

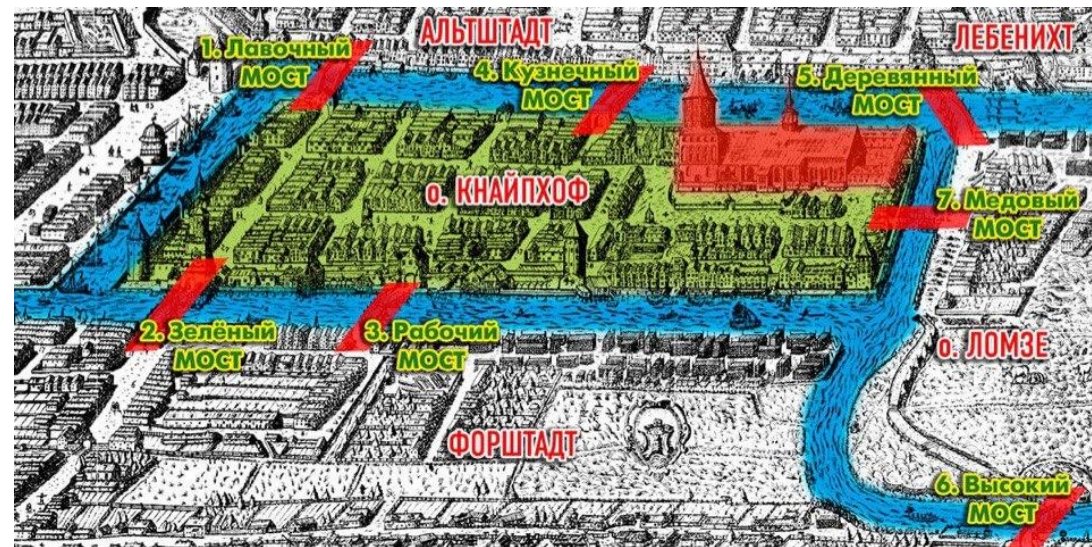
Обходы в графах

Эйлеровы и гамильтоновы графы

Эйлеровы графы

Задача о семи кёнигсбергских мостах

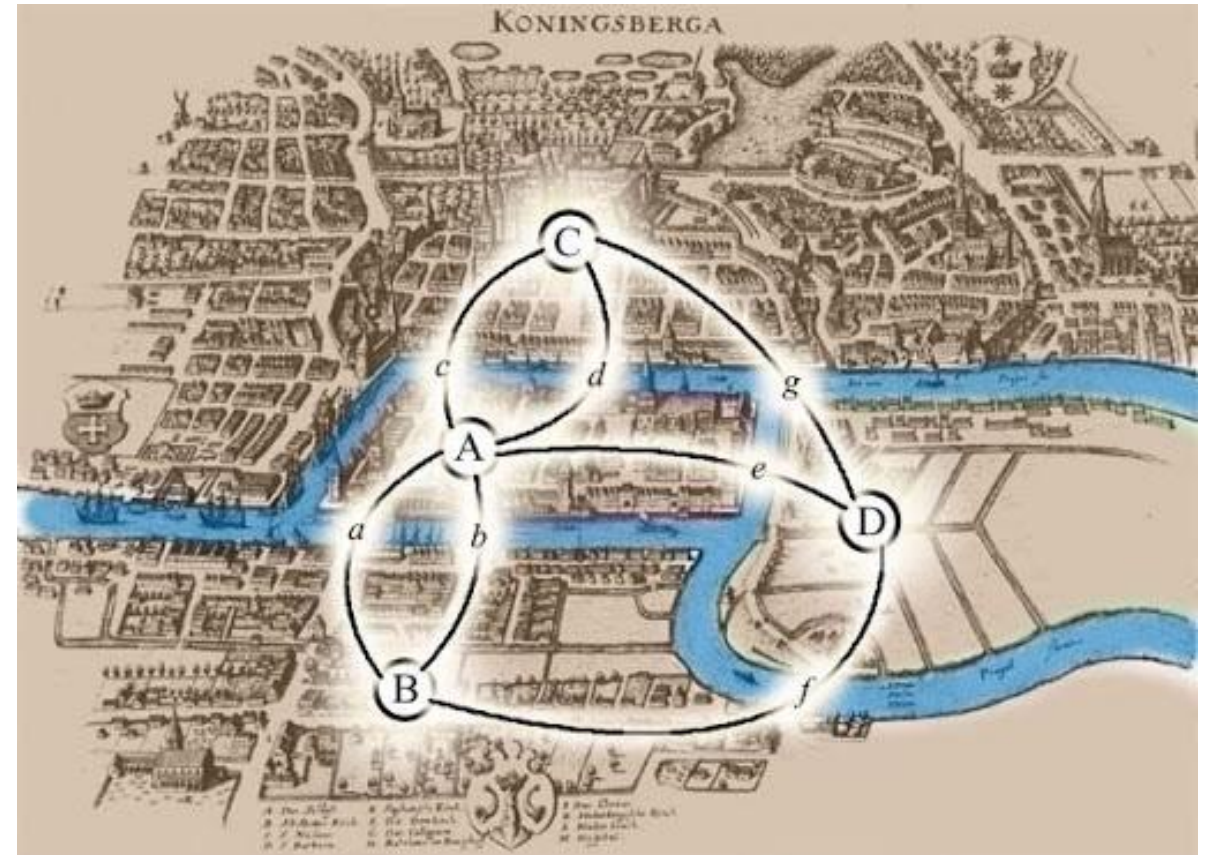
Проблема семи мостов Кёнигсберга или Задача о кёнигсбергских мостах — старинная математическая задача, в которой спрашивалось, как можно пройти по всем семи мостам Кёнигсберга, не проходя ни по одному из них дважды. Впервые была решена в 1736 году немецким и русским математиком Леонардом Эйлером.



Издавна среди жителей Кёнигсберга была распространена такая загадка: как пройти по всем мостам (через реку Преголя), не проходя ни по одному из них дважды. Многие кёнигсбержцы пытались решить эту задачу как теоретически, так и практически, во время прогулок. Впрочем, доказать или опровергнуть возможность существования такого маршрута никто не мог.



В письме итальянскому математику и инженеру Мариони от 13 марта 1736 года Эйлер пишет о том, что он смог найти правило, пользуясь которым, легко определить, можно ли пройти по всем мостам, не проходя дважды ни по одному из них.

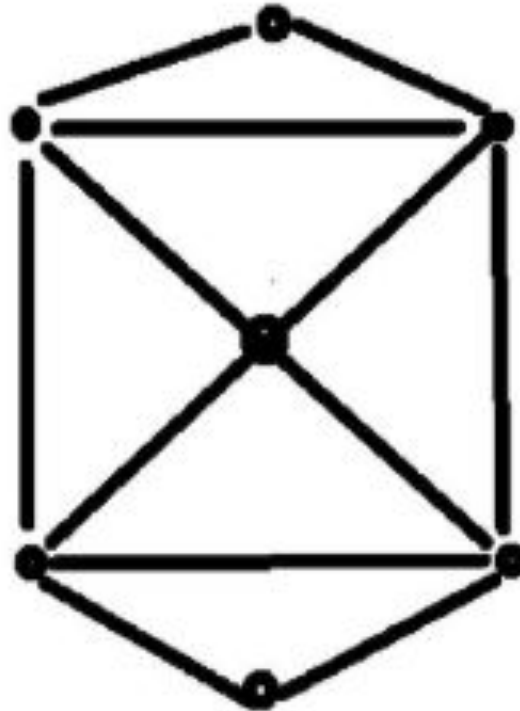


Решение задачи по Эйлеру

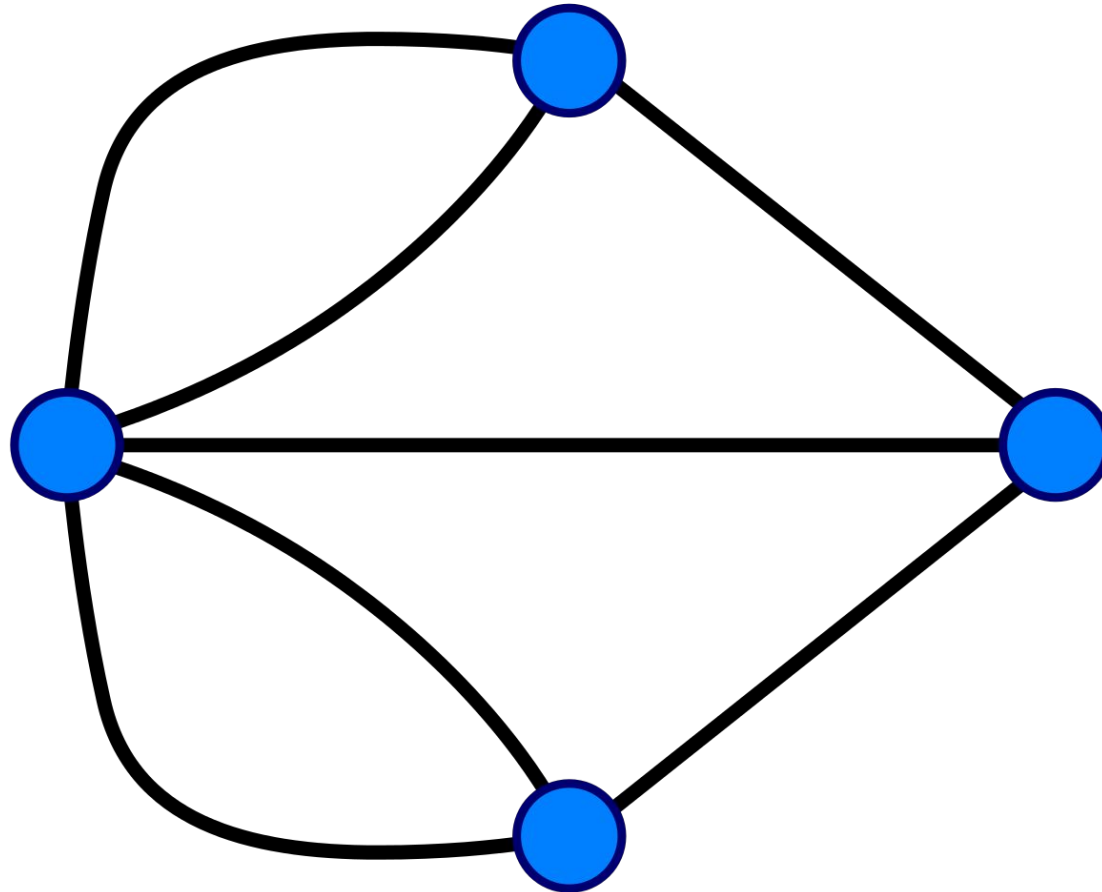
На упрощённой схеме части города (графе) мостам соответствуют линии (дуги графа), а частям города — точки соединения линий (вершины графа). В ходе рассуждений Эйлер пришёл к следующим выводам:

1. 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.

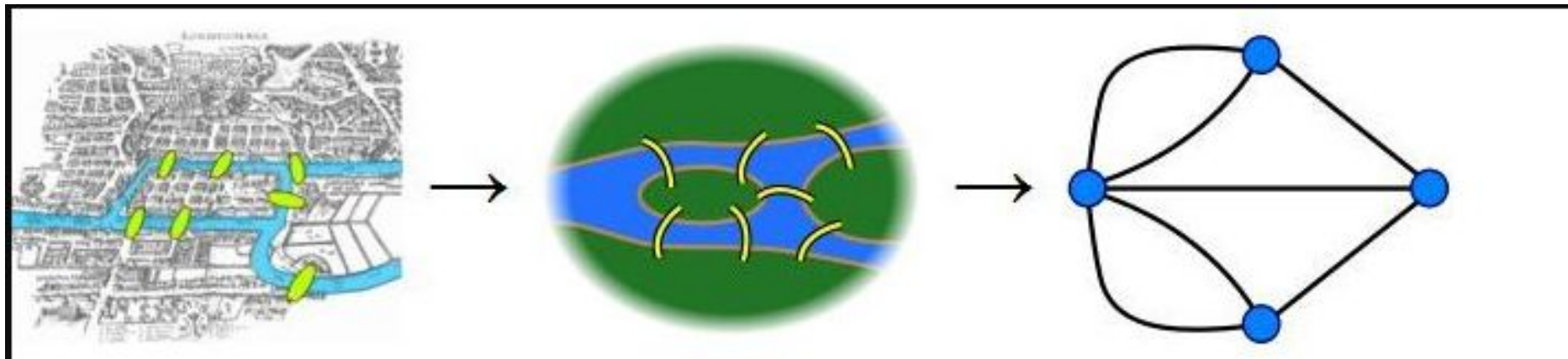
2. Если все вершины графа чётные, то можно, не отрывая карандаша от бумаги, начертить граф, при этом можно начинать с любой вершины графа и завершить его в той же вершине.



3. Граф с более чем двумя нечётными вершинами невозможно начертить одним росчерком.

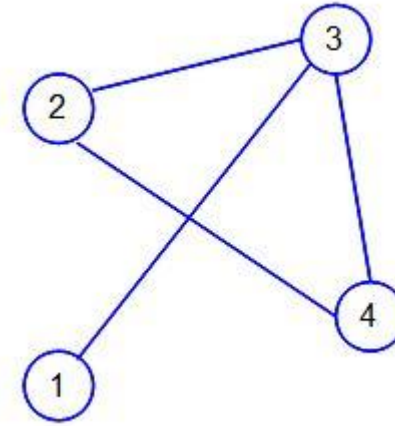


4. Граф кёнигсбергских мостов имел четыре (синим) нечётные вершины (то есть все), следовательно, невозможно пройти по всем мостам, не проходя ни по одному из них дважды

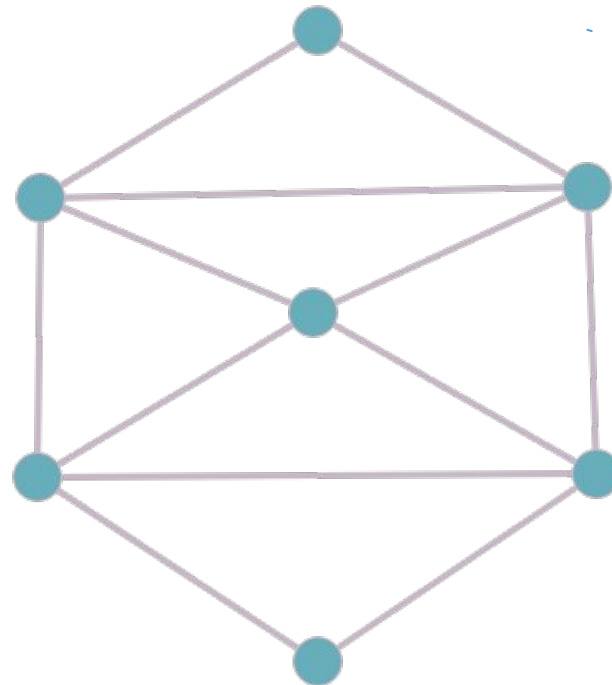


Эйлеровой цепью в графе называется цепь, содержащая все рёбра графа.

