PAGE NO.:
DATE: 17 / Jan / 2019

UNIT - 1

★ **Operating System** – An operating system is a system program that enables the computer hardware to communicate and operate with the computer software.

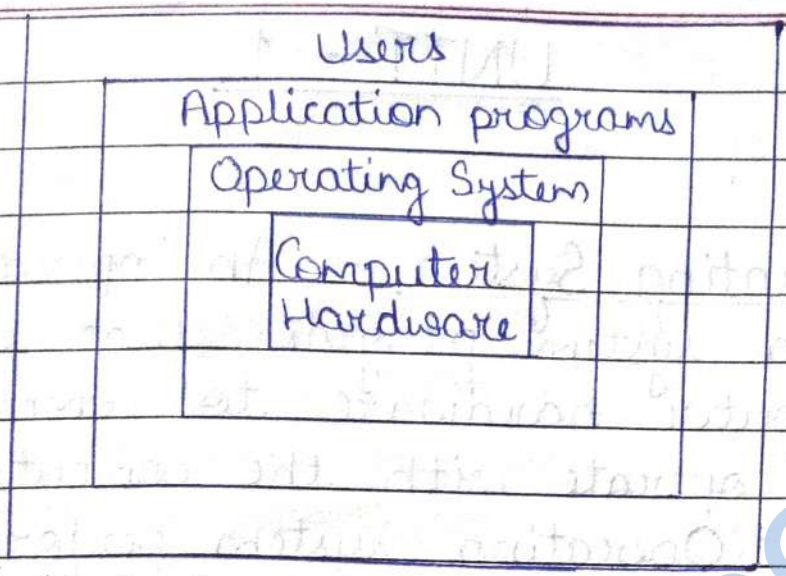
Operating system perform basic task such as recognising input from the keyboard and sending output to the display screen, keeping track of files and directories on the disk and controlling peripheral devices such as disk driver and printer.

The operating system is also responsible for security ensuring that unauthorise users do not access the system.

Some examples of operating system are – Unix, MS-DOS, MS-Windows, Windows-NT, VMS and UM.

"An operating system is a program (system software) which act as an interface between user and computer hardware."

A computer system can be logically divided into 4 components –



(Logical Computer Hierarchy)

* Functions of operating system :-

1.) Memory management function -

- Primary (main) memory - Find free space in memory and allocate it to different process. Provide direct access storage for CPU. Process must be in main memory to execute.

- Operating system must -
(i) Decide when to load each process into memory.
(ii) Decide how much memory space to allocate each process.
(iii) Decide when a process should be remove from memory.

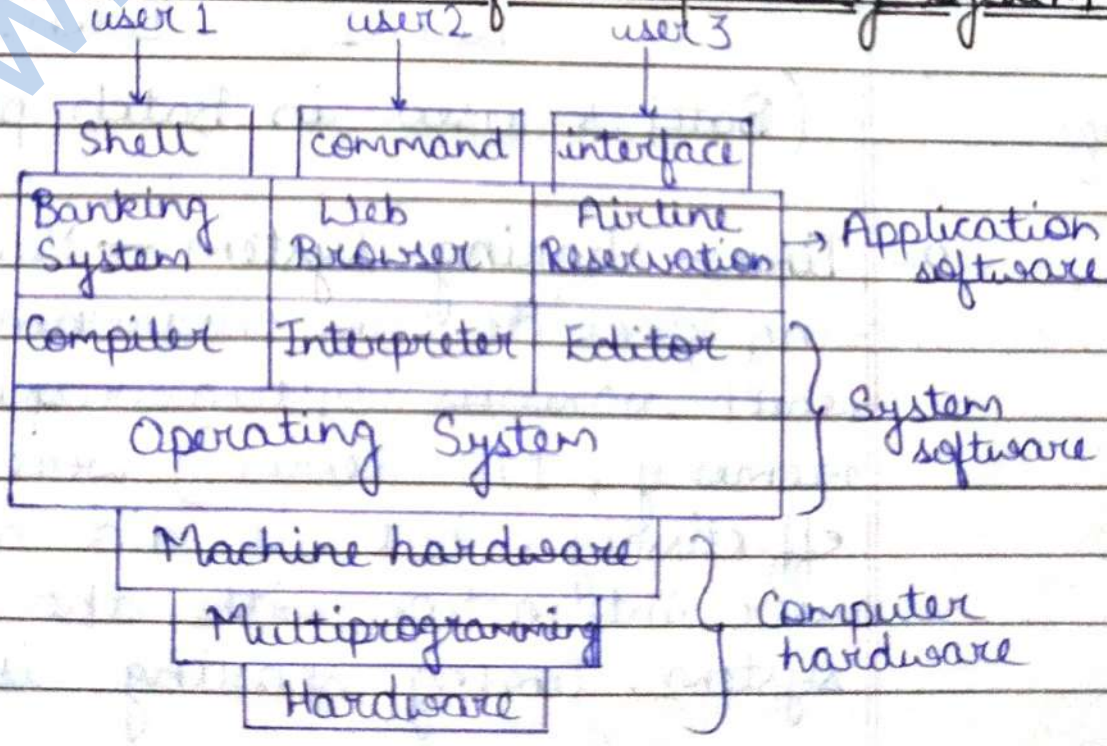
(iv) Protect memory space.
(v) Allocate - deallocate space for process.

