CSCI 1411: Fundamentals of Computing
Lab 7
Due Date: October 13, 2023

Name: Brandon Perez

Goals:

- Use of decision structure
- Testing

Development Environment: IDLE

Deliverables:

1) This completed document with required screen shots and algorithms.
2) Python file created for the first and second parts of the lab. Name the file using the following format: `lastnameLab07Part1.py` and `lastnameLab07Part2.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.

Part I – Calculating Grades

Problem Statement: Write a Python program to convert numerical score into letter grade. It will ask for three test scores, calculate average of the test scores and convert it into a letter grade using the following mapping:

Average of Test Score	Letter Grade
90 ≤ Average Test Score ≤ 100	A
80 ≤ Average Test Score < 90	B
70 ≤ Average Test Score < 80	C
60 ≤ Average Test Score < 70	D
Average Test Score < 60	F

Notes:

1. Assume that test scores are int.
2. If the average is above 100 then your program must display an error message and set the letter_grade to undefined.
3. If any of the test score is non-numerical or negative, then your program must display an error message and stop its execution (see below for a sample code).

Algorithm (Pseudocode):

```
1. Display 'Enter test score 1'
2. Input first test score (call it test1).
3. Display 'Enter test score 2'
4. Input second test score (call it test2).
5. Display 'Enter test score 3'
6. Input third test score (call it test3).
7. Calculate average using the following formula:

   $Average = \frac{test1 + test2 + test3}{3}$

8. Display average score (format it to three decimal places).
9. Convert average into letter grade (call this letter_grade)
10. Display letter_grade
```

Convert this algorithm into Python code and test it using the following data: If you get the correct result (shown in the last two column) then your program is working as expected.

Run Number	**Input**			**Output**	
	Test Score 1	**Test Score 2**	**Test Score 3**	**Average**	**Letter Grade**
1	100	90	80	90.000	A
2	66	33	88	62.333	D
3	ABC			Error Message	
4	100	-90		Error Message	
5	200	95	85	126.666	Undefined

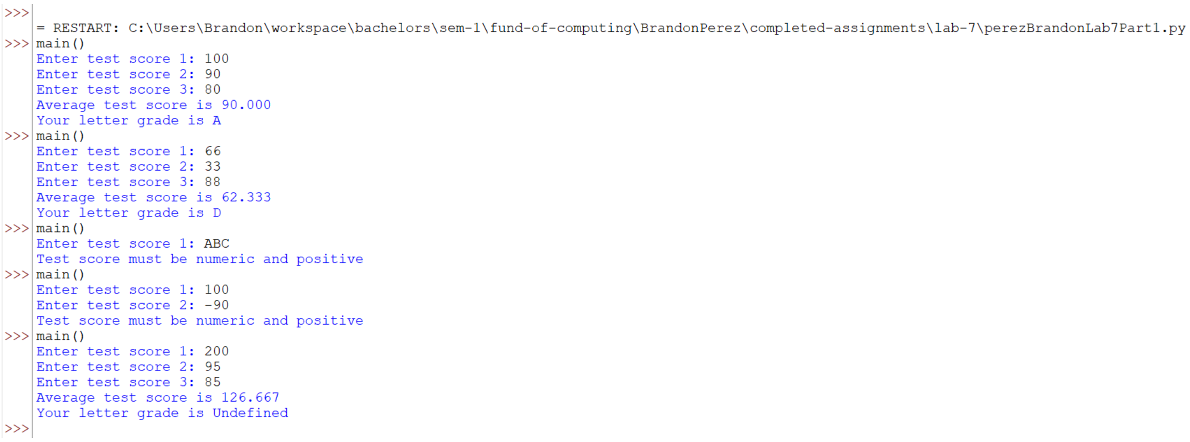
Sample Code (to test for non-numerical and negative input):

```
# Ask for and read in test score 1 without converting it into int
    test_score_1 = input ('Enter test score 1: ')

    #Test to make sure that input is numeric and positive
    if (test_score_1.isnumeric ()):
        # if it is numeric then convert it into int
        test1 = int (test_score_1)
    else:
        #Display error message
        print ('Test score must be numeric and positive')
        #End program by executing return statement
        return
```

