CSCI 127: Joy and Beauty of Data
Summer 2021
Final Exam

Thursday, June 17, 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, June 18th, 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, 4, 5 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 may also use the official documentation for numpy, matplotlib, and pandas**

You CAN NOT use search engines to access external resources (Youtube, Stack Overflow, W3Schools, etc), Discord, or other students (previous or current). If you need clarification on a certain question, then you should ask Reese. Reese will make a public announcement to clarify to the entire class if needed.

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) (20 pts)

1.1) Match the following Python statements with the correct description

____ Creating a Dictionary A. `hello = [ ]`
____ Creating a List B. `hi = hello()`
____ Creating an Object C. `hi = Hello()`
____ Calling a function D. `hello = { }`

1.2)The following code will result in errors. . In your own words, **briefly** explain (1) why the error is happening and (2) how you could modify the code to eliminate the error.

```
class Bike():
   def __init__(self,color,wheels):
        self.color = color
        self.wheels = wheels
   def get_color(self):
        return color
   def get_wheels(self):
        return wheels

my_bike = Bike("red",2)
print(my_bike.get_color())
```

Write your answer in this box

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

```
class Car():
   def __init__(self,make,model,color):
        self.make = make
        self.model = model
        self.color = color
   def str(self):
        answer = ""
        answer += self.color + " " + self.make + " " + self.model
        return answer

car1 = Car("Dodge","Charger","Black")
print(car1)
```

a. Black Dodge Charger
b. Dodge Charger Black
c. It will print out the location of the object in memory
d. The program will result in an error

1.4) The following code will produce an error. In your own words, **briefly** explain (1) why the error is happening and (2) how you could modify the code to eliminate the error.

```
import numpy as np
array = np.zeros(5,dtype=int)

array[2] = "Hello There"
print(array[2])
```

Write your answer in this box

1.5) Consider the following code that creates an array and reshapes it into some **X** by **Y** array. Which of the following provides valid values for X and Y that would not cause errors?

```
import numpy as np
a = np.array([1,2,3,4,5,6])
a = a.reshape(X,Y)
print(a)
```

a. X = 1, Y =5
b. X = 3, Y = 2
c. X = 2, Y = 2
d. X = 3, Y = 3

1.6) True/False: `print(names[0])` will always print out the first key-value pair of a dictionary called `names`

a. True
b. False

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

```
capitals = {"Montana":"Helena", "Idaho":"Boise", "Utah":"Salt Lake City"}

capitals["Montana"] += capitals["Utah"]

print(capitals["Montana"])
```

a. Helena
b. HelenaSalt Lake City
c. Salt Lake City
d. Utah

1.8) What is the Python method name for the constructor in a class?

```
a. self
b. __str__
c. __init__
d. class
```

1.9) Suppose you want to write out to a file named “answer.txt”. What is the correct python statement?

a. file = open(answer.txt,w)
b. f = open("answer.txt","r")
c. my_file = open("answers.txt")
d. output_file = open("answer.txt","w")

1.10) Free question (There is no wrong answer) 😊

What was your favorite lab or program in the class?

Write your answer in this box

Take a deep breath…

I believe in you!

Good luck!

Question 2 (20 pts)

Consider a dictionary that contains enrollment information about computer science courses at MSU. The keys of the dictionary are the course subject and number (such as “CSCI 127”). The values of the dictionary are a `list` that contain information about the **instructor**, **location**, and **number of students enrolled**.

```
courses = {
             #instructor     #location  #number of enrolled students
"CSCI 127": ["Reese Pearsall", "Barnard 103", 32],
"CSCI 107": ["John Paxton", "Leon Johnson 339", 71],
"CSCI 460": ["Travis Peters", "Barnard 108", 65],
"CSCI 232": ["Brendan Mumey", "Reid 101", 95],
"CSCI 347": ["Veronika Neeley", "Norm Asbjornson 201", 80],
"CSCI 538": ["Binhai Zhu", "Barnard 323", 14],
"CSCI 112": ["Mary Ann Cummings", "Barnard 103", 55],
"CSCI 366": ["Carson Gross", "Reid 202", 107],
"CSCI 132": ["Daniel DeFrance", "Norm Asbjornson 165", 121]
}
```

Using the starting code below, supply the missing function that will sum up the total number of students enrolled in all **upper division courses** (highlighted in red above). An upper division course is defined as a course number that is 300 or greater. For example, CSCI 347 is an upper division course, but CSCI 232 is *not* an upper division course.

```
def main():

    courses = {
                 #instructor     #location  #number of enrolled students
        "CSCI 127": ["Reese Pearsall", "Barnard 103", 32],
        "CSCI 107": ["John Paxton", "Leon Johnson 339", 71],
        "CSCI 460": ["Travis Peters", "Barnard 108", 65],
        "CSCI 232": ["Brendan Mumey", "Reid 101", 95],
        "CSCI 347": ["Veronika Neeley", "Norm Asbjornson 201", 80],
        "CSCI 538": ["Binhai Zhu", "Barnard 323", 14],
        "CSCI 112": ["Mary Ann Cummings", "Barnard 103", 55],
        "CSCI 366": ["Carson Gross", "Reid 202", 107],
        "CSCI 132": ["Daniel DeFrance", "Norm Asbjornson 165", 121]
        }

    calculate_student_enrollment(courses)


main()
```

When the program is run, it should produce the following output

```
The total number of students enrolled in upper division classes is 266
```

Note: Your solution should work for **any** dictionary that follow a similar format, not just this example. I will test your solution on a slightly different dictionary that is structured the same way.

Hint: The course number will always be the last 3 characters of the dictionary keys

Question 3 (20 pts)

Using the given code below, supply the missing class, methods, and functions using object-oriented programming. Each Musician has a *name*, *number of albums*, *number of Grammy awards*, and *their most popular song*. Consider the following starting code:

```
def main():
                    #name, # of albums, #number of grammys, most popular song
    elton = Musician("Elton John",30,5,"Your Song")
    print(elton)

    beyonce = Musician("Beyonce",6,28,"Single Ladies (Put a Ring on It)")
    print(beyonce)

    billie = Musician("Billie Eilish",1,7,"Bad Guy")
    print(billie)

    musician_list = [elton, beyonce, billie]
    most_grammys(musician_list)

main()
```

For example, Elton John has 30 released albums and has won 5 Grammys. His most popular song is “Your Song”. The **most_grammys** function takes in a list of Musician objects as an input parameter and should calculate which artist has won the most Grammy awards.

When the program is run, it should produce the following output

```
Name: Elton John
Number of Studio Albums: 30
Number of Grammy Awards: 5
Most Popular Song: Your Song

Name: Beyonce
Number of Studio Albums: 6
Number of Grammy Awards: 28
Most Popular Song: Single Ladies (Put a Ring on It)

Name: Billie Eilish
Number of Studio Albums: 1
Number of Grammy Awards: 7
Most Popular Song: Bad Guy

The artist with most Grammys is Beyonce with 28 awards
```

Question 4 (20 pts)

Supply the missing code below using Object Oriented Programming. The **StoreInventory** class represents a store's inventory (such as Kohls) and the prices of items. The **SaleEvent** class represents a special sales event where a discount is applied to **all items** in the store. In the code below, the Winter Sales event offers 50% off on all items in the store.

You will need to implement the magic method for the multiplication (*) operator so that the discount gets applied to all the items in a stores inventory. The magic method should return a new numpy array that contains the discounted prices. For example, when the sale event's discount is 50% off (.50):

```
[10.99, 4.50, 30.00, 19.99] → [5.50, 2.25, 15.00, 19.99]
```

When your program runs, it should produce the following output:

```
Winter Sale event at Kohls
Our new prices are:
$5.50
$2.25
$15.00
$9.99
```

```
import numpy as np #do not import anything else


#Your code should go above this comment
def print_new_prices(prices):
    #this functions prints out the prices seen in the output
    for each_item in prices:
        print("${:.2f}".format(each_item))

def main():
    #original price
    kohls_prices = np.array([10.99, 4.50, 30.00, 19.99])
    kohls_inventory = StoreInventory("Kohls", kohls_prices)

    winter_sale = SaleEvent("Winter Sale",.50)

    #all items are now 50% off
    sales_prices = kohls_inventory * winter_sale

    print(winter_sale.getName(),"event at",kohls_inventory.getName())
    print("Our new prices are: ")
    print_new_prices(sales_prices)

main()
```

Note: All the printing is done for you. You just need to define the necessary classes and methods that return the correct value(s)

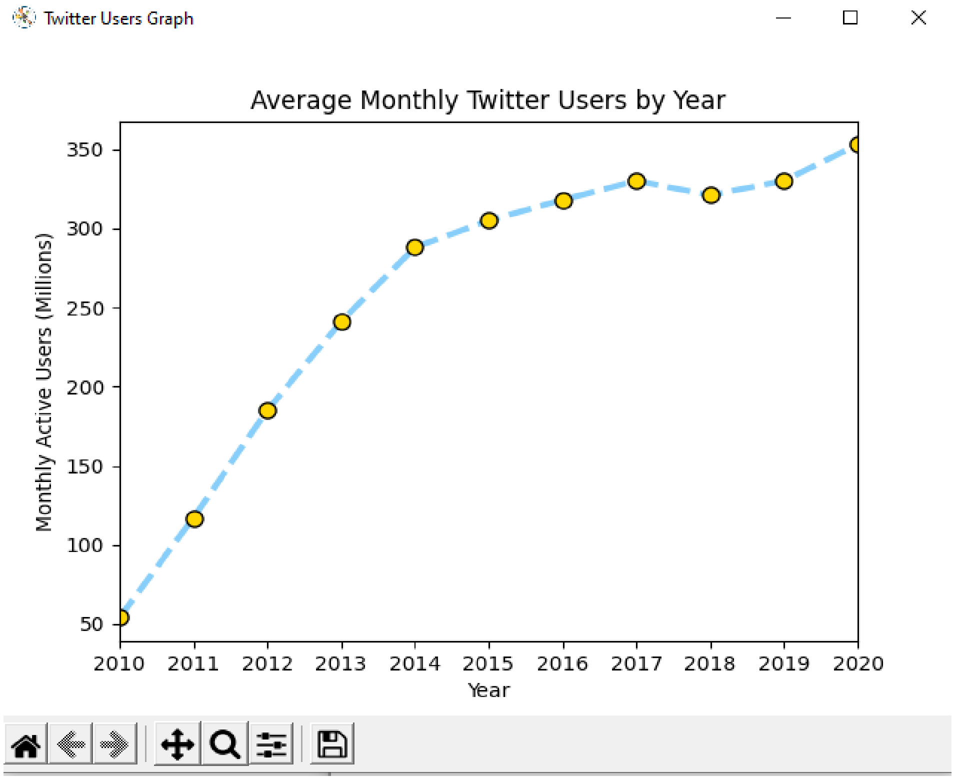
(You may want to reference the magic methods page we looked at in class: https://www.python-course.eu/python3_magic_methods.php -- This is an approved website you can use on this exam 😊)

Question 5 (20 pts)

Year	Average Monthly Active User (Millions)
2010	54
2011	117
2012	185
2013	241
2014	288
2015	305
2016	318
2017	330
2018	321
2019	330
2020	353

Since its creation in 2010, Twitter has become one of the most popular social media platforms. Consider the following data of average monthly active Twitter users since 2010

Using this data, write a Python program that generates a line chart that looks **exactly** like this:



Hint: Read the documentation for matplotlib.pyplot.plot

The chart has a **"lightskyblue" dashed** line. The markers are **yellow circles**. The edge of the markers is **black**

While this problem can be solved using only **Matplotlib**, you may also use **Pandas** and **Numpy**. You may use the following starting code (that contains the data from above in a python list) in your program.

```
year = [2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020]

active_users = [54, 117, 185, 241, 288, 305, 318, 330, 321, 330, 353]
```