

2002 Mid-Atlantic Regional Programming Contest

Welcome to the 2002 Programming Contest. Before you start the contest, please be aware of the following notes (this material will be repeated on the real problem set):

1. There is one (1) practice problem. Please submit solutions or request clarifications **for this problem only**. Unless you have a real question about the problem, please submit at most one clarification request, and at most two runs. It is important that everyone have a chance to see how the system works.
2. All solutions must read from standard input and write to standard output. In C this is scanf/printf, in C++ this is cin/cout, and in Java this is System.in/System.out. From your workstation you may test your program with an input file by redirecting input from a file:

```
program < file.in
```

3. Solutions for problems submitted for judging are called runs. Each run will be judged. Runs for each particular problem will be judged in the order they are received. However, it is possible that runs for different problems may be judged out of order. For example, you may submit a run for B followed by a run for C, but receive the response for C first. **DO NOT** request clarifications on when a response will be returned. If you have not received a response for a run within 90 minutes of submitting it, **you may have a runner ask the site judge to determine the cause of the delay. Under no circumstances should you ever issue a clarification request about a submission you have not received a judgment for.**

The judges will respond to your submission with one of the following responses. In the event that more than one response is applicable, the judges may respond with any of the applicable responses.

Response	Explanation
Correct	The run has been judged correct.
Incorrect Output	The program generated output that is not correct.
Incorrect Output Format	The program's output is not in the correct format.
Incomplete Output	The program failed to generate all required output.
Syntax Error	The program failed to compile on the judge's machine.
Run-Time Error	The program experienced a run-time error.
Time-Limit Exceeded	The program failed to complete within 30 seconds.

4. In the event that you feel a problem statement is ambiguous, you may request a clarification. Read the problem carefully before requesting a clarification. If the judges do not believe that you have discovered an ambiguity in the problem, you will receive the response, "The problem statement is not ambiguous, no clarification is necessary." If you receive this response, you should read the problem description more carefully. If you still feel there is an ambiguity, you will have to be more specific or descriptive of the ambiguity you have found. If the problem statement is ambiguous in specifying the correct output for particular input, please include that input data in the clarification request.

Practice Set

Additionally, you may submit a clarification request asking for the correct output for input you provide. The judges will seek to respond to these requests with the correct output. These clarification requests will be answered only when no clarifications regarding ambiguity are pending. The judges reserve the right to suspend responding to these requests during the contest.

If a clarification, including output for a given input, is issued during the contest, it will be broadcast to all teams.

5. The submission of abusive programs or clarification requests to the judges will be considered grounds for immediate disqualification.
6. Good luck, and HAVE FUN!!!

Practice Problem: Financial Management

Larry graduated this year and finally has a job. He's making a lot of money, but somehow never seems to have enough. Larry has decided that he needs to grab hold of his financial portfolio and solve his financing problems. The first step is to figure out what's been going on with his money. Larry has his bank account statements and wants to see how much money he has. Help Larry by writing a program to take his closing balance from each of the past twelve months and calculate his average account balance.

Input

The input will be twelve lines. Each line will contain the closing balance of his bank account for a particular month. Each number will be positive and displayed to the penny. No dollar sign will be included.

Output

The output will be a single number, the average (mean) of the closing balances for the twelve months. It will be rounded to the nearest penny, preceded immediately by a dollar sign, and followed by the end-of-line. There will be no other spaces or characters in the output.

Example

For the input:

```
100.00
489.12
12454.12
1234.10
823.05
109.20
5.27
1542.25
839.18
83.99
1295.01
1.75
```

The output should be:

```
$1581.42
```