https://course.acciojob.com/idle?question=b2d01431-402d-4ff1-bae1-107a104d73e4

- EASY
- Max Score: 30 Points

Queue Using Two Stacks

You will be given Q queries. You need to implement a queue using two stacks according to those queries. Each query will belong to one of these two types:

1 x: Enqueue element x into the end of the nth queue. Returns true after the element is enqueued.

2: Dequeue the element at the front of the nth queue. Returns -1 if the queue is empty, otherwise, returns the dequeued element.

Note

Enqueue means adding an element to the end of the queue, while Dequeue means removing the element from the front of the queue.

Input Format

First line contains an integer Q denoting the number of queries .

A Query Q is of 2 Types

(i) 1 x (a query of this type means pushing "x" into the queue)

(ii) 2 (a query of this type means to pop element from queue and print the poped element)

The second line contains Q queries seperated by space.

Output Format

Print all the space seperated integer after performing Dequeue.

Example 1

Input

```
7
1 2 1 3 2 1 4 1 6 1 7 2
```

Output

```
2 3
```

Explanation

For this input, we have the number of queries, "Q" = 7.

Operations performed on the queue are as follows:

push(2): Push element ‘2’ into the queue.

push(3): Push element ‘3’ into the queue.

pop(): Pop the top element from the queue. This returns 2.

push(4): Push element ‘4’ into the queue.

push(6): Push element ‘6’ into the queue.

push(7): Push element ‘7’ into the queue.

pop(): Pop the top element from the queue. This returns 3.

Example 2

Input

```
7
1 11 1 51 1 26 2 1 6 2 2
```

Output:

```
11 51 26
```

Explanation

For this input, we have the number of queries, Q = 7.

Operations performed on the queue are as follows:

push(11): Push element ‘11’ into the queue.

push(51): Push element ‘51’ into the queue.

push(26): Push element ‘26’ into the queue.

pop(): Pop the top element from the queue. This returns 11.

push(6): Push element ‘6’ into the queue.

pop(): Pop the top element from the queue. This returns 51.

pop(): Pop the top element from the queue. This returns 26.

Constraints

1 <= Q <= 10^3

1 <= P <= 2

1 <= X <= 10^5

Topic Tags

- Queues
- Stacks

My code

```
// n java
import java.util.*;


class Main
{
	public static void main(String args[])
	{
		Scanner sc = new Scanner(System.in);

			Queue g = new Queue();

			int q = sc.nextInt();
			while(q>0)
			{
				int QueryType = sc.nextInt();
				if(QueryType == 1)
				{
					int a = sc.nextInt();
					g.enqueue(a);
				}
				else if(QueryType == 2)
```

```
                System.out.print(g.dequeue()+" ");
            q--;
            }
            System.out.println();




    }
}


class Queue
{
    Stack<Integer> st1 = new Stack<Integer>();
    Stack<Integer> st2= new Stack<Integer>();

    /*The method pop which return the element poped out of the
stack*/
    int dequeue()
    {
          // Your code here
        int t=-1;
          //Write your code here
           while(!st1.isEmpty())
                st2.add(st1.pop());
             if(!st2.isEmpty())
              t=st2.pop();
              while(!st2.isEmpty())
                st1.add(st2.pop());
```

```
            return t;
    }

    /* The method push to push element into the stack */
    void enqueue(int x)
    {
        // Your code here
        st1.add(x);
    }
}
```