1.进程的用途：进程可以控制和协调程序的执行。程序是存放在磁盘上的，如果不执行程序，程序就没有任何作用；而执行程序时，就会启动一个进程，用来存储、维护、管理“控制程序运行的那些东西”，将程序和系统为程序分配的资源结合起来，从而完成程序的运行。

2.代码段A：父进程会创建31个子进程，总共有32个进程。代码段B：父进程会创建5个子进程，总共有6个进程。

代码段A中，第一次循环，父进程通过fork()创建1个子进程1；第二次循环中，父进程依然会通过fork()创建一个子进程2，但此时第一次循环中的子进程1和父进程是相同的，也会通过fork()创建一个子进程3，此时子进程1是子进程3的父进程，换句话说，此时已经有4个进程在运行，且这4个进程在第二次循环结束之后都会继续执行相同的操作。第三次循环中，之前的每个子进程都继续创建一个子进程，总共就会有8个进程；第四次，总共有16个进程；第五次循环，进程总数就会变为32。

代码段B中，加了一条判断进程pid是否为0的语句，而对于一个父进程来说，其子进程的pid为0。第一次循环中，父进程创建一个子进程1，但第二次循环中，父进程pid不为0，还是可以继续创建子进程2；而第一次循环中的子进程1的pid为0，无法继续创建它的子进程，而是直接跳出循环；第三次循环中，父进程继续创建子进程3，子进程2也像子进程1一样无法创建它的子进程；第四次，第五次循环均是如此。所以，只有父进程在5次循环中创建了5个子进程，总共有6个进程。

3.僵尸进程：一个进程从调用了exit函数结束其运行到这个进程被回收（消失）的这段时间中，这个进程的状态叫僵尸态，该进程就是僵尸进程。

父进程调用wait时，父进程即被阻塞等待子进程退出。wait()函数可以查询父进程的子进程是否退出。如果父进程收到子进程已退出的信息，就销毁退出的进程。wait()函数中有一个参数status，作用是收集子进程退出后的信息（子进程是否正常退出等等）。如果父进程单纯的只想销毁退出后的子进程，那么status的具体值可以被忽略。

4.Linux系统中的信号是实现进程间通信的方式之一，是一种特殊的异步消息。内核根据用户或系统的需要将信号发送给进程，进程会对信号进行处理，可以对不同的信号进行响应，执行信号所要求的操作（如终止进程、读终端等）。

5.当信号被阻塞而无法到达进程时，内核会修改未决信号集中该信号对应的位为1，从而将该信号设置为未决状态。解除阻塞后，信号被传递到进程，未决状态转变为递达状态。

未决状态是阻塞状态或者说信号被屏蔽时表现的一种状态，它表示某一时刻有信号到达（不是递达状态），但由于阻塞而没有被处理。阻塞有时指的并非一种状态，而是一个动作（类似开关），而未决状态就是阻塞动作的体现。

6.

（1）用semget()函数创建新的信号量，将信号量初始化为1。

（2）用shmget()函数创建一段共享内存，用shmat()函数将共享内存映射到虚拟地址空间。

（3）进行P操作（获取信号量），向共享内存中写入数据，写入结束后，进行V操作释放信号量。

（4）进行P操作（获取信号量），从共享内存中读数据，读数据结束后，进行V操作释放信号量。

（5）删除共享内存和信号量。