

Лабораторная работа №9 по курсу дискретного анализа: алгоритмы на графах

Выполнил студент группы М8О-307Б-21 МАИ *Кажекин Денис Андреевич*

Условие

Задан взвешенный ориентированный граф, состоящий из n вершин и m ребер. Вершины пронумерованы целыми числами от 1 до n . Необходимо найти длины кратчайших путей между всеми парами вершин при помощи алгоритма Джонсона. Длина пути равна сумме весов ребер на этом пути. Обратите внимание, что в данном варианте веса ребер могут быть отрицательными, поскольку алгоритм умеет с ними работать. Граф не содержит петель и кратных ребер.

Метод решения

Общее решение строится на последовательном применении алгоритма Форда-Беллмана, который умеет работать с отрицательными весами (но не с отрицательными циклами, которые мы за одно и обнаруживаем), а затем Дейкстры. Это необходимо, потому что Дейкстра умеет работать только с неотрицательными весами на ребрах. Мы используем алгоритм Форда-Беллмана на графе, который дополняется новой искусственной вершиной S , из которой достижимы все остальные вершины графа. Таким образом, мы получаем массив потенциалов для каждой вершины, каждый из которых является кратчайшим путем до соответствующей вершины. Имея этот массив, можно по формуле $w'(uv) = [w(uv) + \text{phi}(u)] - \text{phi}(v)$ переопределить веса ребер, чтобы они стали неотрицательными. После этого применяем алгоритм Дейкстры, находя кратчайшие пути для всех пар вершин графа.

Алгоритмическая сложность программы $O(V^2 \log(V) + V E)$

Описание программы

Программа реализована с одним source файлом `jhonson.cpp`

Дневник отладки

Вначале был неверный ответ из-за использования типа данных `int` для определения номеров вершин, ребер и соответствующих им весов, а также `INT_MAX` для определения значения бесконечности. После изменения на `long int` и `LONG_MAX` ошибка была решена.

Далее столкнулся с `time_limit` на 19 тесте. Получилось исправить переписав алгоритм Дейкстры на приоритетной очереди.

Тест производительности

n/m	100	2100	4100	6100
100	26.304 ms	180.682 ms	271.856 ms	381.492 ms
200	26.656 ms	241.564 ms	628.513 ms	1235.41 ms
300	26.175 ms	303.103 ms	717.383 ms	1396.9 ms

Заметим, что на при количестве ребер в 100 для разного количества вершин, алгоритм выполняет работу одинаково, а при увеличении ребер, разница между работами увеличивается

Недочёты

Недочеты не выявлены

Выводы

Сделав лабораторную работу, я разобрался с алгоритмом Форда-Беллмана, Дейкстры, которые лежат в основе алгоритма Джонсона. Научился работать с графами. Считаю эту тему важной, потому что графы имеют широкое применение в прикладных задачах.