University of Pittsburgh

Introduction to Operating Systems
CS 1550

Spring 2023

Sherif Khattab

ksm73@pitt.edu

(Some slides are from **Silberschatz, Galvin and Gagne ©2013)**

Announcements

- Upcoming deadlines
  - Homework 7 is due **this Friday**
  - Quiz 1 and Lab 2 due on Tuesday 2/28 at 11:59 pm
  - Project 2 is due Friday 3/17 at 11:59 pm
- Midterm exam on Thursday 3/2
  - In-person, on paper, closed book
  - Study guide, old exam, and practice Midterm on Canvas
- Midterm Review Session tomorrow 3/1 at 5:30 pm
  - Recorded
  - Same Zoom link as Student Support Hours

Previous lecture …

- How to implement Condition Variables and Locks using semaphores
- CPU scheduling
  - FCFS

Problem of the Day: CPU Scheduling

How does the ***short-term scheduler*** select the next process to run?

Shortest Job First (Shortest Process Next)

- Selection function: the process with the shortest expected CPU burst time
  - I/O-bound processes will be selected first
- Decision mode: non-preemptive
- The required processing time, i.e., the CPU burst time, must be estimated for each process

Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2

SJF / SPN Critique

- Possibility of starvation for longer processes
- Lack of preemption is not suitable in a time sharing environment
- SJF/SPN implicitly incorporates priorities
  - Shortest jobs are given preference
  - CPU bound processes have lower priority, but a process doing no I/O could still monopolize the CPU if it is the first to enter the system

Priorities

- Implemented by having multiple ready queues to represent each level of priority
- Scheduler selects the process of a higher priority over one of lower priority
- Lower-priority may suffer starvation
- To alleviate starvation allow dynamic priorities
  - The priority of a process changes based on its age or execution history

Round-Robin

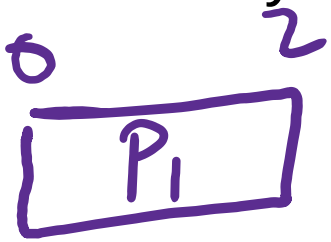
- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2

Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

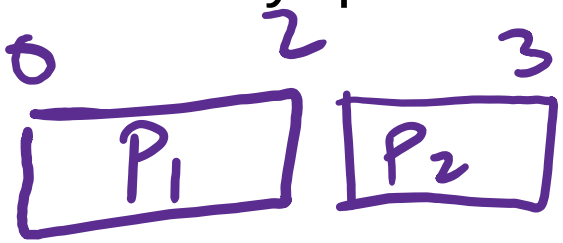
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

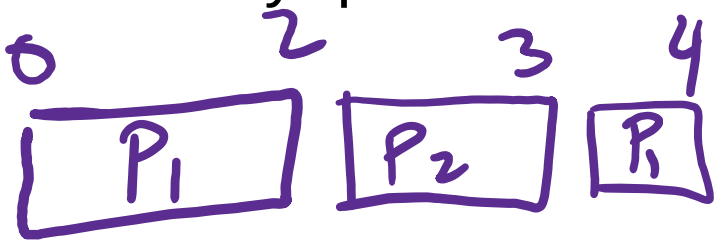
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

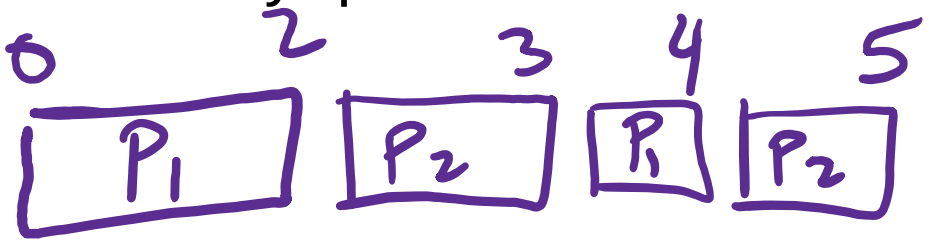
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

