

7254 Immortality of Frog

N frogs are attempting to prolong their life-span. They live in the bottom of a well which can be described as a two-dimensional $N \times N$ grid. Grid (i, j) is located in the i -th row and the j -th column. At the beginning, the i -th frog lives in the bottom of i -th column, i.e. the place below grid $(1, i)$.

The frogs are so devout that God decides to give them a chance. In each row i , a horizontal membrane ranging from (i, L_i) to (i, R_i) inclusively is created. A capsule of elixir is placed at one of the grids of the membrane with uniform probability.

Now the frogs are jumping upwards to pursue immortality. The i -th frog would be in grid (j, i) after j jumps. When a frog arrives at a grid that contains a capsule of elixir, it will eat the capsule and gain immortality. After that, it continues jumping upwards until it gets out of the well.

A membrane is considered “bad” if it covers less than N grids. The frogs are very sensitive, so they can only endure passing through 10 bad membrane. When a frog reaches the 11-th bad membrane, it thinks that there is no hope to get out of the well, so it will go back to the bottom of well and live there until death, even though it has eaten a capsule of elixir already.

The frogs are friends, so they want all of them gain immortality and live a happy life out of the well. They want to know the probability P that every frog eats exactly one capsule of elixir and gets out of the well.

Input

The first line of input contains a number T indicating the number of test cases ($T \leq 100$).

Each test case starts with a line containing an integer N as described above ($1 \leq N \leq 1000$). The second line contains N space separated integers L_1, L_2, \dots, L_N . The third line contains N space separated integers R_1, R_2, \dots, R_N ($1 \leq L_i \leq R_i \leq N$)

Output

For each test case, output a single line consisting of ‘Case # X : Y ’. X is the test case number starting from 1. Y is an integer defined as the probability P multiplies $\prod_{i=1}^N (R_i - L_i + 1)$. As the answer could be huge, you only need to output it *module* 105225319.

Sample Input

```
2
2
1 2
1 2
2
1 1
2 2
```

Sample Output

```
Case #1: 1
Case #2: 2
```