САНКТ-ПЕТЕРБУРГСКОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАНИЕ   
«КОЛЛЕДЖ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ»

**ОТЧЕТ**

**по практической работе №17-19**

**«Морской бой»**

Выполнил студент 41 гр.

Березко В.В.

Преподаватель: Фомин А. В.

Оценка: \_\_\_\_\_\_\_\_\_

Санкт-Петербург 2016

**Оглавление**

[1. Цель работы 3](#_Toc467707416)

[2. Ход выполнения работы 4](#_Toc467707417)

[3. Заключение 8](#_Toc467707418)

# Цель работы

Получить навыки по разработке клеточных игр с возможностью взаимодействия через локальную сеть, а также навыки реализации игрового процесса и его отображения с помощью компонента pictureBox.

# Ход выполнения работы

Для создания игрового приложения «Морской бой» мы разместили на форме следующие компоненты:

2 pictureBox для отображения нашего игрового поля и игрового поля оппонента

tabControl с кнопками и полями ввода для управления сетевым соединением

listBox, 2 radioBatton и 2 кнопки для расстановки кораблей по полю

еще один listBox для отображения ходов и текущего сотояния.

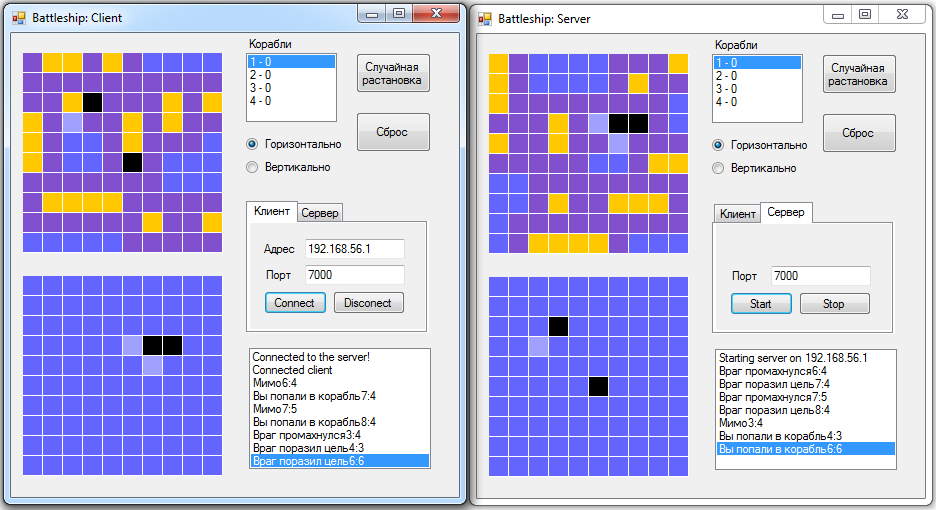


Рисунок - Две главных формы запущенные на одном компьютере

По окончанию игры победителю и проигравшему выводится сообщение:

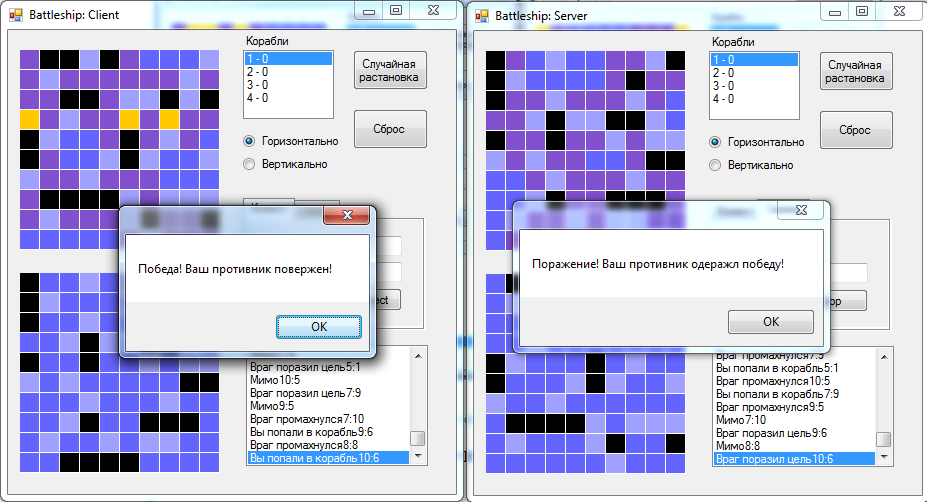


Рисунок - Игра окончена

Для отрисвоки полей используется следующий код:

private void pictureBox1\_Paint(object sender, PaintEventArgs e)

{

e.Graphics.Clear(Color.White);

c\_width = pictureBox1.Width / f\_width;

c\_height = pictureBox1.Height / f\_height;

for(int y = 0; y < f\_height; y++)

for(int x = 0; x < f\_width; x++)

{

switch (f\_data[x, y])

