**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МОЭВМ**

**ОТЧЕТ**

**по учебной практике**

**“Визуализация алгоритма поиска в глубину в ориентированном графе на языке Java. ”**

|  |  |  |
| --- | --- | --- |
| Студентка гр. 5381 |  | Боровикова О.В. |
| Студентка гр. 5381 |  | Буздина М.А |
| Студентка гр. 5381 |  | Кочнева О.Р. |
| Руководитель |  | Фирсов М.А. |

Санкт-Петербург

2017

**ЗАДАНИЕ**

**на учебную практику**

|  |  |  |
| --- | --- | --- |
| Студентка группы 5381 Боровикова О.В. | | |
| Студентка группы 5381 Буздина М.А. | | |
| Студентка группы 5381 Кочнева О.Р. | | |
| Тема практики: визуализация алгоритмов на языке Java | | |
| Задание на практику:  Командная итеративная разработка визуализатора алгоритма на Java с графическим интерфейсом.  Алгоритм: обход ориентированного графа в глубину. | | |
| Сроки прохождения практики: 21.06.2017 – 04.07.2017 | | |
| Дата сдачи отчета: 00.06.2017 | | |
| Дата защиты отчета: 00.06.2017 | | |
|  | | |
| Студентка гр.5381 |  | Боровикова О.В. |
| Студентка гр.5381 |  | Буздина М.А. |
| Студентка гр.5381 |  | Кочнева О.Р. |
| Руководитель |  | Фирсов М.А. |

# Аннотация

## Кратко (в 8-10 строк) указать цель и основное содержание практики.

## 

## 

## 

## 

## 

## 

## 

## 

# Summary

## Briefly (8-10 lines) to describe the the purpose and main contents of the practice work.

## 

## Содержание

[**Введение**](#_usp4xiuwfpq)5

[**1. ТРЕБОВАНИЯ К ПРОГРАММЕ**](#_64n3q5feev0r)7

[1.1. Исходные Требования к программе](#_bocpv7m2495l) 7

[1.2. Второй подраздел первого раздела](#_asenufewjjel) 7

[**2. ПЛАН РАЗРАБОТКИ И РАСПРЕДЕЛЕНИЕ РОЛЕЙ В БРИГАДЕ**](#_ktccdg7vely1) **8**

[2.1. План разработки](#_m9i2oodwv98d) 8

[2.2. Распределение ролей в бригаде](#_12yty5lld6ky) 8

[**3. ОСОБЕННОСТИ РЕАЛИЗАЦИИ**](#_uo131tuddt4h) **10**

[3.1 Использованные структуры данных](#_c6nxh4n9kb61) 10

[3.2 Основные методы](#_1dduxx9qwwgv) 10

[3.3 Спецификация](#_1dduxx9qwwgv) 10

[3.3.1 Описание интерфейса](#_79q1p9oqfmqv) 10

[3.3.2. Описание входных и выходных данных](#_p3wx0xfrz1hj) 11

## 

# Введение

Поиск в глубину – один из методов обхода графа. Стратегия поиска в глубину, как и следует из названия, состоит в том, чтобы идти «вглубь» графа, насколько это возможно. Алгоритм поиска описывается рекурсивно: перебираем все исходящие из рассматриваемой вершины рёбра. Если ребро ведёт в вершину, которая не была рассмотрена ранее, то запускаем алгоритм от этой нерассмотренной вершины, а после возвращаемся и продолжаем перебирать рёбра. Возврат происходит в том случае, если в рассматриваемой вершине не осталось рёбер, которые ведут в нерассмотренную вершину. Если после завершения алгоритма не все вершины были рассмотрены, то необходимо запустить алгоритм от одной из нерассмотренных вершин. В нашей работе будет реализован визуализатор данного алгоритма. При этом будет присутствовать графический интерфейс.

# 1. ТРЕБОВАНИЯ К ПРОГРАММЕ

## 1.1 Исходные Требования к программе

При запуске проекта будет создаваться отдельное окно. Данное окно будет иметь 5 полей:

1. Поле графического представления графа.

Граф строится по данным из файла автоматически при запуске программы, вершины размещаются по окружности; если файл пуст или отсутствует, то поле представление графа при запуске будет пустым.

1. Поле с выводом результата.
2. Поле с выводом промежуточных результатов.
3. Поле входных данных, содержащее представление исходного графа.
4. Поле интерфейса создания графа:

* добавить вершину (вершина добавляется в центр графического поля, затем перемещается пользователем мышкой)
* добавить ребро
* старт
* следующий шаг
* результат
* очистить поля

При нажатии кнопки следующий шаг происходят действия: в поле промежуточных выводов указывается состояние вершины; на графе состояние вершины отображается соответствующим цветом (1 цвет – посещённая вершина, 2 цвет – использованная вершина) и номером порядка посещения и использования. При переходе к следующей вершине ребро, соединяющее вершины, будет подсвечено в цвет 3. При возврате ребро будет подсвечено в цвет 4.

Примерный вид интерфейса представлен на рис.1



Рисунок 1. Интерфейс программы. (Построен на сайте draw.io)

1.2 Изменения к спецификации

Для удобства пользователя было решено ввести кнопку «Выбрать файл», для считывания информации из файла.

Также в окне начальных данных граф представляется не в виде матрицы смежности, а в виде списков смежности.

### 1.3 Описание входных и выходных данных

Входные данные:

Граф, записанный в виде матрицы смежности.

Выходные данные:

Графическое представление графа с подсветкой всех посещенных вершин и соответствующих пройденных ребер. Вывод пошаговой работы алгоритма в текстовое окно. Вывод порядка обхода в глубину.

## 1.3 Пример работы программы

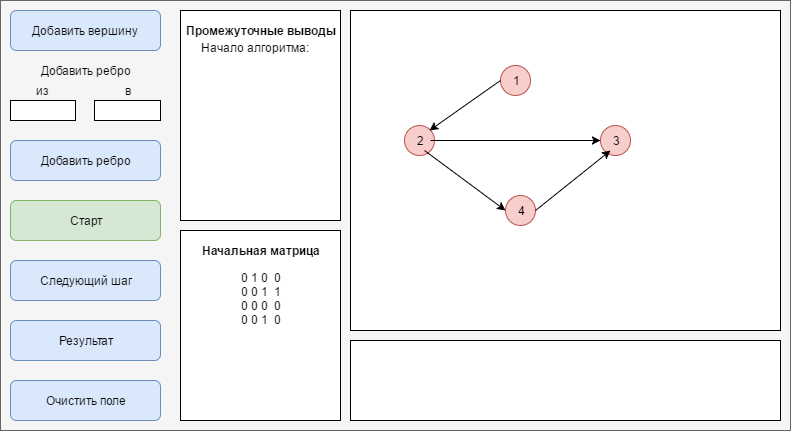


Рисунок 2. (Построен на сайте draw.io)

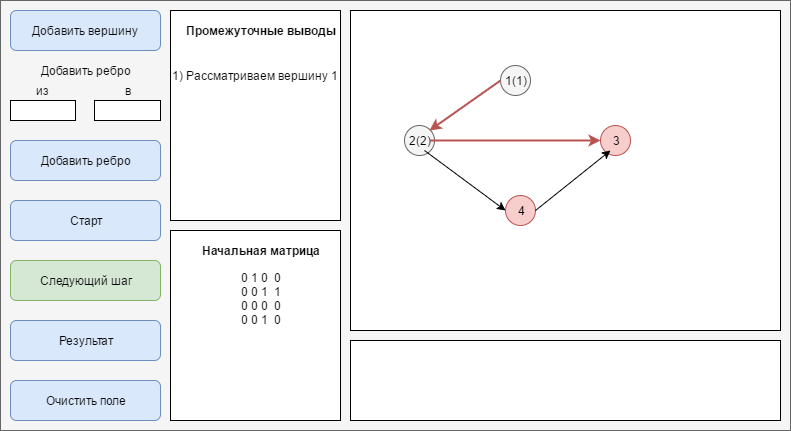


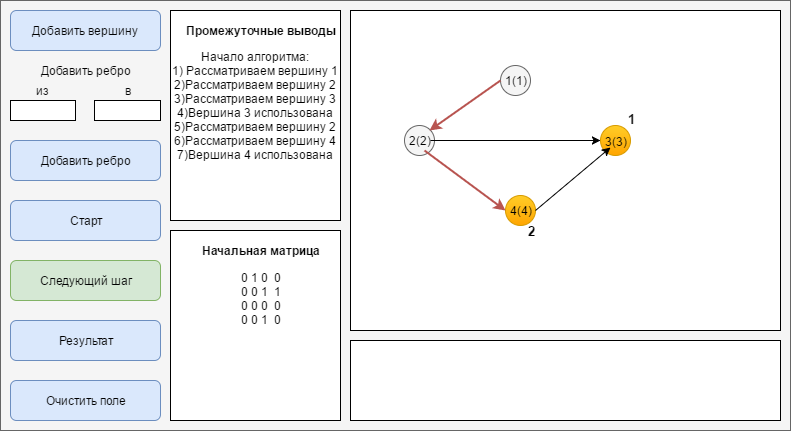
Рисунок 3. (Построен на сайте draw.io)

## 

Рисунок 4. (Построен на сайте draw.io)

## 

Рисунок 5.



