

# 苏州大学实验报告

院系	计算机学院	年级专业	21 计科	姓名	方浩楠	学号	2127405048
课程名称	Linux 操作系统					成绩	
指导教师	曹敏	同组实验者	无	实验日期	2023. 11. 2		

## 实 验 名 称 实验 5

### 一. 实验目的

熟悉 Linux 环境下的基本操作和命令，增强对 Linux 系统的实际应用能力。

学习如何配置和管理 Linux 系统中的环境变量。

掌握文件和目录的常用操作命令。

理解和掌握命令的组合与重定向技巧。

深入了解 Linux 进程控制的概念和方法。

加深对 Linux 环境变量的认识，并学习其配置和应用。

### 二. 实验内容

设置环境变量，调整 `.bash_history` 文件可以保存的命令历史数量。

使用命令批量修改文件名，将大写字母转为小写。

针对 `/etc/passwd` 文件，使用 `sort`、`cut` 和管道符进行数据筛选和排序。

利用通配符查询特定文件和目录。

学习命令输出的重定向，将输出同时显示在屏幕并保存到文件。

创建命令别名，简化长命令的调用。

掌握命令组合符号，如 `;`、`&&` 和 `||` 的使用。

总结 Linux 中的进程控制概念和方法。

对 Linux 环境变量进行综述，探讨其配置和应用。

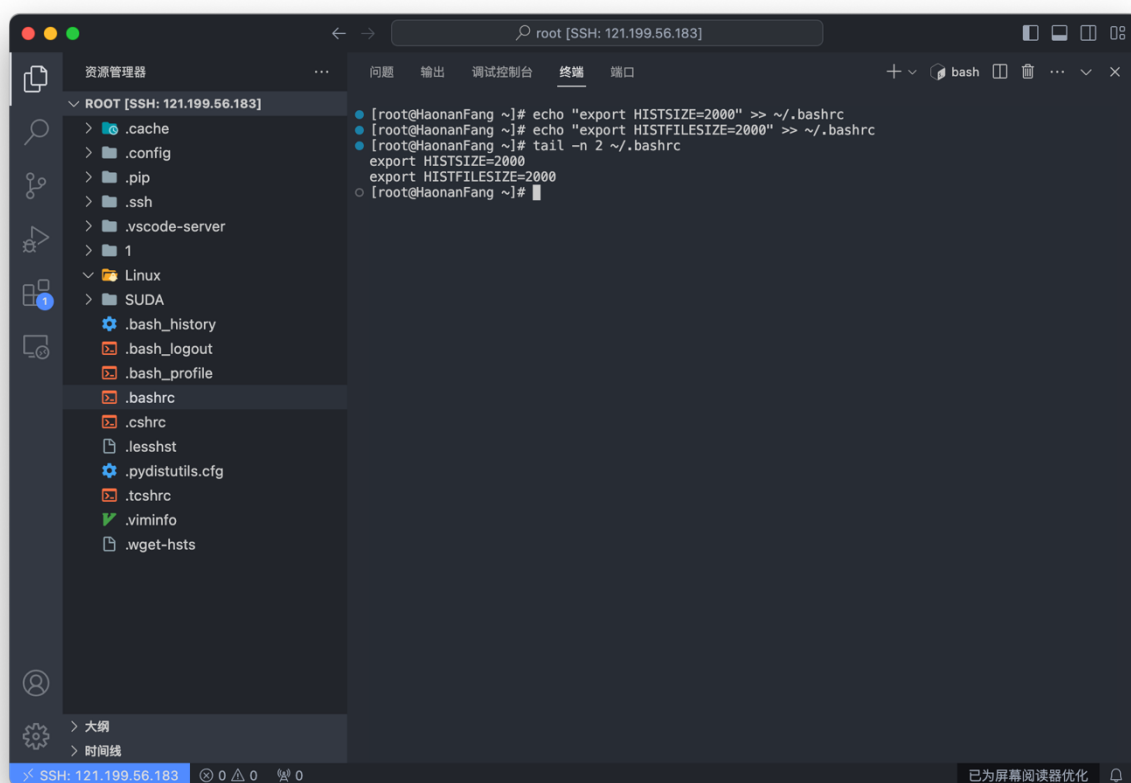
### 三. 实验步骤和结果

1. 设置环境变量，使 `.bash_history` 文件可以保存 2000 条命令历史。

命令:


```
1 echo "export HISTSIZE=2000" >> ~/.bashrc # ~/.bashrc追加内容
2 echo "export HISTFILESIZE=2000" >> ~/.bashrc # 向~/.bashrc追加内容
3 tail -n 2 ~/.bashrc # 查看~/.bashrc最后两行的内容
```

