Sorting machine

Recently Ivancho found a new hobby of his - robotics. So he decided to create something and came up with a machine that sorts balls.

The machine is made from **N** boxes, arranged in a circle, in which the balls are put. In the middle there is a robot, which can take a ball from a random box and put it in a box neighboring it.

As Ivancho was playing with the machine, he realized that it has time come for school. So he decided to pack all the balls in a single box. But because he was short of time, he wanted to do that with as little moves as possible.

Help him by writing a program **sortmach**, which by a given number of boxes and amount of balls in every boxes, finds out the minimal number of moves required to place all the balls in a single box.

**Input**: The first row of the input file **sortmach.in** will contain a single number **N** - the number of boxes.

The next row will contain **N** integers **Ki** -the number of balls in the corresponding box.

**Output**: In the output file **sortmach.out** you should print a single integer **P** - the minimal number of moves.

**Limits**:

3 <= N < 100

0 <= Ki < 50

Time limit: 0.2 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 ("sortmach.in", "r", stdin);*

*freopen ("sortmach.out", "w" , stdout);*

Number of preliminary tests: 4

Number of final tests: 10

**Example tests**:

|  |  |
| --- | --- |
| sortmach.in | sortmach.out |
| 7  10 4 8 5 1 13 11 | 72 |

Output explanation:

The given amount of moves is accomplished by moving all balls in the last box (the one which originally has 11 balls).