ДИСКРЕТНАЯ МАТЕМАТИКА

G.

ЭЛЕМЕНТЫ

ТЕОРИИ ГРАФОВ

G.1. Направленные графы

Различают направленные и ненаправленные графы.

Направленным графом ***G*** называется пара $G = < V, E >$, где

$V = \{ v_i \mid i \in \{1, \dots , n\}\}$ есть непустое множество вершин графа

$E = \{ e_k \mid k \in \{1, \dots , m\}\}$ есть множество дуг графа

$$e_k = < v_s , v_t > , \quad s, t \in \{ 1, \dots , n \}$$

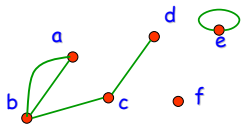
Направленный граф является одной из форм представления отношений и рассмотрена в соответствующем разделе теории множеств.

G.2. Ненаправленные графы

В далнейшем в рамках нашего курса мы рассматриваем симметричные (ненаправленные) графы.

Графом *G* называется пара < *V*(*G*), *E*(*G*) > , где *V* (*G*) непустое множество элементов, называемых *вершинами*, а *E*(*G*) - конечное семейство неупорядоченных пар элементов из (не обязательно различных), называемых *рёбрами*.

Заметим, что употребление слова “семейство” говорит о том, что допускаются кратные рёбра (мультиграф).

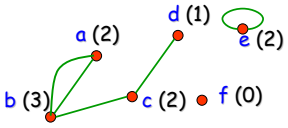


Степень вершины. Лемма о рукопожатиях

Степенью вершины v называется число рёбер $\rho(v)$, инцидентных v (при вычислении степени вершины петлю учитывают 2 раза).

Вершина степени 0 называется *изолированной* вершиной.

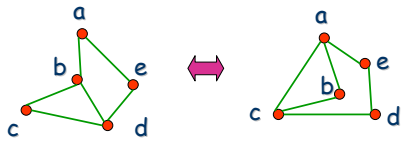
Вершина степени 1 называется *висячей* (*концевой*) вершиной.



Лемма о рукопожатиях : если сложить степени всех вершин графа, то получится чётное число - равное удвоенному числу рёбер.

Изоморфизм графов.

Два графа G_1 и G_2 называются изоморфными, если существует взаимно однозначное соответствие между множествами их вершин, обладающее тем свойством, что число рёбер, соединяющих любые две вершины в G_1, равно числу рёбер, соединяющих соответствующие две вершины в G_2.



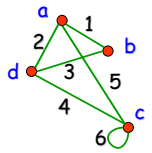
Изоморфизм f от левого графа к правому:
f(a) = e, f(b) = a, f(c) = b, f(d) = c, f(e) = d.

Матрица инциденций

Две вершины v и w грфа G называются *смежными*, если существует соединяющее их ребро; при этом вершины v и w нзываются инцедентными этому ребру (а ребро - инцедентным этим вершинам).

Матрицей инциденций простого графа G с множеством вершин $\{v_1, v_2, ..., v_n\}$ и множеством рёбер $\{e_1, e_2, ..., e_m\}$ называется матрица $A = (a_{pq})$ размера n×m, в которой элемент $a_{pq} = 1$, если вершина v_p инцидентна ребру e_q и $a_{pq} = 0$ в противном случае.

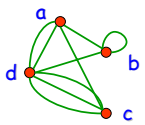
$$M = \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ a \\ b \\ c \\ d \end{array}$$



Матрица смежности. Простой граф

Матрицей смежности графа G с множеством вершин $\{v_1, v_2, ..., v_n\}$ называется матрица $A = (a_{pq})$ размера n×n, в которой элемент a_{pq} равен числу рёбер в G, соединяющих v_p и v_q.

$$A_G = \begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{bmatrix} 0 & 1 & 1 & 2 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 3 \\ 2 & 1 & 3 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ a \\ b \\ c \\ d \end{array}$$



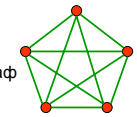
Графы, не содержащие петель и кратных рёбер, мы будем называть *простыми.*

У простого графа a_{pq} =1 для всех p, q.

