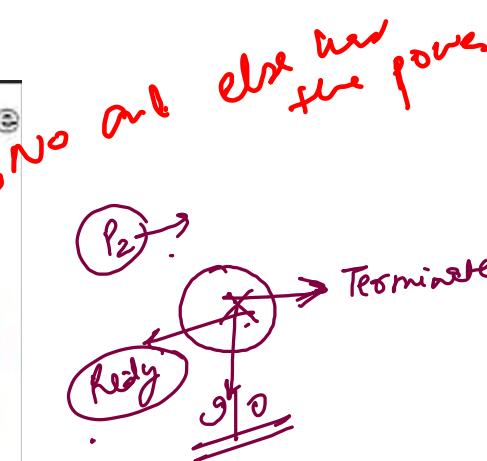


# Operating Systems

- 1) Pre-emptive and non pre-emptive
- 2) Starvation ✓
- 3) Convoy effect ✓
- 4) FCFS
- 5) SJF
- 6) SRTF

# Pre-emptive Scheduling

Sched wh is hairy power



✗

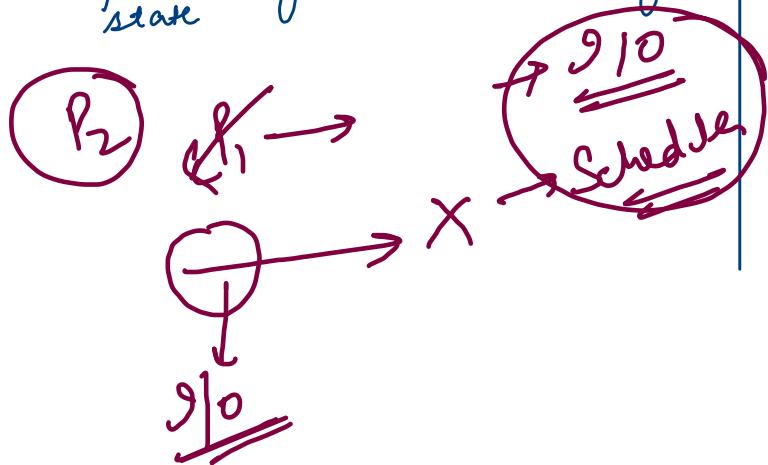
Pre-emptive



CPU is allocated to the process for a limited time.



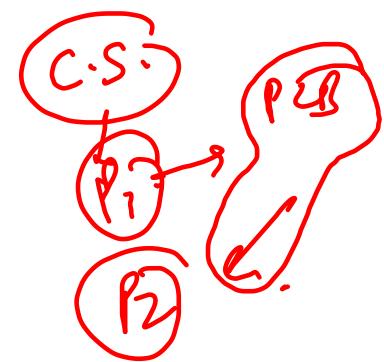
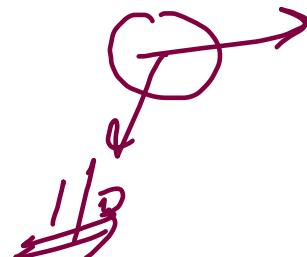
Overhead of switching the process from ready state to running state



