B.C.A. (Sem – IV)

B.C.A. - 404

Operating System

Purushottam Singh

Purushottam Singh

You Can Download Materials From This Link ⇒ **mrpurusingh.blogspot.in**

Unit- 4

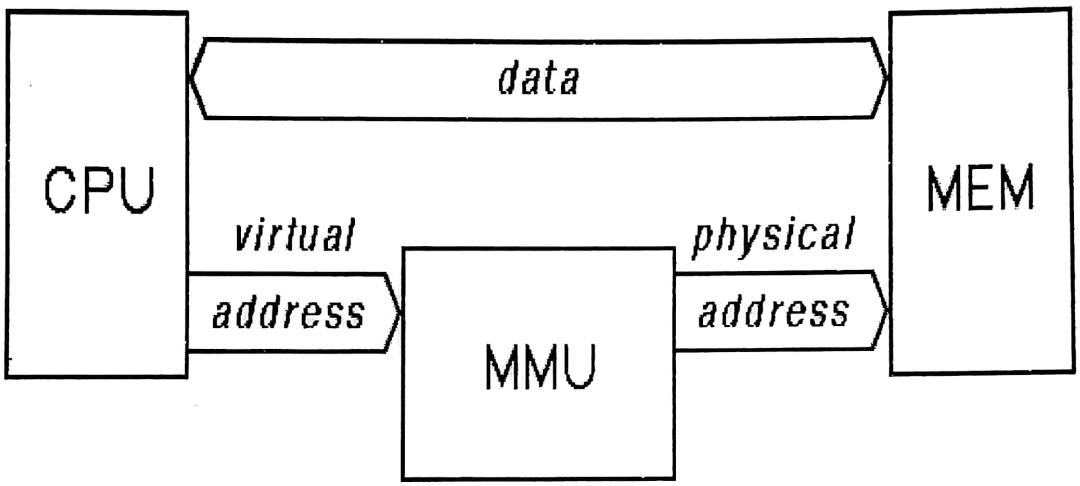
Purushottam Singh

You Can Download Materials From This Link ⇒ **mrpurusingh.blogspot.in**

Unit-4

Memory management unit: -

- A memory management unit (MMU), sometimes called paged memory management unit (PMMU), is a computer hardware component responsible for handling accesses to memory requested by the CPU.
- Its functions include translation of virtual addresses to physical addresses (i.e., virtual memory management), memory protection, cache control, bus arbitration and in simpler computer architectures.
- The MMU normally translates virtual page numbers to physical page numbers via an associative cache called a translation look-aside buffer (TLB).
- Hardware device that maps logical/virtual to physical address.
- In MMU the value in the relocation register is added to every address generated by a program at the time the address is sent to memory.
- The program deals with logical addresses; it never sees the real physical addresses.

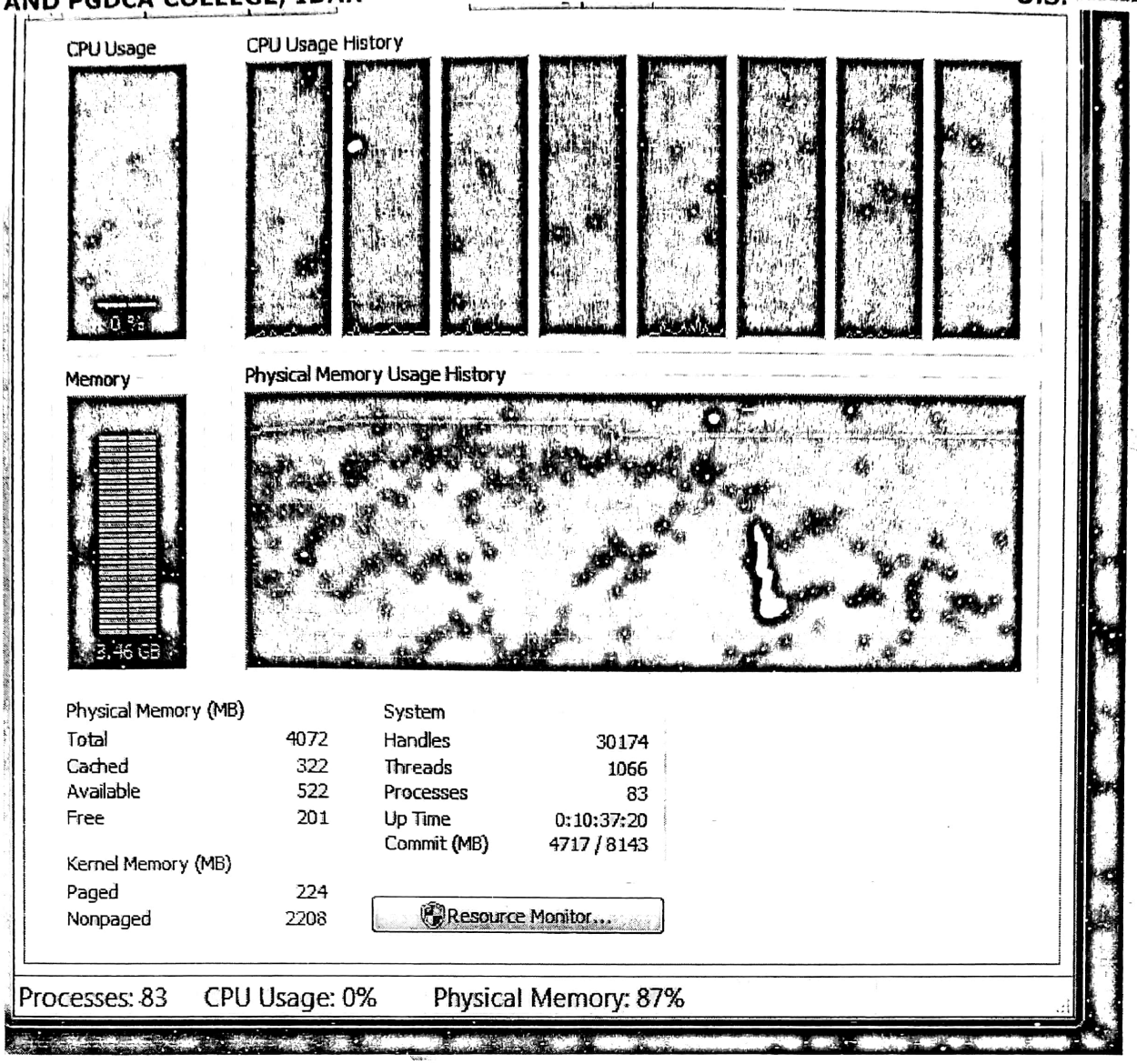


Virtual memory: -

- Virtual memory is a common part of most operating systems on desktop computers.
- Most computers today have something like 256 or 512 megabytes of RAM available for the CPU to use.
- Unfortunately, that amount of RAM is not enough to run all of the programs that most users expect to run at once.
- For example, if you load the operating system, an e-mail program, a Web browser and word processor into RAM simultaneously, 256 megabytes is not enough to hold it all.
- If there were no such thing as virtual memory, then once you filled up the available RAM your computer would have to say, "Sorry, you can not load any more applications. Please close another application to load a new one."
- With virtual memory, computer is look at RAM for areas that have not been used recently, that data are copied onto the hard disk. This frees up space in RAM to load the new application.
- The area of the hard disk that stores the RAM image is called a page file.

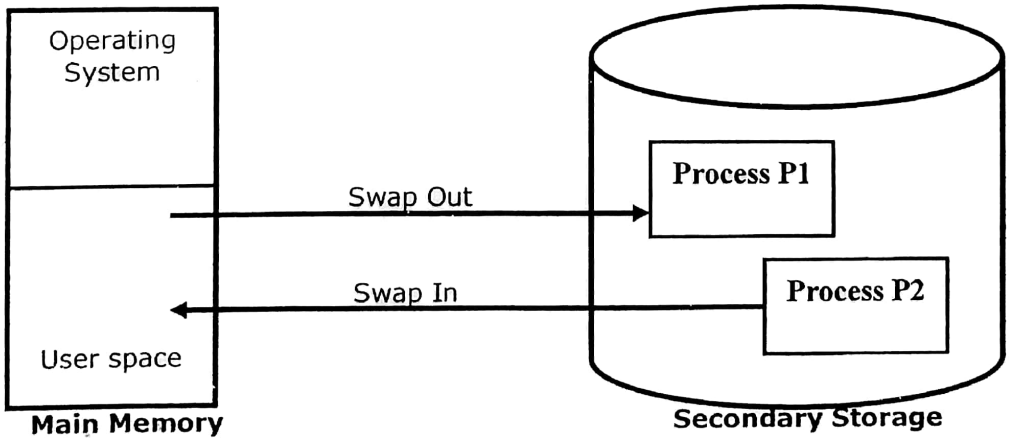
Physical memory: -

- The physical memory of a computer usually consists of Random Access Memory (RAM) chips and hard drives.
- RAM is the amount of real storage, and is the total amount of memory installed on a computer.
- Physical memory also referred to as the physical storage or the real storage.
- Physical memory is a term used to describe the total amount of memory installed in the computer.
- For example, if the computer has two 64MB memory modules installed, it has a total of 128MB of physical memory.



Swapping: -

- A process can be swapped out temporarily from memory and stored in to secondary storage, and then back into memory for continued execution.
- This swapping policy is used for priority based scheduling.
- If a higher priority process arrives in memory and wants service, the memory manger can swap out the lower priority process in secondary storage.
- The higher priority process can load into memory and execute that process.
- When the higher priority process finished, the lower priority process can be swapped back from secondary storage into memory and continued the execution.
- This process is also known as roll-out, roll-in process.



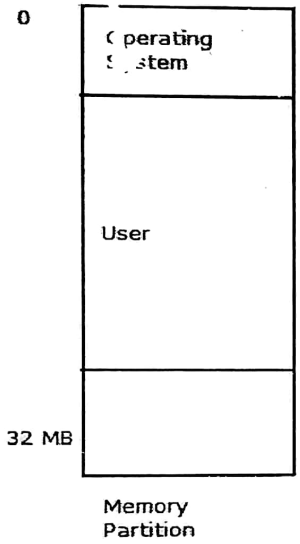
Memory Management Schemes: -

Memory allocation: -

- Memory is a central part of the modern computer systems.
- Memory is large collection storage cells. The size of each storage cell is 1 byte with its own memory address.
- The CPU fetches the instructions from memory according to the value of program counter.
- After the instruction has been executed, result may be stored back in memory.
- There are two Commands of memory allocation. [1] Contiguous memory allocation. [2] Non contiguous memory allocation.

Contiguous memory allocation: -

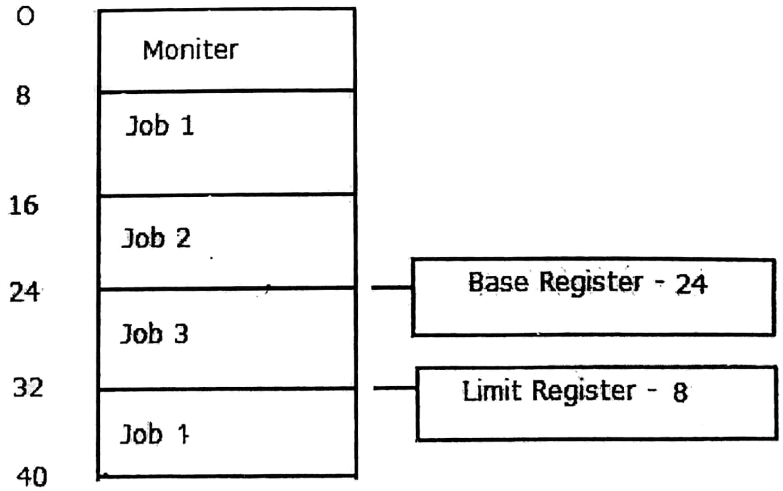
- The main memory must allocate to both, the operating system and the various user process.
- The memory is usually divided into two parts. One for the operating system and another for user process.
- It is possible to place the operating system in either low memory or higher memory.
- It is more common to place the operating system in low memory. The user process can take place in high memory.



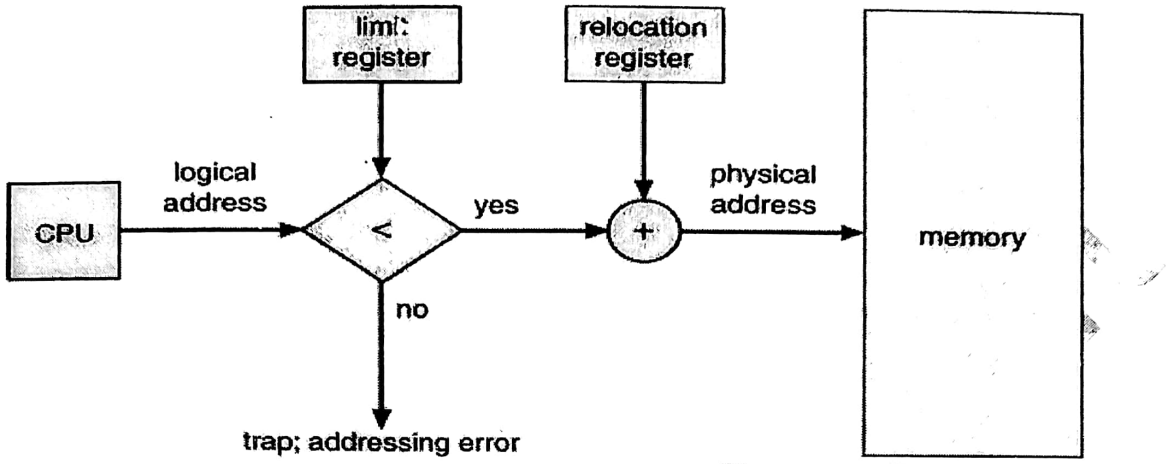
Single partition allocation:

- If the operating system is loaded into low memory and the user processes are executing in high memory, we need to protect the operating system code and data from changes by the user process.
- We also need to protect the user processes from one another.
- We provide this protection by using a relocation register.
- Consider the following figure.
- The base register is also known as relocation register.
- The relocation register contain the value of physical address.

