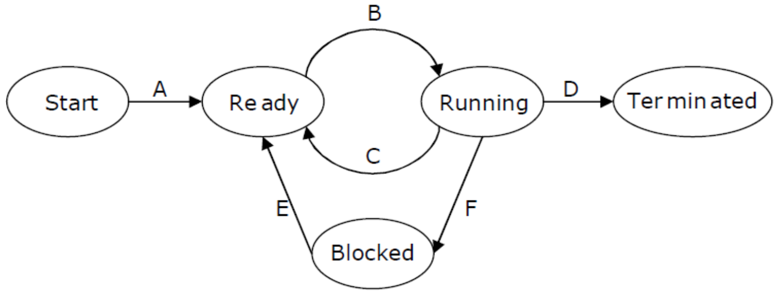
1. In the following process state transition diagram for a uniprocessor system, assume that there are always some processes in the ready state: Now consider the following statements:



I. If a process makes a transition D, it would result in another process making transition A immediately.
II. A process P2 in blocked state can make transition E while another process P1 is in running state.
III. The OS uses preemptive scheduling.
IV. The OS uses non-preemptive scheduling.

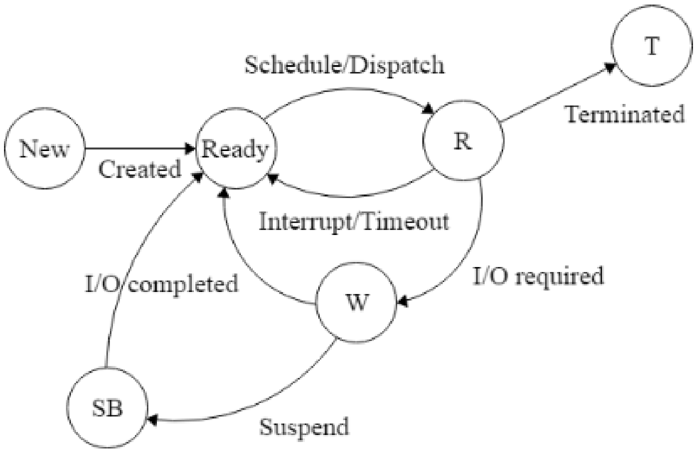
Which of the above statements are TRUE?

(A) I and II
(B) I and III
(C) II and III
(D) II and IV

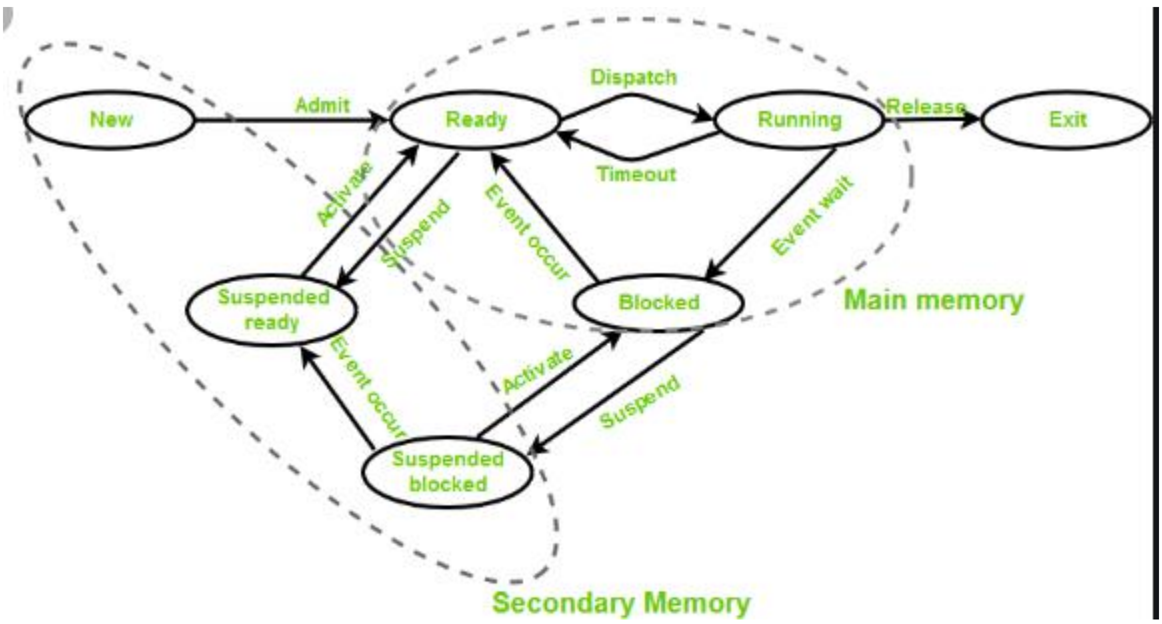
Answer (C)

I is false. If a process makes a transition D, it would result in another process making transition B, not A.
II is true. A process can move to ready state when I/O completes irrespective of other process being in running state or not.
III is true because there is a transition from running to ready state.
IV is false as the OS uses preemptive scheduling.

2. **Discuss six and seven states of process. How it overcomes the shortcoming of the five-state process model?**



Six states



Seven states

3. **For a system with 'n' processors, maximum how many processes can be in the**

A. Ready state ----- No limit
B. Blocked state ----- No limit
C. Running state ------ n

4. **Indicate whether the following transitions are possible directly, also mention the reasons for those transitions**

a. Running to Ready
b. Running to blocked
c. Blocked to Running
d. Running to Terminated

Ans:

a. Possible, when a process's time quantum expires/high priority process comes.

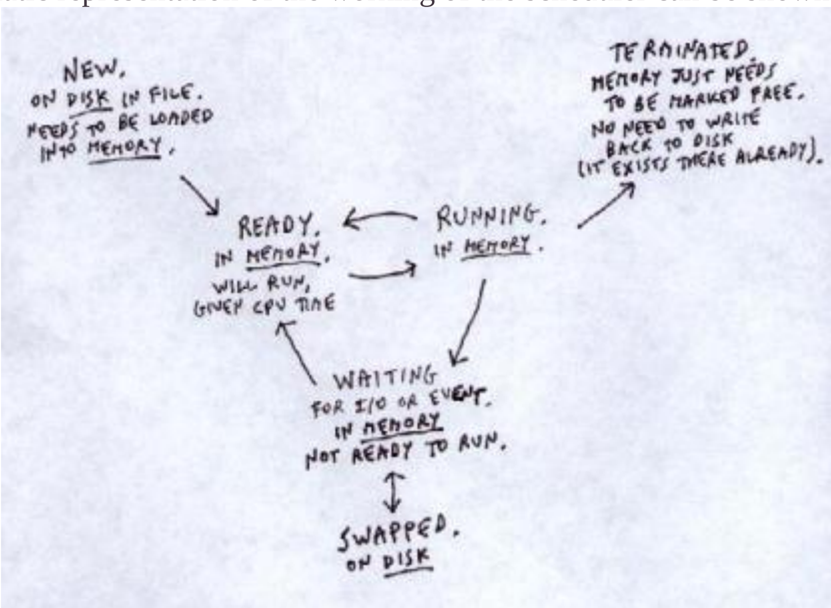
b. Possible, when a process issues an I/O request

c. Not possible

d. Possible, when a process completes successfully.

5. How we can diagrammatically represent the working of scheduler in process states? Explain the term swap-in and swap-out?

The diagrammatic representation of the working of the scheduler can be shown as-



Model of scheduler

Waiting for running– Short term scheduler.
Swap-in and Swap-out-Middle term scheduler.

Swap-in and swap-out:
Swap-out is a process of swapping a process out from the ready queue to a backing store when a process of higher priority arrives and claims for the CPU while **Swap-in** is a process where swapped out process is transferred back to the ready queue from the backing store.

6. Explain Dispatcher. What is the difference between dispatcher and schedular?

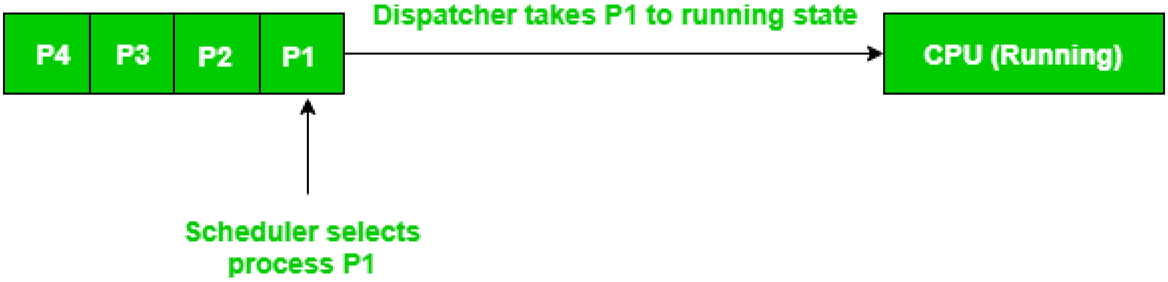
A dispatcher is a special program which comes into play after the scheduler. When the scheduler completes its job of selecting a process, it is the dispatcher which takes that process to the desired state/queue. The dispatcher is the module that gives a process control over the CPU after it has been selected by the short-term scheduler. This function involves the following:

- Switching context

- Switching to user mode
- Jumping to the proper location in the user program to restart that program

The Difference between the Scheduler and Dispatcher –

Consider a situation, where various processes are residing in the ready queue waiting to be executed. The CPU cannot execute all of these processes simultaneously, so the operating system has to choose a particular process on the basis of the scheduling algorithm used. So, this procedure of selecting a process among various processes is done by **the scheduler**. Once the scheduler has selected a process from the queue, the **dispatcher** comes into the picture, and it is the dispatcher who takes that process from the ready queue and moves it into the running state. Therefore, the scheduler gives the dispatcher an ordered list of processes which the dispatcher moves to the CPU over time.



7. **Consider the processes P1, P2, P3, P4 given in the below table, arrives for execution in the same order, with Arrival Time 0, and given Burst Time**

Process	Burst time
P1	21
P2	3
P3	6
P4	2

   a. **Find the average waiting time and average burst time using the FCFS scheduling algorithm.**
   b. **Find the average waiting time and average burst time using the SJF non-pre-emptive scheduling algorithm.**

FCFS without arrival time

PID	Burst time	Waiting time	Turnaround time
P1	21	0	21
P2	3	21	24
P3	6	24	30
P4	2	30	32

Gantt Chart

P1	P2	P3	P4

0   21   24   30   32

Total Waiting time = 0 + 21 + 24 + 30 = 75 ms

Average " " = 75/4 = 18.75 ms

Total Turn Around Time = 21 + 24 + 30 + 32 = 107

Average " " " = 107/4 = 26.75 ms

Non-pre-emptive SJF without arrival time

Gantt

P4	P2	P3	P1

0   2   5   11   32

PID	Burst time	Waiting time	turnaround time
P1	21	11	32
P2	3	2	5
P3	6	5	11
P4	2	0	2

Total Waiting time = 11 + 2 + 5 + 0 = 18

Average " " = 18/4 = 4.5 ms

Total Turnaround time = 32 + 5 + 11 + 2 = 50

Average " " = 50/4 = 12.5

8. **Consider the processes P1, P2, P3, P4 given in the below table, arrives for execution in the same order, with Arrival time, and given Burst time, find the average waiting time and burst time using the FCFS scheduling algorithm.**

Process	Arrival time	Burst time
P1	0	5
P2	2	3
P3	6	2
P4	7	3

FCFS with arrival time

Q1.

Process ID	Arrival Time (milliseconds)	Burst time (milliseconds)
P_1	0	5
P_2	2	3
P_3	6	2
P_4	7	3

Gantt Chart

P_1	P_2	P_3	P_4
0 – 5	5 – 8	8 – 10	10 – 13

ID	Arrival Time	Burst time	Completion time	Turn Around Time	Waiting Time
P_1	0	5	5	5	0
P_2	2	3	8	6	3
P_3	6	2	10	4	2
P_4	7	3	13	6	3

Total Turn Around Time - 5+6+4+6 = 21 ms

Average " " " = 21/4 = $\boxed{5.25 \text{ ms}}$

Total Waiting time = 0+3+2+3 = 8

Average " " = 8/4 = $\boxed{2 \text{ ms}}$

9. **Consider the processes P1, P2, P3, P4 given in the below table, arrives for execution in the same order, with Arrival time, and given Burst time, find the average waiting time and burst time using the pre-emptive scheduling algorithm.**

Process	Arrival time	Burst time
P1	0	7
P2	2	3
P3	3	1
P4	5	4

Solution:

PID	AT	BT	WT	TAT
P1	0	7	8	15
P2	2	3	1	4
P3	3	1	0	1
P4	5	4	1	5

Average WT = 10/4 = 2.5 ms

Average TAT = 25/4 = 6.25 ms