Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования

ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Институт информационных технологий и анализа данных Центр программной инженерии

ОТЧЕТ

к лабораторной работе №2 по дисциплине:

|  |
| --- |
| «Технология программирования» |
| Graphical User Interface |

наименование темы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Выполнил студент группы: | ИСТб-20-2 |  |  |  | Теленьков М.А. |
|  | шифр группы |  | подпись |  | Фамилия ИО |
| Проверил: | доцент |  |  |  | Бахвалова З.А. |
|  | должность |  | подпись |  | Фамилия ИО |

Иркутск. 2021 г.

Оглавление

[Лабораторное задание №2 3](#_Toc84347131)

[Общая постановка задачи 3](#_Toc84347132)

[Задача на for 3](#_Toc84347133)

[Постановка задачи 3](#_Toc84347134)

[Словесное описание алгоритма 3](#_Toc84347135)

[Таблица спецификаций 4](#_Toc84347136)

[Таблица тестов 4](#_Toc84347137)

[Код программы 4](#_Toc84347138)

[Результаты тестов 7](#_Toc84347139)

[Задача на строки 7](#_Toc84347140)

[Постановка задачи 7](#_Toc84347141)

[Словесное описание алгоритма 8](#_Toc84347142)

[Таблица спецификаций 8](#_Toc84347143)

[Таблица тестов 8](#_Toc84347144)

[Код программы 8](#_Toc84347145)

[Результаты тестов 11](#_Toc84347146)

**Выполнил:** Теленьков Максим Алексеевич, группа ИСТб-20-2

# Лабораторное задание №2

### Общая постановка задачи

Создать пользовательский интерфейс для своей первой лабораторной.

* Взять любые две свои задачи из предыдущей лабы, предпочтительно там где больше входных данных и сделать два проекта
* Каждый проект разместить на github аккаунте
* В интерфейсе должны присутствовать поля для ввода данных.
* А также кнопка, по нажатию на которую выскочит сообщение с результатом либо выведется результат на форму
* При закрытии формы сохранять значения введенные на форме, а при открытии восстанавливать

## Задача на for

Ссылка на репозиторий: https://github.com/ValletOr/Lab2-For14.git

### Постановка задачи

**Задача №14. (Красный)** Дана последовательность натуральных чисел. Определить, есть ли в последовательности хотя бы одна n-ка одинаковых “соседних” чисел (n и элементы последовательности вводятся с клавиатуры). В случае положительного ответа определить порядковые номера чисел первой из таких пар.

### Словесное описание алгоритма

**Form1.Button1\_click()**

1. Вывести формулировку задания

**Form1.Button2\_click()**

1. Сохранение введённых данных
2. Перевод строки в массив чисел
3. Проверка на исключения
4. Вызов и вывод Logic.Execution

**Logic.Execution**

1. Заводим переменные под выходное сообщение outMessage, счётчик counter и trigger.
2. Проходим по вссем элементам списка:
   1. Если нынешний элемент равен следующему элементу, то
      1. Счётчик +=1
      2. Если Счётчик равен n, то
         1. trigger = true
         2. outMessage = "Номера первой найденной n-ки:" + номера найденной n-ки
   2. Иначе Счётчик = 0
3. Если trigger = false, то
   1. outMessage = "n пары не найдены"
4. Вернуть outMessage.

### Таблица спецификаций

|  |  |  |
| --- | --- | --- |
| Имя | Назначение | Тип/Диапазон |
| Входные величины | | |
| nTextBox | Поле для ввода от пользователя n-пары | Текст |
| strTextBox | Поле для ввода от пользователя последовательности чисел | Текст |
| Выходные величины | | |
| outMessage | Поле с результатом | Текст |

