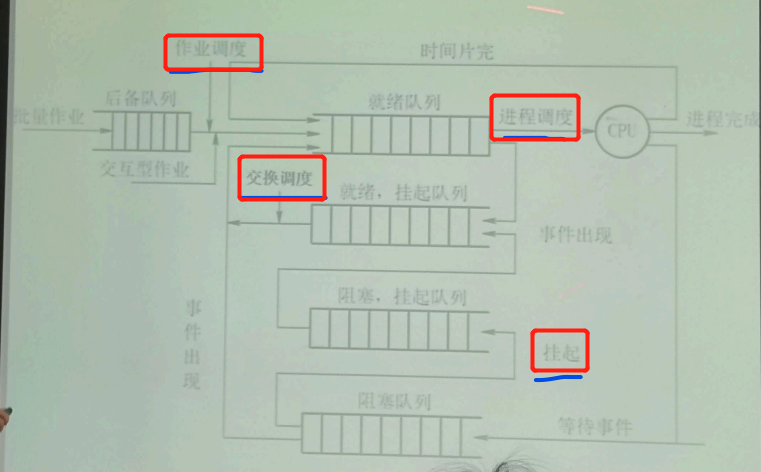
第六章 CPU调度

## 6.1 基本概念

### 6.1.1 多级调度

**1、**

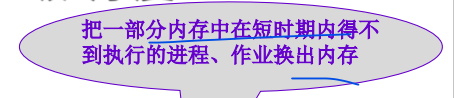
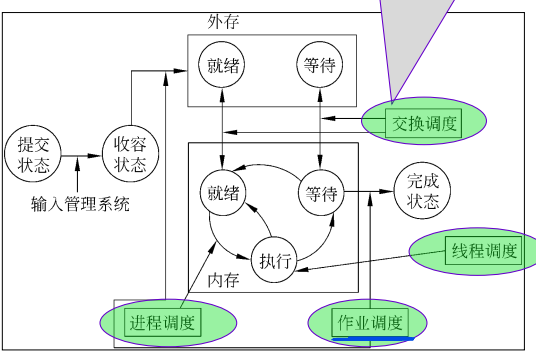


