**Казанский Приволжский Федеральный Университет**

**Институт Вычислительной Математики и Информационных Технологий**

**Представление результатов работы разработки программного обеспечения**

**«Система Repeat - Until»**

Выполнил студент 2 курса группы 09-551

Кафедры технологий программирования

Латыпов Булат Салаваович

Казань

2017

**Общие сведения**

Данное программное обеспечение представляет собой систему, которая запускает 3 программы (.exe) одновременно на экране.

**Общее описание**

Для функционирования данного программного обеспечения необходимо:

* Операционная система Microsoft Windows
* .NET Framework 4.5.2
* 15 Мб оперативной памяти, 1 MHz процессора
* Visual C# for Visual Studio

**Функциональное назначение**

Данное программное обеспечение представляет собой систему, которая запускает 3 программы (.exe) одновременно на экране.

**Описание логической структуры**

*Fifth.exe* – исполняемый файл программы

**Использованные технологии и средства**

Программа реализована на языке C#. Для запуска и корректной работы в среде Microsoft должен быть установлен .NET Framework 4.5.2 или более новой версии.

**Вызов и загрузка**

Для запуска программы необходимо запустить файл

ВАША\_ПАПКА\_С\_ЗАДАНИЯМИ\ Fifth \ Fifth \bin\Debug\ Fifth.exe

**Описание продукта**

Данные вбиваются в консоль. Действия осуществляются с помощью одиночного нажатия кнопок и текстовых строк.

**Задание на разработку**

**Постановка задачи:** реализовать систему, которая запускает 3 программы (.exe) одновременно на экране.

**Требования:**

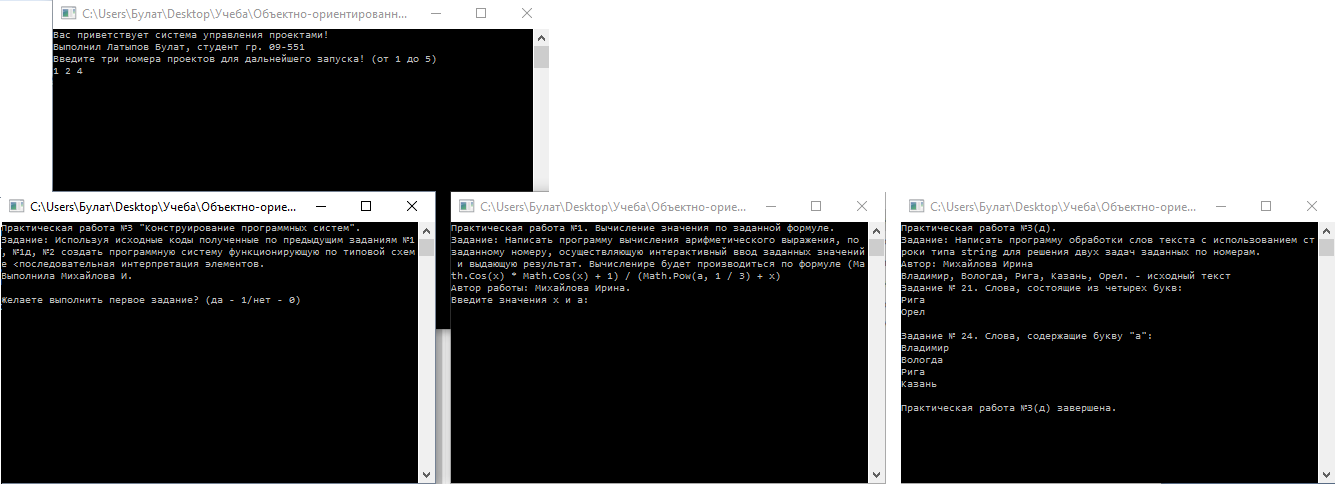
* Программа должна быть совместима с .NET Framework.4.5.2
* Операционная система Windows 7, Windows 8, Windows 10

**Инструментарий:** Visual Studio

**Язык разработки:** C#

**Пример работы**

Результат работы данной программы:



**Листинг исходного кода на С#**

using System;

using System.Threading;

using System.Diagnostics;

using System.Threading.Tasks;

using System.Runtime.InteropServices;

namespace Fifth

{

class Program

{

const string Path = @"..\..\..\..\Exe\";

static void Main(string[] args)

{

Console.WriteLine("Вас приветствует система управления проектами!");

Console.WriteLine("Выполнил Латыпов Булат, студент гр. 09-551");

Console.WriteLine("Введите три номера проектов для дальнейшего запуска! (от 1 до 5)");

string answer = Console.ReadLine();

string[] separate = answer.Split(' ');

if (separate.Length != 3)

{

Console.WriteLine("Неверное количество параметров!");

SystemClosing();

return;

}

Task t = ParallelRunningAsync(separate);

t.Wait();

WaitTheLastProcessClosing(separate);

SystemClosing();

}

static async Task ParallelRunningAsync(string[] separate)

{

int x = 0, y = 245;

Task[] tasks = new Task[3];

for (int i = 0; i < 3; i++)

{

string fileName = Path + separate[i] + ".exe";

tasks[i] = RunAsync(fileName, x, y);

x += 450;

}

await Task.WhenAll(tasks);

}

static Task RunAsync(string fileName, int x, int y)

{

return Task.Run(() =>

{

Process p = new Process();

p.StartInfo.FileName = fileName;

p.Start();

Thread.Sleep(150);

Program.MoveWindow(p.MainWindowHandle, x, y, 450, 300, true);

});

}

[DllImport("user32.dll", SetLastError = true)]

internal static extern bool MoveWindow(IntPtr hWnd, int X, int Y, int nWidth, int nHeight, bool bRepaint);

static void WaitTheLastProcessClosing(string[] separate)

{

foreach (Process p in Process.GetProcessesByName(separate[2]))

{

p.WaitForExit();

break;

}

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

{

foreach (Process p in Process.GetProcessesByName(separate[i]))

p.Kill();

}

}

static void SystemClosing()

{

Console.WriteLine("Завершение работы системы...");

Thread.Sleep(1000);

}

}

}