```java
package A4;

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)
                    pointer = 0;
            }

            for (int j = 0; j < frames; j++)
                mem_layout[i][j] = buffer[j];
        }

        System.out.println("Memory Layout:");
        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) hit / ref_len);
        System.out.println("The number of Faults: " + fault);
    }
}
```

Output :