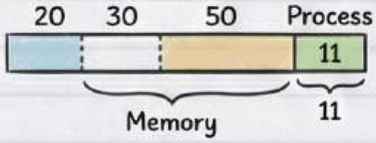
# Operating System
# Module-5 Notes

by pyqfort.com

## Contents Covered:

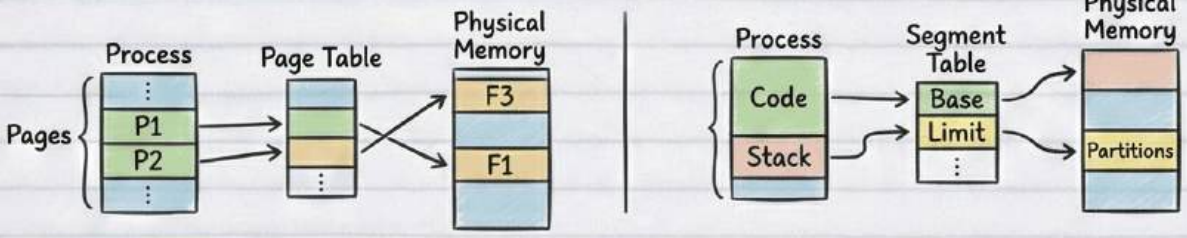
- Intro to Memory Management

- Memory Allocation

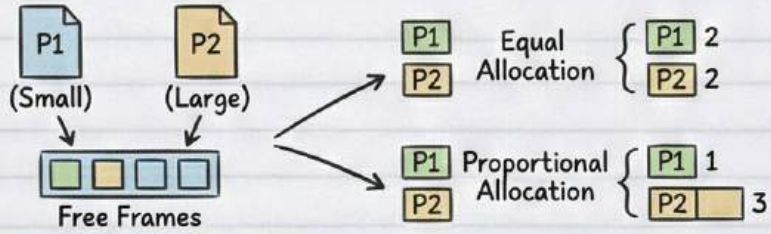



| Algorithm | Hole Selected | Comment |
|---|---|---|
| First Fit | 30 | First sufficient |
| Best Fit | 20 | Smallest sufficient |
| Worst Fit | 50 | Largest sufficient |

