UNIT – IV

MEMORY MANAGEMENT

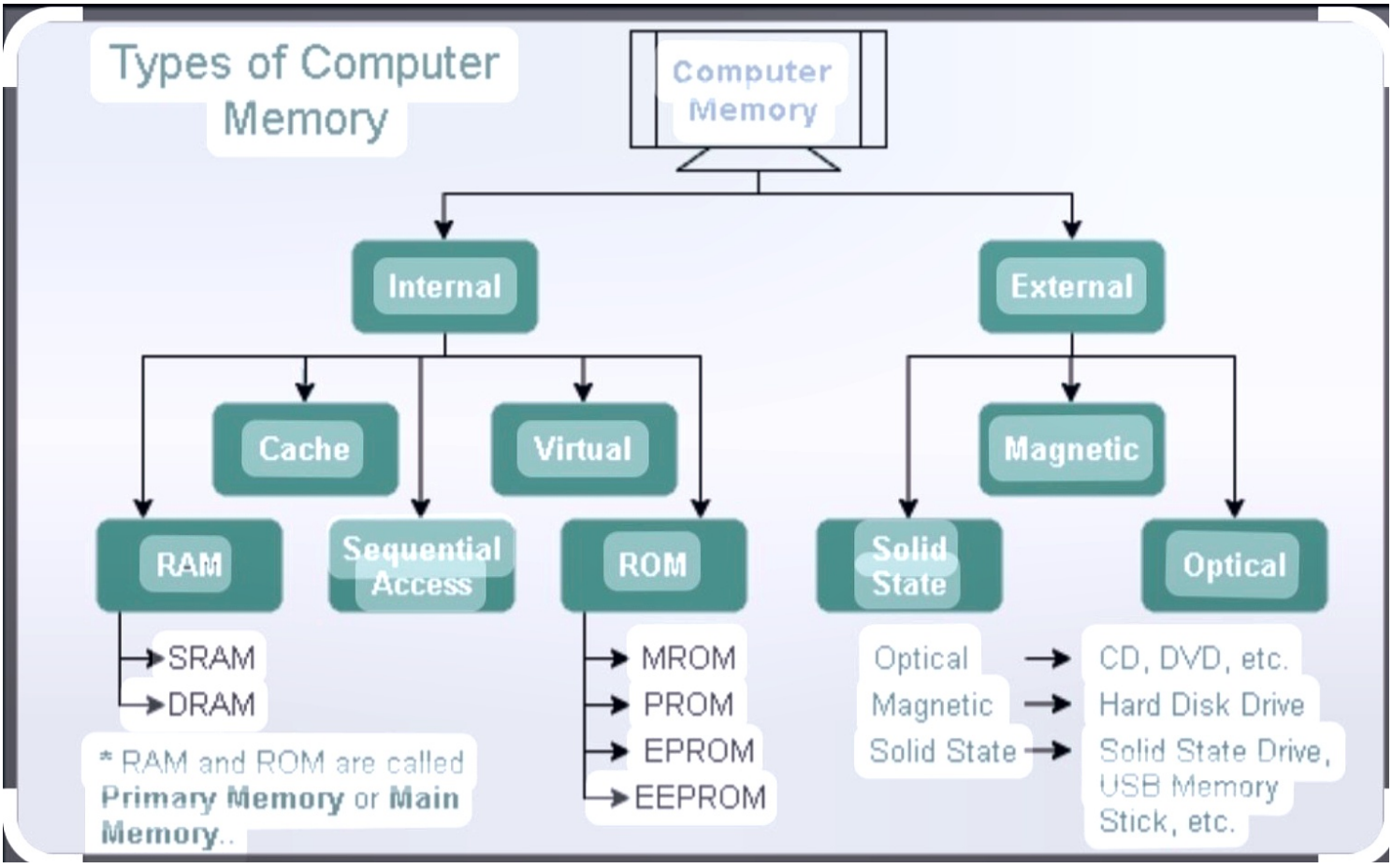
- Single Contiguous Allocation
- Introduction to Multiprogramming
- Partitioned Allocation
- Relocatable Partitioned Memory Management
- Paged Memory Management
- Demand- Paged Memory Management
- Segmented Memory management

Introduction

- In computing, **memory** refers to a device that is used to store information for immediate use in a computer or related computer hardware device.
- The memory management modules of an Operating System are concerned with the management of primary memory.
- Primary memory we mean the memory that the processors directly access for instructions and data.
- Primary memory is often called as core memory
- The term "memory" is often synonymous with the term "primary storage".

Types of Memory:

Memory is the most essential element of a computing system because without it computer can't perform simple tasks. Computer memory is of two basic types – Internal Memory (Primary memory) and External Memory (Secondary memory).


Types of Computer Memory
Computer Memory
Internal
External
Cache
Virtual
Magnetic
RAM
Sequential Access
ROM
Solid State
Optical
SRAM
DRAM
MROM
PROM
EPROM
EEPROM
Optical
Magnetic
Solid State
CD, DVD, etc.
Hard Disk Drive
Solid State Drive, USB Memory Stick, etc.
* RAM and ROM are called Primary Memory or Main Memory..

Memory management concerned with four functions:

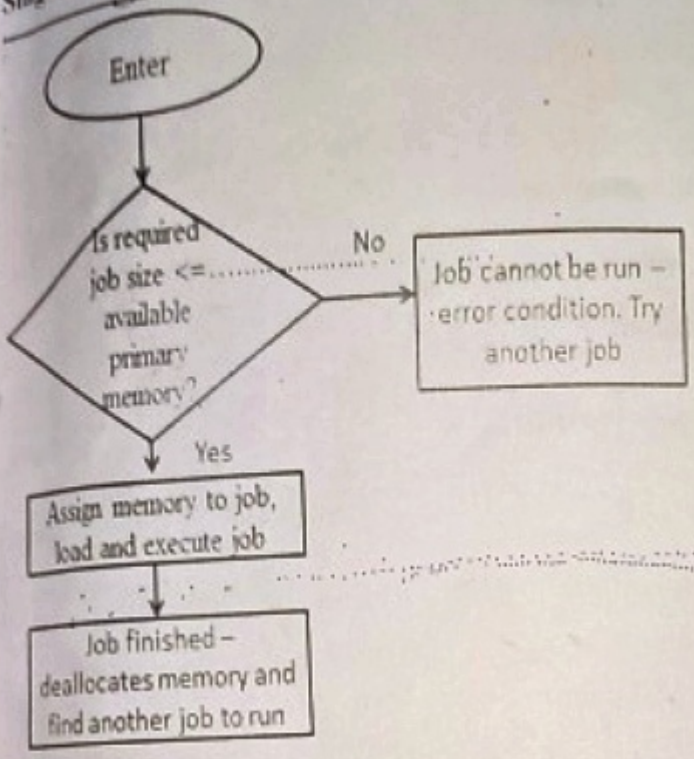
- Keeping track of the status of each location of primary memory.
- Determining allocation policy for memory. i.e. deciding to whom it should allocates, how much, when, and where.
- Allocation technique – once it is decided to allocate memory.
- Deallocation technique and policy

Single Contiguous allocation:

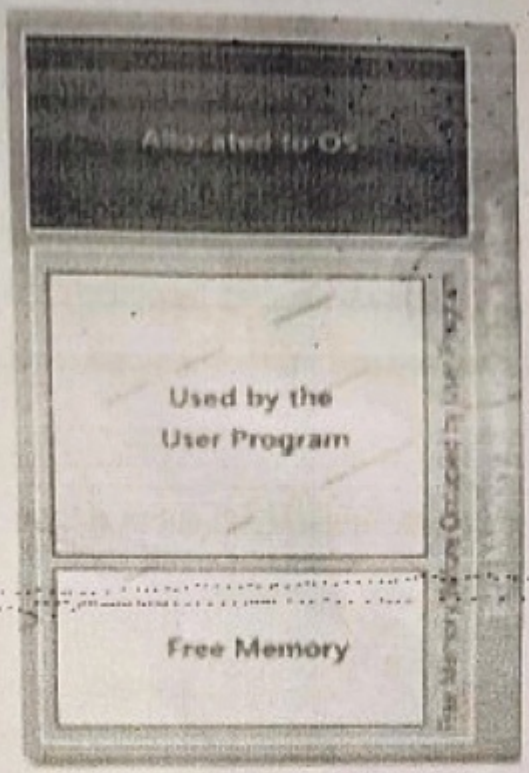
Memory conceptually divided into three contiguous regions.

1. A portion of memory permanently allocated to the operating system.
2. All of the remaining memory is available and allocated to the single job being processed
3. Allocated but unused memory (Wasted)

Single Contiguous allocation algorithm:



Single Contiguous allocation



- **Bit map** keeps track of free blocks in memory, it has one bit for one memory block, bit 0 shows that the block is free and bit 1 shows the block is allocated to some file or a process.
- Contiguous memory allocation leads to fragmentation (breakup). Further fragmentation can either be external or internal.
- Contiguous memory allocation leads to memory wastage and inflexibility.
- If the operating system uses buffered I/O during processing, then contiguous memory allocation can enhance processing speed.

Advantages of Single Contiguous allocation:

- Allocating memory is easy and cheap.
- OS can take first one out of list.
- Eliminates external fragmentation

- Data (page frames) can be scattered all over PM
- Pages are mapped appropriately anyway.
- Allows demand paging and pre-paging
- More efficient swapping.
- No need for considerations about fragmentation.
- Just swap out page least likely to be used.

Disadvantages of Single Contiguous allocation:

- Longer memory access times (page table lookup)
- Can be improved using TLB
- Guarded page tables.
- Inverted page tables.
- Memory requirements (one entry per VM page)
- Improve using Multilevel page tables and variable page sizes (super-pages)
- Guarded page tables
- Page Table Length Register (PTLR) to limit virtual memory size.
- Internal fragmentation.

Fragmentation:

- Fragmentation is an unwanted problem in the operating system in which the processes ar loaded and unloaded from memory, and free memory space is fragmented. Processes can't be assigned to memory blocks due to their small size, and the memory blocks stay unused.

Types of Fragmentation

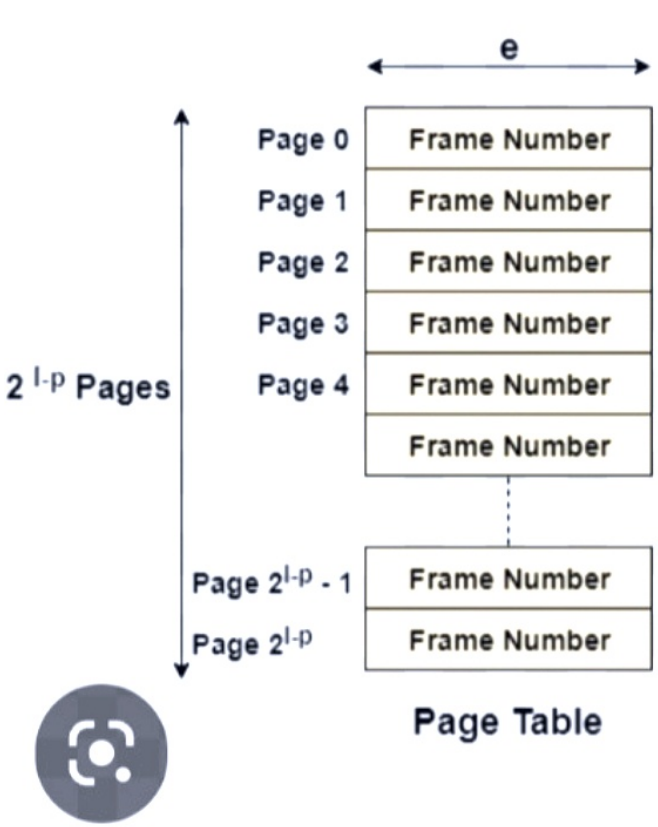
1. **Internal Fragmentation**
2. **External Fragmentation**

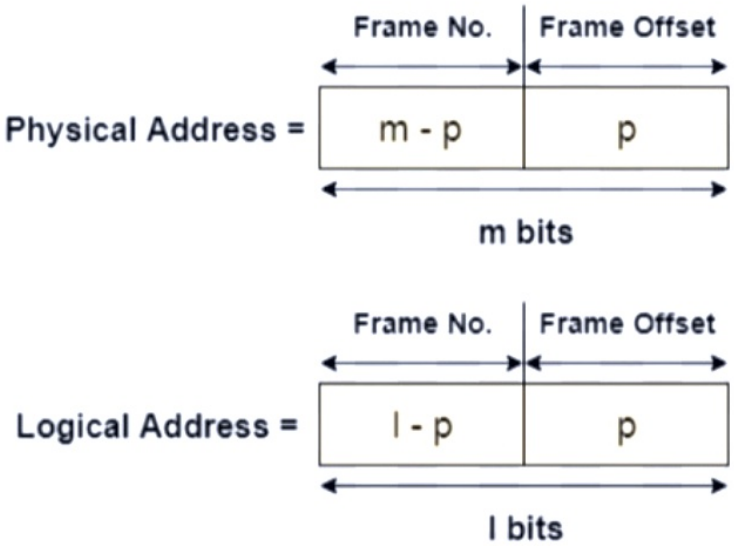
When a process is allocated to a memory block, and if the process is smaller than the amount of memory requested, a free space is created in the given memory block. Due to this, the free space of the memory block is unused, which causes **internal** fragmentation.

External fragmentation happens when a dynamic memory allocation method allocates some memory but leaves a small amount of memory unusable. The quantity of available memory is substantially reduced if there is too much external fragmentation. There is enough memory space to complete a request, but it is not contiguous. It's known as **external fragmentation**.

Paged Memory Management

- A computer can address more memory than the amount physically installed on the system. This extra memory is actually called **virtual memory.**
- Paging technique plays an important role in implementing virtual memory.
- Paging is a memory management technique in which **process address space is broken into blocks of the same size** called **pages.**
- The size of the process is measured in the **number of pages**.
- Similarly, main memory is divided into small fixed-sized blocks of (physical) memory called **frames**.
- The size of a frame is kept the same as that of a page to have optimum utilization of the main memory and to avoid external fragmentation.
- Page address is called **logical address** and represented by **page number** and the **offset**. Logical Address = Page number + page offset
- Frame address is called **physical address** and represented by a **frame number** and the **offset**. Physical Address = Frame number + page offset



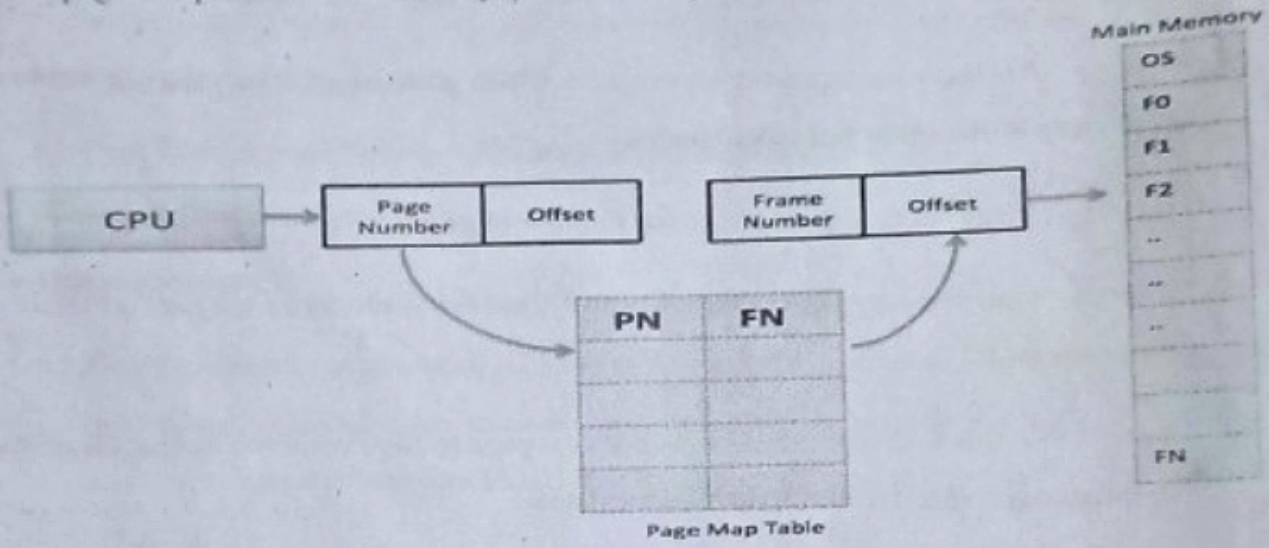


No. of entries in Page Table = No. of the pages in the process

Page Table Size = 2^{l-p} X e bytes

e = m-p (Frame Size) bits

- A data structure called **page map table** is used to keep track of the relation between a page of a process to a frame in physical memory.



- When the system allocates a frame to any page, it translates this logical address into a physical address and create entry into the page table to be used throughout execution of the program.
- When a process is to be executed, its corresponding pages are loaded into any available memory frames. When a computer runs out of RAM, the operating system (OS) will

move idle or unwanted pages of memory to secondary memory to free up RAM for other processes and brings them back when needed by the program.

Advantages :

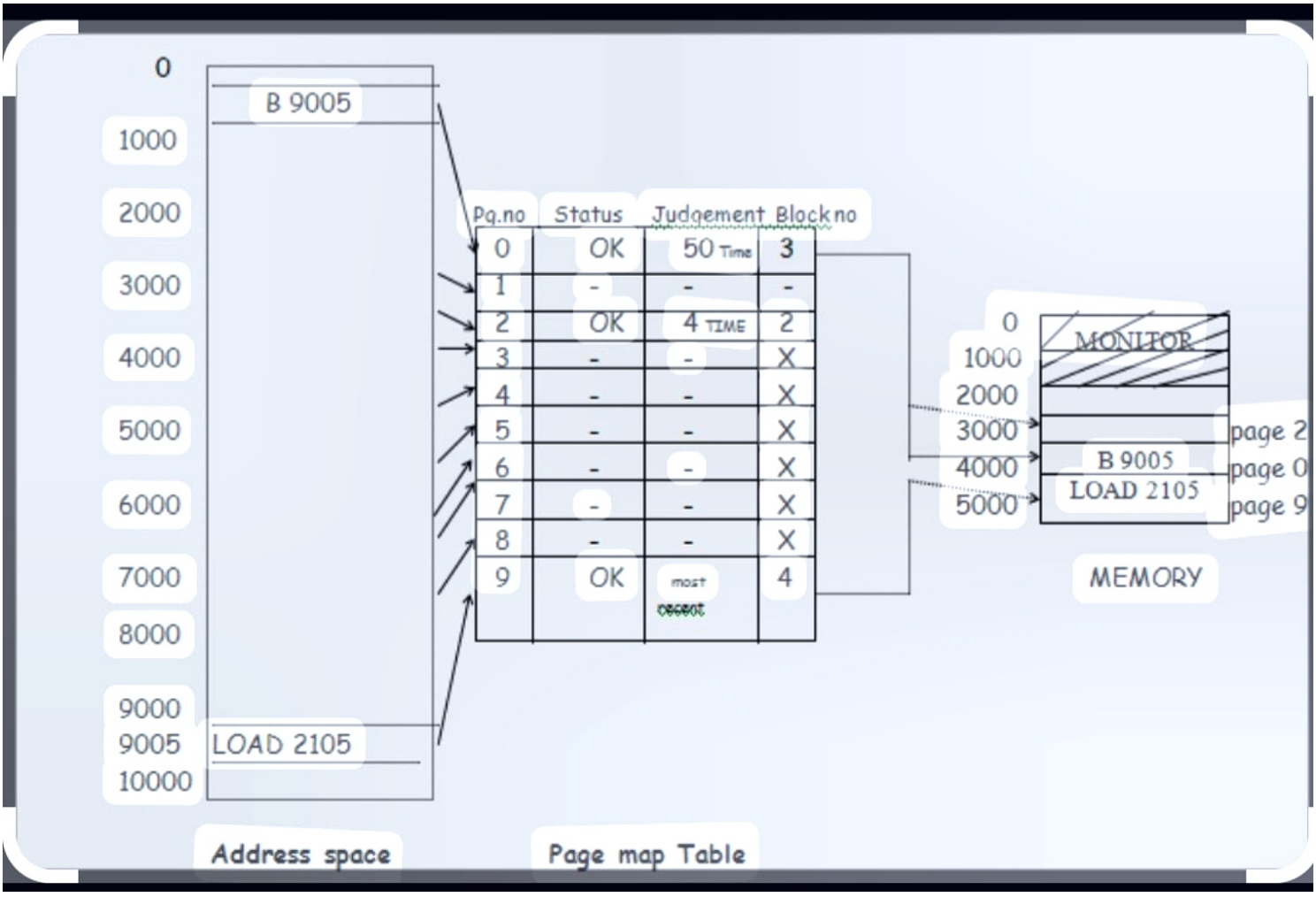
1. It solves the fragmentation problem without physically moving the pages in memory.
2. This allows a higher degree of multiprogramming.
3. The compaction in relocatable partitioned allocation is eliminate.

Dis-Advantages :

1. Page address mapping hardware increase the cost of the computer system.
2. Extra core or extra registers needed for page map tables.
3. There is a possibility of internal fragmentation (or) page breakage may occur.
4. Some memory will still unused if the number of available page frames are not sufficient for the Job's page.

Demand Paged Memory Management

- In the demand paged memory allocation one programmer assumes infinite memory called virtual memory. In which the Job operates under demand.
- Paging virtual memory is large in size then the available memory the demand paging allocation is shown in figure.
- In demand pages memory allocation a particular page is swapped from the secondary storage device if it is required for execution.


0
B 9005
1000
2000
3000
4000
5000
6000
7000
8000
9000
9005
LOAD 2105
10000
Address space
Pg.no
Status
Judgement
Block no
0 OK 50 Time 3
1 - - -
2 OK 4 TIME 2
3 - - X
4 - - X
5 - - X
6 - - X
7 - - X
8 - - X
9 OK most recent 4
Page map Table
0
1000
2000
3000
4000
5000
MONITOR
B 9005
LOAD 2105
page 2
page 0
page 9
MEMORY

- In the above fig. Job address space consists of 10,000B of programs called page and we have only 3000B of memory available. This 10,000B program run in 3000B of memory by using demand paging.

Advantages :

1. Fragmentation is eliminated.
2. Large Virtual memory is available.
3. More efficient use of memory.
4. Demand paging is valuable in time sharing, systems.

Dis-Advantages :

1. Page address mapping hard ware increases the cost of the computer system
2. Extra memory, extra register needed for page map table.

- The page map table consists of a page number, page frame number, status and judgement field. Initially the allocation algorithm places the first three pages of a program in the main memory. The job starts execution in one of these pages. Suppose in page O there is a transfer instruction to the location 9055. The hard ware mechanism uses 9 as index to the page lable that is page 9 is required in the execution. then the pagemap table is checked to find whether the page in the main memory job not if the required page is not in the main memory then the page fault occurs, then the hard ware generates interrept to

