Assigment 3: Project Plan John Waczak

CS 161 Date: May 30, 2018

Tim Alcon

5a

Testing Plan

test	expected output
input: 2, -19, 4	output -19, 2, 4
input: valgrind ./transformArray	output No memory leaks
input: 1, 1, 1	output 1, 1, 1

Pseudo-code

```
define void function called transformArray that takes
two arguments a reference to a pointer to an array of
integers and an integer for the size of the array

decalare a new dynamic array twice the size of the old

loop through the indices of the new array
        if the index is less than the size
                add the old array element to the new one
        when the index is past the old size
                add the old value at index−size to array
                plus 1

delete the old array to free the memory
set the old array equal to the new array
```

5b

Testing Plan

test	expected output
input: place ship at (0,9) R	output: invalid move
input: place ship at (9, 0) C	output: invalid move
input: place ship at (0, 9) C	output: valid move
input: place ship at (9,0) R	output: valid move
input: attack (anything within bounds)	output true

Psuedo-code

```
create ship class with the following attributes
        string name
        int length
        int damage
with the following methods
        constructor that sets name, length, and puts the
          damage to 0

        get methods for all of the attributes
```

take hit method that increments damage by one

create a board class with the following attributes
      10x10 bool array for keeping track of what has been attacked

      10x10 ship pointer array that stores address of ship occupying each position on board

      int shipsUnsunk

with the follwing methods
      constructor that initializes bools to false, pointers to NULL

      getAttacksArrayElement that returns element in bool array for a given (i,j) pair of int coordinates

      getShipsArrayElement that returns pointer to ship given a specific (i,j) pair of int coordinates. Should return NULL if nothing occupies that position

      getNumShipsRemaining that returns shipsUnsunk

      placeShip -- takes a ship object, (i,j) corresponding to position closest to (0,0) top left corner of board, and either R for row or C for column to specify orientation of the board. first make sure (i,j) are valid positions on 10x10 board second we want to check if the ship will fit with given choice of orientation and position. Notice that it is sufficient to check whether or not the opposite side of the ship is on the board to determine if the move is legal as the ship either fits or it doesn't.
            if user picks R for row check to see if (i + length of ship) is valid position on board.

            if user picks C for column check if (j + length of ship) is a valid position on board.

            if that all checks out then define a pointer that points to the ship. Loop from 0 to length of ship−1 and add poniter for corresponding position to the pointer array then add one to shipsUnsunk and return true
            return false if any of the previous checks fail

attack –– takes an (i, j) coordinate pair. Check to see if that position has been hit before in bool array. If it has, that's an invalid move so don't do anything, just return true. Otherwise, turn that position in bool array to true.

When the bool is set to true, get the call the takeHit method for the corresponding ship using the pointer in the pointer array.

If that ship's damage is equal to it's length print out "you sank (ship name)!". Decrease shipsUnsunk by 1.

allShipsSunk –– loop through 10x10 pointer array. If the pointer is not null and that ship's length is equal to it's damage continue. If this fails ever return false. If we make it through the whole array then return true (the game is over).

General question: Since we can't have a main, can we add attributes to these classes if we want to? For example I am thinking I want to add something like an isSunken boolean to the ship class to keep track of whether or not the ship has been sunk before.