Эйлеровым циклом данного графа называется цикл, содержащий все рёбра этого графа.



Эйлерова цепь

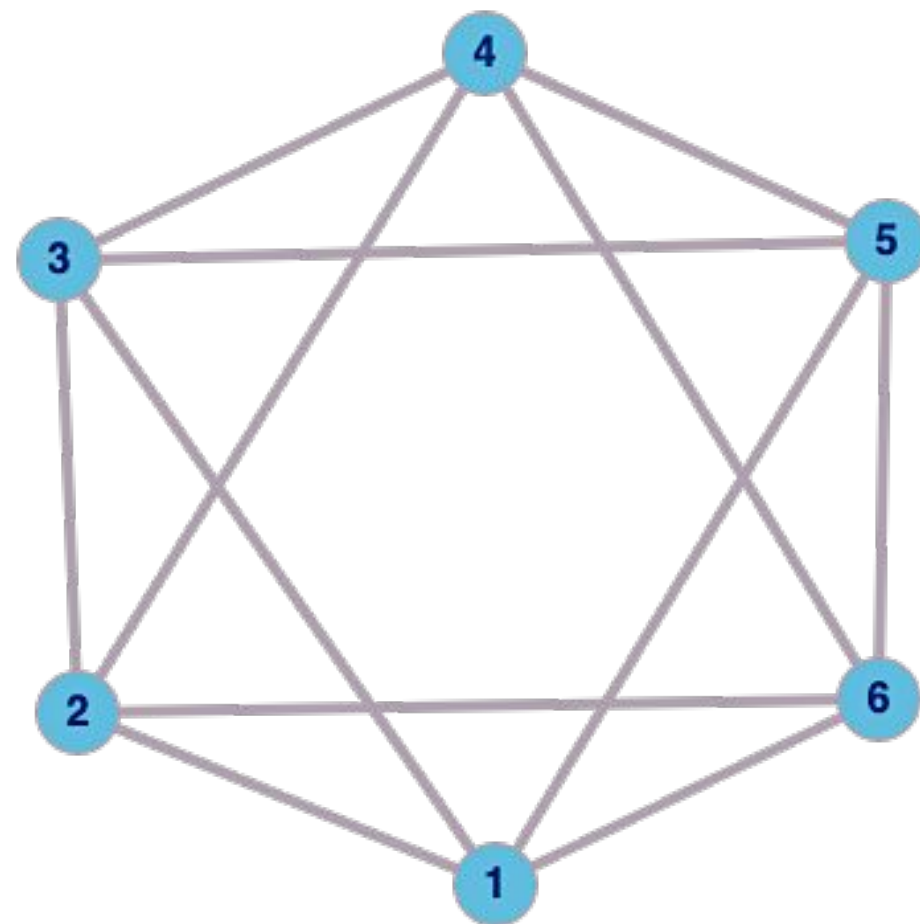
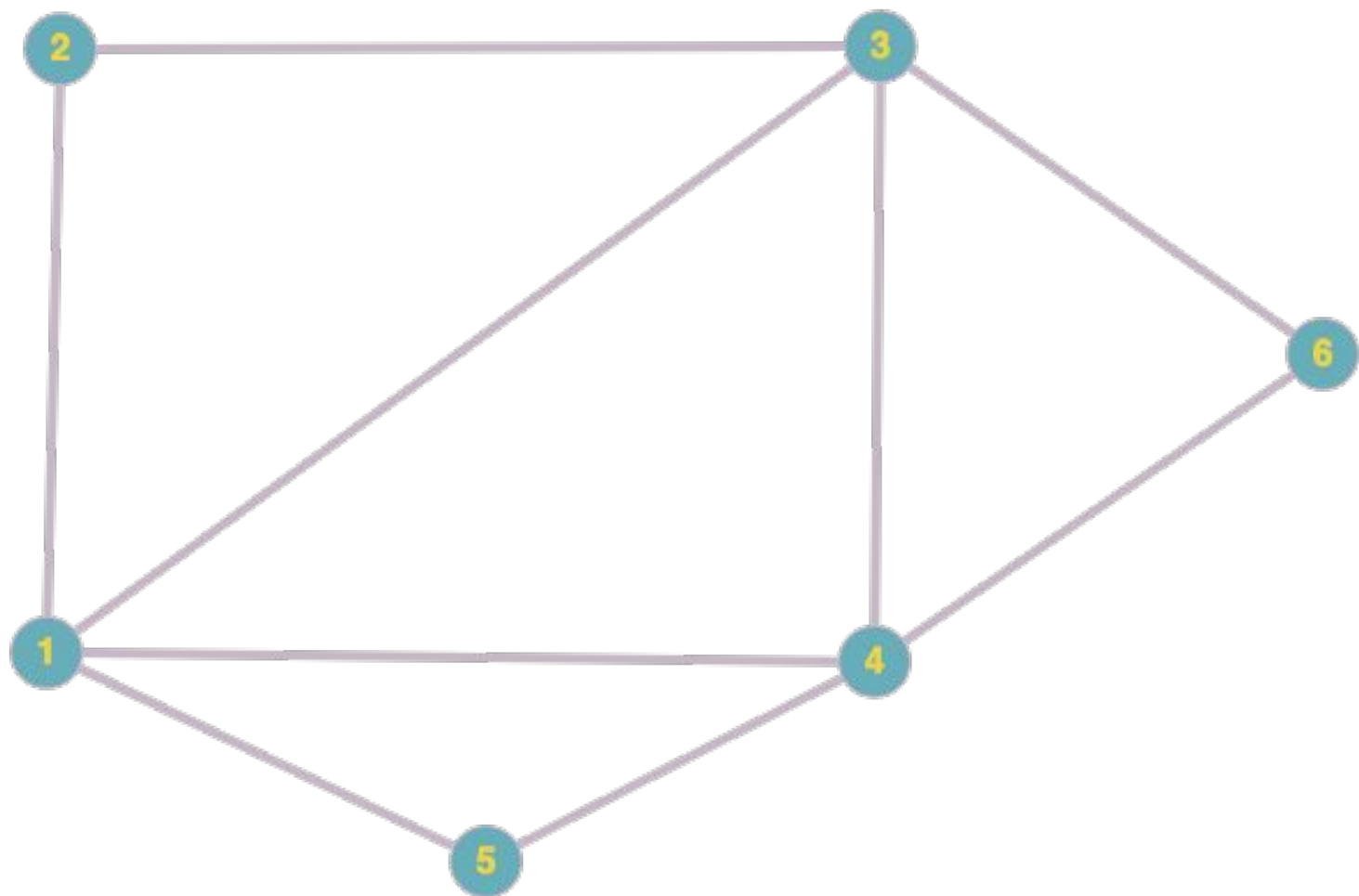


Эйлеров цикл

Эйлеровым графом называется связный граф, в котором есть эйлеров цикл.

Эйлеров граф можно нарисовать, не отрывая карандаша от бумаги и не повторяя линий.

Эйлеровы графы



Теорема Эйлера (1736 г.)

Теорема.

Связный граф является эйлеровым тогда и только тогда, когда степени всех его вершин чётны.

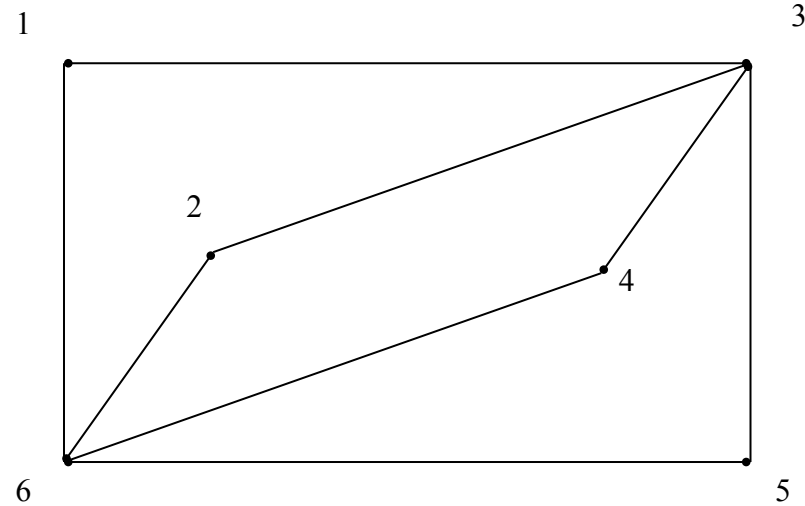
Теорема

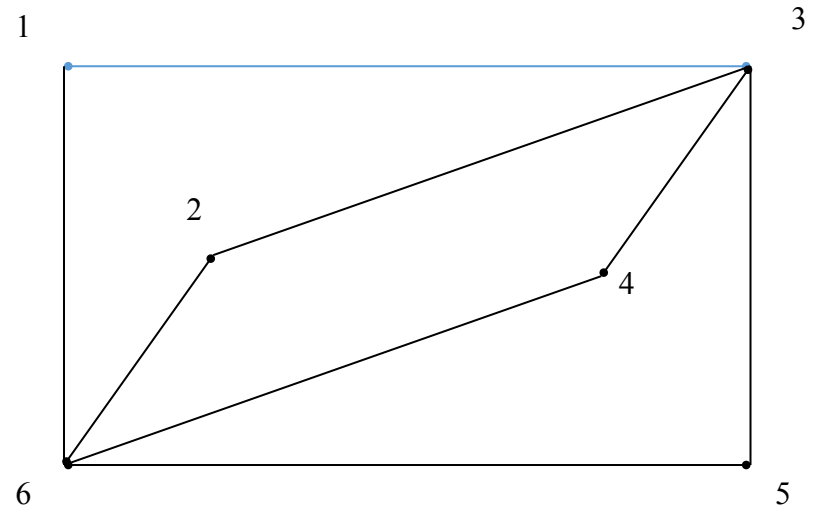
- Эйлеровых графов почти нет

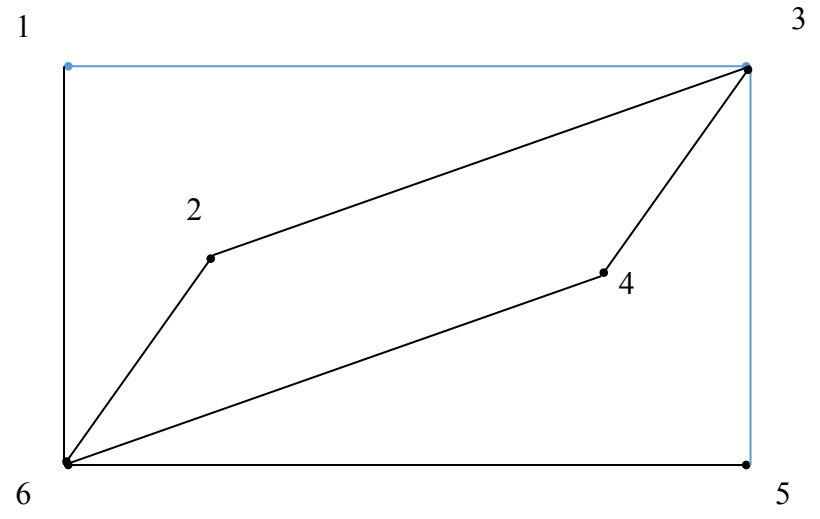
Алгоритм Флёрри.

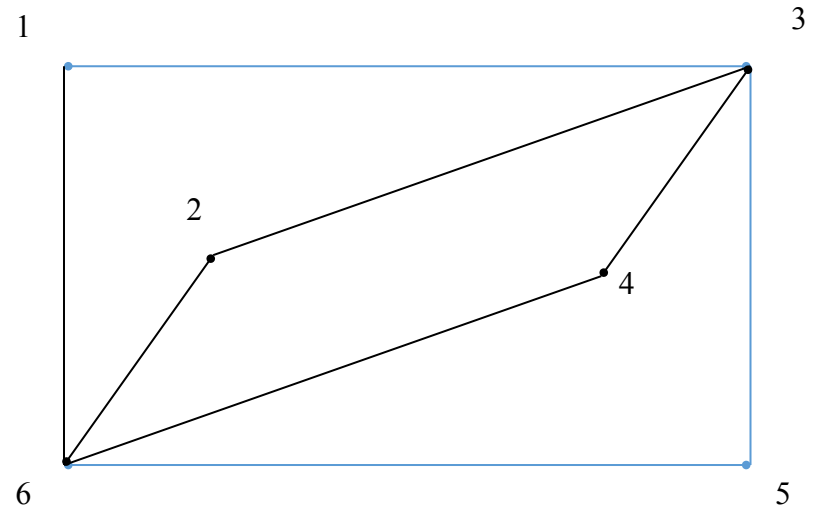
1. Выбираем произвольную вершину u_1 и по некоторому ребру u_1u_2 переходим в вершину u_2 , ребру u_1u_2 присваиваем номер 1.
2. Пусть u_{i+1} – вершина, в которую перешли в результате выполнения предыдущего шага. Выбираем любое ребро $u_{i+1}u_{i+2}$, инцидентное вершине u_{i+1} , однако ребро, являющееся мостом для графа, образованного из рёбер без номеров, выбираем только в том случае, когда нет других возможностей, ребру $u_{i+1}u_{i+2}$ присваиваем номер $i+1$.
3. Процесс завершен, если все рёбра занумерованы.

