Графы.

Источники

- Аляев Ю. А., Тюрин С. Ф. Дискретная математика и математическая логика.
- Андерсон Дж. Дискретная математика и комбинаторика.
- Хаггарти Р. Дискретная математика для программистов.

Задание графа

Граф

Граф – упорядоченная пара множеств: множества элементов, называемых вершинами графа $V = \{v_1, v_2, \dots, v_n\}$, и множества его двухэлементных подмножеств $E = \{\{v_i, v_j\} | v_i \in V, v_j \in V, i \neq j\}$, называемого множеством **рёбер** графа.

$G = G(V, E) = (V, E)$

Говорят, что вершины v_i, v_j связаны ребром e_k, если $\exists\left(e_k = \{v_i, v_j\}\right) \in E$.

В таком случае вершины v_i, v_j называются **смежными** (соседними).

Смежными также называют рёбра , имеющие общую вершину $e_k = \{v_i, v_j\}$, $e_m = \{v_i, v_l\}$.

При этом говорят, что вершина v_i (v_j) и ребро e_k **инцидентны** друг другу.

Орграф

Ориентированным графом (орграфом) называется множество вершин V и множество упорядоченных пар элементов V, называемых **дугами** орграфа.

$$G = (V, E)$$

$$E = \{(v_i, v_j) | v_i \in V, v_j \in V\} \; (E \subseteq A^2)$$

Если $e_k = (v_i, v_j)$, то v_i называется **началом** дуги e_k, v_j – **концом** дуги e_k.

Дуга e_k **выходит** из вершины v_i и **заходит** в вершину v_j.

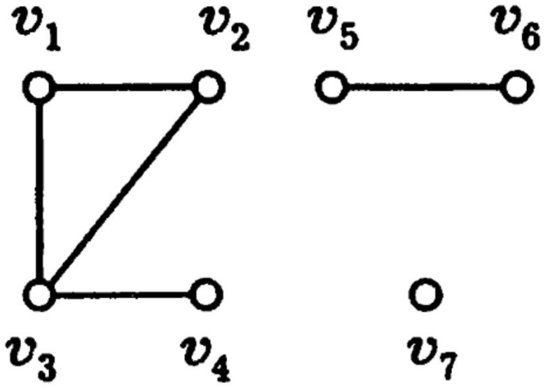
Вершина v_j **непосредственно достижима** из вершины v_i $(v_i \to v_j)$.

Задание графа

1. Перечисление множества вершин и множества рёбер.

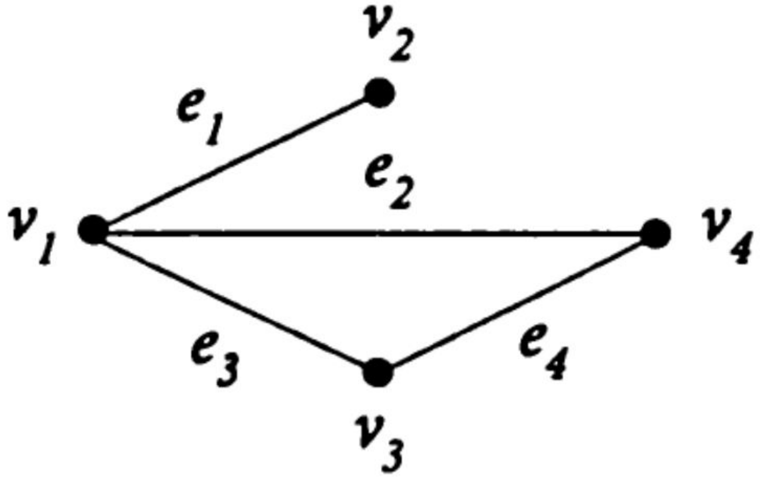
   $G = \left(\{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}, \left\{\{v_1, v_2\}, \{v_1, v_3\}, \{v_2, v_3\}, \{v_3, v_4\}, \{v_5, v_6\}\right\}\right)$

2. Диаграмма, на которой вершины представлены точками (окружностями), а рёбра – линиями, соединяющими соответствующие вершины.



Задание графа

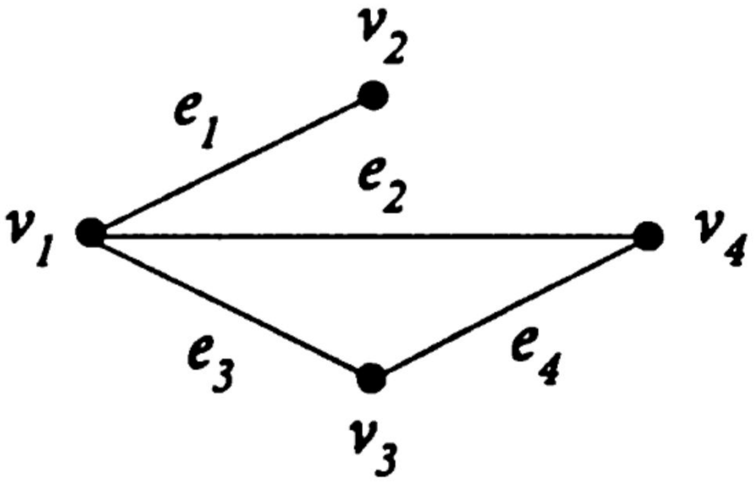
3. **Матрица инцидентности.** Матрица B, строки которой соответствуют вершинам графа, а столбцы – рёбрам. Считается при этом, что вершины и рёбра пронумерованы. Элемент b_{ij} i-й строки и j-го столбца матрицы B равен единице, если i-я вершина инцидентна j-му ребру, и нулю в противном случае.



$$
\begin{array}{c}
 \\ v_1 \\ v_2 \\ v_3 \\ v_4
\end{array}
\begin{array}{c}
\begin{array}{cccc} e_1 & e_2 & e_3 & e_4 \end{array} \\
\begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}
\end{array}
$$

Задание графа

4. **Матрица смежности.** Матрица A, строки и столбцы которой соответствуют вершинам графа, перечисленным в одинаковом порядке. Элемент a_{ij} i-й строки и j-го столбца матрицы B равен единице, если i-я и j-я вершины соединены ребром (являются смежными), и нулю в противном случае.



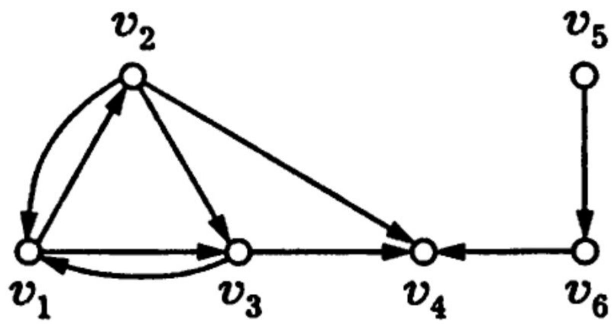
$$
\begin{array}{c|cccc|}
 & v_1 & v_2 & v_3 & v_4 \\
\hline
v_1 & 0 & 1 & 1 & 1 \\
v_2 & 1 & 0 & 0 & 0 \\
v_3 & 1 & 0 & 0 & 1 \\
v_4 & 1 & 0 & 1 & 0 \\
\hline
\end{array}
$$

Задание орграфа

1. Перечисление множества вершин и множества дуг.

$G = (\{v_1, v_2, v_3, v_4, v_5, v_6\}, \{(v_1, v_2), (v_1, v_3), (v_2, v_1), (v_2, v_3), (v_2, v_4), (v_3, v_1), (v_3, v_4), (v_5, v_6), (v_6, v_4)\})$

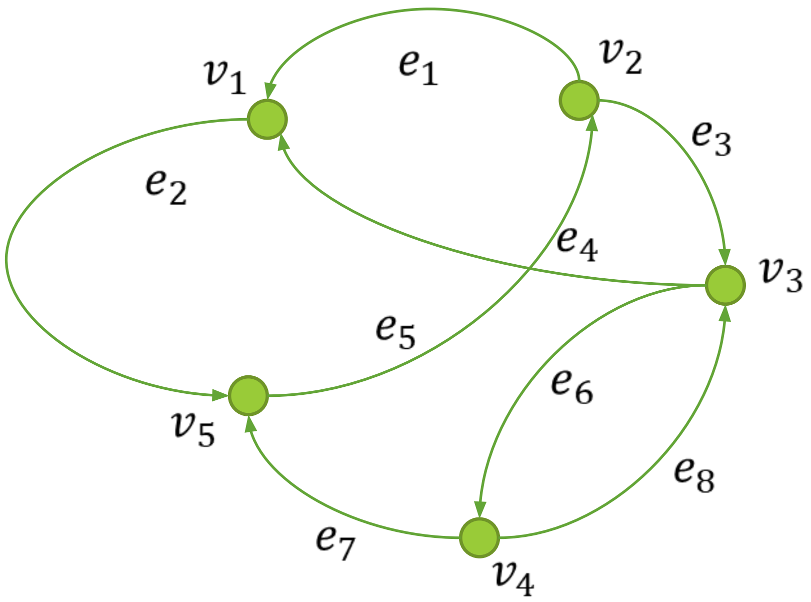
2. Диаграмма, на которой вершины представлены точками (окружностями), а дуги обозначаются стрелкой, направленной из начала дуги к её концу.



Матрица инцидентности орграфа

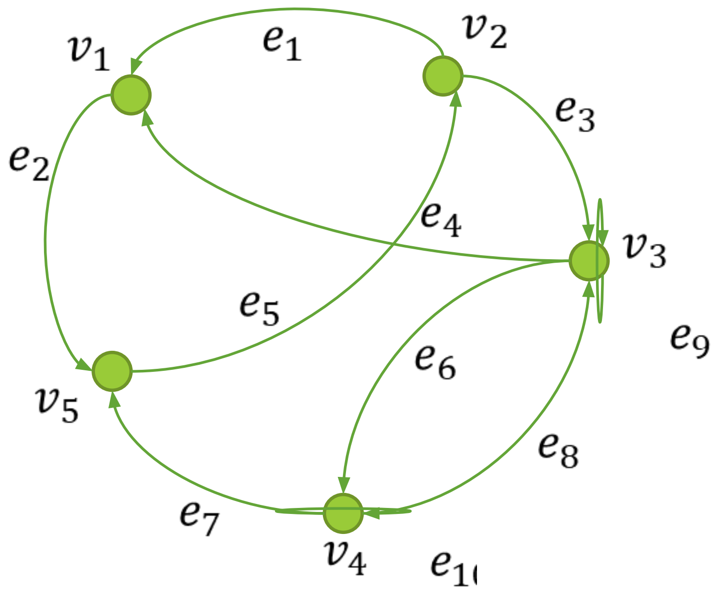
$$b_{ij} = \begin{cases} 1, e_j = (v_i, v_k), \\ -1, e_j = (v_k, v_i), \\ 0, v_i \notin e_j. \end{cases}$$

$$B = \begin{bmatrix} -1 & 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 \\ 0 & -1 & 0 & 0 & 1 & 0 & -1 & 0 \end{bmatrix}$$



Матрица смежности орграфа

$$a_{ij} = \begin{cases} 1, (v_i, v_j) \in E, \\ 0, (v_i, v_j) \notin E. \end{cases}$$



$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$\deg_+ v_i = \sum_{j=1}^{|V|} a_{ij}$$

$$\deg_- v_i = \sum_{j=1}^{|V|} a_{ji}$$

Степень (валентность) вершины

Степенью вершины называется количество рёбер, инцидентных данной вершине.

$\deg(v_i)$

Вершина степени 0 $(\deg(v_i) = 0)$ называется **изолированной**.

Вершина степени 1 $(\deg(v_i) = 1)$ называется **висячей**.

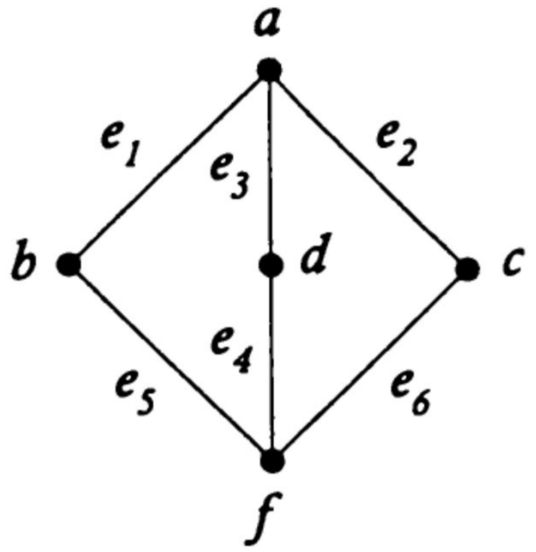
$\deg(a) = 3$

$\deg(b) = 2$

$\deg(c) = 2$

$\deg(d) = 2$

$\deg(f) = 3$



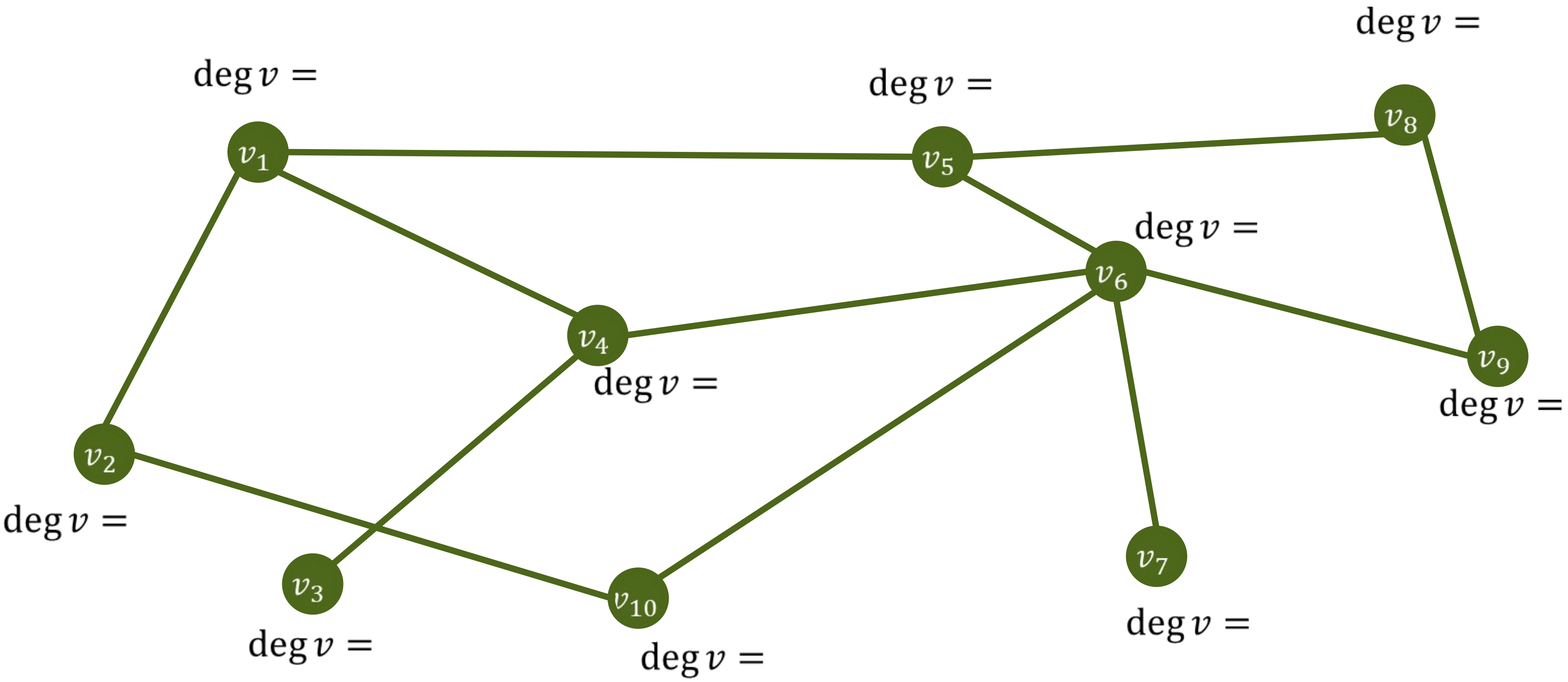
Полустепень

Полустепенью исхода (выхода) вершины называется количество исходящих из вершины рёбер $\deg_+ v_i$.

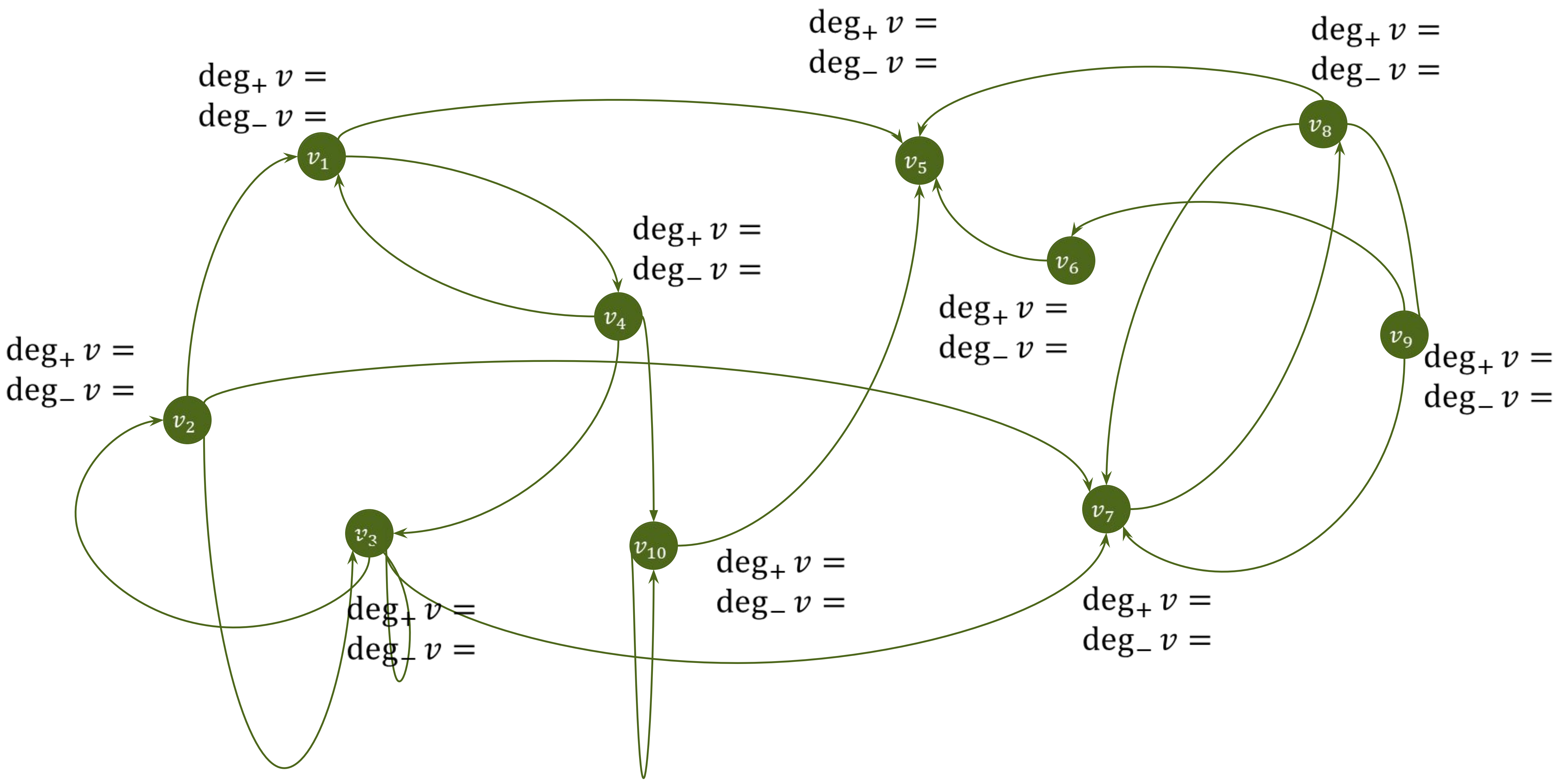
Полустепенью входа (захода) вершины называется количество входящих в вершину рёбер $\deg_- v_i$.

Степень вершины орграфа находится как $\deg v_i = \deg_+ v_i + \deg_- v_i$.

$$\sum_{j=1}^{|V|} \deg_+ v_j = \sum_{j=1}^{|V|} \deg_- v_j$$



$\deg v =$
$\deg v =$
$\deg v =$
v_8
v_1
v_5
$\deg v =$
v_6
v_4
v_9
$\deg v =$
$\deg v =$
v_2
$\deg v =$
v_3
v_{10}
v_7
$\deg v =$
$\deg v =$
$\deg v =$



$\deg_+ v =$
$\deg_- v =$
v_1
$\deg_+ v =$
$\deg_- v =$
v_2
v_3
$\deg_+ v =$
$\deg_- v =$
v_4
$\deg_+ v =$
$\deg_- v =$
v_5
$\deg_+ v =$
$\deg_- v =$
v_6
$\deg_+ v =$
$\deg_- v =$
v_7
$\deg_+ v =$
$\deg_- v =$
v_8
$\deg_+ v =$
$\deg_- v =$
v_9
$\deg_+ v =$
$\deg_- v =$
v_{10}
$\deg_+ v =$
$\deg_- v =$

Теорема

- Сумма степеней вершин графа всегда чётная $\sum_{j=1}^{n} \deg(v_j) = 2q, \ q \in \mathbb{N}$.

Доказательство:

Рассмотрим сумму степеней вершин как сумму количества вершин, принадлежащих рёбрам. Каждое ребро содержит ровно две вершины. Таким образом сумма степеней вершин равна удвоенному количеству рёбер. Следовательно – чётна.

Теорема

- В любом графе количество вершин нечётной степени чётно $\sum_{j:\deg(v_j)=2k+1} \deg(v_j) = 2p,\ p \in \mathbb{N}.$

Доказательство:

От противного: предположим, что граф содержит нечётное количество вершин нечётной степени.

Сумма чётных вершин – всегда чётна. Сумма нечётных, учитывая наше предположение – нечётна. Тогда сумма степеней всех вершин тоже нечётна, что противоречит теореме о чётности суммы степеней вершин в графе.

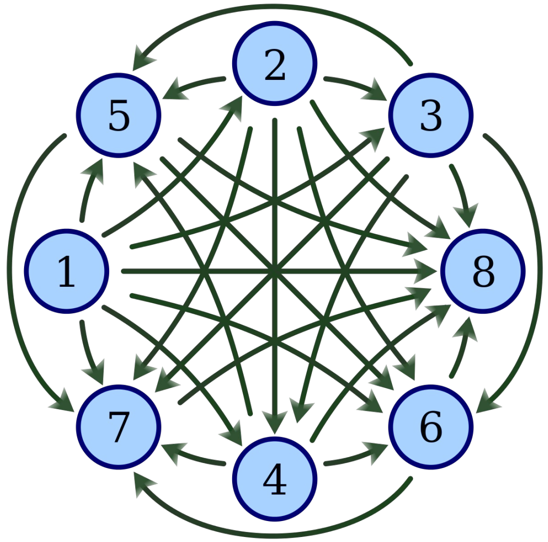
Делаем вывод, что предположение ложно и теорема справедлива.

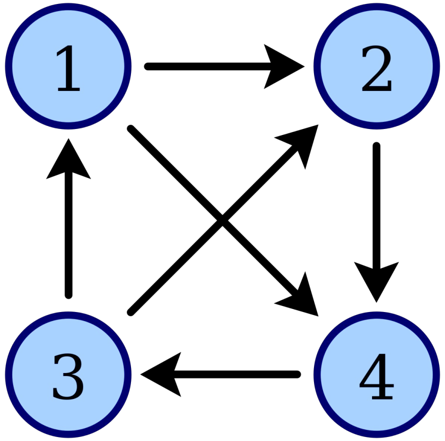
Особые графы

Нуль-граф (O_n) – граф, в котором нет ни одного ребра. Граф, степень каждой вершины которого равна нулю.

Полный граф (K_n) – граф, в котором каждая пара вершин смежна.

Регулярный (однородный) граф — граф, степени всех вершин которого равны. Степень регулярности обозначается $r(G)$. Регулярный граф с вершинами степени k называется k-регулярным, или регулярным графом степени k.

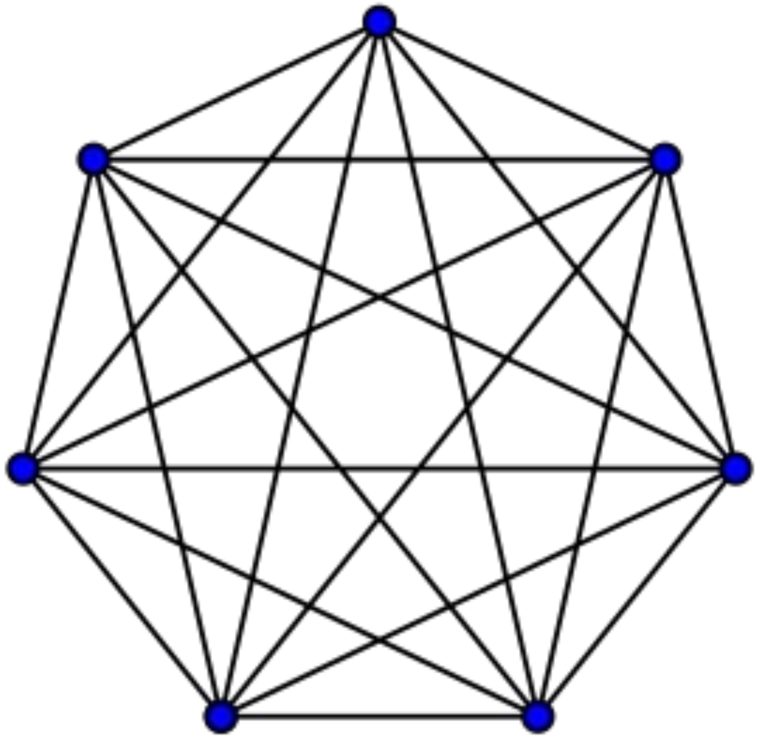


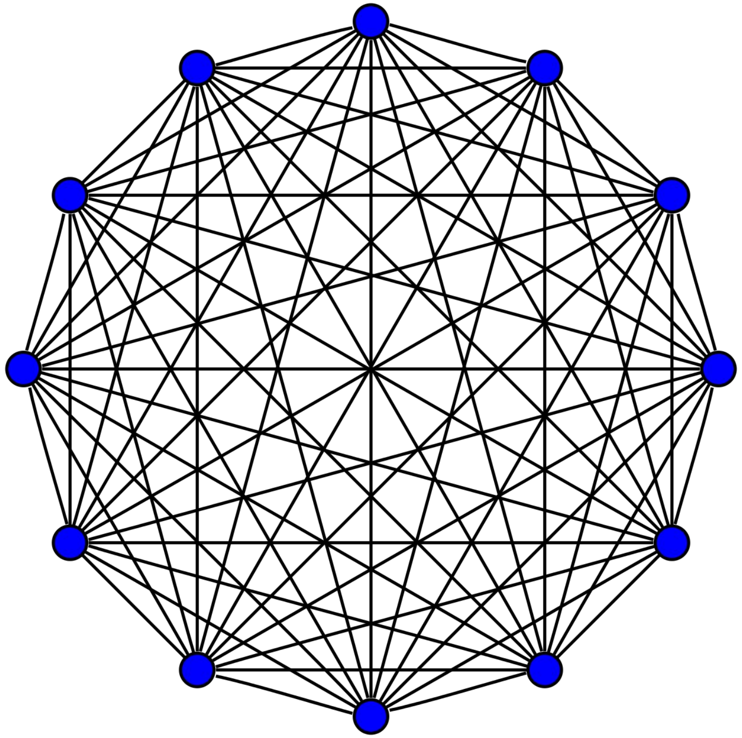


Турнир

Ориентированный граф, в котором любая пара вершин соединена ровно одной дугой, т. е. орграф, ассоциированным графом которого является полный граф.

Любой турнир содержит гамильтонов путь. Строго связный турнир имеет гамильтонов цикл.



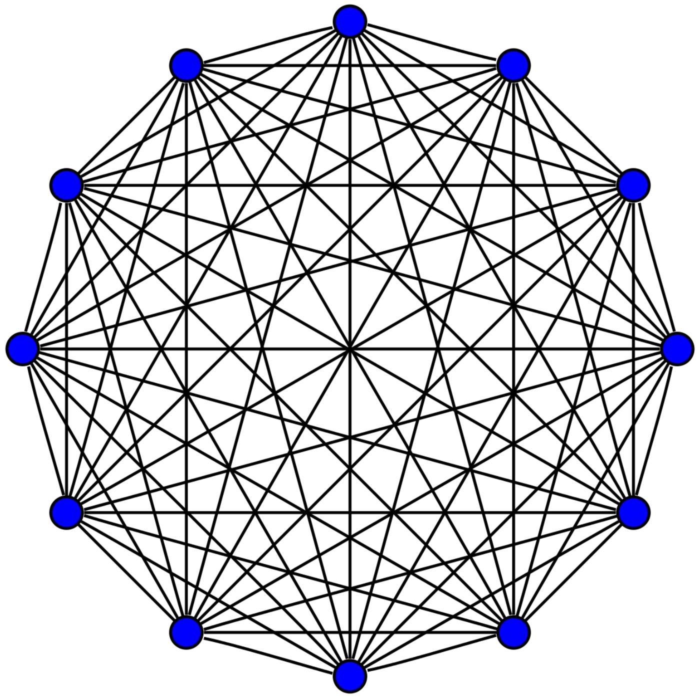


Сколько ребер у полного n-графа?

Сколько ребер у полного n-графа?

$$|K_n| = \frac{n(n-1)}{2}$$

$$(n-1) + (n-2) + \cdots + 2 + 1 + 0 = \frac{(n-1)+0}{2}n = \frac{n(n-1)}{2}$$

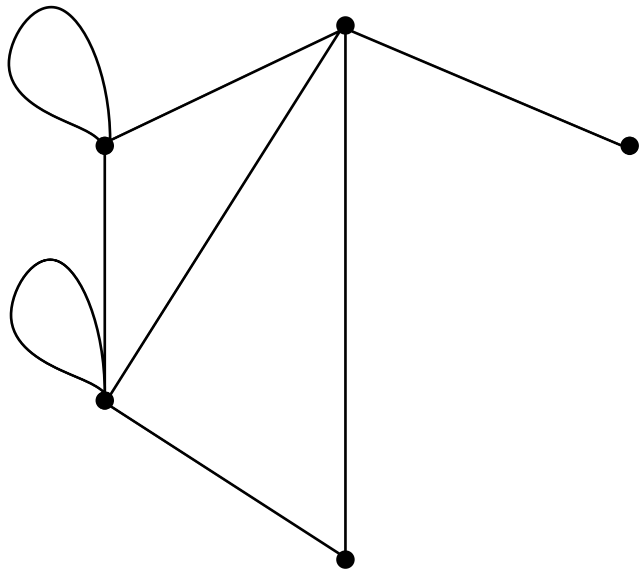


Мультиграфы и псевдографы

Петли

Ребро, началом и концом которого является одна и та же вершина $\{v_i, v_i\}$ называется петлёй.

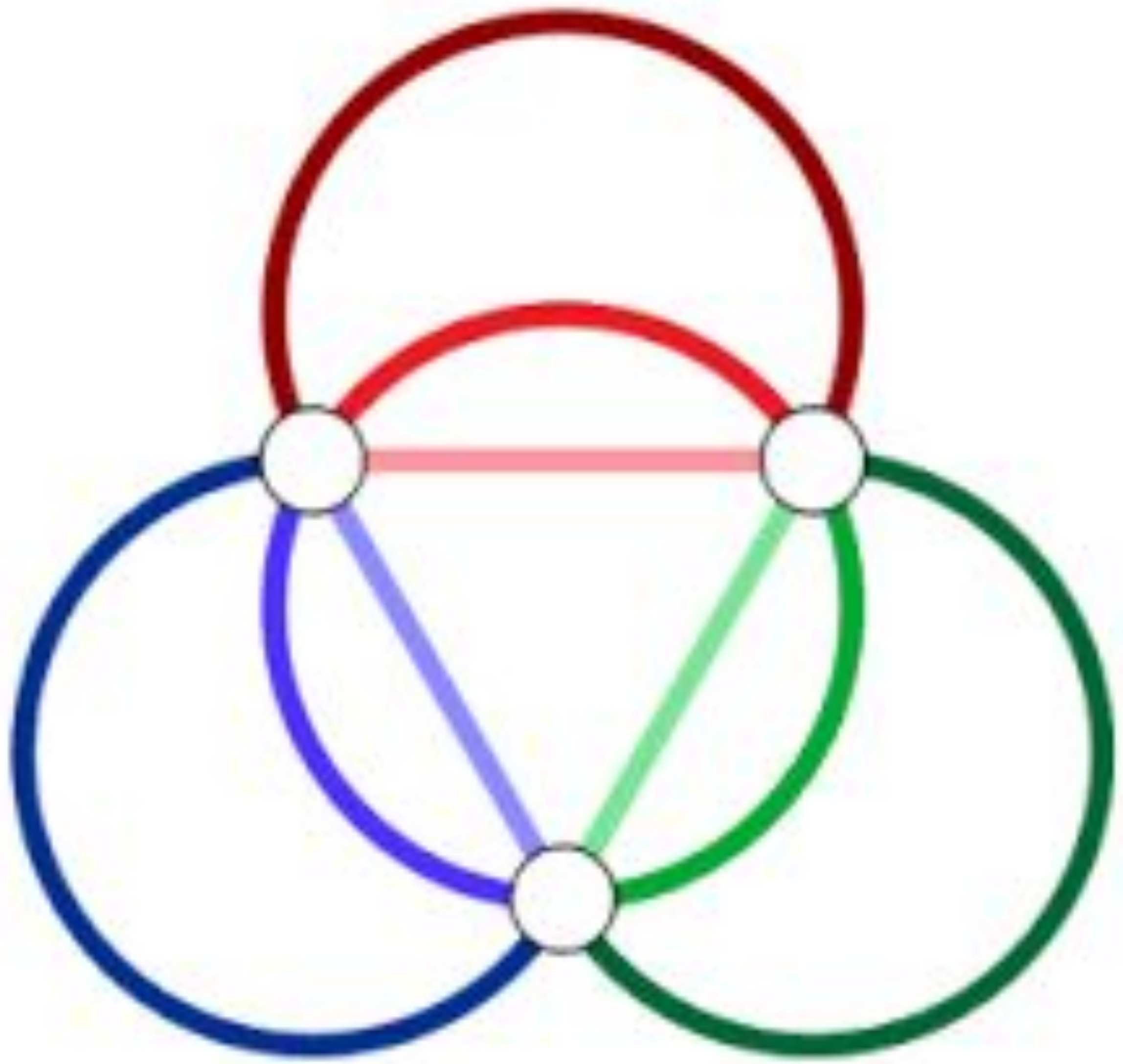
Граф, содержащий петли – графом с петлями.



Кратные рёбра

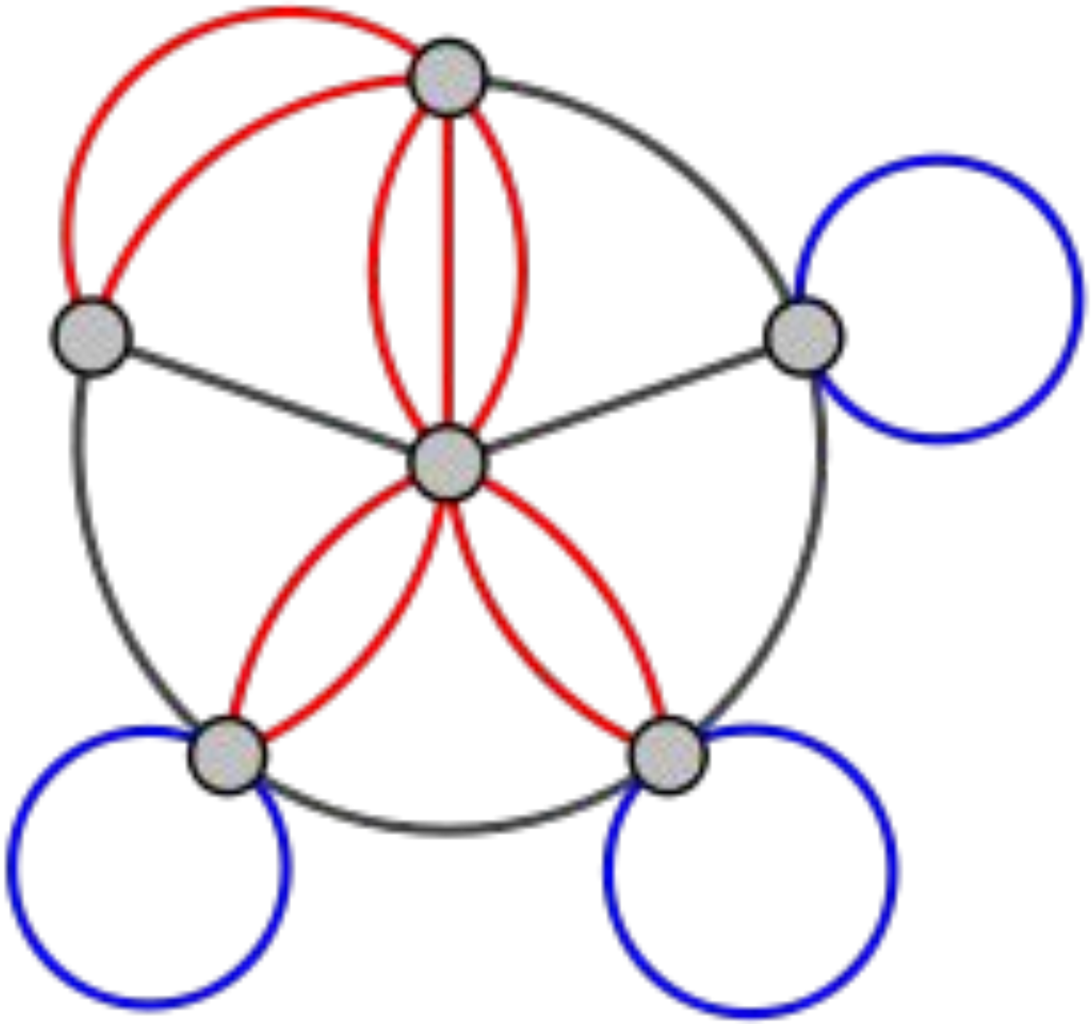
Если в графе существуют несколько рёбер, соединяющих одну и ту же вершину, то такие рёбра называются кратными.

