What is Memory Management?

Memory Management is the process of controlling and coordinating computer memory, assigning portions known as blocks to various running programs to optimize the overall performance of the system.

It is the most important function of an operating system that manages primary memory. It helps processes to move back and forward between the main memory and execution disk. It helps OS to keep track of every memory location, irrespective of whether it is allocated to some process or it remains free.

Swapping:

Swapping is a mechanism in which a process can be **swapped** temporarily out of main memory (or move) to secondary storage (disk) and make that memory available to other processes. At some later time, the **system swaps** back the process from the secondary storage to main memory.

Benefits of Swapping

Here, are major benefits/pros of swapping:

- It offers a higher degree of multiprogramming.
- Allows dynamic relocation. For example, if address binding at execution time is being used, then processes can be swap in different locations. Else in case of compile and load time bindings, processes should be moved to the same location.
- It helps to get better utilization of memory.
- Minimum wastage of CPU time on completion so it can easily be applied to a priority-based scheduling method to improve its performance.

What is Memory allocation?

Memory allocation is a process by which computer programs are assigned memory or space.

Here, main memory is divided into two types of partitions

1. **Low Memory** - Operating system resides in this type of memory.
2. **High Memory**- User processes are held in high memory.

Partition Allocation

Memory is divided into different blocks or partitions. Each process is allocated according to the requirement. Partition allocation is an ideal method to avoid internal fragmentation.

Below are the various partition allocation schemes :

- **First Fit**: In this type fit, the partition is allocated, which is the first sufficient block from the beginning of the main memory.
- **Best Fit:** It allocates the process to the partition that is the first smallest partition among the free partitions.
- **Worst Fit:** It allocates the process to the partition, which is the largest sufficient freely available partition in the main memory.
- **Next Fit:** It is mostly similar to the first Fit, but this Fit, searches for the first sufficient partition from the last allocation point.

What is Paging?

Paging is a storage mechanism that allows OS to retrieve processes from the secondary storage into the main memory in the form of pages. In the Paging method, the main memory is divided into small fixed-size blocks of physical memory, which is called frames. The size of a frame should be kept the same as that of a page to have maximum utilization of the main memory and to avoid external fragmentation. Paging is used for faster access to data, and it is a logical concept.

What is Fragmentation?

Processes are stored and removed from memory, which creates free memory space, which are too small to use by other processes.

After sometimes, that processes not able to allocate to memory blocks because its small size and memory blocks always remain unused is called fragmentation. This type of problem happens during a dynamic memory allocation system when free blocks are quite small, so it is not able to fulfill any request.

Two types of Fragmentation methods are:

1. External fragmentation
2. Internal fragmentation

- External fragmentation can be reduced by rearranging memory contents to place all free memory together in a single block.
- The internal fragmentation can be reduced by assigning the smallest partition, which is still good enough to carry the entire process.

What is Segmentation?

Segmentation method works almost similarly to paging. The only difference between the two is that segments are of variable-length, whereas, in the paging method, pages are always of fixed size.

A program segment includes the program's main function, data structures, utility functions, etc. The OS maintains a segment map table for all the processes. It also includes a list of free memory blocks along with its size, segment numbers, and its memory locations in the main memory or virtual memory.

What is Dynamic Loading?

Dynamic loading is a routine of a program which is not loaded until the program calls it. All routines should be contained on disk in a relocatable load format. The main program will be loaded into memory and will be executed. Dynamic loading also provides better memory space utilization.

What is Dynamic Linking?

Linking is a method that helps OS to collect and merge various modules of code and data into a single executable file. The file can be loaded into memory and executed. OS can link system-level libraries into a program that combines the libraries at load time. In Dynamic linking method, libraries are linked at execution time, so program code size can remain small.

Difference Between Static and Dynamic Loading

Static Loading	Dynamic Loading
Static loading is used when you want to load your program statically. Then at the time of compilation, the entire program will be linked and compiled without need of any external module or program dependency.	In a Dynamically loaded program, references will be provided and the loading will be done at the time of execution.
At loading time, the entire program is loaded into memory and starts its execution.	Routines of the library are loaded into memory only when they are required in the program.

Difference Between Static and Dynamic Linking

Here, are main difference between Static vs. Dynamic Linking:

Static Linking	Dynamic Linking
Static linking is used to combine all other modules, which are required by a program into a single executable code. This helps OS prevent any runtime dependency.	When dynamic linking is used, it does not need to link the actual module or library with the program. Instead of it use a reference to the dynamic module provided at the time of compilation and linking.

Summary:

- Memory management is the process of controlling and coordinating computer memory, assigning portions called blocks to various running programs to optimize the overall performance of the system.
- It allows you to check how much memory needs to be allocated to processes that decide which processor should get memory at what time.
- In Single Contiguous Allocation, all types of computer's memory except a small portion which is reserved for the OS is available for one application
- Partitioned Allocation method divides primary memory into various memory partitions, which is mostly contiguous areas of memory
- Paged Memory Management method divides the computer's main memory into fixed-size units known as page frames
- Segmented memory is the only memory management method that does not provide the user's program with a linear and contiguous address space.
- Swapping is a method in which the process should be swapped temporarily from the main memory to the backing store. It will be later brought back into the memory for continue execution.
- Memory allocation is a process by which computer programs are assigned memory or space.
- Paging is a storage mechanism that allows OS to retrieve processes from the secondary storage into the main memory in the form of pages.
- Fragmentation refers to the condition of a disk in which files are divided into pieces scattered around the disk.
- Segmentation method works almost similarly to paging. The only difference between the two is that segments are of variable-length, whereas, in the paging method, pages are always of fixed size.

- Dynamic loading is a routine of a program which is not loaded until the program calls it.
- Linking is a method that helps OS to collect and merge various modules of code and data into a single executable file.