执行结果:



```
root [SSH: 121.199.56.183]
[root@HaonanFang ~]# echo "export HISTSIZE=2000" >> ~/.bashrc
[root@HaonanFang ~]# echo "export HISTFILESIZE=2000" >> ~/.bashrc
[root@HaonanFang ~]# tail -n 2 ~/.bashrc
export HISTSIZE=2000
export HISTFILESIZE=2000
[root@HaonanFang ~]#
```

2. 把当前目录下文件的文件名的大写字母替换为小写字母  
命令:

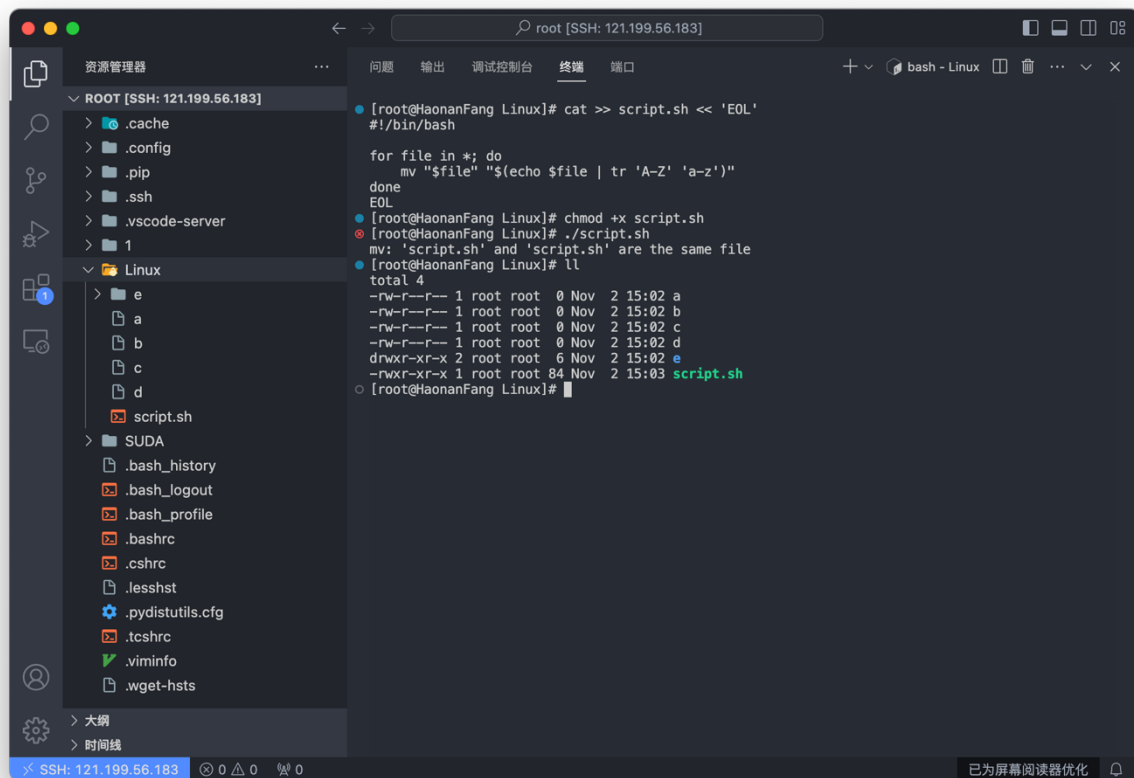


```
1 $ cat >> script.sh << 'EOL'
2 #!/bin/bash
3
4 for file in *; do
5     mv "$file" "$(echo $file | tr 'A-Z' 'a-z')"
6 done
7 EOL
8
9 $ chmod +x script.sh
10 $ ./script.sh
```

