西安交通大学
软件学院

操作系统原理

Operating System Principle

田丽华

5-4 优先级和RR时间片轮转

Priority Scheduling

- A priority number (integer) is associated with each process（每个进程都有自己的优先数[整数]）
- The CPU is allocated to the process with the highest priority (smallest integer ≡ highest priority)（CPU分配给最高优先级的进程[假定最小的整数 ≡ 最高的优先级]）.

1. Preemptive（抢占式）
2. Nonpreemptive （非抢占式）

- SJF is a priority scheduling where priority is the predicted next CPU burst time（SJF是以下一次CPU脉冲长度为优先数的优先级调度）.

Priority Scheduling

进程	运行时间	优先权
P1	10	3
P2	1	1
P3	2	4
P4	1	5
P5	5	2

- 调度顺序：P2、P5、P1、P3、P4
- 平均等待时间8.2ms

Priority Scheduling

1. 静态优先权在进程创建时确定，且在整个生命期中保持不变。

2. 静态优先权的问题 Problem ≡ Starvation – low priority processes may never execute

(问题 ≡ 饥饿 – 低优先级的可能永远得不到运行) .

一个很有意思的例子：当MIT的IBM7094机器于1973年关掉时，人们发现一个于1967年提交的一个低优先权的进程还没有得到运行。

Solution ≡ Aging – as time progresses increase the priority of the process

(解决方法 ≡ 老化 – 视进程等待时间的延长提高其优先数) .

Priority Scheduling

- 动态优先权是指进程的优先权可以随进程的推进而改变，以便获得更好的调度性能
- 改变优先权的因素

进程的等待时间

已使用处理机的时间

资源使用情况

Round Robin (RR)

Each process gets a small unit of CPU time *(time quantum),* usually 10-100 milliseconds. After this time has elapsed, the process is preempted and added to the end of the ready queue

（每个进程将得到小单位的CPU时间[时间片]，通常为10-100毫 秒。

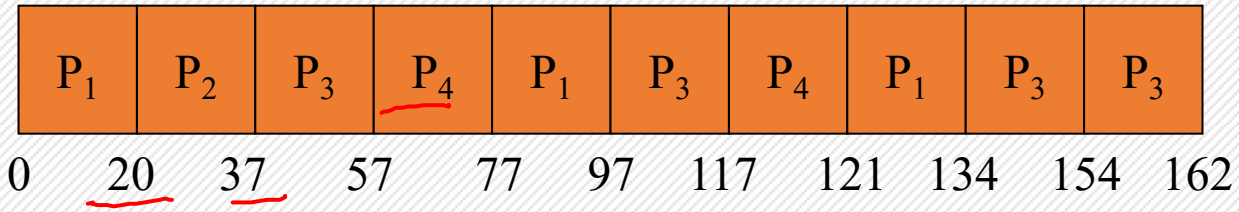
时间片用完后，该进程将被抢占并插入就绪队列末尾）

Round Robin (RR)

Example: RR with Time Quantum = 20

Process	Burst Time
P_1	53
P_2	17
P_3	68
P_4	24

01 The Gantt chart is:



02 Typically, higher average turnaround than SJF, but better response（一般来说，RR的平均周转时间比SJF长，但响应时间要短一些）.

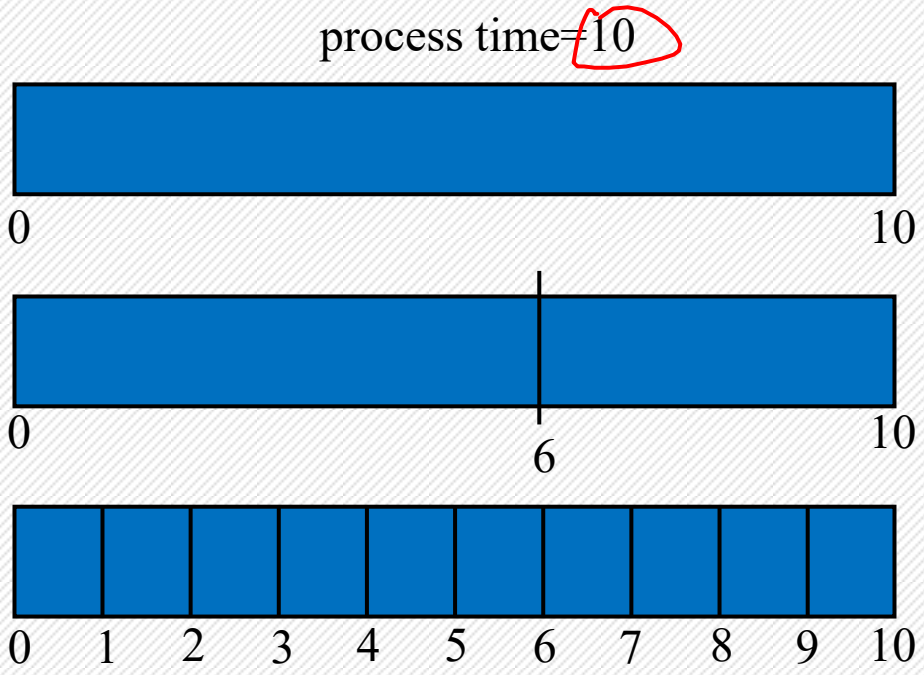
Round Robin (RR)

Performance（特性）

1. *q* large ⇒ FCFS
2. *q* small ⇒ q must be large with respect to context switch, otherwise overhead is too high（*q*相对于切换上下文的时间而言足够长，否则将导致系统开销过大）.

Round Robin (RR)

How a Smaller Time Quantum Increases Context Switches

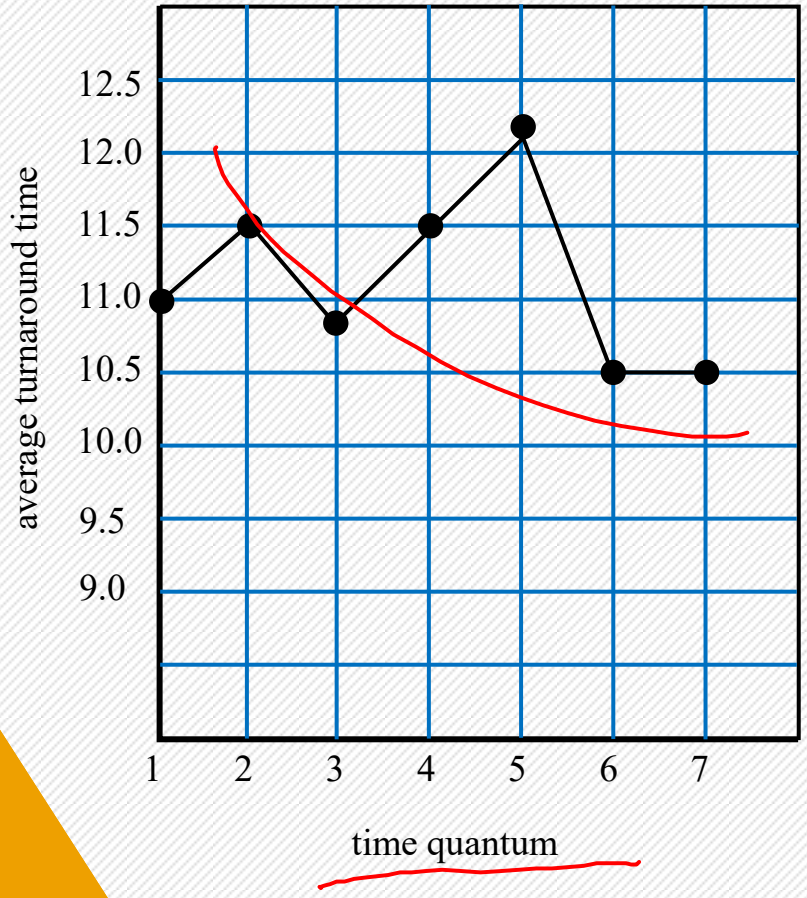


quantum	context switches
12	0
6	1
1	9

Longer quantum yields shorter average turnaround times?

Round Robin (RR)

Turnaround Time Varies With The Time Quantum



process	time
P_1	6
P_2	3
P_3	1
P_4	7

一组进程的平均周转时间并不一定随着时间片的增大而降低。一般来说，如果大多数（80%）进程能在一个时间片内完成，就会改善平均周转时间