Non pre-emptive ✗

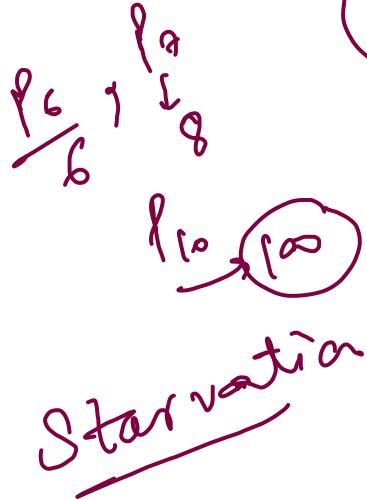
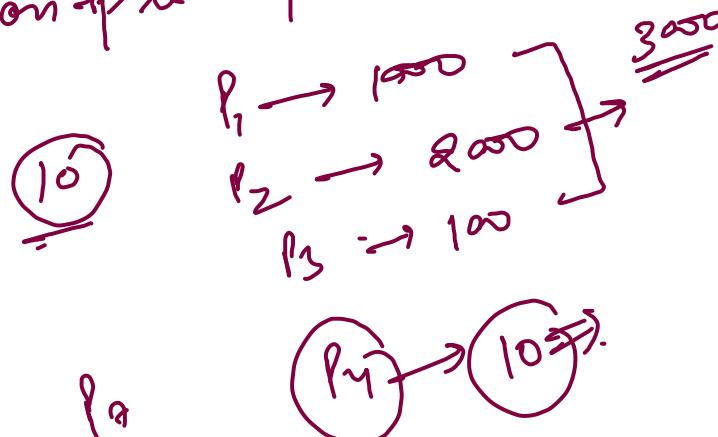
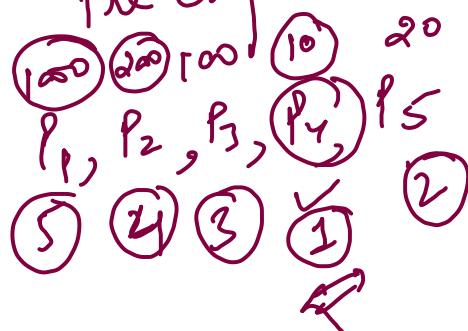
CPU is allocated to the process till it terminates or switches to waiting for state.

No overhead of switching the process from running state to ready state.

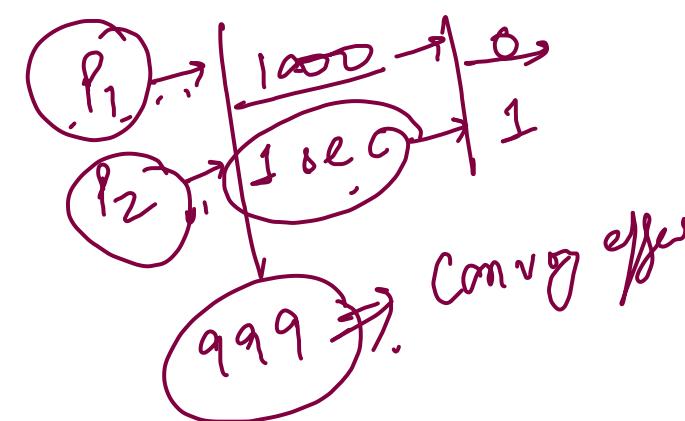


9/10

- 1) Starvation  $\rightarrow$  indefinite time
- 2) Convoy effect  $\rightarrow$  pack slow
- 3) Pre-emptive vs Non-pre-emptive

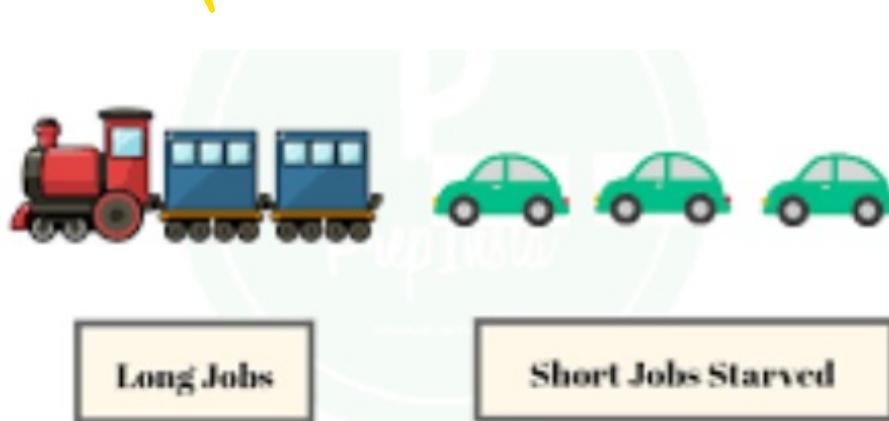


Starvation



Convoy effect

- 1) Starvation - When high priority processes keeps executing & low priority processes get blocked for indefinite time.
- 2) Convoy effect - when one slow process, slows down the performance of entire system.



