

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

ОТЧЕТ
по учебной практике
Тема: Командная визуализация алгоритма Борувки

Студент гр. 8382	_____	Мирончик П.Д.
Студент гр. 8382	_____	Синельников М.Р.
Студент гр. 8382	_____	Янкин Д.О.
Руководитель	_____	Ефремов М.А.

Санкт-Петербург
2020

ЗАДАНИЕ НА УЧЕБНУЮ ПРАКТИКУ

Студент Мирончик П.Д. группы 8382

Студент Синельников М.Р. группы 8382

Студент Янкин Д.О. группы 8382

Тема практики: Командная визуализация алгоритма Борувки

Задание на практику:

Командная итеративная разработка визуализатора алгоритма на Kotlin с графическим интерфейсом.

Алгоритм: Борувки.

Сроки прохождения практики: 29.06.2020 – 12.07.2020

Дата сдачи отчета: 00.07.2020

Дата защиты отчета: 00.07.2020

Студент	_____	Мирончик П.Д.
Студент	_____	Синельников М.Р.
Студент	_____	Янкин Д.О.
Руководитель	_____	Ефремов М.А.

АННОТАЦИЯ

Целью учебной практики является разработка приложения для визуализации алгоритма Борувки. Пользователю предоставляются инструменты для рисования графа и возможность пошагового просмотра работы реализуемого алгоритма. Приложение должно обладать доступным и удобным интерфейсом.

Разработка ведется на языке Kotlin с использованием JavaFX для создания графического интерфейса. Разработка ведется командой из трех человек, за которыми закреплены определенные роли. Выполнение работы осуществляется поэтапно.

SUMMARY

Academia practice aim is to develop application for visualizing Boruvka's algorithm. The user is provided with tools of drawing graph and has opportunity to see all steps of realized algorithm. The application should provide accessible and intuitive interface.

Development is done on Kotlin programming language using JavaFX framework for creating graphical interface. Development team consists of three people, each of them responsible for certain task. Development is done step by step.

СОДЕРЖАНИЕ

	Введение	5
1.	Требования к программе	6
1.1.	Исходные требования к программе	6
1.1.1.	Требования к вводу исходных данных	6
1.1.2.	Требования к визуализации	6
2.	План разработки и распределение ролей в бригаде	7
2.1.	План разработки	7
2.2.	Распределение ролей в бригаде	7

ВВЕДЕНИЕ

Целью учебной практики является реализация приложения для визуализации алгоритма Борувки для нахождения минимального остовного дерева в графе. Пользователю должна быть предоставлена возможность задания графа посредством взаимодействия с графическими элементами. Результат работы алгоритма должен иметь графическое отображение. Должна быть предоставлена возможность просмотра итогового результата алгоритма и просмотра хода его исполнения по шагам.

Работа выполняется командой из трех человек, за каждым из которых закреплены ответственности: за разработку графического интерфейса, за разработку логики алгоритма, за проведение тестирования и сборку проекта.

Необходимо провести модульное тестирование программы. Разработанная программа должна собираться в jar-архив.

1. ТРЕБОВАНИЯ К ПРОГРАММЕ

1.1. Исходные Требования к программе

1.1.1. Требования к вводу исходных данных

Граф должен задаваться рисованием посредством взаимодействия с графическими элементами: интерактивным добавлением вершин, ребер, назначением весов.

1.1.2. Требования к визуализации

Пользователю должно быть доступно графическое изображение графа, интерактивное взаимодействие с графом (перемещение, добавление или удаление элементов), выполнение алгоритма по команде, просмотр состояний графа на каждом шаге алгоритма и просмотр итогового результата.

