# Содержание

[Содержание 0](#_Toc125403766)

[Цель курсовой работы 1](#_Toc125403767)

[Ход работы: 1](#_Toc125403768)

[Задачи проекта 1](#_Toc125403769)

[Интерфейс программы 1](#_Toc125403770)

[Формы 1](#_Toc125403771)

[Переход между формами 2](#_Toc125403772)

[Реализация букв 3](#_Toc125403773)

[Класс букв 3](#_Toc125403774)

[Мешок с фишками 3](#_Toc125403775)

[Выбор случайных букв 4](#_Toc125403776)

[Игровое поле 6](#_Toc125403777)

[Состав поля 6](#_Toc125403778)

[Перенос букв на поле 7](#_Toc125403779)

[Логика работы программы 10](#_Toc125403780)

[Правило 1 10](#_Toc125403781)

[Правило 2 13](#_Toc125403782)

[Правило 3 15](#_Toc125403783)

[Правило 4 и 5 17](#_Toc125403784)

[Правило 6 19](#_Toc125403785)

[Вывод 20](#_Toc125403786)

[Код программы 21](#_Toc125403787)

[Forms1 21](#_Toc125403788)

[Forms2 21](#_Toc125403789)

# Цель курсовой работы

Целью курсовой работы является создание обучаемой игры “Scrabble” в WindowsForms использую язык программирование C#. Параллельно написанию приложения я должен изучить работу с простейшими базами данных, улучшить свои знания в работе с WindowsForms и языком программирования C#.

# Ход работы:

## Задачи проекта

Для того чтобы быстро и эффективно выполнить какую-то большую задачу нужно раздробить её на кучу мелких подзадач.

Чтобы мне написать игру “Scrabble” на C# мне нужно реализовать следующие аспекты:

* Интерфейс программы;
* Реализация букв;
* Игровое поле;
* Логика работы программы.

## Интерфейс программы

### Формы

Интерфейс моей программы будет состоять из двух окон:

1. Окно правил
2. Игровое окно

Окно правил — это первое поле, которое появляется при запуске приложение. В нем кратко написаны правило игры и есть кнопка при нажатии на которую появляется игровое окно (рисунок 1).

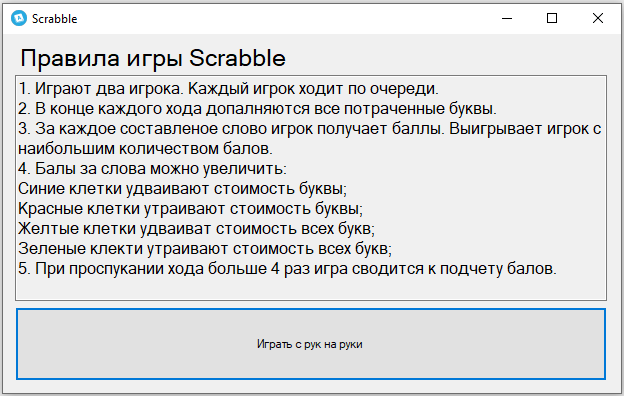


Рисунок Окно правил

Игровое окно представляет собой саму игру. В этом окне находиться игровое поле 15x15, счетчики, показывающий счет каждого игрока, показатели хода, 3 кнопки передачи хода, и 7 ячеек, предназначенных для просмотра набора случайных букв, и окно, позволяющая видеть все слова, которые присутствуют в игре (рисунок 2).

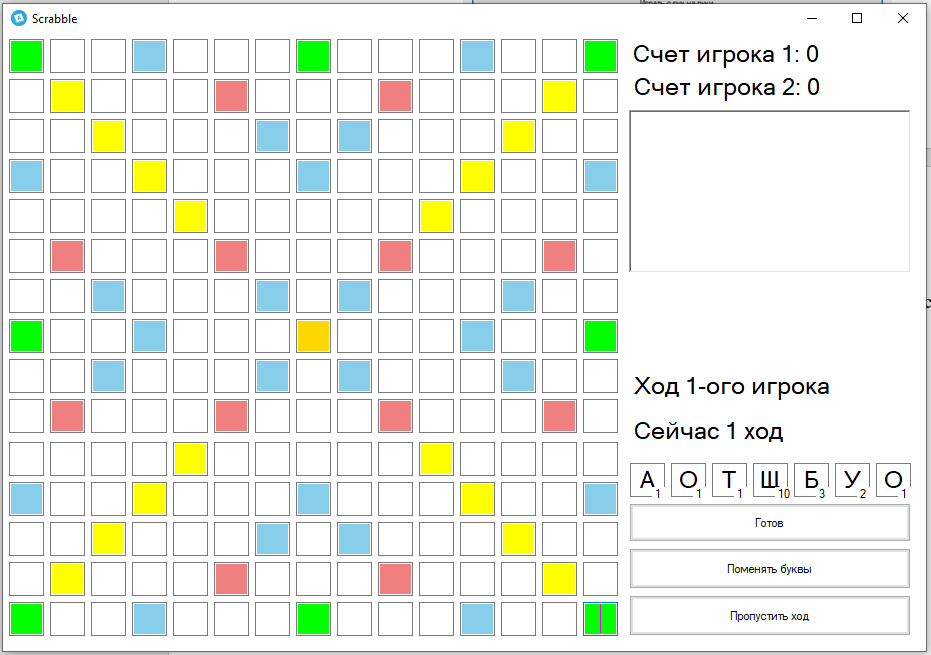


Рисунок Игровое окно

### Переход между формами

Переход из одного окна в другое реализован с помощью создание экземпляра второго окна в коде первого и при использовании команд позволяющие прятать окно (рисунок 3).

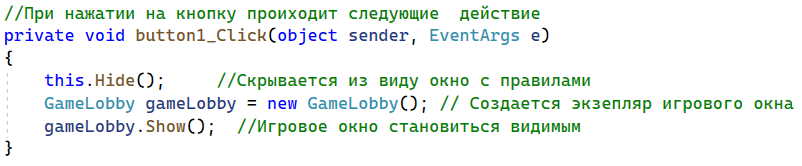


Рисунок Кнопка перехода

Переход из одного окна в другой также происходит когда игра подходит к концу (рисунок 4).

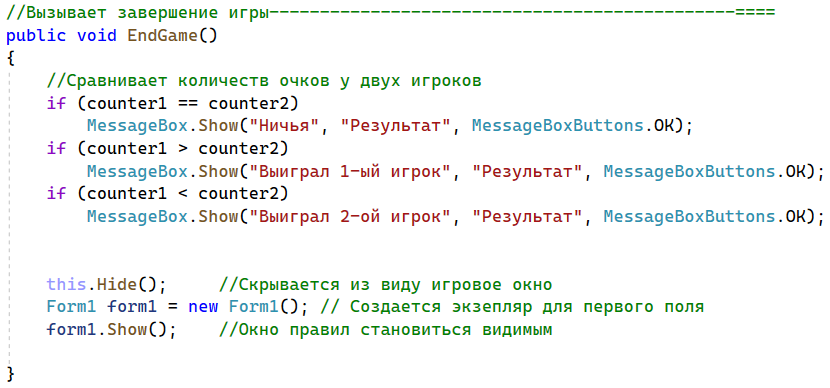


Рисунок метод перехода

## Реализация букв

### Класс букв

В оригинальной игре игроки засовывают руки в мешок и достают оттуда 7 случайных букв. Далее в игре если у какого-то игрока не хватает фишек до 7 то он снова сует руку в мешок и берет от туда не достающее количество букв.

Я создал отдельный класс для буквы (рисунок 5).

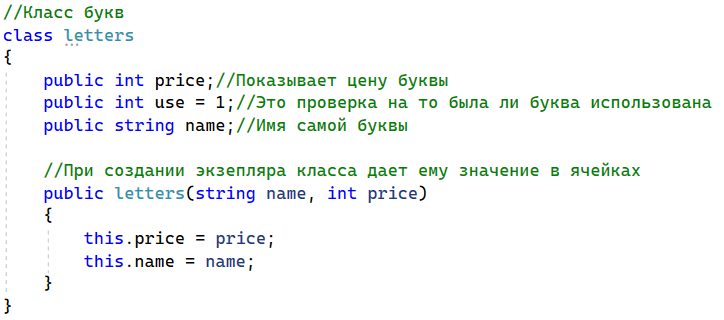


Рисунок Класс letters

### Мешок с фишками

После создание класса на его основании я сделал три массива. Первый массив представляется в виде мешочка со всеми фишками, а два других массива — это фишки игроков (рисунок 6 и 7).

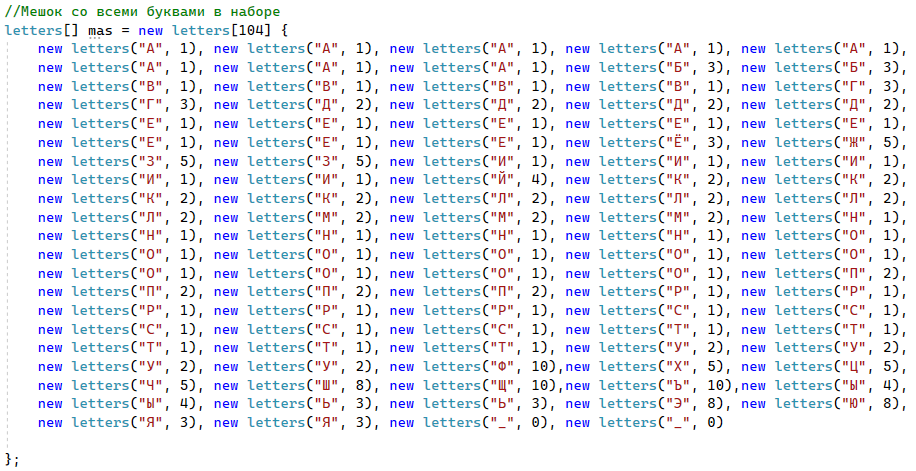


Рисунок Игровой мешок

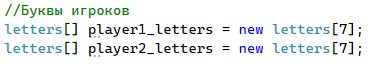


Рисунок Фишки игроков

### Выбор случайных букв

Буквы у игроков должны заполоняться случайным образом, для этого я сделал так что при запуске игрового окна через цикл for ищет случайные буквы для каждого игрока (рисунок 8 и 9).

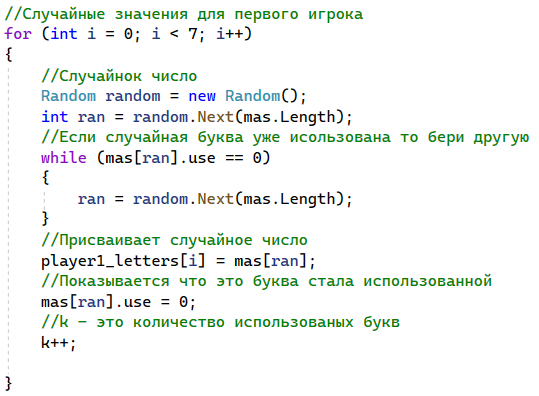


Рисунок Выборка случайных букв

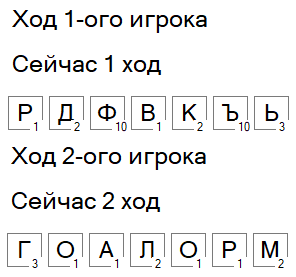


Рисунок Пример случайного выбора

Также после каждого хода идет проверка игровых ячеек, в которых отсутствуют буквы и туда также добавляют случайные буквы (рисунок 10 и 11).

