1.

进程是计算机中的程序关于某数据集合上的一次运行活动，是系统进行资源分配和调度的基本单位，是操作系统结构的基础，可以清晰地刻画动态系统的内在规律，有效管理和调度进入计算机系统主存储器运行的程序。

2.

第一个程序会创建32个进程，第二个程序会创建6个进程。

原因：

第一个程序，每循环一次，系统会复制原程序，接着两个文件都继续向下执行，也就是父进程和子进程都会继续向下执行按照此规律，每一次循环，进程总数变为当前进程数量的两倍。

第二个程序，循环创建进程，加入判断，若当前进程为子进程，则跳出循环。

3.

僵尸进程是当子进程比父进程先结束，而父进程又没有回收子进程，释放子进程占用的资源，此时子进程将成为一个僵尸进程。

调用wait()函数的进程会被挂起，进入阻塞状态，直到子进程变为僵尸态，wait()函数捕获到该进程的退出信息时才会转为运行态，回收子进程资源并返回。

4.

信号用来通知进程发生了异步事件。进程之间可以互相通过系统调用kill发送软中断信号。内核也可以因为内部事件而给进程发送信号，通知进程发生了某个事件。

5. 一个已经产生的信号，但是还没有传递给任何进程，此时该信号的状态就称为未决状态。

信号未决状态用于解决常规信号不可靠问题，通过暂时将信号设为未决状态，避免由于信号发送过于密集使得进程将后收到的信号丢弃的现象发生。

6.

生产者进程:

每次读取输入文件的一帧，

然后将其写到共享内存中。

消费者进程:

每次从共享内存中读到一帧，

处理后，

将其写到输出文件。

两个进程间使用信号量来保证同步处理每一帧。

生产者进程流程：

打开输入文件并初始化帧;

创建并初始化共享内存和信号量;

然后每次读取一帧，

用信号量获取共享内存的权限后，

将读取的帧写入共享内存，

再释放共享内存的权限。

当处理完所有的帧后，

设置结束标志，

并释放相关的资源。

消费者进程的基本流程:

打开输出文件并初始化帧;

获取共享内存和信号量;

每次

得到共享内存的权限后，

从共享内存中读取一帧并获得结束标志

进行帧处理，

释放共享内存的权限。

直到结束标志为真。

最后释放相关的资源。