- Memory Address 0 - 40 MB
- Each Job requires 8 MB



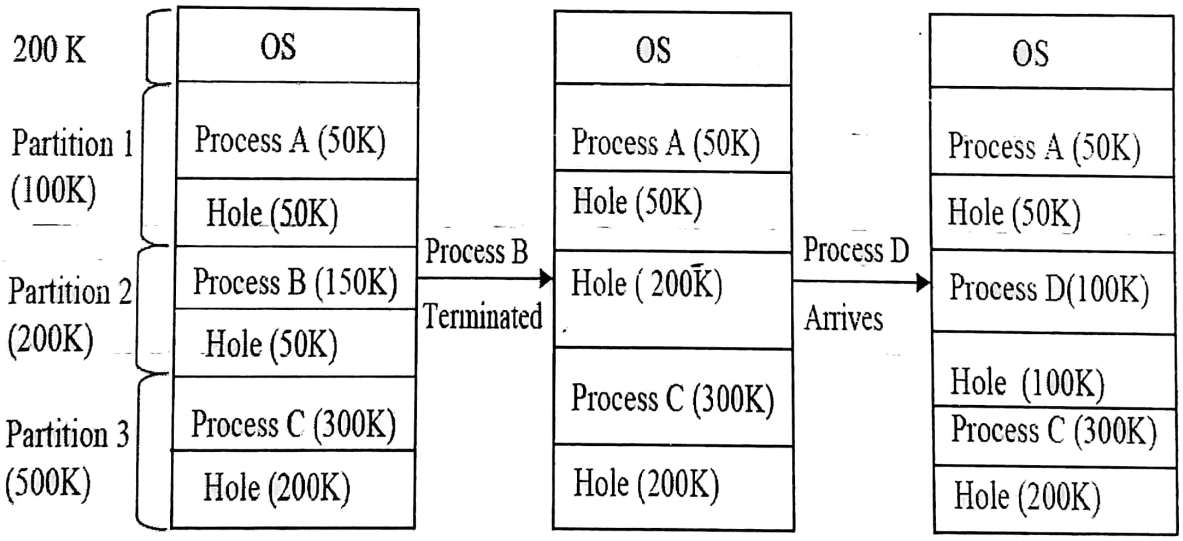
- The limit register contains the range of logical address.
- By using relocation and limit register the memory management unit provides the memory to the particular job.
- Relocation-register scheme used to protect user processes from each other, and from changing operating-system code and data.



Multiple partition allocation:

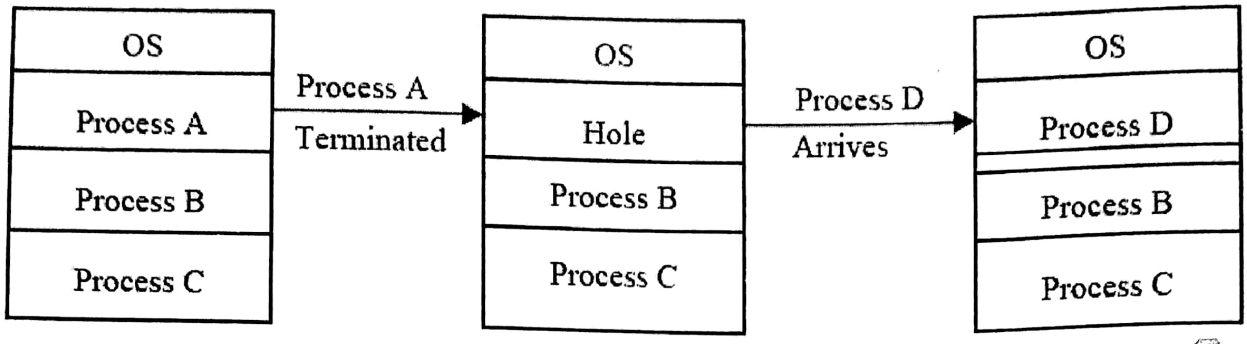
Fixed partition:

- There are several processes are stored into memory at the same time.
- One of the simplest schemes for the memory allocation is to divide the memory into a number of fixed sized partitions.
- Each partition contains only one process at a time.
- When the partition is free, a process is selected from the input queue and allocated to that free partition.
- When the process is completed, that partition becomes available for another process.
- The operating system keeps a table indicating which part of memory are available and which part of memory is allocated.



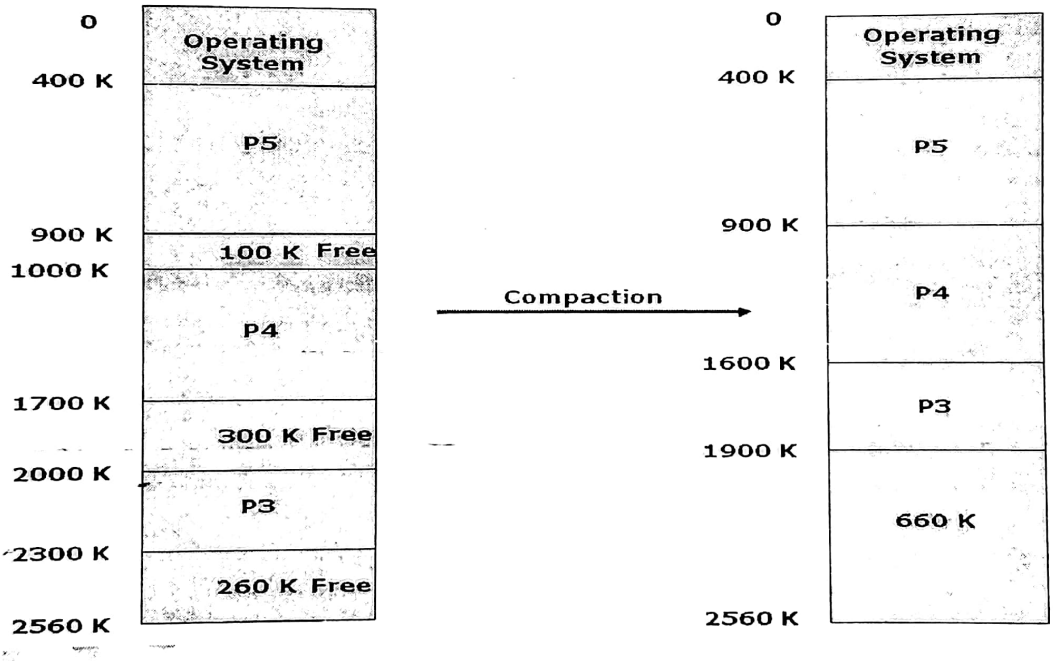
Variable partition:

- Initially, all memory is available for the user process, considered as one large block of available memory.
- That large block is known as a **hole.**
- When a process arrives and needs memory, we search for a **hole** large enough for this process and allocate only as much required.
- **First-fit:** Allocate the first hole that is big enough.
- **Best-fit:** Allocate the smallest hole that is big enough.
- **Worst-fit:** Allocate the largest hole.



External and Internal Fragmentation:

- Multiple partition allocation schemes suffer from external fragmentation.
- As processes are loaded and removed from the memory, the free memory space is broken into little pieces.
- External fragmentation exists when enough total memory space exists to satisfy a request, but it is not contiguous; storage is fragmented into a number of holes.
- Internal fragmentation is memory that is internal to a partition, but is not being used.
- One solution to the problem of external fragmentation is **compaction**.
- The compaction is used to reshuffle the all free memory places together and create large block.
- **External Fragmentation** – Total memory space exists to satisfy a request, but it is not contiguous.
- **Internal Fragmentation** – Allocated memory may be slightly larger than requested memory; this size difference is memory internal to a partition, but not being used.



- Reduce external fragmentation by compaction
  1. Shuffle memory contents to place all free memory together in one large block.
  2. Compaction is possible only if address binding is dynamic, and is done at execution time.

Memory management policy: -

- There are three types of memory management policies available in operating system. [1] Fetch policy [2] Placement policy [3] Replacement policy.

- **Fetch Policy: -**
  1. Demand Paging: - Pages are fetched when needed. The process starts with a flurry of page faults, eventually locality takes over.
  2. Pre- paging: - Pre-paging makes the use of disk storage characteristics. If pages are stored contiguously it may be more efficient to fetch them. The fetch police is ineffective if the extra pages are not referenced.

- **Placement policy: -**
  1. The placement policy is concerned with determining where in real memory a process piece is to reside. With anything other than pure segmentation this is not an issue. (Refer – best-fit, first-fit etc...)

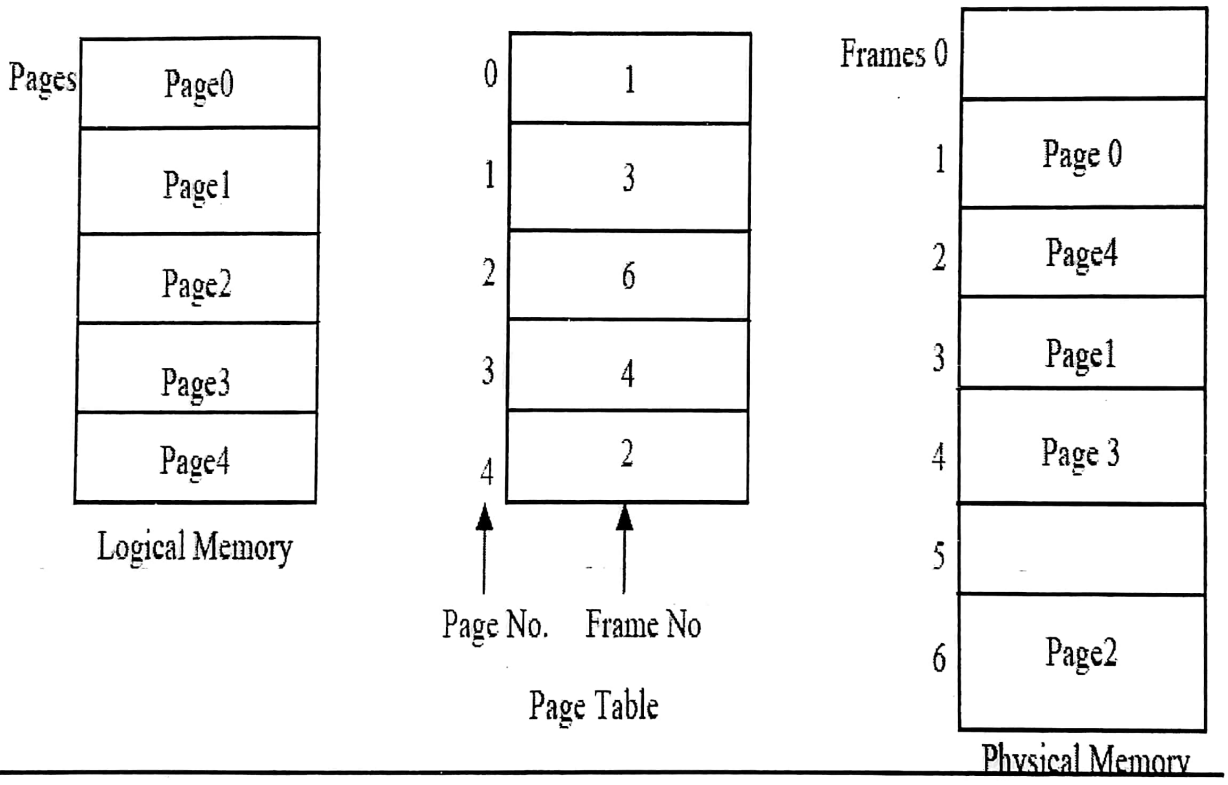
- **Replacement policy: -**
  1. All page frames are used. A page fault has occurred. New page must go into a frame.

Non-contiguous memory management: -

- Another memory allocation method is non-contagious allocation.
- Memory is allocated in non-contagious by two ways.
- One is paging and another is segmentation.

Paging:

- Physical memory is broken into fixed sized blocks called frames.
- Logical memory is broken into blocks of the same size called pages.
- Keep track of all free frames.
- To run a program of size n pages, need to find n free frames and load program.
- The reason behind this is implementation of paging mechanis n using page number and page offset.
- Set up a page table to translate logical to physical addresses.
- Remove/reduce external fragmentation. Internal fragmentation exists.



- Page table: used to translate logical to physical addresses.
- **Address translation scheme: -**
  - Address generated by CPU is divided into:
    - Page number (p)
      - Used as an index into the page table.
      - Page table contains base address of each page in physical memory.
    - page offset (d)
      - Combined with base address to define the physical memory address sent to the memory unit.
  - Use page number to index the page table.
  - Get the page frame start address.
  - Add offset with that to get the actual physical memory address.
  - Access the memory.

