

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по учебной практике
ТЕМА: «Алгоритм Борувки»

Студент гр. 9383	_____	Рыбников Р.А.
Студентка гр. 9383	_____	Сергиенкова А.А.
Студент гр. 9383	_____	Крейсманн К.В.
Руководитель	_____	Ефремов М.А.

Санкт-Петербург
2021

ЗАДАНИЕ НА УЧЕБНУЮ ПРАКТИКУ

Студент Рыбников Р.А. группы 9383

Студентка Сергиенкова А.А. группы 9383

Студент Крейсманн К.В. группы 9383

Тема практики: Алгоритм Борувки

Задание на практику:

Командная разработка визуализации алгоритма на языке Java с графическим интерфейсом.

Алгоритм: Алгоритм Борувки.

Дата сдачи отчёта:

Дата защиты отчёта:

Студент гр. 9383

Рыбников Р.А.

Студентка гр. 9383

Сергиенкова А.А.

Студент гр. 9383

Крейсманн К.В.

Руководитель

Ефремов М.А.

АННОТАЦИЯ

Целью учебной практики является разработка графического приложения для нахождения минимального оставного дерева для заданного графа с помощью алгоритма Борувки.

Программа разрабатывается на языке Java, командой из трёх человек, каждый из которых имеет определённую специализацию.

SUMMARY

The aim of the training practice is to develop a graphical application for finding the minimum abandoned tree for a given graph using Boruvka's algorithm.

The program was developed in the Java language by a team of three people, each of whom has a specific specialization.

СОДЕРЖАНИЕ

Введение

1.	Требования к программе.....	6
1.1.	Исходные требования к программе.....	6
1.2.	Уточнение требований после сдачи прототипа.....	6
2.	План разработки и распределение ролей в бригаде.....	7
2.1.	План разработки.....	7
2.2.	Распределение ролей в бригаде.....	7
3.	Особенности реализации.....	8

ВВЕДЕНИЕ

Целью данной практической работы является разработка графического приложения, выполняющего визуализацию работы алгоритма Борувки. Пользователю программы должна быть предоставлена возможность самостоятельно задать входные данные для алгоритма с помощью графического интерфейса. Результат работы алгоритма должен иметь графическое отображение. Должна быть предоставлена возможность просмотра итогового результата алгоритма и просмотра хода его исполнения по шагам.

Разработка осуществляется на языке Java, командой из трёх человек, каждый из которых имеет определённую специализацию.

1. ТРЕБОВАНИЯ К ПРОГРАММЕ

1.1. Исходные требования к программе

Программа представляет собой визуализацию алгоритма Борувки, нахождения минимального остовного дерева для взвешенного неориентированного графа.

Требования к вводу исходных данных

Для задания графа будут реализованы несколько возможностей: задание графа посредством взаимодействия с графическими элементами; случайная генерация графа.

Требования к визуализации

Пользователю должно быть доступно графическое изображение графа, взаимодействие с ним средствами графического интерфейса, просмотр состояний графа на каждом шаге алгоритма и просмотр конечного результата.



Рисунок 1 – use-case диаграмма.

2. ПЛАН РАЗРАБОТКИ И РАСПРЕДЕЛЕНИЕ РОЛЕЙ В БРИГАДЕ

2.1. План разработки

- Распределение ролей, составление диаграммы прецедентов программы, создана директория с исходным кодом.
Срок выполнения: 02.07.2021
- Создание интерфейса, но пока не рабочего, проектирование классов программы.
Срок выполнения: 04.07.2021
- Реализация случайной генерации графа, реализация алгоритма с отображением результата работы, составить план тестирования.
Срок выполнения: 06.07.2021
- Сделан прототип программы в котором визуализируется как получение и отображение результата сразу, так и пошаговое выполнение алгоритма.
Срок выполнения: 08.07.2021
- Проект полностью готов, программа корректно собирается.
Срок выполнения: 10.07.2021

2.2. Распределение ролей в бригаде

Рыбников Р.А. – Реализация алгоритма, логики программы, документации и тестирования.

Сергиенкова А.А. – Работа с алгоритмом, документация, тестирование.

