Variadic functions in C ★

105 more points to get your gold badge!

Rank: **28761** | Points: **395/500**

C
C language
★★★★

Your Variadic functions in C submission got 50.00 points. Share Tweet

You are now 105 points away from the gold level for your c badge.

Try the next challenge | **Try a Random Challenge**

Problem Submissions Leaderboard Editorial

Variadic functions are functions which take a variable number of arguments. In C programming, a variadic function will contribute to the flexibility of the program that you are developing.

The declaration of a variadic function starts with the declaration of at least one named variable, and uses an ellipsis as the last parameter, e.g.

```
int printf(const char* format, ...);
```

In this problem, you will implement three variadic functions named $sum()$, $min()$ and $max()$ to calculate sums, minima, maxima of a variable number of arguments. The first argument passed to the variadic function is the count of the number of arguments, which is followed by the arguments themselves.

Input Format

- The first line of the input consists of an integer $number\_of\_test\_cases$.
- Each test case tests the logic of your code by sending a test implementation of 3, 5 and 10 elements respectively.
- You can test your code against sample/custom input.
- The error log prints the parameters which are passed to the test implementation. It also prints the sum, minimum element and maximum element corresponding to your code.

Constraints

$1 \ \leq \ number\_of\_test\_cases \ \leq 50$

$1 \leq element \leq 1000000$.

Output Format

"Correct Answer" is printed corresponding to each correct execution of a test implementation."Wrong Answer" is printed otherwise.

Sample Input 0

```
1
```

Sample Output 0

```
Correct Answer
Correct Answer
Correct Answer
```

Change Theme Language: C

```
    elements[1] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[2] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[3] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[4] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[5] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[6] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[7] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[8] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;
    elements[9] = rand() % (MAX_ELEMENT - MIN_ELEMENT + 1) + MIN_ELEMENT;

    fprintf(stderr, "Sending following ten elements:\n");
    for (int i = 0; i < 10; i++) {
        fprintf(stderr, "%d\n", elements[i]);
    }

    int elements_sum = sum(10, elements[0], elements[1], elements[2], elements[3], elements[4],
                           elements[5], elements[6], elements[7], elements[8], elements[9]);
    int minimum_element = min(10, elements[0], elements[1], elements[2], elements[3], elements[4],
                           elements[5], elements[6], elements[7], elements[8], elements[9]);
```

Line: 203 Col: 2

Upload Code as File

□ Test against custom input

Run Code

Submit Code

You have earned 50.00 points!

You are now 105 points away from the gold level for your c badge.

65%

395/500

C

C language

★★★★

Congratulations

You solved this challenge. Would you like to challenge your friends?

Next Challenge

✓ Test case 0

✓ Test case 1

✓ Test case 2

Compiler Message

```
Success
```

✓ Test case 3 🔒

✓ Test case 4 🔒

✓ Test case 5

🔒Hidden Test Case

Unlock this testcase for 5 hackos.

Unlock

Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature