Example-4:

Example data:

Process	Arrival Time	Burst Time
1	0	8
2	1	4
3	2	9
4	3	5

Round Robin, Quantum = 4, no priority – based preemption

P1	P2	P3	P4	P1	P3	P4	P3

0 4 8 12 16 20 24 25 26

Soln.

Process	AT	BT	CT	TAT = CT – AT	WT = TAT – BT
P1	0	8	20	20	12
P2	1	4	8	7	3
P3	2	9	26	24	15
P4	3	5	25	22	17

Average TAT $= \dfrac{20+7+24+22}{4} = \dfrac{75}{4} = 18.25$ Average WT $= \dfrac{12+3+15+12}{4} = 11.75$

Note: In Round Robin, if there are 'n' processes, and 'q' is the time qunatum, then no process waits more than $(n-1)\times q$ time units until the next time quantum.

Problem: Assume that the following jobs are to be executed on a single processor system

Job Id	CPU Burst time
p	4
q	1
r	8
s	1
t	2

The jobs are assumed to have arrived at time 0^+ and in the order p, q, r, s, t. Calculate the departure time (completion time) for job p if scheduling is round robin with time slice 1. **[GATE-1993 : 2 Marks]**

(a) 4 (b) 10 (c) 11 (d) 12

Ans. (c)

Soln. Create the gantt chart with Round Robin scheduling.

0 1 2 3 4 5 6 7 8 9 10 11

p	q	r	s	t	p	r	t	p	r	p	⋯

Problem: Which scheduling policy is most suitable for a time-shared operating systems ?

(a) Shortest Job First (b) Round Robin **[GATE-1995 : 1 Mark]**

(c) First come first serve (d) Elevator

Ans. (b)

Soln. Round Robin scheduling working on time quantum, after certain time every process get back the CPU units for it's completion and the same phenomena used in time sharing system. So Round Robin is best for time sharing system.

Problem: Four jobs to be executed on a single processor system arrive at time 0^+ in the order A, B, C, D. Their burst CPU time requirements are 4, 1, 8, 1 time units respectively. The completion time of A under Round Robin scheduling with time slice of one time unit is

(a) 10 (b) 4 (c) 8 (d) 9

Ans. (d)

Soln. Draw Gantt chart and place jobs according to Round Robin scheduling with time quantum '1' unit.

0	1	2	3	4	5	6	7	8	9
A	B	C	D	A	C	A	C	A	C

So correct answer is '9'.

[GATE-1996 : 2 Marks]

Problem: If the time-slice used in the round robin scheduling policy is more than the maximum time required to execute any process, then the policy will

(a) degenerate to shortest job first
(b) degenerate to priority scheduling
(c) degenerate to first come first serve
(d) None of the above

Ans. (c)

Soln. When time quantum used in round robin scheduling is more than maximum time required to execute any process then its behave like first come first serve.

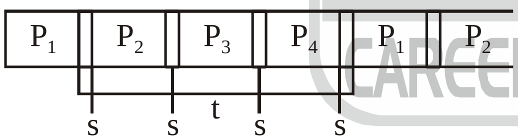
[GATE-2008 : 2 Marks]

Problem: Consider n processes sharing the CPU in a Round Robin fashion. Assuming that each process switch takes s seconds, what must be the quantum size q such that the overhead resulting from process switching is minimized but at the same time each process is guaranteed to get its turn at the CPU at least every t seconds ? **[GATE-1998 : 1 Mark]**

(a) $q \le \frac{t - ns}{n-1}$ (b) $q \ge \frac{t - ns}{n-1}$ (c) $q \le \frac{t - ns}{n+1}$ (d) $q \ge \frac{t - ns}{n+1}$

Ans. (a)

Soln. Let n = 4



$$t \ge n \times s + q \times (n-1)$$

So, $q \le \frac{t - ns}{n-1}$

Problem: A uni-processor computer system only has two processes, both of which alternate 10 ms CPU burst with 90 ms I/O bursts. Both the processes were created at nearly the same time. The I/O of both processes can proceed in parallel. Which of the following scheduling strategies will result in the least CPU utilization (over a long period of time) for this system ?

(a) First come first served scheduling
(b) Shortest remaining time first scheduling
(c) Static priority scheduling with different priorities for the two processes
(d) Round robin scheduling with a time quantum of 5 ms

Ans. (d)

Soln. CPU = 10 ms

I/O = 90 ms