CIS 452 - Operating Systems Concepts

Nathan Bowman

Images taken from Silberschatz book

Scheduling Algorithms -- Round Robin

Round robin (RR) scheduling is designed for time-sharing systems

Unlike priority scheduling, all processes are treated equally

Rather than letting a process run fully, processes are assigned a fixed amount of time on the CPU

Time slice (or **time quantum**) is small, fixed unit of time

Generally 10 - 100 milliseconds -- we will discuss a bit more later

Ready queue is a circular FIFO queue

Scheduler starts at head of queue and

- assigns process to CPU for up to 1 time quantum
- sets a timer

When timer expires, second element in queue becomes head and the steps are repeated

New processes are added to tail of queue

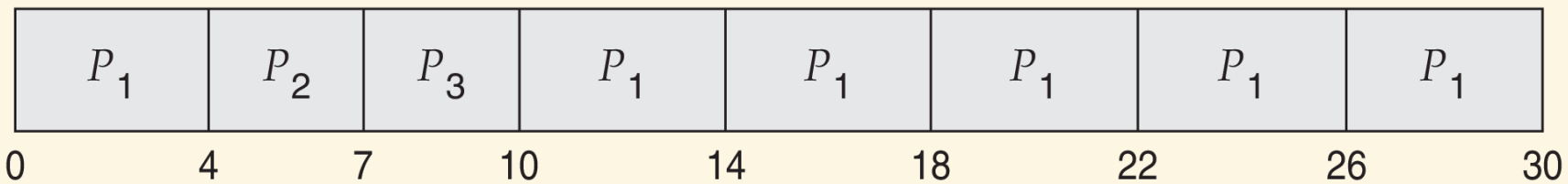
It is possible that a process will run for less than one time quantum if it exits or makes a blocking call

In that case, the process is removed from the ready queue and the rest of the scheduling algorithm proceeds normally

You need to know the time quantum to simulate RR scheduling -- assume a quantum of 4 ms

Process	Burst Time
P1	24
P2	3
P3	3

(times are in milliseconds)


P_1
P_2
P_3
P_1
P_1
P_1
P_1
P_1
0
4
7
10
14
18
22
26
30

RR scheduling is fair, but it does not minimize average waiting time

RR average waiting time: 5.66 ms

SJF average waiting time: 3 ms

Note that RR scheduling must be preemptive. Why?

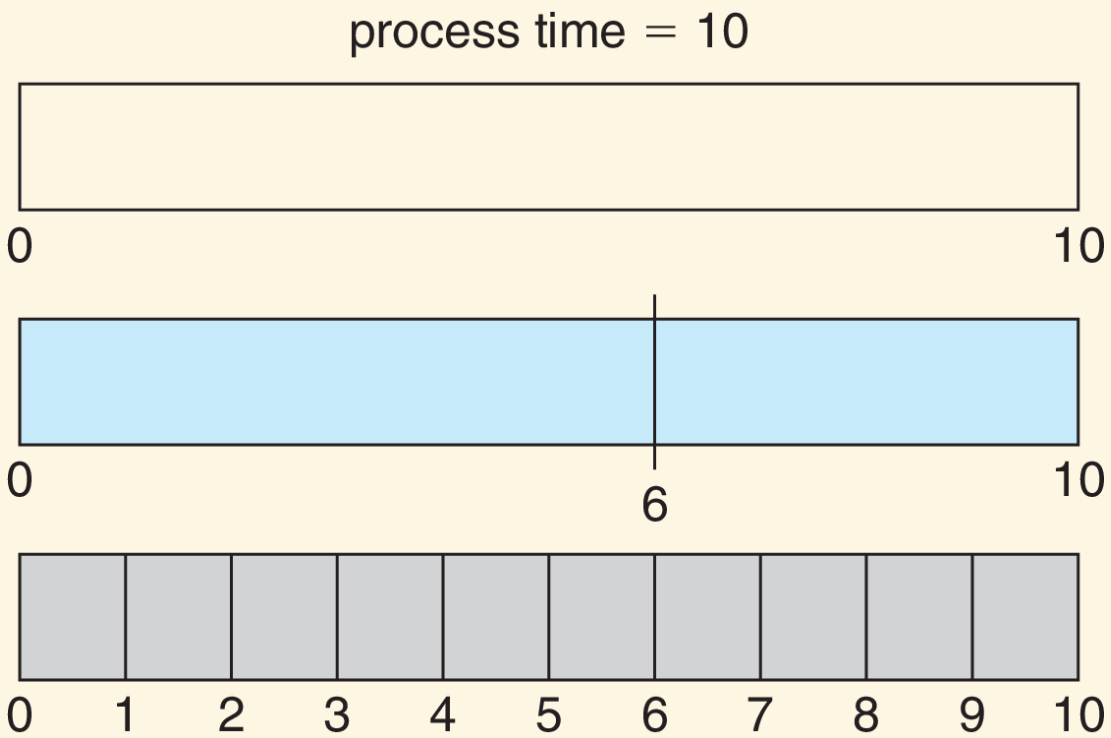
With n processes and a time quantum of q, no process must wait for more than $(n - 1)q$ time units

