Final Project       CS223        Fall 2020

Due : Sunday, November 29, 2020, 11:59 PM

Assignment objectives:   
    - Gain experience using strings, arrays, functions and their interactions   
    - Gain more experience using arrays (one/multi-dimensional)   
    - Gain more experience using File Input/Output   
    - Gain more experience using modular programming (functions)   
    - Reinforce knowledge of previous programming topics   
    - Use of structure   
    - Continue to refine C programming skills by following the C coding standard

- Learn to apply "C" programming to problem solving

Problem Description:   
Your spaceship is navigating an asteroid field; you are to design a system that will log and display the distance between your spaceship and the asteroids. Each asteroid has (x, y, z) position in space, where x, y, and z are the distances in kilometers (east-west, north-south, up-down) from your spaceship which is at 0,0,0.

Your program will read a file **asteroids.txt** that represents the ID and location of each asteroid. The distance between your spaceship and each asteroid is calculated based on asteroid’s X, Y, Z position using the following formula.

√ **X2 + Y2 + Z2**

This program must be secure. The user must input their name and pin number, if both (name and pin) matches the list you have in a text file on the disk (ident.txt), then the user gets access to this program. The program should allow 3 tries for name and password entry, if unsuccessful after 3 tries the program should print a message to the user and terminate.

Program Input/ Requirements

* The name and the pin number of the user
* A file called **ident.txt** which contains names and pin numbers. Example below.

  leia       12345   
       darth     23456   
        r2d2     34567   
        solo      45678   
        jabba    56789   
        yoda      67890

* Input to the program will be generated using random generator for X, Y, and Z coordinate. Random generator will generate numbers between 1 and 1000 (including 1 and 1000, integers). This data will be stored (write to text file) in a text file called **asteroids.txt**. Assume the speed of asteroids is 25Km/Sec. There must be at least 10 asteroids’ data in the file. The text data file **asteroid.txt** has the following format: only the numbers and IDs are in the file not the heading (ID, X,Y,Z)

ID X Y Z

A 600 200 59

B 999 999 15

C 100 1000 555

D 500 202 666

- - - -

- - - -

Program output / Requirements

* 1. Write the following data to a file called **distance.txt**, after calculating the distance

A 600 200 59 (calculated distance)

B 999 999 15 (calculated distance)

C 100 1000 555 (calculated distance)

D 500 202 666 - - - -

- - - - - - - -

* 1. Print an alarm message to screen with X, Y, and Z positions of asteroids within a distance of less than 750 km.

 The alarm message could look like the following: only two digits after the decimal point.

**Warning - Warning - Warning**

**Nearest asteroid B at ??, ??, ??: ?? km away**

**Time to impact ??? seconds**

**asteroid B at at ??, ??, ??: ?? km away**

**asteroid D at at ??, ??, ??: ?? km away**

**- - - - - - - - - -  - - - - - -- - - - - - - - - - - - - -- - - - - -**   
**- - - - - - - - - -  - - - - - -- - - - - - - - - - - - - -- - - - - -**

3- Display a bar chart (character) for distance of the asteroids.

Example (something like below):

A |\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

B |\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

C |\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

D |\*\*\*\*\*\*\*

E |\*\*\*\*\*\*\*\*\*\*\*\*

F |

|

|

0------|------|------|------|------|------|------|-------------------------------------------------

Scale appropriately

1. Rank the asteroids from closest to farthest. That means sorting an array of structures based on the distance member of the structure.
2. Display ranking with the asteroid ID

Example:

Asteroid C distance (must be the closest distance)

Asteroid E distance (must be the next to the closest)

And so on…

More requirements, In addition to our standards:

1. The **ident.txt** file can contain at least 10 names and pin numbers.
2. Your program should be able to process the **minimum of 10 asteroids.**
3. You must assign information read from file **ident.txt** to array of structures then process the data.
4. You assign asteroids information that was read from **asteroid.txt** file to an array of structures then use the array of structures to process data.
5. You must use pointers to process array of structures
6. The main function must do as little as possible other than calling other functions and passing parameters to those functions.
7. Your program should be **modularly designed** with functions designed to do one task and one task well.
8. Do not use global variables. Pass data back and forth via parameters or as return values.
9. Make your functions as general as possible so that they can be called more than once if needed.
10. Comment your code
11. Use few lines of comments at the beginning of each function including main function the functions job.
12. Use defined constants for ALL constants in your program.

General requirement

1. Following comments must be the first lines of your source code

/\* -----------------------------------------------

Submitted By: <Your Name>

Homework Number: <whichever homework you are doing>

Credit to:

<Name or names>

Submitted On: <Date>

By submitting this program with my name,

I affirm that the creation and modification

of this program is primarily my own work.

------------------------------------------------ \*/

1. Next few line of your source code must be comments describing what this program is supposed to do.
2. The first output (display) must be your name
3. The source code file must include comments documenting the design.
4. Indentation and spacing should be used to make the program readable.
5. Throughout this class (including this program you are not permitted to use the following commands: continue, break, exit, or any library functions that perform similarly

**Deliverables:**

Delivery Requirements:   
For this project you must submit a report (in Canvas) that includes the following sections:

* Design, block diagram of functions used.
* Flowchart for every function

**IMPORTANT: You must name your source code as below:**

**Yourusername\_homeworknumber\_cs223**

**Example: behif\_h1\_cs223**