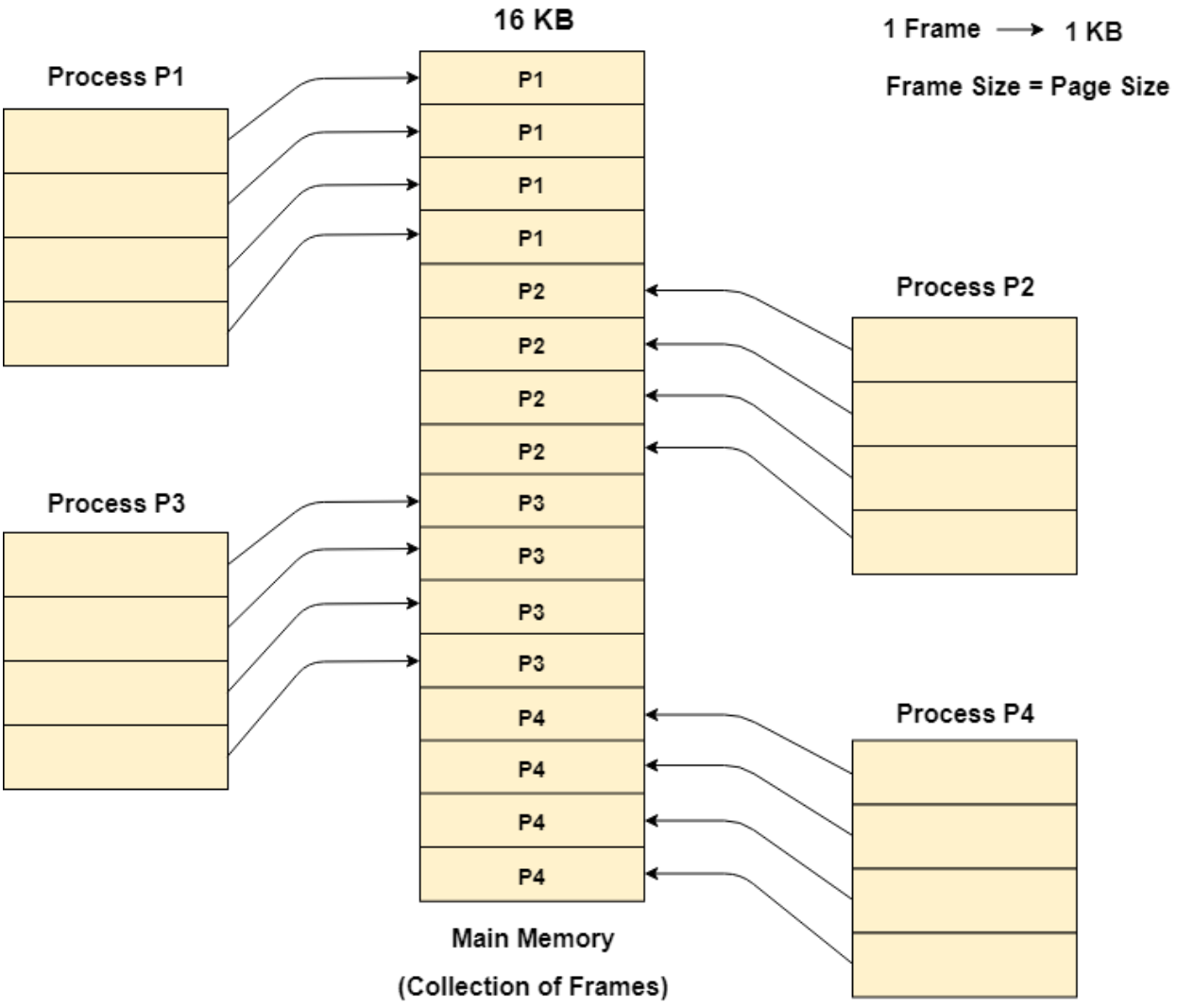
Paging (Non-contiguous)

In Operating Systems, Paging is a storage mechanism used to retrieve processes from the secondary storage into the main memory in the form of pages. The main idea behind the paging is to divide each process in the form of pages. The main memory will also be divided in the form of frames. One page of the process is to be stored in one of the frames of the memory. The pages can be stored at the different locations of the memory but the priority is always to find the contiguous frames or holes. Pages of the process are brought into the main memory only when they are required otherwise they reside in the secondary storage.

