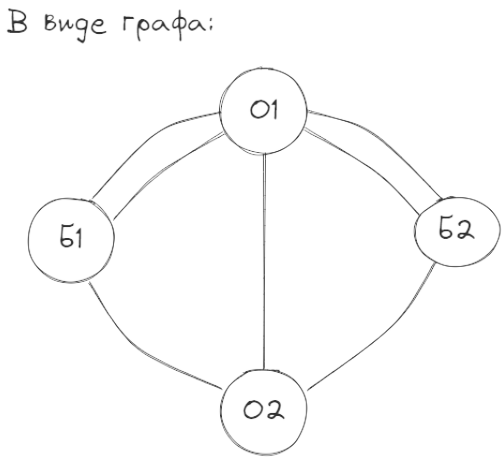
# Теория графов

ИУ6-25Б

2024

## Задача о кёнингсбергских мостах


**Опр.** Графом $G(X, U)$ называется математический объект, заданный парой множеств $X = x_1, x_2, \ldots, x_n$ - множество вершин - и $U = i_1, u_2, \ldots, u_m$ - множество рёбер.

Так как любые 2 вершины могут быть связаны или не связаны ребром, то на множестве вершин задаётся бинарное отношение.

**Опр.** Ребро, заданное неупорядоченной парой $u_l = \{x_i, x_j\}$ называется неориентированным.

**Опр.** Граф, состоящий только из неориентированных рёбер, называет неориентированным, неографом.

**Опр.** Ребро, заданное упорядоченной парой вершин $u_l = (x_i, x_j)$, называется ориентированным.

**Опр.** Граф, состоящий только из ориентированных рёбер, называет ориентированным, орграфом.

**Опр.** Граф, состоящий из ориентированных и неориентированных рёбер, называется смешанным.

**Опр.** Если в графе хотя бы одну пару вершин связывает несколько рёбер, то такой граф называется мультиграфом.

**Опр.** Вершины, определяющие ребро, называются концевыми вершинами.

**Опр.** Если концевые вершины ребра совпадают, то такое ребро называется петлей.

## Понятие смежности и инцидентности

Между вершинами и рёбрами графа имеет место отношение инцидентности: вершина инцидентна ребру, если является одной из его концевых вершин.

Между вершинами или между рёбрами имеет место отношение смежности:

- Два ребра называются смежными, если имеют общую концевую вершину.
- Две вершины называются смежными, если являются концевыми вершинами одного ребра.

Отношение инцидентности имеет место быть между однородными компонентами графа, отношение смежности - между однородными компонентами.

Отношение смежности является отношением толерантности, если принять, что вершина(ребро) смежна сама себе.

**Опр.** Граф называется простым, если в нём нет петель и кратных рёбер.

**Опр.** Граф называется полным, если 2 любые его вершины смежны.

Полный граф обязан быть простым.

**Опр.** Граф называется графом Кёнига (двудольным графом), если множество вершин можно разбить на 2 непересекающихся множества $X_1$ и $X_2$ ($X_1 \cap X_2 = \varnothing$) таких, что для каждого ребра верно, что одна из его вершин принадлежит $X_1$, а другая - $X_2$

- $\forall u_l \in U, u_l = (x_i, x_j) : x_i \in X_1, x_j \in X_2, i \neq j$
- $X_1$ и $X_2$ называются долями двудольного графа

### Понятие смежности

**Опр.** $|X| = n$ - мощность множества вершин называется порядком графа
**Опр.** Число рёбер, инцидентных вершине $x_i \in X$, называется степенью вершины $x_i$
**Обоз.** $p(x_i) = n$
**Опр.** Если $p(x_i) = 1$, то вершина $x_i$ называется висячей
**Опр.** Если $p(x_i) = 0$, то вершина $x_i$ называется изолированной
В случае орграфа:

- $p^+(x_i)$ - полустепень захода - количество ориентированных рёбер, входящих в вершину $x_i$
- $p^-(x_i)$ - полустепень исхода - количество ориентированных рёбер, исходящих из вершины $x_i$
- $p(x_i) = p^+(x_i) + p^-(x_i)$

