Jumping Game

Filename: *jump*

You've been hired by a playground manufacturing company to design jumping blocks for children. Typically, the blocks are set up in a row and children start at the left most block and then continually jump to the adjacent block to the right until they get to the last block. Each block is an individual piece and a set of blocks can be rearranged in any order. Unfortunately, due to safety reasons, some of the orderings of blocks are not permitted. Depending on the age of the children, there are restrictions on how far up they can jump and how far down they can jump.

Consider a situation where there are four blocks of heights 36 inches, 30 inches, 40 inches and 20 inches where the children can jump up no more than 8 inches and can jump down no more than 16 inches. In this scenario, placing the 36 inch block followed by the 40 inch block, followed by the 30 inch block and ending with the 20 inch block would be a valid arrangement since the three jumps involved would be a jump of 4 inches up, 10 inches down and 10 inches down. However, placing the 40 inch block third and the 20 inch block last would be invalid since this would result in a jump down of 20 inches, more than the maximum of 16 inches. Note that since children don't jump up to the first block from the ground and don't jump down to the ground from the last block, there's no restriction on the heights of these blocks based on how far they are from the ground. (They climb onto the first block and get help from friends getting down from the last block.)

The Problem

Given a set of blocks with distinct heights (in inches), and maximum limits for which children can jump up and jump down, determine the number of orderings of those blocks that don't contain any invalid jumping distances for the kids.

The Input

The first line of input will contain a single positive integer, n ($n \leq 100$), the number of sets of blocks to evaluate. The following n lines will each contain a single input case. All the values within an input case will be separated by spaces. The first value for each input case will be a positive integer, b ($2 \leq b \leq 10$), the number of blocks for the input case. This will be followed by the b unique positive integers, each within the range of 1 to 100, inclusive, representing the heights of each of the blocks for the input case. This will be followed by two more positive integers, u ($u \leq 100$) and d ($d \leq 100$), representing the maximum number of inches the children for that input case can jump up and jump down, respectively.

The Output

For each test case, output a single integer representing the number of orderings of the blocks with every jump from left to right being a valid jump according to the specifications given above.

Sample Input

```
3
4 36 30 40 20 8 16
3 10 20 30 6 8
3 10 20 30 8 10
```

Sample Output

```
3
0
1
```