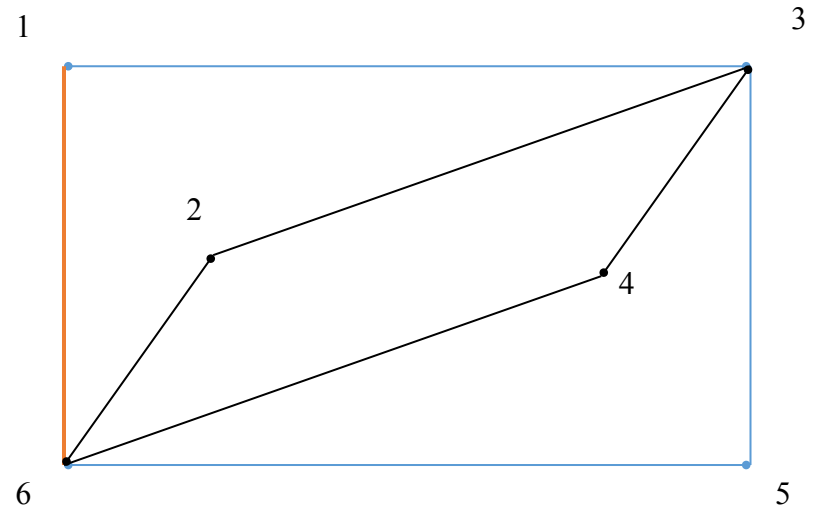
Эйлеров цикл – (1; 3), (3; 5), (5; 6), (6; 4), (4; 3),
(3; 2), (2; 6), (6; 1).

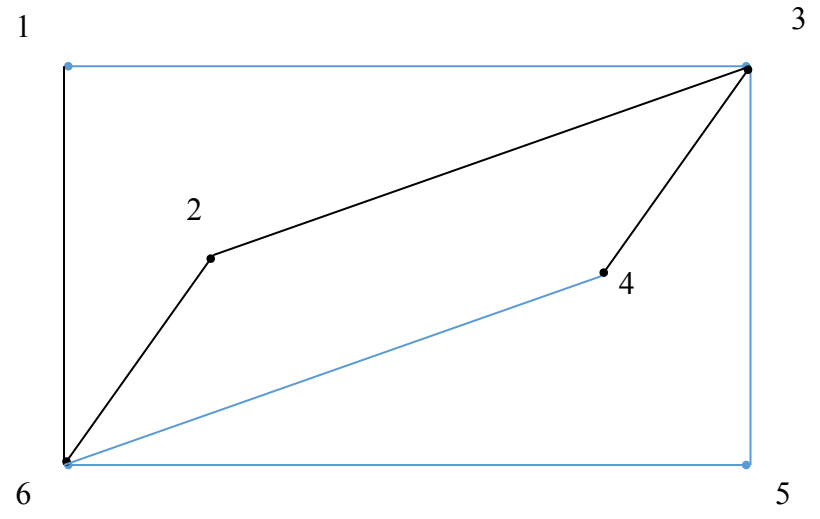


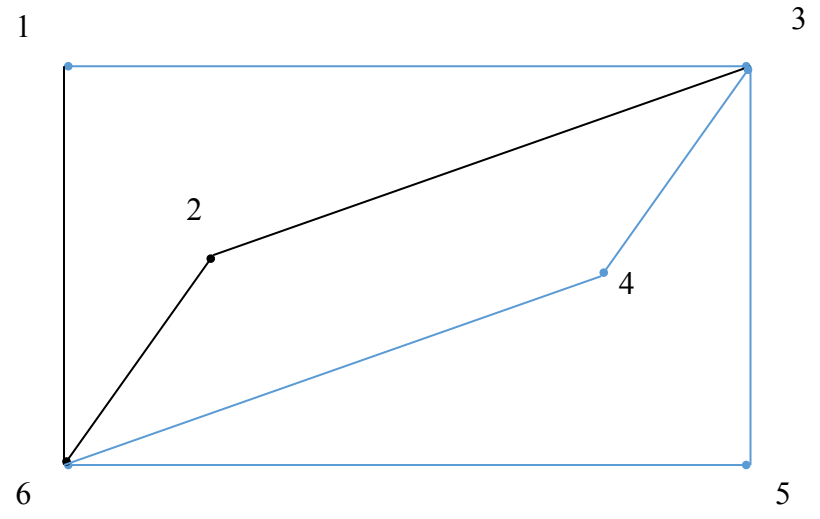


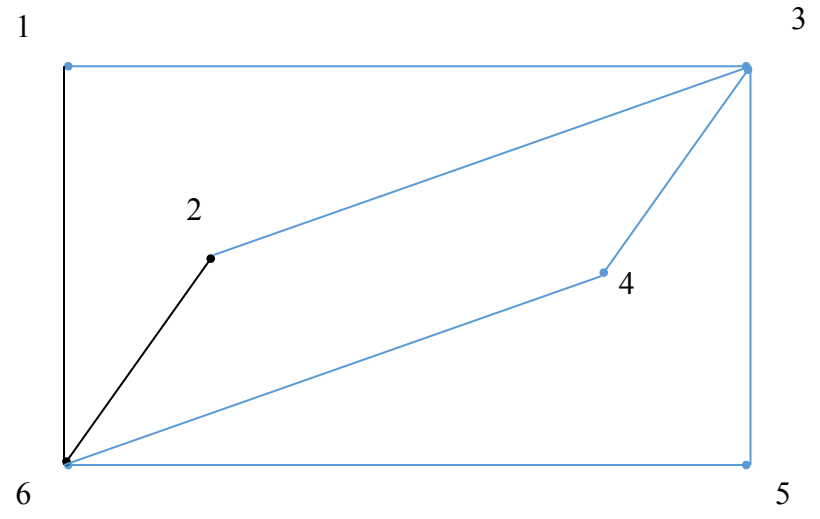


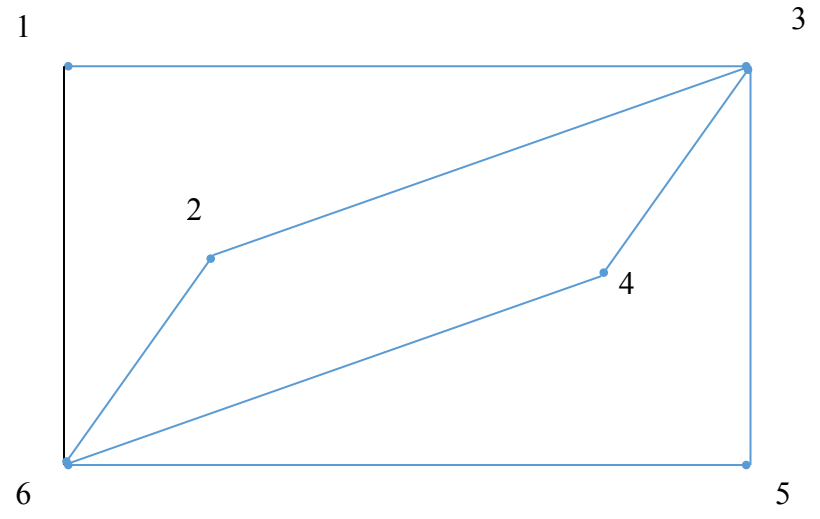


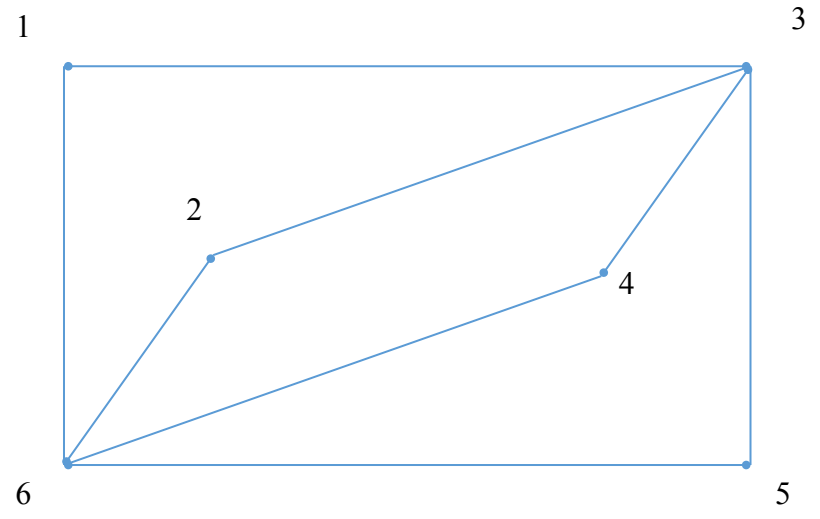












Следствие.

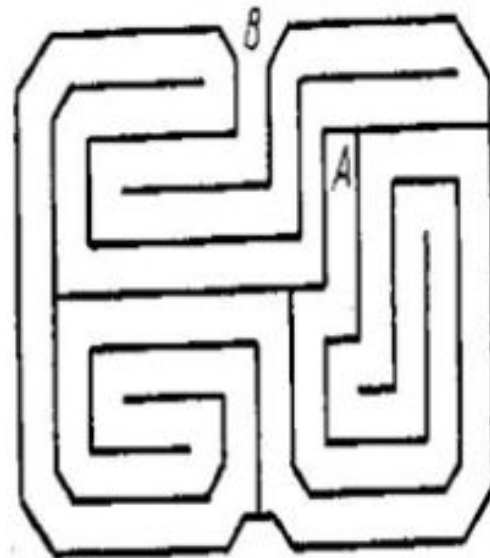
Если связный граф содержит ровно $2k$ вершин нечетной степени, то минимальное число покрывающих его реберно-непересекающихся цепей равно k .

В частности, когда $k = 1$, граф имеет цепь, которая соединяет две вершины нечетной степени и содержит все ребра графа, то есть имеет эйлерову цепь.

Лабиринты и графы

Способ обходить все ребра графа можно использовать, например, для отыскания пути выхода из лабиринта.

Лабиринты состоят из коридоров, перекрестков и тупиков. Их маршруты можно представить в виде графов, в которых рёбра соответствуют коридорам, а вершины – входам, выходам, перекрёсткам и тупикам.



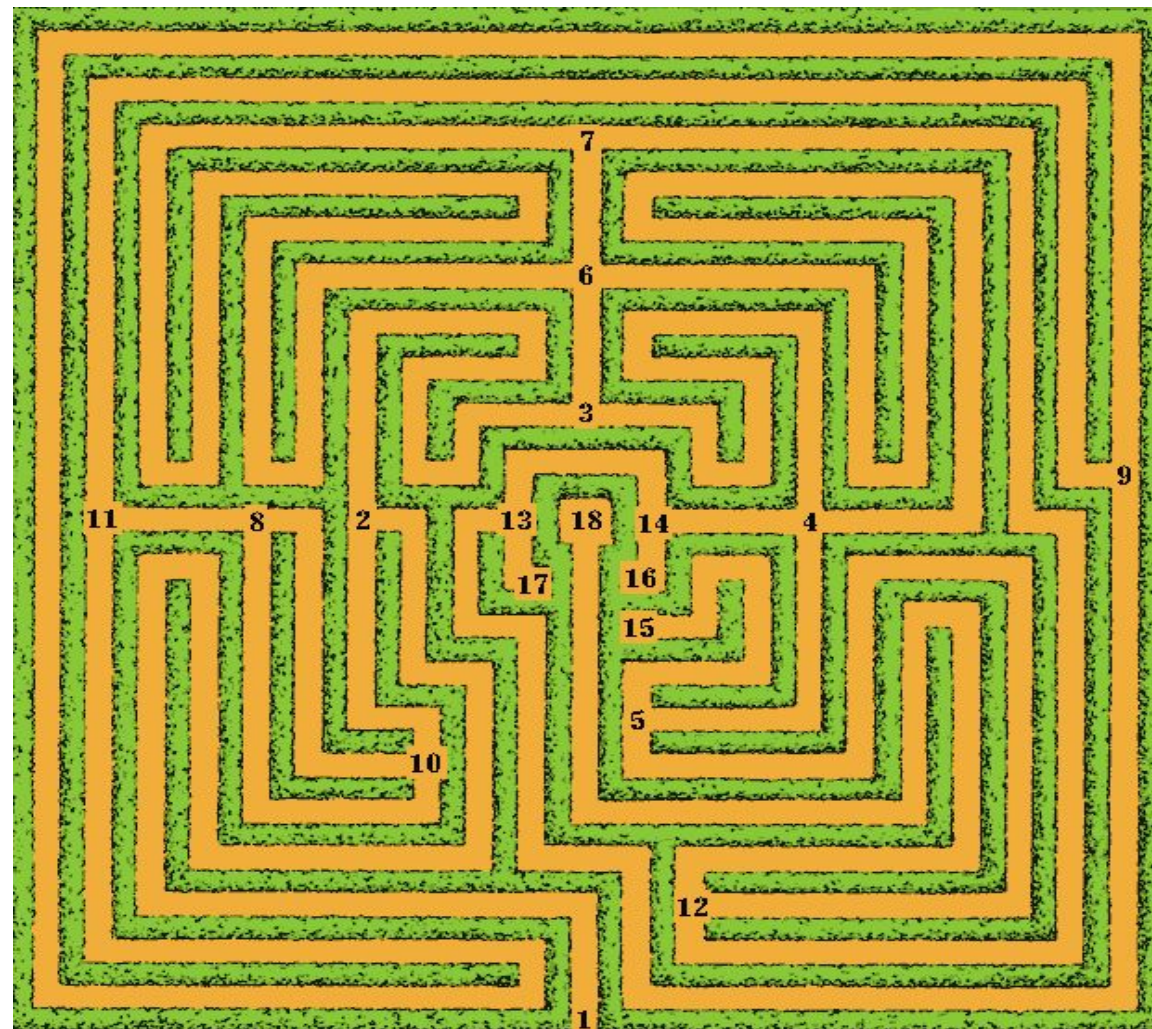
Происхождение задачи о лабиринтах относится к глубокой древности. Причем древние задачу о лабиринтах считали вообще неразрешимой. Человек, попавший в лабиринт, не мог уже из него выйти, если только какое-нибудь чудо или случай не приходили к нему на помощь.



