CS & IT ENGINEERING

Operating System

REVISION

CPU Scheduling- Part-02

Lecture No.- 04

By- Dr. Khaleel Khan Sir

Recap of Previous Lecture

Topic

CPU Scheduling

FCFS, SJF, SRTF

Topics to be Covered

Topic Round Robin (RR)

Topic Priority Scheduling

Topic Multilevel Queue

Topic

Topic : Round Robin (RR)

: PreEmptive Scheduling

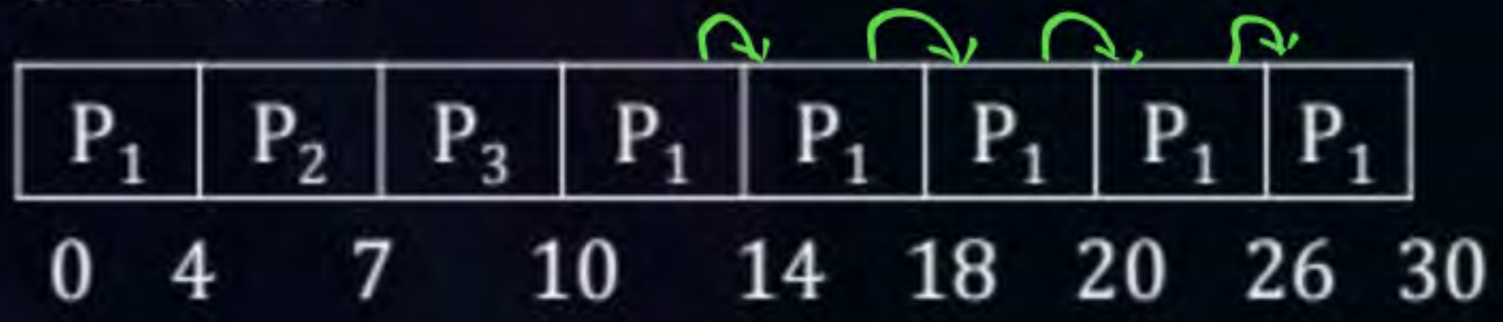
Criteria : $\boxed{A.T + \underline{TQ}}$

- Each process gets a small unit of CPU time (**time quantum** *q*), usually 10-100 milliseconds. After this time has elapsed, the process is preempted and added to the end of the ready queue.
- If there are *n* processes in the ready queue and the time quantum is *q*, then each process gets 1/*n* of the CPU time in chunks of at most *q* time units at once. No process waits more than (*n*-1)*q* time units.
- Timer interrupts every quantum to schedule next process
- Performance
  - *q* large ⇒ FIFO (FCFS)
  - *q* small ⇒ RR
- Note that q must be large with respect to context switch, otherwise overhead is too high

Topic : Example of RR with Time Quantum = 4

Process	Burst Time
P_1	24
P_2	3
P_3	3

- The Gantt chart is:



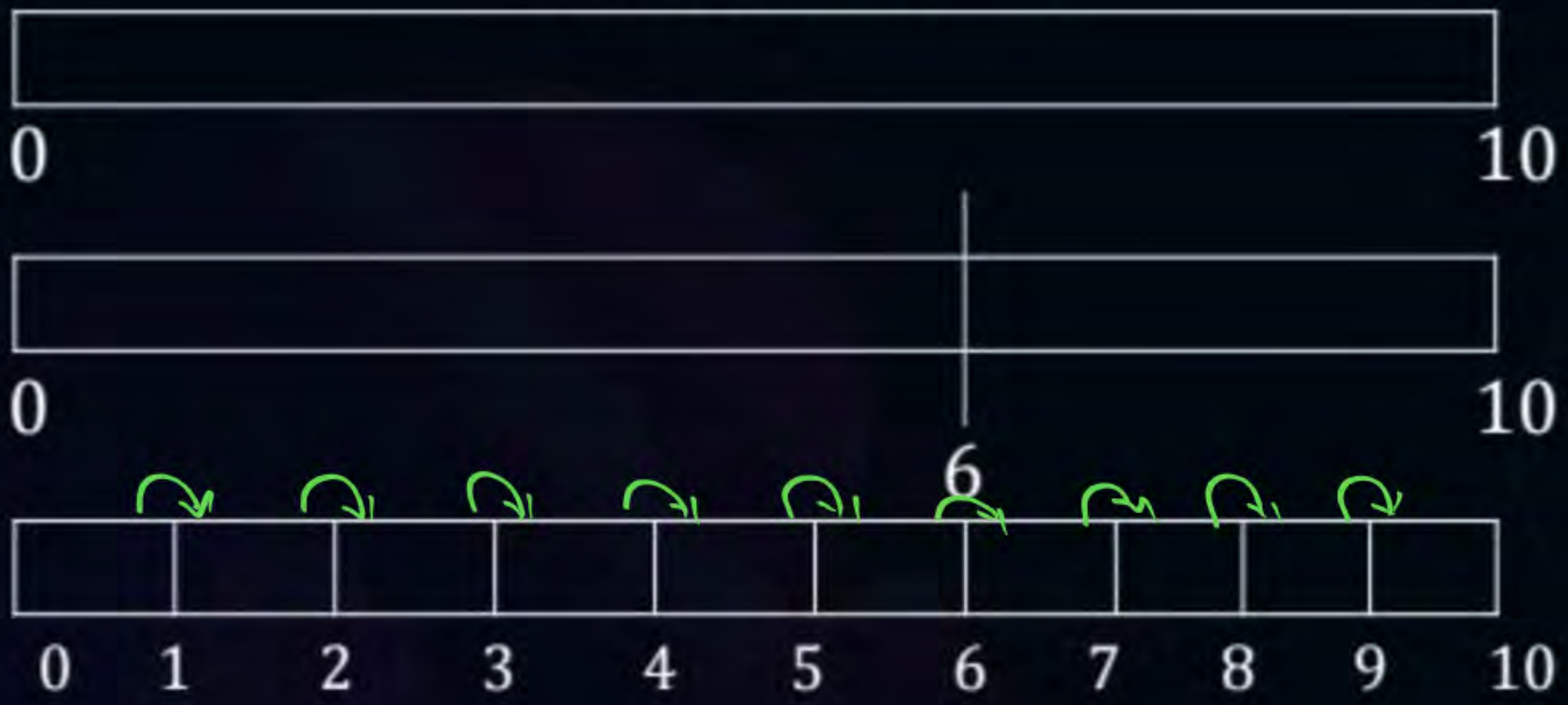
P_1 P_2 P_3 P_1 P_1 P_1 P_1 P_1

0 4 7 10 14 18 20 26 30

- Typically, higher average turnaround than SJF, but better response
- q should be large compared to context switch time
  - q usually 10 milliseconds to 100 milliseconds,
  - Context switch < 10 microseconds

Topic : Time Quantum and Context Switch Time

Process time =10



quantum	Context switches
12 ✓	0
6	1
1	9

$\underline{\underline{TQ = 3}}$

P.NO	A.T	B.T
1	0	6
2	2	4
3	3	5
4	4	2
5	6	8

R.Q ~~P1~~, ~~P2~~, ~~P3~~, ~~P1~~, ~~P4~~, ~~P5~~, ~~P2~~, ~~P3~~, ~~P5~~

cpu	P1	P2	P3	P1	P4	P5	P2	P3	P5	P5
0	3	6	9	12	14	17	18	20	23	25

$W.T = TAT - BT$

$= 16 - 4 = \underline{\underline{12}}$

$Av.TAT = (12 + 16 + 17 + 10 + 19)/5 =$

$$Av.W.T = \frac{6 + (1 + 11) + (3 + 9) + (8) + (8 + 3)}{5} =$$

$Av.R.T = (0 + 1 + 3 + 8 + 8)/5 = 20/5 = 4$

TQ = 2

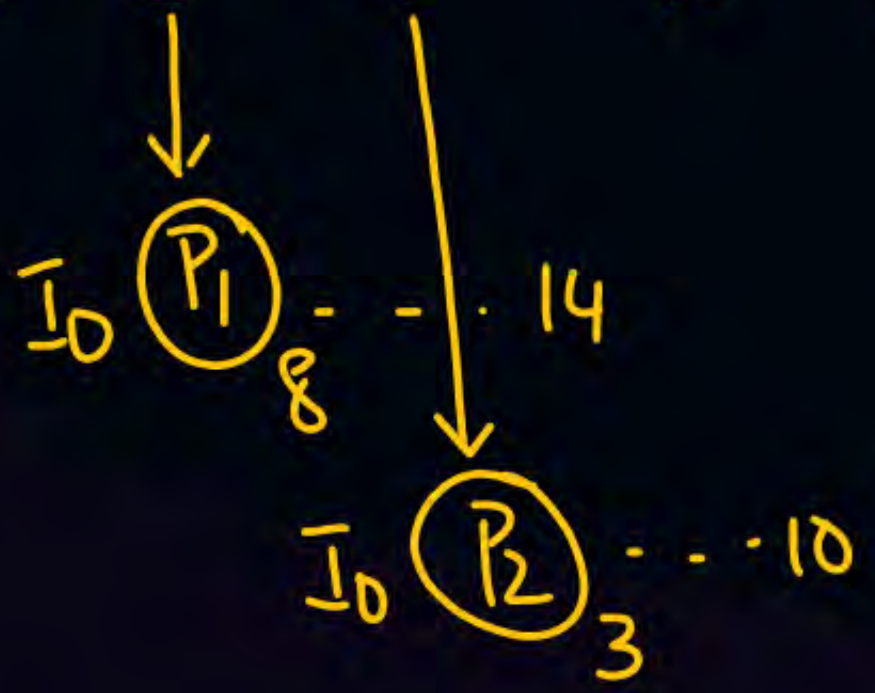
P.NO	A.T	<BT ; IOBT ; BT>
1	0	<4 ; 8 ; 6>
2	1	<3 ; 3 ; 5>

$\frac{\text{\% cpu Idleness}}{L} = 3/21$

$= 1/7$

RQ: ~~P1~~ P2, ~~P1~~, ~~P2~~ ~~P2~~ ~~P2~~ ~~P1~~ ~~P2~~ P1

cpu	P1	P2	P1	P2	▨	P2	P2	P1	P2	P1	P1
0	2	4	6	7	10	12	14	16	17	19	21



Performance of R.R

↓

Time Quantum

- Small
  - → More CS overhead
  - → More Interactive
- Large
  - → Less C.S overhead
  - → Less Interactive
- → Very large
  - → FCFS

Topic : Priority Scheduling

- A priority number (integer) is associated with each process
- The CPU is allocated to the process with the highest priority (smallest integer ≡ highest priority)
  - Preemptive
  - Nonpreemptive
- SJF is priority scheduling where priority is the inverse of predicted next CPU burst time
- Problem ≡ **Starvation** – low priority processes may never execute
- Solution ≡ **Aging** – as time progresses increase the priority of the process

