Bubble Sort

One of the simplest sorting algorithms, the Bubble Sort, can be expressed as (0-based array):

```
procedure bubbleSort( A : list of sortable items )
    n = length(A)
    repeat
        swapped = false
        for i = 1 to n-1 inclusive do
            /* if this pair is out of order */
            if A[i-1] > A[i] then
                /* swap them and remember something changed */
                swap( A[i-1], A[i] )
                swapped = true
            end if
            end for
    until not swapped
end procedure
```

Now, given an array of N integers, you have to find out how many swap opeartions occur if the Bubble Sort algorithm is used to sort the array.

Input

Input begins with a line containing an integer **T(1<=T<=100)**, denoting the number of test cases. Then T test cases follow. Each test case begins with a line containing an integer **N(1<=N<=10000)**, denoting the number of integers in the array, followed by a line containing **N** space separated 32-bit integers.

Output

For each test case, output a single line in the format **Case X: Y**, where **X** denotes the test case number and **Y** denotes the number of swap operations needed modulo 10000007.

Example

Input:

```
1
4
```

3 2 1 4

Output:
Case 1: 3