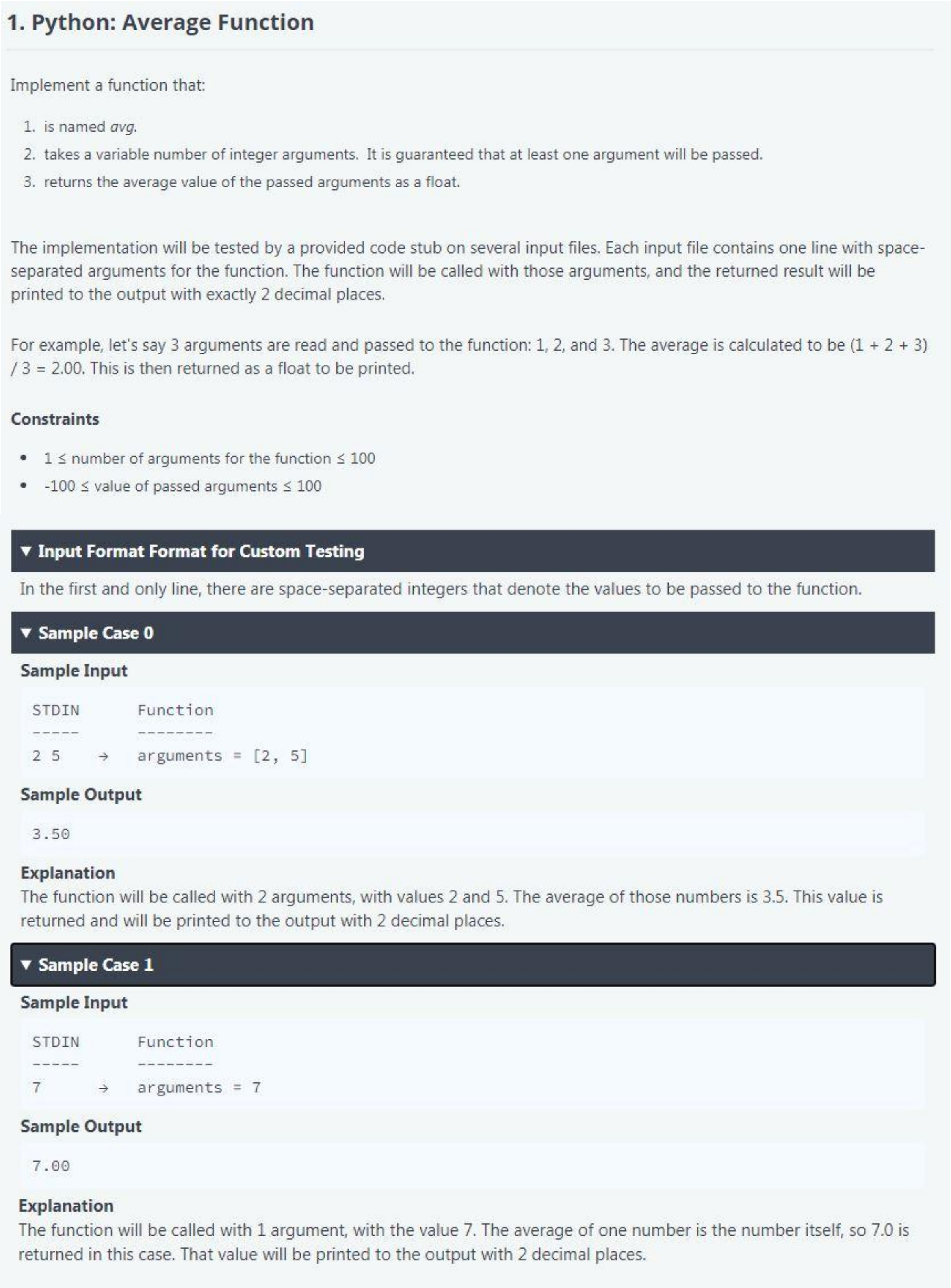

1. Python: Average Function

Implement a function that:

1. is named *avg*.
2. takes a variable number of integer arguments. It is guaranteed that at least one argument will be passed.
3. returns the average value of the passed arguments as a float.

The implementation will be tested by a provided code stub on several input files. Each input file contains one line with space-separated arguments for the function. The function will be called with those arguments, and the returned result will be printed to the output with exactly 2 decimal places.

For example, let's say 3 arguments are read and passed to the function: 1, 2, and 3. The average is calculated to be (1 + 2 + 3) / 3 = 2.00. This is then returned as a float to be printed.

Constraints

- 1 ≤ number of arguments for the function ≤ 100
- -100 ≤ value of passed arguments ≤ 100

▼ Input Format Format for Custom Testing

In the first and only line, there are space-separated integers that denote the values to be passed to the function.

▼ Sample Case 0

Sample Input

```
STDIN       Function
-----       --------
2 5    →    arguments = [2, 5]
```

Sample Output

```
3.50
```

Explanation

The function will be called with 2 arguments, with values 2 and 5. The average of those numbers is 3.5. This value is returned and will be printed to the output with 2 decimal places.

▼ Sample Case 1

Sample Input

```
STDIN       Function
-----       --------
7      →    arguments = 7
```

Sample Output

```
7.00
```

Explanation

The function will be called with 1 argument, with the value 7. The average of one number is the number itself, so 7.0 is returned in this case. That value will be printed to the output with 2 decimal places.