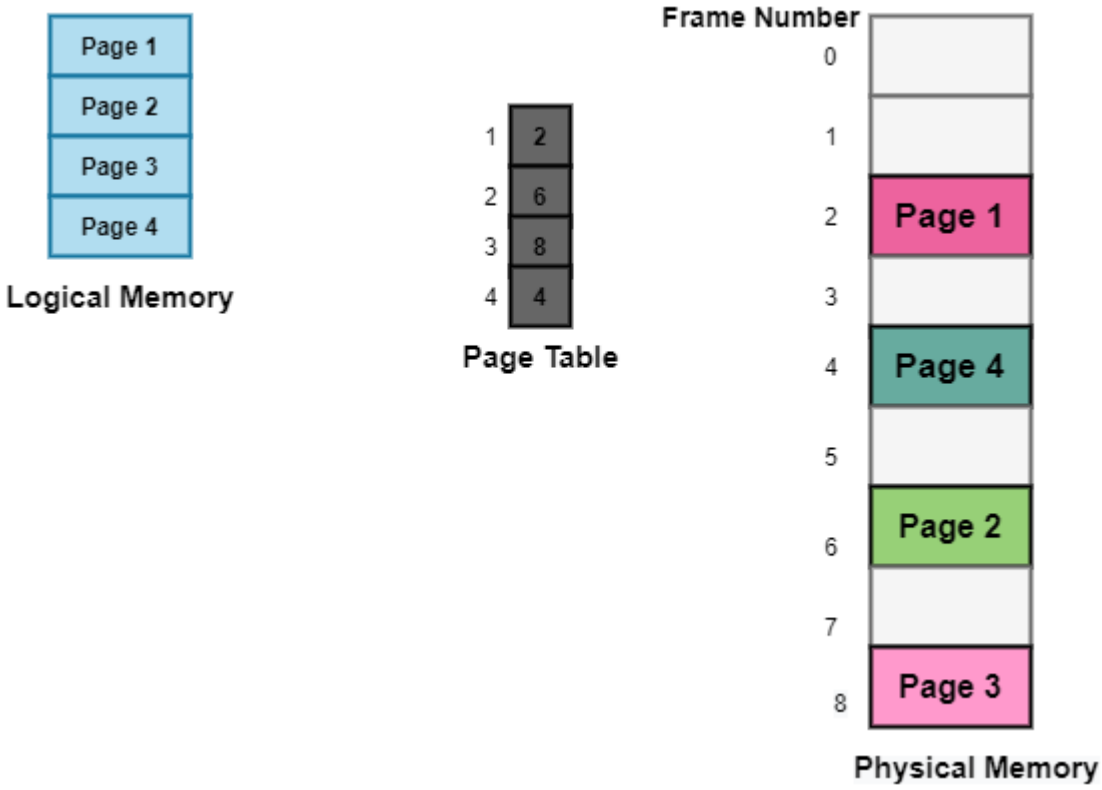
Different operating system defines different frame sizes. The sizes of each frame must be equal. Considering the fact that the pages are mapped to the frames in Paging**, page size needs to be as same as frame size**.



Paging

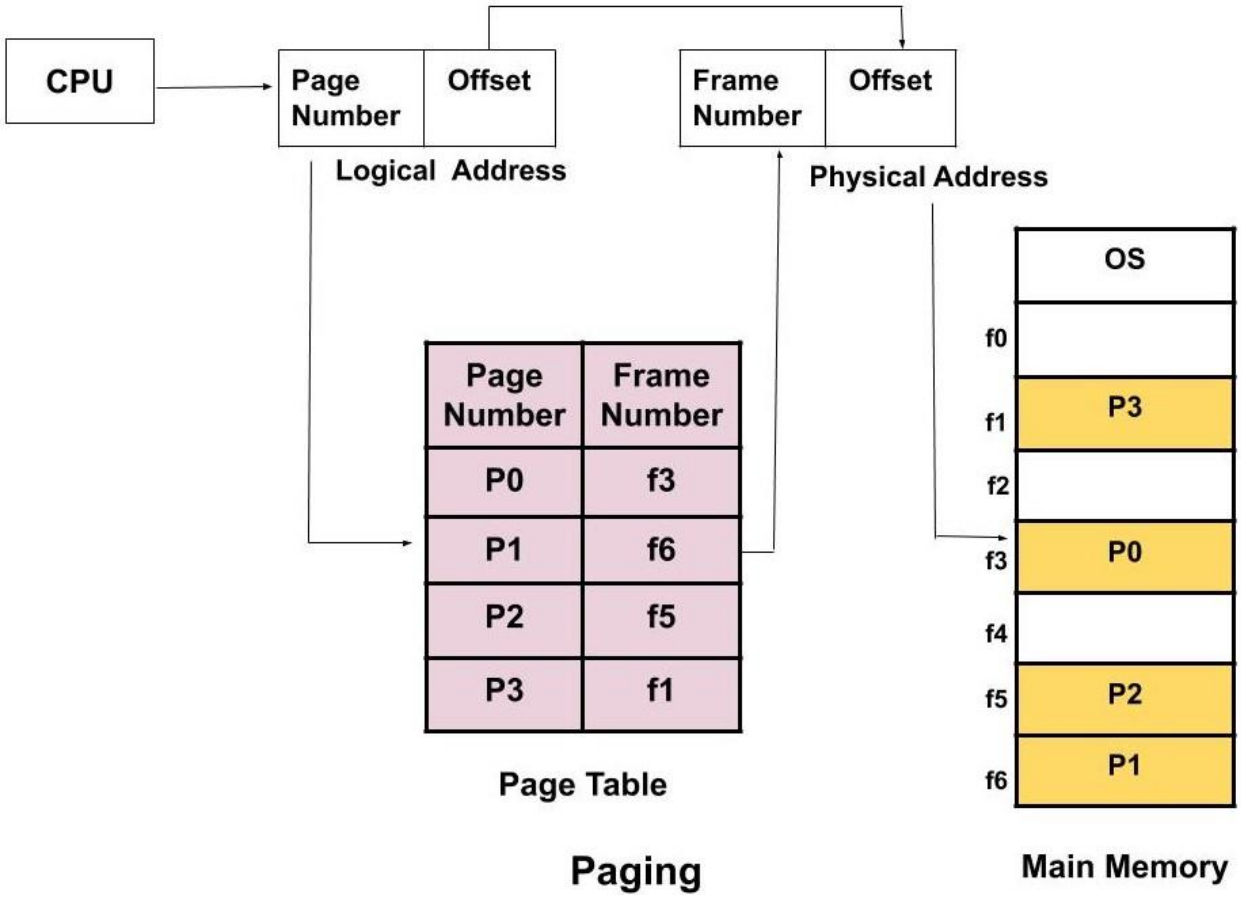
The data structure that is used by the virtual memory system in the operating system of a computer in order to store the mapping between physical and logical addresses is commonly known as **Page Table**. The logical address that is generated by the CPU is translated into the physical address with the help of the page table.

Thus page table mainly provides the corresponding frame number where that page is stored in the main memory.

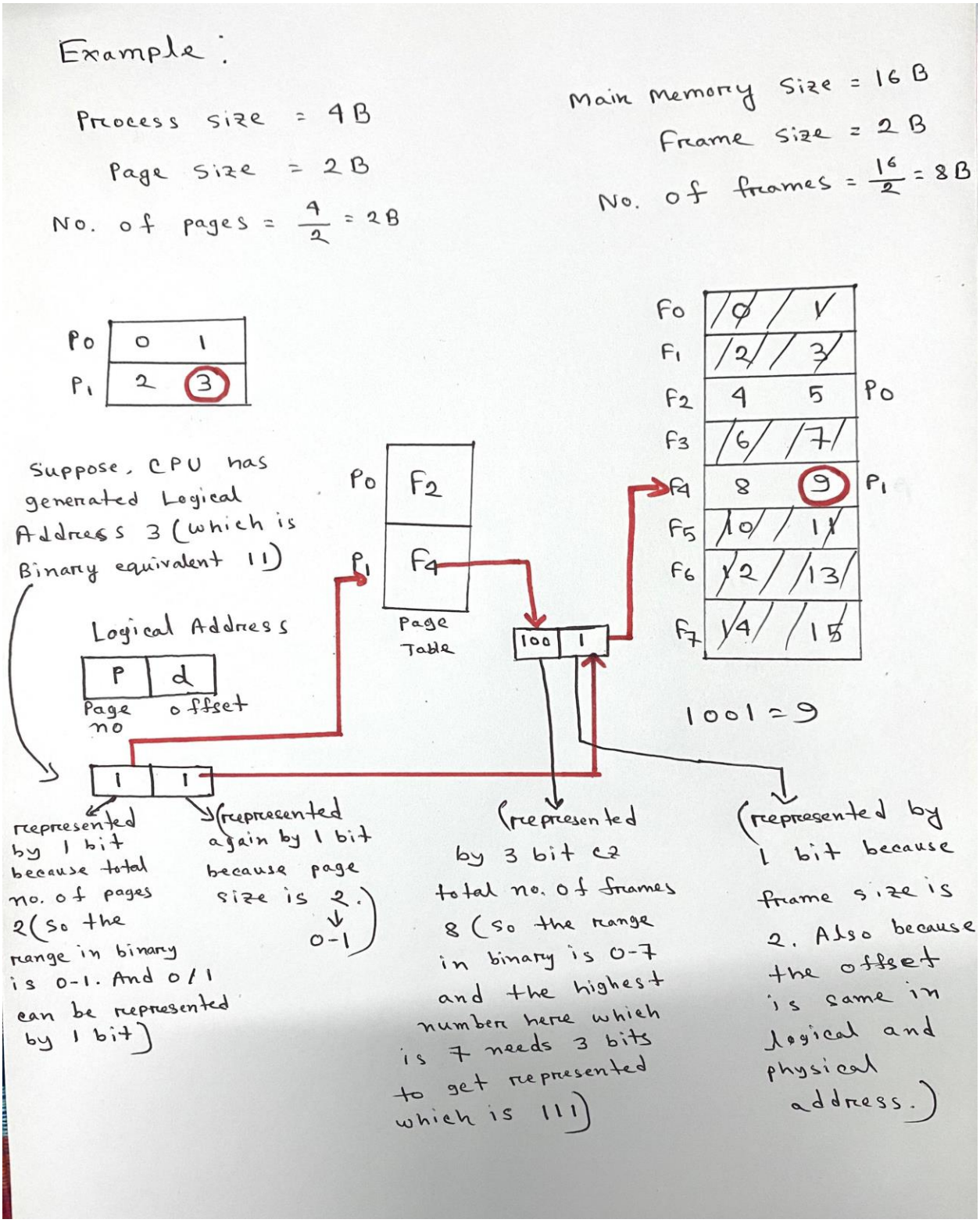


Logical Address to Physical Address

In operating systems, there is always a requirement of mapping from logical address to the physical address. CPU generates logical address for each page of the process. This contains two parts: page number and offset. The frame number of the desired page is determined by its entry in the page table. A physical address is generated which also contains two parts : frame number and offset. The Offset will be similar to the offset of the logical address therefore it will be copied from the logical address. The frame number and the offset from the physical address is mapped to the main memory in order to get the actual word address.



