

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: Алгоритм Форда-Фалкерсона

Студент гр. 7304

Абдульманов Э.М

Преподаватель

Филатов А.Ю

г. Санкт-Петербург
2019

Цель работы

Освоить алгоритм Форда-Фалкерсона и реализовать его на языке программирования c++.

Задача

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). В ответе выходные рёбра отсортируйте в лексикографическом порядке по первой вершине, потом по второй (в ответе должны присутствовать все указанные входные рёбра, даже если поток в них равен 0).

Теоретические сведения

Сеть – ориентированный взвешенный граф, имеющий один исток и один сток.

Исток – вершина, из которой рёбра только выходят.

Сток – вершина, в которую рёбра только входят.

Поток – абстрактное понятие, показывающее движение по графу.

Величина потока – числовая характеристика движения по графу (сколько всего выходит из стока = сколько всего входит в сток).

Пропускная способность – свойство ребра, показывающее, какая максимальная величина потока может пройти через это ребро.

Максимальный поток (максимальная величина потока) – максимальная величина, которая может быть выпущена из стока, которая может пройти через все рёбра графа, не вызывая переполнения ни в одном ребре.

Фактическая величина потока в ребре – значение, показывающее, сколько величины потока проходит через это ребро.

Ход работы

Был реализован алгоритм Форда-Фалкерсона на языке программирования C++. Этот алгоритм находит максимальный поток в сети и в дополнение выводит фактические потоки для каждой дуги. Он работает следующим образом.

Шаг 1. Создается остаточный граф, который называется `network` и изначально этот граф совпадает с исходным графом, потому что потоки в графе `network` равны 0. Переходим на шаг 2.

Шаг 2. С помощью поиска в ширину(BFS) ищется путь от истока в сток, если такого пути нет, переходим на шаг 5, иначе на шаг 3.

Шаг 3. Когда путь был найден, мы в этом пути ищем дугу с наименьшей пропускной способностью(\min_flow). И переходим на шаг 4.

Шаг 4. Происходит реконструкция графа *network*. Для всех дуг, которые попали в путь от истока в сток и для противоположных им дуг, пересчитывается их пропускная способность следующим образом. Если дуга входит в путь: пропускная способность – \min_flow . Для обратных дуг: пропускная способность + \min_flow . Если пропускная способность стала равна нулю(то есть дуга насытилась), то она удаляется из графа *network*. И переходим на шаг 2.

Шаг 5. Вычисляется максимальный поток, это сумма потоков из истока. И выводятся фактические потоки для дуг.

Примеры работы программы

1. Пример работы алгоритма

Входные данные:

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

Выходные данные:

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

Вывод

В ходе данной лабораторной работы был реализован алгоритм Форда-Фалкерсона на языке программирования `c++`, который ищет максимальный поток в сети. Для поиска дополняющего пути использовался поиск в ширину, который ищет путь за время $O(|V|+|E|)$. Сложность самого алгоритма составляет $O((|V|+|E|)*f)$, где f – максимальный поток в сети. Если величина пропускной способности хотя

бы одного из рёбер - иррациональное число, то алгоритм может работать бесконечно, даже не обязательно сходясь к правильному решению.