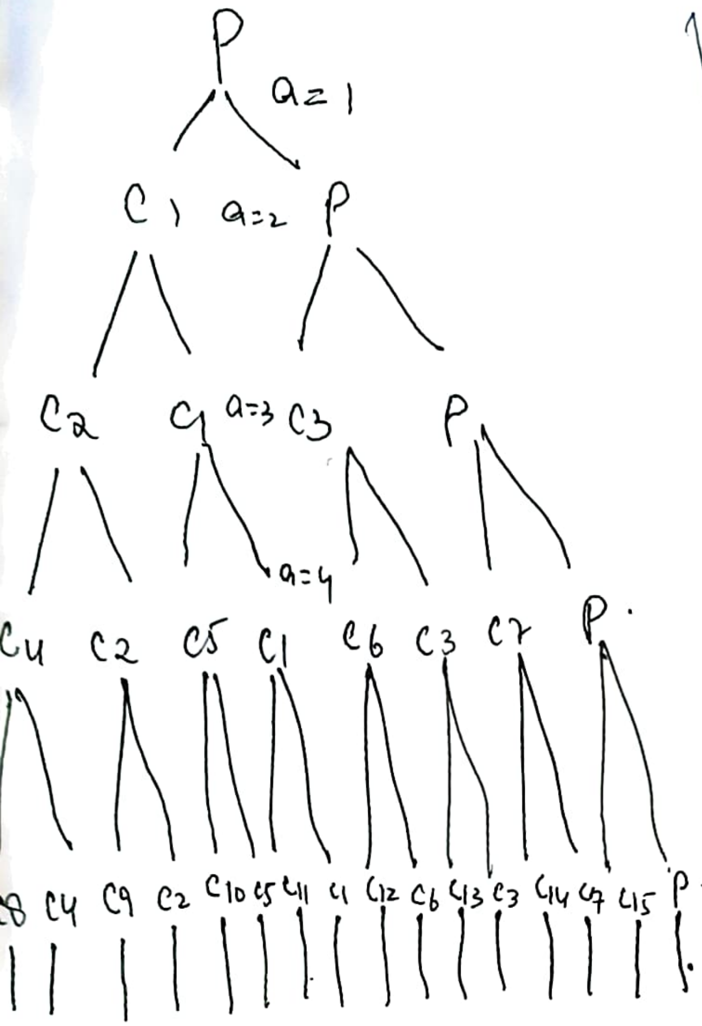
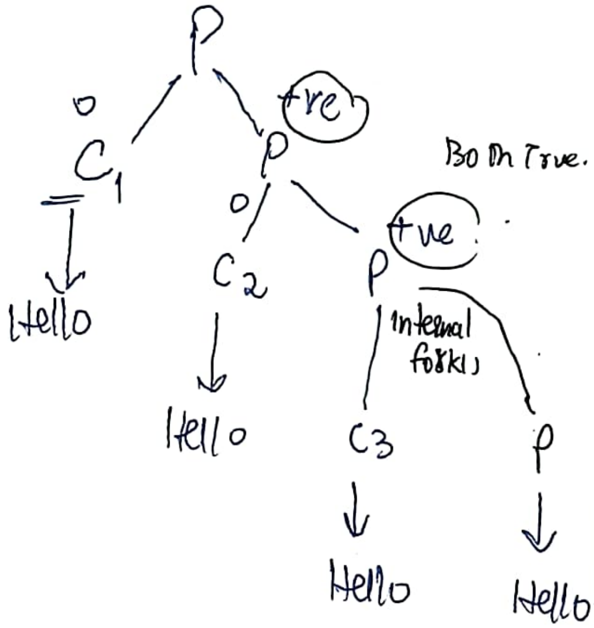
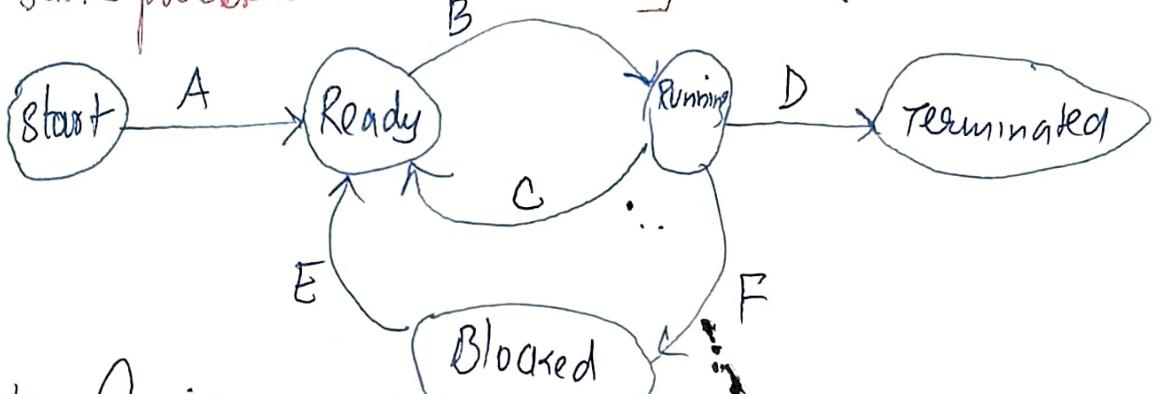



```
if( fork() && fork())
    fork();
printf("Hello");
```