Topic : Example of Priority Scheduling

A·T=0

Process	Burst Time	Priority	
P_1	10	3	
P_2	1	1	— H
P_3	2	4	
P_4	1	5	— L
P_5	5	2	

- Priority scheduling Gantt Chart

P_2	P_5	P_1	P_3	P_4

0 1 6 16 18 19

- Average waiting time = 8.2

Topic : Priority Scheduling w/Round-Robin

⊛ ✓

- Run the process with the highest priority. Processes with the same priority run round-robin

A.T = 0   TQ = 2

$\mathcal{L} = 27$

- Example:

Process	Burst Time	Priority
P_1	4	3
P_2	5	2
P_3	8	2
P_4	7	1
P_5	3	3

CPU	P_4	P_2	P_3	P_2	P_3	P_2	P_3	P_3	P_1	P_5	P_1	P_5	

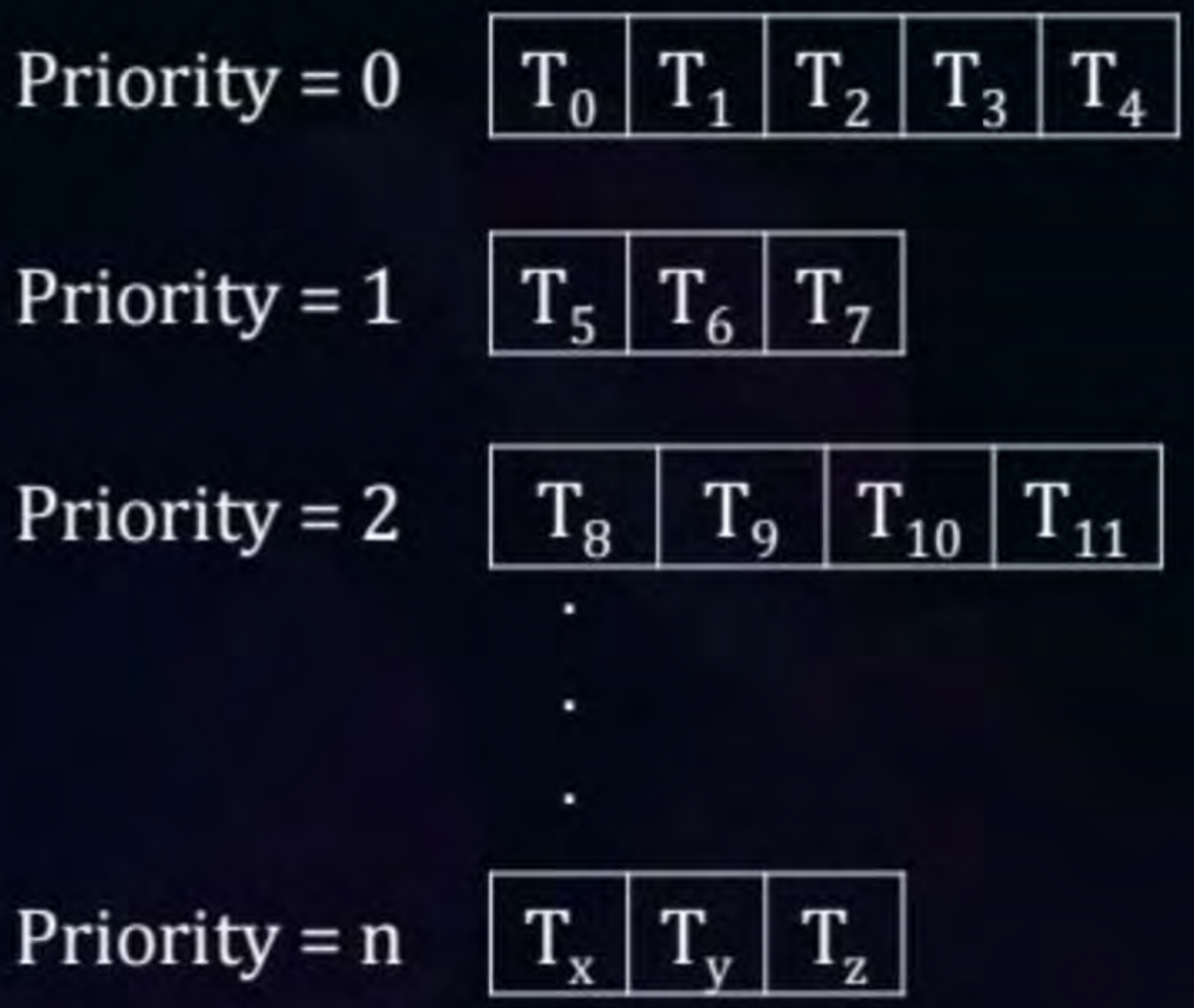
0 7 9 11 13 15 16 18 20 22 24 26 27

Topic : Multilevel Queue

- The ready queue consists of multiple queues
- Multilevel queue scheduler defined by the following parameters:
  - Number of queues
  - Scheduling algorithms for each queue
  - Method used to determine which queue a process will enter when that process needs service
  - Scheduling among the queues

Topic : Multilevel Queue

- With priority scheduling, have separate queues for each priority.
- Schedule the process in the highest-priority queue!



