ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО»
Институт машиностроения, материалов и транспорта
Высшая школа автоматизации и робототехники

КУРСОВАЯ РАБОТА

**Способы представления графов**

по дисциплине «Объектно-ориентированное программирование»

Выполнил: студент гр. 3331506/90401 Колесов С.А.

Преподаватель: Ананьевский М.С.

Санкт-Петербург

2022

## Введение

Граф - это топологическая модель, которая состоит из множества вершин и множества соединяющих их рёбер. При этом значение имеет только сам факт, какая вершина с какой соединена. В свою очередь вершина - точка в графе, отдельный объект, а ребро - неупорядоченная пара двух вершин, которые связаны друг с другом.

Графы являются очень полезной в программировании структурой, поскольку зачастую задачи компьютерной науки можно представить в виде графа и решить с помощью одной из его техник. Графы находят широкое применение в современной науке и технике. Они используются и в естественных науках (физике и химии) и в социальных науках (например, социологии), но наибольших масштабов применение графов получило в информатике и сетевых технологиях.

Для удобной работы с графами, их представляют в памяти различными способами, рассмотрим основные способы представления графов и оценим их характеристики по памяти и скорости доступа к элементам.

## Особенности графов

Простой граф $\boldsymbol{G}(\boldsymbol{V},\ \boldsymbol{E})$ – совокупность двух множеств – непустого множества $V$ и множества $E$ неупорядоченных пар различных элементов множества $V$. Множество $V$ называется множеством вершин, а множество $E$ называется множеством ребер. Пример визуального представления простого графа показан на рисунке 1.

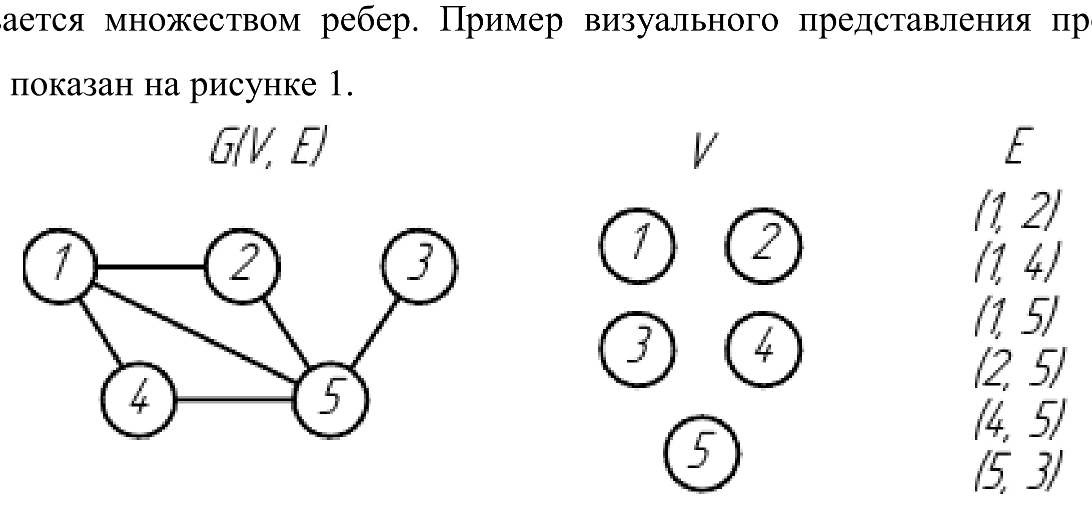



Рисунок 1 – Простой граф

Если в графе $\boldsymbol{G}(\boldsymbol{V},\ \boldsymbol{E})$ в множестве ребер *E* разрешены элементы типа $\boldsymbol{e} = \{\boldsymbol{v},\ \boldsymbol{v}\}$, то такой граф называется псевдографом. Другими словами если в графе ребра могут быть петлями, то есть начинаться и заканчиваться в одной вершине, то такой граф $\boldsymbol{G}(\boldsymbol{V},\ \boldsymbol{E})$ называется псевдографом. Пример псевдографа представлен на рисунке 2.

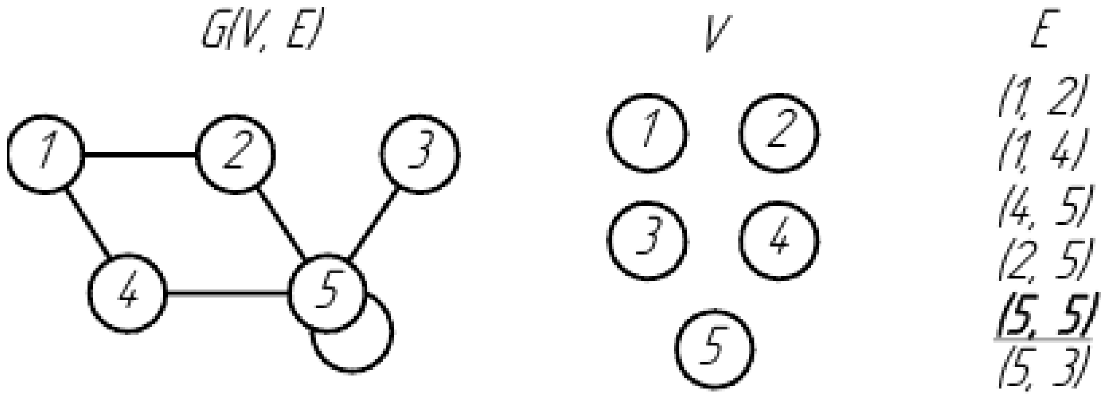



Рисунок 2 – Псевдограф

Если в графе $\boldsymbol{G}(\boldsymbol{V},\ \boldsymbol{E})$ в множество ребер *E*, содержит одинаковые элементы, то есть если *E* не множество, а семейство, то такие элементы называются кратными ребрами, а граф $\boldsymbol{G}(\boldsymbol{V},\ \boldsymbol{E})$ называется мультиграфом. Пример мультиграфа представлен на рисунке 3.

