SPOS Practical\Page Placement Strategy\LeastRecentlyUsed.java

```java
import java.util.*;

public class LeastRecentlyUsed {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        ArrayList<Integer> arr = new ArrayList<>();
        int noofpages, capacity, hit = 0, fault = 0, index = 0;
        boolean isFull = false;
        double hitRatio, faultRatio;
        System.out.print("Enter the number of pages you want to enter: ");
        noofpages = sc.nextInt();
        int pages[] = new int[noofpages];
        for (int i = 0; i < noofpages; i++) {
            pages[i] = sc.nextInt();
        }
        System.out.print("Enter the capacity of frame: ");
        capacity = sc.nextInt();
        int frame[] = new int[capacity];
        int table[][] = new int[noofpages][capacity];
        for (int i = 0; i < capacity; i++) {
            frame[i] = -1;
        }
        System.out.println("------------------------------------------------------------------------");
        for (int i = 0; i < noofpages; i++) {
            if (arr.contains(pages[i])) {
                arr.remove((Integer) pages[i]);
            }
            arr.add(pages[i]);
            int search = -1;
            for (int j = 0; j < capacity; j++) {
                if (frame[j] == pages[i]) {
                    search = j;
                    hit++;
                    System.out.printf("%4s","H");
                    break;
                }
            }
            if (search == -1) {
                if (isFull) {
                    int min_loc = noofpages;
                    for (int j = 0; j < capacity; j++) {
                        if (arr.contains(frame[j])) {
                            int temp = arr.indexOf(frame[j]);
                            if (temp < min_loc) {
                                min_loc = temp;
                                index = j;
                            }
                        }
                    }
                }
                frame[index] = pages[i];
                fault++;
```

```
                System.out.printf("%4s","F");
                index++;
                if (index == capacity) {
                    index = 0;
                    isFull = true;
                }
            }
            System.arraycopy(frame, 0, table[i], 0, capacity);
        }
        System.out.println("\n-------------------------------------------------------------------
------");
            for (int i = 0; i < capacity; i++) {
                for (int j = 0; j < noofpages; j++)
                    System.out.printf("%3d ", table[j][i]);
                System.out.println();
            }

        System.out.println("-------------------------------------------------------------------
----");
        hitRatio = ((double)hit / noofpages) * 100;
        faultRatio = ((double)fault / noofpages) * 100;
        System.out.println("Page Fault: " + fault + "\nPage Hit: " + hit);
        System.out.printf("Hit Ratio:%.2f \nFault Ratio:%.2f ", hitRatio, faultRatio);

    }
}
```