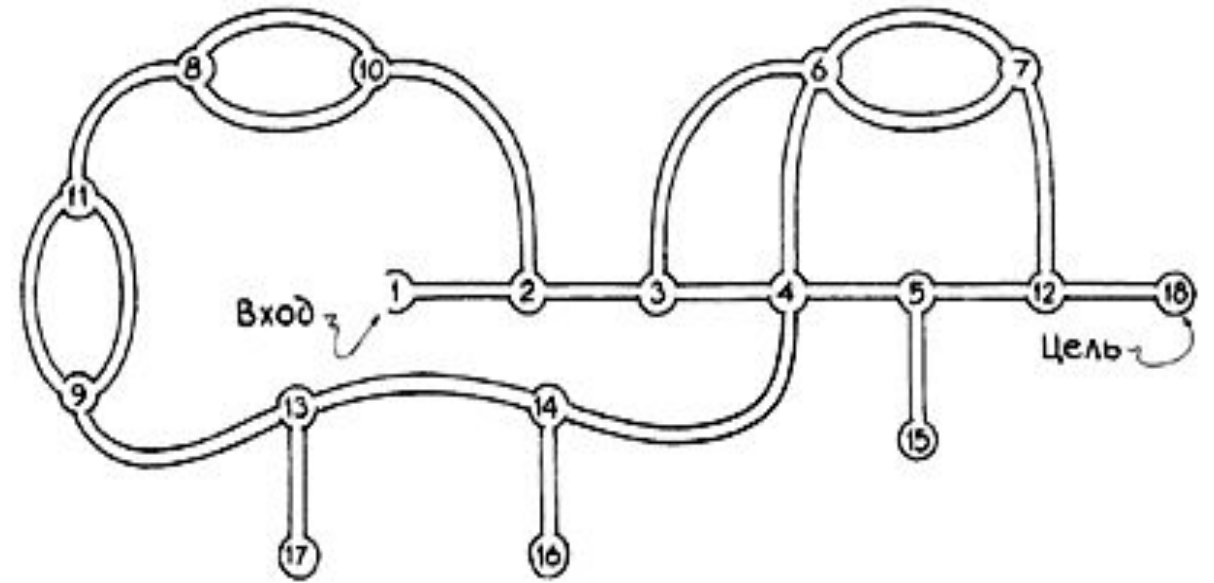
Один из методов состоит в том, чтобы в каждой узловой точке выбирать одно и то же направление. Например, можно всегда сворачивать на крайнюю правую ветвь. Если этот путь закончится тупиком, следует вернуться к узловой точке и выбрать следующую ветвь (если считать справа). Может оказаться, что в результате вы пройдёте по каждой ветви дважды — по одному разу в каждом направлении, но в конце концов вы доберётесь до цели. На обратном пути можно либо продолжать выбирать крайние правые ветви в каждом узле (и в этом случае вы, вероятно, пройдёте по новым областям лабиринта), либо каждый раз сворачивать на крайнюю левую ветвь (и тогда вы в точности повторите первоначальный маршрут). Метод выбора одной и той же — правой или левой — ветви называют соответственно *правилом правой или левой руки*.

Правило руки применимо только к так называемым *односвязным лабиринтам*. Этот термин означает, что лабиринт не содержит замкнутых маршрутов, т.е. таких, которые образуют замкнутую петлю. Замкнутый маршрут возникает в том случае, если существует ограниченный стенками «остров», который не соединяется с другими стенками лабиринта. Лабиринт с одним или более островами называется *многосвязным*.

Первый многосвязный садовый лабиринт был сооружён в 1820-е годы в Чевнинге в Великобритании. Он состоит из восьми сцепленных друг с другом островов.



С лабиринтом гораздо легче разобраться, если его схему топологически преобразовать к простой структуре, называемой сетью. В этой структуре все узлы сохранены, но ветви выпрямлены. На сети чевнингского лабиринта легко увидеть прямой маршрут 1–2–3–4–5–12–18, ведущий к цели. Подходящим является и другой маршрут, а именно 1–2–3–6–7–12–18, содержащий такое же количество узлов.



Сеть для лабиринта в Чевнинге

В терминах графов *задача о лабиринте* может быть сформулирована следующим образом. Определить *метод, позволяющий найти маршрут в графе, который начинается в заданной вершине a_0 и наверняка приводит в другую заданную вершину a_1 (выход)*. Очевидно, чтобы не было бесконечного блуждания по циклическим маршрутам, необходимо иметь возможность запоминать уже пройденные вершины и ребра; поэтому нужно предполагать, что заблудившийся располагает средствами пометить каким-то способом проходимые им ребра и вершины.



Первый систематический процесс для нахождения выхода из лабиринта, по-видимому, был предложен **Винером** (1873). Его правило таково. Из данной вершины a_0 следует проходить по ребрам графа возможно дальше, выбирая в каждой вершине еще не пройденное ребро. Из вершины, от которой дальше двигаться нельзя, маршрут возвращается до такой вершины, в которой есть еще не использованное ребро. Последняя операция тогда будет состоять в возвращении по всей линии к вершине a_0 . Ясно, что такой маршрут покроет все ребра. Однако он будет содержать много повторяющихся участков, и, чтобы их различить, требуется нечто вроде “нити Ариадны”.

Reignac-sur-Indre Maze (Франция)

Один из красивейших и крупнейших лабиринтов мира – Reignac-Sur-Indre во французской исторической провинции Турень – сооружен в 1996 году и занимает площадь более 20 га.

Ежедневно его посещают 85000 туристов, которые долго блуждают по его переходам. Этот лабиринт замечателен еще тем, что он каждый год разный. Благодаря особому методу посадки растений (кукуруза, подсолнухи) и разметке поверхности он «вырастает» каждую весну, в августе дает урожай и исчезает, а следующей весной появляется снова.



Longleat Hedge Maze (Англия)

Этот лабиринт состоит из 16000 деревьев английского тиса и является самым длинным в мире. Его сформировал в 1975 году дизайнер Грег Брайт (Greg Bright), площадь лабиринта 0.6 гектара (60 соток), длина всех ходов 2.7 километра.

В отличие от большинства подобных сооружений, этот лабиринт трехмерный, так как внутри него находятся шесть деревянных мостиков, с которых можно посмотреть и оценить маршрут. В центре лабиринта находится смотровая башня, которая является конечной его точкой, откуда можно еще раз рассмотреть в подробностях весь комплекс.



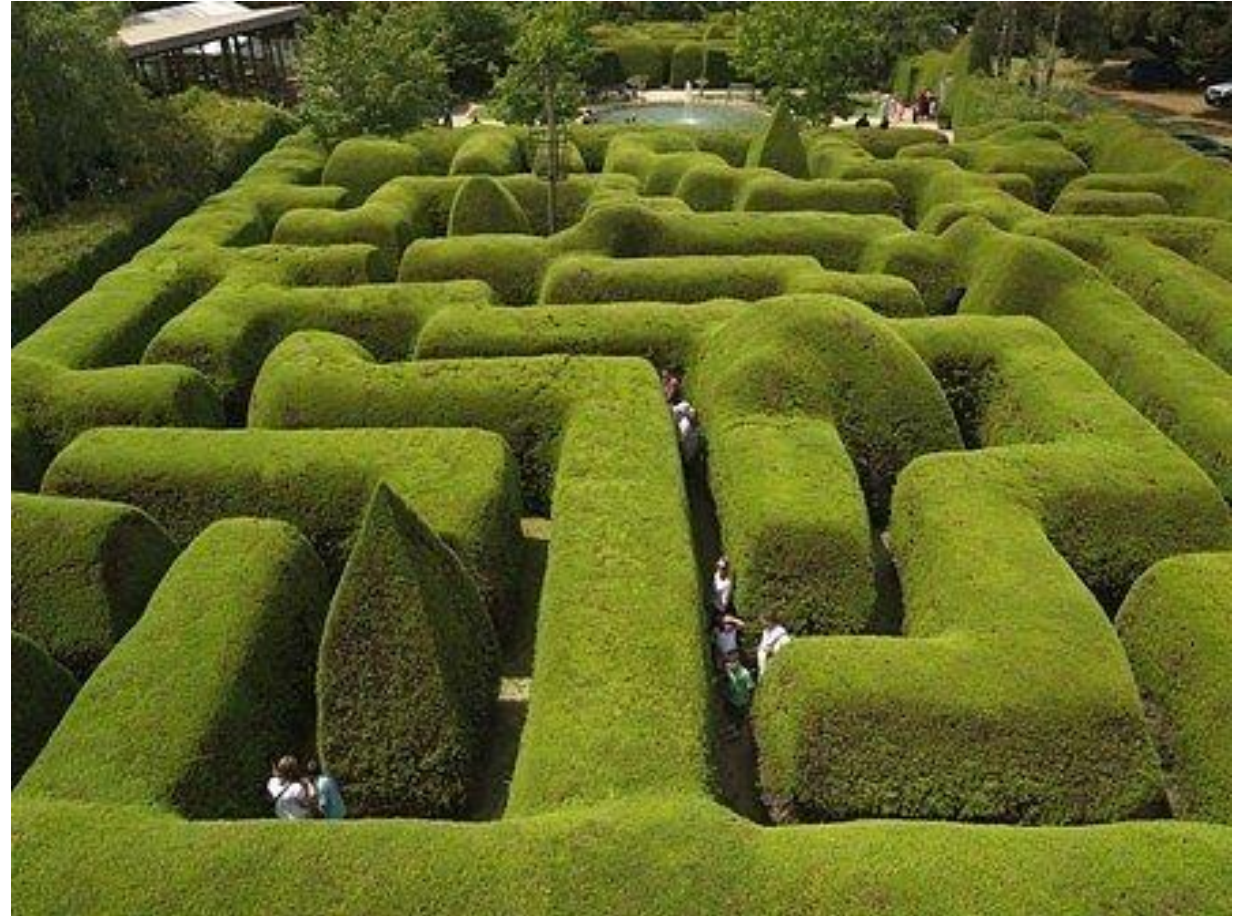
York Maze (Англия)



Состоит из полутора миллионов деревьев, покрывает площадь 32 акра, это примерно 15 футбольных полей. Разработан с использованием современных технологий спутникового позиционирования, что позволило добиться точности порядка десятков сантиметров. Создан неким Томом Pearcy в честь 40-летия картины Star Trek.

Ashcombe Maze (Австралия)

Он не может похвастаться огромным размером или длиной переходов, но это самый старый и известный классический лабиринт — живая изгородь, расположенный в городе Shoreham на востоке Mornington Peninsula, австралийского штата Виктория. Высота «стен» три метра, толщина два метра. Он интересен еще и тем, что здесь есть розовый лабиринт, где цветут 217 разновидностей роз на 1200 кустах. Интересно посмотреть, как он выглядит во время цветения...





Pineapple Garden Maze (Гавайи)

Очень большой и красочный лабиринт Ананасовый Сад, содержащий более 5 км запутанных ходов на площади 1.2 гектара. В его составе более 14000 экзотических гавайских деревьев, включая гибискус, кротон, панакс, ананас и геликонию. Находится этот лабиринт в Waimea Bay на ананасовых плантациях компании Dole на Гавайях, и в период цветения выглядит сверху очень впечатляюще. А тех, кто достигнет центра лабиринта, ждет фреш и ананасовый щербет.



Snake Maze (Англия)

62-летний садовник Майкл Бли (Michael Blee) потратил несколько месяцев, создавая этот Змеиный лабиринт на ферме Gore Farm неподалеку от Рочестера. Основной строительный элемент — трехметровые живые изгороди. Этот мистер Бли очень продуктивный садовник — это его десятый лабиринт, и самый сложный из всех им сформированных. Теперь Майкл надеется попасть в книгу рекордов Гиннеса если не по размеру, то по количеству созданных им зеленых лабиринтов.





Лабиринт Виллы Пизани (Италия)

Созданный в начале 18 века, Il Labirinto считается самым сложным лабиринтом среди всех существующих в мире. Он расположен в городе Stra неподалеку от Венеции на территории виллы Pisani. Роскошный сад был любимым лабиринтом Наполеона и местом встреч Муссолини и Гитлера. Представьте, как изменился бы ход исторических событий, если бы они заблудились на тропах лабиринта.



Peace Maze (Ирландия)

Ирландский лабиринт официально был открыт в 2001 году, его площадь 1.1 гектара. Длина всех дорожек составляет 3147 метра, весь лабиринт состоит из 6000 деревьев тиса, большинство из которых выращены в декабре 2000 года жителями Северной Ирландии — от этого произошло его название — «Мирный лабиринт»





Hampton Court Maze (Англия)

Расположен возле Темзы на западе Лондона на территории корта Hampton. Деревья, которые его составляют, входят в композицию сада, посаженного в период с 1689 по 1695 год Джорджем Лондоном и Генри Вайзом (George London, Henry Wise). Площадь его невелика по сравнению с ранее упомянутыми — «всего» 1350 квадратных метров, длина дорожек — 800 метров. Основное достоинство лабиринта — его известность среди британцев, о нем писал Джером К. Джером в своей новелле «Трое в лодке, не считая собаки». Неудивительно, что ежегодно лабиринт посещают сотни тысяч людей.

Davis” Mega Maze (США)

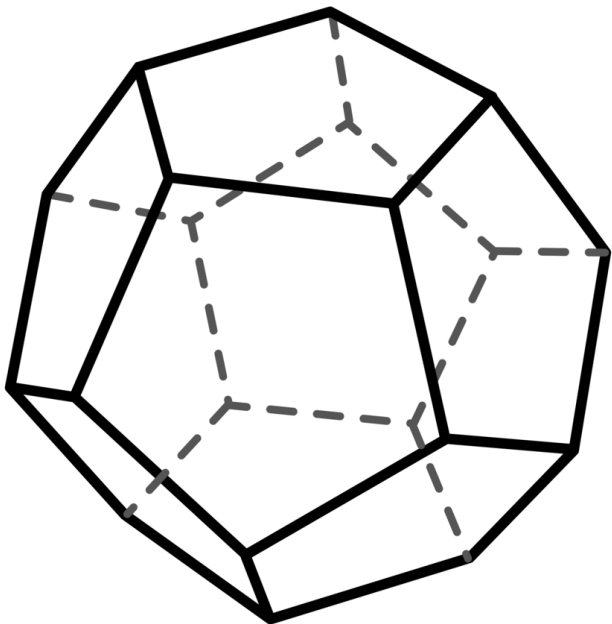
Этот лабиринт расположен в городе Стерлинг, штат Массачусетс, открыт в 1998 году. Он принадлежит фермерам в седьмом поколении, держащим владения под названием Davis” Farmland. Из года в год лабиринт меняется, изначально он был разработан в Англии фирмой Dorset дизайнером лабиринтов (есть и такие) Адрианом Фишером (Adrian Fisher), ежегодно на его поддержание в отличном виде тратится не более 12800 часов напряженной работы.



Правило Тарри (обхода лабиринта).

При обходе лабиринта, попадая в любой перекрёсток A впервые по некоторому пути a , при возвращении в этот перекрёсток A избегать пользоваться путём a до тех пор, пока это возможно, и лишь только в этом случае идти по пути a в обратную сторону, когда все остальные пути из A будут пройдены дважды.

Гамильтоновы графы

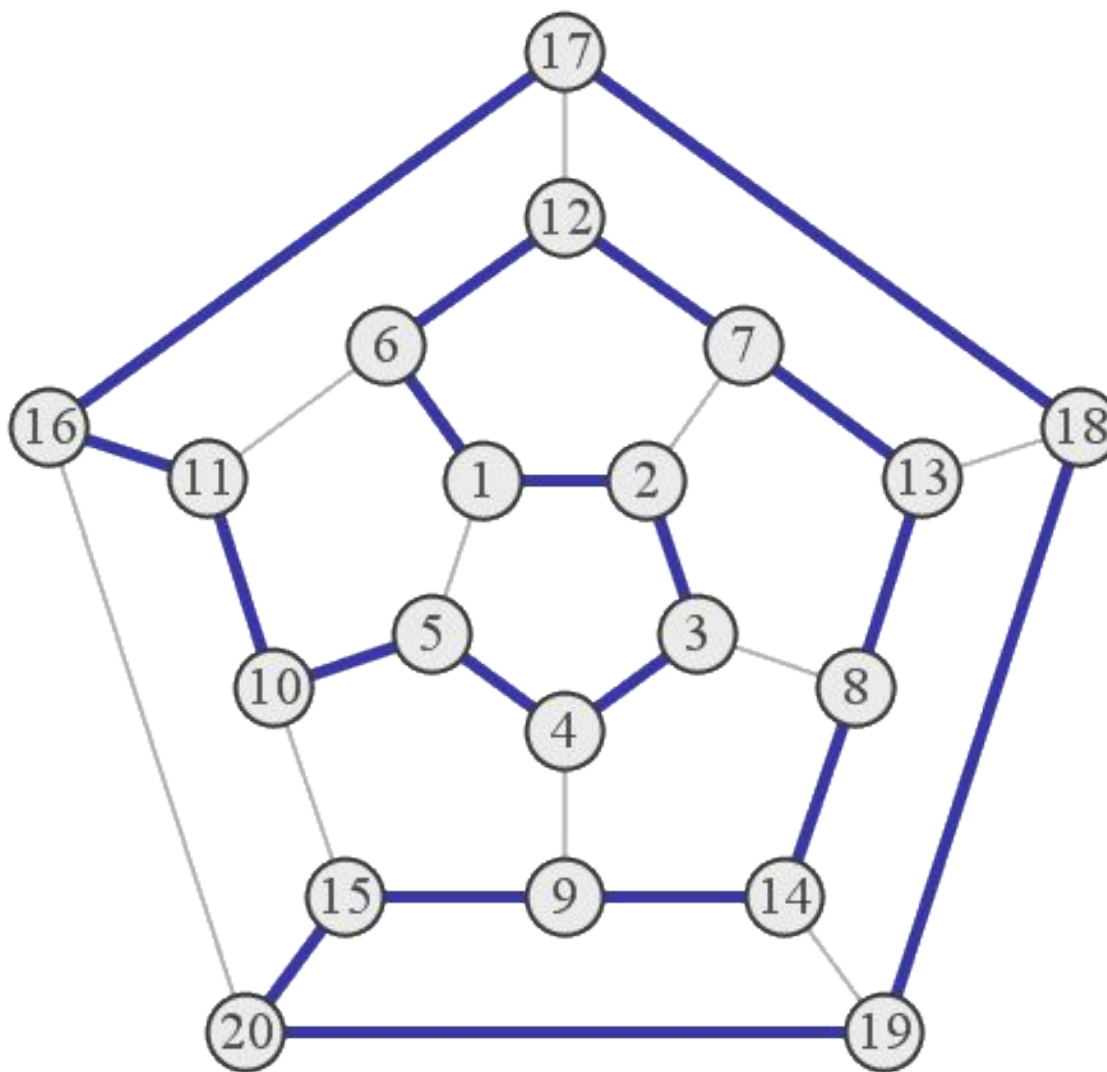


Название графа связано с именем ирландского математика У. Гамильтона, который в 1859 г. предложил игру «Кругосветное путешествие».

В этой игре каждой из двадцати вершин додекаэдра приписано название одного из крупных городов мира. Нужно, переходя от одного города к другому по ребрам додекаэдра, посетить каждый город один раз и вернуться в исходный город.

Эта игра сводится к отысканию в графе додекаэдра простого цикла, проходящего через каждую вершину этого графа.

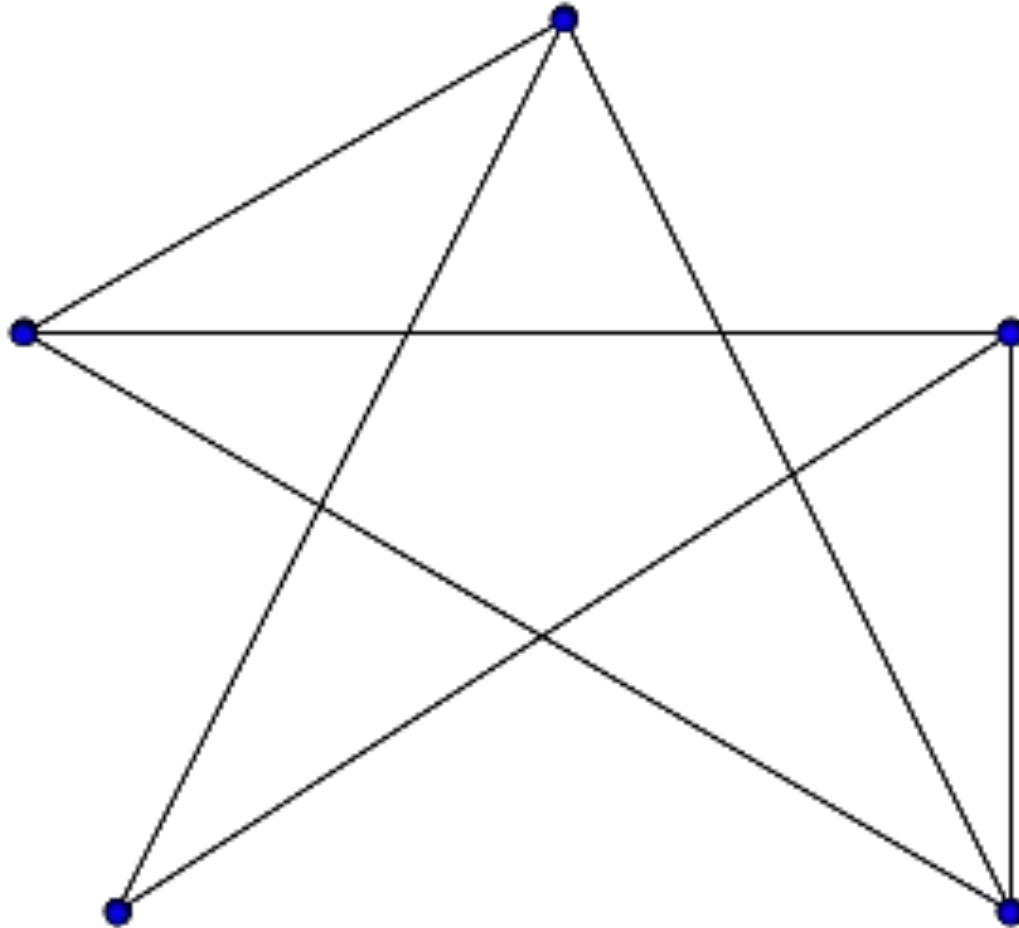
Игра «Кругосветное путешествие»



Простой цикл, содержащий каждую вершину графа, называется **ГАМИЛЬТОНОВЫМ ЦИКЛОМ**.

Простая цепь, содержащая каждую вершину графа, называется **ГАМИЛЬТОНОВОЙ ЦЕПЬЮ**.

Гамильтоновым графом называется граф, в котором есть цикл, содержащий каждую вершину этого графа.



Граф Петерсена имеет гамильтонов путь, но не гамильтонов цикл.

Граф Петерсена является **гипогамильтоновым** — удаление любой вершины делает граф гамильтоновым.

Гиперкуб порядка не
меньше 3 имеет
гамильтонов цикл.
Его описывает код Грея.

Возникает задача об
отыскании всех неподобных
ГЦ в гиперкубе

В 1958г показано, что для Q_3 это 1, для Q_4 это 9.

В 1972г с помощью ЭВМ обнаружили, что для Q_5 это 237230.

Эта ситуация иллюстрирует эмпирическое правило, которое часто обнаруживается при решении комбинаторных задач:

1. Для достаточно малых n задача тривиальна
2. Для следующих значений n задача решается вручную
3. Для следующих значений n задача может быть решена с помощью ЭВМ.
4. Для следующих значений n решение получить невозможно

Условия гамильтоновости

Всякий полный граф является
гамильтоновым.

Теорема.

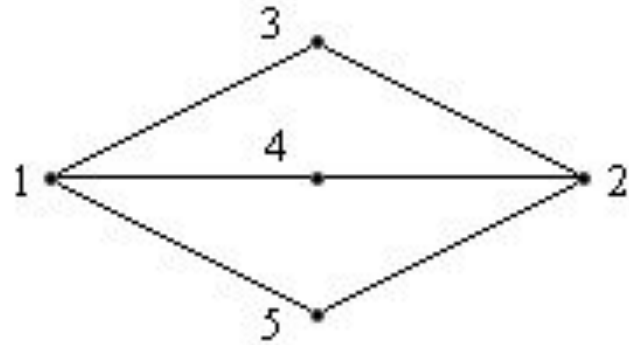
Если граф G имеет разрезающее ребро, то он не может иметь гамильтонов цикл. Если компоненты графа, полученные путем удаления разрезающего ребра, имеют гамильтонов цикл, то граф G имеет гамильтонову цепь.

Если граф содержит висячую вершину,
то он гамильтоновым не является

Легко видеть, что односвязные графы негамильтоновы.

Значит, гамильтоновы графы двусвязные, т. е. они связности 2 и более.

Тэта-графом называют граф, содержащий две вершины степени 3, соединенные тремя простыми попарно непересекающимися цепями длины не менее двух:



Если двусвязный граф содержит тета-граф, то он негамильтонов граф.

Теорема Хватала

Теорема (Хватал):

Пусть:

- G — связный граф,
- $n = |VG| \geq 3$ — количество вершин,
- $d_1 \leq d_2 \leq \dots \leq d_n$ — его последовательность степеней.

Тогда если $\forall k \in \mathbb{N}$ верна импликация:

$$d_k \leq k < n/2 \Rightarrow d_{n-k} \geq n - k, (*)$$

то граф G гамильтонов.

- https://neerc.ifmo.ru/wiki/index.php?title=%D0%A2%D0%B5%D0%BE%D1%80%D0%B5%D0%BC%D0%B0_%D0%A5%D0%B2%D0%B0%D1%82%D0%B0%D0%BB%D0%B0

Теорема Оре — результат в [теории графов](#), доказанный в 1960 году [норвежским](#) математиком [Ойстином Оре](#).

Теорема даёт достаточное условие для того, чтобы граф был [гамильтоновым](#), по существу утверждая, что граф с «достаточно большим числом рёбер» должен содержать [гамильтонов цикл](#). В частности, теорема рассматривает суммы степеней пар несмежных вершин — если каждая такая пара в сумме даёт как минимум общее число вершин графа, граф является гамильтоновым.

Формальное утверждение

Пусть G — (конечный и простой) граф с $n \geq 3$ вершинами. Обозначим через $\deg v$ степень вершины v в G , то есть число инцидентных вершине v рёбер. Теорема Оре утверждает, что если

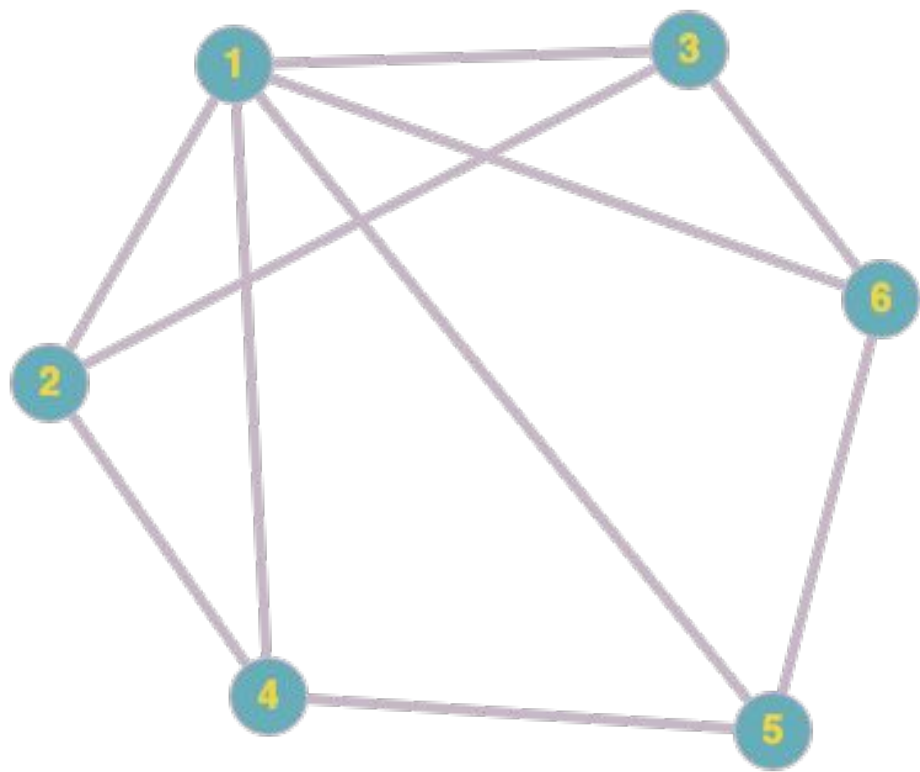
$$\deg v + \deg w \geq n \text{ для любой пары несмежных вершин } v \text{ и } w \text{ графа } G, (*)$$

то G является гамильтоновым.