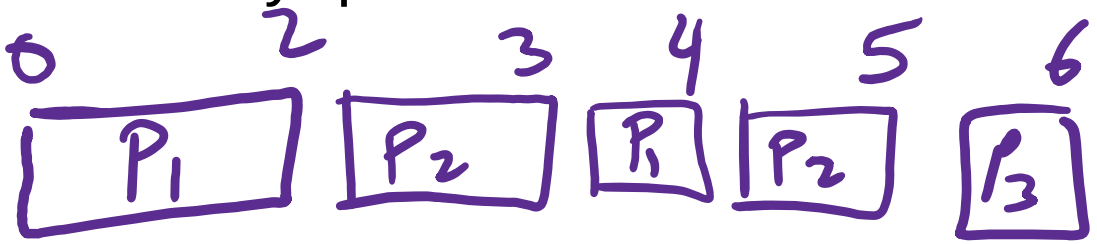
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

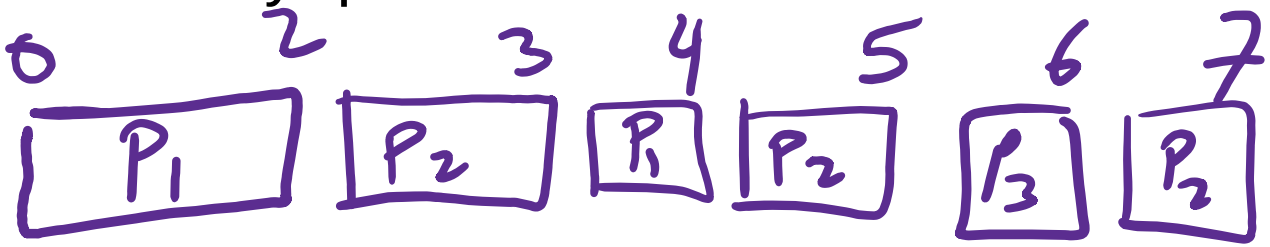
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

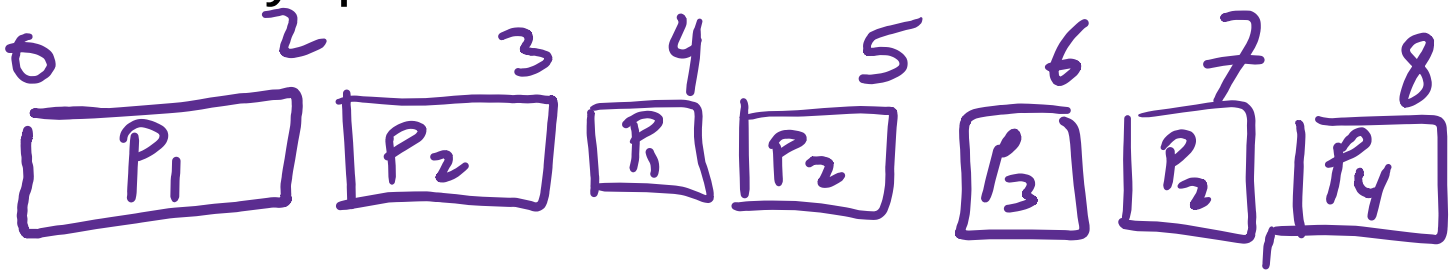
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

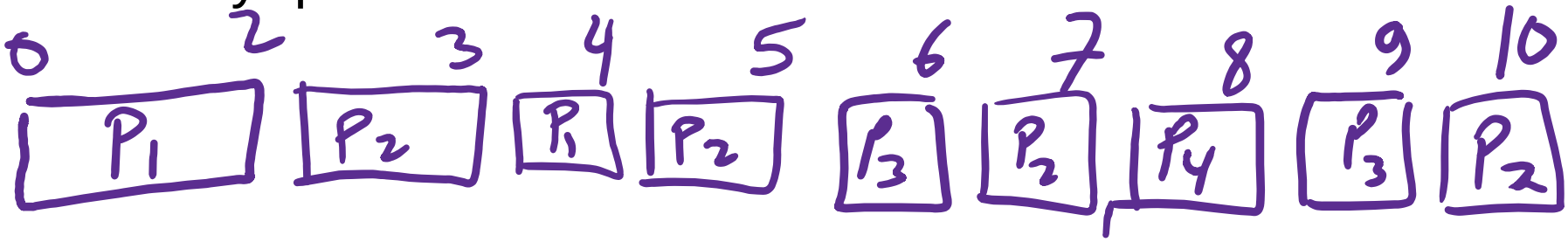
- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2

Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

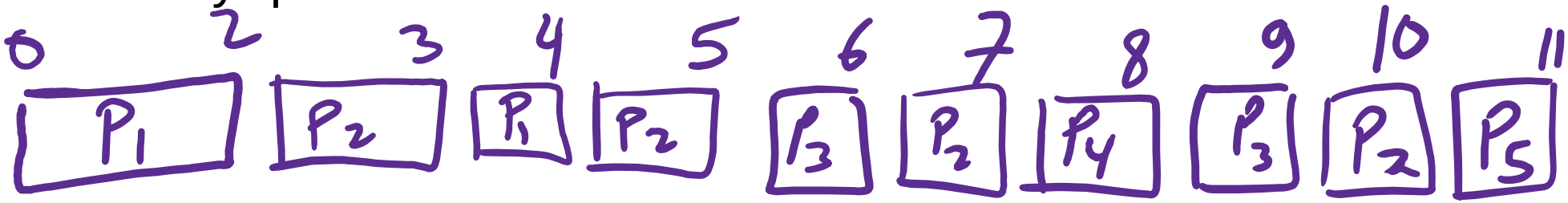
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

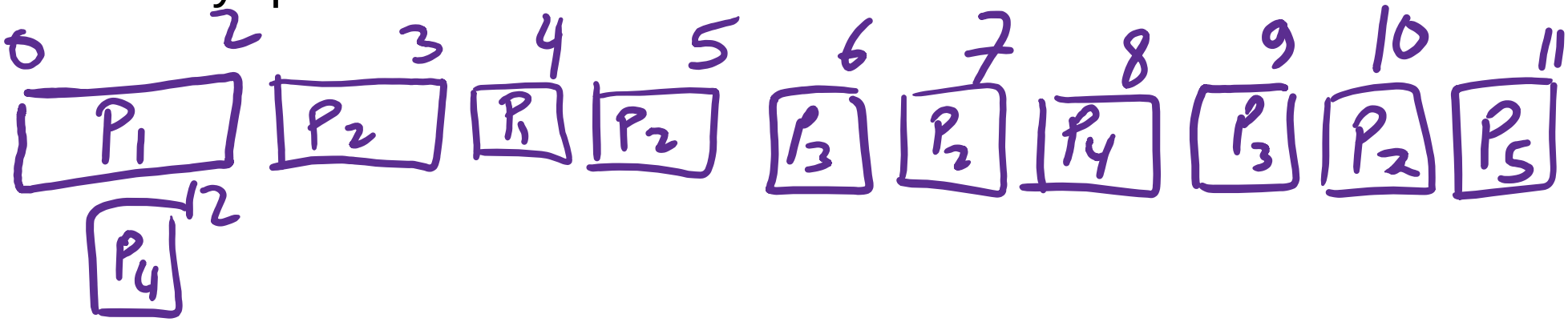
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

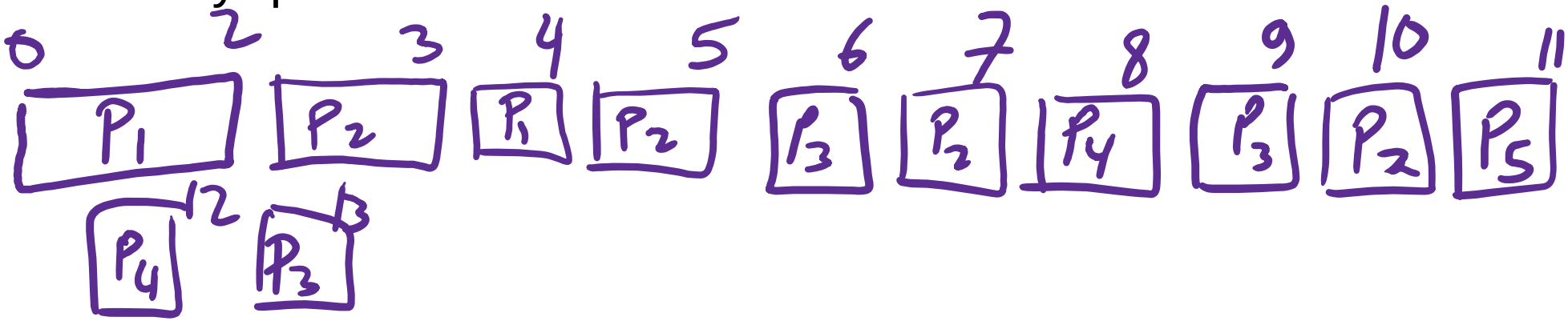
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

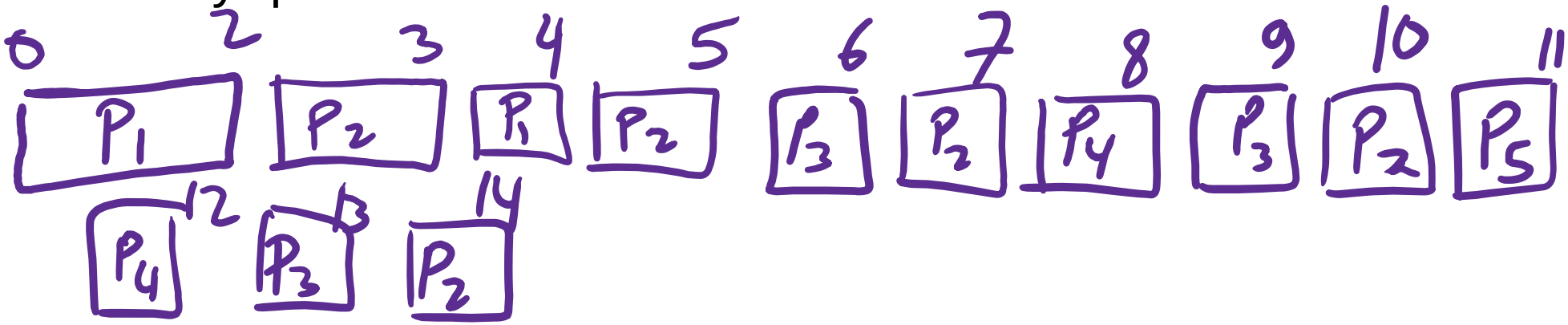
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

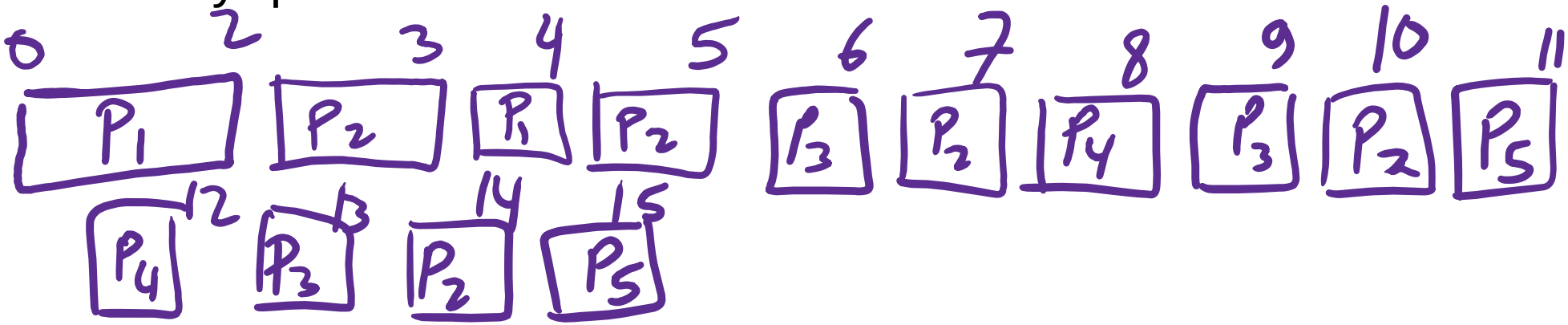
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