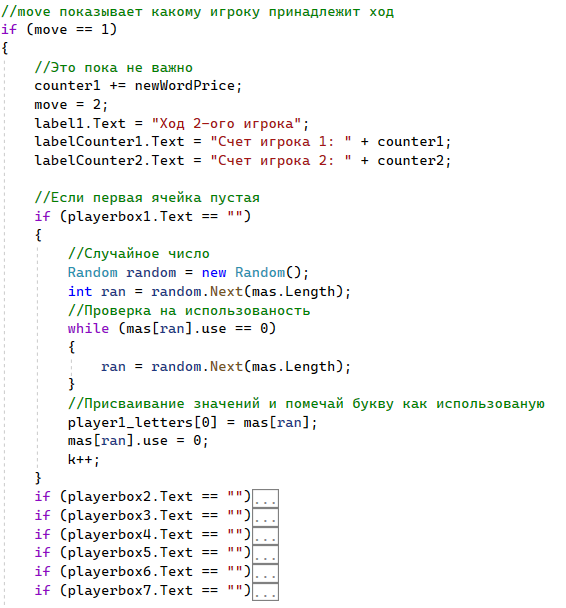


Рисунок Проверка на пустые ячейки

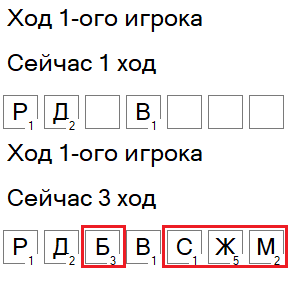


Рисунок Пример сдачи новых фишек

## Игровое поле

### Состав поля

Игровое поле игры “Scrabble” состоит из 225 ячеек, которые расположены в виде квадрата 15x15 (рисунок 12).

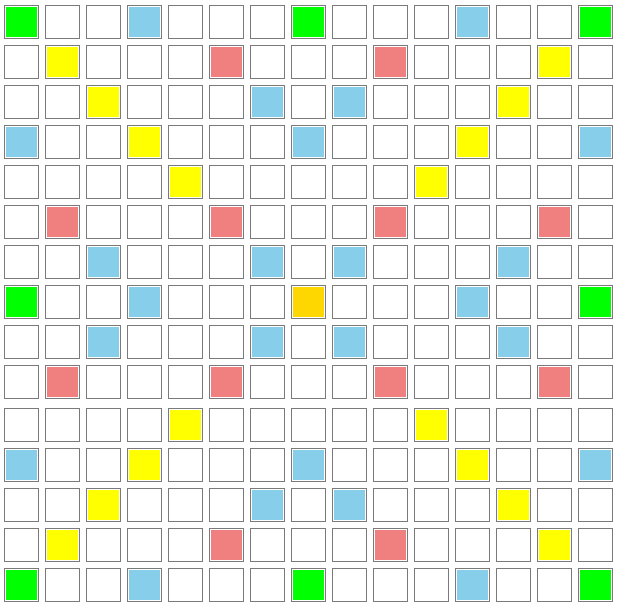


Рисунок Игровое поле

Я решил реализовать это поле через такой элемент как Textbox. Каждая ячейка моей игры является отдельным Textbox.

Также на игровом поле есть ячейки различных цветов. Цветов всего 5. У каждого цвета есть своя функция.

Золотой цвет, показывает, что при первом ходе слово обязательно должно пройти через него.

Синий и красный цвет, говорят, что при попадании на них цена буквы умножаются на 2 и 3 соответственно.

При попадании на желтый и зеленый умножаются цена всего слова на 2 и 3 соответственно.

Этот элемент поля я реализовал через метод WordPoints. В этой функции также идет проверка на то идет первое слово в золотой ячейке (рисунок 13 и 14).

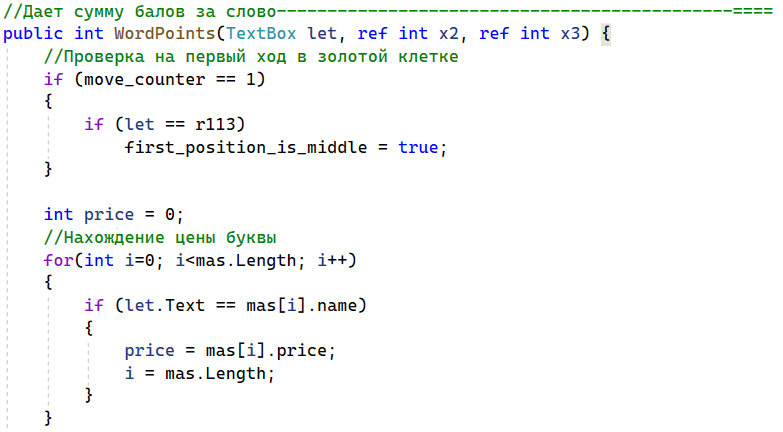


Рисунок WordFinder часть 1

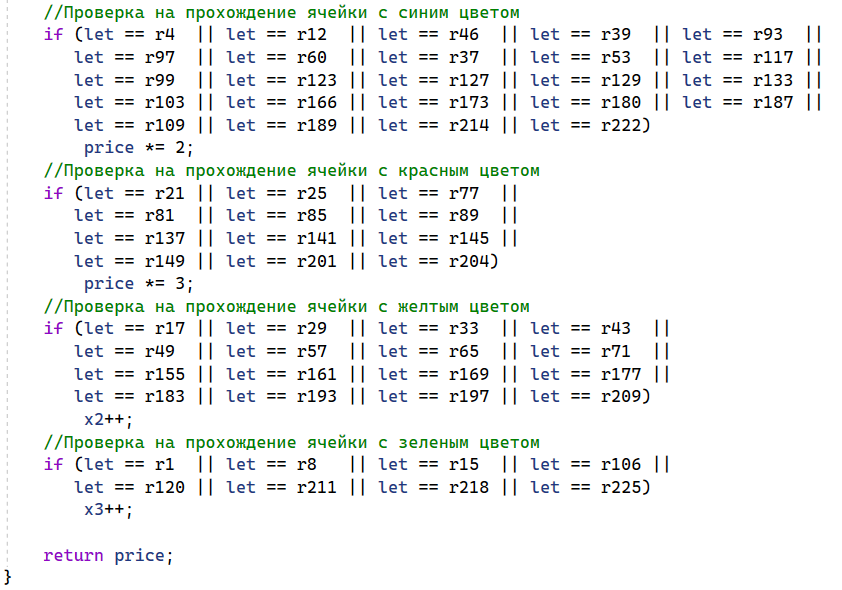


Рисунок WordFinder часть 2

### Перенос букв на поле

Для того чтобы можно было переносить буквы из “руки” на поле я создал переменную, которая временно содержит в себе букву, и написал функцию для того, чтобы переносить буквы из руки на поле (рисунок 15 и 16).

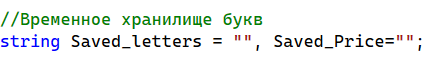


Рисунок Аргумент - временного хранилища

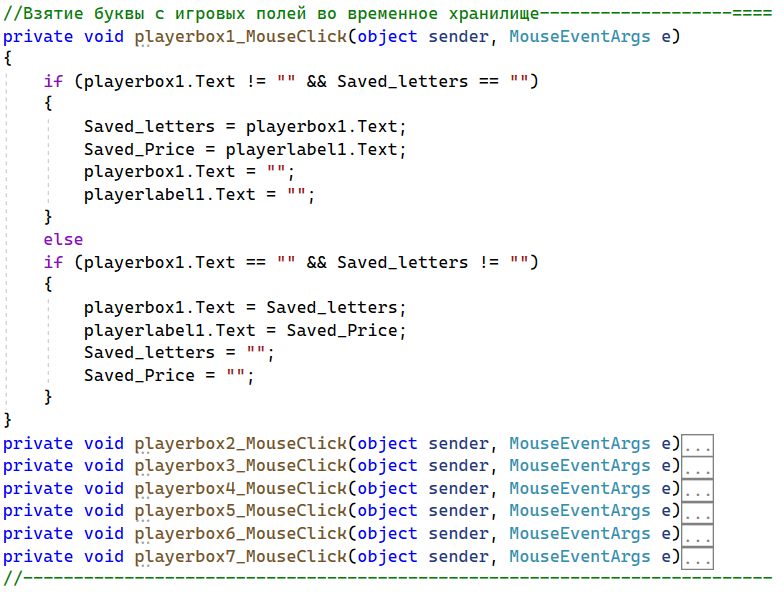


Рисунок Перенос буквы с временного хранилища на поле и на оборот

Переносить буквы из одной ячейки в другую можно только во время одного хода, чтобы это сделать я создал массив move\_counter\_letters (рисунок 17,18,19,20,21,22).



Рисунок Аргумент - проверки хода

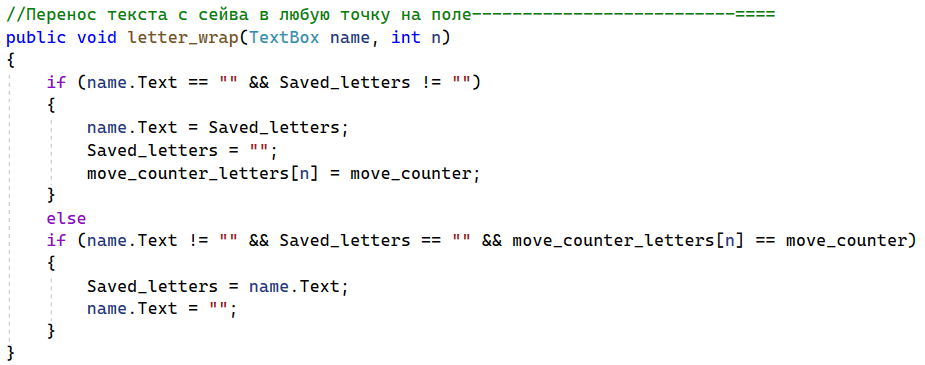


Рисунок Перенос букв из рук на поле



Рисунок Начальное позиция

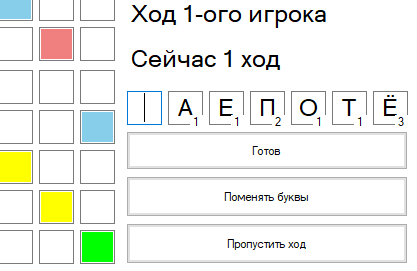


Рисунок Взятие буквы



Рисунок Подстановка буквы

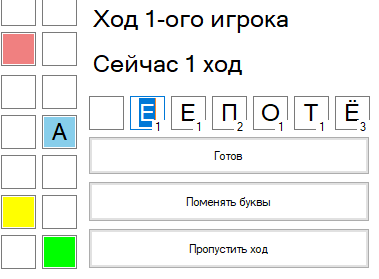


Рисунок Поставить букву снова в руку и поставить другую

## Логика работы программы

Логика программы — это работы игры по правилам игры “Scrabble”.

Правила Scrabble:

1. Слово, составленное на первом ходу, должно проходить через золотую клетку
2. Если слова не существует, то его нельзя использовать
3. Игрок может поменять все свои фишки, но при этом он передает ход
4. Игрок может пропускать ход
5. При пропускании хода больше 4 раз подряд игра заканчивается.
6. Выигрывает игрок с наибольшим количеством

### Правило 1

Первое правило я долго не мог реализовать, но потом я это сделал с лёгкостью. Для начало я создал массив textbox-ов (рисунок 23).

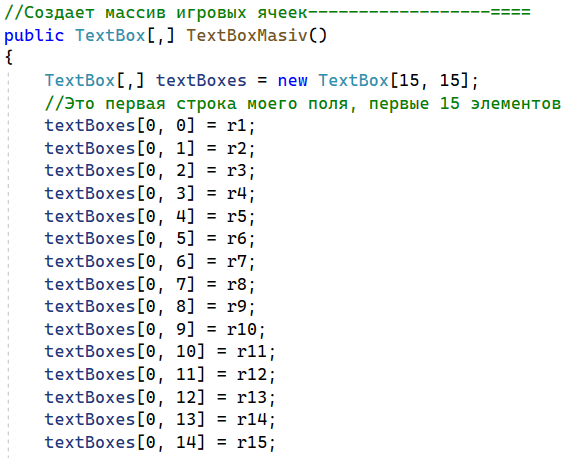


Рисунок Массив Textbox

После создание массива textbox-ов, я сделал метод поиска слов в игровом поле, WordFinder. Эту функция слишком большой получилось поэтому поделим ее на две условные части:

1. Поиск слова:

В поиске слова я прохожусь по двумерному массиву ища и соединяя отдельные буквы (рисунок 24).