## Рисунок 6.

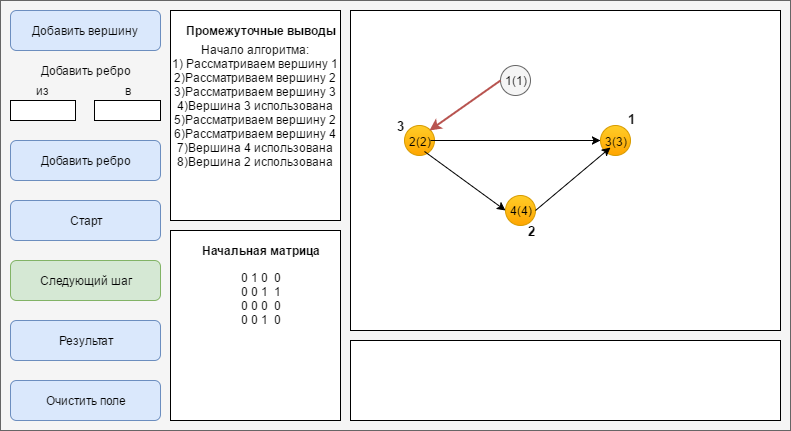


Рисунок 7.

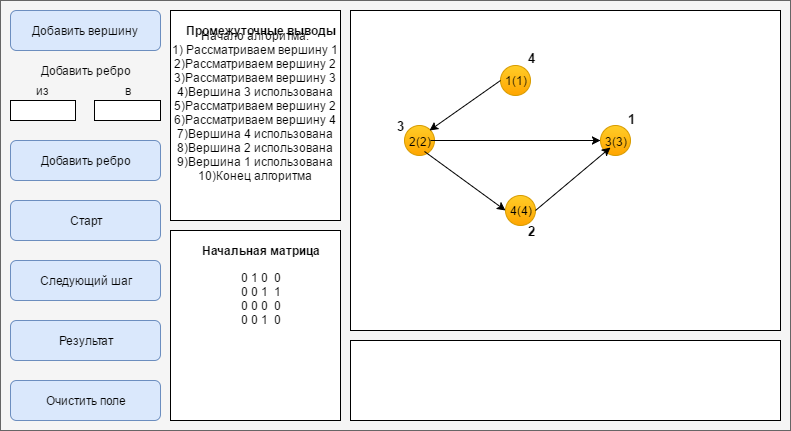


Рисунок 8.

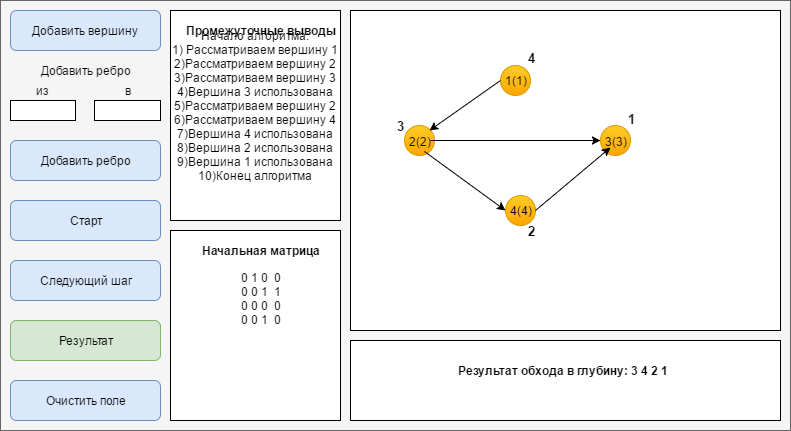


Рисунок 9.

## 

## 

## 

## 

## 

## 

## 

## 2. ПЛАН РАЗРАБОТКИ И РАСПРЕДЕЛЕНИЕ РОЛЕЙ В БРИГАДЕ

## 2.1. План разработки

22.06.2017 (четверг) – разработка спецификации

23.06.2017 – согласование спецификации с руководителем

(24.06 -25.06).2017 – разработка интерфейса

26.06.2017 (понедельник) - предоставление планируемого интерфейса (прототип), с отсутствующей функциональностью.

27.06.2017 – реализация графического представления графа, на основе входных данных с файла и осуществление работы алгоритма с выводом результата в соответствующее окно(частичная функциональность)

28.06.2017 (среда) - сдача 1-ой версии с добавлением частичной функциональности к пользовательскому интерфейсу;

29.06.2017 – реализация пошаговой работы алгоритма, реализация графического представления графа, на основе входных данных с использованием пользовательского интерфейса, тестирование программы. Оформление отчета.

30 июня (пятница) - сдача финальной версии. Представление проекта с полной функциональностью.

## 2.2. Распределение ролей в бригаде

В таблице 1 представлен состав бригады и распределение обязанностей

Таблица 1.

|  |  |
| --- | --- |
| Имя студентки | Обязанность |
| Боровикова Ольга | Осуществление пошаговой реализации алгоритма на графической части интерфейса. Тестирование программы. |
| Буздина Мария | Реализация алгоритма поиска в глубину. Осуществление пошаговой работы алгоритма и выведения результатов на текстовой панели графического интерфейса. |
| Кочнева Ольга | Разработка прототипа. Реализация интерфейса и необходимого функционала для обеспечения добавления ребер и вершин на изображении графа. |

## 3. ОСОБЕННОСТИ РЕАЛИЗАЦИИ

## 3.1 Использованные структуры данных

## 3.2 Основные методы

## 3.3 Спецификация