Chapter: CPU/ Process Scheduling

Basic Concept

- CPU Scheduling is basis of Multi-programmed OS
- Objective of Multi-programming is to have some processes running all the time to maximize CPU Utilization.

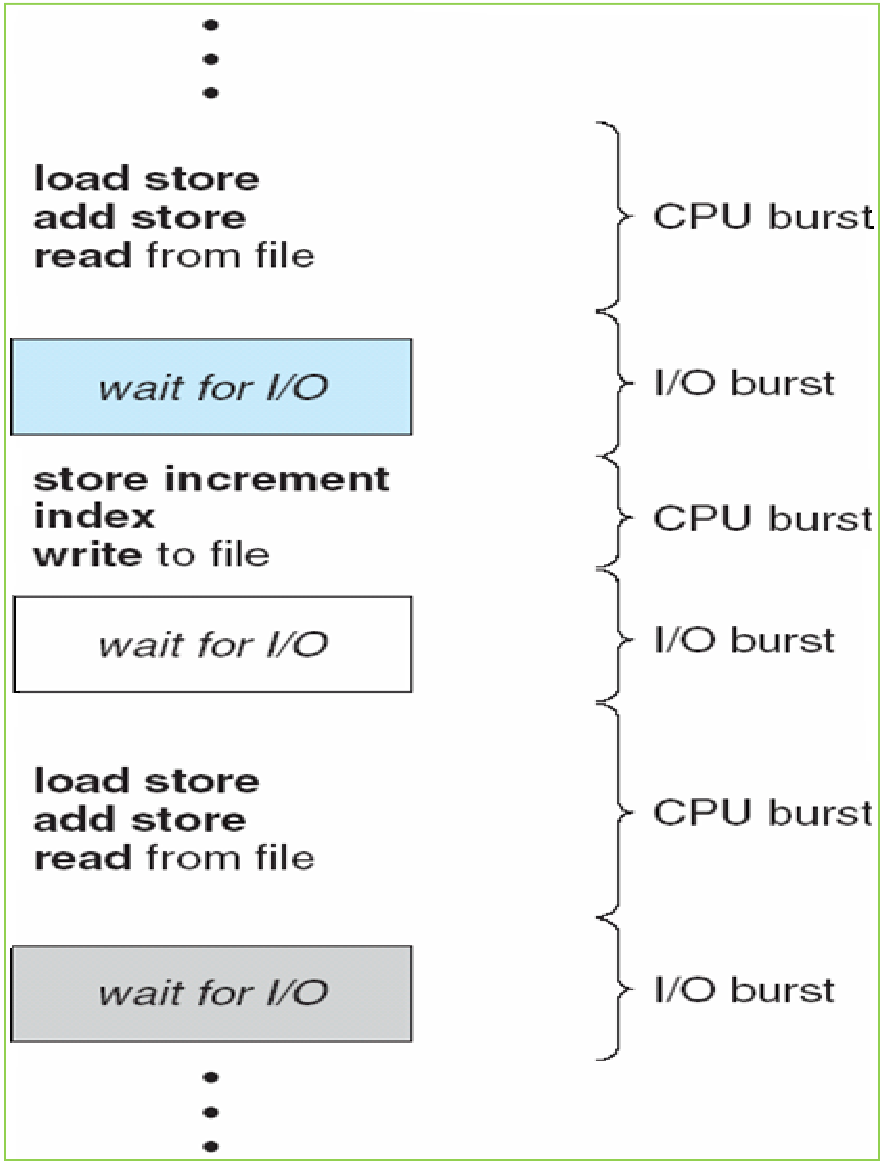
Sequence of CPU and I/O Bursts

- **CPU Burst-** when process is executed in CPU

- **CPU Burst Time:** The amount of **time** the **process uses the processor**
- Types of **CPU bursts**:
  - Long **bursts** -- process is **CPU** bound (spend max. time with CPU
  - Short **bursts** -- process is I/O bound (spend max. time with I/O)

- **I/O Burst-** when CPU is waiting for I/O completion for further execution.

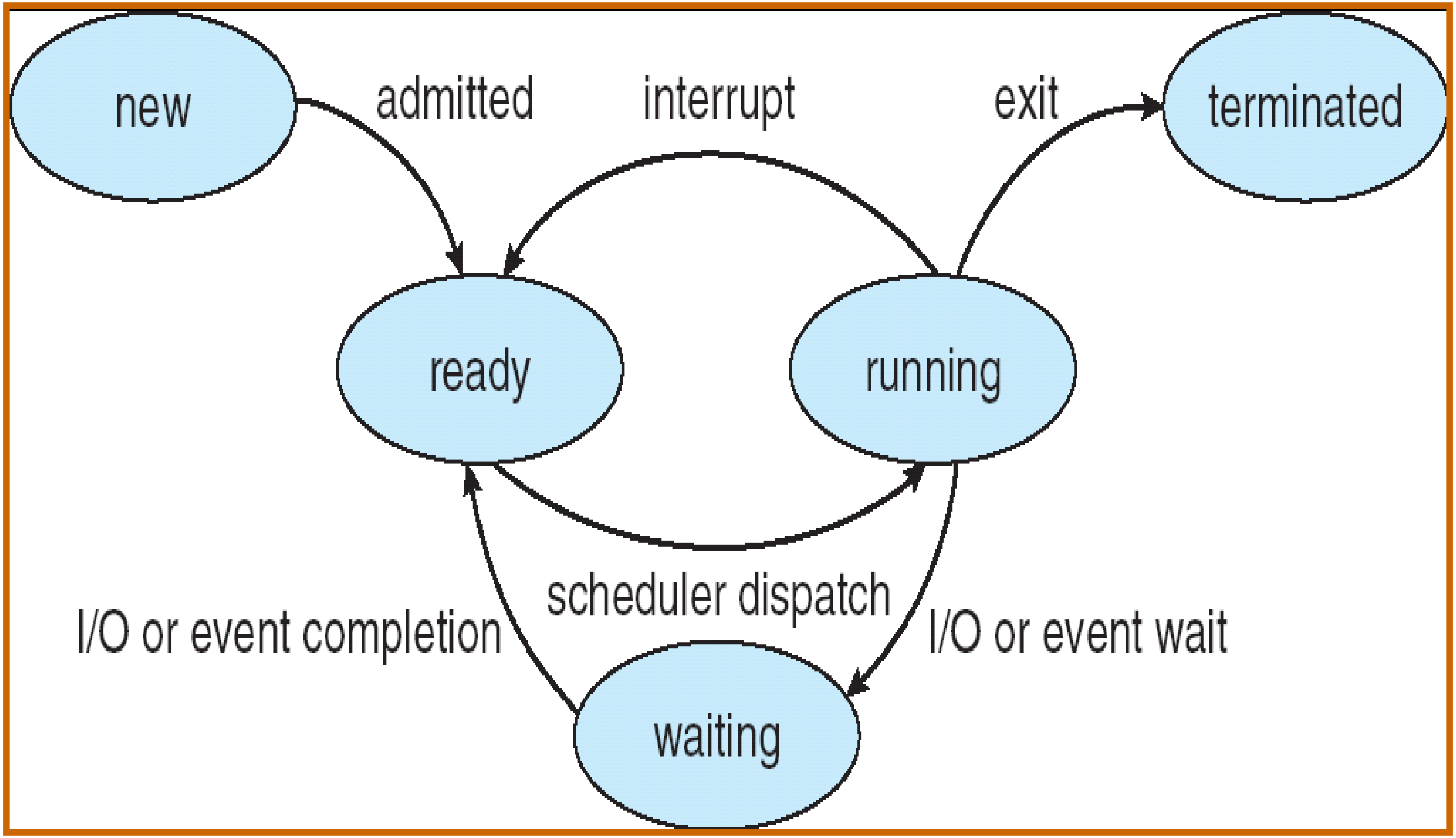
Sequence of CPU and I/O Bursts



CPU Scheduling Decisions

1. When process switches from Running State to Waiting State (i/o request or wait)	1. **Non-preemptive**
2. When process switches from Running to Ready State (interrupt)	2. **Pre-emptive**
3. When process switches from Waiting State to Ready State (at completion of i/o)	3. **Pre-emptive**
4. When a process terminates	4. **Non-Preemptive (allow)**

