1：先来先服务（fcfs）

基本思想：按进程（作业）进入就绪（后备）队列的先后次序来分配处

理机（为其创建进程）。  
一般采用非剥夺的调度方式。

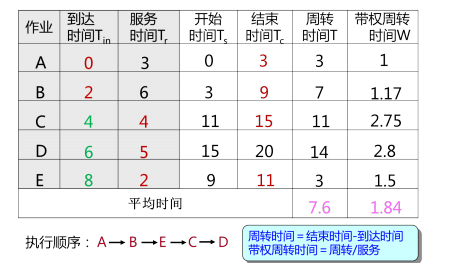
例子



FCFS 调度算法的平均作业周转时间与作业提交的顺序有关。  
FCFS 调度算法的特点：  
简单，但效率不高。  
有利于 CPU 繁忙型作业。  
不利于 I/O 繁忙型作业。  
现在操作系统中，已很少用该算法作为主要调度策略，尤其是在分时系统  
和实时系统中。但它常被结合在其它调度策略中使用。

2：短作业优先（sjf）

用于作业调度  
主要任务是从后备队列中选择一个或若干个估计运行时间最短的作业，将  
它们调入内存运行。  
类似地，用于进程调度的是短进程优先调



优点  
能有效降低作业的平均等待时间。  
能有效缩短进程的平均周转时间。  
提高了吞吐量。  
缺点  
对长作业不利。  
没有考虑作业的紧迫程度。  
作业执行时间、剩余时间仅为估计。

SJF 算法虽然是优化的，但在 CPU 调度中很难实现。

3：优先级调度算法（psa）

优先级调度算法(Priority-Scheduling Algorithm, PSA) 以作业的紧迫程  
度为优先级。  
系统选择优先级最高的几个作业装入内存。  
优先级调度算法也用于进程调度，系统在可运行的进程中选择优先级最  
高者使其投入运行。  
优先级的类型  
   静态优先级  
   动态优先级

静态优先级

　　优先权在创建进程时确定，且在进程的整个运行期间保持不变。一般用  
　　整数表示，小表示优先级高。  
　　确定原则：  
　　进程类型（系统进程 > 用户进程）  
　　进程对资源的需求（要求少的有较高的优先权）  
　　用户要求（紧急程度和付费情况）  
　　优点：简单，开销小。  
　　缺点：公平性差（对低优先权进程）

动态优先级

　　动态优先级在进程的存在过程中不断发生变化。  
　　动态优先级的变化取决于：  
　　进程的等待时间  
　　进程的运行时间  
　　进程使用资源的情况  
　　动态优先权确定方法的资源利用率高，公平性好，但开销较大，实现较  
　　为复杂。  
　　高响应比优先算法 (HRRN)采用动态优先权。

4：高响应比优先调度算法（hrrn）

FCFS 只考虑了作业的等待时间，忽略了运行时间。SJF 只考虑了作业  
的运行时间，忽略了等待时间。  
高响应比优先调度算法(Highest Response Ratio Next, HRRN) 既考虑了  
作业的等待时间，也考虑了作业的运行时间，是一种动态优先级调度算  
法。  
优先权 =  
（等待时间 + 要求服务时间）/  
   要求服务时间  
响应比RP =  
(等待时间 + 要求服务时间)/要求服务时间 =响应时间/要求服务时间  
周转时间：从提交到完成的时间间隔。响应时间：在交互式系统中，从提交请求到产生首次响应的时间，而不是  
到产生输出结果所需的时间

根据优先权公式可知

如等待时间相同，则要求服务时间越短其优先权越高 →SJF。  
如要求服务时间相同，优先权决定于等待时间 →FCFS。  
对长作业，若等待时间足够长，优先权也高，也能获得 CPU

