



Process Scheduling Algorithms

Round Robin (RR)

What is Robin Round?

- Round Robin is the pre-emptive version of first-come, first-served scheduling.
- Processes are dispatched in a first-in-first-out sequence but are given a fix time to execute, called as **time quantum or time sliced**.
- If a process does not complete within the time slice, cpu is pre-empted and given to next process waiting in queue.
- If a process gets blocked due to an I/O operation before its time slice expires, it is, of course, enters a blocked because of that I/O operation. Once that operation completes, it is placed on the end of the run queue and waits its turn.
- A big advantage of round robin scheduling over non-pre-emptive schedulers is that it dramatically improves average response times.
- By limiting each task to a certain amount of time, the operating system can ensure that it can cycle through all ready tasks, giving each one a chance to run.



Example

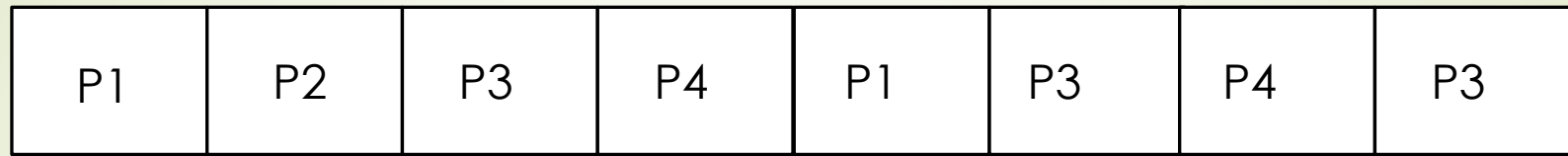
Process	Arrival Time	Execution Time
P1	0	5
P2	1	3
P3	2	8
P4	3	6

Time Quantum = 3ms



0 3 6 9 12 14 17 20 22

P1	$5 - 3 = 2$	$2 - 3 = -1$	Finish
P2	$3 - 3 = 0$	Finish	
P3	$8 - 3 = 5$	$5 - 3 = 2$	$2 - 3 = -1$
P4	$6 - 3 = 3$	$3 - 3 = 0$	Finish



0 3 6 9 12 14 17 20 22

Average Time:

Process	Wait Time : Service Time - Arrival Time
P1	$(0 - 0) + (12 - 3) = 9$
P2	$(3 - 1) = 2$
P3	$(6 - 2) + (14 - 9) + (20 - 17) = 12$
P4	$(9 - 3) + (17 - 12) = 11$

Average Wait Time: $(9+2+12+11) / 4 = 8.5$



Advantage and disadvantage

- **Advantage:** Round robin scheduling is fair in that every process gets an equal share of the CPU. It is easy to implement and, if we know the number of processes on the run queue, we can know the worst-case response time for a process.
- **Disadvantage:** Giving every process an equal share of the CPU is not always a good idea. For instance, highly interactive processes will get scheduled no more frequently than CPU-bound processes.