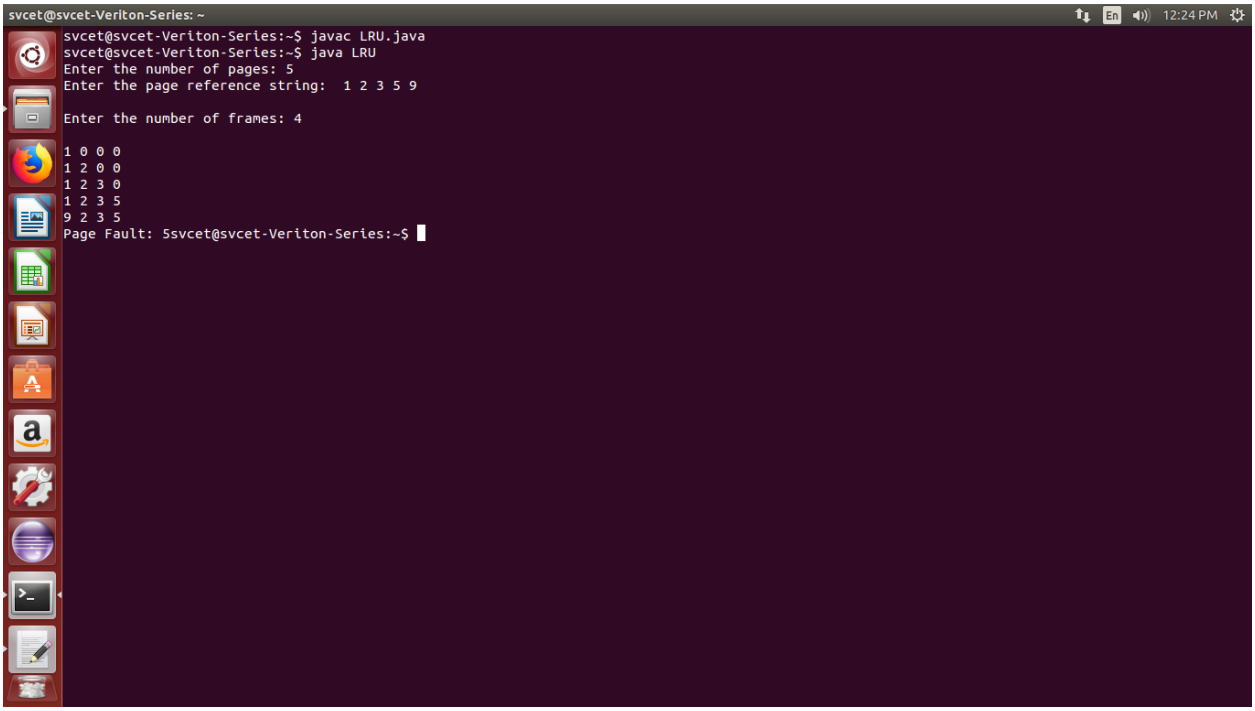
INPUT CODE

```
import java.util.Scanner;
public class LRU{
   public static int min(int counter[],int nFrames)
{
   int minimum = counter[0];
   int pos = 0;
   for(int i=0;i<nFrames;i++) { if(minimum > counter[i])
        pos = i;
   }
 return pos;
}
   public static void main(String[] args) {
      // TODO code application logic here
      Scanner s = new Scanner(System.in);
   int n,recent = 0,pageFault = 0,nFrames;

   System.out.print("Enter the number of pages: ");
   n = s.nextInt();
   int pageString[] = new int[n];
   System.out.print("Enter the page reference string: ");
   for(int i=0;i<n;i++)
      pageString[i]=s.nextInt();

    System.out.print("\nEnter the number of frames: ");
    nFrames = s.nextInt();
    int frames[] = new int[nFrames];
    int counter[] = new int[nFrames];

    for(int i=0;i<nFrames;i++)
```

```
{       frames[i] = 0;
   counter[i] = 0;//here 0 referes an empty space in frame
}

for(int i=0;i<n;i++)
{int flag =0;
   for(int j=0;j<nFrames;j++)
   {
    if(frames[j] == pageString[i])
      {flag=1;
       counter[j] = recent++; //counter holds which frame is recently used,
                              //recently used page in frame will have a bigger number
                              //and least recently used page in frame will have a lower number
      break;
      }
   }
   if(flag == 0)
   {
      for(int j=0;j<nFrames;j++)
      {if(frames[j] == 0)
         {   frames[j] = pageString[i];
            counter[j] = recent++;
            flag=1;
            pageFault++;
            break;
         }
      }
   }

   if(flag == 0){
      int PositionToreplace = min(counter,nFrames);
```

```
            frames[PositionToreplace] = pageString[i];
            counter[PositionToreplace] = recent++;
            pageFault++;
        }

        //print frames
        System.out.println();
      for(int j=0;j<nFrames;j++)
      {
          System.out.print(frames[j]+" ");
      }
    }
     System.out.print("\nPage Fault: "+pageFault);
    }
}
```

OUTPUT


svcet@svcet-Veriton-Series: ~
12:24 PM
svcet@svcet-Veriton-Series:~$ javac LRU.java
svcet@svcet-Veriton-Series:~$ java LRU
Enter the number of pages: 5
Enter the page reference string: 1 2 3 5 9
Enter the number of frames: 4
1 0 0 0
1 2 0 0
1 2 3 0
1 2 3 5
9 2 3 5
Page Fault: 5svcet@svcet-Veriton-Series:~$

INPUT:

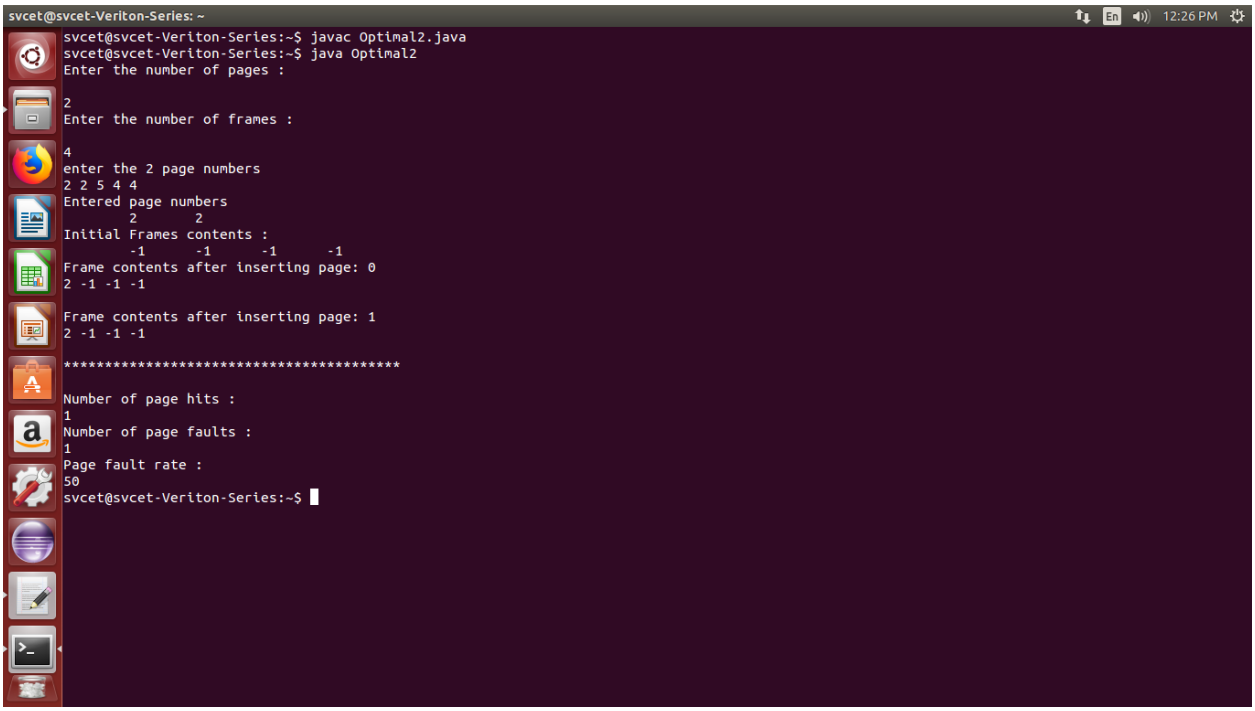
```
import java.io.*;
import java.util.*;
public class Optimal2
{
public static void main(String args[])
{
Scanner sc = new Scanner(System.in);
int n,f,pageHit = 0,pageFault = 0,pointer = 0;
boolean isFull = false;
System.out.println("Enter the number of pages :\n ");
n = sc.nextInt();
System.out.println("Enter the number of frames :\n");
f = sc.nextInt();
int frame[] = new int[f];
int pages[] = new int[n];
System.out.println("enter the"+" "+n+" " +"page numbers");
for(int i=0;i<n;i++)
{
pages[i] = sc.nextInt();
}
System.out.println("Entered page numbers");
for(int i=0;i<n;i++)
{
System.out.print("\t"+pages[i]);
}
for(int i=0;i<f;i++)
{
frame[i] = -1;
}
```

```
System.out.println("\nInitial Frames contents :");
for(int i=0;i<f;i++)
{
System.out.print("\t"+frame[i]);
}
for(int i=0;i<n;i++) //start searching empty frames
{
int search = -1;
for(int j=0;j<f;j++)
{
if(frame[j] == pages[i])
{
search = j;
pageHit++;
break;
}
}
if(search == -1) //if no page found
{
if(isFull)
{
int index[] = new int[f];
boolean index_flag[] = new boolean[f];
for(int j = i+1 ;j<n;j++)
