# Memory Organization:

The storage device which stores the instructions & data required for operation or execution of program is termed as memory system of the computer. These storage devices plus the algorithms implemented by h/w or s/w are needed to manage the stored information.

Memory Characteristics: 8.

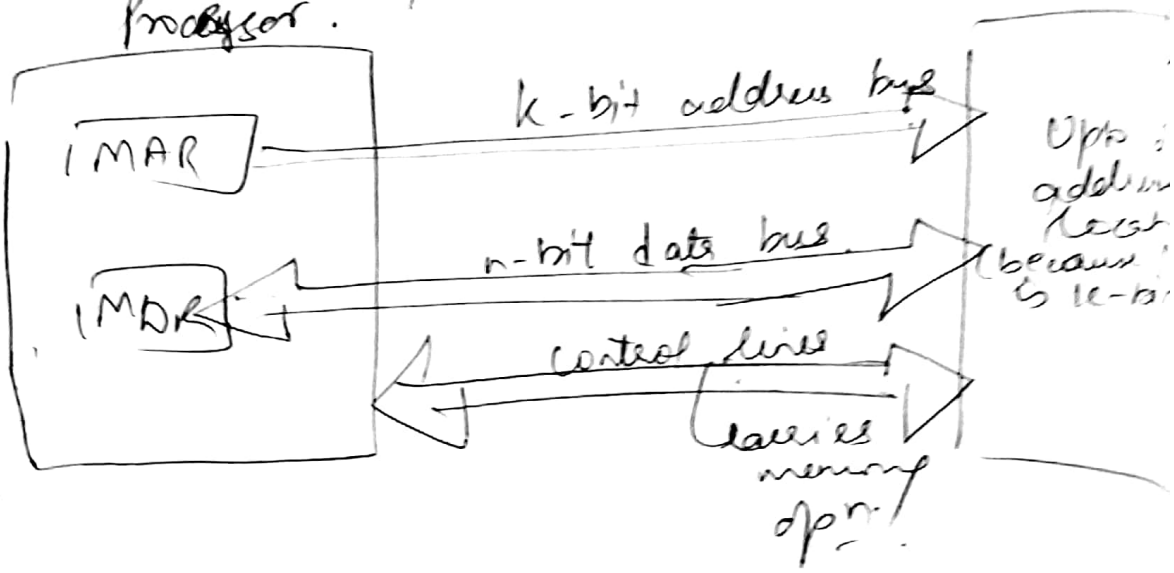
(1.) Location of memory (within CPU or outside CPU).
(2.) Capacity - (Measured in Bytes, KB, MB, GB, etc.,
(3.) Unit of Transfer (Data Rate b/w CPU & Memory, I/O)
(4.) Access Method - (Sequential or Random)
(5.) Performance (Overall CPU Execution Time)
(6.) Physical Characteristics of Memory Device
→ Power consumption
Is Erasable.
Is Cost per Bit Storage.
(7.) Bandwidth => rate of transfer.

Note: Max. size of memory that can be used in any computer is determined from no. of address lines provided by the processor. E.g.
if processor has 20 address lines, it is capable of addressing upto $2^{20} = 1$ (Mega) Memory locations. Similarly, processors whose instructions generate 32-bit address can access a memory

Scanned by CamScanner

that contains upto $2^{32}$ = 4 (Giga) memory

Max. no. of bits that can be transferred from memory or to the memory depend on the data lines supported by the processor, i.e.




Types of Memory:-

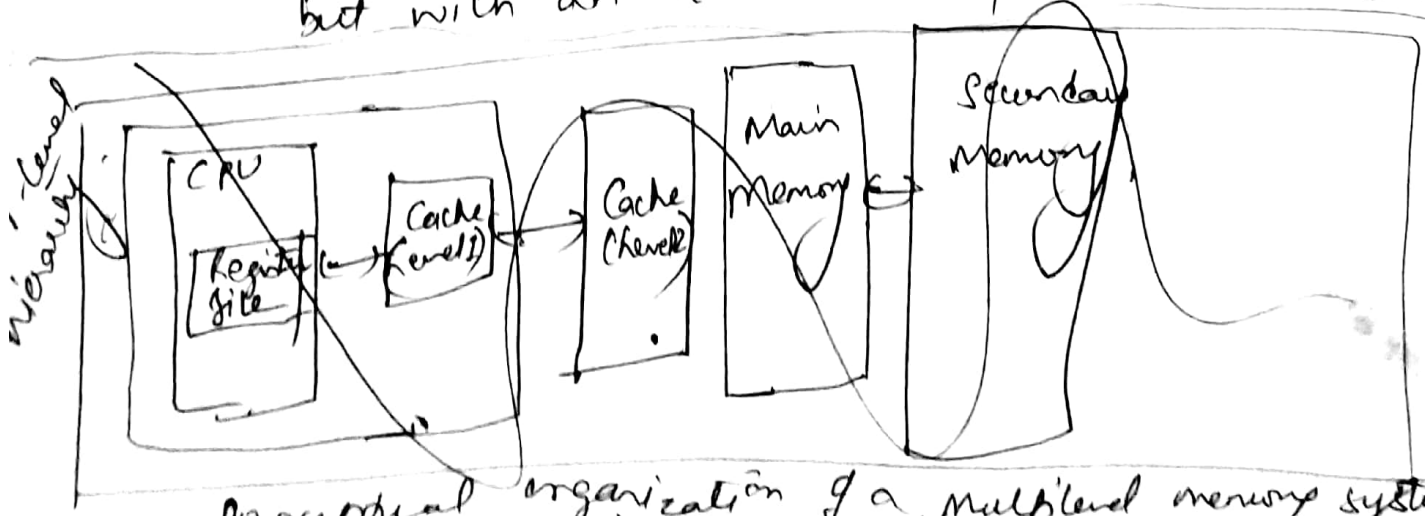
(1) CPU Registers:- These are very high-speed storage devices present in the CPU for temporary storage of instructions & data

(2) Main Memory (primary memory):- The memory Unit that directly communicates with CPU is called as the Main Memory. This has fairly fast device stores programs & data are in active use. It's access is slow than registers b'coz of larger capacity & physical separation ~~b/w~~ c. from CPU. Its capacity is b/w 1 & 2^10 MBytes.

Scanned by CamScanner

(3) Auxillary Memory: Devices that provide backup storage are called auxillary memory. ~~The most~~ common auxillary They are used for storing system programs, large data files, & other backup information. The info. stored in it is transfered to main memory when required. It is much larger in capacity but also much slower than main memory. Storge capacity is generally in gigabytes & access times are measured in millisecondes. E.g. Magnetic disks & Tapes.

(4) Cache: It is very high-speed memory device used to increase the speed of processing by making current pgm & data available to CPU at a rapid rate. It is employed in computer system to compensate for the speed differential b/w main memory access time & processor logic. It is positioned logically b/w CPU registers & main memory. It's storage capacity is less than that of main memory, but with an access time of one to three cycles.




Conceptual organization of a multilevel memory system in a computer.

Scanned by CamScanner

The cache is used for storing segments of pgm currently being executed in the CPU & temporary data frequently needed in present calculations.

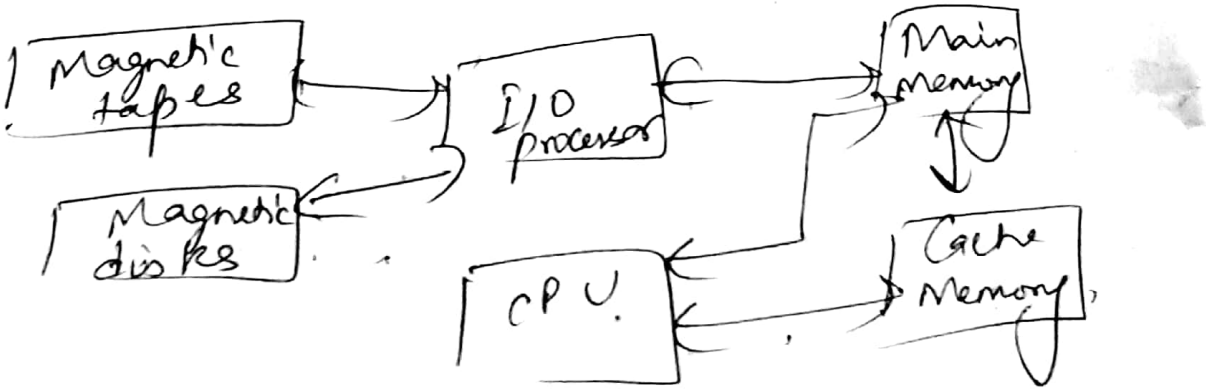
Diff. b/w auxillary memory & cache memories.

(1.) Cache holds those parts of pgm & data that are most heavily used while auxillary memory holds those parts that are not presently used by CPU.

(2.) CPU has direct access to both cache & main memory but not to auxillary memory.

## Memory Hierarchy. (Diagram).

Memory Hierarchy system consists of all the storage devices employed in a comp. system from slow but high-capacity auxilary memory to a relatively faster main memory, to an even smaller & faster cache memory accessible to the high-speed processing logic). Major components in a typical memory hierarchy are:




Scanned by CamScanner

## Levels of Memory Hierarchy:

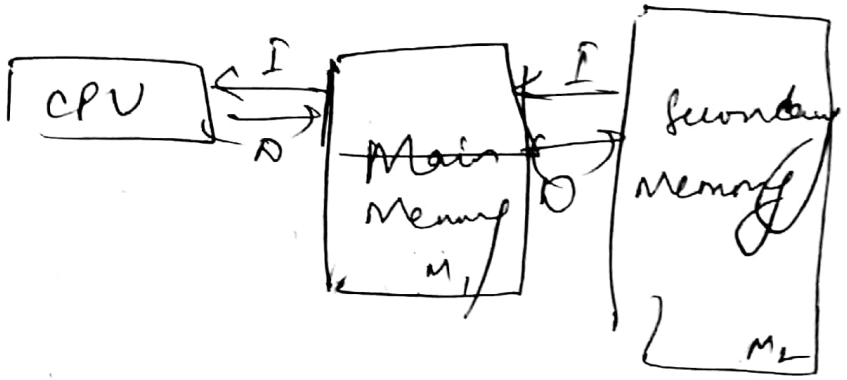
The I/O processor manages data transfers b/w auxiliary memory & main memory, cache organization is concerned with transfer of info. b/w main memory & CPU, Thus, each device is indexed with a diff. level in memory hierarchy system. The reason for having 2 or 3 levels of memory hierarchy is economics. Not all accumulated info. is needed by CPU is available at one place, at one time ∴, it is more economical to use low-cost storage devices to serve as back up for storing the info. that is not currently used by CPU. As the storage capacity of memory increases, the cost per bit for storing binary information decreases and the access time of memory becomes longer. The auxiliary memory has a large capacity, is relatively inexpensive, but has low access speed compared to main memory. The cache memory is very small, relatively expensive, & has very high access speed. Thus, as the memory access speed increased, so does its relative cost. The overall goal of using a memory hierarchy is to obtain the highest possible average access speed while minimizing the total cost of the entire memory system.

Scanned by CamScanner

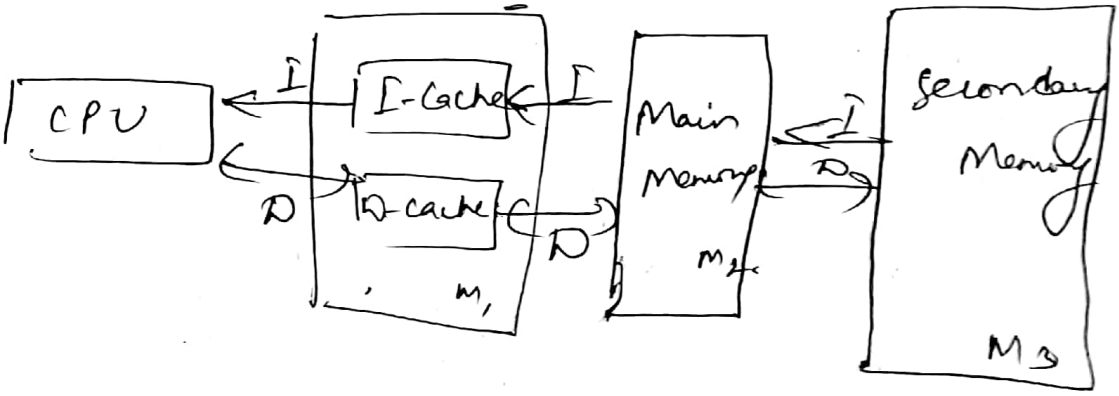
# Different levels of Memory hierarchy

M = Level
I = Instruct
D = Data

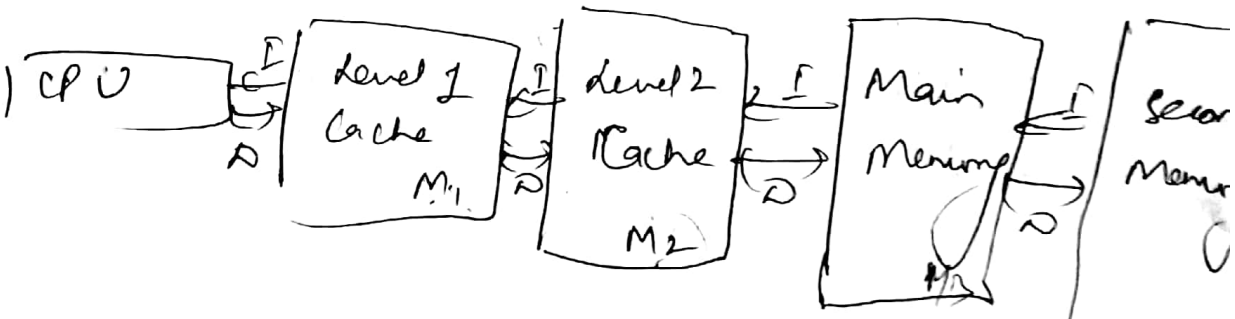
Ca: 2-Level




Three - Level.

