OPERATING SYSTEMS (CS-3000)

Lecture-3

DR. JAISHREE MAYANK
ASSISTANT PROFESSOR
IIITDM KANCHEEPURAM

Objective of this lecture

- Performance Metrics for CPU Scheduling
- Types of CPU Scheduling Strategies

Multi-Programming

- We have many processes ready to run.
- **Degree of multiprogramming** – The number of processes that can reside in the ready state at maximum

CPU Scheduling

- Decide the order of execution of processes when multiple are competing for system resources
- **Where** – In the Ready State
- **Who(se) Responsibility –** Short Term Scheduler
- **When** - Wherever there is queuing for the CPU!
  - Ready to Running
- **Function** – allocate CPU to process
- **Pre-emption** – Process is forcefully removed from CPU.
- **Non pre-emption** – Processes are not removed until they complete the execution.

Scheduling Criteria

- Increase CPU Utilization
  - CPU should not be idle as minimum time as possible.
- Increase Throughput
  - complete as many processes as possible per unit time
- Minimize Average Waiting Time
  - Waiting time: The time duration taken by a process waiting in the ready queue.
- Minimize Average Turn Around Time
  - Turnaround time: the time taken for a single process from start to completion.
- Minimize Average Response time
  - Response time: the time taken from the point that when the process enters into the ready queue to the point when the process goes into the running state
- Fairness
  - ensure that each process is given a fair share of the CPU based on some particular policy

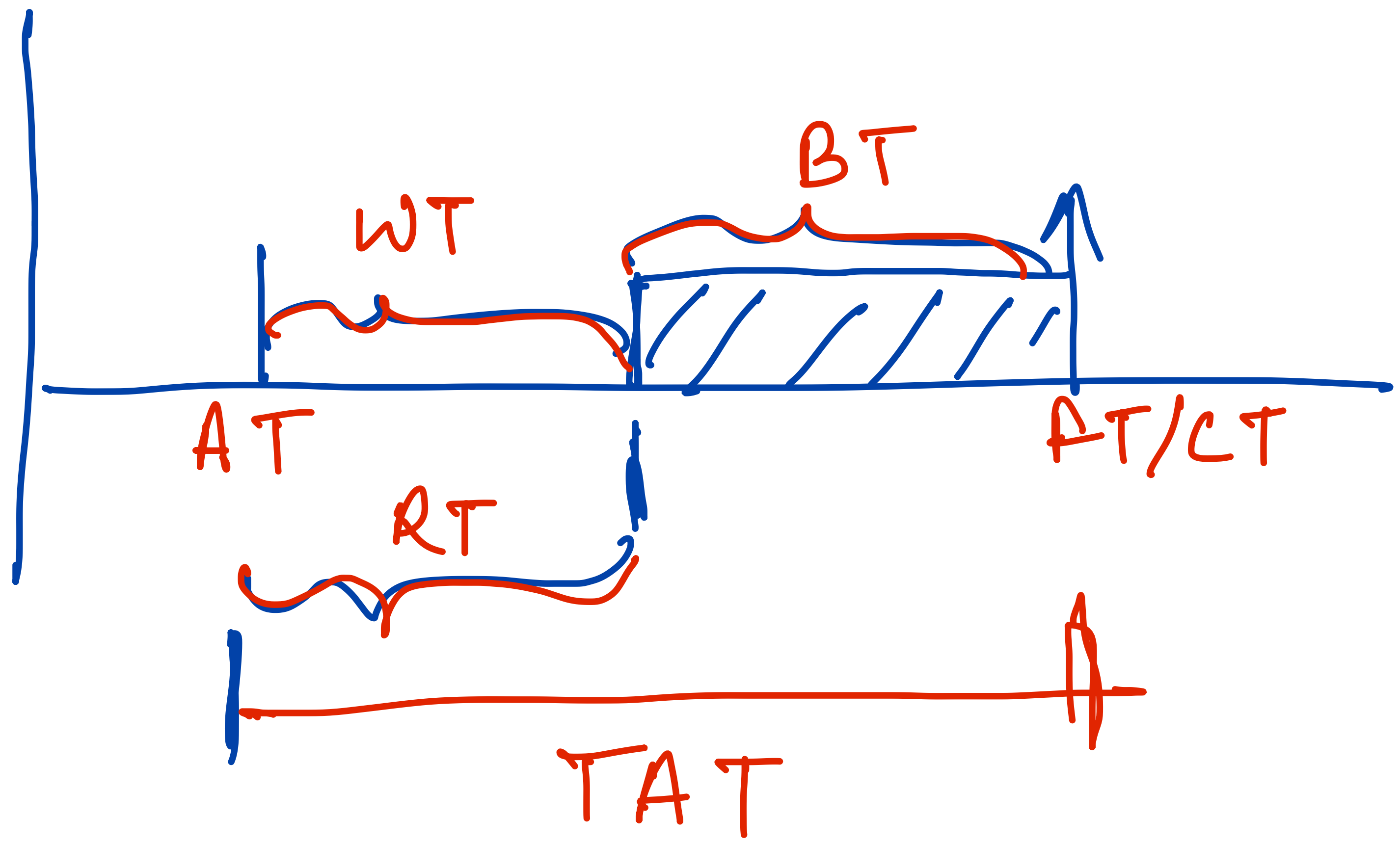
Types of CPU Scheduling

- **Non preemptive**
  - FCFS
  - SJF
  - Priority
  - HRRN

- **Preemptive**
  - SRT
  - RR
  - Priority

Different Time based Parameters w.r.t. Process

- **Arrival Time (AT)** - Time the Process comes to the Ready State
- **Burst Time (BT)** – Execution time of the process – also referred as Service Time
- **Completion Time (CT)** – Time at which process completes its execution.
- **Turn Around Time (TAT)** - Time required for an application (process) to give an output to the end user
  - **TAT = CT - AT**
- **Waiting Time (WT)** - Time Difference between turn around time and burst time.
  - **WT = TAT – BT**
- **Response Time** – Time for the System to Respond to Process or User (First Response time on System Clock)
  - Time Since the Request is Submitted (AT) and the First Response Time


