PAGE REPLACEMENT ALGORITHMS

FIFO

Reference string

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

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

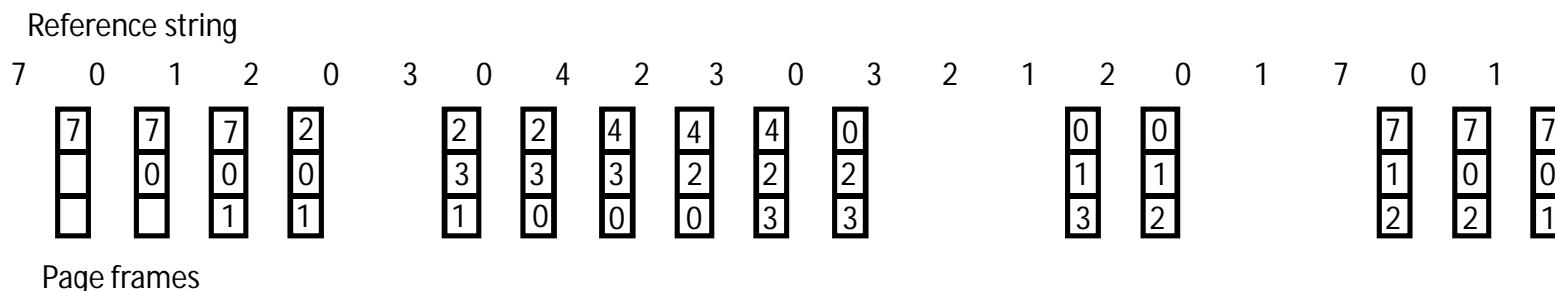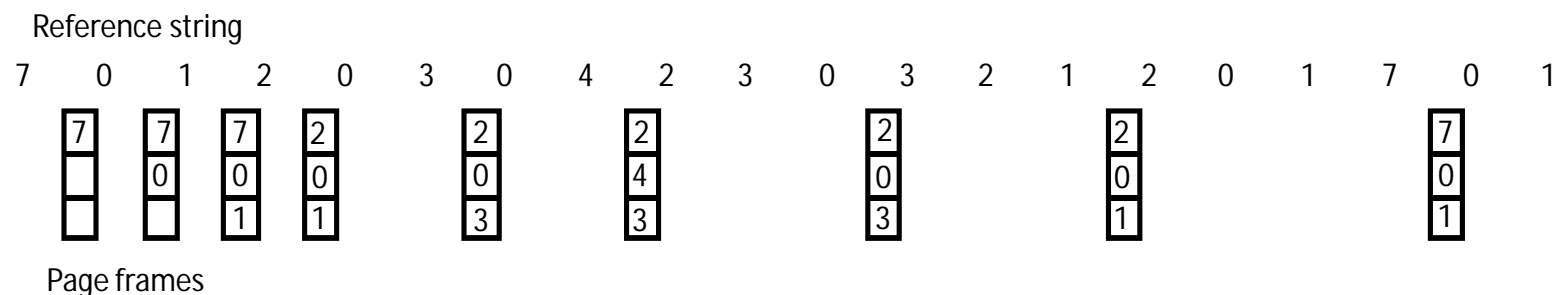Page frames

FIFO algorithm selects the page that has been in memory the longest time
Using a queue - every time a page is loaded, its
          -          identification is inserted in the queue
Easy to implement
May result in a frequent page fault

Optimal Replacement (OPT) - Lowest page fault rate of all algorithms

> Replace that page which will not be used for the longest period of time

Reference string

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

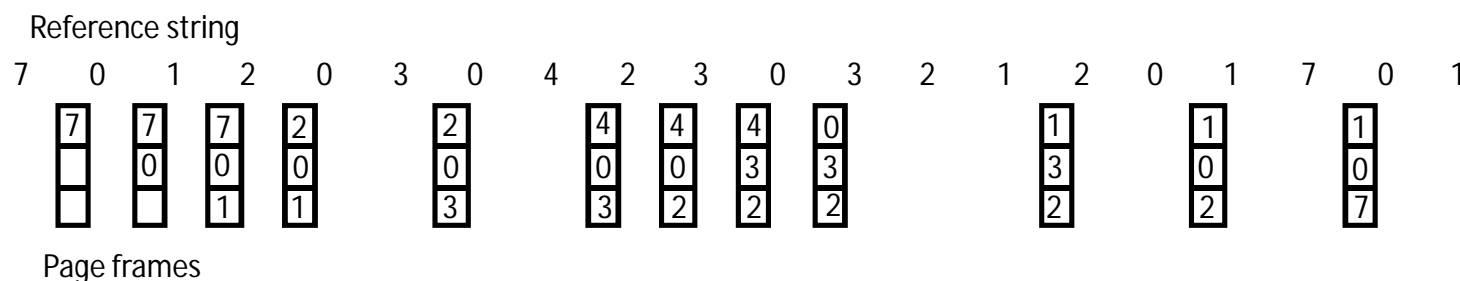| 7 | 7 | 7 | 2 | | 2 | | 2 | | 2 | | 2 | | 7 |
| | 0 | 0 | 0 | | 0 | | 4 | | 0 | | 0 | | 0 |
| | | 1 | 1 | | 3 | | 3 | | 3 | | 1 | | 1 |

Page frames

PAGE REPLACEMENT ALGORITHMS

<u>LRU</u>

    - OPT is difficult to implement since it requires future knowledge

    - LRU uses the recent past as an approximation of near future.

> Replace that page which has not been
> used for the longest period of time

Reference string

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

```
7   7   7   2       2       4   4   4   0           1       1       1
    0   0   0       0       0   0   3   3           3       0       0
        1   1       3       3   2   2   2           2       2       7
```

Page frames

- LRU may require substantial hardware assistance
- The problem is to determine an order for the frames
  defined by the time of last use