	<p>Министерство образования и науки 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)</p>
---	---

ФАКУЛЬТЕТ _____ Информатика и системы управления
 КАФЕДРА _____ Программное обеспечение ЭВМ и информационные технологии

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ 8
По дисциплине «Типы и структуре данных»

Название **«Графы»**

Студент Дубов Андрей Игоревич
 фамилия, имя, отчество

Группа ИУ7-33Б

Вариант 13

Тип лабораторной работы Учебная

Студент	_____	<u>Дубов А. И.</u>
	<i>подпись, дата</i>	<i>фамилия, и.о.</i>
Преподаватель	_____	<u>Рыбкин Ю. А.</u>
	<i>подпись, дата</i>	<u>Силантьева А. В.</u>
		<i>фамилия, и.о.</i>

2022 г.

Оглавление

Условие задачи.....3

Описание технического задания.....3

Входные данные:.....3

Выходные данные:.....3

Аварийные ситуации:.....3

Описание структуры данных.....3

Описание алгоритма.....3

Набор тестов.....4

Вывод.....4

Ответы на контрольные вопросы.....4

Условие задачи

Обработать графовую структуру в соответствии с заданным вариантом. Обосновать выбор необходимого алгоритма и выбор структуры для представления графов. Ввод данных осуществить на усмотрение программиста. Результат выдать в графической форме.

Описание технического задания

В системе двусторонних дорог за проезд каждой дороги взимается некоторая пошлина. Найти путь из города А в город В с минимальной величиной $S + P$, где S – сумма длин дорог пути, а P – сумма пошлин проезжаемых дорог.

Входные данные:

Файл с данными о длинах дорог, номер команды или вершины для обработки.

Выходные данные:

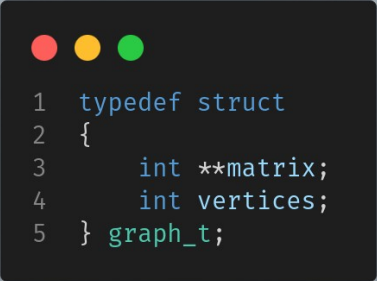
Дерево, информация о частотности слова.

Аварийные ситуации:

1. Некорректный ввод номера команды или номера вершины
2. Отрицательные длины рёбер
3. Поиск для путей несуществующих вершин

Описание структуры данных

Структура стека

A code editor window with a dark background and three colored window control buttons (red, yellow, green) at the top left. It contains C code for a graph structure.

```
1 typedef struct
2 {
3     int **matrix;
4     int vertices;
5 } graph_t;
```

matrix – матрицы длин ребер

vertices – количество вершин

Описание алгоритма

1. Пользователю предлагается инициализировать граф и посмотреть кратчайшие пути
2. Пока пользователь не введет 0 (выход из программы), ему будет предложено вводить номера команд и выполнять действия по выбору.
3. При обработке кратчайших путей алгоритм добавляет начальную вершины в очередь, смотрит для нее все соединенные необработанные вершины, потом добавляет в очередь не добавленные ранее по порядку от минимальной длины пути до вершины к большей. Алгоритм работает пока очередь не опустеет. Алгоритм более известен как алгоритм Дейкстры. Сложность Алгоритма $O(n^2)$

Набор тестов

	Название теста	Пользователь вводит	Вывод
1	Некорректный ввод команды	45	No such option or wrong input
2	Пустой ввод	Пустой ввод.	No such option or wrong input
3	Команда 0	0	Выход из программы

Оценка эффективности

Для конкретной вершины в среднем за 50 запусков поиск кратчайших маршрутов до всех вершин 22136.

Вывод

При помощи алгоритма Дейкстры можно достаточно быстро и легко искать пути в графе. Для этого алгоритма удобно использовать матрицу стоимостей, поэтому представление графа было выбрано именно в таком виде.

Ответы на контрольные вопросы

- 1) Что такое граф?

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

- 2) Как представляются графы в памяти?

Графы в памяти могут представляться различным способом. Один из видов представления невзвешенных графов – это матрица смежности $B(n \times n)$; В этой матрице элемент $b[i, j] = 1$, если ребро, связывающее вершины V_i и V_j существует и $b[i, j] = 0$, если ребра нет. У неориентированных графов матрица смежности всегда симметрична.

- 3) Какие операции возможны над графами?

Включение элемента

Исключение элемента

Обход графа

- 4) Какие способы обхода графов существуют?

Обход графа в глубину

Обход графа в ширину

- 5) Где используются графовые структуры?

Графовые структуры используются, к примеру, в задачах на пути (Схема дорог, метро, проезд между городами, странами, кратчайшая дорога от одной точки до другой и тд.)

- 6) Какие пути в графе Вы знаете?

Эйлера пути - когда граф имеет цикл, содержащий все рёбра графа по одному разу.

Гамильтонова пути - простой путь, проходящий через каждую вершину графа ровно 1 раз.

- 7) Что такое каркасы графа?

Дерево в котором содержатся все вершины искомого графа и некоторые его рёбра. (Без циклов)

При использовании алгоритмов поисков в ширину и глубину графы обходят разными способами, получая при этом некоторые подграфы, которые имеют специфические названия: каркасы (остовы или стягивающие деревья).