|  |  |
| --- | --- |
|  | **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 ПО ПРЕДМЕТУ "ТИПЫ И СТРУКТУРЫ ДАННЫХ" №7**

Студент \_Золотухин Алексей Вячеславович\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*фамилия, имя, отчество*

Группа \_\_\_\_\_\_ИУ7-34Б\_\_\_\_\_\_\_\_\_

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_Золотухин А. В.\_\_\_\_

*подпись, дата фамилия, и.о.*

Принял \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_Силантьева А. В. \_

*подпись, дата фамилия, и.о.*

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Москва, 2021**

**Задача:**

Задана система двусторонних дорог (путь между двумя городами в противоположные стороны может быть разным). Для каждой пары городов найти длину кратчайшего пути между ними.

**Входные данные**

При запуске программа читает данные из файла “in.txt”, получает на вход количество городов. После чего производится ввод графа в виде матрицы смежности. Длины дорог — целые положительные числа.

**Выходные данные**

Программа выводит матрицу стоимостей с длинами кратчайших путей от одной вершины до другой. Также создаётся графическое представление графа.

**Аварийные ситуации**

При любом некорректном вводе программа выводит сообщение об ошибке и завершает работу.

**Структура данных**

#define V 20

typedef struct

{

    size\_t v;

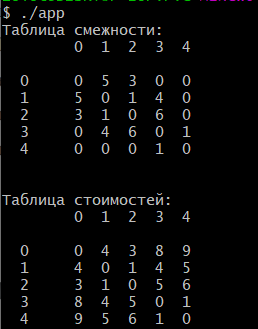
    int m[V][V];

} matrix\_t;

Граф хранится в виде матрицы смежности, где G[i][j] = val, если из i в j ведёт дорога длины val и равно 0 в противном случае. Такое представление хранит избыточную информацию, однако максимально удобно в использовании.

Для решения данной задачи я использовал алгоритм Дейкстры, как самый гибкий в использовании алгоритм. Сложность алгоритма - O(V^2), но так как я этот алгоритм использую для каждой вершины, то получается O(V^3).

**Тесты**

****

Если прочесть информацию из входного файла не удалось то будет выведено соответствующее сообщение об ошибке





**Примечание**

Данная программа может быть использована для решения логистических проблем.

**Ответы на контрольные вопросы**

***1. Что такое граф?***

Граф – это конечное множество вершин и ребер, соединяющих их, т. е.: G = < V,E >, где V – конечное непустое множество вершин; Е – множество ребер (пар вершин).

***2. Как представляются графы в памяти?***

Графы в памяти могут представляться различным способом. Один из видов представления графов – это матрица смежности B(n\*n); В этой матрице элемент b[i,j]=1, если ребро, связывающее вершины Vi и Vj существует и b[i,j]=0, если ребра нет. У неориентированных графов матрица смежности всегда симметрична. Во многих случаях удобнее представлять граф в виде так называемого списка смежностей. Список смежностей содержит для каждой вершины из множества вершин V список тех вершин, которые непосредственно связаны с этой вершиной.

***3. Какие операции возможны над графами?***

Перечислим основные операции по работе с графовыми структурами:

* поиск кратчайшего пути от одной вершины к другой (если он есть);
* поиск кратчайшего пути от одной вершины ко всем другим;
* поиск кратчайших путей между всеми вершинами;
* поиск эйлерова пути (если он есть);
* поиск гамильтонова пути (если он есть).

***4. Какие способы обхода графов существуют?***

Один из основных методов проектирования графовых алгоритмов – это поиск (или обход графа) в глубину (depth first search, DFS), при котором, начиная с произвольной вершины v0, ищется ближайшая смежная вершина v, для которой, в свою очередь, осуществляется поиск в глубину (т.е. снова ищется ближайшая, смежная с ней вершина) до тех пор, пока не встретится ранее просмотренная вершина, или не закончится список смежности вершины v (то есть вершина полностью обработана). Другой метод обхода графа – поиск в ширину (breadth first search, BFS). Обработка вершины v осуществляется путем просмотра сразу всех новых соседей этой вершины. При этом полученный путь является кратчайшим путем из одной вершины в другую

5. ***Где используются графовые структуры?***

Графовые структуры используются почти в каждой сфере науки, начиная от структурных формул веществ в химии и заканчивая генеалогическим древом выдающейся личности в истории.

***6. Какие пути в графе Вы знаете?***

Произвольный путь в графе, проходящий через каждое ребро графа точно один раз, называется эйлеровым путем. При этом, если по некоторым вершинам путь проходит неоднократно, то он является непростым. Если путь замкнут, то имеем эйлеров цикл. Для существования эйлерова пути в связном графе необходимо и достаточно, чтобы граф содержал не более двух вершин нечетной степени. Путь в графе, проходящий в точности один раз через каждую вершину графа (а не каждое ребро) и соответствующий цикл называются гамильтоновыми и существуют не для каждого графа, как и эйлеров путь. В отличие от эйлеровых путей неизвестно ни одного простого необходимого и достаточного условия для существования гамильтоновых путей. Неизвестен даже алгоритм полиномиальной сложности, проверяющий существование гамильтонова пути в произвольном графе.

***7. Что такое каркасы графа?***

Остовное дерево или каркас графа – это подграф, который : 1) содержит все вершины графа, 2) является деревом.