CMSC 201 Section 60

Fall 2024

Sample Exam for Midterm Exam #2

Test Date:

November 8, 2023

Test Value:

130 points

Instructions:

You will have 75 minutes to complete this exam, starting from the time you are told to begin. As with Exam 1, this in-class test is open-book, open-note. You may NOT collaborate with any other person during the exam - either fellow students in the room, or anyone outside the room. You may NOT use GhatGPT, Bard, or any other AI-based aide.

Section 1: Multiple Choice and True/False - 45 points

There are 15 questions in this section; each is worth 3 points. No partial credit will be given for questions in this section.

1. Which of the following is true about the recursive case in a recursive solution in Python?
    a. There must be one and only one recursive case
    b. The recursive case must result in a simpler problem. That is, it must get you closer to a base case.
    c. A and B are both true.
    d. None of the above is true.
2. If I have a 2-dimensional list, l, and refer to a cell as l[x][y], does x represent the column number or the row number?
    a. The column number
    b. The row number
    c. Neither of these; that's not a valid way to reference a cell in a 2D list in Python.
    d. Sorry, I don't have a good fourth option. Don't pick this.
3. True or False: lists, like dictionaries, are unordered data structures in Python
    a. True

b. False

4. Suppose that we have the string

```
s = "antidisestablishmentarianism"
```

What does the following code produce?

```
y = []
for i in range(len(s)):
    if i%2 == 0:
        y.append(s[i])
x = ''.join(y)
print(x)
```

a. atdssalsmnains

b. niietbihetraim

c. Antidisestablishmentarianism

d. None of the above; the program crashes due to an error

5. Suppose I have the following dictionary:

```
schools = {
    "UMBC":"Retrievers",
    "UMCP": "Terrapins",
    "JHU": "Blue Jays"
  }
```

What will be printed if I try to get a value from the dictionary using the statement:

```
print(schools.get("MSU", -1))
```

a. Nothing; the program will crash because "MSU" is not a key in the dictionary
b. "Bears"
c. None (that is, the special value None of type NoneType)
d. None of the above is correct

6. Suppose we have the dictionary `schools` as defined in problem 5. Now we execute the statement

```
Y = schools.pop("MSU")
print(Y)
```

What is printed as the result of these statements?

a. None (that is, the special value None of type NoneType)
b. "Bears"
c. Nothing; the program will crash because "MSU" is not a key in the dictionary
d. None of the above is correct

7. True or false: a list variable is a valid key for a Python dictionary
   a. True
   b. False

8. True or false: you must have exactly one base case in a recursive Python function
   a. True
   b. False

9. What's wrong with the following Python code?

```
with open("my_results.txt","w") as outfile:
    a = 4
    b = 3.14159
    outfile.write(a, b)
```

   a. There's nothing wrong with that code; it will work as presented
   b. The file "my_results.txt" doesn't exist, so that will cause an error
   c. Only strings can be written to files; not ints or floats. So you can't write a and b without first casting them to strings
   d. All of the above are correct; this code is a mess

10. True or false: every problem that can be solved recursively also has an iterative solution, but the reverse is not true
   a. True
   b. False

11. The Python hex() function converts decimal numbers to hexadecimal numbers. What is the result of hex(255)?
   a. ff
   b. 0xff
   c. 0x100
   d. None of the above is correct

12. What happens when you try to open a file for reading in Python with

```
with open ("data.txt","r") as infile:
      info = infile.read()
```

And there is no file called “data.txt” on your computer?

a. Python creates a new empty file for you, and info contains an empty string
b. Your program crashes and you get an error message
c. Both of the above are true
d. None of the above are true

13. Why should you not access or change a global variable within a function?

a. It is poor coding style
b. It can easily result in bugs that are difficult or impossible to find and fix
c. You never need to do that; if you need to access the variable within the function you can simply pass it as an argument/parameter
d. All of the above are true

14. If you have a 2D list, every row must contain the same number of elements

a. True
b. False

15. Suppose you have the following recursive Python function that determines whether or not a string is a palindrome

```
def recursive_palindrome(word):
    if len(word) <= 1:
        return True
    elif word[0] != word[-1]:
        return False
    else:
        return (recursive_palindrome(word[1:-1]))
```

How many base cases are in this function?

a. Three
b. Zero
c. One
d. None of the above is correct

Section 2 - Short Answer - 45 points

There are nine questions in this section worth 5 points each. Partial credit will be given for questions in this section, so please make an attempt to answer each. As with Midterm 1, “short answer” means you really shouldn’t take more than one or two sentences or a few lines of code to answer each question.

