**\*\*\* MidTerm \*\*\***

Each question is worth 20 points. The extra credit is optional.

**1. Programming Exercise 3-13 (Shipping Charges), page 164 (10pts)**

* The weight must be a decimal value. (2.5pts)
* The shipping charge must be a decimal value - 2 decimal places should be displayed. (2.5pts)
* The weight cannot be zero or a negative value. (5pts)

**Output:**









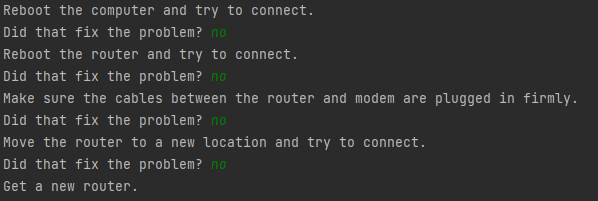


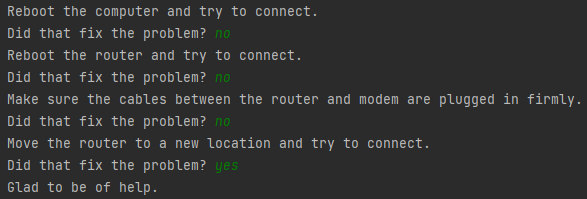


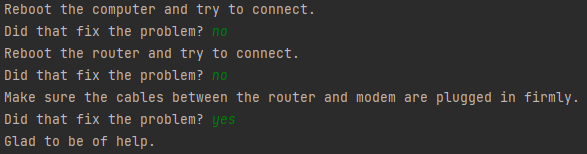
**2. Programming Exercise 3-17 (Wi-Fi Diagnostic Tree), page 165 (20pts)**

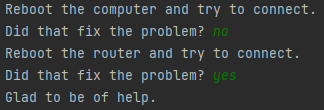
* Pay attention to the flow chart - Figure 3-19, page 166

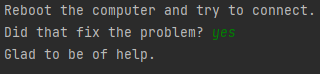
**Output:**







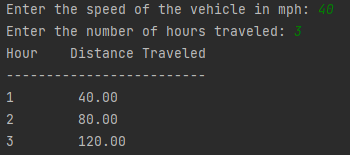




**3. Programming Exercise 4-4 (Distance Traveled), page 214 (10pts)**

* The speed must be a decimal value. (2.5pts)
* The time must be an integer value. (2.5pts)
* Distance traveled should display 2 decimal places. (5pts)

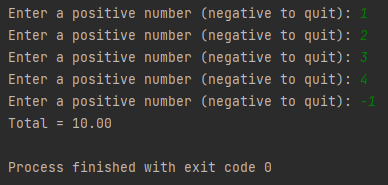
**Output:**

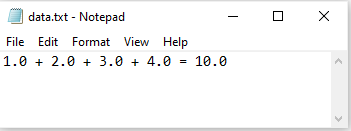
****

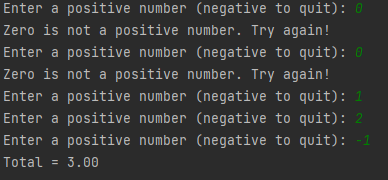
**4. Programming Exercise 4-8 (Sum of Numbers), page 214 (10pts)**

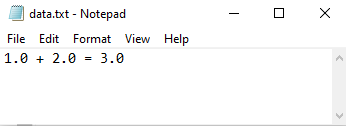
* The number must be a decimal value. (1pts)
* The sum should display 2 decimal places. (2pts)
* If zero is entered, then ask the user to enter a positive number again. (2pts)
* Sent data to a file named data4.txt. (5pts)

Output:





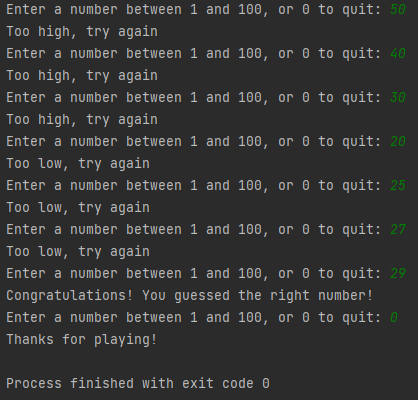


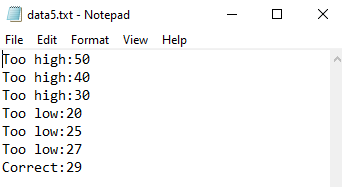
****

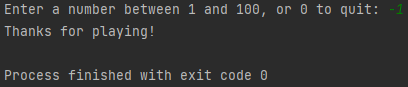
**5. Programming Exercise 5-20 (Random Number Guessing Game), page 298 (10pts)**

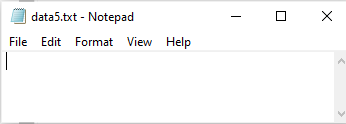
* Add validation. If a negative number, a number greater than 100 or 0 is entered, terminate the program with a “Thanks for playing!” message. (3 pts)
* Send all your input to a file named data5.txt (7 pts)

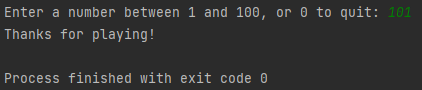
Output:

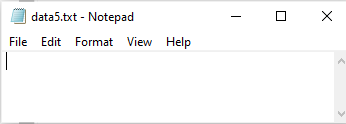
****

****

****

****

****

****

**Extra Credit: Programming Exercise 4-13 (Population), page 215**

**ALL requirements MUST be met to get 20 points.**

* The number organisms must be an integer value.
* The average must be a decimal value.
* The number of days must be an integer value.
* If any of the above values (number organisms, average, number of days) are negative, then your program needs to keep asking for the value until a valid (positive number) value is entered.
* On day 1, the population will match the number of organisms. On the next days, the following formula should be used:

New Number of organisms = Number of organisms + (Number of organisms \* Average)

For example:

organisms = 2.0, Average = 30%, days = 10

Day Population

1 2.00

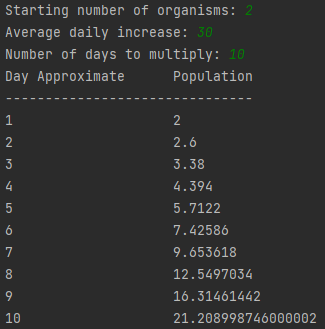
2 organisms = organisms + (organisms \* average) => 2.0 + (2.0\*0.3) => 2.6

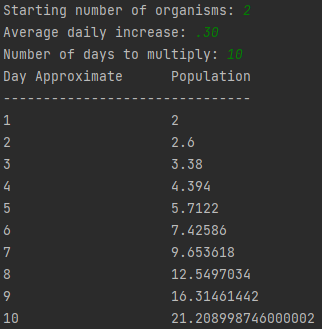
3 organisms = organisms + (organisms \* average) => 2.6 + (2.6\*0.3) => 3.38

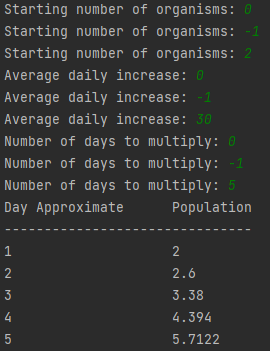
4 organisms = organisms + (organisms \* average) => 3.38 + (3.38\*0.3) => 4.394

5 organisms = organisms + (organisms \* average) => 4.394 + (4.394\*0.3) => 5.7122

**Output:**







**Note:** Zip your work and name the file **MidTerm\_yourfirstname\_yourlastname.zip**. Upload it to Canvas.

Your zip file should have the following:

* Lab1.py
* Lab2.py
* Lab3.py
* Lab4.py
* Lab5.py
* extra-credit.py