

## 第一题：进程这个概念的意义是什么

进程可以唯一表示操作系统上运行的一个程序。因此，它的主要作用就是实现在操作系统上，同时运行多个程序，并且给这多个程序分配各种运算、储存资源

## 第二题：两端代码有什么不同？

第一段代码：每一次在for循环中都要进行一次fork。第一次for循环变成了两个进程，第二次是4个，以此类推，循环结束时一共有32个进程

```
int i;
for(i=0;i<5;i++){
    pid=fork();
}
```

第二段代码：每一个for循环都会进行一次fork，但是产生子进程之后，立刻就在子进程中把for循环给停了，所以只有最初的进程可以走完整个for循环

```
int i;
for(i=0;i<5;i++){
    if(pid=fork()==0){
        break;
    }
}
```

## 第三题：什么是僵尸进程？简述wait()捕获僵尸进程的过程。

僵尸进程：一个已经执行过exit的进程，但是其计算、储存资源还没有被系统回收。

wait()如何捕获僵尸进程：wait会把一个进程挂起，直到有一个子进程变成僵尸进程之后，让系统把它回收，并且继续进行当前进程。如果在wait执行之前，就有进程结束了，那么也会立刻回收那个僵尸进程，然后继续当前进程的执行。

## 第四题：简述信号在Linux系统中的作用

信号本质上来讲就是一个进程发送给另一个进程的异步消息，用来告诉另一个进程某些重要的事情发生了。另一个进程受到信号以后，应该立刻停止当前的操作，并且执行信号处理程序。某种意义上讲，这是软件层面对中断机制的一种模拟

## 第五题：信号什么时候处于未决状态，有什么用处

未决状态：如果某个进程的进程控制块中的信号掩码把某个信号给屏蔽了，那么当这个进程收到该信号的时候，这个信号就会变成未决状态。

用处：如果没有未决状态的话，信号发出去，如果被屏蔽了，就相当于废弃了，那就没有意义了。但是因为信号未决机制的存在，这个信号会在将来不屏蔽的时候被送到进程中。

## **第六题：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**

定义一个信号量，然后不同的进程之间从共享内存读写之前，先进行P操作，结束的时候进行V操作