



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Программное обеспечение ЭВМ и информационные технологии

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 7 **“Графы”**

Студент Татаринова Дарья Алексеевна
фамилия, имя, отчество

Группа ИУ7-34Б

Выполнил Татаринова Д.А.
подпись, дата *фамилия, и.о.*

Принял Силантьева А.В.
подпись, дата *фамилия, и.о.*

2021 г.

Цель работы

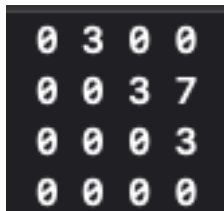
Цель работы – реализовать алгоритмы обработки графовых структур: поиск различных путей, проверку связности, построение остовых деревьев минимальной стоимости.

Задание

Найти самый длинный простой путь в графе.

Входные данные

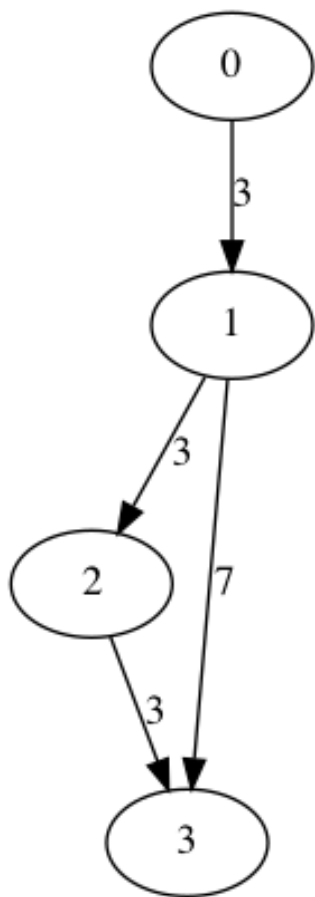
- Количество вершин в графе.
- Матрица целых чисел. Если две точки не соединены путем, то вводится 0.



0	3	0	0
0	0	3	7
0	0	0	3
0	0	0	0

Выходные данные

1. Граф, изображенный при помощи GRAPHVIZ.
2. Две точки, которые соединяет самый длинный простой путь, и длина этого пути.
3. Если найдено несколько максимальных путей, то будет выведен тот, у которого индекс начальной вершины меньше.



Способ обращения к программе

Программа может быть вызвана через консоль.

Команда сборки:

```
gcc -std=c99 -Wall -Werror -Wextra -Wpedantic -Wfloat-conversion -Wvla -  
Wfloat-equal -c main.c
```

```
gcc -std=c99 -Wall -Werror -Wextra -Wpedantic -Wfloat-conversion -Wvla -  
Wfloat-equal -c graph.c
```

```
gcc main.o graph.o -o app.exe
```

Аварийные ситуации

- Длина пути не целое число.

Структуры данных

Граф представлен в виде матрицы.

Описание алгоритма

- Самый длинный путь A между двумя заданными вершинами s и t во взвешенном графе G — это то же самое, что и кратчайший путь в графе $-G$, полученном из G путем замены всех весов на веса с обратным знаком.
- Поиск самого короткого пути реализован по алгоритму Флойда.

Вывод

Для реализации данной задачи был использован алгоритм Флойда, который очень прост в реализации. Сложность алгоритма же кубическая.

Хранение графа в виде матрицы смежности удобно тем, что по матрице можно понять расстояние между вершинами, или узнать, существует ли ребро между вершинами.

1. Что такое граф?

Граф — это конечное множество вершин и ребер, соединяющих их, $G = \langle V, E \rangle$, где V — конечное непустое множество вершин; E — множество ребер (пар вершин).

Если пары E (ребра) имеют направление, то граф называется ориентированным (орграф), если иначе — неориентированный (неорграф). Если в пары E входят только различные вершины, то в графе нет петель. Если ребро графа имеет вес, то граф называется взвешенным.

Неорграф называется связным, если существует путь из каждой вершины в любую другую.

2. Как представляются графы в памяти?

В памяти удобно представлять граф в виде матрицы смежности или списка смежности.

Матрица смежности $B(n \times n)$ – элемент $b[i,j]=1$, если существует ребро, связывающее вершины i и j , и $=0$, если ребра не существует.

Список смежностей – содержит для каждой вершины из множества вершин V список тех вершин, которые непосредственно связаны с ней. Входы в списки смежностей могут храниться в отдельной таблице, либо же каждая вершина может хранить свой список смежностей.

3. Какие операции возможны над графами?

Обход вершин и поиск различных путей: поиск кратчайшего пути от одной вершины к другой (если он есть), поиск кратчайшего пути, поиск эйлерова пути, поиск гамильтонова пути.

4. Какие способы обхода графов существуют?

Обход в ширину (BFS – Breadth First Search) - обработка вершины V осуществляется путём просмотра сразу всех «новых» соседей этой вершины, которые последовательно заносятся в очередь просмотра.

Обход в глубину (DFS – Depth First Search) - начиная с некоторой вершины v_0 , ищется ближайшая смежная ей вершина v , для которой в свою очередь осуществляется поиск в глубину до тех пор, пока не встретится ранее просмотренная вершина, или не закончится список смежности вершины v (то есть вершина полностью обработана). Если нет новых вершин, смежных с v , то вершина v считается использованной, идет возврат в вершину, из которой попали в вершину v , и процесс продолжается до тех пор, пока не получим $v = v_0$. При просмотре используется стек.

5. Где используются графовые структуры?

Графовые структуры могут использоваться в задачах, где между элементами могут быть установлены произвольные связи. Наиболее распространенное использование таких структур — при решении различных задач о путях.

6. Какие пути в графе Вы знаете?

Эйлеровый путь - путь в графе, проходящий через каждое ребро ровно один раз. (путь может проходить по некоторым вершинам несколько раз – в этом случае он является непростым)

Гамильтонов путь - путь, проходящий через каждую вершину ровно один раз.

Такие пути могут не существовать в графах.

7. Что такое каркасы графа?

Каркас графа – дерево, в которое входят все вершины графа, и некоторые (не обязательно все) его ребра. Для построения каркасов графа используются алгоритмы Крускала и Прима.