- **图形用户界面（Graphical User Interface, GUI）：**
Linux系统支持多种图形用户界面，最常用的是X Window System。X Window System提供了窗口管理、图形显示和用户交互等功能。用户可以选择不同的窗口管理器和桌面环境，如GNOME、KDE、Xfce等。
- **Shell（命令行解释器）：**
Shell是用户与Linux系统进行交互的命令行解释器。它接收用户输入的命令，并将其转发给操作系统执行。常见的Linux Shell包括Bash（Bourne Again Shell）、Zsh（Z Shell）和Fish（Friendly Interactive Shell）等。
- **文件系统（File System）：**
Linux系统支持多种文件系统，如Ext4、XFS、Btrfs等。文件系统负责组织和管理文件和目录的存储、访问和权限。它提供了文件的层次结构和文件的元数据（如权限、所有者和创建时间）。
- **网络协议和服务：**
Linux系统支持广泛的网络协议和服务，如TCP/IP、HTTP、FTP、SSH等。它可以作为服务器操作系统，提供Web服务器、文件共享、邮件服务等。

Linux系统具有以下主要特点：

- **完全免费、源代码开放：**
Linux系统是免费获取的，并且其源代码是公开的。这意味着任何人都可以免费使用、修改和分发Linux系统。开放的源代码使得用户能够深入了解系统的运作方式，并且可以根据需要进行自定义和优化。
- **多用户多任务、支持多进程和多线程：**
Linux系统具有多用户和多任务的能力，多个用户可以同时登录并使用系统，每个用户可以同时运行多个程序。此外，Linux系统支持多进程和多线程，使得多个任务可以同时执行，提高了系统资源的利用效率。

- **丰富的网络功能：**

Linux系统提供了强大的网络功能，支持各种网络协议和服务。它可以作为网络服务器提供Web服务、文件共享、邮件服务等。Linux系统还提供了丰富的网络管理和配置工具，方便用户设置和管理网络连接。
- **可靠的安全性和稳定性：**

Linux系统以其良好的安全性和稳定性而受到广泛赞誉。由于开放源代码的特性，Linux系统经过全球开发者的审查和测试，可以及时发现和修复安全漏洞。此外，Linux系统的设计和架构使其能够长时间运行而不受系统崩溃和故障的影响。
- **完全兼容POSIX标准：**

Linux系统完全兼容POSIX（可移植操作系统接口）标准。POSIX是一种操作系统接口标准，定义了一组兼容的API（应用程序接口），使得在符合POSIX标准的操作系统上开发的应用程序能够在Linux系统上无需修改或仅需轻微修改地运行。
- **支持多种平台：**

Linux系统支持多种硬件平台，包括个人计算机、服务器、嵌入式系统等。这种灵活性使得Linux能够运行在各种设备和场景中，并为不同用户提供所需的功能和性能。
- **良好的用户接口：**

Linux系统提供了多种用户接口，包括命令行界面和图形用户界面。命令行界面提供了强大的控制和配置能力，而图形用户界面提供了直观的可视化操作和友好的用户体验。用户可以根据自己的喜好和需求选择适合的界面。

这些特点使得Linux系统成为一种强大、灵活且可靠的操作系统，广泛应用于各个领域和行业。

Linux发行版本

- **Ubuntu：**

Ubuntu是最受欢迎和广泛使用的Linux发行版之一。它注重易用性和用户友好性，并提供了一个强大的图形用户界面（如GNOME），以及丰富的软件包和支持。
- **Fedora：**

Fedora是由Red Hat支持的社区驱动的发行版。它专注于引入最新的软件和技术，并将其测试和集成到稳定的发行版中。Fedora采用了GNOME作为默认的图形用户界面。
- **Debian：**

Debian是一个非常稳定且具有广泛软件库的发行版。它强调自由软件的使用和分发，也是许多其他发行版的基础，如Ubuntu和Linux Mint。
- **CentOS：**

CentOS是以Red Hat Enterprise Linux（RHEL）为基础的开源发行版。它专注于提供一个免费的企业级操作系统，具有稳定性和可靠性，适用于服务器和企业环境。
- **Arch Linux：**

Arch Linux是一种面向技术用户的发行版，注重简洁、灵活和定制性。它采用滚动发布模式，意味着软件包会持续更新，用户可以保持系统最新。
- **openSUSE：**

openSUSE是一种面向桌面和服务器的发行版，注重易用性和可靠性。它提供了多个桌面环境选择，如KDE和GNOME，并具有强大的配置和管理工具。

这只是一小部分常见的Linux发行版，还有其他许多发行版可供选择，如Linux Mint、Gentoo、Slackware、Redhat等。每个发行版都有其自己的特点和目标受众，用户可以根据自己的需求和偏好选择适合的发行版。

NFS的步骤

安装和配置NFS服务器：

- 在NFS服务器上安装NFS软件包。
- 配置NFS服务器的共享目录。编辑NFS服务器的配置文件（通常是 `/etc/exports`），指定要共享的目录和允许访问该目录的客户端。

启动NFS服务：

- 启动NFS服务器进程，以便客户端可以连接和访问共享。
- 在大多数Linux发行版中，可以使用类似以下命令启动NFS服务：

```
sudo service nfs start
```

配置NFS客户端：

- 在NFS客户端上安装NFS软件包。
- 创建本地目录，用于挂载NFS共享。
- 编辑NFS客户端的配置文件（通常是 `/etc/fstab`），添加挂载NFS共享的条目。例如：

```
server:/shared_directory /mnt/nfs_share nfs defaults 0 0
```

挂载NFS共享：

- 在NFS客户端上使用 `mount` 命令挂载NFS共享。如果在步骤3中已经配置了 `/etc/fstab`，可以使用 `mount -a` 命令挂载所有已配置的共享。
- 也可以直接使用 `mount` 命令挂载指定的共享目录。例如：

```
sudo mount server:/shared_directory /mnt/nfs_share
```

验证和访问NFS共享：

- 使用 `df -h` 命令验证NFS共享是否已成功挂载到本地目录。
- 在本地目录中访问和操作NFS共享的文件和目录。

vi的三种模式

Vi编辑器具有三种主要模式，它们分别是：

1. 命令模式 (Command Mode)：

当你打开Vi编辑器时，默认进入命令模式。在命令模式下，你可以执行一系列编辑器命令，如移动光标、复制粘贴、删除文本、保存文件等。你可以使用键盘上的字母键或特殊的快捷键来执行这些命令。

2. 插入模式 (Insert Mode)：

在命令模式下，你可以通过按下 `i` 键进入插入模式。在插入模式下，你可以像在普通文本编辑器中一样输入文本内容。你可以使用键盘上的各种字符键进行文本输入，而不会触发编辑器命令。按下 `Esc` 键可以退出插入模式并返回到命令模式。

3. 底行模式 (Ex Mode) :

在命令模式下, 你可以通过按下冒号 (:) 进入底行模式。底行模式允许你执行更高级的操作, 如搜索替换文本、设置编辑器选项、执行批量操作等。在底行模式下, 你可以输入命令, 并按下 `Enter` 键来执行。执行完命令后, 编辑器将返回到命令模式。

在Vi编辑器的不同模式之间切换是通过按下特定的键或组合键来完成的。例如, 按下 `i` 键可以从命令模式切换到插入模式, 按下 `Esc` 键可以从插入模式或底行模式切换回命令模式。

需要注意的是, Vi编辑器的默认模式是命令模式, 所以在进行文本编辑之前, 你需要进入插入模式才能输入内容。

文件系统创建步骤

1. **选择文件系统类型**: 首先需要确定要使用的文件系统类型, 例如Ext4、NTFS、FAT32等。选择文件系统类型应该基于操作系统和应用程序的要求以及所需的功能和性能。
2. **准备磁盘分区**: 如果你要在磁盘上创建文件系统, 则需要先进行分区操作。这涉及到使用分区工具 (如fdisk、Disk Management等) 在磁盘上创建分区, 并为每个分区分配空间。
3. **格式化分区**: 一旦分区完成, 你需要使用适当的命令或工具对每个分区进行格式化。这将在分区上创建文件系统结构, 并准备用于存储数据的空间。格式化命令通常包含文件系统类型和分区设备名称。例如, 在Linux上, 可以使用 `mkfs` 命令格式化分区, 如 `mkfs.ext4 /dev/sda1`。
4. **挂载文件系统**: 已经格式化的文件系统需要挂载到文件系统树中的某个目录, 以便可以访问和使用它。在Linux中, 你可以使用 `mount` 命令挂载文件系统。例如, `mount /dev/sda1 /mnt` 将分区 `/dev/sda1` 挂载到目录 `/mnt`。
5. **配置自动挂载 (可选)**: 如果你希望文件系统在系统启动时自动挂载, 可以进行相应的配置。在Linux中, 你可以编辑 `/etc/fstab` 文件, 添加一行条目来描述要自动挂载的文件系统。这样, 每次系统启动时, 文件系统都会被自动挂载。

需要注意的是, 上述步骤的具体命令和操作可能因操作系统和文件系统类型的不同而有所变化。因此, 在实际操作时, 应参考所使用的操作系统和文件系统的文档和指南, 以获取准确的步骤和命令。

用户分类/区别

1. 根用户 (Root User) :

- 根用户, 也称为超级用户, 是系统中具有最高权限的用户账户。
- 根用户可以执行系统级任务, 如安装软件、修改系统配置、管理其他用户账户等。
- 根用户的用户名通常是“root”, 其UID为0。

2. 系统用户 (System Users) :

- 系统用户是由系统和服务进程使用的特殊用户账户。
- 系统用户通常没有登录权限, 并且用于执行特定的系统任务和服务, 如网络服务、打印服务等。
- 系统用户的用户名通常以字母“_”或“-”开头, 其UID一般小于1000。

3. 普通用户 (Regular Users) :

- 普通用户是由系统管理员创建的一般用户账户。
- 普通用户用于日常计算和操作, 其权限受到限制, 无法对系统关键部分进行修改。
- 普通用户的权限可以通过用户组和访问控制列表 (ACL) 进行细粒度的控制。
- 普通用户的用户名和UID都有一定的自由度, 可以根据实际需要进行设置。

4. 特权提升 (Privilege Escalation) :

- 在Linux系统中, 普通用户可以通过特权提升来获取临时的超级用户权限。
- 一种常见的特权提升方式是使用 `sudo` 命令, 允许用户以超级用户的权限执行特定的命令。
- 特权提升允许普通用户在需要时获得临时的超级用户权限, 以完成特定的任务, 同时保持系统的安全性。

这些用户分类和区别帮助用户权限的分离和安全性的管理。根用户具有最高权限, 可以执行敏感任务, 而普通用户的权限受到限制, 可以避免意外或恶意操作对系统造成影响。通过正确配置和管理用户账户, 可以提高系统的安全性和可维护性。

Linux常用命令及参数 (不全, 看书)