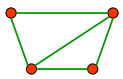
Подграф

Подграфом графа G называется граф, все вершины которого принадлежат $V(G)$, а все рёбра принадлежат E(G).

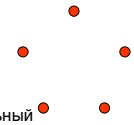
G_1
Полный граф



G_2
Подграф графа G_1

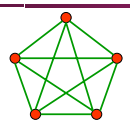


G_3
Максимальный пустой подграф графа G_1

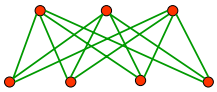


G_4
Подграф графов G_1, G_2, и G_3; максимальный пустой подграф графа G_2

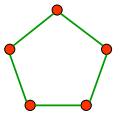
Некоторые типы графов



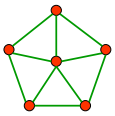
K_5
полный граф



$K_{3,4}$
двудольный граф



C_5
циклический граф



W_5
граф “колесо”

Укладка графов. Планарный граф.

Жордановой кривой на плсскости называется непрерывная кривая, не имеющая самопересечений.

Говорят, что граф может быть уложен в данном пространстве, если он изоморфен некоторому графу, изображенному в этом пространстве при помощи точек, представляющих вершины графов, и жордановых кривых, представляющих рёбра, причём эти кривые не пересекаются друг с другом.

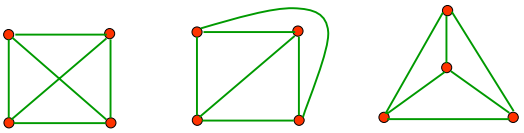
Граф, который может быть уложен на плоскости, наывается *планарным графом*.

Например, графы K_5 и $K_{3,3}$ не являются планарными.

Планарный и плоский графы

Плоским называется граф, изображённый на плоскости так, что никакие его два ребра не пересекаются.

Планарный граф изоморфен плоскому.



Например, эти три графа планарны, но только два из них плоские.

Маршрут. Связность графов.

Маршрутом называется последовательность ребер графа такая, что два соседних ребра имеют общую вершину. Маршрут называется *цепью*, если все его рёбра различны и *простой цепью*, если все вершины различны (кроме, может быть, начальной и конечной вершин). Замкнутая простая цепь называется *циклом*.

Граф называется *связным*, если для любых его двух вершин v и w существует простая цепь из *v* в *w*. Несвязный граф состоит из нескольких *компонент*.

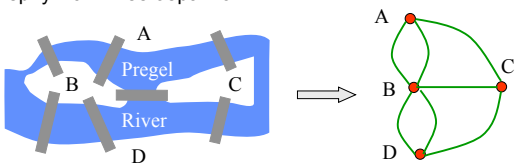
Разрез

Разделяющим множеством связного графа называется такое множество его рёбер, удаление которого приводит к несвязному графу.

Разрезом называется такое разделяющее множество, никакое собственное подмножество которого не является разделяющим. Если разрез состоит из единственного ребра, то это ребро называется *мостом* или *перешейком*.

G.3. Задача о кенигсбергских мостах

Найти такую точку города, выйдя из которой можно пройти по всем мостам города по одному разу и вернуться в неё обратно.

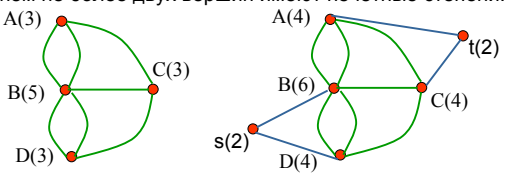


Граф называется *эйлеровым*, если существует замкнутая цепь содержащий по одному разу каждое ребро заданнрго графа (эйлеров цикл). Если снять ограничение на замкнутость цепи, то граф называется *полуэйлеровым*.

Задача о кенигсбергских мостах

Лемма. Если степень каждой вершины графа не меньше двух, то граф содержит цикл.

Теорема. Граф эйлеров, если и только если он связен и степени всех его вершин чётны.
Связный граф полуэйлеров, если и только если в нём не более двух вершин имеют нечётные степени.



Граф справа в отличие от графа слева является эйлеровым графом.

Алгоритм построения эйлеровой цепи

Алгоритм построения эйлеровой цепи (алгоритм Флёри).

Выходя из произвольной вершины, идём по рёбрам графа произвольным образом, соблюдая лишь следующие правила:

1) стираем рёбра по мере их прохождения и стираем также изолированные вершины, которые при этом образуются;

2) на каждом этапе идём по мосту только тогда, когда нет другой возможности.

G.4. Кратчайший путь. Алгоритм Дейкстры.

Пусть G - взвешенный граф, причём все веса рёбер - неотрицательные числа. Пусть Z - какой-нибудь путь, соединяющий вершины a и z; *длиной пути* Z из a в z будем считать сумму весов всех входящих в данный путь рёбер.

Алгоритм Дейкстры основан на двух идеях:

1) присвоении вершинам графа меток и правиле пересчёта меток, при этом окончательные метки - длины соответствующих кратчайших путей;

2) экстремальном свойстве кратчайшего пути, состоящем в следующем: если кратчайший путь из вершины x в вершину y проходит через вершину b, то его отрезок от вершины x до вершины b - кратчайший путь от x до b, а его отрезок от b до y - кратчайший путь от b до y.

Алгоритм Дейкстры.

procedure Dijkstra(G: взвешенный связный простой граф с вершинами $a = v_0, v_1, \ldots, v_n = z$ и положительными весами $w(v_i, v_j)$, где $w(v_i, v_j) = \infty$, если $\{v_i, v_j\}$ не является ребром графа G)

for i := 1 ***to*** n

$L(v_i) := \infty$

L(a) := 0

S := ∅

{теперь метки инициализироанны таким образом, что метка вершины a есть ноль, метки остальных вершин есть ∞, а строимое множество вершин S пока есть пустое множество}

Алгоритм Дейкстры.

while $z \notin S$

begin

u := вершина не принадлежащая S с минимальной меткой L

$S := S \cup \{u\}$

for все вершины v не принадлежащие S

if L(u) + w(u, v) < L(v) ***then*** L(v) := L(u) + w(u, v)

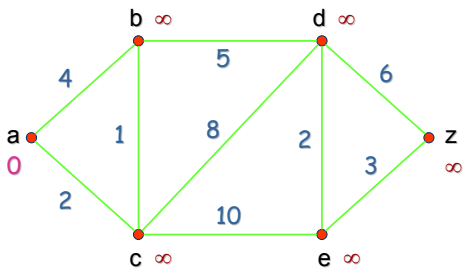
{выбирается вершина с минимальной меткой, вкключается в множество S и обновляются метки не принадлежащие S}

end

{L(z) = длина кратчайшего пути от a к z}

Пример. Начальная установка меток

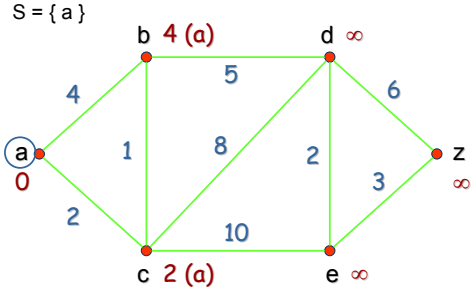
Вначале вершине v присваивается окончательная метка, а каждой из остальных присваивается временная метка ∞.



Шаг 1

На каждом шаге одной вершине с временной меткой присваивается окончательная метка и поиск продолжается (пересчитываются метки).

S = { a }

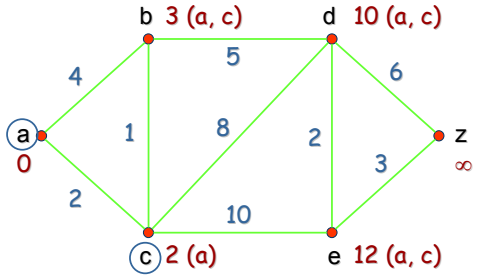


Шаг 2

Выбирается вершина **c** имеющая наименьшую метку

S = { a, c }

Пересчитываются метки вершин, для которых существуют рёбра, идущие из S (временные метки)

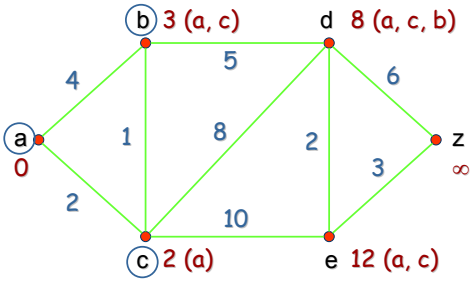


Шаг 3

Выбирается вершина **b** имеющая наименьшую метку

S = { a, c, b }

Пересчитываются метки вершин, для которых существуют рёбра, идущие из b.

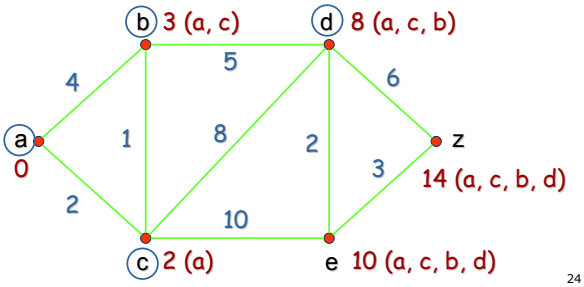


Шаг 4

Выбирается вершина **d** имеющая наименьшую метку

S = { a, c, b, d }

Пересчитываются метки вершин, для которых существуют рёбра, идущие из d.

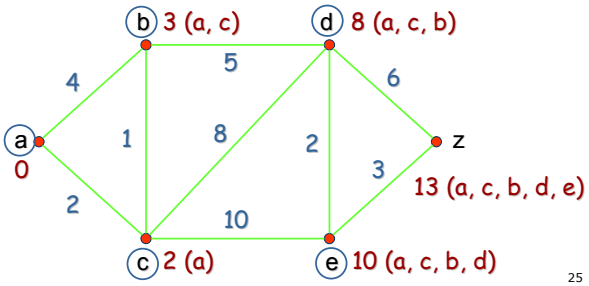


Шаг 5

Выбирается вершина **e** имеющая наименьшую метку

S = { a, c, b, d, e }

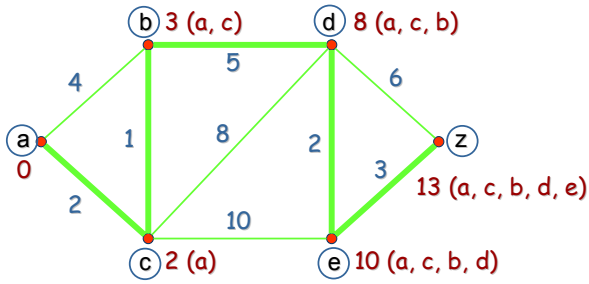
Пересчитываются метки вершин, для которых существуют рёбра, идущие из e.



Останов алгоритма

S = { a, c, b, d, e, z }

Кратчайший путь: a, c, b, d, e, z



G.5. Деревья

Лесом называется граф, не содержащий циклов.

Связный лес называется *деревом*.

Пусть граф *G* имеет *n* вершин. Тогда следующие утверждения эквивалентны:

1) *G* является деревом;

2) *G* не содержит циклов и имеет *n-1* рёбер;

3) *G* связен и имеет *n-1* рёбер;

4) *G* связен и каждое его ребро является мостом;

5) любые две вершины графа *G* соединены ровно одной простой цепью;

6) *G* не содержит циклов, но добавляя к нему любое новое ребро, мы получаем ровно один цикл.

Деревья

Сумма степеней всех вершин дерева равна 2n -2.

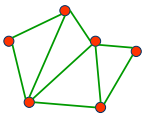
В связном графе *G* удаление одного ребра, принадлежащего некоторому выбранному циклу, не нарушает связности оставшегося графа. Применим эту процедуру к одному из оставшихся циклов, и так до тех пор, пока не останется ни одного цикла. В результате получим дерево, связывающее все вершины рафа *G*; оно называется *остовным деревом* графа *G*.

Число удалённых в этой прцедуре рёбер называется *циклическим рангом* (или *цикломатическим числом*) графа и равно в общем случае для графа *G* с *n* вершинами, *m* рёбрами и *k* компонентами:

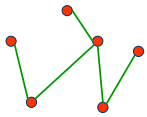
$$\gamma(G) = m - n + k$$

Деревья

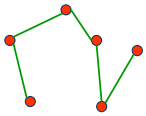
Являются ли эти графы деревьями ?



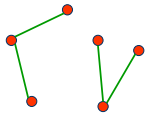
Нет



Да



Да



Нет. Это лес.

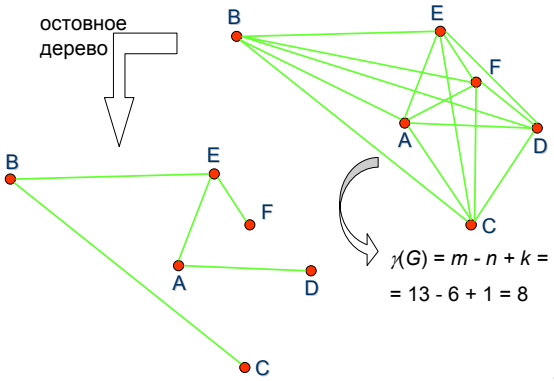
Деревья

В связном графе *G* удаление одного ребра, принадлежащего некоторому выбранному циклу, не нарушает связности оставшегося графа. Применим эту процедуру к одному из оставшихся циклов, и так до тех пор, пока не останется ни одного цикла. В результате получим дерево, связывающее все вершины рафа *G*; оно называется *остовным деревом* графа *G*.

Число удалённых в этой прцедуре рёбер называется *циклическим рангом* (или *цикломатическим числом*) графа и равно в общем случае для графа *G* с *n* вершинами, *m* рёбрами и *k* компонентами:

$$\gamma(G) = m - n + k$$

Деревья



$\gamma(G) = m - n + k =$

$= 13 - 6 + 1 = 8$

Перечиление деревьев

Помеченный граф с n вершинами - это граф, у которого все вершины “помечены” целыми числами от 1 до n .

Формально имеем распределение меток как взаимно однозначное соответствие между множеством вершин и множеством {1, 2,..., n} ; тогда помеченным графом называется пара < G, φ > , где G - граф, а φ - распределение меток в G .

Теория перечисления графов занимается подсчётом числа неизоморфных графов. Два помеченных графа < G_1, φ_1 > и < G_2, φ_2 > называются изоморфными, если существует изоморфизм между G_1 и G_2, сохраняющий распределение меток в этих графах.

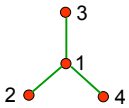
Перечисление деревьев

Теорема Кэли. Существует ровно n^{n-2} различных помеченных деревьев с n вершинами.

Например, число всех неизоморфных помеченных деревьев с четырьми вершинами равно 16.

1/2 * 4! + 4 = 16 =

$= 4^{4-2}$

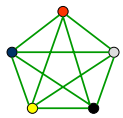


Различными считаются деревья, не переводимые друг в друга изоморфизмом, сохраяющим нумерацию вершин.

G.6. Задачи раскраски графов

Графы, рассмартиваемые в этом разделе, являются неориентированными и не имеют петель.
В общем случае задача раскраски графов формулируется следующим образом.
Задан неориентированный граф ***G*** с множеством вершин ***V*** и множеством ребер ***E*** . Требуется раскрасить его вершины в минимальное число цветов так, чтобы соседние вершины были окрашены в различные цвета.
Минимально необходимое число красок называется *хроматическим числом* графа.

Например, для раскраски данного графа требуется пять красок.



G.6.1. Поиск минимальных разбиений

Часто решение логических задач связывается с поиском некоторой совокупности подмножеств заданного множества, удовлетворяющей определённым требованиям, и оптимальной в том или ином смысле, например минимальной по числу выбранных подмножеств.

Приведём одну из формулировок подобной задачи.
Задано некоторое множество *V* . На множестве *V* определено отношение несовместимости. Требуется найти такое разбиение множества *V* на минимальное число подмножеств (блоков) так, чтобы любые два элемента принадлежащие одному блоку не находились бы в отношении несовместимости.

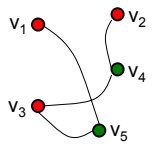
Поиск разбиения сводится к раскраске графа описывающего отношение несовместимости.

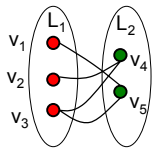
Задача размещения отдыхающих

Пример. Задано отношение несовместимости на множестве

$V = \{v_1, v_2, v_3, v_4, v_5\}$

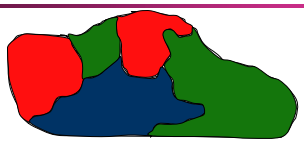
	v_1	v_2	v_3	v_4	v_5
v_1	0	0	0	0	1
v_2	0	0	0	1	0
v_3	0	0	0	1	1
v_4	0	1	1	0	0
v_5	1	0	1	0	0



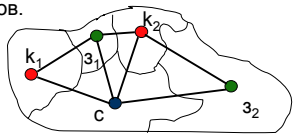


Получаем разбиение: $V = \{ \{ v_1, v_2, v_3 \}, \{ v_4, v_5 \} \}$

G.6.2. Задача раскраски карт



Задача раскраски карт сводится к задаче раскраски планарных графов.



Теорема. Для оптимальной раскраски планарных графов достаточно четырёх цветов.
(1977, K. Appel, W. Haken, J. Koch)

G.6.3.Решение задача расраски графов

Решение путем сведения к задаче о кратчайшем покрытии.

Данная задача сводится к поиску минимального числа максимальных пустых подграфов, которые в своей совокупности содержат все вершины заданного графа.

Утверждение. Вершины, которые не принадлежат пустому подграфу, должны покрывать все ребра заданного графа (действительно, иначе обе вершины непокрытого ребра должны принадлежать этому подграфу, что невозможно).

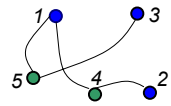
На основании приведенного утверждениия построение пустого подграфа можно свести к поиску подмножества вершин покрывающего **все** ребра заданного графа.

Алгоритм

1. Найти все максимальные пустые подграфы графа *G* .
2. Построить матрицу бннарного отношения принадлежности вершин графа найденным подграфам.
3. Найти кратчайшее покрытие этой булевой матрицы.
4. Перейти к разбиению множества вершин (т.е. устранить возможные пересечения между множествами вершин выбранных максимальных пустых подграфов).

Пример

Дан граф:



1.Построение максимальных пустых подграфов.

$f_e(1, 2, 3, 4, 5) = (1 \vee 5) \,\&\, (2 \vee 4) \,\&\, (3 \vee 4) \,\&\, (3 \vee 5) =$

$= (1\&2\&3) \vee (1\&3\&4) \vee (2\&3\&5) \vee (4\&5)$

Минимальные подмножества вершин покрывающие все ребра будут:

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

Взяв дополнения найденных подмножеств, получим все максимальные пустые подграфы:

$G_1 = \{4, 5\}, G_2 = \{2, 5\}, G_3 = \{1, 4\}, G_4 = \{1, 2, 3\}$

Пример

2. Построение матрицы бинарного отношения принадлежности вершин графа найденным подграфам.

	1	*2*	*3*	*4*	*5*
G_1	0	0	0	1	1
G_2	0	1	0	0	1
G_3	1	0	0	1	0
G_4	1	1	1	0	0

3. Решение задачи покрытия.

Минимальное покрытие будет: **{**G_1, G_4**}**

Таким образом имеем решение: **{{*4, 5*}, {*1, 2, 3*}}**