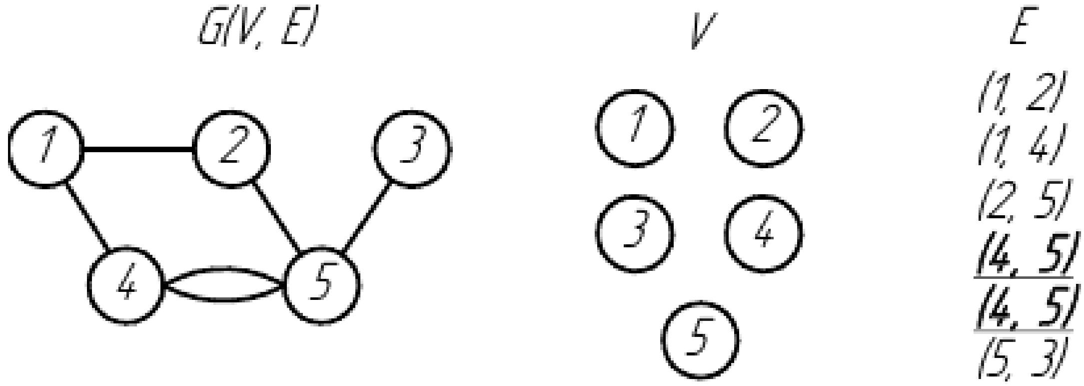



Рисунок 3 – Мультиграф

Если множество ребер *E* в графе $\boldsymbol{G}(\boldsymbol{V},\ \boldsymbol{E})$ – множество упорядоченых пар различных элементов множества *V*, то такой граф называется ориентированным или же орграфом. Тогда множество *E* будет состоять из элементов $\boldsymbol{e} = \{\boldsymbol{v_1},\ \boldsymbol{v_2}\}$, где вершина $\boldsymbol{v_1}$ – начало дуги, а $\boldsymbol{v_2}$ – конец дуги. Пример ориентированного графа представлен на рисунке 4.

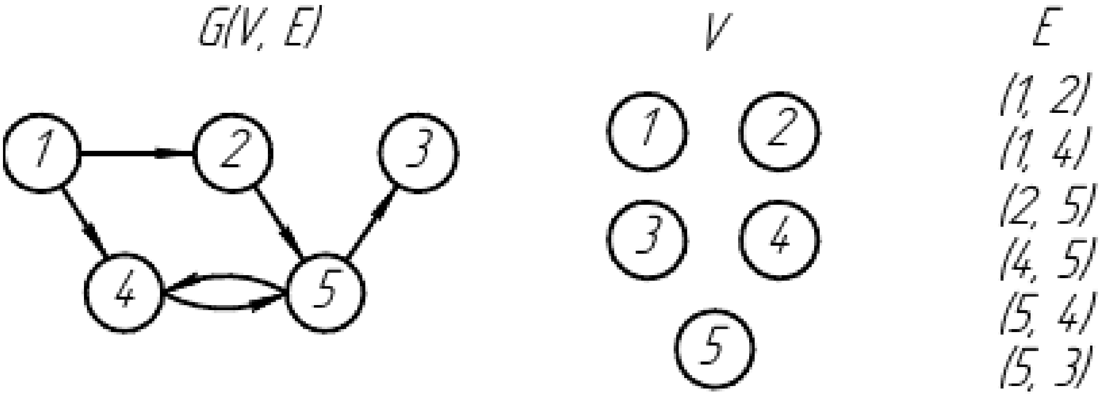



Рисунок 4 – Ориентированный граф

Если в графе присутствуют как ориентированные так и не ориентированные ребра, то есть граф является совокупностью трех множеств $\boldsymbol{G}(\boldsymbol{V},\ \boldsymbol{E},\ \boldsymbol{U})$: множества вершин $V$, множества упорядоченных пар различных вершин $E$ и множества неупорядоченных пар различных вершин $U$. Пример смешанного графа представлен на рисунке 5.

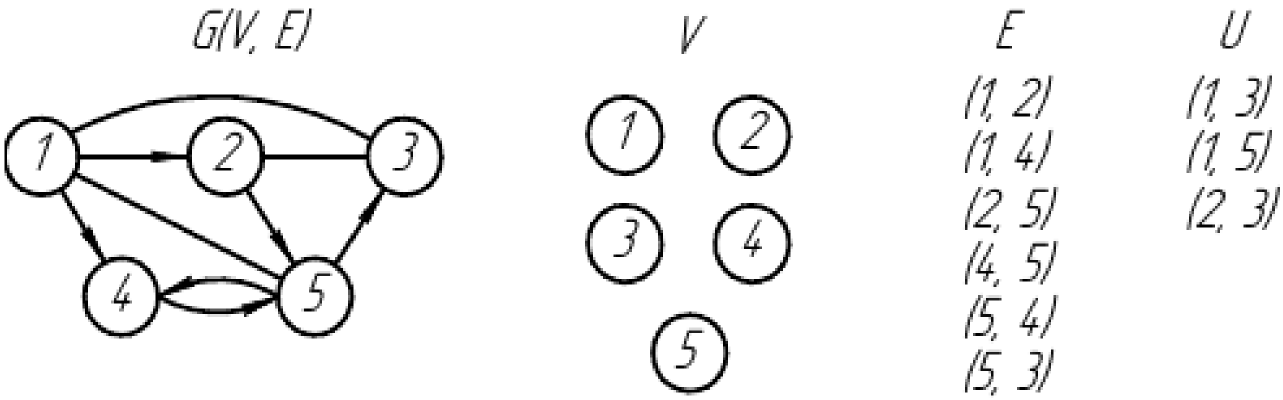



Рисунок 5 – Смешанный граф

Если каждому ребру графа поставлено в соответствие некоторое число – вес ребра, тогда граф называется взвешенным. В таком случае множество $E$ будет состоять из элементов $\boldsymbol{e} = \{\boldsymbol{v_1},\ \boldsymbol{v_2},\ \boldsymbol{w}\}$, где $\boldsymbol{w}$ – вес ребра. Пример взвешенного графа представлен на рисунке 6.

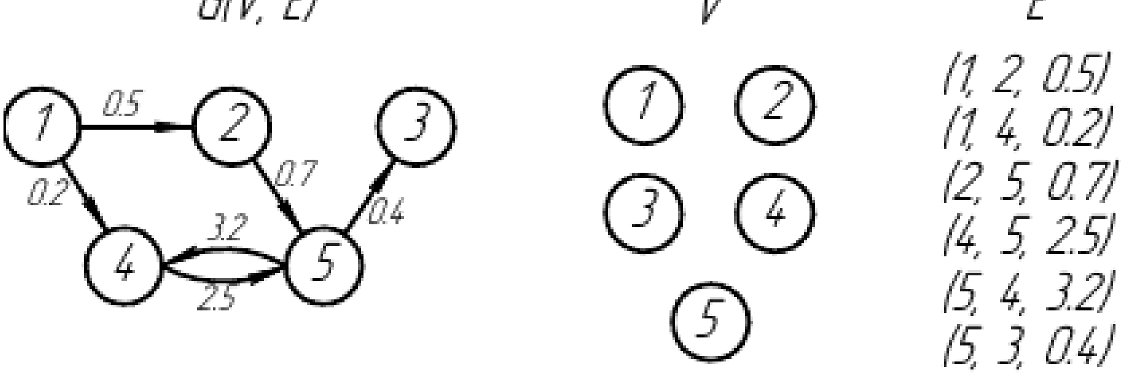



Рисунок 6 – Взвешенный граф

Если каждым ребром графа $\boldsymbol{G(V, E)}$ могут соединяться не только две вершины, но любое подмножество вершин графа, то такой граф называется гиперграфом. В таком случае $E$ - семейство непустых (необязательно различных) подмножеств множества $V$, называемых рёбрами гиперграфа. Гиперграф относится как к теории множеств так и к теории графов, поэтому его стоит учитывать как отдельный вид графа, так как с его помощью решаются определенные типы задач. Пример гиперграфа представлен на рисунке 7.

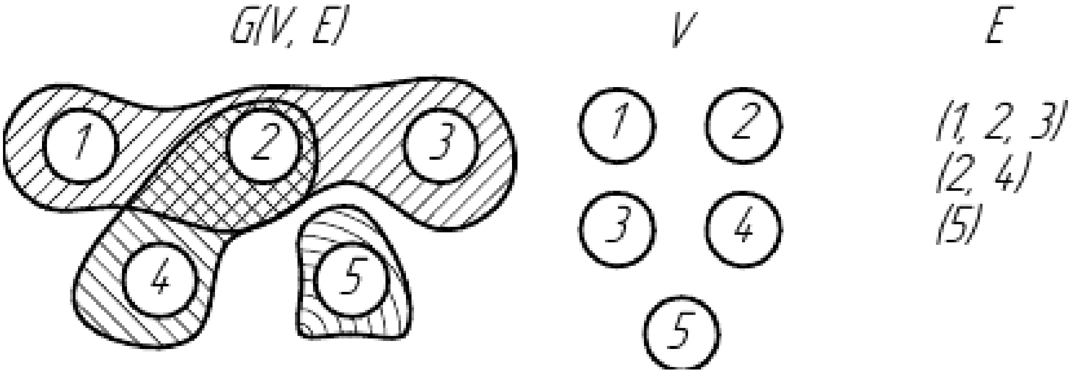



Рисунок 7 - Гиперграф

## Способы представления графов

В зависимости от вида графа, его сложности и характера работы с данными о графе используют различные способы представления графа. Основными способами представления графа являются:

- Матрица смежности;
- Матрица инцидентности;
- Список смежности;
- Список рёбер(инцидентности);

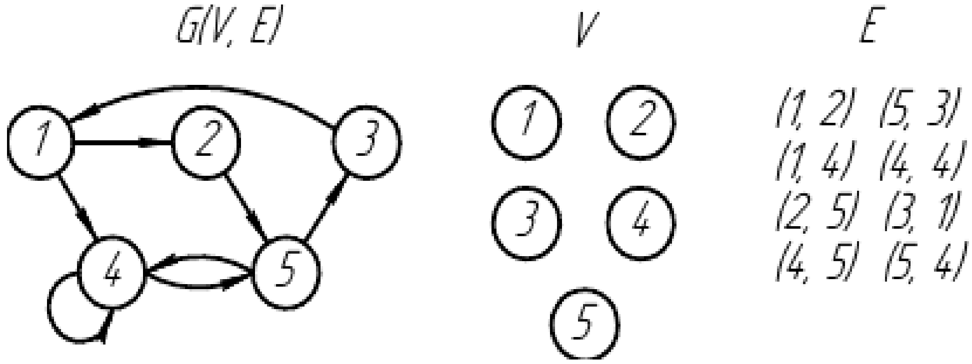
Рассмотрим каждый способ подробнее, и определим какие преимущества имеют те или иные способы представления графа.

## Матрица смежности

Смежность – понятие, используемое только в отношении двух ребер или в отношении двух вершин: два ребра инцидентные одной вершине, называются смежными; две вершины, инцидентные одному ребру, также называются смежными.

Граф $\boldsymbol{G}(\boldsymbol{V},\ \boldsymbol{E})$ представляется в виде матрицы $A$ размером $\boldsymbol{n} \times \boldsymbol{n}$, где $n$ – количество вершин в которой столбцы и строки соответствуют вершинам графа. В случае простого графа в элементе матрицы $\boldsymbol{a_{i,j}}$ стоит 0 если между i-ой и j-ой вершиной нет ребра и 1 если оно есть, то есть вершины смежны между собой. Таким образом матрица смежности для простого графа – бинарная матрица, которая содержит нули на главной диагонали. Если граф неориентированный, то матрица смежности симметрична относительно главной диагонали, если ориентированный – нет, в первом случае объем используемой памяти можно сократить вдвое $\boldsymbol{\theta}(|\boldsymbol{V}|^{2}/\boldsymbol{2})$. В случае псевдо и мультиграфов элемент матрицы $\boldsymbol{a_{i,j}}$ равен числу ребер из i-ой вершины в j-ую, в свою очередь петли соответствуют диагональным элементам матрицы и в случае ориентированного графа считаются за два ребра. Если граф взвешенный элементы матрицы смежности вместо чисел 0 и 1, указывающих на присутствие или отсутствие ребра, содержат веса самих ребер.

Матрица смежности предпочтительна только в случае плотных графов с большим числом ребер, так как она требует объем памяти $\boldsymbol{\theta}(|\boldsymbol{V}|^{\boldsymbol{2}})$, что критично для больших графов. Если граф разрежён, то большая часть памяти напрасно будет тратиться на хранение нулей, зато в случае неразреженных графов матрица смежности достаточно компактно представляет граф в памяти, используя примерно $\boldsymbol{n^2}$ бит памяти, по одному биту на элемент, что может быть на порядок лучше других способов представления. Однако уделять по одному биту на элемент возможно только при невзвешенном графе и если не присутствуют кратные ребра, тогда матрица бинарна. Но в противном случае на один элемент может приходиться несколько байт и тогда матрица смежности очень неэффективный по памяти способ представления. Также если граф взвешенный и в нем имеются кратные ребра, то такой граф нельзя представить в виде матрицы смежности. Пример представления графа в виде матрицы смежности показан на рисунке 8.




Матрица смежности

| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 1 | 0 |
| 2 | 0 | 0 | 0 | 0 | 1 |
| 3 | 1 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 0 | 1 | 1 |
| 5 | 0 | 0 | 1 | 1 | 0 |

Рисунок 8 – Матрица смежности

Достоинства матрицы смежности:

- Сложность проверки наличия ребра между двумя вершинами $\boldsymbol{O(1)}$;
- Эффективное использование памяти для плотных графов.

Недостатки матрицы смежности:

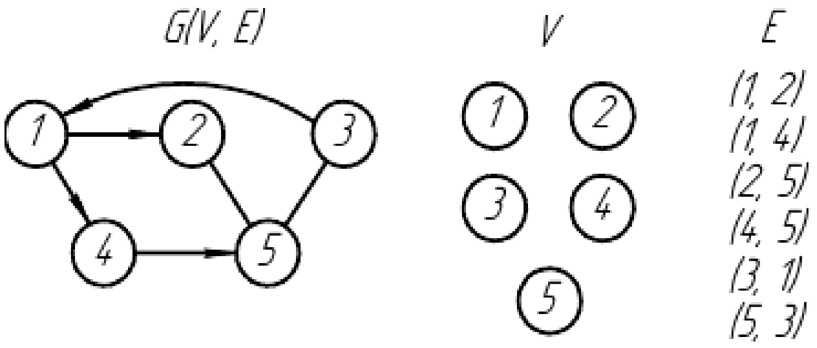
- Объем памяти $\boldsymbol{\theta(|V|^2)}$, что критично для больших неплотных графов;
- Сложность перебора всех вершин смежных с данной: $\boldsymbol{O(|V|)}$;
- Нельзя представить взвешенный мультиграф;
- Нельзя представить гиперграф;

## Матрица инцидентности

Инцидентность – понятие, используемое только в отношении ребра и вершины: две вершины (или два ребра) инцидентными быть не могут. Если $\boldsymbol{v_1}$, $\boldsymbol{v_2}$ – вершины, а $\boldsymbol{e = \{v_1,\ v_2\}}$ – соединяющее их ребро, тогда вершина $\boldsymbol{v_1}$ и ребро $\boldsymbol{e}$ инцидентны, также как $\boldsymbol{v_2}$ и $\boldsymbol{e}$.

Граф $\boldsymbol{G(V,\ E)}$ представляется в виде матрицы *A* размером $\boldsymbol{n \times m}$, где $n$ – количество вершин, а $m$ – количество вершин в которой столбцы матрицы соответствуют ребрам, строки - вершинам графа. В случае простого графа в элементе матрицы $\boldsymbol{a_{i,j}}$ стоит 0 если между i-ой вершиной и j-ым ребром нет инцидентности и наоборот 1 если она есть. В случае ориентированного графа каждой дуге $\boldsymbol{e = \{v_1,\ v_2\}}$, ставится в соответствующем $e$ столбце: 1 в строке вершины $\boldsymbol{v_1}$, и -1 в строке вершины $\boldsymbol{v_2}$, если связей между ребром и вершиной нет, то в соответствующую ячейку ставится 0. В случае если представляется псевдограф, то есть присутствуют петли, то в столбец ставится одна 1, иначе в матрице в каждом столбце должно быть ровно две ненулевых ячейки. Если представляется гиперграф, то в столбце тоже может быть отличное от двух число ненулевых ячеек. В случае взвешенного графа вместо 1 и -1 в соответствующие ячейки записывается вес самих ребер.

Матрица инцидентности может использоваться для любых видов графов. Так как в столбце данной матрицы в случае простого графа находится только два ненулевых элемента, то уже при небольшом числе вершин матрица будет разреженной от чего память будет неэффективно использоваться. Однако для невзвешенного и неориентированного графа матрица будет бинарной, от чего ее можно более компактно хранить. Матрица инцидентности не позволяет проверять наличие ребер между двумя вершинами быстрее чем за $O(|E|)$, что довольно долго, а удалять ребро быстрее чем за $O(|V|)$, так как последний столбец копируется в столбец удаляемого ребра. Пример представления графа в виде матрицы инцидентности показан на рисунке 9.




Матрица инцидентности

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 0 | -1 | 0 |
| 2 | -1 | 0 | 1 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 1 | 1 |
| 4 | 0 | -1 | 0 | 1 | 0 | 0 |
| 5 | 0 | 0 | 1 | -1 | 0 | 1 |

Рисунок 9 – Матрица инцидентности

Достоинства матрицы смежности:

- Можно представить любой вид графа;
- Удобно определять число входов и выходов в вершину;

Недостатки матрицы смежности:

- Объем памяти $\boldsymbol{\theta}(|\boldsymbol{V}| \cdot |\boldsymbol{E}|)$, матрица разрежена;
- Сложность перебора всех вершин смежных с данной: $\boldsymbol{O}(|\boldsymbol{E}|)$;
- Сложность удаления ребра $\boldsymbol{O}(|\boldsymbol{V}|)$;

## Список смежности

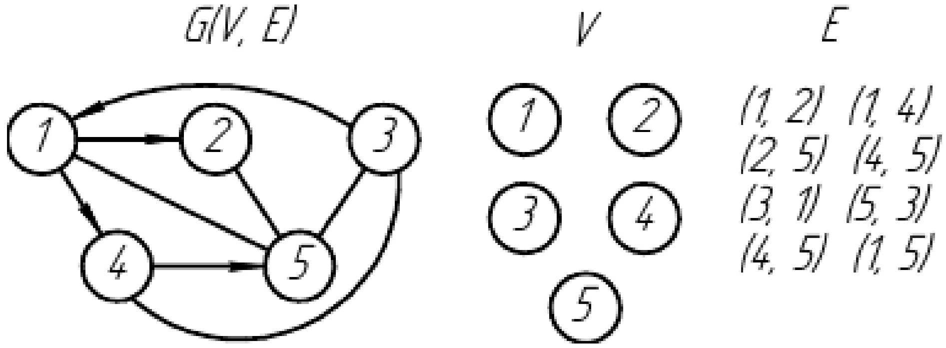
Граф $\boldsymbol{G}(\boldsymbol{V}, \boldsymbol{E})$ представляется в виде списка где каждой вершине соответствует строка в которой хранится список смежных вершин. Так как количество смежных вершин у различных вершин различно, данная структура представляет собой не матрицу, а список списков. В случае взвешенного графа в списке смежных вершин также указывается вес ребра для каждой смежной вершины. Если граф неориентированный, то из-за того что рассматривается смежность вершин будет иметь место дублирование и сумма длин всех списков будет $2|E|$, при ориентированном графе дублирования не будет и память будет использоваться более эффективно. Недостатком списка смежности является, то что в плотном графе для определения смежности двух вершин, требуется поиск по списку сложностью $\boldsymbol{O}(|\boldsymbol{V}|)$.

Для реализации данной структуры используется несколько способов, различающиеся особенностями ассоциации вершин и коллекциями соседей, и способами представления ребер и вершин:

- Использование хэш-таблицы для ассоциации каждой вершины со списком смежных вершин. Нет явного представления ребер.
- Вершины представляются в массиве, индекс элемента соответствует номеру вершины, каждый элемент массива ссылается на однонаправленный связный список соседних вершин.
- Объектно-ориентированный список смежности, который содержит классы вершин и ребер. В списке вершин, каждый объект вершины содержит ссылку на коллекцию рёбер, а каждый объект ребра содержит ссылки на вершины начала и конца ребра.

В отличии от матричных представлений списки смежности нельзя представить в бинарном виде, поэтому их выгодно использовать, если граф разрежен, так как объем используемой памяти: $\boldsymbol{\theta}(|\boldsymbol{V}| + |\boldsymbol{E}|)$, в отличии от матричных представлений в памяти не хранится преобладающее количество нулей. Также список смежности выгоден в операциях над графом по определению степени вершины, вставке и удалении вершины и обходе графа.

Пример списка смежности представлен на рисунке 10.




Список смежности

| | | | |
|---|---|---|---|
| 1 | 2 | 4 | 5 |
| 2 | 5 | | |
| 3 | 1 | 4 | 5 |
| 4 | 3 | 5 | |
| 5 | 2 | 3 | 4 |

Рисунок 10 – Список смежности

Достоинства списка смежности:

- Объем памяти $\boldsymbol{\theta}(|\boldsymbol{V}| + |\boldsymbol{E}|)$, оптимально для не насыщенных графов
- Позволяет быстро перебирать смежные вершины, и обходить граф;

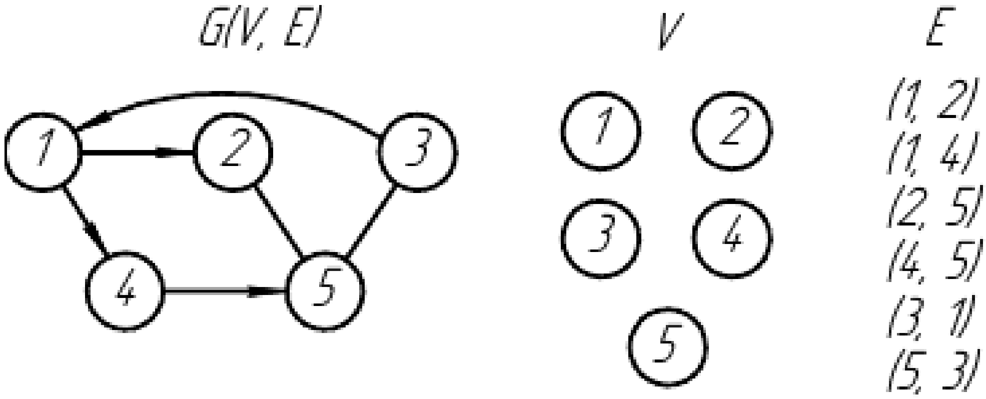
Недостатки списка смежности:

- Не оптимален для насыщенных графов (количество ребер $\sim\boldsymbol{n}^{2}$);
- Для взвешенных графов приходится усложнять структуру списка смежных вершин добавлением веса;
- Сложность определения ребра между вершинами $\boldsymbol{O}(|\boldsymbol{V}|)$.

**Список ребер(инцидентности)**

Граф $\boldsymbol{G}(\boldsymbol{V},\ \boldsymbol{E})$ представляется в виде списка где каждому ребру соответствует строка в которой хранится список инцидентных данному ребру вершин. В отличии от списка смежности в реализации списка ребер в массиве хранятся не объекты вершин с ссылками на списки смежных вершин, а объекты ребер с ссылками на список инцидентных вершин. Также как и матрица инцидентности, с помощью списка ребер можно представить любой граф, в том числе и гиперграф. В случае взвешенного графа, помимо инцидентных вершин хранится вес ребра, а если граф смешанный то неориентированные ребра записываются в список ребер дважды. В общем случае список ребер представляет из себя таблицу с количеством строк равным количеству ребер и двумя столбцами в случае невзвешенного графа и тремя столбцами в случае взвешенного графа.

Список рёбер наиболее компактный способ представления графов, поэтому его применяют для внешнего хранения или обмена данными. На этом достоинства списка ребер заканчиваются, так как такие операции как: определение ребра между двумя вершинами, удаление вершины, обход графа, определение степени вершины и т.п. делать не оптимально в данном представлении. Пример списка ребер представлен на рисунке 11.




Список ребер

| | | |
|---|---|---|
| 1 | 1 | 2 |
| 2 | 1 | 4 |
| 3 | 4 | 5 |
| 4 | 3 | 1 |
| 5 | 2 | 5 |
| 6 | 5 | 2 |
| 7 | 3 | 5 |
| 8 | 5 | 3 |

Рисунок 11 – Список рёбер

Достоинства списка смежности:

- Объем памяти $\boldsymbol{\theta}(|\boldsymbol{E}|)$, наиболее компактно;
- Можно представить любой вид графа;

Недостатки списка смежности:

- Не оптимален для насыщенных графов (количество ребер $\sim\boldsymbol{n}^{\boldsymbol{2}}$);
- Операции над графом, за исключением удаления ребра, имеют сложность $\boldsymbol{O}(|\boldsymbol{E}|)$, что критично для насыщенных графов;

### Сравнение способов представления

Каждый способ представления графа имеет свои преимущества и недостатки, сравним их по объему памяти и сложности выполнения операций над графом, таких как: обход графа, проверка наличия ребра $\{\boldsymbol{v_1},\ \boldsymbol{v_2}\}$, определение степени вершины, вставка или удаление вершины, вставка или удаление ребра. Результаты сравнения представлены в таблице 1.

Таблица 1 – Сравнение способов представления графа

| Операция | Матрица смежности | Матрица инцидентности | Список смежности | Список рёбер |
|---|---|---|---|---|
| Объем используемой памяти | $\boldsymbol{\theta}(\|\boldsymbol{V}\|^{\boldsymbol{2}})$ | $\boldsymbol{\theta}(\|\boldsymbol{V}\| \cdot \|\boldsymbol{E}\|)$ | $\boldsymbol{\theta}(\|\boldsymbol{V}\| + \|\boldsymbol{E}\|)$ | $\boldsymbol{\theta}(\|\boldsymbol{E}\|)$ |
| Проверка на наличие ребра $\{\boldsymbol{v_1},\ \boldsymbol{v_2}\}$ | $\boldsymbol{O}(\boldsymbol{1})$ | $\boldsymbol{O}(\|\boldsymbol{E}\|)$ | $\boldsymbol{O}(\|\boldsymbol{V}\|)$ | $\boldsymbol{O}(\|\boldsymbol{E}\|)$ |
| Определение степени вершины | $\boldsymbol{O}(\|\boldsymbol{V}\|)$ | $\boldsymbol{O}(\|\boldsymbol{E}\|)$ | $\boldsymbol{O}(\boldsymbol{1})$ | $\boldsymbol{O}(\|\boldsymbol{E}\|)$ |
| Вставка или удаление вершины | $\boldsymbol{O}(\|\boldsymbol{V}\|)$ | $\boldsymbol{O}(\|\boldsymbol{E}^{\boldsymbol{2}}\|)$ | $\boldsymbol{O}(\boldsymbol{d})$ | $\boldsymbol{O}(\|\boldsymbol{E}^{\boldsymbol{2}}\|)$ |
| Вставка или удаление ребра | $\boldsymbol{O}(\boldsymbol{1})$ | $\boldsymbol{O}(\boldsymbol{1})$ | $\boldsymbol{O}(\boldsymbol{1})$ | $\boldsymbol{O}(\|\boldsymbol{1}\|)$ |
| Обход графа | $\boldsymbol{O}(\|\boldsymbol{V}\|^{\boldsymbol{2}})$ | $\boldsymbol{O}(\|\boldsymbol{V}\| \cdot \|\boldsymbol{E}\|)$ | $\boldsymbol{O}(\|\boldsymbol{V}\| + \|\boldsymbol{E}\|)$ | $\boldsymbol{O}(\|\boldsymbol{E}^{\boldsymbol{2}}\|)$ |
| Представление любого вида графа | Нет | Да | Нет | Да |

Для представления графа в алгоритмах в основном используется список смежности, благодаря преимуществам в скорости выполнения основных операций, однако для передачи данных о графе и хранении используется список ребер благодаря большей компактности.

Для подтверждения аналитических характеристик способов представления графов была получена зависимость времени выполнения операций удаления вершин и ребер от количества ребер в графе. При получении зависимости использовался граф из 1000 вершин и со случайно составленными ребрами, количество которых менялось от 1 тыс. до 1024 тыс., увеличиваясь на два, поэтому ось количества ребер в графе логарифмическая. Зависимости показывают время затрачиваемое на удаление 10 случайных вершин и на удаление 10 случайных ребер. Графики полученных зависимостей представлены на рисунках 12 и 13.

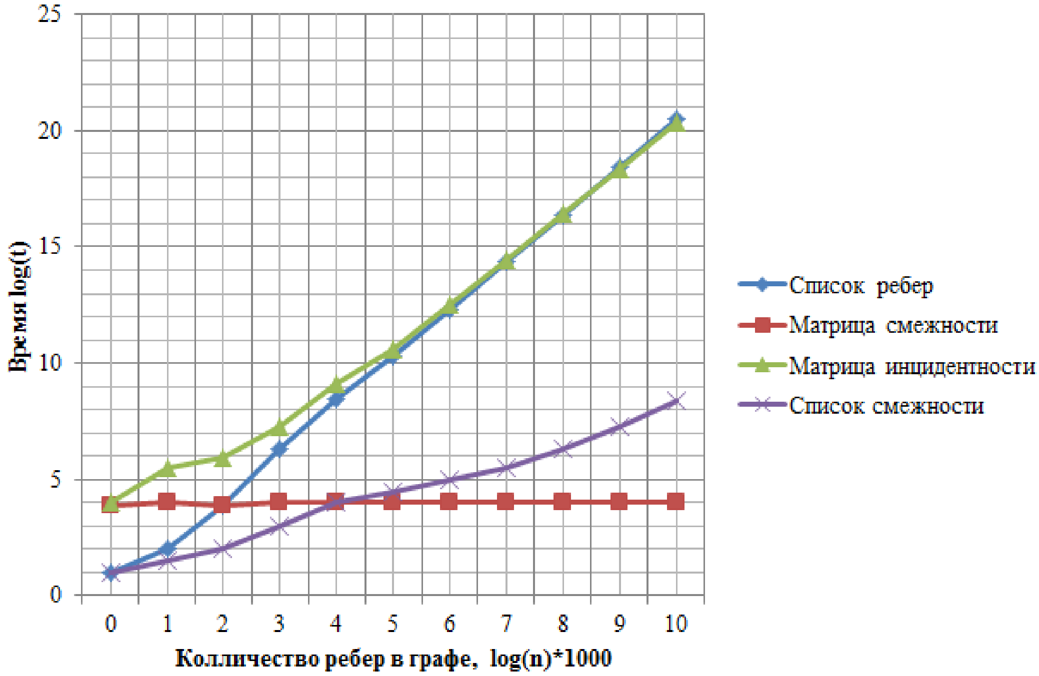



Рисунок 12 – График зависимости времени от количества ребер, для операции удаления вершины

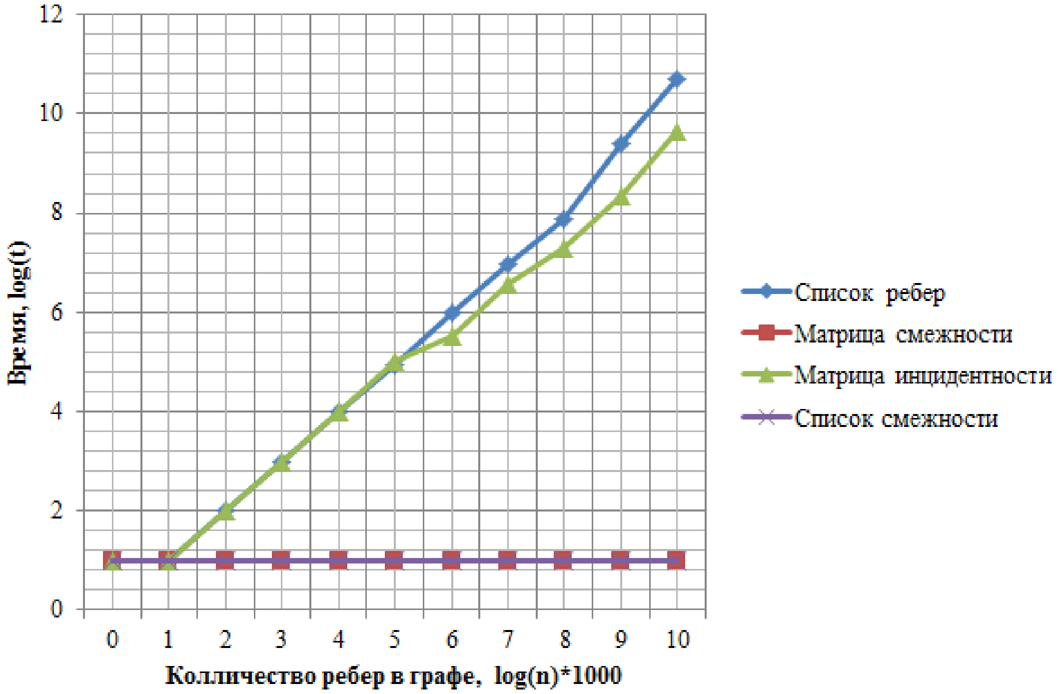



Рисунок 13 – График зависимости времени от количества ребер, для операции удаления ребра

Можно заключить, что данные графики подтверждают теоретическую сложность выполнения операций удаления ребра и вершины для разных способов представления графа.

# Заключение

Для представления графа в алгоритмах в основном используется список смежности, благодаря преимуществам в скорости выполнения основных операций, так как он позволяет быстро получать соседей вершин, однако для передачи данных о графе и хранении используется список ребер благодаря большей компактности. В свою очередь матрицу смежности выгодно использовать для насыщенных графов, но в случае взвешенных графов или мультиграфов, структура усложняется, и объем используемой памяти становится еще больше. Матрица инцидентности используется крайне редко из-за большого объема используемой памяти, однако может применяться для быстрого нахождения циклов в графе.

## Список литературы


1. Кормен, Т., Лейзерсон, Ч., Ривест, Р., Штайн, К. Алгоритмы: построение и анализ, под ред. И. В. Красикова. — 2-е изд. — М.: Вильямс, 2005. — 1296 с.

2. Харари Ф. Теория графов. — М.: Мир. — 1973. — 300 с.

3. Граф (математика) [Электронный ресурс] // https://ru.wikipedia.org/wiki/Граф_(математика)#Простой_граф.

4. Графы. Способы представления графа в программе [Электронный ресурс] // https://pro-prof.com/forums/topic/graph-representations.

5. Понятие и представление графа: матрица смежности, список смежности [Электронный ресурс] // https://brestprog.by/topics/graphs.