|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **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: Светличная Алина Алексеевна

Группа: ИУ7 – 33Б

Проверил: Силантьева Александра Васильевна

*2021 г.*

# **Описание технического задания**

Техническое задание:

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.

Условие задачи:

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

Входные данные:

* Тип заполнения (вручную или из файла)
* Количество вершин графа
* Последовательность вершин с длиной дорог между ними и стоимостью проезда
* Пара вершин для поиска кратчайшего пути

Выходные данные:

Последовательность вершин кратчайшего пути по сумм.

Обращение к программе:

Запускается через терминал командой ./app.exe в директории с программой.

Возможные аварийные ситуации:

1. Некорректный тип заполнения
2. Некорректные символы
3. Число не является возможным вариантом (отлично от 0 и 1)
4. Некорректный файл
5. Имя файла некорректно для считывания
6. Файл невозможно открыть (несуществующий файл)
7. Некорректное количество вершин
8. Некорректные символы
9. Не натуральное число
10. Некорректное заполнение матрицы смежностей
11. Некорректные символы
12. Номера вершины не принадлежит промежутку [1, количество вершин]
13. Длина пути меньше 1
14. Стоимость проезда меньше 1
15. Некорректная пара вершин для поиска кратчайшего пути
16. Некорректные символы
17. Номера вершины не принадлежит промежутку [1, количество вершин]
18. Вершины совпадают
19. Граф несвязный

# **Описание структур данных**

Поля структуры, описанной ниже, для хранения узла графа (одной дороги):

length – длина пути

price – стоимость проезда

**typedef struct**

**{**

**int length;**

**int price;**

**}road\_t;**

Поля структуры, описанной ниже, для хранения всего графа:

size – количество дорог в графе

\*\*matrix – динамическая матрица смежностей графа

\*path – кратчайший путь между заданными вершинами

**typedef struct**

**{**

**int** **size;**

**road\_t \*\*matrix;**

**char \*path;**

**} graph\_t;**

# **Описание алгоритма**

Для поиска кратчайшего пути используется алгоритм Дейкстры. Он подходит под условие задачи, так как в графе не может быть отрицательных весов. Алгоритм Дейкстры не подходит для несвязного графа, и так же не подходит для графа, где есть отрицательные весы у дуг. Но, так как в моей задаче таких случаев быть не может, то данный алгоритм идеально подходит для решения задачи. Так как вершины хранятся в простом массиве и для поиска минимума используется линейный алгоритм, то сложность алгоритма составляет O(V \* V + E) = O(V2).

Данная задача абсолютно явным образом может быть использована при разработке приложений для построения маршрутов (особенно, если заменить стоимость проезда, например, на стоимость бензина на путь, задача станет еще универсальнее).

**Тесты**

|  |  |  |
| --- | --- | --- |
| Аварийная ситуация | Код ошибки | Уведомление |
| Некорректный тип заполнения | | |
| Тип заполнения - k | ERROR\_SYMBOLS | Ошибка: встречены некорректные символы |
| Тип заполнения - 2 | ERROR\_VALUE | Ошибка: встречено некорректное значение |
| Некорректный файл | | |
| Имя файла – “\n” | ERROR\_SYMBOLS | Ошибка: встречены некорректные символы |
| Несуществующий файл | ERROR\_FILE | Ошибка: невозможно открыть файл |
| Некорректное количество вершин | | |
| Количество вершин - ! | ERROR\_SYMBOLS | Ошибка: встречены некорректные символы |
| Количество вершин - -2 | ERROR\_VALUE | Ошибка: количество вершин должно быть натуральным |
| Некорректное заполнение матрицы смежностей | | |
| Вершина - ! | ERROR\_SYMBOLS | Ошибка: встречены некорректные символы |
| Первая вершина - 0 | ERROR\_VALUE | Ошибка: встреченo некорректные значение первой вершины |
| Первая вершина -5 (при количестве вершин 4) | ERROR\_VALUE | Ошибка: встреченo некорректные значение первой вершины |
| Вторая вершина - -3 | ERROR\_VALUE | Ошибка: встреченo некорректные значение второй вершины |
| Вторая вершина -5 (при количестве вершин 4) | ERROR\_VALUE | Ошибка: встреченo некорректные значение второй вершины |
| Длина пути - 0 | ERROR\_VALUE | Ошибка: встреченo некорректные значение длины пути |
| Стоимость проезда - -12 | ERROR\_VALUE | Ошибка: встреченo некорректные значение стоимости пути |
| Некорректная пара вершин для поиска кратчайшего пути | | |
| Вершина - ety | ERROR\_SYMBOLS | Ошибка: встречены некорректные символы |
| Первая вершина - 0 | ERROR\_VALUE | Ошибка: встреченo некорректные значение первой вершины |
| Первая вершина -5 (при количестве вершин 4) | ERROR\_VALUE | Ошибка: встреченo некорректные значение первой вершины |
| Вторая вершина - -3 | ERROR\_VALUE | Ошибка: встреченo некорректные значение второй вершины |
| Вторая вершина -5 (при количестве вершин 4) | ERROR\_VALUE | Ошибка: встреченo некорректные значение второй вершины |
| Вершины равны | ERROR\_EQUAL\_VERTEX | Ошибка: вершины совпадают |
| Граф несвязный | | |
| Из вершины A вершина B не достигается | ERROR\_GRAPH | Ошибка: граф не является связным |

**Контрольные вопросы**

**Что такое граф?**Граф – конечное множество вершин и соединяющих их ребер; G = <V, E>. Если пары Е (ребра) имеют направление, то граф называется ориентированным; если ребро имеет вес, то граф называется взвешенным.

**Как представляются графы в памяти?**

С помощью матрицы смежности или списков смежности.

**Какие операции возможны над графами?**

Обход вершин, поиск различных путей, исключение и включение вершин.

**Какие способы обхода графов существуют?**

Алгоритм в ширину, алгоритм в глубину.

**Где используются графовые структуры?**Графовые структуры могут использоваться в задачах, в которых между элементами могут быть установлены произвольные связи (в отличии от дерева могут быть не иерархические).

**Какие пути в графе Вы знаете?**Эйлеров путь, непростой путь, гамильтонов путь.

**Что такое каркасы графа?**

Каркас графа – дерево, в которое входят все вершины графа, и некоторые (необязательно все) его рёбра.

**Вывод**

В ходе лабораторной познакомились со сложной структурой представления информации – графов. Мною была выбрана форма представления графовой структуры – матрица смежности, алгоритм Дейкстры. Так как задача о реальном объекте в виде графовых структур, а именно о дорожном полотне, то мы можем заранее предположить, что граф будет являться смежным и не содержать отрицательных ребер, для такого случая наиболее прост именно данный алгоритм. Однако существуют и иные алгоритмы, не ставящие такие условия, например алгоритм Флойда-Уоршалла.