**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук  
Департамент программной инженерии

|  |  |
| --- | --- |
| СОГЛАСОВАНО  Старший преподаватель департамента программной инженерии факультета компьютерных наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ С. А. Шершаков «\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2019 г. | УТВЕРЖДАЮ  Академический руководитель образовательной программы «Программная инженерия» профессор департамента программной инженерии, канд. техн. наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В. В. Шилов «\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2019 г. |

|  |  |
| --- | --- |
| ***Подп. и дата*** |  |
| ***Инв. № дубл.*** |  |
| ***Взам. инв. №*** |  |
| ***Подп. и дата*** |  |
| ***Инв. № подл*** | RU.17701729.04.01-01 ТЗ 01-1-ЛУ |

**ПРОГРАММА-РАСШИРЕНИЕ MICROSOFT VISIO ДЛЯ ИМПОРТА ГРАФОВ В ФОРМАТЕ DOT**

**Пояснительная записка**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.04.01-01 ТЗ 01-1-ЛУ**

Исполнитель  
студент группы БПИ173  
\_\_\_\_\_\_\_\_\_\_ / Переплетчиков А. И. /  
«\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2019 г.

**Москва 2019**

УТВЕРЖДЕНRU.17701729.04.01-01 ТЗ 01-1-ЛУ

|  |  |
| --- | --- |
| Подп. и дата |  |
| Инв. № дубл. |  |
| Взам. инв. № |  |
| Подп. и дата |  |
| Инв. № подл | RU.17701729.04.01-01 |

**ПРОГРАММА-РАСШИРЕНИЕ MICROSOFT VISIO ДЛЯ ИМПОРТА ГРАФОВ В ФОРМАТЕ DOT**

**Пояснительная записка**

**RU.17701729.04.01-01 ТЗ 01-1**

**Листов 28**

**Москва 2019**

**Содержание**

[**1. Введение** 3](#_Toc451347290)

[**1.1. Наименование программы** 3](#_Toc451347291)

[**1.2. Основания для разработки** 3](#_Toc451347292)

[**2. Назначение и область применения** 4](#_Toc451347293)

[**2.1. Назначение разработки** 4](#_Toc451347294)

[**2.1.1. Функциональное назначение** 4](#_Toc451347295)

[**2.1.2. Эксплуатационное назначение** 4](#_Toc451347296)

[**2.2. Область применения** 4](#_Toc451347297)

[**3. Технические характеристики** 5](#_Toc451347298)

[**3.1. Постановка задачи на разработку программы** 5](#_Toc451347299)

[**3.2. Описание и обоснование алгоритма и функционирования программы** 5](#_Toc451347300)

[**3.2.1. Описание алгоритма и функционирования программы** 5](#_Toc451347301)

[**3.2.2. Обоснование выбора алгоритма** 6](#_Toc451347302)

[**3.3. Описание и обоснование метода организации входных и выходных данных** 7](#_Toc451347303)

[**3.3.1. Описание метода организации входных и выходных данных** 7](#_Toc451347304)

[**3.3.2. Обоснование метода организации входных и выходных данных** 7](#_Toc451347305)

[**3.4. Описание и обоснование выбора состава технических и программных средств** 7](#_Toc451347306)

[**3.4.1. Состав технических и программных средств** 7](#_Toc451347307)

[**3.4.2. Обоснование выбора состава технических и программных средств** 8](#_Toc451347308)

[**4. Технико-экономические показатели** 9](#_Toc451347309)

[**4.1. Предполагаемая потребность** 9](#_Toc451347310)

[**4.2. Ориентировочная экономическая эффективность** 9](#_Toc451347311)

[**4.3. Экономические преимущества разработки по сравнению с отечественными и зарубежными аналогами** 9](#_Toc451347312)

[**Приложение 1. Список используемой литературы** 10](#_Toc451347313)

[**Приложение 2. Описание и функциональное назначение классов** 11](#_Toc451347314)

[**Приложение 3. Описание и функциональное назначение полей, методов и свойств** 12](#_Toc451347315)

# **1. Введение**

## **1.1. Наименование программы**

Наименование программы: «Программа-расширение Microsoft Visio для импорта графов в формате DOT».

## **1.2. Основания для разработки**

Основанием для разработки программы является Приказ НИУ ВШЭ № 6.18.1-02/1112-19 от 11.12.2015 г.

Программа разрабатывается в рамках выполнения курсовой работы по теме «Программа-расширение Microsoft Visio для импорта графов в формате DOT».

# **2. Назначение и область применения**

## **2.1. Назначение разработки**

### **2.1.1. Функциональное назначение**

Программа предназначена для импорта графов в виде файла с расширением .dot и .gv в программу пакета Microsoft Office™ Microsoft Visio и визуализации графов в виде вершин, связанных ребрами. Визуализированный в Microsoft Visio граф должен иметь все атрибуты, указанные в импортируемом файле (цвета, толщина ребер, названия вершин и прочие атрибуты, поддерживаемые языком DOT).

### **2.1.2. Эксплуатационное назначение**

Программа будет использоваться для работы с графами в широко распространённом инструменте Microsoft Visio с возможностью в дальнейшем модификации с учетом имеющегося функционала инструмента, в том числе, в учебных и научных целях.

## **2.2. Область применения**

Программа будет применяться для облегчения работы с графами, записанных на языке описания графов DOT, в программе Microsoft Visio: для импорта содержимого DOT файла на страницу документа Visio в виде схем и корректного экспорта графа-схемы в DOT файл с учетом всех изменений, произведенных пользователем во время работы с документом.

# **3. Технические характеристики**

## **3.1. Постановка задачи на разработку программы**

Программа должна обеспечивать возможность выполнения следующих функций:

1. возможность импорта файла расширения .dot или .gv, в котором содержится информация на языке описания графов DOT;
2. визуализация графа, записанного в импортируемом файле, на отдельной странице документа Visio при помощи стандартных фигур Microsoft Visio с учетом всех атрибутов вершин и ребер, указанных в импортируемом файле;
3. импорт нового DOT файла без перезапуска программы;
4. хранение связанных пар страницы и представленного на ней графа;
5. экспорт содержимого с учетом всех изменений, произведенных пользователем, обратно в .dot или .gv файл.

## **3.2. Описание и обоснование алгоритма и функционирования программы**

### **3.2.1. Описание алгоритма и функционирования программы**

Программа-расширение Microsoft Visio для импорта графов в формате DOT использует набор инструментов Visual Studio Tools for Office (VSTO) для взаимодействия с объектной моделью Microsoft Visio и библиотеку Graphviz4Net для взаимодействия с DOT графами. Сценарий работы программы может отличаться в зависимости от действий пользователя, упрощенно главный алгоритм выглядит следующим образом:

1. При помощи диалогового окна выбора файла пользователь может выбрать файл с расширением .dot или .gv;
2. Основной класс программы ThisAddIn создает новую страницу для активного документа Visio, связывая ее с новым объектом класса VisioGraph, наследующего класс DotGraph, в конструктор которого в строковом формате передается содержимое раннее открытого файла;
3. С помощью анализатора для формальных языков, написанного при помощи инструмента ANTLR (*ANother Tool for Language Recognition*), входящего в библиотеку Graphviz4Net, содержимое файла транслируется в объект класса DotGraph;
4. Запускается метод отображения содержимого импортируемого файла на соответствующей странице документа Visio. Сначала в левом верхнем углу страницы размещаются фигуры, соответствующие вершинам графа, для каждой фигуры устанавливается соответствующие стили, указанные в числе атрибутов вершины. Затем каждому ребру импортируемого графа на странице документа сопоставляется объект типа «соединительная линия (connector)», связывающая соответствующие вершины;
5. После добавления всех вершин и ребер на страницу документа Visio программа запускает алгоритм «лэйаутинга (layout algorithm)» для корректного и понятного отображения получившейся схемы на странице.

Программа поддерживает не только возможность импорта графа из файла на страницу документа, но и экспорта содержимого страницы обратно в файл с учетом всех изменений, произведенных пользователем. Для того, чтобы фиксировать изменения содержимого страницы, используются события, порождаемые при работе пользователя с документом:

1. При возникновении события BeforePageDeleted (перед удалением страницы) программа также удаляет из коллекции соответствующий данной странице объект VisioGraph;
2. При возникновении события BeforeShapeDeleted (перед удалением фигуры) программа проверяет, какой тип фигуры был удален (проверяет наличие данной фигуры в словаре vertices или edges) и в зависимости от этого удаляет в графе либо ребро, либо вершину со всеми смежными ей ребрами;
3. При возникновении события ConnectionsAdded (после добавления новой соединительной линии) программа добавляет соответствующее ребро в объект графа;
4. При возникновении события ConnectionsDeleted (после удаления соединительной линии) программа удаляет из графа соответствующее ребро, связывавшее вершины;

Программа также реагирует на изменение пользователем стилей фигур страницы (изменение цвета, текста и т.д.). При попытке пользователя экспортировать граф в файл открывается диалоговое окно сохранения файла, объект класса DotGraph записывается выбранный файл на языке DOT.