### Таблица тестов

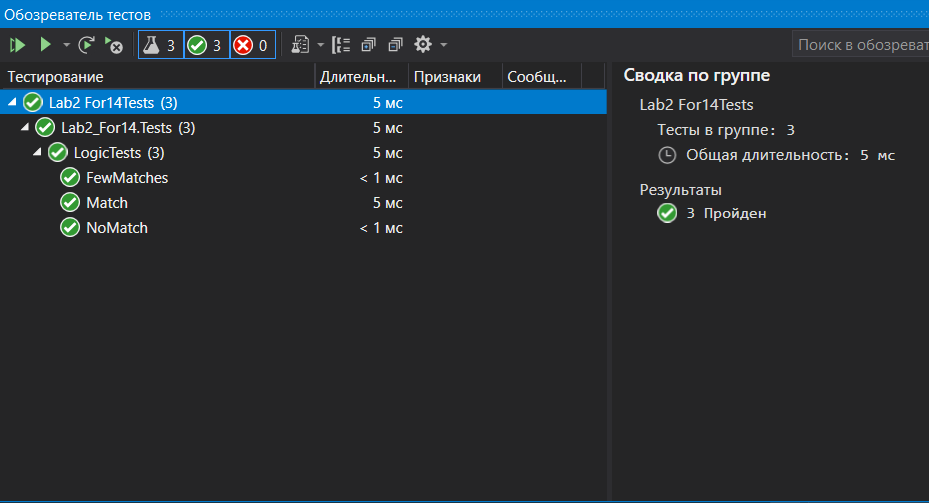
| Номер теста | Что проверяем | Входные данные | Выходные данные |
| --- | --- | --- | --- |
| 1 | Нет совпадений | n = 4  var numbers = new int[] = {1, 2, 2, 3, 3, 3, 2, 2} | n пары не найдены |
| 2 | Есть одно совпадение | n = 3  int[] = {1, 2, 2, 3, 3, 3, 2, 2} | Номера первой найденной n-ки:4 5 6 |
| 3 | Есть несколько совпадений | n = 2  int[] = {1, 2, 2, 3, 3, 3, 2, 2} | Номера первой найденной n-ки:2 3 |

### Код программы

|  |
| --- |
| 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 Lab2\_For14  {  public partial class Form1 : Form  {  public Form1()  {  InitializeComponent();  //Восстанавливаем последние введённые данные  strTextBox.Text = Properties.Settings.Default.strTextBox.ToString();  nTextBox.Text = Properties.Settings.Default.nTextBox.ToString();  }  private void button1\_Click(object sender, EventArgs e)  {  MessageBox.Show("Дана последовательность натуральных чисел. Определить, есть ли в последовательности хотя бы одна n-ка одинаковых “соседних” чисел (n и элементы последовательности вводятся с клавиатуры). В случае положительного ответа определить порядковые номера чисел первой из таких пар.", "Формулировка задания");  }  private void button2\_Click(object sender, EventArgs e)  {  string[] strMass = strTextBox.Text.Split(" ", StringSplitOptions.RemoveEmptyEntries);  int[] numbers = new int[strMass.Length];  int n = 0;  bool excTrigger = false;  //Сохраняем последние введённые данные  Properties.Settings.Default.strTextBox = strTextBox.Text;  Properties.Settings.Default.nTextBox = nTextBox.Text;  Properties.Settings.Default.Save();  //Переводим массив строк в массив чисел  try  {  for (int i = 0; i < numbers.Length; i++)  {  numbers[i] = int.Parse(strMass[i]);  }  }  catch (Exception)  {  MessageBox.Show("Ошибка при конвертации строки в числа.", "Ахтунг!", MessageBoxButtons.OK, MessageBoxIcon.Error);  excTrigger = true;  }  //Переводим n текстовое в n числовое  try  {  n = int.Parse(nTextBox.Text);  }  catch (Exception)  {  MessageBox.Show("Ошибка при конвертации строки n в число.", "Ахтунг!", MessageBoxButtons.OK, MessageBoxIcon.Error);  excTrigger = true;  }  //Вызов логики  if (!excTrigger)  {  resultTextBox.Text = Logic.Execution(numbers, n);  strTextBox.Text = "";  nTextBox.Text = "";  }  }  private void strTextBox\_KeyPress(object sender, KeyPressEventArgs e)  {  if (e.KeyChar ==(char)Keys.Enter)  {  nTextBox.Focus();  }  }  private void nTextBox\_KeyPress(object sender, KeyPressEventArgs e)  {  if (e.KeyChar == (char)Keys.Enter)  {  button2.PerformClick();  strTextBox.Focus();  }  }  }  public static class Logic  {  //Основная логика  public static string Execution(int[] numbers, int n)  {  bool trigger = false;  int counter = 0, i = 0;  string outMessage = "";  while ((trigger != true) && (i < numbers.Length-1))  {  if (numbers[i] == numbers[i + 1])  {  counter += 1;  if (counter == n - 1)  {  trigger = true;  outMessage = "Номера первой найденной " + n + "-ки: ";  for (int j = (i + 1) - (n - 1); j <= i + 1; j++)  {  outMessage += j+1 + " ";  }  }  }  else  {  counter = 0;  }  i++;  }  if (!trigger)  {  outMessage = "Пары не найдены";  }  return outMessage;  }  }  } |

### Результаты тестов





## Задача на строки

Ссылка на репозиторий: https://github.com/ValletOr/Lab2-Str11.git

### Постановка задачи

**Задача №11. (Красный)** Даны два слова. Для каждой буквы первого слова определить, входит ли она во второе слово. Повторяющиеся буквы первого слова не рассматривать. Например, если заданные слова процессор и информация, то для букв первого из них ответом должно быть: нет да да да нет нет.

