Highest Response Ratio next (HRRN):

SJF & SRTF → best algo → But only problem is long jobs wait longer
→ Starvation → for long jobs.

Shorter jobs get selected

If longer jobs waits for more time, they too get selected.

$$\text{Response Ratio} = \frac{W.T + B.T}{B.T} = \left(1 + \frac{WT}{BT}\right)$$

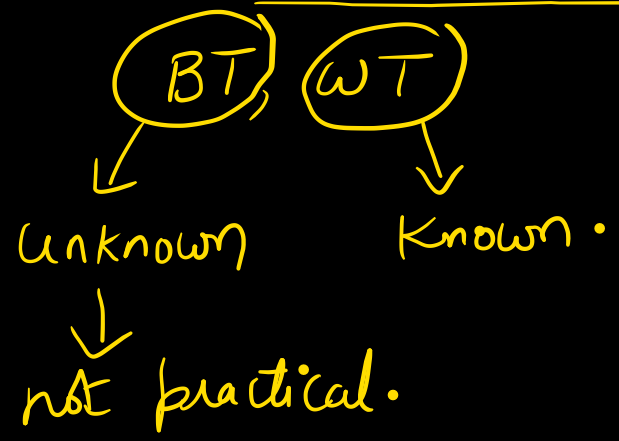
So we select process with highest Response Ratio.

$$RR = \left(1 + \frac{WT}{BT}\right)$$

$BT \downarrow \rightarrow RR \uparrow$ ∴ Shorter jobs get priority.

$WT \uparrow \rightarrow RR \uparrow$ which means if a process waits for too long → it will be selected.

Criteria → Response Ration — $\left(1 + \frac{WT}{BT}\right)$

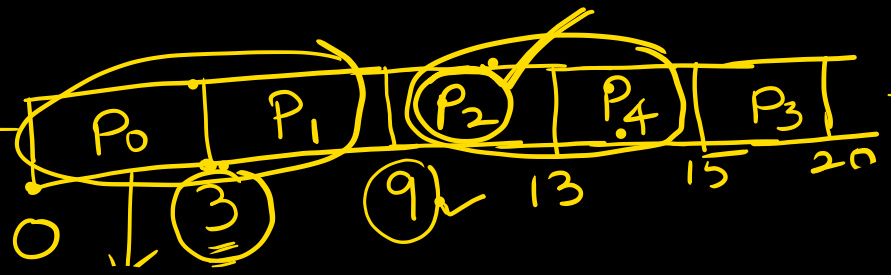


not practical.

Non preemptive algo.

Non PRE

PNO	AT	BT
0	0	3
1	2	6
2	4	4
3	6	5
4	8	2



→ Longer jobs also get selected if they wait more.

SJF:

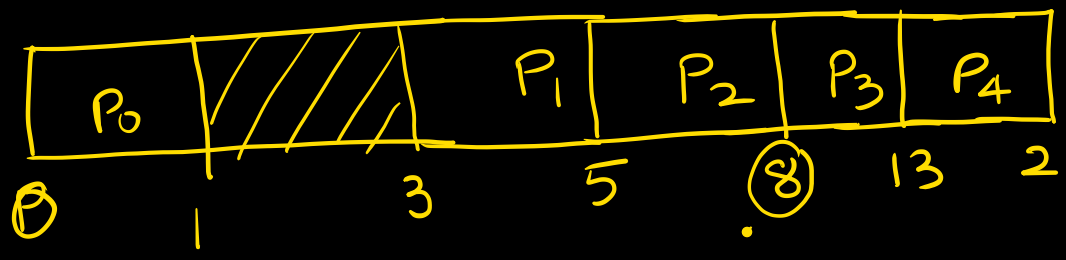
→ only shorter jobs first.

HRRN

PNO	AT	BT	CT	TAT
0	0	1	1	1
1	3	2	5	2
2	5	3	8	3
3	6	5	13	7
4	7	8	21	14

CT of P3.

RR Same → AT Same → PNO. (Low).



$RR_3 = \frac{2+5}{5} = \frac{7}{5} = 1.4$

$RR_4 = \frac{1+8}{8} = \frac{9}{8} = 1.1$

Best throughput - avg WT - Avg TAT → SJF & SRTF

Practical → ~~RR~~ Round Robin.