2.) Process management function – Allocates the processor to execute a selected process.

3.) Input-Output device management function- It allocates a device to process.

4.) File management function – Keep track of all information about files, how they are open and close.
A file system supports directories which contain file and other directories name, size, data created, last modified etc.

* Layered abstraction of an operating system-



Multiprogramming → Context switch

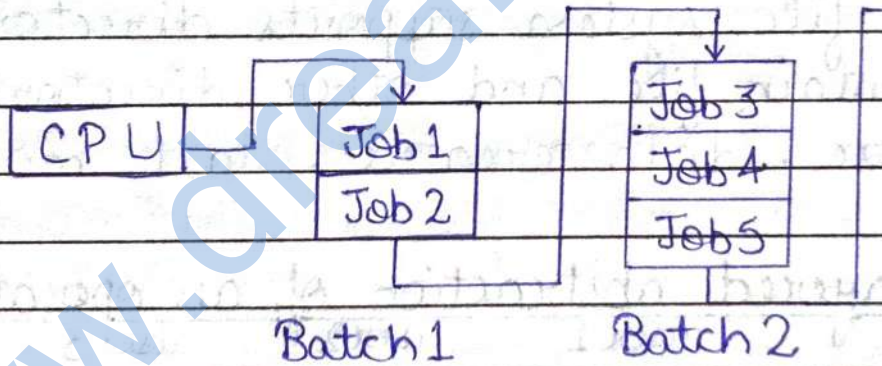
Multitasking → Time-sharing

PAGE NO :

DATE : 19 / Jan / 2019

1- Batch system
2- Time sharing system
3- Real time system
4- Parallel system
5- Distributed system

1- **Batch processing system** – Batch is define as a group of jobs with similar name.
Computer executes each batch sequentially processing all jobs of a batch considering them as a single process is called batch processing.



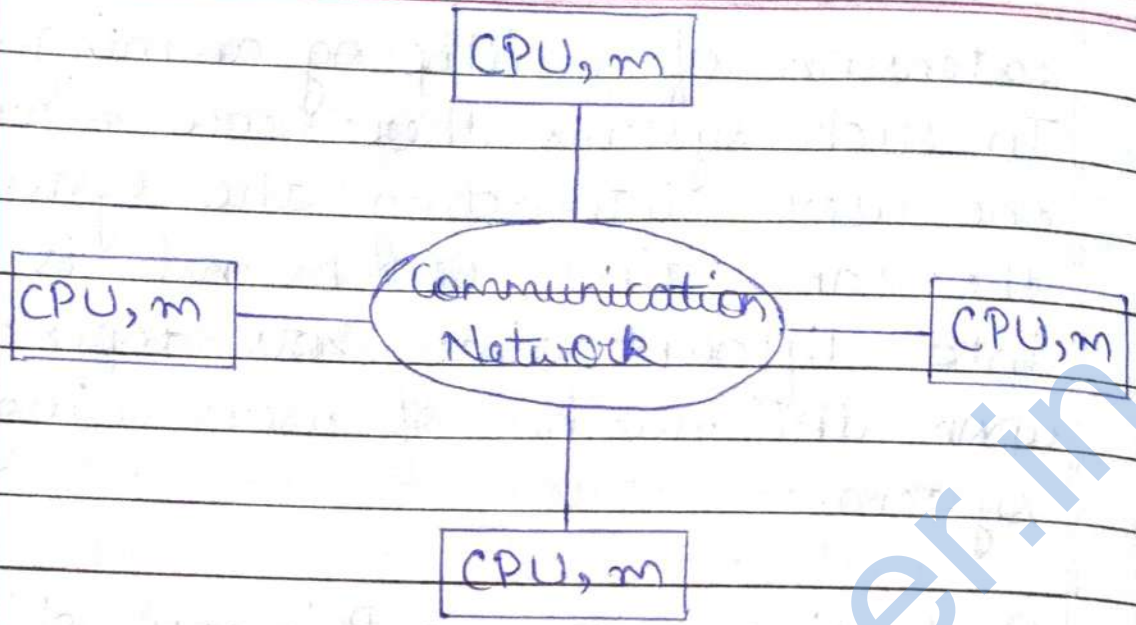
(Batches used in Batch processing)

2- **Time-sharing system** – Multiprogramming provide an environment in which various system resources (CPU, memory, I/O device) were utilize effectively, but it did not provide user interaction with the computer system. Timing sharing is a logical

extension of multiprogramming.
In such systems there are more than one user interacting the system at the same time. CPU bound is divided into different time slots depending upon the number of users using the system.

3- Real-time system — Primary objective of real time system is to provide quick response time and meet a scheduling dead line.
Real time system has many events that must be accepted and process in a short time or within certain deadline.

4- Distributed system — Distributed operating system are the operating system for a network of computers connected by communication network through a message passing mechanism.
A distributed operating system manage the hardware and software resources of a distributed system.



5- Parallel system - Parallel operating system are the interface between parallel computers and the applications that are executed on them. They translate the hardware capabilities into concepts usable by programming language.

Client - server system - A client-server application is a distributed system made up of both client and server software.
Client - server communication involves two components, namely a client and a server. There are usually multiple clients in communication with a single server. The clients send requests to the server and the server

responds to the client request.

* **Memory management :-** It is the functionality of an operating system which manage primary memory and move process back between main memory and disk during execution.

Functions of memory management -

1- Keep track of every memory location.
2- Track of how much memory is allocated.
3- Takes the decision, which process will get memory and when.
4- It updates the memory status when it is allocated or free.

* **Physical and logical address space -**

Logical address space - It is generated by the CPU. It is also known as virtual address.

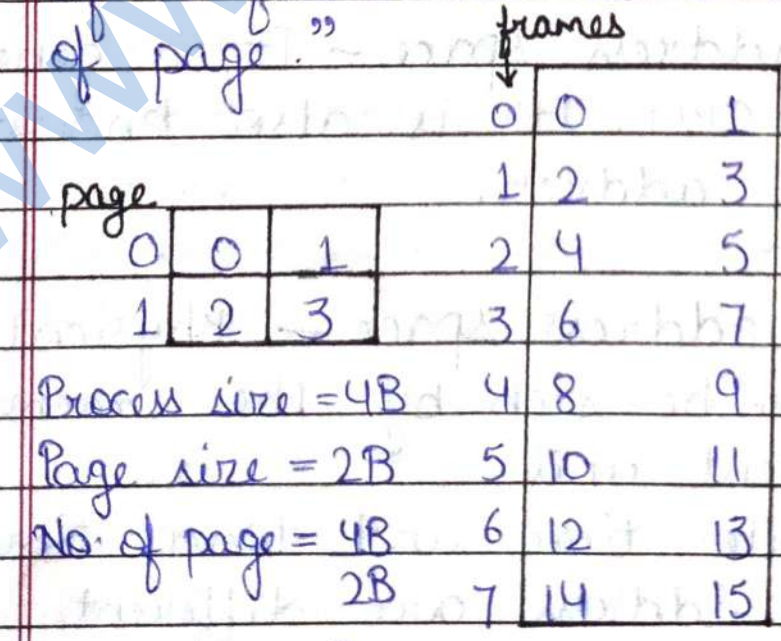
Physical address space - Physical address space can be seen by the memory management unit.
In compile time and load time memory address are different in

execution time, physical address space are same.

* **Paging :-** Paging is a memory management scheme that remove the requirement of contiguous allocation of physical memory. This scheme permits the physical address space of a process to be non-contiguous.

The physical memory is divided into a number of fix size blocks called frames and the logical address space is also divided into fix size block called page.

"When a process is to be executed its pages are loaded into any available memory frames. The size of a frame is same as the size of page."



Process size = 4B
Page size = 2B
No. of page = $\frac{4B}{2B}$ = 2

Memory size = 16B
Frame size = 2B
No. of process = $\frac{16B}{2B}$ = 8 frames

If the capability of physical memory is m and size of each page is P, then the number of frames in physical memory will be

$$f = \frac{m}{P}$$

Ex- Consider a logical address space of 8 page of 1024 words. Each map onto a physical memory of 32 frames, then—

1.) How many bits are there in physical address?

2.) How many bits are there in logical address?

Sol: Let the number of bits in the physical memory = m

Then the size of physical memory = 2^m

Given

no. of pages = $8 = 2^3$

no. of frames = $32 = 2^5$

Size of each frame = size of each page

$= 1024$

$= 2^{10}$

$$f = \frac{m}{P}$$

$$2^5 = \frac{2^m}{2^{10}}$$

$2^m = 2^{10} \times 2^5$

$\boxed{m = 15}$

no. of page = $\frac{\text{size of logical memory}}{\text{size of each page}}$

$2^3 = \frac{2^n}{2^{10}}$

$2^n = 2^3 \times 2^{10}$

$\boxed{n = 13}$

★ **Segmentation** – Segmentation is another technique for the non-contiguous storage allocation. It is different from paging as pages are physical in nature and hence are of fix size, whereas segments are logical division of a program and hence are of variable size.

It is a memory management scheme that support the user view of memory rather than system view of memory. Each segment has a name and length which is loaded into physical memory.

★ **Virtual memory** – It is a virtual resource of a computer. It is an

illusion that a computer system resources process more memory that is actually having. This makes a process independent of the size of real memory (main memory). It also permits large number of process to share a computer system without constraining each other.

* Demand Paging :- In demand paging, a page is break into the memory for its execution only when it is demanded otherwise it is remain in backing storage (disk).
This method is the combination of paging and swapping method. The main requirement of this method is that the complete program should be present in the backing storage in the form of page.

- Swapping - Swapping is used to swap the content of the program from disk to main memory when it is required.

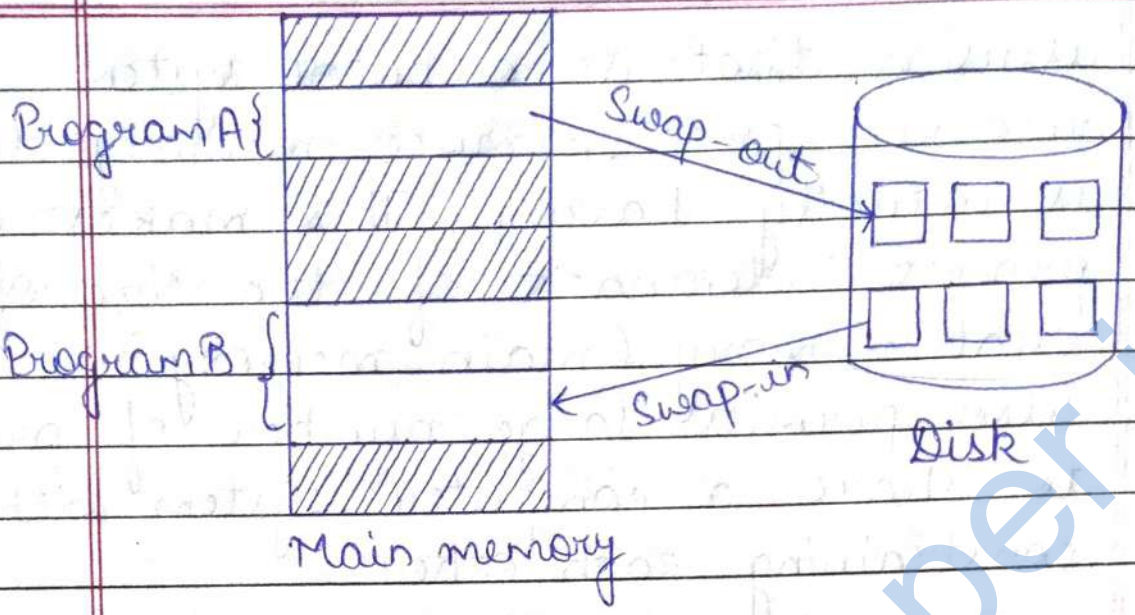


fig:- Swapping with paging

Imp.

