Inversions

Write a program that, given a permutation P of the set $\{1, \ldots, n\}$, computes the number of pairs (i, j) such that $1 \leqslant i < j \leqslant n$ and $P[i] > P[j]$ (such a pair is called an *inversion* in P).

Your program should work in time $O(n \log n)$ on a randomly chosen permutation.

Input

The first line contains integer z ($1 \leqslant z \leqslant 2 \cdot 10^9$) – the number of data sets. Each data set is as follows:

The first line contains the number n ($1 \leqslant n \leqslant 10000$) denoting the size of the permutation P. The second line contains n consecutive entries of the permutation P, separated by a space.

Output

The number of inversions in P.

Example

For the input:

```
5
3
1 2 3
3
3 2 1
10
1 4 2 5 3 9 10 8 7 6
2
1 2
2
2 1
```

the output is:

```
0
3
12
0
1
```