1. “进程”这个概念能够方便Linux系统进行程序运行的管理，可以让“死”的程序被计算机执行，从而实现了计算机的各种功能，并且合理分配系统资源。
2. 代码A每次循环后，进程的数量都是当前进程的两倍，所以5次循环后进程总数为32个。这是因为代码A实际执行的时候，子进程也会创建进程。

代码B总共创建了5个子进程，因为代码B每次循环的时候都会判断，如果不是父进程，就会跳出循环。

1. 僵尸进程就是一个进程调用了exit（）函数回收自己之后，在回收过程中它不再是进程，不能继续完成自己所有的回收，就是说这个进程调用了exit（）之后不能完全结束并且消失，会在内存中留下数据结构，这个数据结构就叫做僵尸进程。僵尸进程必须等待父进程来回收。

如果调用wait（）进程，那么函数就会被挂起，进入阻塞状态，直到子进程变成僵尸进程。Wait函数捕获到这个子进程的推出信息才会转换为运行姿态，然后回收子进程的资源，并且返回。如果没有变成僵尸进程的子进程，拿wait（）函数就会一直阻塞。

1. 信号是用来提醒进程的某些异步事件的发生的，是Linux系统中进程间通信的方式之一。从而实现对进程的一些操作。进程是收到信号以后根据信号的处理函数来执行的。
2. 信号在发送的信号被在阻塞，无法到达进程的时候，信号会处于未决状态。

作用：未决状态允许进程对信号的阻塞，防止信号打断一些敏感操作。内核能够通过信号来pa短是否再次向进程发送这个信号。

1. 分别创造两个进程，一个写，一个读。信号量集中有两个信号量，一个代表能否读，一个代表能否写。

写进程中，通过shmget函数创建一个共享内存，然后用shmat函数将进程和共享内存绑定。然后向共享内存里写入数据，最后用shmdt函数解除内存的绑定。

读进程中，通过shmget函数创建一个共享内存，然后用shmat函数将进程和共享内存绑定，然后向共享内存写入数据，最后用shmdt函数解除绑定。最后用shmctl函数删除共享内存。

编译写程序，写入数据，编译读程序，读取数据。