1. ls: 列出目录内容

- `-l`: 以长格式显示
- `-a`: 显示所有文件, 包括隐藏文件
- `-h`: 以人类可读的格式显示文件大小

2. cd: 切换目录

- `cd`: 切换到当前用户的主目录
- `cd [目录路径]`: 切换到指定目录路径

3. pwd: 显示当前工作目录的路径

4. mkdir: 创建目录

- `mkdir [目录名]`: 创建一个新目录
- `-p`: 创建多级目录, 若父目录不存在则创建

5. rm: 删除文件或目录

- `rm [文件名]`: 删除指定文件
- `-r`: 递归删除目录及其内容
- `-f`: 强制删除, 无需确认

6. cp: 复制文件或目录

- `cp [源文件] [目标文件]`: 将源文件复制到目标文件
- `-r`: 递归复制目录及其内容

7. mv: 移动文件或目录, 或重命名文件

- `mv [源文件/目录] [目标位置]`: 将源文件/目录移动到目标位置
- `mv [旧文件名] [新文件名]`: 将文件重命名为新名称

8. cat: 查看文件内容

- `cat [文件名]`: 显示文件的全部内容
- `-n`: 显示行号

9. grep: 在文件中搜索指定模式

- `grep [模式] [文件名]`: 在文件中查找包含指定模式的行
- `-i`: 忽略大小写
- `-r`: 递归搜索目录中的文件

10. chmod: 修改文件或目录的权限

- `chmod [权限] [文件名]`: 设置文件或目录的权限
- `+/ -`: 添加或取消权限
- `r/w/x`: 读、写、执行权限

11. find: 在文件系统中查找文件

- `find [路径] [条件]`: 在指定路径下根据条件查找文件
- `-name`: 按文件名匹配
- `-type`: 按文件类型匹配

12. tar: 归档和解压文件

- `tar -cvf [归档文件] [文件/目录]`: 创建归档文件
- `tar -xvf [归档文件]`: 解压归档文件

13. ifconfig/ip: 查看和配置网络接口

- `ifconfig` 或 `ip addr`: 显示网络接口的信息
- `ifconfig [接口名] up/down`: 启用/禁用网络接口

14. top: 实时显示系统运行状况和进程信息

15. ps: 显示系统中正在运行的进程

- `ps aux`: 显示所有进程的详细信息

16. kill: 终止运行中的进程

- `kill [进程ID]`: 向指定进程发送终止信号

17. man: 查看命令的手册页

- `man [命令]`: 查看特定命令的帮助文档

18. wget: 从网络下载文件

- `wget [URL]`: 下载指定的文件或网页

19. chown: 更改文件或目录的所有者

- `chown [新所有者] [文件/目录]`: 更改文件或目录的所有者

20. chgrp: 更改文件或目录的所属组

- `chgrp [新所属组] [文件/目录]`: 更改文件或目录的所属组

21. history: 显示命令历史记录

- `history`: 显示最近执行的命令列表
- `!n`: 重新执行历史记录中的第n个命令

22. ln: 创建链接文件

- `ln -s [源文件] [链接文件]`: 创建符号链接文件

23. grep: 在文件中搜索指定模式

- `grep [模式] [文件]`: 在文件中查找匹配指定模式的行

重定向符

在Linux中，重定向符用于控制命令输入和输出的流向。以下是一些常用的重定向符号：

1. `>`: 输出重定向（覆盖）

- `command > file`：将命令的标准输出覆盖到文件中，如果文件不存在则创建，如果文件已存在则覆盖其内容。
- 2. `>>`：输出重定向（追加）
 - `command >> file`：将命令的标准输出追加到文件中，如果文件不存在则创建，如果文件已存在则在末尾追加。
- 3. `<`：输入重定向
 - `command < file`：将文件的内容作为命令的标准输入。
- 4. `2>`：错误输出重定向（覆盖）
 - `command 2> file`：将命令的错误输出（标准错误）覆盖到文件中。
- 5. `2>>`：错误输出重定向（追加）
 - `command 2>> file`：将命令的错误输出（标准错误）追加到文件中。
- 6. `|`：管道
 - `command1 | command2`：将command1的输出作为command2的输入，实现两个命令之间的数据传递。
- 7. `&>`：输出和错误输出重定向（覆盖）
 - `command &> file`：将命令的标准输出和错误输出同时覆盖到文件中。
- 8. `&>>`：输出和错误输出重定向（追加）
 - `command &>> file`：将命令的标准输出和错误输出同时追加到文件中。

日期时间操作

- `date`：显示当前日期和时间
- `date "+%Y-%m-%d"`：以指定的格式显示当前日期（例如：2023-06-02）
- `date "+%H:%M:%S"`：以指定的格式显示当前时间（例如：13:45:30）
- `date -s "2023-06-02"`：设置系统日期为指定日期
- `date -s "13:45:30"`：设置系统时间为指定时间

日期格式（更全面）

- `%Y`：四位数的年份（例如：2023）
- `%m`：两位数的月份（01-12）
- `%d`：两位数的日期（01-31）
- `%H`：24小时制的小时数（00-23）
- `%I`：12小时制的小时数（01-12）
- `%M`：分钟数（00-59）
- `%S`：秒数（00-59）
- `%A`：完整的星期几名称（例如：Sunday）
- `%a`：缩写的星期几名称（例如：Sun）
- `%B`：完整的月份名称（例如：January）
- `%b` 或 `%h`：缩写的月份名称（例如：Jan）
- `%p`：AM或PM

- `%j`：年份中的一天（001–366）
- `%U`：一年中的第几周，以周日为一周的起始（00–53）
- `%W`：一年中的第几周，以周一为一周的起始（00–53）
- `%w`：星期几的数字表示（0–6，其中0代表周日）
- `%C`：年份的前两位数
- `%y`：两位数的年份（00–99）
- `%Z`：时区偏移量（例如：+0800）
- `%Z`：时区名称
- `%%`：百分号
- `date "+%Y-%m-%d %H:%M:%S"`：以YYYY-MM-DD HH:MM:SS格式显示当前日期和时间（例如：2023-06-02 13:45:30）
- `date "+%A, %B %d, %Y"`：以完整的星期几、月份和日期显示当前日期（例如：Sunday, June 02, 2023）

☆☆☆ ls -l 命令解释（重点）

在Linux中，`ll` 命令通常是 `ls -l` 命令的别名，用于以长格式显示文件和目录的详细信息。执行 `ll` 命令后，会显示类似以下的输出结果：

```
-rw-r--r--  1 user group 4096 Jun  2 13:45 file.txt
drwxr-xr-x  2 user group 4096 Jun  2 13:45 directory
```