### **3.2.2. Обоснование выбора алгоритма**

Выбор данного алгоритма обоснован функциональными требованиями, представленными в техническом задании проекта – весь заявленной функционал реализован в полном объеме. Набор инструментов VSTO использован, т.к. он является наиболее распространенным и простым средством разработки расширений для программ пакета Microsoft Office. Библиотека Graphviz4Net была выбрана среди множества других библиотек для работы с графами, т.к. она включает в себя ANTLR парсер для языка DOT, возможность чтения и записи в DOT файл, а также поставляется с открытым исходным кодом и доступна для возможных изменений.

## **3.3. Описание и обоснование метода организации входных и выходных данных**

### **3.3.1. Описание метода организации входных и выходных данных**

#### **3.3.1.1. Описание метода организации входных данных**

Программа позволяет вводить входные данные (данные о дифференциальном уравнении для решения, границах области, на которой дифференциальное уравнение решается, и граничных условий, параметрах нейронной сети и ее обучения) через текстовые поля или выпадающие списки окна Windows Forms.

#### **3.3.1.2. Описание метода организации выходных данных**

Программа выводит результат решения дифференциального уравнения в виде цветового графика искомой функции на заданной области (с возможностью посмотреть значение искомой функции в каждой из отображаемых точек заданной области) и информации о максимальной ошибке нейронной сети после ее обучения.

### **3.3.2. Обоснование метода организации входных и выходных данных**

#### **3.3.2.1. Обоснование метода организации входных данных**

Описанный в п. 3.3.1.1 настоящего документа метод организации входных данных выбран, потому что ввод входных данных через текстовые поля или выпадающие списки окна Windows Forms является оптимальным и удобным для пользователя.

#### **3.3.2.2. Обоснование метода организации выходных данных**

Обоснование выбора описанного в п. 3.3.1.2 настоящего документа метода организации выходных данных приведено в п. 3.2.2 настоящего документа при обосновании выбора алгоритма построения цветового графика искомой функции.

## **3.4. Описание и обоснование выбора состава технических и программных средств**

### **3.4.1. Состав технических и программных средств**

#### **3.4.1.1. Состав технических средств**

Для нормального функционирования программы требуется компьютер, оснащенный следующими техническими компонентами:

1. процессор не ниже Intel Pentium/Celeron, AMD K6/Athlon/Duron или совместимый с ними с тактовой частотой не ниже 1 ГГц;
2. 512 Мб ОЗУ или более;
3. жесткий диск с объемом свободной памяти не менее 1,5 Гб;
4. VGA-совместимые видеоадаптер и монитор с разрешением не ниже 1280х800;
5. клавиатура и мышь.

#### **3.4.1.2. Состав программных средств**

Для нормального функционирования программы требуется компьютер, оснащенный следующими программными компонентами:

1. операционная система Microsoft Windows XP (SP2, SP3) / Vista / 7 / 8 / 8.1 / 10;
2. библиотека Microsoft .NET Framework 4.5 и выше.

### **3.4.2. Обоснование выбора состава технических и программных средств**

#### **3.4.2.1. Обоснование выбора состава технических средств**

При процессоре ниже указанного процесс решения дифференциального уравнения и отрисовки цветового графика искомой функции будет выполняться крайне долго.

При количестве памяти ОЗУ ниже указанного его может не хватить для успешного решения дифференциального уравнения и отрисовки цветового графика искомой функции.

Не менее 1,5 Гб свободного места на жестком диске требуется для корректной работы операционной системы, библиотеки Microsoft .NET Framework и данной программы.

При разрешении экрана ниже указанного окна программы могут отображаться некорректно.

Клавиатура и мышь требуются для ввода входных данных и управления программой.

#### **3.4.2.2. Обоснование выбора состава программных средств**

Поскольку программа выполнена с использованием библиотеки Microsoft .NET Framework 4.5.2, для ее выполнения требуется библиотека Microsoft .NET Framework версии не ниже 4.5, которая, в свою очередь, требует одну из указанных операционных систем.

