

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

**ОТЧЕТ  
по лабораторной работе №3  
по дисциплине «Построение и анализ алгоритмов»  
ТЕМА: «ПОТОКИ»**

Студентка гр. 7304

\_\_\_\_\_

Юруть Е.А.

Преподаватель

\_\_\_\_\_

Филатов А.Ю.

Санкт-Петербург  
2019

## Цель работы

Найти максимальный поток в сети, а также фактическую величину потока, протекающего через каждое ребро, используя алгоритм Форда-Фалкерсона.

## Задание

Найти максимальный поток в сети, а также фактическую величину потока, протекающего через каждое ребро, используя алгоритм Форда-Фалкерсона.

Сеть (ориентированный взвешенный граф) представляется в виде триплета из имён вершин и целого неотрицательного числа - пропускной способности (веса).

Входные данные:

NN - количество ориентированных рёбер графа

v0 - исток

vn - сток

vi vj ωij - ребро графа

vi vj ωij - ребро графа

...

Выходные данные:

Pmax - величина максимального потока

vi vj ωij - ребро графа с фактической величиной протекающего потока

vi vj ωij - ребро графа с фактической величиной протекающего потока

...

В ответе выходные рёбра отсортируйте в лексикографическом порядке по первой вершине, потом по второй

## Описание алгоритма Форда-Фалкерсона

1. Обнуляем все потоки. Остаточная сеть изначально совпадает с исходной сетью.

2. В остаточной сети находим путь из источника в сток. Если такого пути нет, останавливаемся.

3. Пускаем через найденный путь максимально возможный поток:

1) На найденном пути в остаточной сети ищем ребро с минимальной пропускной способностью tmp.

2) Для каждого ребра на найденном пути увеличиваем поток на tmp, а в противоположном ему - уменьшаем на tmp.

3) Модифицируем остаточную сеть. Для всех рёбер на найденном пути, а также для противоположных им рёбер, вычисляем новую пропускную способность. Если она стала ненулевой, добавляем ребро к остаточной сети, а если обнулилась, стираем его.

4. Возвращаемся на шаг 2.

## Описание функций

*int* *FF()* - вычисление максимального потока в графе, алгоритм Форда-Фалкерсона.

*void* *initPrevs()* – инициализация списка вершин.

*void* *DFS(int v)* – поиск в глубину.

*bool* *getPath(int v)* – поиск пути.

## Тестирование

```
7
a
f
a b 7
a c 6
b d 6
c f 9
d e 3
d f 4
e c 2
12
a b 6
a c 6
b d 6
c f 8
d e 2
d f 4
e c 2

2
s
e
s e 2
e s 1
2
e s 0
s e 2

9
a
d
a b 4
a e 2
e f 2
f c 2
b c 4
b g 2
g h 2
h d 2
c d 4
6
a b 4
a e 2
b c 2
b g 2
c d 4
e f 2
f c 2
g h 2
h d 2
```

## Вывод

В ходе лабораторной работы мы подробно ознакомились с алгоритмом Форда-Фалкерсона. В результате выполнения лабораторной работы была разработана программа, которая вычисляет максимальный поток в ориентированном графе. Решена проблема реализации обратного хода по ориентированному графу.