

07. Functions 1

- Linear function
- Absolute value function
- Quadratic function
- User defined function
- GCD
- XOR

Exercise 1: Pens and pencils

Exercise 2: Complete tasks

Exercise 3: Single number I

Exercise 4: Single number II

Exercise 5: Taming (Feb 2018)

Exercise 6: Cownditioning (Dec 2021)

Pens and pencils

You are given an integer T indicating the amount of money you have. You are also given two integers x and y indicating the price of a pen and pencil respectively. You can spend part or all of your money to buy multiple quantities (or none) of each kind of writing utensil.

Report the number of distinct ways you can buy some number of pens and pencils.

Input (from terminal / stdin)

- The only line contains the integers T, x, y , all in the range $[1, 1e6]$.

Output (to terminal / stdout)

- Report the number of distinct ways to buy the pens and pencils.

Sample input

20 10 5

Sample output

9

If you buy 0 pens, you can buy 0, 1, 2, 3, or 4 pencils.

If you buy 1 pen, you can buy 0, 1, or 2 pencils.

If you buy 2 pens, you cannot buy any pencils.

The total number of ways to buy pens and pencils is $5 + 3 + 1 = 9$.

Complete tasks

You are given a 0-indexed integer array A , where A_i represents the difficulty level of a task. In each round, you can complete either 2 or 3 tasks of the same difficulty level.

Report the minimum rounds required to complete all the tasks, or -1 if it is not possible to complete all the tasks.

Input (from terminal / stdin)

- The first line contains integer N , $1 \leq N \leq 1e5$.
- The next line contains the N difficulty level numbers, all in the range $[1, 1e6]$.

Output (to terminal / stdout)

- Report the min rounds to complete the tasks.

Sample input

```
10
2 2 3 3 2 4 4 4 4 4
```

Sample output

```
4
```

To complete all the tasks, a possible plan is:

In the first round, you complete 3 tasks of difficulty level 2.

In the second round, you complete 2 tasks of difficulty level 3.

In the third round, you complete 3 tasks of difficulty level 4.

In the fourth round, you complete 2 tasks of difficulty level 4.

It can be shown that all the tasks cannot be completed in fewer than 4 rounds, so the answer is 4.

Single number I

Given an array A of integers, every element appears twice except for one. Find that single one.

Input (from terminal / stdin)

- The first line contains integer N , $1 \leq N \leq 1000$.
- The next line contains the N integers, all in the range $[0, 1e6]$.

Output (to terminal / stdout)

- Report the single number.

Sample input

```
3
2 4 4
```

Sample output

```
2
```

Single number II

Given an integer array A where every element appears three times except for one, which appears exactly once. Find the single element.

Input (from terminal / stdin)

- The first line contains integer N , $1 \leq N \leq 1000$.
- The next line contains the N integers, all in the range $[0, 1e6]$.

Output (to terminal / stdout)

- Report the single number.

Sample input

```
7
2 2 4 2 4 5 4
```

Sample output

```
5
```

Taming (Feb 2018)

Early in the morning, Farmer John woke up to the sound of splintering wood. It was the cows, and they were breaking out of the barn again!

Farmer John was sick and tired of the cows' morning breakouts, and he decided enough was enough: it was time to get tough. He nailed to the barn wall a counter tracking the number of days since the last breakout. So if a breakout occurred in the morning, the counter would be 0 that day; if the most recent breakout was 3 days ago, the counter would read 3. Farmer John meticulously logged the counter every day.

The end of the year has come, and Farmer John is ready to do some accounting. The cows will pay, he says! But some entries of his log are missing!

Farmer John is confident that the he started his log on the day of a breakout. Please help him determine, out of all sequences of events consistent with the log entries that remain, the minimum and maximum number of breakouts that may have taken place over the course of the logged time.

INPUT FORMAT (file taming.in):

The first line contains a single integer N ($1 \leq N \leq 100$), denoting the number of days since Farmer John started logging the cow breakout counter.

The second line contains N space-separated integers. The i -th integer is either -1 , indicating that the log entry for day i is missing, or a non-negative integer a_i (at most 100), indicating that on day i the counter was at a_i .

OUTPUT FORMAT (file taming.out):

If there is no sequence of events consistent with Farmer John's partial log and his knowledge that the cows definitely broke out of the barn on the morning of day 1, output a single integer -1 . Otherwise, output two space-separated integers m followed by M , where m is the minimum number of breakouts of any consistent sequence of events, and M is the maximum.

SAMPLE INPUT:

```
4
-1 -1 -1 1
```

SAMPLE OUTPUT:

```
2 3
```

In this example, we can deduce that a breakout had to occur on day 3. Knowing that a breakout also occurred on day 1, the only remaining bit of uncertainty is whether a breakout occurred on day 2. Hence, there were between 2 and 3 breakouts in total.

Cownditioning (Dec 2021)

Farmer John's N cows are very particular about the room temperature in their barn. Some cows like the temperature to be on the cooler side, while others prefer more warmth.

Farmer John's barn contains a sequence of N stalls, numbered $1 \dots N$, each containing a single cow. The i -th cow prefers the temperature of her stall to be p_i , and right now the temperature in her stall is t_i . In order to make sure every cow is comfortable, Farmer John installs a new air conditioning system that is controlled in a somewhat interesting way. He can send commands to the system telling it to either raise or lower the temperature in a consecutive series of stalls by 1 unit – for example "raise the temperature in stalls 5...8 by 1 unit". The series of stalls could be as short as just a single stall.

Please help Farmer John determine the minimum number of commands he needs to send his new air conditioning system so that every cow's stall is at the ideal temperature for its resident cow.

INPUT FORMAT (input arrives from the terminal / stdin):

- The first line of input contains N .
- The next line contains the N non-negative integers $p_1 \dots p_N$, separated by spaces.
- The final line contains the N non-negative integers $t_1 \dots t_N$.

OUTPUT FORMAT (print output to the terminal / stdout):

Please write a single integer as output containing the minimum number of commands Farmer John needs to use.

SAMPLE INPUT:

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

SAMPLE OUTPUT:

```
5
```

One optimal set of commands Farmer John can use might be the following:

```
Initial temperatures: 1 2 2 2 1
Increase stalls 2..5: 1 3 3 3 2
Increase stalls 2..5: 1 4 4 4 3
Increase stalls 2..5: 1 5 5 5 4
Decrease stalls 3..4: 1 5 4 4 4
Decrease stalls 3..4: 1 5 3 3 4
```

SCORING:

- Test cases 2-5 satisfy $N \leq 100$.
- Test cases 6-8 satisfy $N \leq 1000$.
- Test cases 9-10 satisfy $N \leq 100,000$.
- In test cases 1-6 and 9, temperature values are at most 100.
- In test cases 7-8 and 10, temperature values are at most 10,000.