

DMOPC '20 Contest 1 P3 - Victor's Algorithm

Time limit: 0.6s **Memory limit:** 256M

I can sort in $\mathcal{O}(1)$ ~ Victor, 2019

Victor has invented a brand new sorting algorithm that promises an $\mathcal{O}(1)$ upper bound! However, there is a catch: this algorithm will only work on a **nice** sequence of numbers. Victor considers a sequence S of length N nice if it is first non-decreasing, then non-increasing. (i.e. S is nice if there exists some position i in the sequence such that $S_1 \leq S_2 \leq \dots \leq S_{i-1} \leq S_i \geq S_{i+1} \geq \dots \geq S_{N-1} \geq S_N$).

You have subsequently found that Victor's algorithm can be extended to any arbitrary sequence of numbers A by converting that sequence of numbers into a nice sequence A' , such that $A'_i \geq A_i$ for all i . Sadly, because counting is very difficult, you can only tell Victor to increment one element in the sequence by 1 once a second.

Write a program that finds the minimum number of seconds required to convert any given A into a valid A' !

Constraints

For all subtasks:

$$-10^9 \leq S_i \leq 10^9 \text{ for all } i.$$

Subtask 1 [20%]

The length of the sequence N is bounded by $1 \leq N \leq 1\,000$.

Subtask 2 [80%]

The length of the sequence N is bounded by $1 \leq N \leq 500\,000$.

Input Specification

The first line will contain N , the length of the sequence.

The next line will contain N space-separated integers S_1, S_2, \dots, S_N .

Output Specification

Output the minimum number of seconds required to convert the given sequence into a nice sequence.

Sample Input 1

```
10
-6 8 6 -1 3 -8 -6 -2 5 -2
```

Sample Output 1

39

Explanation of Sample Output 1

We can make this a nice sequence by incrementing S_4 by 6, S_5 by 2, S_6 by 13, S_7 by 11, and S_8 by 7. Since $6 + 2 + 13 + 11 + 7 = 39$, the answer is 39. It can be shown that this is the minimum possible answer.

Sample Input 2

5
1 2 3 4 5

Sample Output 2

0

Explanation of Sample Output 2

This is already a nice sequence.

Sample Input 3

7
7 6 5 -1 -3 -4 -5

Sample Output 3

0