FCFS

First come - first serve

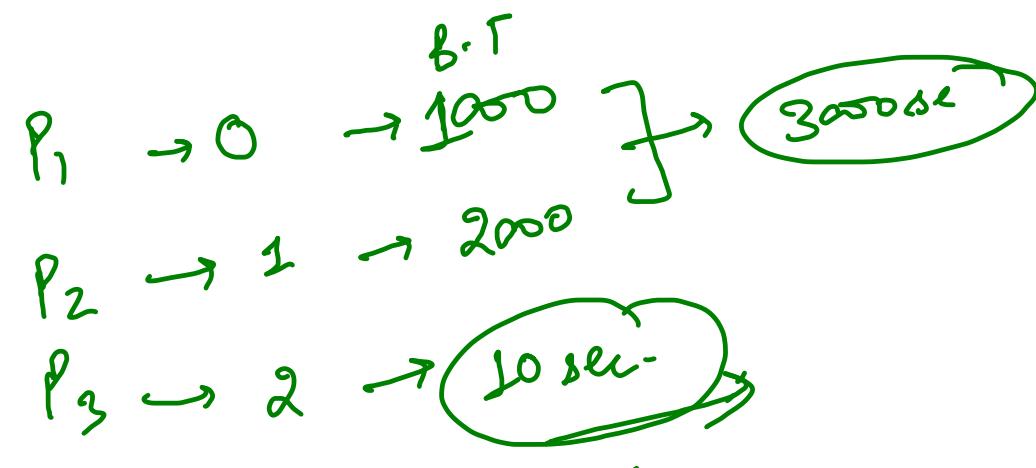
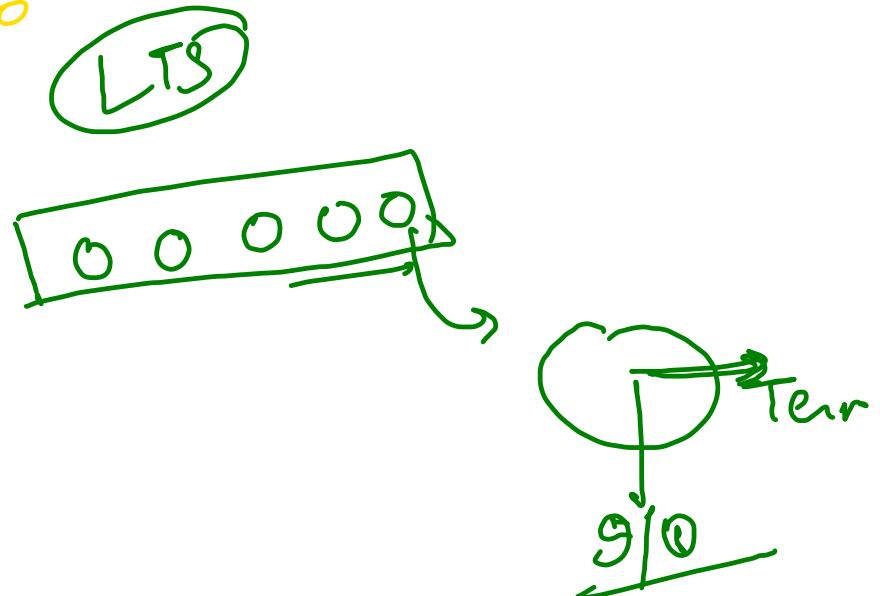
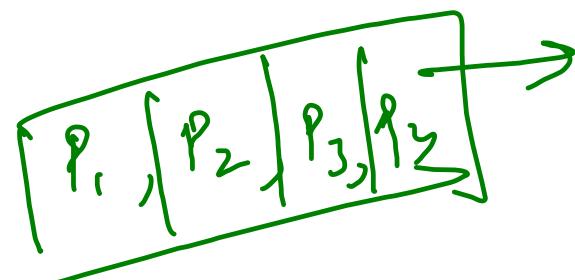
Pehle aao pehle pao!

- 1) FCFS → queue manner
- 2) Order of requests →
- 3) Non-preemptive  
terminate if op.

