

1

将程序和数据封装在一起，抽象形成独立的上下文，减少程序间的耦合，便于进行进程切换，多进程等活动

2

代码段A总计会有32个进程，B总计会有6个进程，`fork()`调用后，子进程会返回0，A中子进程仍然处于循环中，产生新的进程，而在B中，由于子进程返回0，会让子进程跳出循环，就不会再产生新进程了

3

运行完毕已经产生退出值（使用`exit()`或`return`）的进程叫僵尸进程

进程调用`wait()`后自身会被阻塞，如果有已经结束的子进程，子进程就会被销毁，`wait()`返回子进程ID并将参数中的`int* status`设置为子进程的返回值

4

用于系统中的通信，比如协调进程间对共享数据的访问，对其它进程进行操纵，完成事件响应等

5

当信号产生后，并未送达到目的地，此时的信号就叫作未决信号

未决状态实际上是对信号进行缓存的一种方式，一旦接收信号的进程选择将信号阻塞，信号就会以未决信号的方式缓存起来，直到进程解除阻塞。

6

设置两个信号量w和r，对信号量进行修改的操作是原子操作，伪代码如下

读操作

```
1  if (w>0) {
2      \\ refuse to read
3  } else {
4      r++;
5      \\ read
6      r--;
7  }
```

写操作

```
1  if (w>0) {
2      \\ refuse to read
3  } else {
4      w++;
5      while (r>0) {
6          \\ wait util no process read
7      }
8      \\ write
9      w--;
10 }
```