Для произвольного графа $G$ верно, что сумма степеней его вершин равна удвоенному числу его рёбер: $\Sigma p(x_i) = 2|U|$ $x_i \in X$ - лемма о рукопожатиях

Каждое ребро инцидентно двум вершинам, отсюда 2 в правой части. Каждое ребро принимает участие в формировании степени обеих своих концевых вершин.
**Опр.** Если $p^-(x_i) = 0$, то вершина $x_i$ называется стоком. $(x_i = t)$
**Опр.** Если $p^+(x_i) = 0$, то вершина $x_i$ называется источником. $(x_i = s)$
**Опр.** Граф с одним источником и одним стоком называется сетью

## Изоморфизм графов

**Опр.** Рассмотрим графы $G_1(X_1, U_1)$ и $G_2(X_2, U_2)$. Они называются изоморфными, если между множествами вершин $X_1$ и $X_2$ установлено взаимно однозначное соответствие такое, что между $U_1$ и $U_2$ устанавливается также взаимно однозначное соответствие. А именно каждое ребро из $U_2$ инцидентно 2 своим концевым вершинам из $X_2$ $\iff$ когда соответствующие им вершины из $X_1$ инцидентны ребру из множества $U_1$
**Теорема.** Изоморфизм графом является отношением эквивалентности:

1. Рефлексивность - граф изоморфен сам себе $G \leftrightarrow G$
2. Симметричность - $f : G_1 \leftrightarrow G_2 \implies f^{-1} : G_2 \leftrightarrow G_1$
3. Транзитивность:
   - $f_1 : G_1 \leftrightarrow G_2$
   - $f_2 : G_2 \leftrightarrow G_3$
   - $f_3 = f_1 \circ f_2$
   - $f_3 : G_1 \leftrightarrow G_3$

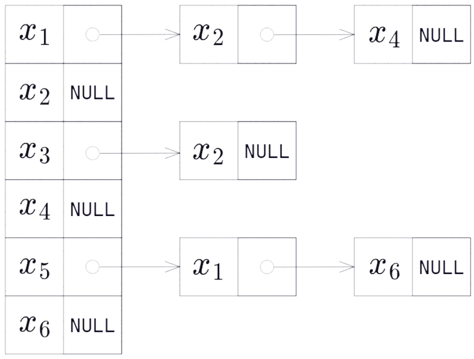
Из теоремы следует, что множество всех графов разбивается на классы эквивалентности такие, что в пределах каждого класса все графы изоморфны, а графы из разных классов не изоморфны (*"с точностью до изоморфизма"*).

### Части графа

Рассмотрим граф $G(X, U)$
**Опр.** Граф $G_1(X_1, U_1)$ называется частью графа $G$, если он находится в отношении включения к нему: $G_1 \subseteq G$ $(X_1 \subseteq X, U_1 \subseteq U)$
**Опр.** Часть графа называется подграфом, если включение строгое: $G_1 \subset G$ $(X_1 \subset X, U_1 \subset U)$
Обозначение удаления ребра: $G - (x_i, x_j)$
**Опр.** Часть графа называется субграфом, если $X_1 = X$, а $U_1 \subset U$

# Способы задания графа

1. Матричный
2. Аналитический - основан на понятии отображения
   - **Обоз.** $\Gamma_{x_i}$ - прямое отображение вершины $x_i$
   - *Пример:*
     - $\Gamma_{x_1} = \{x_2\}$ - исходящие рёбра
     - $\Gamma_{x_2} = \varnothing$
     - $\Gamma_{x_3} = \{x_2\}$
     - $\Gamma_{x_4} = \varnothing$
     - $\Gamma_{x_5} = \{x_1\}$
     - $\Gamma_{x_6} = \varnothing$
     - $\Gamma_{x_1}^{-1} = \{x_5\}$ - входящие рёбра
3. Списковые структуры данных. Вектор Айлифа:




4. Массив рёбер или пар смежных вершин:

| | |
|---|---|
| $x_1$ | $x_2$ |
| $x_5$ | $x_1$ |
| $x_1$ | $x_4$ |
| $x_3$ | $x_2$ |
| $x_5$ | $x_6$ |

## Матричный способ

### 1. Матрица смежности

Матрица смежности - квадратная булева матрица $M$ порядка $n = |X|$.

