Practical Questions

CSCA08 Fall 2017

Question # 1

a) Write a function that takes in a string and returns a tuple that contains the number of upper and lower case letters in that string. Create test cases for this function using DocTest. (But you are already going to do this because you always follow the design recipe and create examples before you start coding. Right?).

b) Write a function that takes in 2 dictionaries and add all the key/values pairs in the second dictionary to the first. If a key appears in both dict1 and dict2, the new value of dict1 is a list containing all the values in dict1 and dict2. Create test cases to test whether this function is working properly. Remember to break up your “test-space” into areas and have test cases covering each area and all boundaries.

c) Write a function that takes in a string and returns the string with each letter being replaced by the next letter in the alphabet. Create your own Exception that will be raised if any data type except a string is passed into your function.

d) Write a function that takes in 2 integers, one for the numerator and one for the denominator of a fraction, and returns a string representation of the fraction in its simplest form. (4,8 -> “1/2”). The function should only ever throw exceptions that you create, try to cover every possible input with an appropriate exception type.

e) Write a function that takes in 2 strings and returns a tuple, where the first element is the number of positions in the strings that have the same character (case-insensitive) and the second element is the number of letters that the strings have in common but not in the same position. Write test cases for this function using black-box testing to see if the function is working properly.

f) Using the function you created above, find someone else in the practical, and try adversarial testing with them. You test their implementation, and they test yours. Can you find weaknesses in your partner’s implementation? How do you go about looking for them?

g) **(Challenge)**
Create a set of test cases that clearly and completely tests whether the triple cut function from the second assignment (A1) is working properly. Remember to keep in mind what triple cut is supposed to do and try to test all boundary cases as well as testing general cases.

Discussion Questions

Question # 2

In lecture, you have looked at the difference between the a few different forms of testing. One form you should be familiar with is DocTest, which uses the examples you create in your docstring to test your function. What are some of the benefits of using DocTest? What are some of the drawbacks? Do you think it is still worth the effort to create the examples in your docstring in order to use Doctest?

Question # 3

2 other forms of testing you learned about in lecture are black box and white (clear) box testing. How is the mindset used to approach each type of testing different? What is the basis behind the test cases for black box and white box testing? What are the benefits of using one over the other? Do you think you will usually use black or white box testing? Do you think there is a middle ground, like gray box testing?

Question # 4, The “Logic Question” **(Challenge)**

Proof: $1 = 2$

Let $a = b$

$$ab = a^2$$

$$ab - b^2 = a^2 - b^2$$

$$b(a - b) = (a + b)(a - b)$$

$$b = a + b$$

$$b = b + b$$

$$b = 2b$$

$$1 = 2$$

Explain what (if anything) is wrong with this proof