**Основы теории графов**

Многие задачи сводятся к рассмотрению совокупности объектов, существенные свойства которых описываются связями между ними. Например: карта автомобильных дорог (рассматривается только возможность связи между отдельными пунктами, не учитывая качества дорог и т.д), различные связи между людьми (родство, соседство, знакомство …)

На практике часто бывает полезно изобразить некоторую ситуацию в виде рисунков, составленных из точек (вершин), представляющих ситуация, и линий (ребер), соединяющих определенные пары этих вершин и представляющих связь между ними. Такие рисунки называют графами. Графы встречаются в разных областях под разными названиями: «структуры» - в гражданском строительстве, «сети» - в электротехнике, «социограммы» - в социологии и экономике, «молекулярные структуры» - в химии.

**Теория графов** систематически и последовательно изучает свойства графов, о которых можно сказать, что они состоят из множеств точек и множеств линий, отображающих связи между этими точками. Основателем теории графов считается Леонард Эйлер (1707-1882), решивший в 1736 году известную в то время задачу о кёнигсбергских мостах.

Издавна среди жителей [Кёнигсберга](https://ru.wikipedia.org/wiki/%D0%9A%D1%91%D0%BD%D0%B8%D0%B3%D1%81%D0%B1%D0%B5%D1%80%D0%B3) была распространена такая загадка: как пройти по всем городским мостам (через реку [Преголя](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B5%D0%B3%D0%BE%D0%BB%D1%8F" \o "Преголя)), не проходя ни по одному из них дважды. Многие пытались решить эту задачу как теоретически, так и практически, во время прогулок.

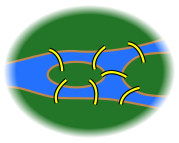
В 1736 году задача о семи мостах заинтересовала выдающегося математика, члена Петербургской академии наук Леонарда Эйлера, о чём он написал в письме итальянскому математику и инженеру [Джованни Джакобо Маринони](https://ru.wikipedia.org/w/index.php?title=%D0%9C%D0%B0%D1%80%D0%B8%D0%BD%D0%BE%D0%BD%D0%B8,_%D0%94%D0%B6%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%B8_%D0%94%D0%B6%D0%B0%D0%BA%D0%BE%D0%B1%D0%BE&action=edit&redlink=1)  от 13 марта 1736 года. В этом письме Эйлер приводит правило, пользуясь которым, легко определить, можно ли пройти по всем мостам, не проходя дважды ни по одному из них. В данном случае ответ был: «нельзя». В письме [Карлу Готлибу Элеру](https://ru.wikipedia.org/w/index.php?title=%D0%AD%D0%BB%D0%B5%D1%80,_%D0%9A%D0%B0%D1%80%D0%BB_%D0%93%D0%BE%D1%82%D1%82%D0%BB%D0%B8%D0%B1&action=edit&redlink=1)  от 3 апреля 1736 года Эйлер обосновывает найденное им правило, а позднее на эту тему Эйлер публикует статью в научном журнале [Петербургской академии наук](https://ru.wikipedia.org/wiki/%D0%9F%D0%B5%D1%82%D0%B5%D1%80%D0%B1%D1%83%D1%80%D0%B3%D1%81%D0%BA%D0%B0%D1%8F_%D0%B0%D0%BA%D0%B0%D0%B4%D0%B5%D0%BC%D0%B8%D1%8F_%D0%BD%D0%B0%D1%83%D0%BA) «Commentarii Academiae Scientiarum Imperialis Petropolitanae».

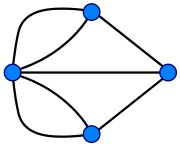
Эта статья была единственной в течении около 100 лет. Интерес к этой науке возродился около середины 19 века в связи с развитием естественных наук (исследование электрических сетей, моделей кристаллов, структур молекул). Оказалось, что многие математические головоломки могут быть сформулированы в терминах теории графов.

На упрощённой схеме города ([графе](https://ru.wikipedia.org/wiki/%D0%93%D1%80%D0%B0%D1%84_(%D0%BC%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0))) мостам соответствуют линии (ребра графа), а частям города — точки соединения линий (вершины графа). В ходе рассуждений Эйлер пришёл к следующим выводам:

* Число нечётных вершин (вершин, к которым ведёт нечётное число рёбер) графа должно быть чётно. Не может существовать граф, который имел бы нечётное число нечётных вершин.
* Если все вершины графа чётные, то можно начертить этот граф без отрыва карандаша от бумаги, при этом можно начинать с любой вершины графа и завершить его в той же вершине.
* Если ровно две вершины графа нечётные, то можно начертить этот граф без отрыва карандаша от бумаги, при этом нужно начинать с одной из нечётных вершин и завершить его в другой нечётной вершине.
* Граф с более чем двумя нечётными вершинами невозможно начертить одним росчерком.

