Tree

Consider an n-vertex binary search tree T containing n keys 1, 2 ... n. A permutation $p = [p_1 ... p_n]$ of the integers 1, 2 ... n is said to be *consistent with the tree T* if the tree can be built from the empty one as the result of inserting integers $p_1, p_2 ... p_n$. Find how many permutations are consistent with the tree T.

Illustration

Exactly 2 permutations are consistent with the tree in the figure below.
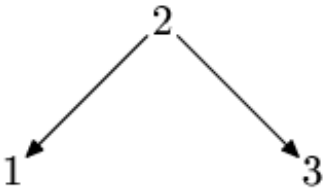
Task

Write a program that for each data set from a sequence of several data sets:

- reads from the input file a description of an input tree T,
- computes the number of permutations consistent with T,
- writes the result to output.

Input

The first line of the input file contains one positive integer d not larger than 10. This is the number of data sets. The data sets follow. Each set of data occupies two consecutive lines of the input file. The first line contains only one integer n, 1 <= n <= 30. This is the number of vertices of the tree. The second line contains a sequence of n integers separated by single spaces. The integers are keys in the input tree given in the prefix order. The first integer in the sequence is the key from the root of the tree. It is followed by the keys from the left subtree written in the prefix order. The sequence ends with the keys from the right subtree, also given in the prefix order.

Output

For each i = 1 ... d, your program should write to the i-th line of output the number of permutations consistent with the tree described in the i-th data set.

Example

Sample input:
5
3
2 1 3
3
1 2 3

1
1
4
2 1 3 4
4
1 4 2 3

Sample output:
2
1
1
3
1