

ОСНОВНЫЕ ПОНЯТИЯ

1 Основные понятия и определения теории графов

1.1. Начнем данный параграф с определения неориентированных и ориентированных графов.

1.1.1. Формальное и достаточно общее определение неориентированного графа таково.

Определение 1.1. Неориентированным графом G называется тройка

$$G = (V, E, I),$$

состоящая из

(1) (конечного) множества вершин $V = V(G)$, например,

$$V = \{1, 2, 3, 4\},$$

(2) (конечного) множества ребер $E = E(G)$, например,

$$E = \{a, b, c, d, e, f, g, h\},$$

(3) а также отображения $I: E \rightarrow V_2$, сопоставляющего любому ребру $e \in E$ неупорядоченную пару вершин $\{x, y\} \in V_2$, которую это ребро соединяет.

Вершины x и y называются *концевыми вершинами* ребра e . При этом говорят, что ребро e *инцидентно* своим концевым вершинам. Про любую из двух вершин x, y , в свою очередь, говорят, что она инцидентна ребру e .

В принципе, возможен случай $x = y$. Ребро $e \in E$, соответствующее паре $\{x, x\}$, называется обычно *петлей*. Кроме того, в общем случае у нас могут быть несколько различных ребер, соединяющих одну и ту же пару вершин $\{x, y\}$. Такие ребра называют кратными ребрами и говорят, что они образуют *мультиребро* графа G . Если пара вершин $\{x, y\}$ соединена между собой единственным ребром, то такое ребро иногда называют простым ребром графа.

Пример 1.2. Зададим отображение I в виде следующей таблицы:

E	a	b	c	d	e	f	g	h
V_2	$\{1, 3\}$	$\{2, 4\}$	$\{1, 3\}$	$\{3, 4\}$	$\{3, 4\}$	$\{1, 2\}$	$\{3, 4\}$	$\{2, 2\}$

Ей соответствует граф G , изображенный на рис.1. В этом графе ребра f и b являются простыми, кратные ребра a и c образуют мультиребро, соединяющее вершины 1 и 3, а ребро h представляет собой петлю.

1.1.2. С понятием инцидентности тесно связано важное понятие степени вершины графа G .

Определение 1.3. В неориентированном графе G *степенью* $\deg(x)$ или *валентностью* вершины x называется количество ребер, инцидентных x . Считается, что петля дает вклад, равный двум, в степень любой вершины. Вершина, степень которой равна нулю, называется *изолированной*. Если все вершины в графе G имеют одинаковую степень, то граф G называют *регулярным*.

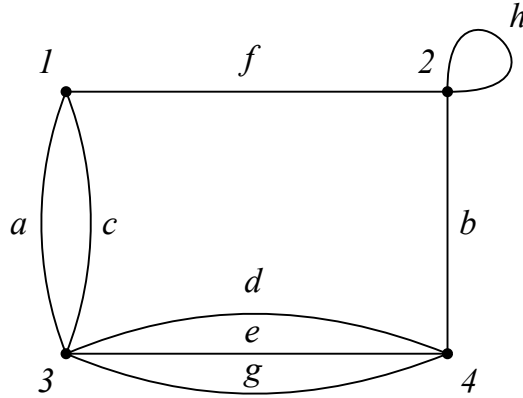


Рис. 1: Пример графа на четырех вершинах

Так, вершина 1 на рис.1 имеет степень, равную трем, а вершина 2 — степень, равную четырем. Следующее утверждение часто называют первой теоремой теории графов.

Теорема 1.4. В неориентированном графе G сумма степеней всех вершин равна удвоенному количеству всех ребер графа:

$$\sum_{x \in V(G)} \deg(x) = 2|E(G)|. \quad (1)$$

Доказательство практически очевидно — любое ребро дает вклад, равный двум, в стоящую слева сумму.

Следствие 1.5. Количество вершин в графе G , имеющих нечетную степень, четно.

1.1.3. Несмотря на достаточную очевидность доказательства теоремы 1.4, оно использует чрезвычайно важный прием, очень часто встречающийся и в теории графов, и в комбинаторике — так называемый двойной подсчет (double counting).

Неформально его можно себе представлять следующим образом. Предположим, что у нас имеется какое-то множество C элементарных объектов C_i , любой из которых мы можем рассматривать как составную часть двух различных, более сложно устроенных объектов A_j и B_k . Тогда количество $c = |C|$ всех элементарных объектов C_i мы всегда сможем сосчитать двумя способами. Именно, для любого сложного объекта первого типа A_j мы можем сосчитать количество a_j элементарных объектов, из которых состоит A_j , а затем просуммировать полученные числа a_j по j от 1 до n , где n есть количество всех сложных объектов первого типа. С другой стороны, мы можем для любого B_k сосчитать количество b_k элементарных объектов, из которых состоит сложный объект второго типа B_k , а затем просуммировать b_k по всем k от 1 до m , где m — количество объектов второго типа. Так как общее количество c всех элементарных объектов C_i , от способа подсчета не зависит, то мы в итоге получаем равенство вида

$$\sum_{j=1}^n a_j = c = \sum_{k=1}^m b_k, \quad (2)$$

которое и отражает некоторый конечный результат процедуры double counting.

При доказательстве первой теоремы теории графов мы в качестве элементарного объекта C_i можем взять так называемое *полуребро* графа G (semi-edges или darts), то есть половинку любого ребра, инцидентного вершине x (см.рис.2, на котором полуребра изображены синим цветом).

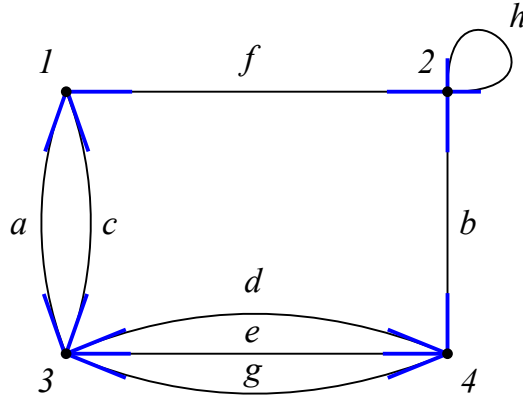


Рис. 2

Каждое ребро состоит ровно из двух полуребер, поэтому количество s всех полуребер равно удвоенному количеству $2m$, $m = |E(G)|$, всех ребер. С другой стороны, любую вершину x_j мы можем представить в виде “ежа”, то есть вершины вместе с инцидентными ей полуребрами. Количество a_j таких полуребер совпадает со степенью $\deg(x_j)$ вершины x_j . Суммируя эти числа a_j по всем $j = 1, \dots, n$, где n — количество всех вершин в графе, мы вновь получаем общее количество полуребер s . Формула (1) при этом есть не что иное, как соотношение (2), переписанное для рассматриваемого частного случая.

1.1.4. Описанную выше процедуру double counting можно формализовать с помощью так называемой матрицы инцидентности M_i . Именно, рассмотрим матрицу M_i размерами $n \times m$, строки которой отвечают объектам A_j , $j = 1, \dots, n$ первого типа, а столбцы — объектам B_k , $k = 1, \dots, m$ второго типа. Назовем объекты A_j и B_k *инцидентными* друг другу, если у нас существует хотя бы один элементарный объект, являющийся частью как A_j , так и B_k . Любой элемент $m_{j,k}$ матрицы M_i равен количеству элементарных объектов, входящих как в A_j , так и в B_k . При таком подходе левая часть равенства (2) подсчитывает количество элементарных объектов суммированием по строкам матрицы M_i , а правая часть (2) — суммированием по столбцам.

В случае неориентированного графа G строки матрицы M_i инцидентности графа G отвечают вершинам этого графа, а столбцы — его ребрам. Элемент $m_{j,k}$ матрицы M_i равен нулю, если вершина x_j и ребро e_k не инцидентны друг другу, $m_{j,k} = 2$ в случае, если e_k есть инцидентная вершине x_j петля, и $m_{j,k} = 1$ в остальных случаях.

Рассмотрим, к примеру, граф G , показанный на рисунках 1 и 2. Для такого графа матрица инцидентности имеет следующий вид:

$$M_i = \left(\begin{array}{c|cccccccc} V \setminus E & a & b & c & d & e & f & g & h \\ \hline 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 2 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 2 \\ 3 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 4 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \end{array} \right).$$

Видно, что сумма элементов в любом столбце этой матрицы равна двум, а сумма элементов в любой строке совпадает со степенью вершины x этого графа. Суммирование по строкам и суммирование по столбцам снова приводит нас к равенству (1).

1.2. Чаще всего на практике встречаются так называемые простые графы, к описанию которых мы и перейдем.

1.2.1. Начнем с определения простого графа.

Определение 1.6. Граф G называется *простым*, если он не содержит петель и мультиребер. Граф, не являющийся простым, часто называют *мультиграфом*.

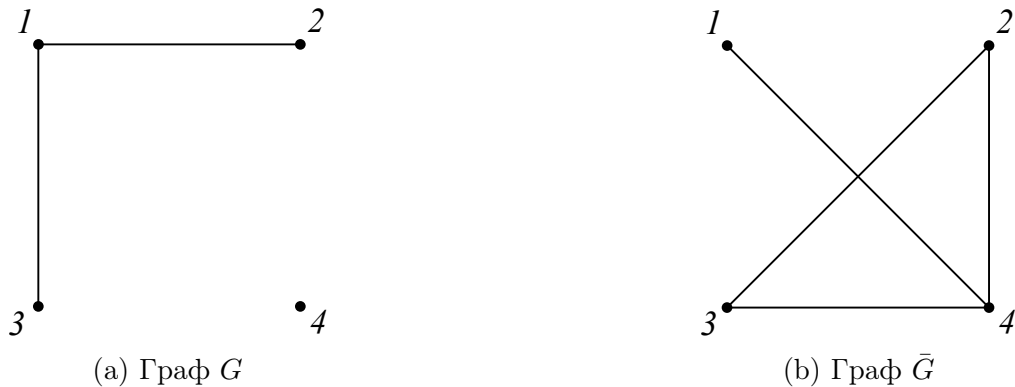


Рис. 3: Примеры простых графов

В качестве примера рассмотрим изображенный на рис.3,а простой граф G , построенный на четырех вершинах и имеющий два ребра. Видно, что в таком графе мы можем не вводить какое-то дополнительное специальное обозначение для ребер — любое ребро однозначно задается парой вершин, которые оно соединяет. Как следствие, для описания простого графа G нам достаточно задать множество V его вершин, а также множество E его ребер в виде некоторого подмножества множества всевозможных неупорядоченных пар множества вершин. Так, для графа G на рис.3,а мы имеем

$$V = \{1, 2, 3, 4\}, \quad E = \{\{1, 2\}, \{1, 3\}\}.$$

Итак, с формальной точки зрения любой простой неориентированный граф G , построенный на n вершинах, можно рассматривать как некоторое подмножество множества $V^{(2)}$ всех двухэлементных подмножеств множества $V(G)$ его вершин:

$$G \subseteq V^{(2)}.$$

Если же говорить менее формально, то простой граф — это граф, построенный на n вершинах, любая пара которых может быть соединена или не соединена ребром.

Сразу заметим, что матрица M_i инцидентности простого графа состоит только из нулей и единиц.

1.2.2. Разберем некоторые важные примеры простых графов. В качестве первого примера рассмотрим так называемый *полный* граф K_n (см.рис.4,а, на котором показан полный граф K_5), в котором любая вершина соединена со всеми оставшимися вершинами графа. Такой граф отвечает всему множеству $V^{(2)}$ двухэлементных подмножеств множества n вершин. Пустому подмножеству множества $V^{(2)}$ отвечает так называемый *пустой* граф, то есть граф, состоящий из n изолированных вершин (рис.4,б).

Пустой граф является дополнением к полному графу K_n (и обозначается он как $\bar{K}_n$) в смысле следующего определения.

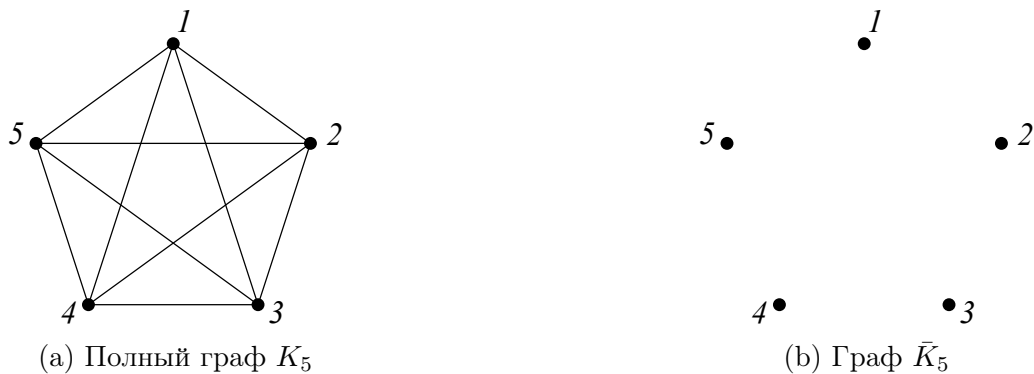


Рис. 4: Полный граф и пустой граф

Определение 1.7. Граф $\bar{G}$ называется *дополнением* к графу G , если множества вершин этих двух графов совпадают, а множество ребер графа $\bar{G}$ дополняет множество ребер $E(G)$ исходного графа G до множества ребер полного графа K_n .

На рис.3 в качестве примера приведены два графа, построенные на четырех вершинах — граф G и граф $\bar{G}$. Граф $\bar{G}$ из графа G можно получить, например, так: взять полный граф K_4 , построенный на том же количестве вершин, что и графы G и $\bar{G}$, и удалить из него ребра, принадлежащие графу G . Полученный в результате этой операции граф $\bar{G}$ и будет являться дополнением к графу G в смысле данного выше определения.

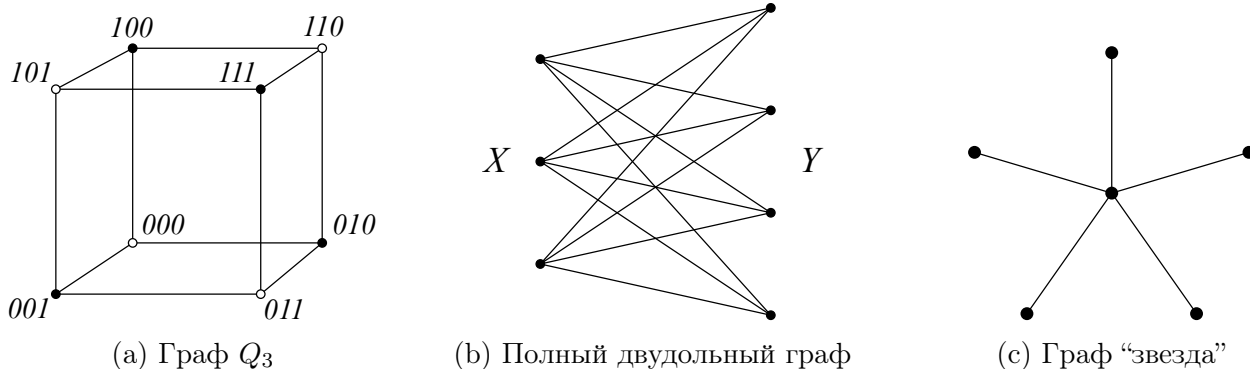


Рис. 5: Двудольные графы

1.2.3. При изучении теории графов нам часто будут встречаться еще несколько важных подклассов простых графов.

Определение 1.8. Граф G называется *двудольным* (рис.5), если множество $V(G)$ его вершин можно разбить на два блока X и Y так, что концы x и y любого ребра $e = \{x, y\} \in E(G)$ лежат в разных блоках этого разбиения. Обозначается двудольный граф с разбиением (X, Y) через $G[X, Y]$.

Простой граф $G[X, Y]$, в котором любая вершина из X соединена ребром с каждой вершиной из блока Y и наоборот, называется *полным двудольным графом* (рис.5,b). Часто такой граф обозначается через $K_{n,m}$, где $n = |X|$, а $m = |Y|$. Граф $K_{n,1}$ называется *звездой* (рис.5,c).

Еще два важных подкласса простого графа, называемые путем P_n и циклом C_n , показаны на рис.6,a и рис.6,b. Количество ребер в графах P_n и C_n характеризует длину таких графов.

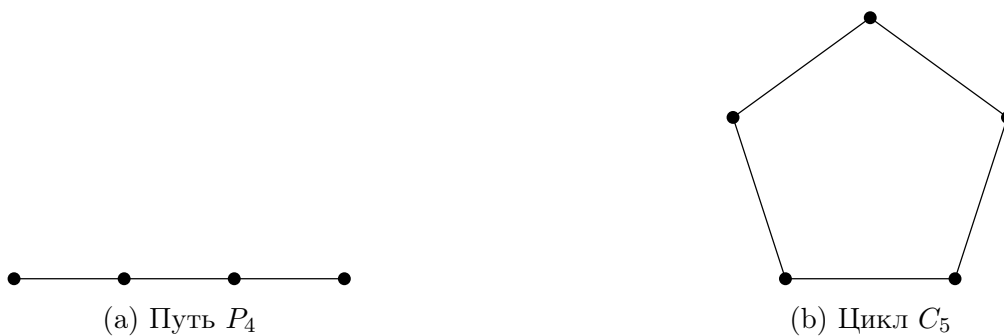


Рис. 6: Путь и цикл

Также часто нам будет встречаться граф W_n , называемый колесом (см.рис.7,а). Индекс n в обозначении W_n такого графа обозначает либо количество вершин в его внешнем цикле, либо количество всех вершин. Мы будем считать, что n — это количество вершин во внешнем цикле графа W_n .

