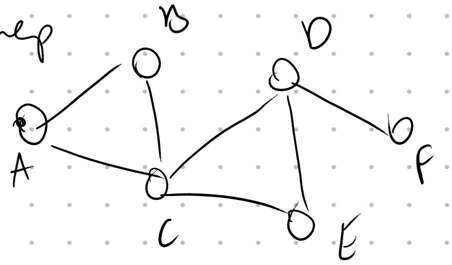
В прошлый раз: (по всем верш) гамильтоновы / (по всем рёбрам) эйлеровы
пути / циклы

Длины путей в граф.

Опр Длина пути в графе – кол-во рёбер в пути.

Пример



ABCDF – путь от A до F
– длина 4 (4 ребра)
ACEDF – длина 4
ACDF – длина 3
ABCEDF – длина 5.

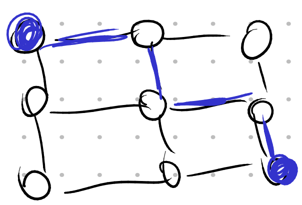
Опр Расстояние между вершинами – мин длина пути между вершинами или $+\infty$, если пути нет.

Обозначение $d(X,Y)$ – расстояние от X до Y.

Пример $d(A,F)=3$

Опр. Диаметр графа – макс расстояние между вершинами графа.

Пример в примере выше диаметр $=3$ (достигается на AF)



– диаметр 4

все другие расстояния ≤ 4

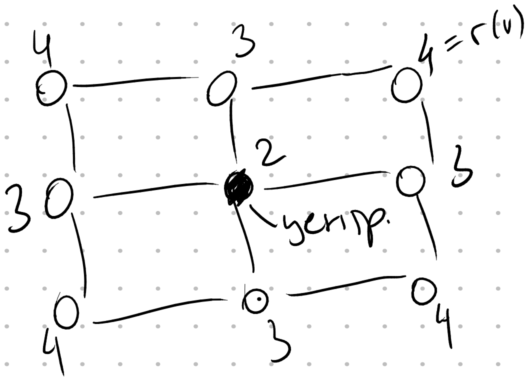
Опр. Для каждой вершины графа $G=(V,E)$ можно посчитать max расстояние до других вершин

$r(v) := \max \{ d(v,s) \mid s \in V \}$

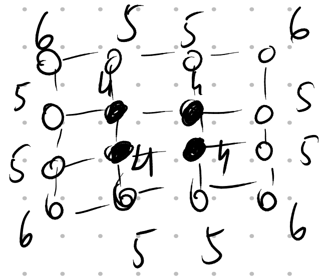
Радиус:

$r(G) = \min\{r(v) \mid v \in V\}$

те вершины, на которых достигается min — это центр.



: $r(G) = 2$ — радиус графа.

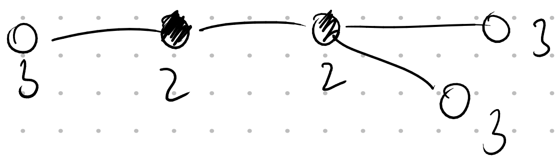


— 4 центра

$r(G) = 4$

центров может быть много

или

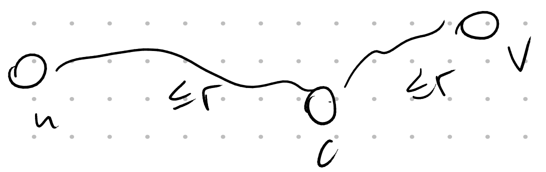


$r(G) = 2$

2 центра

Утв. $G = (V, E)$. $d(G) \le 2r(G)$

Д-во. $\exists$ c — центр графа. $u, v \in V$

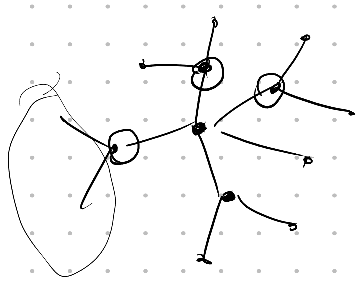


$d(c,u) \le r$

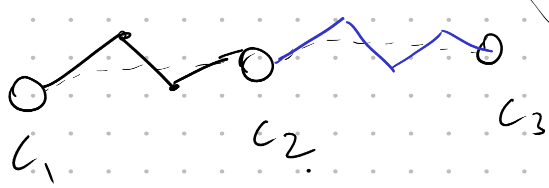
$d(c,v) \le r$

$\Rightarrow d(u,v) \le 2r \Rightarrow d(G) = \max\limits_{u,v} d(u,v) \le 2r$ ■

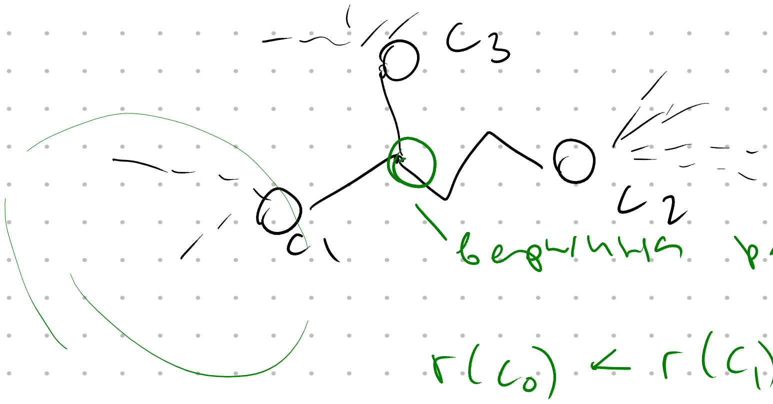
Утв. В дереве ≤ 2 центров.



$\exists$ их 3:



построим пути между c_1 c_2, потом c_2 c_3 (в дереве ровно 1 путь между верш.)



вершина развилки c_0:

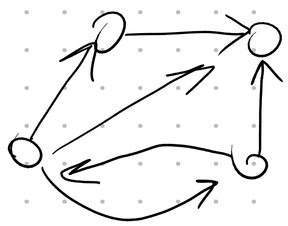
$$r(c_0) < r(c_1) = r(c_2) = r(c_3) = r(G) = r.$$

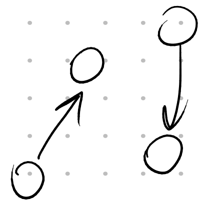
Замечание. Будем дальше иногда использовать ориентированные графы. $G = (V, E)$

(рёбра в ор. графе иногда называют дугами)

$E \in \{(u,v)$ – упорядоченная пара$\}$

Пример

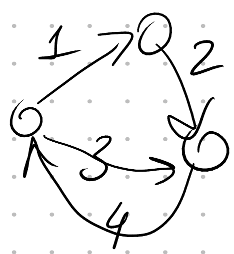


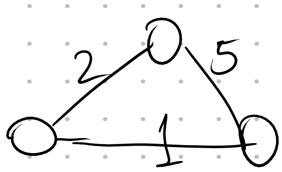


Замечание. У рёбер будут веса

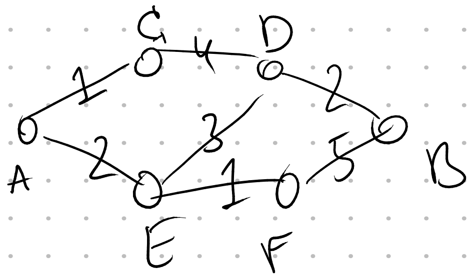
$G = (V, E)$ вес – это $f: E \to \mathbb{R}$

т.е. у число каждому ребру





Расстояние на графе с весами считается как min Σ весов по всем путям



$d(A,B) = ?$

$d(ACDB) = 1+4+2 = 7$

$d(ACDEFB) = 1+4+3+1+5 = 14$

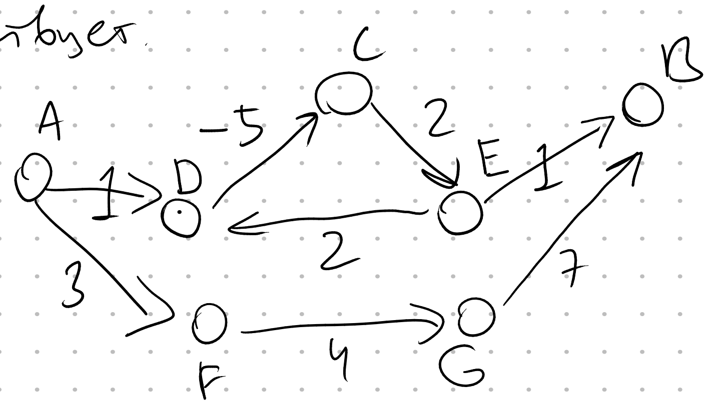
$d(AEFB) = 2+1+5 = 8$

$d(AEDB) = 2+3+2 = 7$

$\min = 7$

$\Rightarrow d(A,B) = 7.$

Замечание: расстояние во взвешенном графе не всегда существует.



$d(F,G) = 4$

$d(G,F) = +\infty$

$d(A,B) = ?$

$d(ADCEB) = 1-5+2+1 = -1$

$d(ADCEDCEB) = 1-5+2+\underbrace{2-5+2}_{=-1}+1 = -2$

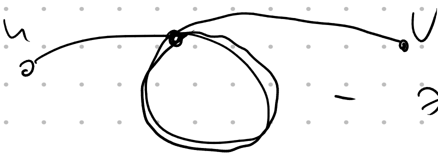
и т.д. $\min = -\infty$

Утв. В графе есть все расстояния $\Leftrightarrow$ в графе нет цикла отр. длины.

Д-во. Если есть цикл < 0

$\Rightarrow$ $\forall$ две вершины этого цикла не имеют расстояния (или $-\infty$)

Если нет расстояния, т.е. для u, v есть пути сколь угодно маленькие, $\exists$ есть путь длиннее $n = |V|$ рёбер. $\Rightarrow$ повтор ~~вершин~~ в пути (без нол* циклов)

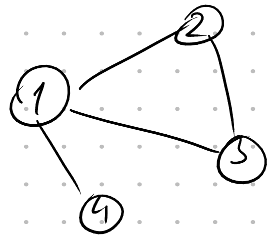


- это и будет отр. цикл

Как хранить граф в компьютере (представление графа в компьютере).

1. Матрица смежности: таблица вершин × вершины

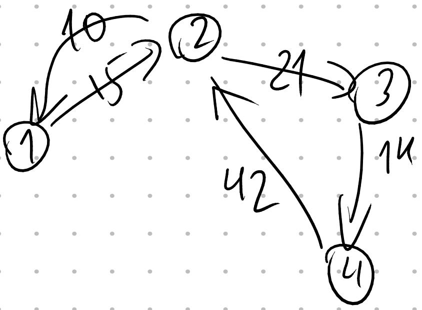
$$a(i,j) = \begin{cases} 0 & \text{если нет ребра} \\ 1 & \text{если есть ребро} \end{cases}$$



$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 1 & 1 & 1 \\ 2 & 1 & 0 & 1 & 0 \\ 3 & 1 & 1 & 0 & 0 \\ 4 & 1 & 0 & 0 & 0 \end{array}$$

- симметрична для неориентированного графа.

Для графов с весами $a(i,j)$ = вес ребра ij или $+\infty$, если нет



$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & +\infty & 15 & +\infty & +\infty \\ 2 & 10 & +\infty & 21 & +\infty \\ 3 & +\infty & +\infty & +\infty & 14 \\ 4 & +\infty & 42 & +\infty & +\infty \end{array}$$

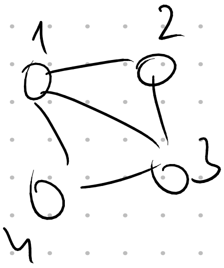
объем памяти: $n^2 = |V|^2$

② Списки смежности

- для каждой вершины храним список соседей

Пример 1: 2(15)   2: 1(10), 3(21)   3: 4(14)   4: 2(42)

Пример



1: 2 3 4

2: 1 3

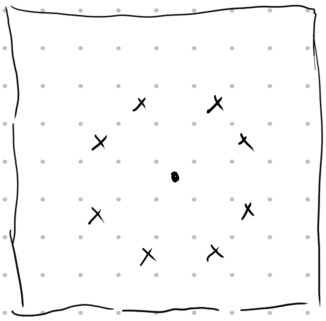
3: 2, 4

4: 1 3

Память $\approx |E|$ кол-во рёбер

③ неявные способы   умеем вычислять всех соседей $\forall$ вершины

Пример. Задача обход конём шахматной доски



граф: вершины = клетки 64 шт
рёбра - вершины через ход коня.

можно для ∀ клетки (вершины) посчитать, куда можно попасть

Задача обхода конём = гамильтонов цикл в этом графе.

Задача дано две вершины u, v, найти $d(u,v)$ и путь, на котором достигается это расстояние.

Замечание Оказывается, что найти путь от u до v это то же самое, что найти путь от u до всех вершин.

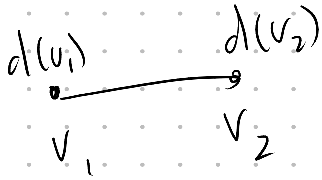
Алгоритм Форда-Беллмана. Дано $G = (V, E)$, f-вес $u \in V$, найти расстояния $d(u,v)$ для $\forall v \in V$,

будем писать $d(v) = d(u,v)$ т.к. u не меняется.

Будем хранить в массиве d текущие найденные расстояния. В начале

$$d(u) = 0 \qquad d(v) = +\infty \quad \text{если } v \neq u$$

Релаксация ребра $e = (v_1, v_2)$



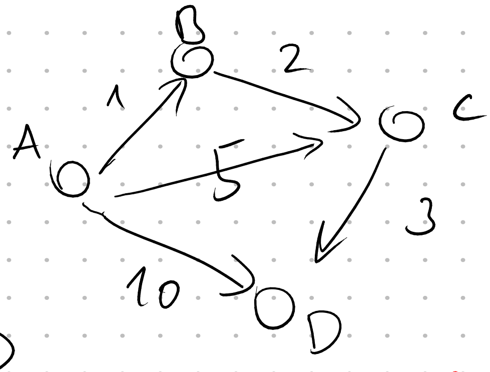
если $d(v_1) + f(v_1, v_2) < d(v_2) \Rightarrow d(v_2) := d(v_1) + f(v_1, v_2)$

Алгоритм: Повторить $n-1$ раз:
перебрать все рёбра e и каждое релаксировать

(в неор. графе $\circ\!-\!\!-\!\circ$ = $\circ \rightleftarrows \circ$, т.е. две релакс. на ребро)

Пример

$n=4$ (4 вершины)



рёбра

A: B(1) C(5) D(10)

B: C(2)

C: D(3)

D:

		A	B	C	D
d:		0	+∞	+∞	+∞
шаг1:	AB	0	1	+∞	+∞
	AC	0	1	5	+∞
	AD	0	1	5	10
	AB:	0	1	3	10
	CD:	0	1	3	6
шаг2:	AD:	//			
	AC	//			
	AD	//			
	AB	//			
	CD	//			
шаг3:	AB	//			
	AC	//			
	AD	//			
	AB	//			
	CD	//			

$\underset{0}{A} \xrightarrow{1} \underset{+\infty}{B}$

$0+1 < +\infty$

$\underset{1}{B} \xrightarrow{2} \underset{5}{C}$ $\quad 1+2<5$

$\underset{3}{C} \xrightarrow{3} D$ 10 $\quad 3+3<10 \quad 6$

$d(A)=0$

$d(B)=1$

← ответ $d(C)=3$

$d(D)=6$

Время работы $\approx |V| \cdot |E| \le |V|^3$