Adv ↗ Simple implementation

↗ No starvation

Dis ↗ Convoy effect  
↗ ↗



# Shortest Job First

"pehle chote chote kaam karlo fir bade kaam karenge"

- 1) Based on selection of smallest burst time of processes
- 2) Non-preemptive
- 3) Service time is <sup>not</sup> known before execution

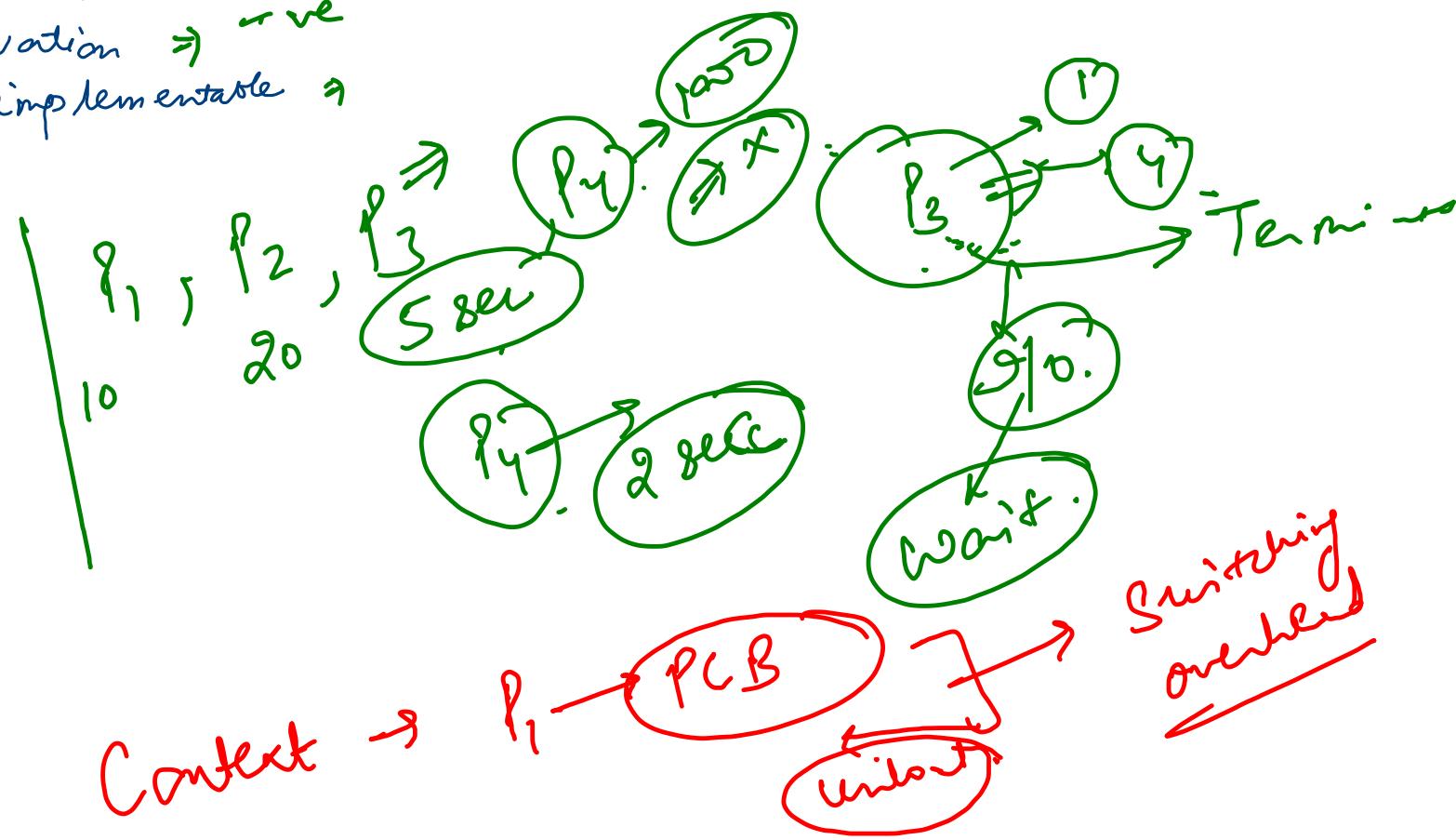
Burst time

Q.

