1. 冯诺依曼体系结构

五大硬件单元：

输入设备

输出设备

存储器（内存）

运算器 运算速度快

控制器 运算速度快

CPU(运算器 控制器)能且只能对内存进行读写,不能访问外设(输入 输出设备)

机械硬盘（200MB/s）读写速率小于固态硬盘

固态硬盘有两种接口类型:①SATA(500-800MB/S) ②PCIE(3000-4000MB/s)

**硬件结构决定软件行为**

二进制鼻祖----冯诺依曼

1. 操作系统(是搞管理的软件)

包含两个部分:①内核 ②外部应用

软硬件资源管理以及为上层应用提供良好的执行环境

库函数与系统调用接口的关系:上下层的封装调用关系；库函数实际就是对系统调用接口的封装

管理的概念:先描述再组织；管理者将被管理者描述起来，然后组织起来进行管理

1. 进程

用户角度:

写的东西叫程序--运行起来的程序就是进程,程序运行起来需要被加载到内存

操作系统角度:

进程就是操作系统对进程的描述(PCB)，每一个进程的描述都是一个结构体，Linux下PCB有自己的名字叫task-struct;

Linux操作系统如何管理进程:使用task-struct结构体**描述**进程，使用双向链表将这些结构体组织起来进行**管理**

进程的描述信息：标识符（PID） 状态 IO 记账

内存指针：保存程序代码和进程相关的数据指针

优先级：因为进程之间存在竞争

程序计数器：保存下一步要进行的命令操作

上下文数据：进程执行时cpu中的寄存器中的数据（eg：休学当兵）

**并行：多个程序在多个CPU下分别同时进行**

**并发：多个进程在一个CPU下采用进程切换方式,在一段时间内让多个进程都得以推进**

cpu的分时机制：多个程序分时轮流进行

时间片：程序在cpu上处理的时间

交互式进程：有反馈的进行

批处理进程：连续不断持续进行

1. 查看进程

ps -ef 查看操作系统所有进程信息

ps -ef |grep [文件] 查看指定文件进程信息

ls -l /proc/进程ID/

ps aux 查看操作系统所有进程信息

getpid(); 系统函数：返回调用函数的进程ID

1. 通过系统调用创建进程--fork()

操作系统以父进程为模板创建子进程 父子进程代码段相同，但是数据各自独有(子进程复制了父进程的pcb)。

fork的返回值起到分流的作用，我们用户通过fork的返回值来判断，到底哪个进程是父进程，哪个是子进程。

对于*父进程*，返回值是*子进程的id*

对于*子进程* 返回值是*0*

出错了返回值是**-1**

1. 进程状态

R运行状态(running):随时可以运行或正在运行

S可中断睡眠状态

D不可中断(磁盘休眠)睡眠状态:进程会等待IO的结束

T停止状态

X死亡状态

Z僵尸状态

*僵尸进程:*

产生原因:

1.子进程先于父进程退出,而父进程没有关心他的状态

2.进程退出为了保存自己退出原因,因此这个退出资源不会被完全释放,等待父进程来获取退出状态,然后释放子进程所以资源

3.假如父进程没有关心子进程的退出,那么这个退出子进程将成为一个僵尸 进程,占据着系统资源不释放,

危害:造成资源泄露 正常进程可能无法创建

*孤儿进程:*

产生原因:

父进程先于子进程退出,那么子进程就成为了孤儿进程,这个孤儿进程将被孤儿院init进程所领养,子进程退出后将由init进程回收释放资源.

1. 优先级

交互式进程:优先级高

批处理进程:

PID:进程ID

PRI:进程的优先级,优先级越小,进程越先被执行,加入nice值后PRI(new)=PRI(old)+nice

NI:进程的nice值,取值范围:-20--19,共40个级别,进程可被执行的优先级的修正后的数值

**nice** 调整优先级

**renice** 程序运行后调整优先级

top 进入后 r 进程PID 输入NICE值

1. 环境变量(操作系统中有特殊功能的全局参数)

如何查看: env echo set

如何设置:export

如何删除:unset

代码中如何操作:getenv() environ

1. 程序地址空间

(虚拟地址空间)

命令行参数环境变量

栈区

共享区

堆区

未初始化全局数据段

初始化全局数据段

代码段

内存地址:对内存的编号,一个地址指向内存的一个位置

页表:记录虚拟地址与物理地址之间的映射关系,内存访问控制

虚拟地址空间是什么:是一个虚拟的地址,是一个结构体

虚拟地址空间有什么用:让每一个进程都能独立.