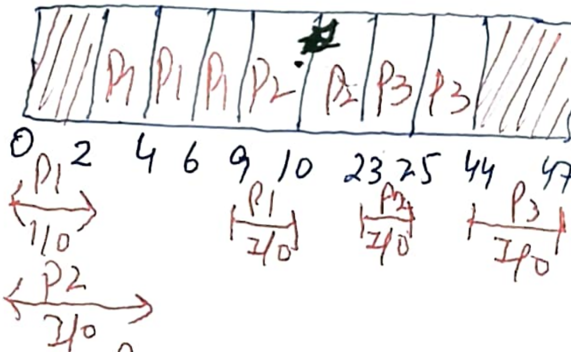
Q1- In the following Process state transition diagram for a Uniprocessor system, assume that there are always some processes in the Ready state:

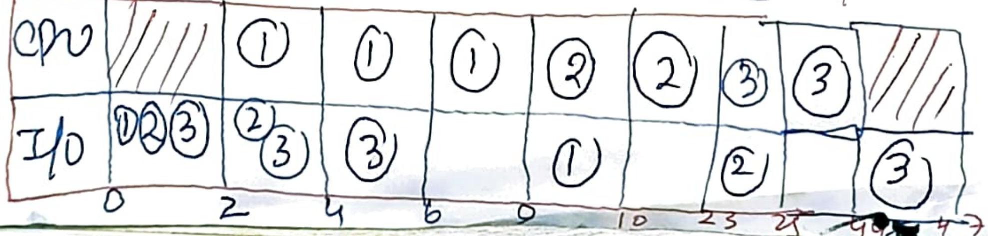


Now, Consider the following Statements:

Q. Consider 3 processes, all arriving at time 0, with BT of 10, 20, and 30 units respectively. Each Process spends the first 20% of exec. time doing I/O, the next 70% of time doing Computation, and the last 10% of time doing I/O again. The OS uses SRTF scheduling. Assume that all I/O operations can be overlapped as much as possible. For what %age of time does the CPU remain idle?

