1.

进程将要执行的程序加载到内存中，系统为进程分配资源。

代码段A中，实际上循环结束后一共有32个进程。因为每一次fork后，相当于复制了原程序，父进程会继续进行下一次循环，子进程同样也会进行下一次循环创建自己的子进程。

代码段B中，循环结束后一共有6个进程，1个父进程，5个子进程，因为每一次循环后，子进程将跳出循环，只有父进程会继续循环。

2.

子进程退出时，虽然其在内存中的内容基本清除了，但是会在进程列表中记录下该子进程的退出状态等资料，如果父进程没有结束并且一直没有对该子进程回收，那么这个记录将一直存在，这就是僵尸进程。

父进程调用wait时，如果此时没有进入僵尸进程的子进程，那么父进程将被阻塞，即不继续运行，而是等待某一个子进程结束成为僵尸进程，然后wait将会回收该僵尸进程留下的退出状态信息，如果wait参数不为null，那么将用int指针保存该子进程的退出状态。

3.

信号用于进程的通讯与一些操作。当有键盘事件，硬件异常或者用户调用Kill，raise等操作时，内核就会产生信号发送给进程，进程可以忽略也可以不忽略信号，然后进行对应操作。

4.

当信号被阻塞时，信号处于未决状态，即信号被发送了，但没有递达。解除阻塞后，信号就会递达。

如果一个信号的处理程序中的一些操作会对进程的一些操作产生影响，譬如进程在对一个全局变量操作，信号的处理程序也会对该全局变量操作，那么为了避免混乱或者不稳定，就可以先阻塞信号使其为未决状态，当进程的操作不受影响后解除阻塞，使信号递达，然后进程再处理信号。

5.

设置两个信号量，一个控制读，一个控制写。初始化读信号量为0，写信号量为1，当一个进程进行完写操作后，释放写信号量和读信号量，即写完后才可以读或者继续写。