目录

[操作系统面试十题 1](#_Toc514426573)

[进程和线程区别 1](#_Toc514426574)

[线程同步方式有哪些 1](#_Toc514426575)

[进程通信方式有哪些 1](#_Toc514426576)

[缓冲区溢出是？有什么危害？原因是？ 2](#_Toc514426577)

[死锁是什么？死锁产生的条件是？如何解决死锁？ 2](#_Toc514426578)

[进程有哪几种状态？ 2](#_Toc514426579)

[分页和分段的区别？ 2](#_Toc514426580)

[操作系统进程调度策略有？ 3](#_Toc514426581)

[进程同步有哪些机制？ 3](#_Toc514426582)

[死锁处理的基本策略和常用方法？ 3](#_Toc514426583)

[操作系统笔试题总结： 3](#_Toc514426584)

# 操作系统面试十题

## 进程和线程区别

进程：系统进行资源分配和调度的一个独立单位

线程：进程的实体，是CPU分配和调度的基本单位

一个进程可以有多个线程，多个线程可以并发执行

<http://www.ruanyifeng.com/blog/2013/04/processes_and_threads.html> 阮一峰对进程和线程的理解以及对线程同步方式的理解。工厂的例子。

## 线程同步方式有哪些

互斥量：采用互斥对象机制，只有拥有互斥对象的线程才有访问公共资源的权限。具体点说，java的synchronized，知道了吧。

信号量：允许同一时刻多个线程访问同一个资源，但是需要控制同一时刻访问此资源的最大线程数量。

事件（信号）：一个线程在处理完一个任务后，主动唤醒另一个线程执行任务。

<http://www.cnblogs.com/lebronjames/archive/2010/08/11/1797702.html>

## 进程通信方式有哪些

管道、系统IPC（包括消息队列、信号量、共享存储）、SOCKET

管道：普通管道PIPE、流管道（s\_pipe）、命名管道（name\_pipe）。管道的数据只能单向流动，且只能在父子进程之间流动。

命名管道，允许无亲缘关系的进程间通信。

信号量：也可以在进程之间使用。

<https://blog.csdn.net/yufaw/article/details/7409596>

## 缓冲区溢出是？有什么危害？原因是？

缓冲区是：高级语言定义的变量，数组，结构体等，都是保存在缓冲区内的，因此所谓的缓冲区可以更抽象地理解为一段可读写的内存区域。

缓冲区溢出：计算机向缓冲区填充数据时超出了缓冲区本身的容量，溢出的数据覆盖在合法数据上。

危害：程序崩溃，导致拒绝服务；跳转并且执行一段恶意代码。

造成原因：程序中没有仔细检查用户输入。

<http://www.cnblogs.com/fanzhidongyzby/archive/2013/08/10/3250405.html>

## 死锁是什么？死锁产生的条件是？如何解决死锁？

死锁：两个或多个并发进程中，如果每个进程持有某种资源而又等待其它进程释放它或他们现在保持着的资源，在未改变这种状态之前都不能向前推进，称这一组进程产生了死锁。相互等待。

死锁的条件：四个条件缺一不可。

互斥条件：一个资源一次只能被一个进程使用。

请求与保持条件：一个进程因请求资源而阻塞时，对已获得的资源保持不放。

不剥夺条件：进程获得的资源，在未完全使用完之前，不能强行剥夺。

循环等待：若干进程之间形成一种头尾相接的环形等待资源关系。

解决死锁的思路：

预防死锁：

避免死锁：

检测死锁：

解除死锁：

## 进程有哪几种状态？

就绪状态：进程已获得除嘞处理机以外所需的所有资源，等待分配处理机资源

运行状态：处于此状态的进程数小于等于CPU数。

阻塞状态：进程等待某种条件，在满足条件之前无法执行。

## 分页和分段的区别？

段是对用户可见的，页是信息的物理单位，是为了管理主存的方便而划分的，对用户不可见。

段的大小不固定，页大小固定，由系统决定

段向用户提供二维地址空间，页向用户提供一维地址空间

段是信息的逻辑单位。

## 操作系统进程调度策略有？

先来先服务，优先级，时间片轮转，多级反馈

## 进程同步有哪些机制？

原子操作：

信号量机制：

自旋锁管程：

会合：

分布式系统：

<http://www.cnblogs.com/sonic4x/archive/2011/07/05/2098036.html>

## 死锁处理的基本策略和常用方法？

常用方法：鸵鸟策略、预防策略、避免策略、检测与解除死锁。

## 操作系统笔试题总结：

<https://blog.csdn.net/hackbuteer1/article/details/6787354>