The following programs will focus your skills on Output, Input, & Variables, Calculations, Decisions Structures, Loops, and String Formatting from Chapters 1 – 5. Please read the instructions carefully before you begin. Remember to save often.

Problem #1: How much should I study outside of class?

Issue:

Your fellow students need help. This is their first year in college and they need to determine how many hours they need to study to get good grades.

Study Hours Per Week Per Class Grade

15 A

12 B

9 C

6 D

0 F

Project Specifications:

1. The user enters their full name and the number of credits they are taking, assume each class is worth 3 credits.
2. The user will then enter the grade they want, assume the same grade for all classes.
3. The program determines the number of hours they have to study, assume they will study the same number of hours for each class.
4. The program displays for each student: student’s name in *proper case*, number of credits, total number of weekly study hours, and grade they should expect to receive. In the following format –

Name: FirstName LastName

Credits: 12

Study Hours: 60

Grade: A

1. At the end of the program, the program displays the total number of students who used the program, the average credits taken, and the average study hours. In the following format –

Total Students: 3

Average Credits: 9

Average Study Hours: 20

All Project Requirements:

1. Use Project #1 design template provided in Canvas
2. The Python file name is Project1.py
   1. Submit the Python file in zybooks AND in Canvas
   2. Submit the Design document and Flowchart files in Canvas ONLY
3. Use compound conditions
4. All variables and constants names must be descriptive
5. All data must be validated!!!
   1. Make sure the datatype is correct
   2. The user can enter either upper or lower case for example their name and/or grade
   3. Make sure the data is within the proper range. For example, users shouldn’t be allowed to enter an X for the letter grade.
6. The user determines how many students they want to enter/process.
7. Follow Programming Standard Guidelines

The design/programming standards for COP1000 vary from the book. All standards outlined supersede what is in the textbook. If you have any questions on these standards, please contact your professor.

* Program designs must contain internal documentation
* Program designs must begin with a description of the program, your name, and date last modified.
  + Program designs must contain a Welcome Screen, which describes the program, and developer name.
  + Variables must be descriptive single letter variables are not permitted
* All User prompts must contain information on how the user should enter the data. Example – “Please enter the amount of the item in numerical format without the dollar sign. For example, enter 10.50 for an item that costs ten dollars and fifty cents.”