Adv. No Convoy effect  $\Rightarrow$  +ve

Disadv. → Starvation  $\Rightarrow$  -ve  
Not implementable  $\Rightarrow$

SJF  
practical X  
B.T. X



## SRTF

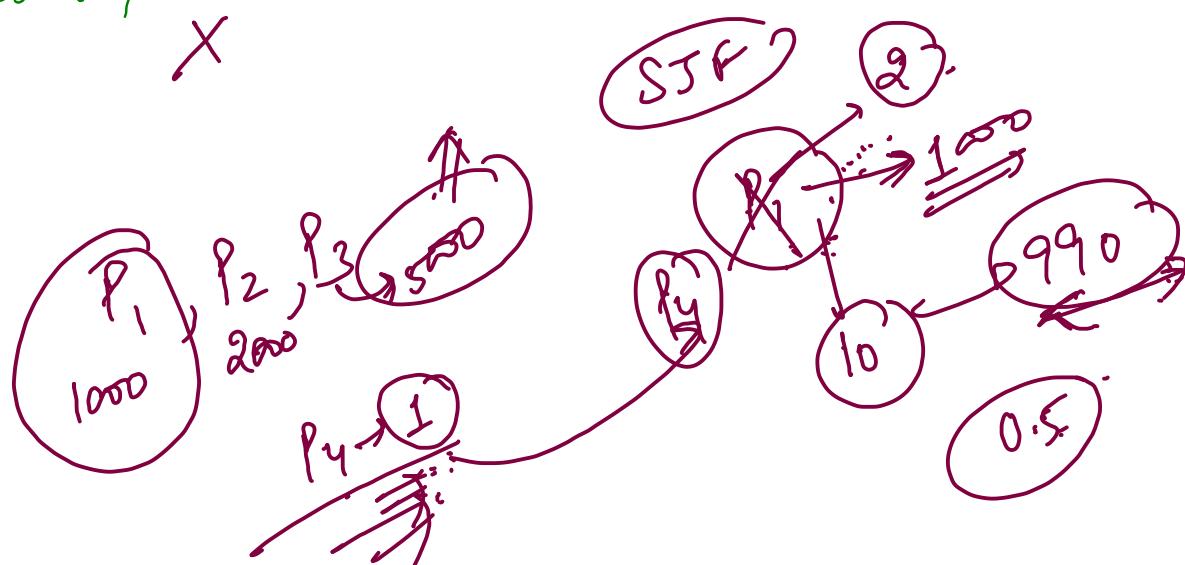
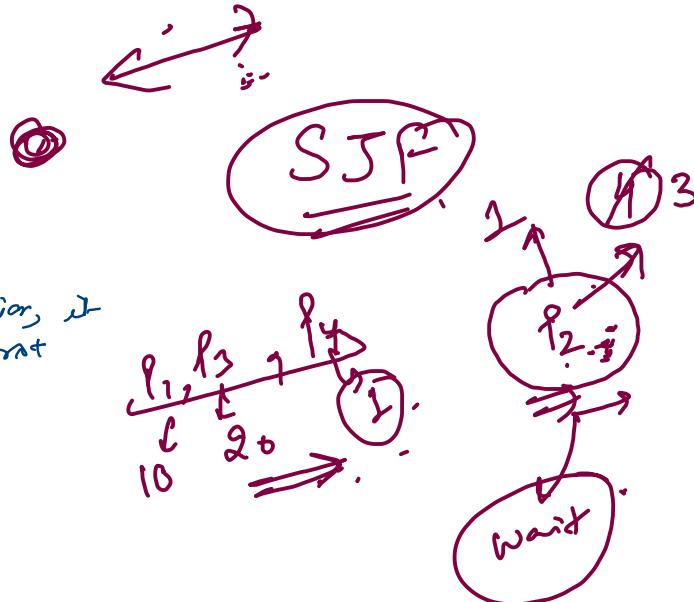
## Shortest Remaining Time First

- 1) Pre-emptive
  - 2) Similar to SJF, but is pre-emptive

If any smaller job comes during execution, it pre-empts & executes the smaller job first (compared to remaining time)

