# 处理机调度典型算法比较

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **算法** | **思想** | **规则** | **可抢占？** | **优点** | **缺点** | **考虑到等待时间&运行时间？** | **会导致饥饿？** | **补充** |
| **FCFS** | 主要从“公平”角度考虑 | 按照作业/进程的先后顺序进行服务 | 非抢占式 | 公平；实现简单 | 对短作业不利 | 等待时间√  运行时间× | 不会 | —— |
| **SJF/SPF** | 追求最少的平均等待时间、最少的平均周转时间、最少的平均带权周转时间 | 最短的作业/进程优先得到服务（“最短”是指要求服务时间最短） | 默认为非抢占式，也有SJF的抢占式版本最短剩余时间优先算法（STR） | “最短的”平均等待/周转时间 | 对长作业不利，可能导致饥饿；难以做到真正的短作业优先 | 等待时间×  运行时间√ | 会 | —— |
| **HRRN** | 要综合考虑作业/进程的等待时间和要求服务的时间 | 在每次调度时先计算各个作业/进程的响应比，选择响应比最高的作业/进程为其服务 | 非抢占式 | 上述两种算法的权衡折中，综合考虑的等待时间和运行时间 | —— | 等待时间√  运行时间√ | 不会 | —— |
| **时间片轮转** | 公平地/轮流地为各个进程服务，让每个进程在一定时间间隔内都可以得到响应 | 按照各进程到达就绪队列地顺序，轮流让各个进程执行一个时间片。若进程未在一个时间片内执行完，则剥夺处理机，将进程重新放到就绪队列队尾重新排队 | 抢占式 | 公平，适用于分时系统 | 频繁切换有开销，不区分优先级 | —— | 不会 | 时间片太大或太小有影响 |
| **优先级调度** | 随着计算机地发展，特别是实时操作系统的出现，越来越多的应用场景需要根据任务的紧急程度来决定处理顺序 | 调度时选择优先级最高的作业/进程 | 有抢占式的，也有非抢占式的 | 区分优先级，适用于实时系统 | 可能导致饥饿 | —— | 会 | 动态/静态优先级 |
| **多级反馈队列** | 对其他调度算法的折中平衡 | 较复杂 | 抢占式 | 平衡优秀 | 一般不说它有缺点，不过可能导致饥饿 | —— | 会 | —— |