CSIS 112 – Lab 8 Battleship Instructions

For this assignment, you are to recreate the last lab in CSIS 111 – Battleship. The difference between the CSIS 111 lab and this one, however, is in how you are to implement it.

In this course, you have learned about tools and techniques that will allow you to write sophisticated software: classes, inheritance, polymorphism, random number generators, pointers, double pointers, the array class, operator overloading, templates, file processing, exception handling, etc.

Some ideas for implementation include creating classes for the grid and ships, creating a ship inheritance hierarchy, using polymorphism to do something with each ship (i.e. place it in its initial location, check its coordinates, etc.), using double pointers to represent the grid(s), generating initial locations of ships on the grid using random numbers, and the list goes on. Use your imagination and be creative.

In addition to writing the code, write a summary (in a Word document) of how you chose to implement your game using the tools and techniques listed above. Be sure to create a list (use bullet points or numbers) in your summary so that I can easily identify how and where you implemented each feature in your code.

You must create a text file called aareadme.txt and put it at the top level of your zipped folder. It should provide instructions on how to use your program.

Here are the general instructions for creating the Battleship game that you did in CSIS 111:

Imagine we are using a two-dimensional array as the basis for creating the game *battleship*. In the game of battleship, a ‘ ~ ‘ character entry in the array represents ocean (i.e., not a ship), a ‘#’ character represents a place in the ocean where part of a ship is present, and a ‘H’ character represents a place in the ocean where part of the ship is present and has been *hit* by a torpedo. Thus, a ship with all ‘H’ characters means the ship has been sunk. Declare a two-dimensional array that is 25 X 25 that represents the entire ocean and an If statement that prints “HIT” if a torpedo hits a ship given the coordinates X and Y. Write a C++ program that will read in a file representing a game board with 25 lines where each line has 25 characters corresponding to the description above. An example file might look like:

~~~~~~~~~~~~~~~~~~~~~~~~~  
~~~~~~~~~~~~~~~~~~~~~~~~~  
~~~~~~~~~~~~~~~~~~~~~~~~~  
~~~~~~#####~~~~~~~~~~~~~~  
~~~~~~~~~~~~~~~~~~~~~~~~~  
~~~~~~~~~~~~~~#~~~~~~~~~~  
~~~~~~~~~~~~~~#~~~~~~~~~~  
~~~~~~~~~~~~~~#~~~~~~~~~~  
~~~~~~~~~~~~~~#~~~~~~~~~~  
~~~~~~~~~~~~~~#~~~~~~~~~~  
~~~~~~~~~~~~~~~~~~~~~~~~~  
~~~~~~~~~~~~~~#####~~~~~~  
~~~~~~~~~~~~~~~~~~#~~~~~~  
~~~~~~~~~~~~~~~~~~#~~~~~~  
~~~~~~~~~~~~~~~~~~#~~~~~~  
~~~~~~~~~~~~~~~~~~#~~~~~~  
~~~######~~~~~~~~~#~~~~~~  
~~~~~~~~~~~~~~~~~~~~~~~~~  
~~~~~~~~~~~~~~~~~~~~~~~~~  
~~~~~~~~~~~~~~~~~~~~~~~~~  
~~~~~~~~~~~~~~~~~~~~~~~~~

You should write a function called Fire that will take an X and Y coordinate and print “HIT” if a ship is hit and “MISS if a ship is missed. If a ship is HIT you should update the array with an ‘H’ character to indicate the ship was hit. If a ship is hit that has already been hit at that location you should print “HIT AGAIN”. You should write a second function called **FleetSunk** that will determine if all the ships have been sunk. Your C++ program must then call these functions until all the ships have been sunk at which point the program should display “The fleet was destroyed!”.

Deliverables:

Submit your entire zipped solution according to the normal lab submission policies by 11:59PM on Wednesday, May 3, 2017.

The aareadme.txt file and Word document summary must be included in your zipped folder to receive full credit for this lab.