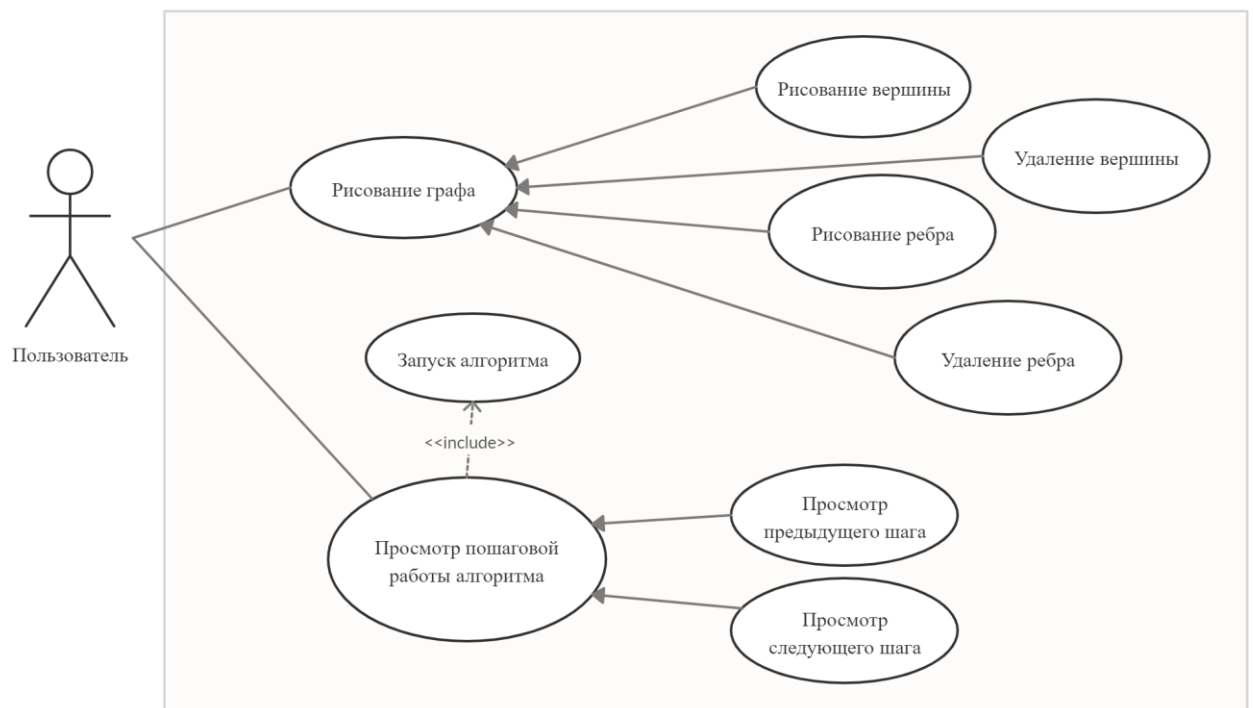


Рисунок 1. Диаграмма прецедентов разрабатываемой программы

2. ПЛАН РАЗРАБОТКИ И РАСПРЕДЕЛЕНИЕ РОЛЕЙ В БРИГАДЕ

2.1. План разработки

К 02.07.2020 должны быть распределены роли между членами бригады, составлена диаграмма прецедентов программы, а также создана директория с исходным кодом и скриптом сборки.

К 04.07.2020 должны быть размещены все элементы интерфейса с использованием заглушек, составлена UML-диаграмма классов программы с пояснением принятых решений.

К 06.07.2020 необходимо реализовать случайную генерацию изначальных графов по параметрам, задаваемых пользователем, с проверкой корректности входных данных; запуск алгоритма из графического интерфейса с отображением конечного результата работы алгоритма; добавить в отчет описание алгоритма, составить UML-диаграмму состояний и план тестирования.

К 08.07.2020 должна быть добавлена возможность визуализации как пошагового выполнения алгоритма, так и просмотра итогового результата; должны быть проведены тесты для созданных структур данных и функций алгоритма согласно плану тестирования; в отчет добавлено описание алгоритма пошагового отображения работы алгоритма.

К 10.07.2020 проект должен быть полностью готов, программа должна корректно собираться, в ходе сборки должны выполняться и успешно завершаться модульные тесты.

2.2. Распределение ролей в бригаде

Мирончик П.Д. отвечает за разработку графического интерфейса.

Янкин Д.О. отвечает за реализацию логики алгоритма.

Синельников М.Р. отвечает за тестирование и сборку приложения.