**Assignment # 3**

**Structures**

***DeadLine:18-2-2018 (11:50 pm)***

**Submission Guidelines: Paste all your code in one word file with screen shots of output for each code and submit on slate.**

**Submissions through email will not be marked.**

|  |
| --- |
| Problem 1: |

Write a program that simulates a soft drink machine. The program should use a structure called *drinkmachine* that stores the following data:

Drink Name  
Drink Cost  
Number of Drinks in Machine

The program should create a dynamic array of size four and type *drinkmachine*. The elements should be initialized with the following data:

**Drink Name Cost Number in Machine**Cola 75 20  
Root Beer 75 20  
Grape Soda 80 20  
Cream Soda 80 20

Each time the program runs, it should enter a loop that performs the following steps:

* A list of drinks is displayed on the screen.
* The user should be allowed to either quit the program or pick a drink.
* If the user selects a drink, he or she will next enter the amount of money that is to be inserted into the drink machine.
* The program should display the amount of change that would be returned and subtract one from the number of that drink left in the machine.
* If the user selects a drink that has sold out, a message should be displayed.
* When the user chooses to quit the program it should display the total amount of money the machine earned.

|  |
| --- |
| Problem 2: |

Write a program that reads students’ names followed by their test scores. Take total number of students as an input from the user. The program should output each student’s name followed by the test scores and the relevant grade. It should also find and print the highest test score and the name of the students having the highest test score.

Student data should be stored in a struct variable of type studentType, which has four components: studentFName and studentLName of type string, testScore of type int (testScore is between 0 and 100), and grade of type char. Suppose that the class has 20 students. Use an array of 20 components of type studentType.

Your program must contain at least the following functions:

a. A function to read the students’ data into the array.

b. A function to assign the relevant grade to each student.

c. A function to find the highest test score.

d. A function to print the names of the students having the highest test score.

e. A function to delete the students with the same “CPGA” and sort the list based on “CGPA” and print the final result.

Your program must output each student’s name in this form: last name followed by a comma, followed by a space, followed by the first name; the name must be left justified. Moreover, other than declaring the variables and opening the input and output files, the function main should only be a collection of function calls.

|  |
| --- |
| Problem 3: |

Write a grading program for a class with the following grading policies.

a. There are two quizzes, each graded on the basis of 10 points.

b. There is one midterm exam and one final exam, each graded on the basis of 100 points.

c. The final exam counts for 50% of the grade, the midterm counts for 25%, and the two quizzes together count for a total of 25%. (Do not forget to normalize the quiz scores. They should be converted to a percentage before they are averaged in.)

Any grade of 90 or more is an A, any grade of 80 or more (but less than 90) is a B, any grade of 70 or more (but less than 80) is a C, any grade of 60 or more (but less than 70) is a D, and any grade below 60 is an F. The program will read in the student’s scores and output the student’s record, which consists of two quiz and two exam scores as well as the student’s average numeric score for the entire course and final letter grade. Define and use a structure for the student record.

|  |
| --- |
| Problem 4: |

Create a structure called employee that contains two members: an employee number (type int) and the employee’s compensation (in dollars; type float). Ask the user to fill in this data for three employees, store it in three variables of type struct employee, and then display the information for each employee.

|  |
| --- |
| Problem 5: |

The user types in the model number, part number, and cost of a part. The program records this data in a structure. However, this structure is only one element in an array of structures. The program asks for the data for different parts, and stores it in the elements of the array. You have to ask user how many records he wants to store and declare array of that size. It then displays the information. Here’s some sample input:

Enter model number: 44

Enter part number: 4954

Enter cost: 133.45

Sample output

Model 44 Part 4954 Cost 133.45

|  |
| --- |
| Problem 6: |

1. Write a grading program for a class with the following grading policies.

a. There are two quizzes, each graded on the basis of 10 points.

b. There is one midterm exam and one final exam, each graded on the basis of 100 points.

c. The final exam counts for 50% of the grade, the midterm counts for 25%, and the two quizzes together count for a total of 25%. (Do not forget to normalize the quiz scores. They should be converted to a percentage before they are averaged in.)

Any grade of 90 or more is an A, any grade of 80 or more (but less than 90) is a B, any grade of 70 or more (but less than 80) is a C, any grade of 60 or more (but less than 70) is a D, and any grade below 60 is an F. The program will read in the student’s scores and output the student’s record, which consists of two quiz and two exam scores as well as the student’s average numeric score for the entire course and final letter grade. Define and use a structure for the student record.