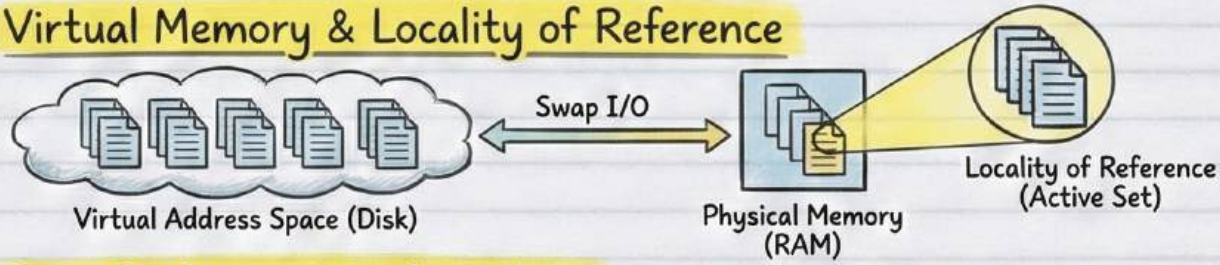
- Paging and Segmentation



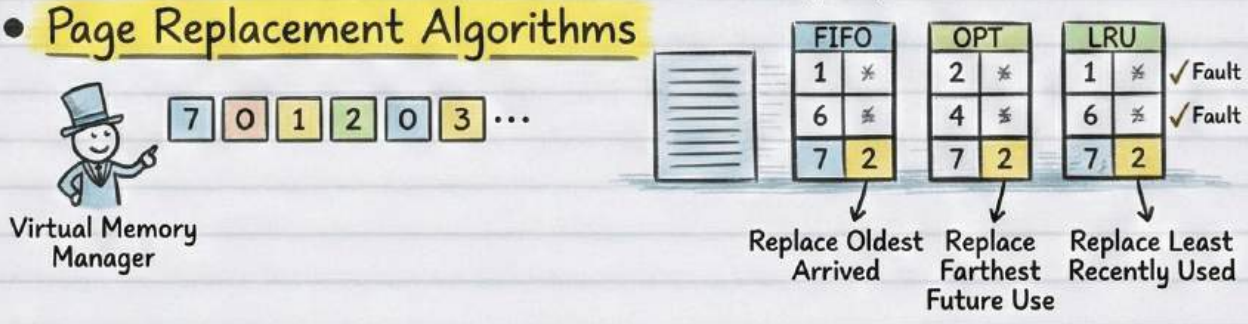

- Page Allocation




- Virtual Memory & Locality of Reference




- Page Replacement Algorithms

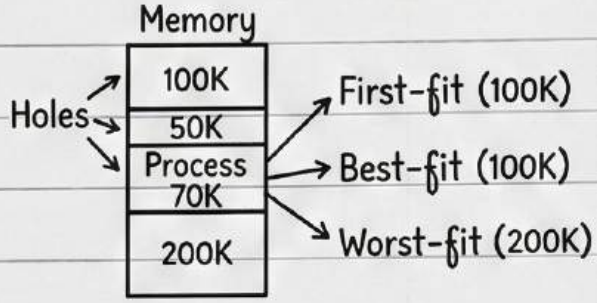


# [illegible]

[illegible] a program [illegible] [illegible] [illegible]

# Contiguous Memory Allocation

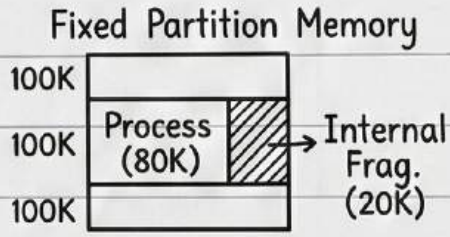
These techniques are algorithms that satisfy the memory requirement of a process from the list of available free holes. They are known as partition selection algorithms.




- There are primarily three types: first-fit, best-fit, and worst-fit.

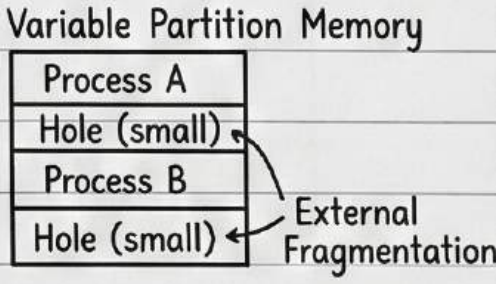
## Fixed and Variable Partition

- Fixed Partitioning: Memory is partitioned at the time of system generation. Partition size is fixed & cannot be changed.




- It suffers from both internal and external fragmentation.

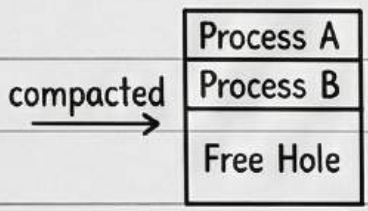
- Variable Partitioning: Number and size of partitions are created at run-time by the OS. A free memory partition is a hole.




- It suffers from external fragmentation.

## Compaction

Compaction method is used to reduce memory wastage in variable partitioning.



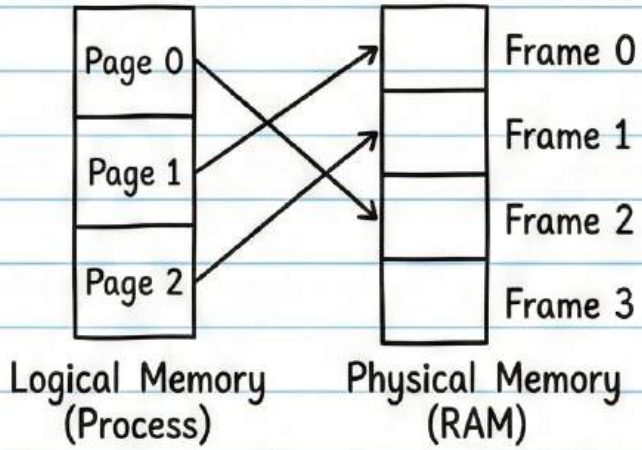
# Paging and Segmentation

## 1. Paging

Paging is a **non-contiguous** allocation method where physical memory is divided into fixed-size blocks called **frames.**