BT
WT
AT
FT/CT
RT
TAT

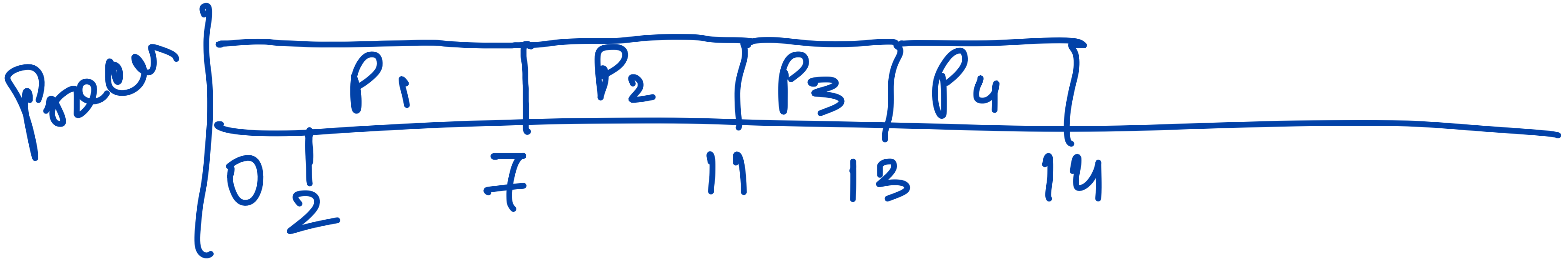
FCFS

PID/P#	AT	BT	CT/FT	TAT (CT-AT)	WT	RT
1	0	7	7	7	0	0
2	2	4	11	9	5	5
3	4	2	13	9	7	7
4	7	1	14	7	6	6
			Sum	32	18	18

Av. TAT = $\frac{32}{4} = 8$

A. WT = $\frac{18}{4} = 4.5$

A. RT = $\frac{18}{4} = 4.5$



GANTT CHART

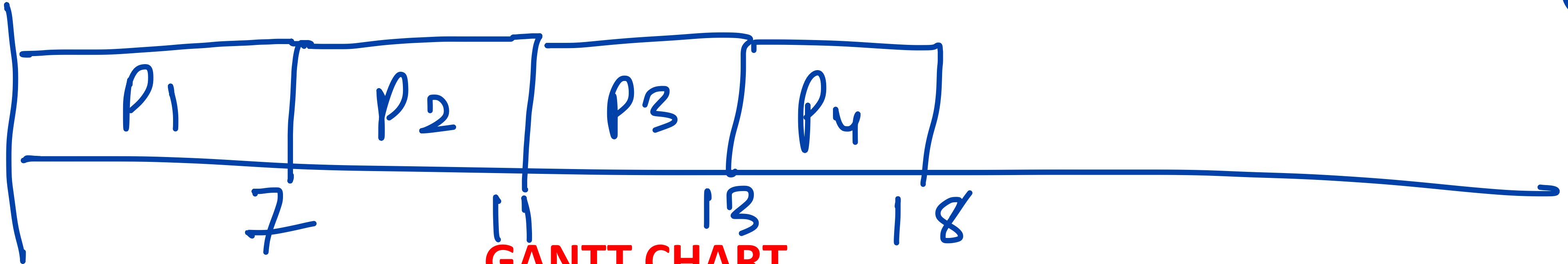
FCFS

PID/P#	AT	BT	CT/FT	TAT	WT	RT
1	0	7	7	7	0	0
2	0	4	11	11	7	7
3	0	2	13	13	11	11
4	0	5	18	18	13	13
			Sum	49	31	31

$$\text{Av TAT} = \frac{49}{4} = 12.25$$

$$\text{Av. WT} = \frac{31}{4} = 7.75$$

$$\text{Av. RT} = \frac{31}{4} = 7.75$$



GANTT CHART

Advantages and Disadvantages

- **Advantages**
  - Simple
  - Fair
- **Disadvantages**
  - Waiting time depends on arrival order
  - Convoy effect

SJF

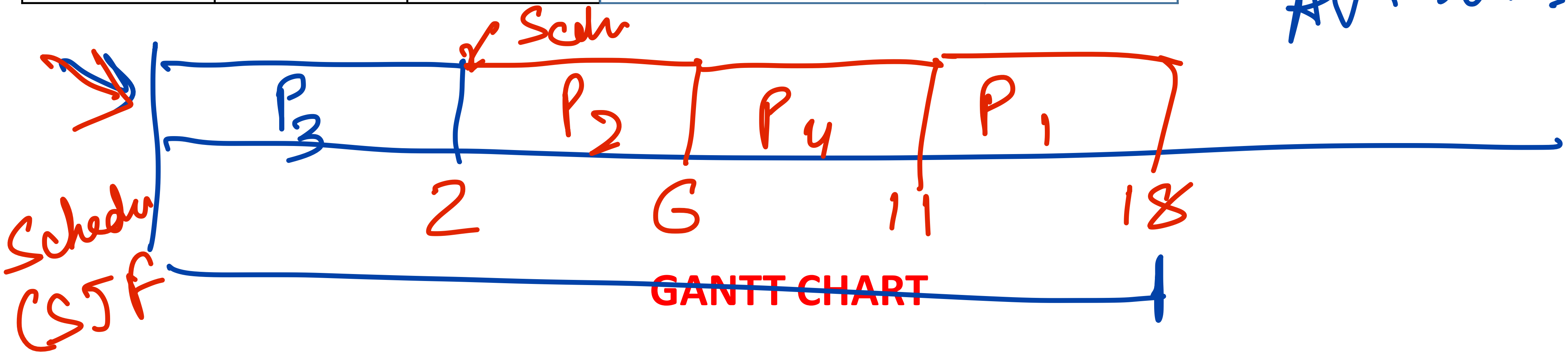
PID/P#	AT	BT	CT/FT	TAT	WT
1	0	7	18	18	11
2	0	4	6	6	2
3	0	2	2	2	0
4	0	5	11	11	6
			Sum	37	19