Priority = 0 T_0 T_1 T_2 T_3 T_4

Priority = 1 T_5 T_6 T_7

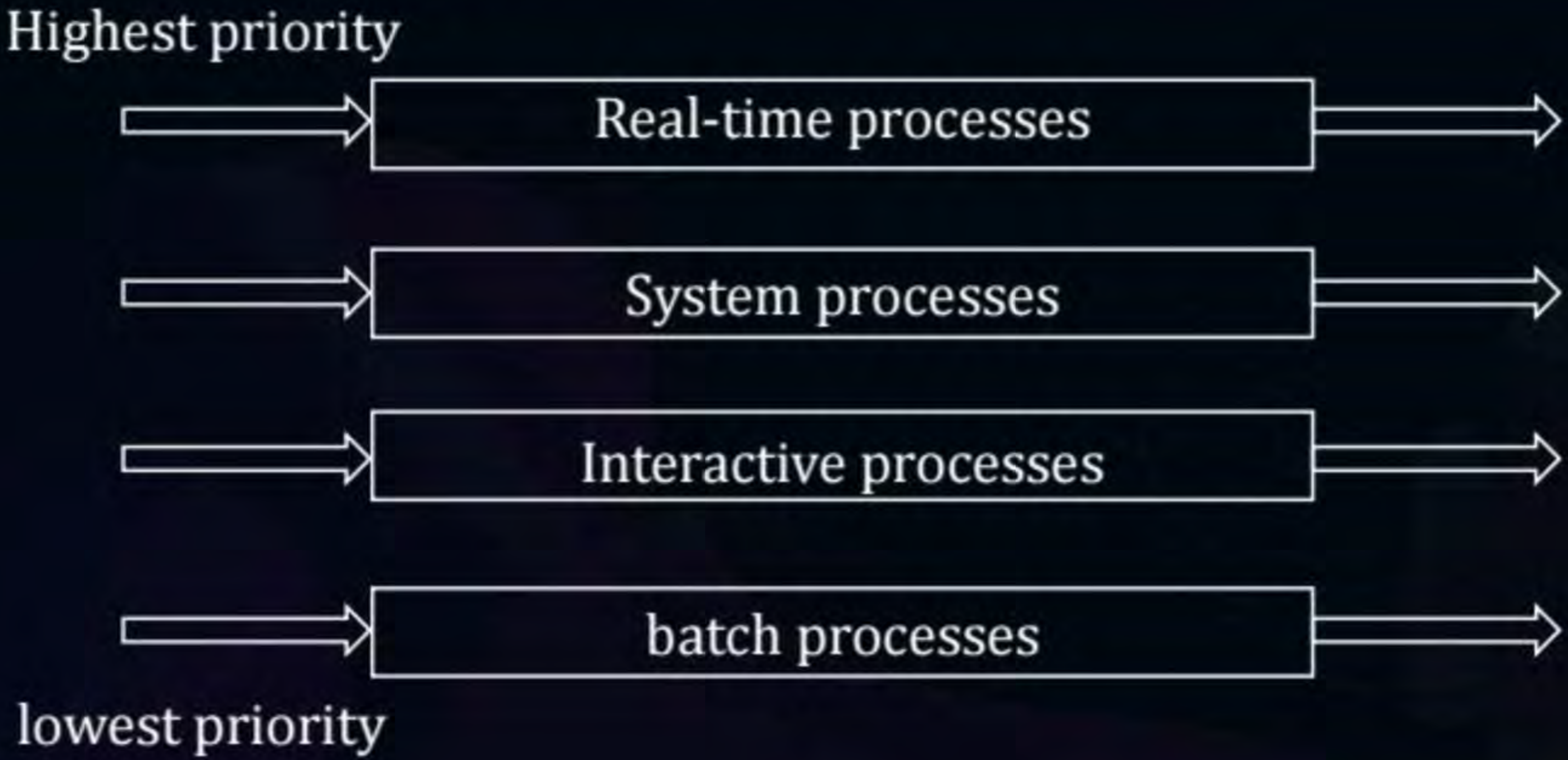
Priority = 2 T_8 T_9 T_{10} T_{11}

.
.
.

