**CSCI 1410 – PA 2 – Magic Eight Ball (40 points)**

Have you ever wanted to predict the future? Well the Magic Eight Ball does just that. The original game was a softball sized “8-ball”. You would ask a question, shake it up and look at the result. There are 20 responses…10 positive, 5 negative, and 5 are vague. For this project, we want to recreate this, but give the ability to read in a set of responses, and add additional responses, and print out all of the responses in alphabetical order. Of course, we have to give seemingly accurate responses, which we will do by giving a random response.

**Program Details:**

You should have a menu with five lettered options. You should accept both capital and lower case letters in your menu options. The menu should do the task, then return to the menu (except in the case of exit). Any incorrect responses should get an error message, followed by a reprint of the menu.

1. Read responses from a file
2. Play Magic Eight Ball
3. Add responses to a file
4. Print out responses alphabetically
5. Print out responses by type (positive, negative, vague)
6. Write responses to a file
7. Exit

Each menu item must be implemented using a function or sets of functions with appropriate input parameters and return values. Functions will have a prototype in a file called functions.h and defined in a file called functions.cpp. Also, you will have a struct that keeps track of the response and whether it is positive, negative, or vague. That struct will also be in the functions.h file.

**Implementation Detail 1:** Remember you will need to #include “functions.h” into both your main.cpp and functions.cpp but do not include .cpp files

**Implementation Detail 2:** You should have an “duplicate guard” in your functions.h like:  
#ifndef FUNCTIONS\_H  
#define FUNCTIONS\_H  
//your code here  
#endif

**Implementation Detail 3:** In most IDEs, you need to make sure all of your files are included in the project. For VS and csegrid, make sure all files are in the same directory. For XCode, you will need to place them in the same directory, then go to Product->Scheme->Edit Scheme and Use custom working directory where your .cpp and .txt files are located. For the csegrid run your program with  
g++ -o pa2.o main.cpp functions.cpp. Then run with ./pa2.o

**Implementation Detail 4:** You must use (at least) one **array**, and **struct**. You may not use vectors or classes.

**Implementation Detail 5:** You can not have any global variables. So you will need to declare variables in main, then pass them (as appropriate) to your functions. Remember that there is no built in size for arrays, so you will have to pass the size (and MAXSIZE when trying to add to an array)

**Implementation Detail 6:** You will need to choose a random response. See the example at <http://www.cplusplus.com/reference/cstdlib/rand/>

Note: You need some additional #include statements. You will use srand ONLY once, then use rand() with the modulus operator (%) to get a number between 0 and size -1

**Extra Credit:** If you have ALL of the functionality listed above, you can get up to 5 points extra credit (So a 45/40) if you implement a Delete Answers function. (Instead of an actual delete you would need to shift everything one character to the left. Make sure to list that in your status if you want the extra credit.

**Implementation Detail 6:** Responses are as follows:

It is certain

positive

It is decidedly so

positive

Without a doubt

positive

Yes definitely

positive

You may rely on it

positive

As I see it, yes

positive

Most likely

positive

Outlook good

positive

Yes

positive

Signs point to yes

positive

Reply hazy try again

vague

Ask again later

vague

Better not tell you now

vague

Cannot predict now

vague

Concentrate and ask again

vague

Don't count on it

negative

My reply is no

negative

My sources say no

negative

Outlook not so good

negative

Very doubtful

negative

**IDE and Deliverables:**

You may use any individual development environment (IDE) that you choose (VS, XCode, csegrid, eclipse), but your code must run on the csegrid. Deliver one file in the format lastnameFirstinitialPA2.zip (containing a main.cpp, functions.h and functions.cpp) to canvas by the due date. Make sure to update the status in the comments to include what IDE you developed it in and whether it fully runs as directed, or has issues/error or only partially works.