作业调度：根据一定的算法选择一个作业进入内存，叫做~

进程调度：根据一定的算法选择一个进程执行，叫做~

挂起：为了提高资源利用率，把一部分内存中在短时间内得不到执行的进程换出内存，其状态变为~

交换调度：根据一定的算法选择一个进程将其调回内存叫做~



四级调度关系

**2、调度层次**

作业调度：将外存作业调入内存，创建PCB和插入就绪队列

交换调度：内外存对换功能，提高系统吞吐量和内存利用率

进程调度：从就绪队列中选择某一个进程占用CPU

线程调度：与进程调度的算法类似，实际调度的是内核级线程

### 6.1.2 CPU调度的时机（简答题会考）★★★★★

**1、四种时机**

进程中止

进程从运行到等待

进程从运行到就绪状态（时间片用完）

进程从等待到就绪（优先级剥夺方式）

**2、非抢占调度**

一旦进程分配到CPU，就会一直占用，知道它中止或切换到等待态

### 6.1.3 调度程序

**1、概念：**

挑选就绪进程的**内核函数**

**2、将CPU的控制权转交给选中的进程：**

① 切换上下文

② 切换到用户态

③ 跳转到用户程序的适当位置并恢复运行

## 6.2 调度准则

**1、哪几方面**

① **公平性**：“合理”的分配CPU（每个进程的等待时间相同）

② CPU利用率高：CPU处于忙状态的时间百分比

③ **吞吐量大**：单位时间完成的任务数量，吞吐量达=CPU使用效率高+上下文切换代价小

④ **响应时间短**（交互式）：从用户输入到产生反应的总时间

⑤ 周转时间短（批处理）：从进程初始化到进程结束的总时间

⑥ 等待时间短：进程在就绪队列中的总时间

**2、调度准则之间的矛盾**

① 公平性和响应时间：

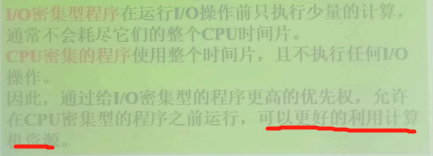
响应时间短->前台任务的优先级高->后台任务得不到CPU->不公平

② 吞吐量和响应时间：

吞吐量大->上下文切换代价小->时间片大->响应时间长

**3、简答题**★★★★★





## 6.3 调度算法

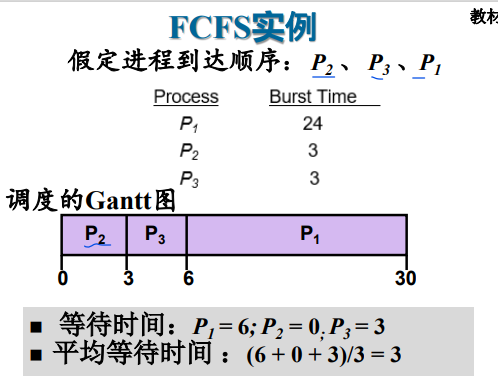
**1、先来先服务（FCFS）适用于作业、进程**

① 先调度先到达的（作业、进程），任务占用CPU后，直到**执行完**或**阻塞**才出让

② 特点：

非抢占、适合长作业、平均等待时间长（波动大）、很少单独使用，经常和其他算法配合使用

③ 实例



**2、短作业优先（作业）**

① 优先调度占用CPU时间最短的任务

② 分类：

非抢占式调度(**SJF**)：一旦进程拥有CPU，一直占用

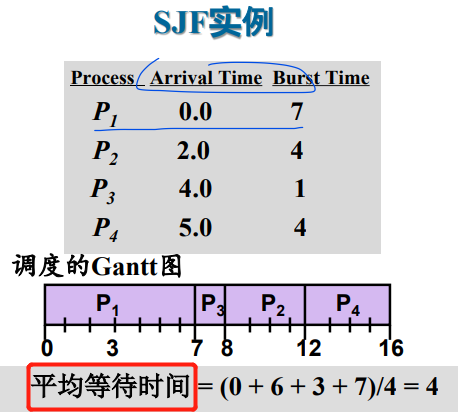
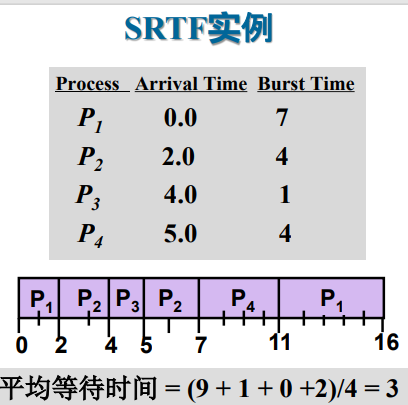
抢占式调度（**SRTF**）：最短剩余时间优先调度（SRTF），当比当前任务剩余时间片更短的任务到达时，抢占CPU

③ 优点：保证了最小的平均等待时间

④ 缺点：可能导致**饥饿**

④ 困难：下一CPU区间大小的预测

⑤ 实例：

**4、优先级（作业非剥夺、进程剥夺）**

① 每个任务（作业、进程）关联一个优先权

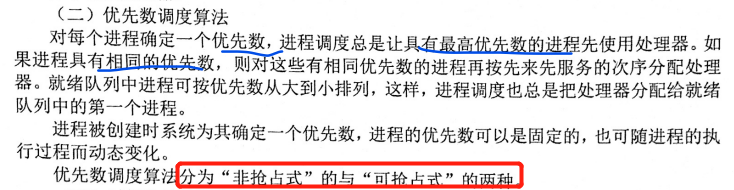
② 优先级：分为内部定义的、外部定义的

③ SJF是优先级调度的特例

④ 缺点：导致**饥饿**

具体体现在：优先级的任务可能永远得不到运行

解决办法：任务随着等待时间的增长提高其优先数（老化Aging）



**5、时间片轮转（RR）进程**

① 例子：

**分时系统**

② 按照时间片来轮转调度：

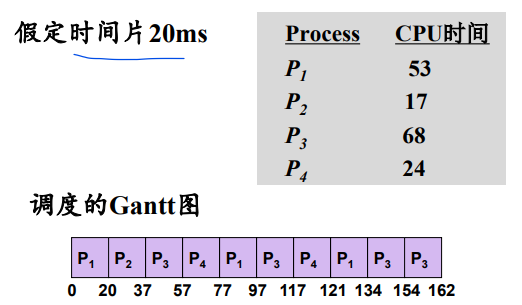
A、把CPU划分成若干时间片；

B、所有就绪进程按照FCFS，排成一个队列

C、把CPU分派给队首进程，执行一个时间片

D、如果时间片够用，立马执行下一个进程；如果不够用，时间片结束时，暂停当前进程的执行，送到**就绪队列**的尾部，通过上下文切换执行当前的队首进程。

③ 实例



时间片越小，增加上下文切换次数

④ 优点和缺点：

优点：定时有响应（每个进程得到固定的时间片），等待时间短（任何进程的等待时间不会超过(n-1)q单位）

缺点：上下文切换次数较多

⑤ 时间片大小的影响：

太大：导致响应时间长（如**时间片500ms × 10进程**，响应需要5秒），退化为FCFS

太小：上下文切换次数过多，开销大

最佳：分时用户得到好的响应时间

折衷：时间片10-100ms，切换时间0.1-1ms（1%的切换代价）

**6、多级队列（进程）**

① 多个就绪队列，不同的队列采用不同的调度方法，如前后台两个队列

② 队列内调度：

前台（交互式）：RR

后台（批处理）：FCFS

③ 队列间调度：

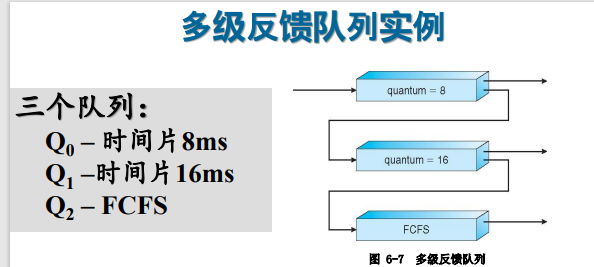
固定优先级调度：前台运行完再运行后台，可能产生饥饿

给定时间片调度：每个队列得到一定的CPU时间，任务在给定时间内执行

**7、多级反馈队列（进程）**

**使用最多**

① 任务可以在队列之间移动，可实现老化。



② 调度程序参数

队列数；

每一队列的调度算法

决定任务升级、降级的方法

决定进程将进入那个队列的方法

**8、高响应比优先（作业）HRNs**

①计算响应比

响应比R = 作业周转时间（执行时间+等待时间）/ 作业执行时间 = 1 + 等待时间 / 执行时间

② 特点：

兼顾到长作业和短作业

是FCFS和SJF的折中，每次调度前要计算响应比，系统开销增加

