# 进程与线程的区别：

进程是一个具有独立功能的程序关于某个数据集合的一次运行活动。它可以申请和拥有系统资源，是一个动态的概念，是一个活动的实体。它不只是程序的代码，还包括当前的活动，通过程序计数器的值和处理寄存器的内容来表示。

进程是一个“执行中的程序”。程序是一个没有生命的实体，只有处理器赋予程序生命时，它才能成为一个活动的实体，我们称其为进程。

通常在一个进程中可以包含若干个线程，它们可以利用进程所拥有的资源。在引入线程的操作系统中，通常都是把进程作为分配资源的基本单位，而把线程作为独立运行和独立调度的基本单位。由于线程比进程更小，基本上不拥有系统资源，故对它的调度所付出的开销就会小得多，能更高效的提高系统内多个程序间并发执行的程度。

线程和进程的区别在于，子进程和父进程有不同的代码和数据空间，而多个线程则共享数据空间，每个线程有自己的执行堆栈和程序计数器为其执行上下文。多线程主要是为了节约CPU时间，发挥利用，根据具体情况而定。线程的运行中需要使用计算机的内存资源和CPU。

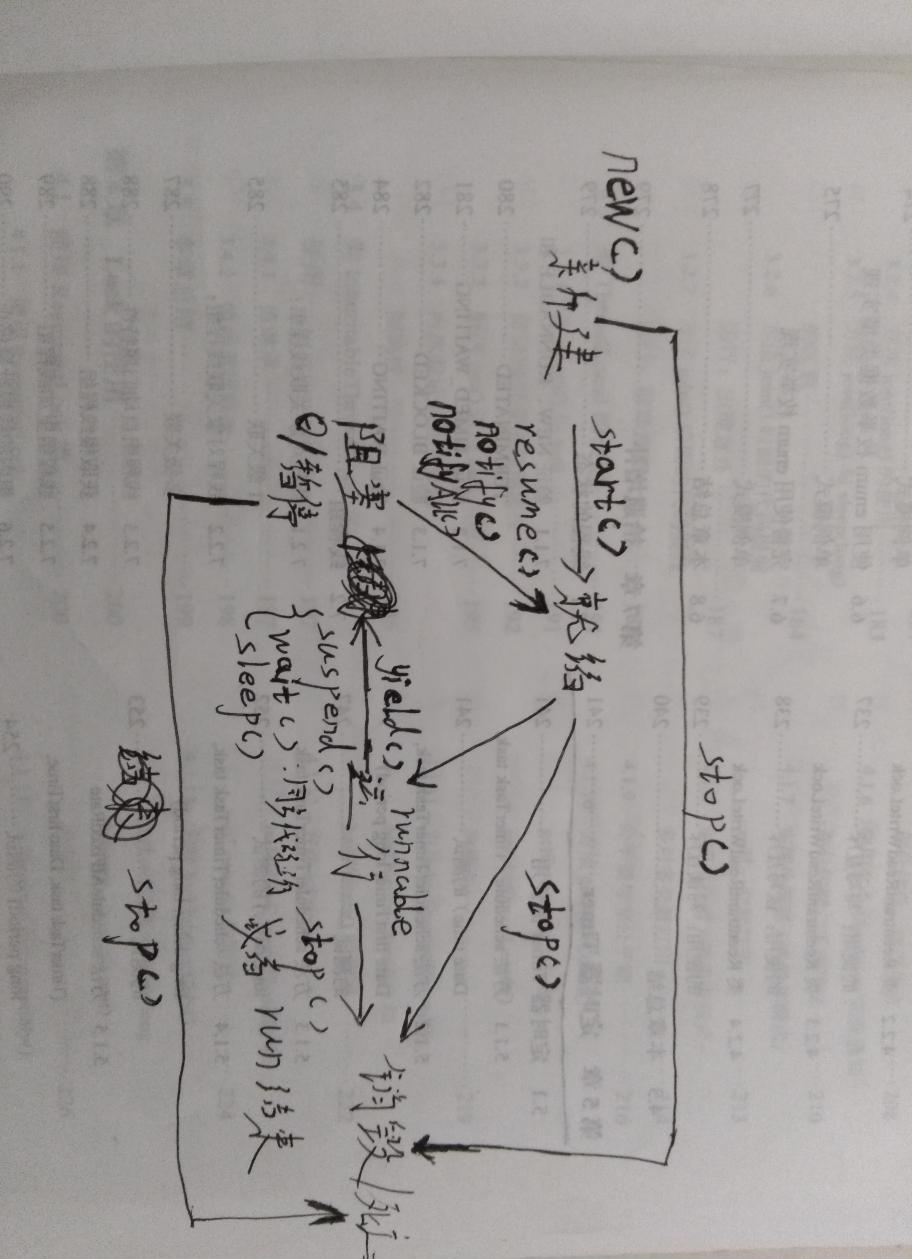
**线程与进程的区别归纳：**

**a.地址空间和其它资源**：进程间相互独立，同一进程的各线程间共享。某进程内的线程在其它进程不可见。

**b.通信：**进程间通信IPC，线程间可以直接读写 进程 数据段（如全局变量）来进行通信——需要进程同步和互斥手段的辅助，以保证数据的一致性。

**c.调度和切换**：线程上下文切换比进程上下文切换要快得多。

d.在多线程OS中，进程不是一个可执行的实体。



作业：

用户在一次运算过程中，或一次事务处理中要求计算机所做的全部工作的总和。通常包括程序（静态），进程（动态），数据，控制命令，有序操作等。