Граф, содержащий кратные рёбра называется мультиграфом.



Псевдограф

Мультиграф, содержащий петли называется псевдографом.

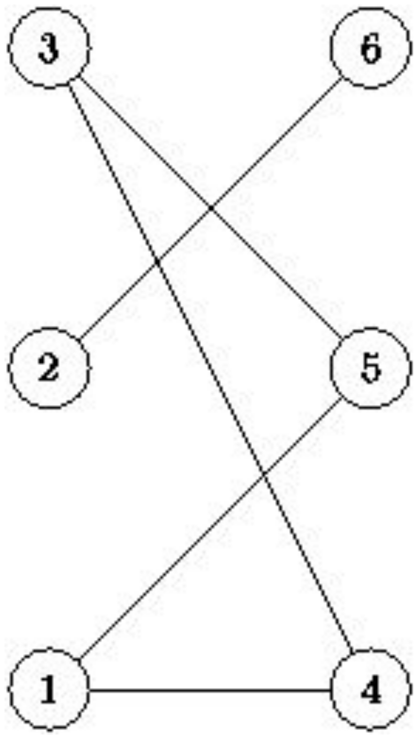


Двудольный граф

Двудольные графы

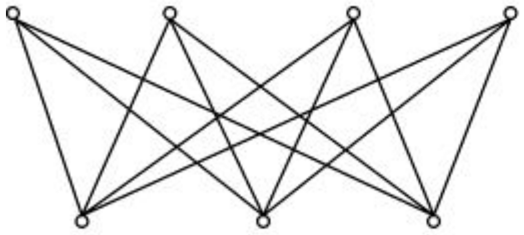
Неориентированный граф $G = (V, E)$– двудольный, если множество его вершин V можно разбить на два подмножества V_1 и V_2 так, что каждое ребро имеет один конец в V_1, а другой в V_2.

Обозначается такой граф $K_{m,n}$, где $m = |V_1|$, $n = |V\_2|$.

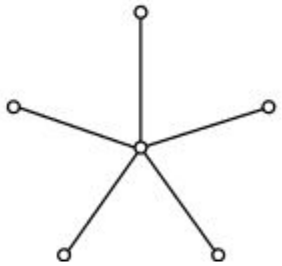


Полный двудольный граф

Полным двудольным графом называется специальный вид двудольного графа, у которого любая вершина первой доли соединена со всеми вершинами второй доли вершин.



$K_{4,3}$



$K_{5,1}$

Граф $К_{m,n}$ *имеет ровно* $m + n$ вершин и $m \cdot n$ *ребер.*

Маршруты и пути

Маршрут

Маршрутом в графе называется последовательность вершин, в которой две соседние вершины соединены рёбрами:

$$v_0 e_1 v_1 e_2 v_3 e_3 \dots v_{k-1} e_k v_k$$

где $e_1 = \{v_{i-1}, v_i\}$.

Часто записывают:

$$v_0 v_1 v_3 \dots v_{k-1} v_k$$

Длиной маршрута называется число составляющих его рёбер.

Вершина v_0 – называется **начальной** вершиной маршрута.

Вершина v_k – называется **конечной** вершиной маршрута.

Вершины $\left\{v_j \middle| j = \overline{1, k-1}\right\}$ – называются **внутренними**.

Маршрут

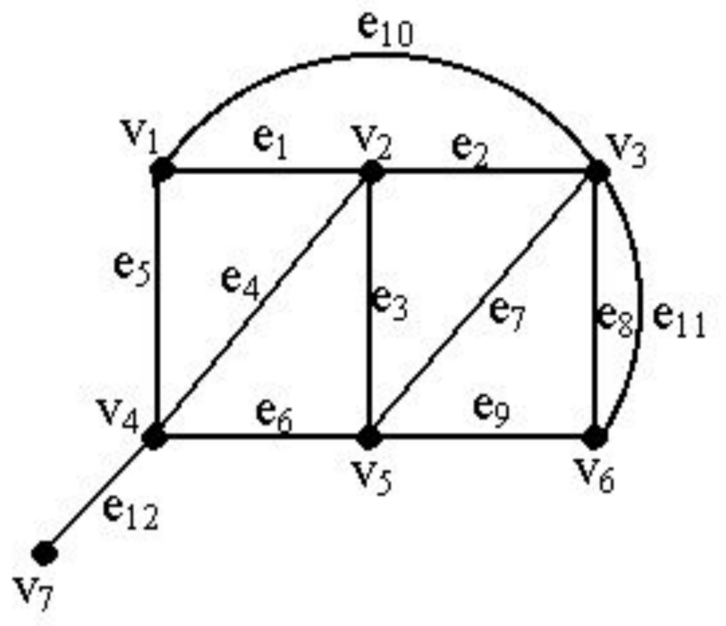
Если $v_0 = v_k$, то маршрут называют **замкнутым**, иначе – **незамкнутым** (открытым).

Незамкнутый:

$(\boldsymbol{v_1}, \boldsymbol{e_1}, \boldsymbol{v_2}, \boldsymbol{e_2}, \boldsymbol{v_3}, \boldsymbol{e_3}, \boldsymbol{v_6}, \boldsymbol{e_9}, \boldsymbol{v_5}, \boldsymbol{e_7}, \boldsymbol{v_3}, \boldsymbol{e_{11}},$

Замкнутый:

$(\boldsymbol{v_1}, \boldsymbol{e_1}, \boldsymbol{v_2}, \boldsymbol{e_2}, \boldsymbol{v_3}, \boldsymbol{e_7}, \boldsymbol{v_5}, \boldsymbol{e_3}, \boldsymbol{v_2}, \boldsymbol{e_4}, \boldsymbol{v_4}, \boldsymbol{e_5},$



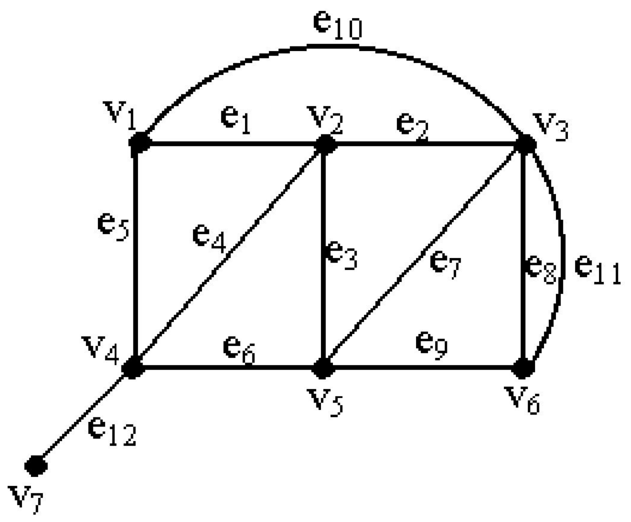
Цепь

Маршрут называется **цепью**, если все его ребра различны.

$$(v_7, e_{12}, v_4, e_5, v_1, e_1, v_2, e_4, v_4, e_6, v_5, e_3, v_2).$$

Цепь называется **простой**, если все её внутренние вершины различны и отличны от начальной и конечной.

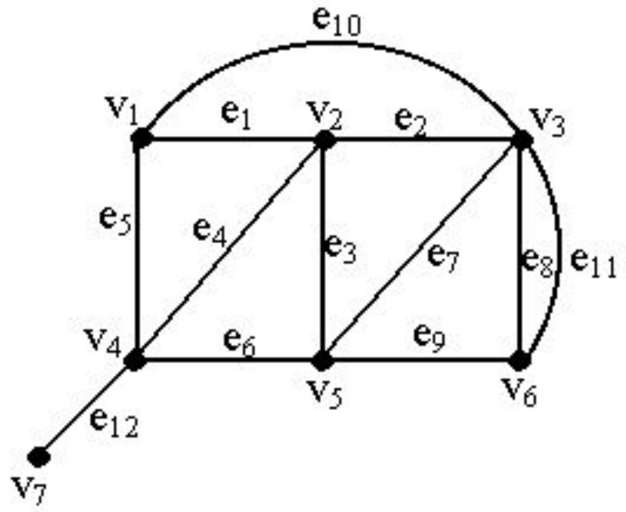
$$(v_1, e_1, v_2, e_2, v_3, e_8, v_6, e_9, v_5).$$



Цикл

Цепь называется **замкнутой**, если её концевые вершины совпадают. Замкнутая цепь называется **циклом**.

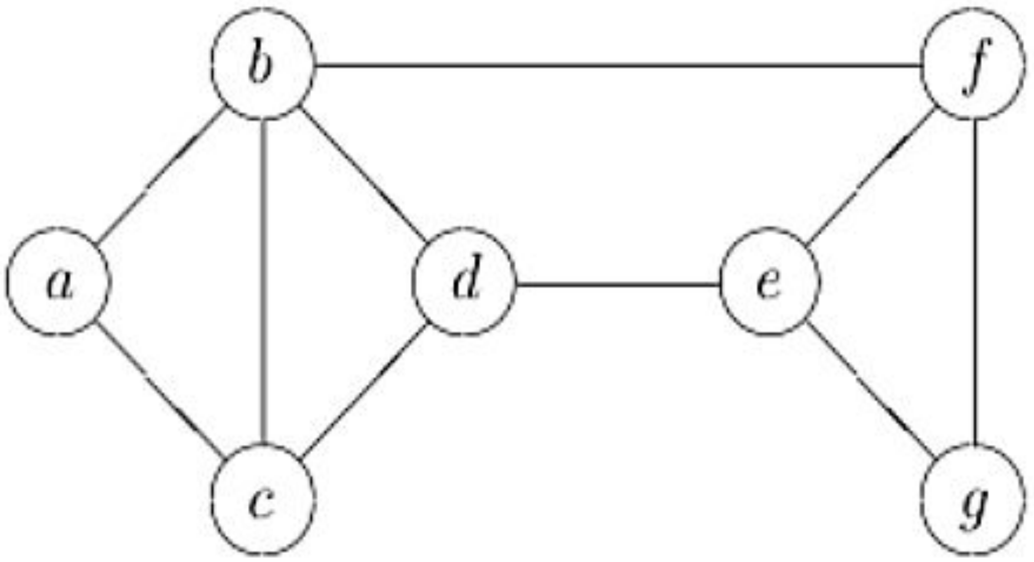
$(\boldsymbol{v_1}, \boldsymbol{e_{10}}, \boldsymbol{v_3}, \boldsymbol{e_2}, \boldsymbol{v_2}, \boldsymbol{e_3}, \boldsymbol{v_5}, \boldsymbol{e_6}, \boldsymbol{v_4}, \boldsymbol{e_5}, \boldsymbol{v_1})$.



Замкнутая простая цепь называется **простым циклом**.

$(\boldsymbol{v_1}, \boldsymbol{e_1}, \boldsymbol{v_2}, \boldsymbol{e_3}, \boldsymbol{v_5}, \boldsymbol{e_6}, \boldsymbol{v_4}, \boldsymbol{e_5}, \boldsymbol{v_1})$.

