**PACE Project Pseudocode**

**Written by: Hunter Pollock, Zhengyu Jiang, and Benjamin Johnson**

**Computer Science I**

**Last Modified: 4/23/21**

**Page 1 of 7**

declare an enumeration called Ships, holding the values {MINESWEEPER = 2, FRIGATE = 3, CRUISER = 4, BATTLESHIP = 5}

define a new two-dimensional char array type called GameBoard

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

**Page 2 of 7**

begin function main

declare GameBoard variable board

call function InitBoard

call function PlayGame

return 0

end function main

----------------------------------------------------------------

**Page 3 of 7**

*Initializes board with values*

*Pre: None*

*Post: board is filled with char values*

begin function InitBoards

pass in GameBoard variable board from function call

for each row in the game board

for each column in the game board

set the value of the board at the specified row and column equal to ‘~’

for each value in Ships

//Will put in FRIGATE in two function calls since there are two Frigate pieces

call function PlaceShip

end function InitBoard

----------------------------------------------------------------

**Page 4 of 7**

*Places a ship into the gameBoard array in a randomized location based on ship type*

*Pre: None*

*Post: A ship is placed onto board in a randomized, valid location*

begin function PlaceShip

pass in Ships variable shipType and GameBoard variable board from function call

declare int variables row and column

declare bool variable isVertical

declare bool variable isValid and set it equal to false

assign isVertical to be either true or false based on a randomly assigned number of either 0 or 1

if shipType is equal to MINESWEEPER

if isVertical is equal to true

assign row a random value between 0 and 8

assign column a random value between 0 and 9

while isValid is equal to false

if any of the rows needed to place the ship already have a ship (the member value is equal to ‘#’)

assign column a random number between 0 and 9

else if the placement is valid

set isValid equal to false

place the ship on the corresponding place on the board (member values now equal ‘#’)

else //ship will be placed horizontally

assign row a random value between 0 and 9

assign column a random value between 0 and 8

while isValid is equal to false

if any of the columns needed to place the ship already have a ship (the member value is equal to ‘#’)

assign row a random number between 0 and 9

else if the placement is valid

set isValid equal to false

place the ship on the corresponding place on the board (member values now equal ‘#’)

else if shipType is equal to FRIGATE

if isVertical is equal to true

assign row a random value between 0 and 7

assign column a random value between 0 and 9

while isValid is equal to false

if any of the rows needed to place the ship already have a ship (the member value is equal to ‘#’)

assign column a random number between 0 and 9

else if the placement is valid

set isValid equal to false

place the ship on the corresponding place on the board (member values now equal ‘#’)

else //ship will be placed horizontally

assign row a random value between 0 and 9

assign column a random value between 0 and 7

while isValid is equal to false

if any of the columns needed to place the ship already have a ship (the member value is equal to ‘#’)

assign row a random number between 0 and 9

else if the placement is valid

set isValid equal to false

place the ship on the corresponding place on the board (member values now equal ‘#’)

else if shipType is equal to CRUISER

if isVertical is equal to true

assign row a random value between 0 and 6

assign column a random value between 0 and 9

while isValid is equal to false

if any of the rows needed to place the ship already have a ship (the member value is equal to ‘#’)

assign column a random number between 0 and 9

else if the placement is valid

set isValid equal to false

place the ship on the corresponding place on the board (member values now equal ‘#’)

else //ship will be placed horizontally

assign row a random value between 0 and 9

assign column a random value between 0 and 6

while isValid is equal to false

if any of the columns needed to place the ship already have a ship (the member value is equal to ‘#’)

assign row a random number between 0 and 9

else if the placement is valid

set isValid equal to false

place the ship on the corresponding place on the board (member values now equal ‘#’)

else //shipType is equal to BATTLESHIP

if isVertical is equal to true

assign row a random value between 0 and 5

assign column a random value between 0 and 9

while isValid is equal to false

if any of the rows needed to place the ship already have a ship (the member value is equal to ‘#’)

assign column a random number between 0 and 9

else if the placement is valid

set isValid equal to false

place the ship on the corresponding place on the board (member values now equal ‘#’)

else //ship will be placed horizontally

assign row a random value between 0 and 9

assign column a random value between 0 and 5

while isValid is equal to false

if any of the columns needed to place the ship already have a ship (the member value is equal to ‘#’)

assign row a random number between 0 and 9

else if the placement is valid

set isValid equal to false

place the ship on the corresponding place on the board (member values now equal ‘#’)

end function ModifyShipBoard

----------------------------------------------------------------

**Page 5 of 7**

*Lets the user play Battleship through a menu system*

*Pre: gameBoard has been declared and has values associated with each member. 5 ships have been placed into the board in valid locations*

*Post: The user plays Battleship until they win the game or forfeit*

begin function PlayGame

pass in GameBoard variable board from function call

declare char variable userInput

declare int variable rowGuess and columnGuess

declare int variable shipsSunk and set it equal to 0

declare bool variable isFinished and set it equal to false

declare bool variable isValidInput and set it equal to false

call function PrintBoard

while isFinished is false

output to console “Guess the row (1-10): \n“

read user input into variable rowGuess

output to console “Guess the column (1-10): \n”

read user input into variable columnGuess

if function CheckUserGuess returns true

change the element at row rowGuess and column columnGuess to ‘H’

output to console “Hit!\n”

else

change the element at row rowGuess and column columnGuess ‘.’

output to console “Miss!\n”

while isValidInput equals false

output to console “Would you like to continue? Type Y or N for your answer:\n”

read user input into userInput

if userInput is equal to Y

break from if statement

set isValidInput equal to true

else if userInput is equal to N

call function RevealShips

output to console “Here’s where the ships were:\n”

call function PrintBoard

set isFinished equal to true

set isValidInput equal to true

else

output to console “Error! Bad input detected. Please try again...\n”

set isValidInput equal to false

set isValidInput equal to false

end function PlayGame

----------------------------------------------------------------

**Page 6 of 7**

*Prints the game board to the console,with ship locations revealed if printShipLocations is true*

*Pre: The board passed in is defined and has values associated with each member*

*Post: The board is printed to the console*

begin function PrintBoard

pass in GameBoard variable board and bool printShipLocations from function call

for each row in the board

for each column in the board

if the value of board at the specified row and column is equal to ‘#’ and printShipLocations is equal to false

Output ‘~’ to the console

else

Output the value of the board at the specified row and column to the console

move to the next line of output in the console

end function PrintBoard

----------------------------------------------------------------

**Page 7 of**

*Checks the user’s guess on where the ship is, returning true if they “hit” the ship or false if they “missed”*

*Pre: board has been defined and has values associated with each member. column and row are valid indexes of board (0-9)*

*Post: The function returns a boolean signifying if the user guessed right or not*

begin function CheckUserGuess

pass in GameBoard variable board, and int variables column and row from function call

if the member of board at row row and column column is equal to true

set the member of board at row row and column column equal to ‘H’

return true

return false

end function CheckUserGuess

----------------------------------------------------------------