P1 | P2 | P3 | P4

Ready Queue

$$\text{Av. TAT} = \frac{37}{4} = 9.25$$

$$\text{Av. WT} = \frac{19}{4} = 4.75$$



SJF

PID/P#	AT	BT	CT/FT	TAT	WT	RT
1	0	7	7	7	0	0
2	2	4	14	12	8	8
3	4	2	10	6	4	4
4	7	1	8	1	0	0
						$\frac{12}{4} = 3$

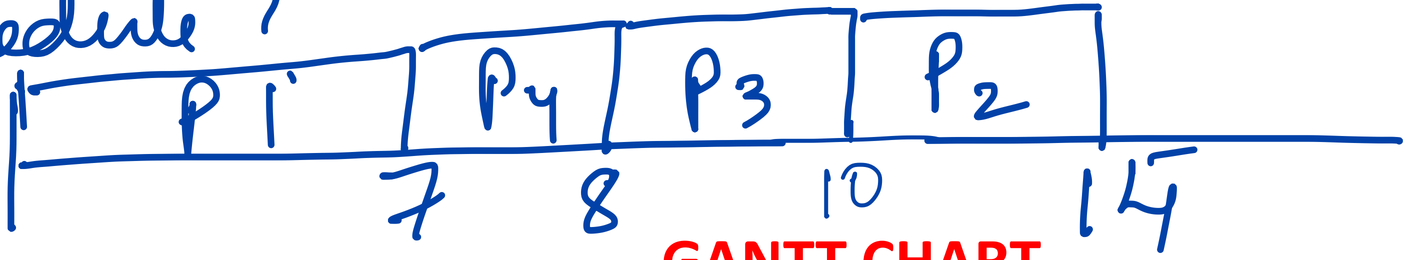
Av. TAT = 6.5

Av. WT = 3

Av RT = 3

Compare with FCFS

Schedule?



GANTT CHART

Advantages and Disadvantages

- **Advantages**
  - Optimal: Min avg. waiting time and response time

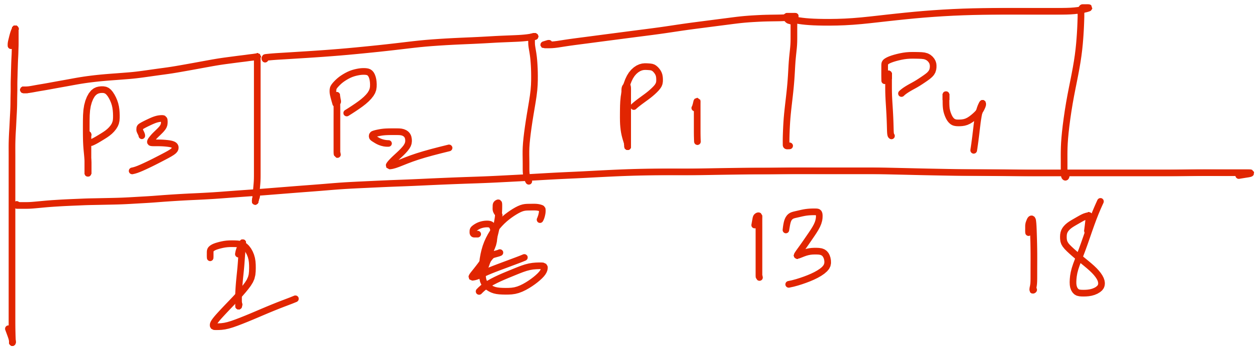
- **Disadvantages**
  - Difficult to predict burst time
  - Starvation