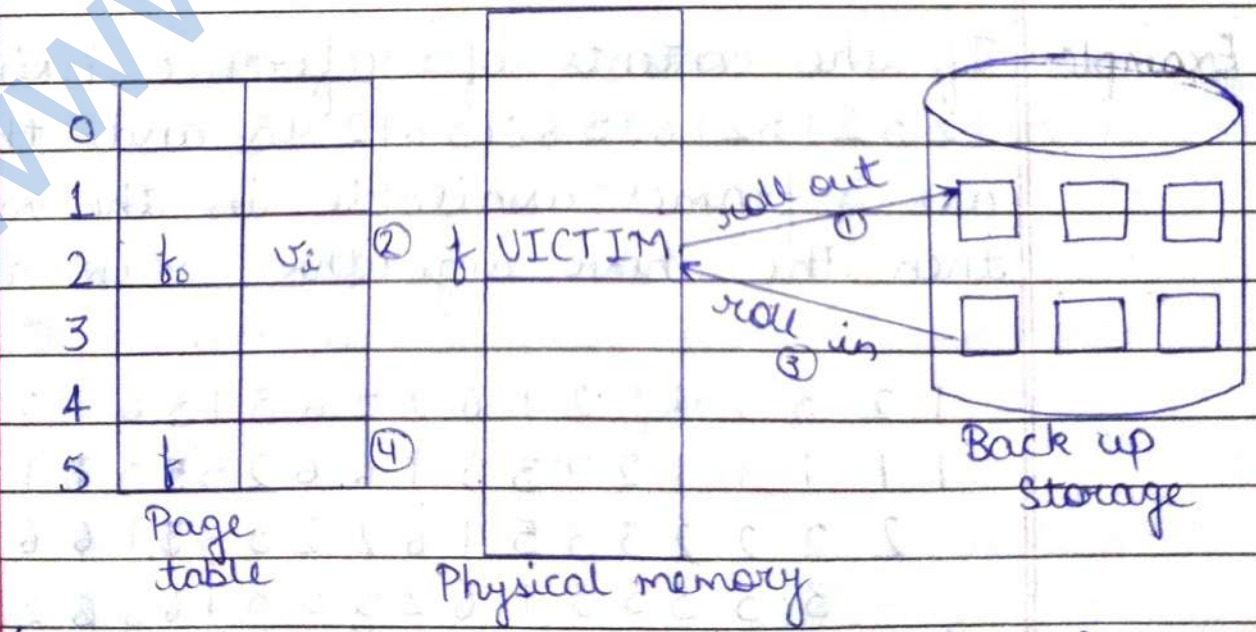
★ **Page Replacement Algorithms :-**

Principle of page replacement – The page replacement is done by swapping. The required page from back-up storage to main memory and vice-versa. This swapping is done by checking the content of physical memory if there is a free frame in the memory then swap-in the required page into the frame is free.

In case there is no free frame in physical memory then first find the frame which is not currently in use. The content of this frame is

swap-out from memory to back-up storage, then bring the required page in the frame which is now free.
Following steps are performed in page fault routine for page replacement -

1- Find the location of desired page on back-up storage.
2- Find the free frame.
(a) If the frame is free, use it.
(b) Otherwise find the frame which is not currently being used this is called as victim frame.
(c) Write the content of victim frame on back-store and change the page-table entries to indicate that the page is no longer in the main memory.
3- Read the desired page into free frame, change page table and frame entries.



(fig:- Example of page replacement)

PAGE NO :
DATE : / /

1- Swap out the victim frame.
2- Update its page table entry.
3- Swap in the desired page.
4- Update its page table entry.

* 1.) FIFO Replacement Algorithm – In this method, the oldest page present in the memory is selected for replacement. For this the page at the head of the queue is replace and when the page is break into the main memory then it is place at the end tail of the queue.

(fig:- Queue used in Page Replacement Algorithm)

mple– If the contents of reference using is 1 2 3 2 1 5 2 1 6 2 5 6 3 1 3 6 1 2 4 3 and there are 3 frames available in the memory then the above algorithm work as –

1	2	3	2	1	5	2	1	6	2	5	6	3	1	3	6	1	2	4	3
1	1	1	1	1	2	2	3	5	1	6	6	2	5	5	3	3	1	6	2
	2	2	2	2	3	3	5	1	6	2	2	5	3	3	1	1	6	2	4
		3	3	3	5	5	1	6	2	5	5	3	1	1	6	6	2	4	3
*	*	*			*		*	*	*	*		*	*		*		*	*	*

PAGE NO :
DATE : 2nd / Feb / 2019

Total no. of page fault = 14

2.) Optimal Replacement Algorithm – This algorithm has lowest page fault rate for all algorithm of page replacement.
"This algorithm states that replace that page which will not be use for longest period of time."
Future knowledge of reference string is required to understand this algorithm following example is consider –