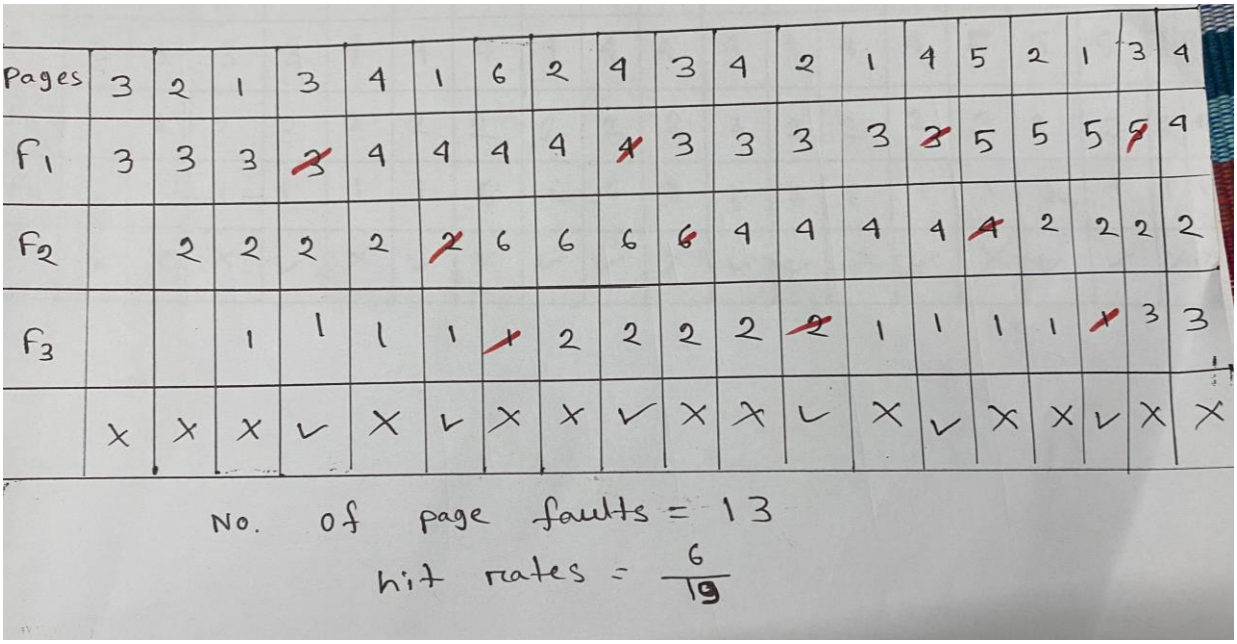
An example of converting logical address into physical address



Page Replacement Algorithms

1. FIFO
2. LRU
3. Optimal
4. MRU

FIFO(First In First Out) – When a page needs to be replaced, the page in the front of the queue is selected for removal.

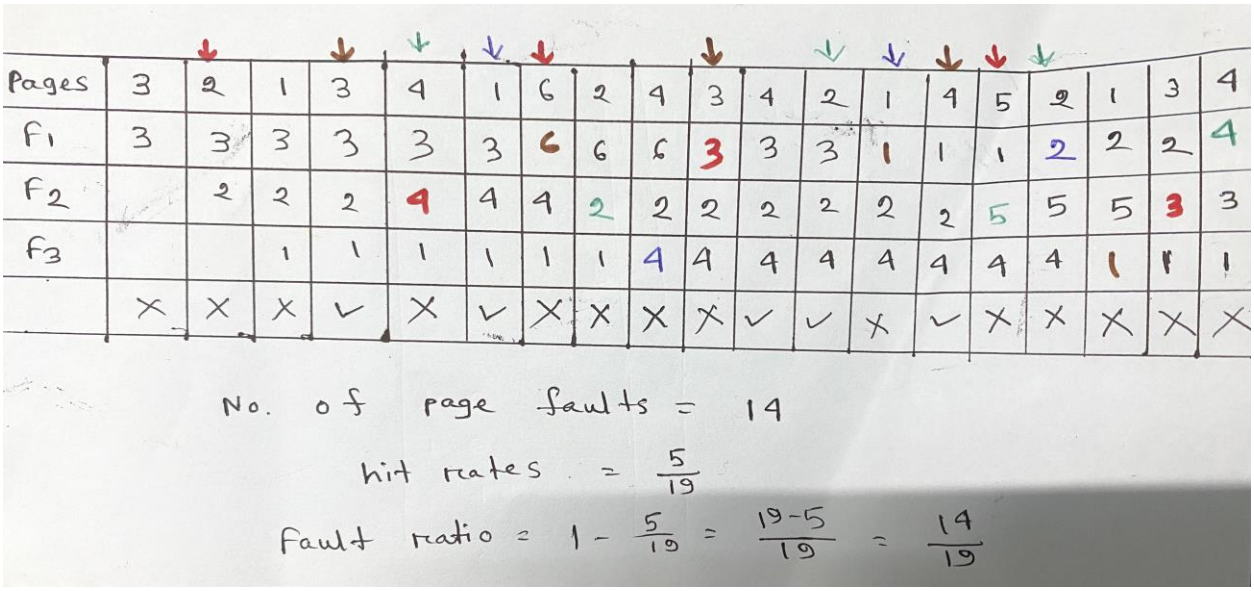


Pages	3	2	1	3	4	1	6	2	4	3	4	2	1	4	5	2	1	3	4
F_1	3	3	3	3	4	4	4	4	4	3	3	3	3	3	5	5	5	5	4
F_2		2	2	2	2	2	6	6	6	6	4	4	4	4	4	2	2	2	2
F_3			1	1	1	1	1	2	2	2	2	2	1	1	1	1	1	3	3
	X	X	X	✓	X	✓	X	X	✓	X	X	✓	X	✓	X	X	✓	X	X

No. of page faults = 13

hit rates = $\frac{6}{19}$

LRU(Least Recently Used) – Page will be replaced which is least recently used.



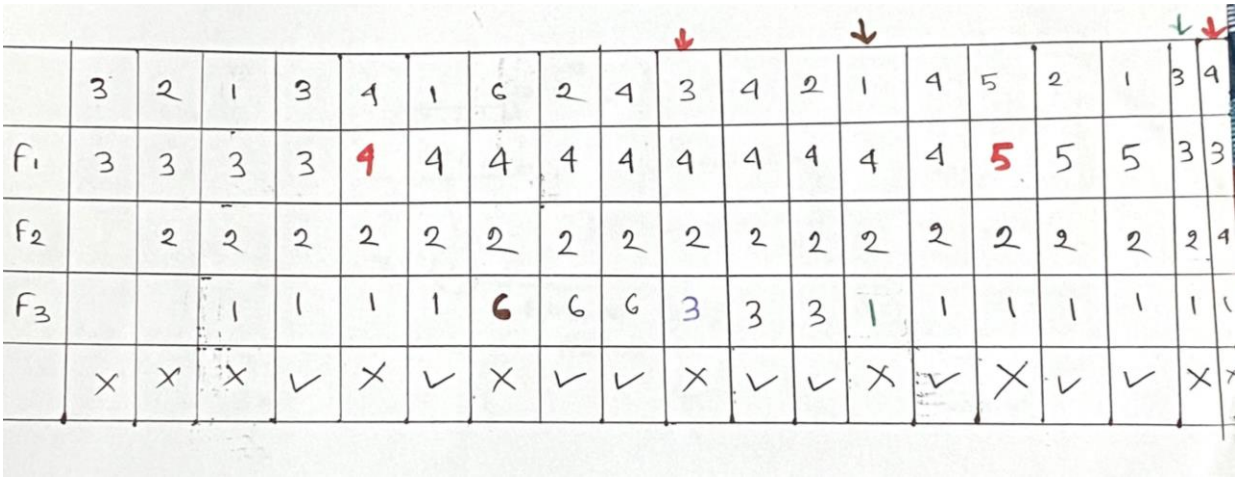
Pages	3	2	1	3	4	1	6	2	4	3	4	2	1	4	5	2	1	3	4
F_1	3	3	3	3	3	3	6	6	6	3	3	3	1	1	1	2	2	2	4
F_2		2	2	2	4	4	4	2	2	2	2	2	2	2	5	5	5	3	3
F_3			1	1	1	1	1	1	4	4	4	4	4	4	4	4	1	1	1
	X	X	X	✓	X	✓	X	X	X	X	✓	✓	X	✓	X	X	X	X	X

No. of page faults = 14

hit rates = $\frac{5}{19}$

Fault ratio = $1 - \frac{5}{19} = \frac{19-5}{19} = \frac{14}{19}$

Optimal – Replace the page which is not used in longest dimension of time in future.



	3	2	1	3	4	1	6	2	4	3	4	2	1	4	5	2	1	3	4
F_1	3	3	3	3	4	4	4	4	4	4	4	4	4	4	5	5	5	3	3
F_2		2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	4
F_3			1	1	1	1	6	6	6	3	3	3	1	1	1	1	1	1	
	X	X	X	✓	X	✓	X	✓	✓	X	✓	✓	X	✓	X	✓	✓	X	