unit 4

DATE :___/___/___
PAGE :__________

Memory Management

Memory is the physical device which is used store program or data on a temporary or permanent basis for use in a computer or other digital electronic device.

There are various types of memories in a computer system and are accessed by various processes for their execution.

→ It is most imp. and more complex task fof OS. Memory management involves treating main memory as a resource to be allocated and shared among no. of active processes ie it is the act of managing computer memory.

→ It checks how much memory is to be allocated to processes. It decides which process will get memory at what time. It tracks whenever some memory gets freed or unallocated and correspondingly it updates the status.

Memory management requirements.

① Relocation

→ programmer doesnot know where the program will be placed in memory when it is executed.

→ While the program is executing, it may be swapped to disk and return to main memory at a different location.

② Protection.

→ processes should not be able to reference memory location in another process without premission..

→ processes should not be able to the crop the OS.

③ sharing

→ Allow several processes to access the same portion of memory in a controlled way.

⇒ Better to allow each process accessed to the same copy of the program rather than have their own separate copy

④ Physical organization.

⇒ Memory available for a program plus its data may be insufficient.

⇒ Programmer doesnot know how much space will be available.

⑤ logical organization.

⇒ Program return in modules.

⇒ Modules can be return and complied independently.

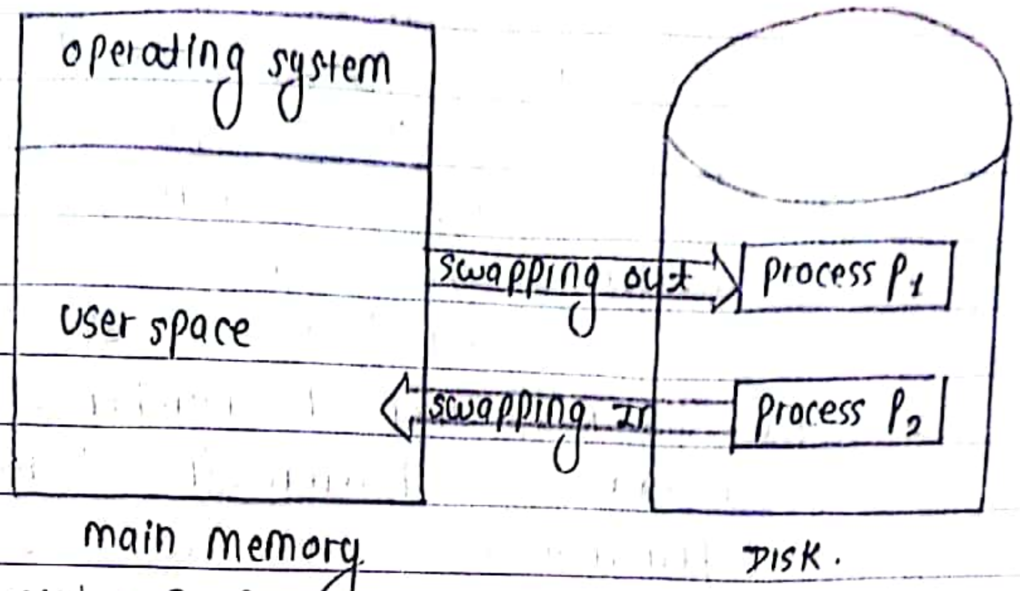
⑥ Swapping.

⇒ Any process which needs to be executed should be kept in main memory.

⇒ However, if the process in the main memory is not getting executed for some reason such as waiting for an event to occur or waiting for an I/O, then such process can be moved back to disk (swapout) and any process in the disk which is ready to be executed can be loaded into the main memory (swap in)

⇒ This process of swapping out or swapping in of processes betn main memory and disk in order to reduce the CPU ideal time is called as swapping.

⇒ The phenomenon of moving processes from main memory to secoundary memory is called swapping out. while the the phenomenon of moving process from secondary memory back to primary memory is called swapping in.



swapping of processes.

Advantages of swapping.

The advantages of swapping are listed below:

→ Swapping helps in acheiving the goal of maximum CPU utilization.

→ Swapping ensures proper memory availability for every process that needs to be executed.

→ Swapping helps to avoid the problem of process starvation means a process should not take much time for execution so that the next process should be executed.

→ CPU performs various tasks simultaneously with the help of swapping so that the processes donot have to wait much longer before execution.

→ Swapping ensures proper RAM (main memory) utilization.

→ Swapping creates a dedicated disk partition in the hard drive for swapped processes which is called swap space.

→ Swapping in OS is an economical process.

→ swapping method can be explained on priority-based sheduling where a high priority process is swapped in and the low priority process is swapped out which improves the performance.

absolute = virtual = logical

Disaduantages of swapping.

Some disaduantages of swapping are listed below:

⇒ If the system deals with the power-cut during bulky swapping activity then the user may lose all information which is related to the program.

⇒ If the swapping method uses an algorithms that is not up to the mark then the number of page faults can be increased and therefore this decreases the complete performance.

⇒ There may be inefficiency in case when there is some common resources used by the processes that are participating in the swapping process.

physical Address and logical address.

The physical address identifies the physical location of required data in memory. The user never directly deals with physical address but can access it by its corresponding logical address.

⇒ The logical address is an address that is generated by the CPU during program execution.

⇒ The logical address is a virtual address as it doesn't exist physically and therefore it is also known as virtual address.

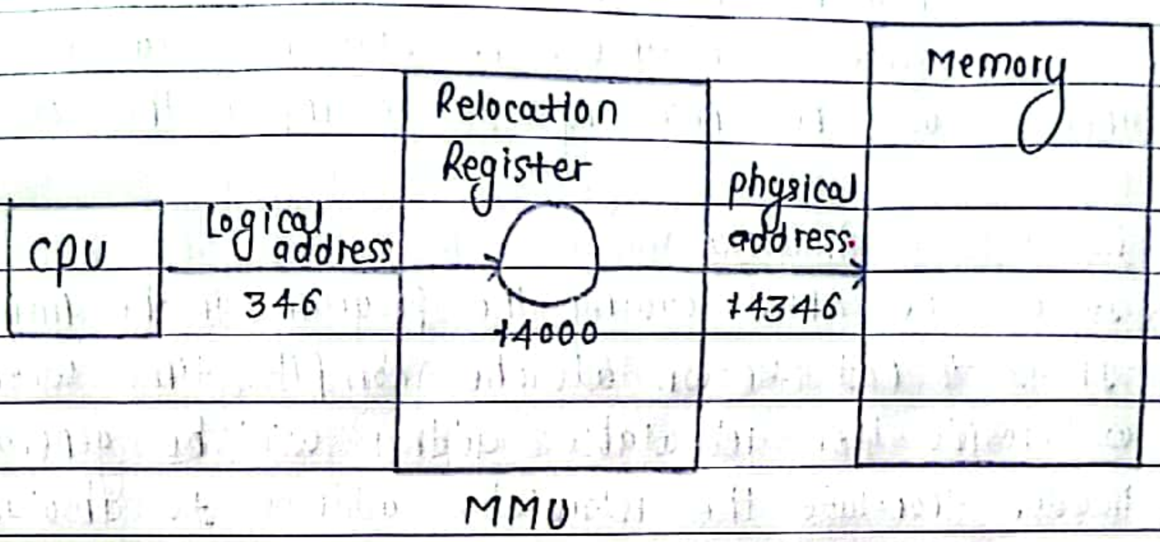
Memory Management Unit (MMU)

⇒ The user only things that he is accessing the data from the logical address.

⇒ The transaction from the logical to the physical address is done by special equipment in the CPU that is called memory management unit

DATE :__/__/__
PAGE :________

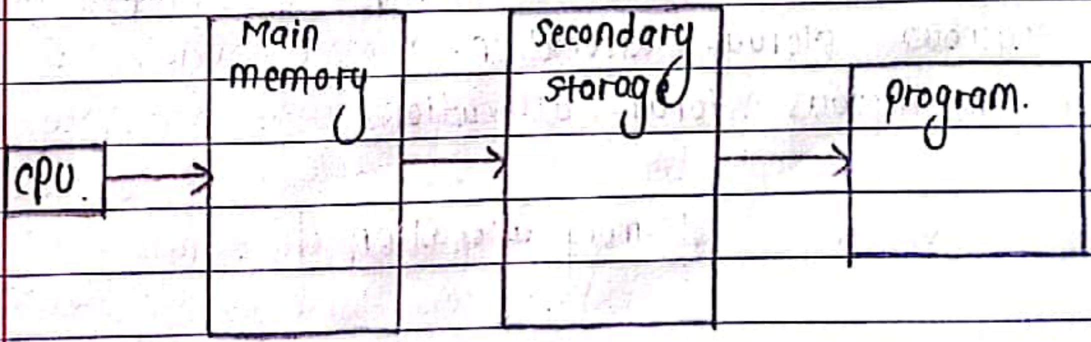
Mapping virtual address to physical address.



