ITSE 1359 – Lab 8 Assignment (Refer to Ch. 8 as needed):

NOTE: Lab 8 requires you to work on and submit two programs. Read this document carefully and comply with submission and grading criteria. Your first problem (Rectangle Calculations) will result in two files. Use the video provided to construct this. The second program (Determine Season) will consist of one file. Use the pseudocode provided to solve this problem.

Note: Sublime Text does not handle user input, so you will have to use an editor that can like Idle or VS code – Unless you want to run from the command prompt (see my YouTube Video on chapter 7)!

Problem #1 – Rectangle Calculations:

Submit 2 files: rectangle_main.py and rectangle_mod.py

Problem Definition: Create a python program that prompts for the width and length of a rectangle and prints the area and perimeter. Your prompts for width and length must be in a loop which ask the user if they want to repeat. This program must use functions in a module. A program title and goodbye are required. Do not forget general and specific comments.

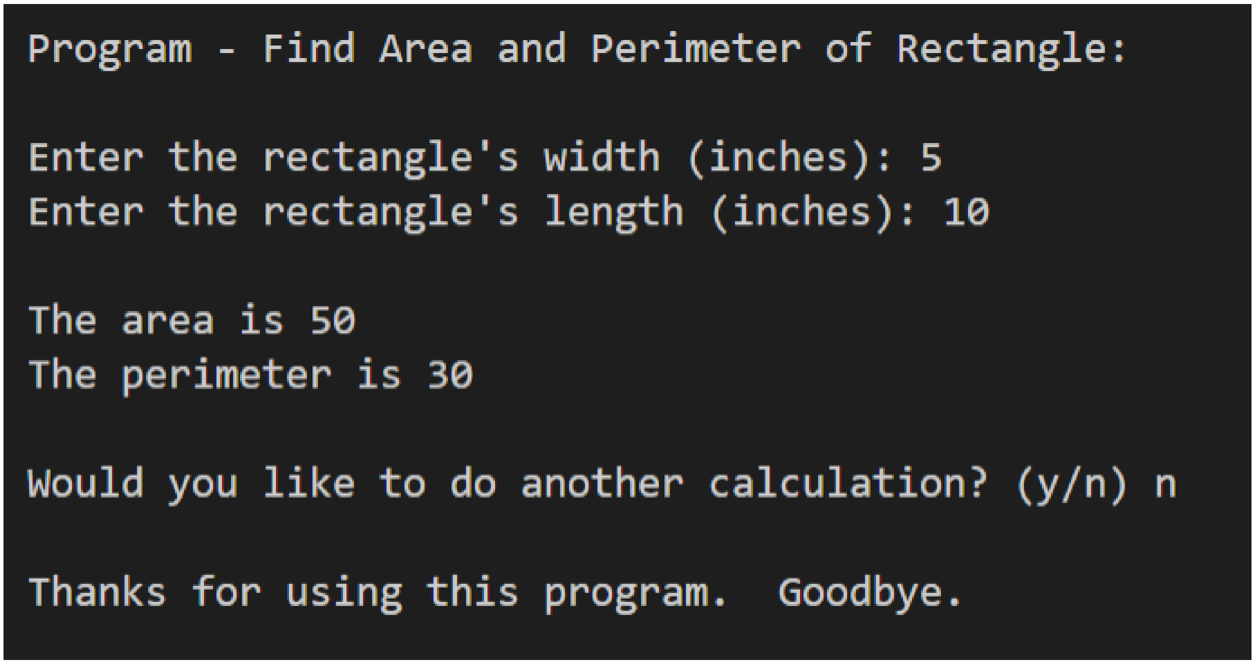
Sentence structure for prompts and output should match screenshot provided. Numbers should be formatted to zero decimal places. This application must utilize functions defined in a module. These functions must be called from your main application utilizing dot notation to reference and utilize the function.

Description of Module: First, create a module called: rectangle_mod.py. In this module define the functions that return area and perimeter. Name them find_area() and find_perimeter() and define them according to these specifications:

- To find the area of a rectangle, use the formula: width * length. Your function to find the area should receive width and length and return area.
- To find the perimeter of a rectangle, use the formula: 2 * (width + length). Your function to find the perimeter should receive width and length and return the perimeter.

Description of main program: Second, create a file called rectangle_main.py. In this file, you will have the main logic of your program and call the methods you created in the module. See pseudocode for program logic. Both rectangle_mod.py and retangle_main.py need to be in same folder and when you run, you must be in the main app.

Screenshot of Output (Rectangle Calculations):



```
Program - Find Area and Perimeter of Rectangle:

Enter the rectangle's width (inches): 5
Enter the rectangle's length (inches): 10

The area is 50
The perimeter is 30

Would you like to do another calculation? (y/n) n

Thanks for using this program.  Goodbye.
```

Pseudocode / Comments for module (rectangle_mod.py):

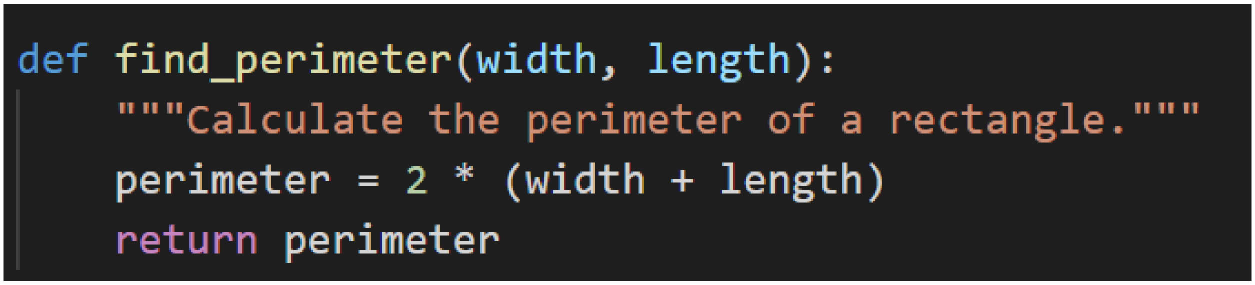
```
# function definition for find_area(receives width and length):

    # docstring stating what function does
    # calculate area in body
    # return area

# function definition for find_perimeter(receives width and length)

    # docstring stating what function does
    # calculate perimeter in body
    # return perimeter
```

Pseudocode / Comments for main app (rectangle_main.py):

```
# import rectangle_mod module
# print name of program
# set repeat flag to True

# while repeat, prompt, calculate, show output, test for again

    # prompt for width and convert to float
    # prompt for length and convert to float
    # calculate area by calling function in module
    # calculate perimeter by calling function in module
    # output shows area and perimeter formatting to zero decimal places
    # prompt for again and set flag accordingly

# say thanks and goodbye
```

Code Examples (rectangle.mod):

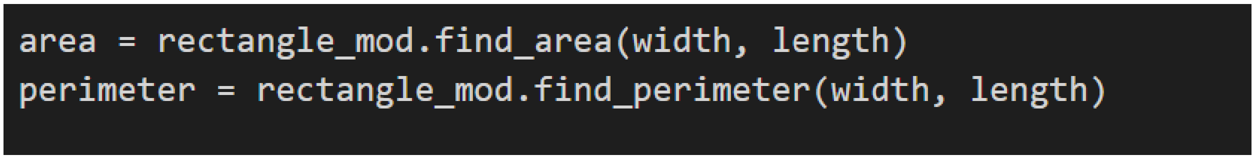
How do I create the function that calculates the perimeter of a rectangle?



```
def find_perimeter(width, length):
    """Calculate the perimeter of a rectangle."""
    perimeter = 2 * (width + length)
    return perimeter
```

Code Examples (rectangle_main):

How to call method in module?



```
area = rectangle_mod.find_area(width, length)
perimeter = rectangle_mod.find_perimeter(width, length)
```

Problem #2 – Determine Season (determine_season.py):

Problem Definition: Create a python program that receives a Fahrenheit temperature and based on this input determines the probable season. This program must receive input from the user at the keyboard and utilize a function to determine the probable season. The function definition must appear in the same file that calls the function. Use the pseudocode provided to build your solution. Do not forget general and specific comments.

Input should be converted to a float using the float() function instead of the int() function. Your prompt for this temperature must be in a loop which askes the user if they want to repeat. Match up with output screen on next page. Make sure the season is shown in title case.

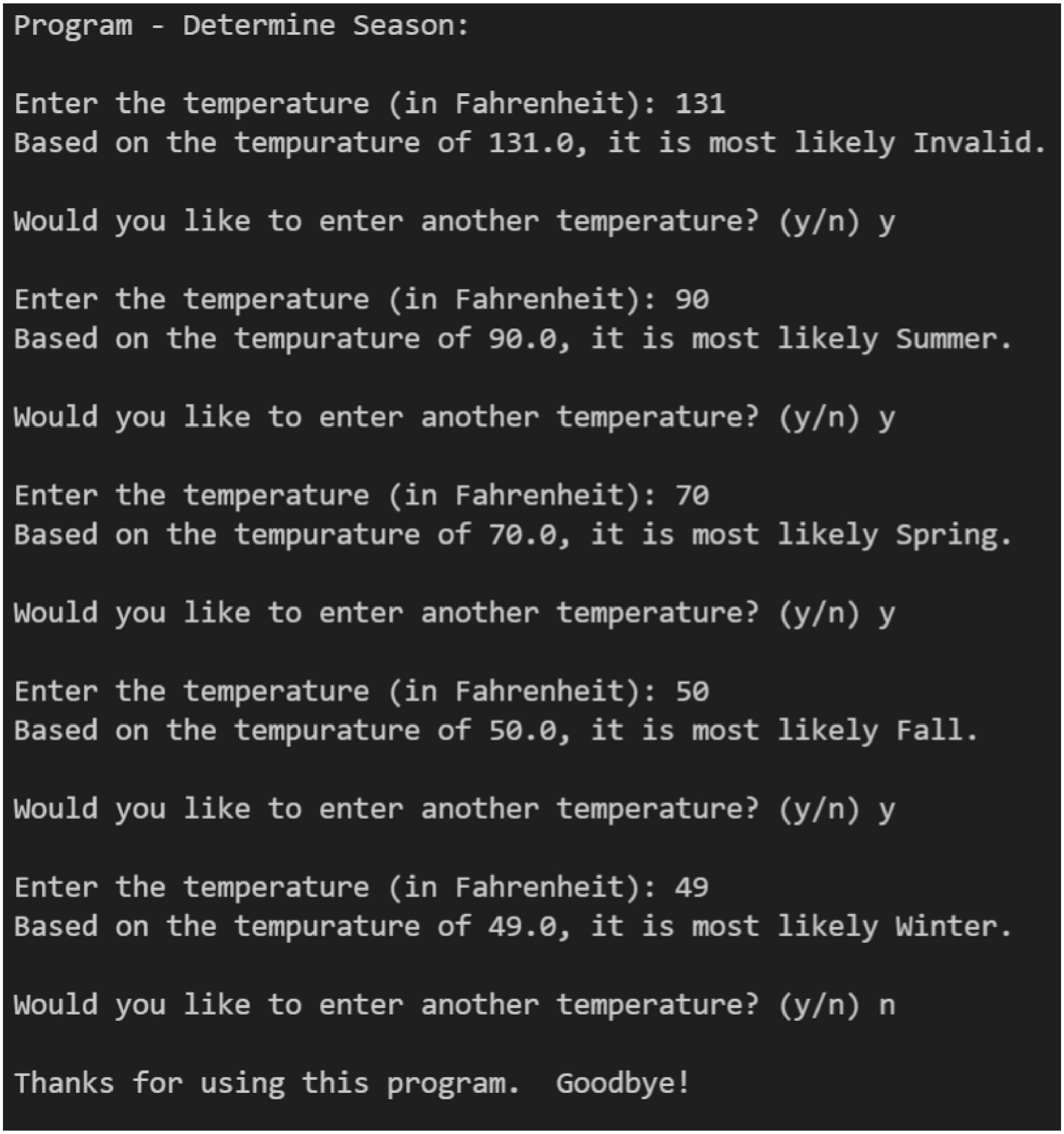
Your method to determine season should receive a temp and return the season. To find the probable season use this logic in your method:

```
If temp > 130 or temp <- 20 then
    season is “invalid”
else if temp >= 90
    season is “summer”
else if temp >= 70 and temp < 90
    season is “spring”
else if temp >= 50 and temp < 70
    season is “fall”
else if temp < 50
    season is “winter”
```

Output should start out with a program title, prompt the user for temp, show the probable season, and ask for repeat. Program exits with a goodbye statement. Sentence structure for prompts and output should match screenshot below.

Name your file: determine_season.py

Screenshot (Determine Season):



```
Program - Determine Season:

Enter the temperature (in Fahrenheit): 131
Based on the tempurature of 131.0, it is most likely Invalid.

Would you like to enter another temperature? (y/n) y

Enter the temperature (in Fahrenheit): 90
Based on the tempurature of 90.0, it is most likely Summer.

Would you like to enter another temperature? (y/n) y

Enter the temperature (in Fahrenheit): 70
Based on the tempurature of 70.0, it is most likely Spring.

Would you like to enter another temperature? (y/n) y

Enter the temperature (in Fahrenheit): 50
Based on the tempurature of 50.0, it is most likely Fall.

Would you like to enter another temperature? (y/n) y

Enter the temperature (in Fahrenheit): 49
Based on the tempurature of 49.0, it is most likely Winter.

Would you like to enter another temperature? (y/n) n

Thanks for using this program.  Goodbye!
```

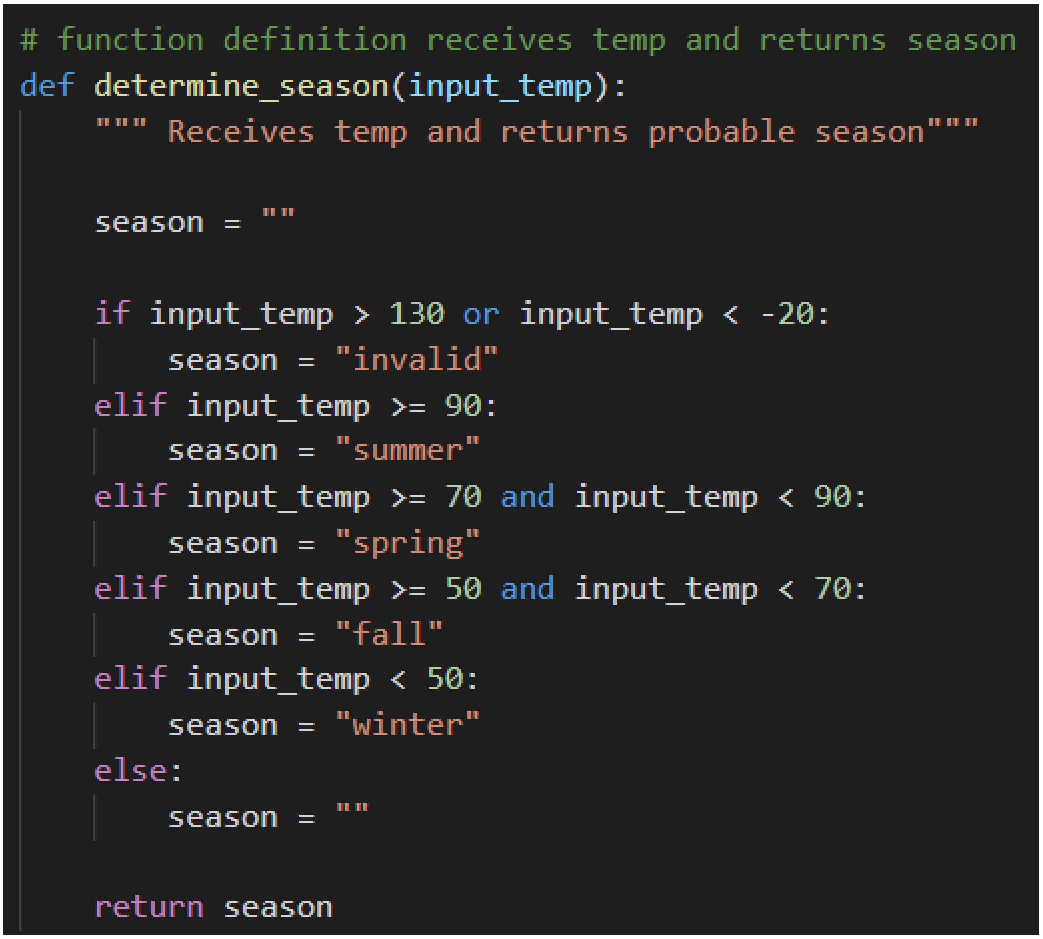
Data type and Input:

If you expect integers to be input and the user enters floating point, your program will crash (ValueError). So, that is why this program is best to receive input as a float.

Pseudocode / Comments (Determine Season):

```
# function definition determine_season – receives temp and returns season

# print name of program
# set strings for prompts
# set repeat flag to True

# while repeat, prompt, find season, show output, test for again:

        # prompt for temp and convert to number
        # find season by calling function, pass temp, assign return to variable
        # print output stating temp and season using title case
        # prompt for again and set flag accordingly

# print thanks and goodbye on exit
```

Example Code (Determine Season - function that returns season):



```python
# function definition receives temp and returns season
def determine_season(input_temp):
    """ Receives temp and returns probable season"""

    season = ""

    if input_temp > 130 or input_temp < -20:
        season = "invalid"
    elif input_temp >= 90:
        season = "summer"
    elif input_temp >= 70 and input_temp < 90:
        season = "spring"
    elif input_temp >= 50 and input_temp < 70:
        season = "fall"
    elif input_temp < 50:
        season = "winter"
    else:
        season = ""

    return season
```

Submit your lab assignment:

VIP: Since this lab assignment involves multiple files, create a folder called “lab8_username”, where username is your college username. Put your three files this folder (rectangle_main.py, rectangle_mod.py, determine_season.py). Zip your folder. See last page of this document for directions on how to create a folder and zip a folder.

Using the Canvas assignment tool, upload your zipped file to the lab 8 assignment. After you have uploaded your zipped file, submit your lab assignment.

All lab assignments must be submitted using the CANVAS assignments tool. Lab Assignments will not be accepted any other way**.** Make sure you submit your work to the right lab assignment number otherwise you will not get credit.

Grading Criteria:

- ✓ You must follow the steps in the Rectangle Calculations video.
- ✓ Don’t forget general comments.
- ✓ Specific comments are optional.
- ✓ Use white space to make your code easy to read.
- ✓ Comply with PEP-8 conventions for variable names, file names, etc.
- ✓ Satisfy the problem definition and other grading standards.
- ✓ Your work should not have syntax errors.
- ✓ Your work must be your own.
- ✓ Match your output screen to screenshot provided.
- ✓ Any deviations from lab specifications will result in points off.
- ✓ If you want to vary, do so on your own.

VIP Videos:

Rectangle Calculations program (step-by-step): Lab 8 YouTube Video

Lectures and other videos: VIEW MY PYTHON PLAYLIST

Closing:

If you have questions about this lab send me a message using canvas inbox or attend zoom office hours. See Unit 0 for link for my office hours. Programming tutors are also available and listed in your Canvas class as announcements.

By-the-way, a great way to get ready for your lab assignments (and exams) is do the Try It Yourself problems in your book. Most of the solutions are on the authors website and I also discuss them in my YouTube video lectures.

Another way to get ready for your lab assignments is to review my lecture notes for each chapter. In Unit 0, find the link for my chapter notes.

Warning: Labs will increase in complexity with each lab assignment. A beginner may need 5 to 15 hours to complete each lab. Please start early because there will be no extensions.

Consider doing this lab over a three-day period:

- ✓ Day 1 – Get confused and run out of time.
- ✓ Day 2 – Research and debug errors.
- ✓ Day 3 – Polish, double-check everything, and submit.

Have an exception free day!

Prof. Benavides

Resources – Creating a Folder and zipping it:

Win 10 - How do I create a folder?

1.) Right-mouse click on the Desktop.
2.) Point to “New” and then click on “Folder”.
3.) Type in the name of your folder and press Enter.

Win 10 - How do I zip a folder? (Do it this way please.)

1.) Right-mouse click on the folder.
2.) Point to “Send to” and then click “Compressed (zipped”) folder”.
3.) Now you see zipped file by same name as folder with a zipper on it.
4.) Do not send a .rar file or any other file type other than .zip.

macOS - How do I create a folder?

1.) Right-mouse click on the Desktop.
2.) Select “New Folder”.
3.) Type in the name of your folder and press Enter.

macOS - How do I zip a folder? (Do it this way please.)

1.) Right-mouse click on the folder.
2.) Select “Compress”. Name of folder appears after word Compress.
3.) Now you see zipped file by same name as folder with a zipper on it.
4.) Do not send a .rar file or any other file type other than .zip.