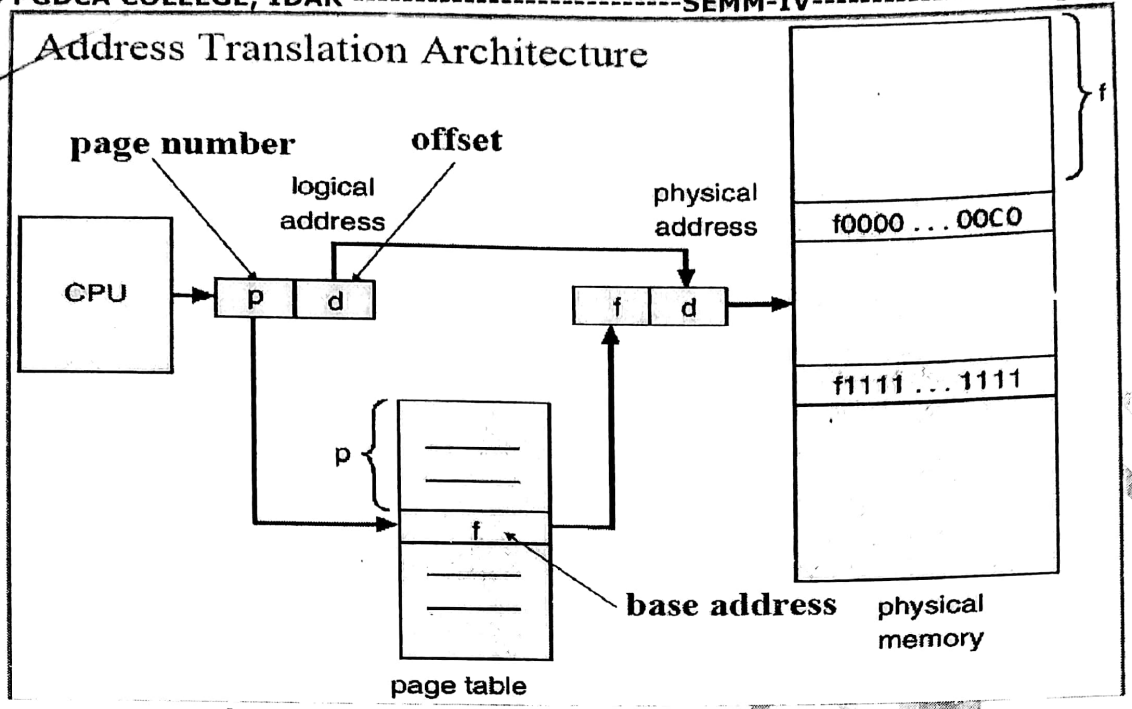


Figure: - Address Translation Architecture

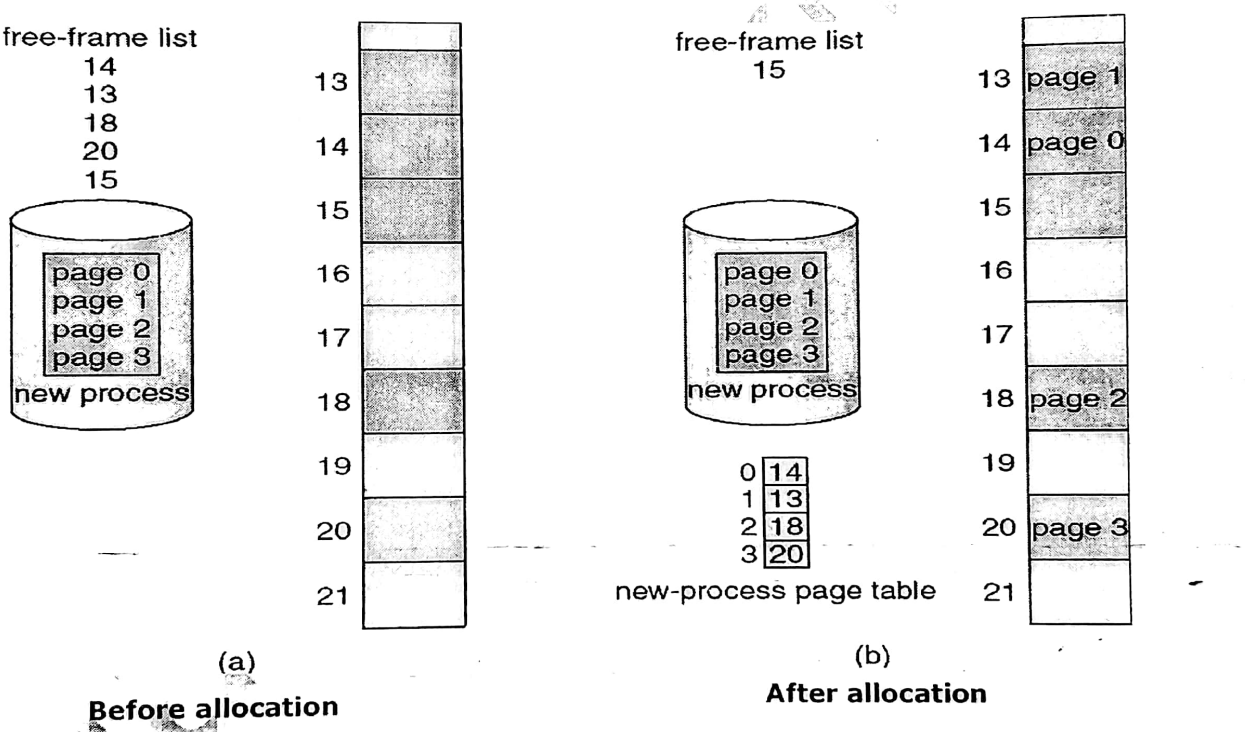
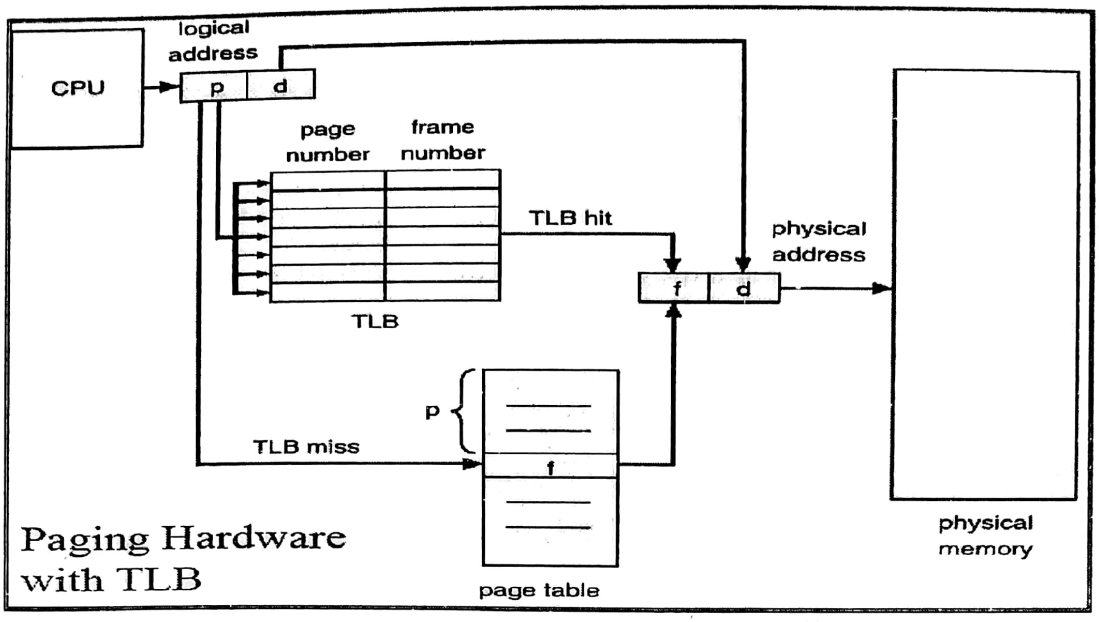


