Stock Maximize

Your algorithms have become so good at predicting the market that you now know what the share price of Wooden Orange Toothpicks Inc. (WOT) will be for the next number of days.

Each day, you can either buy one share of WOT, sell any number of shares of WOT that you own, or not make any transaction at all. What is the maximum profit you can obtain with an optimum trading strategy?

For example, if you know that prices for the next two days are $prices = [1, 2]$, you should buy one share day one, and sell it day two for a profit of 1. If they are instead $prices = [2, 1]$, no profit can be made so you don't buy or sell stock those days.

Function Description

Complete the *stockmax* function in the editor below. It must return an integer that represents the maximum profit achievable.

stockmax has the following parameter(s):

- *prices*: an array of integers that represent predicted daily stock prices

Input Format

The first line contains the number of test cases t.

Each of the next t pairs of lines contain:
- The first line contains an integer n, the number of predicted prices for WOT.
- The next line contains n space-separated integers $prices[i]$, each a predicted stock price for day i.

Constraints

- $1 \le t \le 10$
- $1 \le n \le 50000$
- $1 \le prices[i] \le 100000$

Output Format

Output t lines, each containing the maximum profit which can be obtained for the corresponding test case.

Sample Input

```
3
3
5 3 2
3
1 2 100
4
1 3 1 2
```

Sample Output

```
0
197
3
```

Explanation

For the first case, you cannot obtain any profit because the share price never rises.

For the second case, you can buy one share on the first two days and sell both of them on the third day.

For the third case, you can buy one share on day 1, sell one on day 2, buy one share on day 3, and sell one share on day 4.