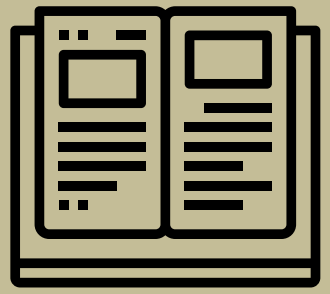
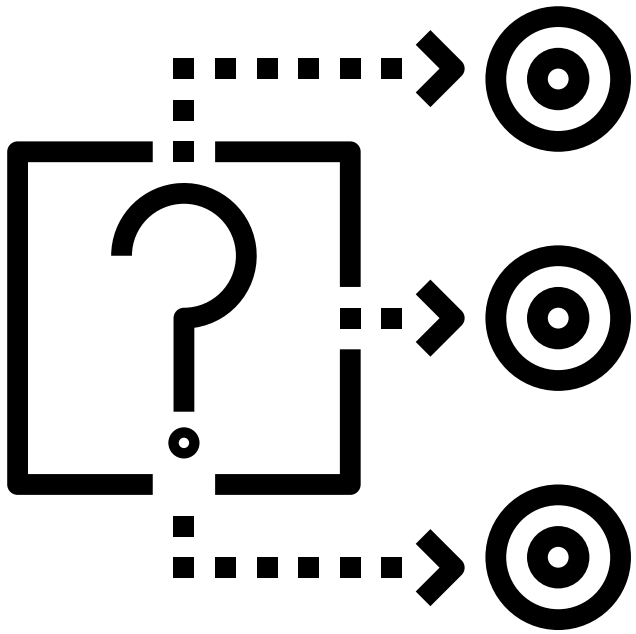


MEMORY

DCS4103 Operating System



Memory Management

Introduction

- Help the computer system in allocating the main memory space
  - to the processes and their data at the time of execution.
- Upgrading performance
- Execution of multiple processes
- Utilization of memory space
- Helps OS to keep track of every memory location

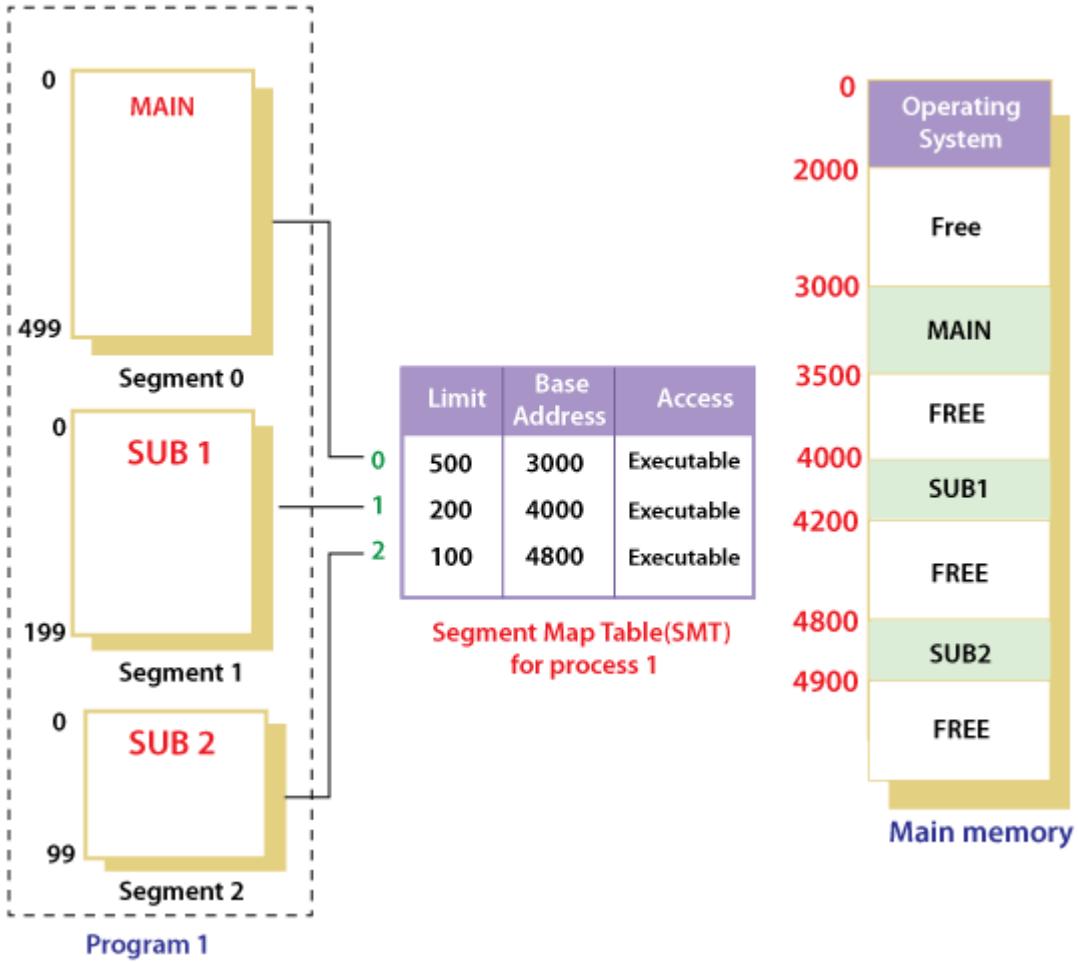
Effectiveness

- Correct relocation of data
- Protection of data from illegal change
- Provision to share the information
- Utilization of small free space

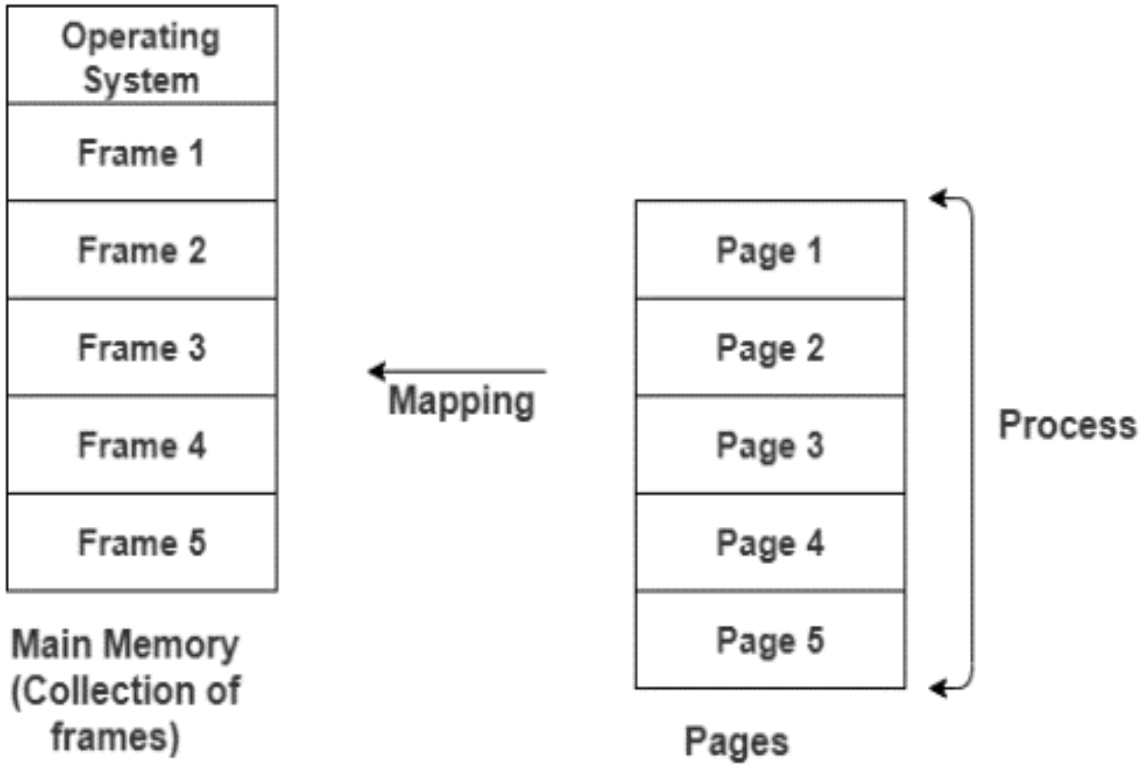
Techniques

Segmentation

- The technique of dividing the physical memory space into multiple blocks
- Segment
  - Each block specific length
  - Determines the availability memory space
- Base Address
  - Starting address
- Displacement or offset value
  - The distance between the actual position of data and the base address of segment



