```
(base) student@composnlab-93:~/Downloads/A43$ javac FIFO.java
(base) student@composnlab-93:~/Downloads/A43$ java FIFO
Please enter the number of Frames:
3
Please enter the length of the Reference string:
15
Please enter the reference string:
7
0
1
2
0
3
0
4
2
3
1
3
0
2
0

  7   7   7   2   2   2   2   4   4   4   1   1   1   1   1
 -1   0   0   0   0   3   3   3   2   2   2   2   0   0   0
 -1  -1   1   1   1   1   0   0   0   3   3   3   3   2   2
The number of Hits: 3
Hit Ratio: 0.2
The number of Faults: 12
(base) student@composnlab-93:~/Downloads/A43$
```

```java
import java.io.*;
public class FIFO {
public static void main(String[] args) throws IOException
{
BufferedReader br = new BufferedReader(new
InputStreamReader(System.in));
int frames, pointer = 0, hit = 0, fault = 0,ref_len;
int buffer[];
int reference[];
int mem_layout[][];
System.out.println("Please enter the number of Frames: ");
frames = Integer.parseInt(br.readLine());
System.out.println("Please enter the length of the Reference string: ");
ref_len = Integer.parseInt(br.readLine());
reference = new int[ref_len];
mem_layout = new int[ref_len][frames];
buffer = new int[frames];
for(int j = 0; j < frames; j++)
buffer[j] = -1;
System.out.println("Please enter the reference string: ");
for(int i = 0; i < ref_len; i++)
{
reference[i] = Integer.parseInt(br.readLine());
}
System.out.println();
for(int i = 0; i < ref_len; i++)
{
int search = -1;
for(int j = 0; j < frames; j++)
{
if(buffer[j] == reference[i])
{
search = j;
hit++;
break;
}
}
if(search == -1)
{
buffer[pointer] = reference[i];
fault++;
pointer++;
if(pointer == frames)
```

```
fault++;
pointer++;
if(pointer == frames)
pointer = 0;
}
for(int j = 0; j < frames; j++)
mem_layout[i][j] = buffer[j];
}
for(int i = 0; i < frames; i++)
{
for(int j = 0; j < ref_len; j++)
System.out.printf("%3d ",mem_layout[j][i]);
System.out.println();
}
System.out.println("The number of Hits: " + hit);
System.out.println("Hit Ratio: " + (float)((float)hit/ref_len));
System.out.println("The number of Faults: " + fault);
}
```

```
(base) student@composnlab-93:~/Downloads/A43$ javac OptimalReplacement.java
(base) student@composnlab-93:~/Downloads/A43$ java OptimalReplacement
Please enter the number of Frames:
4
Please enter the length of the Reference string:
20
Please enter the 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   7   7   3   3   3   3   3   3   3   3   1   1   1   1   1   1   1
 -1   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
 -1  -1   1   1   1   1   1   4   4   4   4   4   4   4   4   4   4   7   7   7
 -1  -1  -1   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
The number of Hits: 12
Hit Ratio: 0.6
The number of Faults: 8
(base) student@composnlab-93:~/Downloads/A43$
```

```java
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
public class OptimalReplacement {
public static void main(String[] args) throws IOException
{
BufferedReader br = new BufferedReader(new
InputStreamReader(System.in));
int frames, pointer = 0, hit = 0, fault = 0,ref_len;
boolean isFull = false;
int buffer[];
int reference[];
int mem_layout[][];
System.out.println("Please enter the number of Frames: ");
frames = Integer.parseInt(br.readLine());
System.out.println("Please enter the length of the Reference string: ");
ref_len = Integer.parseInt(br.readLine());
reference = new int[ref_len];
mem_layout = new int[ref_len][frames];
buffer = new int[frames];
for(int j = 0; j < frames; j++)
buffer[j] = -1;
System.out.println("Please enter the reference string: ");
for(int i = 0; i < ref_len; i++)
{
reference[i] = Integer.parseInt(br.readLine());
}
System.out.println();
for(int i = 0; i < ref_len; i++)
{
int search = -1;
for(int j = 0; j < frames; j++)
{
if(buffer[j] == reference[i])
{
search = j;
hit++;
break;
}
}
if(search == -1)
{
if(isFull)
```

```
43 if(isFull)
44 {
45 int index[] = new int[frames];
46 boolean index_flag[] = new boolean[frames];
47 for(int j = i + 1; j < ref_len; j++)
48 {
49 for(int k = 0; k < frames; k++)
50 {
51 if((reference[j] == buffer[k]) && (index_flag[k] == false))
52 {
53 index[k] = j;
54 index_flag[k] = true;
55 break;
56 }
57 }
58 }
59 int max = index[0];
60 pointer = 0;
61 if(max == 0)
62 max = 200;
63 for(int j = 0; j < frames; j++)
64 {
65 if(index[j] == 0)
66 index[j] = 200;
67 if(index[j] > max)
68 {
69 max = index[j];
70 pointer = j;
71 }
72 }
73 }
74 buffer[pointer] = reference[i];
75 fault++;
76 if(!isFull))
```

```
if(!isFull)
{
pointer++;
if(pointer == frames)
{
pointer = 0;
isFull = true;
}
}
}
for(int j = 0; j < frames; j++)
mem_layout[i][j] = buffer[j];
}
for(int i = 0; i < frames; i++)
{
for(int j = 0; j < ref_len; j++)
System.out.printf("%3d ",mem_layout[j][i]);
System.out.println();
}
System.out.println("The number of Hits: " + hit);
System.out.println("Hit Ratio: " + (float)((float)hit/ref_len));
System.out.println("The number of Faults: " + fault);
}
```

```
(base) student@composnlab-93:~/Downloads/A43$ gedit LRU.java
Failed to register: GDBus.Error:org.freedesktop.DBus.Error.NoReply: Message recipient disconnected from message bus without replying
[2]+  Killed                  gedit OptimalReplacement.java
(base) student@composnlab-93:~/Downloads/A43$ gedit LRU.java
^Z
[1]+  Stopped                 gedit LRU.java
(base) student@composnlab-93:~/Downloads/A43$ javac LRU.java
(base) student@composnlab-93:~/Downloads/A43$ java LRU
Please enter the number of Frames:
4
Please enter the length of the Reference string:
20
Please enter the 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   7   7   3   3   3   3   3   3   3   3   3   3   3   7   7   7
 -1   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
 -1  -1   1   1   1   1   1   4   4   4   4   4   4   1   1   1   1   1   1
 -1  -1  -1   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2   2
The number of Hits: 12
Hit Ratio: 0.6
The number of Faults: 8
(base) student@composnlab-93:~/Downloads/A43$
```

```java
import java.io.*;
import java.util.*;
public class LRU {
public static void main(String[] args) throws IOException
{
BufferedReader br = new BufferedReader(new
InputStreamReader(System.in));
int frames,pointer = 0, hit = 0, fault = 0,ref_len;
Boolean isFull = false;
int buffer[];
ArrayList<Integer> stack = new ArrayList<Integer>();
int reference[];
int mem_layout[][];
System.out.println("Please enter the number of Frames: ");
frames = Integer.parseInt(br.readLine());
System.out.println("Please enter the length of the Reference string: ");
ref_len = Integer.parseInt(br.readLine());
reference = new int[ref_len];
mem_layout = new int[ref_len][frames];
buffer = new int[frames];
for(int j = 0; j < frames; j++)
buffer[j] = -1;
System.out.println("Please enter the reference string: ");
for(int i = 0; i < ref_len; i++)
{
reference[i] = Integer.parseInt(br.readLine());
}
System.out.println();
for(int i = 0; i < ref_len; i++)
{
if(stack.contains(reference[i]))
{
stack.remove(stack.indexOf(reference[i]));
}
stack.add(reference[i]);
int search = -1;
for(int j = 0; j < frames; j++)
{
if(buffer[j] == reference[i])
{
search = j;
hit++;
break;
```

```
break;
}
}
if(search == -1)
{
if(isFull)
{
int min_loc = ref_len;
for(int j = 0; j < frames; j++)
{
if(stack.contains(buffer[j]))
{
int temp = stack.indexOf(buffer[j]);
if(temp < min_loc)
{
min_loc = temp;
pointer = j;
}
}
}
}
buffer[pointer] = reference[i];
fault++;
pointer++;
if(pointer == frames)
{
pointer = 0;
isFull = true;
}
}
for(int j = 0; j < frames; j++)
mem_layout[i][j] = buffer[j];
}
for(int i = 0; i < frames; i++)
{
for(int j = 0; j < ref_len; j++)
System.out.printf("%3d ",mem_layout[j][i]);
System.out.println();
}
System.out.println("The number of Hits: " + hit);
System.out.println("Hit Ratio: " + (float)((float)hit/ref_len));
System.out.println("The number of Faults: " + fault);
}
```