在命令中 第一位代表了文件是什么

- `-`：表示是一个常规文件（Regular file），具有可读、可写和可执行的权限。
- `d`：表示是一个目录（Directory），具有可读和可执行的权限。目录的可写权限表示可以在该目录中创建、删除和重命名文件。
- `l`：表示是一个符号链接（Symbolic link），即指向另一个文件或目录的快捷方式。符号链接没有特定的权限，而是继承了目标文件的权限。
- `b`：表示是一个块设备（Block device），用于访问块设备文件（如硬盘分区）。
- `c`：表示是一个字符设备（Character device），用于访问字符设备文件（如终端设备）。
- `s`：表示是一个套接字（Socket），用于进程间的通信。
- `p`：表示是一个命名管道（Named pipe），用于进程间的通信。

1. 文件/目录权限（File/Directory Permissions）：

- `-`：表示是一个文件，`d`：表示是一个目录，其它见上。
- 接下来的9个字符（例如：`rw-r--r--`）表示文件或目录的权限。每3个字符代表一组权限，分别表示文件所有者、所属用户组和其他用户的权限。每个字符的含义为：
 - `r`：可读权限
 - `w`：可写权限
 - `x`：可执行权限
 - `-`：无相应的权限

2. 硬链接和符号链接数（Hard Links and Symbolic Links Count）：

- 1: 表示文件或目录的硬链接数（硬链接指向相同的索引节点）
- 如果是目录，则接下来的数字表示目录的符号链接数（符号链接是指向其他目录的快捷方式）
- 3. 文件所有者和所属组（File Owner and Group）：
 - user: 表示文件或目录的所有者
 - group: 表示文件或目录的所属组
- 4. 文件大小（File Size）：
 - 4096: 表示文件的大小（以字节为单位），或者对于目录来说，表示目录的大小（目录大小显示为磁盘上所占用的空间）
- 5. 修改日期和时间（Modification Date and Time）：
 - Jun 2 13:45: 表示文件或目录的最后修改日期和时间
- 6. 文件/目录名称（File/Directory Name）：
 - file.txt 和 directory: 表示文件或目录的名称

JDK和Tomcat的部署

在Linux中部署JDK（Java Development Kit）和Tomcat服务器涉及以下步骤：

1. 下载JDK和Tomcat：
 - 访问Oracle官方网站 (<https://www.oracle.com/java/technologies/javase-jdk11-downloads.html>) 下载适用于Linux的JDK版本。
 - 访问Apache Tomcat官方网站 (<https://tomcat.apache.org/download-10.cgi>) 下载适用于Linux的Tomcat版本。
2. 解压JDK文件：
 - 使用 tar 命令解压下载的JDK文件，例如：

```
tar -xf jdk-11.0.12_linux-x64_bin.tar.gz
```
3. 将JDK移动到合适的目录：
 - 创建一个目录，例如 /usr/lib/jvm，并将解压后的JDK文件移动到该目录：

```
sudo mkdir /usr/lib/jvm
sudo mv jdk-11.0.12 /usr/lib/jvm/
```
4. 配置环境变量：
 - 打开 ~/.bashrc 文件或 ~/.bash_profile 文件（根据您的系统环境和Shell设置）：

```
vim ~/.bashrc
```
 - 在文件末尾添加以下内容，修改 /usr/lib/jvm/jdk-11.0.12 为您实际的JDK路径：

```
export JAVA_HOME=/usr/lib/jvm/jdk-11.0.12
export PATH=$JAVA_HOME/bin:$PATH
```
 - 保存文件并执行以下命令使环境变量生效：

```
source ~/.bashrc
```

5. 配置Tomcat:

- 解压下载的Tomcat文件, 例如:

```
tar -xf apache-tomcat-10.0.12.tar.gz
```

- 将解压后的Tomcat目录移动到适当的位置, 例如 `/opt` :

```
sudo mv apache-tomcat-10.0.12 /opt/
```

6. 启动Tomcat:

- 切换到Tomcat目录并执行启动脚本:

```
cd /opt/apache-tomcat-10.0.12/bin  
./startup.sh
```

7. 验证Tomcat的运行:

- 在Web浏览器中访问 `http://localhost:8080`, 您应该看到Tomcat的欢迎页面。