- The CPU generate the logical address (346)
- The MMU will generate the base address (+4000) which is stored in the relocation register.
- The value of relocation register (here +4000) is added to the logical address to get the physical address (ie +4000++4346).

Address binding.

- Address binding is the process of mapping from one address space to another address space.



- The addresses user in a source code. The variable, names, constants and instruction levels are the basic elements of the symbolic address space.

① Compile Time Binding.

⇒ If it is known at complile time where process / program reside in memory then absolute address is generated.

⇒ If, at some later time, the starting location change, then it will be necessary to recompile this code.

② Load Time Binding

⇒ It is done after loading the program in the memory.

⇒ If it is not known at the compile time where process will reside then relocatable address will be generated.

⇒ Loader translate the relocatable address to absolute address.

⇒ one the process loads, it doesnot move in memory.

③ Run Time Binding.

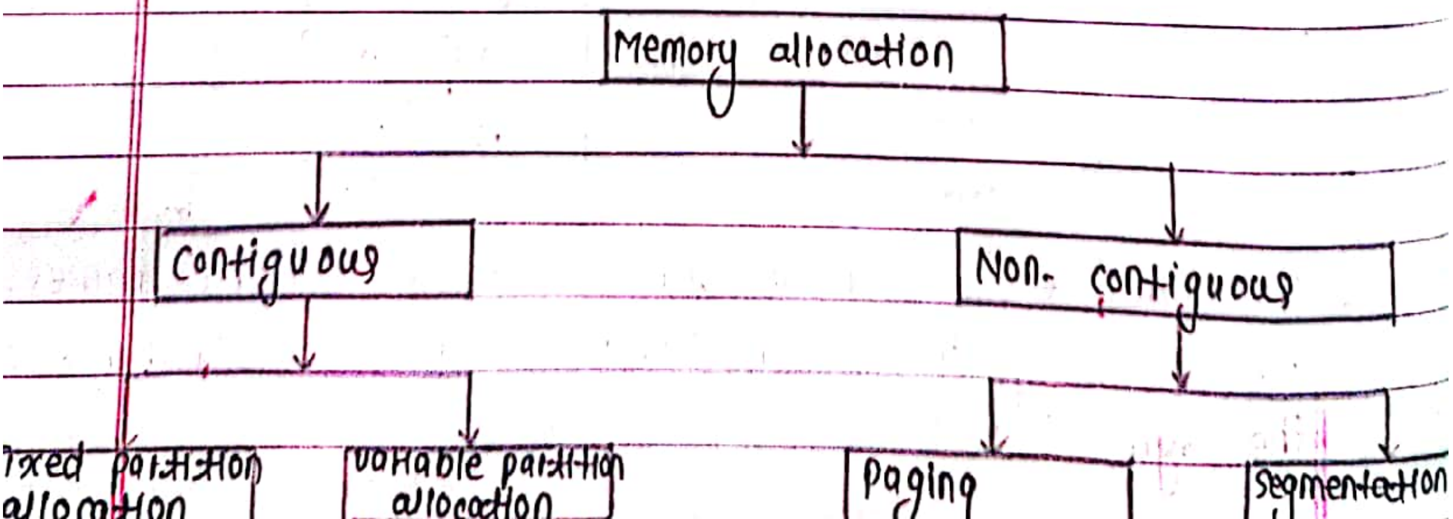
If process can be moved from one memory to another during execution then binding is done at Run time.

Memory Allocation Techniques.

Since the main memory must accommodate both the operating system and the various user process, main memory has tobe allocated in the most efficient way possible. Memory allocation is of two types:

① contiguous storage allocation.

② Non-contiguous storage allocation.



DATE : / /
PAGE :

Differencebetⁿ physical address and logical address.

parameter	Logical address	physical address.
Basic	Logical address is generated by CPU in perspective of program	physical address is the location that exists in memory unit
Address space	Logical address space is set of all logical addresses generated by CPU in reference to program	physical address is set of all physical addresses mapped to the corresponding logical addresses.
Visibility	user can view the logical address of the program.	user can never view physical address of the program.
Generation	Generated by CPU	Computed by MMU.
Access	The user can use the logical address to access the physical address.	The user can indirectly access physical address but not directly.
Editable	Logical address can be change.	physical address will not change.
Also called	Virtual address	Real address.

Contiguous storage allocation.

In contiguous storage allocation, each process occupy a single contiguous section of memory. In a multiprogramming environment, several programs reside in primary memory at a time and the cpu passes it's control rapidly bet^{n} these programs. To support multiprogramming one idea is to divide the main memory into several partition each of which is allocated to a single process. Depending on how and when partition are created, There are two types of memory partition.

1. fixed or static partitioning
2. variable or dynamic partitioning

1 fixed or static partitioning (Multiprogramming with fixed partitioning)

In fixed partitioning, main memory is divided into a no. of static partition at system generation time. There are two alternatives for fixed partitioning is equal size partition and unequal size partition

OS
8MB
8MB
8MB
8MB

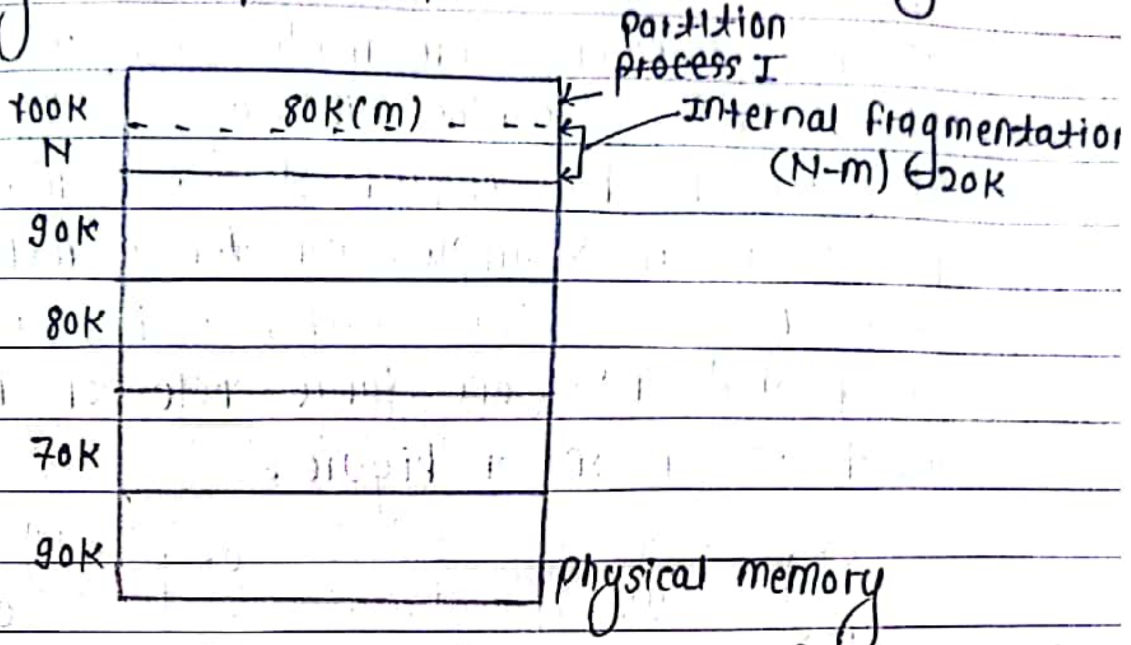
fig (a) Equal size partition

OS
2 MB
6 MB
8 MB
16MB

fig (b) unequal size partition.

fixed Partitioning of 32MB memory

To understand internal fragmentation in detail, consider an example in which we have a physical memory with the following fixed partition.



When a process or program of size 80K(m) arrives, it is accommodate in ~~partition~~ process I but ~~Process~~ partition I is 100K(N) size so, M<N therefore M can be given ~~process~~ partition I the left over unused space is (N-m) =(100-80)K = 20K. This causes internal fragmentation of 20K here.