16. Write Python code to create a dictionary called “baltimore_colleges” with key/value pairs as follows:

Key: “location” Value: [“Baltimore CIty”, “Baltimore County”]
Key: “schools” value: [“UMBC”, “UMB”, “TSU”, “MSU”, “CSU”, “JHU”]

```
baltimore_colleges = { “location”:[“Baltimore City”, “Baltimore
County”],
         “schools”:[“UMBC”, “UMB”, “TSU”, “MSU” “CSU”, “JHU”]}
```

17. If you open a file in append mode (“a” mode) in Python and the file does not already exist, what happens?

The file is created, and is empty

18. Suppose we have the following dictionary defined:

```
canadian_hockey_teams =
      { "montreal":"canadiens",
      "winnipeg":"jets",
      "edmonton":"oilers",
      "vancouver": "canucks",
      "toronto":"maple leafs",
      "Calgary":"flames"
      }
```

Let’s suppose the Winnipeg Jets move to Saskatoon and become the “Wheatfarmers.” Write Python code to remove the Winnipeg entry from this dictionary and then add a Saskatoon key with a value “Wheatfarmers”

canadian_hockey_teams.pop(“winnipeg”)
canadian_hockey_teams[“Saskatoon”] = “Wheatfarmers”

19. In Python file I/O, what’s the difference between the readline () method and the readlines () method?

readline() reads one line at a time. readlines() reads the entire file into a list, with each list element being one line of text

20. Suppose you have the following Python program:

```
def check_input(child):
      child *= 4
      return
```

```
if __name__ == "__main__":
      d = { "name":"Williams",
            "occupation":"plumber",
            "children":2 }
      check_input(d[“children”])
      print (d)
```

When this is executed, the result is

```
{‘name’:’Williams’, ‘occupation’:’plumber’, ‘children’:2}
```

Explain, using the concept of mutable and immutable types, why the number of children did not change.

Since the argument/parameter is an integer, and that’s immutable, it can’t be changed in the function.

21. Explain the difference between the pop() method used for lists and the pop() method used for dictionaries. It is sufficient to describe what each does and point out what’s different.

pop() for lists removes an entry by its index - its place in the list. pop() for dictionaries removes a key value pair for the specified key.

22. Suppose I have a Python file, ‘data.txt’ that has the following content, with the values on each line separated by tabs:

```
H     He    Li
Be    B     C
N     O     F
```

I want to read this data into a 2-dimensional list, ‘elements’ with the values on the first line of the file, the values on the second of the file in the second row, and the values on the third line of the file in the third row. I start with the following code:

```
with  open (‘data.txt’, “r”) as infile:
      data = infile.read()
```

Write Python code that turns the string data into the 2D list that I want

```
elements  = data.split(“\n”)
for in range(len(elements)):
      elements[i] = elements[i].split(“\t”)
```

23. Write the Python Code to generate a 2-dimensional list, curves, that gives you the cubes of the integers from 0 through 19. Make sure there are 4 rows and 5 columns. The first row should have 0 cubed, 1 cubed, 2 cubed, 3 cubed and 4 cubed; the second row has 5 cubed through 9 cubed, and so on.

```
Curves = []
For i in range(4):
      New_row = []
      For j in range(5):
            new_roe.append((i * 5 + j)**3)
      curves.append(new_row)
```

24. Every recursive function could be implemented as an iterative function, and the recursive function will always be more “expensive” in terms of memory and computation requirements. Given that, why would you implement a function using recursion?

It’s sometimes easier for the programmer to understand.

Section 3: Programming - 40 points

Two questions, worth 20 points each. Partial credit WILL be given for answers in this section that are substantially but not completely correct.

25. Write a recursive Python function that takes a negative integer and returns the product of all integers between that number and -1. That is, if your function starts with -5, it should ultimately return -5 * -4 *-3 * -2 * -1 = -120. If it starts with -2, it should return -2 * -1 = 2. You don’t need to write the entire program. Just write the recursive function.

```
Def fact(number):
        If number == -1:
                Return -1
        Else:
                return(number * fact(number+1)
```

26. I've got a data file, states.txt, that contains the names of the 50 United States in alphabetical order. State names are separated by semi-colons. That is, the file looks like Alabama;Alaska;Arizona;Arkansas and so on. Write a Python program that reads in this file and separates it into a list, with each state being an element of the list. Then write a loop that goes through this list and prints out the name of each state that begins with the letter "N".

```
With open ("states.txt", "r") as infile:
        States = infile.read()
        State_list = states.split(";")
        For state in state_list:
                If state[0] == "N":
                        Print (state)
```