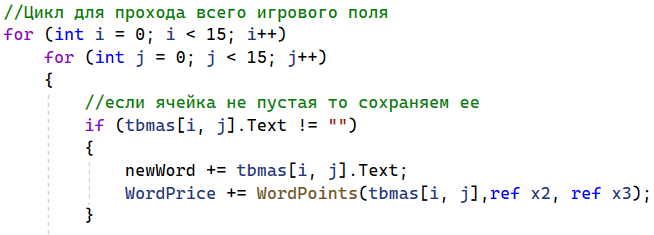


Рисунок Поиск слова

1. Проверка слова:

Проверка слова занимает большую часть этой программы. В этой части проверяется входит слово в словарь, входит ли первое слово золотую ячейку, не является ли слово одним символом (рисунок 25).

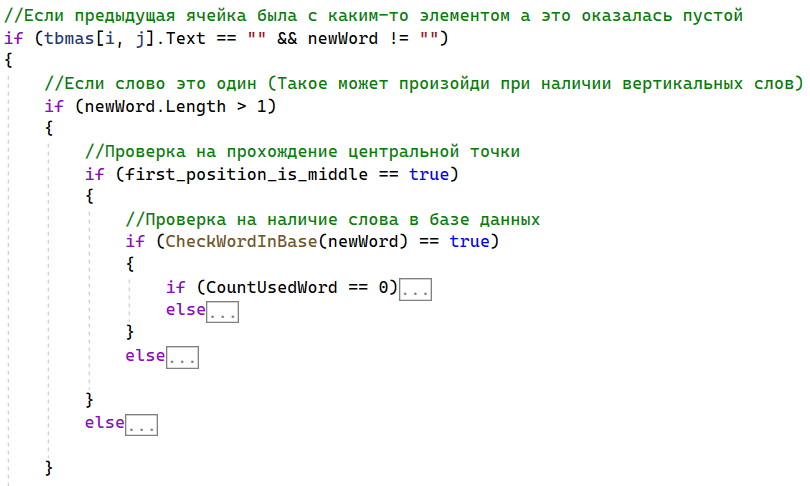


Рисунок Проверка слова

Весь метод WordFinder состоит из двух таких дублированных циклов (рисунок 26, 27, 28).

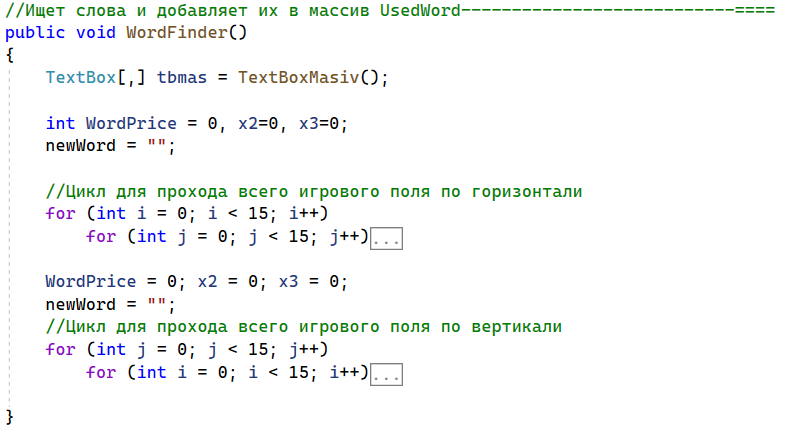


Рисунок WordFinder



Рисунок Слово написано не в центре

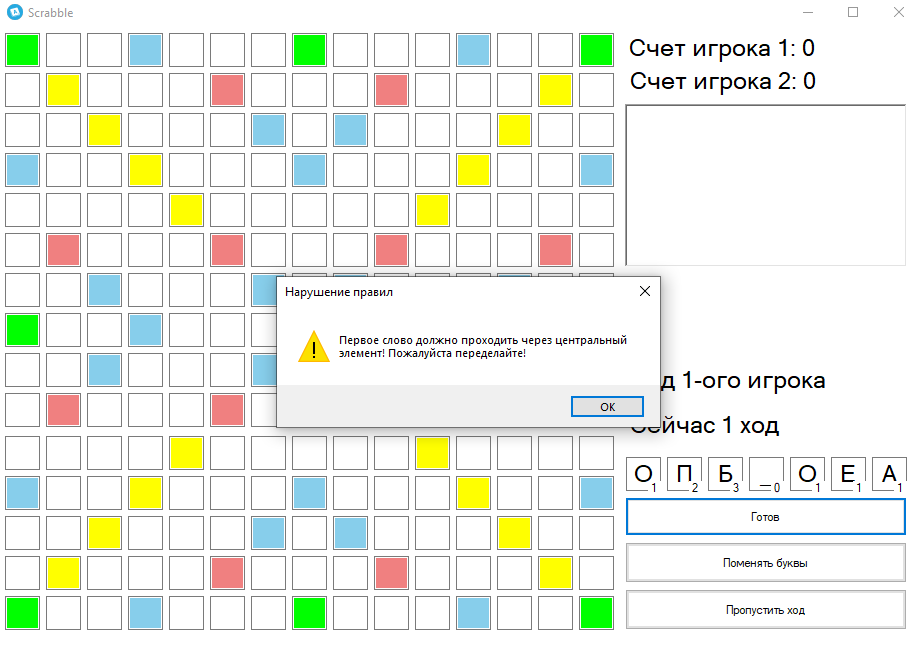


Рисунок Выдача ошибки

### Правило 2

Второе правило было реализовано через метод CheckWordInBase. Для работы этого метода сначала нужно создать текстовый файл, который будет нашей базой данных, хранящей все наши слова. После создание этого файла его надо наполнить словами, я нашел несколько сайтов в котором написаны большинство слов русского языка и загрузил их в нашу БД. После этого я ввожу переменную path, в котором написан путь к нашему файлу, а потом в самом методе открываю поток через stream reader и проверяю если наше слово в базе данных (рисунок 29 и 30).



Рисунок Путь к файлу

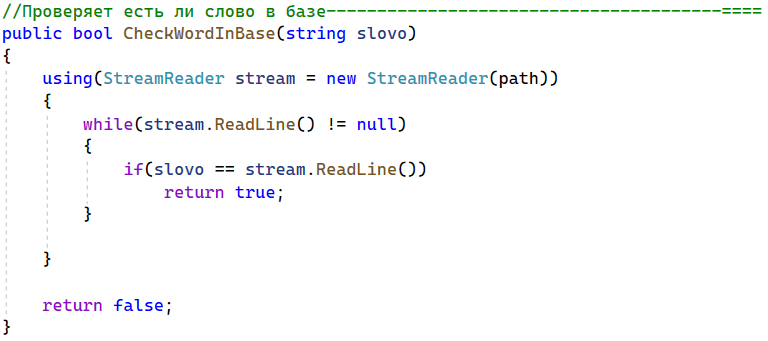


Рисунок Проверка на наличие в словаре

Благодаря этому методу и методу WordFinder, я смог заставить игру добавлять новые слова в свою базу данных (рисунок 31 и 32).



Рисунок Новое слово



Рисунок Вывод просьбы исправить ошибку

### Правило 3

Для реализации смены всех фишек у одного игрока я создал в форме дополнительную кнопку и добавил туда тот же цикл что и при запуске игрового поля (рисунок 33, 34, 35, 36).

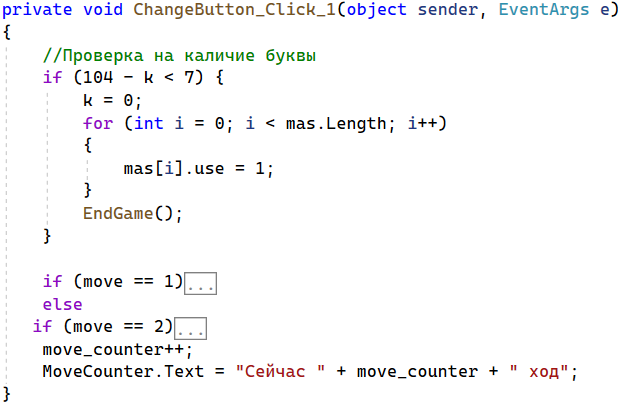


Рисунок Смена игровых фишек

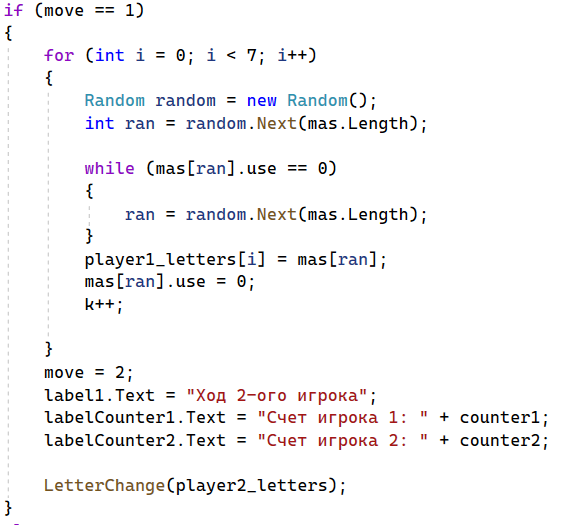


Рисунок Смена игровых фишек



Рисунок Пример



Рисунок Пример

### Правило 4 и 5

Четвертое правило реализована также как и третье.

Пятое реализовано через условие в начале нажатии на кнопку при помощи специальной переменой skip\_move (рисунок 37, 38, 39).

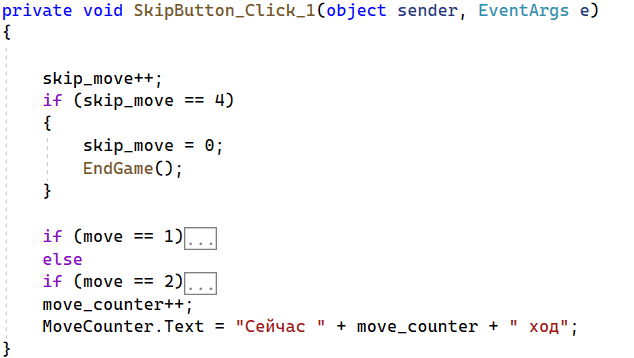


Рисунок метод SkipButton



Рисунок Пример работы 1



Рисунок Пример работы 2

### Правило 6

Чтобы сделать шестое правило я написал два метода это EndGame, который вызывает конец игры, и WordPoints, который считает количество очков в слове (рисунок 40, 41, 42).

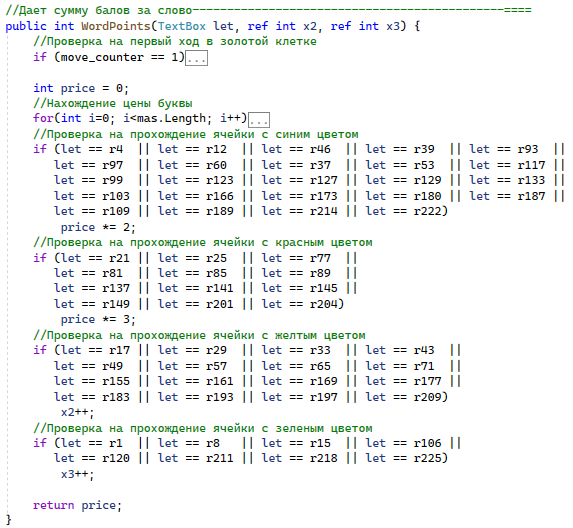


Рисунок WordPoints

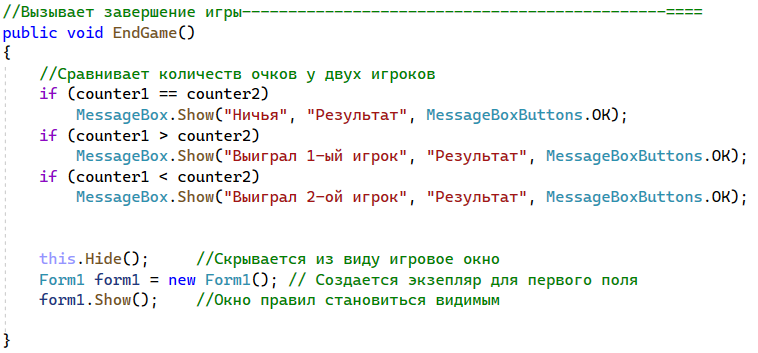


Рисунок EndGame



Рисунок Пример выигрыша

# Вывод

В ходе работы над курсовой работы я узнал новые стороны языки программирования С#, изучил работу WindowsForms, а также научился лучше создавать приложение в MS Visual Studio. Результатом выполнения курсовой работы стала самообучаема игра “Scrabble”.