Priority = n T_x T_y T_z

Topic : Multilevel Queue

- Prioritization based upon process type



Highest priority

Real-time processes

System processes

Interactive processes

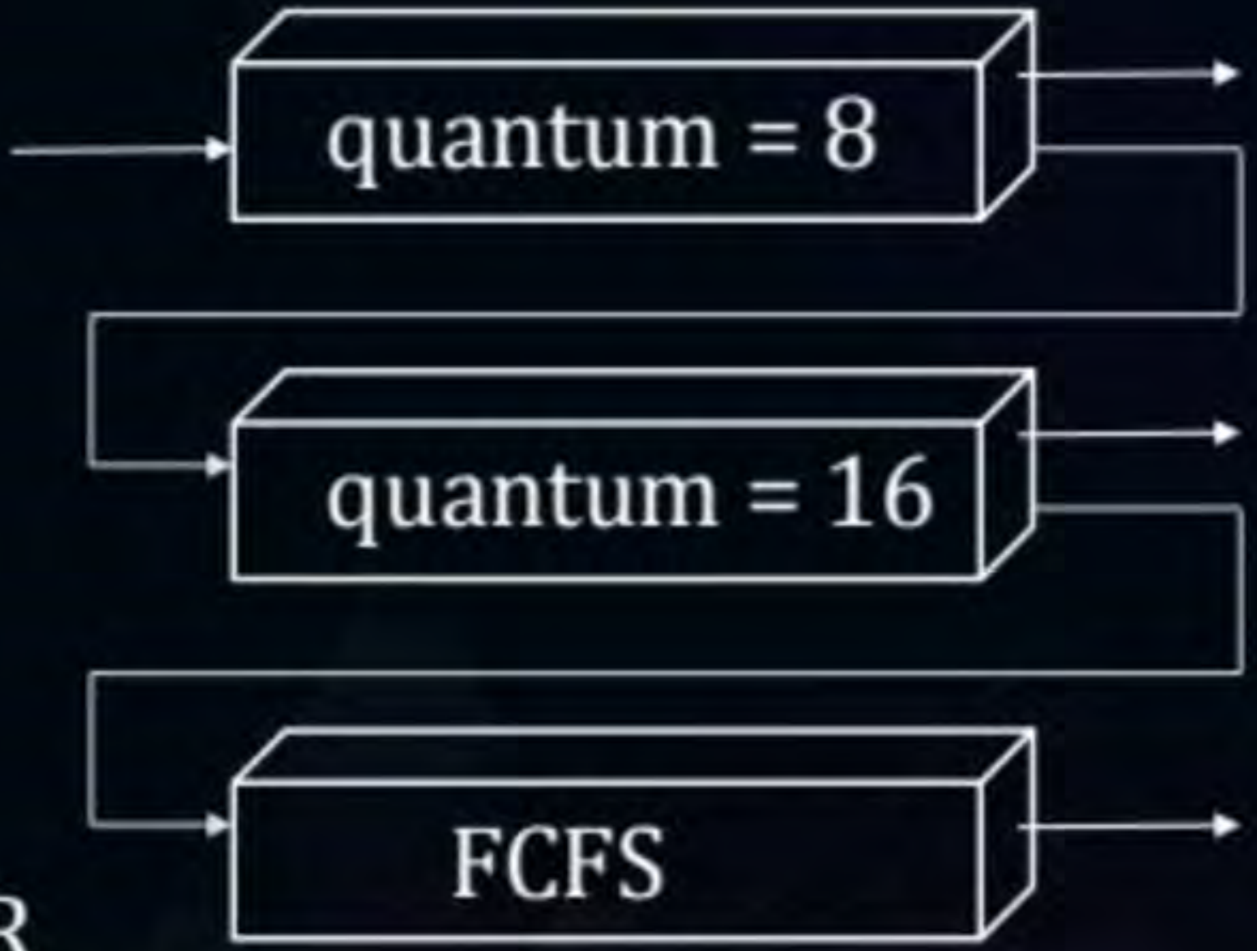
batch processes

lowest priority

Topic : Multilevel Feedback Queue

- A process can move between the various queues.
- Multilevel-feedback-queue scheduler defined by the following parameters:
  - Number of queues
  - Scheduling algorithms for each queue
  - Method used to determine when to upgrade a process
  - Method used to determine when to demote a process
  - Method used to determine which queue a process will enter when that process needs service
- Aging can be implemented using multilevel feedback queue

Topic : Example of Multilevel Feedback Queue



- Three queues:
  - Q_0 – RR with time quantum 8 milliseconds
  - Q_1 – RR time quantum 16 milliseconds
  - Q_2 – FCFS
- Scheduling
  - A new process enters queue Q_0 which is served in RR
    1. When it gains CPU, the process receives 8 milliseconds
    2. If it does not finish in 8 milliseconds, the process is moved to queue Q_1
  - At Q_1 job is again served in RR and receives 16 additional milliseconds
    - If it still does not complete, it is preempted and moved to queue Q_2

Q. Consider a System using Round Robin Scheduling with 10 Processes all arriving at the time 0. Each Process is associated with 20 identical Request. Each Process request consumes 20 ms of CPU time after which it spends 10 ms of time on I/O, thereafter, initiates subsequent Request. Assuming Scheduling Overhead of 2 ms and Time Quantum of 20 ms, Calculate

i. Response time of the 1st request of the 1st Process

ii. Response time of the 1st request of the last Process

iii. Response time of the subsequent request of any Process.