Крейсманн К.В – Работа с логикой взаимодействия с графическим интерфейсом, основная логика взаимодействия интерфейса с алгоритмом, тестирование.

3.ОСОБЕННОСТИ РЕАЛИЗАЦИИ

3.1. Структуры данных

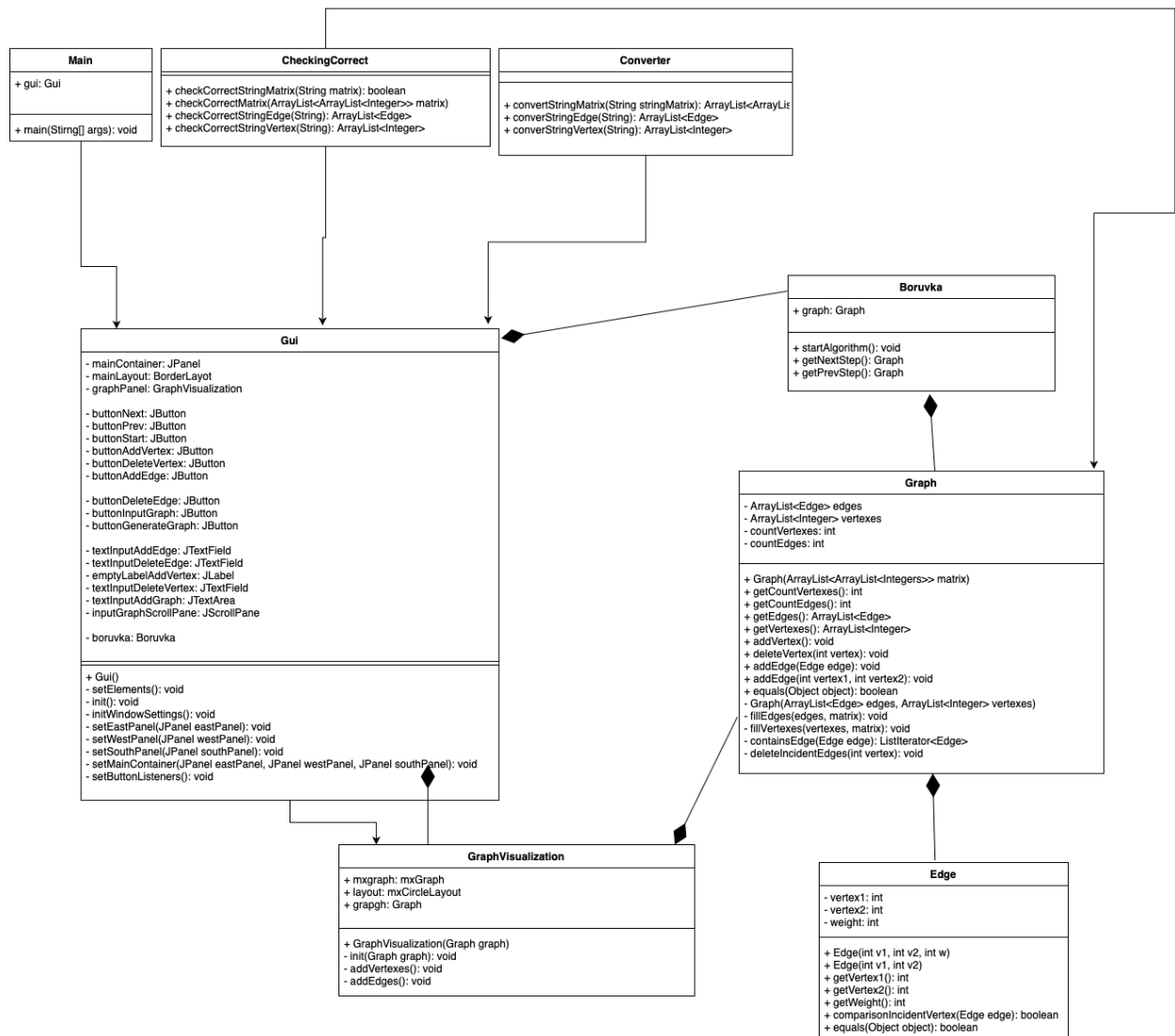


Рисунок 2 – UML диаграмма классов.