PNO	AT	BT	I/O	CPU	I/O
P1	0	10	2	7	1
P2	0	20	4	14	2
P3	0	30	6	21	3





Cpu idle time = 2+ / 47

= 10.638%

Multilevel Queue

P NO	AT	BT	Queue No
P1	0	4	1
P2	0	3	1
P3	0	8	2
P4	10	5	2

P1	P2	P1	P2	P3	P4	P3

0 2 4 6 7 10 15 20

highest Priority → Real-time Processes → foreground
→ System Processes → &
→ Interactive Processes → Background
lowest Priority → Batch Processes → Processes

(Multilevel Queue scheduling)

NOTE: Queue 1 has higher Priority than Q2. while FCFS is used in Queue2.
RR is used in Q1 (TQ=2),

Multilevel Feedback Scheduling (Aging) prevents Starvation

→ Quantum=2 → Q_0 = Highest Priority = RR ⇒ TQ = 2
↓
Quantum=4 → Q_1 = RR ⇒ TQ = 4
↓
FCFS → Q_2 = lowest Priority = FCFS

P NO	AT	BT
P_1	0	8
P_2	1	4
P_3	2	9
P_4	3	5

Gantt Chart

P_1	P_2	P_3	P_4	P_1	P_2	P_3	P_4	P_1	P_3

0 2 4 6 8 12 14 18 21 23 26

Operation on Processes

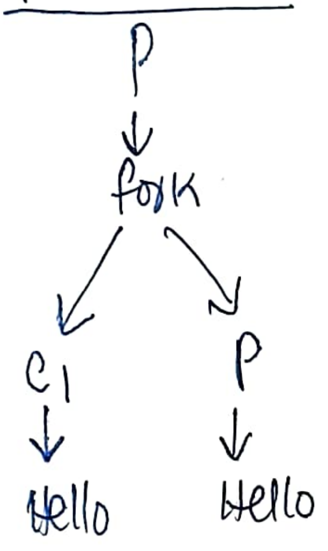
↳ Creation

↳ Termination

fork() — 0 child
— +1 > 0 parent
— -1 < 0 error

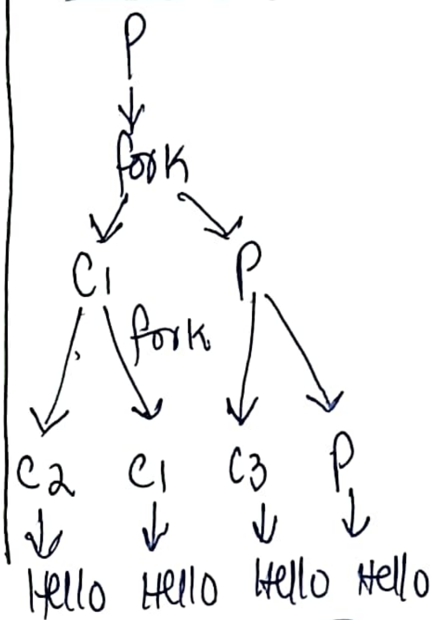
Ex1:
```
main() {
    fork();
    printf("Hello");
}
```

Process Tree



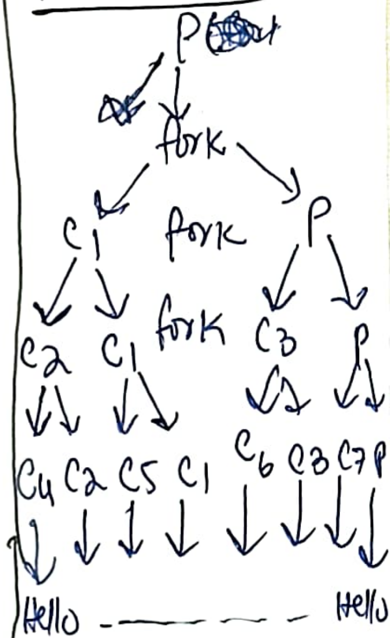
Ex2:
```
main(){
    fork();
    fork();
    printf("Hello");
}
```

