Министерство сельского хозяйства Российской Федерации

ФГБОУ ВО

КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ им. И.Т. Трубилина

Кафедра системного анализа и обработки информации

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

к курсовому проекту

по дисциплине: **Алгоритмизация и программирование**

на тему: Программирование игрового приложения "4 фото 1 слово".

Выполнил студент группы ИТ 1721

Рылов Михаил Леонидович .

Допущен к защите

Руководитель проекта: Доцент, кандидат физико-математических наук, доцент ВАК Павлов Д. А.

Нормоконтролер \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись, расшифровка подписи)

Защищен\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Оценка\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(дата)

Члены комиссии\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Крамаренко Т.А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Ефанова Н.В.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись, дата, расшифровка подписи)

Краснодар

2017 г.

ФГБОУ ВО

КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ им. И.Т. Трубилина

Кафедра системного анализа и обработки информации

**УТВЕРЖДАЮ:**

**Зав. кафедрой**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ЗАДАНИЕ**

**Курсового проекта**

Студенту: ИТ 1721 группы 1 курса

Факультета Прикладной информатики.

специальности «Информационные системы и технологии» .

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

(Ф.И.О., шифр)

Тема проекта: Программирование игрового приложения "4 фото 1 слово". Надо отгадать слово по 4 четырём картинкам. Чтобы составить слово, надо выбрать из представленных букв нужные (или воспользоваться подсказкой).

Объем проекта:

а) пояснительная записка к проекту\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_листа формата А4

б) графическая часть\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_лист формата А4

Рекомендуемая литература: Язык программирования C# 5.0 и платформа .NET 4.5 Эндрю Троелсен.

Срок выполнения проекта: с “2” октября 2017 по “12” января 2018 г.

Срок защиты: “12” января 2018

Дата выдачи задания: “2” октября 2017

Дата сдачи проекта на кафедру: “\_\_\_”\_\_\_\_\_\_\_\_20\_\_ г.

Руководитель проекта \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись, Ф.И.О., звание, степень)

Задание принял студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись, дата)

Краснодар

2017 г.

РЕФЕРАТ

27с., 6 рис., 1 прил., 1 блок-схема

«4 ФОТО 1 СЛОВО», ИГРА, УГАДАТЬ СЛОВО, ПРИЛОЖЕНИЕ, C#

Целью проекта является создание игрового приложения "4 фото 1 слово" в среде Microsoft Visual Studio на языке программирования С#.

Разработанная программа позволяет загадывать слово (по четырём картинкам), накапливать игровые очки за прохождение уровней игры.

**Содержание**

[ВВЕДЕНИЕ 5](#_Toc503534787)

[1 ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ 6](#_Toc503534788)

[1.1 Задачи исследования 6](#_Toc503534789)

[1.2 Сведения из теории 6](#_Toc503534790)

[1.3 Входные и выходные данные 6](#_Toc503534791)

[2 РЕАЛИЗАЦИЯ ПРОЕКТА 7](#_Toc503534792)

[2.1 Алгоритм решения 7](#_Toc503534793)

[2.2 Макет программы 9](#_Toc503534794)

[2.3 Описание программы 10](#_Toc503534795)

[2.4 Результат работы программы 16](#_Toc503534796)

[3 РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ 20](#_Toc503534797)

[ЗАКЛЮЧЕНИЕ 22](#_Toc503534798)

[ЛИТЕРАТУРА 23](#_Toc503534799)

[ПРИЛОЖЕНИЕ 24](#_Toc503534800)

# ВВЕДЕНИЕ

Все более расширяющийся рынок персональной электроники, а с ним и рынок развлечений – неоспоримые аргументы в пользу осуществления проектов по созданию новых игр.

Приспособленность к мобильным устройствам, таким как планшеты и телефоны, позволяет играм расширять свое присутствие на всевозможных платформах – это возможность выпуска на наиболее быстрорастущем сегменте рынка.

Повсеместное распространение и большая доступность инструментальных средств, легкий доступ к информационным обучающим и справочным материалам позволяют уверенно повышать эффективность процесса разработки, и уменьшение сопутствующих трудозатрат.

Цель работы – проектирование игрового процесса и осуществление этапов разработки компьютерной игры.

## ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

# 1.1 Задачи исследования

Задачи исследования – изучить основные сведения из теории, описать входные и выходные данные приложения и сделать подробное описание программы.

# Сведения из теории

Головоломки – это всегда успех у геймеров, и игра 4 картинки 1 слово, вышедшая совсем недавно – не исключение. Это увлекательное приложение создано для тренировки логики и памяти.

Сюжет игры 4 картинки 1 слово прост – игроку предлагается посмотреть на 4 разных картинки и определить, какое слово «спрятано» в них. Изображения даются не просто так: все они объединены тем самым словом, которое и нужно определить. Например, даются изображения бегущего по траве котенка, лежащего мяча на футбольном поле, красивого цветка на поляне и лужайки около дома. Единственное, что находится на всех четырех картинках – это трава. Она и будет тем словом, которое нужно угадать.

# 1.3 Входные и выходные данные

Входными данными будет нажатие кнопки мыши на одну из кнопок массива.

Выходными данными будут выведение пользовательского ответа и информация о правильности или неверности данного ответа , начисление игровых очков, смена уровня и соответственно картинок.

# 2 РЕАЛИЗАЦИЯ ПРОЕКТА

# 2.1 Алгоритм решения

Алгоритм решения, представленный в программе:

1. Создаётся массив кнопок, в котором задаются буквы загаданного слова и рандомные
2. В обработчике событий нажатия на кнопку массива, значение, лежащее в этой кнопке переместиться в пустой массив, предназначенный для ответа
3. Создаётся пустой массив кнопок, в который будут помещаться выбранные буквы предполагаемого ответа пользователем и при желании убираться при помощи клика оттуда в случае ошибки
4. Составленное слово сравнивается с ответом, и в случае совпадения игра переходит на следующий уровень

выход

Для каждого из уровней

начало

полученное слово =ответу

Складываем эти буквы в слово

Ввод пользователем

букв в пустой массив

из данного заполненного

Для каждой из этих букв

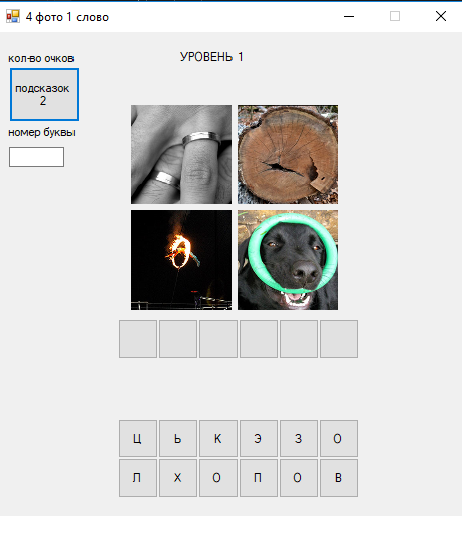
букв

Рисунок 1 – Блок-схема

Блок-схема на рисунке 1 отображает схематичную реализацию программы.

# 2.2 Макет программы

На рисунке 2 отображен основной макет программы, где обозначены основные элементы.



label

TextBox1

button

button

pictureBox

Рисунок 2 – макет программы

# 2.3 Описание программы

Ключевые переменные, которые используются в программе:

* ochki – целочисленная переменная, в которой хранятся игровые очки
* otv – строковая переменная, в которую заносятся буквы, выбранные пользователем для сравнения с заданным ответом
* ur – целочисленная переменная, в которой хранится информация об текущем уровне
* a – строковая переменная, в которой хранится заданный ответ
* string[] otvety – массив, в котором хранятся ответы
* char[] array – массив, в котором хранятся буквы основного поля
* int[] sluch – массив, в котором хранятся случайные числа от 1 до 12, для рандомного расположения букв массива array в основном поле
* Button[,] S – массив кнопок основного поля
* Button[] C – массив кнопок поля ввода

Программа состоит из 3 функций

Первая функция формирует массив кнопок 6 на 2

public void InitB(ref string a) // создание массива 6 на 2

{

Razrez(ref a, ref array, ref sluch);

for (int i = 0; i < 6; i++)

for (int j = 0; j < 2; j++)

{

S[i, j] = new Button();//создание кнопки

S[i, j].Text = "";//текст кнопки(пусто)

S[i, j].Enabled = true;//включает элемент управления

S[i, j].Click += new EventHandler(this.but\_Click);//добавляет обработчик событий нажатия мыши