Techniques



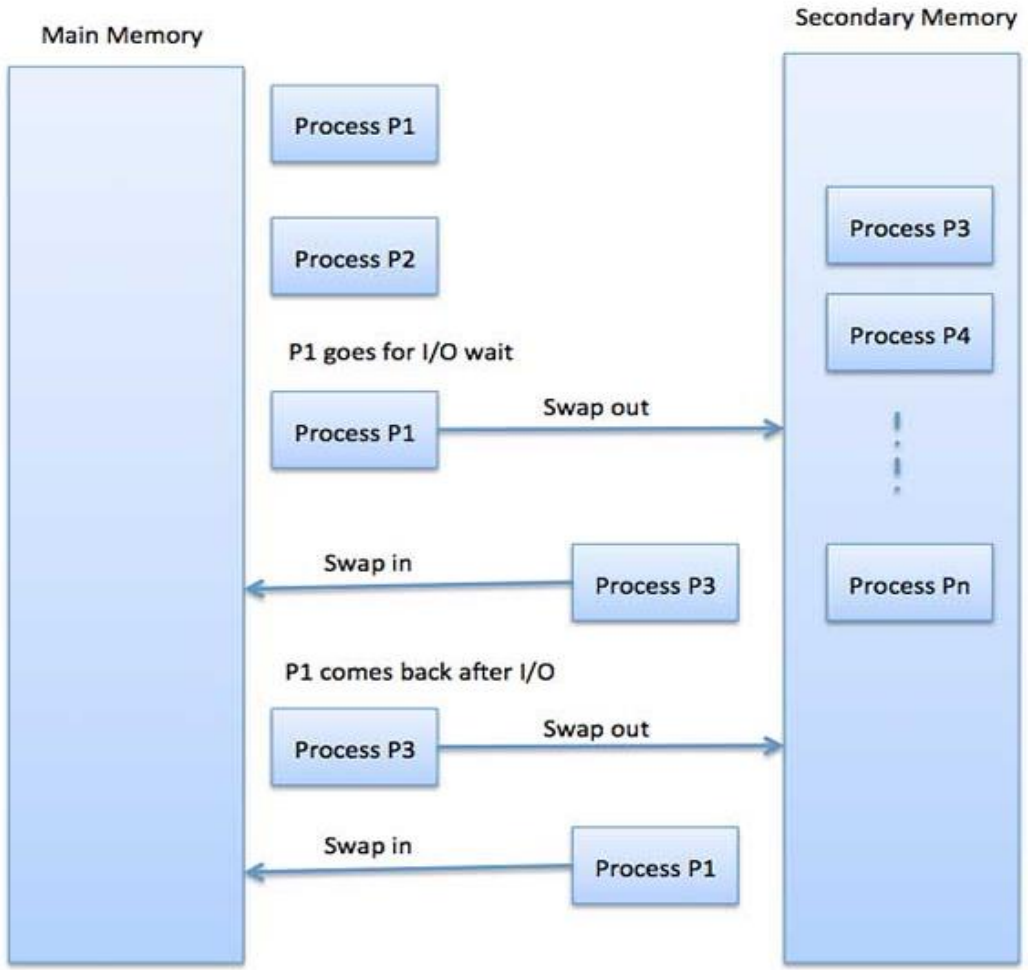
Paging

- The main memory of computer system is organized in the form of equal sized blocks called pages
- Page table
  - Store the address of occupied pages of physical memory
- Virtual address
  - Obtain data from the physical memory location without specifying lengthy memory address in the instruction

Techniques

Swapping

- Swap the process temporarily from the main memory to the backing store
- Backing store
  - Hard disk or some other secondary storage device
  - Offer direct access to the memory images
- Swapping in
  - Place the pages or blocks of data from the hard disk to the main memory
- Swapping out
  - Removing pages or blocks of data from main memory to the hard disk



Allocation

- A process by which computer programs are assigned memory or space.
- Low Memory
  - Operating System
- High Memory
  - User processes

Schemes

First Fit	Best Fit	Worst Fit	Next Fit
First sufficient block from the beginning of the main memory.	First smallest partition among the free partitions	Largest sufficient freely available partition in the main memory	Search for the first sufficient partition from the last allocation point.

Fragmentation

- Processes not able to allocate to memory blocks because its small size and memory blocks always remain unused
- Insufficient space to load another process
  - Dynamic allocation of main memory processes.