ПРИМЕР



abdbc – маршрут, но не цепь;

abdcb – цепь, но не простая цепь;

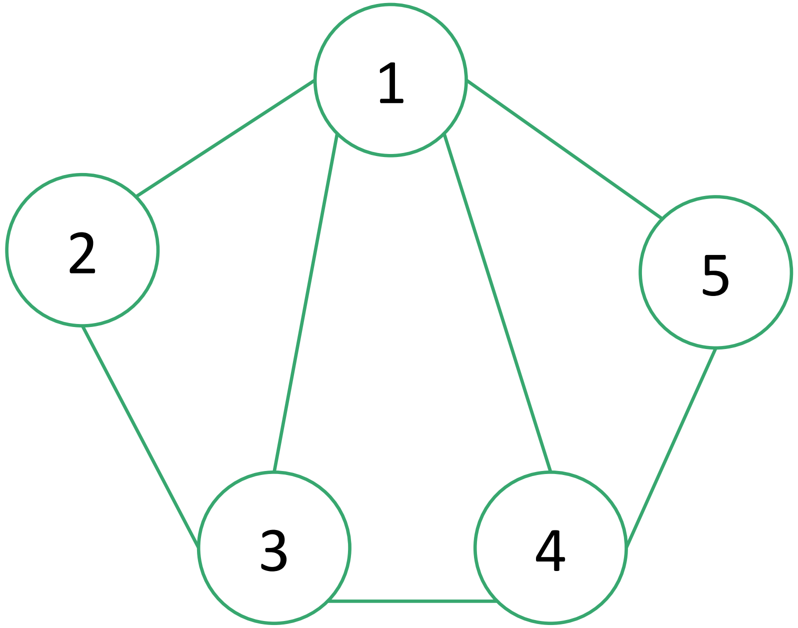
abcde – простая цепь;

abdbca – замкнутый маршрут, но не цикл;

abfedbca – цикл, но не простой цикл;

abca – простой цикл.

Определить



2,3,5,4 – маршрут? **НЕТ**

2,3,4,5,1,4,3- маршрут? **ДА** а цепь? **НЕТ**

3,1,4,5,1,2- цепь? **ДА** простая? **НЕТ**

2,3,1,4,3,1,2 – цикл? **НЕТ** замкнутый маршрут? **ДА**

2,3,1,4,5,1,2- цикл? **ДА** простой? **НЕТ**

2,3,4,5,1,2- цикл? **ДА** простой? **ДА**

Свойства

1. Каждая внутренняя вершина незамкнутой цепи инцидентна не менее, чем рёбрам, концевые вершины инцидентны минимум одному ребру.
2. Если в графе степень каждой вершины не меньше 2, то в нем есть цикл.
3. Число вершин в маршруте на единицу больше числа рёбер, тогда как в цикле число рёбер равно числу вершин.

Свойства

4. В любом цикле, проходящем через некоторое ребро, содержится простой цикл, проходящий через это ребро.
5. Вершина v_j называется достижимой из вершины v_i, если существует маршрут из v_i в v_j.
6. Если вершина v_j достижима из v_i, то есть простая цепь, соединяющая вершины v_i, v_j.

Лемма. Всякий маршрут графа содержит хотя бы одну простую цепь, соединяющую ту же пару вершин.

Доказательство

Рассмотрим маршрут:

$$\left(v_0, e_1, v_1, e_2, v_2, \dots, v_{i-1}, e_i, v_i, \dots, v_{j-1}, e_j, v_i, e_{j+1}, \dots, v_{k-1}, e_k, v_k\right)$$

Если v_i - первая из вершин маршрута, которая входит в него больше одного раза, а $v_j = v_i$ – последняя из совпадающих с v_i вершин маршрута, то его можно заменить более коротким:

$$\left(v_0, e_1, v_1, e_2, v_2, \dots, v_{i-1}, e_i, v_i, e_{j+1}, \dots, v_{k-1}, e_k, v_k\right)$$

Если в полученном маршруте есть еще повторяющиеся вершины, то снова заменяем его более коротким и так далее, пока не выделим маршрут без повторяющихся вершин.

Путь

Последовательность вершин и дуг $v_0 e_1 v_1 e_2 \dots e_{n-1} v_{n-1} e_n v_n$, где $e_k = (v_{k-1}, v_k)$ ($v_{k-1} \to v_k$) называется **путем** (ориентированным маршрутом) из вершины v_0 в вершину v_n. **Длиной пути** называется количество дуг (сумма весов дуг), входящих в него.

Простой путь – все вершины которого, кроме, возможно, первой и последней различны.

Простой замкнутый путь ненулевой длины называется **контуром**.

Граф, не содержащий контуров, называется **бесконтурным**.

Если существует путь из v_0 в v_n говорят, что вершина v_n достижима из v_0.

Для любого пути, ведущего из v_0 в v_n, в ориентированном графе существует простой путь из v_0 в v_n.

Свойства орграфа

- В каждом бесконтурном графе имеется хотя бы одна вершина, полустепень исхода которой равна нулю.
- Вершины бесконтурного ориентированного графа можно перенумеровать так, что каждая дуга идёт из вершины с меньшим номером в вершину с большим номером.

Связность

Связность графа

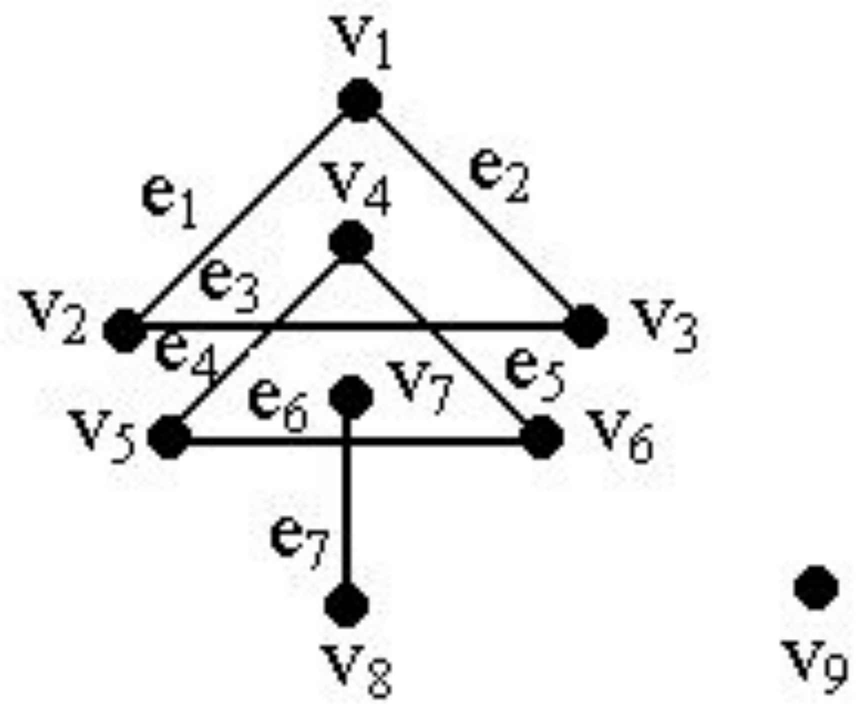
Две различные вершины графа ***называются связными***, если существует соединяющая их простая цепь. В противном случае две вершины называются ***несвязными***.

Граф ***называется связным***, если любые две его вершины связные.

Граф ***называется несвязным***, если хотя бы две его вершины несвязные.

Компонента связности – такое множество вершин в графе, и инцидентных их рёбер, что любые две вершины в данном множестве являются связными (максимально полный подграф).

Пример



1 компонента связности:

$$(\{v_1, v_2, v_3\}, \{e_1, e_2, e_3\})$$

2 компонента связности:

$$(\{v_4, v_5, v_6\}, \{e_4, e_5, e_6\})$$

3 компонента связности:

$$(\{v_7, v_8\}, \{e_7\})$$

4 компонента связности:

$$(\{v_9\}, \emptyset)$$

Связность орграфа

Ориентированный граф называется связным, если для любых двух его вершин v_i, v_j, вершина v_j достижима из вершины v_i, или вершина v_i достижима из вершины v_j.

Ориентированный граф называется сильно связным, если для любых двух его вершин v_i, v_j, вершина v_j достижима из вершины v_i, или вершина v_i достижима из вершины v_j.

Неориентированный граф $G_1 = (V_1, E_1)$ называют ассоциированным с ориентированным графом $G = (V, E)$, если

$$V_1 = V,\ E_1 = \{\{v_i, v_j\} | (v_i, v_j) \in E\ OR\ (v_j, v_i) \in E, v_i \neq v_j\}.$$

Ориентированный граф называется слабо связным, если ассоциированный с ним неориентированный граф связный.

Матрицы связности и достижимости

Матрицей связности графа называется матрица $S(G) = [s_{ij}]$, где

$$s_{ij} = \begin{cases} 1, \text{если } i = j \text{ или существует маршрут, соединяющий } v_i \text{ и } v_j, \\ 0, \text{если } i \neq j \text{ и не существует маршрута, соединяющего } v_i \text{ и } v_j. \end{cases}$$

Матрицей достижимости орграфа называется матрица $T(G) = [s_{ij}]$, где

$$t_{ij} = \begin{cases} 1, \text{если } v_j \text{ достижима из } v_i, \\ 0, , \text{если } v_j \text{ достижима из } v_i. \end{cases}$$

Матрицы связности и достижимости

Пусть дан граф $G = (V, A)$, с матрицей смежности $A = \left(a_{ij}\right)_{n\times n}$. Матрица смежности даёт информацию о всех маршрутах длины 1. Для поиска маршрутов длины 2 можно найти композицию отношения A с самим собой:

$$A^{\odot 2} = \vee_k \left(a_{ik} \wedge a_{kj}\right)$$

Для определения наличия маршрутов длины не более k между вершинами необходимо вычислить

$$P(G) = E \vee A \vee A^{\odot 2} \vee A^{\odot 3} \vee \cdots \vee A^{\odot k}$$

В таком случае при использовании алгебраического умножения:

$$\mathrm{K}(G) = E + A + A^2 + A^3 + \cdots + A^k$$

результатом будет матрица количества путей длины от 0 до k между вершинами.

Для орграфа матрица достижимости вычисляется так же, как и матрица связности графа.

Эйлеров граф

Задача **о семи кёнигсбергских мостах**

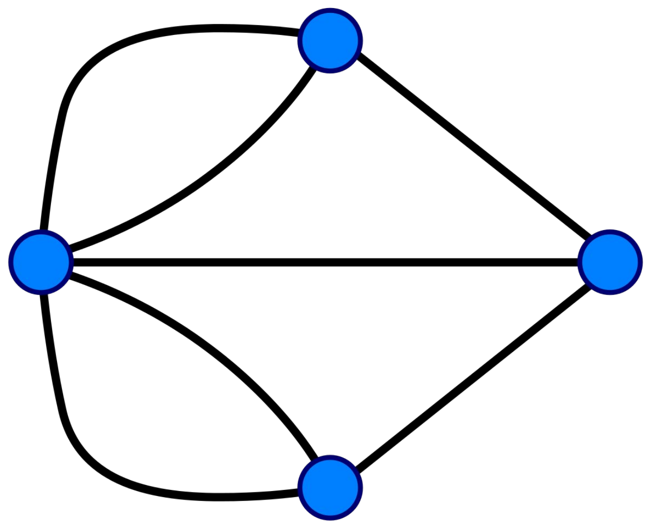
Издавна среди жителей Кёнигсберга была распространена такая загадка: как пройти по всем мостам (через реку Преголя), не проходя ни по одному из них дважды. Многие кёнигсбержцы пытались решить эту задачу как теоретически, так и практически, во время прогулок. Впрочем, доказать или опровергнуть возможность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



Эйлеров цикл и Эйлерова цепь

Цикл называется **эйлеровым**, если он содержит все ребра графа.

Граф называется **эйлеровым**, если в нем найдётся эйлеров цикл.

Связный граф является эйлеровым тогда и только тогда, когда он не содержит вершин нечётной степени.

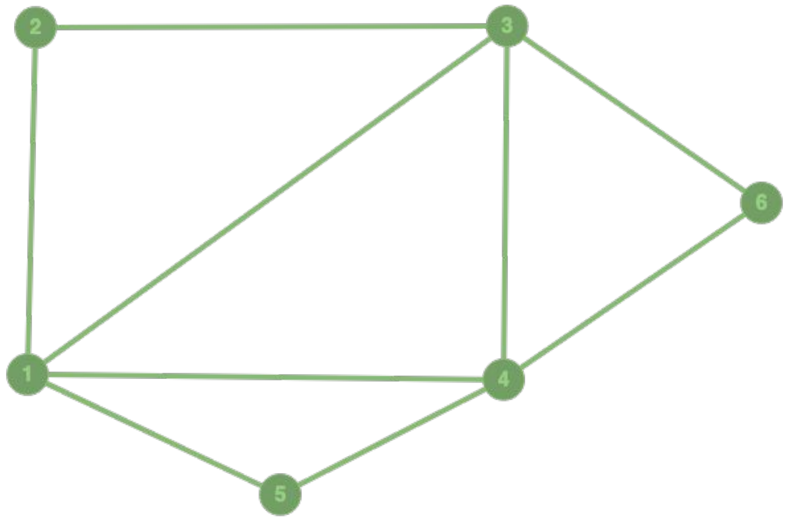
Цепь, содержащая все ребра графа, называется **эйлеровой**.

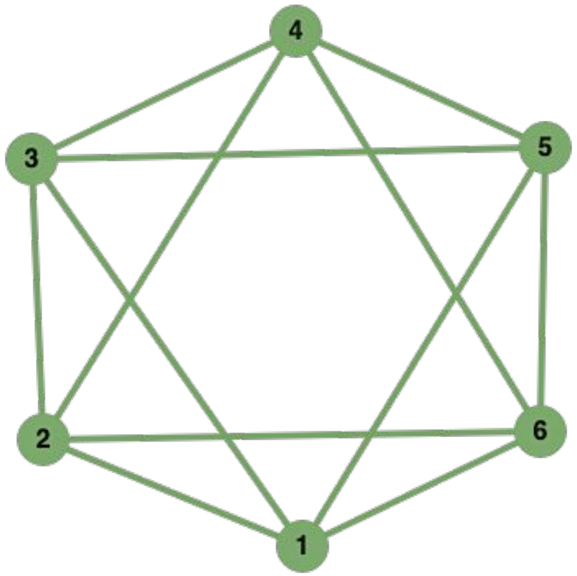
Граф, обладающий эйлеровой цепью, называется **квазиэйлеровым**.

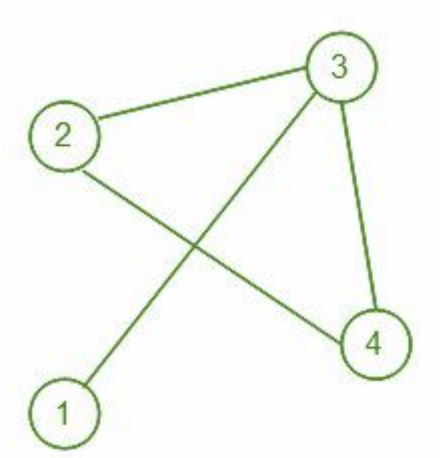
Граф является квазиэйлеровым, если в нем не более двух вершин нечётной степени.

В квазиэйлеровом графе существующие у него две вершины нечётной степени всегда будут являться концами любой эйлеровой цепи.

Эйлеровы графы



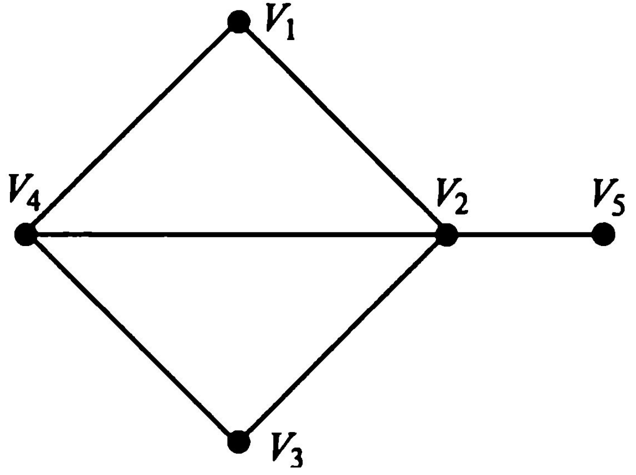


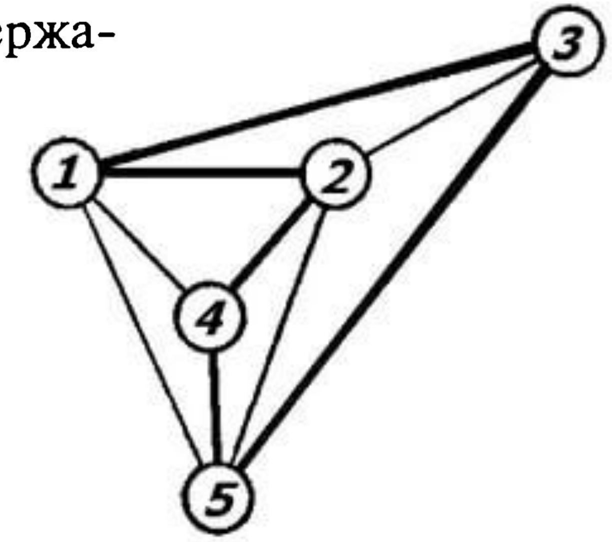


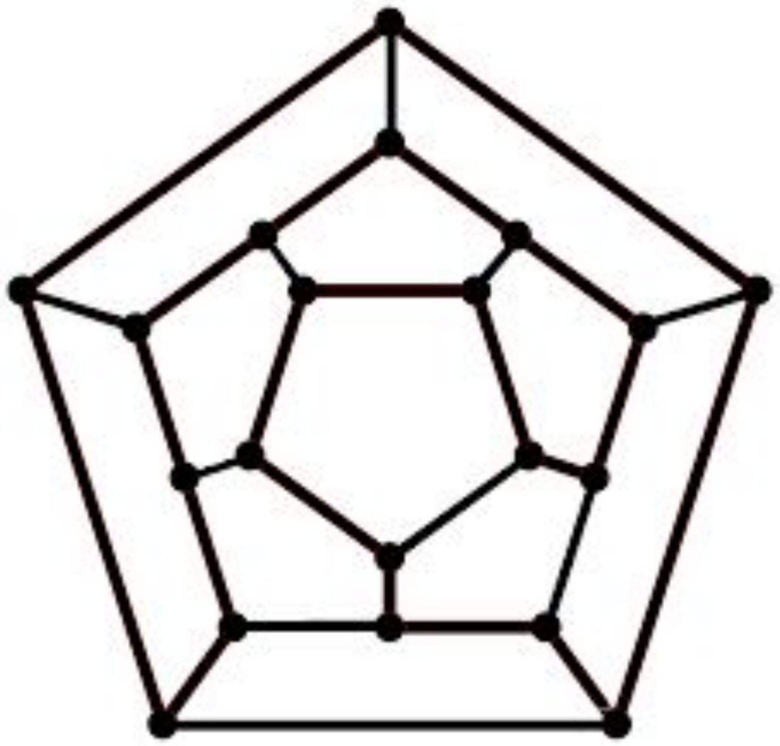
Гамильтонов граф

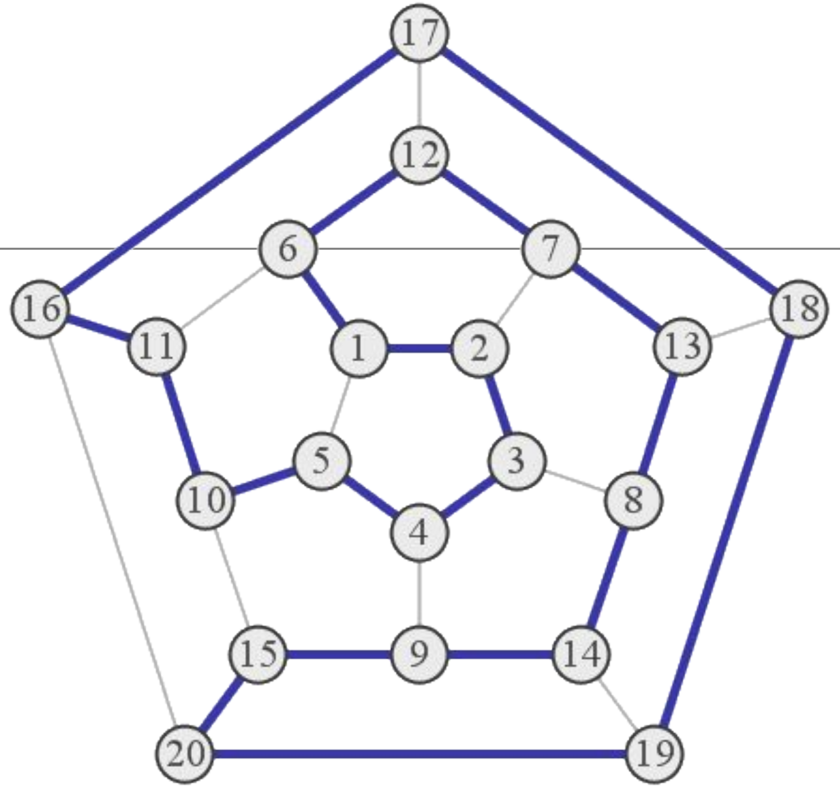
Гамильтонов граф

Гамильтоновым $\frac{\textbf{путем}}{\textbf{циклом}}$ графа G называется $\frac{\text{путь}}{\text{цикл}}$, проходящий через каждую его вершину только один раз. Граф, содержащий гамильтонов цикл, называется **гамильтоновым**.









Игра "Вокруг света"

Гамильтонов граф

Критерий существования гамильтонова цикла в произвольном графе еще не найден.

Факты о существовании гамильтоновых циклов в графе:

Всякий полный граф является гамильтоновым

Если *граф*, помимо простого *цикла*, проходящего через все его вершины, содержит и другие ребра, то он также является гамильтоновым.

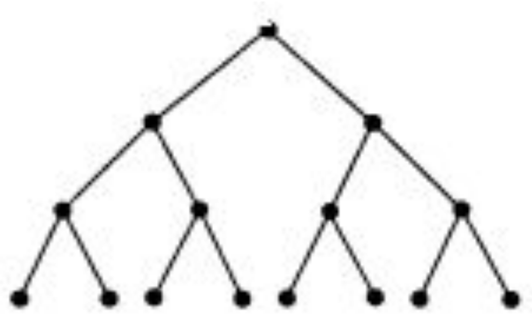
Если *гамильтонов граф* объединить с ещё одной вершиной ребром так, что образуется *висячая вершина*, то такой *граф* гамильтоновым не является,

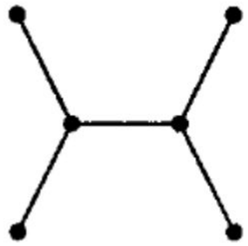
Не является гамильтоновым и *граф*, представляющий собой *простой цикл* с "перекладиной" (тэта граф), на которой расположены одна или несколько вершин.

Деревья

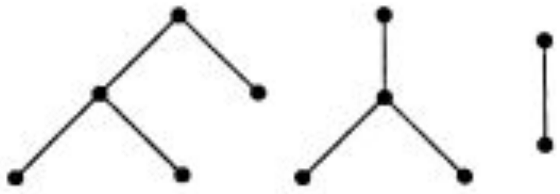
Определения

Дерево - связный, неориентированный граф без циклов.



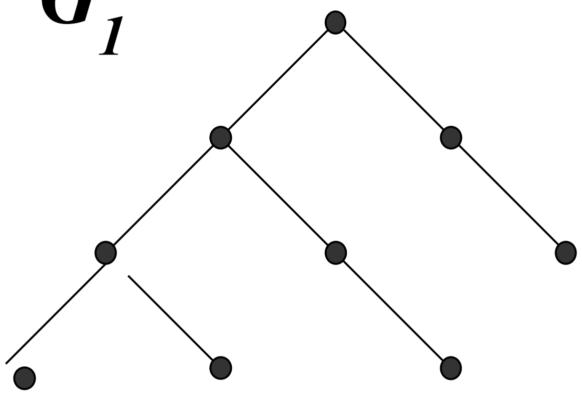


Лес – граф, компоненты которого являются деревьями.

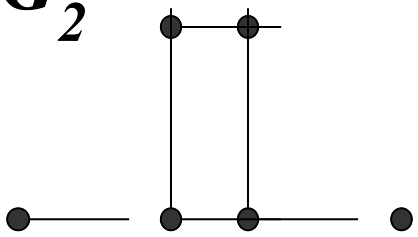


Какой из графов является деревом?