Process States



Scheduling

- Non-Preemptive
- Preemptive

Non-Preemptive or Cooperative Scheduling

- Once the CPU is allocated to a process, process keeps the CPU until:
  - it releases when it completes
  - by switching to waiting state

  E.g : 1. Windows 3.x and Apple Macintosh operating systems uses non-preemptive scheduling

  2. Windows (also 10) uses a round-robin technique with a *multi-level feedback queue* for priority scheduling
- Process is executed till completion. It cannot be interrupted.
  Eg First In First Out

Preemptive Scheduling

- The running process is interrupted for some time and resumed later on, when the priority task has finished its execution.

- CPU /resources is/are taken away from the process when some high priority process needs execution.

Dispatcher

- A module that gives control of CPU to the process selected by short-term scheduler.
- Functions:
  - Switching Context
  - Switching to User mode
  - Jumping to proper location to restart the program.
- The dispatcher should be as fast as possible, given that it is invoked during every process switch
- Dispatch Latency:
  - Time taken for the dispatcher **to stop one process and start another running.**

Scheduling Criteria

- Which algorithm to use in a particular situation

1. **CPU Utilization:** CPU should be busy to the fullest

2. **Throughput:** No. of processes completed per unit of time.

Turnaround Time: The time interval from submitting a process to the time of completion.

Turnaround Time= Time spent to get into memory + waiting in ready queue + doing I/O + executing on CPU

(It is the amount of time taken to execute a particular process)

Scheduling Criteria

4. **Waiting Time:** Time a process spends in ready queue.

Amount of time a process has been waiting in the ready queue to acquire control on the CPU.

5. **Response Time:** Time from the submission of a request to the first response, Not Output

6. **Load Average:** It is the average number of processes residing in the ready queue waiting for their turn to get into the CPU.

Scheduling Algorithm Optimization Criteria

- Max CPU utilization
- Max throughput
- Min turnaround time
- Min waiting time
- Min response time

First-Come, First-Served (FCFS)

- Processes that request CPU first, are allocated the CPU first

- FCFS is implemented with FIFO queue.

- A process is allocated the CPU according to their arrival times.

- When process enters the ready queue, its PCB is attached to the Tail of queue, When CPU is free, it is allocated to the process selected from Head/Front of queue.

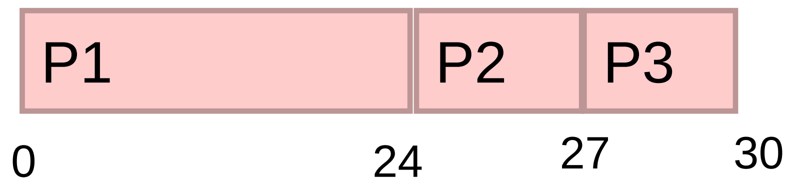
First-Come, First-Served (FCFS)

- “Run until Completed:” FIFO algorithm
- Example: Three processes arrive in order P1, P2, P3.
  - P1 burst time: 24
  - P2 burst time: 3
  - P3 burst time: 3
- Draw the Gantt Chart and compute Average Waiting Time and Average Turn Around Time/Completion Time.

Sol: As arrival time is not given assume order of arrival as: P1,P2,P3

First-Come, First-Served (FCFS)

- Example: Three processes arrive in order P1, P2, P3.
  - P1 burst time: 24
  - P2 burst time: 3
  - P3 burst time: 3
- Waiting Time
  - P1: 0
  - P2: 24
  - P3: 27



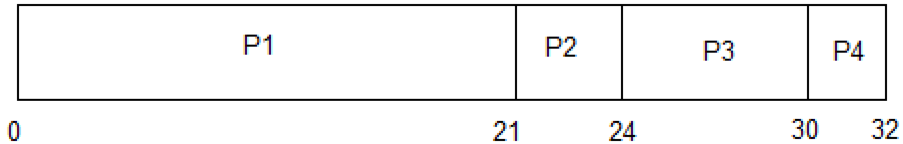
- Turnaround Time/Completion Time:
  - P1: 24
  - P2: 27
  - P3: 30

