PRACTICAL 8

AIM: Write a program that implements the FIFO page-replacement algorithm.

CODE:

```
import java.util.HashSet;
import java.util.LinkedList;
import java.util.Queue;
class test
{
    static int pageFaults(int pages[], int n, int capacity)
    {
        HashSet<Integer>s=new HashSet<>(capacity);
        Queue<Integer>indexes=new LinkedList<>();
        int page_faults=0;
        for (int i=0;i<n;i++)
        {
            if(s.size()<capacity)
            {
                if(!s.contains(pages[i]))
                {
                    s.add(pages[i]);
                    page_faults++;
                    indexes.add(pages[i]);
                }
            }
            else
```

```
                {
                    if(!s.contains(pages[i]))
                    {
                        int val=indexes.peek();
                        indexes.poll();
                        s.remove(val);
                        s.add(pages[i]);
                        indexes.add(pages[i]);
                        page_faults++;
                    }
                }
            }
            return page_faults;
}
public static void main(String args[])
{
    int pages[]={7,0,1,2,0,3,0,4,2,3,0,3,2};
    int capacity=4;
    System.out.println(pageFaults(pages,pages.length,capacity));
   }
}
```

OUTPUT:

```
[Running] cd "c:\Users\Atharva\Dropbox (Old)\PC\Documents\Practicals\Principles of Operating Systems\Practical 8\" && javac test.java && java test
7
```