Министерство образования и науки Российской Федерации

федеральное государственное автономное образовательное учреждение

высшего профессионального образования

**«Северный (Арктический) федеральный университет имени М.В. Ломоносова»**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | | | | | | | | | | | | |  |
|  | Информационных систем и технологий | | | | | | | | | | | | |  |
|  | (наименование кафедры) | | | | | | | | | | | | |  |
|  |  | | | | | | | | | | | | |  |
|  | Данилов Алексей Александрович | | | | | | | | | | | | |  |
|  | (фамилия, имя, отчество студента) | | | | | | | | | | | | |  |
|  |  | | | | | | | | | | | | |  |
|  | Институт | ИМИКТ | | | курс | 2 | | | группа | 28 | | |  |  |
|  |  | | | | | | | | | | | | |  |
|  | |  | | --- | | 230400.62 Информационные системы и технологии | | (код и наименование направления подготовки/специальности) | | | | | | | | | | | | | |  |
|  | **ОТЧЁТ** | | | | | | | | | | | | |  |
|  | о лабораторном практикуме | | | | | | | | | | | | |  |
|  |  | | | | | | | | | | | | |  |
|  | По дисциплине | |  | Технология программирования | | | | | | | | | |  |
|  |  | |  |  | | |  |  | | |  |  | |  |
|  |  | |  |  | | |  |  | | |  |  | |  |
|  |  | |  |  | | |  |  | | |  |  | |  |
|  |  | |  |  | | |  |  | | |  |  | |  |
|  |  | |  |  | | |  |  | | |  |  | |  |
|  | Отметка о зачёте | |  |  | | | | | | |  |  | |  |
|  |  | |  |  | | |  |  | | |  | (дата) | |  |
|  |  | |  |  | | |  |  | | |  |  | |  |
|  | Руководитель практикума | | | | | |  |  | | |  |  | |  |
|  |  | |  |  | | | | | | |  |  | |  |
|  | Доцент | |  |  | | | | | | |  | И. С. Майоров | |  |
|  | (должность) | |  | (подпись) | | | | | | |  | (инициалы, фамилия) | |  |
|  |  | |  |  | | |  |  | | |  |  | |  |
|  |  | |  |  | | |  |  | | |  |  | |  |
|  |  | |  |  | | |  |  | | |  |  | |  |
|  |  | |  |  | | |  |  | | |  |  | |  |
|  |  | |  |  | | |  |  | | |  |  | |  |
|  |  | |  |  | | |  |  | | |  |  | |  |
|  |  | | | | | | | | | | | | |  |
|  | Архангельск  2014 | | | | | | | | | | | | |  |

ЛИСТ ДЛЯ ЗАМЕЧАНИЙ

1. Задание

Вычислить значение функции f(x), для введенного пользователем x:

Модифицировать программу для вычисления следующей функции:

1. Планируемая реализация

Используя делегаты и события реализовать расчёт функций для соответствующих x. Реализовать корректный ввод переменной x.

1. Итоговая реализация

Для организации вызова методов расчёта функций используется делегат, объявленный на уровне пространства имён DAA.TP.lab6.

Фрагмент программного кода

public delegate void SystemEventHandler(double x);

Имя этого делегата означает, что этот делегат − обработчик события. Для хранения и работы с событием был создан класс Systems.

Фрагмент программного кода

class Systems

{

public event SystemEventHandler CountSystem;

public void CountingSystem(double x)

{

if (CountSystem != null){

CountSystem(x);

}

}

}

Тип этого события указывает на обработчика события. Другие методы могут подписываться на событие CountSystem. В методе CountingSystem() проверяется, подписались ли какие-либо методы на событие CountSystem, и если такие имеются, то происходит событие CountSystem().

Методы подписываются на это событие в методе Main().

Фрагмент программного кода

static void Main()

{

double x = GetValue("Введите x:");

Systems syst = new Systems();

syst.CountSystem += FirstSystemHandler;

syst.CountSystem += SecondSystemHandler;

syst.CountingSystem(x);

Console.ReadKey();

}

Сначала осуществляется корректный ввод переменной x (о нём ниже), затем создаётся экземпляр класса Systems. На событие подписываются методы FirstSystemHandler() и SecondSystemHandler(). В программе методы подписываются на событие неявно. Явно метод подписывается следующим образом.

Фрагмент программного кода

syst.CountSystem += new SystemEventHandler(FirstSystemHandler);

Оператор +=, обычно использующийся при вычислениях, здесь скрывает механизм подписки обработчика на событие. По факту при подписывании на событие вызывается метод add у этого события.

Событие происходит при вызове метода CountingSystem(), вызывается его обработчик − делегат SystemEventHandler() и предаёт выполнение сначала методу FirstSystemHandler(), а затем методу SecondSystemHandler().

Метод FirstSystemHandler() осуществляет расчёт значения первой системы для полученного x, SecondSystemHandler() − второй системы для того же x. Метод GetValue() осуществляет корректный ввод переменной.

Фрагмент программного кода

static void FirstSystemHandler(double x)

{

if (x >= 0){

Console.WriteLine("Результат первой системы для x >= 0: {0}", Math.Cos(4 \* x));

}

else{

double func = Math.Pow(Math.Pow(x, 8) - Math.Pow(x, 7) + 7, 9);

if (double.IsInfinity(func)){

Console.WriteLine("Результат расчёта первой системы для x < 0 - бесконечно большая функция");

}

else{

Console.WriteLine("Результат расчёта первой системы для x < 0: {0}", func);

}

}

}

static void SecondSystemHandler(double x)

{

if (x >= 4){

double func = Math.Sqrt(Math.Pow(x, 2) - 16);

if (double.IsInfinity(func)){

Console.WriteLine("Результат расчёта второй системы для x >= 4 - бесконечно малая функция");

}

else{

Console.WriteLine("Результат расчёта второй системы для x >= 4: {0}", func);

}

}

else{

Console.WriteLine("Результат расчёта второй системы для x < 4: {0}", Math.Sin(7 \* x + 2));

}

}

private static double GetValue(string msg)

{

double x;

do{

Console.Write(msg);

string str = Console.ReadLine();

if (double.TryParse(str, out x)){

return double.Parse(str);

}

Console.WriteLine("Входная строка имеет не верный формат.");

}

while (true);

}

ЗАКЛЮЧЕНИЕ

В рамках данной лабораторной работы была реализована программа, осуществляющая расчет систем уравнений, с использованием делегатов.