### Reverse First K elements of Queue

Given an integer **K**and a [queue](http://www.geeksforgeeks.org/queue-data-structure/) of integers, we need to reverse the order of the first**K** elements of the queue, leaving the other elements in the same relative order.

Only following standard operations are allowed on queue.

* enqueue(x) : Add an item x to rear of queue
* dequeue() : Remove an item from front of queue
* size() : Returns number of elements in queue.
* front() : Finds front item.  
  **Note:** The above operations represent the general processings. In-built functions of the respective languages can be used to solve the problem.

**Example 1:**

**Input:**

5 3

1 2 3 4 5

**Output:**

3 2 1 4 5

**Explanation:**

After reversing the given

input from the 3rd position the resultant

output will be 3 2 1 4 5.

**Example 2:**

**Input:**

4 4

4 3 2 1

**Output:**

1 2 3 4

**Explanation:**

After reversing the given

input from the 4th position the resultant

output will be 1 2 3 4.

**Expected Time Complexity** : O(N)  
**Expected Auxiliary Space**: O(K)

**Constraints:**  
1 <= N <= 1000  
1 <= K <= N

**Company Tags**

[**Amazon**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Amazon)

//{ Driver Code Starts

// Initial Template for Java

import java.util.\*;

class ModifyQueue {

public static void main(String[] args) {

// Taking input using class Scanner

Scanner sc = new Scanner(System.in);

// Taking total number of testcases

int t = sc.nextInt();

while (t-- > 0) {

// Taking count of total number of elements

int n = sc.nextInt();

// Taking count of total elements

// that need to be reversed

int k = sc.nextInt();

// Creating a Queue

Queue<Integer> q = new LinkedList<>();

// adding all the elements to the Queue

while (n-- > 0) {

q.add((int)sc.nextInt());

}

// Creating an object of class GfG

CodingMaxima g = new CodingMaxima ();

// calling modifyQueue of class GfG

// and passing Queue and k as arguments

// and storing the reuslt in a new Queue

Queue<Integer> ans = g.modifyQueue(q, k);

// Printing all the elements from the

// new Queue and polling them out

while (!ans.isEmpty()) {

int a = ans.peek();

ans.poll();

System.out.print(a + " ");

}

System.out.println();

}

}

}

// } Driver Code Ends

// User function Template for Java

class CodingMaxima {

// Function to reverse first k elements of a queue.

public Queue<Integer> modifyQueue(Queue<Integer> q, int k) {

Stack<Integer> stack=new Stack<>();

Queue<Integer> ans=new LinkedList<Integer>();

while(k-->0){

stack.push(q.poll());

// q.poll();

}

while(!stack.isEmpty()){

ans.add(stack.pop());

}

while(!q.isEmpty()){

ans.add(q.poll());

//q.poll();

}

return ans;

}

}