

Расчётная работа.

Теория графов.

1 Цель работы

- 1.1 Ознакомиться с понятием графов.
- 1.2 Узнать какие способы представления графов существуют.
- 1.3 Научиться решать теоретико-графовые задачи.

2 Задачи

- 2.1 Придумать алгоритм решения теоретико-графовой задачи.
- 2.2 Реализовать алгоритм решения задачи на языке программирования C++.
- 2.3 Протестировать алгоритм.

3 Вариант

- 3.1 Вариант - 4.9. Структура представления графа - матрица инцидентности.

4 Список ключевых понятий

- 4.1 Граф представляет собой пару $G = (V, E)$, где V - множество, элементы которого называются вершинами, а E - набор неупорядоченных пар $\{v_1, v_2\}$ из вершин, элементы которых называются ребрами.
- 4.2 Путь в графе — последовательность вершин, в которой каждая вершина соединена со следующей ребром.
- 4.3 Неориентированный граф — это тип графа, в котором рёбра не имеют направления. Это означает, что если между двумя вершинами существует ребро, то можно перемещаться от одной вершины к другой в обоих направлениях.
- 4.4 Окружение вершины v — множество всех вершин графа G , смежных с вершиной v .
- 4.5 Матрица инцидентности — одна из форм представления графа, в которой указываются связи между инцидентными элементами графа рёбрам и вершинам. Столбцы матрицы соответствуют ребрам, строки — вершинам. Ненулевое значение в ячейке матрицы указывает связь между вершиной и ребром (их инцидентность).
- 4.6 Взвешенный граф — это граф, в котором каждое ребро обозначается числом. Это число — его вес:

5 Описание алгоритма решения

- 5.1 Чтение графов из файлов: открываем указанный файл и считываем количество вершин и рёбер. Создаем двумерный вектор (матрицу) для представления графа. Считываем данные из файла и заполняет матрицу, где 1 обозначает наличие ребра, а 0 — его отсутствие.
- 5.2 Пересечение графов: принимаем два графа и находим их пересечение. Для каждого ребра в первом графе проверяется, существует ли соответствующее ребро во втором графе: если ребро найдено, проверяется, совпадают ли соответствующие значения в матрицах для всех вершин, если совпадают, ребро добавляется в результирующую матрицу.
- 5.3 Вывод графов: выводим матрицу смежности графа в формате матрицы инцидентности

6 Тестирование программы:

- 1. Первый тестовый пример.

Первый граф в виде матрицы инцидентности:

```

4 3
1 0 0
0 1 1
1 1 0
0 0 1

```

Рис. 1: graph1.txt

Второй граф в виде матрицы инцидентности:

```

5 5
1 1 1 0 0
1 0 0 1 1
0 1 0 1 0
0 0 1 0 0
0 0 0 0 1

```

Рис. 2: graph2.txt

Граф пересечения в виде матрицы:

```

3 3
1 0 0
0 1 0
1 1 0
0 0 0

```

Рис. 3: graph3.txt

2. Второй тестовый пример.

Первый граф в виде матрицы инцидентности:

```

6 6
0 0 0 0 1 1
0 1 1 0 1 0
0 0 1 1 0 0
1 1 0 0 0 0
1 0 0 0 0 0
0 0 0 1 0 1

```

Рис. 4: graph4.txt

Второй граф в виде матрицы инцидентности:

```

6 6
1 0 0 0 0 1
1 1 0 0 0 0
0 1 1 0 0 0
0 0 1 1 0 0
0 0 0 1 1 0
0 0 0 0 1 1

```

Рис. 5: graph5.txt

Граф пересечения в виде матрицы:

```

6 6
0 0 0 0 1 1
0 0 1 0 1 0
0 0 1 0 0 0
1 0 0 0 0 0
1 0 0 0 0 0
0 0 0 0 0 1

```

Рис. 6: graph6.txt

6.1 Вывод

В этой работе мы ознакомились с понятием графов, с представлением графов в виде матрицы инцидентности для нахождения пересечения неориентированных графов, написали алгоритм и код для решения задачи.

Список литературы

- [1] Свободная энциклопедия "Википедия"[Электронный ресурс]. - Режим доступа: <https://ru.wikipedia.org/wiki/Граф>
- [2] Сайт "habr"[Электронный ресурс]. - Режим доступа: <https://habr.com/ru/companies/otus/articles/568026/>
- [3] Сайт "habr"[Электронный ресурс]. - Режим доступа: <https://habr.com/ru/articles/734642/>