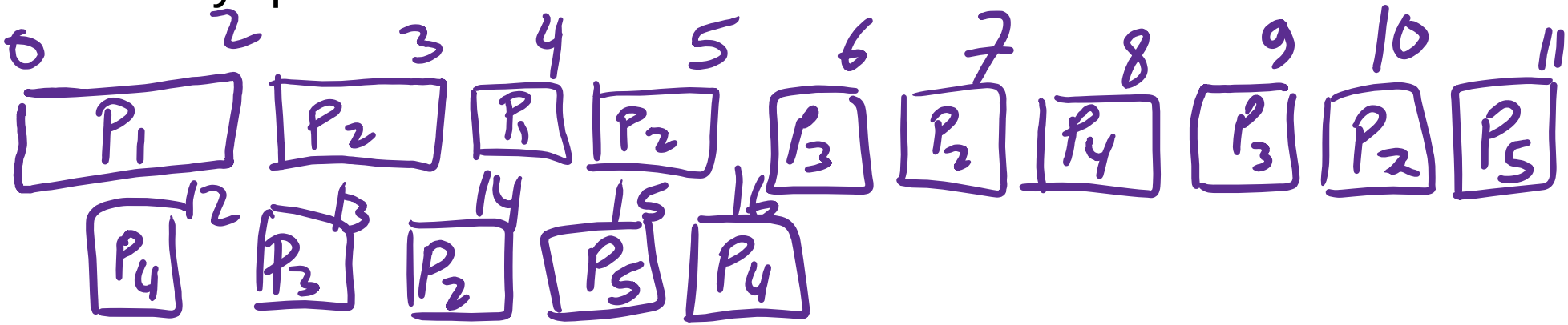
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

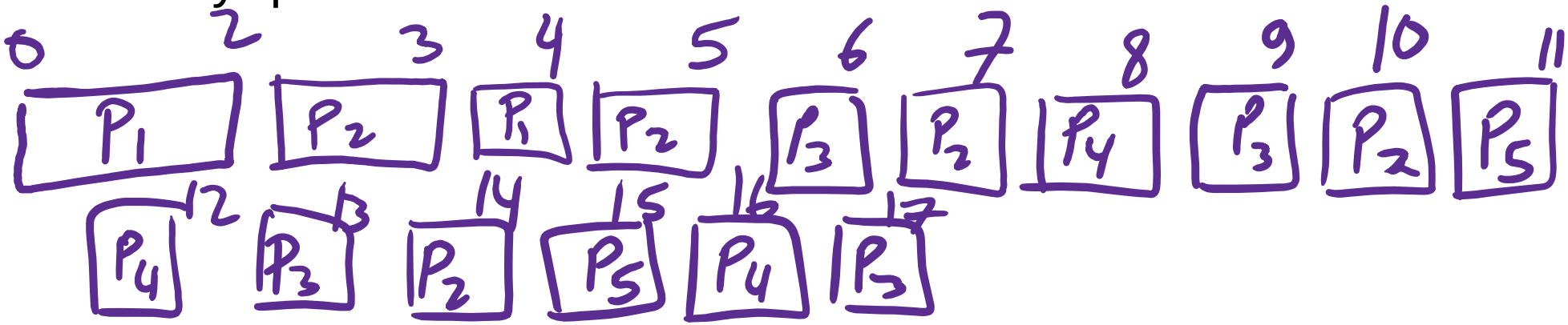
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