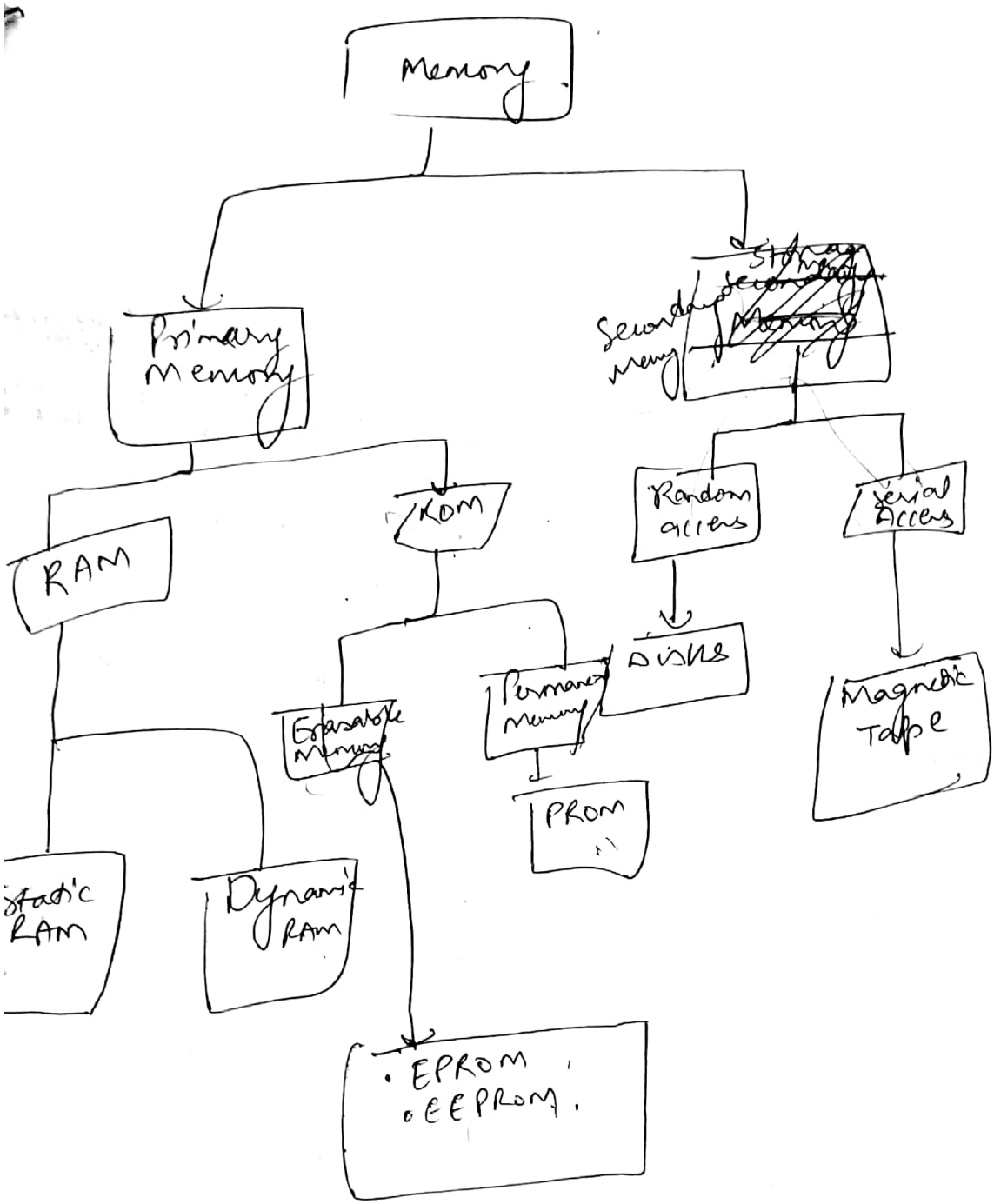



Four - Level.




All info. stored in $M_i$ at any time is also stored in $M_{i+1}$, but not vice-versa.

Scanned by CamScanner




Memory Hierarchy.

Scanned by CamScanner

# Locality of reference principle.

It is also known as principle of l.o.r
It is defined as the phenomenon of having same value or related storage locations being frequently accessed. Such locations are made local by their contents to cache for e.g. When a program is executed in CPU, it repeatedly refers to period of time so that CPU can learn when required set of instructions in memory that constitute the l
Every time a subroutine is called, its set of instructions are fetched from memory. from Mo
Hence.

Types.

(1) Temporal locality :- It refers to the reuse specific data and/or resources within relati small time durations. If at any pt. i time, a particular memory location is re then it is likely that same location will be referenced again in the near future so, adjacent references to the same memor location are considered. Thus, it becomes better to store a copy of it in some special memory storage, which can be accessed faster.

```
for(i=0;i<10;i++)
    sum = sum + a[i];
```

sum → Temporal locality.

Scanned by CamScanner

Spatial locality :- If a particular memory location is referenced at a particular time, then it is likely that nearby memory locations will be referenced in the near future. This is spatial locality. Special case → sequential locality → occurs when relevant data elements are arranged & accessed linearly. E.g. simple traversal of 1-D array.

Reasons for locality :-

(1.) Predictability :- Many of computer programs problems are decidable & hence the program behaves predictably. So, locality is considered to be one type of predictable behavior in computer system.

