STL queue

Implement different operations on a queue q .

Input:
The first line of input contains an integer **T** denoting the no of test cases . Then T test cases follow. The first line of input contains an integer **Q** denoting the no of queries . Then in the next line are **Q** space separated queries .
A query can be of four types
1. a x (Pushes an element x at the end of the queue q)
2. b (if queue is not empty pops the front element and prints it, else prints -1)
3. c (prints the size of the queue)
4. d (if queue is not empty prints the front element of the queue, else prints -1)
5. e (if queue is not empty prints the last element of the queue else prints -1)

Output:
The output for each test case will be space separated integers denoting the results of each query .

Constraints:
1<=T<=100
1<=Q<=100

Example:

Input
2
5
a 4 a 6 a 7 b c
4
a 55 a 11 d e

Output
4 2
55 11

Explanation :
For the first test case
There are five queries. Queries are performed in this order
1. a 4 { queue q has 4 }
2. a 7 {queue q has 4,7 }
3. a 6 {queue q has 4,7,6}
4. b {pop 4 from queue q and prints it queue now has 7,6}
5. c {prints the size of the queue q ie 2}

For the sec test case
There are three queries. Queries are performed in this order
1. a 55 {queue q has 55 }
2. a 11 {queue q has 55 ,11 }
3. d {prints the front element of the queue q ie. 55 }
4. e {prints the end element of the queue q ie 11 }