

CSCI 127: Joy and Beauty of Data

Summer 2021

Midterm Exam

Thursday, May 27, 2021
Instructor: Reese Pearsall

Print your name: _____

Print your Net ID: _____

By submitting this exam, I make the following truthful statements:

- I have not received, I have not given, nor will I give or receive, any assistance to another student taking this exam, nor have I discussed this exam with past students of this course.
- I will not use any non-instructor approved materials to assist me on this exam.
- I will not plagiarize someone else's work and turn it in as my own.
- I understand that acts of academic dishonesty may be penalized to the full extent allowed by the Montana State University Student Code of Conduct, including receiving a failing grade for the exam and/or course. I recognize that I am responsible for understanding the provisions of the Montana State University Student Code of Conduct as they relate to this academic exercise.

Submission Instructions

This exam will be due to D2L by 8:00 AM on Friday, May 28th, 2021. This is a **hard** deadline. Late submissions will receive an automatic 0 for this exam..

The first four pages (**This page and question 1**) will be submitted to the appropriate D2L dropbox as a PDF

- You may print out the exam, fill out the first four pages by hand, scan it/take a picture, and submit as a PDF.
- Or you may fill out the first four pages virtually (i.e. in Word, PDF editor, etc) and save your answers as a PDF.

Problems 2, 3, and 4 will be submitted to a separate D2L dropbox. You will develop a python solution for each of those problems. You may have one single .py file for your all your answers, or you may have a separate .py file for each of those problems.

Make sure you name your files accordingly and place your name at the top of each of your .py file in a comment.

You may use notes, lecture slides, in class examples, previous labs/programs, the textbook, lecture recordings, and your computer on this exam.

You CAN NOT use search engines to access external resources (Youtube, Stack Overflow, W3Schools, etc), Discord, or other students (previous or current).

Be sure to attempt every problem. Even if you cannot fully solve one of the programming problems, you should try to get as much done as you can. I will give out as much partial credit as I can.

Read each question carefully and make sure that you answer everything asked for.

Question 1 (Multiple Choice/Short Answer) (20 pts)

1.1) Which of the following is **not** a Python data type?

- a. String (str)
- b. List (list)
- c. Integer (int)
- d. Function (def)

1.2) Consider the following Python statements. What will be the value of a at the end of the program?

```
a = 12 % 6  
a = a + 1  
a = a ** 2
```

- a. 0
- b. 1
- c. 2
- d. 3

1.3) What will be the output of the python program given below?

```
nums = ["5", "12", "7", "14"]  
if 12 in nums:  
    print("Success!")  
else:  
    print("Failure!")
```

- a. Success!
- b. Failure!
- c. True
- d. False

1.4) What are the two requirements for recursion?

- 1.
- 2.

1.5) The following code will produce an error. Why? (You can assume there are **no** indentation errors)

```
def my_function(num1, num2, num3):  
    print(num1, num2, num3)
```

```
my_function(22, 15, "Hello", 1.5)
```

- a. Because the function doesn't return anything.
- b. Because when the function is called, it passes the function too many input parameters/arguments.
- c. Because when the function is called, it passes different data types (two different integers, a string, and a float).
- d. All of the above are true

1.6) Consider the following list of information

```
states = [ "Montana", "Idaho", "Wyoming", "Colorado", "Utah", "Nevada" ]
```

Complete the Python print statement below using list slicing to produce the following output:

```
['Wyoming', 'Colorado', 'Utah']
```

```
#put your answer inside the print statement below  
print( )
```

1.7) What will be the output of the Python program given below? (You can assume there are **no** indentation errors)

```
def func1(x,y):  
    x += 2
```

```
def main():  
    x = 3  
    y = 1  
    func1(x,y)  
    print(x)
```

```
main()
```

- a. 2
- b. 3
- c. 5
- d. The program will result in an error

1.8) True or False: The following code will produce an error: (You can assume there are **no** indentation errors)

```
x = 7
if(x = 3 or x = 7):
    print("Hello there!")
```

- a. True
- b. False

1.9) Which of the following Python **range()** statements would produce the following sequence of numbers:

[0,3,6,9,12]

- a. range(0,15,3)
- b. range(0,12)
- c. range(0,12,3)
- d. range(12,0,3)

1.10) What will be the output of the python program given below?

```
s = "Computer"
s[3] = "b"
print(s)
```

- a. Computer
- b. Combuter
- c. b
- d. The program will not run due to an error

Question 2 (20 pts)

Write a Python program that will ask the user for a number of cents (5¢, 99¢, 23¢, etc). The program should calculate and print out the number of coins needed to create that amount of cents using **ONLY** pennies (1¢) and dimes (10¢). Your program should minimize the number of coins used.

Below shows three different example runs of how the output of your program should look

```
Enter amount of cents: 23
Coins needed: 2 dimes and 3 pennies
```

```
Enter amount of cents: 90
Coins needed: 9 dimes and 0 pennies
```

```
Enter amount of cents: 7
Coins needed: 0 dimes and 7 pennies
```

There is no starting code. You will develop your solution from scratch, but your output should look the same as the sample output above

Question 3 (25 pts)

In Fall of 2020, Montana State University implemented several different policies and guidelines to help prevent the spread of Covid-19. One of the policies was to limit the number of students that could attend lectures in person.

CSCI 107 is a class where lectures occurred 3 days during the week: Monday, Wednesday and Friday. The professor of the class implemented the following rule for in person attendance:

- Students with last names starting with **A – H** (inclusive) can attend lectures on Monday
- Students with last names starting with **I—Q** (inclusive) can attend lectures on Wednesday
- Student with last names starting with **R—Z** (inclusive) can attend lectures on Friday

Using the starting code below, write a Python function (`which_day`) that will take in the last name of a student. The function should determine and print out which day (Monday, Wednesday or Friday) the student can attend in person lectures.

Below are three example runs of how the output of your program should look. You can assume that the user will enter a last name that begins with a valid character (*uppercase or lowercase*)

```
Enter your last name: Smith
You should attend lecture in person on Friday
```

```
Enter your last name: harmon
You should attend lecture in person on Monday
```

```
Enter your last name: Johnson
You should attend lecture in person on Wednesday
```

Your function should go above this comment. Do not change any of the code below:

```
last_name = input("Enter your last name: ")
which_day(last_name)
```

Question 4 (35 pts)

In CSCI 127, the **lowest** lab grade will be dropped at the end of the semester. Consider a nested list that contains information about a student's grade for each lab

For example, consider the following nested list: `[["Lab 1", 80], ["Lab 2", 85], ["Lab 3", 75], ["Lab 4", 100]]`

80% was the score the student received for Lab 1, 85% for Lab 2, 75% for Lab 3, etc.

Write a python function that will take in a nested list of this information that will determine which Lab Assignment will be dropped. If there is a tie for the lowest lab grade, then you can just use the lowest lab grade that occurred first in the list.

Given the following starting code:

```
# Your function will go above this comment. You should not modify anything below.

# [ [lab1, lab1-score], [lab2, lab2-score], [lab3, lab3-score]... ]
labs = [["Lab 1", 80], ["Lab 2", 85], ["Lab 3", 75], ["Lab 4", 100]]

answer = find_lowest_grade(labs)
print("At the end of the semester,", answer, "will be dropped")
```

Your program should produce the following output:

```
At the end of the semester, Lab 3 will be dropped.
```

Important Note: Your function should work with **any** nested list of a similar format, not just the given list

For example, if `labs = [["Lab 3", 100], ["Lab 1", 50], ["Lab 2", 75]]`

Then your same program should produce the following output:

```
At the end of the semester, Lab 1 will be dropped.
```