

МИНОБРНАУКИ РОССИИ
Санкт-Петербургский государственный
электротехнический университет
«ЛЭТИ» им. В.И. Ульянова (Ленина)
Кафедра МО ЭВМ

ОТЧЕТ
по учебной практике
Тема: «Поиск компонент сильной связности»

Студентка гр. 8304	_____	Николаева М. А.
Студентка гр. 8304	_____	Мельникова О. А.
Студент гр. 8304	_____	Щука А. А.
Руководитель	_____	Фирсов М. А.

Санкт-Петербург

2020

ЗАДАНИЕ

на учебную практику

Студентка Николаева М. А. гр. 8304

Студентка Мельникова О. А. гр. 8304

Студент Щука А. А. гр. 8304

Тема практики: Поиск компонент сильной связности

Задание на практику:

Пользователь с помощью графического интерфейса конструирует невзвешенный ориентированный граф. Пользователь может добавлять/удалять вершины графа. Помимо этого, пользователь может задавать ребра в графе, нажав на одну вершину и потянув указатель мыши к другой вершине. После запуска алгоритма программа визуализирует алгоритм, а также выводит текстовые данные, поясняющие ход выполнения алгоритма. Также входные данные могут быть получены из файла.

Алгоритм: алгоритм Косарайю.

Дата сдачи отчета: __.07.2020

Дата защиты отчета: __.07.2020

Студентка гр. 8304

Николаева М. А.

Студентка гр. 8304

Мельникова О. А.

Студент гр. 8304

Щука А. А.

Руководитель

Фирсов М. А.

АННОТАЦИЯ

Целью работы является получения навыков работы с такой парадигмой программирования, как объектно-ориентированное программирование. Для получения данных знаний выполняется один из вариантов мини-проекта. В процессе выполнения мини-проекта необходимо реализовать графический интерфейс к данной задаче, организовать ввод и вывод данных с его помощью, реализовать сам алгоритм, научиться работать в команде. В данной работе в качестве мини-проекта выступает поиск компонент сильной связности (визуализация алгоритма Косарайю). Также при разработке выполняется написание тестирования, для проверки корректности алгоритма.

СОДЕРЖАНИЕ

АННОТАЦИЯ.....	3
ВВЕДЕНИЕ.....	5
1. ТРЕБОВАНИЯ К ПРОГРАММЕ.....	6
1.1 Исходные требования к программе.....	6
1.1.1 Требования к входным данным.....	6
1.1.2 Требования к визуализации	6
1.1.3 Требования к алгоритму и данным	7
1.1.4 Требования к выходным данным	7
1.2 Требования к программе после уточнения работы	7
1.2.1 Требования к входным данным.....	7
1.2.2 Требования к визуализации	7
1.2.3 Требования к алгоритму и данным	7
1.2.4 Требования к выходным данным	7
2. ПЛАН РАЗРОБОТКИ И РАСПРЕДЕЛЕНИЕ РОЛЕЙ В БРИГАДЕ	7
2.1 План разработки.....	7
2.2 Распределение ролей в бригаде	8
3. ОСОБЕННОСТИ РЕАЛИЗАЦИИ	9
3.1 Используемые структуры данных	9
3.1 Основные методы.....	9
4. ТЕСТИРОВАНИЕ	9
4.1 Написание UNIT Tests	9
4.2 Ручное тестирование программы	9
ЗАКЛЮЧЕНИЕ	9
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	9
ПРИЛОЖЕНИЕ А. UML ДИАГРАММА	9
ПРИЛОЖЕНИЕ Б. ИСХОДНЫЙ КОД.....	9

ВВЕДЕНИЕ

Основная цель практики – реализация мини-проекта, который является визуализацией алгоритма. В данной работе это алгоритм Косарайю. Для выполнения этой цели были поставлены задачи: реализация алгоритма, разработка GUI к проекту, написание тестирования.

Алгоритм Косарайю (в честь американского учёного индийского происхождения Самбасивы Рао Косарайю) — алгоритм поиска областей сильной связности в ориентированном графе. Чтобы найти области сильной связности, сначала выполняется поиск в глубину (DFS) на обращении исходного графа (то есть против дуг), вычисляя порядок выхода из вершин. Затем мы используем обращение этого порядка, чтобы выполнить поиск в глубину на исходном графе (в очередной раз берём вершину с максимальным номером, полученным при обратном проходе). Деревья в лесе DFS, которые выбираются в результате, представляют собой сильные компоненты связности.

1. ТРЕБОВАНИЯ К ПРОГРАММЕ

1.1 Исходные требования к программе

1.1.1 Требования к входным данным

Для корректной работы алгоритма требуется:

- множество вершин графа
- множество ребер графа

1.1.2 Требования к визуализации

Программа должна обладать простым и понятным интерфейсом.

Прототип интерфейса представлен на рисунке 1.

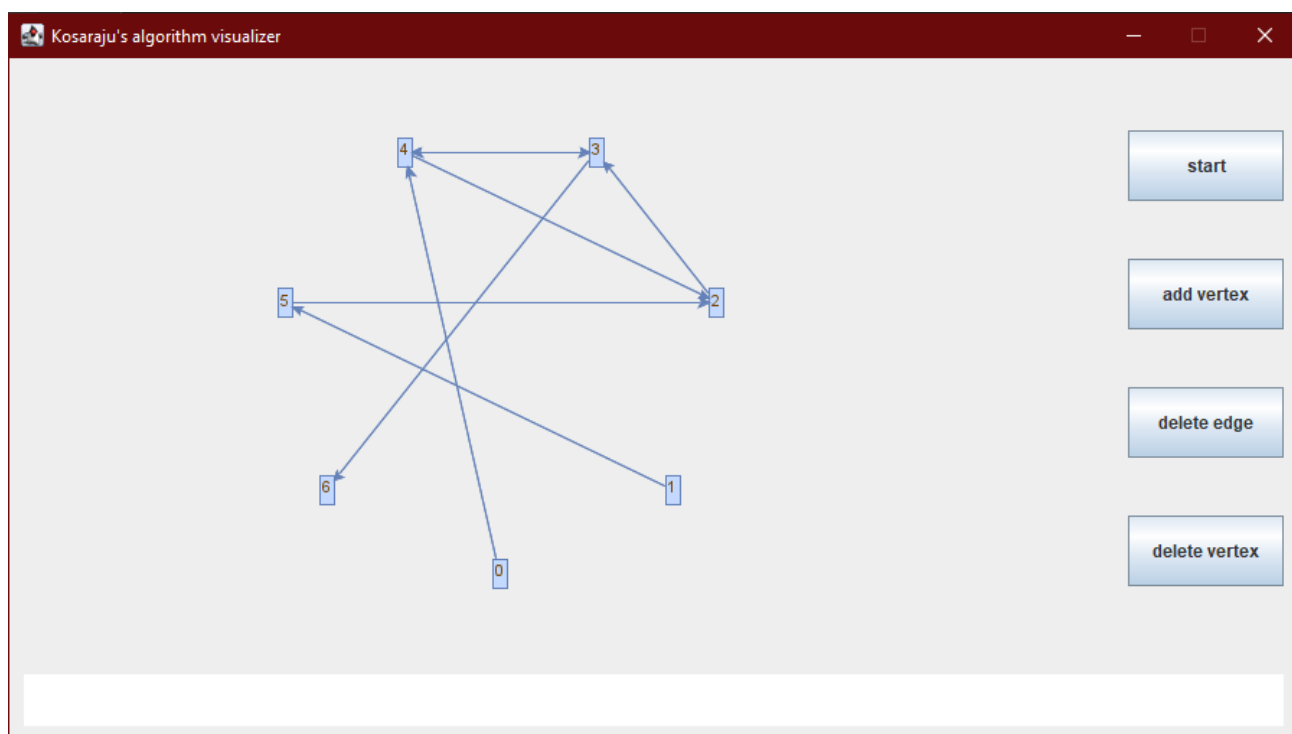


Рисунок 1 - Прототип интерфейса

Программа имеет интерактивную область с графом с возможностью выбора вершин, ребер и запуска алгоритма. С правой стороны расположены функциональные кнопки, снизу расположено поле для вывода текстовой информации для пояснения алгоритма.

1.1.3 Требования к алгоритму и данным

Алгоритм получает на вход граф, заданный пользователем, и в процессе выполнения передает промежуточные состояния графа для визуализации.

1.1.4 Требования к выходным данным

Выходные данные: компоненты сильной связности в исходном графе.

1.2 Требования к программе после уточнения работы

1.2.1 Требования к входным данным

1.2.2 Требования к визуализации

1.2.3 Требования к алгоритму и данным

1.2.4 Требования к выходным данным

2. ПЛАН РАЗРОБОТКИ И РАСПРЕДЕЛЕНИЕ РОЛЕЙ В БРИГАДЕ

2.1 План разработки

1. Обсудить задание, распределить роли, выбрать необходимые средства разработки и структуры данных. Данный пункт задания необходимо выполнить до 1 июля 2020 года.

2. Создать прототип GUI. Данный пункт задания необходимо выполнить к 2 июля 2020 года.

3. Реализовать структуры данных, необходимые для алгоритма. Данный пункт задания необходимо выполнить к 3 июля 2020 года.

4. Реализовать алгоритм без использования GUI. Данный пункт задания необходимо выполнить к 4 июля 2020 года.

5. Добавить JavaDoc комментарии для генерации документации к алгоритму и структуре данных. Данный пункт задания необходимо выполнить до 5 июля 2020 года.

6. Реализация основного GUI. Данный пункт задания необходимо выполнить к 6 июля 2020 года.

7. Связывание структур данных алгоритма и визуализации. Данный пункт задания необходимо выполнить к 7 июля 2020 года.

8. Первичная сборка проекта и первичное тестирование его функций. Данный пункт задания необходимо выполнить к 8 июля 2020 года.

9. Отладка ошибок, улучшение GUI и добавление возможности считывания входных данных из файла. Данный пункт задания необходимо выполнить к 9 июля 2020 года.

10. Окончательная сборка проекта и добавление полноценного тестирования всех модулей программы. Данный пункт задания необходимо выполнить до 11 июля 2020 года.

2.2 Распределение ролей в бригаде

- Николаева М. А.:
 - Тестирование программы;
 - Реализация ввода-вывода;
 - Проектирование, организация командной работы.
- Щука А. А.:
 - Создание основного GUI;
 - Объединение отдельных модулей программы;
 - Рефакторинг кода.
- Мельникова О. А.:
 - Создание алгоритма и структур данных;
 - Оформление пояснительной записки;
 - Расширение возможностей GUI.

3. ОСОБЕННОСТИ РЕАЛИЗАЦИИ

3.1 Используемые структуры данных

3.1 Основные методы

4. ТЕСТИРОВАНИЕ

4.1 Написание UNIT Tests

4.2 Ручное тестирование программы

ЗАКЛЮЧЕНИЕ

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

ПРИЛОЖЕНИЕ А. UML ДИАГРАММА

ПРИЛОЖЕНИЕ Б. ИСХОДНЫЙ КОД