CSCI 1411: Fundamentals of Computing
Lab 9
Due Date: October 27, 2023

Name: Brandon Perez

Goals:

- Loops
- Nested loops
- List
- Error handling using while loop.

Development Environment: IDLE

Deliverables:

1) This completed document with required screen shots and algorithms.
2) Python file created for the lab. Name the file using the following format: `lastnameLab09.py`.

How to take a **screen shot**:

- For a Windows 10: Use Snipping Tool to copy and press CTRL + V to paste screen shot.
- For Mac: Press Shift + Command (⌘) + 4 to copy and press Command (⌘) + V to paste screen shot.

Prime and Composite Numbers

Problem Statement: A positive number $n \geq 2$ is prime number if no number between 2 and $\sqrt{n}$ (inclusively) evenly divides n. Write a python program that will ask the user for a positive number x and displays all prime and composite numbers between 2 and x. Hint: You will have to use nested loop.

Your program will complete the following steps:

1. It will ask for and read in an int number verifying that it is ≥ 2. If the input is a negative number, 0 or 1 then it will display an error message and the ask the user to enter another number. Use a while loop for handling error condition (see sample algorithm below)
2. It will generate a list of all prime numbers and composite numbers between 2 and the given number. Hint: Use two lists to keep track of all prime numbers and composite numbers.
3. It will display the list of prime numbers and composite numbers. Note that the list of composite numbers may be empty. If there are no composite numbers, then it must display a message that there are no composite numbers between 2 and the given number.

Algorithm (Task 1 above):

```
1.   Set error to True
2.   Use while loop to iterate as long as error is True
3.      Display 'Enter a number greater than or equal to 2'
4.      Input the number (call this x)
5.      If x is < 2 then
6.         Display 'Number must be greater than or equal to 2'
7.      If x >= 2 then set Error to False
8.   Display 'This program generates and displays list of all
              prime and composite numbers between 2 and x'
9.   Create an empty list for prime numbers (call this prime)
10.  Create an empty list for composite numbers (call this
     composite)
```

You will need nested loops to generate the list of all the prime numbers and composite numbers between 2 and x (Task 2 above). The outer loop will iterate over all the numbers from 2 to x (call this j). Write your algorithm for outer loop in the following box:

Algorithm 1 (Outer Loop)
Iterate through all numbers from 2 through x using variable j Do inner loop to check if j is composite if j is not in composite list, add j to prime

Inner loop will test j to see if it is a prime number or composite number. This can be achieved by repeatedly dividing it be all the numbers between 2 and √j. If j can be completely divided (there is no remainder) by any of the number in the range then it is a composite number and you can add it to the list of composite numbers. If it cannot be completely divided by any numbers between 2 and √j then it is prime numbers and you can add it to the list of prime numbers. Write the algorithm for the inner loop in the following box:

Algorithm 2 (Inner Loop)
iterate through all numbers between 2 and int(sqrt(j)) inclusive using variable i. if j%i==0 add j to composite and break out of inner loop (no need to check every possible divisibility)

After generating the list of all prime numbers and composite numbers display the list using for loops (Task 3 above). You can use the `len` function to check the length of the list. If the length of any of the list is 0 then display the message to that regards (see sample I/O below). Write the algorithm for displaying the lists in the following box:

Algorithm 3 (Displaying the List)

Output the header List of Prime Numbers
Output the primes list and just omit the brackets by converting to substring (saves on runtime)

