Solve the problem using C++ STL Vector

Implement different operations on a vector A .

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 five types

1. a x (Adds an element x to the vector A at the end)

2. b (Sorts the vector A in ascending order)

3. c (Reverses the vector A)

4. d (prints the size of the vector)

5. e (prints space-separated values of the vector)

5. f (Sorts the vector A in descending order)

6. g x (Checks if x exist in the vector A)

7. h y (Erase ‘yth’ element from the vector check for error - Whether the vector contains yth element or not)

8. i (Erase last element from the vector, check for error - Whether the vector contains any elements or not)

9. j x (Performs binary search to check x exist in vector A or not, before performing binary_search make sure you sort the array, also if multiple elements exist in the vector print the last occurrence)

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

6

a 4 a 6 a 7 b c e

4

a 55 a 11 d e

Output

7 6 4

2 55 11

Explanation :

For the first test case

There are six queries. Queries are performed in this order

1. a 4 { Vector has 4 }

2. a 7 {vector has 7 }

3. a 6 {vector has 6}

4. b {sorts the vector in ascending order, vector now is 5 6 7}

5. c {reverse the vector }

6. e {prints the element of the vectors 7 6 4}

For the sec test case

There are four queries. Queries are performed in this order

1. a 55 (vector A has 55}

2. a 11 (vector A has 55 ,11}

3. d (prints the size of the vector A ie. 2)

4. e (prints the elements of the vector A ie 55 11)