# Код программы

## Forms1

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Scrablle

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

}

//При нажатии на кнопку проиходит следующие действие

private void button1\_Click(object sender, EventArgs e)

{

this.Hide(); //Скрывается из виду окно с правилами

GameLobby gameLobby = new GameLobby(); // Создается экзепляр игрового окна

gameLobby.Show(); //Игровое окно становиться видимым

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

}

}

## Forms2

using System;

using System.IO;

using System.Collections.Generic;

using System.ComponentModel;

using System.ComponentModel.Design;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Runtime.InteropServices;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Scrablle

{

public partial class GameLobby : Form

{

//Класс букв

class letters

{

public int price;//Показывает цену буквы

public int use = 1;//Это проверка на то была ли буква использована

public string name;//Имя самой буквы

//При создании экзепляра класса дает ему значение в ячейках

public letters(string name, int price)

{

this.price = price;

this.name = name;

}

}

//Мешок со всеми буквами в наборе

letters[] mas = new letters[104] {

new letters("А", 1), new letters("А", 1), new letters("А", 1), new letters("А", 1), new letters("А", 1),

new letters("А", 1), new letters("А", 1), new letters("А", 1), new letters("Б", 3), new letters("Б", 3),

new letters("В", 1), new letters("В", 1), new letters("В", 1), new letters("В", 1), new letters("Г", 3),

new letters("Г", 3), new letters("Д", 2), new letters("Д", 2), new letters("Д", 2), new letters("Д", 2),

new letters("Е", 1), new letters("Е", 1), new letters("Е", 1), new letters("Е", 1), new letters("Е", 1),

new letters("Е", 1), new letters("Е", 1), new letters("Е", 1), new letters("Ё", 3), new letters("Ж", 5),

new letters("З", 5), new letters("З", 5), new letters("И", 1), new letters("И", 1), new letters("И", 1),

new letters("И", 1), new letters("И", 1), new letters("Й", 4), new letters("К", 2), new letters("К", 2),

new letters("К", 2), new letters("К", 2), new letters("Л", 2), new letters("Л", 2), new letters("Л", 2),

new letters("Л", 2), new letters("М", 2), new letters("М", 2), new letters("М", 2), new letters("Н", 1),

new letters("Н", 1), new letters("Н", 1), new letters("Н", 1), new letters("Н", 1), new letters("О", 1),

new letters("О", 1), new letters("О", 1), new letters("О", 1), new letters("О", 1), new letters("О", 1),

new letters("О", 1), new letters("О", 1), new letters("О", 1), new letters("О", 1), new letters("П", 2),

new letters("П", 2), new letters("П", 2), new letters("П", 2), new letters("Р", 1), new letters("Р", 1),

new letters("Р", 1), new letters("Р", 1), new letters("Р", 1), new letters("С", 1), new letters("С", 1),

new letters("С", 1), new letters("С", 1), new letters("С", 1), new letters("Т", 1), new letters("Т", 1),

new letters("Т", 1), new letters("Т", 1), new letters("Т", 1), new letters("У", 2), new letters("У", 2),

new letters("У", 2), new letters("У", 2), new letters("Ф", 10),new letters("Х", 5), new letters("Ц", 5),

new letters("Ч", 5), new letters("Ш", 8), new letters("Щ", 10),new letters("Ъ", 10),new letters("Ы", 4),

new letters("Ы", 4), new letters("Ь", 3), new letters("Ь", 3), new letters("Э", 8), new letters("Ю", 8),

new letters("Я", 3), new letters("Я", 3), new letters("\_", 0), new letters("\_", 0)

};

//Счетскик пропусков

int skip\_move = 0;

//Счетчик использованных букв

int k = 0;

//Переход хода

int move = 1;

//Счетчик очков

int counter1 = 0, counter2 = 0;

//Буквы игроков

letters[] player1\_letters = new letters[7];

letters[] player2\_letters = new letters[7];

//Счетчик ходов

int move\_counter = 0;

bool first\_position\_is\_middle=false;

//Число возможных слов за игру

static int N = 100;

string[] UsedWord = new string[N];

int CountUsedWord = 0;

//Имя файла строки

string path = @"C:\Users\Admin\source\repos\Scrablle\resources\vocabulary.txt";

//Пауза хода

bool stop = false;

//Переменные для поиска новых слов

public string newWord = "";

public int newWordPrice = 0;

public GameLobby()

{

InitializeComponent();

//Случайные значения для первого игрока

for (int i = 0; i < 7; i++)

{

//Случайнок число

Random random = new Random();

int ran = random.Next(mas.Length);

//Если случайная буква уже исользована то бери другую

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

//Присваивает случайное число

player1\_letters[i] = mas[ran];

//Показывается что это буква стала использованной

mas[ran].use = 0;

//k - это количество использованых букв

k++;

}

//Случайные значения для второго игрока

for (int i = 0; i < 7; i++)

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player2\_letters[i] = mas[ran];

mas[ran].use = 0;

k++;

}

playerbox1.Text = player1\_letters[0].name;

playerbox2.Text = player1\_letters[1].name;

playerbox3.Text = player1\_letters[2].name;

playerbox4.Text = player1\_letters[3].name;

playerbox5.Text = player1\_letters[4].name;

playerbox6.Text = player1\_letters[5].name;

playerbox7.Text = player1\_letters[6].name;

playerlabel1.Text = player1\_letters[0].price.ToString();

playerlabel2.Text = player1\_letters[1].price.ToString();

playerlabel3.Text = player1\_letters[2].price.ToString();

playerlabel4.Text = player1\_letters[3].price.ToString();

playerlabel5.Text = player1\_letters[4].price.ToString();

playerlabel6.Text = player1\_letters[5].price.ToString();

playerlabel7.Text = player1\_letters[6].price.ToString();

move\_counter++;

MoveCounter.Text = "Сейчас " + move\_counter + " ход";

}

private void StartButton\_Click\_1(object sender, EventArgs e)

{

WordFinder();

if (stop == false)

{

string slovo = "";

for (int i = 1; i <= CountUsedWord; i++)

slovo += i + "-ое слово " + UsedWord[i - 1] + "\r\n";

richTextBox1.Text = slovo;

//move показывает какому игроку принадлежит ход

if (move == 1)

{

//Это пока не важно

counter1 += newWordPrice;

move = 2;

label1.Text = "Ход 2-ого игрока";

labelCounter1.Text = "Счет игрока 1: " + counter1;

labelCounter2.Text = "Счет игрока 2: " + counter2;

//Если первая ячейка пустая

if (playerbox1.Text == "")

{

//Случайное число

Random random = new Random();

int ran = random.Next(mas.Length);

//Проверка на использованость

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

//Присваивание значений и помечай букву как использованую

player1\_letters[0] = mas[ran];

mas[ran].use = 0;

k++;

}

if (playerbox2.Text == "")

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player1\_letters[1] = mas[ran];

mas[ran].use = 0;

k++;

}

if (playerbox3.Text == "")

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player1\_letters[2] = mas[ran];

mas[ran].use = 0;

k++;

}

if (playerbox4.Text == "")

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player1\_letters[3] = mas[ran];

mas[ran].use = 0;

k++;

}

if (playerbox5.Text == "")

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player1\_letters[4] = mas[ran];

mas[ran].use = 0;

k++;

}

if (playerbox6.Text == "")

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player1\_letters[5] = mas[ran];

mas[ran].use = 0;

k++;

}

if (playerbox7.Text == "")

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player1\_letters[6] = mas[ran];

mas[ran].use = 0;

k++;

}

LetterChange(player2\_letters);

}

else

if (move == 2)

{

counter2 += newWordPrice;

move = 1;

label1.Text = "Ход 1-ого игрока";

labelCounter1.Text = "Счет игрока 1: " + counter1;

labelCounter2.Text = "Счет игрока 2: " + counter2;

if (playerbox1.Text == "")

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player2\_letters[0] = mas[ran];

mas[ran].use = 0;

k++;

}

if (playerbox2.Text == "")

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player2\_letters[1] = mas[ran];

mas[ran].use = 0;

k++;

}

if (playerbox3.Text == "")

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player2\_letters[2] = mas[ran];

mas[ran].use = 0;

k++;

}

if (playerbox4.Text == "")

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player2\_letters[3] = mas[ran];

mas[ran].use = 0;

k++;

}

if (playerbox5.Text == "")

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player2\_letters[4] = mas[ran];

mas[ran].use = 0;

k++;

}

if (playerbox6.Text == "")

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player2\_letters[5] = mas[ran];

mas[ran].use = 0;

k++;

}

if (playerbox7.Text == "")

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player2\_letters[6] = mas[ran];

mas[ran].use = 0;

k++;

}

LetterChange(player1\_letters);

}

skip\_move = 0;

move\_counter++;

MoveCounter.Text = "Сейчас " + move\_counter + " ход";

}

stop = false;

}

private void ChangeButton\_Click\_1(object sender, EventArgs e)

{

//Проверка на каличие буквы

if (104 - k < 7) {

k = 0;

for (int i = 0; i < mas.Length; i++)

{

mas[i].use = 1;

}

EndGame();

}

if (move == 1)

{

for (int i = 0; i < 7; i++)

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player1\_letters[i] = mas[ran];

mas[ran].use = 0;

k++;

}

move = 2;

label1.Text = "Ход 2-ого игрока";

labelCounter1.Text = "Счет игрока 1: " + counter1;

labelCounter2.Text = "Счет игрока 2: " + counter2;

LetterChange(player2\_letters);

}

else

if (move == 2)

{

for (int i = 0; i < 7; i++)

{

Random random = new Random();

int ran = random.Next(mas.Length);

while (mas[ran].use == 0)

{

ran = random.Next(mas.Length);

}

player2\_letters[i] = mas[ran];

mas[ran].use = 0;

k++;

}

move = 1;

label1.Text = "Ход 1-ого игрока";

labelCounter1.Text = "Счет игрока 1: " + counter1;

labelCounter2.Text = "Счет игрока 2: " + counter2;

LetterChange(player1\_letters);

}

move\_counter++;

MoveCounter.Text = "Сейчас " + move\_counter + " ход";

}

private void SkipButton\_Click\_1(object sender, EventArgs e)

{

skip\_move++;

if (skip\_move == 4)

{

skip\_move = 0;

EndGame();

}

if (move == 1)

{

move = 2;

label1.Text = "Ход 2-ого игрока";

labelCounter1.Text = "Счет игрока 1: " + counter1;

labelCounter2.Text = "Счет игрока 2: " + counter2;

LetterChange(player2\_letters);

}

else

if (move == 2)

{

move = 1;

label1.Text = "Ход 1-ого игрока";

labelCounter1.Text = "Счет игрока 1: " + counter1;

labelCounter2.Text = "Счет игрока 2: " + counter2;

LetterChange(player1\_letters);

}

move\_counter++;

MoveCounter.Text = "Сейчас " + move\_counter + " ход";

}

//Смена игровых карт на руке игрока-------------------------------------====

private void LetterChange(letters[] player)

{

playerbox1.Text = player[0].name;

playerbox2.Text = player[1].name;

playerbox3.Text = player[2].name;

playerbox4.Text = player[3].name;

playerbox5.Text = player[4].name;

playerbox6.Text = player[5].name;

playerbox7.Text = player[6].name;

playerlabel1.Text = player[0].price.ToString();

playerlabel2.Text = player[1].price.ToString();

playerlabel3.Text = player[2].price.ToString();

playerlabel4.Text = player[3].price.ToString();

playerlabel5.Text = player[4].price.ToString();

playerlabel6.Text = player[5].price.ToString();

playerlabel7.Text = player[6].price.ToString();

}

//--------------------------------------------------------------------------

//Создает массив игровых ячеек------------------====

public TextBox[,] TextBoxMasiv()

{

TextBox[,] textBoxes = new TextBox[15, 15];

