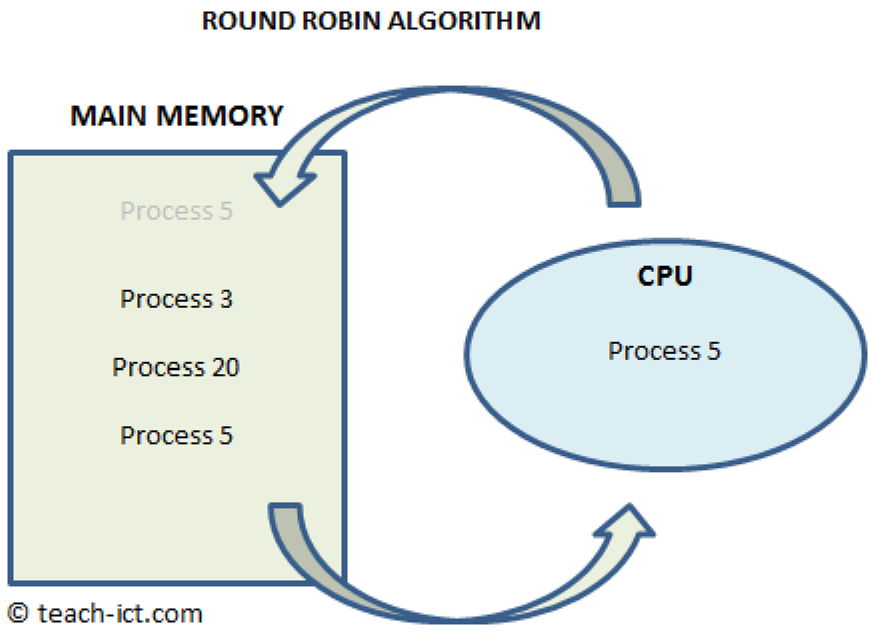
Round Robin



Inputs:

- A set of processes with **Arrival Time (AT)** and **Burst Time (BT)**
- **Time Quantum (q)** – how long a process can run before being interrupted to run another process, if any.

Core Idea:

- Each process from ready processes gets the CPU for q time units.
- If it's not finished, it goes to the **end of the ready queue**.
- New arrivals are added to the **end of the ready queue** as they come.

Simple Algorithm

1️⃣ Initialization

- Set current time = 0
- Create an empty Ready Queue
- Sort all processes by Arrival Time

2️⃣ Add Processes to Ready Queue

- Insert into the Ready Queue all processes with Arrival Time ≤ current time (0)
- The order is FCFS by arrival time

3️⃣ Scheduling Loop (Main Logic)

Repeat until all processes finish:

- Remove the first process from the Ready Queue (FCFS)
- Run the process for: min(Time Quantum q, Remaining Burst Time)
- During execution: If any new process arrives, insert it at the end of the Ready Queue immediately
- After execution:

  If the process finished → mark it as completed and remove it.
  Else → put it at the end of the ready queue
- Stop when all processes are completed

Question

Schedule the following process using RR. Then calculate the Turnaround Time, the Waiting Time, the Response Time (RT), and the averages.

Time Quantum to use: q = 2, q = 4, q = 6

Processes:

Process	**Arrival Time (AT)**	**Burst Time (BT)**
P1	0	5
P2	1	4
P3	1	6

1️⃣ Round Robin with q = 2

Step-by-Step Table: P1 Arrival – 0, Burst 5 - P2 Arrival – 1, Burst 4 - P3 Arrival – 1, Burst 6

Time	Process (runs, remaining)	Ready Queue after running
0–2	P1 (runs 2, remaining 3)	P2,p3,p1
2–4	P2 (runs 2, remaining 2)	P3, P1,p2
4–6	P3 (runs 2, remaining 4)	P1, P2,p3
6–8	P1 (runs 2, remaining 1)	P2, P3,p1
8–10	P2 (runs 2, remaining 0)	P3, P1
10–12	P3 (runs 2, remaining 2)	P1, P3
12–13	P1 (runs 1, remaining 0)	P3
13–15	P3 (runs 2, remaining 0)	–

Summary Table (q = 2)

Process	Start Time	Completion Time (CT)	Response Time (RT = first start – AT)	Waiting Time (WT = TAT – BT)	Turnaround Time (TAT = CT – AT)
P1 - 0	0	13	0	8	13
P2 - 1	2	10	1	5	9
P3 - 1	4	15	3	8	14

Averages (q = 2):

- Average RT = (0+1+3)/3 = 1.33
- Average WT = (8+5+8)/3 = 7
- Average TAT = (13+9+14)/3 = 12

2️⃣ Round Robin with q = 4

Step-by-Step Table: P1 Arrival – 0, Burst 5 - P2 Arrival – 1, Burst 4 - P3 Arrival – 1, Burst 6

Time	Process (runs, remaining)	Ready Queue after running
0–4	P1 (runs 4, remaining 1)	P2, p3, p1
4–8	P2 (runs 4, remaining 0)	P3, P1
8–12	P3 (runs 4, remaining 2)	P1, P3
12–13	P1 (runs 1, remaining 0)	P3
13–15	P3 (runs 2, remaining 0)	–

Summary Table (q = 4)

Process	Start Time	Completion Time (CT)	Response Time (RT)	Waiting Time (WT)	Turnaround Time (TAT)
P1 – 0	0	13	0	8	13
P2 – 1	4	8	3	3	7
P3 - 1	8	15	7	8	14

Averages (q = 4):

- Average RT = (0+3+7)/3 = 3.33
- Average WT = (8+3+8)/3 = 6.33
- Average TAT = (13+7+14)/3 = 11.33

3️⃣ Round Robin with q = 6

Step-by-Step Table: P1 Arrival – 0, Burst 5 - P2 Arrival – 1, Burst 4 - P3 Arrival – 1, Burst 6

Time	Process (runs, remaining)	Ready Queue after running
0–5	P1 (runs 5, remaining 0)	P2, p3
5–9	P2 (runs 4, remaining 0)	P3
9–15	P3 (runs 6, remaining 0)	–

Summary Table (q = 6)

Process	Start Time	Completion Time (CT)	Response Time (RT)	Waiting Time (WT)	Turnaround Time (TAT)
P1 – 0	**0**	**5**	**0**	**0**	**5**
P2 - 1	**5**	**9**	**4**	**4**	**8**
P3 - 1	**9**	**15**	**8**	**8**	**14**

Averages (q = 6):

- Average RT = (0+4+8)/3 = 4
- Average WT = (0+4+8)/3 = 4
- Average TAT = (5+8+14)/3 = 9

✅ Observations Across q Values

q	Avg RT	Avg WT	Avg TAT
2	1.33	7	12
4	3.33	6.33	11.33
6	4	4	9

- Smaller **q** → more context switches, higher waiting time.
- Larger **q** → fewer switches, less waiting time, some processes may finish in one slice.