进程和线程的区别

1.进程是运行中的程序，线程是进程的内部的一个执行序列

2.**进程是资源分配的最小单位**，**线程是**程序执行的最小单位（**资源调度的最小单位**）

3.**进程有自己的独立地址空间**，**每启动一个进程，系统就会为它分配地址空间，建立数据表来维护代码段、堆栈段和数据段**，这种操作非常昂贵。而**线程是共享进程中的数据的，使用相同的地址空间，因此CPU切换一个线程的花费远比进程要小很多**，同时创建一个线程的开销也比进程要小很多。

4.线程之间的通信更方便，同一进程下的线程共享全局变量、静态变量等数据，而进程之间的通信需要以通信的方式（IPC)进行。不过如何处理好同步与互斥是编写多线程程序的难点

5.但是多进程程序更健壮，多线程程序只要有一个线程死掉，整个进程也死掉了，而一个进程死掉并不会对另外一个进程造成影响，因为进程有自己独立的地址空间。

6.进程与线程的资源

**线程共享：进程代码段、进程的公有数据**(利用这些共享的数据，线程很容易的实现相互之间的通讯)、进程打开的文件描述符、信号的处理器、进程的当前目录和进程用户ID与进程组ID。

**线程独有：栈（保存其运行状态和局部自动变量）、程序计数器**。

<http://liruiha.com/2019/03/20/progress-thread01/>

进程与线程的同步

什么是同步：**同步就是数据保持一致**，无论是进程还是线程，都是实现了代码执行流程的分支，多个分支同时进行。多个分支互不干扰，但是又有些数据需要共享，让这些数据对所有分支保持一致即为同步。

什么是通信：**通信就是数据传输**，数据存在两块不同的内存区域。通过某种方式互相传递。

但是在进程线程中，比如面试官问你进程同步有那些方式，管道算是同步还是通信？干脆也懒得区分，还是按传统习惯，同步，通信一并处理吧，理解成同一个玩意

<https://cloud.tencent.com/developer/article/1129585>

<https://www.nowcoder.com/discuss/374134?type=post&order=time&pos=&page=2>

**进程同步方式：**

<https://www.cnblogs.com/MrListening/p/5858358.html>

<https://www.jianshu.com/p/c1015f5ffa74>

**管道**，只局限与父子进程。

**信号**，进程间传递信号，捕获到信号后执行对应绑定的代码，和QT的信号槽类似。可以实现进程通信的“单播”、“广播”。

**信号量**，信号量本身无法传递数据，配合共享内存使用，类似于线程中的锁，用于保护临界资源。

**共享内存**，进程间最常用的数据同步方式。与信号量配合使用。

**消息队列**，也是非常常见的同步方式，把数据放入队列，内核逐一处理发送至目的线程。

**线程同步方式：**

**信号**，使用方法和进程几乎一样，但是是另一套相似的API，不可以互换。

**信号量**，和进程类似，功能和互斥锁基本一样。

**互斥锁**，保护临界资源。

**控制变量**，常和互斥锁配合使用，控制线程执行的先后。暂时挂起线程还锁，解决线程为获得数据等待其他线程，导致长时间占用锁

操作系统中的进程调度策略有哪几种

<https://blog.csdn.net/qq_35642036/article/details/82809812?depth_1-utm_source=distribute.pc_relevant.none-task&utm_source=distribute.pc_relevant.none-task>