Граф кёнигсбергских мостов имел четыре нечётные вершины (то есть все) — следовательно, невозможно пройти по всем мостам, не проходя ни по одному из них дважды.



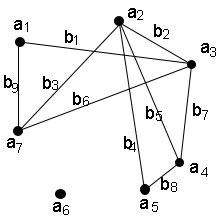


**Теория графов** - один из обширнейших разделов дискретной математики, широко применяется в решении экономических и управленческих задач, в программировании, химии, конструировании и изучении электрических цепей, коммуникации, психологии, социологии, лингвистике, других областях знаний.

Теория потоков в сетях возникла из рассмотрения реальных задач, таких как перевозка грузов по системе железных дорог, перекачка нефти по трубопроводам, управление запасами и т. д.

Основы теории графов — факты, понятия и утверждения, знать которые должен любой человек, занимающийся современным программированием или дискретной математикой.

**Графы строят** для того, чтобы отобразить отношения на [множествах](https://function-x.ru/sets1.html). Пусть, например, множество *A* = {*a*1, *a*2, ... *a*n} - множество людей, а каждый элемент будет отображён в виде точки. Множество *B* = {*b*1, *b*2, ... *b*m} - множество связок (прямых, дуг, отрезков - пока не важно). На множестве *A* задано отношение знакомства между людьми из этого множества. **Строим граф** из точек и связок. Связки будут связывать пары людей, знакомых между собой. Естественно, число знакомых у одних людей может отличаться от числа знакомых у других людей, а некоторые вполне могут и не быть ни с кем знакомы (такие элементы будут точками, не соединёнными ни с одной другой). Вот и получился граф!



"точки" называют вершинами графа, а "связки" - рёбрами графа.

Теория графов не учитывает конкретную природу множеств A и B. Существует большое количество самых разных конкретных задач, при решении которых можно временно забыть о специфическом содержании множеств и их элементов. Эта специфика никак не сказывается на ходе решения задачи, независимо от её трудности! Например, при решении вопроса о том, можно ли из точки a добраться до точки e, двигаясь только по соединяющим точки линиям, неважно, имеем ли мы дело с людьми, городами, числами и т.д. Но, когда задача решена, мы получаем решение, верное для любого содержания, которое было смоделировано в виде графа. Не удивительно поэтому, что теория графов - один из самых востребованных инструментов при создании искусственного интеллекта: ведь искусственный интеллект может обсудить с собеседником и вопросы любви, и вопросы музыки или спорта, и вопросы решения различных задач, причем делает это без всякого перехода (переключения), без которого в подобных случаях не обойтись человеку.

**Определение 1. Графом называется** система объектов произвольной природы (вершин) и связок (рёбер), соединяющих некоторые пары этих объектов.

**Определение 2.**Пусть V – (непустое) множество вершин, элементы v∈V – вершины. Граф G = G(V) с множеством вершин V есть некоторое cемейство пар вида: e = (a, b), где a,b∈V, указывающих, какие вершины остаются соединёнными. Каждая пара e = (a, b) - ребро графа. Множество U - множество рёбер e графа. Вершины a и b – концевые точки ребра e.

**Графы как структура данных.**Широким применением теории графов в компьютерных науках и информационных технологиях обусловлено добавлением к вышеизложенным определениям понятия графа как структуры данных. В компьютерных науках и информационных технологиях граф определяется как нелинейная структура данных. Что же тогда - линейная структура данных и чем от них отличаются графы? Линейные структуры данных характеризуются тем, что связывают элементы отношениями типа "простого соседства". Линейными структурами данных являются, например, массивы, таблицы, списки, очереди, строки. В противоположность им нелинейные структуры данных - такие, в которых элементы располагаются на различных уровнях иерархии и подразделяются на три вида: исходные, порождённые и подобные. Итак, граф - нелинейная структура данных.

Слово граф греческого происхождения, от слов "пишу", "описываю». То есть, любой граф описывает отношения. И наоборот: любое отношение можно описать в виде графа.

Одно из центральных понятий теории графов, опираясь на которое строятся другие понятия - понятие инцидентности. Друг другу инцидентны две вершины графа, если они соединены ребром; вершина и ребро графа инцидентны, если вершина является началом или концом ребра.