{
for(int k = 0;k < f;k++)
{
if((pages[j] == frame[k]) && (index_flag[k] == false))
{
index[k] = j;
index_flag[k] = true;
```

```
break;
}
}
}
int max = index[0];
pointer = 0;
if(max == 0)
{
max = 200;
}
for(int j=0;j<f;j++)
{
if(index[j] == 0)
{
index[j] = 200;
}
if(index[j] > max)
{
max = index[j];
pointer = j;
}
}//end for
} //end of isFull
frame[pointer] = pages[i];
pageFault++;
if(!isFull)
{
pointer++;
if(pointer == f)
{
pointer = 0;
```

```
isFull = true;
}
}
} //end if
System.out.print("\nFrame contents after inserting page:"+" "+i+"\n");
for(int k=0;k<f;k++)
{
System.out.print(frame[k]+" ");
}
System.out.println();
} //end for
System.out.println("\n**************************************");
System.out.println("\nNumber of page hits :\n" + pageHit);
System.out.println("Number of page faults :\n" + pageFault);
int pageFR = (pageFault*100) / n;
System.out.println("Page fault rate :\n" + pageFR);
}
}
```

OUTPUT


svcet@svcet-Veriton-Series: ~
En
12:26 PM
svcet@svcet-Veriton-Series:~$ javac Optimal2.java
svcet@svcet-Veriton-Series:~$ java Optimal2
Enter the number of pages :
2
Enter the number of frames :
4
enter the 2 page numbers
2 2 5 4 4
Entered page numbers
2 2
Initial Frames contents :
-1 -1 -1 -1
Frame contents after inserting page: 0
2 -1 -1 -1
Frame contents after inserting page: 1
2 -1 -1 -1
****
Number of page hits :
1
Number of page faults :
1
Page fault rate :
50
svcet@svcet-Veriton-Series:~$