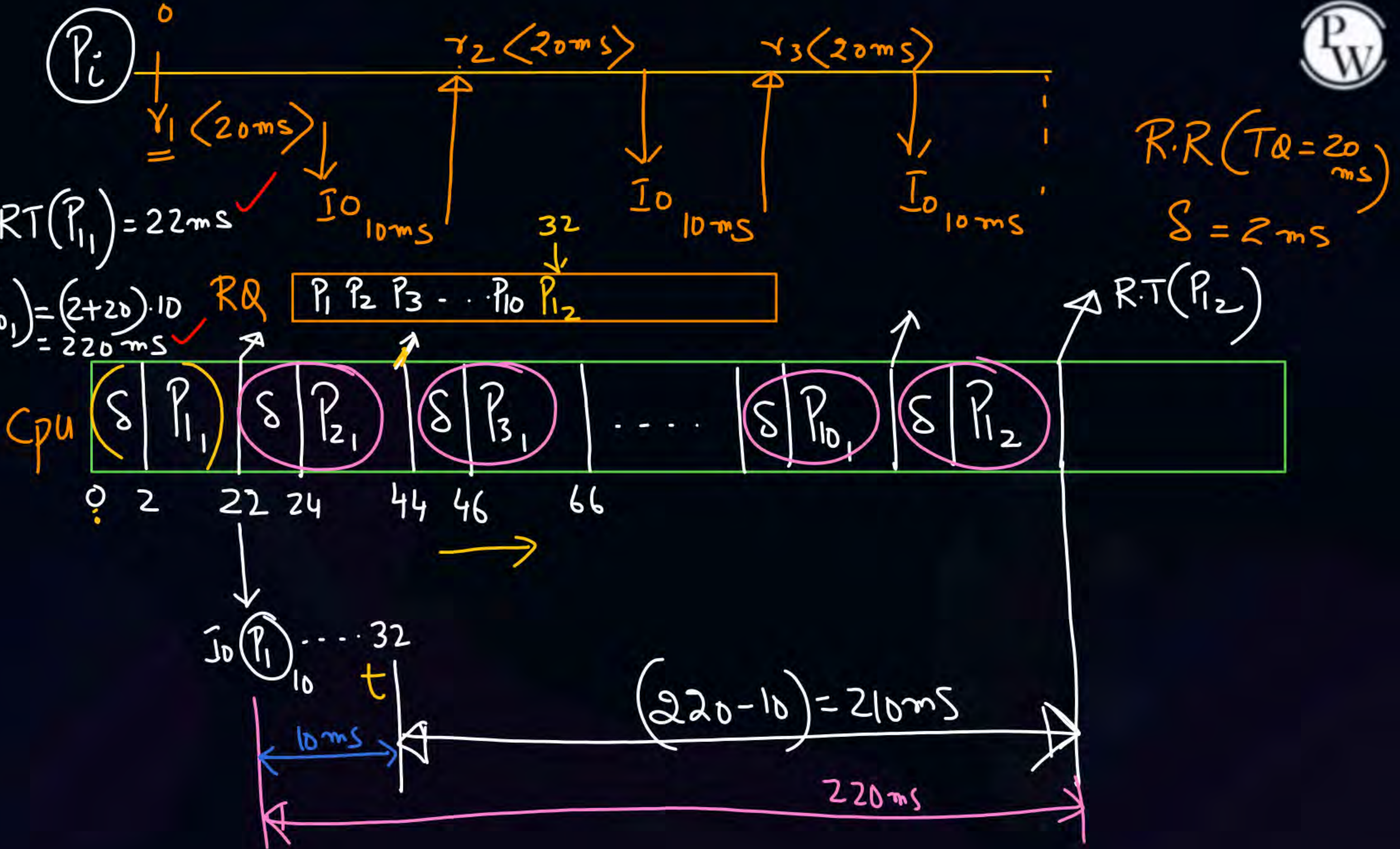
(Next Request)

R.R (TQ = 20 ms)

δ = 2 ms

(i) RT(P_{1_1}) = 22ms ✓

(ii) RT(P_{10_1}) = (2+20)·10
= 220 ms ✓

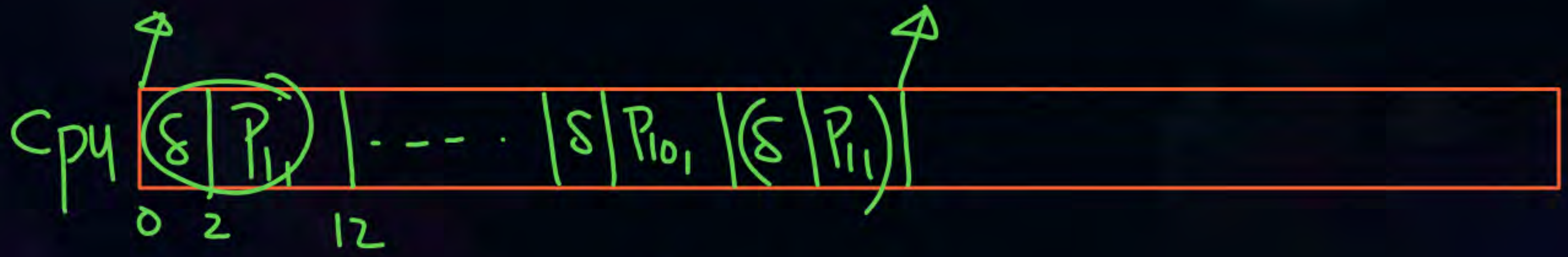


$\underline{TQ = 10\,ms}$; $\delta = 2\,ms$

a) $R.T(P_{1,1}) = 132$ ✓

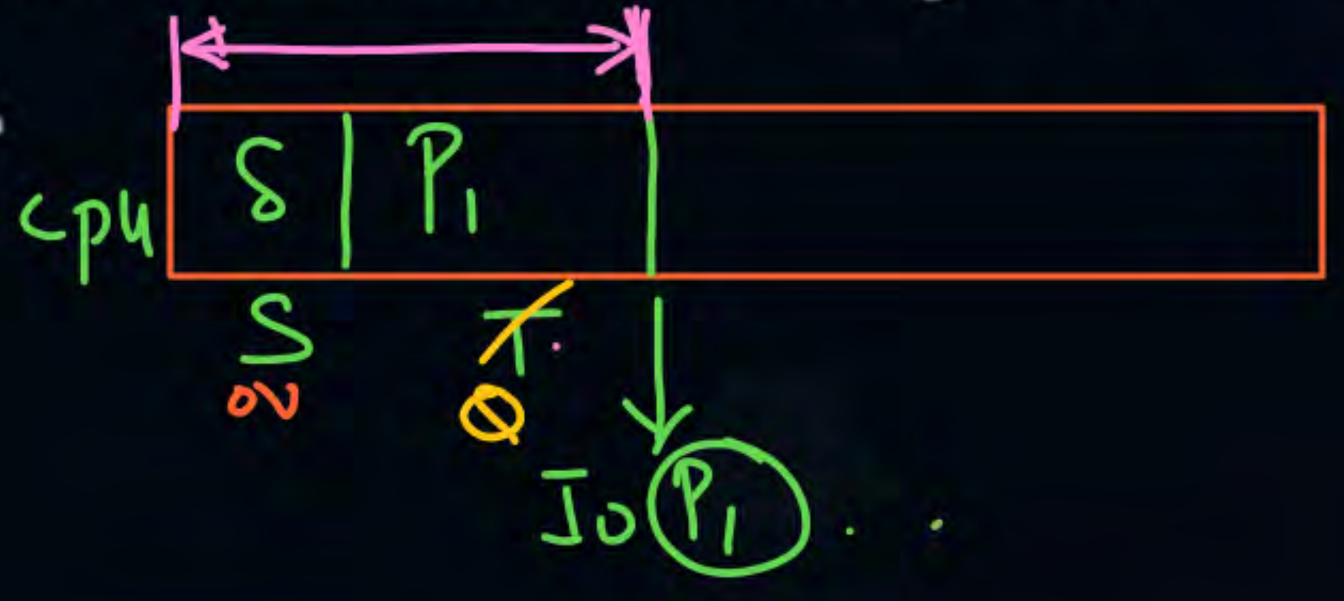
b) $R.T(P_{10,1}) = (2+10)\cdot 10 + (2+10)\cdot 10$

