

**CSC 1100**  
**Problem Solving and Programmin**  
**Fall 2014**  
**Project 01**  
**80 points**  
**Due 11/06/2014 (11:00 P.M.)**

**(80 points, 40 points each)**

**Project Problem 1:**

The cost to become a member of a fitness center is as follows:

- (a) the senior citizens discount is 30%;
- (b) if the membership is bought and paid for 12 or more months, the discount is 15%;
- (c) if more than five personal training sessions are bought and paid for, the discount on each session is 20%.

Write a menu-driven program that determines the cost of a new membership. Your program must contain a **function** that displays the general information about the fitness center and its charges, a **function** to get all of the necessary information to determine the membership cost, and a **function** to determine the membership cost. Use appropriate parameters to pass information in and out of a function. (Do not use any global variables.)

**Project Problem 2:**

Jason opened a coffee shop at the beach and sells coffee in three sizes: small (9oz), medium (12oz), and large (15oz). The cost of one small cup is \$1.75, one medium cup is \$1.90, and one large cup is \$2.00. Write a menu-driven program that will make the coffee shop operational.

Your program should allow the user to do the following:

- a. Buy coffee in any size and in any number of cups.
- b. At any time show the total number of cups of each size sold.
- c. At any time show the total amount of coffee sold.
- d. At any time show the total money made.

Your program should consist of at least the following functions: a **function** to show the user how to use the program, a **function** to sell coffee, a **function** to show the number of cups of each size sold, a **function** to show the total amount of coffee sold, and a **function** to show the total money made. Your program should not use any global variables and special values such as coffee cup sizes and cost of a coffee cup must be declared as named constants.

## **Extra Credit (70 points, 35 points each)**

### **Extra Credit Problem 1:**

A company hired 10 temporary workers who are paid hourly and you are given a data file that contains the last name of the employees, the number of hours each employee worked in a week, and the hourly pay rate of each employee. You are asked to write a program that computes each employee's weekly pay and the average salary of all the workers. The program then outputs the weekly pay of each employee, the average weekly pay, and the names of all the employees whose pay is greater than or equal to the average pay. If the number of hours worked in a week is more than 40 hours, then the pay rate for the hours over 40 is 1.5 times the regular hourly rate. Use two parallel arrays: a one-dimensional array to store the names of all the employees, and a two-dimensional array of 10 rows and 3 columns to store the number of hours an employee worked in a week, the hourly pay rate, and the weekly pay. Your program must contain at least the following functions—a function to read the data from the file into the arrays, a function to determine the weekly pay, a function to output the names of all the employees whose pay is greater than or equal to the average weekly pay, and a function to output each employee's data.

### **Extra Credit Problem 2:**

Jason, Samantha, Ravi, Sheila, and Ankit are preparing for an upcoming marathon. Each day of the week, they run a certain number of miles and write them into a notebook. At the end of the week, they would like to know the number of miles run each day, the total miles for the week, and average miles run each day. Write a program to help them analyze their data. Your program must contain parallel arrays: an array to store the names of the runners and a two-dimensional array of five rows and seven columns to store the number of miles run by each runner each day. Furthermore, your program must contain at least the following functions: a function to read and store the runners' names and the numbers of miles run each day; a function to find the total miles run by each runner and the average number of miles run each day; and a function to output the results. (You may assume that the input data is stored in a file and each line of data is in the following form: runnerName milesDay1 milesDay2 milesDay3 milesDay4 milesDay5 milesDay6 milesDay7.)

### **Restrictions:**

You must work individually. Use only material from class or from the text, chapters 1-8. All code must be the work of the individual. Do not share your code or copy from external resources.

### **Grading:**

The grade of each program will be based on the creation of a program that works correctly, up to some details (40%), the clear and efficient organization of the program (20%), the appropriate use of functions (15%), the production of clear output, with readable formatting and without unnecessary repetition (15%), and the composition of informative comments (10%). Programs must compile.

**Submission format**

Solve each problem and include the entire project of each problem in a folder. Name the folder (CSC1100\_Project1). Compressed the folder and upload it to the blackboard using the appropriate folder by the due date. No email or hard copy is accepted.