执行结果:

创建一个 **bash** 脚本用来执行该命令

保存该文件,同时使用 **chmod** 给该文件添加权限,然后运行该文件

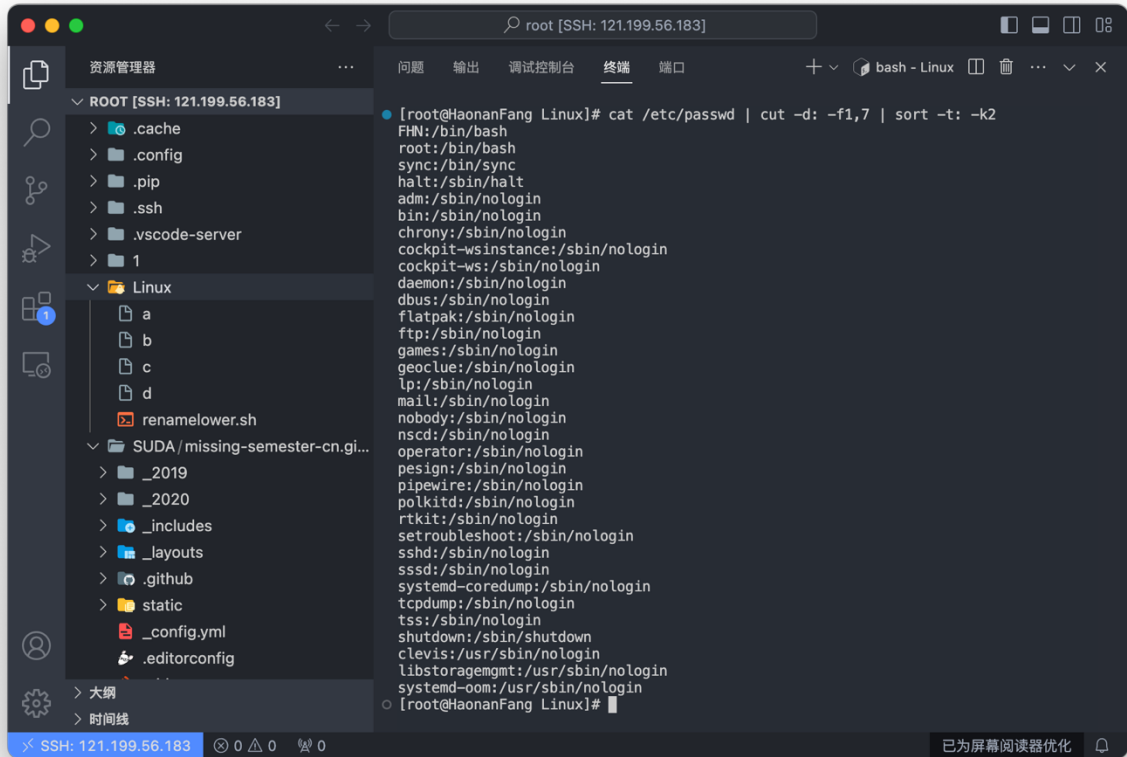


发现运行后文件夹中的大写字母的文件均变成了小写字母的文件

3. 针对/etc/passwd 文件，自己设计场景，实现 sort、cut 和管道符|的搭配使用  
场景：得到所有用户的用户名和他们的默认 shell，并按照 shell 进行排序  
命令：



执行结果：



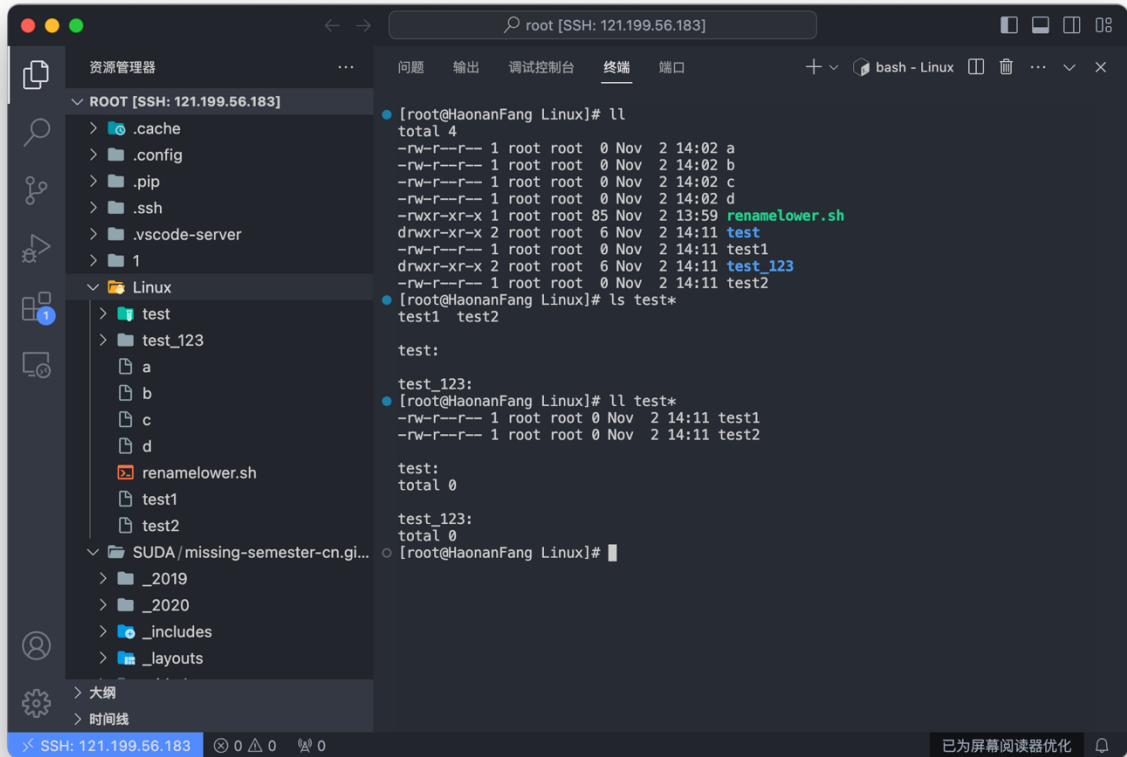
```
[root@HaonanFang Linux]# cat /etc/passwd | cut -d: -f1,7 | sort -t: -k2
FHw:/bin/bash
root:/bin/bash
sync:/bin/sync
halt:/sbin/halt
adm:/sbin/nologin
bin:/sbin/nologin
chrony:/sbin/nologin
cockpit-wsinstance:/sbin/nologin
cockpit-ws:/sbin/nologin
daemon:/sbin/nologin
dbus:/sbin/nologin
flatpak:/sbin/nologin
ftp:/sbin/nologin
games:/sbin/nologin
geoclue:/sbin/nologin
lp:/sbin/nologin
mail:/sbin/nologin
nobody:/sbin/nologin
nscd:/sbin/nologin
operator:/sbin/nologin
design:/sbin/nologin
pipewire:/sbin/nologin
polkitd:/sbin/nologin
rtkit:/sbin/nologin
setroubleshoot:/sbin/nologin
sshd:/sbin/nologin
sssd:/sbin/nologin
systemd-coredump:/sbin/nologin
tcpdump:/sbin/nologin
tss:/sbin/nologin
shutdown:/sbin/shutdown
clevis:/usr/sbin/nologin
libstoragemgmt:/usr/sbin/nologin
systemd-oom:/usr/sbin/nologin
[root@HaonanFang Linux]#
```

4. 当前目录下以 test 开头的文件和目录列出。

命令:



执行结果:

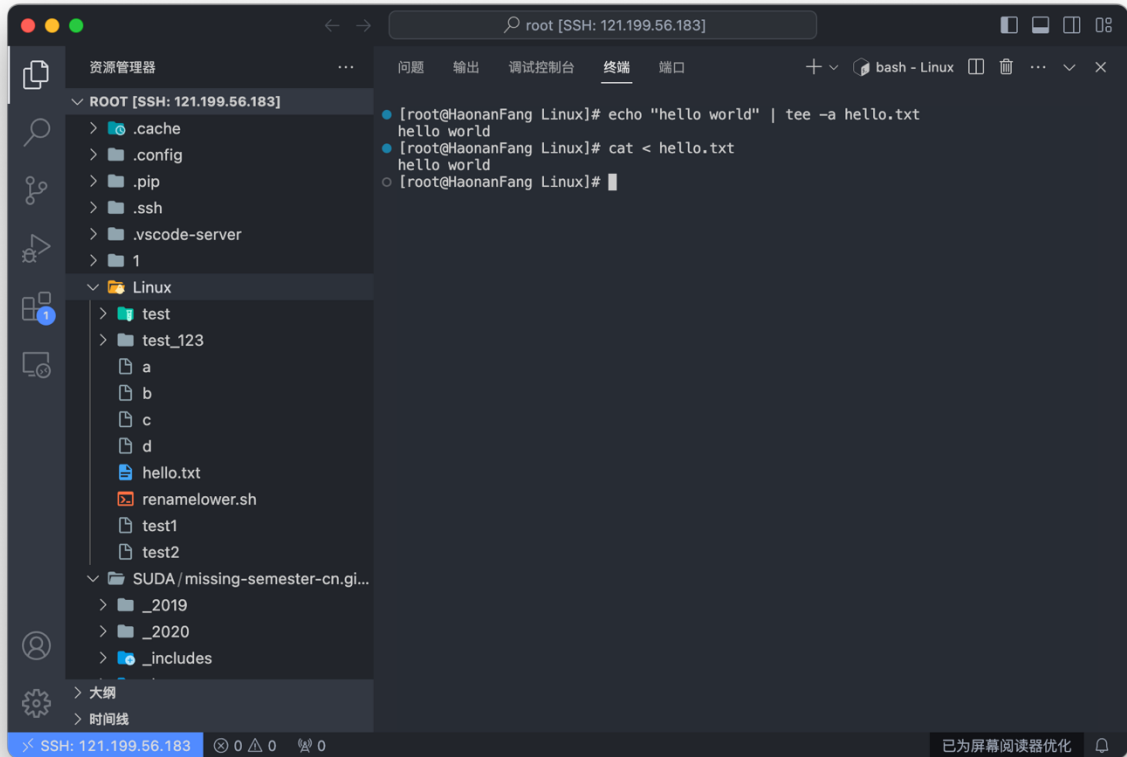


5. 把一个命令的输出内容打印到屏幕，同时重定向到一个文件内  
命令:



执行结果:

发现 hello world 被同时输入到了屏幕和 hello.txt 中



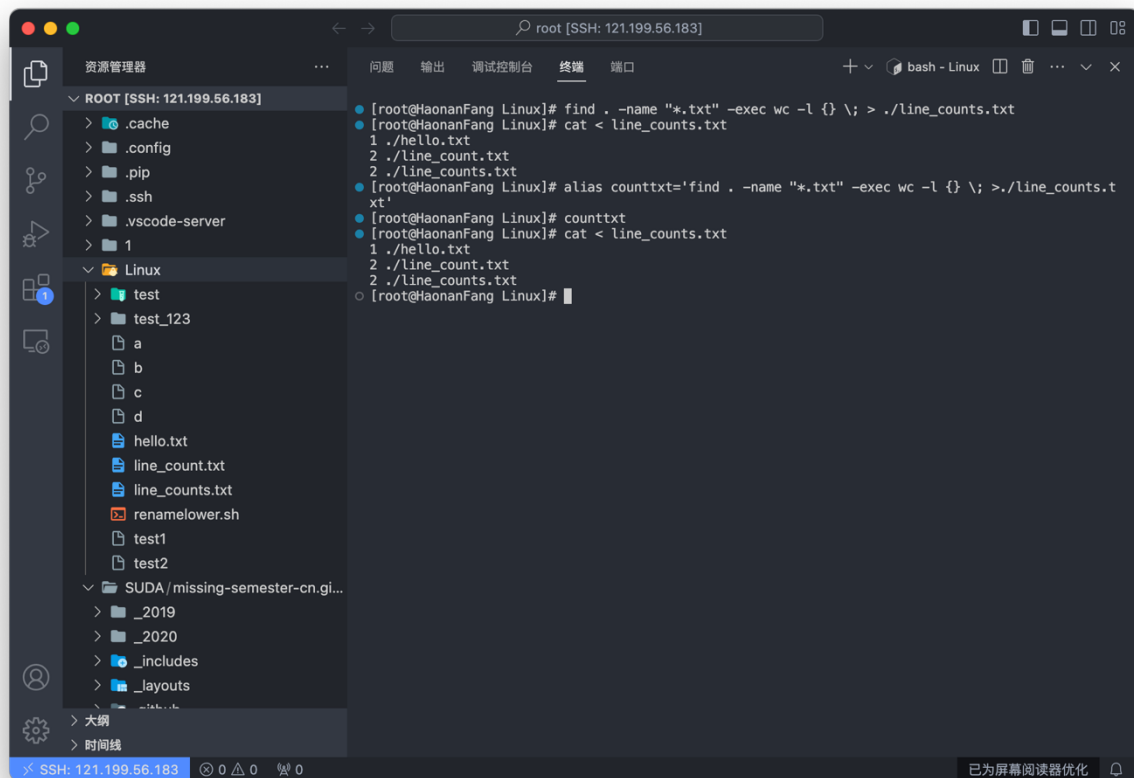
6.

假设我们有以下长命令，它查找所有 .txt 文件，计算这些文件的行数，并将结果输出到 line\_counts.txt 文件中，我们可以为上述长命令创建一个简短的别名，如 counttxt,这样输入 counttxt 就可以快速计算行数

命令:



执行结果:



7.

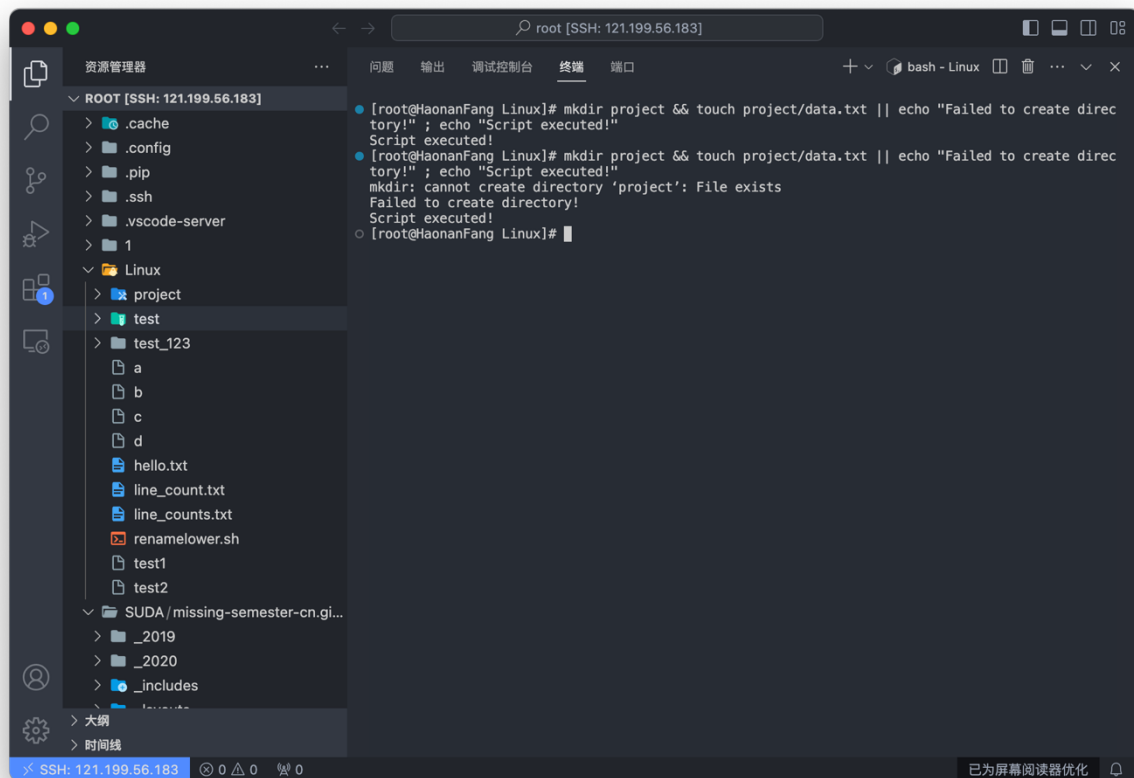
创建一个名为 `project` 的目录。如果成功创建目录，将在其中创建一个名为 `data.txt` 的空文件。在 `mkdir` 命令失败时（例如，因为 `project` 目录已存在）打印一个错误消息不论之前的命令是否成功，你都希望在终端中打印出 `"Script executed!"`。

命令:



执行结果:





## 7. Linux 进程控制综述，介绍和总结

### 进程简介：

在 Linux 中，一个进程是一个运行中的程序的实例。每个进程都有一个独特的进程 ID (PID)。

### 进程的状态：

运行态 (Running)：进程正在执行或在等待执行。

中断态 (Interruptible sleep)：进程正在等待某些条件成真，例如等待 I/O 操作完成。

不可中断态 (Uninterruptible sleep)：进程在等待某些特定条件，但不能被其他事件中断。

僵尸态 (Zombie)：进程已结束，但其父进程还没有获取其退出状态。

停止态 (Stopped)：进程已停止，不再执行。

### 进程管理命令：

ps: 显示当前进程状态。

top: 动态显示进程状态。

pstree: 显示进程树。

kill: 向进程发送信号。

nice 和 renice: 调整进程优先级。

### 进程的创建和终止：

fork(): 创建一个新的子进程，这是最常用的进程创建方法。

exec(): 在当前进程上下文中执行一个新程序。

`wait()`: 父进程等待子进程终止。

`exit()`: 结束进程执行。

后台和前台进程:

在 Linux 中, 可以使用 `&` 将进程放到后台运行。

`fg` 命令可以将后台进程调到前台。

`bg` 命令可以将停止的进程放到后台运行。

## 8. Linux 环境变量综述, 介绍和总结

环境变量简介:

环境变量在 Linux 中是存储系统设置和应用程序配置的键值对。它们可以被 shell 和程序使用。

常见的环境变量:

`PATH`: 定义命令查找路径。

`HOME`: 用户的主目录。

`USER`: 当前登录的用户名。

`PS1`: shell 提示符。

`LD_LIBRARY_PATH`: 动态链接库查找路径。

设置和显示环境变量:

`export`: 设置环境变量。例如: `export VARNAME=value`。

`unset`: 删除环境变量。

`printenv`: 显示环境变量。

`env`: 列出所有环境变量。

配置文件:

在 Linux 中, 有多个文件和目录用于配置环境变量:

`/etc/environment`: 系统范围内的环境变量。

`/etc/profile` 和 `/etc/profile.d/*`: 系统范围内的 shell 启动脚本。

`~/.bashrc` 和 `~/.bash_profile`: 用户级别的 shell 启动脚本

## 四. 实验总结

在此次实验中, 我们系统地学习和应用了 Linux 中的核心概念和常用命令。通过对环境变量的配置和管理, 我们理解了其在 Linux 系统中的重要性。文件和目录操作的实践增强了我们的文件管理能力。同时, 通过命令的组合、重定向和别名的应用, 我们提高了命令行的使用效率。进程控制和环境变量的总结加深了我们对 Linux 系统内部工作机制的理解。

此外, 实验中的操作和实践进一步巩固了我们的理论知识, 使我们能够更加熟练地应对日常的系统管理和配置任务。总的来说, 这次实验不仅提高了我们的技能和知识, 还增强了对 Linux 系统的兴趣和信心。