namespace ConsoleApp1

{

internal class Program

{

protected static int origRow;

protected static int origCol;

protected static void WriteAt(string s, int x, int y)

{

try

{

Console.SetCursorPosition(origCol + x, origRow + y);

Console.Write(s);

}

catch (ArgumentOutOfRangeException e)

{

//Console.Clear();

Console.WriteLine("YOUR OUT OF BOUNDS HACKER >:(");

}

}

static void Main(string[] args)

{

int x = 10;

int y = 10;

int[,] grid = new int[y, x];

Player player = new Player();

BuildBorder(grid, player);

while (true)

{

Player(grid, player);

Tail(player);

}

}

static void BuildBorder(int[,] grid, object instance)

{

Player player = (Player)instance;

//building in the grid

int sizeY = grid.GetLength(0);

int sizeX = grid.GetLength(1);

for (int i = 0; i < sizeY; i++)

{

if (i == 0 || i == sizeY - 1)

{

for (int j = 0; j < sizeX; j++)

{

grid[i, j] = 1;

}

}

for (int j = 0; j < sizeX; j++)

{

if (j == 0 || j == sizeX - 1)

{

grid[i, j] = 1;

}

}

}

//displaying the visual

int Y\_off = player.Yoff;

int X\_off = player.Xoff;

int y\_size = grid.GetLength(0);

int x\_size = grid.GetLength(1) \* 2;

//WriteAt("██", j, i);

for (int i = 0; i < y\_size; i++)

{

if (i == 0 || i == y\_size - 1)

{

for (int j = 0; j < x\_size; j += 2)

{

WriteAt("[]", j + X\_off, i + Y\_off);

}

}

else

{

for (int j = 0; j < x\_size; j += 2)

{

if (j == 0 || j == x\_size - 2)

{

WriteAt("[]", j + X\_off, i + Y\_off);

}

}

}

}

}

static void Player(int[,] grid, object instance)

{

Player player = (Player)instance;

int Y\_off = player.Yoff;

int X\_off = player.Xoff;

WriteAt(" ", 0, 0);

var input = ConsoleKey.Enter;

if (Console.KeyAvailable == true)

{

input = Console.ReadKey().Key;

grid[player.y, player.x] = 0;

//WriteAt(" ", player.x \* 2 + X\_off, player.y + Y\_off);

}

switch (input)

{

case ConsoleKey.E:

//Swing(grid, player); break;

player.lengthX.Enqueue(player.x);

player.lengthY.Enqueue(player.y);

break;

case ConsoleKey.W:

player.y--;

if (grid[player.y, player.x] == 1)

{

player.y++;

}

break;

case ConsoleKey.A:

player.x--;

if (grid[player.y, player.x] == 1)

{

player.x++;

}

break;

case ConsoleKey.S:

player.y++;

if (grid[player.y, player.x] == 1)

{

player.y--;

}

break;

case ConsoleKey.D:

player.x++;

if (grid[player.y, player.x] == 1)

{

player.x--;

}

break;

}

grid[player.y, player.x] = 2;

WriteAt("[]", player.x \* 2 + X\_off, player.y + Y\_off);

}

static void Swing(int[,] grid, object instance)

{

Player player = (Player)instance;

int x = player.x \* 2;

int y = player.y;

WriteAt("]>-----------------", x + 2, y + 1);

Thread.Sleep(120);

WriteAt(" ", x + 2, y + 1);

for (int i = player.x; i < grid.GetLength(0); i++)

{

grid[i, y] = 0;

}

}

static void Tail(object instance)

{

Thread.Sleep(100);

Player player = (Player)instance;

int x = player.x;

int y = player.y;

player.lengthX.Enqueue(x);

player.lengthY.Enqueue(y);

WriteAt("[]", x \* 2, y);

try

{

WriteAt(" ", player.lengthX.Dequeue() \* 2, player.lengthY.Dequeue());

}

catch

{

}

}

}

}