

USACO 2023 December Contest, Bronze

USACO 2023 December Contest, Bronze

Problem 1. Candy Cane Feast

INPUT FORMAT (pipe stdin)

OUTPUT FORMAT (pipe stdout)

SAMPLE INPUT&OUTPUT

INPUT1

OUTPUT1

SPECIFICATION1

SCORING

Problem 2. Cowntact Tracing 2

INPUT FORMAT (pipe stdin)

OUTPUT FORMAT (pipe stdout)

SAMPLE INPUT&OUTPUT

INPUT1

OUTPUT1

SPECIFICATION1

INPUT2

OUTPUT2

SPECIFICATION2

SCORING

Problem 3. Farmer John Actually Farms

INPUT FORMAT (pipe stdin)

OUTPUT FORMAT (pipe stdout)

SAMPLE INPUT&OUTPUT

INPUT1

OUTPUT1

SPECIFICATION1

INPUT2

OUTPUT2

SPECIFICATION2

SCORING

Problem 1. Candy Cane Feast

Farmer John's cows have quite the sweet tooth, and they especially enjoy eating candy canes! FJ has N total cows, each with a certain initial height and he wants to feed them M candy canes, each also of varying height ($1 \leq N, M \leq 2 \cdot 10^5$).

FJ plans to feed the candy canes one by one to the cows, in the order they are given in the input. To feed a candy cane to his cows, he will hang the candy cane so that initially the candy cane is just touching the ground. The cows will then line up one by one, in the order given by the input, and go up to the candy cane, each eating up to their height (since they cannot reach any higher). The candy cane stays suspended in place where it is initially set up and is not lowered to the ground, even after cows eat the bottom of the candy cane. It is possible a cow may eat nothing during her turn, if the base of the candy cane is already above that cow's height. After every cow has had their turn, the cows grow in height by how many units of candy cane they ate, and Farmer John hangs the next candy cane and the cows repeat the process again (with cow 1 again being the first to start eating the next candy cane).

INPUT FORMAT (pipe stdin)

The first line contains N and M .

The next line contains the initial heights of the N cows, each in the range $[1, 10^9]$.

The next line contains the heights of the M candy canes, each in the range $[1, 10^9]$.

OUTPUT FORMAT (pipe stdout)

The final heights of each of the N cows on separate lines.

Note that the large size of integers involved in this problem may require the use of 64-bit integer data types (e.g., a "long long" in C/C++).

SAMPLE INPUT&OUTPUT

INPUT1

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

OUTPUT1

```
1 7
2 2
3 7
```

SPECIFICATION1

The first candy cane is 66 units tall.

1. The first cow eats the portion of the first candy cane up to height 33, after which the remaining portion of the first candy cane occupies heights $[3, 6]$.
2. The second cow is not tall enough to eat any of the remaining portion of the first candy cane.
3. The third cow eats two additional units of the first candy cane. The remaining portion of the first candy cane, occupying heights $[5, 6]$, is not eaten.

Next, each cow grows by the amount she ate, so the heights of the cows become $[3 + 3, 2 + 0, 5 + 2] = [6, 2, 7]$.

The second candy cane is 1 unit tall, and the first cow eats all of it.

SCORING

- Inputs 1-9: $N, M \leq 10^3$
- Inputs 10-13: No additional constraints.

Problem credits: Agastya Goel

Problem 2. Cowntact Tracing 2

Farmer John has N cows in a line ($1 \leq N \leq 3 \cdot 10^5$). Unfortunately, there is a sickness spreading throughout.

Initially, some cows start off infected. Every night, an infected cow spreads the sickness to the cows on their left and right (if they exist). Once a cow is infected, she stays infected.

After some amount of nights, Farmer John realizes that the issue has gotten out of control, so he tests his cows to determine who has the sickness. Find the minimum number of cows that could have started with the sickness.

INPUT FORMAT (pipe stdin)

The first line contains N , the number of cows that Farmer John has.

The next line contains an N character bitstring of only 1s and 0s where a 1 represents an infected cow and a 0 represents an uninfected cow after some number of nights.

OUTPUT FORMAT (pipe stdout)

Output a single integer: the minimum number of cows that could have started with the sickness.

SAMPLE INPUT&OUTPUT

INPUT1

```
1 5
2 11111
```

OUTPUT1

```
1 1
```

SPECIFICATION1

Suppose the middle cow was the only cow to start off infected. Then the cows would be infected in the following order:

```
1 0 nights:    00100 (the third cow is initially infected)
2 1 night:  -> 01110 (the second and fourth cows are now infected)
3 2 nights: -> 11111 (the first and fifth cows are now infected)
4 3 nights: -> 11111 (all cows already were infected, so no additional
   cows are infected)
5           -> ...
```

After two or more nights, the final state of the cows would look like the input. There are many other initial states and number of nights that could have produced the input state, such as:

```
1 0 nights:    10001
2 1 night:  -> 11011
3 2 nights: -> 11111
```

or:

```
1 0 nights:    01001
2 1 night:  -> 11111
```

or:

```
1 0 nights:    01000
2 1 night:  -> 11100
3 2 nights: -> 11110
4 3 nights: -> 11111
```

All of these initial states contain at least one infected cow.

INPUT2

```
1 6
2 011101
```

OUTPUT2

```
1 4
```

SPECIFICATION2

The only initial state and number of nights that could have led to this final state is if no nights have passed and each of the four infected cows in the input started off with the sickness.

SCORING

- Inputs 1-5: $N \leq 1000$
- Inputs 6-10: No additional constraints.

Problem credits: Suhas Nagar

Problem 3. Farmer John Actually Farms

Farmer John is growing N ($1 \leq N \leq 2 \cdot 10^5$) plants of asparagus on his farm! However some of his plants have genetic differences, so some plants will grow faster than others. The initial height of the i th plant is h_i inches, and after each day, the i th plant grows by a_i inches.

FJ likes some of his plants more than others, and he wants some specific plants to be taller than others. He gives you an array of distinct values $t_1, \dots, t_N$ containing all integers from 0 to $N - 1$ and he wants the i th plant to have exactly t_i other plants that are taller than it. Find the minimum number of days so that FJ's request is satisfied, or determine that it is impossible.

INPUT FORMAT (pipe stdin)

The first will consist of an integer T , denoting the number of independent test cases ($1 \leq T \leq 10$).

The first line of each test case consists of an integer N .

The second line consists of N integers h_i ($1 \leq h_i \leq 10^9$) denoting the initial height of the i th plant in inches.

The third line consists of N integers a_i ($1 \leq a_i \leq 10^9$) denoting the number of inches the i th plant grows each day.

The fourth line consists of N distinct integers t_i denoting the array that FJ gives you.

It is guaranteed that the sum of N over all test cases does not exceed $2 \cdot 10^5$.

OUTPUT FORMAT (pipe stdout)

Output T lines, the answer to each test case on a different line. If it is not possible, output -1 .

Note that the large size of integers involved in this problem may require the use of 64-bit integer data types (e.g., a "long long" in C/C++).

SAMPLE INPUT&OUTPUT

INPUT1

```
1 6
2 1
3 10
4 1
5 0
6 2
7 7 3
8 8 10
9 1 0
10 2
```

```
11 3 6
12 10 8
13 0 1
14 2
15 7 3
16 8 9
17 1 0
18 2
19 7 7
20 8 8
21 0 1
22 2
23 7 3
24 8 8
25 1 0
```

OUTPUT1

```
1 0
2 3
3 2
4 5
5 -1
6 -1
```

SPECIFICATION1

In the first sample input, there are 6 test cases.

In the first test case, there is only one plant, so the condition is satisfied on day 0.

In the second test case, we need the first plant to be shorter than the second plant. After day 1, the heights are 15 and 13. After day 2, the heights are both 23. After day 3, the heights are 31 and 33, and that's the first day in which the condition is satisfied.

The third and fourth test cases are similar to the second.

In the fifth test case, both plants start with an initial height of 7 and a growth rate of 8. So they will always have identical heights, and therefore the condition is never satisfied.

In the sixth test case, the condition is not satisfied initially and the growth rates are the same. So the condition can never be satisfied.

INPUT2

```
1  2
2  5
3  7 4 1 10 12
4  3 4 5 2 1
5  2 1 0 3 4
6  5
7  4 10 12 7 1
8  3 1 1 4 5
9  2 4 3 1 0
```

OUTPUT2

```
1  4
2  7
```

SPECIFICATION2

In the second sample input, there are 2 test cases.

In the first test case, the final heights after day 4 are 19, 20, 21, 18, 16.

In the second test case, the final heights after day 7 are 25, 17, 19, 35, 36.

SCORING

- Input 1: $N \leq 2$
- Inputs 2-3: $N \leq 50$ and $a_i, h_i \leq 10^3$
- Inputs 4-6: $N \leq 10^3$
- Inputs 7-11: No additional constraints.

Problem credits: Chongtian Ma