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

Факультет компьютерных наук  
Департамент программной инженерии

|  |  |
| --- | --- |
| СОГЛАСОВАНО  Доцент департамента больших данных и информационного поиска факультета компьютерных наук, к.ф.-м.н.  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В. Л. Чернышев «\_\_» \_\_\_\_\_\_\_\_\_\_\_ 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.17701729.04.13-01 ТЗ 01-1-ЛУ |

**ПРОГРАММА ДЛЯ МОДЕЛИРОВАНИЯ ДВИЖЕНИЯ ТОЧЕК НА ОРИЕНТИРОВАННОМ МЕТРИЧЕСКОМ ГРАФЕ, С УСЛОВИЕМ**

**СИНХРОНИЗАЦИИ В ВЕРШИНАХ**

**Техническое задание**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.04.13-01 ТЗ 01-1-ЛУ**

Исполнитель

студент группы БПИ196

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / А. А. Баранова /

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2019 г.

**Москва 2019**

УТВЕРЖДЕНRU.17701729.04.13-01 ТЗ 01-1-ЛУ

|  |  |
| --- | --- |
| Подп. и дата |  |
| Инв. № дубл. |  |
| Взам. инв. № |  |
| Подп. и дата |  |
| Инв. № подл | RU.17701729.04.13-01 ТЗ 01-1-ЛУ |

**ПРОГРАММА ДЛЯ МОДЕЛИРОВАНИЯ ДВИЖЕНИЯ ТОЧЕК НА**

**ОРИЕНТИРОВАННОМ МЕТРИЧЕСКОМ ГРАФЕ, С УСЛОВИЕМ**

**СИНХРОНИЗАЦИИ В ВЕРШИНАХ**

**Техническое задание**

**RU.17701729.04.13-01 ТЗ 01-1**

**Листов 19**

**Москва 2019**

Содержание

[1. ВВЕДЕНИЕ 4](#_Toc40704212)

[1.1. Наименование программы 4](#_Toc40704213)

[1.2. Краткая характеристика области применения 4](#_Toc40704214)

[2. ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ 5](#_Toc40704215)

[2.1 Документы, на основании которых ведется разработка 5](#_Toc40704216)

[2.2 Наименование темы разработки 5](#_Toc40704217)

[3. НАЗНАЧЕНИЕ РАЗРАБОТКИ 6](#_Toc40704218)

[3.1. Функциональное назначение 6](#_Toc40704219)

[3.2. Эксплуатационное назначение 6](#_Toc40704220)

[4. ТРЕБОВАНИЯ К ПРОГРАММЕ 7](#_Toc40704221)

[4.1. Требования к функциональным характеристикам 7](#_Toc40704222)

[4.1.1. Требования к составу выполняемых функций 7](#_Toc40704223)

[4.1.2. Требования к организации входных данных 8](#_Toc40704224)

[4.1.3. Требования к организации выходных данных 8](#_Toc40704225)

[4.2. Требования к интерфейсу 8](#_Toc40704226)

[4.3. Требования к надежности 9](#_Toc40704227)

[4.3.1. Требования к обеспечению надежного (устойчивого) функционирования программы 9](#_Toc40704228)

[4.3.3. Отказы из-за некорректных действий оператора 9](#_Toc40704229)

[4.4. Условия эксплуатации 9](#_Toc40704230)

[4.4.1. Климатические условия 9](#_Toc40704231)

[4.4.2. Требования к квалификации оператора 9](#_Toc40704232)

[4.5. Требования к составу и параметру технических средств 10](#_Toc40704233)

[4.6. Требования к информационной и программной совместимости 10](#_Toc40704234)

[4.7. Требования к маркировке и упаковке 10](#_Toc40704235)

[4.8. Требования к транспортированию и хранению 10](#_Toc40704236)

[5. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ 11](#_Toc40704237)

[5.1. Состав программной документации должен включать в себя следующие компоненты: 11](#_Toc40704238)

[5.2. Специальные требования к программной документации 11](#_Toc40704239)

[6. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ 12](#_Toc40704240)

[6.1. Ориентировочная экономическая эффективность 12](#_Toc40704241)

[6.2. Предполагаемая потребность 12](#_Toc40704242)

[6.3. Экономические преимущества разработки по сравнению с отечественными и зарубежными аналогами 12](#_Toc40704243)

[7. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ 13](#_Toc40704244)

[7.1. Необходимые стадии разработки, этапы и содержание работ 13](#_Toc40704245)

[7.2. Сроки и исполнители 14](#_Toc40704246)

[8. ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ 15](#_Toc40704247)

[8.1. Виды испытаний 15](#_Toc40704248)

[8.2. Общие требования к приемке работы 15](#_Toc40704249)

[9. СПИСОК ЛИТЕРАТУРЫ 16](#_Toc40704250)

[ПРИЛОЖЕНИЕ 1 17](#_Toc40704251)

[ПРИЛОЖЕНИЕ 2 18](#_Toc40704252)

# **1. ВВЕДЕНИЕ**

* 1. **Наименование программы**

Наименование программы – «Программа для моделирования движения точек на ориентированном метрическом графе, с условием синхронизации в вершинах»

«The Program for Modeling the Movement of Points on Directed Metric Graph, with the Condition of Synchronization at the Vertices».

* 1. **Краткая характеристика области применения**

Программа предназначена для моделирования движения точек на сильно связных ориентированных метрических графах, с условием синхронизации в вершинах. Программный инструмент позволяет строить графы и моделировать на них движение точек двух разных типов: стандартного (синхронизация в вершинах по установленным порогам) и модели песка (синхронизация по степени вершины); получать визуализацию процесса распространения точек на графе, изображение зависимости числа точек от времени в виде графика, изображение графика распределения размеров лавин при моделировании движения песка; сохранять анимацию процесса в формате GIF-изображения.

Данная программа несет научно-образовательный характер и может использоваться исследователями, работающими в различных областях математики и информатики при рассмотрении разнообразных задач на графах или изучении абелевой модели песка. Также программа потенциально может найти применение в нейробиологии при моделировании различных процессов в нейронных сетях с использованием данной математической модели.

# **2. ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ**

* 1. **Документы, на основании которых ведется разработка**

Основанием для разработки является приказ декана факультета компьютерных наук И.В. Аржанцева "Об утверждении тем, руководителей курсовых работ студентов образовательной программы «Программная инженерия» факультета компьютерных наук" № 2.3-02/1112-04 от 11.12.2019.

**2.2 Наименование темы разработки**

Наименование темы разработки – «Программа для моделирования движения точек на ориентированном метрическом графе, с условием синхронизации в вершинах»

Условное обозначение темы разработки – «The Program for Modeling the Movement of Points on Directed Metric Graph, with the Condition of Synchronization at the Vertices».

Программа выполняется в рамках темы курсовой работы в соответствии с учебным планом подготовки бакалавров по направлению 09.03.04 «Программная инженерия» Национального исследовательского университета «Высшая школа экономики», факультет компьютерных наук, департамент программной инженерии.

# **3. НАЗНАЧЕНИЕ РАЗРАБОТКИ**

## **3.1. Функциональное назначение**

Функциональным назначением разрабатываемого приложения является работа с выбранным пользователем из архива ранее созданных, либо новым, построенным им в специальном редакторе, сильно связным ориентированным метрическим графом с целью моделирования движения на нем точек, с условием синхронизации в вершинах, а также получения минимального анализа поведения этих точек и визуализации происходящих на графе процессов.

## **3.2. Эксплуатационное назначение**

Программа может быть использована преподавателями, студентами или исследователями, работающими в различных областях математики и информатики при рассмотрении разнообразных задач на графах, или нейробиологии - при моделировании различных реальных процессов в нейронных сетях с использованием данной математической модели. Таким образом, продукт разработки позволит решать задачи, возникающие при составлении теоретических моделей в ходе научных исследований.

# **4. ТРЕБОВАНИЯ К ПРОГРАММЕ**

## **4.1. Требования к функциональным характеристикам**

### **4.1.1. Требования к составу выполняемых функций**

Программа должна обеспечивать возможность выполнения следующих функций:

1) Создание и редактирование графа:

* построение графа с использованием специальных инструментов путем проставления вершин на поле для отрисовки и соединения этих вершин дугами с заданием их весов;
* удаление вершин (с последующим удалением инцидентных дуг) и дуг графа;
* перемещение любой из вершин графа с его последующей перерисовкой, а также перемещение графа целиком;
* масштабирование изображения графа;
* изменение цветов отрисовки графа;
* отменить последнее действие и вернуть последнее отмененное действие;
* изменение радиуса вершин;
* очистка поля для рисования;
* выбор файла для сохранения графа и само сохранение (изображением, файлом с данными графа, папкой с изображением и файлом с данными) в специальном формате .dgmm [см. Приложение 2].;
* открытие графа из файла .dgmm для редактирования и/или работы с ним;
* генерация случайного сильно связного графа;
* генерация треугольной и квадратной решетки.

2) Задание дополнительных параметров, необходимых для работы алгоритма:

* порогов, по которым будет происходить синхронизация при стандартном моделировании;
* периодов восстановления вершин после выпускания точек;
* начальных состояний вершин, то есть количества точек в них в начальный момент времени;
* типа моделирования (стандартный или модель песка);
* дополнительных действий (построение графика, сохранение GIF)
* выбор скорости распространения.

3) Анализ графа и введенных значений на корректность.

4) Моделирование движения точек:

* запуск движения, его остановка и сброс;
* параллельное с анимацией заполнение данных графиков;
* сохранение данных построенных графиков (изображением и/или .csv файлом);
* сохранение созданного GIF-изображения в выбранный файл;
* выбор вершины стока перед началом моделирования движения песка;
* выбор вершины для добавления песчинки после завершения движения в процессе моделирования движения песка;

### **4.1.2. Требования к организации входных данных**

Программа должна предоставлять пользователю возможность открыть ранее созданный в этой программе и сохраненный в .dgmm файл граф для его дальнейшего редактирования и/или работы с ним, а также возможность создать новый в редакторе.

### **4.1.3. Требования к организации выходных данных**

Программа должна выводить полученные в ходе работы результаты в отведенные окна и по желанию пользователя сохранять их в файл.

## **4.2. Требования к интерфейсу**

1) Приложение должно иметь интуитивно понятный оконный Windows-интерфейс, содержащий основное окно, в котором происходит основная работа с графом, настройка параметров моделирования, управление процессом моделирования, а так же вспомогательные окна для генерации случайных графов и решеток и окна для отображения графиков, полученных в процессе моделирования;

2) Интерфейс должен быть реализован на международном языке (английский);

3) Интерфейс окна для выбора типа входных данных должен быть простым, позволяющим выбрать способ предоставления программе графа;

4) Создание нового графа должно происходить на специальном поле, а инструменты отрисовки, расположенные в боковом меню, должны быть удобны в использовании;

5) Окно создания и редактирования графа должно содержать кнопку для сохранения графа в файл, а также кнопку, при нажатии которой будет осуществляться запуск работы алгоритма или же вывод соответствующего сообщения, если анализ входных данных показал их некорректность;

6) Среди окон, предоставляющих результаты моделирования, должно быть основное окно приложения, в которое будет выводиться анимация движения, а также окна для вывода графиков, в которых должна быть реализована функция сохранения графиков;

7) Приложение должно иметь верхнее меню, реализующее возможности сохранения, открытия нового проекта, открытия графа из файлов, возврата к стартовому окну, предоставления информации о разработчике и кнопка запуска движения, отображающаяся, когда пользователь находится в редакторе, а также кнопки остановки моделирования и сброса, отображающиеся, когда моделирование запущено.

## **4.3. Требования к надежности**

**4.3.1. Требования к обеспечению надежного (устойчивого) функционирования программы**

Для корректной работы программы необходимо обеспечить устойчивую работу компьютера, на котором она запущена, его бесперебойное, а также защиту от воздействия различных видов вредоносного программного обеспечения.

Выполнение программы не должно аварийно завершаться ни при каком наборе входных данных, программа должна обрабатывать все исключительные ситуации.

**4.3.3. Отказы из-за некорректных действий оператора**

Отказ функционала программы возможен вследствие некорректных действий пользователя при пользовании операционной системой. Для недопущения отказа программы вследствие некорректного ввода данных оператором, следует предусмотреть обработку всех исключительных ситуаций, обеспечить проверку входных данных на корректность и выводить сообщение об ошибке при необходимости.

## **4.4. Условия эксплуатации**

**4.4.1. Климатические условия**

Требований к климатическим условиям эксплуатации не предъявляется.

**4.4.2. Требования к квалификации оператора**

Пользователь программы должен обладать базовыми навыками работы с персональным компьютером и оконными приложениями, а также (рекомендовано) обладать базовыми знаниями в теории графов.

## **4.5. Требования к составу и параметру технических средств**

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

1. персональный компьютер, оснащенный 32-разрядным (x86) или 64-разрядным (x64) процессором с тактовой частотой 1 ГГц и выше (рекомендуется процессор Intel Core i7 с частотой 1.9ГГц и выше или аналогичный процессор);
2. 1024 МБ оперативной памяти или больше;
3. не менее 4.5 ГБ свободного места на жестком диске;
4. видеокарта и монитор с разрешением не менее чем 1258x753 точек;
5. клавиатура и мышь.

## **4.6. Требования к информационной и программной совместимости**

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

1. операционная система Microsoft Windows 7 SP1 или более поздняя версия (кроме Windows 8);
2. установленный Microsoft .NET Framework 4.7;
3. среда программирования – Microsoft Visual Studio 2015 и выше;
4. программа должна быть реализована на языке С#.

## **4.7. Требования к маркировке и упаковке**

Приложение распространяется на компакт диске (CD), флэш носителе или посредством сети Интернет. Носитель должен иметь маркировку с обозначением наименования изделия, темы разработки, фамилии, имени и отчества исполнителя и руководителя разработки, учебной группы и года выпуска изделия.

## **4.8. Требования к транспортированию и хранению**

Требования к транспортированию и хранению программы соответствуют стандартным требованиям к транспортированию и хранению соответствующих электронных и бумажных носителей информации.

# **5. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ**

## **5.1. Состав программной документации должен включать в себя следующие компоненты:**

1. «Программа для моделирования движения точек на ориентированном метрическом графе, с условием синхронизации в вершинах». Техническое задание (ГОСТ 19.201-78)
2. «Программа для моделирования движения точек на ориентированном метрическом графе, с условием синхронизации в вершинах». Программа и методика испытаний (ГОСТ 19.301-78)
3. «Программа для моделирования движения точек на ориентированном метрическом графе, с условием синхронизации в вершинах». Пояснительная записка (ГОСТ 19.404-79)
4. «Программа для моделирования движения точек на ориентированном метрическом графе, с условием синхронизации в вершинах». Руководство оператора (ГОСТ 19.505-79)
5. «Программа для моделирования движения точек на ориентированном метрическом графе, с условием синхронизации в вершинах». Текст программы (ГОСТ 19.401-78)

## **5.2. Специальные требования к программной документации**

Документы к программе должны быть выполнены в соответствии с ГОСТ 19.106-78 и ГОСТами к каждому виду документа (см. п. 5.1.);

Пояснительная записка должна быть загружена в систему Антиплагиат через LMS «НИУ ВШЭ».

Документация и программа сдаются в электронном виде в формате .pdf или .docx. в архиве формата .zip или .rar;

За один день до защиты комиссии все материалы курсового проекта:

* техническая документация,
* программный проект,
* исполняемый файл,
* отзыв руководителя
* лист Антиплагиата

должны быть загружены одним или несколькими архивами в проект дисциплины «Курсовой проект 2019-2020» в личном кабинете в информационной образовательной среде LMS (Learning Management System) НИУ ВШЭ.

# **6. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ**

## **6.1. Ориентировочная экономическая эффективность**

В рамках данной работы расчёт экономической эффективности не предусмотрен.

Приложение сможет бесплатно предоставлять необходимый материал для преподавателей, студентов и исследователей в области математики и информатики, занимающихся исследованиями в области теории графов, а также для исследователей, использующих графы в качестве математических моделей реальных процессов.

## **6.2. Предполагаемая потребность**

Программа может быть использована преподавателями, студентами и исследователями в области математики, информатики и нейробиологии для исследования движения точек на ориентированных метрических графах, а также различных процессов путем их моделирования посредством таких графов.

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

Преимуществами программы являются:

* бесплатность, отсутствие встроенных покупок;
* простой интуитивно понятный интерфейс;
* англоязычный интерфейс;
* свободное распространение.

# **7. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ**

## **7.1. Необходимые стадии разработки, этапы и содержание работ**

1. Техническое задание
   1. Обоснование необходимости разработки программы
      1. Постановка задачи
      2. Выбор и обоснование критериев эффективности и качества разрабатываемой программы
   2. Научно-исследовательские работы
      1. Определение структуры входных и выходных данных
      2. Предварительный выбор методов решения задачи
   3. Разработка и утверждение технического задания
      1. Определение требований к программе
      2. Определение стадий, этапов и сроков разработки программы и документации на неё
      3. Согласование и утверждение технического задания
2. Технический проект

2.1 Разработка технического проекта

2.1.1 Уточнение структуры входных и выходных данных

2.1.2 Разработка алгоритма решения задачи

2.1.3 Определение формы представления входных и выходных данных

2.1.4 Разработка структуры программы

2.2 Утверждение технического проекта

2.2.1 Разработка пояснительной записки

* + 1. Согласование и утверждение технического проекта

1. Рабочий проект
   1. Разработка программы
      1. Программирование и отладка программы
   2. Разработка программных документов
      1. Разработка программных документов в соответствии с требованиями ГОСТ 19.101-77
   3. Испытания программы
      1. Разработка, согласование и утверждение программы и методики испытаний
      2. Корректировка программы и программной документации по результатам испытаний
2. Подготовка и передача программы
   1. утверждение даты защиты программного продукта;
   2. подготовка программы и программной документации для презентации и защиты;
   3. представление разработанного программного продукта руководителю и получение отзыва;
   4. загрузка Пояснительной записки в систему Антиплагиат через ЛМС НИУ ВШЭ;
   5. загрузка материалов курсового проекта (курсовой работы) в ЛМС, проект дисциплины «Курсовая работа 2019» (п. 5.2);
   6. Защита программного продукта (курсового проекта) комиссии.

## **7.2. Сроки и исполнители**

Разработка должна закончиться к 22 мая 2020 года.

Исполнитель: **Баранова Анастасия Андреевна**, студент группы БПИ196 факультета компьютерных наук НИУ ВШЭ.

# **8. ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ**

## **8.1. Виды испытаний**

Проверка программного продукта, в том числе и на соответствие техническому заданию, осуществляется исполнителем вместе с заказчиком согласно «Программе и методике испытаний», а также пункту 5.2.

## **8.2. Общие требования к приемке работы**

Общие требования к приемке работы описаны в документе «Программа и методика испытаний» (ГОСТ 19.301-79).

Защита выполненного проекта осуществляется комиссии, состоящей из преподавателей департамента программной инженерии, в утверждённые приказом декана ФКН сроки.

# **9. СПИСОК ЛИТЕРАТУРЫ**

1. Winfried Just. Chapter 6. Neuronal Networks: A Discrete Model // Mathematical Concepts and Methods in Modern Biology: Using Modern Discrete Models / Winfried Just, Sungwoo Ahn и David Terman (авторы), Robeva, R., Hodge, T. (редакторы). – USA: Academic Press, 2013. – 179-211 с.
2. ГОСТ 19.101-77 Виды программ и прогрaммных документов. //Единaя система прогрaммной документaции. – М.: ИПК Издaтельство стaндaртов, 2001. – 126 с
3. ГОСТ 19.102-77 Стадии разработки. //Единая система программной документации. – М.: ИПК Издaтельство стaндaртов, 2001. – 126 с.
4. ГОСТ 19.103-77 Обозначения программ и прогрaммных документов. //Единая система программной документации. – М.: ИПК Издaтельство стaндaртов, 2001. – 126 с.
5. ГОСТ 19.104-78 Основные надписи. //Единая система программной документации. – М.: ИПК Издaтельство стaндaртов, 2001. – 126 с.
6. ГОСТ 19.105-78 Общие требования к программным документам. //Единaя системa программной документaции. – М.: ИПК Издaтельство стaндaртов, 2001. – 126 с.
7. ГОСТ 19.106-78 Требования к программным документам, выполненным печaтным способом. //Единая система программной документации. – М.: ИПК Издaтельство стaндaртов, 2001. – 126 с.
8. ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издaтельство стaндaртов, 2001. – 126 с.
9. ГОСТ 19.603-78 Общие правила внесения изменений. //Единая система программной документации. – М.: ИПК Издaтельство стaндaртов, 2001. – 126 с.
10. ГОСТ 19.604-78 Правила внесения изменений в программные документы, выполненные печaтным способом. //Единая система программной документации. – М.: ИПК Издaтельство стaндaртов, 2001. – 126 с.
11. Евстигнеев В.А. Толковый словарь по теории графов в информатике и программировании / Евстигнеев В.А., Касьянов В.Н. – Россия, Наука, Сибирское предприятие РАН, 1999.
12. Калинин, Н. С. Модель пересыпания песка и дивизоры на графах [Электронный ресурс]: курс лекций — Электрон. дан. — Дубна: Летняя школа «Современная математика», 2017. — Режим доступа: <https://www.mccme.ru/dubna/2017/courses/kalinin.html>, свободный. (дата обращения: 14.05.20).

**ПРИЛОЖЕНИЕ 1**

**Терминология**

Ниже приведен список необходимых терминов для ознакомления.

**Вершина** – базовое понятие. Точка, где могут сходиться/выходить рёбра и/или дуги.

**Ребро** – базовое понятие. Ребро соединяет две вершины графа.

**Дуга** – ребро, имеющее направление. Упорядоченная пара вершин (v, w), где вершину v называют началом, а w - концом дуги. Можно сказать, что дуга v → w ведет от вершины v к вершине w, при этом вершина w смежная с вершиной v.

**Граф** – базовое понятие. Включает множество вершин и множество рёбер, являющееся подмножеством декартова квадрата множества вершин (то есть каждое ребро соединяет ровно две вершины).

**Орграф** – ориентированный граф G = (V, E) есть пара множеств, где V - множество вершин (узлов), E – множество дуг.

**Сильно связный граф** – ориентированный граф, в котором все вершины сильно связаны, т. е. существует путь из первой во вторую и из второй в первую и т. д.

**Метрический граф** – граф, каждое ребро которого имеет заданное вещественное время прохождения.

**Инцидентность** — отношение между ребром (дугой) и его концевыми вершинами, т.е. ребро e = (a,b) инцидентно вершинам a и b и вершины a, b инцидентны ребру e = (a,b).

**ПРИЛОЖЕНИЕ 2**

**Описание формата .dgmm**

Файл формата .dgmm содержит информацию об орграфе, созданном в программе, в сереализованном с использованием XML сериализации виде.

Программа позволяет открыть файл формата .dgmm, и при этом осуществляет корректное открытие только файлов данного формата, созданных или измененных в этой программе.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 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 |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |