

****

《实用操作系统》实验报告

（一）

**姓　　名 ：宋泽涛**

**学　　号 ：25120222201292**

**学 院 ：信息学院**

**专 业 ：软件工程**

**2024年 9 月**

# 实验目的

* 熟悉Linux系统的基本指令
* 熟悉Linux系统的基本软件（如vi、gdb、gcc等）
* 查看Linux系统中进程的信息

# 实验环境

* + PC + Linux操作系统
  + vi、gcc、gdb

# 实验内容

* + 熟悉linux环境的基本使用命令
  + 熟悉使用vi等文本编辑工具
  + 熟悉使用gdb、gcc等编程工具
  + 利用who、w、ps和top等指令查看Linux下的进程执行情况。

# 实验具体实现

* **Linux使用体会（与Window、Mac等相比）**

1. 与Windows和macOS的封闭系统不同，Linux的源码完全开放，这意味着用户可以根据自己的需求进行自定义。无论是修改系统内核、安装个性化的桌面环境，还是针对特定应用进行优化，都能够实现。这种高度的自由度是其他系统难以企及的。
2. 与Windows和macOS在不断迭代中对硬件要求越来越高相比，Linux依旧能够在老旧电脑上流畅运行。特别是一些轻量级的发行版，如Lubuntu、Xubuntu等，它们能够充分利用硬件资源，保持流畅的用户体验。
3. Linux系统默认支持多种编程语言、编译器和工具链，使用包管理工具（如apt、yum等）能够快速安装开发环境。而Windows的命令行工具较为复杂，macOS虽然有类似的Unix环境，但相比Linux的社区资源和开源工具库，稍显逊色。

* **Linux进程信息（写出具体含义）**

1. PID：进程的唯一标识符。每个运行的进程都有一个唯一的PID，它是操作系统分配给进程的整数。
2. PPID：父进程的PID。每个进程都有一个父进程，PPID指示该进程的父进程的PID。
3. UID和EUID：UID是启动进程的用户的标识符。EUID表示进程实际运行时的权限用户ID，通常与UID相同，但如果进程使用了setuid权限，它可以不同。
4. GID和UGID：GID表示进程所属的用户组ID，EGID则是进程实际使用的组ID。
5. RSS：进程实际占用的物理内存大小（以KB为单位）。

* **gcc和dgb的使用方式**

1. gcc的基本语法：gcc [选项] 源文件 -o 输出文件。其中，-o用于指定输出的可执行文件名。如果没有指定，则默认生成a.out文件。以下是一些常用的选项
   1. -o <output>：指定输出的可执行文件名。
   2. -Wall：打开所有常见的编译警告，帮助检测潜在的代码问题。
   3. -g：在编译时生成调试信息，以便在GDB中调试使用。
   4. -c：只编译不链接，生成目标文件（.o文件）。
   5. -I<directory>：指定头文件的搜索路径。
2. dgb通常用于调试C、C++程序。以下是dgb的常用指令
   1. run [args]：启动并运行程序，args可以传递给程序的命令行参数。
   2. break <location>：在指定位置设置断点，位置可以是函数名、文件名:行号等。
   3. continue：在遇到断点后继续执行程序。
   4. print <variable>：打印变量的当前值。
   5. info locals：查看当前作用域内的所有局部变量及其值。
   6. backtrace（bt）：显示当前调用栈，方便分析程序的调用顺序。