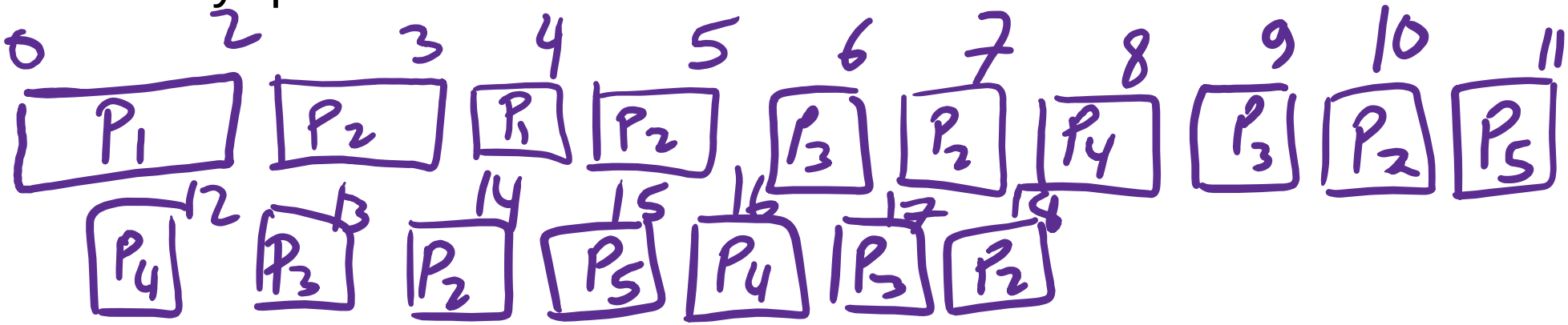
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

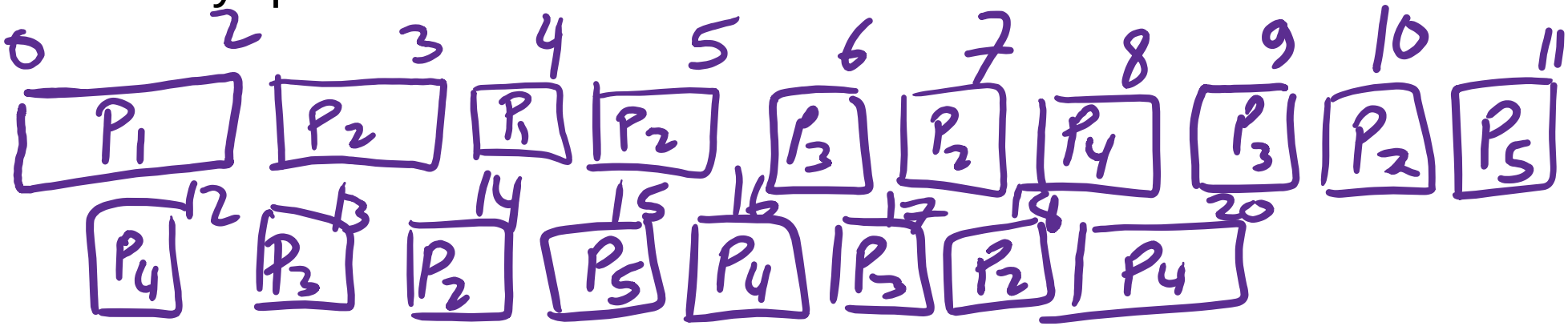
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



Round-Robin

- Selection function: same as FCFS
- Decision mode: pre-emptive
  - a process is allowed to run until the time slice period (quantum, typically from 10 to 100 ms) has expired
  - a clock interrupt occurs and the running process is put on the ready queue

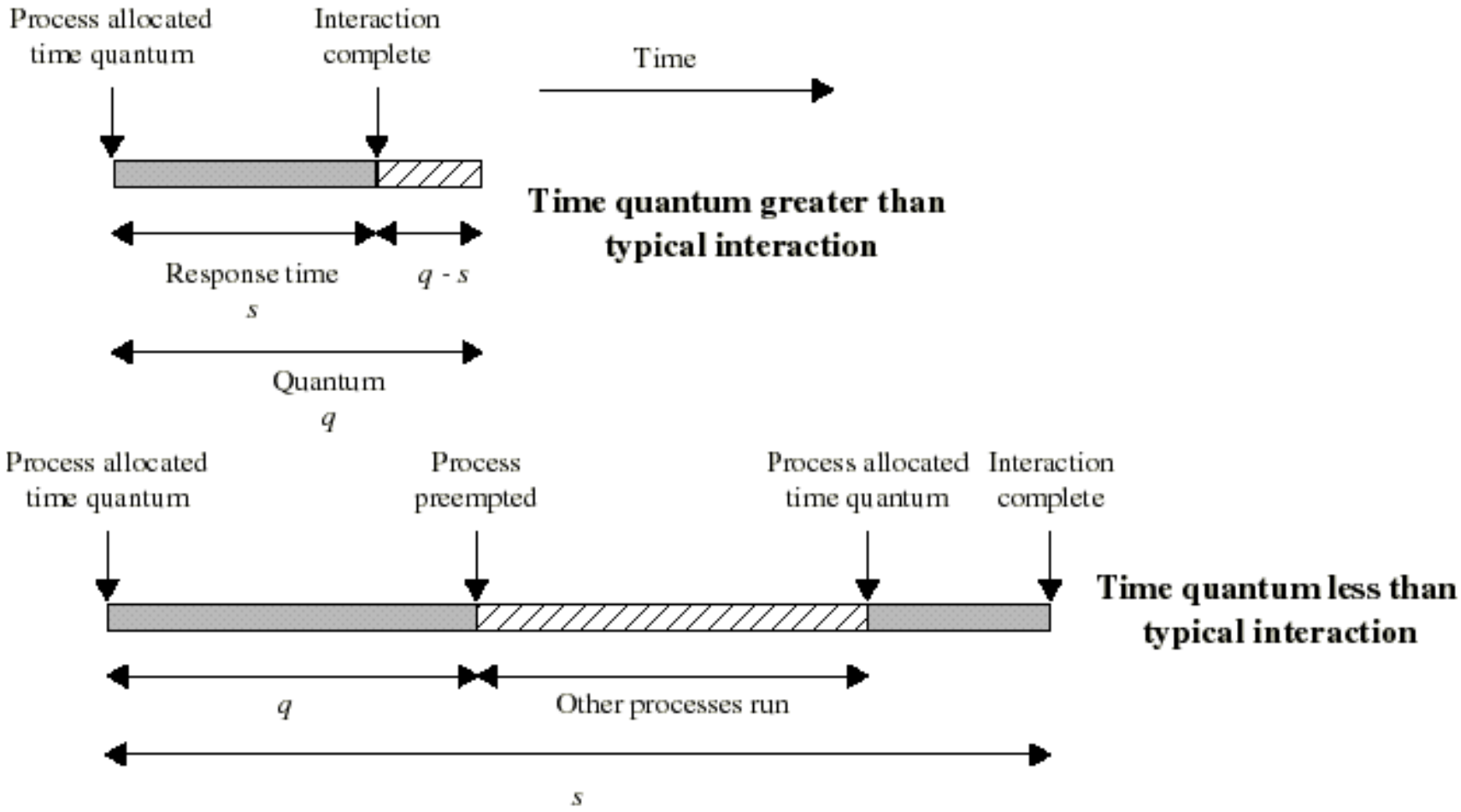
Process	Arrival Time	Service Time
1	0	3
2	2	6
3	4	4
4	6	5
5	8	2