Adv - No convoy effect  $\Rightarrow$   
Avg Wait time  $\downarrow$

Disadv → Starvation ↗  
Not implementable



## SRTF

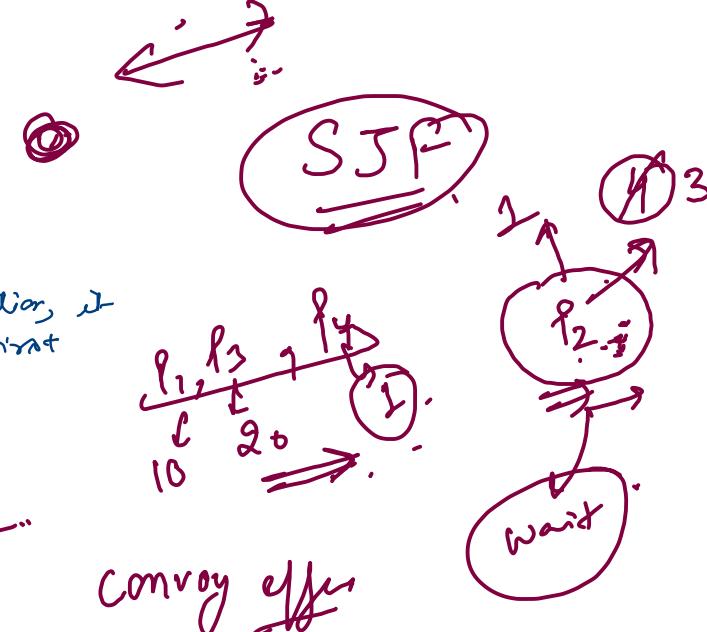
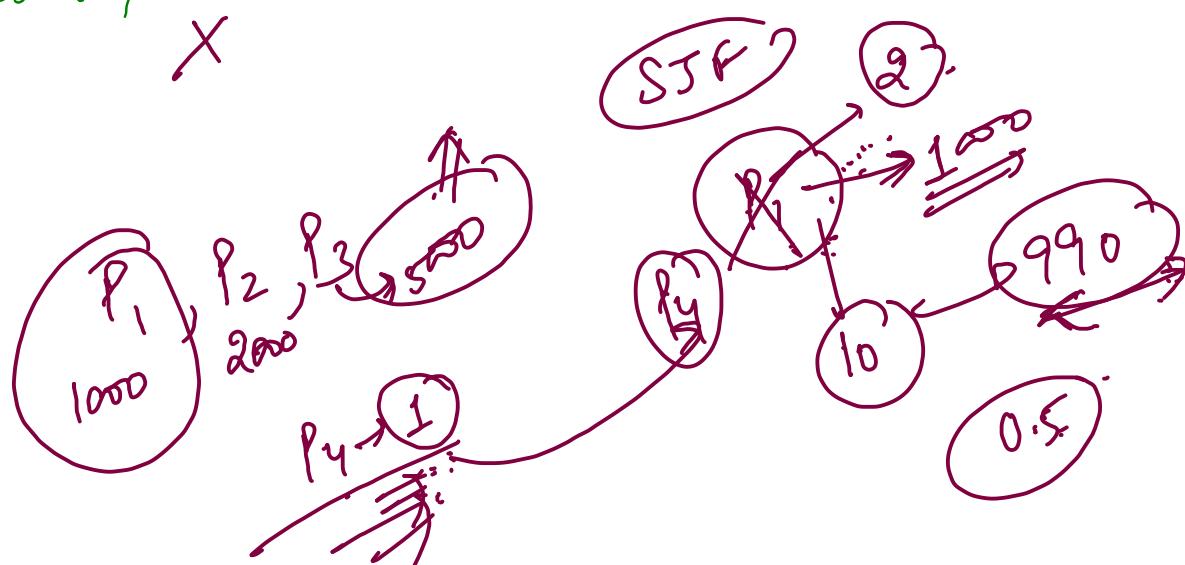
Shortest Remaining Time First

- 1) Pre-emptive
- 2) Similar to SJF, but is pre-emptive

If any smaller job comes during execution, it pre-empts & executes the smaller job first (compared to remaining time)

Adv - No convoy effect  $\Rightarrow$   
Avg Wait time  $\downarrow$

Dis adv  $\rightarrow$  Starvation  $\Rightarrow$   
Not implementable



FCFS  $\rightarrow$  N.P  
SJF  $\rightarrow$  N.P  
SRTF  $\rightarrow$  Q.

