

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

Факультет компьютерных наук
Департамент программной инженерии

СОГЛАСОВАНО

Старший преподаватель департамента
программной инженерии факультета
компьютерных наук

УТВЕРЖДАЮ

Академический руководитель
образовательной программы
«Программная инженерия»
профессор департамента программной
инженерии, канд. техн. наук

_____ Д. В. Пантюхин
«__» _____ 2016 г.

_____ В. В. Шилов
«__» _____ 2016 г.

**ПРОГРАММА НЕЙРОСЕТЕВОГО РЕШЕНИЯ
ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ В ЧАСТНЫХ
ПРОИЗВОДНЫХ**

Пояснительная записка

ЛИСТ УТВЕРЖДЕНИЯ

RU.17701729.503100-01 81 01-1-ЛУ

Исполнитель
студент группы БПИ154
_____ / Ригин А. М. /
«__» _____ 2016 г.

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл	RU.17701729.503100 -01 81 01-1-ЛУ

Москва 2016

УТВЕРЖДЕН
RU.17701729.503100-01 51 01-1-ЛУ

**ПРОГРАММА НЕЙРОСЕТЕВОГО РЕШЕНИЯ
ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ В ЧАСТНЫХ
ПРОИЗВОДНЫХ**

Пояснительная записка

RU.17701729.503100-01 81 01-1

Листов 28

Инв. № подл	Подп. и дата
Взам. инв. №	Инв. № дубл.
Подп. и дата	Подп. и дата
Инв. № подл	Подп. и дата
RU.17701729.503100-01 81 01-1	

Москва 2016

Содержание

1. Введение.....	3
1.1. Наименование программы	3
1.2. Основания для разработки	3
2. Назначение и область применения	4
2.1. Назначение разработки	4
2.1.1. Функциональное назначение	4
2.1.2. Эксплуатационное назначение.....	4
2.2. Область применения.....	4
3. Технические характеристики.....	5
3.1. Постановка задачи на разработку программы	5
3.2. Описание и обоснование алгоритма и функционирования программы	5
3.2.1. Описание алгоритма и функционирования программы	5
3.2.2. Обоснование выбора алгоритма	6
3.3. Описание и обоснование метода организации входных и выходных данных	7
3.3.1. Описание метода организации входных и выходных данных	7
3.3.2. Обоснование метода организации входных и выходных данных	7
3.4. Описание и обоснование выбора состава технических и программных средств	7
3.4.1. Состав технических и программных средств	7
3.4.2. Обоснование выбора состава технических и программных средств.....	8
4. Техничко-экономические показатели.....	9
4.1. Предполагаемая потребность.....	9
4.2. Ориентировочная экономическая эффективность	9
4.3. Экономические преимущества разработки по сравнению с отечественными и зарубежными аналогами.....	9
Приложение 1. Список используемой литературы	10
Приложение 2. Описание и функциональное назначение классов.....	11
Приложение 3. Описание и функциональное назначение полей, методов и свойств.....	12

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

1. Введение

1.1. Наименование программы

Наименование программы: «Программа нейросетевого решения дифференциальных уравнений в частных производных».

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 нейросетевого решения дифференциальных уравнений в частных производных».

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

2. Назначение и область применения

2.1. Назначение разработки

2.1.1. Функциональное назначение

Программа будет применяться для решения дифференциальных уравнений, представляющих собой в левой части полиномы из искомой функции, ее производных, коэффициентов, свободной функции, с заданными граничными условиями, с помощью аппроксимации искомой функции, используя нейронную сеть. У пользователя будет возможность ввести данные об уравнении, выбрать параметры нейронной сети и ее обучения, посмотреть визуализированный результат.

2.1.2. Эксплуатационное назначение

Программа будет использоваться 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.

2.2. Область применения

Программа будет применяться для решения дифференциальных уравнений, представляющих собой в левой части полиномы из искомой функции, ее производных, коэффициентов, свободной функции, с заданными граничными условиями, с помощью аппроксимации искомой функции, используя нейронную сеть.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

3. Технические характеристики

3.1. Постановка задачи на разработку программы

Программа должна обеспечивать возможность выполнения следующих функций:

- 1) задание дифференциального уравнения для решения, границ области, на которой дифференциальное уравнение решается, и граничных условий;
- 2) настройка параметров нейронной сети и ее обучения: количество нейронов в слоях нейронной сети, количество итераций обучения, шаг градиентного спуска;
- 3) решение заданного дифференциального уравнения нейросетевым способом (с помощью аппроксимации искомой функции, используя нейронную сеть);
- 4) вывод визуализированных результатов решения дифференциального уравнения в форме значений искомой функции на заданной области;
- 5) прерывание процесса решения дифференциального уравнения во время его выполнения;
- 6) решение нового дифференциального уравнения без перезапуска программы.

3.2. Описание и обоснование алгоритма и функционирования программы

3.2.1. Описание алгоритма и функционирования программы

Искомая функция представляется в виде нейросети – четырехслойного персептрона.

В процессе решения дифференциального уравнения данная нейросеть аппроксимирует искомую функцию. Для этих целей выполняется задаваемое пользователем количество итераций обучения нейросети.

На каждой из итераций происходит следующее:

- 1) программа проходит по всем заданным точкам заданной области для решения уравнения (границы прямоугольной области и количество точек по X и по Y задается пользователем, шаг по X и по Y рассчитывается исходя из этих данных);
- 2) в каждой из точек программа методом обратного распространения ошибки (запоминая производные выходов нейронов нейросети сразу после их первого расчета) рассчитывает производные уравнения и граничных условий по каждому из весовых коэффициентов нейросети (предварительно запоминая все необходимые данные, такие как выходы всех нейронов нейросети в данной точке и необходимые производные всех нейронов нейросети в данной точке по входам нейросети);
- 3) производные уравнения и граничных условий по каждому из весовых коэффициентов нейросети во всех заданных точках складываются;
- 4) из каждого весового коэффициента нейросети вычитается сумма производных по нему уравнения и граничных условий, умноженная на шаг градиентного спуска, заданный пользователем.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

После выполнения всех итераций в нашей нейросети мы получаем аппроксимированную с определенной точностью искомую функцию, после чего программа рассчитывает максимальную ошибку нейросети по уравнению и по граничным условиям, проходя по большему количеству точек заданной области, чем при обучении нейросети.

После выполнения данных действий производится отрисовка цветового графика искомой функции на заданной области:

- 1) считается, что элемент управления PictureBox для рисования соответствует заданной области для решения уравнения, левый верхний угол данного элемента управления считается соответствующим точке (X_{\min} , Y_{\min}) заданной области, правый нижний угол – соответствующим точке (X_{\max} , Y_{\max}) заданной области, для остальных точек данного элемента управления также рассчитываются соответствующие точки (X , Y) заданной области, по их расположению на элементе управления;
- 2) для каждой из точек элемента управления для рисования берется рассчитанная соответствующая точка (X , Y) и для нее рассчитывается значение искомой функции (значение выхода нейросети, с помощью которой производилась аппроксимация искомой функции), значения во всех точках запоминаются;
- 3) находится наименьшее и наибольшее значения искомой функции по всем точкам элемента управления для рисования;
- 4) считается, что каждое из найденных значений функции в точках элемента управления для рисования соответствует определенному цвету по шкале от (0, 255, 255) до (255, 0, 255) по системе RGB, где (0, 255, 255) соответствует наименьшему значению искомой функции по всем точкам элемента управления для рисования, а (255, 0, 255) – наибольшему значению, таким образом, для каждой точки определяется цвет, он запоминается и отрисовывается, так отрисовывается весь цветовой график искомой функции;
- 5) в случае необходимости повторной отрисовки графика при сохранении размеров элемента управления для рисования производится отрисовка графика из уже сохраненных цветов, повторных расчетов значений функции и цветов не производится;
- 6) в случае изменения размеров элемента управления для рисования производится перерисовка графика, при этом заново выполняются пп. 1 – 4.

3.2.2. Обоснование выбора алгоритма

Алгоритм аппроксимации искомой функции с помощью четырехслойного персептрона с применением метода обратного распространения ошибки и с нахождением производных данной нейросети по входам выбран, поскольку аналитически несложно доказать, что данный алгоритм действительно применим для нейросетевого решения дифференциальных уравнений в частных производных.

Алгоритм построения цветового графика искомой функции выбран, поскольку он является одним из наиболее оптимальных способов вывода результатов решения дифференциального уравнения в частных производных.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

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 1280x800;
- 5) клавиатура и мышь.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

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.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

4. Техничко-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.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

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 С#. Базовый курс: учеб. пособие, 2-е изд., перераб. и доп. – М.: Финансы и статистика, 2013.
3. Шилдт, Г. С# 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.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

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	Задаёт функцию активации arctg.
LinearActivationFunction	Задаёт линейную функцию активации $y = x$.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

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

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

			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

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

			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

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

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	double	Вычисляет максимальную ошибку на левой части уравнения.
ComputeOutputsAndDifferents	private	void	double, double	Вычисляет выходы и производные нейронов

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

				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

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

				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

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

			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.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

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.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

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

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

				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.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

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 у 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 у 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 у в заданной точке.
NetDifferentXXWeight	public	double	ActivationNetwork, double[[]], double[[]],	Вычисляет производную

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

			double[], ref double[], ref bool[], ref double[], ref bool[], ref double[], ref bool[], int, int, int, double, double	второй производной f''_{xx} нейросети по заданному весовому коэффициенту у 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 у 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 у 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 у в заданной точке.
ComputeNeuronOutput	public	double	ActivationNetwork, Neuron, int, double, double	Вычисляет выход заданного нейрона нейросети в заданной точке.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

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.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

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.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

			ref bool[], double, double	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

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

			ref bool[], int, int, int, double, double	коэффициент у 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 у 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 у 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

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

				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.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата

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					

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.503100-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. инв №	Инв. № дубл.	Подп. и дата