UNIT -03 CS-405 OPERATING SYSTEM DETAILED NOTES

CPU Scheduling : Process Concept & Scheduling Concepts :-

Process Concept

Introduction

Computer me jab koi program execute hota hai to use **Process** kehte hain.

Program ek passive entity hota hai, jabki Process ek active entity hota hai.

Example:

```
Program = MS Word.exe

Running MS Word = Process
```

Definition

"A Process is a program in execution."

OR

"Process is an active entity that contains program code, data and execution state."

Program vs Process

⭐⭐⭐⭐⭐⭐ EXAM FAVOURITE

Program	Process
Passive Entity	Active Entity
Stored on Disk	Stored in Memory
Static	Dynamic
No Execution	Under Execution
Example: notepad.exe	Running Notepad

Components of a Process

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Every Process consists of:

```
1. Program Code (Text Section)

2. Data Section

3. Stack

4. Heap

          5. Program Counter
```

Process Memory Layout Diagram

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM

```
-----------------
```

```
Program Code

-----------------

Data Section

-----------------

Heap

-----------------

           Stack

        -----------------
```

Process States

Process execution ke dauran different states me move karta hai.

```
New

Ready

Running

Waiting

         Terminated
```

Process State Diagram

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

```
           New

            |

            v
```

Explanation of States

New State

Process create ho raha hai.

Ready State

CPU milne ka wait kar raha hai.

Running State

CPU execute kar raha hai.

Waiting State

I/O operation ka wait kar raha hai.

Terminated State

Execution complete ho gayi.

Process Control Block (PCB)

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Operating System har process ki information PCB me store karta hai.

PCB Contains

```
Process ID

Process State

Program Counter

CPU Registers

Memory Information

        Scheduling Information
```

PCB Diagram

```
-------------------

Process ID

Process State

CPU Registers

Program Counter
```

```
        Memory Info

        -------------------
```

Advantages of Process Concept

Multitasking Possible

Better Resource Utilization

Concurrent Execution

Efficient CPU Usage

Scheduling Concepts

Introduction

CPU ek time par sirf ek process execute kar sakta hai.

Agar multiple processes ready state me hain to OS decide karta hai ki pehle kis process ko CPU milega.

Is decision-making process ko CPU Scheduling kehte hain.

Definition

"CPU Scheduling is the process of selecting a process from the ready queue and allocating CPU to it."

Need of CPU Scheduling

Without Scheduling:

❌ CPU Idle

❌ Poor Performance

❌ Long Waiting Time

With Scheduling:

✅ Better CPU Utilization

✅ Higher Throughput

✅ Reduced Waiting Time

CPU Scheduling Diagram

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM

```
Ready Queue

-----------------

P1

P2

P3

P4

-----------------

        |

        v

       CPU

                  |
```

```
                    v

               Execution
```

Objectives of Scheduling

⭐⭐⭐⭐⭐ Frequently Asked

1. Maximize CPU Utilization

2. Increase Throughput

3. Reduce Waiting Time

4. Reduce Turnaround Time

5. Improve Response Time

6. Fairness

Scheduling Cycle

```
Process Created

      ↓

Ready Queue

      ↓

CPU Assigned

      ↓

Running

                  ↓
```

```
            Completed
```

Important Terms

CPU Utilization

Percentage of CPU busy time.

Throughput

Number of completed processes per unit time.

Turnaround Time

`Completion Time - Arrival Time`

Waiting Time

Time spent in ready queue.

Response Time

Time between request and first response.

Process Concept vs Scheduling Concept

⭐⭐⭐⭐⭐⭐ EXAM TABLE

Process Concept	**Scheduling Concept**
Program in Execution	CPU Allocation

Describes Process	Describes CPU Selection
Uses PCB	Uses Ready Queue
Focus on Execution	Focus on Scheduling

Real Life Example

Suppose Railway Reservation Counter hai.

Processes

```
Customer1

Customer2

            Customer3
```

Scheduling

Counter decide karega:

`Pehle kis customer ko serve karna hai?`

Ye Scheduling hai.

Viva Questions

Q1. What is a Process?

Program in execution.

Q2. What is CPU Scheduling?

Process selection for CPU allocation.

Q3. What is PCB?

Process Control Block.

Q4. What is Ready State?

Waiting for CPU.

Q5. Why is Scheduling needed?

To improve CPU utilization.

Frequently Asked RGPV Questions

---

7 Marks

1. Explain Process State Diagram.
2. Explain CPU Scheduling.
3. Explain Process Memory Layout.

---

14 Marks

Q. Explain Process Concept with neat diagram.

Q. Explain CPU Scheduling and its objectives.

Q. Explain Process State Diagram and PCB.

PYQ Trend Analysis

Topic	Frequency

Process Concept	⭐⭐⭐⭐⭐⭐
PCB	⭐⭐⭐⭐⭐
Process State Diagram	⭐⭐⭐⭐⭐⭐
Scheduling Concepts	⭐⭐⭐⭐⭐
Objectives of Scheduling	⭐⭐⭐⭐

Expected 2026 Questions

🔥 Explain Process Concept with Process State Diagram.

🔥 Explain Process Control Block (PCB).

🔥 Explain CPU Scheduling and its objectives.

🔥 Differentiate Program and Process.

🔥 Explain Process States with diagram.

One-Minute Revision

```
PROCESS

↓

Program in Execution

↓

PCB

↓

States

New
```

```
Ready
Running
Waiting
Terminated

                  ↓

        CPU Scheduling

                  ↓

            Select Process from Ready Queue
```

Memory Trick: NRRWT

```
N → New

R → Ready

R → Running

W → Waiting

           T → Terminated
```

🎯 **Exam Tip:** Unit-3 me **Process State Diagram + PCB Diagram** sabse important diagrams hain. Inhe exam me zarur banao. Ye dono diagrams alone 3–4 extra marks dilwa sakte hain. 🚀📚

Types of Schedulers:-

Introduction

Operating System me ek samay par bahut saare processes hote hain.

OS ko decide karna hota hai:

```
Kaunsa process memory me aayega?

Kaunsa process CPU par chalega?

Kaunsa process memory se bahar jayega?
```

Ye kaam **Schedulers** karte hain.

Definition

"A Scheduler is an Operating System component that selects processes from different queues and allocates system resources to them."

Need of Schedulers

Without Scheduler:

❌ CPU Idle

❌ Memory Waste

❌ Poor Performance

With Scheduler:

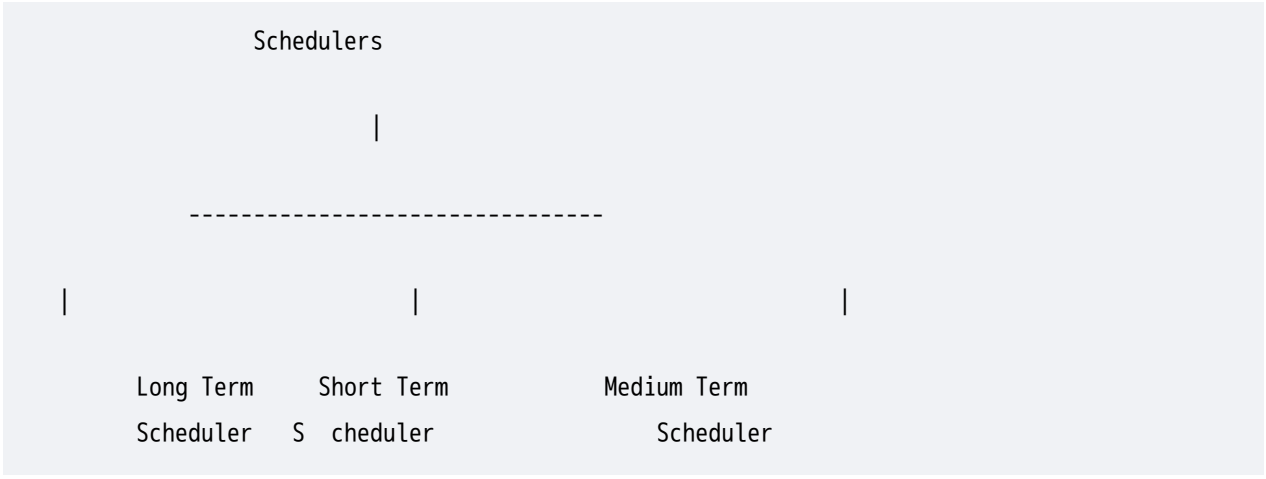
✅ Better CPU Utilization

✅ Efficient Memory Usage

✅ Faster Execution

Types of Schedulers

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM



---

1. Long-Term Scheduler

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

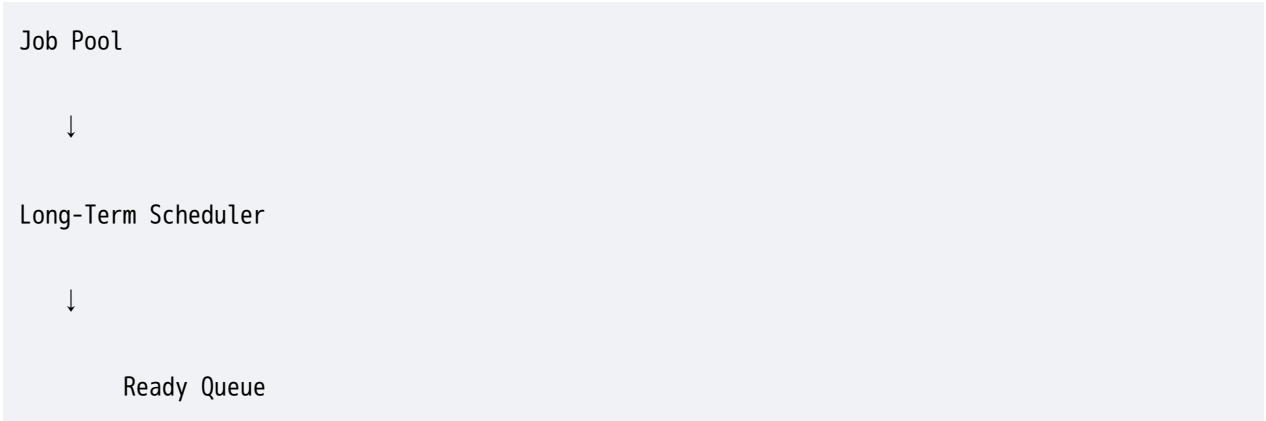
Also Called

`Job Scheduler`

Definition

Long-Term Scheduler jobs ko Job Pool se select karke Main Memory me load karta hai.

Working



Diagram

```
Jobs on Disk

      ↓

Long-Term Scheduler

      ↓

       Ready Queue
```

Functions

✔ New Processes Select Karna

✔ Degree of Multiprogramming Control Karna

✔ Memory me Process Load Karna

Characteristics

Slow Speed

Less Frequently Used

Controls Multiprogramming

Example

Suppose:

`100 Jobs`

Disk par hain.

Long-Term Scheduler:

`10 Jobs`

Memory me load karega.

Advantages

✔ Better Memory Management

✔ Controls System Load

---

2. Short-Term Scheduler

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Also Called

`CPU Scheduler`

Definition

Ready Queue se process select karke CPU allocate karta hai.

Working

```
Ready Queue

      ↓

Short-Term Scheduler

      ↓

           CPU
```

Diagram

```
Ready Queue

P1

P2

P3

P4

      ↓

Short-Term Scheduler

                ↓

              CPU
```

Functions

✔ CPU Allocation

✔ Context Switching

✔ Process Selection

Characteristics

Very Fast

Runs Frequently

Most Important Scheduler

Example

Ready Queue:

```
P1

P2

          P3
```

Scheduler decide karega:

`P1 First`

Advantages

✔ Better CPU Utilization

✔ Fast Response

---

3. Medium-Term Scheduler

⭐⭐⭐⭐⭐ Frequently Asked

Also Called

`Swapper`

Definition

Medium-Term Scheduler processes ko memory se temporarily remove karta hai aur baad me wapas load karta hai.

Working

```
Memory

      ↓

Medium-Term Scheduler

      ↓

Swap Out

                ↓

             Disk
```

Diagram

```
Main Memory

       ↓

Medium-Term Scheduler

       ↓

Disk

               ↓

        Swap In
```

Functions

✔ Swapping

✔ Memory Management

✔ Reduce Multiprogramming

Characteristics

Moderate Speed

Memory Utilization Improve Karta Hai

Example

Memory Full Ho Gayi:

```
P1

P2

P3

          P4
```

OS:

`P4`

ko temporarily disk me bhej dega.

Advantages

✔ Better Memory Utilization

✔ Prevents Memory Overload

Complete Working of Schedulers

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM

```
Job Pool

    ↓
```

```
Long-Term Scheduler
     ↓
Ready Queue
     ↓
Short-Term Scheduler
     ↓
CPU
     ↓
Running Process
     ↓
          Blocked
               ↓
     Medium-Term Scheduler
               ↓
          Disk
```

Comparison of Schedulers

⭐⭐⭐⭐⭐⭐ EXAM FAVOURITE TABLE

Feature	**Long-Term**	**Short-Term**	**Medium-Term**
Also Called	Job Scheduler	CPU Scheduler	Swapper
Selects From	Job Pool	Ready Queue	Memory
Selects To	Ready Queue	CPU	Disk

Speed	Slow	Fast	Medium
Frequency	Low	Very High	Medium
Main Function	Process Admission	CPU Allocation	Swapping
Controls	Multiprogramming	CPU Utilization	Memory Usage

Long-Term vs Short-Term Scheduler

Long-Term	**Short-Term**
Selects Jobs	Selects Processes
Controls Multiprogramming	Controls CPU
Slow	Very Fast
Less Frequent	Frequent

Advantages of Schedulers

Better CPU Utilization

Efficient Memory Usage

Reduced Waiting Time

Increased Throughput

Improved System Performance

Real Life Example

Suppose Railway Station hai.

Long-Term Scheduler

`Station me kitne passengers aayenge?`

Short-Term Scheduler

`Kaun passenger pehle ticket lega?`

Medium-Term Scheduler

`Waiting Room me bhejna.`

Viva Questions

Q1. What is a Scheduler?

OS component that selects processes.

Q2. Which scheduler is called CPU Scheduler?

Short-Term Scheduler.

Q3. Which scheduler is called Job Scheduler?

Long-Term Scheduler.

Q4. Which scheduler performs swapping?

Medium-Term Scheduler.

Q5. Which scheduler is fastest?

Short-Term Scheduler.

Frequently Asked RGPV Questions

---

7 Marks

1. Explain Types of Schedulers.
2. Compare Long-Term and Short-Term Scheduler.
3. Explain role of Medium-Term Scheduler.

---

14 Marks

Q. Explain Types of Schedulers with neat diagram.

Q. Compare Long-Term, Short-Term and Medium-Term Schedulers.

Q. Explain working of schedulers in Operating System.

PYQ Trend Analysis

Topic	Frequency
Types of Schedulers	⭐⭐⭐⭐⭐⭐
Long-Term Scheduler	⭐⭐⭐⭐⭐
Short-Term Scheduler	⭐⭐⭐⭐⭐⭐
Medium-Term Scheduler	⭐⭐⭐⭐
Comparison Table	⭐⭐⭐⭐⭐⭐

Expected 2026 Questions

🔥 Explain Types of Schedulers with diagram.

🔥 Compare Long-Term, Short-Term and Medium-Term Schedulers.

🔥 Explain CPU Scheduler.

🔥 Explain Job Scheduler.

🔥 Explain Swapping and Medium-Term Scheduler.

One-Minute Revision

```
Schedulers

↓

Long-Term Scheduler
(Job Scheduler)

↓

Short-Term Scheduler
(CPU Scheduler)

            ↓

        Medium-Term Scheduler
       (Swapper)
```

Memory Trick: JCS

```
J → Job Scheduler (Long-Term)

C → CPU Scheduler (Short-Term)

          S → Swapper (Medium-Term)
```

🎯 **Exam Tip:** Types of Schedulers ke question me **Comparison Table + Complete Working Diagram** zarur banao. Ye examiner ko direct dikhata hai ki concept clear hai

aur 14 marks ka answer easily 4 pages tak bhar sakta hai. 🚀📚

Process State Diagram:-

Introduction

Process execution ke dauran ek process different states se pass hota hai.

Operating System process ko manage karne ke liye in states ko maintain karta hai.

In states aur transitions ko dikhane wale diagram ko **Process State Diagram** kehte hain.

Definition

"Process State Diagram is a graphical representation of various states of a process and the transitions between them during execution."