Доказательство

Утверждение эквивалентно тому, что любой негамильтонов граф G нарушает условие (*). Пусть G — негамильтонов граф с $n \geq 3$ вершинами. Пусть граф H образован из G путём добавления по одному рёбер, не образующих гамильтонов цикл, пока есть возможность такие рёбра добавлять. Рассмотрим любые две несмежные вершины x и y графа H . Добавление ребра xy в H создаёт по меньшей мере один гамильтонов цикл, а рёбра, отличные от xy в этом цикле, должны образовывать гамильтонов путь $v_1 v_2 \dots v_n$ в H с $x = v_1$ и $y = v_n$. Для каждого индекса i в диапазоне $2 \leq i \leq n$, рассмотрим два возможных ребра в H из v_1 в v_i и из v_{i-1} в v_n . Максимум одно из этих рёбер может присутствовать в H , поскольку в противном случае цикл $v_1 v_2 \dots v_{i-1} v_n v_{n-1} \dots v_i$ был бы гамильтоновым. Таким образом, общее число рёбер, инцидентных v_1 или v_n , не превосходит числа возможных i , что равно $n - 1$. Поэтому H не удовлетворяет условию (*), которое требует, чтобы общее число рёбер ($\deg v_1 + \deg v_n$) было больше или равно n . Поскольку степени вершин G не превосходят степеней в H , то G также не удовлетворяет требованию (*).

Теорема Оре



Граф является
гамильтоновым



$$|G| = 6$$

$$\begin{aligned} \deg(2) + \deg(6) &= \\ &= 3 + 3 = 6 \end{aligned}$$

Алгоритм

Палмер описывает следующий простой алгоритм построения гамильтонового цикла в графе, удовлетворяющем условию Оре.

1. Выстроим вершины в цикл произвольным образом, игнорируя смежность в графе.
2. Если цикл содержит две последовательные несмежные вершины, v_i и v_{i+1} , осуществим следующие шаги:
 - Находим индекс j , для которого четыре вершины v_i, v_{i+1}, v_j и v_{j+1} все различны и граф содержит рёбра из v_i в v_j и из v_{j+1} в v_{i+1}
 - Часть цикла между v_{i+1} и v_j (включительно) выстраиваем в обратном порядке.

Каждый шаг увеличивает число последовательных смежных пар на одну или две пары (зависит от того, являются ли v_i и v_{i+1} уже смежными), так что внешний цикл алгоритма может отработать максимум n раз, прежде чем он прервётся, где n — число вершин данного графа. По причинам, аналогичным приведённым в доказательстве теоремы, индекс j должен существовать, в противном случае несмежные вершины v_i и v_{i+1} имеют слишком маленькую суммарную степень. Поиск i и j с последующим инвертированием части цикла можно осуществить за время $O(n)$. Таким образом, общее время работы алгоритма равно $O(n^2)$.

Теорема Оре является обобщением [теоремы Дирака](#), утверждающей, что если каждая вершина имеет степень не меньшую $n/2$, граф является гамильтоновым. Ясно, что если граф удовлетворяет условию Дирака, сумма степеней пар вершин будет не меньше n .

В свою очередь, теорема Оре обобщена до [теоремы Бонди-Хватала](#). Можно определить замыкание графа, добавляя для каждой пары несмежных с суммарной степенью вершин как минимум n ребро, соединяющее эти вершины. Если граф удовлетворяет условию теоремы Оре, его замыкание является [полным графом](#). Теорема Бонди-Хватала утверждает, что граф гамильтонов в том и только в том случае, если его замыкание является гамильтоновым графом. Поскольку полный граф гамильтонов, теорема Оре немедленно вытекает из этой теоремы как следствие.

Теорема Г. Дирака (1952 г.)

Если $|G| = n \geq 3$ и для любой вершины v графа G выполняется неравенство $\deg(v) \geq n/2$, то G – гамильтонов граф.

Теорема (Дирак):

Пусть G — неориентированный граф и δ — минимальная степень его вершин. Если $n \geq 3$ и $\delta \geq n/2$, то G — гамильтонов граф.

Доказательство:

▷ Пусть C — цикл наибольшей длины в графе G . По лемме его длина $l \geq \delta + 1$. Если C — гамильтонов, то теорема доказана. Предположим обратное, т. е. $G \setminus C \neq \emptyset$. Рассмотрим путь $P = x \dots y : P \cap C = \{y\}$ наибольшей длины m . Заметим, что по условию $\delta \geq n/2$, а значит $\delta \geq n - \delta > n - l = |V(G \setminus C)|$. Поэтому каждая вершина из $G \setminus C$ смежна с некоторыми вершинами из C . Заметим, что вершина x не может быть смежна:

- с вершинами из C , расстояние от которых до y (по C) не превышает m . Действительно, пусть вершина $v \in C$ и расстояние от v до y по циклу меньше либо равно m . Тогда этот участок цикла можно заменить на $v \rightarrow x \rightarrow P \rightarrow y$, длина которого $m + 1$. Таким образом образуется цикл большей длины, что противоречит предположению о максимальной длине цикла C .
- двум смежным вершинам на C . Пусть $u, v \in C$ и $\{(u, v), (u, x), (x, v)\} \in E$. Тогда заменив ребро (u, v) на $u \rightarrow x \rightarrow v$, увеличим длину цикла на 1.
- вершинам из $G \setminus (C \cup P)$, поскольку P максимальный.

Получаем $\deg x \leq m + (l - 2m)/2 = l/2 < n/2 \leq \delta$. Противоречие.

◁

Лемма (о длине цикла):

Пусть G — произвольный неориентированный граф и δ — минимальная степень его вершин. Если $\delta \geq 2$, то в графе G существует цикл C длиной $l \geq \delta + 1$.

Доказательство:

▷

Рассмотрим путь максимальной длины $P = v_0 v_1 \dots v_s$. Все смежные с v_0 вершины лежат на P . Обозначим $k = \max\{i : v_0 v_i \in E\}$. Тогда $\delta \leq \deg v_0 \leq k$. Цикл $C = v_0 v_1 \dots v_k v_0$ имеет длину $l = k + 1 \geq \delta + 1$

◁

Теорема (Дирак — альтернативное доказательство):

Пусть G — неориентированный граф и δ — минимальная степень его вершин. Если $n \geq 3$ и $\delta \geq n/2$, то G — гамильтонов граф.

Доказательство:

▷

Для $\forall k$ верна импликация $d_k \leq k < n/2 \Rightarrow d_{n-k} \geq n - k$, поскольку левая её часть всегда ложна. Тогда по теореме Хватала G — гамильтонов граф.

◁

Теорема (Вывод из теоремы Оре):

Пусть G — неориентированный граф и δ — минимальная степень его вершин. Если $n \geq 3$ и $\delta \geq n/2$, то G — гамильтонов граф.

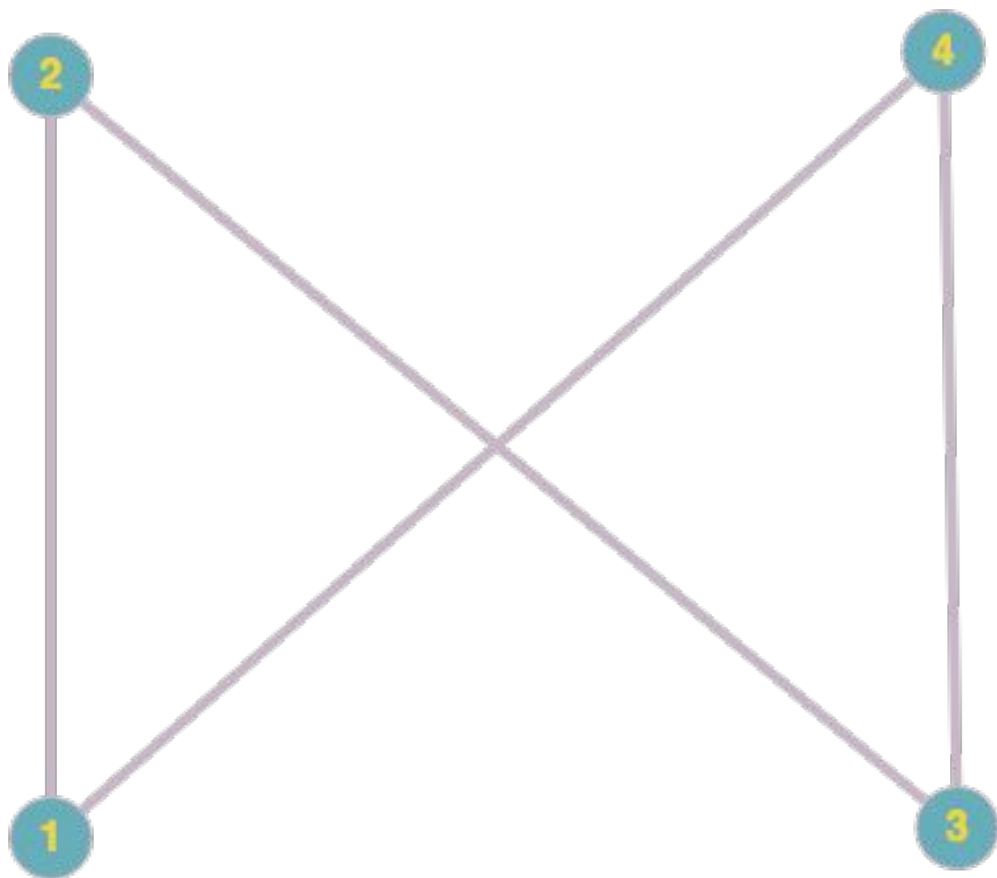
Доказательство:

▷

Возьмем любые неравные вершины $u, v \in G$. Тогда $\deg u + \deg v \geq \frac{n}{2} + \frac{n}{2} = n$. По теореме Оре G — гамильтонов граф.

◁

Теорема Г. Дирака



Граф является
гамильтоновым



$$|G| = 4$$
$$\deg(1) = \deg(2) = \deg(3) =$$
$$= \deg(4) = 2$$

Упр.

Подобрать примеры, иллюстрирующие работу теорем

Нетрудно заметить, что во всяком n -вершинном графе, удовлетворяющем условиям любой из рассмотренных выше теорем, число ребер не меньше чем $(n-1)^2/4$. С другой стороны, n -вершинный планарный граф, как мы знаем, содержит не более $3n-6$ ребер. Поэтому рассмотренные выше достаточные условия заведомо неприменимы к планарным графам.

Легко указать 2-связный негамильтонов планарный граф. Таким графом является, например, тэта-граф. Привести пример 3-связного планарного графа без гамильтонова цикла уже не столь просто. Такой граф, оказывается, должен содержать не менее 11 вершин. Этот факт был установлен Д. Барнеттом и Е. Юковичем (1970 г.). Найденный ими негамильтонов планарный 3-связный граф с минимальным числом вершин изображен на рис. 44.5. В том, что этот

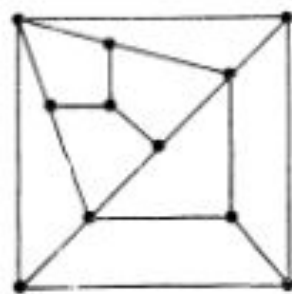


Рис. 44.5

граф не содержит гамильтонова цикла, можно убедиться, проделав не слишком большой в данном случае перебор вариантов. Проще, однако, заметить, что он — двудольный и, следовательно, не содержит циклов нечетной длины, в частности, гамильтоновых циклов.

Следующая теорема — один из наиболее сильных результатов, касающихся гамильтоновых графов.

Теорема У. Татта (1946 г.)

Всякий 4-связный планарный граф является гамильтоновым.

опускается.

Следствие 44.6 (Х. Уитни, 1932 г.). Если в максимальном плоском графе G каждый 3-цикл является границей некоторой грани, то G — гамильтонов граф.

Пример максимального планарного негамильтонова графа G с 11 вершинами приведен на рис. 44.6. Отсутствие гамильтонова цикла следует из того, что шесть вершин графа G , помеченных знаком «+», попарно не смежны, а на цикле длины 11 могут лежать не более пяти таких вершин. Согласно следствию 44.6, этот граф должен содержать 3-цикл, не являющийся границей никакой грани. Такими 3-циклами являются, например, (a, b, c) и (a, c, d) .

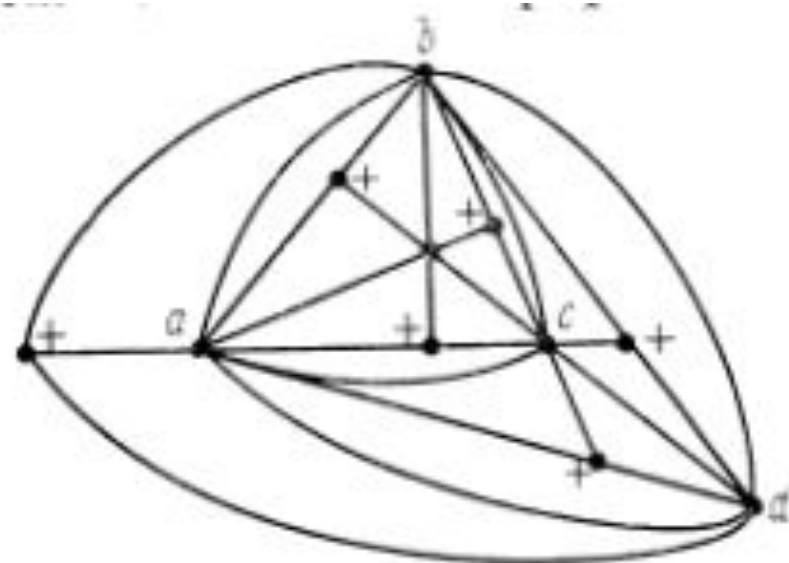


Рис. 44.6

Если в n -вершинном графе G фиксировать одну вершину и обход всегда начинать с нее, то всякому гамильтонову циклу очевидным образом будет соответствовать перестановка элементов множества V_G . Тем самым найти гамильтонов цикл либо убедиться в отсутствии такого цикла можно путем перебора $(n - 1)!$ перестановок. Если граф G гамильтонов, то проделать этот перебор в полном объеме придется только в случае крайнего невезения — когда нужная, т. е. отвечающая гамильтонову циклу перестановка встретится последней в этом процессе. Если же G — негамильтонов граф, то действуя подобным образом, придется в любом случае проверить все $(n - 1)!$ перестановок.

На практике не пользуются столь бесхитрым способом распознавания гамильтоновости, а применяют различные алгоритмы частичного перебора. Однако и там наблюдается такой же эффект — т. е. негамильтоновость

установить, как правило, труднее, чем гамильтоновость. В этой ситуации были бы полезны необходимые условия гамильтоновости. Такие условия нужны и для построения примеров негамильтоновых графов с заданными свойствами. К сожалению, для графов общего вида необходимые условия гамильтоновости неизвестны, за исключением уже упоминавшегося банального факта, что гамильтонов граф должен быть 2-связным. Иначе обстоит дело с планарными графами.

Теорема 44.7 (Э. Я. Гринберг, 1968 г.). Пусть f_k — число k -граней плоского графа G . Если G — гамильтонов граф, то для каждого $k \geq 3$ существуют такие целые неотрицательные числа f'_k, f''_k , что

$$\begin{aligned} f'_k + f''_k &= f_k, \\ \sum_{k \geq 3} f'_k (k - 2) &= \sum_{k \geq 3} f''_k (k - 2). \end{aligned} \tag{1}$$

Доказательство смотри, например, у Мельникова

Установление негамильтоновости графа путем перебора всех разбиений чисел f_k и последующей проверки выполнения равенства (1) едва ли проще полного перебора перестановок, о котором упоминалось выше. Польза теоремы 44.7 в другом — иногда удается, проанализировав числа f_k , сразу сделать вывод об отсутствии подходящего разбиения, т. е. вывод о негамильтоновости графа. Рассмотрим в качестве примера граф, изображенный

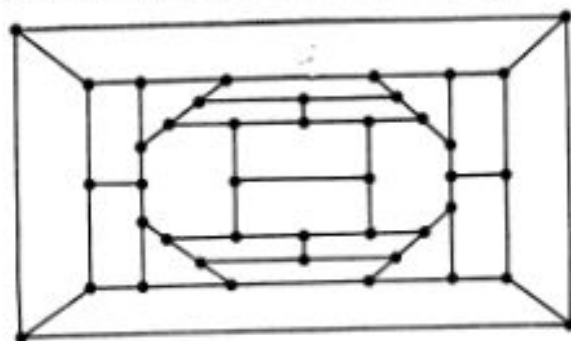


Рис. 44.8

на рис. 44.8. Для этого графа $f_4 = 1$, $f_8 = 18$, $f_{16} = 4$, а остальные f_k равны 0. Поэтому соотношение (1) должно иметь вид

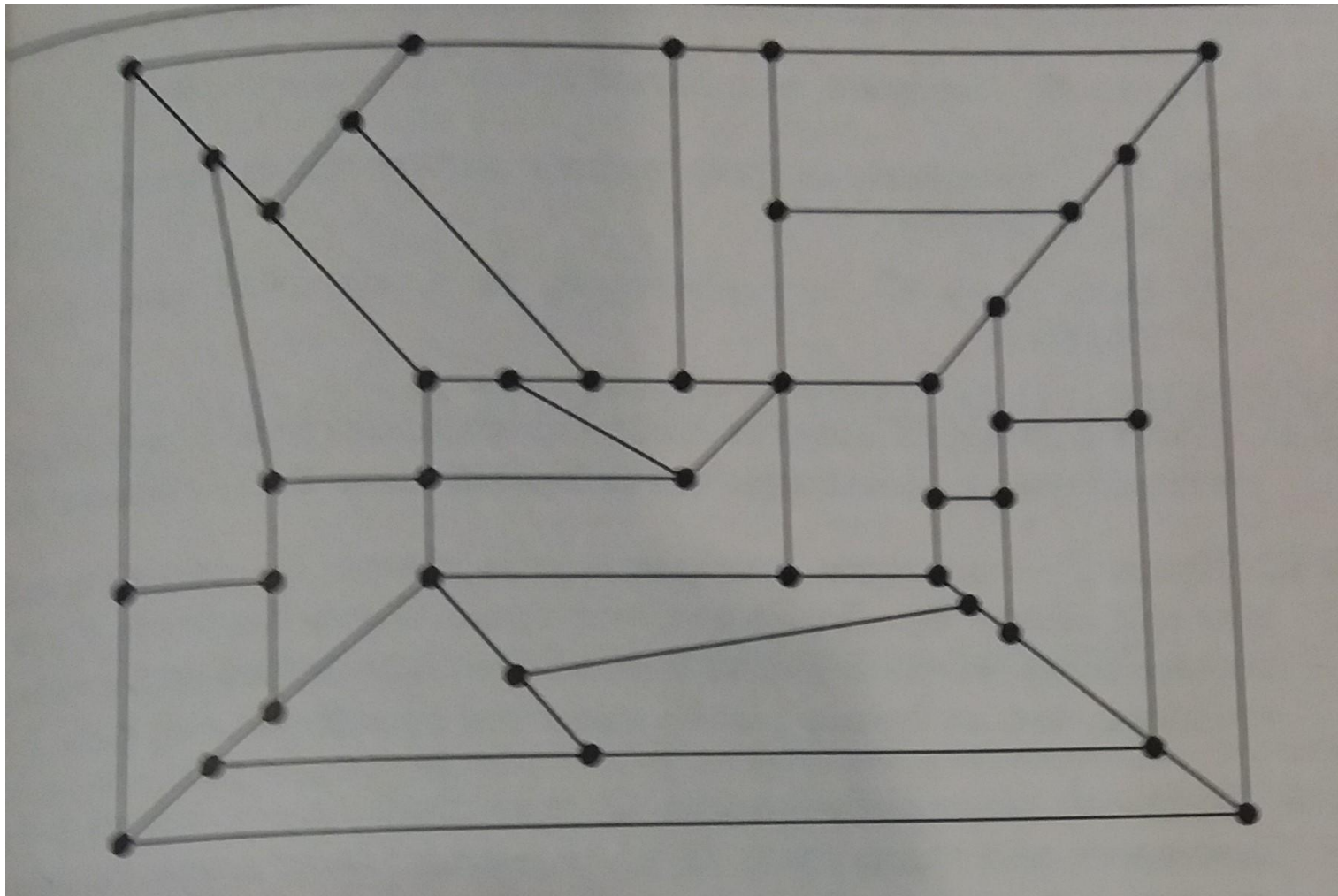
$$2f'_4 + 3f'_8 + 6f'_{16} = 2f''_4 + 3f''_8 + 6f''_{16},$$

откуда следует, что $f'_4 \equiv f''_4 \pmod{3}$. Это, однако, невозможно, поскольку $f'_4 + f''_4 = f_4 = 1$. Таким образом,

нужного разбиения чисел f_k не существует, и значит, граф негамильтонов.

Рассмотрим еще один пример негамильтонового графа.

Покажите, что в этом графе нет гамильтонова цикла



Утверждение 1. Пусть в графе G имеется гамильтонов цикл. Тогда количество $k := c(G - S)$ компонент связности U_1, \dots, U_k , получающихся в результате удаления вершин некоторого непустого подмножества $S \subset V(G)$ графа G , не превосходит количества удаленных вершин:

$$c(G - S) \leq |S|. \quad (3.1)$$

Почти все графы гамильтоновы

Укажем некоторые задачи, интерпретация которых состоит в необходимости построения гамильтоновых циклов.

Задача о шахматном коне. Можно ли, начиная с произвольного поля на шахматной доске, ходить конем в такой последовательности, чтобы пройти через все поле и вернуться в исходное?

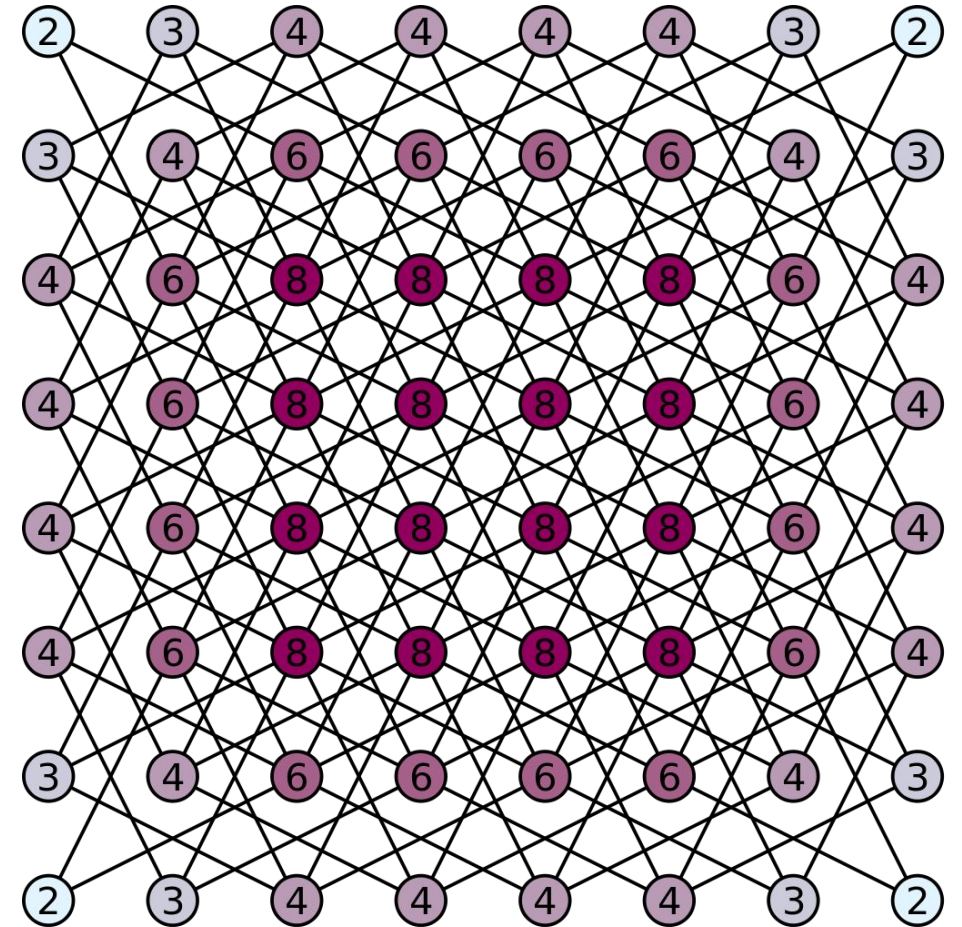
Задача о званом обеде. Обед накрыт на круглом столе. Среди приглашенных некоторые являются друзьями. При каких условиях можно рассадить всех так, чтобы по обе стороны каждого из присутствующих сидели его друзья?

Задача о коммивояжере. Коммивояжер должен объездить все n городов. Для того чтобы сократить свои расходы, он хочет построить такой маршрут, чтобы объездить все города точно по одному разу и вернуться в исходный с минимумом затрат.

Задача о ходе коня

В терминах [теории графов](#) каждый маршрут коня, проходящий через все поля шахматной доски, соответствует [гамильтонову пути](#) (или циклу, если маршрут замкнутый) в [графе](#), вершинами которого являются поля доски, и два поля соединены ребром, если с одного можно попасть на другое за один ход коня. Для доски 8×8 количество всех замкнутых маршрутов коня (гамильтоновых циклов) без учёта направления обхода равно 13 267 364 410 532. Количество всех незамкнутых маршрутов (с учётом направления обхода) равно 19 591 828 170 979 904.

<https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%B4%D0%B0%D1%87%D0%B0%D0%BE%D1%85%D0%BE%D0%B4%D0%B5%D0%BA%D0%BE%D0%BD%D1%8F>



Оценка пространственной сложности алгоритмов

Задача коммивояжера

Оптимизационная постановка задачи относится к классу NP-полных задач, как и большинство её частных случаев.

NP-полная задача — в теории алгоритмов задача с ответом «да» или «нет» из класса NP, к которой можно свести любую другую задачу из этого класса за полиномиальное время (то есть при помощи операций, число которых не превышает некоторого полинома в зависимости от размера исходных данных). Таким образом, NP-полные задачи образуют в некотором смысле подмножество «типовых» задач в классе NP: если для какой-то из них найден «полиномиально быстрый» алгоритм решения, то и любая другая задача из класса NP может быть решена так же «быстро».

Задачи коммивояжера решаются посредством различных методов, выведенных в результате теоретических исследований. Все эффективные методы (сокращающие полный перебор) — методы эвристические. В большинстве эвристических методов находится не самый эффективный маршрут, а приближённое решение. Зачастую востребованы алгоритмы, постепенно улучшающие некоторое текущее приближенное решение. Выделяют следующие группы методов решения задач коммивояжера, которые относят к простейшим:

- Полный перебор;

Полный перебор (или метод «грубой силы») — метод решения задачи путем перебора всех возможных вариантов. Сложность полного перебора зависит от количества всех возможных решений задачи. Если пространство решений очень велико, то полный перебор может не дать результатов в течение нескольких лет или даже столетий.

Случайный перебор;
Обычно выбор решения можно представить последовательностью выборов. Если делать эти выборы с помощью какого-либо случайного механизма, то решение находится очень быстро, так что можно находить решение многократно и запоминать «рекорд», т. е. наилучшее из встретившихся решений. Этот наивный подход существенно улучшается, когда удается учесть в случайном механизме перспективность тех или иных выборов, т. е. комбинировать случайный поиск с эвристическим методом и методом локального поиска. Такие методы применяются, например, при составлении расписаний для Аэрофлота.

Жадные алгоритмы (метод ближайшего соседа, метод включения ближайшего города, метод самого дешевого включения);
Жадный алгоритм – алгоритм нахождения наикратчайшего расстояния путём выбора самого короткого, ещё не выбранного ребра, при условии, что оно не образует цикла с уже выбранными рёбрами. «Жадным» этот алгоритм назван потому, что на последних шагах приходится жестоко расплачиваться за жадность. При решении задачи коммивояжера жадный алгоритм превратится в стратегию «иди в ближайший (в который еще не входил) город». Жадный алгоритм, очевидно, бессилён в этой задаче. Рассмотрим для примера сеть (рис. 2), представляющую узкий ромб. Коммивояжер стартует из города 1. Алгоритм «иди в ближайший город» выведет его в город 2, затем 3, затем 4; на последнем шаге придется платить за жадность, возвращаясь по длинной диагонали ромба. В результате получится не кратчайший, а длиннейший тур.

Метод минимального остовного дерева (деревянный алгоритм);

В основе алгоритма лежит утверждение: «Если справедливо неравенство треугольника, то для каждой цепи верно, что расстояние от начала до конца цепи меньше (или равно) суммарной длины всех ребер цепи». Это обобщение расхожего убеждения, что прямая короче кривой. Деревянный алгоритм для решения задачи коммивояжера будет следующим: строится на входной сети задачи коммивояжера кратчайшее остовное дерево и удваиваются все его ребра. В результате получаем граф — связный с вершинами, имеющими только четные степени. Затем строится эйлеров цикл, начиная с вершины 1, цикл задается перечнем вершин. Просматривается перечень вершин, начиная с 1, и зачеркивается каждая вершина, которая повторяет уже встреченную в последовательности. Останется тур, который и является результатом алгоритма.

Доказано, что деревянный алгоритм ошибается менее чем в два раза, поэтому такие алгоритмы называют приближительными, а не просто эвристическими.

- Метод имитации отжига.

Экзотическое название данного алгоритма связано с методами имитационного моделирования в статистической физике, основанными на технике Монте-Карло. Исследование кристаллической решетки и поведения атомов при медленном остывании тела привело к появлению на свет вероятностных алгоритмов, которые оказались чрезвычайно эффективными в комбинаторной оптимизации. Впервые это было замечено в 1983 году. Сегодня этот алгоритм является популярным как среди практиков благодаря своей простоте, гибкости и эффективности, так и среди теоретиков, поскольку для данного алгоритма удастся аналитически исследовать его свойства и доказать асимптотическую сходимость. Алгоритм имитации отжига относится к классу пороговых алгоритмов локального поиска. На каждом шаге этого алгоритма для текущего решения i_k в его окрестности $N(i_k)$ выбирается некоторое решение j и, если разность по целевой функции между новым и текущим решением не превосходит заданного порога t_k , то новое решение j заменяет текущее. В противном случае выбирается новое соседнее решение. На практике применяются различные модификации более эффективных методов:

Метод ветвей и границ;

Метод ветвей и границ предложен в 1963 году группой авторов Дж. Литлом, К. Мурти, Д. Суини, К. Кэролом. Широко используемый вариант поиска с возвратом, фактически является лишь специальным частным случаем метода поиска с ограничениями⁴. Ограничения в данном случае основываются на предположении, что на множестве возможных и частичных решений задана некоторая функция цены и что нужно найти оптимальное решение, т.е. решение с наименьшей ценой. Для применения метода ветвей и границ функция цены должна обладать тем свойством, что цена любого частичного решения не превышает цены любого расширения этого частичного решения (Заметим, что в большинстве случаев функция цены неотрицательна и даже удовлетворяет более сильному требованию). Столь большой успех применения данного метода объясняется тем, что авторы первыми обратили внимание на широту возможностей метода, отметили важность использования специфики задачи и сами воспользовались спецификой задачи коммивояжера.

В основе метода ветвей и границ лежит идея последовательного разбиения множества допустимых решений на подмножества. На каждом шаге метода элементы разбиения подвергаются проверке для выяснения, содержит данное подмножество оптимальное решение или нет. Проверка осуществляется посредством вычисления оценки снизу для целевой функции на данном подмножестве. Если оценка снизу не меньше рекорда — наилучшего из найденных решений, то подмножество может быть отброшено. Проверяемое подмножество может быть отброшено еще и в том случае, когда в нем удается найти наилучшее решение. Если значение целевой функции на найденном решении меньше рекорда, то происходит смена рекорда. По окончании работы алгоритма рекорд является результатом его работы. Если удастся отбросить все элементы разбиения, то рекорд — оптимальное решение задачи. В противном случае, из неотброшенных подмножеств выбирается наиболее перспективное (например, с наименьшим значением нижней оценки), и оно подвергается разбиению. Новые подмножества вновь подвергаются проверке и т.д.

Метод генетических алгоритмов;
«Отцом-основателем» генетических алгоритмов считается Джон Холланд, книга которого «Адаптация в естественных и искусственных системах» (1975) является основополагающим трудом в этой области исследований. Генетический алгоритм — это эвристический алгоритм поиска, используемый для решения задач оптимизации и моделирования путём случайного подбора, комбинирования и вариации искомым параметров с использованием механизмов, напоминающих биологическую эволюцию. Является разновидностью эволюционных вычислений. Отличительной особенностью генетического алгоритма является акцент на использование оператора «скрещивания», который производит операцию рекомбинации решений-кандидатов, роль которой аналогична роли скрещивания в живой природе. Генетические алгоритмы служат, главным образом, для поиска решений в многомерных пространствах поиска.

Алгоритм с возвратами

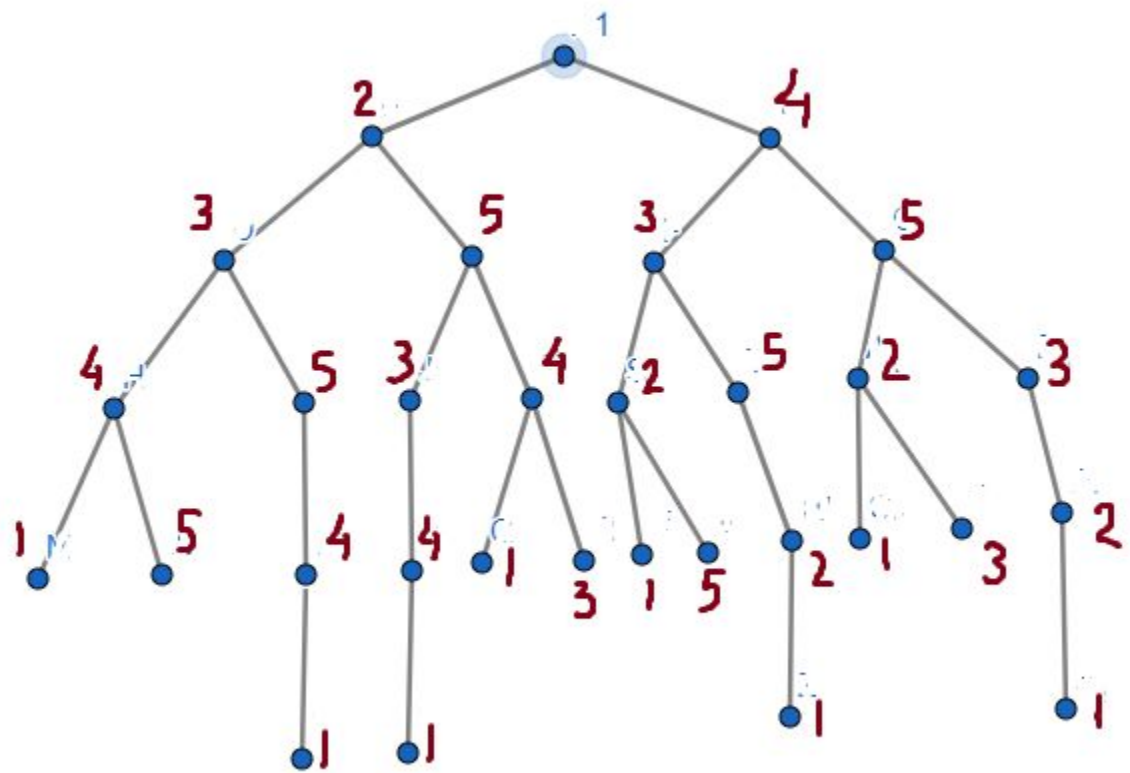
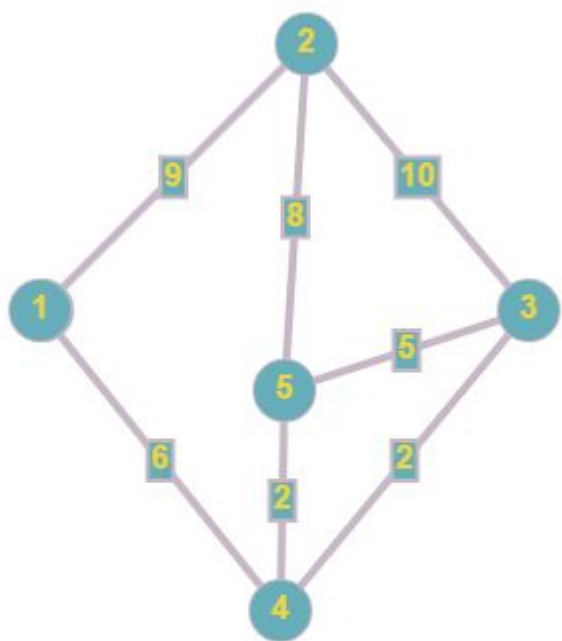
Будем искать решение в виде последовательности вершин $\{v_1, v_2, \dots, v_n\}$, начиная от корневой вершины v_1 . Имея частичное решение $\{v_1, v_2, \dots, v_i\}$, $i < n$, пытаемся найти такое допустимое значение v_{i+1} , относительно которого можно предположить, что оно может расширить частичное решение до $\{v_1, v_2, \dots, v_{i+1}\}$. Если такое предположение возможно, то v_{i+1} включается в частичное решение и процесс продолжается. Если никакая вершина не может претендовать на v_{i+1} , то следует возврат на шаг к v_{i-1} , а v_i из данного частичного решения исключается.

Процесс поиска нового допустимого значения продолжается, начиная с v_{i-1} . Когда v_n найдено, отмечают $\{v_1, v_2, \dots, v_n\}$ как решение.

Если поставлена задача отыскания всех решений, то попытка получения каждого нового решения осуществляется возвратом на шаг от v_n в предположении, что решение как бы получено не было. Таким образом, отыскивается множество всех решений, которое, в частности, оказывается пустым, если гамильтонова цикла на графе не существует.

Из описания алгоритма с возвратами видно, что задача построения всех гамильтоновых циклов на графе G может быть решена путем сопоставления исходному графу некоторого другого графа G' , построенного из частично пересекающихся цепей $\{v_1, v_2, \dots, v_i\}$, ($i \leq n$). Подмножество G' , состоящее из цепей $\{v_1, v_2, \dots, v_n\}$, таких что v_n смежна с v_1 на G , представляет множество решений задачи. Граф G' называется *соотнесенным* графом.

Рассмотрим на примере



Всего имеем четыре гамильтоновых цикла:

a) $\{1, 2, 3, 5, 4, 1\}$,

b) $\{1, 2, 5, 3, 4, 1\}$,

c) $\{1, 4, 3, 5, 2, 1\}$,

d) $\{1, 4, 5, 3, 2, 1\}$.

Стоимости передачи информации каждым из этих циклов вычисляются ниже:

a) $9+10+2+2+6 = 29$,

b) $9+8+2+5+6 = 30$,

c) $6+5+2+8+9 = 30$,

d) $6+2+2+10+9 = 29$.

Решению поставленной задачи отвечают гамильтоновы циклы a) и d).

Можно показать, что алгоритмы с возвратами имеют вычислительную сложность $O(c^n)$, где $c > 1$ — константа, определяемая свойствами присоединенного графа. Такое быстрое действие при больших n значительно выше, чем в случае применения алгоритмов факториальной сложности. Например, возвращаясь к рассмотренному выше примеру оценки быстрого действия алгоритма с возвратами при $n = 20$, видим, что в случае когда $c = 2$ (граф G двоичное дерево), имеем всего 2^{20} или порядка 10^6 вычислительных операций. На любом современном компьютере это практически не займет времени. Однако, при увеличении размерности задачи всего лишь в несколько раз снова возникают проблемы реального времени. Это способствует развитию новых идей построения гамильтоновых циклов.

Алгоритм деревя

Приближенный алгоритм
решения задачи коммивояжера —
алгоритм Дрейфа

Пусть $D = [d_{ij}]_{n \times n}$ — м-ца расстояний,
задающая n -ую ком-ву, $d_{ij} > 0$

D — симметричная м-ца

Будем говорить, что м-ца D удовлетв.

тр-ву треугольника, если $d_{ij} + d_{jk} \geq d_{ik}$

$$\forall 1 \leq i, j, k \leq n$$

Содержательно выполнимые тр-ва Тр-ва означают, что путь из города i в город k через город j не л.б. короче (дешевле), чем путь непосредственно из г. i в г. k .

(В общем случае, когда n -ая м-ца расстояний не удовлетв. тр-ву, то некоторые из тр-ва могут нарушаться.)

Алг-м Дрейфа

1. Дано полное графа с м-цей весов D найти мин. остовное дерево T .
2. Построить мультиграф G , используя две копии каждого ребра T .
3. Найти ~~минимум~~^{цикл} ~~минимум~~ в G и исключить в него обход.

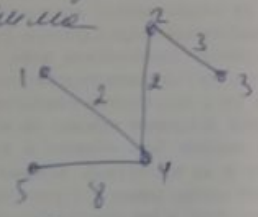
Вход строится путем выбора из
 чисел v_k в v_{k-1} в v_{k-2} по формуле
 включительно каждого целого числа
 из $\{1, 2, \dots, n\}$. По стр. v_{k-1} марта
 марта v_{k-1} по v_{k-2} в нем по v_{k-1}
 один раз.

Пример

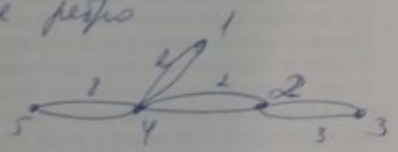
$$P = \begin{pmatrix} \infty & 4 & 6 & 2 & 9 \\ 4 & \infty & 3 & 2 & 9 \\ 6 & 3 & \infty & 5 & 9 \\ 2 & 2 & 5 & \infty & 8 \\ 9 & 9 & 9 & 8 & \infty \end{pmatrix}$$

Итерационной проверкой убеждаемся,
 что v_{k-1} и v_{k-2} v_{k-1} v_{k-2} v_{k-1} v_{k-2}

1. Строим минимальное дерево, напр.
 с пом. алгоритма Крускала



2. Выбираем любое ребро



3. Строим эйлеров цикл: напр., 5-4-1-4-2-3-2-4-5
 Узнаем из него повторные включения:

$$5 \begin{matrix} 4 \\ 1 \\ 2 \\ 3 \\ 5 \end{matrix}$$

$$\text{стоимость} = 8 \cdot 2 \cdot 4 + 3 \cdot 9 = 26$$

полученное решение на 1 приватной оптимально

12345

$$4+3+5+8+9=29$$

13245

$$6+5+2+8+9=28$$

13425

$$6+5+2+9+9=31$$

13524

$$6+9+9+2+2=28$$

13542

$$6+9+8+2+4=29$$

12354

$$4+3+~~5~~+9+8+2=26 \leftarrow \text{invalid}$$

12435

$$4+2+5+9+9=29$$

12453

$$4+2+8+9+6=29$$

12534

$$4+9+9+5+2=29$$

12543

$$4+9+8+5+6=32$$

13254

$$6+3+9+8+1=27$$

13452

$$6+5+8+9+4=32$$

14235

$$2+2+3+9+9=25 \leftarrow \text{invalid}$$

14253

$$2+2+9+9+6=28$$

14325

$$2+5+3+9+9=28$$

14352

2+

14532

2+

14523

2+

15234

15243

15324

15342

15423

15432