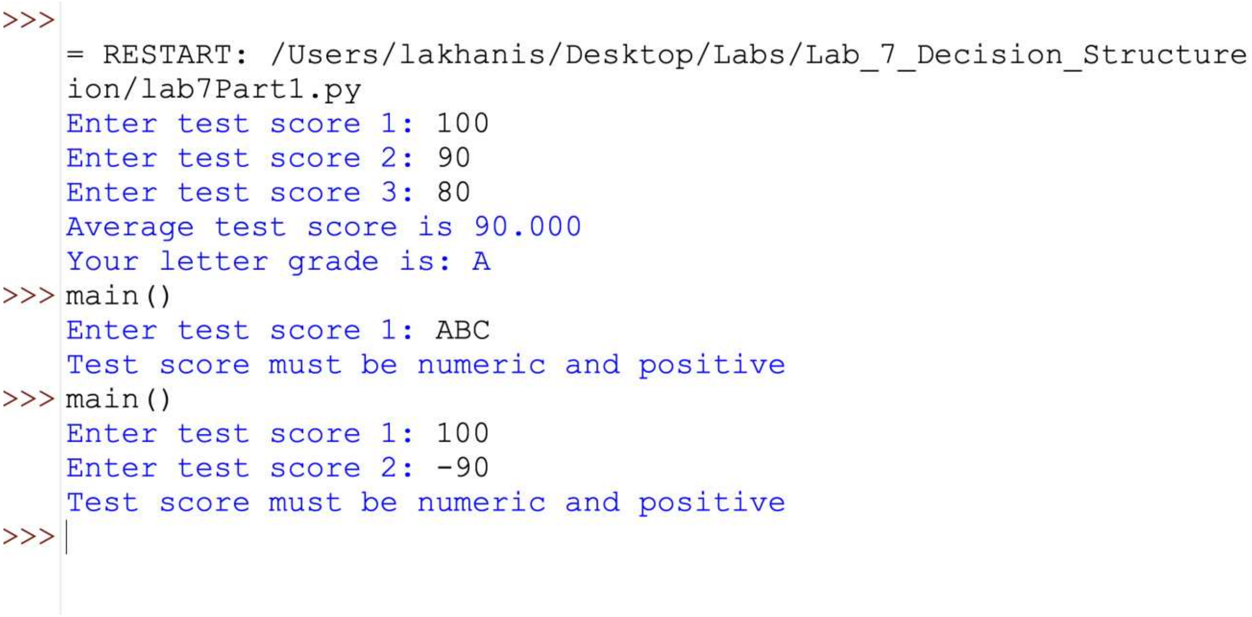
Run your program and take a screen shot of your results and paste it in the box below:

Screen Shot 1



```
>>>
    = RESTART: C:\Users\Brandon\workspace\bachelors\sem-1\fund-of-computing\BrandonPerez\completed-assignments\lab-7\perezBrandonLab7Part1.py
>>> main()
    Enter test score 1: 100
    Enter test score 2: 90
    Enter test score 3: 80
    Average test score is 90.000
    Your letter grade is A
>>> main()
    Enter test score 1: 66
    Enter test score 2: 33
    Enter test score 3: 88
    Average test score is 62.333
    Your letter grade is D
>>> main()
    Enter test score 1: ABC
    Test score must be numeric and positive
>>> main()
    Enter test score 1: 100
    Enter test score 2: -90
    Test score must be numeric and positive
>>> main()
    Enter test score 1: 200
    Enter test score 2: 95
    Enter test score 3: 85
    Average test score is 126.667
    Your letter grade is Undefined
>>>
```

Sample I/O:



```
>>>
    = RESTART: /Users/lakhanis/Desktop/Labs/Lab_7_Decision_Structure
    ion/lab7Part1.py
    Enter test score 1: 100
    Enter test score 2: 90
    Enter test score 3: 80
    Average test score is 90.000
    Your letter grade is: A
>>> main()
    Enter test score 1: ABC
    Test score must be numeric and positive
>>> main()
    Enter test score 1: 100
    Enter test score 2: -90
    Test score must be numeric and positive
>>>
```

Part II – Cost Calculator

A software company sells a package that retails for $99. Quantity discounts are given according to the following table:

Quantity	Discount
10 to 19	20%
20 to 49	30%
50 to 99	40%
100 or more	50%

Write a program that asks the user to enter the number of packages purchased. The program should then display the discount percentage, amount of the discount (can be 0) and the total amount of the purchase after the discount. Format the output to two decimal places, include % sign after the discount percentage and $ before the discount amount & total amount. Use the f string to display all numerical data using two decimal places. If qty is negative or non-numerical then your program must display appropriate error message. Perform the calculations as follows:

- amount = quantity * 99
- discount amount = amount * discount percentage /100
- total amount = amount – discount amount

Algorithm (Pseudocode) (write your answer in the box below):

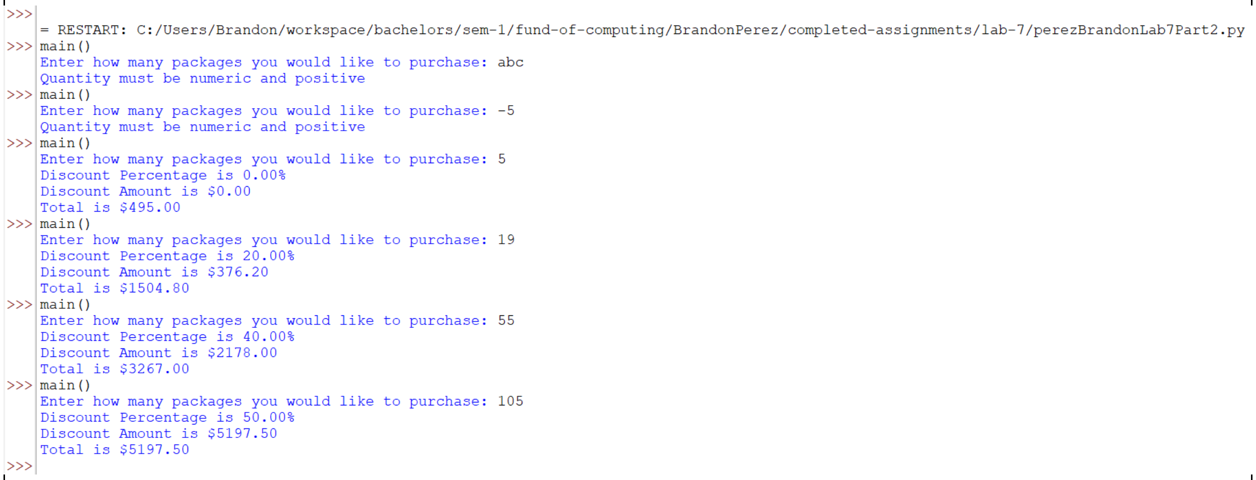
```
Display "Enter how many packages you would like to purchase: "
Input value as int quantity
If quantity > 0:
   discount_percent = 0;
   If quantity >= 100 :
        discount_percent = 50;
   Else If quantity >= 50 :
        discount_percent = 40;
   If quantity >= 20 :
        discount_percent = 30;
   Else If quantity >= 10 :
        discount_percent = 20;
   Calculate amount, discount_amount, and total_amount using the formulas given
   Display discount_percent%, amount, and total amount (format to two decimals)
Else
   Print an error message explaining that quantity can not be negative.
```

Use the quantity given in the following table to test your program for different possible outcomes. If you get the correct results (shown in the last three columns) then your program is working as expected.

Run Number	**Quantity**	**Output**		
		Discount Percentage	**Discount Amount**	**Total Amount**
1	abc	Display Error Message		
2	-5	Display Error Message		
3	5	0.00%	$0.00	$495.00
4	19	20.00%	$376.20	$1504.80
5	55	40.00%	$2178.00	$3267.00
6	105	50%	$5197.50	$5197.50

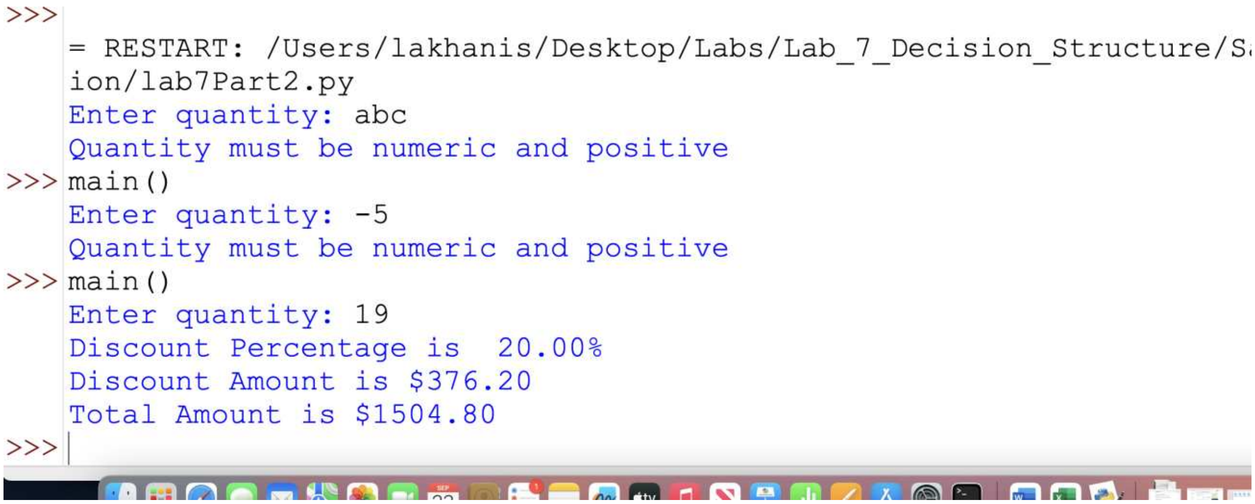
Run your program and take a screen shot of your result and paste it in the box below:

Screen Shot 2



```
>>>
    = RESTART: C:/Users/Brandon/workspace/bachelors/sem-1/fund-of-computing/BrandonPerez/completed-assignments/lab-7/perezBrandonLab7Part2.py
>>> main()
    Enter how many packages you would like to purchase: abc
    Quantity must be numeric and positive
>>> main()
    Enter how many packages you would like to purchase: -5
    Quantity must be numeric and positive
>>> main()
    Enter how many packages you would like to purchase: 5
    Discount Percentage is 0.00%
    Discount Amount is $0.00
    Total is $495.00
>>> main()
    Enter how many packages you would like to purchase: 19
    Discount Percentage is 20.00%
    Discount Amount is $376.20
    Total is $1504.80
>>> main()
    Enter how many packages you would like to purchase: 55
    Discount Percentage is 40.00%
    Discount Amount is $2178.00
    Total is $3267.00
>>> main()
    Enter how many packages you would like to purchase: 105
    Discount Percentage is 50.00%
    Discount Amount is $5197.50
    Total is $5197.50
>>>
```

Sample I/O



```
>>> 
    = RESTART: /Users/lakhanis/Desktop/Labs/Lab_7_Decision_Structure/S
    ion/lab7Part2.py
    Enter quantity: abc
    Quantity must be numeric and positive
>>> main()
    Enter quantity: -5
    Quantity must be numeric and positive
>>> main()
    Enter quantity: 19
    Discount Percentage is  20.00%
    Discount Amount is $376.20
    Total Amount is $1504.80
>>> 
```

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 7:

Criteria	Rating
Part I (Screen shot 1)	Screen shot included – 5 points No screen shot included – 0 points
Part I: Python Program	Prompts for and read in the test scores – 5 points Reads in the test scores without prompt – 2 points Does not read in the test scores – 0 points
Part I: Python Program	Validate the test scores for being numeric and positive – 5 points Does not validate the test score for being numeric and positive – 0 points
Part I: Python program	Correctly calculates the average score – 5 points Does not correctly calculate the average score – 0 points
Part I: Python Program	Displays the average score formatted to three decimal places – 5 points Displays the average score without formatting – 2 points Does not display the average score – 0 points
Part I: Python Program	Correctly maps the average score to letter grade – 5 points Mapping of average score to letter grade is incorrect – 0 points
Part I: Python Program	Displays the letter grade with text message – 5 points Displays the letter grade without appropriate text message – 2 points Does not display the letter grade – 0 points
Part II (Screen shot 2)	Screen shot included – 5 points No screen shot included – 0 points
Part II: Algorithm:	Algorithm is included – 5 points Algorithm is not included – 0 points
Part II: Python Program	Prompts for and read in the quantity – 5 points Reads in the quantity without prompt – 2 points Does not read in the quantity – 0 points
Part II: Python Program	Correctly calculates the discount percentage – 5 points Does not correctly calculate the discount percentage – 0 points
Part II: Python Program	Correctly calculates the discount amount – 5 points Does not correctly calculate the discount amount – 0 points
Part II: Python Program	Correctly calculates the total amount – 5 points Does not correctly calculate the total amount – 0 points
Part II: Python Program	Displays the discount percentage, discount amount, and total amount with correct format and text message – 5 points Displays the discount percentage, discount amount, and total amount without correct format and/or text message – 2 points Does not display all the required output – 0 points
Total Points	70