https://course.acciojob.com/idle?question=89a5f158-cacc-427d-a317-0967668d8f2b

- EASY
- Max Score: 30 Points
-

# Implement Queue using stack - enQueue O(1)

---

**Implement Queue using two stack $S1$ and $S2$. You need to complete the push and pop function of Queue class. You are given 2 types of query $1$ for push an integer into queue and $2$ for enqueue the value from the queue and print.**

**Note: implement enQueue in O(1)**

## Input Format

**First line contains $q$ of queries.**

**Followed by $q$ lines.**

**Query of type $1$ is followed by an integer $x$ to push element in the queue.**

**Query of type $2$ is for dequeue the last value from the queue and print.**

## Output Format

**Print the value for dequeue operations in the query given.**

## Example 1

Input

```
5
1 2
1 3
2
```

```
1 4
2
```

Output:

```
2 3
```

Explanation

First we push 2 , then for second query we push 3, for third query we dequeue 2 and print, for 4th query we push 4 and for 5th query we dequeue 3 and print.

## Example 2

Input

```
3
2
1 2
2
```

Output:

```
-1 2
```

Explanation:

In the first query we don't have any element but we use dequeue query so we print -1, in the 2nd query we push 2 and in the 3rd we dequeue 2 and print 2.

## Constraints

**1 <= Total number of queries <= 100**

**1 <= value in each query <= 100**

### Topic Tags

- Queues

- Stacks

# My code

```
// in java
import java.io.*;
import java.util.*;

class StackQueue
{
   Stack<Integer>st1;
        Stack<Integer>st2;
        public StackQueue()
        {
                st1=new Stack<>();
                st2=new Stack<>();
        }
    //Function to push an element in queue by using 2 stacks.
    void Push(int x)
    {
          //Write your code here
         st1.add(x);
    }
    //Function to pop an element from queue by using 2 stacks.
    int Pop()
    {
              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;
    }
}
public class Main {
    public static void main(String args[]) throws IOException {
        Scanner sc = new Scanner(System.in);
        StackQueue s = new StackQueue();
        int q = sc.nextInt();
        ArrayList<Integer> ans= new ArrayList<>();
        while(q>0)
        {
            int QueryType = sc.nextInt();
            if(QueryType == 1)
            {
                int a = sc.nextInt();
                s.Push(a);
            }
            else if(QueryType == 2)
            ans.add(s.Pop());
        q--;
        }
        for(int x:ans)
        System.out.print(x+" ");
     System.out.println();
    }
}
```