{

case 0: brush.Color = Color.FromArgb(100, 100, 255); break; // вода

case 1: brush.Color = Color.FromArgb(255, 200, 0); break; // корабль

case 2: brush.Color = Color.FromArgb(129, 80, 206); break; // рядом с кораблем

case 3: brush.Color = Color.FromArgb(0, 0, 0); break; // убил

case 4: brush.Color = Color.FromArgb(160, 160, 255); break; // промазал

}

e.Graphics.FillRectangle(brush, x \* c\_width, y \* c\_height, c\_width - 1, c\_height - 1);

}

}

Для расстановки кораблей по полю было написано несколько функций:

bool checkCell(int n, int[,] arr, int x, int y, bool chk)

{

if(n == 0)

{

for(int i = x-1; i < x+2; i++)

for (int j = y-1; j < y+2; j++)

{

if (i < 0 || i >= arr.GetLength(0) || j < 0 || j >= arr.GetLength(1)) continue;

if (arr[i, j] == 0) arr[i, j] = 2;

if (arr[i, j] == 1) return false;

}

}

else if(chk)

{

for (int i = x; i < x + 2; i++)

for (int j = y - 1; j < y + 2; j++)

{

if (i < 0 || i >= arr.GetLength(0) || j < 0 || j >= arr.GetLength(1)) continue;

if(arr[i, j] == 0) arr[i, j] = 2;

if (arr[i, j] == 1) return false;

}

}

else

{

for (int i = x - 1; i < x + 2; i++)

for (int j = y; j < y + 2; j++)

{

if (i < 0 || i >= arr.GetLength(0) || j < 0 || j >= arr.GetLength(1)) continue;

if (arr[i, j] == 0) arr[i, j] = 2;

if (arr[i, j] == 1) return false;

}

}

return true;

}

private void pictureBox1\_MouseDown(object sender, MouseEventArgs e)

{

if (listBox1.SelectedIndex < 0) return;

int c\_x = e.X / c\_width;

int c\_y = e.Y / c\_height;

instalShip(c\_x, c\_y, radioButton1.Checked);

if(ship.Sum() == 0) Ready();

pictureBox1.Invalidate();

}

bool instalShip(int c\_x,int c\_y,bool check)

{

int count = listBox1.SelectedIndex + 1;

if (ship[listBox1.SelectedIndex] < 1) return false;

int[,] f\_temp\_data = new int[f\_width, f\_height];

Array.Copy(f\_data, f\_temp\_data, f\_data.Length);

if (check)

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

{

if (c\_x + i >= f\_temp\_data.GetLength(0)) return false;

if (!checkCell(i, f\_temp\_data, c\_x + i, c\_y, check)) return false;

f\_temp\_data[c\_x + i, c\_y] = 1;

}

else

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

{

if (c\_y + i >= f\_temp\_data.GetLength(1)) return false;

if (!checkCell(i, f\_temp\_data, c\_x, c\_y + i, check)) return false;

f\_temp\_data[c\_x, c\_y + i] = 1;

}

Array.Copy(f\_temp\_data, f\_data, f\_temp\_data.Length);

ship[listBox1.SelectedIndex]--;

listBox1.Items.Clear();

for (int i = 0; i < 4; i++)

listBox1.Items.Add((i + 1) + " - " + ship[i]);

listBox1.SelectedIndex = count - 1;

return true;

}

Для реализации сетевого взаимодействия был написан класс Packet:

public class Packet

{

public int tp, X, Y;

public Packet()

{

tp = 0; X = 0; Y = 0;

}

public Packet(byte[] data)

{

int[]arr = new int[data.Length / sizeof(int)];

Buffer.BlockCopy(data, 0, arr, 0, arr.Length);

this.tp = arr[0];

this.X = arr[1];

this.Y = arr[2];

}

public Packet(int tp,int X, int Y)

{

this.tp = tp;

this.X = X;

this.Y = Y;

}

public byte[] ToBytes()

{

int[] arr = new int[3];

arr[0] = tp;

arr[1] = X;

arr[2] = Y;

byte[] result = new byte[arr.Length \* sizeof(int)];

Buffer.BlockCopy(arr, 0, result, 0, result.Length);

return result;

}

public static string GetIP4Address()

{

IPAddress[] ips = Dns.GetHostAddresses(Dns.GetHostName());

foreach (IPAddress i in ips)

{

if (i.AddressFamily == System.Net.Sockets.AddressFamily.InterNetwork)

return i.ToString();

}

return "127.0.0.1";

}

}

# Заключение

Благодаря проделанной работе, я закрепил на конкретных примерах теоретические знания, полученные при изучении свойств, методов и событий стандартных компонентов построения графического пользовательского интерфейса. Получил навыки по разработке клеточных игр с возможностью сетевого взаимодействия, а также навыки реализации игрового процесса и его отображения при помощи компонента pictureBox. Было разработано прикладное приложение «Морской бой» которое позволяет играть в игру морской бой на компьютере через локальную сеть.