Array Simulation

I am retired now, so no work, a lot of time to spare and a lot of problems to share. Well, I am thinking of the old days when I was a problem solver as I am stuck with a tough problem that I want to share with you.

Given an array and some operations on the array, you have to print the final state of the array. Say, the array is `a[]`, the size is **n** and indexed from **0** to **n-1**. The operations are:

1. `S D`. **D** is an integer. **D** will be added with all the elements of the array.
2. `M D`. **D** is an integer. All the elements of the array will be multiplied by **D**.
3. `D K`. **K** is a non zero integer. All the elements of the array will be divided by **K**(integer division).
4. `P Y Z`. **Y**and **Z** are integers. It will swap the elements **a[Y]** and **a[Z]**.
5. `R`. Reverses the array, i.e `[1, 2, 4, 3]`, becomes `[3, 4, 2, 1]`.

Input

Input starts with an integer **T (≤ 100)**, denoting the number of test cases.

Each case contains two integers **n (1 ≤ n ≤ 100)** and **m (0 ≤ m ≤ 101)**. The next line contains **n** space separated integers denoting the elements of the array. Each of the next **m** lines contains an operation defined above. You can assume that no operation will overflow/underflow the 32 bit signed integer range or access any invalid array reference.

Output

For each case, print the case number first. In the next line you have to print the elements of the array. Two elements should be separated by a single space. There should be no trailing space after the last integer of the array.

Sample Input	Sample Output
2 5 3 1 2 3 4 5 P 0 1 S 1 R 4 2 2 7 8 1 M 10 D 5	Case 1: 6 5 4 2 3 Case 2: 4 14 16 2