- Average Waiting Time: (0+24+27)/3 = (51/3)= 17 milliseconds
- Average Completion Time: (24+27+30)/3 = 81/3=27 milliseconds

H.W.: Example: First-Come, First-Served (FCFS)

PROCESS	BURST TIME
P1	21
P2	3
P3	6
P4	2

The average waiting time will be = (0 + 21 + 24 + 30)/4 = 18.75 ms



This is the GANTT chart for the above processes

First Come First Serve

Process id	A T	BT	CT	TAT	WT
1	0	2			
2	3	1			
3	5	6			

Turn around time= Completion Time- Arrival Time

Waiting Time= Turn around Time-Burst Time

First Come First Serve

Process id	A T	BT	CT	TAT	WT
1	0	2			
2	3	1			
3	5	6			

Solve: FCFS Arrival Time

Process	A T	BT	CT	TAT	WT
P1	0	4			
P2	1	3			
P3	2	1			
P4	3	2			
P5	4	5			

Turn around time= Completion Time- Arrival Time

Waiting Time= Turn around Time-Burst Time

Shortest Job First

- Processes with least execution time are selected first.
- CPU is assigned to process with less CPU burst time.
- SJF:
  - Non-Preemption: CPU is always allocated to the process with least burst time and Process Keeps CPU with it until it is completed.
  - Pre-Emption: When a new process enters the queue, scheduler **checks its execution time and compare with the already running process**.
  - If Execution time of running process is more, CPU is taken from it and given to new process.

Shortest Job First(Preemptive)

Q1. Consider foll. Processes with A.T and B.T

Process	A.T	B.T
P1	0	9
P2	1	4
P3	2	9

Cal. Completion time, turn around time and avg. waiting time.

SJF(Pre-emptive)-> SRTF

Shortest Job First(Preemptive)

Process	A.T	B.T
P1	0	9
P2	1	4
P3	2	9

Shortest Job First(Preemptive)

Q1. Consider foll. Processes with A.T and B.T

Process	A.T	B.T
P1	0	5
P2	1	3
P3	2	3
P4	3	1

Cal. Completion time, turn around time and avg. waiting time.

Shortest Job First(Preemptive)

Process	A.T	B.T
P1	0	5
P2	1	3
P3	2	3
P4	3	1

H.W: Shortest Job First(Non-Preemptive)

Q1. Consider foll. Processes with A.T and B.T

Process	A.T	B.T
P1	1	7
P2	2	5
P3	3	1
P4	4	2
P5	5	8

Cal. Completion time, turn around time and avg. waiting time.

H.W: Shortest Job First(Non-Preemptive)

Process	A.T	B.T
P1	1	7
P2	2	5
P3	3	1
P4	4	2
P5	5	8

H.W: Practice: Shortest Job First (Non Preemption)

- P1 burst time: 15
- P2 burst time: 8
- P3 burst time: 10
- P4 burst time: 3

H.W: Practice: Shortest Job First (Preemption)

Process	Arrival Time	Burst Time	CT	Turn around time	Waiting Time
P1	0	8			
P2	1	4			
P3	2	9			
P4	3	5			

Turn around time= Completion Time- Arrival Time

Waiting Time= Turn around Time-Burst Time

Practice: Shortest Job First (Preemption)

Process	Arrival Time	Burst Time	CT	Turn around time	Waiting Time
P1	0	7	16		
P2	2	4	7		
P3	4	1	5		
P4	5	4	11		

Turn around time= Completion Time- Arrival Time

Waiting Time= Turn around Time-Burst Time

Practice: Shortest Job First (Preemption)

Process	Arrival Time	Burst Time	CT	Turn around time	Waiting Time
P1	0	7	16		
P2	2	4	7		
P3	4	1	5		
P4	5	4	11		