Priority based Non-Preemptive Scheduling

PID/P#	Priority	AT	BT	CT/FT	TAT	WT
1	3	0	7			
2	2	0	4			
3	1	0	2			
4	4	0	5			

→ first based on priority

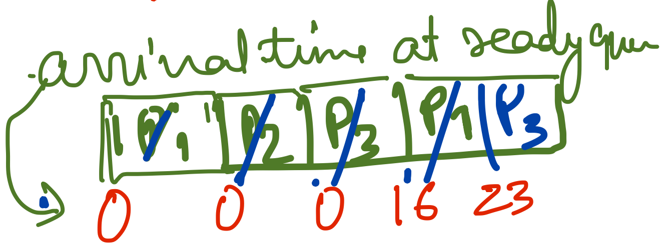
→ for same priority processes, use FCFS

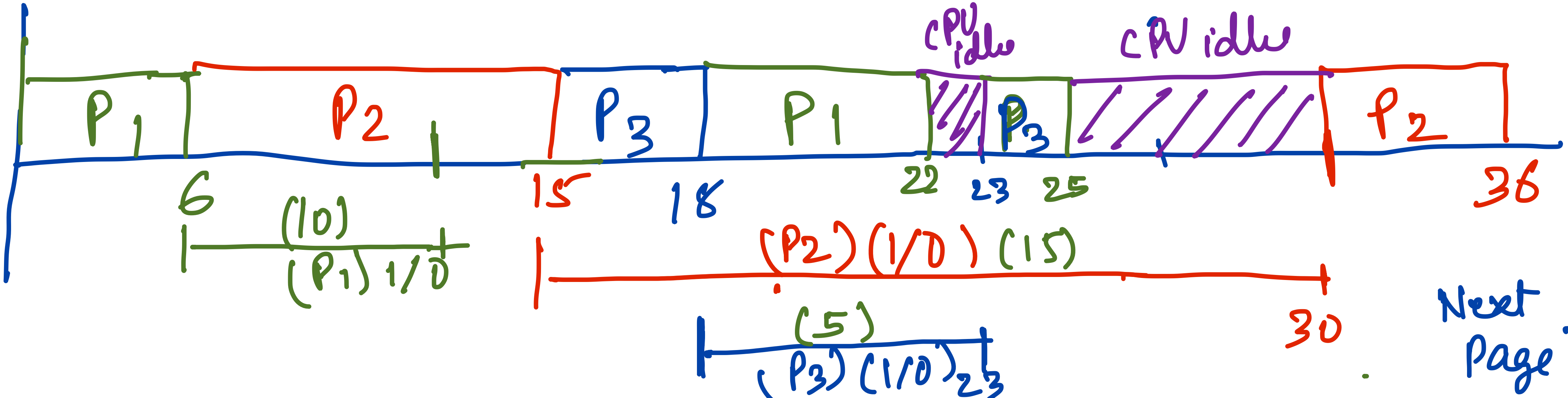


GANTT CHART

	Process time	AT	30% CPU	50% I/O	20% CPU	CT	TAT	WT	RT
P_1	20	0	6	10	4	22	22	2	0
P_2	30	0	9	15	6	36	36	6	6
P_3	10	0	3	5	2	25	25	15	15

Apply FCFS, (Assume each processes requested different I/O device)





Next Page →

CPU utilization $= \frac{30}{36} \times 100$

Throughput $= \frac{\text{no. of proc}}{\max(CT) - \min(AT)} = \frac{3}{36}$

How to estimate the Burst time?

- Static
  - Process size
  - Process type
- Dynamic
  - linear averaging
  - Exponential averaging

* Static will not given proper estimation

Linear Averaging

If $P_1, P_2, \ldots P_n$ → processes has completed

$t_1, \ldots \ldots t_n$ → actual burst time after completion

Expected Burst time

$$\tau_{n+1} = \frac{1}{n}\left(\sum_{i=1}^{n} t_i\right)$$

t_i ⇒ actual burst time of ith process

τ_i → Predicted / Expected burst time

Exponential Averaging

$P_1 \ldots \ldots P_n \rightarrow$ Process

$t_1 \ldots \ldots t_n \rightarrow$ actual burst time

$\tau_1 \ldots \tau_n \rightarrow$ predicted

for any nth process

weighting factor.

$$\tau_{(n+1)} = \alpha t_n + (1-\alpha)\tau_n$$

$\tau_{(n+1)}$: predicted of $(n+1)^{th}$ process

t_n: actual for nth

τ_n: predicted of n^{th}

Example

	P1	P2	P3	P4
t_i	4	8	6	7

→ It is given to verify our approach (actual burst time)

Given
$\alpha = 0.5$, $\tau_1 = 10$

Find τ_2, τ_3, τ_4.

$\tau_2 = 0.5 \times \underset{(t_1)}{4} + (1-0.5) \times \underset{(\tau_1)}{10} = 7$

$\tau_3 = 0.5 \times \underset{(t_2)}{8} + 0.5 \times \underset{(\tau_2)}{7} = 7.5$

$\tau_4 = 0.5 \times \underset{(t_3)}{6} + 0.5 \times \underset{(\tau_3)}{7.5} = 6.75$

Note :-

* Exponential averaging may predict burst time accurately sometimes,
* 'α' plays a major role for predicting the burst time.
* deciding 'α' is not easy, it depends on many factors.