Process Tree



Ex3:
```
main()
{
    fork();
    fork();
    fork();
    printf("Hello");
}
```

Process Tree



NOTE: ① No. of Hello statements printed = 2^n

② No. of child processes created is $2^n - 1$

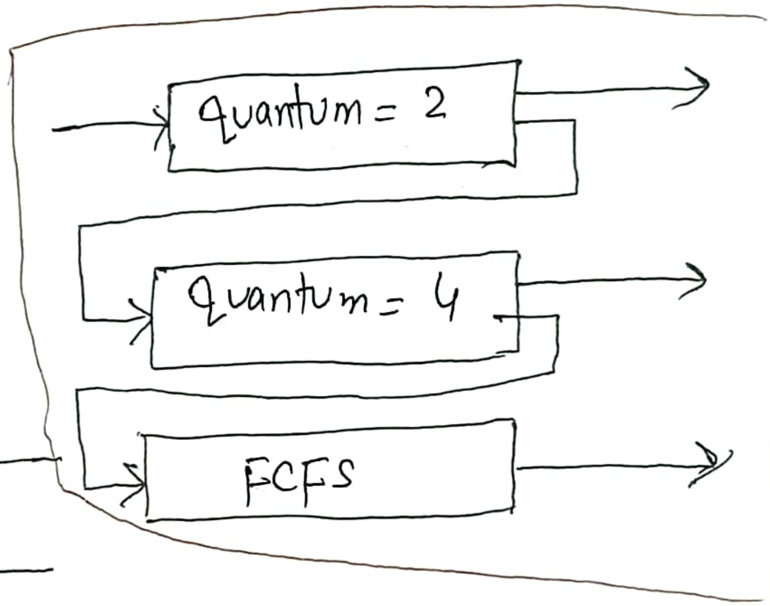
Zombie Process

Orphan Process

Process Information

Process	AT	BT	
P1	0	8	3
P2	1	4	× 1
P3	2	9	4
P4	3	5	× 2

Q_0 (Highest Priority) = RR TQ=2
Q_1 = RR TQ=4
Q_2 Lowest P = FCFS



Gantt Chart

P1	P2	P3	P4	P1	P2	P3

0 2 4 6 8 12 14 18

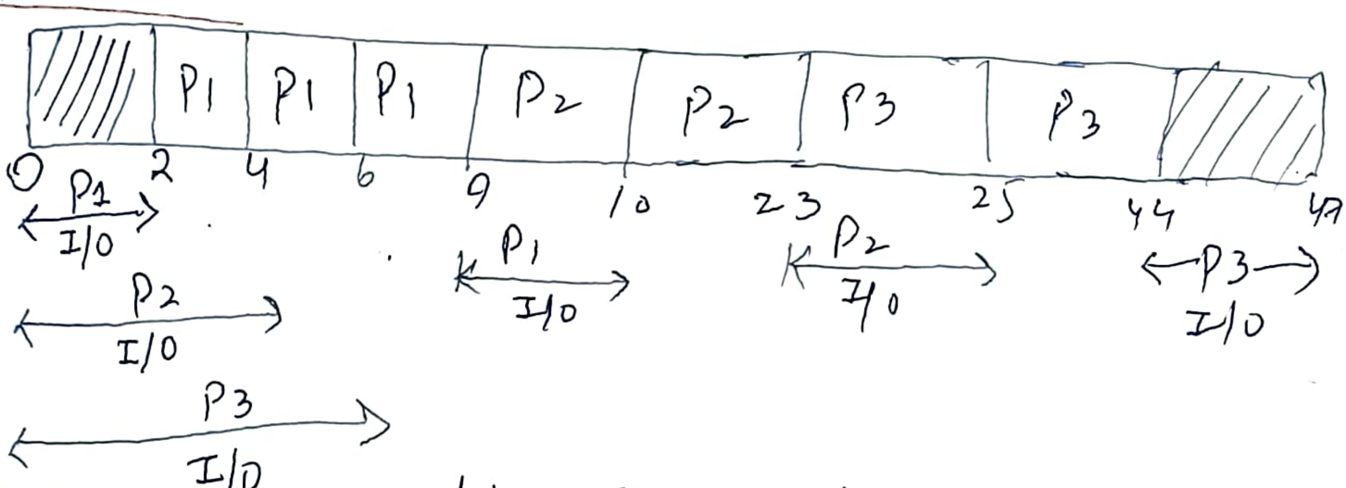
P4	P1	P3

18 21 24 27

Q:-

PNO	AT	BT	I/O	CPU Burst	I/O Burst
P1	0	10	2	7	1
P2	0	20	4	14	2
P3	0	30	6	21	3

Gantt Chart



∴ Percentage of time CPU remains idle $= \frac{5 \times 100}{47}$

$= 10.638\%$

Round Robin :- Criteria: "Time Quantum" ; Mode: "Preemptive"

TQ = 2

Process No	AT	BT	CT	TAT	WT	RT
P1	0	5	12	12	7	0
P2	1	4	11	10	6	1
P3	2	2	6	4	2	2
P4	4	1	9	5	4	4

Ready Queue

~~P1~~	~~P2~~	~~P3~~	~~P1~~	~~P4~~	P2	P1	

0

Running Queue (Gantt chart)

P1	P2	P3	P1	P4	P2	P1

0 2 4 6 8 9 11 12

Pre-emptive Priority Scheduling

Criteria : "Priority"
Mode : "Preemptive"

NOTE: Higher the no. Higher the Priority

Priority	Process No	AT	BT	CT	TAT	WT
10	P1	0	5	12	12	7
20	P2	1	4	8	7	3
