单道批处理系统中，一个作业单位进入内存并独占系统资源，知道运行结束后下一个作业才能进入内存。当作业进行IO操作时，CPU只能处于等待状态。CPU利用率低下。

多道批处理系统，在内存中可以同时存在若干个作业，作业执行顺序与进入内存的次序没有严格对应的关系。一个作业在等待IO处理时，CPU调度另外一个作业运行。

因此CPU的利用率显著提高

分时系统-般进程调度算法是时间片轮转算法。多个终端用户可以通过彼此的终端相互独立地使用计算机，计算机按照时间片轮转算法，给每一个用户进程分配合适的时间片，处理用户请求。

常用排序算法的性能与待排数组的循序的关系作一个总结：  
  
1）冒泡：无关；  
2）选择：无关；  
3）插入：有关，有序程度越大，比较越少；  
4）shell：有关，它的基本思想基于插入排序；  
5）合并：有关，有序程度愈大，合并过程的比较次数越少；  
6）堆排序：有关，有序程度越大，建立堆下沉操作越少；  
7）快排序：有关，如果选择最后值作为阀值，那么有序程度越好，就越可能退化成O(n^2)；  
  无关，随机选择阀值，那么与排序程度无关。

对称矩阵压缩后存储到一维数组中，假设存储下三角的元素，那么存储的元素个数为：第一行1个、第二行2个、第三行3个。。。第10行10个，一共1+2+3+...+10=55个元素（即n(n+1)/2）

线程间共享的资源  
1.文件描述符表  
2.每种信号的处理方式  
3.当前工作目录  
4.用户ID和组ID

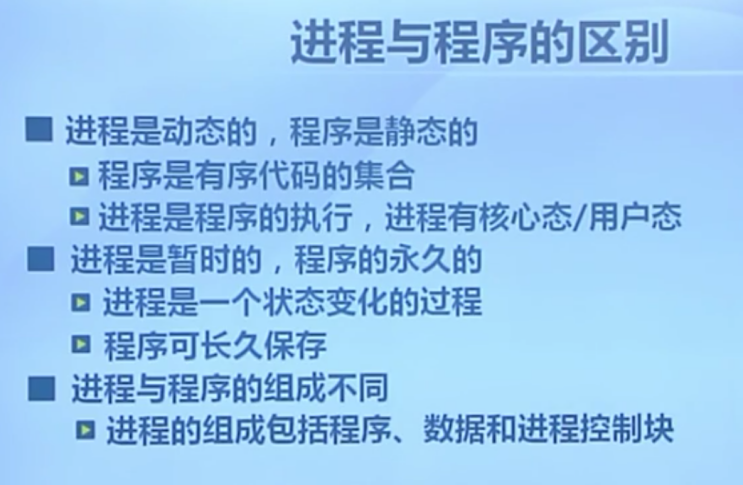
5.内存地址空间

线程间非共享资源  
1.线程id  
2.处理器现场和栈指针(内核栈)  
3.独立的栈空间(用户空间栈)  
4.errno变量  
5.信号屏蔽字  
6.调度优先级

设文件F1的当前引用计数值为1，先建立F1的符号链接（软链接）文件F2，再建立F1的硬链接文件F3，然后删除F1。此时，F2和F3的引用计数值分别是（）。

建立符号链接时，引用计数值直接复制；建立硬链接时，引用计数值加1。删除文件时，删除操作对于符号链接是不可见的，这并不影响文件系统，当以后再通过符号链接访问时，发现文件不存在，直接删除符号链接；但对于硬链接则不可以直接删除，引用计数值减1，若值不为0，则不能删除此文件，因为还有其他硬链接指向此文件。

当建立F2时，F1和F2的引用计数值都为1。当再建立F3时，F1和F3的引用计数值就都变成了2。当后来删除F1时，F3的引用计数值为2-1=1，F2的引用计数值一直不变。



线程的同步主要是指执行的有先有后这样的顺便，比如一个线程的执行依赖另一个线程的某种消息或者条件，当他没有得到这个消息的时候应该等待，直到消息到达才被唤醒

线程的互斥主要是指执行中的线程对共享的进程系统资源的排它性，当有多个线程都要使用某一个共享资源时，任何时刻最多只允许一个线程去使用，其他使用该资源的线程必须等待，直到占用资源者释放资源