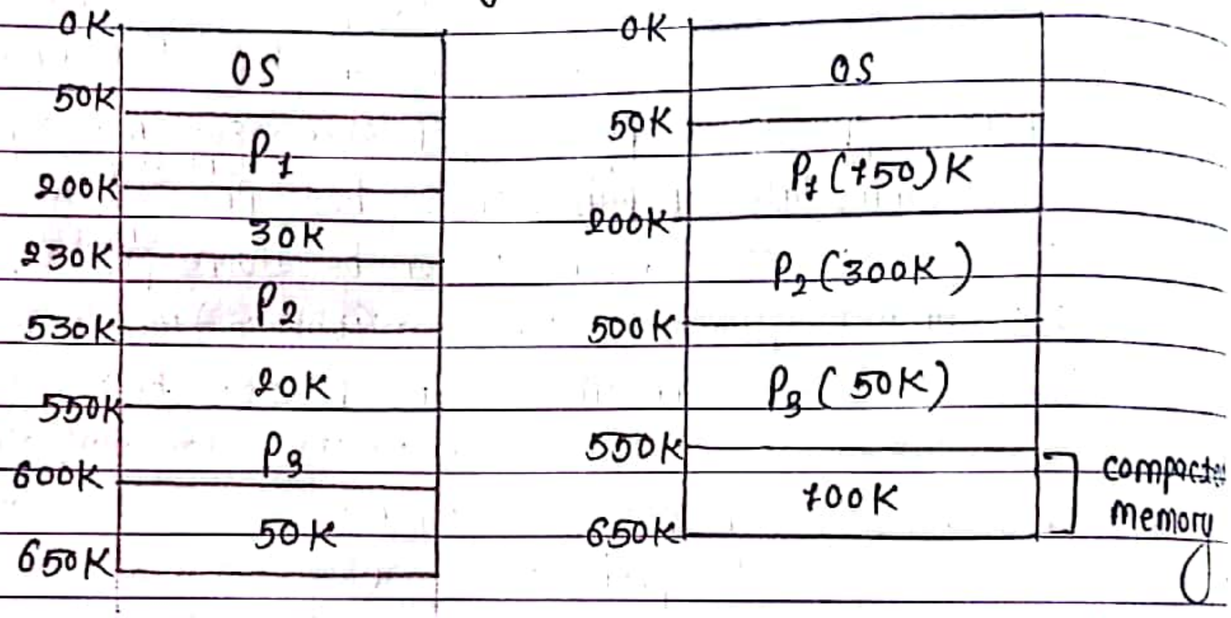
② Variable or dynamic partitioning.

To overcome the probleth with fixed partitioning, the concept of dynamic partition was introduced. With dynamic partitioning, the partition are variable length and number. When a brought into main memory, it is allocated exactly as much memory as it require and no more.

External fragmentation

External fragmentation exist when there is a enough total memory space to satisfy a requesting process but the available space are non-contiguous; storage is fragmented into a large no. of small holes (free spac

To solve this problem, compaction is employed. Compaction is a technique by which the processes are relocated in such as way that the small chunks of free memory are made contiguous to each other clubbed together into a ~~stg~~ single free portition that may be big enough to accommodate additional process as an example consider a memory that has three holes of size 30K, 20K, 50K that have been compacted into one large hole or block of 100K which is shown in figure.

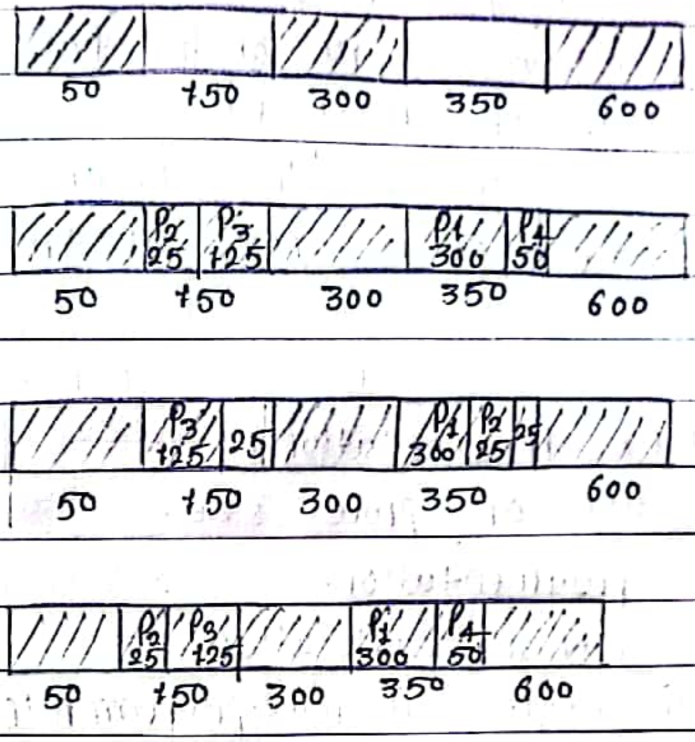


Memory placement algorithms.

In order to load a new process, OS checks for the free memory partition with the help of PDT (partition description table) if the search is found to be successfull. then the entry is marked as "allocated". when the process terminates or swapped out, it is updated as "free". Three most common stati to allocate free partition to the new process are

1. First fit
2. Best fit

③ worst fit.



- first fit

In a first fit approach, the first free portition is allocated large enough to accomodate the process.

- Best fit

In best fit approach, the smallest free portition is allocated that meets the requirement of the process

- worst fit

In a worst fit approach, the largest available partition is allocated to the newly entered process.

Advantages of fixed or static portitioning.

⇒ Easy implementation.
⇒ External fragmentation.
⇒ Internal fragmentation.
⇒ Limiting the process size.
⇒ Lesser degree of multiprogramming.
⇒ Lower OS overhead.

Disadvantages

⇒ Requires entire program to be stored contiguously.

⇒ Jobs are allocated space on the basis of first available partition of required size.

⇒ work well only if all the Jobs are of same size or if the sizes are known ahead of time.

Advantages of variable or dynamic partitioning.

⇒ No internal fragmentation.

⇒ No limitation on process size.

⇒ External fragmentation.

⇒ Complex memory allocation.

⇒ Dynamic degree of multiprogramming.

Disadvantages.

⇒ Limited flexibility in accomodating changing memory requirements of processes

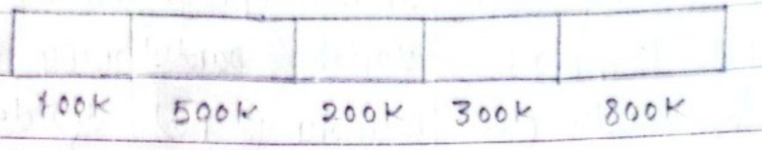
⇒ Wastage of memory resources when a partition is not fully utilized.

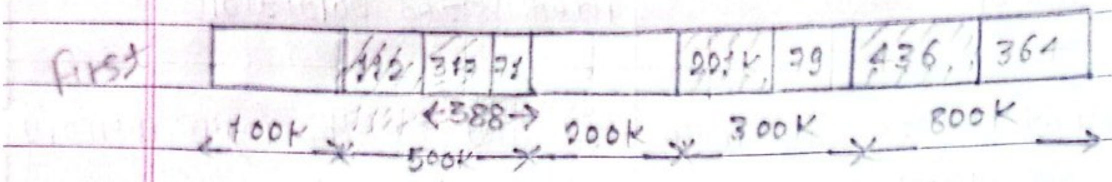
⇒ High fragmentation when processes of different sizes are present in the system.

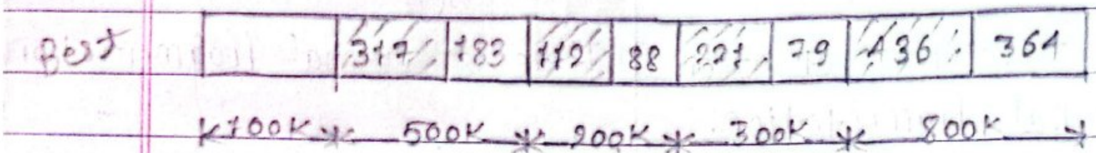
Differences betn fixed partitioning and variable partitionin

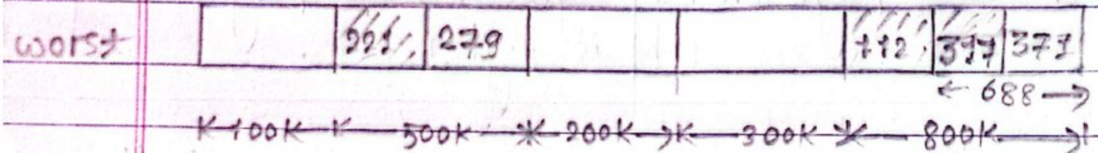
	fixed partitioning	Variable partitoning
1	In a multi-programming with fixed partitioning the main memory is divided into fixed sized partitions	In multi-programming with variable partitioning the main memory is not divided into fixed sized partitions.
2	It doesnot utilize the main memory effectively.	It utilizes the main memory effectively.
3	There is presence of external and internal fragmentation.	There is extenal fragmentation.
4	Degree of multi-programming is less.	Degree of multi-programming is higher.
5	It is more easier to implement.	It is less easier to implement
6	There is limitation on size of process.	There is no limitation on size of process.
7	only one process can be placed in a partition.	In variable partitioning, the process is allocated chunk of free memory.

Q. Given memory partition is 100K ,500K, 200K ,300K and 800K (in ordered) how would each of first fit, best fit and would worst fit algorithm plases. processes of 112K ,317K, 221 K and 436K in order









Non-contiguous storage allocation:

Employing compaction technique to avoid external fragmentation can be expensive. Another possible soln to external fragmentation is to have non-contiguous storage allocation. There are two main methods of non-contiguous storage allocation.

(1) paging

A memory management ~~skim~~ scheme that premits the physical (address space of a process to be non-contiguous is called paging. paging avoids external fragmentation

6
4

500
317
183

388
317
71

300
221
79

800
112

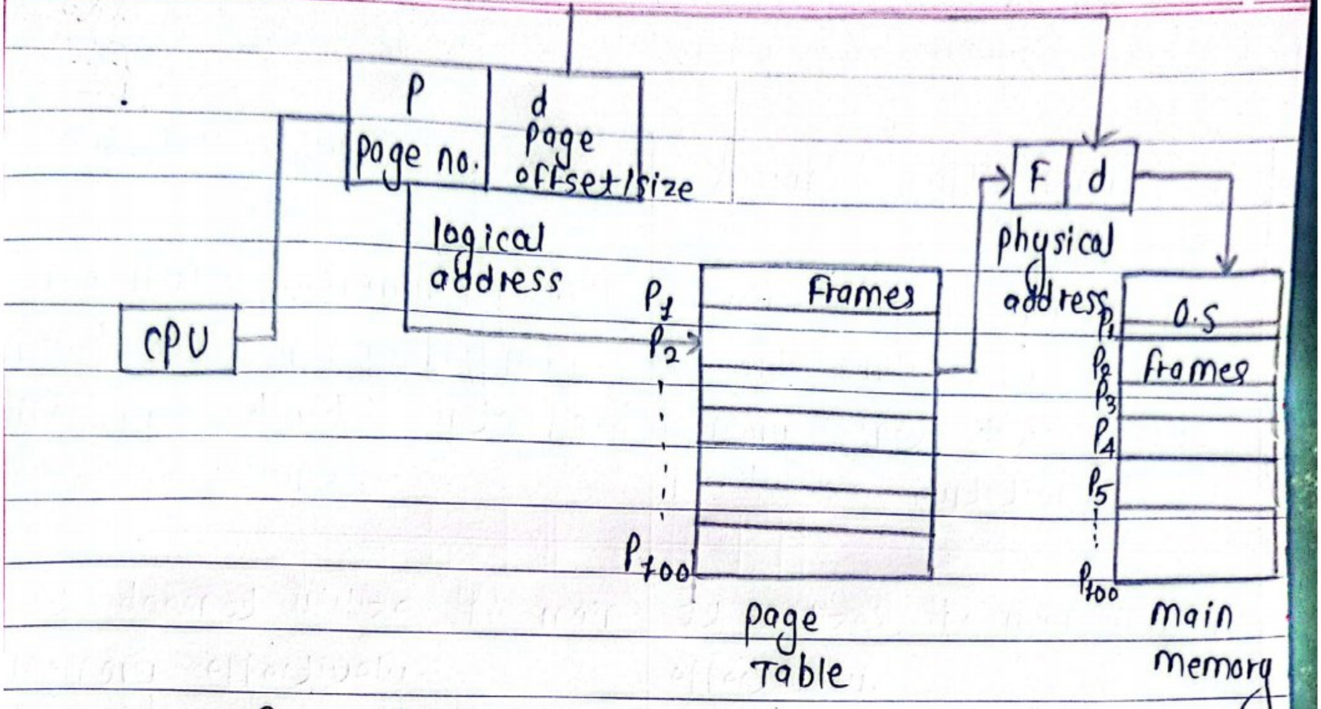


Fig : paging

as well as the need for compaction.

The basic method for implementing paging involves breaking physical memory into fixed size block called frames and breaking logical memory into block of same size called pages.

The hardware support for paging is illustrate in figure. Every address generated by the cpu is divided into a two parts : a page number (p) and page offset (d). The page no. is used as an index into a page Table. The page Table contains the base address of each page in physical memory. This base address is combined with the page offset ~~to~~ define the physical memory address that is sent to a memory unit.