Priority Scheduling:

Priority

Static

The priority given to a process doesn't change

Dynamic

The priority of a process keeps changing.

not in syllabus.

Priority scheduling

Preemptive

If a higher priority process arrives, then current process will be preempted

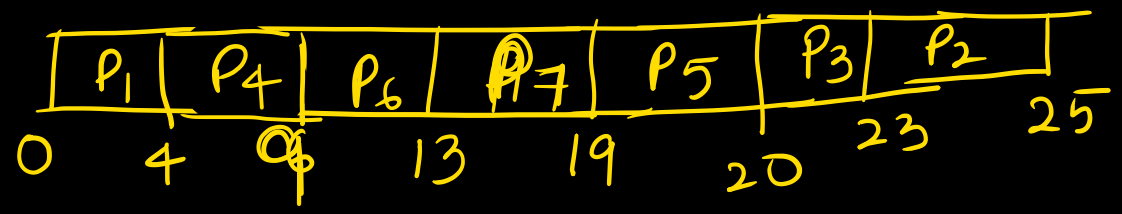
Non preemptive

Once scheduled, it will not stop till end.

Non preemptive priority scheduling: Criteria -> Priority:

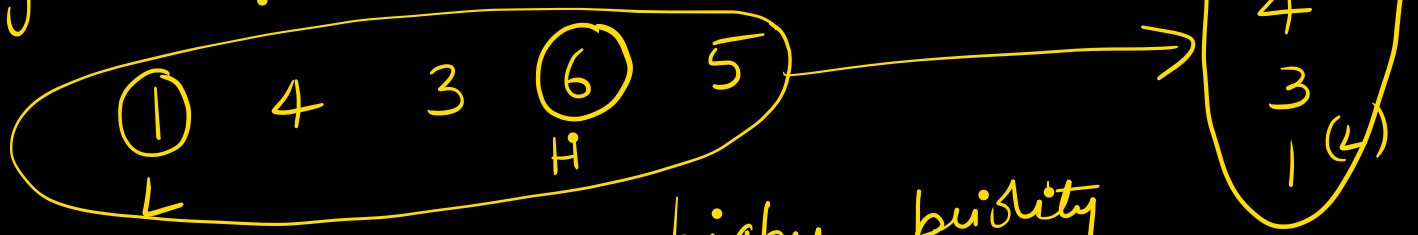
CT of P_2

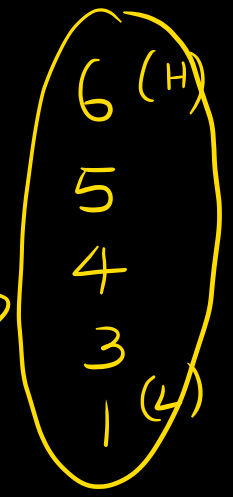
PNO	Priority	AT	BT	CT	TAT	WT	RT
1	2(L)	0	4	4	4	0	0
2	4	1	2	25	24	22	22
3	6	2	3	23	21	18	18
4	10	3	5	9	6	1	1
5	8	4	1	20	16	15	15
6	12(H)	5	4	13	8	4	4
7	9	6	6	19	13	7	7



Two types of priorities

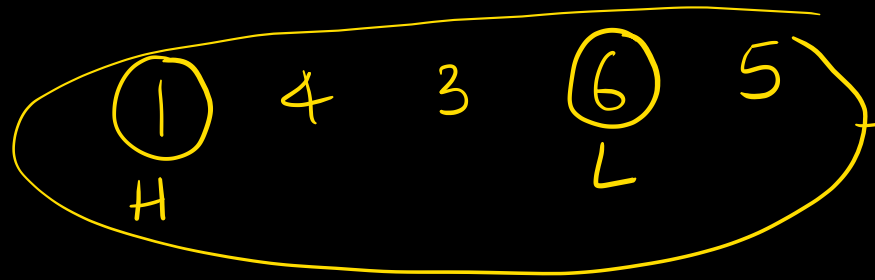
1) Higher number means higher priority.

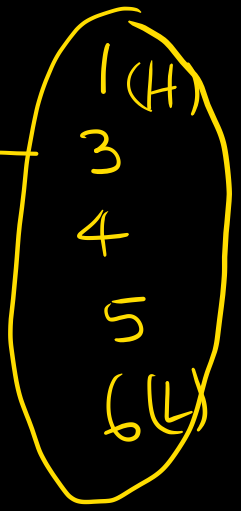




If same → AT same → Lower number (lower).

2) Lower number means higher priority



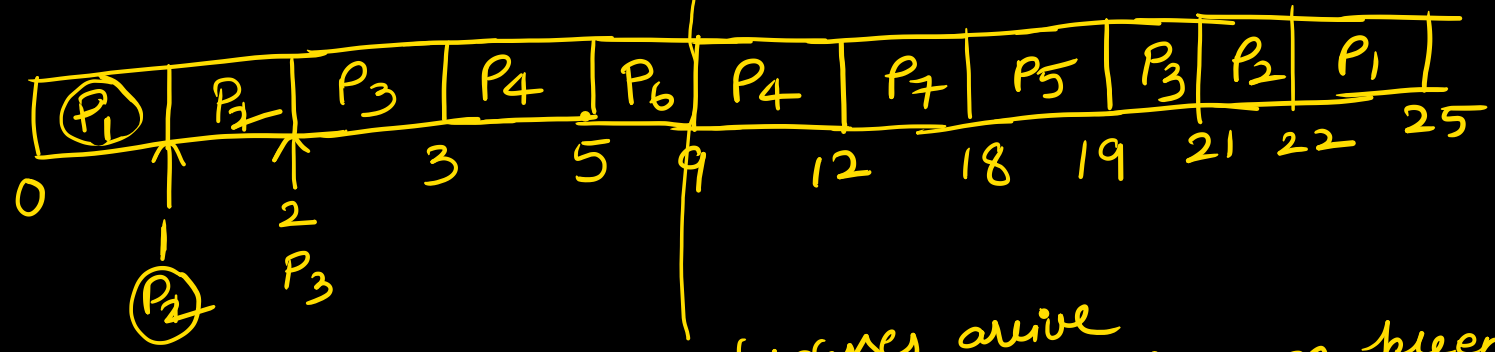


-> In exam they will mention which method is used.

Gate: Preemptive priority:

CT of P1

PNO	Pri	AT	BT	CT	TAT	WT	RT
1	2(4)	0	~~4~~ 3	~~25~~ 25	25	21	0
2	4	1	~~2~~ 1	22	21	19	0
3	6	2	~~3~~ 2	21	19	16	0
4	10	3	~~5~~ 3	12	9	4	0
5	8	4	1	19	15	14	14
×6	12(11)	5	~~4~~ 0	9	4	0	0
7	9	6	6	18	12	6	6

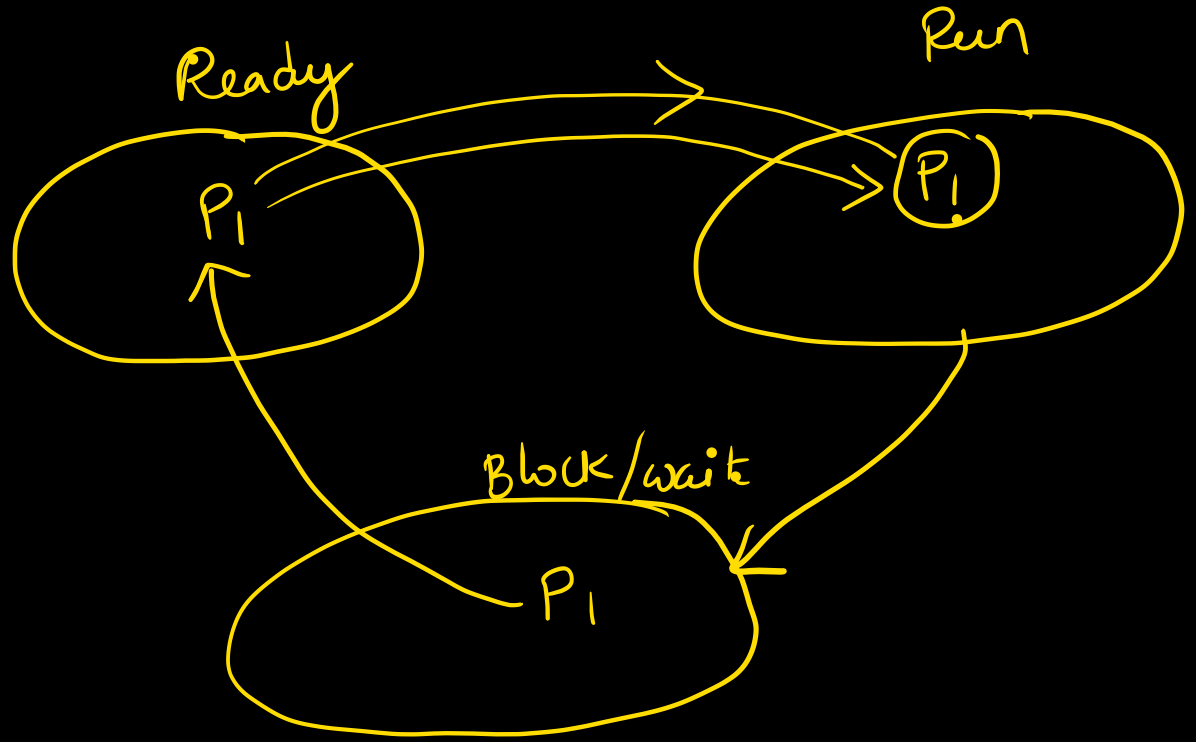


all processes arrive
From here it is like non preemptive.

SRTF with processes containing CPU and I/O time

input/output.