- $M_{ij} = 1$, если $x_i$ и $x_j$ смежны
- $M_{ij} = 0$, если $x_i$ и $x_j$ не смежны

| | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $x_6$ |
|---|---|---|---|---|---|---|
| $x_1$ | | 1 | | 1 | 1 | |
| $x_2$ | 1 | | 1 | | | |
| $x_3$ | | 1 | | | | |
| $x_4$ | 1 | | | | | |
| $x_5$ | 1 | | | | | 1 |
| $x_6$ | | | | | 1 | |

- Сумма элементов равна числу рёбер графа
- Самый быстрый, но затратный по памяти способ
- Матрицу смежности можно построить для рёбер

**2. Матрица инциденций**

Матрица порядка $m \times n$, где $m = |X|, n = |U|$
Для неографов:

- $H_{ij} = 1$, если $x_i$ инцидентна $u_j$
- $H_{ij} = 0$, иначе
- Сумма столбца $= 2$

Для орграфов:

- $H_{ij} = 1$, если $x_i$ инцидента $u_j$ и является конечной для него
- $H_{ij} = 01$, если $x_i$ не инцидента $u_j$
- $H_{ij} = -1$, если $x_i$ инцидента $u_j$ и является начальной для него
- Сумма столбца $= 0$

По строкам матрицы можно вычислить степени и полустепени вершин.
**Опр.** Маршрутом, соединяющим вершины $x_i, x_j$ в графе, называется чередующаяся последовательность $x_i u_q x_w u_e \dots u_k x_j$

- Любые 2 соседних элемента связаны отношением инцидентности
- Любые 2 соседних через один связаны отношением смежности относительно стоящего между ними
- Маршрут не учитывает ориентацию рёбер
- Вершины и рёбра могут повторяться

**Опр.** Маршрут без повторяющихся рёбер называют цепью
**Опр.** Цепь, все вершины которой различны, называют простой
**Опр.** Простая цепь, начальная и конечная вершины которой совпадают, называется циклом
**Опр.** Цикл в орграфе называют контуром
**Опр.** Цепь в орграфе называют путём
**Опр.** Число рёбер, составляющих в цепь, называют длиной цепи или пути

## Связь понятий маршрута и связности

**Опр.** Граф является связным тогда и только тогда, когда любые 2 различные его вершины соединены маршрутом
**Опр.** Сильная связность в орграфе подразумевает, что между 2 любыми вершинами существует путь, слабая связность игнорирует направление рёбер
Отношение связности на множестве вершин графа является отношением эквивалентности:

- Рефлексивно, так любая вершина связна сама с собой
- Симметрично, так как для любого маршрута существует обратный по тем же рёбрам
- Транзитивно, так как $x_i \to x_j, x_j \to x_k \implies x_i \to x_k$

Отношение связности разбивает множество вершин графа на классы эквивалентности.
**Опр.** Вершина графа $x_i$ называется точкой сочленения, если её удаление из графа увеличивает число компонент связности.
**Опр.** Если существует хотя бы одна точка сочленения, то граф называется разделимым (иначе неразделимым)
**Теорема.** Вершина $x_k$ связного графа $G(X, U)$ является точкой сочленения, если и только если в графе $\exists x_i \neq x_j \neq x_k$ такие, что любой путь или любая цепь между $x_i$ и $x_j$ проходит через $x_k$
**Опр.** Если в графе 1 точка сочленения, он называется односвязным.
**Опр.** Граф называют $i$-связным, если для нарушения его связности необходимо удалить не менее $i$ вершин. $i$ называют числом связности.

## Обходы (Эйлеровы и Гамильтоновы)

**Опр.** Эйлеровым циклом (обходом) в неографе называют цикл, включающий все рёбра графа и проходящий по каждому ребру только один раз

Можно выделить часть графа, в которой существует эйлеров обход.

**Опр.** Граф, содержащий эйлеров цикл, называют эйлеровым

**Теорема Эйлера.** Граф является эйлеровым тогда и только тогда, когда все его вершины имеют чётную степень - сформулирована для неографов

**Доказательство.**

**Необходимость** ($\Rightarrow$).

Пусть связные неограф $G$ эйлеров. Следовательно, цикл в $G$ проходит через каждую вершину. Для каждой вершины верно: цикл входит в вершину по одному ребру, а выходит по другому. Таким образом, каждая вершина инцидентна чётному числу рёбер. Так как эйлеров цикл, согласно определению, содержит все рёбра графа, то степени всех вершин чётны.

**Достаточность** ($\Leftarrow$).

Пусть степени всех вершин связного неографа $G$ чётны. Начнём построение цепи $C_1$ из вершины $x_1$. Попав в очередную вершину $x_i$ по некоторому ребру, всегда возможно выйти из неё по другому ребру, так как степень каждому вершины чётная. Тогда окончание цепи $C_1$ придётся на вершину $x_1$, то есть $C_1$ будет циклом. Если при получении цикла $C_1$ были пройдены все ребра графа $G$, то $G$ - эйлеров граф.

Если были пройдены не все рёбра графа, то будем считать, что пройденные рёбра помечены. Тогда для любой вершины, имеющей непомеченные рёбра среди инцидентных ей рёбер, верно: число непомеченных рёбер чётное. Найдём первую такую вершину $x_i$ и начнём от неё построение цепи $C_2$ по непомеченным рёбрам. В силу чётности степеней всех вершин и чётности числа непомеченных рёбер имеем: $C_2$ заканчивается в вершине $x_i$ и будет циклом. Так как граф связный, то циклы $C_1, C_2$ имеют точку пересечения $x_i$. Если после построения цикла $C_2$ в графе ещё остались непомеченные рёбра, аналогично описанному выше строится цикл $C_3$ и т.д., пока непомеченных рёбер не останется. Так как каждый из построенных циклов $C_1, C_2, \ldots, C_n$ является эйлеровым подграфом графа $G$, то в силу связности графа $G$ получаем: граф $G$ - эйлеров.$\blacksquare$

Для эйлеровых графов существует алгоритм Флёри для построения одного из возможных эйлеровых циклов:

1. Выбираем вершину $x_i$, рассматриваем произвольное ребро и помечаем его как первое, переходим в вершину $x_{i+1}$

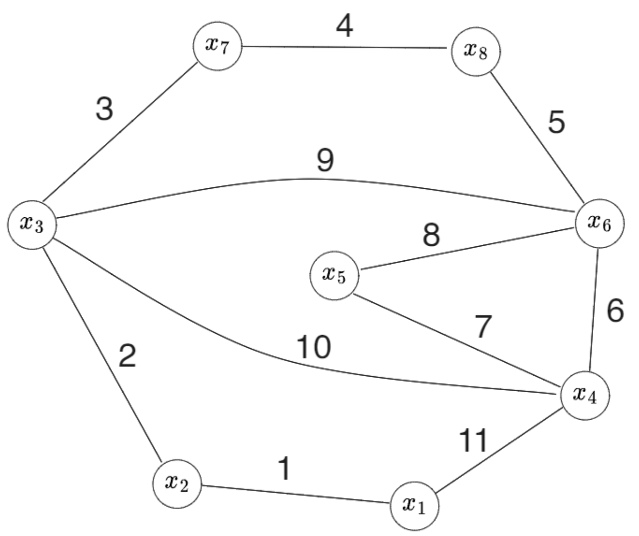
2. Циклически повторяем пункт 1 для $x_{i+1}$, в качестве номеров используем последовательные натуральные числа

3. Находясь в вершине $x_l$ не следует:

   (a) выбирать инцидентное ребро, второй концевой вершиной которого является предыдущая вершина, если есть другие варианты

   (b) выбирать ребро-"перешеек то есть такое ребро, удаление которого нарушит связность графа

*Пример:*



**Теорема.** Связный неограф является эйлеровым тогда и только тогда, когда он может быть представлен объединением циклов, не пересекающихся по рёбрам.

## Гамильтоновость в неографе

**Опр.** Гамильтоновым обходом(циклом) называется в неографе называется обход(цикл), содержащий все вершины и проходящий через каждую из них только один раз.

- Определить гамильтоновость графа сложнее, чем эйлеровость.
- Тут пример с пятиугольниками?

**Задача коммивояжёра.** Есть несколько пунктов, соединённых дорогами разной длины, и требуется обойти все, затратив меньшее количество сил.

Вычислительная сложность задачи нахождения Гамильтонова обхода в графе в общем случае: $O(n!)$

Задача коммивояжёра является **NP-полной**, то есть относится к классу задач, алгоритм решения которых можно применить для похожих задач.

Нет универсального алгоритма построения Гамильтонова цикла. Есть алгоритмы, упрощающие эту задачу при определённых требованиях к начальному графу:

1. Алгоритм поиска Гамильтонова обхода в условиях теорем Дирака и Оре
2. Алгоритм, улучшающий полный перебор за счёт использования динамического программирования
3. Алгоритм, улучшающий полный перебор за счёт применения метода ветвей и границ
4. Из ML: алгоритм поиска ближайшего соседа

Критериев Гамильтоновости столь же простых как критерий эйлеровости не существует, но существует ряд теорем о достаточных условиях Гамильтоновости.

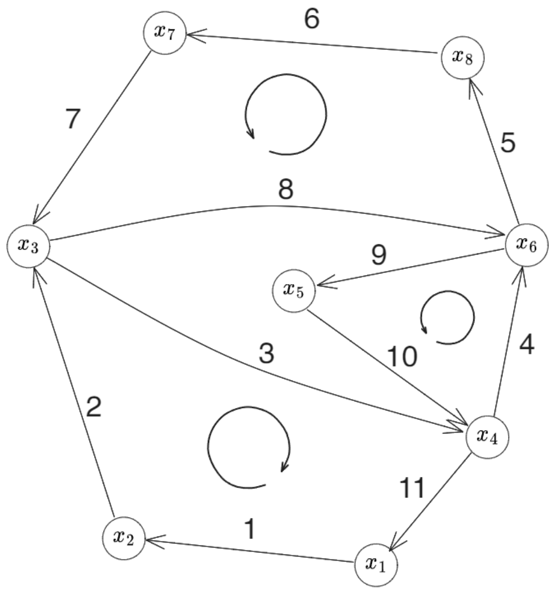
**Теорема Оре.** Дан неограф $G(X, U)$ порядка $n \geq 2$. Если для любой пары вершин $x_i, x_j$ выполняется неравенство $p(x_i) + p(x_j) \geq n$, то $G$ - Гамильтонов граф.

## Эйлеровость в орграфе

Дан орграф $G(X, U)$

1. Если $G$ - эйлеров, то $\forall x \in X$ $p^+(x_i) = p^-(x_i)$
2. Если $G$ - эйлеров, то он является объединением контуров, не пересекающихся по рёбрам.

*Пример:*




**Теорема.** Связный орграф $G(X, U)$ содержит открытый эйлеров путь тогда и только тогда, когда в нём найдётся 2 различных вершины $x, y$ $(x \neq y)$ такие, что $p^-(x) = p^+(x) + 1$ и $p^-(y) = p^+(y) - 1$, а для всякой иной вершины $x_i \in X \setminus x, y$ верно $p^-(x_i) = p^+(x_i)$

### Гамильтоновость в орграфе

Можно доказать гамильтоновость только в частном случае
**Теорема.** (одно из *достаточных* условий)
Дан сильно связный орграф $G(X, U)$ без петель и кратных рёбер порядка $n \geq 2$. Если для любой пары различных несмежных вершин $x_i, x_j$ выполняется неравенство $p(x_i) + p(x_j) \geq 2n - 1$, то орграф $G$ содержит Гамильтонов контур.

Если теорема выполняется, то гарантированно существует гамильтонов обход, а если не выполняется, то он может как быть, так и не быть.

## Деревья

**Опр.** Неориентированное дерево - связный неограф без циклов.
**Опр.** Произвольный неограф без циклов называется лесом.

### Свойства деревьев:

$G(X, U)$ - неориентированные дерево
$|X| = n, |U| = m$

1. $m = n - 1$
2. Если $x_i, x_j \in X$, то их соединяет единственная простая цепь
   - существование цепи следует из связности
   - единственность из отсутствия циклов
3. Если $x_i, x_j \in X$ не смежны, то введение в дерево ребра $(x_i x_j)$ даёт граф, содержащий ровно один цикл
4. Всякое неориентированное дерево содержит по крайней мере 2 концевые вершины
5. **Теорема Кайли.** Число различных деревьев, которые можно построить на $n$ вершинах равно $2^{n-2}$

**Опр.** Орграф $G(X, U)$ называют ориентированным деревом(ордеревом, корневым деревом), если выполняются следующие условия:

- существует ровно одна вершина (корень), не имеющая предшествующих ей: $p^+(x_1) = 0$
- для всех остальных вершин $p^+(x_i) = 1, \forall i \neq 1$

**Опр.** Висячие вершины дерева называются листьями.
**Опр.** Путь из корня в лист называется ветвью.
**Опр.** Длина наибольшей ветви называется высотой дерева.
**Опр.** Расстояние (число рёбер) от корня до вершины называется уровнем этой вершины.
**Опр.** Все вершины одного уровня называются ярусом.
Если из ордерева удалить корень, то оно распадётся на $k$ деревьев $\{T_1, T_2, \ldots, T_k\}$. На этом множестве деревьев можно задать отношение порядка. Если рекурсивно из этих поддеревьев снова удалить корни задать порядок и продолжить этот процесс, пока все поддеревья не станут вершинами, то получим упорядоченное множество всех вершин дерева. Это позволяет использовать деревья для описания иерархий объектов.

*Примеры:*

- Разбор математических выражений
- Файловая система
- Описание сложных программных систем

**Опр.** Если полустепень исхода каждой отличной от листа вершины равна 2 и все листья дерева располагаются в одном ярусе, то дерево называется полным бинарным деревом.

Используя индукцию по высоте, можно доказать, что число листьев полного бинарного дерева высоты $h$ равно $2^h$:

1. $h = 0, n = 2^0 = 1$

2. $h = 1, n = 2^1 = 2$

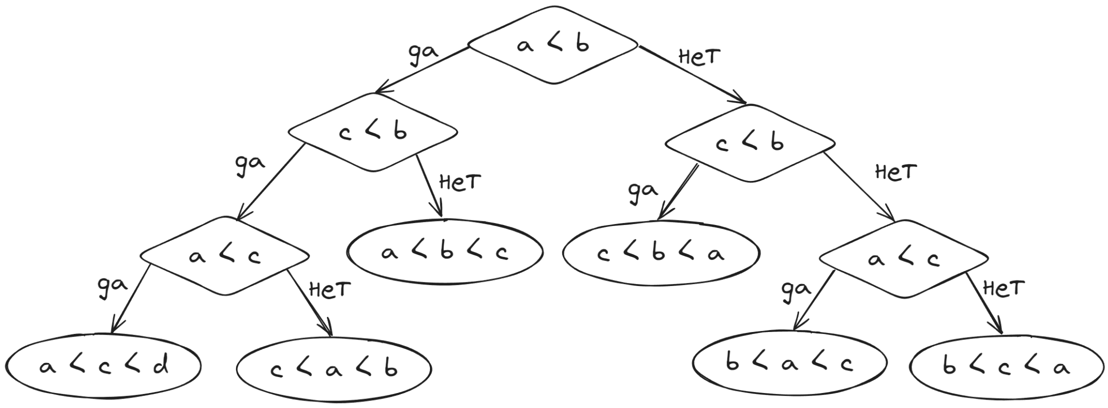
3. $h = k, n = 2^k$

4. Чтобы получить дерево с высотой $h = k + 1$, каждому листу дерева высоты $h = k$ добавить 2 листа, тогда на $k + 1$ уровне получится $2 \cdot 2^k = 2^{k+1}$ листьев

**Теорема.** Произвольное бинарное дерево с $n$ листьями имеет высоту не меньше $\log_2 n$

$\{a_1, a_3, \ldots, a_n\}$ - последовательность из $n$ элементов. Нужно ввести на ней отношение порядка - это задача сортировки.

В общем случае придётся рассмотреть $n!$ перестановок. Все сравнения приведут к построению бинарного дерева (дерева решений).




Дерево решений имеет $n!$ вершин, тогда сортировка даёт пути от корня к каждому листу. Число операций пропорционально высоте дерева - $\log_2(n!)$

Приближение для $n! \approx \sqrt{2\pi n}\left(\frac{n}{e}\right)^n$

Тогда получим минимально возможную сложность для алгоритма классической сортировки $O(n \log n)$

В общем случае ордерево - это связный, но не сильно связный орграф. Компонентами связности ордерева являются поддеревья на множестве вершин, образующие путь из корня в некоторый лист. - ???

## Поиск в глубину

$i$ - номер яруса
$k$ - номер ребра

0. $i = 0, k = 0$

1. Выбираем вершину, смежную текущей из яруса $i = i + 1$ по непомеченному ребру, новую вершину объявляем текущей, пройденное ребро помечаем числом $k = k + 1$

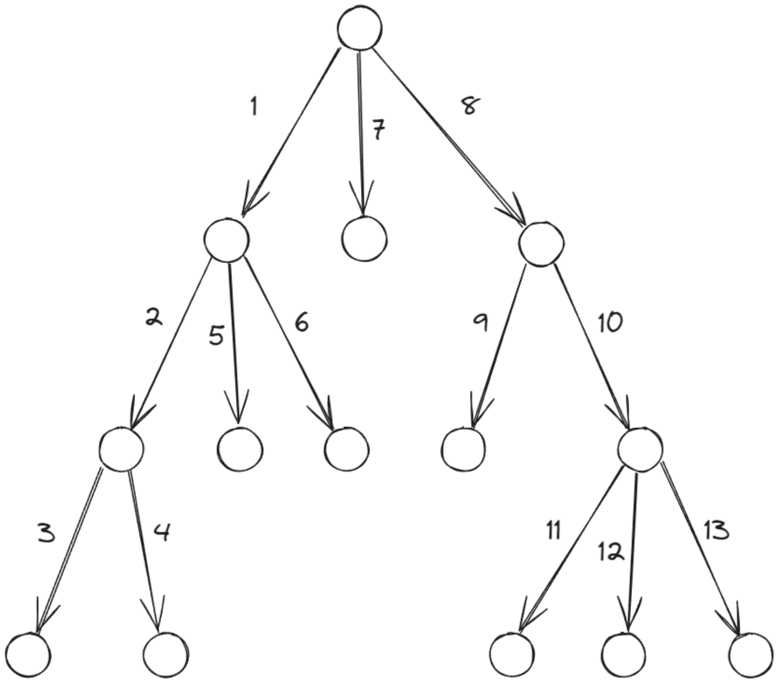
2. Повторяем пункт 1 для текущей вершины

3. Если текущая вершина является листом, то возвращаемся в смежную ей вершину ярусом выше $i = i - 1$ и переходим к пункту 1

4. Если для текущей вершины нет ни одного инцидентного ей непомеченного ребра, возвращаемся в смежную ей вершину ярусом выше $i = i - 1$

5. Алгоритм останавливается, когда помечены все рёбра и $i = 0$

*Пример:*




Поиск в глубину - обход дерева по ярусам
Оба алгоритма имеют одинаковую вычислительную сложность при обходе всех вершин
Если требуется найти конкретную вершину/ребро, то:

- Поиск в глубину применяется для "широких" деревьев
- Поиск в ширину применяется для "узких" деревьев

Аналогичные подходы можно применять также для неориентированных деревьев

## Остовы

**Опр.** Граф $G'(X', U')$ называется остовным подграфом графа $G(X, U)$ $(X' = X, U' \subseteq U)$. Если остовный подграф является деревом, его называют остовным деревом или просто остовом.

То есть остов - это дерево построенное на множестве вершин графа, множество рёбер которого является подмножеством множества вершин исходного графа.

**Теорема.** Число рёбер произвольного неографа $G$, которое нужно удалить для получения остова, не зависит от порядка их удаления и равно $m - n + k$, где $m = |U|, n = |X|$, $k$ - число компонент связности.

- $\nu(G) = m - n + k$ - циклический ранг (циклическое число), число рёбер, которые нужно удалить, чтобы получить остов
- $\nu^*(G) = n - k$ - коциклический ранг (коранг), число рёбер в остове

### Задача Штейнера

На плоскости задано $n$ точек, нужно соединить эти точки отрезками прямых так, что сумма длин этих отрезков была минимальной. Разрешается добавлять точки и длина определяется весом ребра.

#### Постановка задачи в теории графов:

Дан граф $G_0(X_0, \varnothing)$ (задаёт $n$ точек). Требуется найти неограф $G(X, U)$ такой, что:

- каждому ребру соответствует некоторое число (вес) $\geq 0$
- $G(X, U)$ - дерево
- множество вершин $X_0$ является подмножество нового множества вершин $X$: $X_0 \subseteq X$
- сумма весов рёбер дерева $G$ должна быть минимальной из возможных

$G$ - неориентированное дерево, потому что каждую пару вершин должна соединять только одна цепь, иначе сумма весов будет не минимальной, так как, если существуют 2 цепи, возникают альтернативные пути и неоднозначность решения.

Решения задачи Штейнера в общем случае не существует.

Решения при некоторых ограничениях: алгоритм Прима и алгоритм Краскала.

## Алгоритм Прима

Ограничение: в ходе работы алгоритма не меняется множество вершин. В результате работы получается остовное дерево макс./мин. веса.

Дан граф $G(X, U, \Omega)$ ($\Omega$ - множество весов)

$|U| = |\Omega|$, $\Omega = \{\omega_1, \ldots, \omega_n\}$

В основе алгоритма лежит расширение исходного поддерева до остовного. Алгоритм итерационный, на каждой итерации число вершин дерева увеличивается не менее, чем на 1.

Множество вершин $X$ разбивается на 2 подмножества:

$X = X_1 \cup X_2, X_1 \cap X_2 = \varnothing$

Введём понятие расстояния между $X_1$ и $X_2$:

$d(X_1, X_2) = \min\{\omega(x_i, x_j) : x_i \in X_1, x_j \in X_2\}$

**Ход работы алгоритма:**

1. $X_1 = \{x_1\}, X_2 = X \setminus X_1, U = \varnothing$
2. Находим ребро $(x_i, x_j) : x_i \in X_1, x_j \in X_2, \omega(x_i, x_j) = d(X_1, X_2)$ и полагаем $X_1 = X_1 \cup \{x_j\}$, $X_2 = X_2 \setminus \{x_j\}$, $U' = U' \cup \{(x_i), x_j\}$
3. Если $X_1 = X$, то конец алгоритма, иначе повторить шаг 2.

Если требуется найти остовное дерево наибольшего веса, то изменяем формулу расстояния (с min на max).

## Паросочетания

Например, для графа из последовательно соединённых 6 вершин: $x_1 - x_2 - x_3 - x_4 - x_5 - x_6$ возможны паросочетания:

- $\{(x_1, x_2), (x_3, x_4), (x_5, x_6)\}$
- $\{(x_2, x_3), (x_4, x_5)\}$

**Опр.** Максимальным паросочетание называется паросочетание, в которое больше нельзя добавить ребро.

**Опр.** Паросочетание наибольшей мощности называется паросочетание из наибольшего возможного количества рёбер.

**Опр.** Совершенным пваросочетанием называется паросочетание, охватывающее все вершины.

Например, распределение задач между работниками может производиться как поиск паросочетаний в двудольном графе.