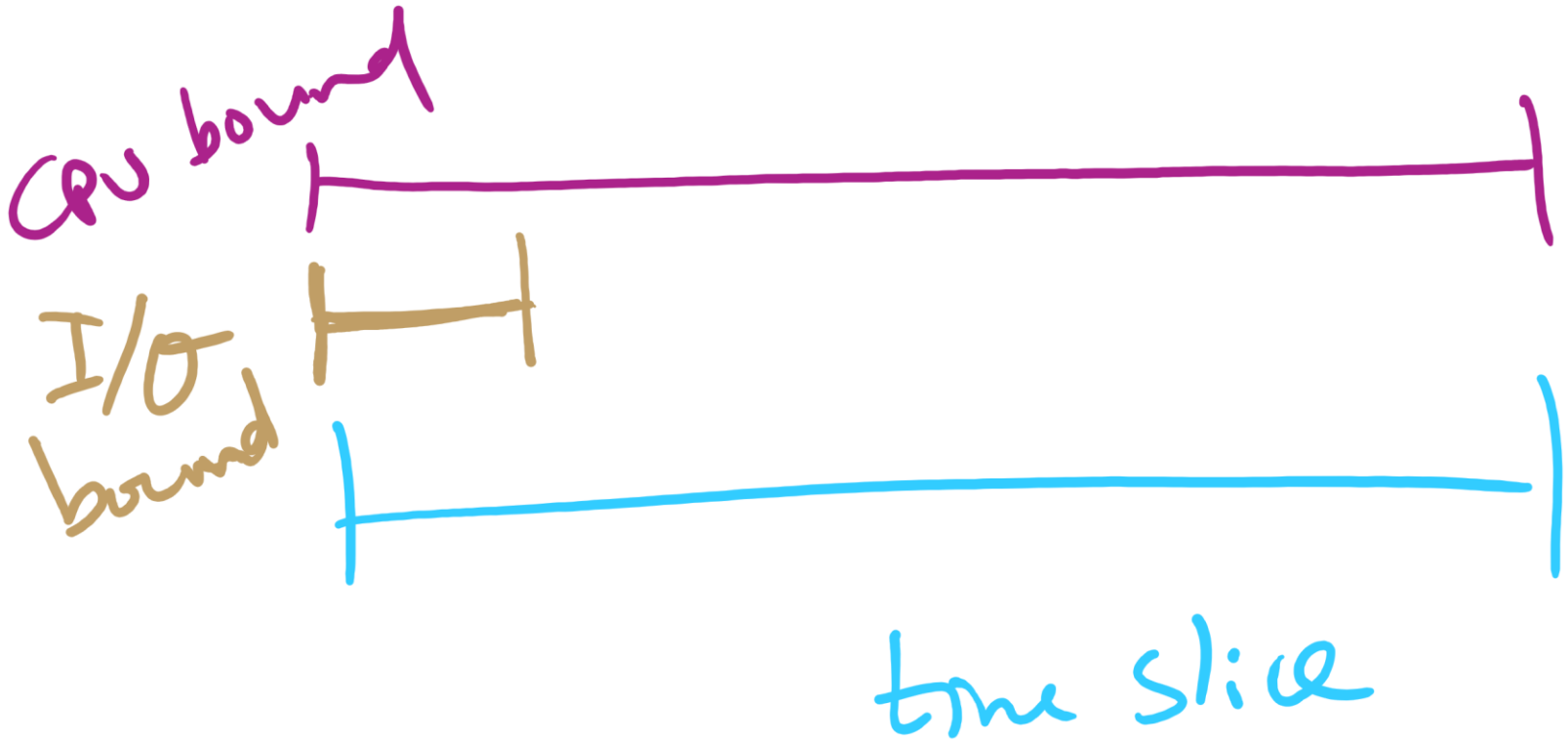
RR Time Quantum

- Quantum must be substantially larger than the time required to handle the clock interrupt and dispatching
- Quantum should be larger then the typical interaction
  - but not much larger, to avoid penalizing I/O bound processes

RR Time Quantum



Quantum Length



Round Robin: critique

- Still favors CPU-bound processes
  - An I/O bound process uses the CPU for a time less than the time quantum before it is blocked waiting for an I/O
  - A CPU-bound process runs for all its time slice and is put back into the ready queue
    - May unfairly get in front of blocked processes