J. Bottles

time limit per test: 2 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

Nick has n bottles of soda left after his birthday. Each bottle is described by two values: remaining amount of soda a_i and bottle volume b_i ($a_i \le b_i$).

Nick has decided to pour all remaining soda into minimal number of bottles, moreover he has to do it as soon as possible. Nick spends x seconds to pour x units of soda from one bottle to another.

Nick asks you to help him to determine k — the minimal number of bottles to store all remaining soda and t — the minimal time to pour soda into k bottles. A bottle can't store more soda than its volume.
All remaining soda should be saved.

Input

The first line contains positive integer n ($1 \le n \le 100$) — the number of bottles.

The second line contains n positive integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 100$), where a_i is the amount of soda remaining in the i-th bottle.

The third line contains n positive integers $b_1, b_2, \ldots, b_n$ ($1 \le b_i \le 100$), where b_i is the volume of the i-th bottle.

It is guaranteed that $a_i \le b_i$ for any i.

Output

The only line should contain two integers k and t, where k is the minimal number of bottles that can store all the soda and t is the minimal time to pour the soda into k bottles.

Examples

input

```
4
3 3 4 3
4 7 6 5
```

output

```
2 6
```

input

```
2
1 1
100 100
```

output

```
1 1
```

input

```
5
10 30 5 6 24
10 41 7 8 24
```

output

```
3 11
```

Note

In the first example Nick can pour soda from the first bottle to the second bottle. It will take 3 seconds. After it the second bottle will contain $3+3=6$ units of soda. Then he can pour soda from the fourth bottle to the second bottle and to the third bottle: one unit to the second and two units to the third. It will take $1+2=3$ seconds. So, all the soda will be in two bottles and he will spend $3+3=6$ seconds to do it.