# **4. Технико-экономические показатели**

## **4.1. Предполагаемая потребность**

Программа будет использоваться преподавателями, студентами и исследователями в области математики и информатики для решения дифференциальных уравнений в частных производных, а также для демонстрации и исследования нейросетевого способа решения дифференциальных уравнений в частных производных, оценки качества, точности и быстродействия данного способа, в том числе, в учебных и научных целях.

## **4.2. Ориентировочная экономическая эффективность**

Программа может бесплатно дать необходимый материал для преподавателей, студентов и исследователей в области математики и информатики, занимающихся исследованиями в области дифференциальных уравнений и нейронных сетей и изучением данных областей.

## **4.3. Экономические преимущества разработки по сравнению с отечественными и зарубежными аналогами**

Программа бесплатна и имеет русскоязычный интерфейс.

# **Приложение 1. Список используемой литературы**

1. Хайкин, С. Нейронные сети: полный курс, 2-е изд., испр.: пер. с англ. – М.: ООО  
   «И. Д. Вильямс», 2006.
2. Подбельский В. В. Язык C#. Базовый курс: учеб. пособие, 2-е изд., перераб. и доп. – М.: Финансы и статистика, 2013.
3. Шилдт, Г. C# 4.0: полное руководство: пер. с англ. – М.: ООО «И. Д. Вильямс», 2013.
4. Microsoft Developer Network (MSDN) [Электронный ресурс] // URL: <https://msdn.microsoft.com/ru-ru/default.aspx> (Дата обращения: 14.05.2016, режим доступа: свободный).
5. Единая система программной документации: сборник, офиц. изд. – М.: ИПК Издательство стандартов, 2001.

# **Приложение 2. Описание и функциональное назначение классов**

Описываются только классы, содержащие вручную написанные поля, методы и свойства. Полностью автоматически сгенерированные классы не описываются.

*Табл. 1. Описание и функциональное назначение классов*

|  |  |
| --- | --- |
| Класс | Назначение |
| Form1 | Представляет форму Form1 – главное окно программы. |
| FormGraph | Представляет форму FormGraph – окно с результатами решения дифференциального уравнения (окно цветового графика искомой функции). |
| FormErrorDialog | Представляет форму FormErrorDialog – диалоговое окно с информацией об ошибке. |
| NetDifferentiating | Статический класс. Представляет статические методы для вычисления производных нейросети по входам и по весовым коэффициентам в заданной точке. |
| ArctgActivationFunction | Задает функцию активации arctg. |
| LinearActivationFunction | Задает линейную функцию активации y = x. |

# **Приложение 3. Описание и функциональное назначение полей, методов и свойств**

Описываются только вручную написанные поля, методы и свойства. Полностью автоматически сгенерированные поля, методы и свойства не описываются.

*Табл. 2. Описание и функциональное назначение полей класса Form1*

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Модификатор доступа | Тип | Назначение |
| net | private | ActivationNetwork | Нейронная сеть, используемая для аппроксимации искомой функции. |
| thread | private | Thread | Поток для выполнения процесса обучения нейросети. |
| coeff | private | double | Коэффициент при f. |
| coeffx | private | double | Коэффициент при f’x. |
| coeffy | private | double | Коэффициент при f’y. |
| coeffxx | private | double | Коэффициент при f’’xx. |
| coeffxy | private | double | Коэффициент при f’’xy. |
| coeffyx | private | double | Коэффициент при f’’yx. |
| coeffyy | private | double | Коэффициент при f’’yy. |
| freeConst | private | double | Свободная константа. |
| xmin | private | double | Левая граница заданной прямоугольной области для решения уравнения. |
| ymin | private | double | Верхняя граница заданной прямоугольной области для решения уравнения. |
| xmax | private | double | Правая граница заданной прямоугольной области для решения уравнения. |
| ymax | private | double | Нижняя граница заданной прямоугольной области для решения уравнения. |
| borderConst | private | double | Константа граничных условий. |
| gradStep | private | double | Шаг градиентного спуска. |
| pbStep | private | double | Шаг элемента управления progressBar1. |
| pbValue | private | double | Текущее значение элемента управления progressBar1. |
| neuronOutputs | private | double[][] | Выходы всех нейронов нейросети в текущей точке. |
| differentsX | private | double[][] | Производные всех нейронов нейросети f’x в текущей точке. |
| differentsY | private | double[][] | Производные всех нейронов нейросети f’y в текущей точке. |
| differentsXX | private | double[][] | Вторые производные всех нейронов нейросети f’’xx в текущей точке. |
| differentsXY | private | double[][] | Вторые производные всех нейронов нейросети f’’xy в текущей точке. |
| differentsYX | private | double[][] | Вторые производные всех нейронов нейросети f’’yx в текущей точке. |
| differentsYY | private | double[][] | Вторые производные всех нейронов нейросети f’’yy в текущей точке. |
| activFuncDerivatives | private | double[][] | Производные функций активации всех нейронов нейросети в текущей точке. |
| activFuncSecondDerivatives | private | double[][] | Вторые производные функций активации всех нейронов нейросети в текущей точке. |
| activFuncThirdDerivatives | private | double[][] | Третьи производные функций активации всех нейронов нейросети в текущей точке. |
| isActivFuncDerivativeReady | private | bool[][] | Показывает, какие производные функций активации нейронов нейросети в текущей точке уже посчитаны. |
| isActivFuncSecondDerivativeReady | private | bool[][] | Показывает, какие вторые производные функций активации нейронов нейросети в текущей точке уже посчитаны. |
| isActivFuncThirdDerivativeReady | private | bool[][] | Показывает, какие третьи производные функций активации нейронов нейросети в текущей точке уже посчитаны. |
| freeFunctionIndex | private | int | Индекс выбранного элемента из ComboBoxFreeFunction (выпадающего списка, отвечающего за выбор свободной функции). |
| borderFunctionIndex | private | int | Индекс выбранного элемента из ComboBoxBorderFunction (выпадающего списка, отвечающего за выбор функции граничных условий). |
| xPointsCount | private | int | Количество точек по X |
| yPointsCount | private | int | Количество точек по Y |
| iterations | private | int | Количество итераций обучения нейросети |

