

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

ОТЧЁТ
по учебной практике
Тема: Визуализация алгоритма Форда-Беллмана.

Студент гр. 9383	_____	Камзолов Н.А.
Студентка гр. 9383	_____	Гладких А.А.
Студент гр. 9383	_____	Моисейченко К.А.
Руководитель	_____	Фиалковский М.С.

Санкт-Петербург

2021

ЗАДАНИЕ НА УЧЕБНУЮ ПРАКТИКУ

Студент Камзолов Н.А. группы 9383

Студентка Гладких А.А. группы 9383

Студент Моисейченко К.А. группы 9383

Тема практики: Визуализация алгоритма Форда-Беллмана

Задание на практику:

Командная итеративная разработка визуализатора алгоритма на Kotlin с графическим интерфейсом.

Алгоритм: алгоритм Форда-Беллмана

Сроки прохождения практики: 1.07.2021 – 14.07.2021

Дата сдачи отчета: __.07.2021

Дата защиты отчета: __.07.2021

Студент гр. 9383		Камзолов Н.А.
Студентка гр. 9383		Гладких А.А.
Студент гр. 9383		Моисейченко К.А.
Руководитель		Фиалковский М.С.

АННОТАЦИЯ

Основная цель практики – изучение основ языка программирования Kotlin и разработка приложения с графическим интерфейсом на Kotlin (визуализатора алгоритма Форда-Беллмана). В процессе работы предстоит реализовать алгоритм Форда-Беллмана, разработать прототип интерфейса приложения, протестировать написанную программу и исправить найденные в ней ошибки. Нашей командой было решено разработать визуализацию алгоритма в качестве Android-приложения.

SUMMARY

The main goal of the practice is to learn basics of Kotlin programming language and develop an application with a graphical interface in Kotlin (Ford-Bellman algorithm visualizer). In the work we will have to implement the Ford-Bellman algorithm, develop a prototype of the application interface, test the program and fix the errors if found. Our team decided to develop a visualization of the algorithm as an Android application.

СОДЕРЖАНИЕ

	Введение	5
1.	Требования к программе	6
1.1.	Описание задачи	6
1.2.	Архитектура проекта	6
1.3.	Формат входных и выходных данных	6
1.4.	Основные типы данных	6
1.5.	Графический интерфейс	7
2.	План разработки и распределение ролей в бригаде	11
2.1.	План разработки	11
2.2.	Распределение ролей в бригаде	11

ВВЕДЕНИЕ

Задача практики состоит в разработке Android-приложения, визуализирующего работу алгоритма Форда-Беллмана. Граф задаётся пользователем. Алгоритм находит кратчайшие пути от одной из вершин графа до всех остальных. Граф и начальная вершина задаются пользователем.

1. ТРЕБОВАНИЯ К ПРОГРАММЕ

1.1. Описание задачи

Приложение будет показывать работу алгоритма Форда-Беллмана по шагам. В приложении можно будет редактировать граф (добавлять, удалять вершины и рёбра) и выбирать начальную вершину для поиска кратчайшего пути.

1.2. Архитектура проекта

Основной архитектурный паттерн - MVVM (Model-View-ViewModel). Таким образом для каждого экрана приложения создан:

1. Layout файл, который отвечает за графическое отображение всех элементов UI(View).
2. Класс ViewModel, который отвечает за всю логику экрана и хранение данных на данный момент времени (ViewModel).
3. Класс Fragment, который отвечает за обновление UI, является посредником между View и ViewModel.

1.3. Формат входных и выходных данных

В качестве входных данных принимается заданный пользователем в режиме редактирования граф, а также начальная вершина для поиска кратчайших путей.

1.4. Основные типы данных

1. Список инцидентности для работы с графом в алгоритме. Представляет из себя `map<String, Neighbours>`, где `Neighbours` - массив пар из имени соседней вершины и длине ребра, проведенного к этой вершине.

2. Список инцидентности для корректного отображения графа. Представляет из себя `map<String, VertexInfo>`, но где `VertexInfo` – это большая структура данных для хранения информации об отображении вершины и всех

инцидентных ей ребер. При вводе графа пользователем, вся информация об отображении графа попадает именно сюда.

1.5. Интерфейс

Интерфейс программы состоит из:

Начальное окно с кнопками начала работы с алгоритмом («Опробовать алгоритм»), подробного пояснения алгоритма («Как работает?»), список разработчиков («Разработчики») и кнопкой выхода из приложения.



Рисунок 1 – Прототип интерфейса начального окна



Рисунок 2 – Прототип окна пояснения алгоритма

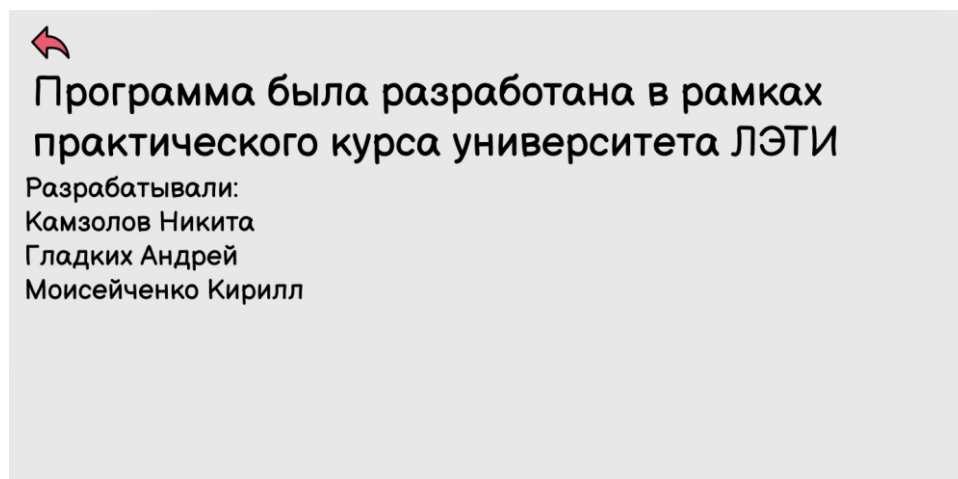


Рисунок 3 – Прототип окна «Разработчики»

Интерфейс основного окна состоит из кнопок «Назад», «Шаги алгоритма», «Помощь», «Редактировать граф», «Удалить вершину», «Удалить ребро», «Завершить редактирование».

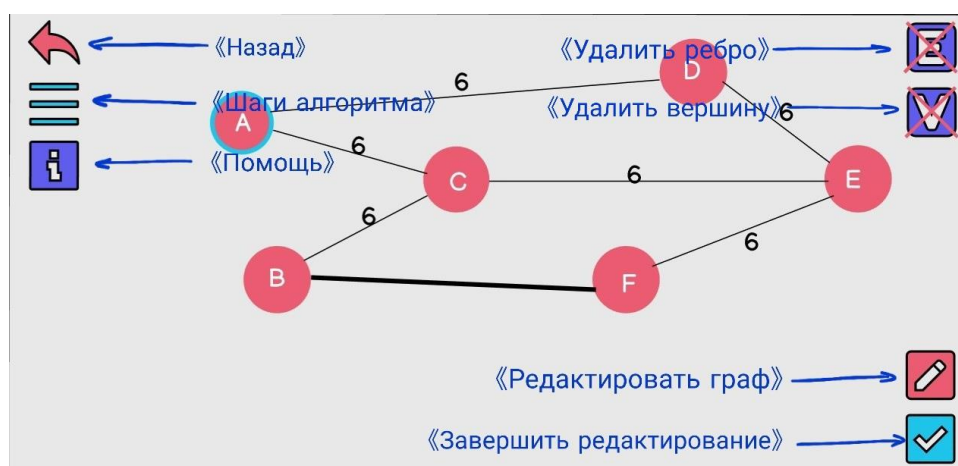


Рисунок 4 – Прототип интерфейса окна работы алгоритма

Рисунок 5 – Прототип интерфейса при нажатии кнопки «Помощь»

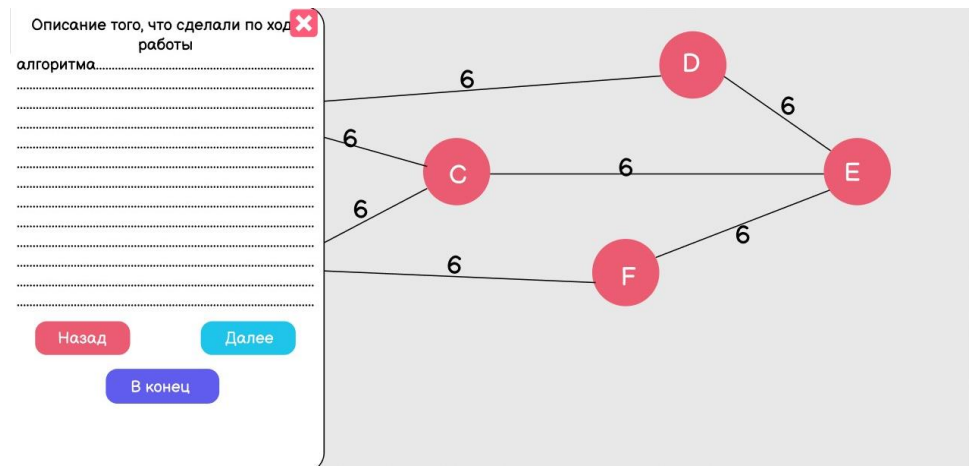


Рисунок 6 – Прототип интерфейса при нажатии кнопки «Шаги алгоритма»

Пользователь задаёт граф, добавляя новые вершины и рёбра в режиме редактирования.

Для того, чтобы добавить новую вершину, необходимо нажать на любое место на экране, а затем ввести имя вершины во всплывающем окне.



Рисунок 7 – Всплывающее окно для ввода имени новой вершины

При нажатии на уже существующую вершину, появляется кнопка «Удалить вершину».

Чтобы добавить новое ребро, необходимо нажать сначала на начальную, затем на конечную вершину и ввести вес ребра во всплывающем окне.

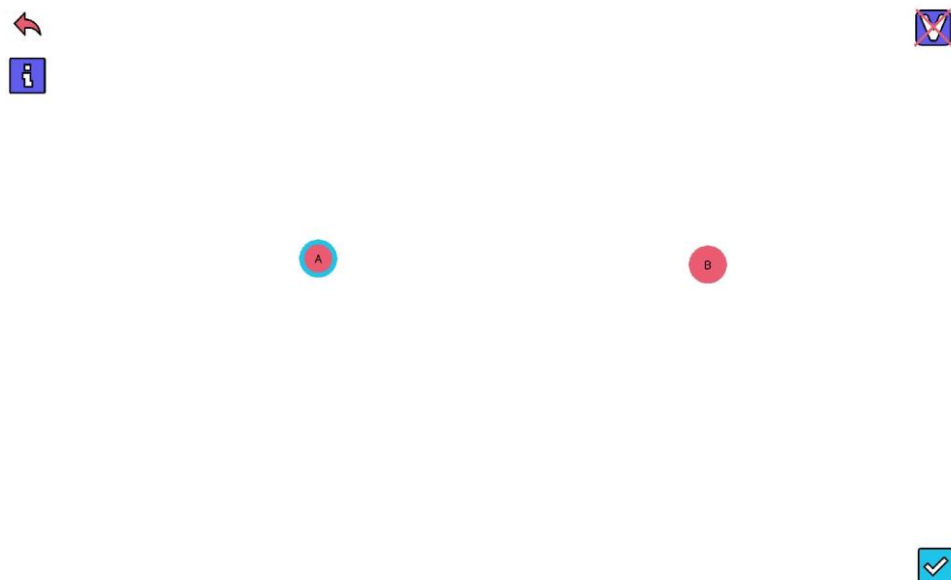


Рисунок 8 – Нажатие на уже существующую вершину



Рисунок 9 – Всплывающее окно для ввода веса нового ребра



Рисунок 10 – Добавление нового ребра

При выборе уже существующего ребра, появляется кнопка «Удалить ребро».



Рисунок 11 – Выбор уже существующего ребра

2. ПЛАН РАЗРАБОТКИ И РАСПРЕДЕЛЕНИЕ РОЛЕЙ

2.1. План разработки

До 02.07.2021 – Распределение по бригадам и выбор темы мини-проекта

До 05.07.2021 – Сдача вводного задания

До 07.07.2021 – Сдача прототипа графического интерфейса

До 09.07.2021 – Сдача первого этапа

До 12.07.2021 – Сдача второго этапа

До 14.07.2021 – Сдача финальной версии и отчёта

2.2. Распределение ролей в команде

Камзолов Н.А. – лидер, фронтенд, ответственный за android архитектуру

Гладких А.А. – алгоритмист, тестировщик

Моисейченко К.А. – алгоритмист, документация