Advantages

- Fast Data Writes
- Fewer Failures
- Storage Optimization

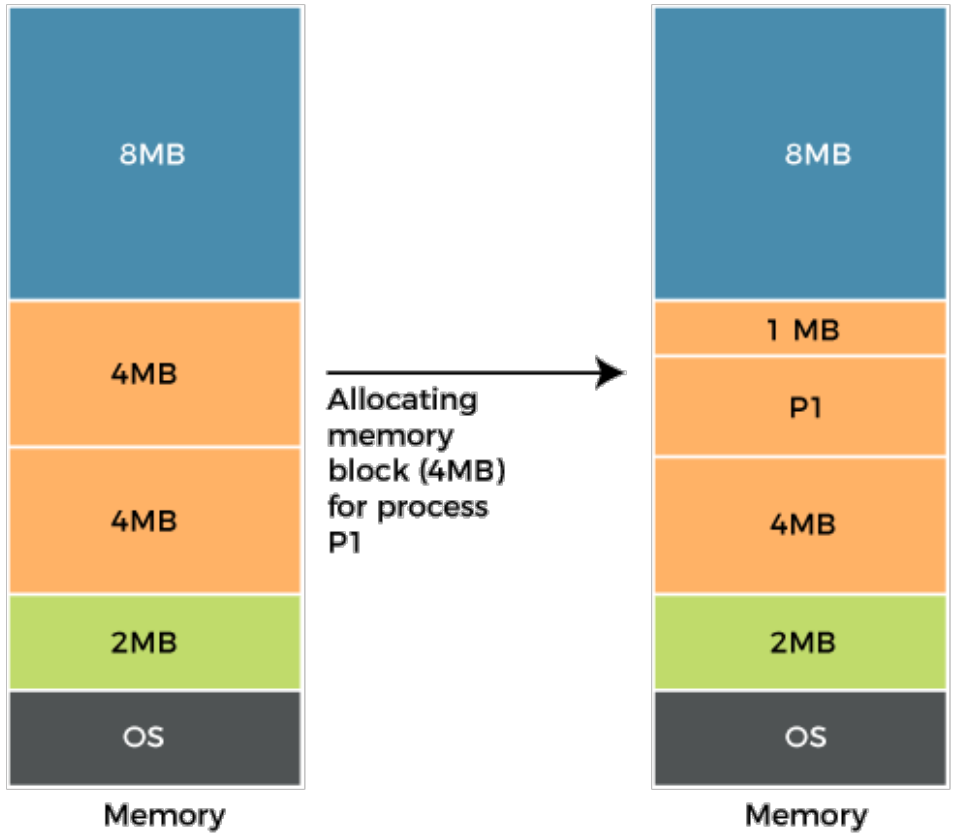
Disadvantages

- Need for regular defragmentation
- Slower Read Times

Fragmentation

Internal Fragmentation

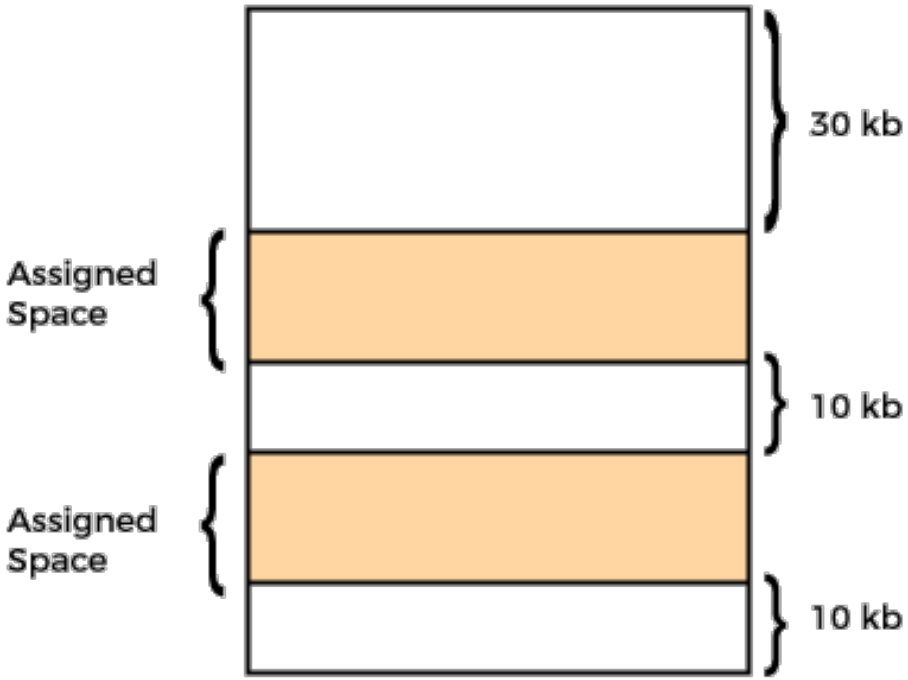
- Unused free space of the memory block
- Memory space allocate to the process is smaller than the amount of memory requested
- Solution
  - Assign the smallest partition, which is still good enough to carry the entire process.



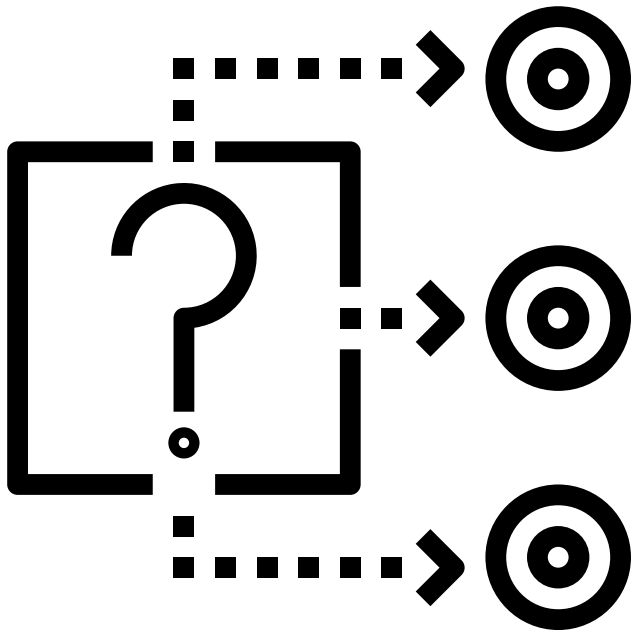
Fragmentation

External Fragmentation

- Enough memory space to complete a request, but it is not contiguous
- Dynamic memory allocation method allocates some memory
  - However, it leaves a small amount of memory unusable
- Reduce quantity of available memory
- Solution
  - Rearrange memory contents to place all free memory together in a single block.



Process 05 needs 45kb memory space



Virtual Memory

Introduction

- A storage scheme that provides user an illusion of having a very big main memory
- Transferring pages of data from random access memory to disk storage.
- Paging file
  - Space for virtual memory to move data from RAM
  - Freed up for computer to complete the task

Advantages

- Increase degree of Multiprogramming
- User can run large application with less real RAM.
- No need to buy more memory RAMs.

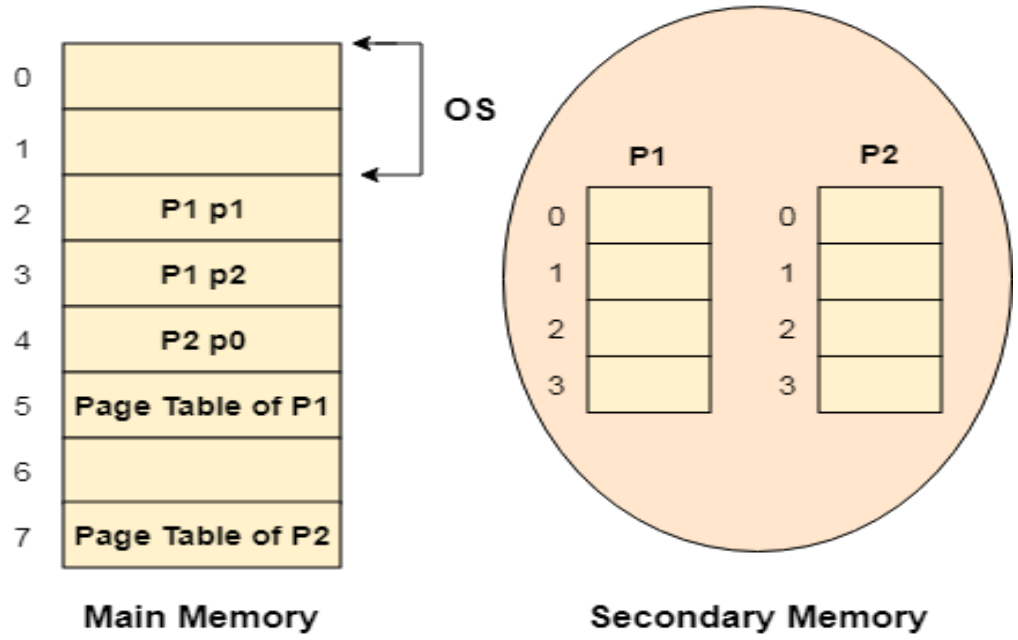
Disadvantages

- The system becomes slower since swapping takes time.
- More time in switching between applications.
- Lesser hard disk space for user's use.

Introduction

	Frame	Present / Absent	D Bit	Reference bit	Protection
0		0	0	0	
1	2	1	0	1	
2	3	1	1	0	
3		0	0	0	

Page Table of P1

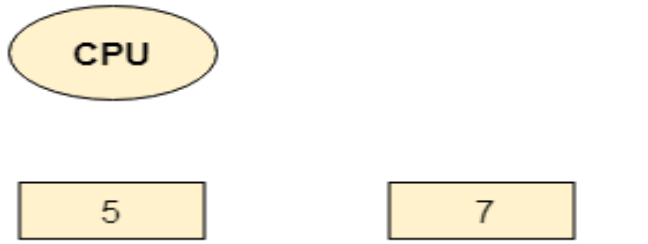


Main Memory

Secondary Memory

	Frame	Present / Absent	D Bit	Reference bit	Protection
0	4	1	0	1	
1		0	0	0	
2		0	0	0	
3		0	0	0	

Page Table of P2

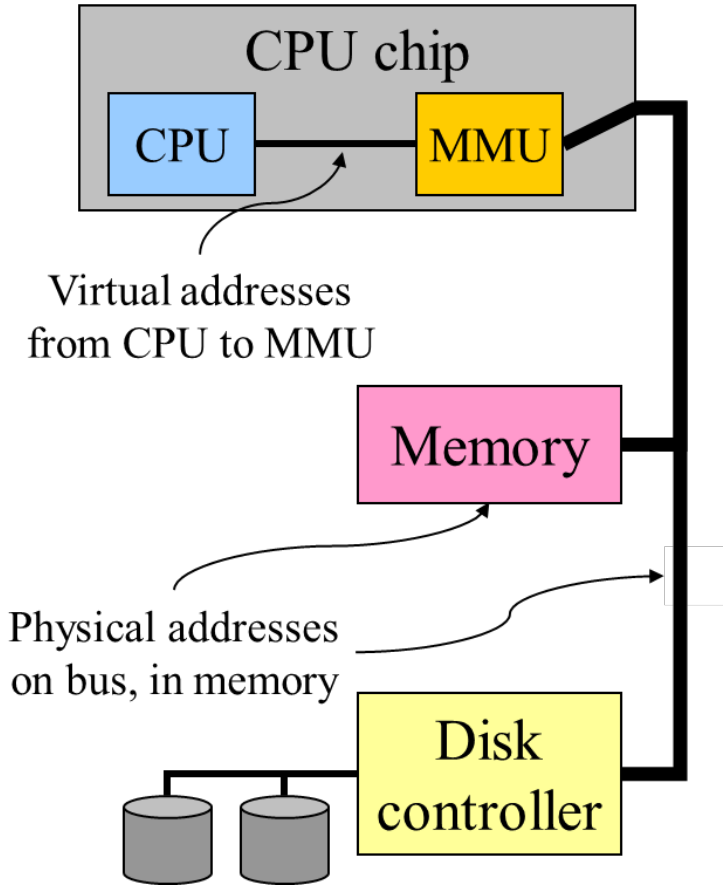


Page Table Base (P1)

Page Table Base (P2)

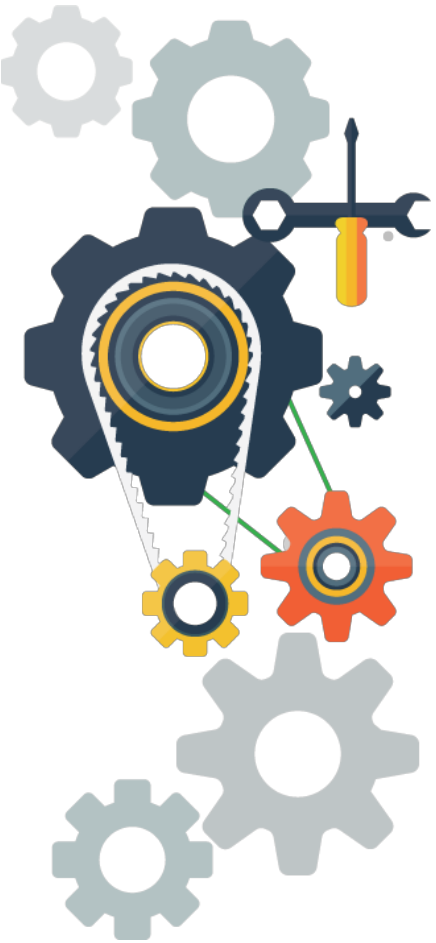
Virtual and Physical Addresses

- Program uses virtual addresses
  - Addresses local to the process
  - Hardware translates virtual address to physical address
- Translation done by the Memory Management Unit
  - Usually on the same chip as the CPU
  - Only physical addresses leave the CPU/MMU chip
- Physical memory indexed by physical addresses

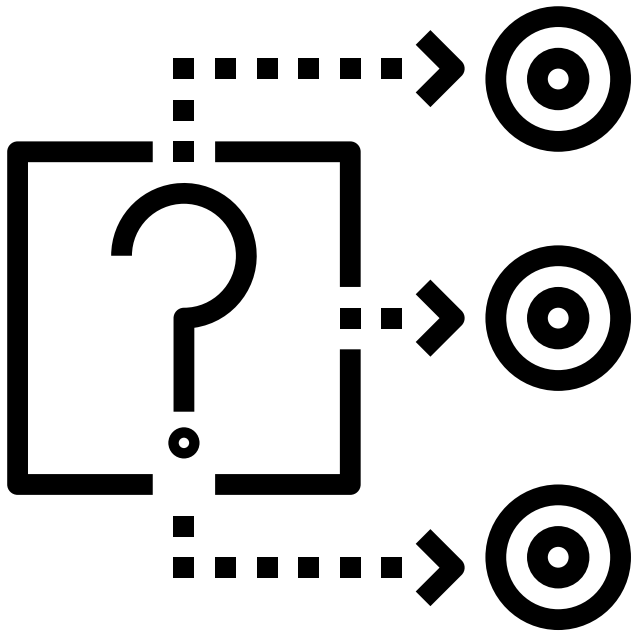


Demand Paging

- The pages of a process which are least used, get stored in the secondary memory.
- A page is copied to the main memory when its demand is made, or page fault occurs.
- Page replacement algorithms
  - Select the pages which will be replaced



First In First Out (FIFO)	Optimal Algorithm	Least Recently Used
Has been in the virtual address of the memory for the longest time.	The time to the next reference is the longest. Difficult implementation	Has not been used for the longest time Easy implementation



File Management

Introduction

Objectives

- ❑ Provides I/O support for a variety of storage device types.
- ❑ Minimizes the chances of lost or destroyed data
- ❑ Helps OS to standardized I/O interface routines for user processes.
- ❑ Provides I/O support for multiple users in a multiuser systems environment.

File structure	Description
Text	Series of characters that is organized in lines.
Object	Series of bytes that is organized into blocks.
Source	Series of functions and processes

Attributes

Attributes	Description
Name	▪ Information stored in a human-readable form ▪ One directory cannot have two files with the same name.
Identifier	▪ Extension which identifies the type of the file
Location	▪ Points to file location on device.
Type	▪ Vidéo files, audio files, text files, exécutable files
Size	▪ The number of bytes acquired by the file in the memory.
Protection	▪ Set of permissions to the different group of Users.
Time, date and security	▪ The time and date on which the file is last modified ▪ For protection, security, and used for monitoring

Operations

Operation	Description
Create	Create a file in the file system
Open	Invoke the open system call and passes the file name to the file system
Write	Write the information (name, length) into a file.
Read	Reads the contents from a file
Re-position or Seek operation	Call re-positions (forward/backward) the file pointers from the current position to a specific place in the file
Delete	Delete directory entry, regain disk space.
Truncate	Delete the file except deleting attributes
Close	Deallocate all the internal descriptors that were created when the file was opened.
Append	Adds data to the end of the file.
Rename	Rename the existing file.

Access Method

Sequential

- Access records in a certain pre-defined sequence
- Process information one by one
- Example : Compilers