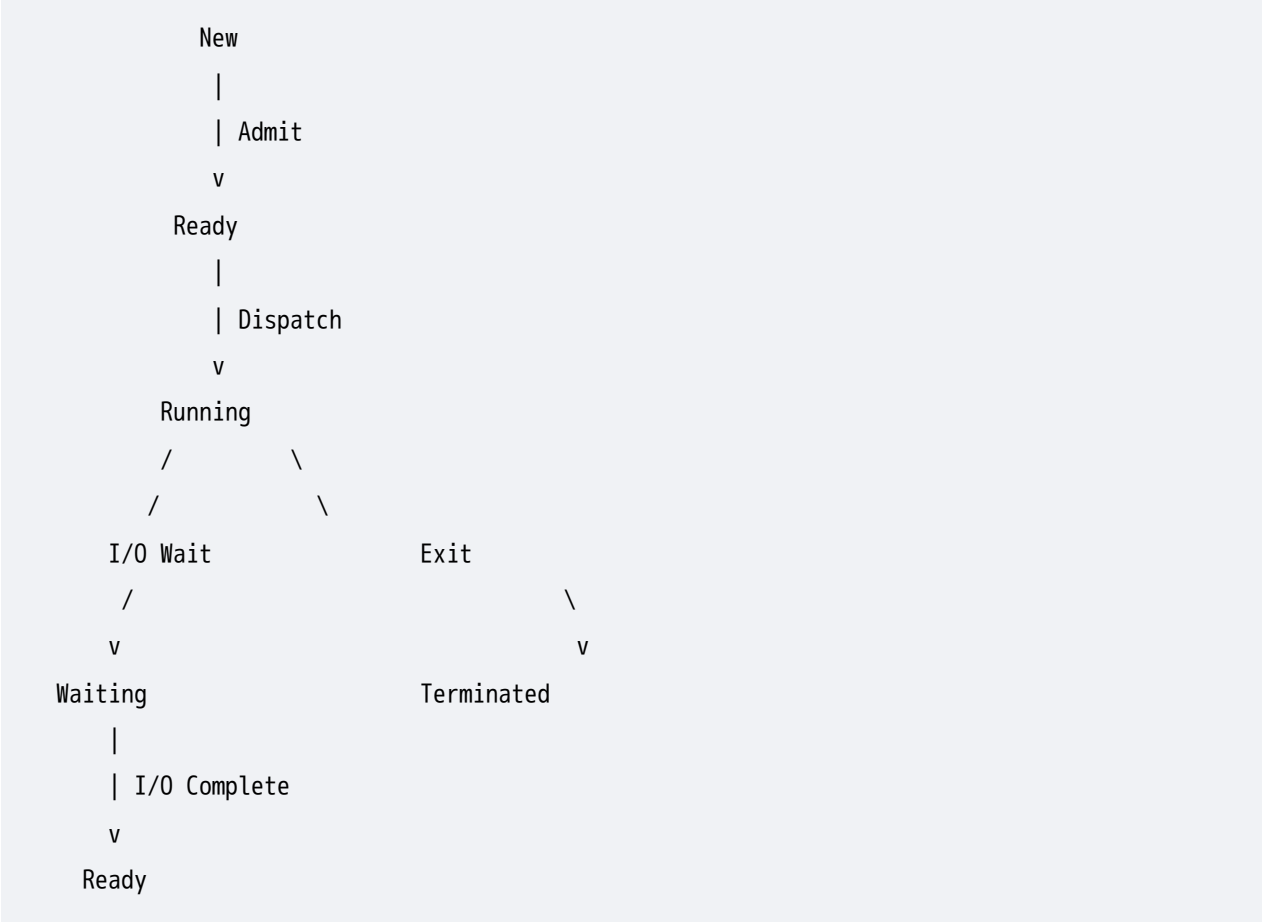
Five-State Process Model

Most Operating Systems me process 5 states me hota hai:

```
1. New

2. Ready

3. Running

4. Waiting (Blocked)

          5. Terminated
```

Process State Diagram

⭐⭐⭐⭐⭐ EXAM DIAGRAM



Explanation of Process States

---

1. New State

⭐⭐⭐⭐⭐ Important

Definition

Jab process create hota hai tab wo New State me hota hai.

Activities

✔ Process Creation

✔ Memory Allocation

✔ PCB Creation

Example

```
User opens Chrome

↓

Process Created

↓

New State
```

---

2. Ready State

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Process memory me loaded hai aur CPU milne ka wait kar raha hai.

Characteristics

✔ Ready to Execute

✔ Waiting for CPU

✔ Stored in Ready Queue

Example

```
P1

P2

P3

Ready Queue
```

Diagram

```
Ready Queue

P1

P2

P3
```

---

3. Running State

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Process CPU par execute ho raha hai.

Characteristics

✔ CPU Assigned

✔ Instructions Executing

✔ Active State

Example

```
CPU

↓

Process P1 Running
```

---

4. Waiting (Blocked) State

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Process kisi event ya I/O operation ke complete hone ka wait kar raha hai.

Reasons

✔ Disk Read

✔ Keyboard Input

✔ Printer Output

✔ Network Request

Example

```
Process

↓

Printer Request

↓

      Waiting State
```

---

5. Terminated State

⭐⭐⭐⭐⭐ Important

Definition

Jab process execution complete kar leta hai to wo terminated state me chala jata hai.

Reasons

✔ Successful Completion

✔ Error

✔ User Termination

Example

```
Program Finished

↓

Terminated
```

State Transitions

⭐⭐⭐⭐⭐⭐ EXAM FAVOURITE

Transition	Reason
New → Ready	Process Admitted
Ready → Running	CPU Assigned
Running → Waiting	I/O Request

Waiting → Ready	I/O Complete
Running → Terminated	Execution Finished

Detailed Process Flow

```
New
  ↓
Ready
  ↓
Running
  ↓
Waiting
  ↓
Ready
  ↓
          Running
              ↓
         Terminated
```

Process Control Block (PCB)

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

OS har process ki information PCB me store karta hai.

PCB Contains

```
Process ID

Process State

Program Counter

CPU Registers

Memory Information

          Scheduling Information
```

PCB Diagram

```
-------------------

Process ID

Process State

CPU Registers

Program Counter

          Memory Info

            -------------------
```

Importance of Process State Diagram

Helps CPU Scheduling

Process Tracking

Resource Management

Multitasking Support

Efficient Execution

Real Life Example

Suppose Railway Reservation Counter hai.

New

Customer arrives.

Ready

Queue me wait kar raha hai.

Running

Counter par ticket ban raha hai.

Waiting

Payment processing.

Terminated

Ticket mil gaya.

Advantages

Better Process Management

Efficient CPU Utilization

Reduced Waiting Time

Improved Performance

Two-State vs Five-State Model

Two-State Model	Five-State Model
Running	New
Not Running	Ready
	Running
	Waiting
	Terminated

Viva Questions

Q1. What is a Process?

Program in execution.

Q2. How many states are present in basic process state model?

5 States.

Q3. What is Ready State?

Waiting for CPU.

Q4. What is Waiting State?

Waiting for I/O completion.

Q5. What is PCB?

Process Control Block.

Frequently Asked RGPV Questions

---

7 Marks

1. Draw and explain Process State Diagram.
2. Explain State Transitions.
3. Explain Waiting State.

---

14 Marks

Q. Draw and explain Process State Diagram with neat diagram.

Q. Explain various states of a process and their transitions.

Q. Explain Process State Model and PCB.

PYQ Trend Analysis

Topic	Frequency
Process State Diagram	⭐⭐⭐⭐⭐⭐
Ready State	⭐⭐⭐⭐⭐
Waiting State	⭐⭐⭐⭐⭐
PCB	⭐⭐⭐⭐⭐
State Transitions	⭐⭐⭐⭐⭐

Expected 2026 Questions

🔥 Draw and explain Process State Diagram.

🔥 Explain various process states.

🔥 Explain state transitions with diagram.

🔥 Explain Process Control Block.

🔥 Explain Ready, Running and Waiting States.

One-Minute Revision

```
Process States

↓

New
 ↓
Ready
 ↓
Running
 ↓
Waiting
            ↓
        Ready
            ↓
        Running
           ↓
   Terminated
```

Memory Trick: NRRWT

```
N → New

R → Ready

R → Running

W → Waiting
```

```
        T → Terminated
```

🎯 **Exam Tip:** Agar exam me "Process State Diagram" aaye to sabse pehle diagram banao, phir 5 states aur state transitions explain karo. PCB ka small diagram bhi add kar do. Isse 14 marks ka answer easily 4 pages bhar jayega aur full marks milne ke chances bahut badh jate hain. 🚀📚

Scheduling Algorithms:-

Introduction

CPU Scheduling ka main objective hai Ready Queue me se process select karke CPU allocate karna.

Jo rule process selection ke liye use hota hai use **Scheduling Algorithm** kehte hain.

Definition

"A Scheduling Algorithm is a method used by the Operating System to decide which process will get the CPU next."

Need of Scheduling Algorithms

Without Scheduling:

❌ CPU Idle Time

❌ Long Waiting Time

❌ Poor Performance

With Scheduling:

✅ Better CPU Utilization

✅ Higher Throughput

✅ Reduced Waiting Time

✅ Better Response Time

CPU Scheduling Diagram

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM

```
Ready Queue

P1
P2
P3
P4

      ↓

CPU Scheduler

                ↓

          CPU Execution
```

Types of Scheduling Algorithms

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM

```
Scheduling Algorithms

        |

-----------------------------------------------
```

```
FCFS

SJF

SRTF

Priority

        Round Robin

        Multilevel Queue
```

---

1. FCFS (First Come First Serve)

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Jo process pehle Ready Queue me aata hai usse pehle CPU diya jata hai.

Diagram

`P1 → P2 → P3 → P4`

Example

Arrival Order:

```
P1

P2

          P3
```

Execution:

```
P1
```

```
↓

P2

↓

          P3
```

Advantages

✔ Simple

✔ Easy Implementation

✔ Fair

Disadvantages

❌ Convoy Effect

❌ High Waiting Time

---

2. SJF (Shortest Job First)

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Sabse kam Burst Time wale process ko pehle CPU diya jata hai.

Example

Process	**Burst Time**
P1	10

P2	4
P3	2

Execution Order:

`P3 → P2 → P1`

Advantages

✔ Minimum Average Waiting Time

✔ Efficient

Disadvantages

❌ Burst Time Prediction Difficult

❌ Starvation Possible

---

3. SRTF (Shortest Remaining Time First)

⭐⭐⭐⭐⭐ Important

Definition

SJF ka preemptive version.

Sabse kam remaining burst time wala process execute hota hai.

Example

```
Current Process

↓
```

```
New Shorter Process Arrives

↓

      CPU Switch
```

Advantages

✔ Better Response Time

Disadvantages

❌ Context Switching Overhead

---

4. Priority Scheduling

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Highest Priority wale process ko CPU diya jata hai.

Example

Process	Priority
P1	3
P2	1
P3	2

Execution:

`P2 → P3 → P1`

Types

Preemptive

Non-Preemptive

Advantages

✔ Important Jobs First

Disadvantages

❌ Starvation

Solution

`Aging`

Technique use karte hain.

---

5. Round Robin (RR)

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Har process ko fixed Time Quantum diya jata hai.

Diagram

```
P1 → P2 → P3 → P4
      ↑
```

```
            Repeat
```

Example

Time Quantum = 2 ms

```
P1 = 5

P2 = 4

          P3 = 3
```

Execution:

```
P1

P2

P3

P1

P2

          P3
```

Advantages

✔ Fair Scheduling

✔ Good Response Time

✔ Time Sharing Systems ke liye Best

Disadvantages

❌ More Context Switching

---

6. Multilevel Queue Scheduling

⭐⭐⭐⭐ Frequently Asked

Definition

Ready Queue ko multiple queues me divide kar diya jata hai.