S[i, j].Location = new Point(120+i\*40, j \* 40 + 390);//задает местоположение элемента

S[i, j].Size = new Size(40, 40);//задает размер элемента

S[i, j].Text = array[p].ToString();

S[i, j].TabStop = false;//убирает табулирование

p++;//увеличивает счётчик

this.Controls.Add(S[i, j]);//возвращение коллекции элементов управление

}

}

Функция Razrez является основной для заполнения массива кнопок 6 на 2:

Random rnd = new Random();

void Razrez(ref string a, ref char[] array,ref int[] sluch)

{

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

array[i] = a[i];

for (int i = a.Length; i < 12; i++)

array[i] = Convert.ToChar(rnd.Next('А', 'Я' + 1));

for (int i = 0; i < 12; i++)

{

sluch[i] = rnd.Next(0, 12);

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

{

if (sluch[i] == sluch[j])

{

i--;

break;

}

}

}

for (int i = 0; i<12; i++)

{

char temp = array[sluch[i]];

array[sluch[i]] = array[i];

array[i] = temp;

}

}

В этой функции в массив array вначале заносятся значение переменной a, которая является правильным ответом:

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

{

array[i] = a[i];

}

Оставшееся место в массиве заполняем случайными символами из алфавита:

for (int i = a.Length; i < 12; i++)

{

array[i] = Convert.ToChar(rnd.Next('А', 'Я' + 1));

}

Потом «перемешаем» элементы в массиве:

for (int i = 0; i < 12; i++)

{

sluch[i] = rnd.Next(0, 12);

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

{

if (sluch[i] == sluch[j])

{

i--;

break;

}

}

}

for (int i = 0; i<12; i++)

{

char temp = array[sluch[i]];

array[sluch[i]] = array[i];

array[i] = temp;

}

Третья функция InitC создает пустой массив кнопок, куда будут заноситься значения при нажатии кнопки мыши массива InitB:

public void InitC(ref string a,ref string b)

{

for (int j = 0; j < a.Length; j++)

{

C[j] = new Button();

if (b.Length>0)

{

if (a.Length > b.Length)

{

for (int i = b.Length; i < a.Length; i++)

C[i].Visible = false;

}

a = b;

}

Array.Resize(ref C, a.Length);//меняем размер массива

C[j].Text = "";

C[j].Click += new System.EventHandler(this.buton\_Click);

C[j].Location = new Point((480 - a.Length \* 40) / 2 + j \* 40, 290);

C[j].Size = new Size(40, 40);

C[j].TabStop = false;

this.Controls.Add(C[j]);

}

}

При переходе на новый уровень, если новый ответ меньше старого(уменьшается размер массива), то:

if (b.Length>0)

{

if (a.Length > b.Length)

{

for (int i = b.Length; i < a.Length; i++)

C[i].Visible = false;

}

a = b;

}

Для реализации функций InitC и InitB их надо вызвать в Form1:

public Form1()

{

InitializeComponent();

InitB(ref a);

InitC(ref a,ref b);

}

Для работы программы нужны 2 обработчика события: but\_Click и buton\_Click.

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

{

Button button = (Button)sender;

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

{

if (C[i].Text == "")

{

C[i].Text = button.Text;

button.Enabled = false;

break;

}

}

for (vr = 0; vr < a.Length; vr++)

otv += C[vr].Text;

if (otv == a)

{

label2.Visible = true;

if (Array.IndexOf(otvety, a) == otvety.Length - 1)

{

pictureBox5.Visible = true;

}

else

button1.Visible = true;

}

else

{

vr = 0;

otv = "";

}

}`

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

{

Button button = (Button)sender;

for (int i = 0; i < 6; i++)

{

for (int j = 0; j < 2; j++)

{

if (button.Text == S[i, j].Text && S[i, j].Enabled == false)

{

S[i, j].Enabled = true;

button.Text = null;

}

}

}

}

Кнопка button1 предназначена для смены уровней игры. После правильного пользовательского ответа button1 появляется:

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

{

ur++;

otv = "";

int kar = ur - 1;

{

pictureBox1.Image = Image.FromFile(kar+"\_0.jpg");

pictureBox2.Image = Image.FromFile(kar+"\_1.jpg");

pictureBox3.Image = Image.FromFile(kar+"\_2.jpg");

pictureBox4.Image = Image.FromFile(kar+"\_3.jpg");

}

label3.Text = ur.ToString();

b = otvety[ur - 2];

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

C[i].Visible = false;

p = 0;

for (int i = 0; i < 6; i++)

for (int j = 0; j < 2; j++)

S[i, j].Visible = false;

InitC(ref a, ref b);

InitB(ref a);

if (Array.IndexOf(otvety, a) == otvety.Length)

pictureBox5.Visible = true;

button1.Visible = false;

label2.Visible = false;

ochki += 2;

button2.Text = "подсказок"+ochki.ToString();

}

Также за реализации подсказок в игре отвечает button2:

private void button2\_Click(object sender, EventArgs e)

{

try

{

int xz = Convert.ToInt32(textBox1.Text) - 1;

if (ochki > 0)

{

string temp = C[xz].Text;

C[xz].Text = Convert.ToString(a[xz]);

for (int i = 0; i < 6; i++)

for (int j = 0; j < 2; j++)

{

if (S[i, j].Text == C[xz].Text)

{

S[i, j].Enabled = false;

j = 2;

i = 6;

}

}

if (C[xz].Text != "")

{

for (int i = 0; i < 6; i++)

for (int j = 0; j < 2; j++)

{

if (S[i, j].Text == temp && S[i,j].Enabled==false)

{

S[i, j].Enabled = true;

j = 2;

i = 6;

}

}

C[xz].Text = Convert.ToString(a[xz]);

}

otv = "";

for (int k = 0; k < a.Length; k++)

otv += C[k].Text;

if (otv == a)

{

button1.Visible = true;

label2.Visible = true;

}

ochki--;

button2.Text = "подсказок"+ochki.ToString();

}

}

catch

{

MessageBox.Show("укажите корректный номер буквы");

}

}

# 2.4 Результат работы программы

При нажатии кнопки основного поля или при использовании подсказок в поле ввода отображаются выбранные (подсказанные) буквы как на рисунке 3.

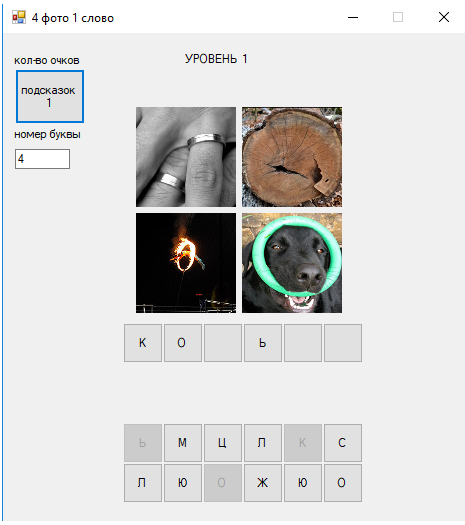


Рисунок 3 – результат 1

При вводе правильного слова в окне приложения отображается информация о верности ответа и кнопка “далее”, для перехода к следующему уровню(рисунок 4).

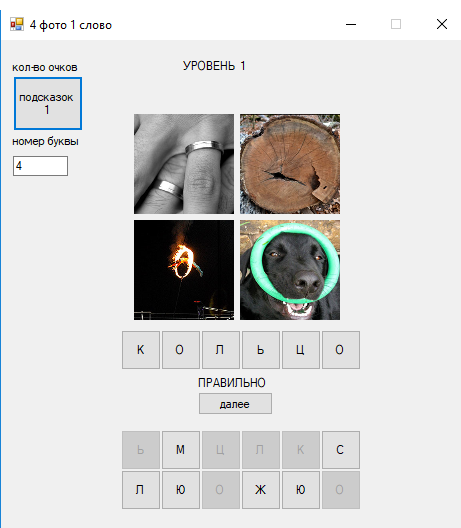


Рисунок 4 – результат 2

По прохождению всех уровней отображается картинка как на рисунке 5.