//Это первая строка моего поля, первые 15 элементов

textBoxes[0, 0] = r1;

textBoxes[0, 1] = r2;

textBoxes[0, 2] = r3;

textBoxes[0, 3] = r4;

textBoxes[0, 4] = r5;

textBoxes[0, 5] = r6;

textBoxes[0, 6] = r7;

textBoxes[0, 7] = r8;

textBoxes[0, 8] = r9;

textBoxes[0, 9] = r10;

textBoxes[0, 10] = r11;

textBoxes[0, 11] = r12;

textBoxes[0, 12] = r13;

textBoxes[0, 13] = r14;

textBoxes[0, 14] = r15;

textBoxes[1, 0] = r16;

textBoxes[1, 1] = r17;

textBoxes[1, 2] = r18;

textBoxes[1, 3] = r19;

textBoxes[1, 4] = r20;

textBoxes[1, 5] = r21;

textBoxes[1, 6] = r22;

textBoxes[1, 7] = r23;

textBoxes[1, 8] = r24;

textBoxes[1, 9] = r25;

textBoxes[1, 10] = r26;

textBoxes[1, 11] = r27;

textBoxes[1, 12] = r28;

textBoxes[1, 13] = r29;

textBoxes[1, 14] = r30;

textBoxes[2, 0] = r31;

textBoxes[2, 1] = r32;

textBoxes[2, 2] = r33;

textBoxes[2, 3] = r34;

textBoxes[2, 4] = r35;

textBoxes[2, 5] = r36;

textBoxes[2, 6] = r37;

textBoxes[2, 7] = r38;

textBoxes[2, 8] = r39;

textBoxes[2, 9] = r40;

textBoxes[2, 10] = r41;

textBoxes[2, 11] = r42;

textBoxes[2, 12] = r43;

textBoxes[2, 13] = r44;

textBoxes[2, 14] = r45;

textBoxes[3, 0] = r46;

textBoxes[3, 1] = r47;

textBoxes[3, 2] = r48;

textBoxes[3, 3] = r49;

textBoxes[3, 4] = r50;

textBoxes[3, 5] = r51;

textBoxes[3, 6] = r52;

textBoxes[3, 7] = r53;

textBoxes[3, 8] = r54;

textBoxes[3, 9] = r55;

textBoxes[3, 10] = r56;

textBoxes[3, 11] = r57;

textBoxes[3, 12] = r58;

textBoxes[3, 13] = r59;

textBoxes[3, 14] = r60;

textBoxes[4, 0] = r61;

textBoxes[4, 1] = r62;

textBoxes[4, 2] = r63;

textBoxes[4, 3] = r64;

textBoxes[4, 4] = r65;

textBoxes[4, 5] = r66;

textBoxes[4, 6] = r67;

textBoxes[4, 7] = r68;

textBoxes[4, 8] = r69;

textBoxes[4, 9] = r70;

textBoxes[4, 10] = r71;

textBoxes[4, 11] = r72;

textBoxes[4, 12] = r73;

textBoxes[4, 13] = r;

textBoxes[4, 14] = r75;

textBoxes[5, 0] = r76;

textBoxes[5, 1] = r77;

textBoxes[5, 2] = r78;

textBoxes[5, 3] = r79;

textBoxes[5, 4] = r80;

textBoxes[5, 5] = r81;

textBoxes[5, 6] = r82;

textBoxes[5, 7] = r83;

textBoxes[5, 8] = r84;

textBoxes[5, 9] = r85;

textBoxes[5, 10] = r86;

textBoxes[5, 11] = r87;

textBoxes[5, 12] = r88;

textBoxes[5, 13] = r89;

textBoxes[5, 14] = r90;

textBoxes[6, 0] = r91;

textBoxes[6, 1] = r92;

textBoxes[6, 2] = r93;

textBoxes[6, 3] = r94;

textBoxes[6, 4] = r95;

textBoxes[6, 5] = r96;

textBoxes[6, 6] = r97;

textBoxes[6, 7] = r98;

textBoxes[6, 8] = r99;

textBoxes[6, 9] = r100;

textBoxes[6, 10] = r101;

textBoxes[6, 11] = r102;

textBoxes[6, 12] = r103;

textBoxes[6, 13] = r104;

textBoxes[6, 14] = r105;

textBoxes[7, 0] = r106;

textBoxes[7, 1] = r107;

textBoxes[7, 2] = r108;

textBoxes[7, 3] = r109;

textBoxes[7, 4] = r110;

textBoxes[7, 5] = r111;

textBoxes[7, 6] = r112;

textBoxes[7, 7] = r113;

textBoxes[7, 8] = r114;

textBoxes[7, 9] = r115;

textBoxes[7, 10] = r116;

textBoxes[7, 11] = r117;

textBoxes[7, 12] = r118;

textBoxes[7, 13] = r119;

textBoxes[7, 14] = r120;

textBoxes[8, 0] = r121;

textBoxes[8, 1] = r122;

textBoxes[8, 2] = r123;

textBoxes[8, 3] = r124;

textBoxes[8, 4] = r125;

textBoxes[8, 5] = r126;

textBoxes[8, 6] = r127;

textBoxes[8, 7] = r128;

textBoxes[8, 8] = r129;

textBoxes[8, 9] = r130;

textBoxes[8, 10] = r131;

textBoxes[8, 11] = r132;

textBoxes[8, 12] = r133;

textBoxes[8, 13] = r134;

textBoxes[8, 14] = r135;

textBoxes[9, 0] = r136;

textBoxes[9, 1] = r137;

textBoxes[9, 2] = r138;

textBoxes[9, 3] = r139;

textBoxes[9, 4] = r140;

textBoxes[9, 5] = r141;

textBoxes[9, 6] = r142;

textBoxes[9, 7] = r143;

textBoxes[9, 8] = r144;

textBoxes[9, 9] = r145;

textBoxes[9, 10] = r146;

textBoxes[9, 11] = r147;

textBoxes[9, 12] = r148;

textBoxes[9, 13] = r149;

textBoxes[9, 14] = r150;

textBoxes[10, 0] = r151;

textBoxes[10, 1] = r152;

textBoxes[10, 2] = r153;

textBoxes[10, 3] = r154;

textBoxes[10, 4] = r155;

textBoxes[10, 5] = r156;

textBoxes[10, 6] = r157;

textBoxes[10, 7] = r158;

textBoxes[10, 8] = r159;

textBoxes[10, 9] = r160;

textBoxes[10, 10] = r161;

textBoxes[10, 11] = r162;

textBoxes[10, 12] = r163;

textBoxes[10, 13] = r164;

textBoxes[10, 14] = r165;

textBoxes[11, 0] = r166;

textBoxes[11, 1] = r167;

textBoxes[11, 2] = r168;

textBoxes[11, 3] = r169;

textBoxes[11, 4] = r170;

textBoxes[11, 5] = r171;

textBoxes[11, 6] = r172;

textBoxes[11, 7] = r173;

textBoxes[11, 8] = r174;

textBoxes[11, 9] = r175;

textBoxes[11, 10] = r176;

textBoxes[11, 11] = r177;

textBoxes[11, 12] = r178;

textBoxes[11, 13] = r179;

textBoxes[11, 14] = r180;

textBoxes[12, 0] = r181;

textBoxes[12, 1] = r182;

textBoxes[12, 2] = r183;

textBoxes[12, 3] = r184;

textBoxes[12, 4] = r185;

textBoxes[12, 5] = r186;

textBoxes[12, 6] = r187;

textBoxes[12, 7] = r188;

textBoxes[12, 8] = r189;

textBoxes[12, 9] = r190;

textBoxes[12, 10] = r191;

textBoxes[12, 11] = r192;

textBoxes[12, 12] = r193;

textBoxes[12, 13] = r194;

textBoxes[12, 14] = r195;

textBoxes[13, 0] = r196;

textBoxes[13, 1] = r197;

textBoxes[13, 2] = r198;

textBoxes[13, 3] = r199;

textBoxes[13, 4] = r200;

textBoxes[13, 5] = r201;

textBoxes[13, 6] = r202;

textBoxes[13, 7] = r203;

textBoxes[13, 8] = r204;

textBoxes[13, 9] = r205;

textBoxes[13, 10] = r206;

textBoxes[13, 11] = r207;

textBoxes[13, 12] = r208;

textBoxes[13, 13] = r209;

textBoxes[13, 14] = r210;

textBoxes[14, 0] = r211;

textBoxes[14, 1] = r212;

textBoxes[14, 2] = r213;

textBoxes[14, 3] = r214;

textBoxes[14, 4] = r215;

textBoxes[14, 5] = r216;

textBoxes[14, 6] = r217;

textBoxes[14, 7] = r218;

textBoxes[14, 8] = r219;

textBoxes[14, 9] = r220;

textBoxes[14, 10] = r221;

textBoxes[14, 11] = r222;

textBoxes[14, 12] = r223;

textBoxes[14, 13] = r224;

textBoxes[14, 14] = r225;

return textBoxes;

}

//------------------------------------------------

//Ищет слова и добавляет их в массив UsedWord---------------------------====

public void WordFinder()

{

TextBox[,] tbmas = TextBoxMasiv();

int WordPrice = 0, x2=0, x3=0;

newWord = "";

//Цикл для прохода всего игрового поля по горизонтали

for (int i = 0; i < 15; i++)

for (int j = 0; j < 15; j++)

{

//если ячейка не пустая то сохраняем ее

if (tbmas[i, j].Text != "")

{

newWord += tbmas[i, j].Text;

WordPrice += WordPoints(tbmas[i, j],ref x2, ref x3);

}

//Если предыдущая ячейка была с каким-то элементом а это оказалась пустой

if (tbmas[i, j].Text == "" && newWord != "")

{

//Если слово это один (Такое может произойди при наличии вертикальных слов)

if (newWord.Length > 1)

{

//Проверка на прохождение центральной точки

if (first\_position\_is\_middle == true)

{

//Проверка на наличие слова в базе данных

if (CheckWordInBase(newWord) == true)

{

if (CountUsedWord == 0)

{

UsedWord[CountUsedWord] = newWord;

CountUsedWord++;

if (x2 > 0)

WordPrice \*= 2;

if (x3 > 0)

WordPrice \*= 3;

newWordPrice = WordPrice;

x2 = 0;

x3 = 0;

WordPrice = 0;

}

else

{

int CheckNewWord = 0;

for (int a = 0; a < CountUsedWord; a++)

{

if (UsedWord[a] == newWord)

CheckNewWord++;

}

if (CheckNewWord == 0)

{

UsedWord[CountUsedWord] = newWord;

CountUsedWord++;

if (x2 > 0)

WordPrice \*= 2;

if (x3 > 0)

WordPrice \*= 3;

newWordPrice = WordPrice;

x2 = 0;

x3 = 0;

WordPrice = 0;

}

}

}

else

{

DialogResult result = MessageBox.Show($"Такого слово как {newWord} нет, но если вы уверены что оно есть то, хотите ли добавить это слово в список сущестующих слов?", "Ошибка в слове!", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (result == DialogResult.Yes)

{

using (StreamWriter stream = new StreamWriter(path, true))

stream.WriteLine(newWord);

if (CheckWordInBase(newWord) == true)

{

if (CountUsedWord == 0)

{

UsedWord[CountUsedWord] = newWord;

CountUsedWord++;

if (x2 > 0)

WordPrice \*= 2;

if (x3 > 0)

WordPrice \*= 3;

newWordPrice = WordPrice;

x2 = 0;

x3 = 0;

WordPrice = 0;

stop = false;

}

else

{

int CheckNewWord = 0;

for (int a = 0; a < CountUsedWord; a++)

{

if (UsedWord[a] == newWord)

CheckNewWord++;

}

if (CheckNewWord == 0)

{

UsedWord[CountUsedWord] = newWord;

CountUsedWord++;

if (x2 > 0)

WordPrice \*= 2;

if (x3 > 0)

WordPrice \*= 3;

newWordPrice = WordPrice;

x2 = 0;

x3 = 0;

WordPrice = 0;

}

}

}

}

else if (result == DialogResult.No)

{

ClearArea();

if (move == 1)

LetterChange(player1\_letters);

if (move == 2)

LetterChange(player2\_letters);

stop = true;

}

}

}

else

{

LetterChange(player1\_letters);

for (int s = 0; s < CountUsedWord; s++)

{

UsedWord[s] = "";

}

CountUsedWord = 0;

newWord = "";

newWordPrice = 0;

ClearArea();

stop=true;

MessageBox.Show("Первое слово должно проходить через центральный элемент! Пожалуйста переделайте!", "Нарушение правил", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

newWord = "";

x2 = 0;

x3 = 0;

WordPrice = 0;

}

}

WordPrice = 0; x2 = 0; x3 = 0;

newWord = "";

//Цикл для прохода всего игрового поля по вертикали

for (int j = 0; j < 15; j++)

for (int i = 0; i < 15; i++)

{

if (tbmas[i, j].Text != "")

{

newWord += tbmas[i, j].Text;

WordPrice += WordPoints(tbmas[i, j], ref x2, ref x3);

}

if (tbmas[i, j].Text == "" && newWord != "")

{

if (newWord.Length > 1)

{

if (first\_position\_is\_middle == true)

{

if (CheckWordInBase(newWord) == true)

{

if (CountUsedWord == 0)

{

UsedWord[CountUsedWord] = newWord;

CountUsedWord++;

if (x2 > 0)

WordPrice \*= 2;

if (x3 > 0)

WordPrice \*= 3;

newWordPrice = WordPrice;

x2 = 0;

x3 = 0;

WordPrice = 0;

}

else

{

int CheckNewWord = 0;

for (int a = 0; a < CountUsedWord; a++)

{

if (UsedWord[a] == newWord)

CheckNewWord++;

}

if (CheckNewWord == 0)

{

UsedWord[CountUsedWord] = newWord;

CountUsedWord++;

if (x2 > 0)

WordPrice \*= 2;

if (x3 > 0)

WordPrice \*= 3;

newWordPrice = WordPrice;

x2 = 0;

x3 = 0;

WordPrice = 0;

}

}

}

else

{

DialogResult result = MessageBox.Show($"Такого слово как {newWord} нет, но если вы уверены что оно есть то, хотите ли добавить это слово в список сущестующих слов?", "Ошибка в слове!", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (result == DialogResult.Yes)

{

using (StreamWriter stream = new StreamWriter(path, true))

stream.WriteLine(newWord);

if (CheckWordInBase(newWord) == true)

{

if (CountUsedWord == 0)

{

UsedWord[CountUsedWord] = newWord;

CountUsedWord++;

if (x2 > 0)

WordPrice \*= 2;

if (x3 > 0)

WordPrice \*= 3;

newWordPrice = WordPrice;

x2 = 0;

x3 = 0;

WordPrice = 0;

stop = false;

}

else

{

int CheckNewWord = 0;

for (int a = 0; a < CountUsedWord; a++)

{

if (UsedWord[a] == newWord)

CheckNewWord++;

}

if (CheckNewWord == 0)

{

UsedWord[CountUsedWord] = newWord;

CountUsedWord++;

if (x2 > 0)

WordPrice \*= 2;

if (x3 > 0)

WordPrice \*= 3;

newWordPrice = WordPrice;

x2 = 0;

x3 = 0;

WordPrice = 0;

}

}

}

}

else if (result == DialogResult.No)

{

ClearArea();

if (move == 1)

LetterChange(player1\_letters);

if (move == 2)

LetterChange(player2\_letters);

stop = true;

}

}

}

else

{

LetterChange(player1\_letters);

for (int s = 0; s < CountUsedWord; s++)

{

UsedWord[s] = "";

}

CountUsedWord = 0;

newWord = "";

newWordPrice = 0;

ClearArea();

stop = true;

MessageBox.Show("Первое слово должно проходить через центральный элемент! Пожалуйста переделайте!", "Нарушение правил", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

newWord = "";

x2 = 0;

x3 = 0;

WordPrice = 0;

}

}

}

//-------------------------------------------------------------------------

//Дает сумму балов за слово---------------------------------------------====

public int WordPoints(TextBox let, ref int x2, ref int x3) {

//Проверка на первый ход в золотой клетке

if (move\_counter == 1)

{

if (let == r113)

first\_position\_is\_middle = true;

}

int price = 0;

//Нахождение цены буквы

for(int i=0; i<mas.Length; i++)

{

if (let.Text == mas[i].name)

{

price = mas[i].price;

i = mas.Length;

}

}

//Проверка на прохождение ячейки с синим цветом

if (let == r4 || let == r12 || let == r46 || let == r39 || let == r93 ||

let == r97 || let == r60 || let == r37 || let == r53 || let == r117 ||

let == r99 || let == r123 || let == r127 || let == r129 || let == r133 ||

let == r103 || let == r166 || let == r173 || let == r180 || let == r187 ||

let == r109 || let == r189 || let == r214 || let == r222)

price \*= 2;

//Проверка на прохождение ячейки с красным цветом

if (let == r21 || let == r25 || let == r77 ||

let == r81 || let == r85 || let == r89 ||

let == r137 || let == r141 || let == r145 ||

let == r149 || let == r201 || let == r204)

price \*= 3;

//Проверка на прохождение ячейки с желтым цветом

if (let == r17 || let == r29 || let == r33 || let == r43 ||

let == r49 || let == r57 || let == r65 || let == r71 ||

let == r155 || let == r161 || let == r169 || let == r177 ||

let == r183 || let == r193 || let == r197 || let == r209)

x2++;

//Проверка на прохождение ячейки с зеленым цветом

if (let == r1 || let == r8 || let == r15 || let == r106 ||

let == r120 || let == r211 || let == r218 || let == r225)

x3++;

return price;

}

//-------------------------------------------------------------------------

//Проверяет есть ли слово в базе---------------------------------------====

public bool CheckWordInBase(string slovo)

{

using(StreamReader stream = new StreamReader(path))

{

while(stream.ReadLine() != null)

{

if(slovo == stream.ReadLine())

return true;

}

}

return false;

}

//-------------------------------------------------------------------------

//Временное хранилище букв

string Saved\_letters = "", Saved\_Price="";

//Взятие буквы с игровых полей во временное хранилище-------------------====

private void playerbox1\_MouseClick(object sender, MouseEventArgs e)

{

if (playerbox1.Text != "" && Saved\_letters == "")

{

Saved\_letters = playerbox1.Text;

Saved\_Price = playerlabel1.Text;

playerbox1.Text = "";

playerlabel1.Text = "";

}

else

if (playerbox1.Text == "" && Saved\_letters != "")

{

playerbox1.Text = Saved\_letters;

playerlabel1.Text = Saved\_Price;

Saved\_letters = "";

Saved\_Price = "";

}

}

private void playerbox2\_MouseClick(object sender, MouseEventArgs e)

{

if (playerbox2.Text != "" && Saved\_letters == "")

{

Saved\_letters = playerbox2.Text;

Saved\_Price = playerlabel2.Text;

playerbox2.Text = "";

playerlabel2.Text = "";

}

else

if (playerbox2.Text == "" && Saved\_letters != "")

{

playerbox2.Text = Saved\_letters;

playerlabel2.Text = Saved\_Price;

Saved\_letters = "";

Saved\_Price = "";

}

}

private void playerbox3\_MouseClick(object sender, MouseEventArgs e)

{

if (playerbox3.Text != "" && Saved\_letters == "")

{

Saved\_letters = playerbox3.Text;

Saved\_Price = playerlabel3.Text;

playerbox3.Text = "";

playerlabel3.Text = "";

}

else

if (playerbox3.Text == "" && Saved\_letters != "")

{

playerbox3.Text = Saved\_letters;

playerlabel3.Text = Saved\_Price;

Saved\_letters = "";

Saved\_Price = "";

}

}

private void playerbox4\_MouseClick(object sender, MouseEventArgs e)

{

if (playerbox4.Text != "" && Saved\_letters == "")

{

Saved\_letters = playerbox4.Text;

Saved\_Price = playerlabel4.Text;

playerbox4.Text = "";

playerlabel4.Text = "";

}

else

if (playerbox4.Text == "" && Saved\_letters != "")

{

playerbox4.Text = Saved\_letters;

playerlabel4.Text = Saved\_Price;

Saved\_letters = "";

Saved\_Price = "";

}

}

private void playerbox5\_MouseClick(object sender, MouseEventArgs e)

{

if (playerbox5.Text != "" && Saved\_letters == "")

{

Saved\_letters = playerbox5.Text;

Saved\_Price = playerlabel5.Text;

playerbox5.Text = "";

playerlabel5.Text = "";

}

else

if (playerbox5.Text == "" && Saved\_letters != "")

{

playerbox5.Text = Saved\_letters;

playerlabel5.Text = Saved\_Price;

Saved\_letters = "";

Saved\_Price = "";

}

}

private void playerbox6\_MouseClick(object sender, MouseEventArgs e)

{

if (playerbox6.Text != "" && Saved\_letters == "")

{

Saved\_letters = playerbox6.Text;

Saved\_Price = playerlabel6.Text;

playerbox6.Text = "";

playerlabel6.Text = "";

}

else

if (playerbox6.Text == "" && Saved\_letters != "")

{

playerbox6.Text = Saved\_letters;

playerlabel6.Text = Saved\_Price;

Saved\_letters = "";

Saved\_Price = "";

}

}

private void playerbox7\_MouseClick(object sender, MouseEventArgs e)

{

if (playerbox7.Text != "" && Saved\_letters == "")

{

Saved\_letters = playerbox7.Text;

Saved\_Price = playerlabel7.Text;

playerbox7.Text = "";

playerlabel7.Text = "";

}

else

if (playerbox7.Text == "" && Saved\_letters != "")

{

playerbox7.Text = Saved\_letters;

playerlabel7.Text = Saved\_Price;

Saved\_letters = "";

Saved\_Price = "";

}

}

//--------------------------------------------------------------------------

//Позваляет менять позицию букву только во время своего хода

int[] move\_counter\_letters = new int[226];

//Вызывает завершение игры----------------------------------------------====

public void EndGame()

{

//Сравнивает количеств очков у двух игроков

if (counter1 == counter2)

MessageBox.Show("Ничья", "Результат", MessageBoxButtons.OK);

if (counter1 > counter2)

MessageBox.Show("Выиграл 1-ый игрок", "Результат", MessageBoxButtons.OK);

if (counter1 < counter2)

MessageBox.Show("Выиграл 2-ой игрок", "Результат", MessageBoxButtons.OK);

this.Hide(); //Скрывается из виду игровое окно

Form1 form1 = new Form1(); // Создается экзепляр для первого поля

form1.Show(); //Окно правил становиться видимым

}

//--------------------------------------------------------------------------

//Очистка поля---------------------------------------------------------====

public void ClearArea()

{

if (move\_counter\_letters[1] == move\_counter)

r1.Text = "";

if (move\_counter\_letters[2] == move\_counter)

r2.Text = "";

if (move\_counter\_letters[3] == move\_counter)

r3.Text = "";

if (move\_counter\_letters[4] == move\_counter)

r4.Text = "";

if (move\_counter\_letters[5] == move\_counter)

r5.Text = "";

if (move\_counter\_letters[6] == move\_counter)

r6.Text = "";

if (move\_counter\_letters[7] == move\_counter)

r7.Text = "";

if (move\_counter\_letters[8] == move\_counter)

r8.Text = "";

if (move\_counter\_letters[9] == move\_counter)

r9.Text = "";

if (move\_counter\_letters[10] == move\_counter)

r10.Text = "";

if (move\_counter\_letters[11] == move\_counter)

r11.Text = "";

if (move\_counter\_letters[12] == move\_counter)

r12.Text = "";

if (move\_counter\_letters[13] == move\_counter)

r13.Text = "";

if (move\_counter\_letters[14] == move\_counter)

r14.Text = "";

if (move\_counter\_letters[15] == move\_counter)

r15.Text = "";

if (move\_counter\_letters[16] == move\_counter)

r16.Text = "";

if (move\_counter\_letters[17] == move\_counter)

r17.Text = "";

if (move\_counter\_letters[18] == move\_counter)

r18.Text = "";

if (move\_counter\_letters[19] == move\_counter)

r19.Text = "";

if (move\_counter\_letters[20] == move\_counter)

r20.Text = "";

if (move\_counter\_letters[21] == move\_counter)

r21.Text = "";

if (move\_counter\_letters[22] == move\_counter)

r22.Text = "";

if (move\_counter\_letters[23] == move\_counter)

r23.Text = "";

if (move\_counter\_letters[24] == move\_counter)

r24.Text = "";

if (move\_counter\_letters[25] == move\_counter)

r25.Text = "";

if (move\_counter\_letters[26] == move\_counter)

r26.Text = "";

if (move\_counter\_letters[27] == move\_counter)

r27.Text = "";

if (move\_counter\_letters[28] == move\_counter)

r28.Text = "";

if (move\_counter\_letters[29] == move\_counter)

r29.Text = "";

if (move\_counter\_letters[30] == move\_counter)

r30.Text = "";

if (move\_counter\_letters[31] == move\_counter)

r31.Text = "";

if (move\_counter\_letters[32] == move\_counter)

r32.Text = "";

if (move\_counter\_letters[33] == move\_counter)

r33.Text = "";

if (move\_counter\_letters[34] == move\_counter)

r34.Text = "";

if (move\_counter\_letters[35] == move\_counter)

r35.Text = "";

if (move\_counter\_letters[36] == move\_counter)

r36.Text = "";

if (move\_counter\_letters[37] == move\_counter)

r37.Text = "";

if (move\_counter\_letters[38] == move\_counter)

r38.Text = "";

if (move\_counter\_letters[39] == move\_counter)

r39.Text = "";

if (move\_counter\_letters[40] == move\_counter)

r40.Text = "";

if (move\_counter\_letters[41] == move\_counter)

r41.Text = "";

if (move\_counter\_letters[42] == move\_counter)

r42.Text = "";

if (move\_counter\_letters[43] == move\_counter)

r43.Text = "";

if (move\_counter\_letters[44] == move\_counter)

r44.Text = "";

if (move\_counter\_letters[45] == move\_counter)

r45.Text = "";

if (move\_counter\_letters[46] == move\_counter)

r46.Text = "";

if (move\_counter\_letters[47] == move\_counter)

r47.Text = "";

if (move\_counter\_letters[48] == move\_counter)

r48.Text = "";

if (move\_counter\_letters[49] == move\_counter)

r49.Text = "";

if (move\_counter\_letters[50] == move\_counter)

r50.Text = "";

if (move\_counter\_letters[51] == move\_counter)

r51.Text = "";

if (move\_counter\_letters[52] == move\_counter)

r52.Text = "";

if (move\_counter\_letters[53] == move\_counter)

r53.Text = "";

if (move\_counter\_letters[54] == move\_counter)

r54.Text = "";

if (move\_counter\_letters[55] == move\_counter)

r55.Text = "";

if (move\_counter\_letters[56] == move\_counter)

r56.Text = "";

if (move\_counter\_letters[57] == move\_counter)

r57.Text = "";

if (move\_counter\_letters[58] == move\_counter)

r58.Text = "";

if (move\_counter\_letters[59] == move\_counter)

r59.Text = "";

if (move\_counter\_letters[60] == move\_counter)

r60.Text = "";

if (move\_counter\_letters[61] == move\_counter)

r61.Text = "";

if (move\_counter\_letters[62] == move\_counter)

r62.Text = "";

if (move\_counter\_letters[63] == move\_counter)

r63.Text = "";

if (move\_counter\_letters[64] == move\_counter)

r64.Text = "";

if (move\_counter\_letters[65] == move\_counter)

r65.Text = "";

if (move\_counter\_letters[66] == move\_counter)

r66.Text = "";

if (move\_counter\_letters[67] == move\_counter)

r67.Text = "";

if (move\_counter\_letters[68] == move\_counter)

r68.Text = "";

if (move\_counter\_letters[69] == move\_counter)

r69.Text = "";

if (move\_counter\_letters[70] == move\_counter)

r70.Text = "";

if (move\_counter\_letters[71] == move\_counter)

r71.Text = "";

if (move\_counter\_letters[72] == move\_counter)

r72.Text = "";

if (move\_counter\_letters[73] == move\_counter)

r73.Text = "";

if (move\_counter\_letters[74] == move\_counter)

r.Text = "";

if (move\_counter\_letters[75] == move\_counter)

r75.Text = "";

if (move\_counter\_letters[76] == move\_counter)

r76.Text = "";

if (move\_counter\_letters[77] == move\_counter)

r77.Text = "";

if (move\_counter\_letters[78] == move\_counter)

r78.Text = "";

if (move\_counter\_letters[79] == move\_counter)

r79.Text = "";

if (move\_counter\_letters[80] == move\_counter)

r80.Text = "";

if (move\_counter\_letters[81] == move\_counter)

r81.Text = "";

if (move\_counter\_letters[82] == move\_counter)

r82.Text = "";

if (move\_counter\_letters[83] == move\_counter)

r83.Text = "";

if (move\_counter\_letters[84] == move\_counter)

r84.Text = "";

if (move\_counter\_letters[85] == move\_counter)

r85.Text = "";

if (move\_counter\_letters[86] == move\_counter)

r86.Text = "";

if (move\_counter\_letters[87] == move\_counter)

r87.Text = "";

if (move\_counter\_letters[88] == move\_counter)

r88.Text = "";

if (move\_counter\_letters[89] == move\_counter)

r89.Text = "";

if (move\_counter\_letters[90] == move\_counter)

r90.Text = "";

if (move\_counter\_letters[91] == move\_counter)

r91.Text = "";

if (move\_counter\_letters[92] == move\_counter)

r92.Text = "";

if (move\_counter\_letters[93] == move\_counter)

r93.Text = "";

if (move\_counter\_letters[94] == move\_counter)

r94.Text = "";

if (move\_counter\_letters[95] == move\_counter)

r95.Text = "";

if (move\_counter\_letters[96] == move\_counter)

r96.Text = "";

if (move\_counter\_letters[97] == move\_counter)

r97.Text = "";

if (move\_counter\_letters[98] == move\_counter)

r98.Text = "";

if (move\_counter\_letters[99] == move\_counter)

r99.Text = "";

if (move\_counter\_letters[100] == move\_counter)

r100.Text = "";

if (move\_counter\_letters[101] == move\_counter)

r101.Text = "";

if (move\_counter\_letters[102] == move\_counter)

r102.Text = "";

if (move\_counter\_letters[103] == move\_counter)

r103.Text = "";

if (move\_counter\_letters[104] == move\_counter)

r104.Text = "";

if (move\_counter\_letters[105] == move\_counter)

r105.Text = "";

if (move\_counter\_letters[106] == move\_counter)

r106.Text = "";

if (move\_counter\_letters[107] == move\_counter)

r107.Text = "";

if (move\_counter\_letters[108] == move\_counter)

r108.Text = "";

if (move\_counter\_letters[109] == move\_counter)

r109.Text = "";

if (move\_counter\_letters[110] == move\_counter)

r110.Text = "";

if (move\_counter\_letters[111] == move\_counter)

r111.Text = "";

if (move\_counter\_letters[112] == move\_counter)

r112.Text = "";

if (move\_counter\_letters[113] == move\_counter)

r113.Text = "";

if (move\_counter\_letters[114] == move\_counter)

r114.Text = "";

if (move\_counter\_letters[115] == move\_counter)

r115.Text = "";

if (move\_counter\_letters[116] == move\_counter)

r116.Text = "";

if (move\_counter\_letters[117] == move\_counter)

r117.Text = "";

if (move\_counter\_letters[118] == move\_counter)

r118.Text = "";

if (move\_counter\_letters[119] == move\_counter)

r119.Text = "";

if (move\_counter\_letters[120] == move\_counter)

r120.Text = "";

if (move\_counter\_letters[121] == move\_counter)

r121.Text = "";

if (move\_counter\_letters[122] == move\_counter)

r122.Text = "";

if (move\_counter\_letters[123] == move\_counter)

r123.Text = "";

if (move\_counter\_letters[124] == move\_counter)

r124.Text = "";

if (move\_counter\_letters[125] == move\_counter)

r125.Text = "";

if (move\_counter\_letters[126] == move\_counter)

r126.Text = "";

if (move\_counter\_letters[127] == move\_counter)

r127.Text = "";

if (move\_counter\_letters[128] == move\_counter)

r128.Text = "";

if (move\_counter\_letters[129] == move\_counter)

r129.Text = "";

if (move\_counter\_letters[130] == move\_counter)

r130.Text = "";

if (move\_counter\_letters[131] == move\_counter)

r131.Text = "";

if (move\_counter\_letters[132] == move\_counter)

r132.Text = "";

if (move\_counter\_letters[133] == move\_counter)

r133.Text = "";

if (move\_counter\_letters[134] == move\_counter)

r134.Text = "";

if (move\_counter\_letters[135] == move\_counter)

r135.Text = "";

if (move\_counter\_letters[136] == move\_counter)

r136.Text = "";

if (move\_counter\_letters[137] == move\_counter)

r137.Text = "";

if (move\_counter\_letters[138] == move\_counter)

r138.Text = "";

if (move\_counter\_letters[139] == move\_counter)

r139.Text = "";

if (move\_counter\_letters[140] == move\_counter)

r140.Text = "";

if (move\_counter\_letters[141] == move\_counter)

r141.Text = "";

if (move\_counter\_letters[142] == move\_counter)

r142.Text = "";

if (move\_counter\_letters[143] == move\_counter)

r143.Text = "";

if (move\_counter\_letters[144] == move\_counter)

r144.Text = "";

if (move\_counter\_letters[145] == move\_counter)

r145.Text = "";

if (move\_counter\_letters[146] == move\_counter)

r146.Text = "";

if (move\_counter\_letters[147] == move\_counter)

r147.Text = "";

if (move\_counter\_letters[148] == move\_counter)

r148.Text = "";

if (move\_counter\_letters[149] == move\_counter)

r149.Text = "";

if (move\_counter\_letters[150] == move\_counter)

r150.Text = "";

if (move\_counter\_letters[151] == move\_counter)

r51.Text = "";

if (move\_counter\_letters[152] == move\_counter)

r152.Text = "";

if (move\_counter\_letters[153] == move\_counter)

r153.Text = "";

if (move\_counter\_letters[154] == move\_counter)

r154.Text = "";

if (move\_counter\_letters[155] == move\_counter)

r155.Text = "";

if (move\_counter\_letters[156] == move\_counter)

r156.Text = "";

if (move\_counter\_letters[157] == move\_counter)

r157.Text = "";

if (move\_counter\_letters[158] == move\_counter)

r158.Text = "";

if (move\_counter\_letters[159] == move\_counter)

r159.Text = "";

if (move\_counter\_letters[160] == move\_counter)

r160.Text = "";

if (move\_counter\_letters[161] == move\_counter)

r161.Text = "";

if (move\_counter\_letters[162] == move\_counter)

r162.Text = "";

if (move\_counter\_letters[163] == move\_counter)

r163.Text = "";

if (move\_counter\_letters[164] == move\_counter)

r164.Text = "";

if (move\_counter\_letters[165] == move\_counter)

r165.Text = "";

if (move\_counter\_letters[166] == move\_counter)

r166.Text = "";

if (move\_counter\_letters[167] == move\_counter)

r167.Text = "";

if (move\_counter\_letters[168] == move\_counter)

r168.Text = "";

if (move\_counter\_letters[169] == move\_counter)

r169.Text = "";

if (move\_counter\_letters[170] == move\_counter)

r170.Text = "";

if (move\_counter\_letters[171] == move\_counter)

r171.Text = "";

if (move\_counter\_letters[172] == move\_counter)

r172.Text = "";

if (move\_counter\_letters[173] == move\_counter)

r173.Text = "";

if (move\_counter\_letters[174] == move\_counter)

r.Text = "";

if (move\_counter\_letters[175] == move\_counter)

r175.Text = "";

if (move\_counter\_letters[176] == move\_counter)

r176.Text = "";

if (move\_counter\_letters[177] == move\_counter)

r177.Text = "";

if (move\_counter\_letters[178] == move\_counter)

r178.Text = "";

if (move\_counter\_letters[179] == move\_counter)

r179.Text = "";

if (move\_counter\_letters[180] == move\_counter)

r80.Text = "";

if (move\_counter\_letters[181] == move\_counter)

r181.Text = "";

if (move\_counter\_letters[182] == move\_counter)

r182.Text = "";

if (move\_counter\_letters[183] == move\_counter)

r183.Text = "";

if (move\_counter\_letters[184] == move\_counter)

r184.Text = "";

if (move\_counter\_letters[185] == move\_counter)

r185.Text = "";

if (move\_counter\_letters[186] == move\_counter)

r186.Text = "";

if (move\_counter\_letters[187] == move\_counter)

r187.Text = "";

if (move\_counter\_letters[188] == move\_counter)

r188.Text = "";

if (move\_counter\_letters[189] == move\_counter)

r189.Text = "";

if (move\_counter\_letters[190] == move\_counter)

r190.Text = "";

if (move\_counter\_letters[191] == move\_counter)

r191.Text = "";

if (move\_counter\_letters[192] == move\_counter)

r192.Text = "";

if (move\_counter\_letters[193] == move\_counter)

r193.Text = "";

if (move\_counter\_letters[194] == move\_counter)

r194.Text = "";

if (move\_counter\_letters[195] == move\_counter)

r195.Text = "";

if (move\_counter\_letters[196] == move\_counter)

r196.Text = "";

if (move\_counter\_letters[197] == move\_counter)

r197.Text = "";

if (move\_counter\_letters[198] == move\_counter)

r198.Text = "";

if (move\_counter\_letters[199] == move\_counter)

r199.Text = "";

if (move\_counter\_letters[200] == move\_counter)

r200.Text = "";

if (move\_counter\_letters[201] == move\_counter)

r201.Text = "";

if (move\_counter\_letters[202] == move\_counter)

r202.Text = "";

if (move\_counter\_letters[203] == move\_counter)

r203.Text = "";

if (move\_counter\_letters[204] == move\_counter)

r204.Text = "";

if (move\_counter\_letters[205] == move\_counter)

r205.Text = "";

if (move\_counter\_letters[206] == move\_counter)

r206.Text = "";

if (move\_counter\_letters[207] == move\_counter)

r207.Text = "";

if (move\_counter\_letters[208] == move\_counter)

r208.Text = "";

if (move\_counter\_letters[209] == move\_counter)

r209.Text = "";

if (move\_counter\_letters[210] == move\_counter)

r210.Text = "";

if (move\_counter\_letters[211] == move\_counter)

r211.Text = "";

if (move\_counter\_letters[212] == move\_counter)

r212.Text = "";

if (move\_counter\_letters[213] == move\_counter)

r213.Text = "";

if (move\_counter\_letters[214] == move\_counter)

r214.Text = "";

if (move\_counter\_letters[215] == move\_counter)

r215.Text = "";

if (move\_counter\_letters[216] == move\_counter)

r216.Text = "";

if (move\_counter\_letters[217] == move\_counter)

r217.Text = "";

if (move\_counter\_letters[218] == move\_counter)

r218.Text = "";

if (move\_counter\_letters[219] == move\_counter)

r219.Text = "";

if (move\_counter\_letters[220] == move\_counter)

r220.Text = "";

if (move\_counter\_letters[221] == move\_counter)

r221.Text = "";

if (move\_counter\_letters[222] == move\_counter)

r222.Text = "";

if (move\_counter\_letters[223] == move\_counter)

r223.Text = "";

if (move\_counter\_letters[224] == move\_counter)

r224.Text = "";

if (move\_counter\_letters[225] == move\_counter)

r225.Text = "";

}

//-------------------------------------------------------------------------

//Перенос текста с сейва в любую точку на поле--------------------------====

public void letter\_wrap(TextBox name, int n)

{

if (name.Text == "" && Saved\_letters != "")

{

name.Text = Saved\_letters;

Saved\_letters = "";

move\_counter\_letters[n] = move\_counter;

}

else

if (name.Text != "" && Saved\_letters == "" && move\_counter\_letters[n] == move\_counter)

{

Saved\_letters = name.Text;

name.Text = "";

}

}

private void r1\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r1,1);

}

private void r2\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r2,2);

}

private void r3\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r3,3);

}

private void r4\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r4,4);

}

private void r5\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r5,5);

}

private void r6\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r6,6);

}

private void r7\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r7,7);

}

private void r8\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r8,8);

}

private void r9\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r9,9);

}

private void r10\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r10,10);

}

private void r11\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r11, 11);

}

private void r12\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r12, 12);

}

private void r13\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r13, 13);

}

private void r14\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r14, 14);

}

private void r15\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r15, 15);

}

private void r16\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r16, 16);

}

private void r17\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r17, 17);

}

private void r18\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r18, 18);

}

private void r19\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r19, 19);

}

private void r20\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r20, 20);

}

private void r21\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r21, 21);

}

private void r22\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r22, 22);

}

private void r23\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r23, 23);

}

private void r24\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r24, 24);

}

private void r25\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r25, 25);

}

private void r26\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r26, 26);

}

private void r27\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r27, 27);

}

private void r28\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r28, 28);

}

private void r29\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r29, 29);

}

private void r30\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r30, 30);

}

private void r31\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r31, 31);

}

private void r32\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r32, 32);

}

private void r33\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r33, 33);

}

private void r34\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r34, 34);

}

private void r35\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r35, 35);

}

private void r36\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r36, 36);

}

private void r37\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r37, 37);

}

private void r38\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r38, 38);

}

private void r39\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r39, 39);

}

private void r40\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r40, 40);

}

private void r41\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r41, 41);

}

private void r42\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r42, 42);

}

private void r43\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r43, 43);

}

private void r44\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r44, 44);

}

private void r45\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r45, 45);

}

private void r46\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r46, 46);

}

private void r47\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r47, 47);

}

private void r48\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r48, 48);

}

private void r49\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r49, 49);

}

private void r50\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r50, 50);

}

private void r51\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r51, 51);

}

private void r52\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r52, 52);

}

private void r53\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r53, 53);

}

private void r54\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r54, 54);

}

private void r55\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r55, 55);

}

private void r56\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r56, 56);

}

private void r57\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r57, 57);

}

private void r58\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r58, 58);

}

private void r59\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r59, 59);

}

private void r60\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r60, 60);

}

private void r61\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r61, 61);

}

private void r62\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r62, 62);

}

private void r63\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r63, 63);

}

private void r64\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r64, 64);

}

private void r65\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r65, 65);

}

private void r66\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r66, 66);

}

private void r67\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r67, 67);

}

private void r68\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r68, 68);

}

private void r69\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r69, 69);

}

private void r70\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r70, 70);

}

private void r71\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r71, 71);

}

private void r72\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r72, 72);

}

private void r73\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r73, 73);

}

private void r\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r, 74);

}

private void r75\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r75, 75);

}

private void r76\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r76, 76);

}

private void r77\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r77, 77);

}

private void r78\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r78, 78);

}

private void r79\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r79, 79);

}

private void r80\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r80, 80);

}

private void r81\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r81, 81);

}

private void r82\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r82, 82);

}

private void r83\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r83, 83);

}

private void r84\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r84, 84);

}

private void r85\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r85, 85);

}

private void r86\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r86, 86);

}

private void r87\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r87, 87);

}

private void r88\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r88, 88);

}

private void r89\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r89, 89);

}

private void r90\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r90, 90);

}

private void r91\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r91, 91);

}

private void r92\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r92, 92);

}

private void r93\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r93, 93);

}

private void r94\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r94, 94);

}

private void r95\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r95, 95);

}

private void r96\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r96, 96);

}

private void r97\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r97, 97);

}

private void r98\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r98, 98);

}

private void r99\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r99, 99);

}

private void r100\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r100, 100);

}

private void r101\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r101, 101);

}

private void r102\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r102, 102);

}

private void r103\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r103, 103);

}

private void r104\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r104, 104);

}

private void r105\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r105, 105);

}

private void r106\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r106, 106);

}

private void r107\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r107, 107);

}

private void r108\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r108, 108);

}

private void r109\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r109, 109);

}

private void r110\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r110, 110);

}

private void r111\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r111, 111);

}

private void r112\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r112, 112);

}

private void r113\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r113, 113);

}

private void r114\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r114, 114);

}

private void r115\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r115, 115);

}

private void r116\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r116, 116);

}

private void r117\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r117, 117);

}

private void r118\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r118, 118);

}

private void r119\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r119, 119);

}

private void r120\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r120, 120);

}

private void r121\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r121, 121);

}

private void r122\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r122, 122);

}

private void r123\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r123, 123);

}

private void r124\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r124, 124);

}

private void r125\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r125, 125);

}

private void r126\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r126, 126);

}

private void r127\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r127, 127);

}

private void r128\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r128, 128);

}

private void r129\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r129, 129);

}

private void r130\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r130, 130);

}

private void r131\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r131, 131);

}

private void r132\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r132, 132);

}

private void r133\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r133, 133);

}

private void r134\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r134, 134);

}

private void r135\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r135, 135);

}

private void r136\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r136, 136);

}

private void r137\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r137, 137);

}

private void r138\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r138, 1358);

}

private void r139\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r139, 139);

}

private void r140\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r140, 140);

}

private void r141\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r141, 141);

}

private void r142\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r142, 142);

}

private void r143\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r143, 143);

}

private void r144\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r144, 144);

}

private void r145\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r145, 145);

}

private void r146\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r146, 146);

}

private void r147\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r147, 147);

}

private void r148\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r148, 148);

}

private void r149\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r149, 149);

}

private void r150\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r150, 150);

}

private void r151\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r151, 151);

}

private void r152\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r152, 152);

}

private void r153\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r153, 153);

}

private void r154\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r154, 154);

}

private void r155\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r155, 155);

}

private void r156\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r156, 156);

}

private void r157\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r157, 157);

}

private void r158\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r158, 158);

}

private void r159\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r159, 159);

}

private void r160\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r160, 160);

}

private void r161\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r161, 161);

}

private void r162\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r162, 162);

}

private void r163\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r163, 163);

}

private void r164\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r164, 164);

}

private void r165\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r165, 165);

}

private void r166\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r166, 166);

}

private void r167\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r167, 167);

}

private void r168\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r168, 168);

}

private void r169\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r169, 169);

}

private void r170\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r170, 170);

}

private void r171\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r171, 171);

}

private void r172\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r172, 172);

}

private void r173\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r173, 173);

}

private void r174\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r174, 174);

}

private void r175\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r175, 175);

}

private void r176\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r176, 176);

}

private void r177\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r177, 177);

}

private void r178\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r178, 178);

}

private void r179\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r179, 179);

}

private void r180\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r180, 180);

}

private void r181\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r181, 181);

}

private void r182\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r182, 182);

}

private void r183\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r183, 183);

}

private void r184\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r184, 184);

}

private void r185\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r185, 185);

}

private void r186\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r186, 186);

}

private void r187\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r187, 187);

}

private void r188\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r188, 188);

}

private void r189\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r189, 189);

}

private void r190\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r190, 190);

}

private void r191\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r191, 191);

}

private void r192\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r192, 192);

}

private void r193\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r193, 193);

}

private void r194\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r194, 194);

}

private void r195\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r195, 195);

}

private void r196\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r196, 196);

}

private void r197\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r197, 197);

}

private void r198\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r198, 198);

}

private void r199\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r199, 199);

}

private void r200\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r200, 200);

}

private void r201\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r201, 201);

}

private void r202\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r202, 202);

}

private void r203\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r203, 203);

}

private void r204\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r204, 204);

}

private void r205\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r205, 205);

}

private void r206\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r206, 206);

}

private void r207\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r207, 207);

}

private void r208\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r208, 208);

}

private void r209\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r209, 209);

}

private void r210\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r210, 210);

}

private void r211\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r211, 211);

}

private void r212\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r212, 212);

}

private void r213\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r213, 213);

}

private void r214\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r214, 214);

}

private void r215\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r215, 215);

}

private void r216\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r216, 216);

}

private void r217\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r217, 217);

}

private void r218\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r218, 218);

}

private void r219\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r219, 219);

}

private void r220\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r220, 220);

}

private void r221\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r221, 221);

}

private void r222\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r222, 222);

}

private void r223\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r223, 223);

}

private void r224\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r224, 224);

}

private void r225\_MouseClick(object sender, MouseEventArgs e)

{

letter\_wrap(r225, 225);

}

//-----------------------------------------------------------------------------------------

private void GameLobby\_Load(object sender, EventArgs e)

{

}

}

}