

Longest Monotone Subsequence

Description

You are given an array of n positive integers a_1, a_2, \dots, a_n .

Each integer a_i can be taken out x_i times. Calculate the number of **non-decreasing subsequence** of length S that can be taken out from the array. Note that S is the length of the **longest non-decreasing subsequence** for the array.

Input Format

The first line includes one integer n , which is the length of the array.

The second line includes n integer a_1, a_2, \dots, a_n .

The third line includes n integer x_1, x_2, \dots, x_n , which is the number that the i -th integer can be taken out.

Output Format

Please output two line.

The first line includes one integer S , which is the length of the longest non-decreasing subsequence for the array.

The second line includes one integer, which is the answer.

Sample

Sample Input

```
4
3 6 2 5
3 1 1 3
```

Sample Output

```
2
4
```

For 40% testcases: $1 \leq n \leq 20$.

For 100% testcases: $1 \leq n \leq 200, 1 \leq a_i, x_i \leq 500$.

Explanation

The length of the longest non-decreasing subsequence for [3,6,2,5] is 2.

Therefore we can take out 4 subsequences: [3, 6], [2, 5], [3, 5], [3, 5] (since 3 and 5 can be taken out 3 times).

