Practical Questions

CSCA08 Fall 2017 – Week 7

Question # 1

Remember to close your files after you have used them, especially when writing to a file.

a) Open the file “my_file.txt” and print out everything in that file on a single line (you have to create a file called my_file.txt before you can open it)

b) Open a file and print out each line of the file with the newline character removed and a semicolon (;) added after each line

c) Open the file “my_test_file.txt” and write your name and shoe size to that file in 2 separate lines.

d) Open the file “my_test_file.txt” and write your name and height to the file in separate lines

e) Open the file “my_test_file.txt” and add your weight[1] to the bottom of the file without erasing what was written before.

f) Open the file “my_test_file.txt” and edit the file so all the information you entered has labels before it (“Name:” and your name, “Height:” and your height, “Weight:” and your weight all one separate lines). This should work for anyone’s file (i.e., your values shouldn’t be hard-coded, they should be read from the file)

g) Create a function that takes in a list of string and writes those string to a file, with each line containing 2 strings each surrounded by braces (or mustache brackets or curly brackets) and separated by a underscore

h) Write a program that will continually prompt the user to enter the name, height, weight, eye color, and hair color for different people until “EXIT” is entered. Every time these 5 parameters are entered, store them all on one line of a file, separated by tabs, under headers at the top that describe what each piece of information is (Name, Height, Weight, eye color, hair color).

i) Modify and improve the mad_libs.py file posted on the lecture notes site. Have fun with it. Work in a group, and post your best results on Piazza.

[1] The TAs won’t withhold points if your height/weight/shoe size isn’t accurate. Feel free to lie. But if you’re going to lie, have fun with it. At least say you’re 23m tall and weigh 1.3kg

Question # 2

Trace the following code using the memory model

a)
```
my_list = [1, 2, 3, 4]
print(my_list)
my_list[1] = 1
print(my_list)
```

b)
```
my_list = [“a”, “B”, “C”, ”d”]
print(my_list)
my_list[1] = [1,2]
print(my_list)
```

c)
```
my_list = [ 4, 3, 2, 1]
print(my_list)
my_list[1:3] = [True, False]
print(my_list)
my_list[0] = my_list[:]
print(my_list)
```

d)
```
my_list =[“MATA31”,”CSCA08”,”CSCA67”,”MATA37”,”CSCA08”]
print(my_list)
my_list[0] = my_list[:]
my_list[4] = my_list[:]
print(my_list)
```

e)
```
my_list = [“True”,[1, 2, 3, 4], 2, 9]
print(my_list)
my_list[1][3] = bool(my_list[0])
print(my_list)
```

f) my_func(my_list):
    my_list[0:(len(my_list)//2)] = my_list[(len(my_list)//2):len(my_list)]

```
list1 = [1, 2, 3, 4]
list2 = [9, 8, 7, 6, 5]
print(list1,list2)
print(list2)
my_func(list1)
my_func(list2)
print(list1,list2)
```

g) Write a function that takes in a list and changes each elements in the list to a negative if the element at that location was a positive number, positive if the number at that location was a negative number, and “SOMethiNG” for anything else
example: input my_list: [1, -2 True, “a”]
output:my_list:[-1, 2 , “SOMethiNG”, “SOMethiNG”]

Discussion Questions

Question # 3

You have learned in lecture that when writing to a file, either you can overwrite a file completely and start adding lines to an empty file or you can append to the end of an already created text file, adding new lines starting from the last line of the file. However, is there a way to append text to the end of a line or the beginning? Can you add a period to the end of every line or add 3 whitespaces to the beginning of every line without reading the whole file and rewriting every line with the changes you want made?

Question # 4

Lists are mutable but strings are not. Why? There’s no reason that the developers of Python couldn’t have made them both mutable or both immutable. Why do you think they chose to implement them the way they did? If you were creating your own programming language, would you make the same decision? Why or why not?

Question # 5, The “Logic Question” **(Challenge)**

I’ve developed a new board game. You start with your piece on square 0, and move forward with the roll of a die (i.e., you roll a number between 1 and 6, and move forward that many spaces). The board has 100 spaces in total.

The trick to this game is that you don’t have to get to the end of the board to win, you just have to land on a “win” square. And you get to choose which square will be the winner before the game starts.

1. *Which square should you choose?*
2. *What if you get to choose 2 squares that will win the game for you? What about 3?*
3. *What if instead you had a ‘lose’ square, and had to try to avoid it? Where would you place that? What if you had 2 or 3 of them?*

(Bonus)

What are your odds of winning/losing in the above scenarios?

(Bonus #2)

Write a program that plays the game, simulates many games, and tells you how many you won/lost for various choices of win/lose square positions