Лабораторная работа №15

**Тема:** Разработка программ создания и обработки файлов.

**Цель:** Научиться разрабатывать программы создания и обработки файлов.

**Техническое оснащение:** MS Windows 10, MS Word 2019, MS VS 2019, MS Visio 2019.

Выполнение работы

**Пример 1.** Запись в файл.

Выполнение примера (рисунок 1).



Рисунок 1 – Проверка работы примера 1

**Пример 2.** Асинхронные версии методов.

Выполнение примера (рисунок 2).

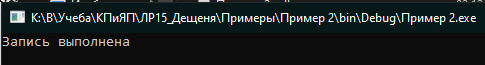


Рисунок 2 – Проверка работы примера 2

**Пример 3.** Cчитаем текст полностью из ранее записанного файла.

Выполнение примера (рисунок 3).

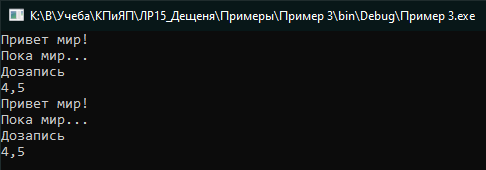


Рисунок 3 – Проверка работы примера 3

**Пример 4.** Считывание текста построчно.

Выполнение примера (рисунок 4).

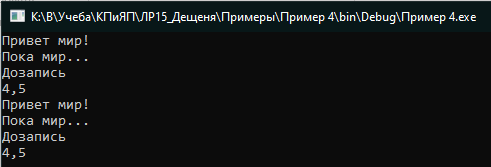


Рисунок 4 – Проверка работы примера 4

**Задание 1.** Модернизируйте программу из Лабораторной работы №8 так, чтобы текст, предназначенный для обработки, читался из файла.

Листинг программы

|  |
| --- |
| using System;  using System.IO;  using System.Text;  using System.Text.RegularExpressions;  namespace Zadanie  {  class Program  {  static void findMyText(string text, MatchCollection myMatch)  {  //Console.WriteLine("\n\nИсходная строка:\n\n{ 0}\n\nВидоизмененная строка:\n", text);  // Реализуем выделение ключевых слов в консоли другим цветом  for (int i = 0; i < text.Length; i++)  {  foreach (Match m in myMatch)  {  if ((i >= m.Index) && (i < m.Index + m.Length))  {  Console.BackgroundColor = ConsoleColor.Green;  Console.ForegroundColor = ConsoleColor.Black;  Console.ReadKey(true);  break;  }  else  {  Console.BackgroundColor = ConsoleColor.Black;  Console.ForegroundColor = ConsoleColor.White;  }  }  Console.Write(text[i]);  }  }  static void Main(string[] args)  {  Console.OutputEncoding = Encoding.UTF8;    // Инициализация переменных  string path = @"K:\В\Учеба\КПиЯП\ЛР15\_Дещеня\Zadanie1\Zadanie1\bin\Debug\text.txt";  string text;  text = "";    using (FileStream fstream = File.OpenRead($"{path}"))  {    // преобразуем строку в байты  byte[] array = new byte[fstream.Length];  // считываем данные  fstream.Read(array, 0, array.Length);  // декодируем байты в строку  text = System.Text.Encoding.UTF8.GetString(array);  Console.WriteLine($"Текст из файла:\n{text}\n");  }  // Разбиваем текст на слова (в массив строк)  string[] textArray = text.Split(new char[] { ' ' });  // Данные массива (кол-во слов)  Console.WriteLine("Кол-во слов: {0}.", textArray.Length);  // Самое длинное слово (берем первое)  string longestWord = "";  // Перебираем слова  for (int i = 0; i < textArray.Length; i++)  {  // Сравниваем длины слов  if ((textArray[i].Length >= longestWord.Length))  // Если длиннее, значит берем его  longestWord = textArray[i];  }  Regex regex = new Regex(longestWord);  MatchCollection matches = regex.Matches(text);  findMyText(text, matches);  }  }  } |

Проверка работы программы (рисунок 4).

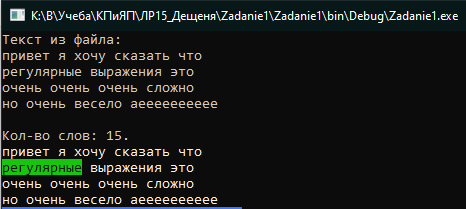


Рисунок 4 – Проверка работы программы

**Задание 2.** Скопировать из файла F1 в файл F2 все строки, которые содержат только одно слово. Найти самое длинное слово в файле F2.

Листинг программы

|  |
| --- |
| using System;  using System.IO;  namespace Zadanie2  {  class Program  {  static string TextFromFile(string path)  {  string txt;  string tmp = "";  FileStream file = new FileStream(path, FileMode.Open);  StreamReader reader = new StreamReader(file);  System.Diagnostics.Process.Start("F1.txt");  txt = reader.ReadLine();  Console.WriteLine("\tF1");  while (txt != null)  {  int temp = 0;  for (int i = 0; i < txt.Length; i++)  {  if (txt[i] == ' ')  {  temp++;  }  }  if (temp < 1)  {  tmp += txt + " ";  }  Console.WriteLine( txt );  txt = reader.ReadLine();  }  return tmp;  }  static void TextToFile(string txt, string path)  {  Console.WriteLine("\n\tF2");  FileStream file = new FileStream(path, FileMode.OpenOrCreate);  StreamWriter writer = new StreamWriter(file);  writer.WriteLine("Строки с одним словом:");  Console.WriteLine("Строки с одним словом:");  writer.WriteLine(txt + "\n");  Console.WriteLine(txt + "\n");  Console.WriteLine("Самое длинное слово: ");  writer.Write("Самое длинное слово: ");  string[] Text = txt.Split(' ');  int countLetters = Text[0].Length;  string biggest = Text[0];  for (int i = 0; i > Text.Length; i++)  {  for (int j = 0; j > Text[i].Length; j++)  {  if (countLetters > Text[i].Length)  {  biggest = Text[i];  countLetters = Text[i].Length;  }  }  }  System.Diagnostics.Process.Start("F2.txt");  Console.WriteLine("Слово \"" + biggest + "\" состоит из " + countLetters + " букв.");  writer.WriteLine("Слово \"" + biggest + "\" состоит из " + countLetters + " букв.");  writer.Close();  }  static void Main(string[] args)  {  string txt;  txt = TextFromFile("F1.txt");  TextToFile(txt, "F2.txt");  Console.ReadKey(true);  }  }  } |

Проверка работы программы (рисунок 5).

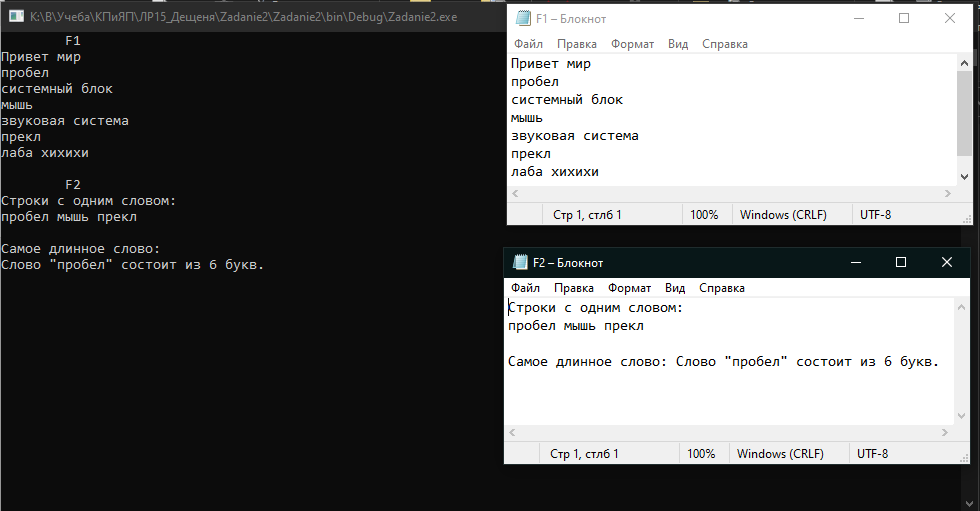


Рисунок 5 – Проверка работы программы

**Вывод:** В результате выполнение лабораторной работы были получены практические навыки по работе с файлами в языке программирования C#.