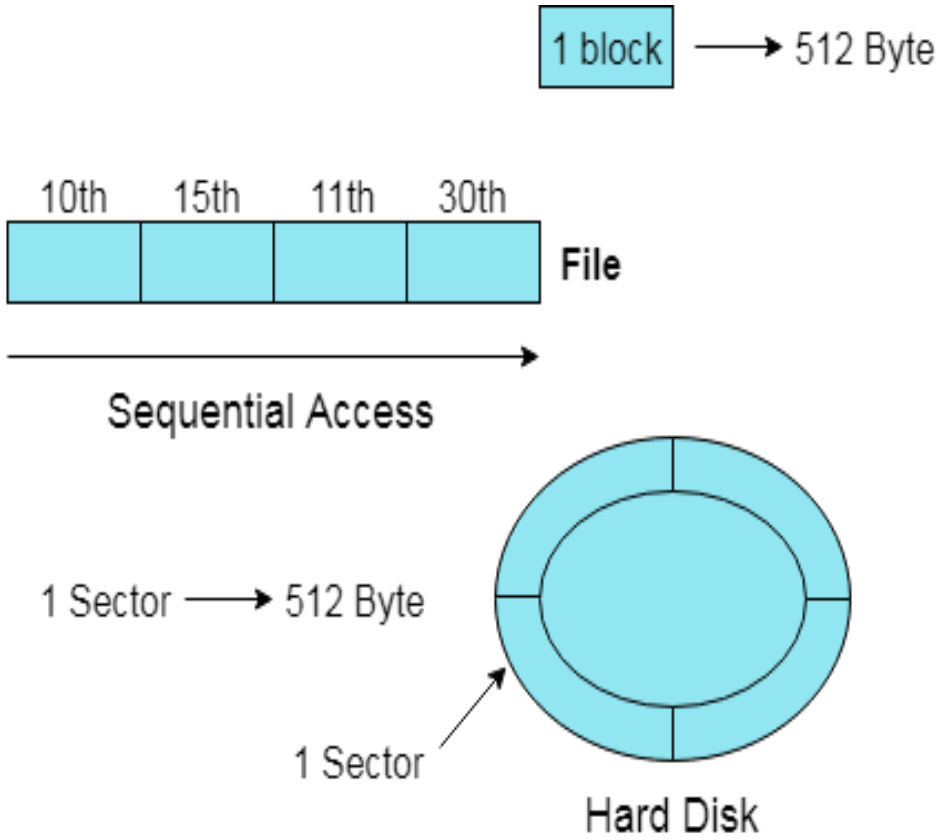
Direct Random

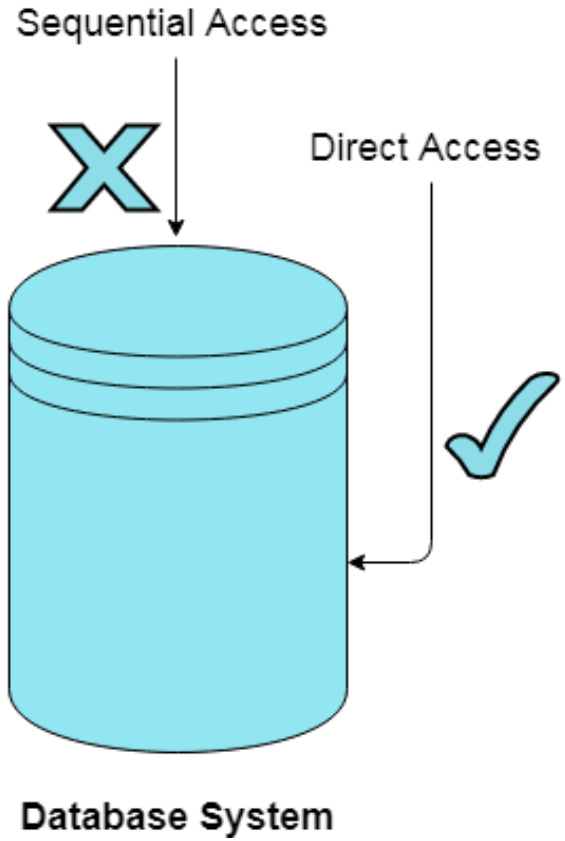
- Access records directly
- Each records has its own address on which can be directly accessed for reading and writing

Index Sequential

- Index is searched sequentially, and its pointer can access the file directly
- Use multiple index level
- Reduces the time needed to access a single record

Access Method





Space Allocation

Contagious

- Directory comprises the addresses of index blocks of the specific files.
- An index block is created, having all the pointers for specific files.
- All files should have individual index blocks to store the addresses for disk space.

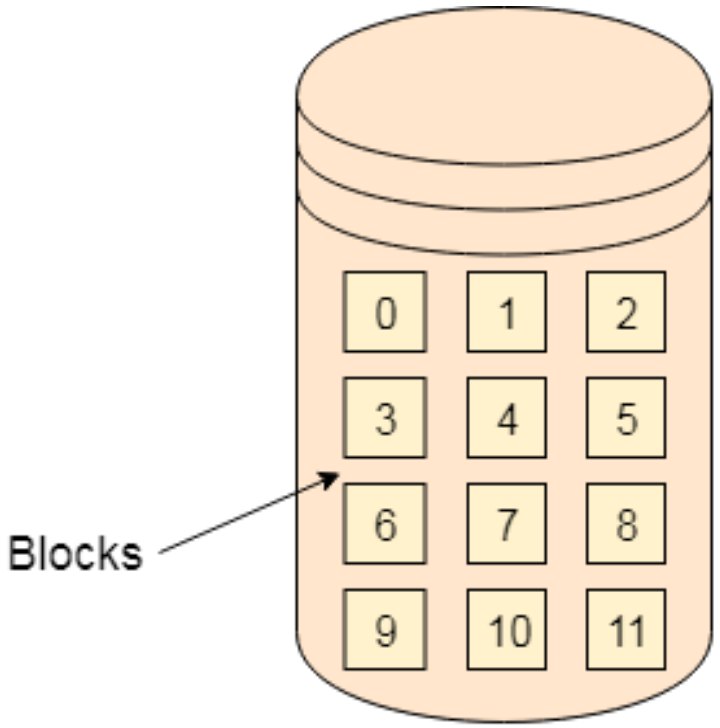
Advantages

- Simple to implement.
- Excellent read performance.
- Supports Random Access into files.