*Табл. 3. Описание и функциональное назначение методов класса Form1*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| ComputeDiffEqValue | private | double | double, double | Вычисляет значение левой части уравнения в заданной точке. |
| ComputeDiffEqValueFromOutputsAndDifferents | private | double | double, double | Вычисляет значение левой части уравнения в заданной точке с использованием ранее посчитанных выходов и производных нейронов нейросети. |
| ComputeFreeFuncValue | private | double | int, double, double | Вычисляет значение функции из comboBoxFreeFunction (свободная функция уравнения) или comboBoxBorderFunction (функция граничных условий) в заданной точке. |
| ComputeFuncValue | private | double | double, double | Вычисляет значение искомой функции (заданной нейросетью) в заданной точке. |
| ComputeNetMaxMistake | private | double | out double | Вычисляет максимальную ошибку на левой части уравнения. |
| ComputeOutputsAndDifferents | private | void | double, double | Вычисляет выходы и производные нейронов нейросети в заданной точке. |
| ComputeWeightInDiffEqDifferent | private | double | int, int, int, double, double | Вычисляет производную левой части уравнения по заданному весовому коэффициенту в заданной точке. |
| RunNetworkTraining | private | void | - | Выполняет процесс обучения нейросети. |
| RunNetworkTrainingIteration | private | void | - | Выполняет одну итерацию обучения нейросети. |
| buttonAbort\_Click | private | void | object, EventArgs | Обработчик события. Вызывается при нажатии на кнопку buttonAbort («Прервать»). Прерывает процесс решения дифференциального уравнения. |
| buttonStart\_Click | private | void | object, EventArgs | Обработчик события. Вызывается при нажатии на кнопку buttonStart («Решить уравнение»). Считывает и проверяет на корректность все входные данные, и если все входные данные корректны, то запускает процесс решения дифференциального уравнения, иначе выводит на экран диалоговое окно FormErrorDialog с информацией об ошибке. |
| Form1\_FormClosing | private | void | object, FormClosingEventArgs | Обработчик события. Вызывается при закрытии главного окна программы. Проверяет, был ли уже инициализирован поток thread, и если да, то прерывает его. |

В классе Form1 отсутствуют вручную написанные свойства.