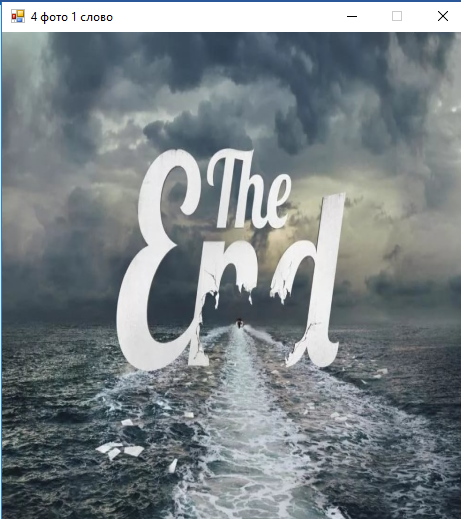


Рисунок 5 – результат 3

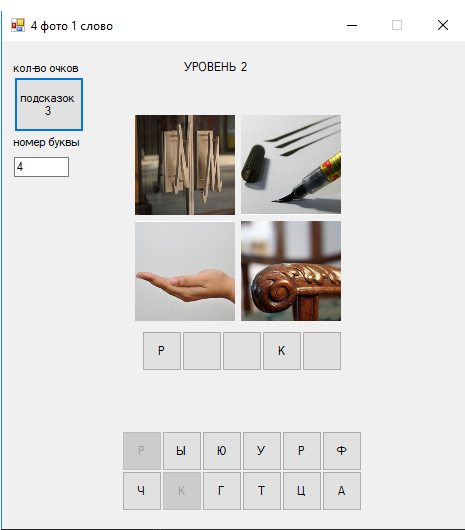
# 3 РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

При запуске программы появится основное окно приложения (смотреть Рисунок 2): 4 картинки, пустое поле для ввода, основное поле, где находятся буквы, в левом верхнем углу – количество очков, которые можно менять на подсказки.

Для переноса буквы из основного поля в поле ввода нажмите на выбранную букву, и она отразится в поле ввода.

Для возврата буквы в основное поле, нажмите на неё в поле ввода.

Для использования подсказок введите в textbox номер буквы, которую хотите открыть и нажмите на кнопку подсказки(за каждую подсказку снимается 1 игровое очко, за прохождение уровня вы получаете 2 очка).



textBox

Кнопка подсказки

Поле ввода

Основное поле

Рисунок 6 – главное окно приложения

# ЗАКЛЮЧЕНИЕ

Программа реализована и работает согласно требованиям и цели работы.

Целью проекта является создание игрового приложения "4 фото 1 слово" в среде Microsoft Visual Studio на языке программирования C#.

Требованиями задания является создание игрового приложения “4 фото 1 слово” и в котором пользователю необходимо отгадать слово, по четырём картинкам. Если пользователь затрудняется с ответом, то он может воспользоваться подсказками.

## ЛИТЕРАТУРА

1. Эндрю Тролсен, Филипп Джепикс. Язык программирования C# 5.0 и платформа .NET 4.5 [Текст] / Э. Тролсен ; пер. с англ. – М. : ООО «И.Д.Вильямс», 2015. – 1440 с.: ил.
2. Хейлсберг Андерсон. Язык программирования C#. Классика Computers Science [Текст] / Х. Андерсон; пер. с англ. – СПб.: Питер, 2012. – 784 с.: ил.
3. Эндрю Стиллмен, Дженнифер Грин. Изучаем C# [Текст] / Э. Стиллмен; пер. с англ. – СПб.: Питер, 2012. – 704 с.: ил.
4. Билл Вагнер. Эффективное программирование на C#. 50 способов улучшения кода [Текст] / Б. Вагнер; пер. с англ. – М. : ООО «И.Д.Вильямс», 2017. – 224 с.: ил.
5. Справочная система MSDN.

## ПРИЛОЖЕНИЕ

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 WindowsFormsApp49

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

InitB(ref a);

InitC(ref a,ref b);

}

int ochki = 2;

int vr;

int[] sluch = new int[12];

string b="";

int ur = 1;

string otv = "";

int p = 0;

string a = "КОЛЬЦО";

string[] otvety = new string[] { "РУЧКА", "ЛЕТО", "СИНИЙ", "СКОРОСТЬ","ВОДА"};

char[] array = new char[12];

Button[,] S = new Button[6, 2];

Button[] C = new Button[10];

Random rnd = new Random();

void Razrez(ref string a, ref char[] array,ref int[] sluch)

{

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

array[i] = a[i];

for (int i = a.Length; i < 12; i++)

array[i] = Convert.ToChar(rnd.Next('А', 'Я' + 1));

for (int i = 0; i < 12; i++)

{

sluch[i] = rnd.Next(0, 12);

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

{

if (sluch[i] == sluch[j])

