⚔ The Sword Problem ⚔

Programming Assignment 1

At midnight you bought a powerful sword in your favorite online game. The price of this digital ingame item is changing every minute and you can sell it anytime for the current price. The price for each minute is fixed in advance on a secret list that only the game developers know—and you.

Though you like your new sword very much and it certainly will be your best friend in many dangerous quests, you would not hesitate any minute to sell it in order to buy a so-called epic item. Epic items are much better and their price does never change. But which epic item should you choose? Your choice depends on the item's fun factor and the time you would have to wait before you could buy it!

Whereas you know the fun factor by intuition, the waiting time has to be calculated. As a first step, you compute, for each epic item, the minimum gain that you have to make by selling your powerful sword in order to have enough ingame money to afford that epic item. The gain is the money that you would earn by selling your sword, that is, the difference between the price for which you would sell the sword and the price for which you bought the sword.

Now, given the secret list of the sword's prices for each one of the n minutes (starting at minute 0 when you bought the sword) and your precomputed list of the required gains for each one of the k epic items, write a function that, for each epic item, either computes the waiting time in minutes or, if the gain is never reached at any minute, outputs the value of n (note that there is no ambiguity in the output as the last price of the list refers to the (n-1)th minute). Your function should run in less than a minute as you don't want to miss any minute!

Example

Let [4, 6, 3, 8, 6, 10] be the list of the prices (n=6). Thus the sword costs

- 4 ingame coins at the zeroth minute,
- 6 ingame coins at the first minute,
- 3 ingame coins at the second minute,
- 8 ingame coins at the third minute,
- 6 ingame coins at the forth minute, and
- 10 ingame coins at the fifth minute.

Since we bought our sword at the zeroth minute, we bought it for 4 ingame coins.

Let [1, 7, 4] be the list of the precomputed gains (k=3).

- Thus, to buy the first epic item, our gain has to be at least 1. If we sell our sword at the zeroth minute, our gain is 4-4=0<1, which is too small. If we sell our sword at the first minute, our gain is 6-4=2≥1, which is enough. Thus, our output is 1 for the first epic item.

Send Feedback

- To buy the second epic item, our gain has to be at least 7. We observe that, for any minute, the gain is always less than 7. Therefore, we cannot obtain the gain of value 7 and, as defined above, we write n in the output which has the value 6.
- To buy the third epic item, our gain has to be at least 4. Observe that the third minute is the first one with sufficient gain (as 8-4=4). Thus, we write 3 in the output

Summarized, the correct output is [1, 6, 3].

Details

You can assume that

- n and k do not exceed 200.000,
- all the prices of the sword and all the required gains are given as integers between 1 and 10^9,
- the intended solution takes less than 10 seconds for such values on the Szkopuł webserver.

Formally, the input of your program is given via the standard input and your output has to be written into the standard output. The format of the input and the output is described at the end of this page. However, you don't need to worry about it. The following code (one in Python and one in C++) automatically reads the input and writes the output. You can use it as a base of your program! (But you are free to write everything from scratch yourself.)

[Example Code in Python] (https://ideone.com/w0UgXo) [Example Code in C++] (https://ideone.com/G7BNDC)

Rules

- The **deadline is on March 31** one minute before midnight (23:59 Polish time zone).
- Discussing the algorithm and the implementation with other people is not allowed.
- We will compare all solutions and take measures if we suspect plagiarism.
- Only C++ and Python are allowed.
- Using standard libraries is allowed—including numpy for Python (as long they work on the Szkopuł webserver).
- Upload your program on the Szkopuł webserver anytime before the deadline and how often you want. Shortly after each upload, you will see which tests you passed or failed and what your total score is.
- Yes, there is basically no limit on how often you can upload a solution.
- The first test (Test 0) is the example test from above. It does not give you any points.
- The inputs of the tests are secret.
- You get 0 points for a test if your program does not work, has the wrong output or is too slow.
- To get the full score (100 points), your program has to pass all the tests within the respective time limits. Note that this is only possible if the underlying algorithm has a sufficiently small time complexity. Use techniques as in the lecture to make your algorithm faster!

  There are two time limits: If you are below the first time limit, you get full points. If you are between the two time limits, you get less points. If you are above the second time limit, you get 0 points. The second time limit is twice

Send Feedback

as large as the first one. In Szkopuł, only the value of the second time limit is displayed. There is also a generous memory limit that you have to pass.

- **Activity points**: You can earn one activity point by submitting a test on Moodle. The test should contain an input and a corresponding output in the format as specified below. It will be published to help other students.
- In the code above, you find in the comments some code by which you can read the input from a given file (input.txt) and write the output to another file (your-output.txt) and compare it to the intended output (output.txt). By this, you can use the test data submitted by other students to test your code.

Format

The input format:

- The first line contains the numbers n and k.
- The next line contains the prices of the sword where the i-th price corresponds to the price at the i-th minute where 0≤i<n. Thus, the 0-th price is the price that you paid for the sword.
- The third line contains the k gains. The i-th gain is the minimum gain that you have to make by selling your sword in order to buy the i-th epic item.

The output format:

> The i-th line contains the waiting time for the i-th epic item. That is, if it contains the value j, then the j-th minute is the first minute when you could earn enough to buy the i-th epic item.

The example test from above has the following input and output: [Example Input] (https://pastebin.com/829CB91q) [Example Output] (https://pastebin.com/W0GL7CwS)

Powered by OIOIOI (https://www.github.com/sio2project/oioioi), from the SIO2 Project (http://sio2project.mimuw.edu.pl).

Send Feedback