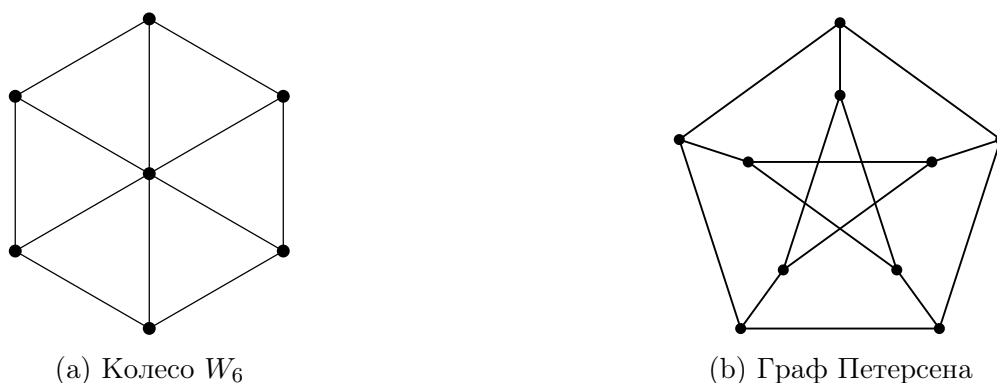


Рис. 7: Колесо и граф Петерсена

Важный класс простых подграфов представляют собой так называемые *регулярные графы*, то есть графы, у которых все вершины имеют одну и ту же степень k . Также их называют k -регулярными графами или регулярными графами степени k . В случае $k = 3$ такие графы называются кубическими. На рис.7,б показан пример кубического графа, который достаточно часто будет встречаться у нас в приложениях — так называемый граф Петерсена (Petersen graph). Графы C_n и K_n также являются регулярными графами степени 2 и $n - 1$ соответственно.

Наконец, имеется важный класс графов, являющихся одновременно k -регулярными и двудольными — так называемые k -кубы Q_k . Вершины графа Q_k можно пометить бинарными строками длины k (см.рис.5,с). Ребра же в таком графе проводятся только между теми вершинами, бинарные последовательности которых отличаются только в одной из k позиций. Пример графа Q_3 показан на рис.5,с.

1.3. Наряду с неориентированными, в теории графов также изучаются и так называемые ориентированные графы (или орграфы).

1.3.1. Начнем с мы вновь с достаточно формального и общего определения ориентированного мультиграфа D .

Определение 1.9. Если в тройке

$$D = (V, E, I),$$

в которой V есть множество вершин, а E — множество ребер, отображение I ставит в соответствие любому ребру e упорядоченную пару вершин $(x, y) \in V \times V$, то такая тройка называется *ориентированным графом* (или *орграфом*). В таком случае говорят, что ребро e *выходит* из вершины x и *входит* в вершину y . На рисунке такое ребро помечается стрелкой, указывающей направление данного ребра.

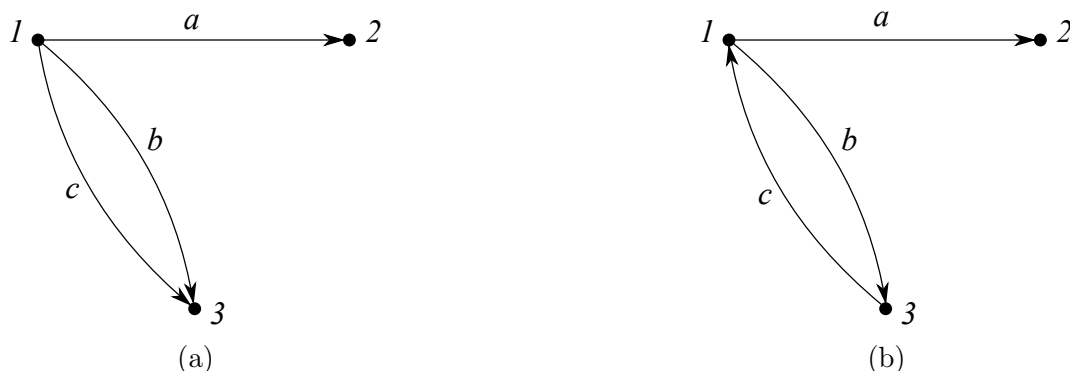


Рис. 8: Примеры ориентированных графов

В качестве примера на рис.8,а показан ориентированный граф D , построенный на множестве вершин $V = \{1, 2, 3\}$ и имеющий три ребра. Ребро a выходит из вершины 1 и входит в вершину 2 (то есть отвечает упорядоченной паре $(1, 2)$), а ребра b и c исходят из вершины 2 и приходят в вершину 3 (то есть им соответствует упорядоченная пара $(2, 3)$ вершин графа D). Таким образом, орграф D на рис. задается тройкой (V, E, I) , в которой множество V вершин и множество E ребер имеют вид

$$V = \{1, 2, 3\}, \quad E = \{a, b, c\},$$

а отображение $I: E \rightarrow V \times V$ задается таблицей вида

E	a	b	c
$V \times V$	$(1, 2)$	$(2, 3)$	$(2, 3)$

1.3.2. В орграфе различают исходящую ($\text{outdeg}(x)$) и входящую ($\text{indeg}(x)$) степень любой вершины $x \in V(D)$, а также просто степень, равную сумме входящей и исходящей степеней. Так, в графе, изображенном на рис.8,а, вершина 1 имеет входящую степень, равную единице, и исходящую степень, равную двойке.

Так как любое ориентированное ребро в орграфе D вносит вклад, равный единице, в сумму всех исходящих степеней вершин орграфа D , а также вклад, равный единице, в сумму всех входящих степеней вершин орграфа G , то для орграфа справедливо равенство вида

$$\sum_{x \in V(D)} \text{indeg}(x) = |E(D)| = \sum_{x \in V(D)} \text{outdeg}(x),$$

являющееся аналогом первой теоремы теории графов (1.4).

1.3.3. Как и для неориентированного графа, важным частным случаем орграфа является простой орграф.

Определение 1.10. Орграф D называется *простым*, если он не содержит петель, а также кратных *упорядоченных* ребер, то есть ребер, отвечающих одинаковым упорядоченным парам вершин.

Упорядоченность вершин в этом определении важна. Так, граф из рис.8,а простым не является — в нем ребра b и c отвечают одной и той же упорядоченной паре $(2, 3)$ вершин. Изображенный же на рис.8,б граф D' является простым. Несмотря на то, что в этом графе по-прежнему имеются два ребра, соединяющих вершины 2 и 3, направлены эти ребра в разные стороны. Иными словами, эти ребра отвечают различным упорядоченным парам вершин — ребро b отвечает упорядоченной паре $(2, 3)$, а ребро c — упорядоченной паре $(3, 2)$.

1.4. Следующим важным понятием в теории графов является понятие смежности вершин.

1.4.1. Начнем с определения смежных вершин в неориентированном графе G .

Определение 1.11. Говорят, что в неориентированном графе G вершина y смежна с вершиной x , если в этом графе существует ребро $\{x, y\}$.

На множестве вершин V смежность задает некоторое отношение. Для неориентированного графа G это отношение является симметричным: если вершина x смежна с вершиной y , то и вершина y смежна с вершиной x .

1.4.2. Для ориентированного графа ситуация несколько сложнее.

Определение 1.12. Говорят, что в ориентированном графе D вершина y смежна с вершиной x , если в этом графе существует ребро (x, y) , исходящее из вершины x и входящее в вершину y . (см.рис.8,а).

Для изображенного на рис.8,а орграфа G вершина 2 смежна с вершиной 1.

Сразу заметим, что если существует ребро (x, y) и не существует ребро (y, x) , то вершина y смежна с вершиной x , а вот x вершиной, смежной с y , уже не является. Так, для изображенного на рис.8,а орграфа вершина 1 смежной с вершиной 2 не является. Вершина x будет смежной с y только в случае, когда в орграфе D существует ребро (y, x) (см. вершины 1 и 2 на рис.8,б). Иными словами, в ориентированном графе D отношение смежности на множестве V вершин симметричным не является.

1.5. Для хранения графа в памяти компьютера, как правило, используются две структуры, тесно связанные с понятием смежности — матрица смежности и список смежности.

1.5.1. Матрица смежности — это матрица M_a размерами $n \times n$, любой элемент a_{ij} которой описывает количество ребер, идущих из вершины i в вершину j . Так, для примера 1.2 соответствующая графу G матрица смежности имеет следующий вид:

$$M_a = \begin{bmatrix} 0 & 1 & 2 & 0 \\ 1 & 1 & 0 & 1 \\ 2 & 0 & 0 & 3 \\ 0 & 1 & 3 & 0 \end{bmatrix}$$

Для ориентированного графа D , показанного на рис.8,а, матрица смежности равна

$$M_a = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix}$$

Заметим, что для неориентированного графа матрица смежности всегда симметрична. Как следствие, все собственные значения этой матрицы являются вещественными числами.

В случае простого графа или орграфа все диагональные элементы $a_{ii} = 0$, так что сумма собственных значений такой матрицы, совпадающая со следом матрицы M_a , также равняется нулю. Элементы, не лежащие на диагонали, равны единице в случае, если существует ребро, идущее из вершины i в вершину j , и нулю в случае, если такового ребра не существует.

1.5.2. Список смежности — это линейный массив L_a размера n , каждый элемент a_i которого содержит список (мультимножество) вершин, смежных с вершиной i . Для примера 1.2 соответствующий список имеет следующий вид:

1 : смежна с вершинами 2, 3, 3;
 2 : смежна с вершинами 1, 2, 4;
 3 : смежна с вершинами 1, 1, 4, 4, 4;
 4 : смежна с вершинами 2, 3, 3, 3.

Для ориентированного графа, показанного на рис.8,а, список смежности записывается так:

1 : список смежности пуст;
 2 : смежна с вершиной 1;
 3 : смежна с вершинами 2, 2.

2 Маршруты, пути, циклы в графе. Связные графы и орграфы

2.1. Следующая важная группа понятий теории графов связана с обходом графа вдоль некоторой последовательности его вершин и ребер. В этой связи нам понадобятся такие понятия, как маршруты, пути, циклы в графе, а также некоторые связанные с ними характеристики графа.

2.1.1. Начнем с определения маршрута в мультиграфе G .

Определение 2.1. *Маршрутом* (walk) в графе G из вершины x_0 в вершину x_k называется чередующаяся последовательность

$$W := x_0, e_1, x_1, e_2, x_2, \dots, x_{k-1}, e_k, x_k$$

вершин $x_i \in V$ и ребер $e_i \in E$, соединяющих вершины x_{i-1} и x_i . И вершины, и ребра в такой последовательности могут повторяться. Количество ребер в маршруте W называется *длиной* k этого маршрута.

В случае простого графа любой маршрут W полностью определяется последовательностью

$$x_0, x_1, x_2, \dots, x_{k-1}, x_k,$$

вершин $x_i \in V(G)$, любые два последовательных элемента x_{i-1}, x_i которой являются смежными вершинами (т.е. соединены между собой ребром $e_i = \{x_{i-1}, x_i\} \in E(G)$).

Вершины x_0 и x_k называют часто начальной и конечной вершинами маршрута W , а остальные вершины — внутренними его вершинами. Говорят также, что вершины x_0 и x_k связаны маршрутом W , а сам маршрут называют x_0x_k -маршрутом.

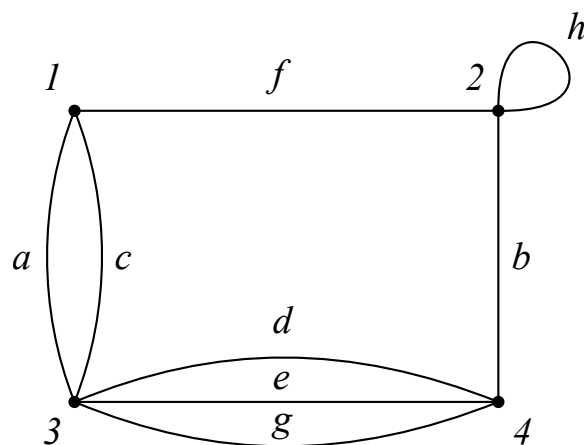


Рис. 9

Определение 2.2. Если все ребра $e_1, \dots, e_k$ в маршруте различны, то такой маршрут называется *путем* (в английской литературе — trail) из вершины x_0 в вершину x_k . Если также и все вершины в данном пути различны, то такой путь называется *простым* (в английской литературе — path).

В качестве примера рассмотрим разные маршруты в графе, показанном на рис.9. Маршрут

$$1, a, 3, d, 4, e, 3, d, 4, b, 2, h, 2$$

путем не является — в нем повторяется ребро d . Маршрут

$$1, a, 3, d, 4, e, 3, g, 4, b, 2, h, 2$$

представляет собой путь (trail), который не является простым — в нем повторяются вершины 3, 4, 2. Наконец, маршрут

$$1, a, 3, e, 4, b, 2$$

является простым путем (path) из вершины 1 в вершину 2.

2.1.2. Следующая серия определений связана с замкнутыми маршрутами в графах, то есть маршрутами, в которых начальная x_0 и конечная x_k вершины совпадают.

Определение 2.3. Замкнутым путем (closed trail) или составным циклом (circuit) в графе G называется путь, в котором $x_0 = x_k$. Замкнутый простой путь называется простым циклом (cycle). Часто слово “простой” опускают, понимая под циклом замкнутый простой путь.

Так, в графе G , показанном на рис.9, путь $1, a, 3, e, 4, d, 3, c, 1$ является составным циклом, а путь вида $1, a, 3, e, 4, b, 2, f, 1$ — простым циклом. Замкнутый маршрут вида $1, a, 3, e, 4, e, 3, c, 1$ составным циклом не является — в нем повторяется ребро e .

В простом графе невозможны циклы длины меньшей, чем три. Циклы длины три часто называются треугольниками. Графы, в которых такие циклы отсутствуют, называются графами, свободными от треугольников (triangle-free graphs).

Определение 2.4. Обхватом графа (girth) называется длина наименьшего цикла в нем. Если в графе циклы отсутствуют, то обхват такого графа считают равным бесконечности.

С учетом данного определения простые графы, свободные от треугольников — это графы, обхват которых больше трех.

В случае мультиграфов возможны и циклы длины 2, и циклы длины 1. Циклы длины 2 — это циклы типа $3, d, 4, e, 3$ на рис.9, появляющиеся при обходе двух ребер одного и того же мультиребра. Циклы длины 1 — это петли графа G . Так, петлю h на рис.9 можно рассматривать как цикл $2, h, 2$ длины 1.

Изолированная вершина по определению является замкнутым путем длины 0, однако циклом она не является.

2.1.3. Теперь мы готовы определить понятие связных графов.

Определение 2.5. Если вершины $x, y \in V$ графа G соединены хотя бы одним путем, то такие вершины называются *связанными*.

Несложно проверить, что связность задает на множестве V вершин графа G отношение эквивалентности. Это отношение делит граф на классы эквивалентности, называемые *компонентами связности графа*.

Определение 2.6. В случае, когда в графе G существует лишь одна компонента связности, то есть в случае, когда любые две вершины x, y графа соединены хотя бы одним путем, граф называется *связным*. В противном случае граф называется *несвязным*.

2.1.4. С понятием пути в графе тесно связаны и такие важные понятия, как расстояние между вершинами в графе, эксцентриситет вершины, диаметр и радиус графа.

Определение 2.7. Расстоянием $d(x, y)$ между двумя связанными вершинами $x, y \in V(G)$ называется длина наименьшего пути между ними.

Понятно, что такой путь обязательно является простым. В случае, когда вершины не являются связанными, полагают по определению, что $d(x, y) = \infty$.

Определение 2.8. Диаметр графа называется максимальное расстояние между его вершинами:

$$\text{diam}(G) := \max_{x, y \in V(G)} d(x, y).$$

В случае несвязного графа считается, что $\text{diam}(G) = \infty$.

Определение 2.9. Эксцентриситетом $\varepsilon(x)$ вершины $x \in V(G)$ называется максимальное расстояние от x до любой другой вершины графа G :

$$\varepsilon(x) := \max_{y \in V(G)} d(x, y).$$

Определение 2.10. Радиусом $r(G)$ графа G называется минимальный из эксцентриситетов вершин графа G . Вершины, на которых этот минимум достигается, называются *центральными вершинами* графа G . Множество всех центральных вершин называется *центром* графа.

2.2. Рассмотрим теперь понятие связности в ориентированном графе D .

2.2.1. Начнем с определения связанных вершин в орграфе.

Определение 2.11. Вершины x и y орграфа D называются *связанными*, если в D существуют хотя бы один путь из x в y и хотя бы один путь из y в x .

Определение 2.12. Орграф D называется *сильно связным*, если любые две его вершины являются связанными.

Иногда наряду с этим понятием для орграфа вводят понятие слабой связности.

Определение 2.13. Орграф D называется *слабо связным*, если соответствующий ему неориентированный граф G , получающийся заменой всех ориентированных ребер на неориентированные, является связным.

2.2.2. Как и в случае неориентированного графа, в орграфе D отношение связности является отношением эквивалентности. Как следствие, множество всех вершин $V(D)$ орграфа D разбивается с помощью этого отношения эквивалентности на классы попарно связанных вершин, которые называются *компонентами сильной связности*.

Сразу заметим, что в неориентированном графе ребер между различными компонентами связности не существует. В орграфе такие ребра могут существовать, однако направлены все они будут лишь от одной компоненты связности к другой. Именно, справедливо следующее достаточно очевидное утверждение.

Лемма 2.14. Пусть H_1, H_2 есть две различные компоненты сильной связности графа D , и пусть существует ребро $e \in E(D)$ из H_1 в H_2 . Тогда ребра из H_2 в H_1 отсутствуют.

Доказательство. Действительно, если ребро из H_2 в H_1 существует, то любые две вершины в множестве $H_1 \cup H_2$ вершин оказываются связанными. Иными словами, $H_1 \cup H_2$ представляет собой компоненту сильной связности графа D , что противоречит предположению о том, что H_1 и H_2 есть две различные компоненты сильной связности. $\square$

Как следствие, по любому орграфу D можно построить так называемый граф $C(D)$ компонент сильной связности графа D , вершинами которого будут компоненты сильной связности графа D , а ребрами — ребра графа D , направленные из одной компоненты сильной связности D к другой. Основное свойство такого орграфа $C(D)$ состоит в том, что в таком графе нет циклов.

Теорема 2.15. В орграфе $C(D)$ циклы отсутствуют, то есть он, как еще говорят, представляет собой ациклический орграф (DAG — directed acyclic graph).

Доказательство. Если бы в таком графе существовал цикл вида $H_1 \rightarrow H_2 \rightarrow \dots \rightarrow H_n \rightarrow H_1$, то любые две вершины в объединении $H_1 \cup H_2 \cup \dots \cup H_n$ оказались бы связанными. Действительно, внутри каждой компоненты H_i мы, по определению сильной связности, можем попасть из любой вершины в любую вершину H_i . Вершины же из разных компонент H_i и H_j мы также можем всегда связать с помощью пути, идущего из H_i в H_j , а также пути, соединяющего компоненты H_j и H_i . $\square$

3 Подграф графа G . Основные операции над графами

3.1. Вернемся к основным определениям теории графов и введем очень важное понятие подграфа графа G . Заодно мы введем две основные операции над графами — операцию удаления ребра и операцию удаления вершины в графе G .

3.1.1. Начнем с формального определения подграфа H графа G .

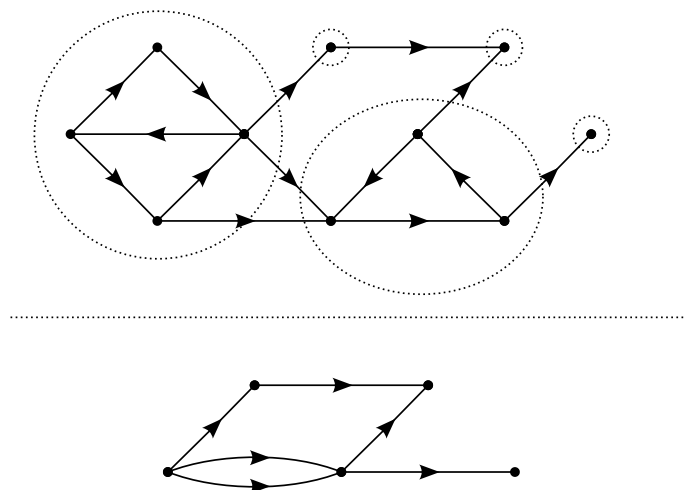
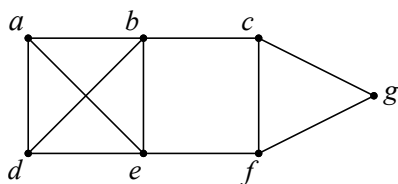


Рис. 10: Оргаф D и граф компонент сильной связности графа D

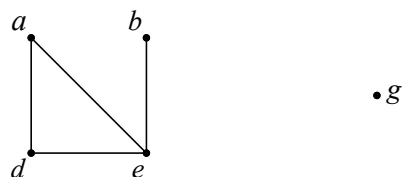
Определение 3.1. Подграфом графа G называется граф H , для которого выполнены следующие три условия:

1. $V(H) \subseteq V(G)$;
2. $E(H) \subseteq E(G)$;
3. любое ребро $e \in E(H)$, соединяющее пару вершин x и y в H , должно соединять ту же самую пару вершин в графе G .

По отношению к графу H граф G иногда называют *надграфом* или *суперграфом*.



(a) Граф G



(b) Подграф H графа G

Рис. 11

В качестве примера рассмотрим графы G и H , изображенные на рис.11,a,b. Утверждается, что граф H является подграфом графа G . Действительно, множество вершин графа H есть подмножество вершин графа G , а множество ребер $E(H)$ — подмножество множества $E(G)$. Рассмотрим теперь любое ребро графа H , например, ребро $\{a, e\}$ графа H . Это же ребро соединяет те же самые вершины a и e и в графе G . Так как это верно для всех ребер $e \in E(H)$, то граф H действительно является подграфом графа G .

3.1.2. Данное выше определение можно сделать несколько более конструктивным, если ввести две основные операции над графами — операцию удаления ребра и операцию удаления вершины. Оказывается, любой подграф H графа G — это граф, полученный из исходного графа G с помощью этих двух операций.

Начнем с более простой операции удаления ребра e . При такой операции множество вершин графа G не меняется, а из множества ребер удаляется элемент $e \in E(G)$. Полученный в результате этой операции граф обозначается $G - e$. Очевидно, что он является подграфом графа G . Так, в приведенном на рис.11 примере мы в графе G удалили ребро, соединяющее вершины a и b .

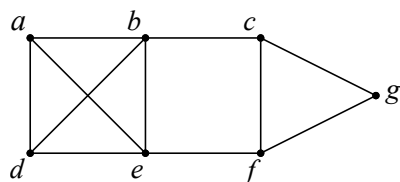
В более общем случае мы таким образом можем удалить сразу несколько ребер, принадлежащих некоторому подмножеству S множества $E(G)$ ребер графа G . Полученный в результате этих операций подграф обозначается $G - S$.

Перейдем теперь к чуть более сложной операции удаления вершины. Предположим, что мы хотим удалить в графе G вершину x . Если эта вершина является изолированной, то нам ничего больше делать не нужно. Если же этой вершине инцидентны какие-то ребра, то мы обязаны будем также вместе с вершиной x удалить и их. Действительно, ранее эти ребра вели в вершину x . Вести в никуда ребра не могут, так что нам вместе с вершиной x приходится удалять и все ребра, инцидентные данной вершине. Так, в примере мы удалили в графе G вершину c . Вместе с ней мы вынуждены были удалить и все три инцидентных ей ребра.

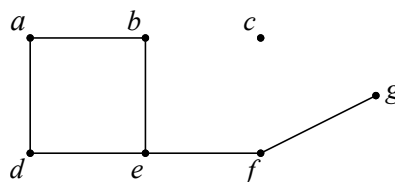
Полученный в результате удаления вершины x граф обозначается обычно $G - x$. Очевидно, что граф $G - x$ также является подграфом исходного графа G .

В более общем случае $S \in V(G)$ мы с помощью операции удаления вершин из подмножества S получаем подграф $G - S$, в котором по сравнению с исходным графом G удалены все вершины подмножества S вместе со всеми ребрами, инцидентными этим вершинам.

3.1.3. Как мы уже заметили, любой подграф графа G получается из G последовательным выполнением двух операций — удаления вершин и удаления ребер. Естественным кажется рассмотреть два частных случая этой ситуации. Первый — это случай, когда мы в графе G удаляем только ребра, а второй — когда в G мы удаляем только вершины.



(a) Граф G



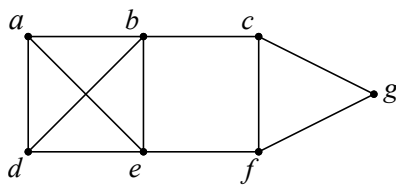
(b) Остовный подграф H графа G

Рис. 12

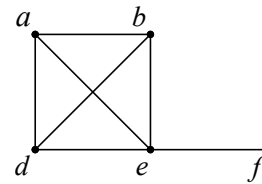
Если мы при получении из графа G подграфа H используем лишь операцию удаления ребер, то мы получаем подграф, множество вершин которого совпадает с множеством $V(G)$ вершин исходного графа. Такой подграф называется *остовным* подграфом (spanning subgraph) графа G (смотри рис.12). В частности, любой простой граф G , построенный на n вершинах, является остовным подграфом полного графа K_n .

Второй частный случай — когда мы в графе G удаляем одну или несколько вершин. В результате такой операции мы получаем подграф H , *индуцированным подмножеством оставшихся вершин графа G* . Иными словами, подграфом H графа G , индуцированным подмножеством вершин S , называется граф, полученный из G удалением всех вершин, не принадлежащих множеству S , вместе со всеми инцидентными этим вершинам ребрами (смотри рис.13).

3.1.4. Достаточно часто в приложениях встречаются важные с практической точки зрения под-



(a) Граф G



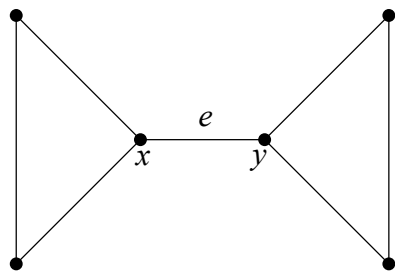
(b) Подграф H графа G , индуцированный подмножеством вершин a, b, d, e и f

Рис. 13

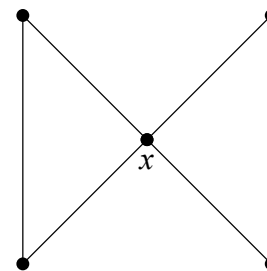
классы остовных подграфов. По историческим причинам некоторые из них имеют специальные названия. Так, остовный 1-регулярный подграф носит название *совершенного паросочетания* в графе G , а остовный k -регулярный подграф для произвольного натурального $k \geq 1$ называется *k -фактором* графа G . Остовный путь в графе носит название гамильтонова пути в графе G , а остовный цикл — гамильтонова цикла в графе G .

В случае мультиграфа G очень часто рассматривают простой граф, получающийся из исходного графа G удалением всех петель и заменой мультиребер на простые ребра. Такой граф, очевидно, является остовным подграфом исходного графа.

3.2. Вернемся к понятию связности в графах. Операции удаления ребра и вершины позволяют нам ввести важные понятия моста и точки сочленения в графе.



(a) Граф G_1



(b) Граф G_2

Рис. 14

3.2.1. Рассмотрим графы, показанные на рис.14. Видно, что связный граф G_1 (рис.14,a) перестает быть таковым после удаления ребра e , связывающего вершины x и y .

Определение 3.2. Ребро $e \in E(G)$ в связном графе G называется *мостом*, если получающийся после его удаления граф $G - e$ становится несвязным. В случае несвязного графа G мостом называется ребро, после удаления которого количество компонент связности увеличивается на единицу.

Теперь рассмотрим связный граф G , показанный на рис.14,b. У этого графа мостов нет, однако имеется вершина x , после удаления которой граф $G - x$ становится несвязным.

Определение 3.3. Вершина x называется *точкой сочленения* графа G , если после ее удаления количество компонент связности графа $G - x$ увеличивается по сравнению с количеством компонент связности исходного графа G .

Заметим, что даже в случае связного графа G удаление в нем точки сочленения может приводить к появлению как двух, так и большего количества связных компонент.

3.2.2. Докажем простое утверждение, характеризующее точку сочленения в связном графе G .

Утверждение 3.4. Вершина x в связном графе G , построенном на $n \geq 3$ вершинах, есть точка сочленения графа G тогда и только тогда, когда в G существуют отличные от x вершины y и z , такие, что x содержится в каждом пути из y в z .

Доказательство. Предположим вначале, что x есть точка сочленения в связном графе G . После удаления x в графе $G - x$ найдутся по крайней мере две различные связные компоненты G_1 и G_2 . Выбрав произвольную вершину y в G_1 и произвольную вершину z в G_2 , мы получим искомую пару вершин. Действительно, если бы в G нашелся путь, не проходящий через вершину x , то в графе $G - x$ вершины y и z оказались бы связанными, что невозможно.

Теперь предположим, что в G существуют описанная в утверждении пара вершин y и z . Удаление вершины x в таком графе уничтожает любой путь из y в z , то есть превращает эти вершины в несвязанные. Это, в свою очередь, означает, что вершины y и z , лежащие ранее в одной компоненте связности графа G , стали принадлежать разным компонентам связности графа $G - x$. Иными словами, удаление x увеличивает количество связных компонент графа, то есть x есть точка сочленения в G . $\square$

3.2.3. Теперь докажем наиболее важное утверждение, характеризующее мосты в графе G .

Утверждение 3.5. Ребро $e = \{x, y\}$ в простом связном графе G является мостом тогда и только тогда, когда e не принадлежит ни одному из циклов графа G .

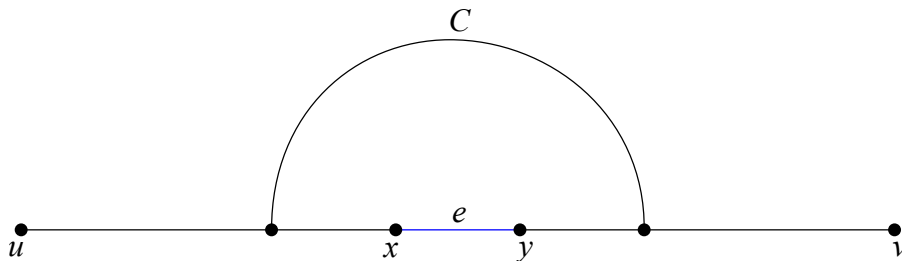


Рис. 15

Доказательство. Пусть ребро e является мостом в графе G . Удаление e приводит к появлению вместо G двух связных компонент G_1 и G_2 , таких, что $x \in G_1$ и $y \in G_2$. Выберем произвольную пару вершин $u \in V(G_1)$ и $v \in V(G_2)$ в этих компонентах (см. рис. 15). Предположим, что ребро e принадлежит какому-то циклу C в графе G . Как и любые вершины в связном графе, вершины u и x в G_1 связаны между собой некоторым ux -путем. Аналогично, вершины y и v связаны в G_2 yv -путем. После удаления ребра e вершины x и y остаются связанными между собой некоторым xy -путем, проходящим через оставшиеся ребра цикла C . В силу транзитивности отношения связности отсюда следует, что произвольные вершины $u \in G_1$ и $v \in G_2$ соединены между собой некоторым путем, что противоречит определению моста в G . Следовательно, e ни какому циклу принадлежать не может.

Обратно, предположим, что e не является мостом в графе G , то есть его удаление оставляет граф $G - e$ связным. Но тогда концы x и y ребра e в графе $G - e$ оказываются связанными некоторым xy -путем. Добавление к этому пути ребра e превращает этот путь в цикл. $\square$

Замечание 3.6. Доказанные выше утверждения являются, по сути, аналогами вершинной и реберной теорем Менгера, описывающих k -связные графы. Изучению таких графов будет посвящена отдельная глава.

3.3. Мы ввели операции удаления ребра и вершины. Нам понадобятся еще несколько операций над графами.

3.3.1. Начнем с операций объединения и пересечения графов. Пусть G и H есть пара простых графов.

Определение 3.7. Объединением графов G и H называется граф $F = G \cup H$, множество вершин и множество ребер которого представляют собой объединение соответствующих множеств графов G и H :

$$V(F) = V(G) \cup V(H), \quad E(F) = E(G) \cup E(H).$$

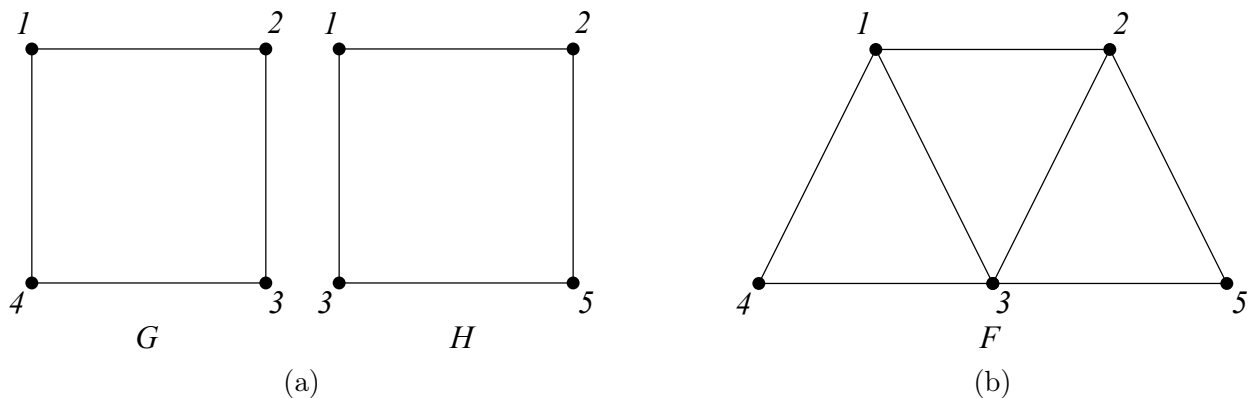


Рис. 16: Операция объединения графов G и H

На рис.16,b в качестве примера показан граф F , полученный в результате объединения графов G и H , показанных на рис.16,a.

В случае, если множества вершин (а следовательно, и ребер) графов G и H не пересекаются, то мы получаем так называемое несвязное объединение $G + H$ таких графов. По сути, любой несвязный граф представляет собой несвязное объединение двух или более связных компонент.

Определение 3.8. Пересечением графов G и H называется граф $F = G \cap H$, множество вершин и множество ребер которого представляют собой пересечение соответствующих множеств графов G и H :

$$V(F) = V(G) \cap V(H), \quad E(F) = E(G) \cap E(H).$$

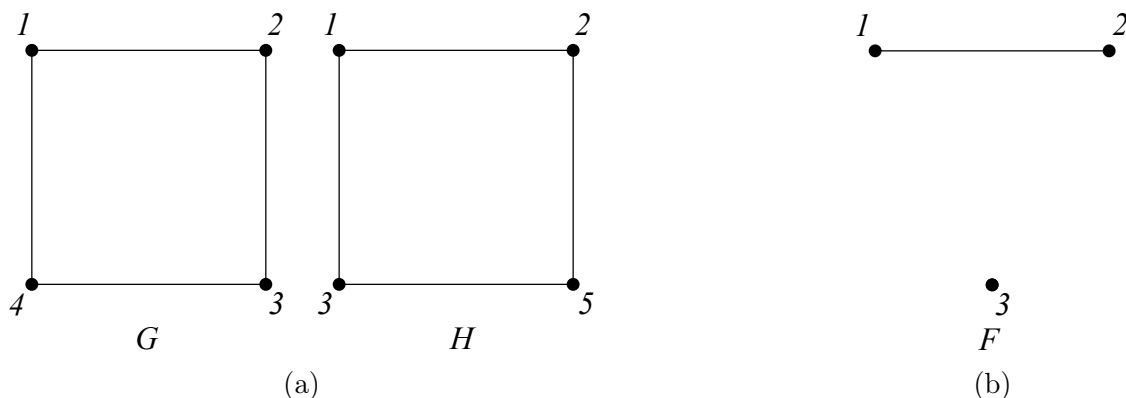


Рис. 17: Операция объединения графов G и H

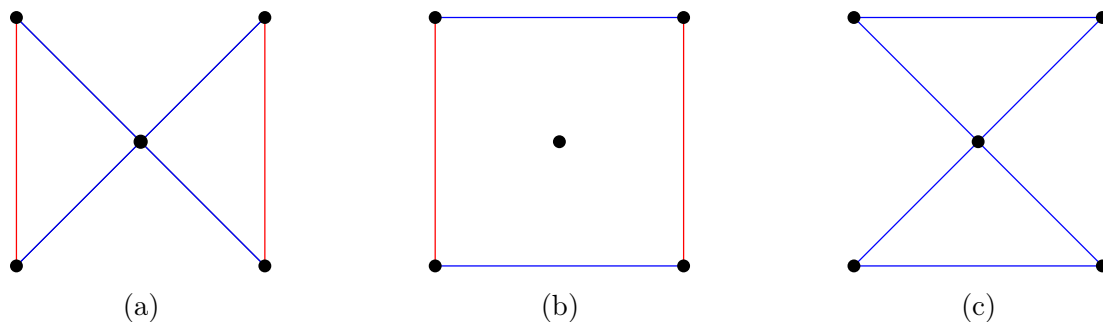


Рис. 18: Операция симметрической разности двух остовных подграфов

На рис.17,b приведен граф F , полученный в результате пересечения графов G и H , показанных на рис.17,a.

3.3.2. При изучении остовных подграфов H_1 и H_2 одного и того же графа G достаточно часто встречается операция Δ их симметрической разности. С помощью такой операции мы из остовных подграфов H_1 и H_2 получаем остовный подграф $H = H_1 \Delta H_2$, состоящий из ребер, принадлежащих H_1 , но не принадлежащих H_2 , и наоборот. На рис.18,c для примера показан остовный подграф H графа G , являющийся симметрической разностью подграфов H_1 и H_2 полного графа K_4 , нарисованных на рис.18,a и рис.18,b. Общие для H_1 и H_2 ребра помечены на этих рисунках красным цветом, а ребра, встречающиеся в одном подграфе и не встречающиеся в другом — синим цветом. Результирующий подграф H_3 состоит только из ребер, окрашенных синим цветом.

3.3.3. Помимо удаления ребра, в теории графов достаточно часто встречается и еще одна важная операция — операция стягивания ребра.



Рис. 19: Операция стягивания ребра

Определение 3.9. Говорят, что граф $G \setminus e$ получен из графа G стягиванием ребра e , если граф $G \setminus e$ получается из G удалением ребра e и стягиванием инцидентных e вершин x и y в одну (см.рис.19).

4 Изоморфизм и автоморфизм графов

4.1. Постараемся вначале ответить на вопрос, сколько же существует различных графов, построенных на n вершинах.

4.1.1. Сразу заметим, что в случае мультиграфов мы наряду с количеством вершин должны также фиксировать и количество ребер. Действительно, любые две вершины мультиграфа соединить произвольным количеством ребер, поэтому количество мультиграфов, построенных на n вершинах, без ограничения на количество ребер может быть сколь угодно большим. Поэтому мы ограничимся пока что задачей подсчета простых графов.

4.1.2. Количество g_n простых неориентированных графов достаточно легко сосчитать, используя формальное определение простого графа как некоторого подмножества множества $V^{(2)}$. Действительно, количество всех двухэлементных подмножеств n -элементного множества вершин (то есть мощность множества $V^{(2)}$) равно $|V^{(2)}| = \binom{n}{2}$. Нас же интересует множество Σ всех подмножеств $V^{(2)}$. Количество элементов в этом множестве, как известно, равно

$$|\Sigma| = 2^{|V^{(2)}|} = 2^{\binom{n}{2}}.$$

Следовательно, количество g_n всех простых графов на n вершинах равняется $2^{\binom{n}{2}}$.

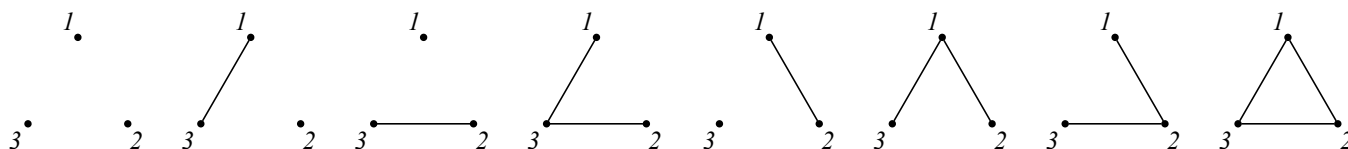


Рис. 20: Восемь простых графов на трех вершинах

Например, существует ровно $g_3 = 2^{\binom{3}{2}} = 2^3 = 8$ различных простых графов, построенных на трех вершинах (смотри рис.20).

4.1.3. Столь же просто подсчитать количество различных простых орграфов. Действительно, всего существует $n(n-1)$ упорядоченных пар отличных друг от друга вершин. Как следствие, количество различных простых орграфов, построенных на n вершинах, равно $2^{n(n-1)}$.

4.2. Заметим, что многие из приведенных на рис.20 графов похожи друг на друга в том смысле, что отличаются они друг от друга только перенумерацией вершин. Таковыми, в частности, являются три графа, имеющих только лишь одно ребро, а также три графа, имеющих ровно два ребра. Формализовать эту похожесть можно с помощью понятия изоморфизма графов.

4.2.1. Определим вначале понятие изоморфизма для простых неориентированных графов.

Определение 4.1. Говорят, что два простых графа G_1 и G_2 *изоморфны* друг другу, если существует взаимно-однозначное отображение $\varphi: V(G_1) \rightarrow V(G_2)$, которое сохраняет отношение смежности. Последнее означает, что если в графе G_1 некоторая пара вершин $\{x, y\}$ соединена ребром, то в графе G_2 соответствующая ей пара вершин $\{\varphi(x), \varphi(y)\}$ также соединена ребром, и наоборот.

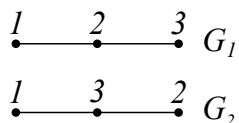


Рис. 21: Два изоморфных графа

Пример 4.2. Рассмотрим графы G_1 и G_2 , изображенные на рис.21. Они изоморфны друг другу, так как существует взаимно-однозначное отображение $\varphi: \{1, 2, 3\} \rightarrow \{1, 2, 3\}$ вида

$$\varphi(1) = 1, \quad \varphi(2) = 3, \quad \varphi(3) = 2,$$

при котором ребро $\{1, 2\}$ графа G переходит в ребро $\{\varphi(1), \varphi(2)\} = \{1, 3\} \in E(G_2)$, а ребро $\{2, 3\}$ графа G переходит в ребро $\{\varphi(2), \varphi(3)\} = \{3, 2\}$ графа G_2 .

4.2.2. Как и любой изоморфизм, изоморфизм графов вводит отношение эквивалентности на множестве всех неориентированных простых графов на n вершинах. Отношение эквивалентности, в свою очередь, разбивает все множество таких графов на классы эквивалентности — классы изоморфных друг другу графов. В теории графов эти классы эквивалентности имеют специальное название.

Определение 4.3. Любой описанный выше класс эквивалентности графов называется *непомеченным* графом. В этом смысле обычный простой граф (то есть любой представитель класса эквивалентности) часто называется *помеченным* графом.

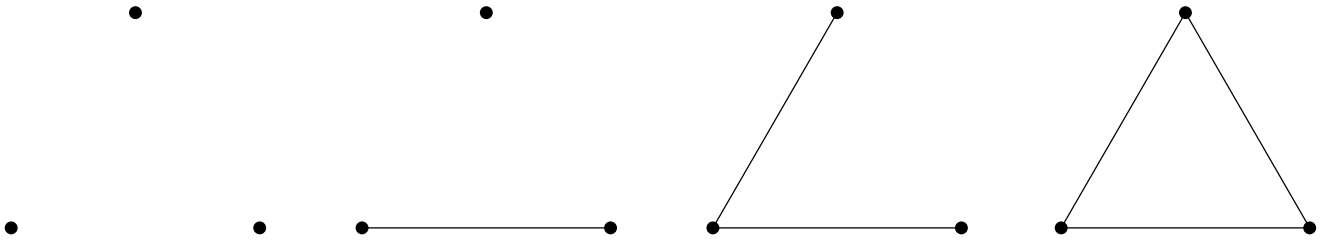


Рис. 22: Четыре простых непомеченных графа на трех вершинах

Так, из рис.20 видно, что существует четыре различных непомеченных графа на трех вершинах: один отвечает пустому графу, второй — трем помеченным графам с одним ребром, третий — трем помеченным графам с двумя ребрами, и наконец четвертый — полному графу K_3 (см.рис.22). Задача перечисления непомеченных графов на n вершинах для произвольного значения параметра n является значительно более сложной задачей по сравнению с задачей перечисления соответствующих помеченных графов.

4.2.3. Давайте теперь для полноты картины дадим определение изоморфизма для более общего случая неориентированных мультиграфов. Как мы помним, в общем случае любой неориентированный граф G задается тройкой, состоящей из множества $V(G)$ вершин, множества $E(G)$ ребер, а также отображения $I(G)$, сопоставляющего любому ребру $e \in E(G)$ неупорядоченную пару вершин $\{x, y\}$, которые это ребро e соединяет.

Определение 4.4. Говорят, что графы

$$G_1 = (V(G_1), E(G_1), I(G_1)) \quad \text{и} \quad G_2 = (V(G_2), E(G_2), I(G_2))$$

изоморфны друг другу, если существует пара взаимно-однозначных отображений

$$(\varphi, \vartheta), \quad \varphi: V(G_1) \rightarrow V(G_2), \quad \text{и} \quad \vartheta: E(G_1) \rightarrow E(G_2),$$

сохраняющих отношение смежности в графах G_1 и G_2 . Последнее в данном случае означает, что если отображение $I(G_1)$ сопоставляет в графе G_1 ребру $e \in E(G_1)$ пару $\{x, y\}$, то в графе G_2 отображение $I(G_2)$ сопоставляет ребру $\vartheta(e) \in E(G_2)$ пару вершин $\{\varphi(x), \varphi(y)\}$, и наоборот.

4.3. Часто на практике достаточно важно уметь отвечать на вопрос, являются ли два различных графа изоморфными друг другу или нет. Оказывается, проверка двух графов на изоморфизм является в общем случае довольно-таки нетривиальной задачей.

4.3.1. Начнем вначале с простого примера.

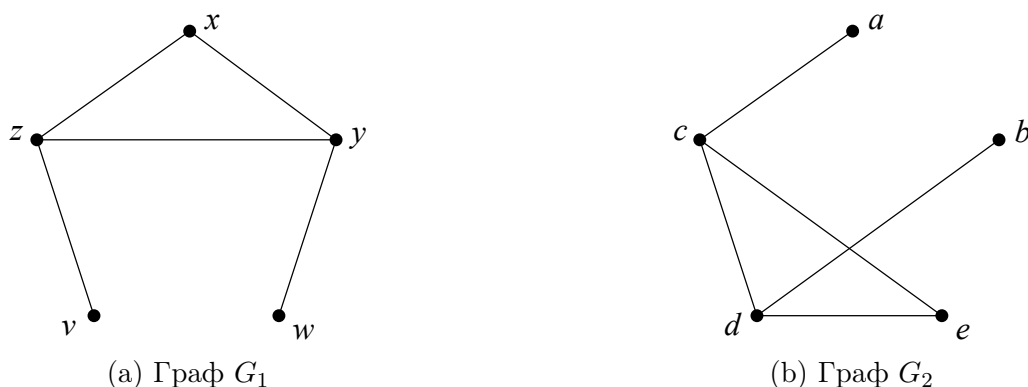


Рис. 23: Установление изоморфизма между графами

Пример 4.5. Рассмотрим два графа, изображенных на рис.23. Возникает вопрос, являются ли два таких графа изоморфными друг другу.

Для ответа на этот вопрос нам нужно, вообще говоря, найти взаимно-однозначное отображение φ из множества вершин $V(G_1)$ в множество $V(G_2)$, сохраняющее отношение смежности. Легче всего в таких простых задачах действовать следующим образом: нужно удалить пометки вершин у первого и у второго графов, пометить вершины первого графа числами от 1 до n , а затем попытаться разметить вершины второго графа теми же числами от 1 до n так, чтобы такой изоморфизм φ был бы очевиден.

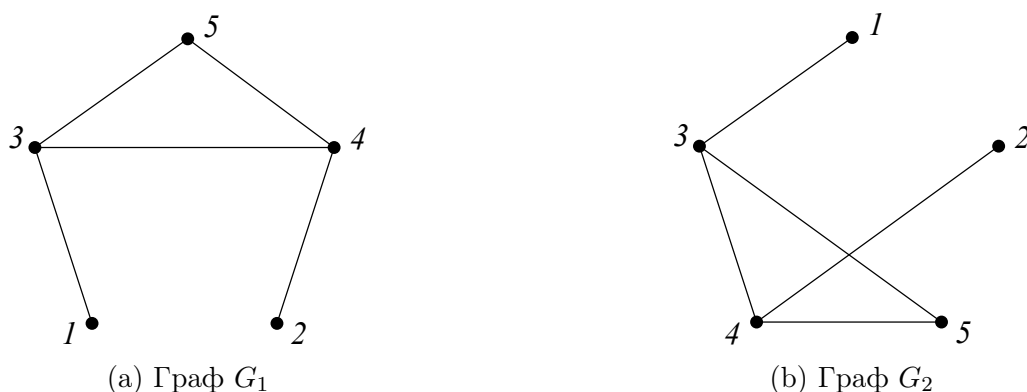


Рис. 24: Установление изоморфизма между графами

В данном конкретном примере сделать это довольно просто (см.рис.24): у графов G_1 и G_2 имеются всего лишь две вершины, степень которых равна единице. В графе G_1 они помечены цифрами 3 и 4. Поэтому нам будет удобно теми же самыми цифрами пометить одновалентные вершины и в графе G_2 . Далее, в графе G_1 эти вершины соединены с вершинами 1 и 2, имеющими степень, равную трем. Пометим аналогичные вершины в G_2 теми же цифрами. Нам остается пометить в графе G_2 оставшуюся вершину цифрой 5 и убедиться, что отображение $\varphi(i) = i$ задает изоморфизм этих графов.

Пример 4.6. Рассмотрим теперь графы, показанные на рис.25,а. Для таких графов уже не вполне очевидно то, что они изображают один и тот же непомеченный граф. Для доказательства этого факта можно, например, попытаться разметить вершины двух графов так, чтобы пара вершин $\{x, y\}$, связанная ребром в графе G_1 , оказалась бы связанной ребром и в графе G_2 , и наоборот. В данном примере это можно все же сделать достаточно просто, заметив, что граф G_2 состоит из двух непересекающихся циклов длины 4. Соответственно, выделив в графе G_1 замкнутый цикл аналогичной длины, мы можем уже однозначно пометить оставшиеся вершины графа G_1 так, чтобы списки смежности этих графов совпали (смотри рис.25,б).

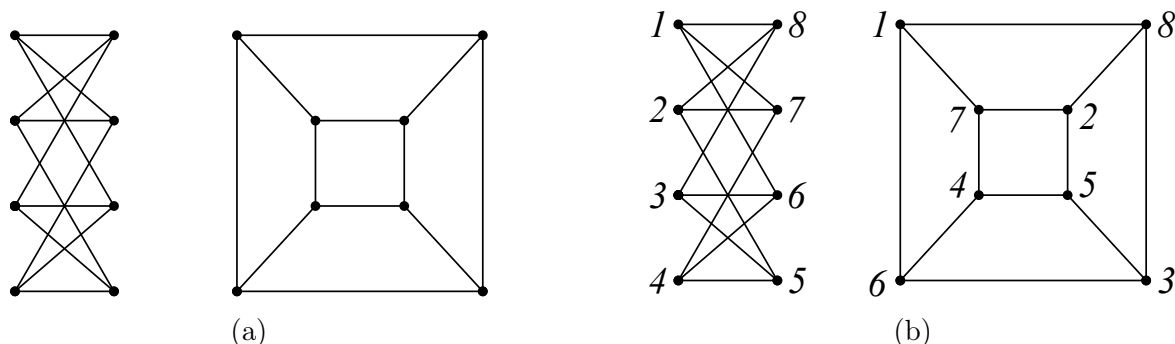


Рис. 25: Установление изоморфизма между графами

4.3.2. В практических задачах довольно часто встречаются графы, в которых содержатся сотни тысяч вершин. Для таких графов никакие визуальные методы оценки уже не работают. Не существует пока и никаких достаточно простых критериев, позволяющих достаточно быстро ответить на вопрос, являются ли два заданных графа изоморфными или нет. Конечно же, есть очевидные необходимые условия изоморфности двух графов — так, например, у изоморфных графов последовательности степеней их вершин должны, очевидно, совпадать. Но в случае, если подобного рода очевидные условия выполнены, то не остается ничего другого, как проверять заданную пару графов на изоморфность простым перебором.

Для этого вершины обоих графов помечают числами из одного и того же множества $[n]$, затем разметку первого графа фиксируют, а пометки второго графа начинают переставлять, вообще говоря, всеми $n!$ способами. Если на каком-то шаге матрицы смежности или же списки смежности этих двух графов совпадут, то графы являются изоморфными друг другу. В противном случае они неизоморфны.

Такого рода алгоритмы крайне неэффективны хотя бы потому, что при больших n значение $n!$ крайне велико. Существуют достаточно сложные алгоритмы, позволяющие более эффективно проверять на изоморфизм специальные подклассы графов, однако никакого быстрого алгоритма, позволяющего делать это в общем случае, пока не существует.

4.4. Наряду с изоморфизмом, в теории графов вводится также понятие автоморфизма — изоморфизма графа G в себя.

4.4.1. Прежде чем давать формальное определение автоморфизма, вернемся еще раз к очень важным понятиям непомеченных и помеченных графов. Давайте подумаем, сколькими способами мы можем разметить один и тот же непомеченный граф на n вершинах. Очевидно, что мы это можем сделать $n!$ способами. Однако при этом возникает вопрос — действительно ли все такие способы пометки дадут нам *различные* помеченные графы?

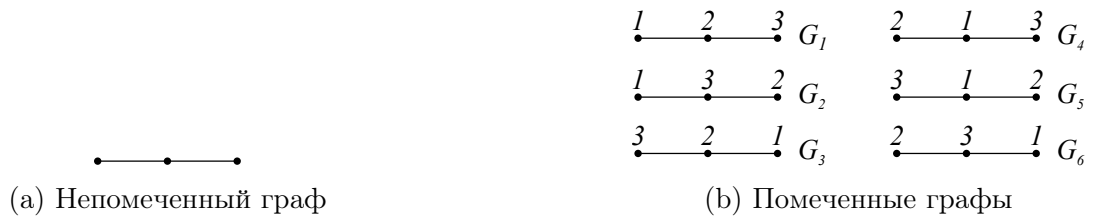


Рис. 26

В качестве примера рассмотрим простой немеченный граф на трех вершинах, имеющий два ребра (рис.26,а). Пометим для него $3!$ способами его вершины. В результате мы получим 6 помеченных графов, изображенных на рис.26,б. Однако мы с вами уже перебрали все помеченные графы, построенные на трех вершинах, и выяснили, что существует только лишь три *различных* помеченных графа, имеющих два ребра. Следовательно, среди шести графов, показанных на рис.26,б, половина лишняя в том смысле, что среди этих графов имеются графы, представляющие собой один и тот же помеченный граф.

Несложно, в частности, убедиться в том, что графы G_1 и G_3 , показанные на рис.26,б, представляют собой один и тот же помеченный граф — первый из них переходит во второй при его развороте на 180° . То же касается и графов G_2 и G_6 , а также графов G_4 и G_5 .

4.4.2. Давайте теперь постараемся разобраться с причиной того, что при разметке вершин немеченного графа $n!$ способами мы получаем лишние графы. Для этого возьмем немеченный граф $\tilde{G}$, как-то разметим его вершины (например, так, как это показано на рис.27,а), а затем рассмотрим поподробнее все функции $\varphi: [3] \rightarrow [3]$, задающие перестановки вершин полученного таким образом помеченного графа G_1 .



Рис. 27: Изоморфные помеченные графы на трех вершинах

Рассмотрим вначале функцию φ_1 , оставляющую единицу на месте и меняющую местами вершины 2 и 3:

$$\varphi_1(1) = 1, \quad \varphi_1(2) = 3, \quad \varphi_1(3) = 2.$$

В результате мы из графа G_1 получим помеченный граф G_2 (рис.27,б). Естественно, что графы G_1 и G_2 изоморфны друг другу, так как они получены из одного и того же немеченного графа. Однако важно то, что это — два различных графа: списки смежности этих двух графов различны:

$$\begin{array}{ll} G_1 : & \begin{array}{l} 1 \text{ смежна с } 2 \\ 2 \text{ смежна с } 1 \text{ и } 3; \\ 3 \text{ смежна с } 2 \end{array} \end{array} \quad \begin{array}{ll} G_2 : & \begin{array}{l} 1 \text{ смежна с } 3 \\ 2 \text{ смежна с } 3 \\ 3 \text{ смежна с } 1 \text{ и } 2. \end{array} \end{array}$$



Рис. 28: Идентичные помеченные графы на трех вершинах

Рассмотрим теперь перестановку φ_2 вершин графа G_1 вида

$$\varphi_2(1) = 3, \quad \varphi_2(2) = 2, \quad \varphi_2(3) = 1.$$

В результате такой перестановки мы из графа G_1 (рис.28,а) получим граф G_3 (рис.28,б). Важно, что у такого графа G_3 список смежности полностью совпадает со списком смежности исходного графа. Следовательно, в результате подобной перестановки φ_2 мы получаем тот же самый помеченный граф, что у нас был до перестановки. Такая перестановка вершин, при которой граф как помеченный объект переходит сам в себя, как раз и называется *автоморфизмом* графа.

Более формальное определение автоморфизма графа звучит следующим образом.

Определение 4.7. Взаимно-однозначное отображение $\varphi : V(G) \rightarrow V(G)$ множества $V(G)$ вершин простого графа G в себя называется автоморфизмом графа G , если она сохраняет отношение смежности в графе G . Последнее означает, что если вершины x и y соединены в графе G ребром $\{x, y\}$, то в том же самом графе G и образы $\varphi(x)$, $\varphi(y)$ этих вершин соединены между собой ребром.

4.4.3. Давайте еще раз сравним определения изоморфизма и автоморфизма графов. На первый взгляд эти определения очень похожи друг на друга, и сразу не очень понятно, чем они отличаются друг от друга. Казалось бы, отличие этих определений состоит в том, что в определении изоморфизма фигурируют различные множества $V_1(G_1)$ и $V_2(G_2)$ вершин, тогда как в определении автоморфизма эти множества совпадают. Однако это отличие в двух определениях абсолютно непринципиально. Как правило, вершины любого графа помечаются элементами одного и того же множества $[n] = \{1, 2, \dots, n\}$ первых n натуральных чисел, так что и в определении изоморфизма, и в определении автоморфизма в роли отображения φ обычно выступает какая-то перестановка $\sigma : [n] \rightarrow [n]$ первых n натуральных чисел.

Ключевое же отличие этих двух определений заключается в следующем. В определении автоморфизма мы в результате отображения φ получаем *тот же самый* граф G , тогда как в случае изоморфизма, не являющегося автоморфизмом, граф G_1 под действием отображения φ переходит в некоторый *другой* граф G_2 , отличный от исходного графа G_1 , причем это верно даже в том случае, когда множества $V_1(G_1)$ и $V_2(G_2)$ вершин этих двух графов совпадают. Легче всего это отличие описать в терминах матрицы или списка смежности графов: в случае автоморфизма список смежности графа G не изменится, тогда как в результате действия изоморфизма списки смежности графов G_1 и G_2 будут, вообще говоря, отличаться друг от друга.

4.4.4. Рассмотрим теперь множество всех автоморфизмов какого-то простого помеченного графа G . Очевидно, что тождественная перестановка id принадлежит этому множеству. Далее, для любого автоморфизма φ обратный к нему автоморфизм φ^{-1} также не меняет граф G , то есть принадлежит тому же множеству автоморфизмов. Наконец, если φ_1 и φ_2 являются автоморфизмами графа, то и их композиция задает нам некоторый автоморфизм. Иными словами, операция композиции автоморфизмов задает на множестве всех автоморфизмов группу $\text{Aut}(G)$, которая называется *группой автоморфизмов* помеченного графа G .

Так, в разобранный выше примере группа автоморфизмов помеченного графа G_1 состоит из двух перестановок — тождественной перестановки id , а также перестановки, меняющей местами вершины 1 и 3. Иными словами, эта группа изоморфна группе $\mathbb{Z}_2$.

4.4.5. Несмотря на то, что группа автоморфизмов определена для помеченного графа G , она, на самом-то деле, отражает внутреннюю симметрию исходного непомеченного графа $\tilde{G}$. Так,

например, мы видим, что показанный на рис.26,а непомеченный граф $\tilde{G}$ на трех вершинах симметричен относительно поворота на 180° . Проблема состоит лишь в том, что не очень понятно, как у непомеченного графа такие симметрии искать. И здесь нам как раз и приходится на помощь помеченный граф: мы берем непомеченный граф $\tilde{G}$, как-то размечаем его вершины, получаем помеченный граф G , вычисляем для него группу автоморфизмов $\text{Aut}(G)$, а затем говорим, что эта группа и отражает внутреннюю симметрию исходного непомеченного графа $\tilde{G}$.

4.4.6. После сделанных замечаний мы можем вернуться к проблеме, с которой мы начинали — к определению количества различных помеченных графов.

Утверждение 4.8. *Количество N различных помеченных графов, получаемых из непомеченного графа $\tilde{G}$ разметкой его вершин, рассчитывается по формуле*

$$N = \frac{n!}{|\text{Aut}(G_1)|}, \quad (3)$$

где G_1 — произвольно выбранный помеченный граф, полученный из $\tilde{G}$ некоторой разметкой его вершин.

Доказательство. Выберем произвольный помеченный граф G_1 и сосчитаем для него группу автоморфизмов $\text{Aut}(G_1)$. Предположим вначале, что порядок этой группы совпадает с $n!$. Это означает, что любая перестановка вершин переводит граф G_1 в себя, то есть у нас получается лишь один помеченный граф, отвечающий исходному непомеченному графу $\tilde{G}$. В этом случае формула (3) верна — она дает $n!/n! = 1$, то есть единственный помеченный граф.

Теперь предположим, что $|\text{Aut}(G_1)| < n!$. В этом случае среди всех $n!$ перестановок вершин найдется перестановка $\tilde{\varphi}$, не принадлежащая $\text{Aut}(G_1)$. Иными словами, эта перестановка переведет нам граф G_1 в некоторый изоморфный, но отличный от G_1 граф G_2 . Применим тогда к графу G_1 все перестановки вида

$$\tilde{\varphi} \cdot \varphi, \quad \varphi \in \text{Aut}(G_1).$$

Количество таких перестановок совпадает, очевидно, с мощностью $|\text{Aut}(G_1)|$ группы автоморфизмов графа G_1 . При этом все эти перестановки переведут граф G_1 в граф G_2 . Действительно, вначале мы с помощью перестановки φ переведем граф G_1 в себя, а затем с помощью $\tilde{\varphi}$ из графа G_1 получим граф G_2 .

Если мы на этом шаге все перестановки вершин графа G_1 перебрали, то мы на этом останавливаемся. В противном случае у нас найдется еще какая-то перестановка $\hat{\varphi}$, не принадлежащая $\text{Aut}(G_1)$ и переводящая G_1 в граф G_3 , отличный от G_1 и G_2 . Продолжая указанный процесс, мы в конце концов исчерпаем все множество Φ перестановок. При этом все это множество у нас окажется разбитым на N блоков, каждый из которых имеет мощность $|\text{Aut}(G_1)|$. Каждый такой блок соответствует перестановкам, переводящим граф G_1 в некоторый изоморфный, но отличный от G_1 помеченный граф G_i . Общее же количество N различных помеченных графов связано с количеством $n!$ всех перестановок множества $V(G_1)$ вершин графа G_1 формулой

$$N \cdot |\text{Aut}(G_1)| = n!,$$

из которой и следует формула (3). □

4.4.7. Переведем проведенные выше рассуждения на язык теории групп. Для этого рассмотрим множество $\mathcal{X}$ всех помеченных графов, построенных на n -множестве $\{1, 2, \dots, n\}$ вершин, а

также действие группы S_n перестановок на этом множестве $\mathcal{X}$. Говорят, что графы G_1 и G_2 изоморфны друг другу, если существует такая перестановка σ множества вершин, которая переводит граф G_1 в граф $G_2 = \sigma \circ G_1$, то есть такая перестановка, которая любое ребро $\{i, j\}$ в графе G_1 переводит в ребро $\{\sigma(i), \sigma(j)\}$. На языке теории групп это означает, что эти графы эквивалентны относительно действия группы S_n , то есть принадлежат одной и той же *орбите*

$$S_n \circ G = \{\sigma \circ G : \sigma \in S_n\}$$

элемента G множества $\mathcal{X}$ всех графов, построенных на n вершинах, под действием группы S_n . Каждая такая орбита на языке теории графов представляет собой некоторый непомеченный граф на n -множестве вершин.

Рассмотрим теперь подмножество $\text{Aut}(G)$ перестановок σ , любой элемент которого оставляет некоторый помеченный граф $G \in \mathcal{X}$ неподвижным:

$$\text{Aut}(G) := \{\sigma \in S_n : \sigma \circ G = G\}.$$

Несложно понять, что такое подмножество перестановок образует подгруппу $\text{Aut}(G)$ группы S_n перестановок. Такая подгруппа на языке теории групп называется *стабилизатором* элемента G множества $\mathcal{X}$. Эта подгруппа (как и любая подгруппа H группы S_n) задает разбиение группы S_n всех перестановок на блоки $\sigma \cdot \text{Aut}(G)$, называемые смежными классами группы S_n по подгруппе $\text{Aut}(G)$. Размер этих блоков одинаков и равен порядку $|\text{Aut}(G)|$ группы $\text{Aut}(G)$. Следовательно, порядок $|S_n|$ всей группы перестановок равен произведению порядка $|\text{Aut}(G)|$ группы автоморфизмов графа G на количество смежных классов $S_n / \text{Aut}(G)$ группы S_n по подгруппе $\text{Aut}(G)$ (теорема Лагранжа):

$$|S_n| = n! = |\text{Aut}(G)| \cdot |S_n / \text{Aut}(G)|.$$

В рассмотренном нами примере мы разбили множество S_3 всех перестановок вершин графа G_1 на блоки, состоящие из перестановок вида $\tilde{\varphi} \cdot \varphi$, где $\varphi \in \text{Aut}(G_1)$, и показали, что размер каждого блока равен двум — порядку группы $\text{Aut}(G_1)$.

Затем мы заметили, что любому такому блоку перестановок отвечает некоторый граф G_i , изоморфный исходному графу G . На языке теории групп данный факт формулируется в виде достаточно важного, и в то же время достаточно просто доказываемого утверждения — так называемой Orbit-Stabiliser Theorem. Согласно этой теореме, имеется взаимно-однозначное соответствие между всеми элементами орбиты $S_n \circ G$ графа G (то есть всеми изоморфными G графами на n вершинах) и фактор-множеством $S_n / \text{Aut}(G)$, то есть множеством всех смежных классов группы S_n по стабилизатору $\text{Aut}(G)$. Следствием этого утверждения как раз и является равенство

$$|S_n| = n! = |\text{Aut}(G)| \cdot |S_n \circ G| = |\text{Aut}(G)| \cdot N,$$

где N — количество графов на n вершинах, изоморфных G .

4.4.8. Дадим еще одно важное определение.

Определение 4.9. Граф G называется *асимметричным*, если его группа автоморфизмов тривиальна, то есть состоит только из тождественной перестановки множества $[n]$ его вершин.

Важно заметить, что чем выше n , тем большее количество графов асимметрично. Следовательно, при больших n количество $\mathcal{M}_n$ непомеченных графов можно оценить по формуле

$$\mathcal{M}_n \sim \frac{2^{\binom{n}{2}}}{n!}.$$

5 Основные свойства деревьев

5.1. Одним из самых важных понятий теории графов является понятие дерева.

5.1.1. Начнем с формального определения этого понятия.

Определение 5.1. Деревом называется простой связный граф без циклов. Произвольный (не обязательно связный) простой граф без циклов называется лесом.

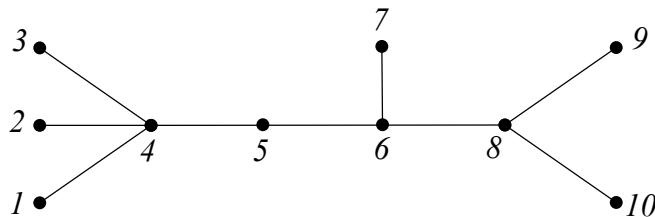


Рис. 29: Пример дерева на десяти вершинах

На рис.29 показан пример дерева T , построенного на 10 вершинах.

5.1.2. Изучим простейшие свойства деревьев. Начнем со следующего важного определения.

Определение 5.2. Вершина графа G , имеющая единичную степень, называется листом.

У дерева T , показанного на рис.29, имеется шесть листьев — вершины 1, 2, 3, 7, 9 и 10.

Лемма 5.3. У любого дерева T , построенного на $n \geq 2$ вершинах, имеется как минимум два листа.

Доказательство. Рассмотрим произвольный простой путь $P = (x_0, x_1, \dots, x_k)$ максимальной длины в T . Для изображенного на рис.29 дерева таким путем является, например, путь $P = (1, 4, 5, 6, 8, 9)$. Очевидно, что такой путь в T всегда существует, причем его длина k равна диаметру $\text{diam}(T)$ дерева T . Покажем, что концы этого пути — вершины x_0 и x_k — обязаны быть листьями.

Предположим, что это не так, то есть предположим, например, что из вершины x_k исходят два или более ребра. Одно из них — это ребро $\{x_{k-1}, x_k\}$. Любое другое исходящее из x_k ребро e не может соединять x_k ни с какой другой из оставшихся вершин пути P — в противном случае мы бы получили в графе цикл. Следовательно, ребро e соединяет x_k с какой-то новой вершиной x_{k+1} графа T . Но в таком случае мы получаем в графе T простой путь $(x_0, x_1, \dots, x_k, x_{k+1})$, длина которого на единицу больше длины пути P . А это, в свою очередь, противоречит тому, что путь P является максимальным. $\square$

5.1.3. Из доказанной леммы достаточно легко следует одно из основных свойств дерева T .

Теорема 5.4. В дереве T , построенном на n вершинах, имеется ровно $(n - 1)$ ребро:

$$|E| = |V| - 1 = n - 1.$$

Доказательство проведем индукцией по количеству n вершин в графе. База индукции очевидна — дерево, состоящее из $n = 1$ вершины, не имеет ни одного ребра. Предположим теперь, что

утверждение доказано для деревьев, построенных на n вершинах, и покажем, что оно остается верным для произвольного дерева с $(n + 1)$ -й вершиной.

Действительно, по доказанной выше лемме 5.3 у любого такого дерева имеется хотя бы один лист x . Удалим теперь этот лист x вместе с инцидентным ему ребром e . Полученный в результате такой операции граф T' останется, очевидно, связным, и дополнительных циклов в нем также не появится. Следовательно, граф T' является деревом, построенным на n вершинах. Но у такого дерева по индукционному предположению имеется $(n - 1)$ ребро. Следовательно, у исходного дерева T имеется ровно n ребер. $\square$

Достаточно очевидно и обратное утверждение.

Теорема 5.5. *Любой простой связный граф G , построенный на n вершинах и имеющий $(n - 1)$ ребро, является деревом.*

Доказательство. Действительно, выберем в графе G любую вершину x и покрасим ее, например, в красный цвет. Затем начнем последовательно выполнять следующие действия: будем выбирать в G произвольную неокрашенную вершину z , смежную с одной из уже окрашенных вершин y , и красить ее в красный цвет. Одновременно с этим будем окрашивать в тот же цвет и ребро, соединившее z с окрашенной вершиной y графа G . То, что такая вершина z всегда найдется, достаточно очевидно — граф G связный, и, выбирая любую из еще неокрашенных вершин, мы можем построить путь, соединяющий ее с одной из уже окрашенных вершин, например, с исходной вершиной x , и одна из вершин этого пути обязательно будет еще неокрашена и смежна с одной из уже окрашенных вершин графа G .

Отметим теперь следующий важный момент: в процессе выполнения этих действий мы каждый раз к уже окрашенным вершинам добавляем какую-то новую вершину графа. Цикл же в графе G может получиться только лишь в случае, когда мы на каком-то шаге пришли в уже окрашенную вершину. Иными словами, циклы у нас в процессе выполнения данного алгоритма появиться не могут. Заметим теперь, что за $(n - 1)$ шаг мы окрасим таким образом все вершины и ребра графа G , получив таким образом окрашенный связный граф без циклов. Тем самым мы показали, что граф G является деревом. $\square$

Следствие 5.6. *Всякий связный граф, построенный на n вершинах, имеет как минимум $(n - 1)$ ребро.*

Доказательство. Рассмотрим произвольный связный граф. Если такой граф не является деревом, то у него имеется хотя бы один цикл. В этом цикле мы всегда можем удалить одно ребро, и граф при этом останется связным. Будем продолжать этот процесс до тех пор, пока в графе не останется ни одного цикла. В результате получим простой связный граф без циклов, то есть дерево. Количество ребер у дерева равно $(n - 1)$. Следовательно, у исходного графа имеется как минимум $(n - 1)$ ребро. $\square$

Замечание 5.7. Данное следствие другими словами можно переформулировать и так: у всякого связного графа существует остовное дерево. Получить это дерево можно, либо удаляя лишние ребра, либо (что более рационально) используя алгоритм, использованный при доказательстве теоремы 5.5. Этот алгоритм, называемый поиском в глубину, очень часто используется в разнообразных приложениях, связанных с теорией графов.

Так, например, с помощью этого алгоритма можно найти все компоненты связности неориентированного графа. Делается это так: вначале мы полагаем, что каждая вершина в графе неокрашена (или не пройдена алгоритмом, $v.\text{visited}=\text{false}$), а число компонент связности графа

равно нулю. Затем в цикле мы перебираем все вершины графа. На каждом шаге цикла проверяем, окрашена ли текущая вершина. Если нет, то мы вызываем для нее процедуру поиска в глубину. В этой процедуре мы окрашиваем текущую вершину ($v.visited = \text{true}$), просматриваем список смежных с ней вершин, и в случае, если в этом списке обнаруживается неокрашенная вершина, рекурсивно вызываем для нее процедуру поиска в глубину. Каждый раз, когда мы выходим из процедуры во внешний цикл, мы увеличиваем количество связных компонент графа на единицу.

5.1.4. Отметим еще две важные характеристики дерева.

Определение 5.8. Простой связный граф называется минимально связным, если удаление любого ребра приводит к нарушению связности графа.

Утверждение 5.9. *Граф T является деревом тогда и только тогда, когда он является минимально связным графом.*

Доказательство практически очевидно. Действительно, предположим, что минимально связный граф T не является деревом. Тогда в нем имеется какой-то цикл. Удаление любого из входящих в этот цикл ребер связность графа не нарушит. Получили противоречие с тем, что T — минимально связный граф.

Предположим теперь, что в дереве T существует ребро $e = \{x, y\}$, после удаления которого граф остается связным. Последнее, в частности, означает, что в полученном после удаления ребра графе существует простой путь P , связывающий вершины x и y . Но тогда в исходном графе T существует цикл $P \cup \{x, y\}$. Это противоречит тому, что T является деревом. $\square$

Следствие 5.10. *Любое ребро в дереве является мостом.*

Утверждение 5.11. *Граф T является деревом тогда и только тогда, когда для любых двух его вершин существует единственный простой путь, соединяющий эти вершины.*

Доказательство. Пусть в T для любой пары вершин существует единственный соединяющий их простой путь. Тогда T является деревом. Действительно, в противном случае в T существует цикл, а любые две принадлежащие этому циклу вершины соединены между собой по крайней мере двумя простыми путями, что невозможно.

Обратно, пусть T является деревом, и пусть в нем существуют два различных простых пути P и Q , соединяющих какие-то две вершины x и y . Рассмотрим в P ребра, не принадлежащие Q , а в Q — ребра, не принадлежащие P . Объединение таких ребер образует, очевидно, один или несколько циклов, что противоречит тому, что T является деревом. $\square$

Следствие 5.12. *Пусть T есть остовное дерево связного графа G , и пусть e есть ребро, не принадлежащее T . Тогда граф $T + e$ содержит единственный цикл.*

Доказательство. Заметим, прежде всего, что любой цикл в графе $T + e$ обязан содержать ребро $e = \{x, y\}$ — в противном случае мы бы получили цикл и в дереве T , чего быть не может. Кроме того, C есть цикл в $T + e$ тогда и только тогда, когда $C - e$ есть xy -путь в исходном дереве T . Но в дереве T xy -путь единственен. Следовательно, $T + e$ содержит единственный цикл, проходящий через ребро e . $\square$

5.1.5. Рассмотрим теперь простой связный граф G , в котором имеется хотя бы одна пара отличных друг от друга остовных деревьев T и T' .

Утверждение 5.13. Пусть e есть ребро остоного дерева T , не принадлежащее остоному дереву T' . Тогда в остоном дереве T' найдется ребро e' , не принадлежащее остоному дереву T , такое, что граф $T - e + e'$ является остоным деревом графа G .

Решение. Удаление ребра $e = \{x, y\}$ разваливает T на две связные компоненты. Обозначим через V_1 и V_2 множества вершин этих двух компонент, такие, что $x \in V_1, y \in V_2$. В дереве T' существует единственный путь, соединяющий вершины x и y . Такой путь обязан иметь ребро e' из V_1 в V_2 . Так как e есть единственное ребро в дереве T между V_1 и V_2 , то ребро e' не принадлежит дереву T . Граф $T - e + e'$ является связным, содержит $n - 1$ ребро, а следовательно, является остоным деревом графа G .

Заметим, что так как в дереве T' имеется единственный путь между вершинами x и y , то добавление к этому пути ребра e приводит к появлению единственного цикла в графе $T' + e$. Удаление же из этого графа ребра e' уничтожает цикл, превращая граф $T' - e' + e$ в связный ациклический подграф — еще одно остоное дерево графа G . $\square$

5.2. Иногда наряду с обычными деревьями и лесами рассматривают так называемые *корневые деревья* и *леса*.

Определение 5.14. *Корневым деревом* называется дерево с выделенной вершиной, называемой *корнем*. *Корневым лесом* называется лес, каждая компонента связности которого представляет собой корневое дерево.

5.2.1. Наглядно корневое дерево можно представлять себе так. Представим дерево в виде некоторого физического объекта, вершинами которого являются некоторые шарниры, а ребрами — соединяющие эти шарниры трубки. Подвешивая такое дерево за какую-то вершину x , мы как раз и получим корневое дерево с корнем в вершине x .

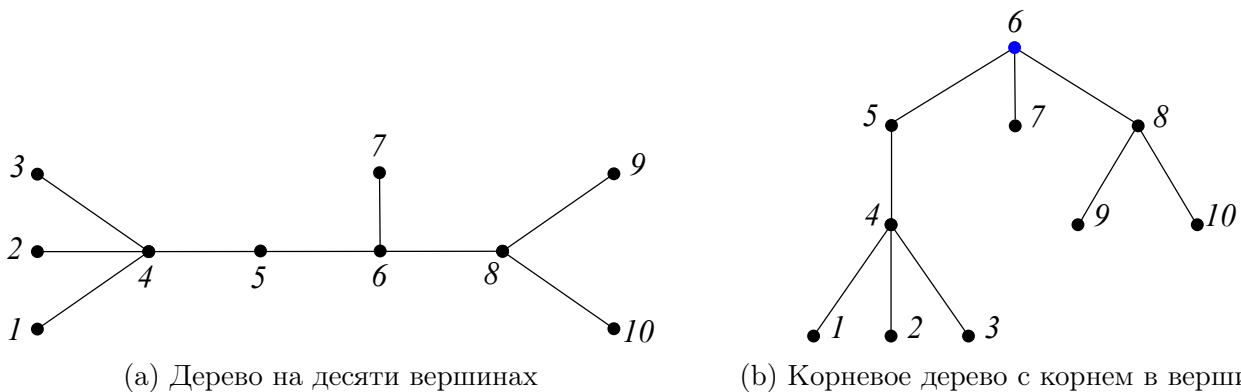


Рис. 30: Дерево и корневое дерево

В качестве примера подвесим показанное на рис.30,а дерево за вершину 6. В результате получим корневое дерево, показанное на рис.30,б.

5.2.2. В любом корневом дереве T_r с корнем в вершине r у нас имеется единственный путь из корня в произвольную вершину x . Длина этого пути определяет *уровень* вершины x в корневом дереве T_r . *Высотой* дерева называется длина наибольшего пути от корня r дерева до вершины x . Ясно, что в последнем случае вершина x обязана быть листом дерева T_r .

Так, изображенное на рис.30,б дерево T_r имеет высоту, равную трем. Уровень вершины 4 равен двум.

5.2.3. Иногда корневое дерево T_r удобно ориентировать от корня r дерева к его листьям (см.рис.31). В такой ориентации входящая степень любой вершины, отличной от корня, будет равняться единице. Корень дерева ориентированного дерева T'_r будет в этом случае источником, то есть вершиной, входящая степень которой равна нулю, а любой лист будет стоком, то есть вершиной, исходящая степень которой равна нулю.

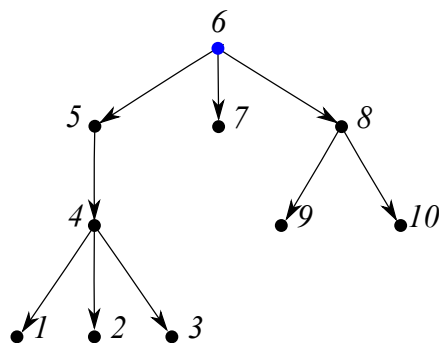


Рис. 31

В дереве T'_r вершина x называется родителем (parent) вершины y , а вершина y — ребенком (child) вершины x , если существует ориентированное ребро (x, y) , ведущее из x в y . Так, на рис.31 вершина 5 является родителем для вершины 4, а вершина 4 — ребенком для вершины 5.

Если существует ориентированный путь из вершины x в вершину z , то вершина x называется предком (ancestor) для вершины z , а вершина z — потомком (descendant) для вершины x . На рис.31 вершины 1, 2 и 3 являются потомками для вершины 5, а вершина 5 — их общим предком.

В случае, если наибольшая исходящая степень вершины в дереве равняется m , то такое дерево называется m -арным деревом. Так, дерево, показанное на рис.31, является 3-арным деревом.

5.2.4. Выбор корня в дереве T задает на множестве $V(T)$ вершин этого дерева частичный порядок, превращая это множество в частично упорядоченное. В одной из следующих глав мы подробнее поговорим о таком множестве, а сейчас лишь дадим его определение.

Определение 5.15. *Частично упорядоченным множеством* называется множество P (конечное или счетное) с введенным на нем бинарным отношением $\preceq$ (так называемым *отношением частичного порядка*), удовлетворяющим следующим трем аксиомам:

1. Рефлексивность: для любого $x \in P$ справедливо $x \preceq x$.
2. Транзитивность: если $x \preceq y$ и $y \preceq z$, то $x \preceq z$.
3. Антисимметричность: если $x \preceq y$ и $y \preceq x$, то $x = y$.

В случае корневого дерева мы будем считать, что $x \preceq y$, если x лежит на пути, соединяющем корень r дерева с вершиной y . Иными словами, это означает, что в соответствующем T_r ориентированном дереве T'_r вершина x является предком для вершины y . Кроме того, любые вершины, лежащие на пути, соединяющем корневую вершину с одним из его листов, сравнимы между собой, то есть, как говорят, образуют *цепь* в рассматриваемом частично упорядоченном множестве.

6 Линейное пространство ребер. Циклы и разрезы

6.1. Перейдем к изучению еще одного важного понятия, тесно связанного с остовным деревом графа G — понятия фундаментального цикла.

6.1.1. Пусть T есть некоторое остовное дерево связного графа G . Ребра графа G , не вошедшие в T , образуют, как говорят, *кодерево* $\bar{T}$ графа G . Кодерево $\bar{T}$ можно еще трактовать как остовный подграф, полученный из G удалением всех ребер, принадлежащих остовному дереву T . Ребра дерева T называют иногда ветвями (branches) этого дерева, а ребра кодерева $\bar{T}$ называют хордами исходного дерева T .

Рассмотрим произвольное ребро $e = \{x, y\}$, принадлежащее кодереву $\bar{T}$. Согласно утверждению ??, граф $T + e$, полученный добавлением к T ребра e , содержит единственный цикл. Этот цикл обозначается обычно C_e и называется *фундаментальным циклом* графа G по отношению к остовному дереву T .

6.1.2. Рассмотрим теперь все множество ребер из $\bar{T}$. Каждому из этих ребер e соответствует свой фундаментальный цикл C_e , полученный добавлением e к T . Множество всех таких циклов образует, как говорят, *фундаментальное множество циклов*, ассоциированное с T . Количество же циклов в этом множестве, совпадающее с количеством ребер в $\bar{T}$, называется *цикловым рангом* $\gamma(G)$ графа G . Понятно, что для любого связного графа G цикловой ранг $\gamma(G)$ не зависит от выбора остовного дерева T в графе и равен $m - n + 1$, где m — количество ребер в графе G .

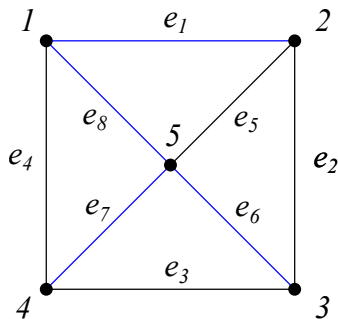


Рис. 32: Остовное дерево графа

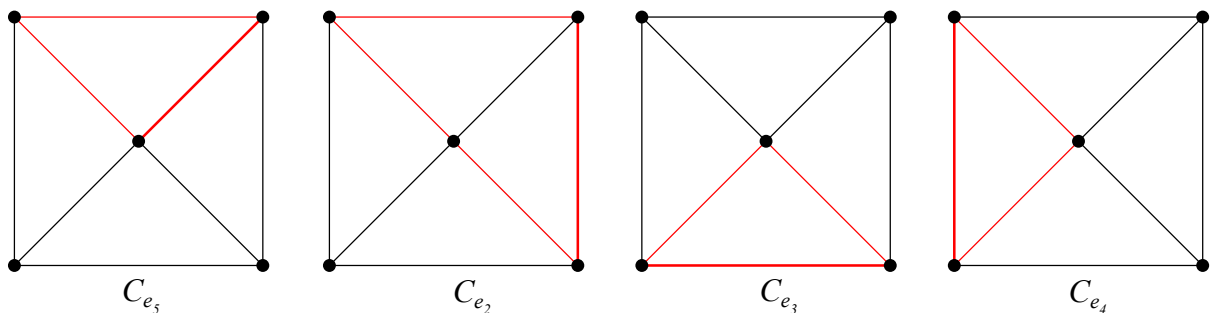


Рис. 33: Фундаментальные циклы графа

В качестве примера рассмотрим граф G , показанный на рис.32. Ребра остовного дерева T на этом рисунке помечены синим цветом. У такого графа имеется четыре ребра e_i , $i = 2, \dots, 5$, входящие в кодерево $\bar{T}$. Как следствие, цикловой ранг $\gamma(G)$ данного графа равен четырем.

Соответствующие каждому из ребер $e_i \in \bar{T}$ фундаментальные циклы C_{e_i} помечены красным цветом на рис.33.

6.2. Важно заметить, что по построению порождающее цикл C_{e_i} ребро e_i не входит ни в какие другие фундаментальные циклы графа G , порожденные остальными ребрами $e_j \in \bar{T}$. Это свойство позволяет рассматривать множество фундаментальных циклов как базис в линейном подпространстве некоторого линейного пространства ребер, к описанию которого мы сейчас и перейдем.

6.2.1. Рассмотрим простой связный граф G , построенный на n вершинах и имеющий m ребер. Введем множество $\mathcal{E}(G)$ всех подмножеств множества $E(G)$ ребер графа G . Его можно рассматривать как линейное пространство $\mathcal{E}$ над двухэлементным полем $\mathbb{F}_2 = \{0, 1\}$ относительно операции Δ симметрической разности. Нулевым элементом в этом пространстве является пустое подмножество ребер $\emptyset$. Кроме того, так как $S \Delta S = \emptyset$ для любого подмножества S ребер (то есть элемента множества $\mathcal{E}(G)$), то обратный элемент $-S$ к S в этом пространстве совпадает с S . Так как множество $E(G)$ всех ребер образует в этом пространстве базис (так называемый стандартный базис пространства $\mathcal{E}$), то размерность этого пространства равна m .

Любой элемент $S \in \mathcal{E}$ в стандартном базисе можно представить в виде бинарной строки длины m . В случае, если ребро $e_i \in S$, на i -й позиции этой строки стоит единица. В противном случае там стоит ноль. Например, множество ребер остоного дерева T , показанного на рис.32, описывается вектором

$$(1, 0, 0, 0, 0, 1, 1, 1).$$

Стандартная операция “исключающее или” (exclusive-or, или xor) двух таких строк задает бинарную операцию на множестве таких строк, эквивалентную операции симметрической разности Δ на множестве $\mathcal{E}(G)$ подмножеств множества ребер графа G . Иными словами, мы можем вместо $\mathcal{E}$ рассматривать изоморфное ему линейное пространство бинарных строк длины m над полем $\mathbb{F}_2$ с операцией xor.

6.2.2. Линейное пространство $\mathcal{E}$ можно превратить в евклидово, введя скалярное произведение его элементов S и Q по формуле

$$(S, Q) := s_1 q_1 + \dots + s_m q_m \in \mathbb{F}_2,$$

где $s_i \in \{0, 1\}$ и $q_i \in \{0, 1\}$ есть коэффициенты в разложении подмножеств S и Q по стандартному базису. Из этого определения видно, что два элемента S и Q ортогональны друг другу тогда и только тогда, когда они пересекаются по четному числу ребер.

6.2.3. Давайте теперь введем достаточно важное понятие декомпозиции графа G .

Определение 6.1. Декомпозицией графа G называется семейство $\mathcal{H}$ попарно реберно непересекающихся подграфов $H_1, \dots, H_k$ графа G , объединение которых дает все множество ребер $E(G)$:

$$\cup_{H_i \in \mathcal{H}} E(H_i) = E(G).$$

Если семейство $\mathcal{H}$ состоит только из путей либо только из циклов, то $\mathcal{H}$ называется декомпозицией G на пути (path decomposition) или на циклы (cycle decomposition).

Сразу заметим, что декомпозиция на пути для любого графа G существует всегда, так как всегда существует так называемая тривиальная декомпозиция, у которой любое подмножество H_i состоит из одиночного ребра e_i графа G . Декомпозиция же на циклы существует не всегда

— любое дерево T представляет тривиальный пример графа, для которого такая декомпозиция отсутствует.

Предположим, что граф G допускает декомпозицию на циклы. В этом случае степень любой вершины $x \in V(G)$ графа G четна. Действительно, эта степень представляет собой удвоенное количество циклов, проходящих через вершину x . Иными словами, мы показали, что любой граф, допускающий декомпозицию на циклы, является *четным* графом.

Определение 6.2. Граф G называется четным, если любая его вершина имеет четную степень.

Оказывается, верно и обратное утверждение.

Теорема 6.3 (Veblen, 1912). *Граф G допускает декомпозицию на циклы тогда и только тогда, когда G четный.*

Доказательство. Нам осталось доказать, что любой четный граф допускает декомпозицию на циклы. Будем доказывать это утверждение индукцией по количеству ребер. В случае пустого графа это утверждение верно — мы G можем разложить на пустое множество циклов. Пусть теперь G является непустым четным графом. Рассмотрим в нем подграф H , индуцированный вершинами графа G с положительными четными степенями. Так как любая вершина этого подграфа больше или равна двум, то такой подграф содержит цикл C . Рассмотрим тогда в графе G остовный подграф $G' := G \setminus E(C)$, полученный удалением ребер цикла C . Этот подграф имеет меньшее количество ребер, а потому допускает разложение на циклы. Но тогда и граф G допускает разложение на те же циклы плюс цикл C . $\square$

6.2.4. Рассмотрим теперь множество всех четных подграфов заданного графа G .

Утверждение 6.4. *Симметрическая разность двух четных подграфов H_1 и H_2 есть снова четный подграф.*

Доказательство. Предположим вначале, что подграфы H_1 и H_2 не имеют общих вершин. В этом случае их симметрическая разность представляет собой пустой подграф, а он по определению является четным. В противном случае рассмотрим произвольную общую вершину x четных подграфов H_1 и H_2 . Обозначим через S_1 подмножество инцидентных x ребер, принадлежащих H_1 , а через S_2 — подмножество ребер, принадлежащих H_2 . Симметрическая разность $S_1 \Delta S_2$ представляет собой объединение двух блоков — подмножества $B_1 = S_1 \setminus (S_1 \cap S_2)$ ребер, принадлежащих H_1 и не принадлежащих H_2 , а также подмножества $B_2 = S_2 \setminus (S_1 \cap S_2)$ ребер, принадлежащих H_2 и не принадлежащих H_1 . Как следствие,

$$|S_1 \Delta S_2| = |S_1 \setminus (S_1 \cap S_2)| + |S_2 \setminus (S_1 \cap S_2)| = |S_1| + |S_2| - 2|S_1 \cap S_2|.$$

Так как $|S_1|$ и $|S_2|$ четны, то четным является и число $|S_1 \Delta S_2|$. Иными словами, все вершины остоного подграфа, полученного в результате симметрической разности двух четных подграфов H_1 и H_2 , имеют четную степень, а следовательно, подграф $H = H_1 \Delta H_2$ является четным подграфом. $\square$

Следствие 6.5. *Множество четных подграфов замкнуто относительно операции Δ , то есть образует в $\mathcal{E}$ линейное подпространство $\mathcal{C}$.*

Учитывая теорему Veblen, мы можем сказать, что это линейное подпространство порождается всеми циклами графа G . Как следствие, подпространство $\mathcal{C}$ называется *пространством циклов* графа G .

6.2.5. Ранее мы сказали, что любой фундаментальный цикл C_{e_i} , связанный с некоторым остовным деревом T графа G , содержит ребро e_i (хорду дерева T), не входящее ни в какой другой фундаментальный цикл. С точки зрения линейной алгебры это означает, что такие циклы являются линейно независимыми: линейная комбинация

$$\alpha \cdot (0, 1, \dots, \underbrace{1}_i, \dots, \underbrace{0}_j, \dots, 0) + \beta \cdot (1, 1, \dots, \underbrace{0}_i, \dots, \underbrace{1}_j, \dots, 0) = (0, 0, \dots, \underbrace{0}_i, \dots, \underbrace{0}_j, \dots, 0)$$

таких циклов равна пустому циклу (нулевому вектору) тогда и только тогда, когда коэффициенты $\alpha = \beta = 0$. Следовательно, мы имеем $m - n + 1$ линейно независимый вектор в пространстве $\mathcal{C}$ циклов графа G . Нам хочется показать, что набор таких векторов образует базис пространства $\mathcal{C}$. В принципе, для этого нужно доказать, что любой цикл графа может быть представлен в виде линейной комбинации этих фундаментальных циклов C_{e_i} . Однако можно поступить несколько проще — на самом деле, нам достаточно показать, что размерность пространства $\mathcal{C}$ равна $m - n + 1$. Если это так, то любой набор из $(m - n + 1)$ -го вектора, и в том числе набор фундаментальных циклов, образует в $\mathcal{C}$ базис.

Для достижения этой цели мы, в свою очередь, рассмотрим в пространстве $\mathcal{E}$ еще одно подпространство — так называемое пространство $\mathcal{B}$ разрезов графа G . Затем мы докажем, что, во-первых, это пространство ортогонально $\mathcal{C}$, а во-вторых, что в этом пространстве существует $n - 1$ линейно независимых векторов. Так как $(n - 1) + (m - n + 1) = m$, то отсюда сразу будет следовать, во-первых, что размерность пространства $\mathcal{C}$ равна $m - n + 1$, а во-вторых, что фундаментальные циклы C_{e_i} образуют в этом пространстве базис.

6.3. Для определения подпространства $\mathcal{B}$ нам понадобятся несколько важных понятий.

6.3.1. Начнем с понятия реберно разделяющего множества F .

Определение 6.6. Подмножество $F \subset E(G)$ называется *реберно разделяющим множеством* связного графа G , если после удаления всех ребер из F граф $G - F$ перестает быть связным.

Рассмотрим в качестве примера граф, показанный на рис.34. Синим цветом на каждом из этих рисунков помечены ребра, после удаления которых граф перестает быть связным. Следовательно, любое из таких подмножеств ребер образует реберно разделяющее множество.

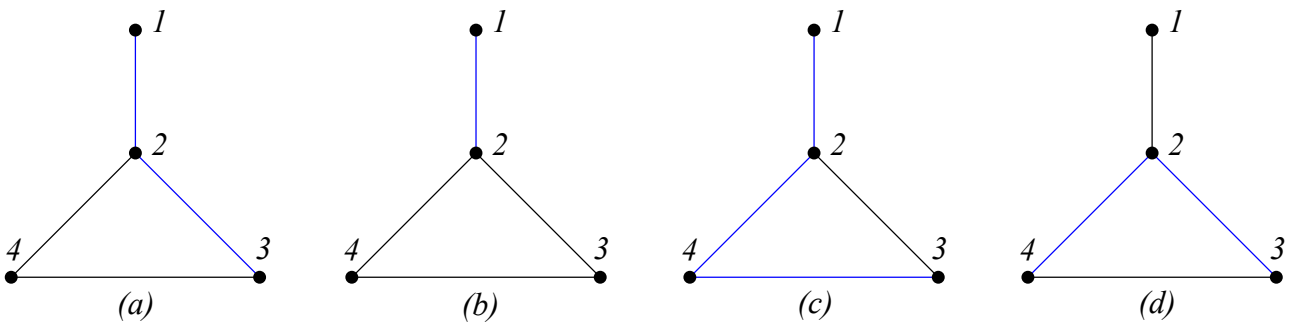


Рис. 34

6.3.2. Следующее важное для нас понятие — это понятие *реберного разреза* (edge cut). Пусть у нас имеются два подмножества S_1 и S_2 множества $V(G)$ вершин графа G (не обязательно непересекающихся). Рассмотрим все ребра графа G , один из концов которых принадлежит S_1 , а второй — S_2 . Множество таких ребер обозначается обычно через $[S_1, S_2]$. Случаю $S_1 = S_2 = S$ отвечает набор ребер, оба конца которых лежат в S . Нас же сейчас будет интересовать противоположный случай, а именно, случай, когда $S_2 = V(G) \setminus S_1$.

Определение 6.7. Пусть S есть некоторое подмножество множества вершин $V(G)$ графа, а $\bar{S} = V(G) \setminus S$. Набор ребер $[S, \bar{S}]$ называется реберным разрезом графа G , связанным с подмножеством S . Иногда он также называется кограницей S и обозначается через $\partial(S)$.

Очевидно, что $\partial(S) = \partial(\bar{S})$, а также то, что $\partial(V(G)) = \emptyset$. Кроме того, граф G является двудольным, если в нем существует такое подмножество S вершин, что $\partial(S)$ совпадает со всем множеством $E(G)$ ребер графа G . Реберный разрез $\partial(x)$, связанный с отдельной вершиной x , является, как говорят, тривиальным реберным разрезом и представляет собой набор ребер, инцидентных x . По аналогии с этим частным случаем размер $|\partial(S)|$ подмножества ребер $\partial(S)$ в случае мультиграфа без петель называется *степенью* подмножества вершин S .

Так как в графе $G - [S, \bar{S}]$ пути между S и $\bar{S}$ отсутствуют, то любой непустой реберный разрез $[S, \bar{S}]$ является реберно-разделяющим множеством. Обратное, конечно же, неверно. Так, реберно-разделяющее множество, показанное синим цветом на рис.34,а, реберным разрезом не является — наряду с ребром $\{1, 2\}$ оно содержит лишнее ребро $\{2, 3\}$. Все остальные реберно-разделяющие множества, показанные на рис.34,б, с и д, являются одновременно и реберными разрезами. Так, на рис.34,б представлен реберный разрез $\partial(1)$, связанный с вершиной 1, на рис.34,с — реберный разрез $\partial(1, 4)$ подмножества вершин $\{1, 4\}$, а на рис.34,д — реберный разрез $\partial(1, 2)$, связанный с подмножеством $\{1, 2\}$ множества вершин.

6.3.3. Наконец, введем последнее нужное нам в этом месте понятие — понятие *минимального реберного разреза* (bond).

Определение 6.8. Реберный разрез называется минимальным (bond), если он представляет собой непустой реберный разрез, любое правильное подмножество которого не является реберным разрезом.

Среди показанных на рис.34 реберных разрезов только два являются минимальными — реберные разрезы $\partial(1)$ (рис.34,б) и $\partial(1, 2)$ (рис.34,д). Реберный разрез $\partial(1, 4)$, показанный на рис.34,с, минимальным не является — он содержит в качестве своего правильного подмножества ребро $\{1, 2\}$, являющееся реберным разрезом, связанным с вершиной 1.

6.3.4. Докажем теперь для связного графа G следующий важный критерий минимальности реберного разреза.

Теорема 6.9. В любом связном графе G любой непустой реберный разрез $\partial(S)$ является минимальным тогда и только тогда, когда граф $G - \partial(S)$ имеет в точности две связные компоненты.

Доказательство. Предположим вначале, что граф $G - \partial(S)$ имеет в точности две компоненты связности G_1 и G_2 . Реберный разрез $F := \partial(S)$, по определению, содержит те и только те ребра, которые соединяют в графе G эти две компоненты. Поэтому, если у F существует правильное подмножество F' , то граф $G - F'$ содержит в качестве подграфов те же самые две компоненты связности G_1 и G_2 графа $G - F$ плюс по крайней мере одно ребро, соединяющее эти компоненты. Следовательно, граф $G - \partial(G)$ является связным, а $\partial(S)$ — минимальным реберным разрезом.

Обратно, предположим, что граф $G - \partial(S)$ содержит более чем две связные компоненты. Это означает, что либо подграф H , индуцированный подмножеством вершин S , либо подграф $\bar{H}$, индуцированный подмножеством вершин $\bar{S}$, содержит по крайней мере две связные компоненты. Пусть, для определенности, таковым является подграф H . В этом случае $S = A \cup B$, причем

между вершинами подмножеств A и B ребра отсутствуют. Но тогда, например, $[A, \bar{A}]$ есть правильное подмножество множества $\partial(S)$, являющееся реберным разрезом, а это противоречит минимальности $\partial(S)$. $\square$

6.3.5. Докажем, наконец, следующее важное утверждение.

Утверждение 6.10. *Симметрическая разность двух различных реберных разрезов есть снова реберный разрез, причем*

$$\partial(X) \triangle \partial(Y) = \partial(X \triangle Y). \quad (4)$$

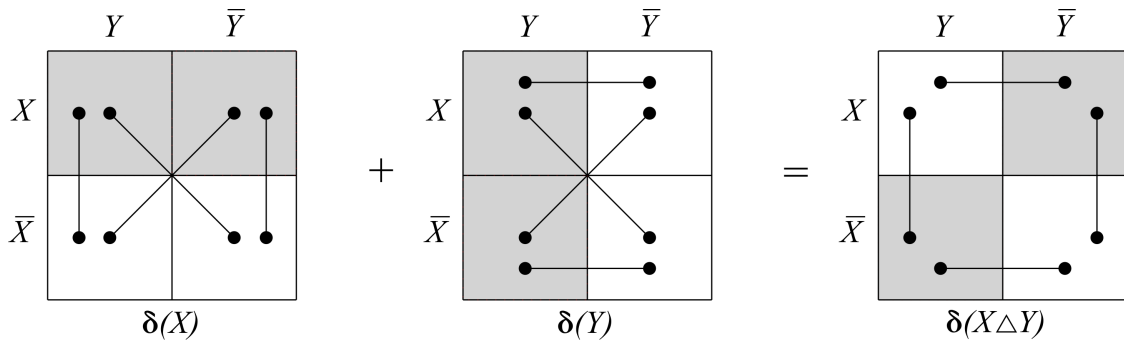


Рис. 35

Доказательство. То, что симметрическая разность двух различных реберных разрезов есть снова реберный разрез, следует из формулы (4). Для ее доказательства рассмотрим рис.35. Все варианты разбиения множества $V(G)$ с помощью двух подмножеств имеют следующий вид:

$$X \cap Y, \quad X \setminus Y, \quad Y \setminus X, \quad \bar{X} \cap \bar{Y}.$$

Все эти варианты можно наглядно представить в виде следующей таблицы:

	Y	$\bar{Y}$
X	$X \cap Y$	$X \setminus Y$
$\bar{X}$	$Y \setminus X$	$\bar{X} \cap \bar{Y}$

Правые нижние четыре клетки этой таблицы образуют разбиение квадрата на четыре части, как раз и описывающие все четыре блока разбиения множества $V(G)$ вершин графа G .

Теперь рассмотрим набор ребер $\partial(X)$. Эти ребра могут исходить либо из $X \cap Y$, либо из $X \setminus Y$, и входить в любые две клетки нижней части описанного выше квадрата. В итоге получается четыре типа ребер (см.рис.35). Аналогично, ребра из набора ∂Y могут выходить из левых клеток квадрата в любую из правых ее клеток. В итоге вновь получаем четыре варианта. Теперь осталось образовать из этих наборов ребер симметрическую разность. Видно, что в нее не будут входить те ребра, которые соединяют диагональные клетки квадрата. Но это и есть $\partial(X \triangle Y)$ — множеству вершин $X \triangle Y$ отвечают правый верхний и левый нижний блоки разбиения $V(G)$, а ребра из $\partial(X \triangle Y)$ соединяют их с двумя оставшимися блоками разбиения $V(G)$. $\square$

Следствие 6.11. *Множество всех реберных разрезов вместе с операцией $\triangle$ образует в линейном пространстве $\mathcal{E}$ всех подмножеств множества $E(G)$ ребер графа G линейное подпространство $\mathcal{B}$, называемое линейным пространством реберных разрезов графа G .*

6.4. Напомним, что в начале этого параграфа мы ввели важное понятие фундаментального множества циклов графа G . С этим понятием тесно связано не менее важное понятие фундаментального множества разрезов в графе G .

6.4.1. Пусть T есть произвольное остовное дерево связного графа G . Удаление любого ребра e этого дерева разбивает все множество вершин дерева T на два блока X и Y . Рассмотрим тогда минимальный реберный разрез $\partial(X) =: \partial_e$, связывающий в исходном графе G вершины из блоков X и Y . Такой реберный разрез называется *фундаментальным разрезом*, связанным с деревом T , а множество всех таких разрезов — *фундаментальным множеством разрезов*, связанным с деревом T .

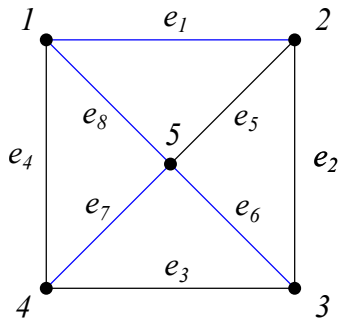


Рис. 36: Остовное дерево графа

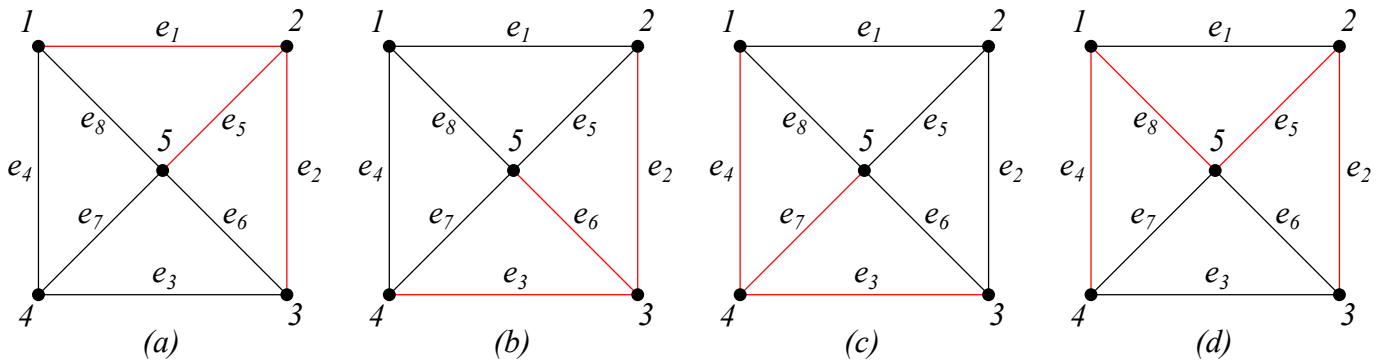


Рис. 37: Фундаментальные циклы графа

6.4.2. В качестве примера рассмотрим граф G , показанный на рис.36. Остовное дерево T этого графа состоит из ребер e_1, e_6, e_7 и e_8 (синие ребра на рисунке). Удаляя ребро e_1 , мы разобьем множество вершин дерева T на два блока — блок, состоящий из единственной вершины 2, а также блок, состоящий из вершин $\{1, 3, 4, 5\}$. Соответствующий этому разбиению реберный разрез $\partial(2) = \partial_{e_1}$ состоит из ребер e_1, e_2, e_5 (ребра, помеченные красным цветом на рис.37,a). Реберный разрез ∂_{e_6} , отвечающий ребру e_6 остоного дерева, равен $\partial(3) = \{e_2, e_3, e_6\}$ (рис.37,b), а реберный разрез ∂_{e_7} , соответствующий ребру e_7 дерева T , есть $\partial(4) = \{e_3, e_4, e_7\}$ (рис.37,c). Наконец, ребру e_8 отвечает реберный разрез $\partial_{e_8} = \partial(1, 2) = \{e_2, e_4, e_5, e_8\}$ (рис.37,d).

Важно заметить, что любой фундаментальный реберный разрез состоит из ребра e_i остоного дерева, которое порождает этот разрез, а также еще из каких-то ребер, которые соединяют X и Y в графе G , но не соединяют их в остоном дереве T . Иными словами, все остальные ребра этого разреза дереву T не принадлежат, то есть все они принадлежат кодереву $\bar{T}$.

Как следствие, ребро e_i остоного дерева T , порождающее соответствующий разрез ∂_{e_i} , по построению входит только в этот фундаментальный разрез и не входит ни в какой другой фун-

даментальный разрез, связанный с другими ребрами e_j остовного дерева T . На языке пространства $\mathcal{B}$ реберных разрезов это означает, что такие циклы образуют набор линейно независимых векторов этого пространства. Количество таких линейно независимых векторов совпадает с количеством ребер в остовном дереве связного графа G и равно $n - 1$.

6.4.3. Как и в случае фундаментальных циклов, мы хотим доказать, что любой реберный разрез графа может быть представлен в виде линейной комбинации некоторых фундаментальных разрезов ∂_{e_i} , то есть что набор фундаментальных разрезов графа G образует базис пространства $\mathcal{B}$. Отсюда, в частности, следовало бы, что размерность данного подпространства равна $n - 1$, а количество всех реберных разрезов связного графа G равно 2^{n-1} . В случае несвязного графа G , содержащего k компонент, размерность $\mathcal{B}$ равна $n - k$.

Можно этот факт доказать и напрямую. Мы же докажем этот факт, равно как и аналогичное утверждение, касающееся фундаментального множества циклов, доказав, что ортогональное дополнение пространства $\mathcal{B}$ разрезов совпадает с пространством $\mathcal{C}$ циклов и наоборот.

Теорема 6.12. *Пространство циклов $\mathcal{C}$ и пространство реберных разрезов $\mathcal{B}$ любого графа связаны соотношениями*

$$\mathcal{C} = \mathcal{B}^\perp, \quad \mathcal{B} = \mathcal{C}^\perp.$$

Доказательство. Заметим, прежде всего, что любой цикл C графа G пересекается с любым реберным разрезом $\partial(X)$ по четному числу ребер. Действительно, любая вершина цикла C входит либо в подмножество X вершин, либо в подмножество $\bar{X} = V(G) \setminus X$ вершин. При движении по циклу C количество переходов из X в $\bar{X}$ совпадает с количеством обратных переходов из $\bar{X}$ в X . Следовательно, $|E(C) \cap \partial(X)|$ есть четное число. А это, в свою очередь, означает, что любой цикл ортогонален реберному разрезу $\partial(X)$, то есть что $\mathcal{C} \subseteq \mathcal{B}^\perp$.

Обратно, рассмотрим произвольное подмножество ребер $F \notin \mathcal{C}$. Согласно теореме Veblen, среди вершин, инцидентных этому подмножеству ребер, найдется хотя бы одна вершина x , которая инцидентна нечетному количеству ребер из F . Следовательно, подмножество F и реберный разрез $\partial(x)$ пересекаются по нечетному количеству ребер, а потому их скалярное произведение равно единице. Иными словами, в пространстве $\mathcal{B}$ реберных разрезов обязательно найдется хотя бы один элемент — разрез $\partial(x)$, не ортогональный F . Поэтому любое подмножество $F \notin \mathcal{C}$ ортогональному дополнению $\mathcal{B}$ не принадлежит. С учетом $\mathcal{C} \subseteq \mathcal{B}^\perp$ отсюда следует, что $\mathcal{C}$ в точности совпадает с $\mathcal{B}^\perp$.

Для доказательства равенства $\mathcal{B} = \mathcal{C}^\perp$ теперь достаточно показать, что $\mathcal{B} = (\mathcal{B}^\perp)^\perp$. Мы уже доказали, что любой реберный разрез ортогонален циклу, то есть $\mathcal{B} \subseteq \mathcal{C}^\perp = (\mathcal{B}^\perp)^\perp$. Но так как

$$|\mathcal{B}| + |\mathcal{B}^\perp| = m = |\mathcal{C}| + |\mathcal{C}^\perp| = |\mathcal{B}^\perp| + |(\mathcal{B}^\perp)^\perp|,$$

то размерности $\mathcal{B}$ и $(\mathcal{B}^\perp)^\perp$ совпадают, а значит, $\mathcal{B} = (\mathcal{B}^\perp)^\perp$. □

Следствие 6.13. *Для любого связного графа G , построенного на n вершинах и имеющего m ребер, размерности пространств $\mathcal{B}$ и $\mathcal{C}$ равны*

$$|\mathcal{B}| = n - 1, \quad |\mathcal{C}| = m - n + 1, \quad (5)$$

а фундаментальные разрезы и фундаментальные циклы образуют в этих пространствах базис.

Решение. Действительно, следствием теоремы 6.12 является равенство

$$|\mathcal{B}| + |\mathcal{C}| = m.$$

Поэтому для доказательства равенств (5) достаточно в пространствах $\mathcal{B}$ и $\mathcal{C}$ найти $n-1$ и $m+n-1$ линейно независимых векторов. А мы такие векторы уже нашли: в пространстве $\mathcal{B}$ таковыми являются фундаментальные разрезы, а в пространстве $\mathcal{C}$ — фундаментальные циклы. $\square$

7 Подсчет остовных деревьев в графе. Матричная теорема о деревьях

7.1. Доказанную в предыдущем параграфе формулу Кэли мы можем также интерпретировать как формулу подсчета количества всех остовных деревьев в полном графе K_n . Основная задача данного параграфа — научиться решать более общую задачу подсчета количества $t(G)$ всех различных остовных деревьев для произвольного связного графа G .

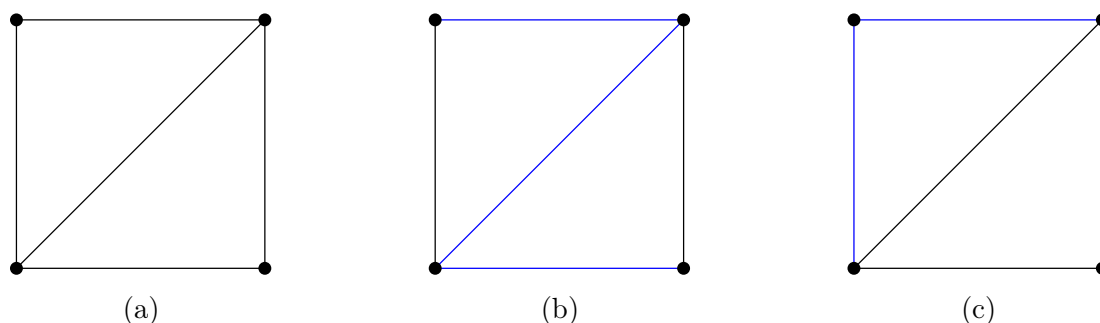


Рис. 38

7.1.1. Начнем с примера графа G , показанного на рис.38,а. У такого графа имеется восемь различных остовных деревьев — четыре из них проходят через диагональ этого графа (см.рис.38,б), а четыре охватывают внешний цикл графа G (рис.38,с).

В рассмотренном примере мы все множество остовных деревьев разбили на два блока, один из которых содержит деревья, проходящие через выбранное ребро e , а второй — деревья, через e не проходящие. Подобного рода рассуждения лежат в основе рекурсивной процедуры подсчета количества $t(G)$ всех остовных подграфов графа G , основанной на следующем утверждении.

Утверждение 7.1. Пусть e — ребро, не являющееся петлей графа G . Тогда

$$t(G) = t(G - e) + t(G \setminus e), \quad (6)$$

где $G - e$ — граф, полученный из G удалением ребра e , а $G \setminus e$ — граф, полученный из G стягиванием этого ребра.

Доказательство. Все множество остовных деревьев графа G разбивается на два блока — блок деревьев, содержащих ребро e , и блок деревьев, e не содержащих. Очевидно, что все остовные деревья, которые не содержат ребро e , являются остовными деревьями графа $G - e$. Поэтому количество таких деревьев совпадает с количеством $t(G - e)$. Давайте теперь докажем, что количество остовных деревьев, содержащих e , совпадает с количеством остовных деревьев графа $G \setminus e$.

Действительно, рассмотрим остовное дерево T графа G , содержащее ребро e . Стягивание этого ребра уменьшает количество ребер и количество вершин в графе $T \setminus e$ на единицу. Кроме того,

эта операция не приводит к потере связности получающегося графа $G \setminus e$. Следовательно, граф $T \setminus e$ вновь является деревом — остовным деревом графа $G \setminus e$. Кроме того, никакие два различных остовных дерева графа G , содержащих e , не сольются в одно в результате этой операции. Обратно, любое остовное дерево в графе $G \setminus e$ превращается в остовное дерево графа G , содержащее e , расщеплением вершины на две и добавлением между ними ребра. $\square$



Рис. 39: Операция стягивания ребра

7.1.2. Сделаем несколько замечаний, касающихся вычислительных аспектов расчета $t(G)$ по формуле (6). Прежде всего, даже если мы стартуем с простого связного графа, то в процессе удаления ребер граф может перестать быть связным, а в процессе стягивания ребер в графе могут появиться петли и мультиребра. Так, например, после стягивания ребра e_1 у графа G , показанного на рис.39,а, ребро e_2 , связывающее в исходном графе G те же вершины, что и e_1 , превращается в петлю.

Со случаем несвязных графов справиться просто — у таких подграфов остовных деревьев не существует, поэтому для таких графов $t(G) = 0$. Мультиребра в графе не мешают нам пользоваться рекуррентным соотношением (6). Однако при появлении петель нам нужно быть осторожнее — рекуррентное соотношение (6) перестает работать в случае, когда ребро e представляет собой петлю. Рассмотрим, к примеру, граф, состоящий из одиночной вершины и петли. У такого графа имеется единственное остовное дерево, состоящее из изолированной вершины. Формула же (6) применительно к данному случаю даст нам $t(G) = 2$. Как следствие, в процессе подсчета по формуле (6) нам нужно такие петли либо просто игнорировать, либо сразу удалять в процессе их появления — на количество остовных деревьев это никак не повлияет.

Наконец, рекурсивный подсчет по формуле (6) требует задания начальных условий. В процессе удаления и стягивания ребер мы в конце концов получим графы, любое ребро которых представляет собой петлю. Если в полученном графе при этом останется одна вершина, то количество остовных деревьев в таком графе равно единице. В противном случае $t(G) = 0$.

7.1.3. Полученное нами рекуррентное соотношение (6) приводит к достаточно громоздким вычислениям даже для случая графов, имеющих не очень большое количество m ребер. С увеличением m количество вычислений экспоненциально возрастает. К счастью, наряду с (6) для подсчета $t(G)$ имеется замкнутая аналитическая формула, полученная еще в 1847 году Кирхгофом. Эта формула носит название матричной теоремы о деревьях (Matrix Tree Theorem) и выражает $t(G)$ через определитель некоторой матрицы. Определитель же квадратной матрицы размером $n \times n$ мы можем сосчитать, используя менее чем n^3 операций.

Рассмотрим связный граф G без петель, построенный на множестве вершин $[n] = \{1, 2, \dots, n\}$. Мы знаем, что матрица M_a смежности такого графа симметрична, и все диагональные элементы этой матрицы равны нулю. Возьмем теперь матрицу M_d размером $n \times n$, в которой диагональные

элементы $d_{i,i}$ равны степеням $\deg(i)$ соответствующих вершин в графе G , а остальные элементы равны нулю, и вычтем из нее матрицу M_a . В результате получим матрицу $L = M_d - M_a$, которая носит название *матрицы Лапласа*.

Теорема 7.2 (Matrix tree theorem). *Количество $t(G)$ остовных деревьев графа G равно определителю матрицы L^* , полученной удалением из матрицы Лапласа L произвольной строки и произвольного столбца.*

Прежде чем доказывать эту теорему, посмотрим, как она работает для примера графа G , показанного на рис.38,а. Для такого графа матрица Лапласа имеет вид

$$L = \begin{pmatrix} 3 & -1 & -1 & -1 \\ -1 & 3 & -1 & -1 \\ -1 & -1 & 2 & 0 \\ -1 & -1 & 0 & 2 \end{pmatrix}.$$

Удаляя из нее первую строку и первый столбец, мы получаем матрицу

$$L^* = \begin{pmatrix} 3 & -1 & -1 \\ -1 & 2 & 0 \\ -1 & 0 & 2 \end{pmatrix},$$

определитель которой равен 8.

7.1.4. При доказательстве матричной теоремы о деревьях удобно перейти от неориентированного графа G к некоторому ориентированному графу D , полученному произвольной ориентацией ребер графа G . Оказывается, что при этом имеется достаточно простая связь между матрицей Лапласа L и матрицей M_i инцидентности орграфа D . Именно, справедлива следующая

Лемма 7.3. *Пусть D есть орграф, полученный из G произвольной ориентацией ребер графа G , а M_i есть матрица инцидентности орграфа D . Тогда*

$$L = M_i \cdot M_i^T.$$

Доказательство. Напомним, что в матрице M_i элемент $m_{i,j} = 1$, если ребро e_j выходит из вершины i , $m_{i,j} = -1$ в случае, если e_j входит в вершину i , и $m_{i,j} = 0$ в остальных случаях. Любой элемент $l_{i,j}$ матрицы $M_i \cdot M_i^T$ есть скалярное произведение i -й и j -й строк матрицы M_i .

В случае $i \neq j$ это скалярное произведение дает нам столько -1 , сколько имеется ребер, соединяющих вершины i и j . Действительно, если $m_{i,k} = 1$ и $m_{j,k} = -1$, то это означает, что в графе G имеется ребро e_k , соединяющее вершины i и j . В противном случае хотя бы один из этих сомножителей равен нулю, поэтому слагаемое $m_{i,k} \cdot m_{j,k} = 0$ и в скалярное произведение вклада не дает.

В случае $i = j$ мы имеем $m_{i,k}^2$, что дает единицу в случае, если ребро e_k инцидентно вершине i . Суммируя по всем k , мы получаем на диагонали число $l_{i,i}$, равное степени вершины i . $\square$

Следствие 7.4. *Матрица L^* , полученная из матрицы Лапласа L удалением произвольной строки и произвольного столбца, равна*

$$L^* = M_i^* \cdot (M_i^*)^T,$$

где M_i^* есть матрица, полученная из M_i удалением произвольной строки.

7.1.5. Приступим теперь к доказательству матричной теоремы о деревьях. Пусть B есть квадратная подматрица $(n-1) \times (n-1)$ матрицы M_i^* . Пусть также G есть неориентированный граф, полученный из D заменой любого ориентированного ребра на неориентированное (underlying graph). Столбцы подматрицы B отвечают какому-то набору из $n-1$ ориентированных ребер в орграфе D . Соответствующие этому набору неориентированные ребра вместе с инцидентными им вершинами образуют в underlying graph G некоторый подграф H . Покажем, что определитель подматрицы B равен ± 1 в случае, если отвечающий этой подматрице подграф H образует остовное дерево графа G , и нулю в противном случае.

Предположим вначале, что подграф H представляет собой остовное дерево. В этом случае в H обязательно найдутся хотя бы два листа. Так как в подматрице B отсутствует только лишь одна строка исходной матрицы M_i инцидентности орграфа D , то в ней обязательно найдется хотя бы одна строка j , отвечающая вершине x подграфа H степени 1. В этой строке имеется единственный ненулевой элемент $m_{j,k} = \pm 1$. Раскладывая определитель матрицы B по j -й строке, мы получаем равенство вида

$$\det B = m_{j,k} \cdot \det B' = \pm 1 \cdot \det B',$$

где B' — подматрица, полученная из B вычеркиванием j -й строки и k -го столбца. На языке подграфа H данная операция отвечает удалению вершины x степени 1 вместе с инцидентным ей ребром. В результате мы получаем подграф $H - x$, который вновь является остовным деревом, но уже графа $G - x$. Продолжая описанный процесс далее, мы на последнем шаге придем к остовному дереву, построенному на двух вершинах, для которого $\det B = \pm 1$.

Теперь допустим, что соответствующий подматрице B подграф H не является остовным деревом. Следовательно, в таком подграфе обязательно найдется цикл C . Покажем, что в этом случае обязательно найдется нетривиальная линейная комбинация столбцов, равная нулю. Отсюда будет следовать, что строки матрицы линейно зависимы, то есть что $\det B = 0$.

Рассмотрим вначале ребра, не входящие в цикл C . Соответствующие им столбцы матрицы B включим в линейную комбинацию с коэффициентами, равными нулю. Предположим теперь, что направление всех ребер, входящих в цикл, совпадает с каким-то из направлений обхода цикла C . В этом случае линейная комбинация соответствующих этим ребрам столбцов равнялась бы нулю (сколько ребер вошло в вершину x цикла, столько же ребер из нее вышло). Однако в цикле C могут найтись и ребра, которые направлены в сторону, противоположную выбранному направлению обхода цикла C . Тогда мы можем включить отвечающие этим ребрам столбцы в линейную комбинацию с коэффициентами, равными -1 , и вновь получим линейную комбинацию столбцов, равную нулевому столбцу.

Для завершения доказательства теоремы нам теперь достаточно воспользоваться формулой Бине-Коши, позволяющей вычислять определитель квадратной матрицы, являющейся произведением двух прямоугольных матриц. Согласно этой формуле,

$$\det L^* = \det M_i^* \cdot (M_i^*)^T = \sum (\det B)^2,$$

где суммирование проводится по всем $(n-1) \times (n-1)$ -подматрицам B матрицы M_i^* . Мы только что показали, что $(\det B)^2 = 1$ в случае, если B отвечает остовному поддереву графа G , и $(\det B)^2 = 0$ в противном случае. Тем самым матричная теорема о деревьях доказана. $\square$