

CSC110 Fall 2021 Assignment 1: Written Questions

Nathaniel Yong

September 22, 2021

Part 1: Data and Comprehensions

1. Imagine this scenario...

- (a) List of string elements. The TO-DO notes would be represented as strings, and because there are multiple notes you would store it in a list. A list works best for storing these notes because it is mutable, and so you can add and remove the TO-DO notes as well as sort them.
- (b) Integer. The number of points would best be represented by an integer because it is a whole number.
- (c) Set of string elements. The names of the fruits would be represented as strings, and because there are multiple fruits you would store it in a set. A set works best because the fruit types are unique, and all the elements of a set are unique.
- (d) Boolean. Whether you won or not is a True or False statement, which is best represented by a boolean.
- (e) Dictionary with integer keys and string values. Each point would be represented by an integer. For example, the 1st point would be integer 1. This would be used as the dictionary key and would be mapped to a string value, the name of the player who won that particular point.

2. Exploring comprehensions.

- (a)
 - i. ['H', 'e', 'l', 'l', 'o', ' ', 'D', 'a', 'v', 'i', 'd']
 - ii. List of strings elements.
- (b)
 - i. {'e', 'a', 'i', 'd', ' ', 'v', 'l', 'o', 'H', 'D'}
 - ii. Set of string elements.
 - iii. Part (a) is a list while part (b) is a set. Both collections have strings as elements. Part (b) has fewer elements because duplicate strings have been removed from the set.
- (c) [False, True, False, False, True, False, False, True, False, False, False]
This expression returns a list of booleans. Each boolean in the list corresponds to a character in the original string of the same index. If the boolean is True, then the original character was a vowel. If the boolean is false, then the original character was not a vowel.
- (d) The first “in” used at “c in vowels”, is used to check if a value is present in a collection. In this case, it is being used to check whether the value c is found in the collection vowels. The second “in” used at “for c in ‘Hello world’”, is being used to iterate through elements in a collection, in conjunction with the keyword “for”. In this case, every character in the string ‘Hello world’ is iterated over by using keywords “for” and “in”.

Part 2: Programming Exercises

Complete this part in the provided `a1_part2.py` starter file. Do **not** include your solution in this file.

Part 3: Pytest Debugging Exercise

- 1. `test_section_average_all_grades_equal` passed.
`test_class_average_no_grades_equal` and `test_class_average_many_students` both failed.

2. `test_section_average_no_grades_equal` failed because the 'grades' list in the test function contains a list of strings instead of a list of floats. This resulted in a `TypeError` when trying to multiply the assignment grades with a float. Removing the quotations fixes this error.

`test_class_average_many_students` failed because when the `student_average` function is called, 'grades' list is sorted in ascending order while the 'weights' list is ordered in descending order. This means that the grades are being multiplied by the weightings in the wrong order. The lowest grades are being weighted the highest and vice versa. Changing the order of the weights list fixes this error.

3. `test_section_average_all_grades_equal` passed because all the students in the 'grades' list in the test function have the same grade for all three assignments, so the order of the assignment grades does not matter when multiplying it with the weightings. Therefore, the incorrect ordering of the 'weights' list does not affect the output of this test as it did with the other two tests.

Part 4: Adding Noise to an Image

Complete this part in the provided `a1.part4.py` starter file. Do **not** include your solution in this file.

Part 5: Removing Noise From an Image

Implementation

Complete this part in the provided `a1.part5.py` starter file. Do **not** include your solution in this file.

Exploration

1. The image is generally the same, although it appears slightly blurrier. The sharpness of the image is a lot lower, and it's harder to make out certain objects. This is probably because the median filter changed many pixels from their original value since it was not the median RGB value in their window. This results in a new image with a lot less sharpness and contrast.
2. The lower the k value, the noisier the image will be, because there's a higher probability of a pixel becoming noisy. When the image is noisier, the median filter does a poorer job of eliminating salt and pepper noise. On k values higher than 5, the median filter generally does a good job of filtering out most noise. However, on k values 5 and lower, a lot of noisiness persists even after using the median filter.
3. The mean filter would be worse because the RGB values of the noisy pixels would be included in the average. These outliers would result in a new pixel not representative of the original pixel. With the median filter, the salt and pepper noise do not usually affect the median.