В **неориентированном** графе

*Маршрут* — чередующаяся последовательность v0,e1,v1,e2,…,eℓ,vℓ (1)  вершин и рёбер, в которой ei={vi−1,vi}

Вершины v0 и vℓ называются *крайними*, а все остальные — *промежуточными* (или *внутренними*). Маршрут, содержащий вершины v0 и vℓ в качестве крайних, называется (v0,vℓ)*-маршрутом*.

Если в графе нет кратных рёбер, то маршрут можно однозначно задать последовательностью вершин.

***Цепь* — маршрут, все рёбра которого попарно различны.**

*Простая цепь* — цепь, все вершины которой, кроме, возможно, крайних, попарно различны.

*Циклический маршрут* — маршрут, крайние вершины которого совпадают.

*Цикл* (или *циклическая цепь*) — циклический маршрут, являющийся цепью.

*Простой цикл* — простая циклическая цепь.

*Гамильтонов цикл* — простой цикл, содержащий все вершины графа.

*Эйлеров цикл* — цикл, содержащий все рёбра графа.

**В ориентированном графе**

*Ориентированный маршрут* (или просто *маршрут*) — последовательность вида (1) для ориентированного графа, в которой ei=(vi−1,vi). Понятия цепи, циклического маршрута и цикла переносятся на случай ориентированного графа без изменений.

*Путь* — ориентированный маршрут, все вершины которого, кроме, возможно, крайних, различны.

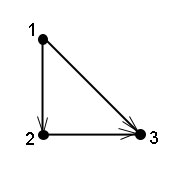
*Контур* — циклический путь.

Если в орграфе существует (u,v)-маршрут, то говорят, что вершина v *достижима из вершины*u. Любая вершина считается достижимой из самой себя.

**Пример 1.**Пусть A - множество чисел 1, 2, 3: A = {1, 2, 3}. Построить граф для отображения отношения "<" ("меньше") на этом множестве.

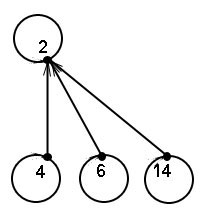
Решение. Очевидно, что числа 1, 2, 3 следует представить в виде вершин графа. Тогда каждую пару вершин должно соединять одно ребро. **Неориентированные** графы - такие, рёбра которых не имели направления. Или, как говорят ещё чаще, порядок двух концов ребра не существенен. В самом деле, граф, построенный в самом начале этого урока и отображавший отношение знакомства между людьми, не нуждается в направлениях рёбер, так как можно утверждать, что "человек номер 1" знаком с "человеком номер 2" в той же мере, как и "человек номер 2" с "человеком номер 1". В нашем же нынешнем примере одно число меньше другого, но не наоборот. Поэтому соответствующее ребро графа должно иметь направление, показывающее, какое всё же число меньше другого. То есть, порядок концов ребра существенен. Такой граф (с рёбрами, имеющими направление) называется **ориентированным** графом или орграфом.

Итак, в нашем множестве A число 1 меньше числа 2 и числа 3, а число 2 меньше числа 3. Этот факт отображаем рёбрами, имеющими направление, что показывается стрелками. Получаем следующий граф:



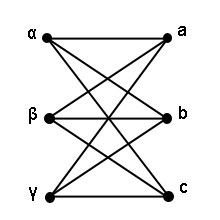
**Пример 2.**Пусть *A* - множество чисел  2, 4, 6, 14: *A* = {2, 4, 6, 14}. Построить граф для отображения отношения "делится нацело на" на этом множестве.

Решение. В этом примере часть рёбер будут иметь направление, а некоторые не будут, то есть строим [**смешанный граф**](https://function-x.ru/graphs2_definitions_classes.html). Перечислим отношения на множестве: 4 делится нацело на 2, 6 делится нацело на 2, 14 делится нацело на 2, и ещё каждое число из этого множества делится нацело на само себя. Это отношение, то есть когда число делится нацело на само себя, будем отображать в виде рёбер, которые соединяют вершину саму с собой. Такие рёбра называются [**петлями**](https://function-x.ru/graphs5.html). В данном случае нет необходимости давать направление петле. Таким образом, в нашем примере три обычных направленных ребра и четыре петли. Получаем следующий граф:



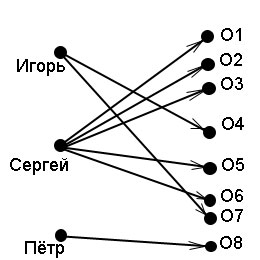
**Пример 3.**Пусть даны множества *A* = {α, β, γ} и *B* = {a, b, c}. Построить граф для отображения отношения "декартово произведение множеств".

Решение. Как известно из определения [**декартова произведения множеств**](https://function-x.ru/sets1.html#paragraph4), в нём нет упорядоченных наборов из элементов одного и того же множества. То есть в нашем примере нельзя соединять греческие буквы с греческими и латинские с латинскими. Этот факт отображается в виде [**двудольного графа**](https://function-x.ru/graphs2_definitions_classes.html), то есть такого, в котором вершины разделены на две части так, что вершины, принадлежащие одной и той же части, не соединены между собой. Получаем следующий граф:



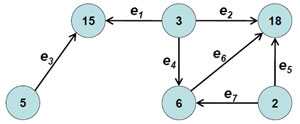
**Пример 4.**В агентстве по недвижимости работают менеджеры Игорь, Сергей и Пётр. Обслуживаются объекты О1, О2, О3, О4, О5, О6, О7, О8. Построить граф для отображения отношений "Игорь работает с объектами О4, О7", "Сергей работает с объектами О1, О2, О3, О5, О6", "Пётр работает с объектом О8".

Решение. Граф, отображающий данные отношения, будет так же двудольным, так как менеджер не работает с менеджером и объект не работает с объектом. Однако, в отличии от предыдущего примера, граф будет ориентированным. В самом деле, например, Игорь работает с объектом О4, но не объект О4 работает с Игорем. Часто, когда такое свойство отношений очевидно, необходимость давать рёбрам направления может показаться "математической тупостью". Но всё же, и это вытекает из строгого характера математики, если отношение носит односторонний характер, то давать направления рёбрам нужно. В приложениях отношений эта строгость окупается, например, в программах, предназначенных для планирования, где тоже применяются графы и маршрут по вершинам и рёбрам должен проходить строго в заданном направлении. Итак, получаем следующий ориентированный двудольный граф:



**Пример 5.**Пусть задано множество *C* = {2, 3, 5, 6, 15, 18}. Построить граф, реализующий отношение, определяющее все пары чисел *a* и *b* из множества *C*, у которых при делении второго элемента на первый получаем частное, которое является целым числом больше 1.

Решение. Граф, отображающий данные отношения, будет ориентированным, так как в условии есть упоминание о втором и первом элементе, то есть, ребро будет направлено от первого элемента ко второму. Из этого однозначно понятно, какой элемент является перым, а какой вторым. Ещё добавим терминологии: ориентированные рёбра принято называть дугами. В нашем графе будет 7 дуг: *e*1 = (3, 15), *e*2 = (3, 18), *e*3 = (5, 15), *e*4 = (3, 6), *e*5 = (2, 18), *e*6 = (6, 18), *e*7 = (2, 6). В этом примере рёбра (дуги) графа просто пронумерованы, но порядковые номера - не единственное, что можно приписать дуге. Дуге можно приписать также весы означающие, например, стоимость пересылки груза из одного пункта в другой. Итак, получаем следующий ориентированный граф:



Как мы уже знаем из теоретической вступительной части, теория графов не учитывает специфическую природу множеств и с помощью одного и того же графа можно задать отношения на множествах с самым разным содержанием. То есть, от этого самого содержания при моделировании задачи можно абстрагироваться. Перейдём к примерам, иллюстрирующим это замечательное свойство теории графов.

**Пример 6.**На кусочке шахматной доски размером 3 Х 3 размещены два белых коня и два чёрных коня так, как показано на рисунке ниже.



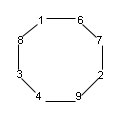
Можно ли переместить коней в состояние, которое изображено на следующем рисунке, не забывая, что две фигуры не могут находиться на одной клетке?



Решение. В конструируемом графе пары вершин будут связаны отношением "ход коня". То есть, одна вершина - та, из которой конь ушёл, а другая - та, в которую пришёл, а промежуточная клетка буквы "г" будет за пределами этого отношения. Получаем следующий граф:



И всё же конструкция получилась громозкой. В ней видны клетки шахматной доски, а многие рёбра графа пересекаются. Нельзя ли абстрагироваться от физического вида шахматной доски и вообразить отношения проще? Оказывается, можно. В новом графе соседними вершинами будут те, которые связаны отношением "ход коня", а не соседние по шахматной доске (рисунок ниже).

Теперь легко увидеть, что ответ на вопрос этой задачи - отрицательный. В начальном состоянии между двумя белыми конями нет чёрного коня, а в конечном состоянии этот чёрный конь должен быть. Рёбра графа размещены так, что два находящихся рядом коня не могут перепрыгнуть друг через друга.