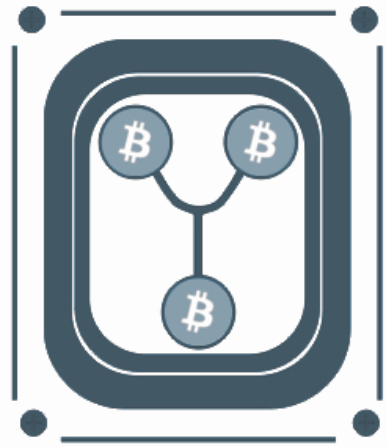
Challenge 1 - Bitcoin to the future

« Prev Next »



In the near future, after winning the Tuenti Challenge, you acquire almost supernatural programming skills. You write a Flux Capacitor Algorithm that is able to predict the exchange rate in euros of Bitcoin (the most used Internet currency) for a period of time. Now, you need to write a program that, given a list of Bitcoin exchange rates for a period of time and your initial budget, calculates the maximum number of euros that you are going to have at the end of the period. That is, the amount of euros you can earn plus your initial budget.

The exchange rate is always an integer, you can't buy a fraction of a bitcoin and you can sell or buy bitcoins at any moment, as many times as you want.

Input

First line contains the number of test cases, **T**, and **T** cases follow (each one in a different line). Each test case consists of one integer **N** ($1 \leq N \leq 100$), indicating your initial budget in euros. In the next line, there is a list of integers indicating the future value of Bitcoin at different times in a fixed period.

Output

The maximum amount of euros that you will have at the end, in a different line for each test case.

Example

Initial budget: 3
List of exchange rates: 1, 2, 10, 6

You buy at 1 euro per bitcoin and sell at 10 euros per bitcoin, so at the end you will have 30 euros.

Sample input

```
2
2
1 2 10 4 1 10
5
1 2 4 20 5 30 4 25 7
```

Sample output

```
200
3750
```

Submit & test your code

To test and submit code we provide a set of tools to help you. Download contest tools if you haven't already done that. You will then be able to test your solution to this challenge with the challenge tokens.

```
Challenge tokens: CHALLENGE_1, CHALLENGE_SUBMIT_1
```

To test your program

```
./test_challenge CHALLENGE_1 path/program
```

A nice output will tell you if your program got the right solution or not. You can try as many times as you need.

To test your program against the input provided in the submit phase

```
./test_challenge CHALLENGE_SUBMIT_1 path/program
```

During the submit phase, in some problems, we might give your program harder inputs. As with the test token, a nice output will tell you if your program got the right solution or not. You can try as many times as you need.

In the actual contest you first need to solve the test phase before submitting the code, you must provide the source code used to solve the challenge and you can only submit once (once your solution is submitted you won't be able to amend it to fix issues or make it faster).

If you have any doubts, please check the info section.

Problem stats

Completion time:	**10 percentile:** 1:06 h **90 percentile:** 43:51 h
Submit exec time:	**min:** 0.19 s **10 percentile:** 0.65 s **90 percentile:** 2.47 s **max:** 1205.00 s
Test tries:	**min:** 1 **10 percentile:** 1 **90 percentile:** 32 **max:** 179
# of completions:	517

« Prev Next »

Tweet about this! #TuentiChallenge3

Share Follow @Tuentieng