Figure: - Example of Paging

TLB (Translation look-aside buffers): -

- In the page table base register there are some time the problem created with the approach of the time required to access a user memory location.
- If want to access location **i**, we must first index into the page table, using the value in the PTBR offset by the page number for **i**.
- This task requires a memory access. It provides us with the page offset to produce the actual address.
- We can then access the desired place in memory. With this scheme, two memory accesses are needs to access a byte.
- Thus memory access is slowed by factor of 2. This delay would be intolerable under most circumstances.
    - The standard solution called associative registers or translation look aside buffer (TLB).
    - A set of associative registers is built of especially high-speed memory.
    - Each register consist of two parts: a key and a value.

- When the associative registers are presented with an item, it is compared with all keys simultaneously.
- If the item is found, the corresponding value field is output. The search is fast and the hardware is expansive.

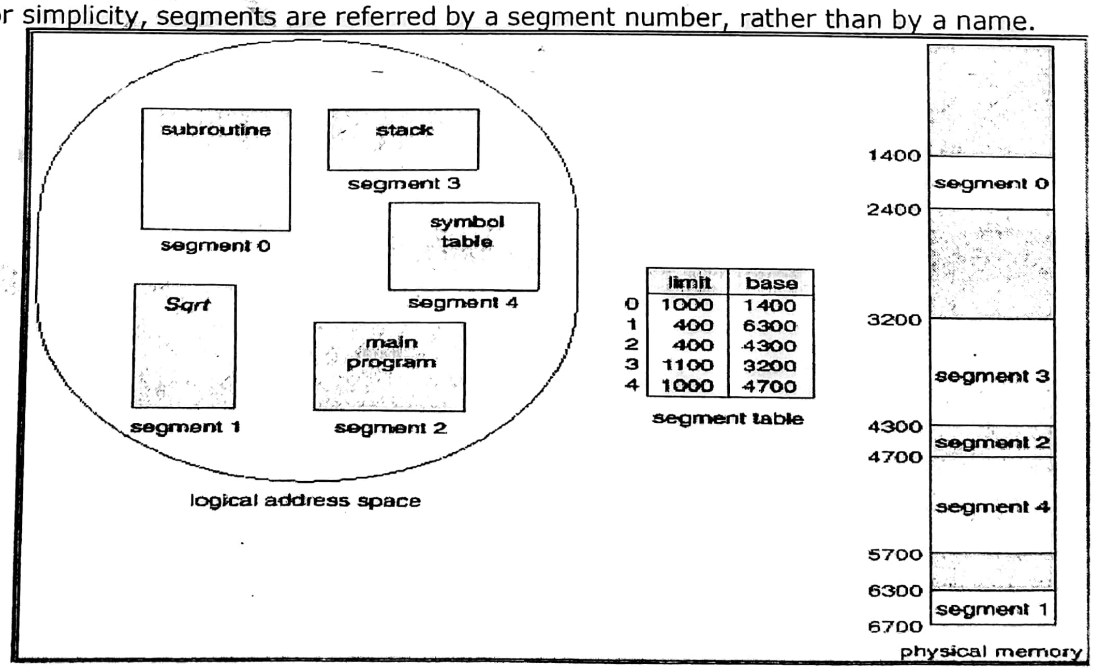


PTBR (Page table base register): -

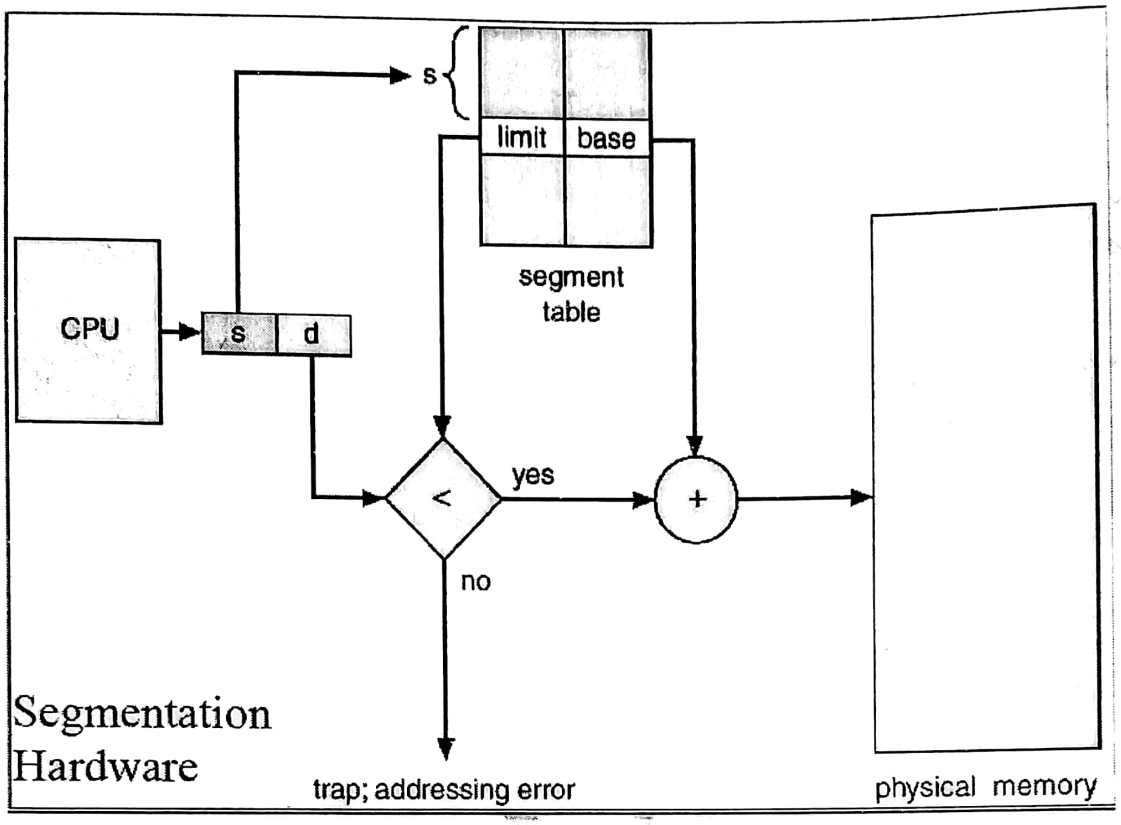
- In the Paging we use the register for the page table which stores the addresses of page.
- In the paging the page table is small is satisfactory if the page table reasonably small.
- Most contemporary computers, the page table is very large.
- For these machines, the use of fast registers to implement the page table is not feasible.
- So the page table kept in main memory and page table base register points to the page table.
- Changing page tables requires changing only this one register, substantially reducing context-switch time.

Segmentation: -

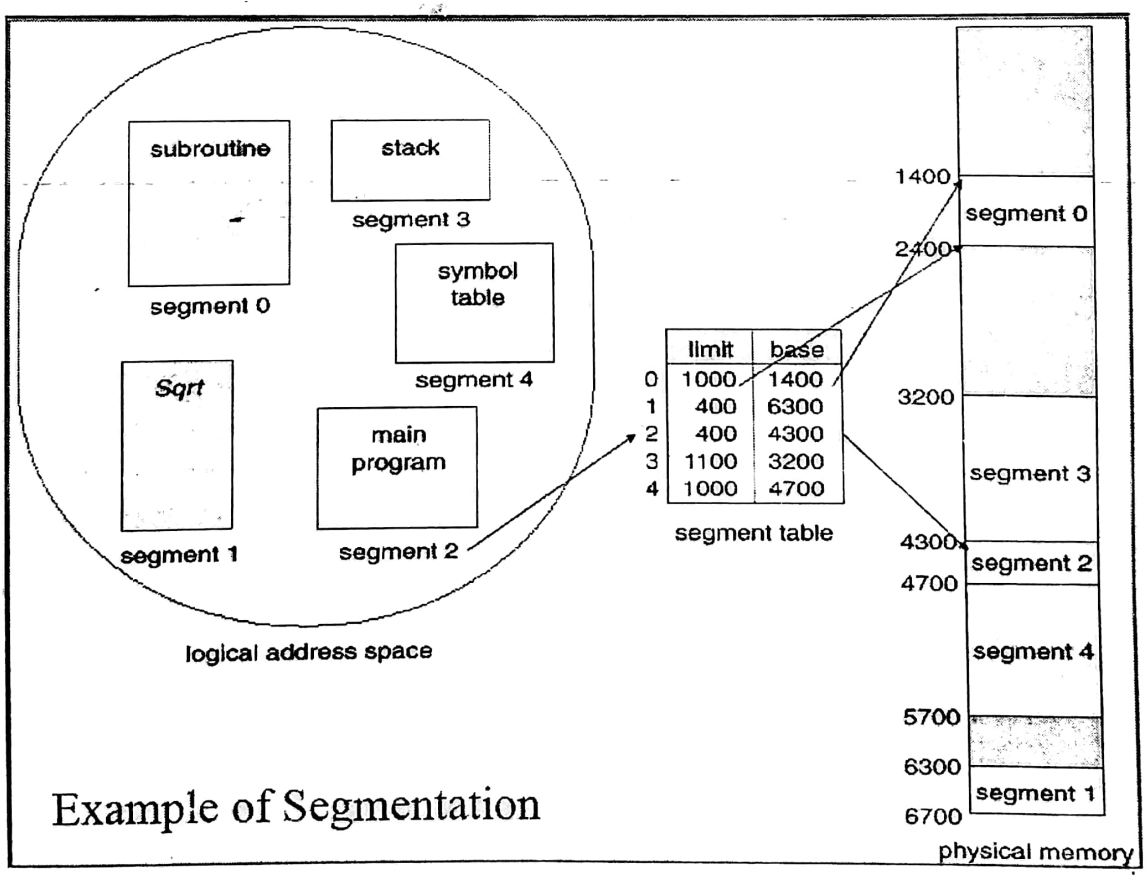
- Segmentation presents an alternative scheme for memory management.
- This scheme divides the logical address space into variable length chunks, called segments, with no proper ordering among them.
- Memory-management scheme that supports user's view of memory.
- A program is a collection of segments. A segment is a logical unit such as: main program, functions, methods, object, local and global variable, arrays etc...
- Each segment has a name and a length with different size.
- For simplicity, segments are referred by a segment number, rather than by a name.



- Thus, the logical addresses are expressed as a pair of segment number and offset within segment.
- It allows a program to be broken down into logical parts according to the user view of the memory, which is then mapped into physical memory.
- This mapping between two is done by segment table, which contains segment base register and its limit register. The segment base register has starting physical address of segment, and segment limit register provides the length of segment.

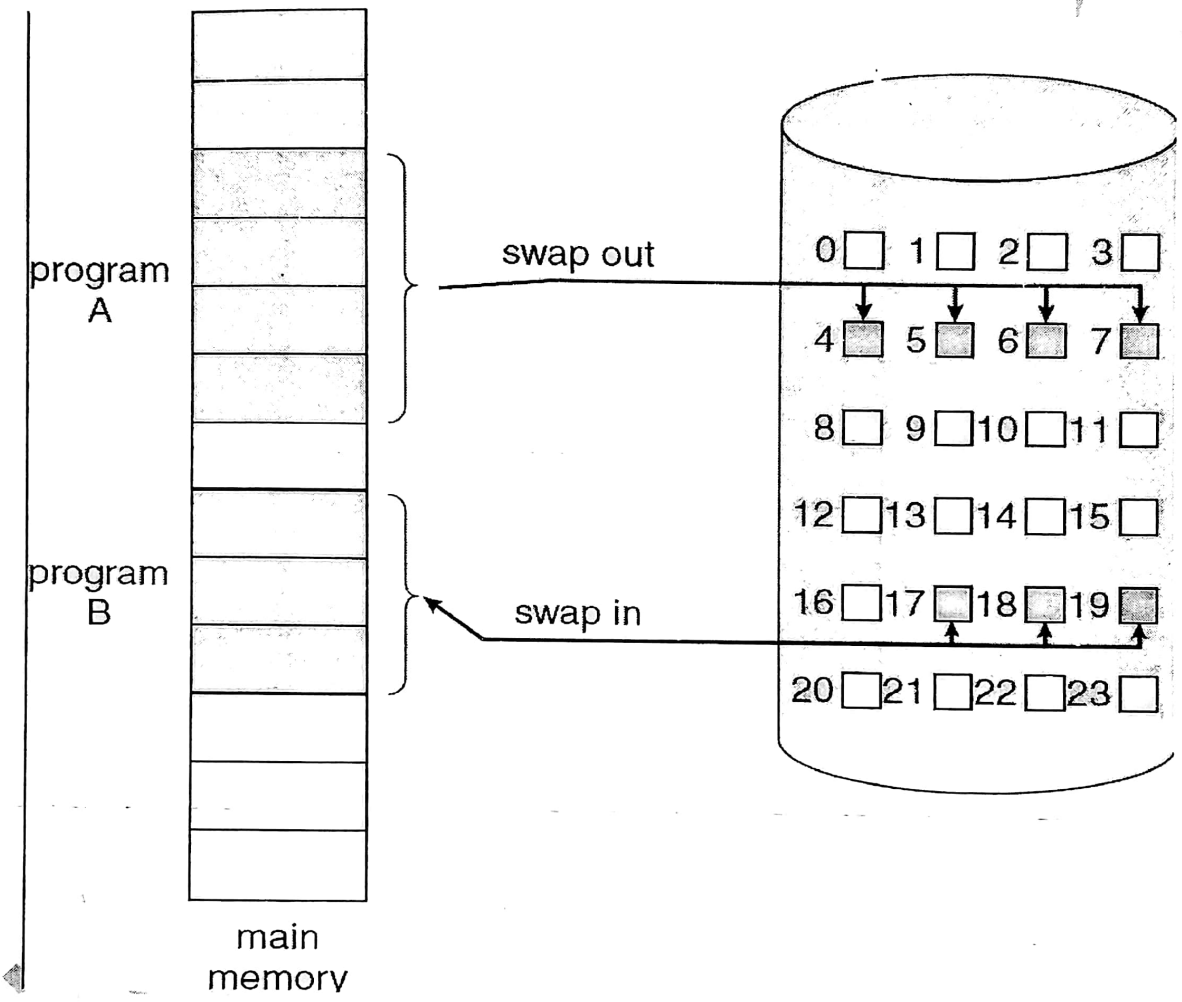


- Consider the following example of segmentation.



Demand paging: -

- In operating systems, demand paging is a method of virtual memory management.
- Consider how an executable program might be loaded from disk into memory.
- One option is to load the entire program in physical memory at program execution time.
- However, a problem with this approach is that we may not initially need the entire program in memory.
- An alternative strategy is to initially load pages only as they are needed.
- This technique is known as demand paging and is commonly used in virtual memory systems.
- A demand-paging system is similar to a paging system with swapping where processes reside in secondary memory (usually a disk).
- When we want to execute a process, we swap it into memory. Rather than swapping the entire process into memory, however, we use a lazy swapper.
- A lazy swapper never swaps a page into memory unless that page will be needed.
- A swapper manipulates entire processes, whereas a pager is concerned with the individual pages of a process. We thus use pager, rather than swapper, in connection with demand paging.

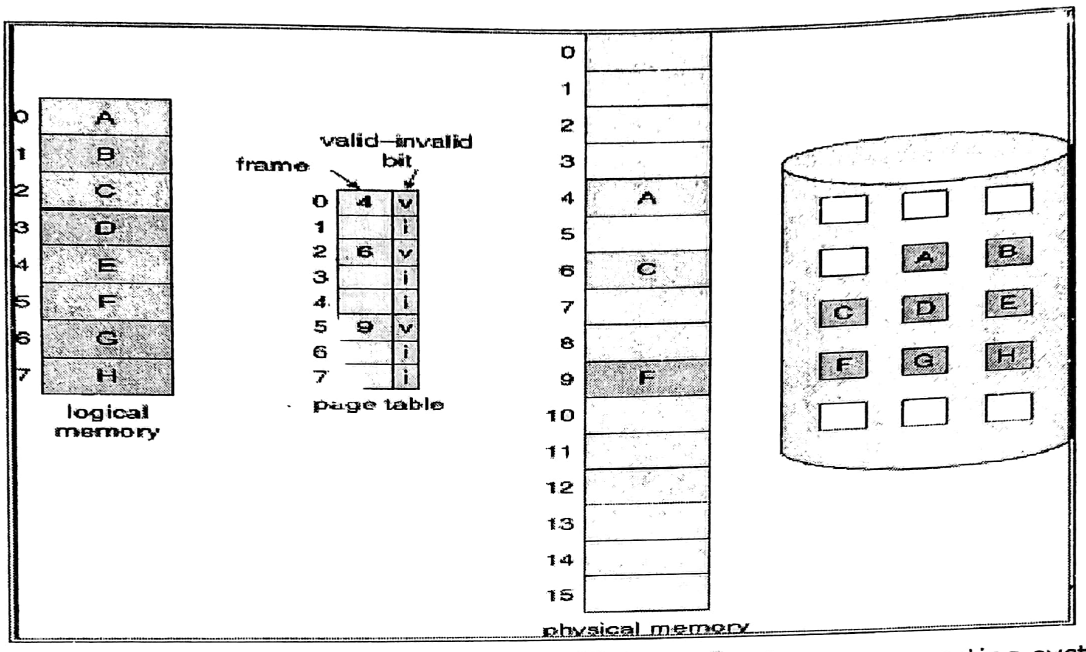


Page fault: -

- A page is a fixed-length block of memory that is used as a unit of transfer between physical memory and external storage like a disk, and a page fault is an interrupt (or exception) to the software raised by the hardware, when a program accesses a page that is mapped in address space, but not loaded in physical memory.
- If a Page-table mapping indicates an absence of the page in physical memory, hardware raises a "Page- Fault".
- OS traps this fault and the interrupt handler services the fault by initiating a disk-read request.
- Once page is brought in from disk to main memory, page-table entry is updated and the process which faulted is restarted.
- May involve replacing another page and invalidating the corresponding page-table entry.
- An invalid page fault or page fault error occurs when the operating system cannot find the data in virtual memory.

- This usually happens when the virtual memory area, or the table that maps virtual addresses to real addresses, becomes corrupt.

Page Table When Some Pages Are Not in Main Memory



- If there is a reference to a page, first reference to that page will trap to operating system:
  - Page fault.
- Operating system looks at another table to decide:
  - Invalid reference -- abort
  - Just not in memory
- Get empty frame
- Swap page into frame
- Reset tables
- Set validation bit = v
- Restart the instruction that caused the page fault

Steps in Handling a Page Fault