Each process gets $1/n$ of total CPU time

Choice of time quantum is very important

Extremely large time quantum makes RR the same as FCFS

Small time quantum results in excessive context switches



quantum	context switches
12	0
6	1
1	9

Want time quantum to be large relative to context switch time

Context switch time generally below 10 microseconds

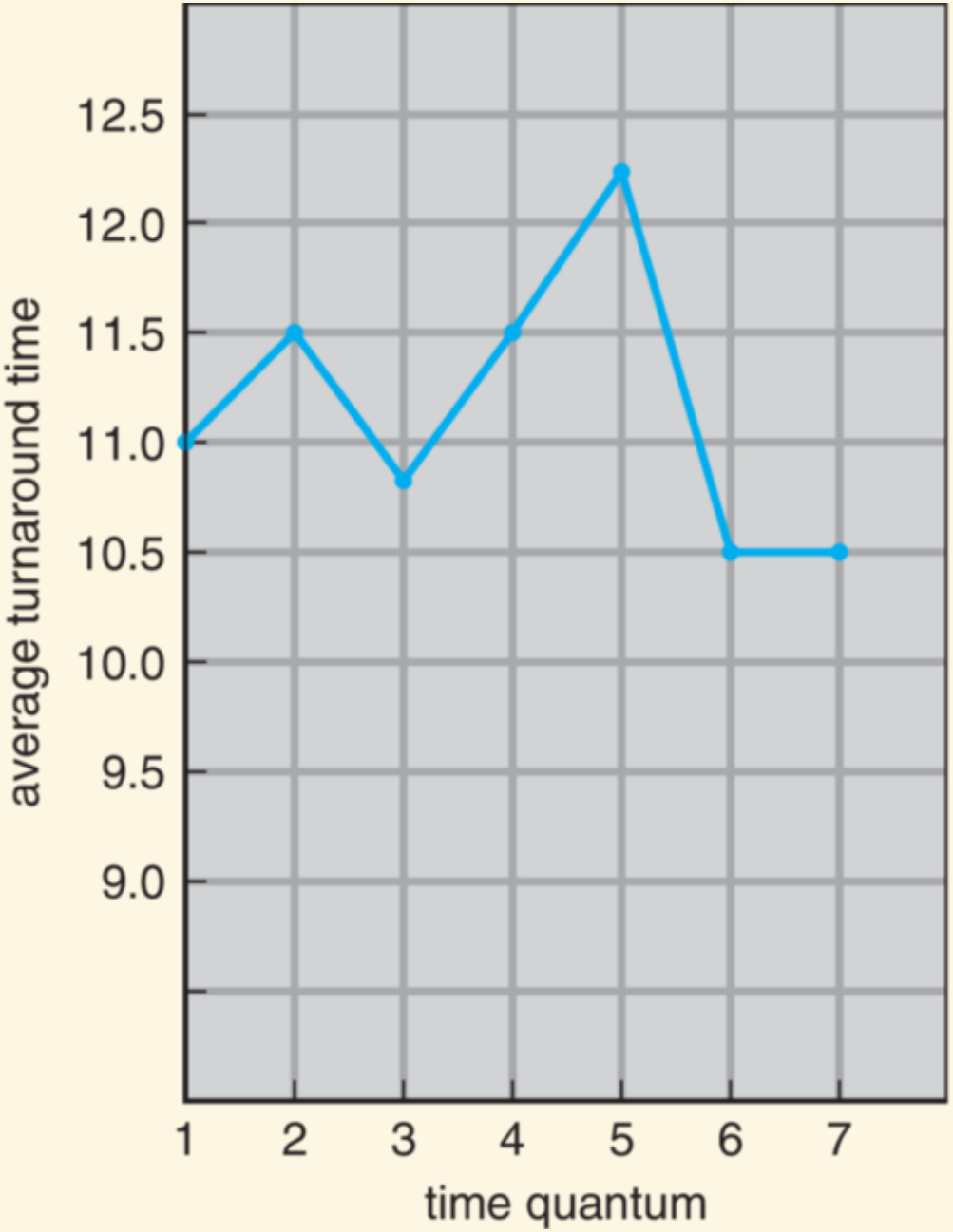
If time quantum is, for example, 10 ms, context switch time will be small fraction of overall time

Quantum choice also affects turnaround time

In general, turnaround time will improve in RR scheduling if most processes finish their CPU burst within one time quantum

A smaller time quantum also means more context switches

However, improvement is not monotonic



process	time
P_1	6
P_2	3
P_3	1
P_4	7

Rule of thumb: 80% of CPU bursts should be shorter than time quantum