Efficient Adding

You are tasked with writing a program that adds a sequence of numbers. But the added challenge is to do so efficiently!

The cost of adding two numbers `a` and `b` is equal to their sum `a+b`. For example: to add 1, 2, and 3, you can do it as follows:

1 + 2 = 3, cost of 3 3 + 3 = 6, cost of 6 Total cost = 9

or

2 + 3 = 5, cost of 5 1 + 5 = 6, cost of 6 Total cost = 11

or

1 + 3 = 4, cost of 4 2 + 4 = 6, cost of 6 Total cost = 10

Your goal is to add the numbers so that the cost is as small as possible.

Input

The first line of input contains a positive number `N` (`2 <= N <= 5000`) that tells you how many numbers there are to add.

The second line of input contains those `N` numbers `0 <= n_1, n_2, ..., n_N <= 100,000`.

Output

The minimum total cost of addition followed by a newline.

Example 1

```
Input:
3
1 2 3

Output:
9
```

Example 2

```
Input:
4
1 2 3 4

Output:
19
```

Bus Tour

Hangzhou is a beautiful city for tourist. The buses with numbers that begin with 'Y' go to scenic spots, train stations, tourist centers and bus stations (all the things that the tourists need). Y1 is a bus route starting from Lingyin Temple and around the West Lake. There is N stops for Y1.

The scenery between two adjacent stops can be measured by a integer scale of "niceness". Positive niceness value indicates nice view and negative value indicates the scenery between stops is dull.

Macro Pool is a reporter for *Tourists' World* who arrived in Hangzhou last week. His task is to find two different stops along the Y1 bus route such that the niceness score of all the segments in between them is the largest.

Can you help him find those stops?

Input

The first line of input contains an integer, N, the number of stops along the Y1 bus route, 2 <= N<= 20,000

Each of the next N-1 lines contains a single integer. The i-th integer indicating niceness between stop i and stop i+1.

The absolute value of niceness will not exceed 10^9.

Output If the maximum possible sum between two stops is not positive, your program should print a line:

"Yet another overrated tourist destination"

Otherwise, your program should identify the beginning bus stop i and the ending bus stop j that identify the segment of the route which yields the maximal sum of niceness.

If more than one segment is equally maximally nice, choose the one with the longest bus ride (largest number of stops, j - i). To break ties further in longest maximal segments, choose the segment that begins with the earliest stop (lowest i).

Print a line in the form:

"The nicest part of Y1 is between stops i and j"

Example 1

```
Input
3
-2
5

Output
The nicest part of Y1 is between stops 2 and 3
```

Example 2

```
Input
4
-1
-1
-1

Output
Yet another overrated tourist destination
```

No Change

Chad has been looking for a rare anime garage kit for years.

He just learned about the seller that has exactly what he wants. The seller is a character known for strange rules of trade: she never gives any change.

Figure 1:

The price of the garage kit is M. Chad has brought N coins with him such that the i-th coin has value a_i. He is determined to get the garage kit and is willing to pay more than M, if necessary, but he wants to overpay as little as possible. Moreover, he wants to minimize the number of coins he uses.

Please help him figure out which coins he should use for the purchase.

Input

The first line contains one positive integer M, the price of the garage kit. The price will not exceed 10,000.

The following line contains one positive integer N, the number of coins Chad has, N <= 100.

The next N lines contain positive integers indicating the value of each coin that Chad has. The values are any positive integers no greater than 10,000.

Chad has brought enough money to buy the kit, so the total value of his coins will always be equal to or greater than the price of the kit.

Output

Output a single line containing two integers: the total amount paid and the total number of coins used.

Example 1

```
Input:
14
3
5
10
20

Output:
15 2
```

Example 2

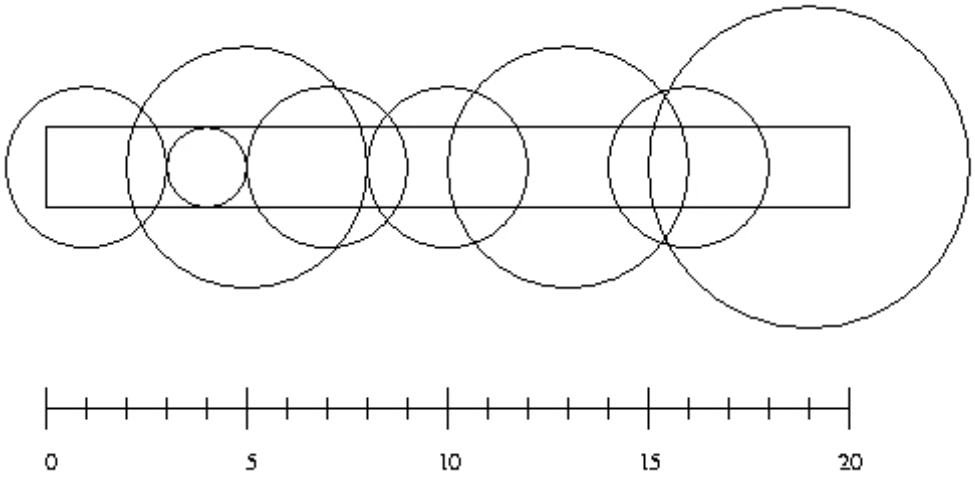
```
Input:
5
3
2
3
5

Output:
5 1
```

Sprinklers

Farmer John has a large field, and he is thinking of planting sweet corn in some part of it. After surveying his field, FJ found that it forms a horizontal strip l meters long and w meters wide.

There are n sprinklers installed at the horizontal center line of the strip. For each sprinkler you are given its radius of operation and its position as the distance from the left end of the center line. Farmer John wants to plant sweet corn in the whole field, but he wishes to turn on as few sprinklers as possible to save the water.

Find the minimum number of sprinklers that need to be turned on in order to water the entire field.



Input

The first line contains integer numbers n, l and w with 1 <= n <= 10,000, 1 <= l <= 10,000,000, and 1 <= w <= 100. The next n lines contain two integers giving the position x (0 <= x <= l) and radius of operation r (1 <= r <= 1000) of a sprinkler.

The picture above illustrates the first case from the sample input.

Output

Output one integer followed by a newline: the minimum number of sprinklers needed to water the entire strip.

If it is impossible to water the entire strip, output -1.

Example 1 (shown in the image above)

```
Input:
8 20 2
5 3
4 1
1 2
7 2
10 2
13 3
16 2
19 4

Output:
6
```

Example 2

```
Input:
3 10 1
```

```
3 5
9 3
6 1

Output:
2
```

Example 3

```
Input:
3 10 1
5 3
1 1
9 1

Output:
-1
```