Software 1: Spring 1 Assessment feedback

Spring 2020

1 Summary

Description	mark awarded	mark scheme
Code correctness (1-5)	80	80
Docstring (1-2)	10	10
Code Quality	7	10
Final Mark	97	100

Table 1: Summary of your SOF1 Programming marks.

2 Detailed Feedback

2.1 Question 1 Docstring

Description	mark awarded	mark scheme
Triple double quote used	1	1
Convention are followed	1	1
parameters are described	2	2
returned value is described	1	1

2.2 Question 2 Docstring

Description	mark awarded	mark scheme
Convention are followed	1	1
parameters are described	1	1
returned value is described	1	1
raised exception is described	2	2

2.3 Code Quality

Description	Good	Average	Poor	Marks
Python: readability of the code, variable name, respect of conventions	Code runs and is free of syntax error, it is easy to read and convention are mostly respected. (3 marks)	Code runs and is free of syntax error, it is relatively easy to read and convention are not consistently respected. The author may have used global variables instead of local variable. (2 marks)	Poor programming skills in general. Code is difficult to read and may contain some syntax errors (1 mark)	3
Python: clarity and simplicity of the solution	The solutions are simple, elegant and adequate (4 marks)	The solutions are over complicated, difficult to understand and/or not efficient (3 marks)	Solutions are far from optimal and are not well thought. (2 mark)	4
Python: comments	Comments are used sparsely and where needed to support the code. (3 marks)	Comments are overused and describe trivial statements. (2 marks)	Poor quality comments and/or Comments are almost non-existent where needed (1 mark)	0
Code Quality				7

2.4 Code Correctness

Description	outcome	mark awarded	mark scheme
Question 1: Test that a value of 0 is returned when the speed is less than or equal to the speed limit.	Success!	2	2
Question 1: Test that fines are calculated correctly for speed below 90 mph.	Success!	4	4
Question 1: Test that fines are calculated correctly for speed of 90 mph or over. That is an additional penalty of 200 pounds should be added.	Success!	4	4
Question 2: Test if a distance of 0 is returned when the sequences are identical.	Success!	1	1
Question 2: Test that the function is case sensitive. That is the lower case and upper case characters are considered different.	Success!	2	2
Question 2: Test that the function calculates the right distance for different sequences.	Success!	5	5

Question 2: Test that the function raise a ValueError if the sequences provided are not comparable, that is do not have the same length.	Success!	2	2
Question 3: Test that given an input containing only one word, the correct result is returned.	Success!	2	2
Question 3: Test that given an input containing only one word and finishing with a white space, the correct result is returned.	Success!	1	1
Question 3: Test that given an input containing only one word and starting with a blank space, the correct result is returned. Watch out for the shift in the indices.	Success!	1	1
Question 3: Test that the correct output is returned when there is no duplicate words and a single blank space between words.	Success!	5	5
Question 3: Test that the correct output is returned when there is no duplicate words and multiple blank spaces between words.	Success!	1	1
Question 3: Test that the correct output is returned when there are duplicate words and multiple indices for the same word. Note that the function should be case incensitive, for example 'One' and 'one' are considered to be the same word.	Success!	5	5
Question 4: Test that the function returns 0 when the length of the rod is 0.	Success!	2	2
Question 4: Test that the function returns the correct value for the simple case where the value of a segment of a rod is equal to its length.	Success!	4	4
Question 4: Test that the function returns the maximum profit for a lenght that is strictly positive. Note that invalid inputs are not considered for this question, and therefore no test is provided to check input.	Success!	9	9
Question 5.i: Test that the object constructed has been assigned the correct values for each attributes when values are provided in the parameters.	Success!	4	4

Question 5.i: Test that the object constructed has been assigned the correct DEFAULT values for each attributes when no values are provided in parameters.	Success!	4	4
Question 5.i: Test that the constructor raises a ValueError if the minimum marks is greater than or equal to the maximum mark.	Success!	2	2
Question 5.ii: Test that adding an empty list of marks does not change the content of the attribute _marks.	Success!	2	2
Question 5.ii: Test that adding a mark outside the range of allowed marks, that is smaller than _min or greater than _max, raises a ValueError.	Success!	4	4
Question 5.ii: Test that adding marks in the valid range modified the content of _marks. Note it does not matter if you decided to sort the marks or not. The test provided ignore the order in which you store the marks.	Success!	4	4
Question 5.iii: Test that the get_distribution returns the correct output when the number of bin divides the range of marks. We test three different number of bins (2, 4, 5) for a range of marks of 20.	Success!	7	7
Question 5.iii: Test if the method get_distribution raises a ValueError if the range of marks is not a multiple of the number of bins.	Success!	3	3
Final Mark		80	80

3 Reading your feedback

- The first column describe the test considered for marks for a given question.
- If your code pass a test fully, you get the full mark for the test, given in the last column titled ”mark scheme”.
- If your code pass only a part of the test, you may be given some marks. You can find the mark awarded in the penultimate column titled ”mark awarded”.
- Each question may have more than one test. In that case partial marks will be awarded for each test you pass.

- The column titled ”outcome” may have several values:
  - **Success!** means you passed the given test and you are awarded all the marks for that test.
  - **Error! Some message!** means that the file containing the answer to the question has a some kind of error and the tests ran but an error occurred at runtime. A mark of 0 is given.
  - **Failed! Syntax Error!** means that the file containing the answer to the question has a syntax error and the tests cannot be ran. A mark of 0 is given. You must check for syntax errors before saving your solution.
  - **Failed! Not Implemented!** means that I could not import your function into the test suite. There might be several reasons for that. You did not save your code in the right file, you misspelled the name of the function or you did not answer the question at all. Another possibilities is that you wrote some tests yourself in your file, and they crash the program. You should only leave the definition of the function in your final submission. Either comment out all your tests or delete them. You should run the tests provided rather than implementing you own.
  - **Failed! some message**. In this case you failed a test, where the output returned by your function is not the one expected. The message try to give you a hint of the problem, for example which type of cases are failing. One common error is to print the result whereas the function is expected to return a value. Read carefully the question.