***BE 1600***

***Introduction to***

***Programming and Computation***

***Python***

**Assignment 01**

40 points

**Due 09/18/2023 (11:45 A.M.)**

Assignment Objectives:

* To write programs that perform simple computations.
* To obtain input from a program’s user by using the input function.
* To use identifiers to name variables.
* To assign data to variables.
* To define named constants.
* To use the operators +, -, \*, /, //, %, and \*\* .
* To write and evaluate numeric expressions.
* To perform numeric type conversion with the int and float functions.
* To learn about f-string
* To learn about turtle module.
* To create a basic graphics program using Turtle.

*Solution for this assignment will not be posted on Canvas; however, the solution of any of the assignment problems can be discussed in the class upon request of a student.*

All assignments must be submitted by the Canvas. **No email or hard copy** is accepted. You must follow the following format:

1. For non-programming questions, use a word file to type your answers. Don’t use the text box on the Canvas to answer the questions or to write comments, we will not read it. State your answer clearly.
2. For programming questions, include only the source file of each programming problem.
3. Submit your files to the Canvas. You must submit your files on time; otherwise, you will receive zero.
4. Use “Add Another File” feature on Canvas to upload each additional file; do not upload the files as a compressed folder.
5. You can upload your files as many times as you like. Only the last attempt counts because the last files you uploaded are the only files your instructor will see.
6. There will be several modules on the Canvas. You need to upload your files using the correct module on the Canvas.
7. Name each file: *Assignment (assignment number)* for the word file [e.g. Assignment 02] and *Assignment (assignment number) \_ (Question number)* for each programming question [e.g. Assignment 02\_Q03].
8. To upload your file(s):

* In Course Navigation, click the ASSIGNMENTS module.
* Click the title of the assignment.
* Click the **Submit** Assignment button.
* Add **File**. ...
* Add Another **File**. ...
* **Submit** Assignment. ...
* View **Submission**.

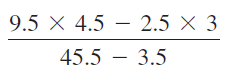
*It is your responsibility to make sure that each file is uploaded correctly. If you uploaded a wrong file, you receive zero; files will not be accepted after due date even if you have a prove that the file is created before the due date.*

***Make sure you review the Cheating & Plagiarism policy on Canvas.***

Use Python Script Window and write a program for each of Q.1 to Q.10. Convert your .py files to text files. Submit 10 text files. Use only Python syntax discussed in class for solving all programming problems.

**Question 01 (3 points) Done**

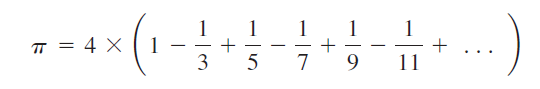
Write a program that displays the result of



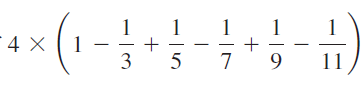
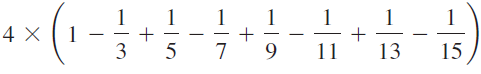
**0.8392857142857143**

**Question 02 (4 points) Done**

π can be computed using the following formula:



Write a program that displays the result of

 and 

**2.9760461760461765**

**3.017071817071818**

**Question 03 (4 points) WROK on it**

The US Census Bureau projects population based on the following assumptions:

* One birth every 7 seconds
* One death every 13 seconds
* One new immigrant every 45 seconds

Write a program to display the population for each of the next five years. Calculate the population of each independently. Assume the current population is 312032486 and one year has 365 days.

Hint: in Python, you can use integer division operator **//** to perform division. The result is an integer. For example, **5 // 4** is **1** (not **1.25**) and **10 // 4** is **2** (not **2.5**).

**population after one year: 314812582**

**population after two years: 317592679**

**population after three years: 320372776**

**population after four years: 323152873**

**population after five years: 325932970**

**Question 04 (3 points)**

Write a program that reads a number in feet, converts it to meters, and displays the result. One foot is **0.305** meters. Here is a sample run:

**Enter a value for feet: 16.5**

**16.5 feet is 5.0325 meters**

**Question 05 (3 points) Done**

How cold is it outside? The temperature alone is not enough to provide the answer. Other factors including wind speed, relative humidity, and sunshine play important roles in determining coldness outside. In 2001, the National Weather Service (NWS) implemented the new wind-chill temperature to measure the coldness using temperature and wind speed. The formula is given as follows:



where ta is the outside temperature measured in degrees Fahrenheit and *v* is the speed measured in miles per hour. twc is the wind-chill temperature. The formula cannot be used for wind speeds below 2 mph or for temperatures below or above 41°F.

Write a program that prompts the user to enter a temperature between -58° F and 41°F and a wind speed greater than or equal to **2** and displays the wind-chill temperature rounded to two decimal places.

Here is a sample run:

**Enter the temperature in Fahrenheit between -58 and 41: 5.3**

**Enter the wind speed miles per hour (must be greater than or equal to 2): 6**

**The wind chill index is -5.57**

**Question 06 (4 points)** **Done**

Write a program that displays the following table:

**a b pow(a, b)**

**------------------------------**

**1 2 1.0**

**2 3 8.0**

**3 4 81.0**

**4 5 1024.0**

**5 6 15625.0**

Use f-string to format the table. Use function pow in math library to calculate the values in the third column. Round the values in the third column to two decimal places.

**Question 07 (4 points) Done**

Write a program that prompts the user to enter a four-digit integer and displays the number in reverse order. Use integer division and reminder operators to separate the digits.

Here is a sample run:

**Enter a four-digit integer: 3125**

**5**

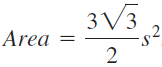
**2**

**1**

**3**

**Question 08 (3 points)**

Write a program that prompts the user to enter the side of a hexagon and displays its area. The formula for computing the area of a hexagon is



where *s* is the length of a side. Round the area to three decimal places.

Here is a sample run:

**Enter the side: 5.5**

**The area of the hexagon is 78.590 # 78.592**

**Question 09 (6 points)**

Write a program that prompts the user for two points and draws a red line between them. The displays the coordinates of the two points, as shown in the below figure. Use the turtle module to draw the line and display the coordinates.

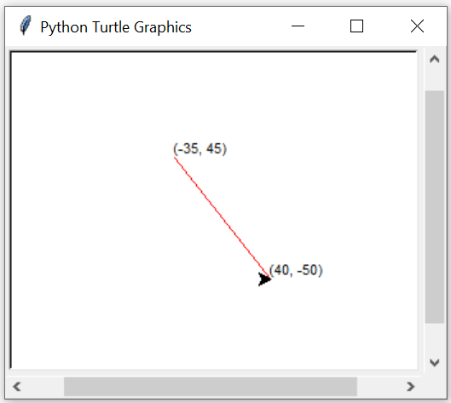
Here is a sample run:

Enter first point x value -35

Enter first point y value 45

Enter second point x value 40

Enter second point y value -50



**d**

**Question 10 (6 points)**

Write a program that prompts the user to enter the center of a rectangle, width, and height, and displays the rectangle, as shown in the below figure.

**Enter x coordinate for the center of the rectangle: -50**

**Enter y coordinate for the center of the rectangle: 25**

**Enter the width of the rectangle: 300**

**Enter the height of the rectangle: 100**