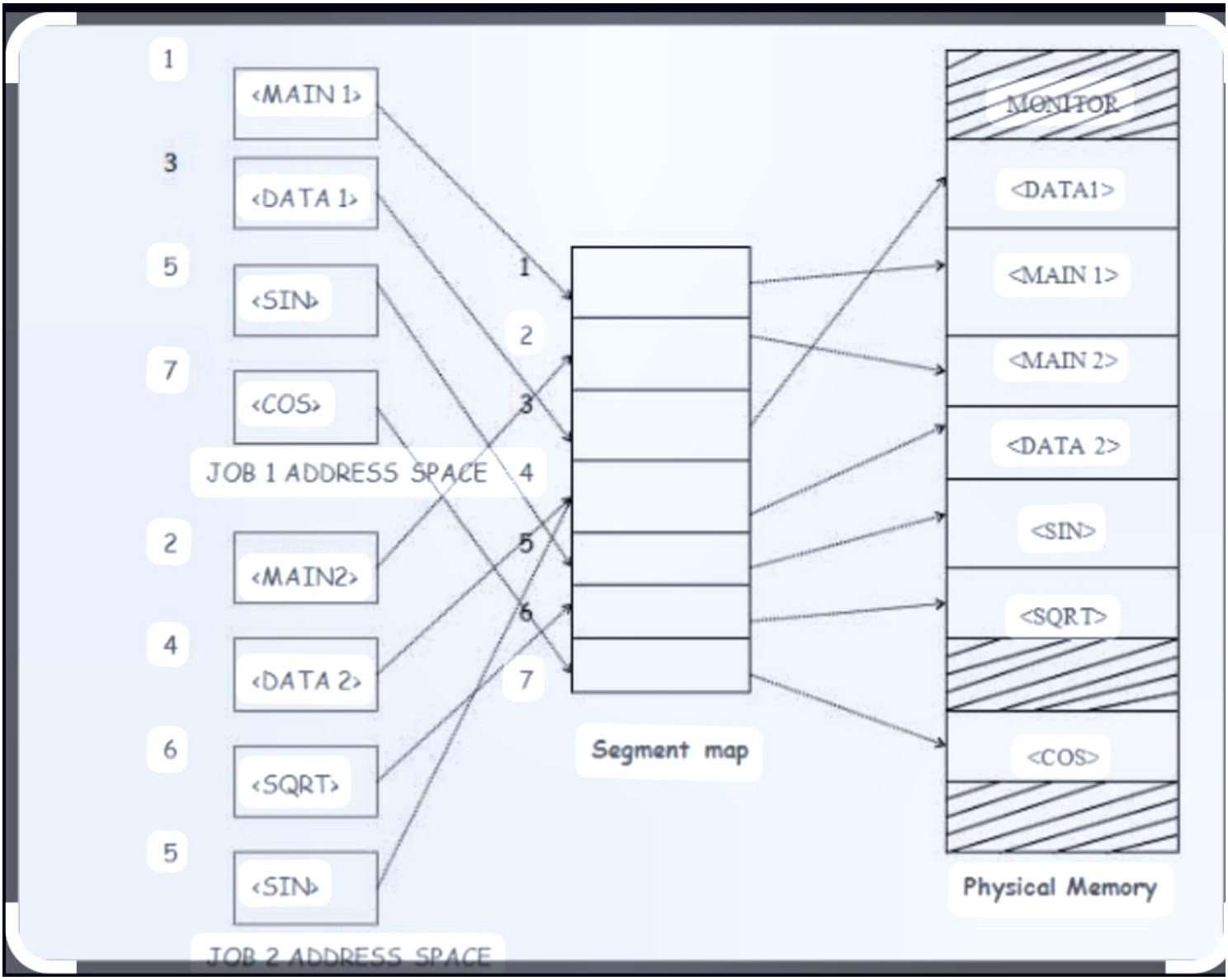
the operating system then the operating system searches the secondary storage device for page 9 and brought into the memory by replacing one of the pages in the memory It may use a page remove algorithms like FIFO(First in first out) LRU(Least Recently Use) for removing a page. The decision which page has to replace it with judgement field in the page map table. It is desirable to keep highly used pages in the memory.

- **A demand paging system is quite similar to a paging system with swapping where processes reside in secondary memory and pages are loaded only on demand, not in advance.**

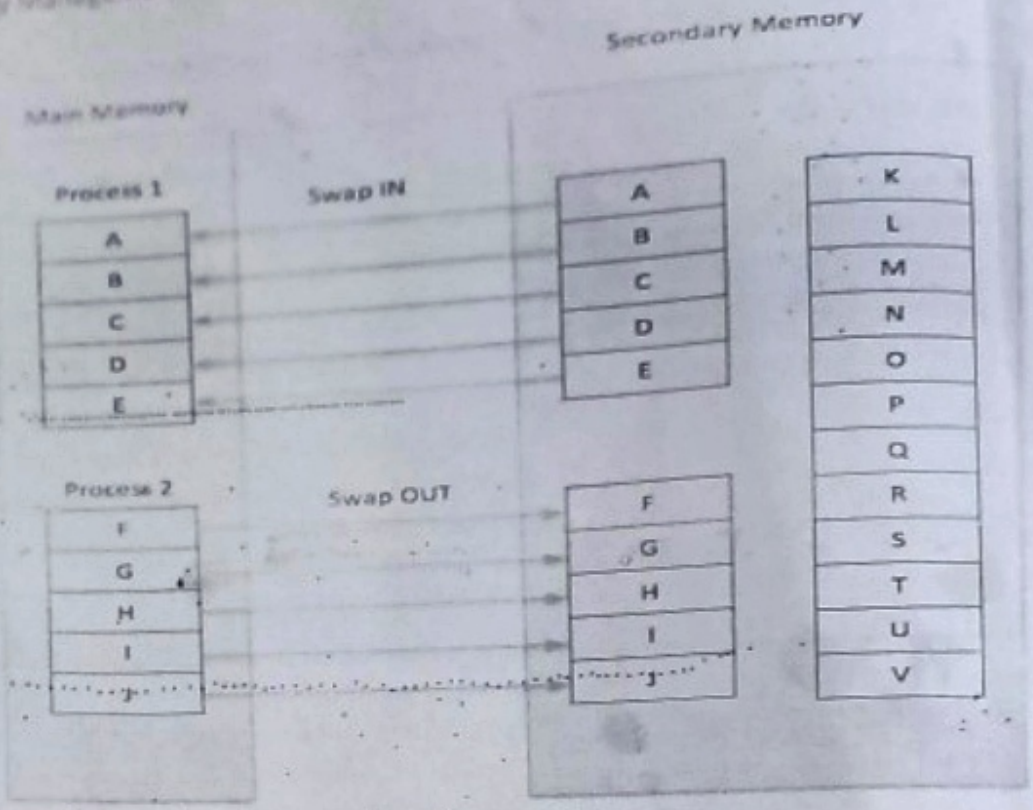
- **When a context switch occurs, the operating system does not copy any of the old program's pages out to the disk or any of the new program's pages into the main memory Instead, it just begins executing the new program after loading the first page and fetches that program's pages as they are referenced.**

Segmented Memory Allocation

- A well structured program consists of modules and also the relations between the modules or designed data a module is also known as a **segment**.
- A segment can be defined as logical group of information such as a subroutine or data area A segment may be a program of subprogram or a data base. A program and its data can be viewed as linked segment each segment has a name and offset.
- The physical addresses assigned to the segments are maintained in a segment table the memory management technique that allocates main memory to segments is called **segmentation**.


1
<MAIN 1>
3
<DATA 1>
5
<SIN>
7
<COS>
JOB 1 ADDRESS SPACE
2
<MAIN2>
4
<DATA 2>
6
<SQRT>
5
<SIN>
JOB 2 ADDRESS SPACE
1
2
3
4
5
6
7
Segment map
MONITOR
<DATA1>
<MAIN 1>
<MAIN 2>
<DATA 2>
<SIN>
<SQRT>
<COS>
Physical Memory

- Virtual memory can also be accomplished segmentation.
- Initially all the segments are stored secondary storage device. At the time of execution. there is a call statement that requires one segment when the required segment is not in the memory then the operating system searches the secondary storage device and bought into the memory.
- In this way the operation system links the calling segment to its coiler. The operation system must find a room for new segments by using compacting technique or by removing one of the segments in the memory.



- While executing a program, if the program references a page which is not available in the main memory because it was swapped out a little ago, the processor treats this invalid memory reference as **a page fault** and transfers control from the program to the operating system to demand the page back into the memory.

Advantages:

1. It eliminates fragmentation.
2. It avoids virtual memory.
3. It helps in dynamic linking.

Dis-Advantages:

1. There is a difficulty in managing segments of different sizes on secondary storage device.
2. The maximum size of the main memory.
3. Memory compacting is required.
4. Increases the hardware cost.

Comparison of Paging and Segmentation

Paging	Segmentation
Main memory is partitioned into frames or blocks	Main memory is partationed into segments
The logical address space is divided into pages by MMU	The logical address space is divided into segments as specified by the program
The scheme suffers from internal fragmentation or page breaks	The scheme suffers from External fragmentation
OS maintains a free frame list , so that searching of free frame is not necessary	OS maintains the particulars of available memory
OS maintains a page map table for mapping between frames and pages	OS maintains a segment map table for mapping
This scheme does not support the users view of memory	This scheme support the users view of memory
Processor use the page number and displacement to calculate absolute address (p, d)	Processor use the segment number and displacement to calculate the absolute address (p, d)