操作系统

叶茂林 2024年3月28日星期四

**物理地址和逻辑地址**

内存单元真正的地址，进程执行指令和访问数据的真正地址，程序暴露给程序员的地址

**虚拟内存**

当物理内存不足时操作系统会将一部分数据暂时存储在磁盘上，使得应用程序认为它拥有连续完整可用的地址

**进程和线程**

进程是资源分配的基本单位，线程是处理机任务调度和执行的基本单位，一个进程可以有多个线程

不同的进程之间的地址空间和资源是相互独立的，同一进程的线程共享进程的地址空间和资源

通信方式不同，创建的过程开销不同

**协程**

用户态的轻量级线程，线程内部调度的基本单位

**管程**

定义公共数据结构为并发进程执行的一组操作，进程调用管程访问临界资源

**临界资源**

同一时刻只能有一个进程使用的资源

**临界区**

进程中访问临界资源的代码段

**守护进程**

后台运行的独立于终端的进程

**创建守护进程**

后台运行程序，创建子进程后结束父进程，调用setsid创建新对话期成为会话组长脱离原终端、原进程组和登陆会话，再次创建子进程退出父进程卸任会话组长，这样就无法打开终端，关闭文件描述符，将当前目录改为根目录，将屏蔽字清零，忽略SIGCHLD信号避免产生僵尸进程

**孤儿进程**

父进程先结束，子进程成为孤儿进程，父进程变成init（1号）进程

**僵尸进程**

子进程结束父进程没有捕获它的退出状态，子进程变成僵尸进程，可以忽略SIGCHLD信号避免僵尸进程

**局部性原理**

时间局部性：最近访问的内存不久后可能还会被访问

空间局部性：当前访问的附近内存不久后可能会被访问

**操作系统中的task\_struct、mm\_struct和vm\_area\_struct有什么联系**

Task\_struct是进程控制块PCB的数据结构，用于管理和控制进程，存储了进程的信息，其中有一个mm\_struct指针指向，mm\_struct是描述进程内存布局的数据结构，其中有一个vm\_area\_struct指针指向一个链表，vm\_area\_struct描述了虚拟地址空间的一个区间，区间少时使用链表管理，区间多时使用红黑树管理

**进程地址空间**

代码段、数据段、BSS段、堆、内存映射段、栈

**大小端**

小端：数据的低字节存储在低地址

大端：数据的高字节存储在低地址

**解决死锁**

忽略死锁：当死锁影响不大，或者发生死锁的概率很低，可以不管它

死锁检测与死锁恢复：

死锁检测：检测是否存在环路等待

死锁恢复：抢占、回滚、杀死

死锁预防：在程序运行前破坏死锁发送的条件

死锁避免：在程序运行时避免发送死锁