Disadvantages

- The disk will become fragmented.
- Difficult to have a file grow.

Space Allocation



Hard Disk

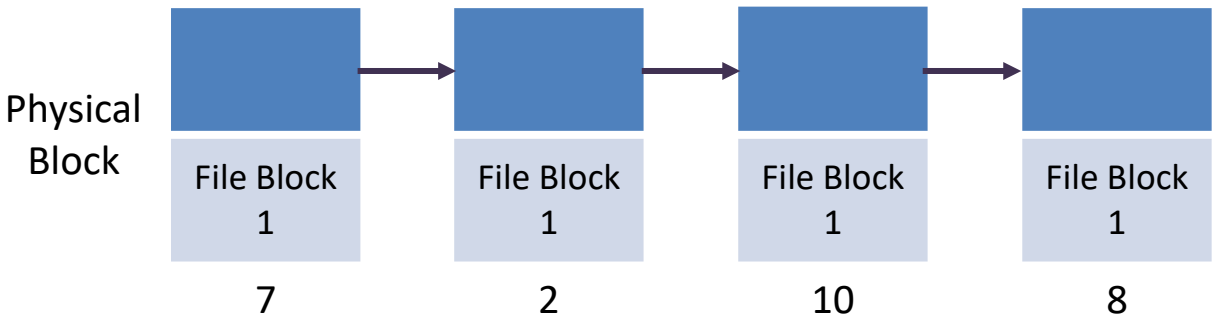
File Name	Start	Length	Allocated Blocks
abc.text	0	3	0,1,2
video.mp4	4	2	4,5
jtp.docx	9	3	9,10,11

Directory

Space Allocation

Linked List

- The directory contains a link or pointer in the first block of a file.
- No external fragmentation
- Used for sequential access files.
- Not ideal for a direct access file



Linked List Allocation

Advantages

- No external fragmentation with linked allocation.
- Any free block can be utilized
- File can continue to grow
- Directory entry will only contain the starting block address.

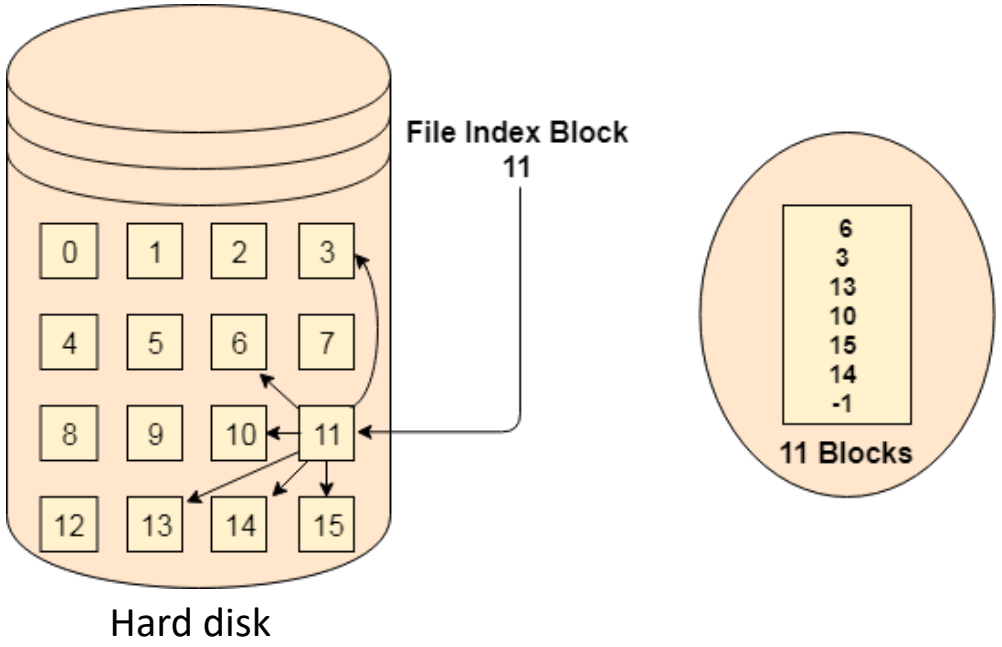
Disadvantages

- Random Access is not provided.
- Pointers require some space in the disk blocks.
- Any of the pointers in the linked list must not be broken
- Need to traverse each block.

Space Allocation

Indexed

- Every file users a contiguous address space on memory.
- The OS assigns disk address is in linear order.
- External fragmentation is the biggest issue.



Advantages

- Supports direct access
- A bad data block causes the lost of only that block.

Disadvantages

- A bad index block could cause the lost of entire file.
- Size of a file depends upon the number of pointers; an index block can hold.
- Having an index block for a small file is totally wastage.
- More pointer overhead

Revision Questions

1. Name the file operations.
2. Discuss and illustrate the file allocation methods.
3. Explain the page replacement algorithms.