$= \underline{\underline{240}}$ ✓



$(2+10)\cdot 10 + (2+10) = 132$

Q. Consider a System using RR Scheduling with TQ of ‘Q’ seconds & CPU Scheduling overhead is ‘S’ seconds. Each Process on an average run for ‘T’ seconds before blocking on I/O. Give a formula for CPU efficiency for each of the following conditions.



1. $Q = \infty \longrightarrow \mu = \frac{T}{(S+T)}$ ✓

2. $Q > T \longrightarrow \mu = \frac{T}{S+T}$

3. $S < \boxed{Q < T} \longrightarrow \mu = \frac{Q}{S+Q}$

4. $Q = S \longrightarrow \mu = 1/2 = 50\%$

5. $Q \approx 0 \longrightarrow \mu \approx 0$

Q.

Consider Processes P_1&P_2 arriving in the ready queue at time 0 with following properties.

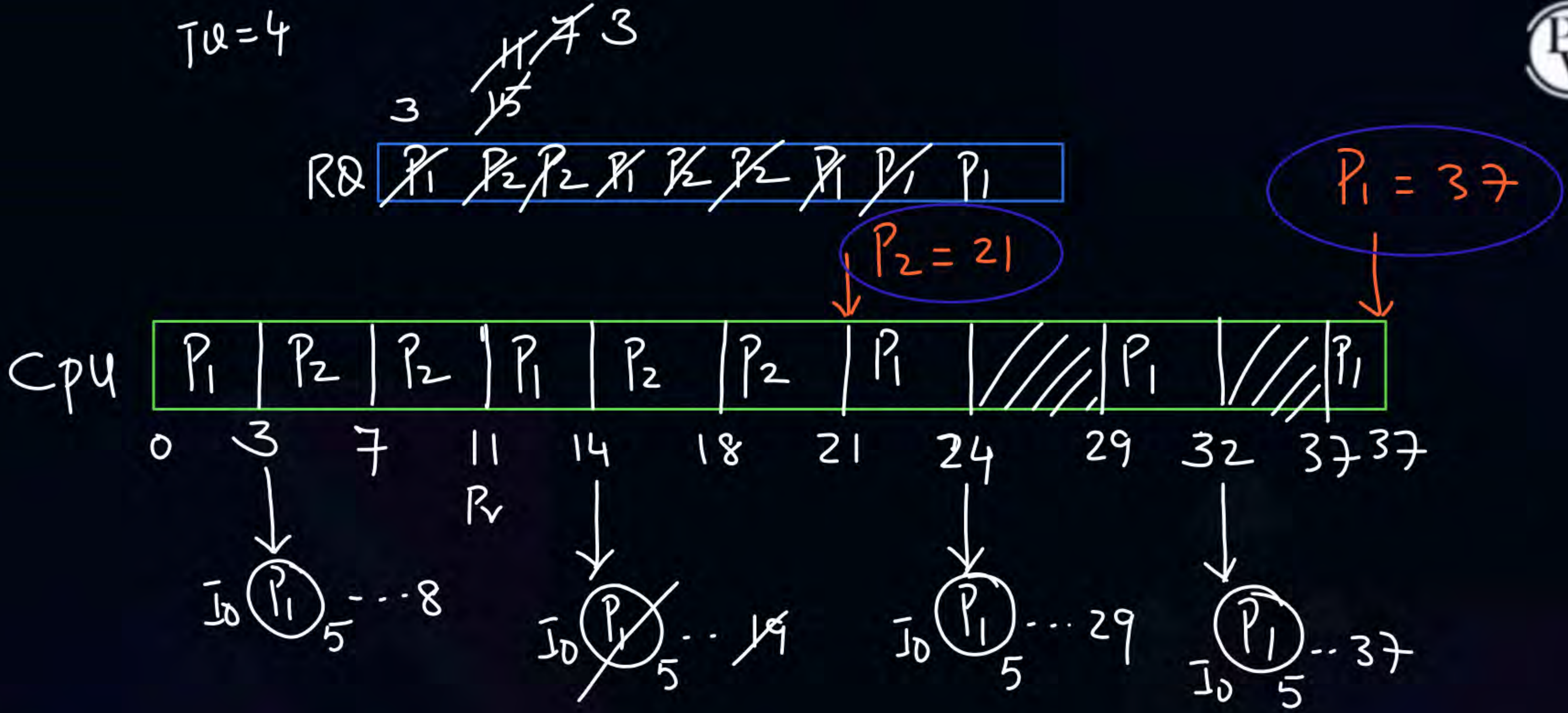
i) P1 needs a total of 12 units of CPU time and 20 units of I/O time. After every 3 units of CPU time $P1$ spends 5 units on I/O.

ii) $P2$ needs a total of 15 units of CPU time and no I/O. P2 arrives just after P1.