1	2	3	2	1	5	2	1	6	2	5	6	3	1	3	6	1	2	4	3
1	1	1	1	1	1	1	1	6	6	6	6	6	6	6	6	6	2	2	2
	2	2	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	4	4
		3	3	3	5	5	5	5	5	5	5	3	3	3	3	3	3	3	3
*	*	*			*			*				*	*				*	*	

Total no. of page fault = 9

3.) LRU (Least Recently Used) Algorithm – In this algorithm, the page has been not used for longest period of time is selected for replacement.
To understand this concept consider the following example –

PAGE NO :
DATE : / /

1	2	3	2	1	5	2	1	6	2	5	6	3	1	3	6	1	2	4	3			
1	1	1	1	3	2	1	5	2	1	6	2	5	6	6	1	3	6	1	2			
	2	2	3	2	1	5	2	1	6	2	5	6	3	1	3	6	1	2	4			
		3	2	1	5	2	1	6	2	5	6	3	1	3	6	1	2	4	3			
*	*	*			*			*		*		*	*				*	*	*			

Total no. of page fault = 11

Ques- 7 0 1 2 0 3 0 4 2 3 0 3

- FIFO Replacement -

7	0	1	2	0	3	0	4	2	3	0	3
7	7	7	0	0	1	2	3	0	4	2	2
	0	0	1	1	2	3	0	4	2	3	3
		1	2	2	3	0	4	2	3	0	0
*	*	*	*		*	*	*	*	*	*	

Total no. of page fault = 10

- Optimal Replacement -

7	0	1	2	0	3	0	4	2	3	0	3
7	7	7	7	7	0	0	2	2	2	3	3
	0	0	0	0	2	2	3	3	3	4	4
		1	2	2	3	3	4	4	4	0	0
*	*	*	*		*		*			*	

Total no. of page fault = 7

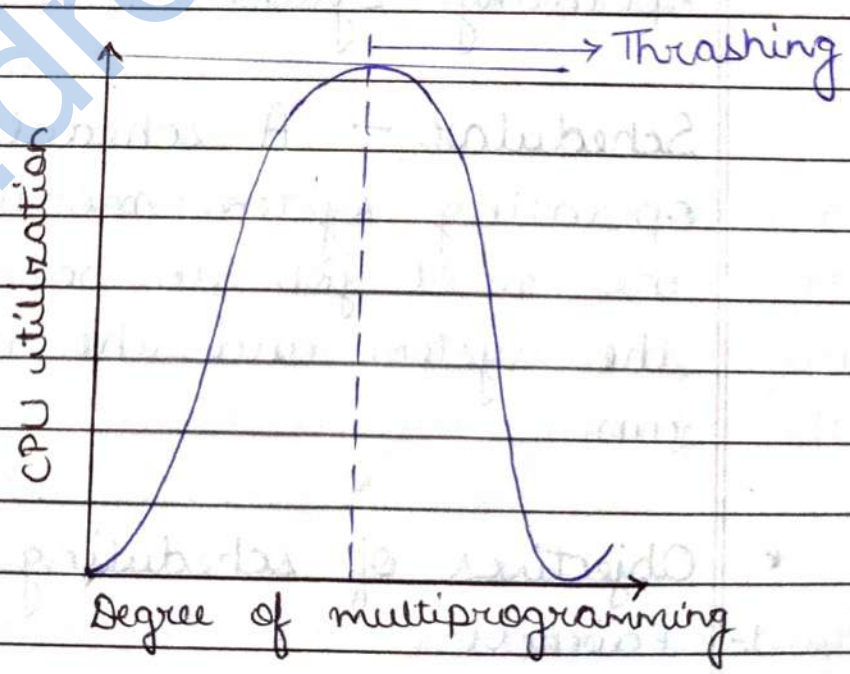
PAGE NO:
DATE: / /

- LRU –

7	0	1	2	0	3	0	4	2	3	0	3	
7	7	7	0	1	2	2	3	0	4	2	2	
	0	0	1	2	0	3	0	4	2	3	0	
		1	2	0	3	0	4	2	3	0	3	
*	*	*	*		*		*	*	*	*		

* **Thrashing :-** The phenomena of moving page from primary to secondary storage or vice-versa consume a lot of compute energy but accomplishes very few results. This situation is called thrashing.

The process is in thrashing if it is spending more time in paging instead of their execution.



(fig :- Thrashing)