Diagram

```
System Processes

Interactive Processes

          Batch Processes
```

Structure

```
Ready Queue

    |

----------------------

System Queue

Interactive Queue

           Batch Queue
```

Advantages

✔ Different Classes of Processes Handle Kar Sakta Hai

Disadvantages

❌ Fixed Queue Assignment

Scheduling Criteria

⭐⭐⭐⭐⭐⭐ EXAM FAVOURITE

Criteria	Goal
CPU Utilization	Maximize
Throughput	Maximize
Turnaround Time	Minimize
Waiting Time	Minimize
Response Time	Minimize

Comparison of Scheduling Algorithms

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT TABLE

Algorithm	Type	Advantage	Disadvantage
FCFS	Non-Preemptive	Simple	High Waiting Time
SJF	Non-Preemptive	Minimum Waiting Time	Starvation
SRTF	Preemptive	Better Response	Overhead
Priority	Both	Important Jobs First	Starvation
RR	Preemptive	Fair	Context Switching

Multilevel Queue	Mixed	Flexible	Complex

FCFS vs SJF

FCFS	**SJF**
Arrival Order	Shortest Job First
Simple	Efficient
Higher Waiting Time	Lower Waiting Time

Priority vs Round Robin

Priority	**Round Robin**
Based on Priority	Based on Time Quantum
Starvation Possible	Fair
Suitable for Real-Time	Suitable for Time Sharing

Real Life Example

Suppose Hospital hai.

FCFS

Jo patient pehle aaya use pehle treatment.

Priority

Emergency patient ko pehle treatment.

Round Robin

Har patient ko equal consultation time.

Advantages of Scheduling Algorithms

Better CPU Utilization

Faster Processing

Reduced Waiting Time

Better Throughput

Improved Response Time

Viva Questions

Q1. What is CPU Scheduling?

CPU allocation process.

Q2. Which algorithm is simplest?

FCFS.

Q3. Which algorithm gives minimum waiting time?

SJF.

Q4. Which algorithm uses Time Quantum?

Round Robin.

Q5. What is Aging?

Technique to prevent starvation.

Frequently Asked RGPV Questions

---

7 Marks

1. Compare FCFS and SJF.
2. Compare Priority and RR.
3. Explain Scheduling Algorithms.

---

14 Marks

Q. Explain various CPU Scheduling Algorithms.

Q. Compare FCFS, SJF, Priority and Round Robin.

Q. Explain CPU Scheduling Algorithms with advantages and disadvantages.

PYQ Trend Analysis

Topic	Frequency
FCFS	⭐⭐⭐⭐⭐⭐
SJF	⭐⭐⭐⭐⭐⭐
Priority Scheduling	⭐⭐⭐⭐⭐
Round Robin	⭐⭐⭐⭐⭐⭐
Comparison Table	⭐⭐⭐⭐⭐⭐

Expected 2026 Questions

🔥 Explain CPU Scheduling Algorithms.

🔥 Compare FCFS and SJF.

🔥 Explain Round Robin with example.

🔥 Explain Priority Scheduling and Aging.

🔥 Compare all Scheduling Algorithms.

One-Minute Revision

```
Scheduling Algorithms

↓

FCFS
SJF
SRTF
         Priority
       Round Robin
      Multilevel Queue
```

Memory Trick: FSPRM

```
F → FCFS

S → SJF

P → Priority

R → Round Robin

          M → Multilevel Queue
```

🎯 **Exam Tip:** Scheduling Algorithms ke answer me **comparison table** aur **FCFS, SJF, RR ke diagrams** zarur banao. RGPV me ye Unit-3 ka sabse important long-answer topic hai aur numerical bhi isi se aate hain. 🚀📚

Algorithm Evaluation:-

Introduction

CPU Scheduling Algorithms jaise FCFS, SJF, Priority aur Round Robin me se kaunsa algorithm better hai ye decide karne ke liye kuch performance parameters use kiye jate hain.

In parameters ko **Scheduling Criteria** ya **Algorithm Evaluation Criteria** kehte hain.

Definition

"Algorithm Evaluation is the process of measuring and comparing the performance of CPU Scheduling Algorithms using different scheduling criteria."

Need of Algorithm Evaluation

Without Evaluation:

❌ Best Algorithm Select Nahi Kar Sakte

❌ Poor Performance

❌ High Waiting Time

With Evaluation:

✅ Better CPU Utilization

✅ Faster Execution

✅ Reduced Waiting Time

✅ Improved System Performance

Scheduling Evaluation Criteria

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM

```
Algorithm Evaluation

         |

--------------------------------

CPU Utilization

Throughput

Turnaround Time

           Waiting Time

         Response Time
```

---

1. CPU Utilization

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

CPU kitne percentage time busy rehta hai.

Formula

CPU Utilization=CPU Busy Time/Total Time

Example

```
CPU Busy = 90 sec

Total Time = 100 sec
```

CPU Utilization:

`90%`

Goal

✅ Maximum CPU Utilization

---

2. Throughput

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Ek unit time me complete hone wale processes ki sankhya.

Formula

Throughput=Number of Completed Processes/Total Time

Example

```
20 Processes

10 Seconds
```

Throughput:

`2 Processes/sec`

Goal

✅ Maximum Throughput

---

3. Turnaround Time (TAT)

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Process submit hone se lekar complete hone tak ka total time.

Formula

Turnaround Time = Completion Time−Arrival Time

Example

```
Arrival Time = 2

Completion Time = 12
```

TAT:

`10 Units`

Goal

✅ Minimum Turnaround Time

Diagram

```
Arrival

   |

   v

Execution

   |

              v

        Completion
```

---

4. Waiting Time (WT)

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Ready Queue me wait karne ka total time.

Formula

Waiting Time=Turnaround Time−Burst Time

Example

```
TAT = 15

BT = 10
```

WT:

`5 Units`

Goal

✅ Minimum Waiting Time

Diagram

```
Ready Queue

↓

Waiting Time

↓

          CPU
```

---

5. Response Time (RT)

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Request submit hone aur first response milne ke beech ka time.

Formula

Response Time=First CPU Allocation−Arrival Time

Example

```
Arrival Time = 0

First CPU Allocation = 3
```

RT:

`3 Units`

Goal

✅ Minimum Response Time

Summary Table

⭐⭐⭐⭐⭐⭐ EXAM FAVOURITE

Criteria	Goal
CPU Utilization	Maximize
Throughput	Maximize
Turnaround Time	Minimize
Waiting Time	Minimize
Response Time	Minimize

Comparison Example

Algorithm	CPU Utilization	Waiting Time
FCFS	Medium	High
SJF	High	Low
RR	High	Medium
Priority	High	Medium

Characteristics of Good Scheduling Algorithm

⭐⭐⭐⭐ Frequently Asked

High CPU Utilization

High Throughput

Low Waiting Time

Low Turnaround Time

Low Response Time

Fairness

Real Life Example

Suppose Bank Counter hai.

Throughput

1 Hour me kitne customers serve hue.

Waiting Time

Customer kitni der line me khada raha.

Turnaround Time

Line me lagne se service complete hone tak ka time.

Response Time

Counter ne pehli baar kab attend kiya.

Advantages of Algorithm Evaluation

Better Algorithm Selection

Improved Performance

Efficient CPU Usage

Reduced Delays

Better User Experience

Viva Questions

Q1. What is Algorithm Evaluation?

Performance measurement of scheduling algorithms.

Q2. What is CPU Utilization?

Percentage of CPU busy time.

Q3. What is Throughput?

Completed processes per unit time.

Q4. What is Turnaround Time?

Completion Time – Arrival Time.

Q5. What is Waiting Time?

Time spent in ready queue.

---

7 Marks

1. Explain Scheduling Criteria.
2. Explain Algorithm Evaluation Parameters.
3. Compare Waiting Time and Turnaround Time.

---

14 Marks

Q. Explain Algorithm Evaluation Criteria used in CPU Scheduling.

Q. Explain CPU Utilization, Throughput, Waiting Time, Turnaround Time and Response Time.

Q. Discuss performance measures of Scheduling Algorithms.

PYQ Trend Analysis

Topic	Frequency
CPU Utilization	⭐⭐⭐⭐⭐
Throughput	⭐⭐⭐⭐⭐
Turnaround Time	⭐⭐⭐⭐⭐⭐
Waiting Time	⭐⭐⭐⭐⭐⭐
Response Time	⭐⭐⭐⭐

Expected 2026 Questions

🔥 Explain Algorithm Evaluation Criteria.

🔥 Explain CPU Utilization and Throughput.

🔥 Explain Turnaround Time and Waiting Time.

🔥 Compare Waiting Time and Response Time.

🔥 Discuss performance measures of Scheduling Algorithms.

One-Minute Revision

```
Algorithm Evaluation

↓

CPU Utilization ↑

Throughput ↑

Turnaround Time ↓

         Waiting Time ↓

         Response Time ↓
```

Memory Trick: CTTWR

```
C → CPU Utilization

T → Throughput

T → Turnaround Time

W → Waiting Time

           R → Response Time
```

🎯 **Exam Tip:** Algorithm Evaluation ke answer me **5 criteria (CPU Utilization, Throughput, TAT, WT, RT)** ki definitions + formulas + summary table zarur likho. Ye topic chhota hai lekin exam me direct 5–7 marks pakke dilata hai. 🚀📚

System Calls for Process Management:-

Introduction

Process Management Operating System ka important function hai.

OS processes ko create, execute, synchronize aur terminate karne ke liye special services provide karta hai.

In services ko access karne ke liye **System Calls** use kiye jate hain.

Definition

"Process Management System Calls are operating system services used to create, control, execute, synchronize and terminate processes."

Need of Process Management System Calls

Without System Calls:

❌ Process Creation Difficult

❌ No CPU Control

❌ No Synchronization

❌ Poor Resource Management

With System Calls:

✅ Easy Process Creation

✅ Process Control

✅ Synchronization

✅ Efficient CPU Utilization

Process Management System Calls Diagram

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM

```
User Program

      |

Process Management

System Calls

      |

Operating System

               |

         Process Control
```

Major Process Management System Calls

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

```
1. fork()

2. exec()

3. wait()

4. exit()

5. getpid()

          6. kill()
```

---

1. fork() System Call

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

fork() ek naya child process create karta hai.

Parent process ki exact copy child process me banti hai.

Diagram

```
Parent Process

        |

      fork()

        |

    ----------------

        |                          |

     Parent               Child
```

Example

```
fork();
```

Advantages

✔ New Process Creation

✔ Multiprocessing Support

---

2. exec() System Call

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

exec() current process ko naye program se replace kar deta hai.

Diagram

```
Old Process

      |

    exec()

      |

           New Program
```

Example

```
execl("/bin/ls","ls",NULL);
```

Advantages

✔ Program Replacement

✔ Efficient Execution

---

3. wait() System Call

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Parent process child process ke complete hone ka wait karta hai.

Diagram

```
Parent

  |

wait()

  |

Child Completes

            |

        Parent Continues
```

Example

```
wait(NULL);
```

Advantages

✔ Synchronization

✔ Prevent Zombie Processes

---

4. exit() System Call

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Process execution terminate karta hai.

Diagram

```
Running Process

      |

    exit()

      |

            Terminated
```

Example

```
exit(0);
```

Advantages

✔ Proper Process Termination

✔ Resource Release

---

5. getpid() System Call

⭐⭐⭐⭐ Frequently Asked

Definition

Current process ka Process ID (PID) return karta hai.

Example

```
getpid();
```

Diagram

```
Process

   |

getpid()

   |

        PID Returned
```

Uses

✔ Process Identification

✔ Debugging

---

6. kill() System Call

⭐⭐⭐⭐ Frequently Asked

Definition

Kisi process ko signal bhejne ke liye use hota hai.

Example

```
kill(pid,SIGKILL);
```

Diagram

```
Process A

    |
```

```
kill()
     |
Process B
              |
       Terminated
```

Uses

✔ Process Control

✔ Process Termination

Complete Process Life Cycle Using System Calls

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM

```
Parent Process
       |
     fork()
       |
Child Process
       |
     exec()
       |
Program Execution
       |
```

```
    wait()

                  |

              exit()

                 |

          Terminated
```

Summary Table

⭐⭐⭐⭐⭐⭐ EXAM FAVOURITE

System Call	Function
fork()	Create Child Process
exec()	Replace Process Image
wait()	Wait for Child
exit()	Terminate Process
getpid()	Get Process ID
kill()	Send Signal / Terminate

Process Creation Example

```
Process P1

      |

    fork()

       |

    ----------------
```

```
              |                                   |

           P1                              P2
```

Advantages of Process Management System Calls

Process Creation

Process Synchronization

Resource Management

Better CPU Utilization

Multitasking Support

Real Life Example

Suppose Company hai.

fork()

New Employee Hire

---

exec()

Employee Department Change

---

wait()

Manager waits for employee report.

---

exit()

Employee resigns.

---

fork() vs exec()

⭐⭐⭐⭐⭐⭐ EXAM TABLE

fork()	**exec()**
Creates New Process	Replaces Current Process
Parent & Child Exist	Old Program Removed
Process Duplication	Program Replacement
New PID Generated	Same PID Continues

---

wait() vs exit()

wait()	**exit()**
Parent Waits	Process Terminates
Synchronization	Resource Release
Used by Parent	Used by Child/Process

Viva Questions

Q1. What is fork()?

Creates a child process.

Q2. What is exec()?

Replaces current process.

Q3. What is wait()?

Parent waits for child completion.

Q4. What is exit()?

Terminates a process.

Q5. What is PID?

Process Identification Number.

Frequently Asked RGPV Questions

---

7 Marks

1. Explain Process Management System Calls.
2. Compare fork() and exec().
3. Explain Process Life Cycle using System Calls.

---

14 Marks

Q. Explain System Calls for Process Management with neat diagram.

Q. Explain fork(), exec(), wait() and exit() with examples.

Q. Discuss Process Management System Calls used in Operating Systems.

PYQ Trend Analysis

Topic	Frequency
fork()	⭐⭐⭐⭐⭐⭐
exec()	⭐⭐⭐⭐⭐
wait()	⭐⭐⭐⭐⭐
exit()	⭐⭐⭐⭐⭐
Comparison Tables	⭐⭐⭐⭐⭐

Expected 2026 Questions

🔥 Explain Process Management System Calls.

🔥 Explain fork(), exec(), wait() and exit().

🔥 Compare fork() and exec().

🔥 Explain Process Life Cycle using System Calls.

🔥 Explain getpid() and kill().

One-Minute Revision

```
Process Management Calls

↓

fork()

exec()

wait()

exit()

          getpid()
```

```
        kill()
```

Memory Trick: FEWGK

```
F → fork()

E → exec()

W → wait()

G → getpid()

          K → kill()
```

🎯 **Exam Tip:** Is topic me **fork() → exec() → wait() → exit() life cycle diagram** zarur banao. RGPV me ye diagram bahut pasand kiya jata hai aur answer ko direct 14-mark level tak pahucha deta hai. 🚀📚

Multiple Processor Scheduling :-

Introduction

Single Processor System me ek hi CPU hota hai.

Lekin modern computers me multiple CPUs/cores hote hain.

Multiple processors ke beech processes ko allocate aur manage karne ki technique ko **Multiple Processor Scheduling** kehte hain.

Definition

"Multiple Processor Scheduling is the process of scheduling tasks among two or more processors in a multiprocessor system."

Need of Multiple Processor Scheduling

✅ Better Performance

✅ Faster Execution

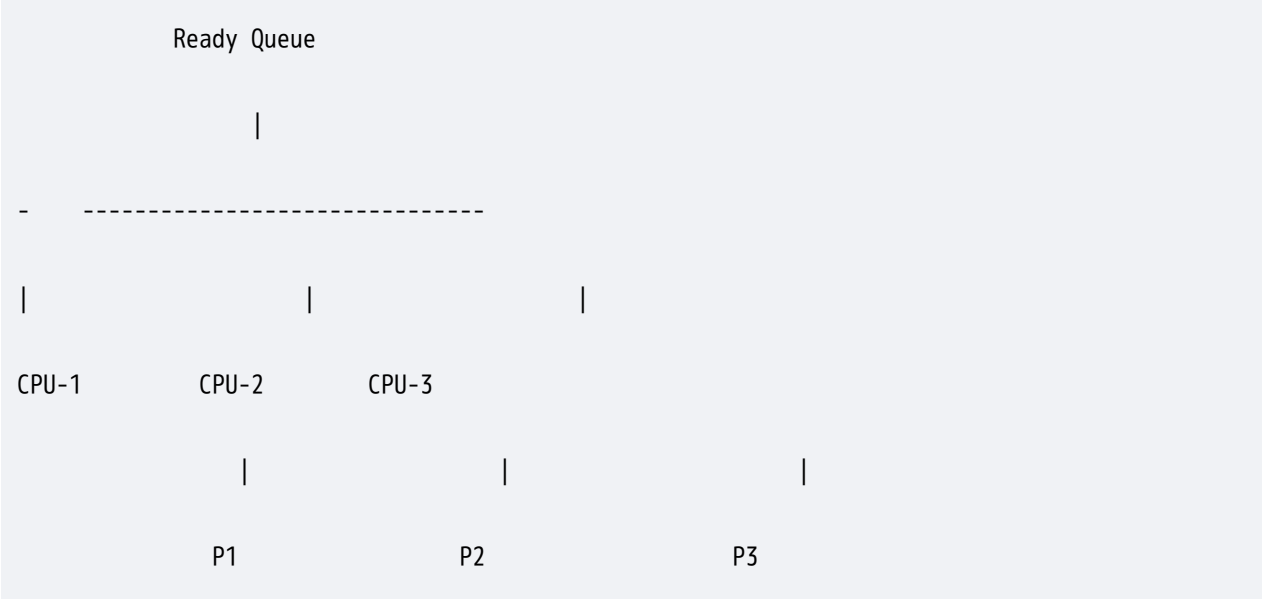
✅ Load Sharing

✅ High Throughput

✅ Better Reliability

Diagram

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM



Types of Multiprocessor Systems

1. Asymmetric Multiprocessing (AMP)

One processor acts as Master.

Other processors act as Slaves.

Diagram

```
Master CPU

        |

------------------

|        |        |

CPU2    CPU3    CPU4
```

Advantages

✔ Simple

✔ Easy Control

Disadvantages

❌ Master CPU Bottleneck

---

2. Symmetric Multiprocessing (SMP)

All processors are equal.

Each processor performs scheduling.

Diagram

```
Ready Queue

       |

------------------
```

```
|          |          |

    CPU1     CPU2       CPU3
```

Advantages

✔ Better Performance

✔ No Single Point Failure

Disadvantages

❌ Complex Synchronization

Load Balancing

⭐⭐⭐⭐⭐ Frequently Asked

Load balancing means distributing workload equally among processors.

Example

```
CPU1 = 80%

CPU2 = 20%
```

Move some tasks from CPU1 to CPU2.

Processor Affinity

Process ko same CPU par run karne ki tendency.

Advantages

✔ Better Cache Performance

✔ Reduced Overhead

Comparison: AMP vs SMP

AMP	SMP
Master-Slave	All Equal
Simple	Complex
Less Efficient	More Efficient
Single Control	Distributed Control

Advantages of Multiple Processor Scheduling

Better CPU Utilization

High Throughput

Faster Processing

Fault Tolerance

Scalability

Viva Questions

Q1. What is Multiprocessor Scheduling?

Scheduling processes among multiple CPUs.

Q2. What is SMP?

Symmetric Multiprocessing.

Q3. What is AMP?

Asymmetric Multiprocessing.

Q4. What is Load Balancing?

Equal distribution of workload.

Q5. What is Processor Affinity?

Keeping process on same CPU.

Concept of Threads:-

Introduction

Traditional process execution me har process ka apna memory space hota hai.

Ek process ke andar multiple execution paths ho sakte hain.

In execution paths ko **Threads** kehte hain.

Definition

"A Thread is the smallest unit of CPU execution within a process."

OR

"Thread is a lightweight process that shares resources of its parent process."

Diagram

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM

```
      Process

          |

    ---------------------------

|            |                |

Thread1    Thread2      Thread3
```

Why Threads?

Without Threads:

❌ Slow Execution

❌ Poor Resource Utilization

With Threads:

✅ Faster Execution

✅ Better Responsiveness

✅ Resource Sharing

Components of a Thread

Each thread has:

```
Thread ID

Program Counter

Registers

          Stack
```

Shared Resources:

```
Code Section

Data Section

         Files
```

Thread Structure Diagram

```
Process

---------------------

Code Section

Data Section

          Files

    ---------------------

       |                   |                          |

     T1                T2                    T3
```

Types of Threads

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

1. User-Level Threads (ULT)

Managed by user-level library.

Kernel ko pata nahi hota.

Diagram

```
Threads

   |

User Library

   |

            Kernel
```

Advantages

✔ Fast Creation

✔ Fast Switching

Disadvantages

❌ One thread blocked ⇒ All blocked

---

2. Kernel-Level Threads (KLT)

Managed directly by Operating System Kernel.

Diagram

```
Thread

  |

Kernel

  |

           CPU
```

Advantages

✔ Better Concurrency

✔ True Parallelism

Disadvantages

❌ Higher Overhead

User Thread vs Kernel Thread

⭐⭐⭐⭐⭐⭐ EXAM TABLE

User Thread	Kernel Thread
Managed by User	Managed by OS
Fast	Slower
Less Overhead	More Overhead
No True Parallelism	True Parallelism

Multithreading

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Multiple threads executing within a single process.

Diagram

```
Web Browser

   |
```

```
---------------------

Download Thread

Render Thread

            Audio Thread
```

Advantages of Threads

Faster Execution

Better Responsiveness

Easy Resource Sharing

Improved CPU Utilization

Reduced Overhead

Process vs Thread

⭐⭐⭐⭐⭐⭐ EXAM FAVOURITE

Process	Thread
Heavyweight	Lightweight
Separate Memory	Shared Memory
More Overhead	Less Overhead
Slow Context Switch	Fast Context Switch
Independent	Part of Process

Real Life Example

Suppose Browser Open Hai:

`Google Chrome`

Thread-1

Display Web Page

Thread-2

Download File

Thread-3

Play Audio

Sab ek saath kaam kar sakte hain.

Viva Questions

Q1. What is a Thread?

Smallest unit of execution.

Q2. What is Multithreading?

Multiple threads in one process.

Q3. What is ULT?

User-Level Thread.

Q4. What is KLT?

Kernel-Level Thread.

Q5. Why are Threads called Lightweight Processes?

Because they require fewer resources.

Frequently Asked RGPV Questions

7 Marks

1. Explain Concept of Threads.
2. Explain User-Level and Kernel-Level Threads.
3. Compare Process and Thread.

---

14 Marks

Q. Explain Concept of Threads with neat diagram.

Q. Explain types of threads and their advantages.

Q. Compare Process and Thread.

Q. Explain Multithreading with suitable examples.

One-Minute Revision

```
Threads

↓

User Level Thread (ULT)

↓

Kernel Level Thread (KLT)

↓

Multithreading
```

```
            ↓

        Advantages
```

Memory Trick: UKMP

```
U → User Thread

K → Kernel Thread

M → Multithreading

           P → Process vs Thread
```

🎯 **Exam Tip:** Threads ke answer me **Process vs Thread comparison table** aur **User Thread vs Kernel Thread table** zarur banao. Ye dono tables examiner ko direct full-mark answer ka impression dete hain. 🚀📚

Memory Management:-

Definition

Memory Management is the process of allocating, managing and controlling main memory among different processes.

Why Memory Management?

✅ Better Memory Utilization

✅ Multitasking Support

✅ Faster Execution

✅ Reduced Memory Waste

Types of Memory Management Techniques

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM

```
Memory Management

      |

---------------------------------

Partitioning

Swapping

Segmentation

Paging

Paged Segmentation
```

---

1. Partitioning

⭐⭐⭐⭐⭐ IMPORTANT

Definition

Main memory ko multiple partitions me divide kiya jata hai.

Har partition me ek process load hota hai.

Types of Partitioning

Fixed Partitioning

Memory fixed size partitions me divide hoti hai.

```
------------------

Partition 1

------------------

Partition 2

------------------

          Partition 3

           ------------------
```

Dynamic Partitioning

Process requirement ke according partition create hota hai.

```
Process P1 = 20 KB

Process P2 = 30 KB

           Process P3 = 40 KB
```

Advantages

✔ Simple

✔ Easy Implementation

Disadvantages

❌ Internal Fragmentation

❌ External Fragmentation

---

2. Swapping

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Process ko temporarily memory se disk par bhejna aur baad me wapas lana swapping kehlata hai.

Diagram

```
Main Memory

     |

Swap Out

     |

Disk

     |

Swap In

                |

      Main Memory
```

Working

Step 1

Memory Full

Step 2

Process Disk me bheja gaya

Step 3

Required hone par wapas memory me load

Advantages

✔ More Processes Execute Ho Sakte Hain

✔ Better Memory Utilization

Disadvantages

❌ Slow Due To Disk Access

---

3. Segmentation

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Program ko logical segments me divide kiya jata hai.

Segments

```
Code Segment

Data Segment

Stack Segment
```

```
            Heap Segment
```

Diagram

```
Program

   |

----------------------

Code

Data

            Stack

           Heap
```

Logical Address

```
< Segment Number, Offset >
```

Advantages

✔ Logical Organization

✔ Easy Protection

✔ Easy Sharing

Disadvantages

❌ External Fragmentation

---

4. Paging

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT TOPIC

Definition

Physical Memory ko fixed-size frames me divide kiya jata hai.

Process ko fixed-size pages me divide kiya jata hai.

Diagram

```
Process

 |

Page 0

Page 1

Page 2

Page 3


Memory

 |

Frame 0

           Frame 1

          Frame 2

          Frame 3
```

Address Format

```
Page Number

+

Offset
```

Page Table

Page ko Frame se map karta hai.

Diagram

```
Page Table

Page 0 → Frame 2

Page 1 → Frame 5

           Page 2 → Frame 8
```

Advantages

✔ No External Fragmentation

✔ Efficient Memory Use

✔ Virtual Memory Support

Disadvantages

❌ Internal Fragmentation

❌ Page Table Overhead

---

5. Paged Segmentation

⭐⭐⭐⭐⭐ IMPORTANT

Definition

Segmentation aur Paging ka combination.

Pehle Program Segments me divide hota hai.

Phir har segment pages me divide hota hai.

Diagram

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM

```
Program

  |

--------------------

Code Segment

Data Segment

Stack Segment

--------------------

Each Segment

↓

       Pages
```

```
        ↓

    Frames
```

Working

Step 1

Segment Create

Step 2

Segment ko Pages me Divide

Step 3

Pages ko Frames me Store

Advantages

✔ Segmentation Benefits

✔ Paging Benefits

✔ Less Fragmentation

Disadvantages

❌ Complex Implementation

❌ More Overhead

Comparison Table

⭐⭐⭐⭐⭐⭐ EXAM FAVOURITE

Technique	Division Basis	Fragmentation
Partitioning	Fixed/Variable Blocks	Internal & External
Swapping	Whole Process	No
Segmentation	Logical Segments	External
Paging	Fixed Pages	Internal
Paged Segmentation	Segment + Page	Very Less

Paging vs Segmentation

⭐⭐⭐⭐⭐⭐ IMPORTANT

Paging	Segmentation
Physical Division	Logical Division
Fixed Size	Variable Size
Internal Fragmentation	External Fragmentation
User Invisible	User Visible

Advantages of Memory Management

Better Memory Utilization

Multiprogramming Support

Faster Execution

Better Resource Sharing

Improved Performance

Viva Questions

Q1. What is Memory Management?

Management of main memory.

Q2. What is Paging?

Division into pages and frames.

Q3. What is Segmentation?

Division into logical segments.

Q4. What is Swapping?

Transfer of process between memory and disk.

Q5. What is Paged Segmentation?

Combination of Paging and Segmentation.

Frequently Asked RGPV Questions

---

7 Marks

1. Compare Paging and Segmentation.
2. Explain Partitioning.

3. Explain Paged Segmentation.

---

14 Marks

Q. Explain Different Memory Management Techniques.

Q. Explain Partitioning, Swapping, Paging and Segmentation.

Q. Compare Paging and Segmentation.

Q. Explain Paged Segmentation with neat diagram.

PYQ Trend Analysis

Topic	Frequency
Paging	⭐⭐⭐⭐⭐⭐
Segmentation	⭐⭐⭐⭐⭐⭐
Paging vs Segmentation	⭐⭐⭐⭐⭐⭐
Swapping	⭐⭐⭐⭐⭐
Partitioning	⭐⭐⭐⭐
Paged Segmentation	⭐⭐⭐⭐

Expected 2026 Questions

🔥 Explain Paging with diagram.

🔥 Explain Segmentation with diagram.

🔥 Compare Paging and Segmentation.

🔥 Explain Different Memory Management Techniques.

🔥 Explain Paged Segmentation.

One-Minute Revision

```
Memory Management

↓

Partitioning

↓

Swapping

↓

Segmentation

↓

         Paging

           ↓

     Paged Segmentation
```

Memory Trick: PSSPP

```
P → Partitioning

S → Swapping

S → Segmentation

P → Paging

           P → Paged Segmentation
```

🎯 **Exam Tip:** Unit-3 me sabse important long questions **Paging**, **Segmentation**, aur **Paging vs Segmentation Comparison** se bante hain. Exam me inke diagrams zarur banana. Ye 14 marks ke guaranteed topics hain. 🚀📚

Comparison of Memory Management Techniques

Introduction

Memory Management Techniques ka use processes ko memory allocate karne ke liye kiya jata hai.

Different techniques ke apne advantages aur disadvantages hote hain.

Important techniques:

```
1. Partitioning

2. Swapping

3. Segmentation

4. Paging

5. Paged Segmentation
```

Overall Comparison Table

⭐⭐⭐⭐⭐ EXAM FAVOURITE

Feature	Partitioning	Swapping	Segmentation	Paging	Paged Segmentation

Memory Division	Partitions	Whole Process	Segments	Pages	Segments + Pages
Size	Fixed/Variable	Whole Process	Variable	Fixed	Mixed
Fragmentation	Internal & External	No	External	Internal	Very Less
Complexity	Low	Low	Medium	Medium	High
Memory Utilization	Moderate	Good	Good	Better	Best
Protection	Limited	Limited	Good	Good	Excellent
Sharing	Difficult	Difficult	Easy	Easy	Very Easy
Speed	Fast	Slow	Fast	Fast	Moderate

Partitioning vs Paging

⭐⭐⭐⭐⭐⭐ MOST ASKED

Partitioning	**Paging**
Memory divided into partitions	Memory divided into frames
Process loaded as whole	Process divided into pages
Internal & External Fragmentation	Only Internal Fragmentation
Simple	More Efficient
Limited Multiprogramming	Better Multiprogramming

Diagram

```
Partitioning

Memory

------------

P1

------------

P2

------------

P3

------------


Paging

Pages → Frames

P1 → F3

P2 → F7

P3 → F2
```

Paging vs Segmentation

⭐⭐⭐⭐⭐⭐ VERY IMPORTANT

Paging	**Segmentation**
Physical Division	Logical Division
Fixed Size Pages	Variable Size Segments
User Invisible	User Visible

Internal Fragmentation	External Fragmentation
Page Table Used	Segment Table Used
Better Memory Utilization	Better Program Structure

Diagram

```
Paging

Process

↓

Page0

Page1

Page2


Segmentation

Program

↓

Code

Data

Stack
```

Segmentation vs Paged Segmentation

⭐⭐⭐⭐⭐ IMPORTANT

Segmentation	**Paged Segmentation**

Only Segments	Segments + Pages
External Fragmentation	Very Less Fragmentation
Simpler	More Complex
Segment Table	Segment + Page Table
Less Overhead	More Overhead

Swapping vs Paging

⭐⭐⭐⭐ Frequently Asked

Swapping	**Paging**
Whole Process Transfer	Page Transfer
Slow	Faster
Disk Dependent	Memory Management Based
Large Overhead	Less Overhead

Fragmentation Comparison

⭐⭐⭐⭐⭐⭐ EXAM FAVOURITE

Technique	**Internal Fragmentation**	**External Fragmentation**
Partitioning	Yes	Yes
Swapping	No	No

Segmentation	No	Yes
Paging	Yes	No
Paged Segmentation	Very Less	Very Less

Memory Utilization Comparison

Technique	**Memory Utilization**
Partitioning	Moderate
Swapping	Good
Segmentation	Good
Paging	Better
Paged Segmentation	Best

Advantages Comparison

Technique	**Main Advantage**
Partitioning	Simple
Swapping	Supports More Processes
Segmentation	Logical Organization
Paging	No External Fragmentation
Paged Segmentation	Combines Benefits of Both

Disadvantages Comparison

Technique	Main Disadvantage
Partitioning	Fragmentation
Swapping	Slow Disk Access
Segmentation	External Fragmentation
Paging	Internal Fragmentation
Paged Segmentation	Complex

Which Technique is Best?

For Simplicity

✅ Partitioning

For Logical Organization

✅ Segmentation

For Memory Utilization

✅ Paging

For Modern Systems

✅ Paged Segmentation

Summary Diagram

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM

```
Memory Management
```

```
        |

-----------------------------------------------

Partitioning

↓

Simple

↓

Fragmentation


Paging

↓

Fixed Pages

↓

No External Fragmentation


Segmentation

↓

Logical Division

            ↓

External Fragmentation


Paged Segmentation

           ↓

Best Features Combined
```

Viva Questions

Q1. Which technique uses pages and frames?

Paging.

Q2. Which technique uses logical segments?

Segmentation.

Q3. Which technique suffers from external fragmentation?

Segmentation.

Q4. Which technique suffers from internal fragmentation?

Paging.

Q5. Which technique combines paging and segmentation?

Paged Segmentation.

Frequently Asked RGPV Questions

5 Marks

1. Compare Paging and Segmentation.
2. Compare Partitioning and Paging.

---

7 Marks

1. Compare Memory Management Techniques.
2. Compare Segmentation and Paging.

---

14 Marks

Q. Compare different Memory Management Techniques.

Q. Compare Paging, Segmentation and Paged Segmentation.

Q. Differentiate Partitioning, Swapping, Segmentation and Paging.

PYQ Focus

⭐⭐⭐⭐⭐⭐ Highest Probability

1. Paging vs Segmentation
2. Partitioning vs Paging
3. Fragmentation Comparison
4. Paged Segmentation

One-Minute Revision

```
Partitioning
↓
Simple
↓
Internal + External Fragmentation

Swapping
↓
Whole Process Transfer

Segmentation
```

```
↓
Logical Division
↓
External Fragmentation

Paging
↓
Pages + Frames
↓
Internal Fragmentation

Paged Segmentation
↓
Best Combination
```

Memory Trick: PSSPP

```
P → Partitioning
S → Swapping
S → Segmentation
P → Paging
P → Paged Segmentation
```

🎯 **Exam Tip:** Agar exam me "Compare Memory Management Techniques" aaye, to **3–4 comparison tables** zarur banao. RGPV examiners tables ko bahut importance dete hain aur ye answer ko direct full-mark category me le jata hai. 🚀📚

Introduction

Kabhi-kabhi program ka size Main Memory se bada hota hai.

Example:

```
Program Size = 500 MB

Available RAM = 256 MB
```

Aise cases me Operating System kuch special techniques use karta hai jisse large programs bhi execute ho sake.

Techniques for Supporting Large Programs

Techniques for Supporting the Execution of Large Programs:-

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM

```
Large Program Execution

        |

---------------------------------

Overlay

Dynamic Loading

          Dynamic Linking

             Virtual Memory
```

---

1. Overlay Technique

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Program ko multiple modules me divide kiya jata hai.

Ek time par sirf required module memory me load hota hai.

Diagram

```
Program

 |

-------------------

Module A

Module B

Module C

-------------------

      Only Required Module

                ↓

        Memory
```

Working

Step 1

Program divide into modules

Step 2

Required module load

Step 3

Unused module replace

Example

```
Compiler

↓

Lexical Analysis

↓

Syntax Analysis

            ↓

       Code Generation
```

Ek time par sirf required phase memory me.

Advantages

✔ Less Memory Required

✔ Large Program Execution

Disadvantages

❌ Programmer Managed

❌ Complex

---

2. Dynamic Loading

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Program ke modules memory me tab load hote hain jab unki actual need hoti hai.

Diagram

```
Program

   |

Function A

Function B

Function C

               |

       Load When Needed
```

Working

```
Function Call

↓

Load Module

↓

        Execute
```

Example

Calculator Program

```
Basic Functions

Always Loaded

Scientific Functions

          Loaded When Used
```

Advantages

✔ Saves Memory

✔ Faster Startup

Disadvantages

❌ Loading Delay

---

3. Dynamic Linking

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Library routines execution time par link hoti hain.

Compile time par complete library include nahi hoti.

Diagram

```
Program

     |

Dynamic Linker

     |

Shared Library

               |
```

```
        Execution
```

Example

```
DLL Files (Windows)

Shared Libraries (.so) in Linux
```

Advantages

✔ Less Memory Usage

✔ Easy Library Updates

✔ Shared Libraries

Disadvantages

❌ Dependency Problems

Dynamic Loading vs Dynamic Linking

⭐⭐⭐⭐⭐ EXAM TABLE

Dynamic Loading	**Dynamic Linking**
Loads Module	Links Library
Done During Execution	Done During Execution
Program Module	Shared Library
Saves Memory	Saves Memory

---

4. Virtual Memory

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Virtual Memory allows execution of programs larger than physical memory.

Diagram

```
Large Program

      |

Virtual Memory

      |

RAM + Disk

                |

          Execution
```

Working

Step 1

Program divided into pages

Step 2

Required pages loaded in RAM

Step 3

Remaining pages stored on disk

Example

```
Program = 1 GB

RAM = 256 MB
```

Still program can run.

Advantages

✔ Execute Large Programs

✔ Better Memory Utilization

✔ Multiprogramming Support

Disadvantages

❌ Page Fault Overhead

❌ Disk Access Delay

Comparison Table

⭐⭐⭐⭐⭐⭐ EXAM FAVOURITE

Technique	Main Idea	Advantage
Overlay	Load Required Module	Less Memory
Dynamic Loading	Load On Demand	Fast Startup

Dynamic Linking	Link Library On Demand	Shared Libraries
Virtual Memory	Use Disk as Extension of RAM	Large Program Execution

---

Summary Diagram

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

```
Large Program Support

↓

Overlay

↓

Dynamic Loading

↓

Dynamic Linking

              ↓

       Virtual Memory
```

Advantages of These Techniques

Better Memory Utilization

Large Program Execution

Reduced RAM Requirement

Improved System Performance

Efficient Resource Usage

Viva Questions

Q1. What is Overlay?

Loading only required modules.

Q2. What is Dynamic Loading?

Loading module when needed.

Q3. What is Dynamic Linking?

Linking libraries during execution.

Q4. What is Virtual Memory?

Extension of RAM using disk.

Q5. Which technique allows programs larger than RAM?

Virtual Memory.

Frequently Asked RGPV Questions

---

7 Marks

1. Explain techniques supporting large programs.
2. Compare Dynamic Loading and Dynamic Linking.

3. Explain Virtual Memory.

---

14 Marks

Q. Explain techniques for supporting the execution of large programs.

Q. Explain Overlay, Dynamic Loading, Dynamic Linking and Virtual Memory.

Q. Compare Dynamic Loading and Dynamic Linking.

PYQ Trend Analysis

Topic	Frequency
Overlay	⭐⭐⭐⭐⭐
Dynamic Loading	⭐⭐⭐⭐⭐
Dynamic Linking	⭐⭐⭐⭐⭐
Virtual Memory	⭐⭐⭐⭐⭐⭐

Expected 2026 Questions

🔥 Explain Overlay Technique.

🔥 Explain Dynamic Loading and Dynamic Linking.

🔥 Compare Dynamic Loading and Dynamic Linking.

🔥 Explain Virtual Memory.

🔥 Explain techniques supporting execution of large programs.

One-Minute Revision

```
Large Program Execution

↓

Overlay

↓

Dynamic Loading

↓

Dynamic Linking

          ↓

        Virtual Memory
```

Memory Trick: ODDV

```
O → Overlay

D → Dynamic Loading

D → Dynamic Linking

          V → Virtual Memory
```

🎯 **Exam Tip:** Is topic me **Overlay + Dynamic Loading + Dynamic Linking + Virtual Memory comparison table** zarur banao. Ye direct 14 marks ka complete answer ban jata hai aur RGPV me bahut baar repeat hua hai. 🚀📚

Virtual Memory – Concept and Implementation by Demand Paging :-

---

Introduction

Kabhi-kabhi program ka size RAM se bada hota hai.

Example:

```
Program Size = 1 GB

RAM Available = 256 MB
```

Phir bhi program execute ho jata hai.

Ye possible hota hai **Virtual Memory** ki wajah se.

Definition

"Virtual Memory is a memory management technique that allows execution of programs larger than the available physical memory by using disk space as an extension of RAM."

Basic Idea

```
Physical Memory (RAM)

+

Secondary Memory (Disk)

=
```

```
Virtual Memory
```

Why Virtual Memory?

Without Virtual Memory:

❌ Large Programs Run Nahi Kar Sakte

❌ Limited Multiprogramming

❌ Poor Memory Utilization

With Virtual Memory:

✅ Large Programs Execute

✅ Better Multiprogramming

✅ Efficient Memory Utilization

Virtual Memory Diagram

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM

```
          Program

             |

       Virtual Memory

             |

------------------------------

|                            |
```

```
RAM                                Disk

(Physical Memory)      (Backing Store)
```

Working of Virtual Memory

Step 1

Program Pages me divide hota hai.

Step 2

Required pages RAM me load hote hain.

Step 3

Remaining pages disk me store rehte hain.

Step 4

Jab page ki need hoti hai to RAM me laya jata hai.

Diagram

```
Program

↓

Page 0

Page 1

Page 2

Page 3
```

```
↓

RAM

Page 0

Page 2

                  ↓

              Disk

          Page 1

          Page 3
```

Advantages of Virtual Memory

Execute Large Programs

Better CPU Utilization

Better Memory Utilization

Increased Multiprogramming

Efficient Resource Sharing

Disadvantages

Page Fault Overhead

Disk Access Slow

Complex Management

Demand Paging

⭐⭐⭐⭐⭐⭐ MOST IMPORTANT

Definition

Demand Paging is a virtual memory technique in which pages are loaded into memory only when they are actually required.

Basic Idea

```
Load Page

Only When Needed
```

Why Demand Paging?

Without Demand Paging:

```
Entire Program Loaded
```

Memory Waste.

With Demand Paging:

```
Only Required Pages Loaded
```

Memory Saved.

Demand Paging Diagram

⭐⭐⭐⭐⭐⭐ EXAM DIAGRAM

```
Program

↓

Page Needed?

↓

YES

↓

Load Page Into RAM

          ↓

       Execute
```

Working of Demand Paging

Step 1

CPU page access karta hai.

Step 2

Check Page Table.

Step 3

Page memory me hai?

If YES

Process Continue.

If NO

Page Fault Occurs.

Page disk se RAM me load hota hai.

Step 4

Execution Continue.

Page Fault

⭐⭐⭐⭐⭐⭐ EXAM FAVOURITE

Definition

Page memory me available nahi hota to Page Fault generate hota hai.

Page Fault Handling Steps

```
CPU Requests Page

↓

Page Not Found

↓

Page Fault

↓

OS Loads Page

↓
```

```
Page Table Updated

↓

Execution Restart
```

Page Fault Diagram

```
CPU

 ↓

Page Table

 ↓

Page Present?

 ↓

NO

 ↓

Page Fault

 ↓

Disk

             ↓

          RAM

            ↓

       Continue
```

Demand Paging Architecture

⭐⭐⭐⭐⭐ MOST IMPORTANT

```
CPU

 ↓

Page Table

 ↓

Physical Memory

 ↓

Page Missing

 ↓

        Backing Store (Disk)

            ↓

      Page Loaded
```

Virtual Memory vs Physical Memory

⭐⭐⭐⭐⭐ EXAM TABLE

Virtual Memory	**Physical Memory**
Logical Memory	Actual RAM
Larger Size	Limited Size
Uses Disk	Uses RAM
Supports Large Programs	Limited by RAM

Paging vs Demand Paging

⭐⭐⭐⭐⭐⭐ IMPORTANT

Paging	Demand Paging
All Pages Loaded	Pages Loaded On Demand
More Memory Used	Less Memory Used
Faster Start	Slower First Access
Higher Memory Requirement	Efficient Memory Usage

Advantages of Demand Paging

Less Memory Usage

Faster Program Loading

Better Multiprogramming

Efficient Resource Utilization

Large Program Support

Disadvantages of Demand Paging

Page Fault Overhead

Disk Access Delay

Increased Complexity

Real Life Example

Suppose Library hai.

Without Demand Paging:

```
Puri Library ki Books

Table par rakh do
```

With Demand Paging:

```
Sirf Required Book

Table par lao
```

Memory save hogi.

Viva Questions

Q1. What is Virtual Memory?

Disk ko RAM extension ki tarah use karna.

Q2. What is Demand Paging?

Pages load only when needed.

Q3. What is Page Fault?

Page not found in RAM.

Q4. Why Virtual Memory is required?

To run large programs.

Q5. What is Backing Store?

Disk area used for virtual memory.

Frequently Asked RGPV Questions

---

7 Marks

1. Explain Demand Paging with diagram.
2. Explain Virtual Memory architecture.
3. Explain Page Fault handling.

---

14 Marks

Q. Explain Virtual Memory and its implementation using Demand Paging.

Q. Explain Demand Paging and Page Fault mechanism.

Q. Explain Virtual Memory architecture with neat diagram.

PYQ Trend Analysis

Topic	Frequency
Virtual Memory	⭐⭐⭐⭐⭐⭐
Demand Paging	⭐⭐⭐⭐⭐⭐
Page Fault	⭐⭐⭐⭐⭐⭐

Demand Paging Diagram	⭐⭐⭐⭐⭐
Comparison Tables	⭐⭐⭐⭐⭐

Expected 2026 Questions

🔥 Explain Virtual Memory with diagram.

🔥 Explain Demand Paging.

🔥 Explain Page Fault handling mechanism.

🔥 Compare Paging and Demand Paging.

🔥 Explain implementation of Virtual Memory using Demand Paging.

One-Minute Revision

```
Virtual Memory

↓

RAM + Disk

↓

Demand Paging

↓

Load Page On Demand

↓

Page Fault

↓
```

```
        Load From Disk

            ↓

      Continue Execution
```

Memory Trick: VDPB

```
V → Virtual Memory

D → Demand Paging

P → Page Fault

          B → Backing Store
```

🎯 **Exam Tip:** Virtual Memory ke answer me **Virtual Memory Diagram + Demand Paging Diagram + Page Fault Handling Diagram** zarur banao. Ye Unit-3 ka highest probability 14-mark question hai aur RGPV me bahut baar repeat hua hai. 🚀📚