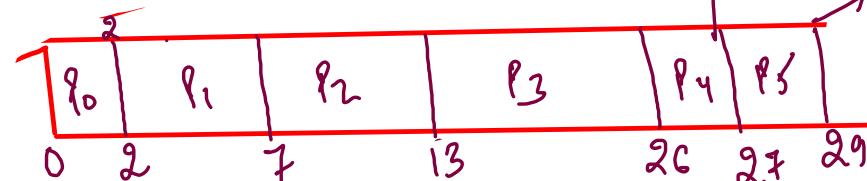
SX      C✓  
S✓      CX  
S✓      CX



Process	Arrival Time	Burst time	CT	TAT	WT
P <sub>0</sub>	0	2	2	2	0
P <sub>1</sub>	2	5	7	5	0
P <sub>2</sub>	4	6	13	9	3
P <sub>3</sub>	5	13	26	21	8
P <sub>4</sub>	6	1	27	21	20
P <sub>5</sub>	7	2	29	22	20

$$TAT = CT - AT$$

$$WT = TAT - BT$$



SJF

FCFS

Queue

%  
P<sub>1</sub>  
P<sub>2</sub>  
P<sub>3</sub>  
P<sub>4</sub>  
P<sub>5</sub>

3+8+20+20  
Avg " 51  
6

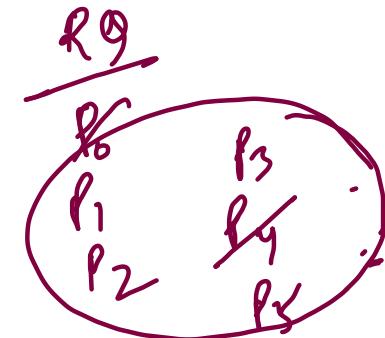
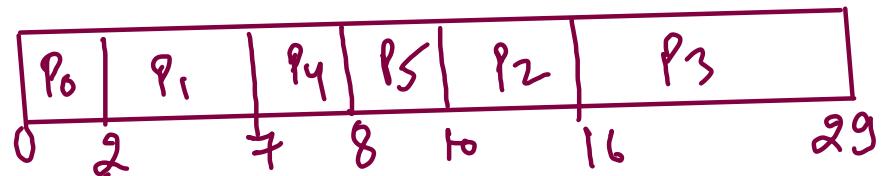
Avg TAT 8) 90  
6

→

Process	Arrival Time	Burst time	CT	TAT	WT
P <sub>0</sub>	0	2	2	2	0
P <sub>1</sub>	2	5	7	5	0
P <sub>2</sub>	4	6	14	12	6
P <sub>3</sub>	5	13	29	24	11
P <sub>4</sub>	6	1	8	2	1
P <sub>5</sub>	7	2	10	3	1
			48/6	= 19	6

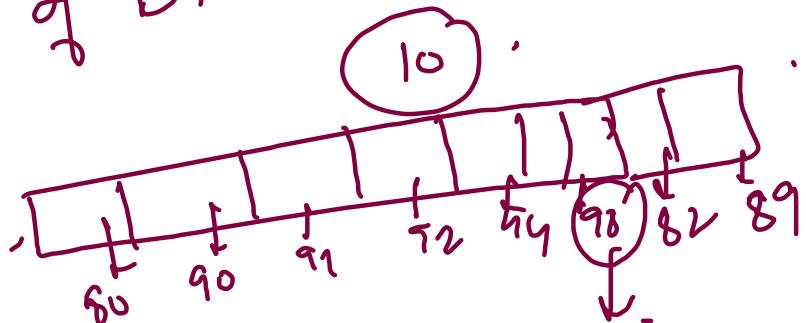
$$TAT = CT - AT$$

$$WT = TAT - BT$$



Comparing on the basis  
of BT

⇒ AT



11  
100

Process	Arrival Time	Burst-time
P <sub>0</sub>	0	2

Thread

Process

new resources are allocated

separate add.  
space

Context switching  
slower

doesn't effect

Independent

Thread

Share resources

Shared address space

Context switching  
faster

Blocking thread block  
process

Interdependent

Multiprogramming

Multi-tasking

Multi-Processing

Multi-threading

Process	Arrival Time	Burst time
P <sub>0</sub>	0	5
P <sub>1</sub>	1	2
P <sub>2</sub>	3	9
P <sub>3</sub>	9	6