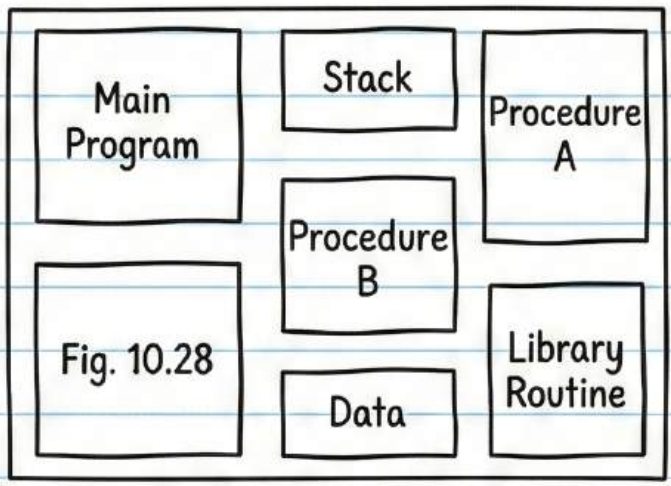
The logical memory of a process is also divided into blocks of the **same size** called **pages.**

This eliminates **external fragmentation.**




## 2. Segmentation

Segmentation is a **memory management technique** where a **logical address space** is divided into **variable-sized segments.** These segments represent logical parts of a program like main program, stack, procedures, etc.



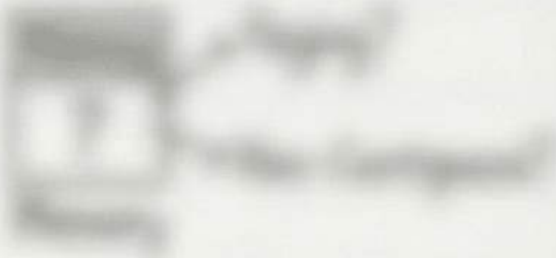

Logical Address Space

## Key Difference

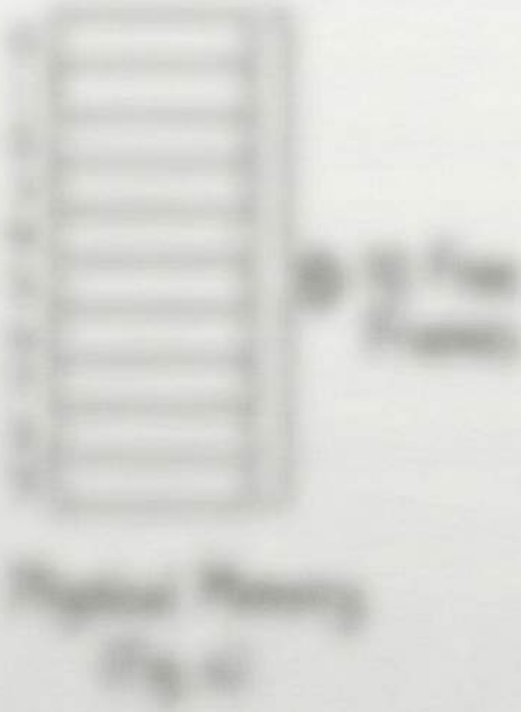
- **Pages** are **physical** divisions of **equal** size.
- **Segments** are **logical** divisions of **variable** size.

## Principle of Operation: Initial Free Frames Setup

To understand the paging concept and the benefits of a non-contiguous method, let's look at an example.




Initially, the physical memory consists of a set of available free frames. In our example (Fig. a), there are 10 free frames in the memory.




Physical Memory
(Fig. a)

These frames are numbered from 0 to 9 and are ready for allocation.

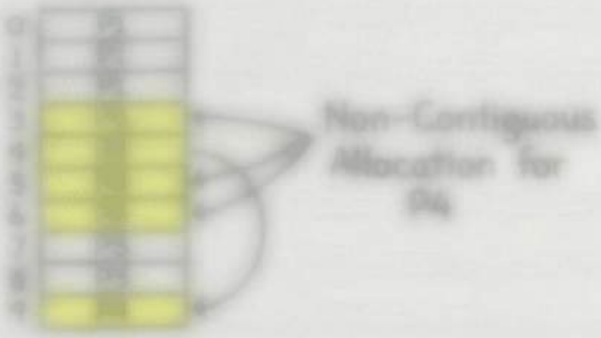
## Initial Contiguous Page Allocation (An Example)

Let's consider a physical memory with 10 free frames (0 to 9) and four processes: P1 (3 pages), P2 (4 pages), P3 (2 pages), and P4 (5 pages).