{

i--;

break;

}

}

}

for (int i = 0; i<12; i++)

{

char temp = array[sluch[i]];

array[sluch[i]] = array[i];

array[i] = temp;

}

}

public void InitB(ref string a)// создание массива 6 на 2

{

Razrez(ref a, ref array, ref sluch);

for (int i = 0; i < 6; i++)

for (int j = 0; j < 2; j++)

{

S[i, j] = new Button();

S[i, j].Text = "";

S[i, j].Enabled = true;

S[i, j].Click += new EventHandler(this.but\_Click);

S[i, j].Location = new Point(120+i\*40, j \* 40 + 390);

S[i, j].Size = new Size(40, 40);

S[i, j].Text = array[p].ToString();

S[i, j].TabStop = false;

p++;

this.Controls.Add(S[i, j]);

}

}

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

{

Button button = (Button)sender;

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

{

if (C[i].Text == "")

{

C[i].Text = button.Text;

button.Enabled = false;

break;

}

}

for (vr = 0; vr < a.Length; vr++)

otv += C[vr].Text;

if (otv == a)

{

label2.Visible = true;

if (Array.IndexOf(otvety, a) == otvety.Length - 1)

{

pictureBox5.Visible = true;

}

else

button1.Visible = true;

}

else

{

vr = 0;

otv = "";

}

}

public void InitC(ref string a,ref string b)

{

for (int j = 0; j < a.Length; j++)

{

C[j] = new Button();

if (b.Length>0)

{

if (a.Length > b.Length)

{

for (int i = b.Length; i < a.Length; i++)

C[i].Visible = false;

}

a = b;

}

Array.Resize(ref C, a.Length);

C[j].Text = "";

C[j].Click += new System.EventHandler(this.buton\_Click);

C[j].Location = new Point((480 - a.Length \* 40) / 2 + j \* 40, 290);

C[j].Size = new Size(40, 40);

C[j].TabStop = false;

this.Controls.Add(C[j]);

}

}

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

{

Button button = (Button)sender;

for (int i = 0; i < 6; i++)

{

for (int j = 0; j < 2; j++)

{

if (button.Text == S[i, j].Text && S[i, j].Enabled == false)

{

S[i, j].Enabled = true;

button.Text = null;

}

}

}

}

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

{

ur++;

otv = "";

int kar = ur - 1;

{

pictureBox1.Image = Image.FromFile(kar+"\_0.jpg");

pictureBox2.Image = Image.FromFile(kar+"\_1.jpg");

pictureBox3.Image = Image.FromFile(kar+"\_2.jpg");

pictureBox4.Image = Image.FromFile(kar+"\_3.jpg");

}

label3.Text = ur.ToString();

b = otvety[ur - 2];

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

C[i].Visible = false;

p = 0;

for (int i = 0; i < 6; i++)

for (int j = 0; j < 2; j++)

S[i, j].Visible = false;

InitC(ref a, ref b);

InitB(ref a);

if (Array.IndexOf(otvety, a) == otvety.Length)

pictureBox5.Visible = true;

button1.Visible = false;

label2.Visible = false;

ochki += 2;

button2.Text = "подсказок"+ochki.ToString();

}

private void button2\_Click(object sender, EventArgs e)

{

try

{

int xz = Convert.ToInt32(textBox1.Text) - 1;

if (ochki > 0)

{

string temp = C[xz].Text;

C[xz].Text = Convert.ToString(a[xz]);

for (int i = 0; i < 6; i++)

for (int j = 0; j < 2; j++)

{

if (S[i, j].Text == C[xz].Text)

{

S[i, j].Enabled = false;

j = 2;

i = 6;

}

}

if (C[xz].Text != "")

{

for (int i = 0; i < 6; i++)

for (int j = 0; j < 2; j++)

{

if (S[i, j].Text == temp && S[i,j].Enabled==false)

{

S[i, j].Enabled = true;

j = 2;

i = 6;

}

}

C[xz].Text = Convert.ToString(a[xz]);

}

otv = "";

for (int k = 0; k < a.Length; k++)

otv += C[k].Text;

if (otv == a)

{

button1.Visible = true;

label2.Visible = true;

}

ochki--;

button2.Text = "подсказок"+ochki.ToString();

}

}

catch

{

MessageBox.Show("укажите корректный номер буквы");

}

}

}

}