

Кафедра компьютерной инженерии и моделирования

Шенгелай Всеволод Михайлович

отчет по лабораторной работе №8  
по дисциплине «**ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ  
ПРОГРАММИРОВАНИЕ**»

Направление подготовки:  
09.03.04 "Программная инженерия"

Оценка - 92



Симферополь, 2021

## **Лабораторная работа №8**

**Тема:** Практическое программирование.

**Цель работы:** Научиться простым приемам практического программирования, запуску чужих процессов из-под C#, использовать различные кодировки текста, "парсить" текст, работать с файлами ввода-вывода, создавать с помощью индексаторов массивы со свойствами (умный массив).

**Перед выполнением лабораторной работы изучена следующая литература:**

1. Презентация лектора курса.
2. Эндрю Троелсен. Язык программирования C# и платформы .NET и .NET CORE. Часть III. Главы 5-6.
3. Сайт MSDN Microsoft.
4. Сайт Metanit.com

**Выполнено 1 задание, описанное в методических указания к выполнению лабораторных работ.**

### **Задание 1:**

В этом задании мы создаём приложение WPF, с помощью которого мы можем запустить процесс Wordpad.exe. Как только пользователь введёт текст, включающий цифры, буквы, точку, как разделитель в числах и знаки минус, и, затем сохранит документ, он может открыть любой txt файл. Написанный нами метод StrRemLet заменяет в строках буквы на пробелы. При наличии двух подряд идущих знаков минус или разделителей, он заменяет их, оставляя один минус или разделитель (соответственно). Буквенные строки сразу же сохраняются в файле txt с кодировкой Windows 1251. Цифровые строки делятся на слова и записываются в «умный» одномерный массив. Затем, с помощью метода TryParse они преобразуются в числа и сохраняются в резиновом массиве ArrayList. При этом у пользователя есть возможность добавлять числа в массив. Числовые данные могут быть сохранены в файл txt.

ссылка: 1

```
private void openNotepadButt_Click(object sender, RoutedEventArgs e)
{
    //Запуск нового процесса
    Process newProc = Process.Start("notepad.exe");
    newProc.WaitForExit();
    newProc.Close(); // освободить выделенные ресурсы
}
```

Рисунок 1. Запуск процесса notepad.exe

ссылка: 1

```
class SmartArray
{
    public double[] arr;
    private double maxIndex;

    Ссылка: 0
    public SmartArray(int maxIndex_arg)
    {
        maxIndex = maxIndex_arg;
        arr = new double[maxIndex_arg];
    }

    Ссылка: 0
    public double this[int NumOfElement]// индексатор
    {
        get
        {
            if (NumOfElement >= maxIndex)
            {
                throw new IndexOutOfRangeException();// исключение
            }
            else
                return arr[NumOfElement];
        }
        set
        {
            if (NumOfElement >= maxIndex)
            {
                throw new IndexOutOfRangeException();
            }
            else
                arr[NumOfElement] = value;
        }
    }
}
```

Рисунок 2. Реализация «умного массива»

```

ССылка: 1
private void downloadFile_Click(object sender, RoutedEventArgs e)
{
    OpenFileDialog openFileDialog = new OpenFileDialog();
    openFileDialog.Filter = "Text files (*.txt)|*.txt|All files (*.*)|*.*";
    openFileDialog.InitialDirectory = @"C:\Users\Vsevolod\source\repos\lab 8\lab 8";
    //Возвращает true, false, или null
    if (openFileDialog.ShowDialog() == true)
    {
        //Выгружаем всё в одну строку-переменную, чтобы найти длину массива
        var linescount = File.ReadAllLines(openFileDialog.FileName);

        string[] str_arr = new string[linescount.Length];
        //Вариант с явным указанием кодировки, но и так работает
        //StreamReader f = new StreamReader(openFileDialog.FileName, Encoding.GetEncoding("utf-32"));
        StreamReader f = new StreamReader(openFileDialog.FileName);
        int arr_counter = 0;
    }
}

```

Рисунок 3. Загрузка строк из файла

**Представлен один проект, реализованный в Visual Studio Common Eddition 2019.** Проект представлен преподавателю в электронной форме, продемонстрирована его работоспособность, разъяснены детали программного кода.