1.

进程用于控制和管理系统资源调配，描述程序执行过程并协调其执行。

2.

A中合计有32个进程产生，而B中合计只有6个进程产生，其原因在于B中每次fork产生新进程，子进程都会break跳出循环，从而不参与下一循环的fork操作。

3.

父进程创建子进程后未通过wait等待子进程结束而后进行，此时子进程若已终止，则其不会被主动回收，造成了进程的终止态存在，即其占用资源，但无用，称为僵尸进程。

进程调用wait()或waitpid()函数，此函数等待子进程结束，清理进程并通过传入的 &status 返回子进程的结束码。

4.

信号用于在进程之间进行消息的异步传递，其可用于在系统崩溃错误发生之前被动终止进程或由人员手动通过键盘输入进行主动终止进程，也可用于在进程之间传递信息等。

5.

当信号产生但未递达，即为未决，通常指由于信号阻塞导致的信号未递达，即信号未决。由于信号递达即被处理，在某些不可中断操作中若不进行信号阻塞可能会导致出错，但若无信号未决状态则会丢失信号，这种丢失无法被信号发送者知晓，故进行阻塞，在阻塞取消后进行递达，保证操作安全，信息完整。

6.

首先通过sem\_open创建读和写的信号量，mode参数取不冲突的一特定值 myshm，保证读写操作仅在两进程间进行。

通过shmget分配或获取一段共享内存，附加IPC\_CREAT和myshm读写权限，而后通过shmat进行连接。

在读操作中通过sem\_wait()控制读信号，在信号示意可读时进行后续读操作，最后sem\_post()控制写操作信号，示意可写。

而在写操作中通过sem\_wait()控制写信号，在信号示意可写时进行后续写操作，最后sem\_post()控制读操作信号，示意可读。

即可实现两进程读写同步