*Табл. 4. Описание и функциональное назначение полей класса FormGraph*

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Модификатор доступа | Тип | Назначение |
| net | private | ActivationNetwork | Нейронная сеть, используемая для аппроксимации искомой функции. |
| colors | private | Color[][] | Рассчитанные цвета для отрисовки цветового графика искомой функции во всех точках элемента управления для рисования pictureBox1. |
| paletteColors | private | Color[] | Рассчитанные цвета для отрисовки палитры. |
| isPaletteReady | private | bool | Показывает, рассчитаны ли уже цвета для отрисовки палитры. |
| funcValues | private | double[][] | Рассчитанные значения искомой функции во всех точках элемента управления для рисования pictureBox1. |
| funcMin | private | double | Наименьшее значение искомой функции по всем точкам элемента управления для рисования pictureBox1. |
| funcMax | private | double | Наибольшее значение искомой функции по всем точкам элемента управления для рисования pictureBox1. |
| xStep | private | double | Шаг по X. |
| yStep | private | double | Шаг по Y. |
| xmin | private | double | Левая граница заданной прямоугольной области для решения уравнения. |
| ymin | private | double | Верхняя граница заданной прямоугольной области для решения уравнения. |
| xmax | private | double | Правая граница заданной прямоугольной области для решения уравнения. |
| ymax | private | double | Нижняя граница заданной прямоугольной области для решения уравнения. |

*Табл. 5. Описание и функциональное назначение методов класса FormGraph*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| FormGraph | public | - | ActivationNetwork, double, double, double, double, double, double | Конструктор класса FormGraph. |
| ComputeFuncValue | private | double | double, double | Вычисляет значение искомой функции (заданной нейросетью) в заданной точке. |
| InitializeFunction | private | void | - | Вычисляет значения функции и цветов во всех точках отрисовываемой области и цветов для палитры, а также вызывает отрисовку графика функции и палитры. |
| FormGraph\_SizeChanged | private | void | object, EventArgs | Обработчик события. Вызывается при изменении размеров окна FormGraph. Вызывает функцию InitializeFunction |
| pictureBox1\_Paint | private | void | object, PaintEventArgs | Обработчик события. Вызывается при отрисовке элемента управления для рисования pictureBox1. Отрисовывает цветовой график искомой функции. |
| pictureBox1\_MouseLeave | private | void | object, EventArgs | Обработчик события. Вызывается при покидании курсором мыши области экрана, занимаемой элементом управления для рисования pictureBox1. Отключает видимость надписей labelX, labelY, labelF, информирующих о координатах выбранной точки и значении искомой функции в ней. |
| pictureBox1\_MouseMove | private | void | object, MouseEventArgs | Обработчик события. Вызывается при движении курсора мыши по области экрана, занимаемой элементом управления для рисования pictureBox1. Отображает координаты выбранной точки и значение искомой функции в ней в надписях labelX, labelY, labelF. |
| pictureBoxPalette\_Paint | private | void | object, PaintEventArgs | Обработчик события. Вызывается при отрисовке элемента управления для рисования pictureBoxPalette Отрисовывает палитру. |

В классе FormGraph отсутствуют вручную написанные свойства.

В классе FormErrorDialog отсутствуют вручную написанные поля и свойства.

*Табл. 6. Описание и функциональное назначение методов класса FormErrorDialog*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| FormErrorDialog | public | - | string | Конструктор класса FormErrorDialog. |
| buttonOK\_Click | private | void | object, EventArgs | Обработчик события. Вызывается при нажатии на кнопку buttonOK («ОК»). Закрывает диалоговое окно FormErrorDialog. |

В классе NetDifferentiating отсутствуют вручную написанные поля и свойства.

*Табл. 7. Описание и функциональное назначение методов класса NetDifferentiating*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| NetDifferentX | public | double | ActivationNetwork, double, double | Вычисляет производную f'x нейросети в заданной точке. |
| NetDifferentY | public | double | ActivationNetwork, double, double | Вычисляет производную f'y нейросети в заданной точке. |
| NetDifferentXX | public | double | ActivationNetwork, double, double | Вычисляет вторую производную f''xx нейросети в заданной точке. |
| NetDifferentXY | public | double | ActivationNetwork, double, double | Вычисляет вторую производную f''xy нейросети в заданной точке. |
| NetDifferentYX | public | double | ActivationNetwork, double, double | Вычисляет вторую производную f''yx нейросети в заданной точке. |
| NetDifferentYY | public | double | ActivationNetwork, double, double | Вычисляет вторую производную f''yy нейросети в заданной точке. |
| NetDifferentWeight | public | double | ActivationNetwork, double[][], ref double[][], ref bool[][], int, int, int, double, double | Вычисляет производную нейросети по заданному весовому коэффициенту в заданной точке. |
| NetDifferentXWeight | public | double | ActivationNetwork, double[][], double[][], ref double[][], ref bool[][], ref double[][], ref bool[][], int, int, int, double, double | Вычисляет производную производной f'x нейросети по заданному весовому коэффициенту в заданной точке. |
| NetDifferentYWeight | public | double | ActivationNetwork, double[][], double[][], ref double[][], ref bool[][], ref double[][], ref bool[][], int, int, int, double, double | Вычисляет производную производной f'y нейросети по заданному весовому коэффициенту в заданной точке. |
| NetDifferentXXWeight | public | double | ActivationNetwork, double[][], double[][], double[][], ref double[][], ref bool[][], ref double[][], ref bool[][], ref double[][], ref bool[][], int, int, int, double, double | Вычисляет производную второй производной f''xx нейросети по заданному весовому коэффициенту в заданной точке. |
| NetDifferentXYWeight | public | double | ActivationNetwork, double[][], double[][], double[][], ref double[][], ref bool[][], ref double[][], ref bool[][], ref double[][], ref bool[][], int, int, int, double, double | Вычисляет производную второй производной f''xy нейросети по заданному весовому коэффициенту в заданной точке. |
| NetDifferentYXWeight | public | double | ActivationNetwork, double[][], double[][], double[][], ref double[][], ref bool[][], ref double[][], ref bool[][], ref double[][], ref bool[][], int, int, int, double, double | Вычисляет производную второй производной f''yx нейросети по заданному весовому коэффициенту в заданной точке. |
| NetDifferentYYWeight | public | double | ActivationNetwork, double[][], double[][], double[][], ref double[][], ref bool[][], ref double[][], ref bool[][], ref double[][], ref bool[][], int, int, int, double, double | Вычисляет производную второй производной f''yy нейросети по заданному весовому коэффициенту в заданной точке. |
| ComputeNeuronOutput | public | double | ActivationNetwork, Neuron, int, double, double | Вычисляет выход заданного нейрона нейросети в заданной точке. |
| ComputeNeuronOutputs | public | double[][] | ActivationNetwork, double, double | Вычисляет выходы нейронов нейросети в заданной точке. |
| DifferentX | private | double | ActivationNetwork, ActivationNeuron, ArctgActivationFunction, int, double, double | Вычисляет производную f'x заданного нейрона нейросети в заданной точке. |
| DifferentsX | public | double[][] | ActivationNetwork, ArctgActivationFunction, double[][], ref double[][], ref bool[][], double, double | Вычисляет производные f'x нейронов нейросети в заданной точке. |
| DifferentY | private | double | ActivationNetwork, ActivationNeuron, ArctgActivationFunction, int, double, double | Вычисляет производную f'y заданного нейрона нейросети в заданной точке. |
| DifferentsY | public | double[][] | ActivationNetwork, ArctgActivationFunction, double[][], ref double[][], ref bool[][], double, double | Вычисляет производные f'y нейронов нейросети в заданной точке. |
| DifferentXX | private | double | ActivationNetwork, ActivationNeuron, ArctgActivationFunction, int, double, double | Вычисляет вторую производную f''xx заданного нейрона нейросети в заданной точке. |
| DifferentsXX | public | double[][] | ActivationNetwork, ArctgActivationFunction, double[][], double[][], ref double[][], ref bool[][], ref double[][], ref bool[][], double, double | Вычисляет вторые производные f''xx нейронов нейросети в заданной точке. |
| DifferentXY | private | double | ActivationNetwork, ActivationNeuron, ArctgActivationFunction, int, double, double | Вычисляет вторую производную f''xy заданного нейрона нейросети в заданной точке. |
| DifferentsXY | public | double[][] | ActivationNetwork, ArctgActivationFunction, double[][], double[][], ref double[][], ref bool[][], ref double[][], ref bool[][], double, double | Вычисляет вторые производные f''xy нейронов нейросети в заданной точке. |
| DifferentYX | private | double | ActivationNetwork, ActivationNeuron, ArctgActivationFunction, int, double, double | Вычисляет вторую производную f''yx заданного нейрона нейросети в заданной точке. |
| DifferentsYX | public | double[][] | ActivationNetwork, ArctgActivationFunction, double[][], double[][], ref double[][], ref bool[][], ref double[][], ref bool[][], double, double | Вычисляет вторые производные f''yx нейронов нейросети в заданной точке. |
| DifferentYY | private | double | ActivationNetwork, ActivationNeuron, ArctgActivationFunction, int, double, double | Вычисляет вторую производную f''yy заданного нейрона нейросети в заданной точке. |
| DifferentsYY | public | double[][] | ActivationNetwork, ArctgActivationFunction, double[][], double[][], ref double[][], ref bool[][], ref double[][], ref bool[][], double, double | Вычисляет вторые производные f''yy нейронов нейросети в заданной точке. |
| DifferentWeight | private | double | ActivationNetwork, ActivationNeuron, ArctgActivationFunction, double[][], int, int, ref double[][], ref bool[][], int, int, int, double, double | Вычисляет производную заданного нейрона нейросети по заданному весовому коэффициенту в заданной точке. |
| DifferentXWeight | private | double | ActivationNetwork, ActivationNeuron, ArctgActivationFunction, double[][], double[][], int, int, ref double[][], ref bool[][], ref double[][], ref bool[][], int, int, int, double, double | Вычисляет производную производной f'x нейросети по заданному весовому коэффициенту в заданной точке. |
| DifferentYWeight | private | double | ActivationNetwork, ActivationNeuron, ArctgActivationFunction, double[][], double[][], int, int, ref double[][], ref bool[][], ref double[][], ref bool[][], int, int, int, double, double | Вычисляет производную производной f'y нейросети по заданному весовому коэффициенту в заданной точке. |
| DifferentXXWeight | private | double | ActivationNetwork, ActivationNeuron, ArctgActivationFunction, double[][], double[][], double[][], int, int, ref double[][], ref bool[][], ref double[][], ref bool[][], ref double[][], ref bool[][], int, int, int, double, double | Вычисляет производную второй производной f''xx нейросети по заданному весовому коэффициенту в заданной точке. |
| DifferentXYWeight | private | double | ActivationNetwork, ActivationNeuron, ArctgActivationFunction, double[][], double[][], double[][], int, int, ref double[][], ref bool[][], ref double[][], ref bool[][], ref double[][], ref bool[][], int, int, int, double, double | Вычисляет производную второй производной f''xy нейросети по заданному весовому коэффициенту в заданной точке. |
| DifferentYXWeight | private | double | ActivationNetwork, ActivationNeuron, ArctgActivationFunction, double[][], double[][], double[][], int, int, ref double[][], ref bool[][], ref double[][], ref bool[][], ref double[][], ref bool[][], int, int, int, double, double | Вычисляет производную второй производной f''yx нейросети по заданному весовому коэффициенту в заданной точке. |
| DifferentYYWeight | private | double | ActivationNetwork, ActivationNeuron, ArctgActivationFunction, double[][], double[][], double[][], int, int, ref double[][], ref bool[][], ref double[][], ref bool[][], ref double[][], ref bool[][], int, int, int, double, double | Вычисляет производную второй производной f''yy нейросети по заданному весовому коэффициенту в заданной точке. |