Using ternary logic dependant on the boolean (x>3) either output
    Header List of Composite Numbers + substring of composites list (saves on runtime)
    No composite numbers between 2 and x

for a more wasteful for loop option, you can do

    iterate through primes printing each with a comma and space

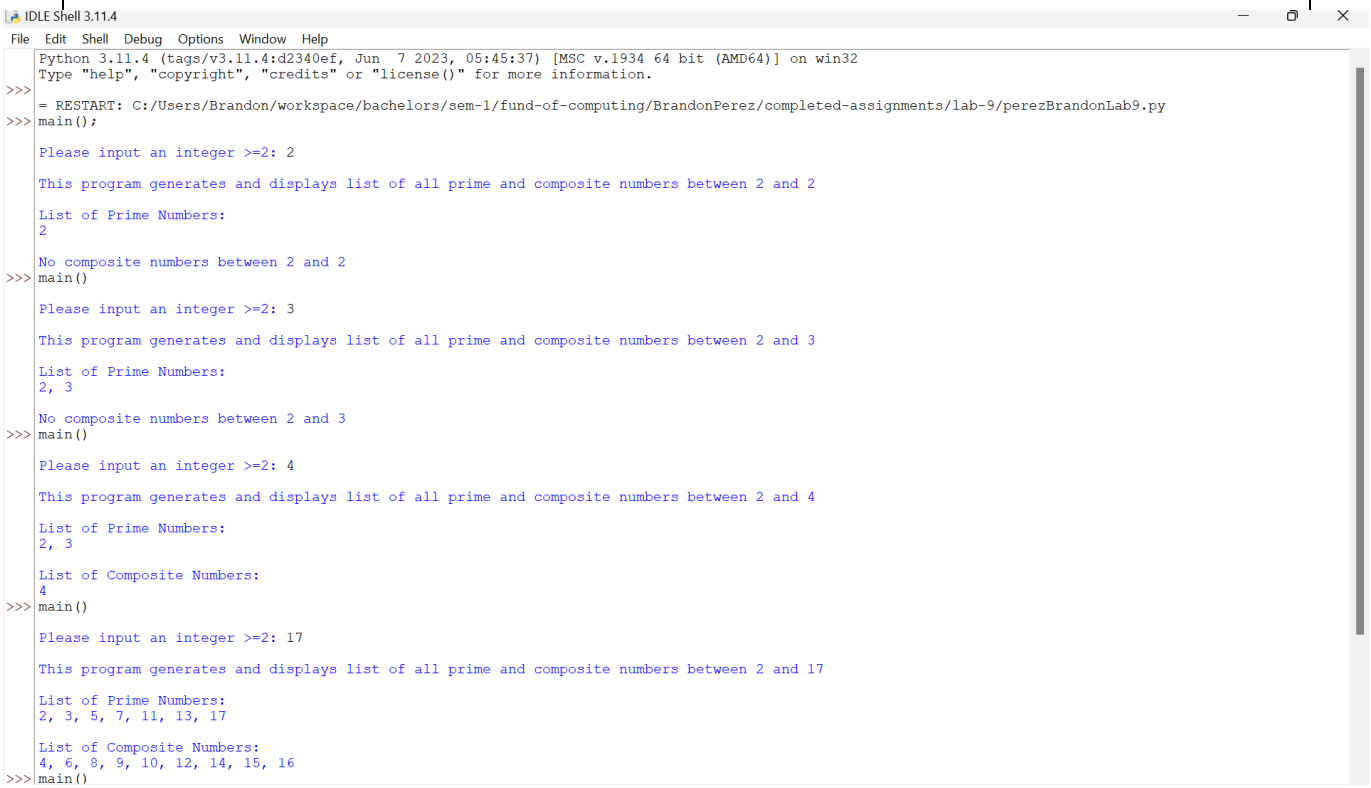
    if length(composites) > 0
        iterate through composites printing each with a comma and space
    else output no composites message

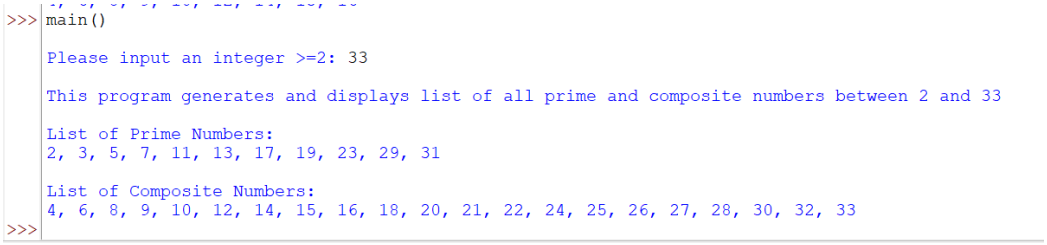
Test your program using the following data.

Run Number	Input	Output	
		Prime Numbers	**Composite Numbers**
1	2	2	No composite numbers
2	3	2 3	No composite numbers
3	4	2 3	4
4	17	2 3 5 7 11 13 17	4 6 8 9 10 12 14 15 16
5	33	2 3 5 7 11 13 17 19 23 29 31	4 6 8 9 10 12 14 15 16 18 20 21 22 24 25 26 27 28 30 32 33

Run your program and take a screen shot of your results and paste it in the box below:

Screen Shot 1





Sample I/O:

```
>>> main ()
Enter a number greater than or equal to 2: 2
This program generates and displays list of all
prime and composite numbers between 2 and 2
List of prime numbers:
2
There are no composite numbers in the range
>>> main ()
Enter a number greater than or equal to 2: 3
This program generates and displays list of all
prime and composite numbers between 2 and 3
List of prime numbers:
2 3
There are no composite numbers in the range
>>> main ()
Enter a number greater than or equal to 2: 4
This program generates and displays list of all
prime and composite numbers between 2 and 4
List of prime numbers:
2 3
List of composite numbers:
4
>>> main ()
Enter a number greater than or equal to 2: 17
This program generates and displays list of all
prime and composite numbers between 2 and 17
List of prime numbers:
2 3 5 7 11 13 17
List of composite numbers:
4 6 8 9 10 12 14 15 16
>>> main ()
Enter a number greater than or equal to 2: 33
This program generates and displays list of all
prime and composite numbers between 2 and 33
List of prime numbers:
2 3 5 7 11 13 17 19 23 29 31
List of composite numbers:
4 6 8 9 10 12 14 15 16 18 20 21 22 24 25 26 27 28 30 32 33
```

Every program should have the following comment block at the top. Make sure to fill in your name, class with section number, due date, brief description of your program, and status of your program:

```
#
# Name:
# Class: CSCI 1411-00X
# Due Date:
# Description:
# Status:
```

Rubric for Lab 9:

Criteria	Rating
Algorithm 1:	Algorithm is included – 5 points Algorithm is not included – 0 points
Algorithm 2:	Algorithm is included – 5 points Algorithm is not included – 0 points
Algorithm 3:	Algorithm is included – 5 points Algorithm is not included – 0 points
Screen Shot 1:	Screen shot is included – 5 points Screen shot is not included – 0 points
Python Program (Input):	Prompts for and read in the number – 5 points Reads in the number without prompt – 2 points Does not read in the number – 0 points
Python Program (Input Validation):	Validate the input to make sure that it is ≥ 2 – 5 points Does not validate the input – 0 points
Python Program (Generate List of Prime Numbers):	Correctly generate list of prime numbers in the given range (2 to x) – 20 points Does not correctly generate the list of prime numbers – 0 points
Python Program (Generate List of Composite Numbers):	Correctly generate list of composite numbers in the given range (2 to x) – 20 points Does not correctly generate the list of composite numbers – 0 points
Python Program (Displays the list of Prime Numbers):	Displays the list with heading (List of Prime Numbers) – 5 points Displays the list of prime numbers without heading – 2 points Does not display the list of prime numbers – 0 points
Python Program (Displays the list of Composite Numbers):	Displays the list with heading (List of Composite Numbers) – 5 points Displays the list of composite numbers without heading – 2 points Does not display the list of composite numbers – 0 points
Python Program (Displays a Message if List Composite Numbers is Empty):	Displays the message – 5 points Does not displays the message – 0 points
Total Points	85