Санкт-Петербургский государственный электротехнический
университет имени В. И. Ленина "ЛЭТИ"

билеты

КОМБИНАТОРИКА И ТЕОРИЯ ГРАФОВ

Студент:	Придчин В.Е.
Группа:	2308
Лектор:	Зяблицева Л.

LaTeX

Санкт-Петербург
2024

1 Основные определения теории графов. Смежность и инцидентность вершин и ребер графа. Степени вершин в графе и орграфе. Теоремы о сумме степеней вершин в графе и орграфе. Матрицы смежности и инцидентности. Найти матрицы смежности и инцидентности указанного графа.

V – множество вершин.
E – множество пар вида $(u, v) : u, v \in V$ (множество ребер).

Опр: Графом – называют совокупность 2-ух множеств непустого множества E

$$E = \{(u, v) : u, v \in V\},\ G(V, E) - \textit{обозначение графа}$$

Опр: Петля – пара вида (v, v) в множестве E.

Опр: Кратные ребра – одинаковые пары в множестве E. Количество кратных ребер - кратность ребра.

Существуют следующие виды графов:

1. Псевдограф – в графе могут быть и *кратные ребра*, и *петли.*
2. Мультиграф – в графе есть *кратные ребра*, но нет *петель.*
3. Простой граф – отсутствуют и *кратные ребра* и *петли.*

Опр: Ориентированный граф (орграф) – граф с ориентированными ребрами.

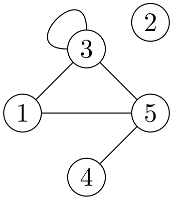
Опр: Если $e = (u, v)$ – ребро неориентированного графа, то u, v - концы ребра.

Опр: Если $e = (u, v)$ – ребро (дуга) ориентированного графа, то u - начало ребра, v - конец ребра.

Опр: a, b *смежные* $\Leftrightarrow e = (u, v)$.

Опр: u, v *инцидентны ребру* $e \Leftrightarrow e = (u, v)$.

Опр: Степенью вершины v неориентированного графа называется количество ребер инцидентных данной вершине, $\delta(v)$ (петлю считают два раза).



$$\delta(1) = 2 \qquad \delta(4) = 1$$
$$\delta(2) = 0 \qquad \delta(5) = 3$$
$$\delta(3) = 4 \quad \Rightarrow sum = 10$$

Теорема: Сумма степеней вершин неориентированного графа равна удвоенному числу ребер

$$\sum_{u \in V} \delta(v) = 2r, \ \textit{где } r \textit{ - число ребер}$$

Док-во: Теорема справедлива, так как вклад каждого ребра равен двум.

□

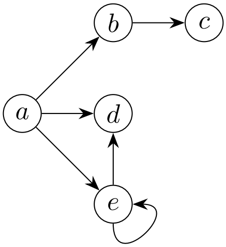
Опр: Если степень вершины равна нулю, то вершина *изолированная*, $\delta(v) = 0$.

Опр: Если степень вершины равна единице, то вершина *висячая*, $\delta(v) = 1$

Опр: Полустепенью исхода(захода) вершина v ориентированного графа называют количество ребер исходящих(заходящих) в данную вершину.
$\delta^-(v)$ – полустепень исхода
$\delta^+(v)$ – полустепень захода

Теорема: Для орграфа справедливо равенство

$$\sum_{u \in V} \delta^-(v) = \sum_{u \in V} \delta^+(v) = r, \ \textit{где } r \textit{ – число ребер}$$



$$\delta^-(a) = 3 \qquad \delta^+(a) = 0$$
$$\delta^-(b) = 1 \qquad \delta^+(b) = 1$$
$$\delta^-(c) = 0 \qquad \delta^+(c) = 1$$
$$\delta^-(d) = 0 \qquad \delta^+(d) = 2$$
$$\delta^-(e) = 2 \qquad \delta^+(e) = 2$$
$$\sum \delta^-(v) = 6; \quad \sum \delta^+(v) = 6$$

Опр: Матрицей смежности графа(орграфа) называют квадратную матрицу размерностью n, где $n = |v|$ (мощность множества вершин), в котором $a_{ij} = k$, где k - число ребер (v_i, v_j)

Опр: Пусть $G(V, E)$ – неориентированный граф. Матрицей инцидентности неориентированного графа называется матрица B размером $n * r$, $|v| = n, |E| = r$, где каждый элемент матрицы:

$$b_{ij} = \begin{cases} 1, & \text{если } v_i \text{ инцидентно ребру } e_j \\ 0, & \text{иначе} \end{cases}$$

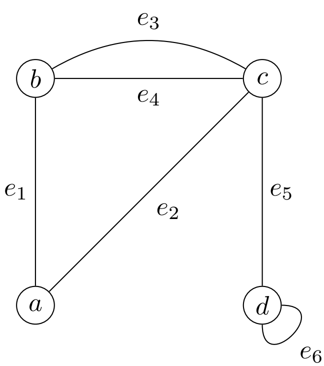
Такая матрица будет симметричной.

Опр: Пусть $G(V, E)$ – ориентированный граф. Матрицей инцидентности ориентированного графа называется матрица B размером $n * r$, $|v| = n, |E| = r$, где каждый элемент матрицы:

$$b_{ij} = \begin{cases} -1, & \text{если ребро } e_j \text{ выходит из } v_i \\ 1, & \text{если ребро } e_j \text{ входит из } v_i \\ 0, & \text{иначе} \end{cases}$$

Если есть петля, то на соответствующее место ставят любое число.

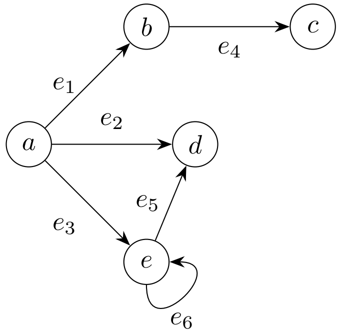
Поиск матрицы смежности A и инцидентности B для неориентированного графа:



$$A = \begin{array}{c} \\ a \\ b \\ c \\ d \end{array} \begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 2 & 0 \\ 1 & 2 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix} \end{array}$$

$$B = \begin{array}{c} \\ a \\ b \\ c \\ d \end{array} \begin{array}{c} \begin{array}{cccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \end{array}$$

Поиск матрицы смежности A и инцидентности B для ориентированного графа:

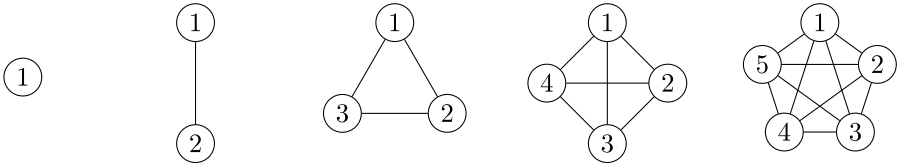


$$A = \begin{array}{c} \\ a \\ b \\ c \\ d \\ e \end{array} \begin{array}{c} \begin{array}{ccccc} a & b & c & d & e \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix} \end{array}$$

$$B = \begin{array}{c} \\ a \\ b \\ c \\ d \\ e \end{array} \begin{array}{c} \begin{array}{cccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{array} \\ \begin{pmatrix} -1 & -1 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & -1 & 1 \end{pmatrix} \end{array}$$

2 Полные и двудольные графы. Число ребер в полном графе с n вершинами и в полном двудольном графе (вывод формул).

Опр: *Полный граф* – простой неориентированный граф у которого любые две вершины смежны.

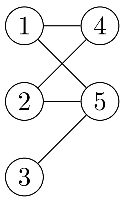


K_1 K_2 K_3 K_4 K_5

Количество полных ребер в графе $K_n = C_n^2 = \frac{n!}{2!(n-2)!} = \frac{n(n-1)}{2}$.

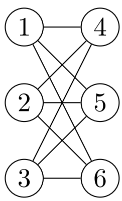
Опр: *Двудольный граф* – граф, если его множество вершин V можно разделить на подмножество V_1 и V_2 такое что: $V_1 \cap V_2 = \varnothing$
$V_1 \cup V_2 = V$

Смежными могут быть только вершины из разных долей графа.



$K_{2,3}$
$V = \{1, 2, 3, 4, 5\}$
$V_1 = \{1, 2, 3\}$
$V_2 = \{4, 5\}$

Опр: *Полный двудольный граф* – каждая вершина одной доли соединяется с другой. K_{t_1,k_2}, где t_1, k_2 – количество вершин в долях графа.



$K_{3,3}$
$V = \{1, 2, 3, 4, 5, 6\}$
$V_1 = \{1, 2, 3\}$
$V_2 = \{4, 5, 6\}$

В полном двудольном графе содержится $t_1 \cdot t_2$ ребер.

3 Изоморфизм и гомеоморфизм графов. Примеры изоморфных и гомеоморфных графов. Способы проверки изоморфизма графов. Инварианты графа. Дополнение графа, проверка изоморфизма графов с помощью дополнений. Выяснить, являются ли графы G1 и G2 изоморфными, гомеоморфными.

Опр: *Изоморфизм графов* – биективное отображение $\varphi : V_1 \to V_2$, такое что:

$$(\forall u, v \in V)((u, v) \in E_1) \Leftrightarrow (\varphi(u), \varphi(v)) \in E_2$$

Изоморфные графы обозначаются: $G_1 \cong G_2$

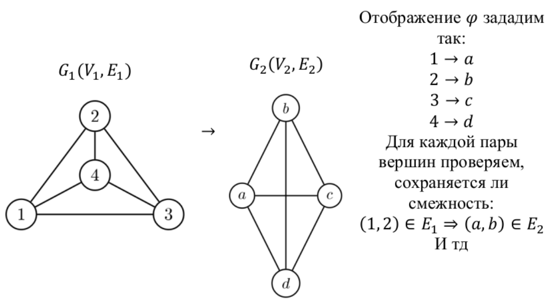
Изоморфные объекты *не различимы* с точки зрения математики. Это экземпляры одного и того же математического объекта. Изоморфные объекты имеют одинаковое число элементов и свойств.

Изоморфизм графов можно определить:

1. По определению (найдя биективное отображение множества вершин, сохраняющее смежность)
2. Перерисовав один из графов так, чтобы изображение совпало с другим графом
3. Сравнив матрицы смежности графов

Теорема: Графы изоморфны тогда и только тогда, когда матрицу смежности одного из них можно получить из матрицы смежности другого путём одновременной перестановки местами i-ой и j-ой строк и столбцов.

Пример изоморфных графов:



Опр: *Инвариантом изоморфизма графов* называется некоторое (обычно числовое) значение или упорядоченный набор значений, характеризующий структуру графа и не зависящий от способа его задания.

Инвариантами графа являются:

1. Количество вершин
2. Количество ребер
3. Набор степеней вершин (упорядоченный)
4. Определитель матрицы смежности
5. Количество компонент связности
6. Хроматическое число и т.д.

Пусть $G(V, E)$ – простой неориентированный граф.

Опр: Дополнением графа G называется граф $\overline{G}(\overline{V}, \overline{E})$, у которого $\overline{V} = V$, в множестве $\overline{E}$ содержатся ребра полного графа, которых нет в множестве E.

Теорема: Простые графы G и H изоморфны тогда и только тогда, когда изоморфны их дополнения.

$$G \cong H \Leftrightarrow \overline{G} \cong \overline{H}$$

Опр: Граф является самодополнительным, если он изоморфен своему дополнению.

Понятие изоморфизма можно дать и для произвольных графов.

Опр: Граф $G_1(V_1, E_1)$ и $G_2(V_2, E_2)$ *изоморфны*, если $\exists$ биективные отображения $\varphi : V_1 \to V_2$ и $h : E_1 \to E_2$ такие, что:

$$e = (u, v) \in E_1 \Leftrightarrow h(e) = (\varphi(u), \varphi(v)) \in E_2$$

4 Маршруты, цепи, циклы в графе. Метрические характеристики графа. Найти эксцентриситет вершины указанного графа, радиус, диаметр графа, центральные и периферийные вершины указанного графа.

Опр: *Маршрут* (путь) в графе – это конечная чередующаяся последовательность смежных вершин и ребер, соединяющих эти вершины.

Маршрут – последовательность $v_1e_1v_2e_2\dots e_{s-1}v_se_sv_{s+1}\dots v_k$, в которой чередуются все вершины и ребра. $e_s = (v_s, v_{s+1})$.

Опр: *Длина маршрута* – количество ребер в маршруте.

Опр: *Нуль-маршрут* – последовательность из одной вершины v.

Опр: *Замкнутый маршрут (циклический)* – последовательность, в которой совпадает начальная вершина с конечной. Иначе – *незамкнутый (открытый)*.

Опр: *Цепь* – незамкнутый маршрут, где все *ребра* попарно различны.

Опр: *Простая цепь* – цепь, где все *вершины* попарно различны.

Опр: *Цикл* – циклический маршрут, где все *ребра* попарно различны.

Опр: *Простой цикл* – цикл, где все *вершины, кроме первой и последней* попарно различны.

Теорема: Из любого цикла можно выделить простой цикл с той же начальной вершиной.

Док-во: Пусть $\exists$ цикл $v_1e_1v_2e_2\dots v_se_s\dots v_je_j\dots v_ne_nv_1$. Если все вершины, кроме первой и последней, различны, то этот цикл простой.
Допустим в цикле есть совпадающие вершины (не первая и не последняя) $v_s = v_j$. Тогда удаляем часть маршрута между этими вершинами.

$$v_1e_1v_2e_2\dots v_se_s\dots v_je_j\dots v_ne_nv_1 \Rightarrow v_1e_1v_2e_2\dots v_se_j\dots v_ne_nv_1$$

Так как $v_s = v_j$, то полученная последовательность является маршрутом. Продолжая получим простой цикл. □

Теорема: Из любой цепи можно выделить простую цепь с теми же начальными и конечными вершинами (док-во аналогично).

Опр: *Связанные вершины в графе* – $\exists$ маршрут с началом в первой вершине и концом на второй.

Опр: *Связный граф* – две любые вершины являются *связанными.*

Пусть $G(V, E)$ – связный неориентированный граф $(v_i, v_j \in V)$.

Опр: *Расстояние между вершинами,* $d(v_i, v_j)$ – длина *кратчайшего* маршрута между этими вершинами.

Опр: *Эксцентриситет вершины графа* – число, максимальное из расстояний между этой вершиной и всеми остальными вершинами.

$$\varepsilon(v_i) = \max d(v_i, v_j), v_j \in V, i \neq j$$

Опр: *Диаметр графа,* $D(G)$ – максимальный из всех эксцентриситетов вершин данного графа.

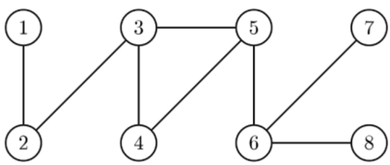
Опр: *Радиус графа,* $R(G)$ – минимальный из всех эксцентриситетов вершин данного графа.

Опр: *Периферическая вершина* – ее эксцентриситет равен диаметру.

Опр: *Центральная вершина* – ее эксцентриситет равен радиусу.

Опр: *Центр графа* – множество всех центральных вершин.

Пример: Найти эксцентриситеты вершин, диаметр и радиус графа, периферийные и центральные вершины.



$\varepsilon(1) = 5, \varepsilon(2) = 4, \varepsilon(3) = 3, \varepsilon(4) = 3,$
$\varepsilon(5) = 3, \varepsilon(6) = 4, \varepsilon(7) = 5, \varepsilon(8) = 5.$
$D(G) = 5, R(G) = 3.$

Периферийные: 1, 7, 8
Центральные: 3, 4, 5

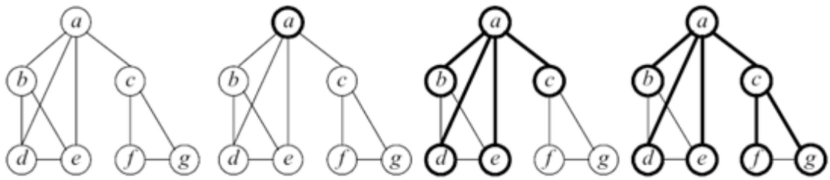
5 Алгоритмы обхода графа в глубину и ширину.

Опр: Обход графа (поиск на графе) — это процесс систематического просмотра всех ребер или вершин графа с целью отыскания ребер или вершин, удовлетворяющих некоторому условию.

Опр: Поиск в ширину (BFS) нужен для нахождения расстояний между вершин в связном графе. Алгоритм по принципу напоминает "пожар".

Алгоритм поиска в ширину:

1. Все вершины окрашиваются в белый цвет.
2. Выбирается первая вершина, раскрашивается в черный цвет и заносится в очередь.
3. Посещается первая вершина из очереди. Все смежные с ней белые вершины раскрашиваются в черный цвет и заносятся в очередь. После этого эта вершина удаляется из очереди.
4. Повторяется шаг 3 до тех пор пока очередь не пуста.



Сложность поиска в ширину при нематричном представлении графа равна $O(n+m)$, так как рассматриваются все n вершин и m ребер. Использование матрицы смежности приводит к оценке $O(n^2)$.

Если граф не является связным, то нужно добавить шаг 5, в котором написать, что нужно повторять 2 шаг до тех пор, пока в графе не останется белых вершин.

Если сначала присвоить первой вершине значение 0, потом смежным с ней вершинам 0+1 =1, и так далее, то в итоге для каждой вершины будет найдено расстояние от первой вершины.

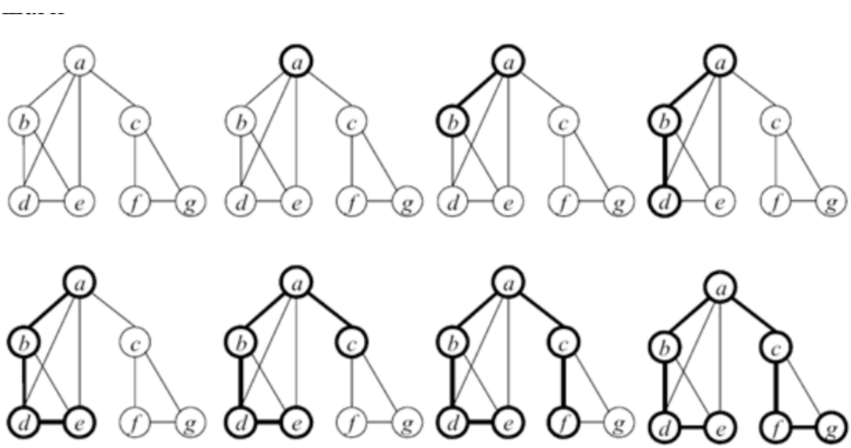
Опр: Поиск в глубину (DFS) исследует конкретную ветку в графе и только потом переходит к другой (если они останутся нерассмотренными).

Вершины графа могут быть раскрашены в три цвета: белый цвет означает, что в вершине еще не были, серый – что в вершине были, но еще вернемся, черный – что были и больше не рассматриваем.

Алгоритм поиска в глубину:

1. Всем вершинам графа присваиваем белый цвет.
2. Выбираем первую вершину и раскрашиваем в серый цвет.
3. Для последней раскрашенной в серый цвет вершины выбираем белую смежную ей вершину (если такая вершина есть), раскрашиваем ее в серый цвет и переходим к шагу 3.
4. Повторяем шаг 3 до тех пор, пока все вершины не будут раскрашены в черный цвет.

Если для рассматриваемой вершины белых смежных с ней вершин нет, то рассматриваемую вершину раскрашиваем в черный цвет и переходим к шагу 3.



Временная сложность алгоритма зависит от представления графа. Если применена матрица смежности, то временная сложность равна $O(n^2)$, а если нематричное представление – $O(n + m)$: рассматриваются все вершины и все ребра.