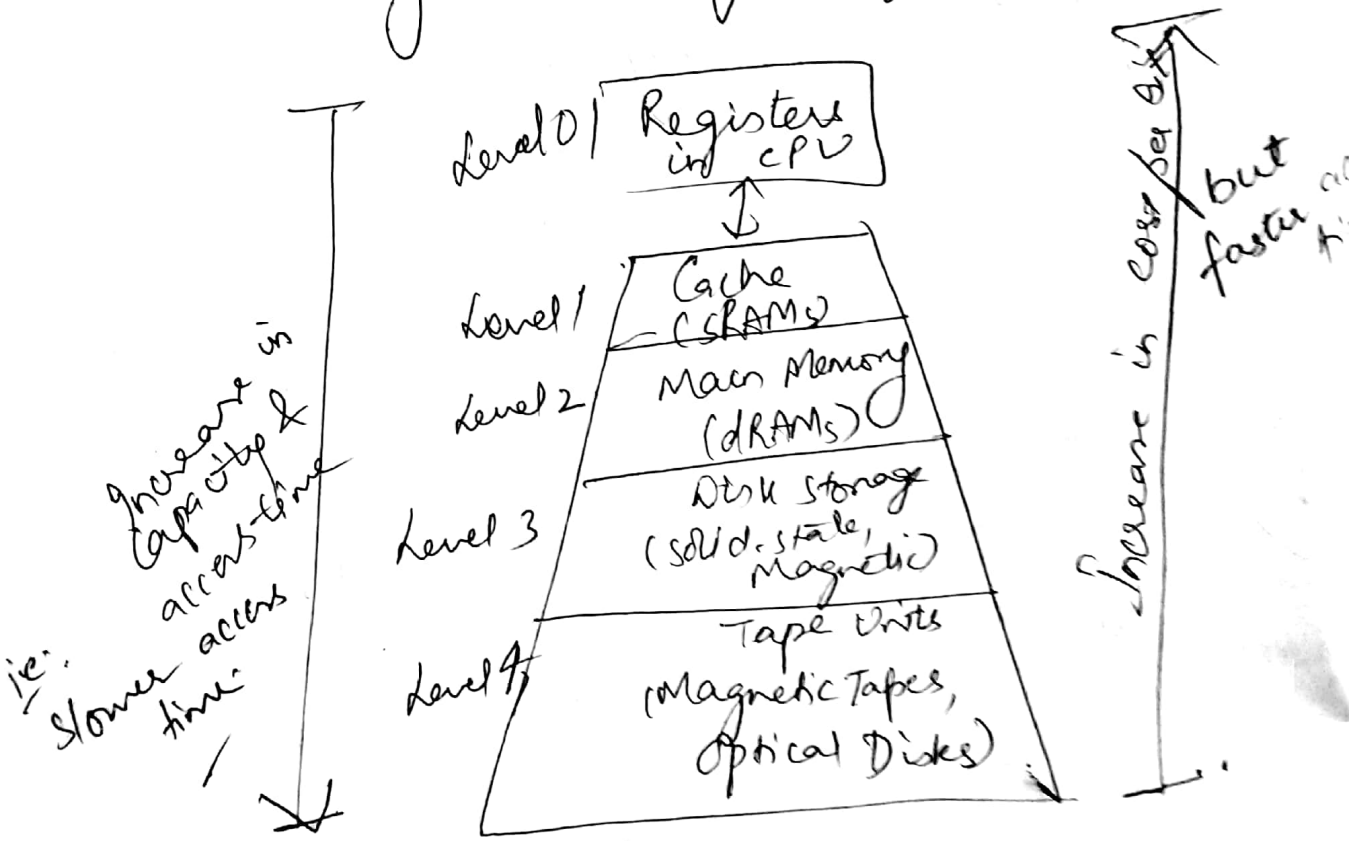
(2.) Structure of the program :- Generally, related data is stored in nearby locations in storage. This means that if a bit of processing is done, single item will be accessed more than once, thus leading to temporal locality of reference. E.g. in addition of five nos, sum is referenced repeatedly. Further, moving to next item implies reading next item, thus leading to spatial locality of reference.

Scanned by CamScanner

Applications of
It is used when -

1. We have a hierarchy in which accessing levels in orders of magnitude is more.
2. Data set is large.
3. Probability of accessing any of members of set is unequal.

Memory Hierarchy diagram:




Advantages of Memory Hierarchy

1. Decrease frequency of accesses to slo
2. Decrease cost/bit
3. Improve average access time.
4. Increase capacity.

Scanned by CamScanner

Morris Mano.

Cache Memory: → Ch: Memory organisation.

Page Replacement Policies + Gain + Thrash

Page fault: → It is an exception which is raised by memory management unit when a program ~~access~~ is to access a page that is present in virtual address space but not loaded physical or main memory.

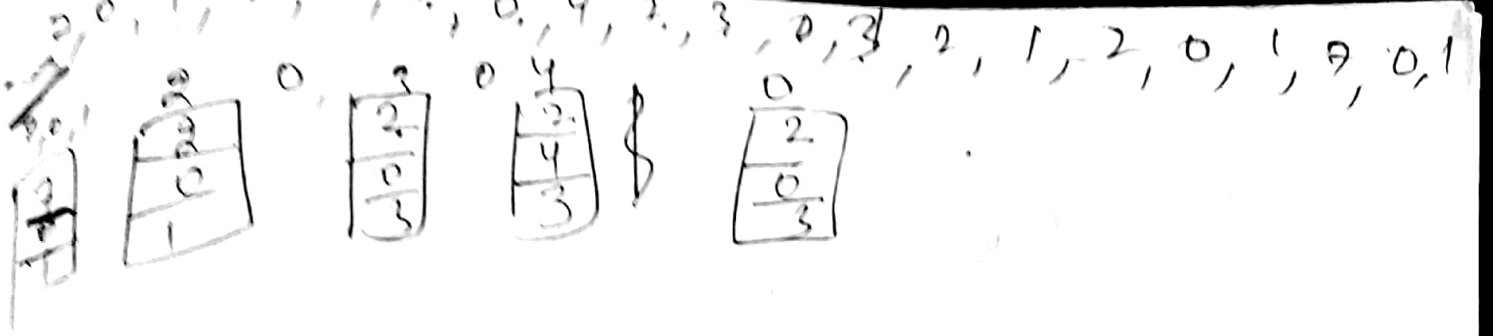
If the page to be removed has been modified then it is needed to be re-written to disk, otherwise if its copy is there in disk, then page can be discarded from main memory.

Page Table: → It is a ds used by virtual memory system to store mapping info b/w virtual addresses & physical addresses. Virtual addresses are those unique to accessing process. Physical addresses are those unique to h/w. i.e.

Why is page replacement algorithm req

→ To select victim page from main memory when page fault occurs, O.S has to remove a page from memory to make room for the page that has to be brought in. System performance increases if a page

Scanned by CamScanner

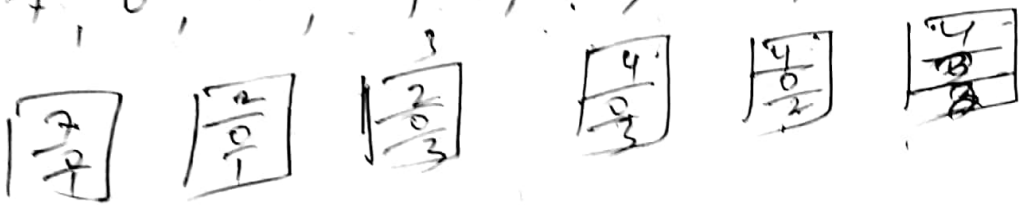


for seq → 1, 2, 3, — — .

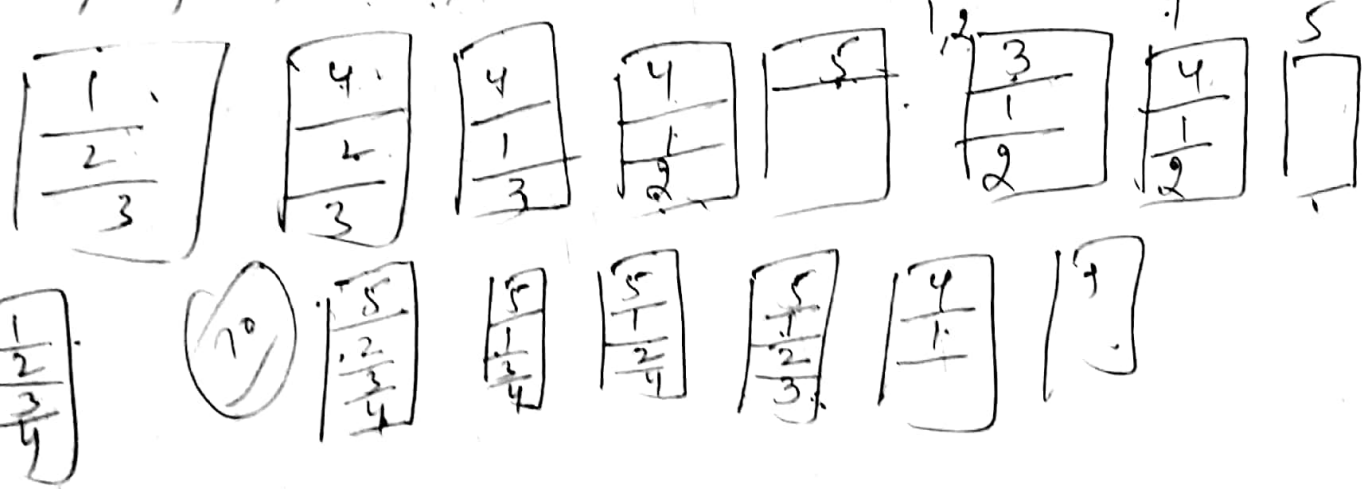
for 4 frames 6 page faults.

LRU page replacement. → It associates with each page the time of that page's last use. When a page must be replaced LRU chooses the page that has not been used for longest period of time.

7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2.



1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4, 5



(4)



(10)

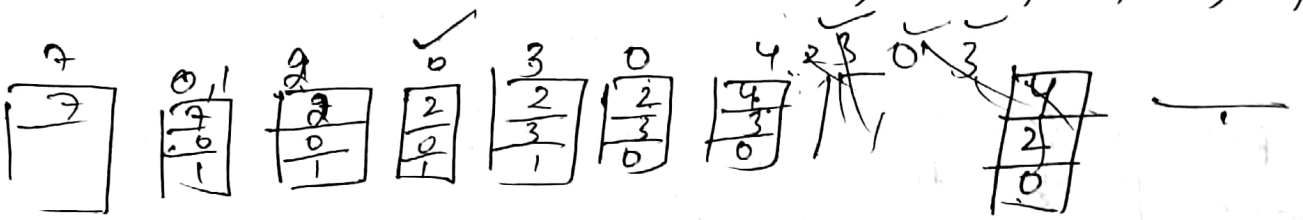
not heavily used is chosen.

Which page replacement algo is better? which is having lowest page fault rate?

E.g.

FIFO →

7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, 1, 2, 0,



We associate with each page - a either time slot or FIFO queue.

---

string = 1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4, 5

→ 3 frames (9 page faults)

→ 4 frames (10 page fault).

that means page-fault rate is increasing as no. of allocated frames is increased. So this is Belady's anomaly.

To remove it, Optimal page replacement policy.

Advantage: guarantee lowest page-fault rate for a fixed no. of frames.

Scanned by CamScanner

| Cache Type | Hit Ratio | Search Speed |
|---|---|---|
| Direct mapping | Good | Best |
| Associative mapping | Best | Moderate |
| N-way Set Associative mapping N>1 | very Good, Best as N increases | Good, worse as N increases |

→ If 4 delete some line if cache is full, in order to bring a new page. Here we do not know, whether in future, which is going to be used frequently.