

DWITE '11 R1 #2 - Penny Game

Time limit: 2.0s **Memory limit:** 64M

DWITE, October 2011, Problem 2

Extra change can be a big burden to carry around with you. So, you decide to leave your stacks of (identical) pennies you have at home. However, these stacks aren't of the same size, and look a little messy. You want to move pennies between these stacks so that all the stacks eventually have the same size; it is guaranteed that you are able to do this. You want to do this by making the smallest number of moves possible, where one move consists of taking one penny from a stack and adding it to another stack. For example, if you have three stacks with 4, 5, and 6 pennies, you have to move one penny from the third stack to the first stack so that they all end up with 5 pennies each. Write a program that helps you out with this task.

There will be 5 test cases. The first line of each test case is the number $1 \leq N \leq 50$, the number of stacks you have. The next N lines each contain a number $1 \leq M \leq 1000$, the number of pennies in each stack.

For each test case, your program should output a single number: the minimum number of moves required to make every stack have the same number of pennies.

Sample Input

```
5
1
5
3
9
7
```

Sample Output

```
6
```

Problem Resource: [DWITE](#)