	(BT	IO	BT)
P_1	~~2~~ 0	~~5~~ 0	$\neq 0$

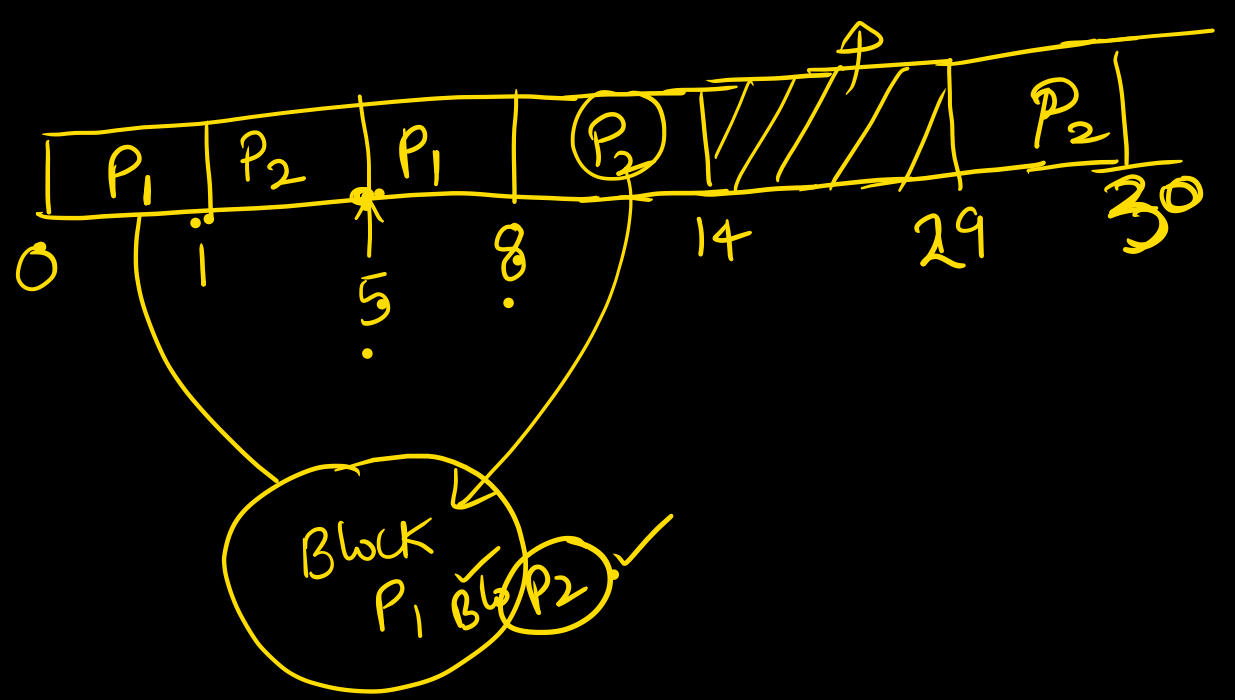


SRTF

SRTF

	BT	I/O	BT	TBT
P_1	1	4	3	4 ~~8~~ 0
P_2	~~10~~ 9	5	~~1~~ 0	

TBT = 7



1 4 3
1 0 1 5 1

Difficult question

Take time

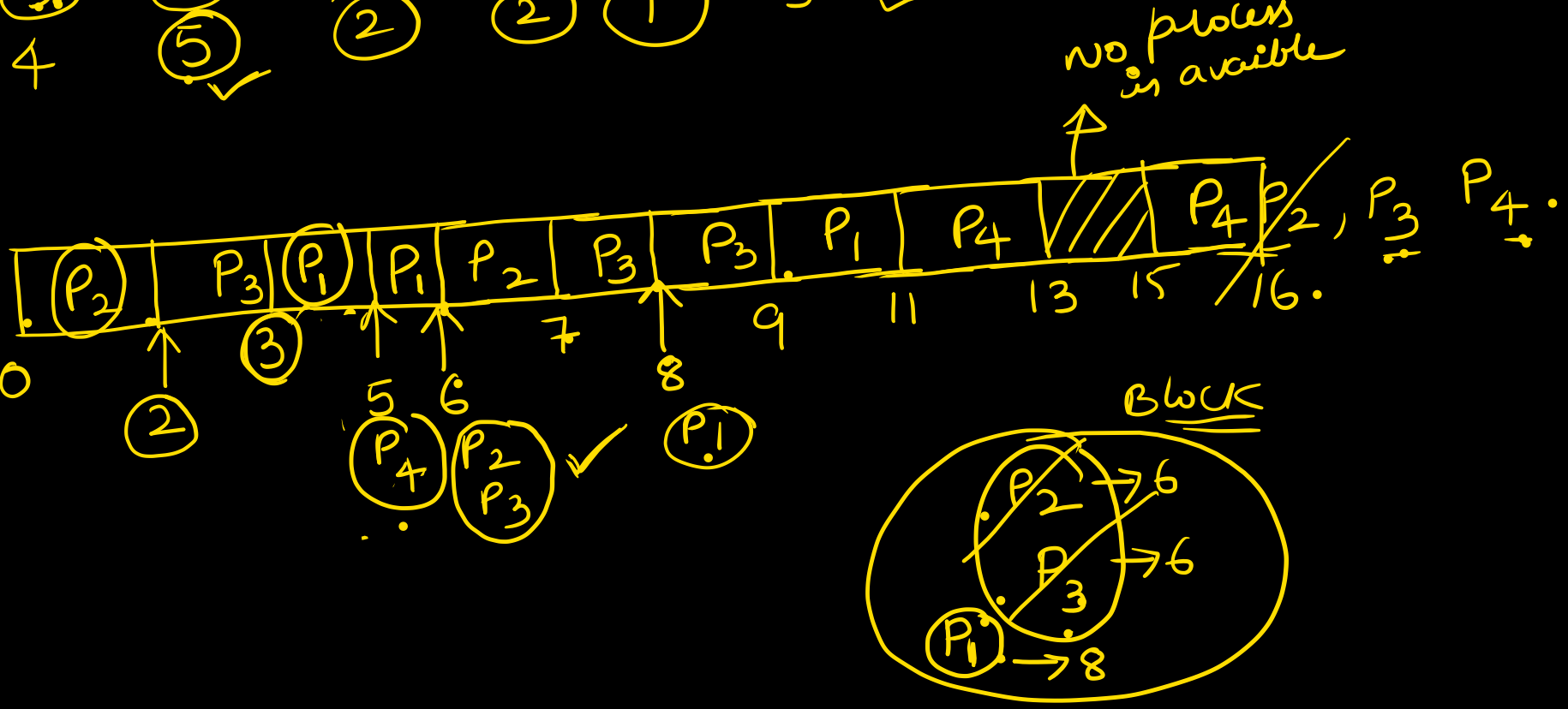
10 minutes do it

Then I will solve

Note: SRTF:

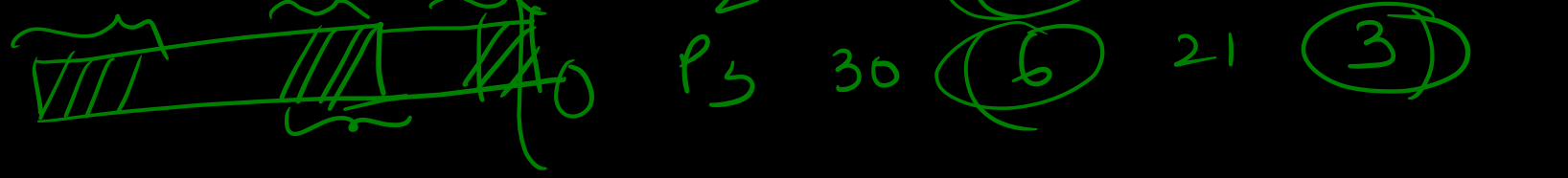
PNO	AT	BT	I/O	BT	TBT
1	0	3	10	2	5
2	0	2	0	1	3
3	2	4	0	2	6
4	5	2	2	1	3

CT of P_4?

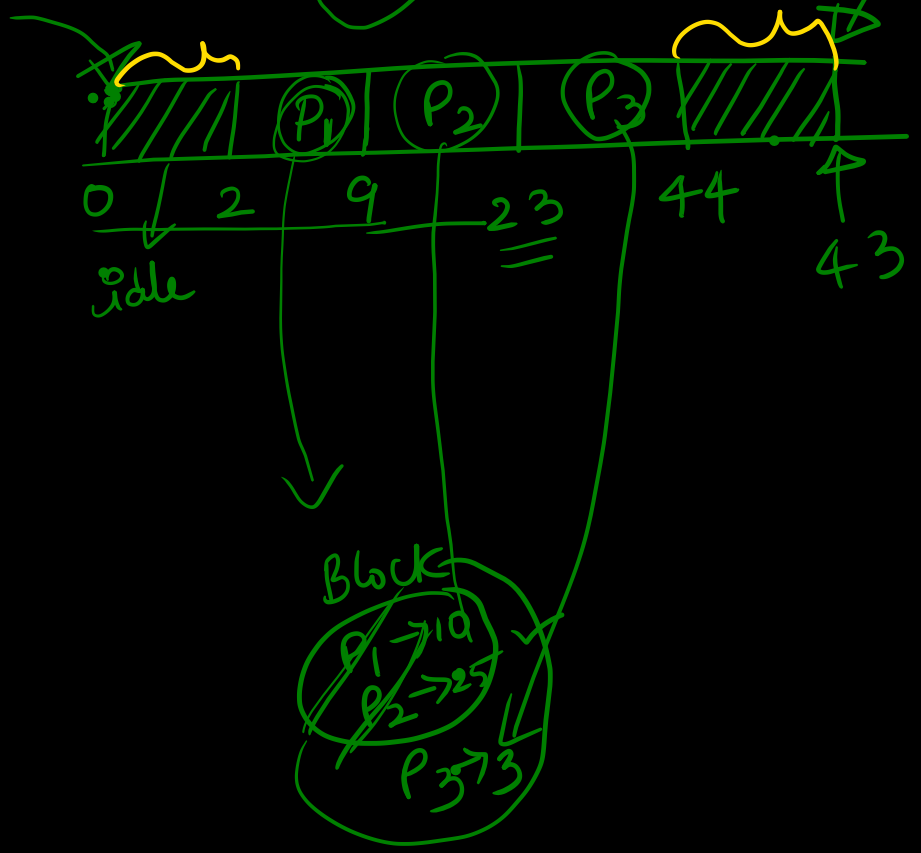


Gate: Consider 3 processes, all arriving at time zero (AT=0), with total execution time of 10, 20 and 30 units respectively. Each process spends the first 20% of execution time doing I/O, the next 70% time doing computation and the last 10% time doing I/O again. The OS uses a SRTF algo and schedules a new process either when the running process gets blocked on I/O or when the running process finishes its compute burst. Assume all I/O operations can be overlapped as much as possible. For what % of time does CPU remain idle?

	AT		BT	I/O	BT	I/O
P_1	0	10	2	7	1	
P_2	0	20	4	14	2	
P_3	0	30	6	21	3	



PNO	AT	IO	BT	I/O
1	0	2	10	1
2	0	4	14	2
3	0	6	21	3



At $t = 0$, we started I/O all 3 processes
At time 47 all 3 completed.

$$\%\ idle = \frac{5}{47} \times 100\%$$

$$= 10.64\%$$