Initially, P1 is allocated the first three contiguous frames (0, 1, 2). Then P2 gets the next four (3, 4, 5, 6), and P3 gets the next two (7, 8).

Now, only one frame (9) is free. P4 needs 5 frames and cannot be allocated.

Later, P2 finishes and releases its frames (3, 4, 5, 6). Now there are 5 free frames (3, 4, 5, 6, 9). These are allocated to P4.



## Process Completion & Memory Release

When a process finishes its execution, it relinquishes the memory frames. If some entry frames frames have become free frames, available for other processes.

Example: An element in a process fly, where PCB finishes, frames 1, 4, 5, and 6 are released.

These newly available, non-contiguous frames can then be allocated for a new process (e.g., PCB).

## Logical to Physical Address Translation (Cont.)

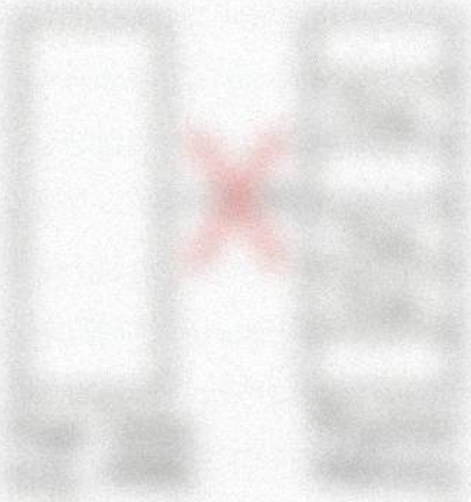
As an example, the page number (p) helps us find the base address (b) for frame number from the page table.

Specifically, it is added to the offset (d) to calculate the final, exact physical address in memory.

# Page Allocation

## The Problem: Large Page Tables

Modern systems with large address spaces (e.g., 32 bits) require large page tables that consume megabytes of memory.



1. If 4 KB page size can lead to a 4 MB page table.
2. If a different to fast configuration space for each large table.

## The Solution: Hierarchical Paging

Multiple page tables are configurable, conforming them to memory.

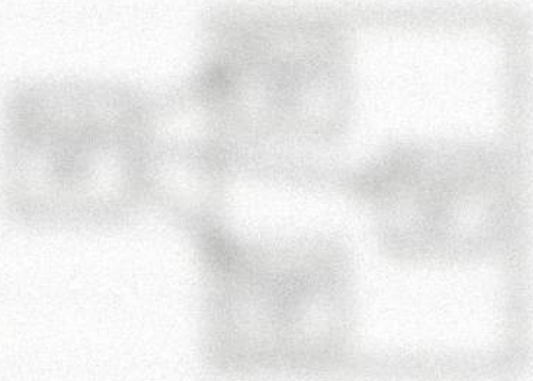
- To manage this, the page table itself is paged into a called hierarchical paging.
- This can be extended to multi-level or hierarchical page table structures.

# Multi Level Pyramid Plan Level

## Plan Problem

[illegible]

## Plan Solution: Plan Level Pyramid



[illegible]

## Logical Problem Structure

[illegible]

## Problem Translation

[illegible]

# Three Level Paging

## Why is it needed?

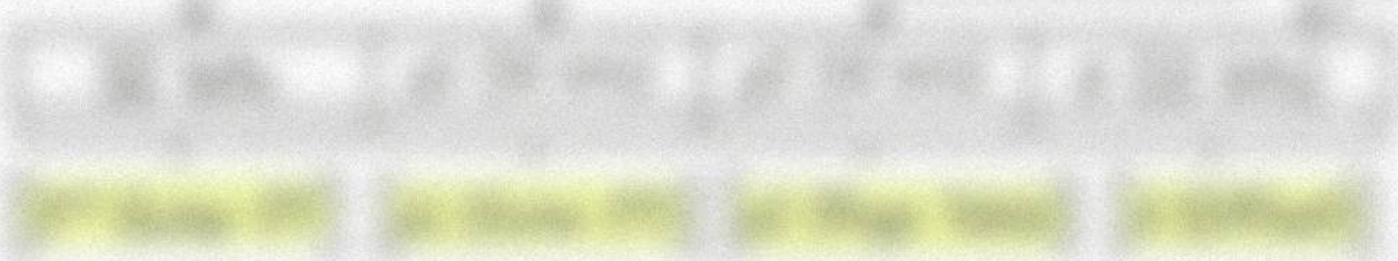
For 64-bit systems, a two-level outer page table (directory) can become too large to fit contiguously.

## The Solution

The directory is further divided to be allocated non-contiguously.

This creates three level paging with four outer page tables and one page table.

Logical Address Structure (on our 64-bit)




Translation uses each part to navigate the memory to find the frame, then add the offset.

# Disadvantage of Multi-Level Paging

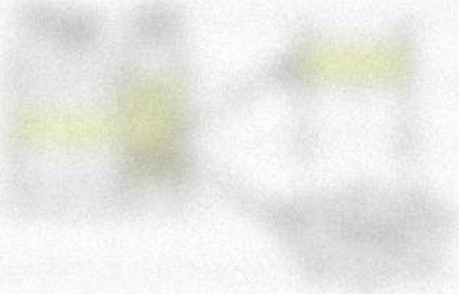
- The main disadvantage is that it increases the memory access.
- As the number of levels increases, the number of memory accesses also increases.
  - Example page table has memory accesses.
  - Two level paging has memory accesses.
  - Three level paging has memory accesses, and so on.
- Which of improper memory can be configurable, the number of memory accesses is large.
- The number of accesses can be reduced with the help of memory mapping using a Full Transition Lookout Buffer, increasing performance.

## Production and Marketing

Production is a single function that has several specialized roles for business operations. Production



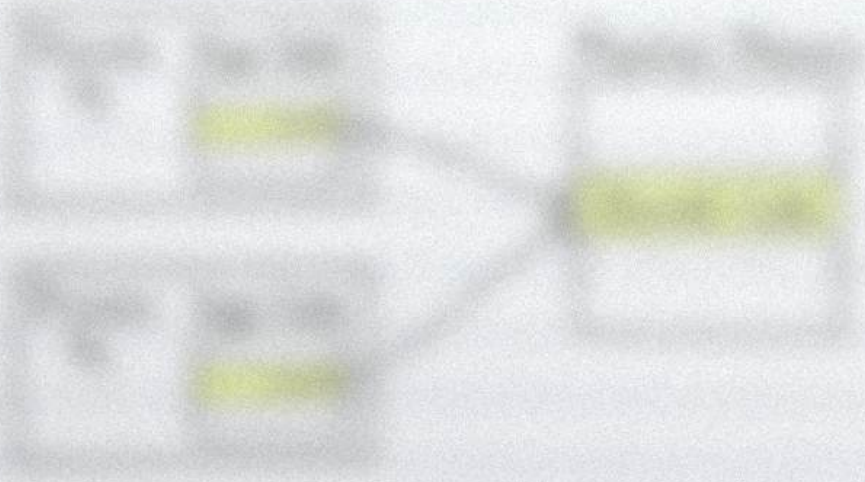
A typical manufacturer will organize a group for each product line with one key personnel as business manager for each, though smaller



## Marketing

The marketing mission involves discovering what the customer buy and satisfying needs.

Marketing uses research that seeks to find out what people want. Marketing



## Relationships of Angles

A ray is a part of a line. It starts at a point and goes on forever in one direction.

The point where two rays meet is called the vertex of the angle. The rays are the sides of the angle.

## Vocabulary

The word **acute** comes from the Latin word *acutus*, which means "sharp." An acute angle looks like the point of a sharp pencil.

When two angles, such as $\angle ABC$ and $\angle CBD$, share a side and a vertex but do not overlap, they are called **adjacent angles**.

The sum of the measures of adjacent angles is equal to the measure of the angle formed by the two outer rays.

$m\angle ABC + m\angle CBD = m\angle ABD$

## Theorems

**Theorem**

**Vertical Angles Theorem**

If two angles are vertical angles, then they are congruent.

If $\angle 1$ and $\angle 2$ are vertical angles, then $\angle 1 \cong \angle 2$.

## Virtual Memory & Locality of Reference

- In virtual memory, the virtual premises is not needed into memory. Only required portions are loaded.
- This is supported by the principle of locality of reference.

Having executive memory reference tend to cluster. The cluster changes over a long period of time.

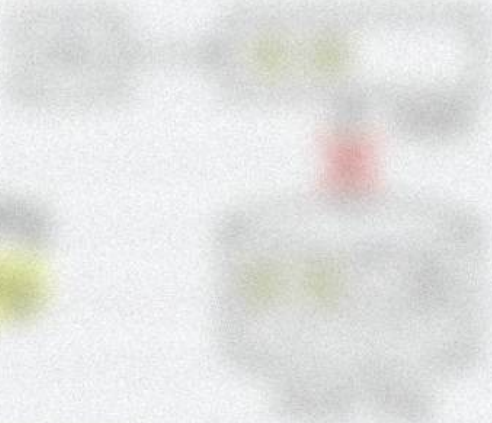
Types of Locality

1. Temporal locality: Recently referenced locations are likely to be referenced again (e.g., loops, subroutines).

2. Spatial locality: Nearby memory locations are referenced (e.g., array traversal, sequential execution).

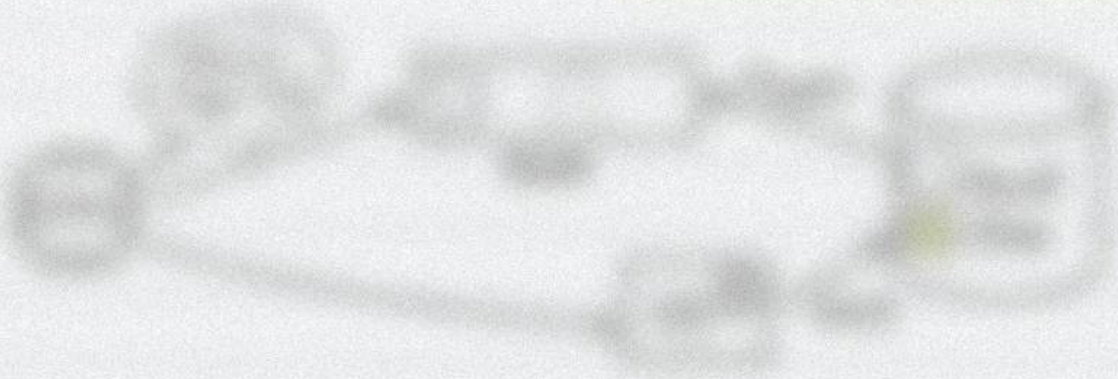
# Demand Paging

- Demand paging is a kind of virtual paging in the memory that are needed at an instant of time of execution.



- The demand paging is the only concept behind virtual memory.
- The thumb rule is that more than a component of a process when it is needed.

- Demand is given rise to many other concepts. Some of them are not needed, memory allocation becomes difficult.
- Further, page faults occur when the pages of segments are not found in the memory. Therefore, the page fault must be handled appropriately.



# Page Replacement Algorithms

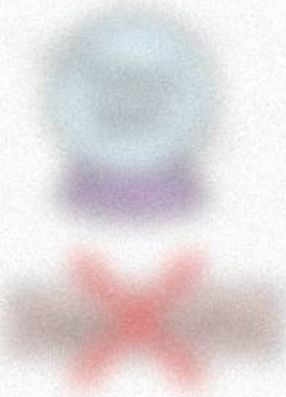
- A strategy to replace an existing page in that a page that needs a page fault can be paged in a lesser as a page replacement algorithm.

## 1. FIFO (First In First Out)

- The oldest page is chosen.
- Implemented through a FIFO queue.
- Page fault rate is high.
- Suffers from Belady's anomaly.

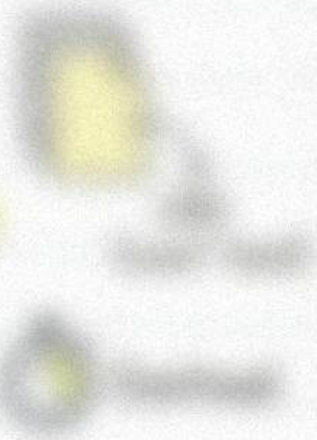
## 2. Optimal

- The page that will not be referenced for the longest time is replaced.
- No way of knowing the future accuracy, reference, therefore cannot be implemented.



## 3. LRU (Least Recently Used)

- Replaces a page that has not been used for the longest period of time in the past.
- Implemented through a short-term algorithm.
- Provides the number of page faults.
- Difficult to implement and cause additional



Full content is
available at
'topmate.io/saumy1905'
or you can navigate
to the specific section
of the subject on
'pyqfort.com'
to access it