

系统编程第12周

一.

在OS层面，内核使用 ECF 用来提供进程概念抽象，进程最重要的两个方面是其提供的在资源分配上的重要抽象，即利逻辑控制流和私有地址空间两个抽象，让其看起来像是在独占 cpu 和主存。

代码段A的子进程会不断生成子进程，类似于树的结构。

代码段A的子进程会直接跳出循环。

原因：子进程中fork返回0

三.

终止而未被回收的进程。

调用 `wait()` 函数等价于调用 `waitpid(-1, &status, 0)` 父进程调用函数后被挂起，直到其所有子进程中的某个终止，然后函数返回中止的 PID 并将该子进程从系统中去除。

四.

信号是软件层面的异常，用来进行进程间通信。

五.

某个信号在被阻塞后第一次发送给进程，该信号就处于未决状态，信号信息储存在 `pending` 位向量里面，作用是防止信息过度丢失，比如正在处理一个 `SIGINT` 信号的过程中时候，`SIGINT` 就会被阻塞，此时如果再发送一个 `SIGINT` 信号，该信号就会处于未决状

态。

六.

将某个信号的 handler 改为接受到该信号后停止读写内存，直到下一次接受到该信号，想要进行读写操作的进程先检测是否接受到该信号次数，若为偶数则向各进程广播一个该信号并开始读写，如果为奇数则等待再次接受到该信号。