### Словесное описание алгоритма

**Form1.button1\_click()**

1. Сохранение введённых значений
2. Проверка первого слова на пустоту
   1. Если пустое: throw exc
3. Вызов и вывод Logic.Execution

**Logic.Execution**

1. Заводим переменные под выходное сообщение outMessage, trigger и буферную строку buffer.
2. Чистим первое слово от повторяющихся символов:
   1. Поочерёдно переносим символы из первого слова во второе
   2. Если символ уже есть в буфере, то не заносим его.
   3. word1 = buffer
3. Двигаемся по буквам первого слова, сравниваем его с буквами второго слова
   1. Если буква первого слова есть во втором слове out message += “да”, иначе outMessage += “нет”.
   2. outMessage += “ ”.
4. Возвращаем outMessage

### Таблица спецификаций

|  |  |  |
| --- | --- | --- |
| Имя | Назначение | Тип/Диапазон |
| Входные величины | | |
| word1Box | Поле для ввода первого слова | Текст |
| word2Box | Поле для ввода второго слова | Текст |
| Выходные величины | | |
| resultBox | Поле для вывода реузльтата | Текст |

### Таблица тестов

| Номер теста | Что проверяем | Входные данные | Выходные данные |
| --- | --- | --- | --- |
| 1 | Нет совпадений и повторов букв | word1 = “Процессор”  word 2 = “Синхрофазотрон” | п – нет  р – да  о – да  ц – нет  е – нет  с - да |
| 2 | Есть совпадения и есть повтор букв | word1 = “Телега”  word2 = “Яд” | т – нет  е – нет  л – нет  г – нет  а - нет |

### Код программы

|  |
| --- |
| 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 Lab2\_Str11  {  public partial class Form1 : Form  {  public Form1()  {  InitializeComponent();  //Вводим значения с прошлого запуска  word1Box.Text = Properties.Settings.Default.word1Box;  word2Box.Text = Properties.Settings.Default.word2Box;  }  private void button1\_Click(object sender, EventArgs e)  {  //Сохранение введённых значений.  Properties.Settings.Default.word1Box = word1Box.Text;  Properties.Settings.Default.word2Box = word2Box.Text;  Properties.Settings.Default.Save();  //Проверка на пустоту в первом слове.  if (string.IsNullOrEmpty(word1Box.Text))  {  MessageBox.Show("Первое слово не введено, невозможно вывести результат.", "Результат", MessageBoxButtons.OK, MessageBoxIcon.Warning);  }  else  {  resultBox.Text = Logic.Execution(word1Box.Text, word2Box.Text);  word1Box.Text = "";  word2Box.Text = "";  }  }  private void word1Box\_KeyPress(object sender, KeyPressEventArgs e)  {  if (e.KeyChar == (char)Keys.Enter)  {  word2Box.Focus();  }  }  private void word2Box\_KeyPress(object sender, KeyPressEventArgs e)  {  if (e.KeyChar == (char)Keys.Enter)  {  button1.PerformClick();  word1Box.Focus();  }  }  private void button2\_Click(object sender, EventArgs e)  {  MessageBox.Show("Даны два слова. Для каждой буквы первого слова определить, входит ли она во второе слово. Повторяющиеся буквы первого слова не рассматривать. Например, если заданные слова процессор и информация, то для букв первого из них ответом должно быть: нет да да да нет нет.");  }  }  public static class Logic  {  public static string Execution(string word1, string word2)  {  string outMessage = "", buffer = "";  bool trigger = false;  int j;  //Приведение букв в словах к одному регистру (нижнему)  word1 = word1.ToLower();  word2 = word2.ToLower();  //Чистка первого слова от повторяющихся букв  for (int i = 0; i < word1.Length; i++)  {  if (!buffer.Contains(word1[i]))  {  buffer += word1[i];  }  }  word1 = buffer;  //Основной расчёт  for (int i = 0; i < word1.Length; i++)  {  trigger = false;  j = 0;  outMessage += word1[i]+" - ";  while ((trigger != true) && (j < word2.Length))  {  if (word1[i] == word2[j])  {  trigger = true;  outMessage += "да";  }  j++;  }  if (!trigger)  {  outMessage += "нет";  }  if (i != word1.Length - 1)  {  outMessage += "\r\n";  }  }  //Возврат значения.  return outMessage;  }  }  } |

### Результаты тестов

