```java
package A4;

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.print("Please enter the number of Frames: ");
        frames = Integer.parseInt(br.readLine());

        System.out.print("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 values: ");
        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;
                }
            }
```

```
46
47              if (search == -1) { // Page Fault
48                  if (isFull) {
49                      int index[] = new int[frames];
50                      boolean index_flag[] = new boolean[frames];
51
52                      for (int j = i + 1; j < ref_len; j++) {
53                          for (int k = 0; k < frames; k++) {
54                              if ((reference[j] == buffer[k]) && (index_flag[k] ==
    false)) {
55                                  index[k] = j;
56                                  index_flag[k] = true;
57                                  break;
58                              }
59                          }
60                      }
61
62                      int max = index[0];
63                      pointer = 0;
64                      if (max == 0)
65                          max = 200;
66
67                      for (int j = 0; j < frames; j++) {
68                          if (index[j] == 0)
69                              index[j] = 200;
70                          if (index[j] > max) {
71                              max = index[j];
72                              pointer = j;
73                          }
74                      }
75                  }
76
77                  buffer[pointer] = reference[i];
78                  fault++;
79
80                  if (!isFull) {
81                      pointer++;
82                      if (pointer == frames) {
83                          pointer = 0;
84                          isFull = true;
85                      }
86                  }
87              }
88
89              for (int j = 0; j < frames; j++) {
90                  mem_layout[i][j] = buffer[j];
91              }
```

```
92          }
93
94          System.out.println("\nPage Replacement Table:");
95          for (int i = 0; i < frames; i++) {
96              for (int j = 0; j < ref_len; j++) {
97                  System.out.printf("%3d ", mem_layout[j][i]);
98              }
99              System.out.println();
100         }
101
102         System.out.println("\nThe number of Hits: " + hit);
103         System.out.println("Hit Ratio: " + (float) hit / ref_len);
104         System.out.println("The number of Faults: " + fault);
105     }
106 }
107
```

Output: