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

Факультет компьютерных наук

Департамент программной инженерии

|  |  |
| --- | --- |
| СОГЛАСОВАНО  Кандидат технических наук, доцент департамента программной инженерии факультета компьютерных наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ К. Ю. Дегтярев  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2019 г. | УТВЕРЖДАЮ  Академический руководитель образовательной программы «Программная инженерия»  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В. Шилов  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_2019 г. |
| |  |  | | --- | --- | | ***Подп. и дата*** |  | | ***Инв. № дубл.*** |  | | ***Взам. Инв. №*** |  | | ***Подп. и дата*** |  | | ***Инв. № подл.*** | RU.17701729.04.0 9-01 33 01-1-ЛУ | | **Программа оценки стоимости разработки ПО с использованием нечетких деревьев решений**  **Руководство программиста**  **ЛИСТ УТВЕРЖДЕНИЯ**  **RU.17701729.501430-01 33 01-1-ЛУ**  Исполнитель  студент группы БПИ 173  \_\_\_\_\_\_\_\_\_\_\_/Королев Д. П./  «\_\_\_»\_\_\_\_\_\_\_\_\_\_2019 г.  **2019** | | |  |

УТВЕРЖДЕНО

RU.17701729.04.09-01 33 01-1-ЛУ

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | ***Подп. и дата*** |  | | ***Инв. № дубл.*** |  | | ***Взам. Инв. №*** |  | | ***Подп. и дата*** |  | | ***Инв. № подл.*** | RU.17701729.04.0 9-01 33 01-1 | | **Программа оценки стоимости разработки ПО с использованием**  **нечетких деревьев решений**  Руководство программиста  RU.17701729.04.09-01 34 01-1  **Листов 13**  **2019** |  |

**АННОТАЦИЯ**

Документ «Руководство программиста» адресован программистам, желающим использовать данную библиотеку в своих программах. В нем содержится информация об основных функциях, реализованных в библиотеке, а также способах работы с ними.

Руководство программиста состоит из пяти основных частей.

В разделе «Назначение и условия применения библиотеки» описаны назначение, функционал и минимальные требования библиотеки.

В разделе «Характеристики библиотеки» указаны её качественные характеристики.

В разделе «Обращение к библиотеке» содержится информация по подключению библиотеки и использованию ее методов в новых программах.

В разделе «Входные и выходные данные» описаны форматы входных и выходных данных, с которыми работают методы библиотеки.

В разделе «Сообщения» содержатся возможные ошибки и исключительные ситуации, возникающие при работе с библиотекой.

**СОДЕРЖАНИЕ**

[1. Назначение и условия применения библиотеки 4](#_Toc8397721)

[1.1. Назначение библиотеки 4](#_Toc8397722)

[1.2. Функции, выполняемые библиотекой 4](#_Toc8397723)

[1.3. Условия, необходимые для выполнения библиотеки 4](#_Toc8397724)

[1.3.1. Требования к составу и параметрам технических средств 4](#_Toc8397725)

[1.3.2. Требования к программным средствам 4](#_Toc8397726)

[2. Характеристики библиотеки 5](#_Toc8397727)

[2.1. Средства контроля правильности исполнения 5](#_Toc8397728)

[3. Обращение к библиотеке 6](#_Toc8397729)

[3.1. Инициализация основного контроллера 6](#_Toc8397730)

[3.2. Загрузка входных данных 6](#_Toc8397731)

[3.3. Построение нечеткого дерева решений 6](#_Toc8397732)

[3.4. Получение входных данных 6](#_Toc8397733)

[3.5 Вывод результатов на экран 6](#_Toc8397734)

[3.6 Примерная последовательность вызовов API 6](#_Toc8397735)

[4.Входные и выходные данные 8](#_Toc8397736)

[4.1. Входные данные 8](#_Toc8397737)

[4.2. Выходные данные 8](#_Toc8397738)

[5. Сообщения 9](#_Toc8397739)

[6. Приложения 10](#_Toc8397740)

[6.1. Список используемой литературы 10](#_Toc8397741)

# 1. Назначение и условия применения библиотеки

## 1.1. Назначение библиотеки

Библиотека предназначена для загрузки данных из файла, созданию нечеткого дерева решений, вычисления результата по нему.

## 1.2. Функции, выполняемые библиотекой

1. Инициализация контроллеров, управляющих процессов.
2. Выгрузка исходных данных в объекты.
3. Создание структуры данных «дерево» с соответствующей нечеткой логикой.
4. Получение результата по построенному дереву.

## 1.3. Условия, необходимые для выполнения библиотеки

### 1.3.1. Требования к составу и параметрам технических средств

Минимальные и рекомендуемые свойства компьютера:

1. Процессор Intel Core i3 с частотой 2000 МГц или более

быстрый (рекомендуется не менее 3000 МГц) с подходящей материнской платой.

1. Не менее 1 ГБ оперативной памяти (рекомендуется не менее 4 ГБ).
2. Не менее 1 ГБ свободного места на жестком диске.
3. Периферия для персонального компьютера, а также клавиатура.
4. Видеокарта и монитор, способных воспроизводить графическое отображение работы программы с разрешением не менее 800х600 точек [5]

### 1.3.2. Требования к программным средствам

1. операционная система Microsoft Windows 7 и новее;
2. установленный Microsoft .NET Framework 4.0.

# 2. Характеристики библиотеки

## 2.1. Средства контроля правильности исполнения

Правильность исполнения проверяется сравнением результатов работы программы с математически верными проведенными предварительно расчетами.

# 3. Обращение к библиотеке

Для использования методов библиотеки FuzzyTreeLib необходимо подключить её в соответствующем проекте.

## 3.1. Инициализация основного контроллера

Для использования основных методов библиотеки необходимо создать объект-контроллер *MainController*. Далее можно воспользоваться любыми открытыми API данного класса.

## 3.2. Загрузка входных данных

Для загрузки в объекты данные из файла нужно воспользоваться классом *DataLoader,* доступным из основного контроллера. В качестве приватного поля в данном классе прописан путь до файла, лежащего в соответствующей директории. Метод *LoadCocomoDataSet()* позволяет сделать это.

## 3.3. Построение нечеткого дерева решений

Для построения нечеткого дерева решений нужно воспользоваться классом *TreeController,* доступным из основного контроллера. Метод *ConstructTree(List<RefAtribut> refAtributs, bool toSort)* принимает список атрибутов, на основе которых можно построить дерево. Также можно отсортировать атрибуты, если это не было сделано прежде.

## 3.4. Получение входных данных

Для получения входных данных нужно воспользоваться классом *InputValue* при получении консольного ввода и классом *DataLoader,* доступным из основного контроллера. В каждом из классов нужно вызвать методы *ReadDouble()* и *FillInputsFromGUI* соответственно.

## 3.5 Вывод результатов на экран

Для вывода результатов на экран в классе *MainController* присутствует метод *GetDoubleResultValue()*. Этот метод возвращает результат всего эксперимента как число с плавающей точкой.

## 3.6 Примерная последовательность вызовов API

Ниже приведена последовательность вызовов API, позволяющих провести эксперимент с консольным вводом входных значений:

\_controllerMain = new MainController(NumberOfDecimals);

// loading sample from site

\_controllerMain.DataLoader.LoadSampleTest(\_controllerMain);

// Normalizing resultValues in Data

\_controllerMain.NormalizeResultValues();

// Reforming values

\_controllerMain.CountReferenceValuesFromDataValues();

\_controllerMain.CountEntropyForAllLexigraphics();

\_controllerMain.CountGrowthFunctionForAllLexigraphics();

Console.WriteLine(\_controllerMain.ToString());

// Counting result

/\* All needed done inside ResultCounter class:

\* 1) construct a tree

\* 2) get input

\* 3) count results

\*/

Double result = \_controllerMain.GetDoubleResultValue();

# 4.Входные и выходные данные

## 4.1. Входные данные

Входные данные поступают вводом пользователем значений из консоли или из графического интерфейса. Пункт 3.4 текущего документа описывает возможности получения входных данных.

## 4.2. Выходные данные

Выходные данные получены вызовом метода в основном контроллере. Пункт 3.5 текущего документа описывает возможности получения результата программно.

# 5. Сообщения

Библиотека не предусматривает сообщений об ошибках, так как подразумевает реализацию проверки корректности входных данных на этапе вызова API данной библиотеки.

# 6. Приложения

## 6.1. Список используемой литературы

1. Стоимостные оценки проекта. Материал свободной энциклопедии «Википедия» [Электронный ресурс] / URL: https://ru.wikipedia.org/Стоимостные\_оценки\_проекта (Дата обращения 22.11.2018, режим доступа: свободный).
2. Лингвистические переменные. Материал свободной энциклопедии «Википедия» [Электронный ресурс] / URL: https://ru.wikipedia.org/Лингвистические переменные (Дата обращения 22.11.2018, режим доступа: свободный).
3. Основной алгоритм построения нечеткого дерева решений. Материал статьи журнала «Mathware & Soft Computing Magazine», том 20, выпуск №1, январь 2015.
4. Единая система программной документации – М.: ИПК Издательство стандартов, 2000.
5. Системные требования для операционных систем Windows ХР [Электронный ресурс] / URL: https://support.microsoft.com/ru-ru/kb/314865 (Дата обращения 20.02.2015, режим доступа: свободный).
6. Факторы, влияющие на стоимость разработки проекта и сроки его выполнения (на англ. языке). Материал статьи конференции «2017 Sudan Conference on Computer Science and Information Technology (SCCSIT)», дата 20 октября 2017.
7. Анализ включения внешних факторов стоимости при оценке стоимости программного обеспечения в базу данных (на англ. языке). Материал статьи конференции «2018 IEEE International Conference on Software Quality, Reliability and Security Companion», дата 20 июля 2018.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | |
| Изм. | Номера листов (страниц) | | | | Всего листов (страниц) в документе | № документа | Входящий № сопроводительного документа и дата | Подпись | Дата |
| измененных | Замененных | новых | аннулированных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |