CptS 121 - Program Design and Development

Lab 2: More Problem Solving with C

Assigned: Week of May 8th, 2019
Due: At the end of the lab session

I. Learner Objectives:

At the conclusion of this programming assignment, participants should be able to:

- Analyze a basic set of requirements for a problem
- Declare variables
- Apply C data types and associated mathematical operators
- Comment a program according to class standards
- Logically order sequential C statements to solve small problems
- Compose a small C language program
- Compile a C program using Microsoft Visual Studio 2015
- Execute a program
- Create basic test cases for a program

II. Prerequisites:

Before starting this programming assignment, participants should be able to:

- Apply basic problem solving strategies
- Summarize key language elements of C including:
    1. Preprocessor directives
    2. Standard and user-defined identifiers
    3. Comments
    4. Punctuation
    5. Operators

III. Overview & Requirements:

This lab, along with your TA, will help you navigate through problem solving in C.

Labs are held in a “closed” environment such that you may ask your TA questions. Please use your TAs knowledge to your advantage. You are required to move at the pace set forth by your TA. Please help other students in need when you are finished with a task. You may program in pairs if you wish. However, I highly encourage you to collaborate in teams of 3 - 4 to analyze the problem requirements. Have a great time! Labs are a vital part to your education in CptS 121 so work diligently.

NOTE: you are not required to define your own functions for this lab! That will be next week!

Tasks: Design, implement, compile, and test C solutions to the following problems. You should always have at least the following steps in your programs: get the input(s), perform the computation(s), and display the result(s). Once you have completed a problem, demonstrate your solution to your TA.

1. Introduce or reintroduce yourself to 3 students in your lab section. This will be your team for today and possibly the rest of the semester! Complete a programming project in chapter 3 of your of the Hanly & Koffman text. The problem states the following:

Write a program that outputs the equation of the perpendicular bisector of the line segment between two points. Your program should perform the following:

- Prompt the user for the coordinates of the two points
- Compute the slope of the line between those two points
- Compute the coordinates of the midpoint of the line segment between the two points by averaging the two x coordinates and the two y coordinates
- Compute the slope of the perpendicular bisector by taking the negative reciprocal of the slope of the line segment
- Compute the y intercept of the perpendicular bisector (you now have the slope *m* of the bisector and a point (x_{mid}, y_{mid}) on the bisector, so the y intercept is $y_{mid} - m\ x_{mid}$)
- Output the labels of the original two points, and output in $y = mx + b$ format the equation of the perpendicular bisector.

Test your program on different pairs of points. Of course, there exist some pairs of points that will make your program not work. You do not have to check programmatically for these points; just consider them.

2. Write a program to calculate your body mass index (BMI). The BMI is a measurement that uses your height and weight to determine if you are underweight, a healthy weight, or overweight. Your program is required to prompt the user for weight in pounds and height in feet. The height must then be converted to inches (recall: 1 foot = 12 inches). Once the BMI has been calculated display the resultant BMI value. Use the equation below to calculate the BMI.

   BMI = ((weight in pounds) / (height in inches)^2) * 703

   Note: a BMI of less than 18 indicates you are underweight, >= 18 and < 25 means you are at a healthy weight, >= 25 and < 30 means you are overweight, and > 30 indicates obesity. You do NOT need to classify the BMI value in the program. This would require "if" statements, which you have not learned yet!

3. Write a program that calculates the Bowl Championship Series (BCS) score of a college football team. A college team's BCS ranking comprises 3 elements. These include results from the Harris Poll, the Coaches Poll, and computer rankings. Each one of these is 1/3 the final ranking. A team's score in the Harris poll is divided by 2,850, which is the maximum number of points any team can receive if all 114 voting members rank the same team as number 1. A team's score in the USA Today poll is divided by 1,475, which is the maximum number of points any team can receive if all 59 voting members rank that same team number 1. The computer rankings is calculated by a magical formula that we are not concerned with. However, we know that this value is between 0 and 1 (call this value computer_ranking). Prompt the user for a team's score in the Harris Poll (a number

between 1 and 2,850) and the Coaches Poll (a number between 1 and 1,475). Also, prompt the user for the computer ranking. Once you have all of the input values divide the Harris Poll score by 2,850 (call the result harris_poll) and the Coaches Poll score by 1,475 (call the result coaches_poll) and print out the intermediate results. The BCS total score may be determined by the following formula:

BCS_score = (harris_poll + coaches_poll + computer_ranking) / 3

Display the BCS total score. The number should be between 0 and 1. 1 would indicate a perfect score.

IV. Submitting Labs:

- You are not required to submit your lab solutions. However, you should keep them in a folder that you may continue to access throughout the semester. You should not store your solutions to the local C: drive on the EME 120/128 machines. These files are erased on a daily basis.

V. Grading Guidelines:

- This lab is worth 10 points. Your lab grade is assigned based on completeness and effort. To receive full credit for the lab you must show up on time and continue to work on the problems until the TA has dismissed you.

Resources:

- J.R. Hanly & E.B. Koffman, *Problem Solving and Program Design in C (8th Ed.)*, Addison-Wesley, 2016