Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA   Batch - 41
OS Assignment-1

Question - 1 : Explain Paging Technique with Example.

Paging is a memory management scheme that eliminates the need for contiguous allocation of physical memory. The process of retrieving processes in the form of pages from the secondary storage into the main memory is known as paging. The basic purpose of paging is to separate each procedure into pages. Additionally, frames will be used to split the main memory.This scheme permits the physical address space of a process to be non – contiguous.

Some important terminologies:

- Logical Address or Virtual Address (represented in bits): An address generated by the CPU
- Logical Address Space or Virtual Address Space(represented in words or bytes): The set of all logical addresses generated by a program
- Physical Address (represented in bits): An address actually available on memory unit
- Physical Address Space (represented in words or bytes): The set of all physical addresses corresponding to the logical addresses

Features of paging:

1. Mapping logical address to physical address.
2. Page size is equal to frame size.
3. Number of entries in a page table is equal to number of pages in logical address space.
4. The page table entry contains the frame number.
5. All the page table of the processes are placed in main memory.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA   Batch - 41
OS Assignment-1

Example:

- If Logical Address = 31 bit, then Logical Address Space = 2^{31} words = 2 G words (1 G = 2^{30})
- If Logical Address Space = 128 M words = $2^7 * 2^{20}$ words, then Logical Address = $\log_2 2^{27}$ = 27 bits
- If Physical Address = 22 bit, then Physical Address Space = 2^{22} words = 4 M words (1 M = 2^{20})
- If Physical Address Space = 16 M words = $2^4 * 2^{20}$ words, then Physical Address = $\log_2 2^{24}$ = 24 bits

The mapping from virtual to physical address is done by the memory management unit (MMU) which is a hardware device and this mapping is known as paging technique.

- The Physical Address Space is conceptually divided into a number of fixed-size blocks, called **frames**.
- The Logical address Space is also splitted into fixed-size blocks, called **pages**.
- Page Size = Frame Size

Let us consider an example:

- Physical Address = 12 bits, then Physical Address Space = 4 K words
- Logical Address = 13 bits, then Logical Address Space = 8 K words
- Page size = frame size = 1 K words (assumption)

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA   Batch - 41
OS Assignment-1

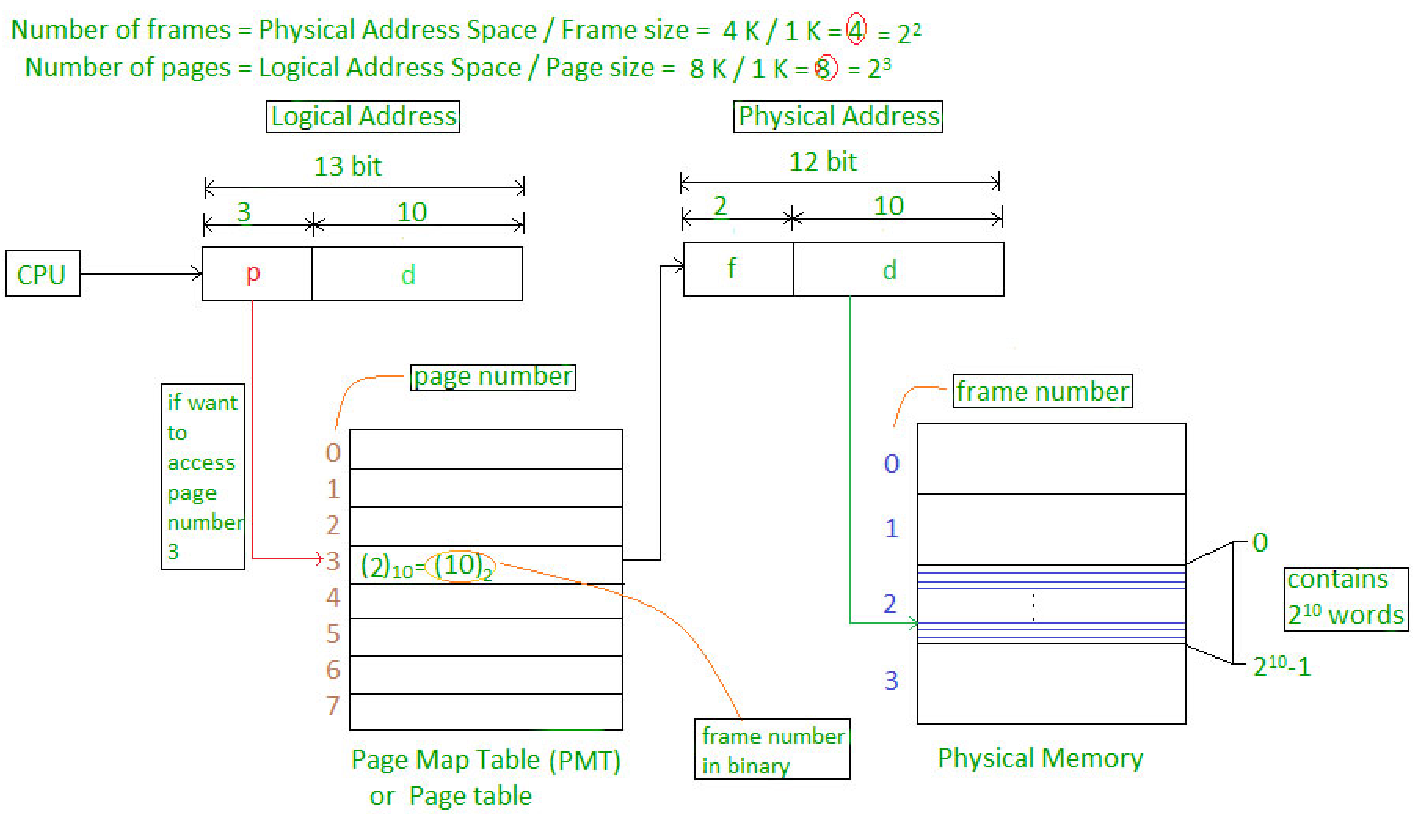
Address generated by CPU is divided into

- **Page number(p):** Number of bits required to represent the pages in Logical Address Space or Page number
- **Page offset(d):** Number of bits required to represent particular word in a page or page size of Logical Address Space or word number of a page or page offset.

Physical Address is divided into

- **Frame number(f):** Number of bits required to represent the frame of Physical Address Space or Frame number.
- **Frame offset(d):** Number of bits required to represent particular word in a frame or frame size of Physical Address Space or word number of a frame or frame offset.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA   Batch - 41
OS Assignment-1



Question - 2 : Explain Optimal Page replacement algorithm with example.

In operating systems, whenever a new page is referred and not present in memory, page fault occurs and Operating System replaces one of the existing pages with newly needed page. Different page replacement algorithms suggest different ways to decide which page to replace. The target for all algorithms is to reduce number of page faults. In this algorithm, OS replaces the page that will not be used for the longest period of time in future.

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA   Batch - 41
OS Assignment-1

Examples :

```
Input : Number of frames, fn = 3
        Reference String, pg[] = {7, 0, 1, 2,
               0, 3, 0, 4, 2, 3, 0, 3, 2, 1,
               2, 0, 1, 7, 0, 1};
Output : No. of hits = 11
         No. of misses = 9

Input : Number of frames, fn = 4
        Reference String, pg[] = {7, 0, 1, 2,
                  0, 3, 0, 4, 2, 3, 0, 3, 2};
Output : No. of hits = 7
         No. of misses = 6
```

The idea is simple, for every reference we do following :

1. If referred page is already present, increment hit count.
2. If not present, find if a page that is never referenced in future. If such a page exists, replace this page with new page. If no such page exists, find a page that is referenced farthest in future. Replace this page with new page.

X	7	0	1	2	0	3	0	4	2	3	0	3	2	3
f4				2	2	2	2	2	2	2	2	2	2	2
f3			1	1	1	1	1	4	4	4	4	4	4	4
f2		0	0	0	0	0	0	0	0	0	0	0	0	0
f1	7	7	7	7	7	3	3	3	3	3	3	3	3	3

Folds = 6 Hits = 8