30	P3	2	2	4	2	0
40	P4	4	1	5	1	0

Gantt Chart

P1	P2	P3	P3	P4	P2	P1

0 1 2 3 4 5 8 12

Non-preemptive Priority Scheduling

Mix BT (CPU & I/O Both)

Mode: Pre-emptive Criteria: Priority based
Find CT of P1, P2, P3, P4

Process	AT	Priority	CPU	I/O	CPU	CT
P1	0	2	1	5	3	10
P2	2	3	3	3	1	15
P3	3	1 H	2	3	1	9
P4	3	4	2	4	1	18

NOTE:- Lower the no., higher the Priority

P1		P2	P3	P2	P1	P1	P3	P1	P2	P4		P2	

0 1 2 3 5 6 7 8 9 10 11 13 14 15 17

P1 I/O

P2 I/O

P3 I/O

P4 I/O

-- P4 18

Categories of Scheduling Algos.

Based on the environments viz
1. Batch 2. Interactive 3. Real Time

↳ Scheduling in Batch Systems: FCFS, SJF, SRTF
↳ Scheduling in Interactive Systems: Priority, RR, Multiple Queues, SRTF, Guaranteed scheduling, Lottery scheduling, Fair-share scheduling.

↳ Scheduling in Real-time Systems:
- Hard Real Time
- Soft Real Time (video whose audio isn't properly synced)

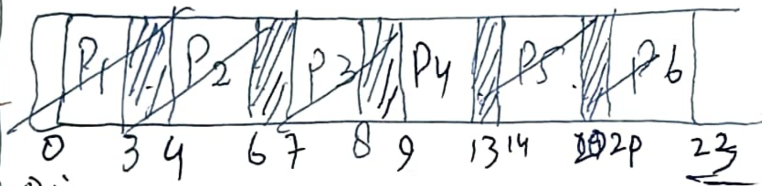
Rate monotonic, Earliest Deadline First.
- Static scheduling
- Dynamic scheduling

02/Sep/2025

Q:- Consider the set of 6 Processes whose AT and BT are given below. If the CPU Scheduling Policy is FCFS & there is 1 unit of overhead in Scheduling the Processes, find the efficiency of the algorithm?

