using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Threading;

namespace Battleships

{

class Program

{

static void Main(string[] args)

{

//dev

while (true)

{

int size = 10;

int[] ship\_nums = new int[4];

ship\_nums[0] = 1;

ship\_nums[1] = 2;

ship\_nums[2] = 1;

ship\_nums[3] = 1;

string[,] display = new string[10, 10];

string[,] enemies = new string[10,10];

Grid\_Size(ref size, ref display, ref enemies);

Place\_Enemies(size, enemies, ship\_nums);

Display(size, display);

Attack(size, ref enemies, ref display);

Console.ReadLine();

}

//dev

while (true)

{

int choice;

Console.WriteLine("Main Menu:");

Console.WriteLine("1. Guessing Practice");

Console.WriteLine("2. Player vs AI");

Console.WriteLine("3. 2 Player");

Console.WriteLine("4. Settings");

Console.WriteLine("5. Quit");

choice = Convert.ToInt32(Console.ReadLine());

switch (choice)

{

case 1:

Console.Clear();

//goto guessing practice

break;

case 2:

Console.Clear();

//goto player vs AI

break;

case 3:

Console.Clear();

//goto 2 player

break;

case 4:

Console.Clear();

//goto settings

break;

case 5:

Console.Clear();

//quit

System.Environment.Exit(0);

break;

}

}

}

static void Grid\_Size(ref int size, ref string[,] display, ref string[,] enemies)

{

size = Convert.ToInt32(Console.ReadLine());

string[,] \_display = new string[size, size];

for (int x = 0; x < size; x++)

{

for (int y = 0; y < size; y++)

{

\_display[y, x] = " -";

}

}

display = \_display;

enemies = \_display;

}

//set grid size

static void Display(int size, string[,] display)

{

Console.Write(" ");

for (int i = 0; i <= size - 1; i++)

{

const string letters = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";

var value = "";

if (i >= letters.Length)

value += letters[i / letters.Length - 1];

value += letters[i % letters.Length];

Console.Write(" " + value);

}

Console.WriteLine();

for (int x = 0; x < size; x++)

{

if ((x + 1) < 10)

{

Console.Write(" " + (x + 1));

}

else

{

Console.Write(x + 1);

}

for (int y = 0; y < size; y++)

{

Console.Write(display[y, x]);

}

Console.WriteLine();

}

}

//display grid

static void Place\_Enemies(int size, string[,] enemies, int[] ship\_nums)

{

for (int length = 2; length <= 5; length++)

{

for (int a = 1; a <= ship\_nums[length - 2]; a++)

{

int[,] temp = new int[2, 5];

Boolean ship\_placed = false;

while (ship\_placed == false)

{

Random rnd = new Random();

int x = rnd.Next(size);

int y = rnd.Next(size);

int dir = rnd.Next(1, 5);

for (int i = 0; i <= length - 1; i++)

{

switch (dir)

{

case 1:

temp[1, i] = x;

temp[0, i] = y + i;

break;

case 2:

temp[1, i] = x + i;

temp[0, i] = y;

break;

case 3:

temp[1, i] = x;

temp[0, i] = y - i;

break;

case 4:

temp[1, i] = x - i;

temp[0, i] = y;

break;

}

}

ship\_placed = true;

for (int i = 0; i <= length - 1; i++)

{

if (temp[0, i] >= size || temp[0, i] <= 0 || temp[1, i] >= size || temp[1, i] <= 0)

{

ship\_placed = false;

}

else if (enemies[temp[1, i], temp[0, i]] == " #")

{

ship\_placed = false;

}

}

}

for (int b = 0; b <= length - 1; b++)

{

enemies[temp[1, b], temp[0, b]] = " #";

}

}

}

}

//place enemies

static void Attack(int size, ref string[,] enemies, ref string[,] display)

{

Console.WriteLine("Enter coordinate (e.g. 3E)");

Console.WriteLine();

string coord = Console.ReadLine();

int y = Convert.ToInt32(Convert.ToString(coord[0])) - 1;

int x = ((int)coord[1] % 32) - 1;

if ((x > 0 && x < size) && (y > 0 && y < size))

{

if (enemies[y, x] == " #")

{

Console.WriteLine("HIT!");

enemies[y, x] = " X";

display[y, x] = " X";

}

}

else

{

Console.WriteLine("Invalid Coordinte");

}

}

//attack

}

}