Title

Project 2

BATTLESHIP Game

Course

CSC-5

42829

Due Date

June 6, 2016

Author

Nicolas Gamez

**Introduction**

Title: Bombing Game

Bombing game is played by put six ship on the board and guessing the location of enemy ship on a board that has been put on the board. Player would try to guess rows and column and bombing the enemy ships. The board is 10 X5 and it is slip for both the player and the ai. When a ship hit it is mark by an X and when it is miss it mark by an O. The game is over when one of the player don’t have any ships left.

Summary

Project size: about 207lines

The number of variables: about 23

The number of method: 7

The project show many concepts that we learned from chapters in the book

And class.

Description

The main point that I programmed this is to show what I learn and how to apply it in a game to play.



Major Variables

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Variable Name | Description | Location |
| char | FREE | “ |” | int main(int argc, char\*\* argv) |
|  | yn | ‘Y’ | int main(int argc, char\*\* argv) |
|  | Player\_grid | [ROW\_SIZE][COL\_SIZE] | int main(int argc, char\*\* argv) |
|  | Ai\_real | [ROW\_SIZE][COL\_SIZE] | int main(int argc, char\*\* argv) |
|  | Ai\_fake | [ROW\_SIZE][COL\_SIZE] | int main(int argc, char\*\* argv) |
| Char const | Ships[SHIPS] | Show where they place a boat | int main() |
|  | SHIP[SHIPS] | Show where you place the boat | Int main() |
| Const in | ROW\_SIZE = 5 | board | Int main() |
|  | const int COL\_SIZE = 10; | board | Int main() |
| Bool | win | When a player wins | bool win() |
| int | eSunk | Enemy ship | int main(int argc, char\*\* argv) |
|  | X,Y | X axis  Y axis | int main(int argc, char\*\* argv) |
|  | i | 0-ships | for (int i = 0; i < SHIPS; i++) |
|  | colmn | 0-9 | for (int colmn = 0; colmn < 10; colmn++) |
|  | row | 0-9 | for (int row = 0; row < 10; row++) |
|  | es | Enemy ship | Bool win() |
|  | as | Your ship | Bool win() |
|  | k | 0-20 | for (int k = 0; k < 20; k++) |
|  |  |  |  |

Reference

1. textbook-Gaddis 8ThED

Program

/\*

\* File: prject 2

\* Author: nicolas gamez

\* Created on April 28, 2016, 2:47 PM

\*/

//system libraries

#include <iostream>

#include <iomanip>

#include <cmath>

#include <cstdlib>

#include <ctime>

#include <fstream> //Writing to a file

using namespace std;

//User Libraries

//Global Constants

const int COL\_SIZE = 10;

const int SHIPS = 6;

const int SHIPE = 6;

const int ROW\_SIZE = 5;

//Function Prototypes

void print(char[][COL\_SIZE], char[][COL\_SIZE], const int);

bool win(int,int);

//Execution Begins Here

int main(int argc, char\*\* argv) {

const char SHIP[SHIPS] = {'S', 'S', 'S', 'S', 'S', 'S'};

const char Ship[SHIPE] = {'E', 'E', 'E', 'E', 'E', 'E'};

const int ROW\_SIZE = 5;

unsigned seed = time(0);

srand(seed);

//Declare Variables

char FREE = '|';

int eSunk = 0; // you sunk these

int aiSunk = 0; // ai sunk these

bool won;

int x, y;

char yn='y';

char player\_grid[ROW\_SIZE][COL\_SIZE];

char ai\_fake[ROW\_SIZE][COL\_SIZE]; //display to the player

char ai\_real[ROW\_SIZE][COL\_SIZE]; //store the ai Ships

do{

//initialize

for (int row = 0; row < ROW\_SIZE; row++) {

for (int colmn = 0; colmn < 10; colmn++) {

player\_grid[row][colmn] = ' ';

ai\_fake[row][colmn]= ' ';

ai\_real[row][colmn]= ' ';

}

}

print(player\_grid, ai\_fake, ROW\_SIZE);

//Players

for (int i = 0; i < SHIPS; i++) {

//ask them to set up their ships

do {

do {

cout << "place your ship " << i + 1 << "(input format col row e.g. 0 0 it a 9X4 grid)" << endl;

cin >> x>>y;

} while (x > 9 || x < 0 || y > 4 || y < 0);

if (player\_grid[y][x] == ' ') {

player\_grid[y][x] = SHIP[i];

print(player\_grid, ai\_fake, ROW\_SIZE);

} else {

cout << "overlap\n";

}

} while (player\_grid[y][x] != SHIP[i]);

}

//ai placement

for (int i = 0; i < SHIPS; i++) {

//ask them to set up their ships

do {

x = rand() % 10;

y = (rand() % 5);

if (ai\_real[y][x] == ' ') {

ai\_real[y][x] = Ship[i];

}

} while (ai\_real[y][x] != Ship[i]);

}

print(player\_grid, ai\_fake, ROW\_SIZE);

cout << endl;

//then if they hit anything mark it in the grid[ row ][ column ]

do{

cout << "Please enter your coordinates:";

cout << "Enter column(0-9): ";

cin.ignore();

cin>>y;

cout << "Enter row(0-4): ";

cin >> x;

cout << endl;

while(y > 9 || y < 0 || x > 4 || x < 0){

cout << "Please enter your coordinates:";

cout << "Enter column(0-9): ";

cin.ignore();

cin>>y;

cout << "Enter row(0-4): ";

cin >> x;

}

if (ai\_real[x][y] == 'E') {

eSunk++;

ai\_fake[x][y] ='X';

cout << "It was a Hit! \n";

} else {

ai\_fake[x][y] = 'O';

cout << "It was a Miss! \n";

}

print(player\_grid, ai\_fake, ROW\_SIZE);

//AI gus

x = rand() % 5;

y = rand() % 10;

cout << x << ", " << y << endl;

if (player\_grid[x][y] == 'S') {

aiSunk++;

player\_grid[x][y] ='X';

cout << "It was a Hit! \n";

} else {

player\_grid[x][y] = 'O';

cout << "It was a Miss! \n";

}

print(player\_grid, ai\_fake, ROW\_SIZE);

}while (!win( eSunk, aiSunk ));

do

{

cout << "DO you want to play again?? : Enter a 'y' or 'n' " << endl;

cin >> yn;

if(yn!='y'&& yn!='Y'&&yn!='n'&&yn!='N')

cout<< "I only respond to the letters 'y' or 'n'"<< "Try again!!"<< endl;

}while(yn!='y' && yn!='Y' && yn!='n' && yn!='N');

}while (yn=='y' || yn=='Y');

cout << "you now quit the game" << endl;

return 0;

}

bool win(int es, int as ) {

if ( es == 6) {

cout << "you win" << endl;

return true;

} else if( as ==6 ) {

cout << "you lose" << endl;

return true;

}else{

return false;

}

}

void print(char grid[][COL\_SIZE], char ai[][COL\_SIZE], const int ROW\_SIZE) {

cout << "Player\n ";

for (int i = 0; i < 10; i++) {

cout << " " << i;

}

cout << endl;

cout << " ";

for (int k = 0; k < 20; k++) {

cout << "-";

}

cout << endl;

for (int row = 0; row < ROW\_SIZE; row++) {

cout << row << " |";

for (int colmn = 0; colmn < COL\_SIZE; colmn++) {

cout << grid[row][colmn] << "|";

}

cout << endl;

}

cout << "AI" << endl;

for (int row = 0; row < ROW\_SIZE; row++) {

cout << row << " |";

for (int colmn = 0; colmn < COL\_SIZE; colmn++) {

cout << ai[row][colmn] << "|";

}

cout << endl;

}

cout << " " << endl;

}