PNO	AT	BT	CT	TAT	WT
P1	0	3			
P2	1	2			
P3	2	1			
P4	3	4			
P5	4	5			
P6	5	2			

Gantt chart



o is a

Now, Useless Time or Wasted Time $= 6 \times \delta = 6 \times 1 = 6$ unit

Total Time $= 23$ unit

Useful time $= 23 - 6 = 17$ unit

$\therefore$ Efficiency$(\eta) = \frac{\text{Useful Time}}{\text{Total Time}} = \frac{17}{23} = 0.7391 = 73.91\%$

NOTE: (1) RR Scheduling is FCFS Scheduling with Preemptive mode.

(2) When Time Quantum tends to infinity, RR Scheduling becomes FCFS Scheduling

(3) The waiting time for the process having the highest priority will always be Zero in Preemptive mode

---

NOTE: → Convoy Effect

→ Starvation

→ Aging

02 Sep 2025

Longest Job First Algorithm

Process NO	AT	BT
1	0	3
2	1	2
3	2	4
4	3	5
5	4	6

Longest remaining Time first Algo

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

	P1	P2	P3	P4	P4	P4	P3	P4	P3	P4	P2	P3	P4	P2	P3

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

P4	P1	P2	P3	P4

17 18 19 20 21

Highest Response Ratio Next (HRRN)

Criteria: Response Ratio (RR) $= \frac{W+S}{S}$

W = Waiting time for a process so far

S = Serivice time or BT

→ HRRN not only favours shorter jobs but also limits the waiting time of longer jobs.

Mode: non-preemptive

PNO	AT	BT	CT	TAT	WT	RT
0	0	3				
1	2	6				
2	4	4				
3	6	5				
4	8	2				

Gantt Chart

P0	P1	P2	P4	P3

0 3 9 13 15 20

$RR2 = \frac{5+4}{4} = 2.25$ ✓

$RR3 = \frac{3+5}{5} = 1.6$

$RR4 = \frac{1+2}{2} = 1.5$

Now, again

$RR3 = \frac{7+5}{5} = \frac{12}{5} = 2.4$

$RR4 = \frac{5+2}{2} = \frac{7}{2} = 3.5$

Scheduling Algorithms

Pre-emptive
- → SRTF
- → LRTF
- → RR
- → Priority Based
- → MLQ

Non pre-emptive
- → FCFS
- → SJF
- → LJF
- → HRRN
- → ~~Multilevel Queue~~
- → Priority Based

② SJF Criteria: Burst Time | Mode: "Non-Preemptive"

Process No	AT	BT	CT	TAT	WT	RT
P_1	1	3	6	5	2	2
P_2	2	4	10	8	4	4
P_3	1	2	3	2	0	0
P_4	4	4	14	10	6	6

Gantt Chart

	P_3	P_1	P_2	P_4

0 1 3 6 10 14

SRTF

④

Process	AT	BT
P_1	0	7
P_2	1	4
P_3	2	8

The Gantt chart for Preemptive SJF scheduling algo is?

Gantt chart

P_1	P_2	P_2	P_1	P_3

0 1 2 5 11 19

③ SRTF : Criteria: "BT" | mode: "Preemptive"

Process No	AT	BT	CT	TAT	WT	RT
P1	0	~~5~~ 4	9	9	4	0
P2	1	~~3~~ 2 1	4	3	0	0
P3	2	4	13	11	7	7
P4	4	1	5	1	0	0

Ganttchart (SRTF)

P1	P2	P2	P2	P4	P1	P3

0 1 2 3 4 5 9 13

RT = cpu first Time − AT

WT = TAT − BT

① FCFS (Criteira: ~~Burst~~ Arrival Time) mode: Non-Preempt

Process No	AT	BT or Exec. Time	CT	TAT	WT	RT
P1	0	2	2	2	0	0
P2	1	2	4	3	1	1
P3	5	3	8	3	0	0
P4	6	4	12	6	2	2

GC

P1	P2		P3	P4

0 2 4 5 8 12