1. **每个概念被创造都有其意义，请简述“进程”这个概念在Linux系统中有什么用途。**

答：进程的用途：进程是工作中的程序，每个命令会对应一个进程，计算机需要知道当前有多少个进程，这样才能为进程分配CPU的使用权。此外，考虑到程序是死的而进程是活的，进程可以通过信号控制被终止或者暂停，而程序不具备这样的灵活性。

1. **阅读以上代码段，回答代码段A和代码段B的执行结果有什么区别？并解释为什么会有这样的区别。**

答：程序一：生成32条进程信息

程序二：生成6条进程信息

原因：程序一中，fork（）执行的每一次都是既创建了父进程的子进程，也创建了子进程的子进程。一份文件复制一次是两次文件，进行五次这样的过程也就是2的5次幂=32

程序二中，由于判断了pid的值（父进程pid！=0,子进程pid=0），因此只会创建父进程的子进程，一共创建了五个子进程，所以加上父进程有六个进程。

1. **用自己的话阐述什么是僵尸进程，并描述进程通过调用 wait() 捕获僵尸态的子进程的过程。**

答：僵尸进程就是已经结束的子进程没有被其父进程回收，资源也没有被回收，处于这样的状态的子进程。

调用wait后，进程被挂起，进入阻塞态。直到出现了处于僵尸状态的子进程，父进程将子进程回收，才继续执行父进程。

1. **请简述信号在Linux系统中的作用。**

答：信号设计之初，是用于提醒进程某件事情已经发生。后来，信号被应用于进程间的通信。不同事件发生时，信号会被内核发送到特定的进程。信号可以终止进程、暂停进程、继续运行进程。

1. **请简述信号什么时候处于未决状态，并简述信号存在未决状态的作用。**

答：信号在产生和传递之间的时间范围内，称此信号是未决的。例如，当一个进程正在处理一个信号，而另一个信号被发送过来时，那么进程将会把后发送过来的信号阻塞，这样后发送过来的信号就处于未决状态，等待之前的信号被处理完。

作用：假如信号传递得过于密集，无法在一个信号处理完之前处理下一个发送过来的信号，没有未决状态的话，下一个发送过来的信号就不会被执行；但这个信号又被发送方发送了，不会再发送一遍，就会产生信号缺失的问题。未决状态正是为了解决这一问题。

1. **请设计一种通过信号量来实现共享内存读写操作同步的方式，文字阐述即可，不需要代码实现。（提示：在写进程操作未完成时，需要防止其他进程从共享内存中读取数据）**

答：具体操作为：获取信号量标识符，然后创建信号量并进行初始化操作。之后，获取共享内存标识符，将当前进程连接该共享内存段，让写数据的程序先运行，然后sleep一段时间等程序写完，再sem\_p占用这个信号量，做完系列操作之后，sem\_v释放信号量，并删除信号量。最后，用shmdt函数进程和共享内存脱离，并删除共享内存