

Cube towers

Ivancho has found a new way to pass the time - by building towers of cubes. He stacks a few cubes on top of each other to make a tower and then arranges the towers in a row.

When Ivancho looked at his last creation he was not very found of how chaotic the heights of his towers were. He decided to fix that by making each tower higher than the previous one in the row by exactly one cube. To do that he can add or remove a few cubes from the top of his towers. He is not limited in the amount of cubes he can use.

Ivancho saw that there are many ways to accomplish his desired order. Now he wonders what is the minimum number of cubes he has to add or remove. Help him with that, by writing a program **towers**, which finds that number.

Input:

The first row of the input file **towers.in** will contain a single integer **N** - the number of towers in the row.

The next row will contain **N** positive integers **H_i** - the corresponding heights of the towers in number of cubes.

Output:

The only row in the output file **towers.out** should contain a single integer **K** - the minimum number of cubes we must add or remove.

Limits:

$N \leq 5000$

$0 \leq H_i \leq 110\ 000$

Time limit: 0.3 sec.

Memory limit: 256 MiB.

Remark: Reading and writing to a file can be done using the appropriate statement. You can use the *freopen* statement from the standard library *cstdio* by adding the flowing two lines at the beginning of your main function:

```
freopen ("towers.in", "r", stdin);
freopen ("towers.out", "w" , stdout);
```

Number of preliminary tests: 4

Number of final tests: 10

Example test:

towers.in	towers.out
6 3 4 3 1 5 4	9

Explanation: The answer is achieved when we make the towers have a height of 1, 2, 3, 4, 5 and 6 cubes.