Compute the Completion times of P_1&P_2 using the following scheduling techniques:

1. SRTF
2. Round Robin with Time Quanta = 4 units

TQ = 4



~~11~~ ~~7~~ 3
3 ~~15~~
RQ
~~P1~~ ~~P2~~ ~~P2~~ ~~P1~~ ~~P2~~ ~~P2~~ ~~P1~~ ~~P1~~ P1
P2 = 21
P1 = 37
CPU
P1 | P2 | P2 | P1 | P2 | P2 | P1 | | P1 | | P1
0 3 7 11 14 18 21 24 29 32 37 37
P2
IO P1 5 ... 8
IO ~~P1~~ 5 ... ~~19~~
IO P1 5 ... 29
IO P1 5 ... 37

$P_1 \rightarrow \underline{\underline{12}}(BT) + 20(IOBT)$

$\hookrightarrow \underline{\underline{3}}(CPU) \quad \hookrightarrow 5(IO)$

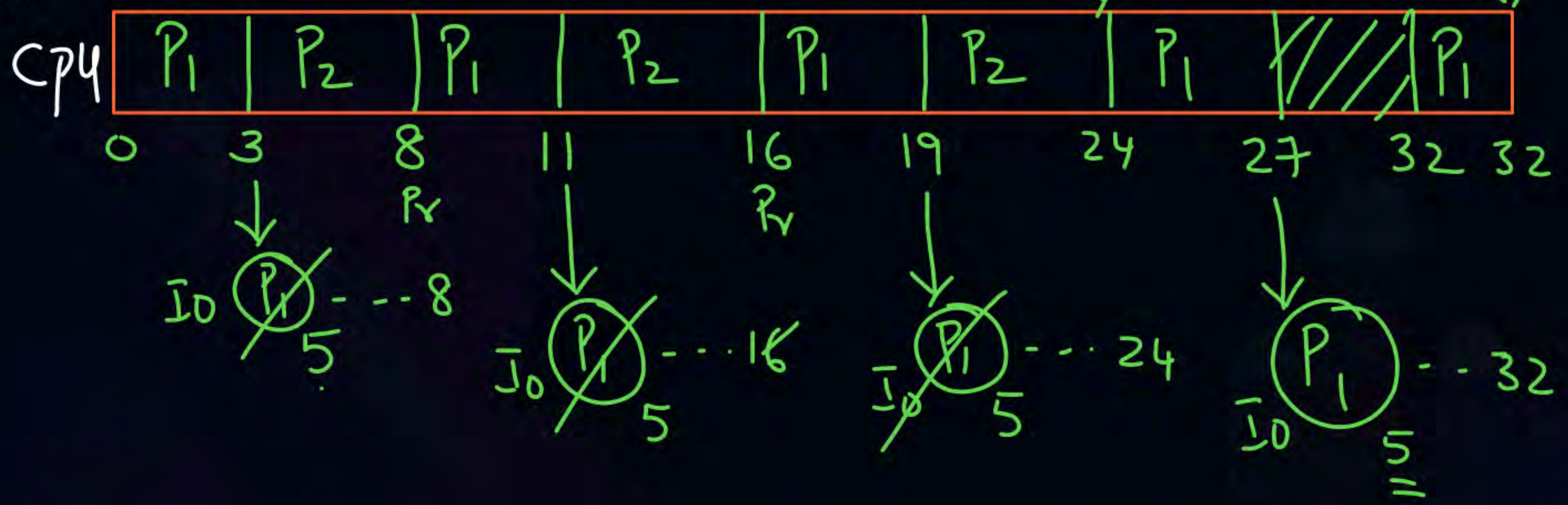
$P_2 \rightarrow \cancel{\underline{\underline{15}}}(B.T)$

$\cancel{10}\ 5$

$\boxed{P_2 = 24}$

$\boxed{P_1 = 32}$

(i) SRTF



Q. Three processes A, B and C each execute a loop of 100 iterations. In each iteration of the loop, a process performs a single computation that requires tc CPU milliseconds and then initiates a single I/O operation that lasts for t_{io} milliseconds. It is assumed that the computer where the processes execute has sufficient number of I/O devices and the OS of the computer assigns different I/O devices to each process. Also the scheduling overhead of the OS is negligible. The processes have the following characteristics:

RQ ~~A~~ ~~B~~ ~~C~~ ~~A~~ B C

Process Id	t_c	t_{io}
A	100 ms	500 ms
B	350 ms	500 ms
C	200 ms	500 ms

cpu	A	B'	C	A	B'	C	B'	C	B'	C

0 50 100 150 200 250 300 350 400 450 500

200 → Io A 500 .. 700

500 → Io C 500 .. 1000

The processes A, B, and C are started at times 0,5 and 10 milliseconds respectively in a pure time-sharing system (round robin scheduling) that uses a time slice of 50 milliseconds. The time in milliseconds at which process C would complete its first I/O operation is (1000) ✓

Q. Which of the following statements (s) is/are correct in the context of CPU Scheduling? <B, C, D> (MSQ)

× A The goal is to only maximize CPU utilization and minimize throughput

[✓ B] Turnaround time includes waiting time TAT = CT - AT

[✓ C] Implementing preemptive scheduling needs hardware support

[✓ D] Round-robin policy can be used even when the CPU time required by each of the processes is not known Apriority.

THANK - YOU