В классе ArctgActivationFunction отсутствуют вручную написанные поля и свойства.

*Табл. 8. Описание и функциональное назначение методов класса ArctgActivationFunction*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| Function | public | double | double | Вычисляет значение функции активации arctg от аргумента x. |
| Derivative | public | double | double | Вычисляет производную функции активации arctg от аргумента x. |
| Derivative2 | public | double | double | Вычисляет производную функции активации arctg от аргумента y. |
| SecondDerivative | public | double | double | Вычисляет вторую производную функции активации arctg от аргумента x. |
| SecondDerivative2 | public | double | double | Вычисляет вторую производную функции активации arctg от аргумента y. |
| ThirdDerivative | public | double | double | Вычисляет третью производную функции активации arctg от аргумента x. |
| ThirdDerivative2 | public | double | double | Вычисляет третью производную функции активации arctg от аргумента y. |

В классе LinearActivationFunction отсутствуют вручную написанные поля и свойства.

*Табл. 9. Описание и функциональное назначение методов класса LinearActivationFunction*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя | Модификатор доступа | Тип | Аргументы | Назначение |
| Function | public | double | double | Возвращает x. |
| Derivative | public | double | double | Возвращает 1. |
| Derivative2 | public | double | double | Возвращает 1. |
| SecondDerivative | public | double | double | Возвращает 0. |
| SecondDerivative2 | public | double | double | Возвращает 0. |
| ThirdDerivative | public | double | double | Возвращает 0. |
| ThirdDerivative2 | public | double | double | Возвращает 0. |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | |
| Изм. | Номера листов | | | | Всего листов (страниц) в документе | № документа | Входя- щий № сопро- водит. докум. и дата | Под- пись | Дата |
| изменен- ных | заменен- ных | новых | аннулиро- ванных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |