DIPLOMA WALLAH

(Your One Stop Hub For Diploma Resources)

OPERATING SYSTEM

AND ADMINISTRATION

Complete Notes Based on Full Syllabus

- Diploma Engineering

4th Semester

www.diplomawallah.in

Notes prepared by **Sangam**

Sagar Sangam

Unit - 06

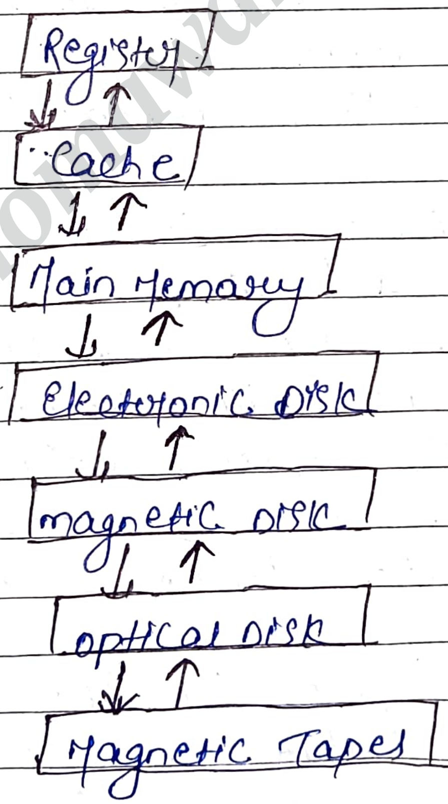
~~Network Management~~

Memory management

Memory management is a critical aspect ~~also~~ of operating system that ensure efficient use of the computer memory resources.

It controls how memory is allocated and deallocated to ~~processes~~ processes, which is ~~the~~ key to both performance and stability.

Below is detailed overview of the various components and technique involved in memory management.



It is important because:-

- Allocate and de-allocate memory before and after process execution.
- To keep track of used memory space by process.
- To minimize fragmentation issue. (The process of dividing computer file.)
- To proper utilization of main memory.

Linking

Linking means combining object files and libraries to make single executable.
It can happen at compile time (static) or run-time (dynamic).

- Static linking:- In static linking, the linker combines all necessary program modules into a single executable program. So there is no runtime dependency.
  Some OS support only static linking in which system language libraries are treated like any other object module.

Ex- In early OS like 'DOS' programs were distributed as a single .exe with all required code inside. (Static linking).

- They did not depend on external .dll file.

Date ______
Page ______

• Dynamic Linking :-
The basic concept of dynamic linking is similar to dyamic loading. In dynamic linking, "Stub" is included for each appropriate library routine reference. A stub is a small piece of code. When the stub executed it check whether the needed routine is already in memory or not. If not available then the program loads the routine into memory.

*Loading
Loading a process into the main memory is done by a loader. There are two different types:-
• Static loading:- Loading is basically loading the entire program into a fixed address. It requires more memory space.
• Dynamic Loading:- Dynamic loading loads program routines into memory only when they are needed. This saves memory by not loading unused routines.
The routines remain on disk is relocatable (Can be loaded at any memory location) format until called.
It allow better memory utilization, especially for large program.

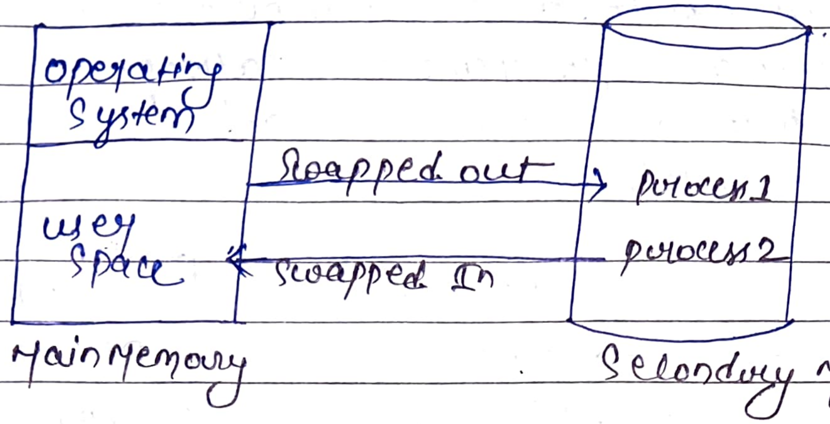
* Swapping

Swapping is a memory management technique in which an entire process is temporarily transferred (swapped out) from the main memory (RAM) to swap space on secondary storage (hard disk/SSD), and later brought back (swapped in) to the main memory for execution.

- The CPU can only work with process as that in RAM.
- If RAM → full, OS choose inactive process and moves it to the swap space (a reserved portion of disk).
- This frees RAM for other active processes.
- When the swapped-out process is needed again, it is brought back (swapped in) to RAM.

Key points

1. Swap space :- The reserved disk area used for swapping (Linux → swap partition, Windows → pagefile.sys).
2. Swapping out :- Moving a process from RAM → swap space.
3. Swapping in :- moving a process from swap space → RAM.
4. Advantage :- supports multiprogramming even with limited RAM.
5. Disadvantage :- Disk is much slower than RAM → frequent swapping (thrashing) reduces performance.



* Memory allocation
It is a process by which the OS assign memory blocks (RAM space) to programs, processes or data, so that they can be executed efficiently.

Types
1. Static Memory Allocation
- Memory is assigned to a program before execution (at compile time).
- Once allocated the size cannot be change.

2. Dynamic Memory Allocation.
- Memory is assigned during program execution (run-time).
- Size can be changed while the program running.

* Memory allocation in OS.

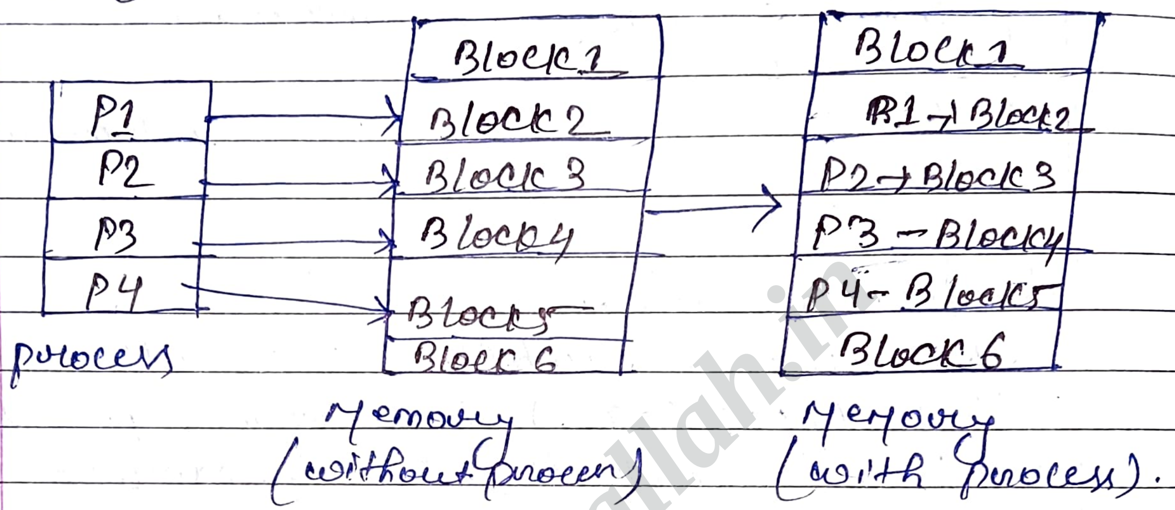
1. Contiguous Memory Allocation.
- Memory allocation is a memory management method where each process is given a single, continuous block of memory.

(i) Static fixed Portition

(ii) Dynamic fixed portition

This mean all the data for a process is stored in a single memory locations.

- easy to manage but can lead to fragmentation.



* Non-Contiguous memory allocation
This method allows processes to be broken into smaller parts, which are placed in different, non-adjacent memory locations.

Technique for non-contiguous memory allocation:-

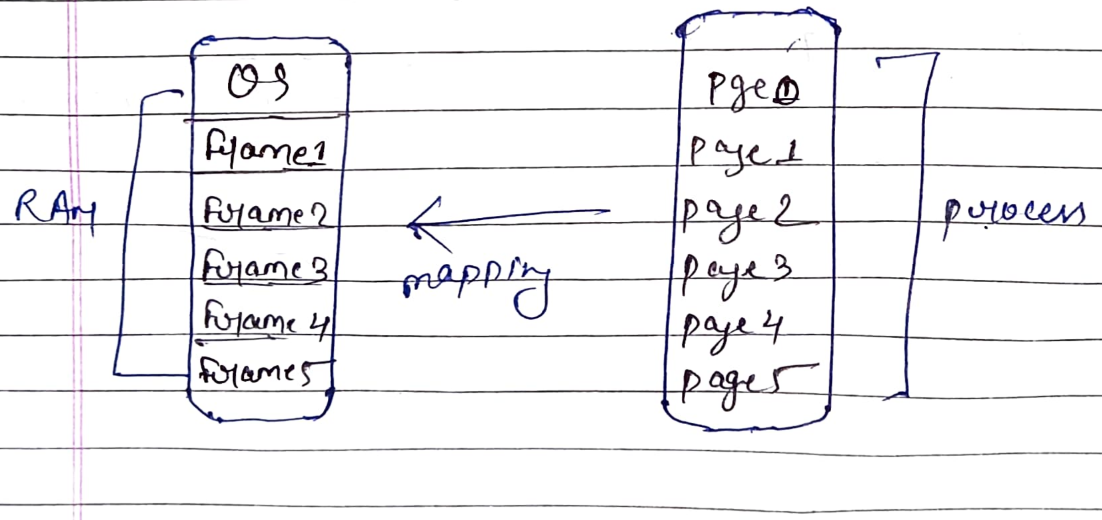
- pagging:- The process is divided into fixed-size blocks called "pages", and the memory is divided into blocks of the same size called "frames". The OS keeps a page table to map logical pages to physical frames.

working

- A process is divided into pages (eg. 4Kb each)
- RAM is divided into frames of the same size (4Kb each).
- The OS finds free frame and places pages there.
- A page table maps logical page number → physical frame numbers.
- When CPU generates a logical address, it is translated using the page table to the actual physical address in RAM.

Ex- • process size = 10KB
- process divided into 3 page (pge 0, 1, 2)
- RAM divided into 4KB frames.
- page can go into any available frames (not necessary contiguous)

page 0 → frame 5
page 1 → frame 2
page 2 — frame 9.

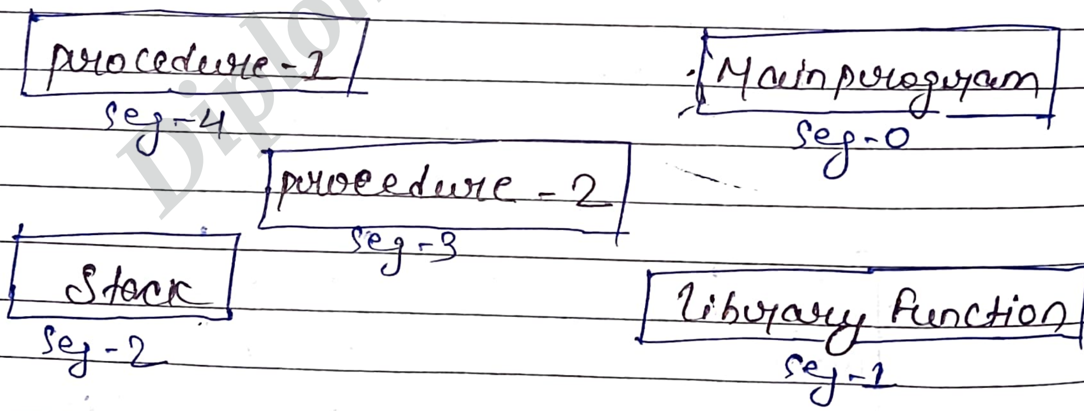


* Segmentation

The process is divided into segment of varying sizes, such as Code, data, stack etc. The OS system maintain a segment table to map logical segments to physical memory.

Key points

- Segment → A variable-sized logical unit (like code segment, data segment)
- Segment Table → Stores the base address (starting point) and length (size) of each segment.
- Logical Address = (Segment number, offset)
- The OS translates this into physical address using the segment table.



Date ____
Page ____

* Fragmentation

Fragmentation is defined as when the process is loaded and removed after execution from memory, it creates a small free hole.

These holes cannot be assigned to new processes because holes are not combined or do not fulfill the memory requirement of the process.

Two types :-

- Internal Fragmentation :- memory allocated fixed size block (partitions) (unused m/m inside alloc block)
  If a process does not fully use the allocated block, the unused space inside the block is wasted.
- External Fragmentation :- Free memory is available but not in a single large block.
  - A process cannot be allocated memory even if the total free memory is sufficient.

* Virtual memory.

Virtual memory is a memory management technique in which the OS uses both RAM and a part of secondary storage (like hard disk/SSD) to give an illusion of having a very large main memory.

In simple words, even if we our computer has less RAM, virtual memory allows programs to run as if there is more memory available by temporarily storing some data on harddisk.

P1 — Pa
P2 — P2
P2 — P3

* Demand pagging

Demand pagging is a memory management technique in which pages are loaded into main memory only when they are required (on demand), instead of loading the entire process at once.

It is a type of lazy loading where the OS loads only the needed part of a program, reducing memory usage and speeding up execution.

* Page Replacement

When a process needs a page that is not in RAM (page fault occurs), the OS must bring it from Secondary memory (disk)

If RAM is already full, the OS must replace (remove) one existing page from RAM to make space.

Common page Replacement Algorithm.

1. FIFO (first in first out) Queue logic.
2. Replace page that not used in longest time future.
3. Least Recently used.
4. Least frequently used.
5. Most Recently used.
6. Clock (second chance algorithm)

Sangam