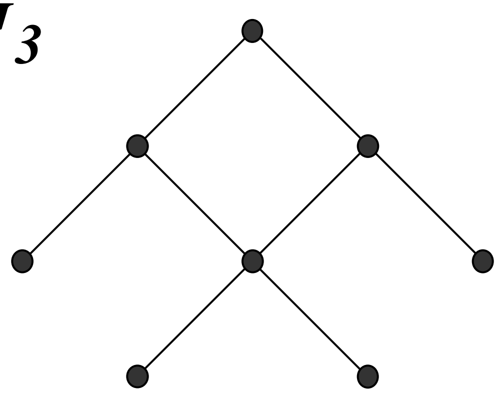
G_1



G_2



G_3



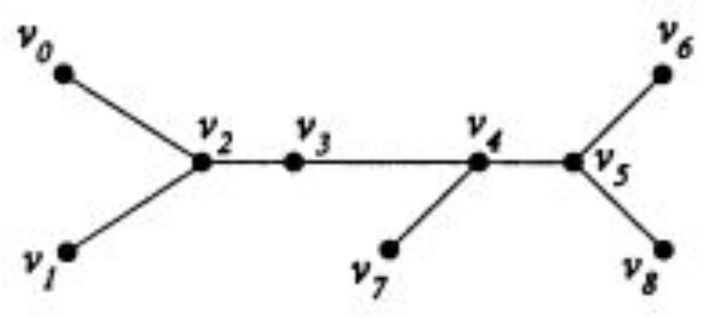
G_1 – дерево G_2 – не дерево G_3 – не дерево

Листья

Если неориентированное дерево имеет хотя бы одно ребро, оно имеет хотя бы две вершины со степенью 1.

Вершины степени 1 называются **листьями.** $(v_0, v_1, v_6, v_7, v_8)$

Другие вершины называются **внутренними вершинами**. (v_2, v_3, v_4, v_5)



Свойства деревьев

Следующие утверждения эквивалентны:

1. G – дерево.
2. G – связный неориентированный граф и в нём число рёбер $|E|$ на единицу меньше числа вершин $|V|$: $|\mathrm{E}| = |\mathrm{V}| - 1$.
3. Любые две вершины дерева соединяет единственная цепь.
4. G – неориентированный граф без циклов и если любую пару несмежных вершин соединить ребром, то G будет содержать единственный цикл.

Свойства деревьев

Теорема.

Для любых двух вершин a и b дерева T существует единственная простая цепь, связывающая их.

Доказательство.

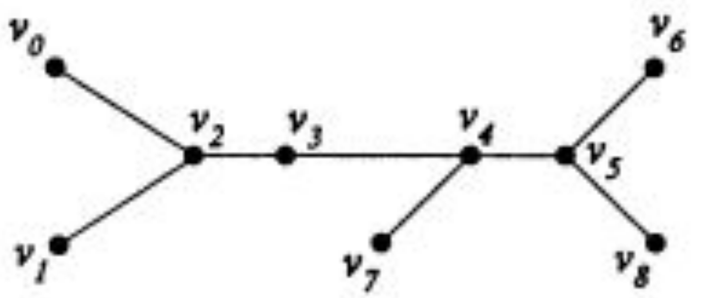
Предположим, что для некоторых вершин a и b соединяющая их цепь не является единственной. Допустим существует цепь $v_0 v_1 v_2 \dots v_n$ длины n и $v'_0 v'_1 v'_2 \dots v'_m$ длины m, где $v_0 = v'_0 = a$ и $v_n = v'_m = b$.

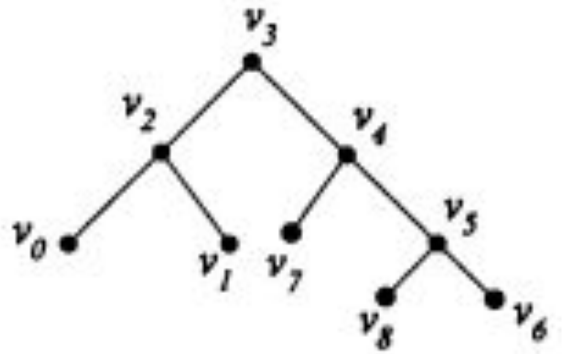
В каждом пути должна существовать первая вершина, начиная с которой соответствующие вершины не совпадают, например $v_i \neq v'_j$ и в каждом из путей должна существовать точка, начиная с которой вершины опять одни и те же $v_k = v'_l$.

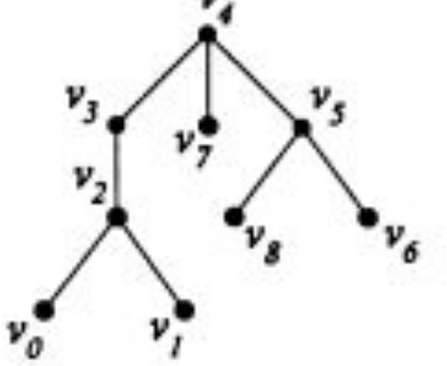
Тогда $v_{i-1} v_i v_{i+1} \dots v_k v'_{l-1} v'_{l-2} \dots v'_j$ является циклом. Следовательно граф T не является деревом, что противоречит условию.

Корень дерева

Пусть дерево представляет физический объект, подвижный в вершинах. Если подвесить дерево за одну из вершин, остальная часть повиснет ниже.







Такая вершина называется **корнем** дерева.

Если корень дерева определен, то дерево называется **корневым деревом.**

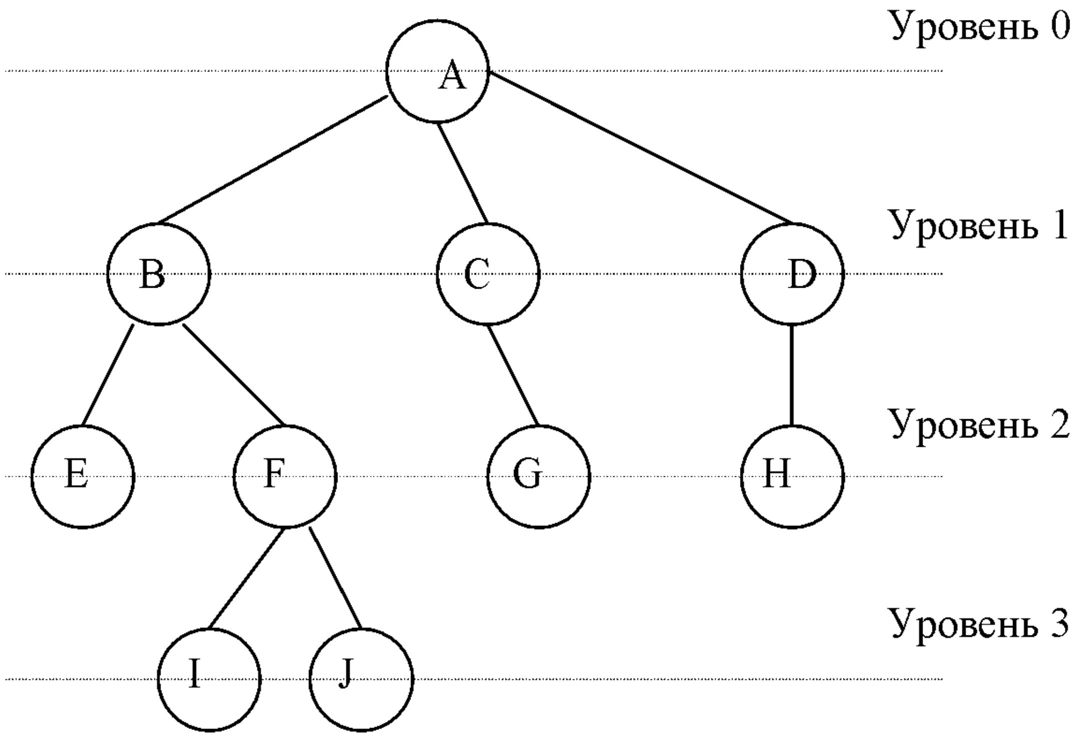
Корневое дерево

Все вершины корневого дерева имеют характеристику, называемую уровнем вершины.

Уровень вершины v определяется расстоянием от корня до вершины v. Узлы одного уровня образуют **ярус** дерева

Высота дерева определяется максимальным расстоянием от корня дерева до листа.

N**-арным** называется дерево, в котором степени вершин, за исключением корня не превосходят $N+1$, а степень корня не превосходит N.



Предки и потомки

Внутренняя вершина называется **родительской (отцом)** по отношению к инцидентным ей вершинам нижнего уровня. Соответствующие вершины называются **сыновьями**. Сыновья одного узла называются **братьями**.

Вершину v называют **потомком** вершины u, если их связывает простая цепь, не проходящая через корень дерева. Вершина u в таком случае называется **предком** вершины v.

Корень дерева является предком любой его вершины.

В $\boldsymbol{N}$**-арном** дереве количество сыновей каждой вершины не превосходит N.

Основные соотношения

1) Если полное (все листья расположены на одном ярусе) N –арное дерево высоты h имеет k листьев, то $h = \log_N k$.
2) Если N–арное дерево имеет k листьев, то $h \geq \log_N k$.
3) Если полное бинарное дерево высоты h имеет v вершин, то $h = \log_2(v+1) - 1$.
4) Если бинарное дерево высоты h имеет v вершин, то $h \geq \log_2(v+1) - 1$.
5) Количество вершин v в полном бинарном дереве высоты h, $v = 2^{h+1} - 1$

Поиск в глубину

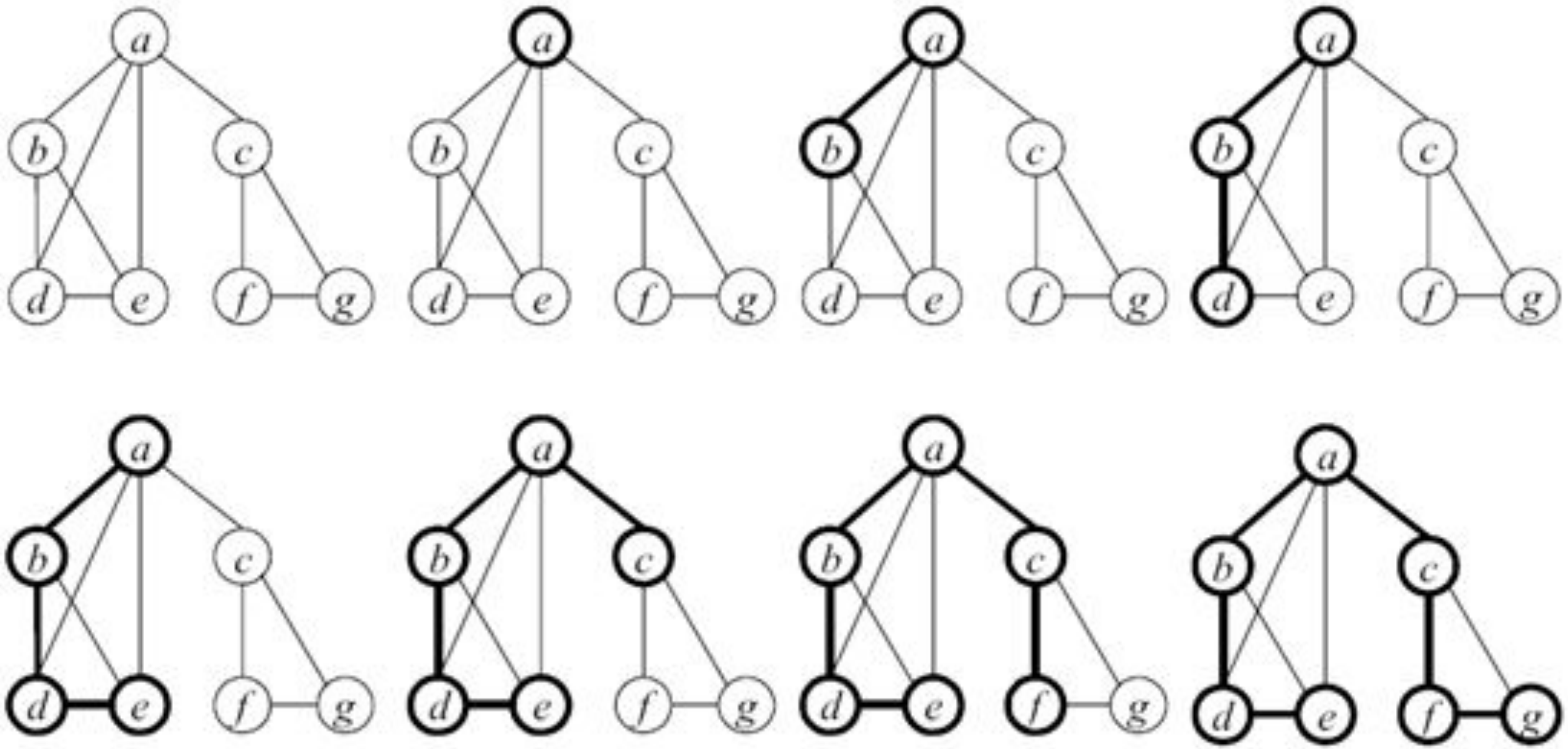
Поиск в глубину (англ. depth-first search, DFS)

Пусть зафиксирована начальная вершина v_0 . Выберем смежную с ней вершину v_1.

Затем для вершины v_1 выбираем смежную с ней вершину из числа ещё не выбранных вершин и т.д.: если мы уже выбрали вершины $v_0, v_1, \dots, v_k$, то следующая вершина выбирается смежной с вершиной v_k из числа невыбранных. Если для вершины v_k такой вершины не нашлось, то возвращаемся к вершине v_{k-1} и для неё ищем смежную среди невыбранных.

При необходимости возвращаемся назад и т.д. Так будут перебраны все вершины графа и поиск закончится.

Поиск в глубину



Алгоритм поиска в глубину

Пусть задан граф $G = (V, E)$. Предположим, что в начальный момент времени все вершины графа окрашены в белый цвет (не посещённые). Выполним следующие действия:

- Пройдём по всем вершинам $v \in V$.
  - Если вершина v белая, выполним для неё $\mathrm{DFS}(v)$.

Процедура DFS (параметр — вершина $u \in V$)

- Перекрашиваем вершину u в серый цвет (обнаруженная).
- Для всякой вершины w, смежной с вершиной u и окрашенной в белый цвет, рекурсивно выполняем процедуру $\mathrm{DFS}(w)$.
- Перекрашиваем вершину u в чёрный цвет (обработанная – все смежные просмотрены).

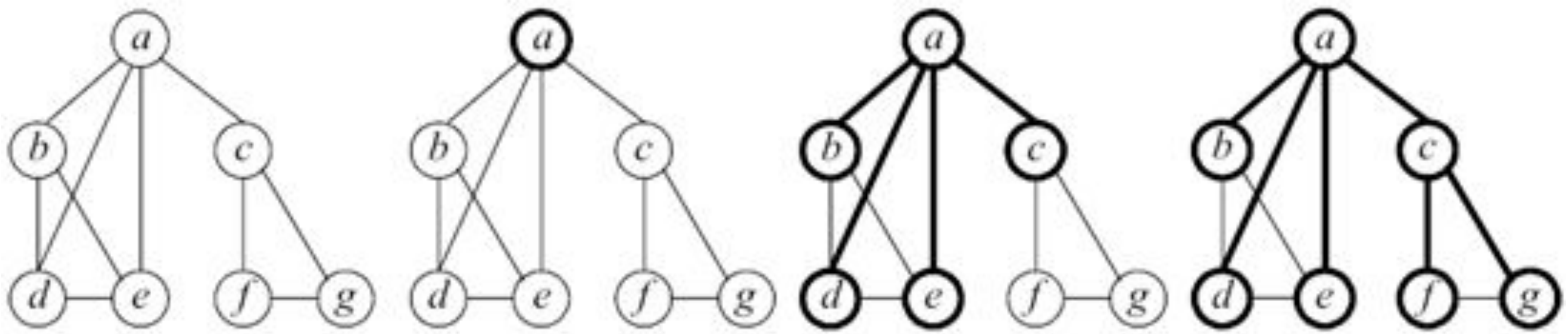
Применения алгоритма поиска в глубину:

- Поиск маршрута в графе;
- Поиск простого цикла;
- Выделение компонент связности;
- Проверка на двудольность;
- Проверка, является ли одна вершина дерева предком другой;
- Поиск наименьшего общего предка;
- Топологическая сортировка (перенумеровать вершины графа таким образом, чтобы каждое ребро вело из вершины с меньшим номером в вершину с большим);
- Поиск мостов, точек сочленения;
- Проход по лабиринту с одним шагом видимости. «Правило левой руки» (идти, ведя левой рукой по стенке) будет поиском в глубину, если лабиринт древовидный.

Поиск в ширину

Поиск в ширину (англ. breadth-first search, BFS)

Пусть зафиксирована начальная вершина v_0 . Рассматриваем все смежные с ней вершины $v_1, v_2, \ldots, v_k$. Затем рассматриваем смежные вершины каждой из рассмотренных вершин $v_1, v_2, \ldots, v_k$, и т.д. Так будут перебраны все вершины графа и поиск закончится.



Поиск в ширину

Пусть задан граф $G = (V, E)$. Предположим, что в начальный момент времени все вершины графа окрашены в белый цвет (не посещённые). Выполним следующие действия:

- Начальная вершина помечается как серая и добавляется в очередь.
- Пока очередь не пуста (пока есть серые вершины)
  - Из очереди берётся очередная вершина. Все смежные с ней белые вершины перекрашиваются в серый и добавляются в очередь.
  - Вершина перекрашивается в чёрный.

Применения алгоритма поиска в ширину

- Выделение компонент связности (подсчёт компонент);
- Поиск кратчайшего пути в невзвешенном графе;
- Поиск в пространстве состояний: нахождение решения задачи с наименьшим числом ходов, если каждое состояние системы можно представить вершиной графа, а переходы из одного состояния в другое — рёбрами графа;
- Нахождение кратчайшего цикла в ориентированном невзвешенном графе;
- Нахождение всех вершин и рёбер, лежащих на каком-либо кратчайшем пути между двумя вершинами.

Числовые характеристики графа

Расстояние между вершинами

Расстоянием между связными вершинами v_i и v_j называется длина минимальной незамкнутой простой цепи, связывающей их.

Расстояние обозначается $d(v_i, v_j)$.

Аксиомы:

1. $d(v_i, v_i) = 0$
2. $d(v_i, v_j) = d(v_j, v_i)$;
3. $d(v_i, v_j) \leq d(v_i, v_m) + d(v_m, v_j)$

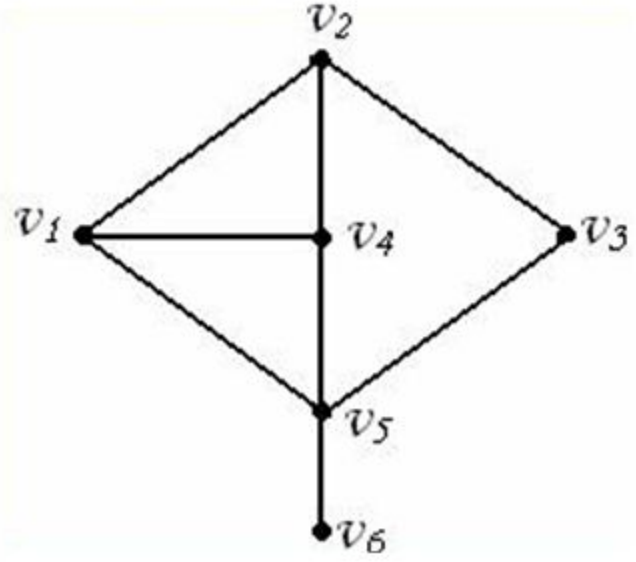
Расстояние между несвязными вершинами обычно считается равным бесконечности

$$d(v_i, v_j) = \infty$$

Эксцентриситет

Для фиксированной вершины v_j максимальное из расстояний от этой вершины до любой другой вершины графа называется ***эксцентриситетом*** вершины v_j.

$$e(v_j) = \max_{v_i \in V_G, v_i \neq v_j} \left(d(v_j, v_i) \right)$$



0	1	2	1	1	2
1	0	1	1	2	3
2	1	0	2	1	2
1	1	2	0	1	2
1	2	1	1	0	1
2	3	2	2	1	0

Диаметр графа

Максимальный среди всех эксцентриситетов вершин называется ***диаметром*** графа G *и* обозначается через $D(G)$.

$$D(G) = \max_{v_i \in V_G} e(v_i)$$

Вершина v_i называется ***периферийной***, если $e(v_i) = D(G)$.

Радиус графа

Минимальный из эксцентриситетов вершин связного графа называется его ***радиусом*** и обозначается через $R(G)$:

$$R(G) = \min_{v_i \in V_G} e(v_i)$$

Вершина *v* называется ***центральной***, если $e(v_i) = R(G)$.

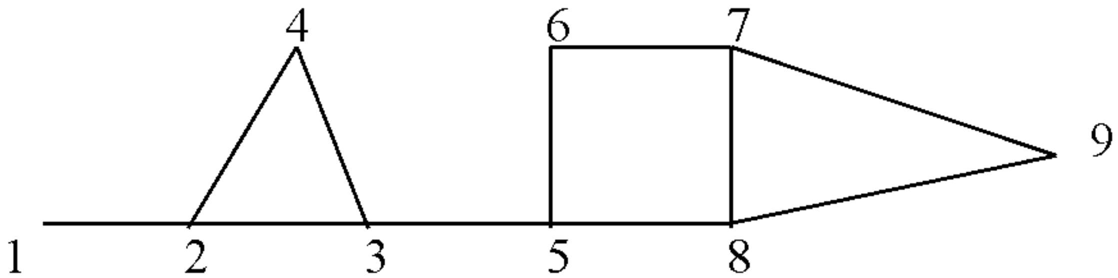
Центр графа

Множество всех центральных вершин графа называется его ***центром*** $Z(G)$.

Граф может иметь единственную центральную вершину или несколько центральных вершин.

Центр графа может совпадать с множеством всех вершин.

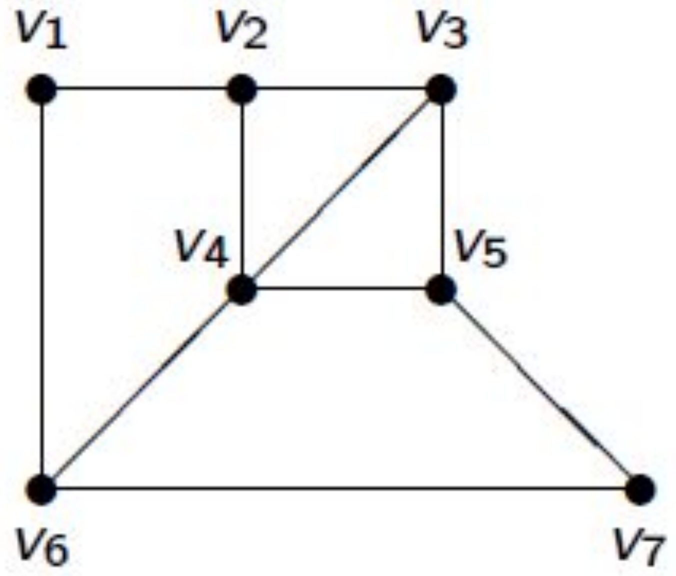
Пример



□□1□ = 5, □□2□ = 4, □□3□ = 3,
□(4) = 4, □(5) = 3, □(6) = 4,
□(7) = 5, □(8) = 4, □(9) = 5

□□□□ = 3;   □□□□□ = 5;
□□□□□ = □3,5□;
□(□□) = {1,7,9}

Найти диаметр, радиус и центры графа

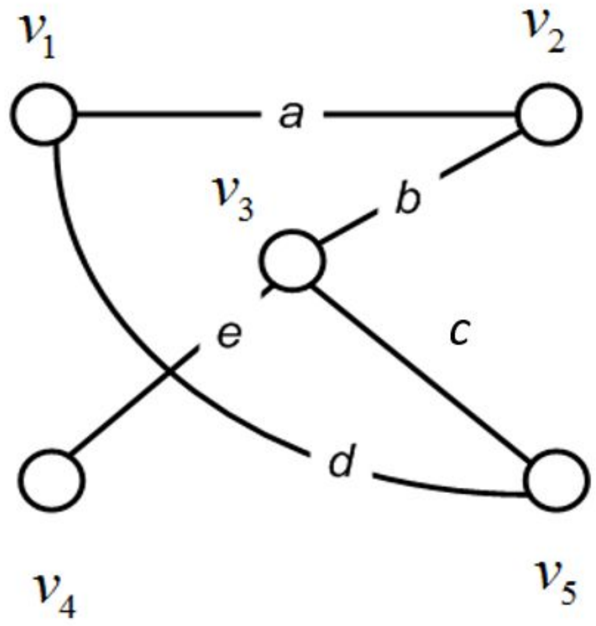


	v_1	v_2	v_3	v_4	v_5	v_6	v_7	$r(v_i)$
v_1	0	1	2	2	3	1	2	3
v_2	1	0	1	1	2	2	3	3
v_3	2	1	0	1	1	2	2	2
v_4	2	1	1	0	1	1	2	2
v_5	3	2	1	1	0	2	1	3
v_6	1	2	2	1	2	0	1	2
v_7	2	3	2	2	1	1	0	3

$D(G) = 3, R(G) = 2,$
$Z(G) = \{v_3, v_4, v_6\},$
$P(G) = \{v_1, v_2, v_5, v_7\}$

Задачи

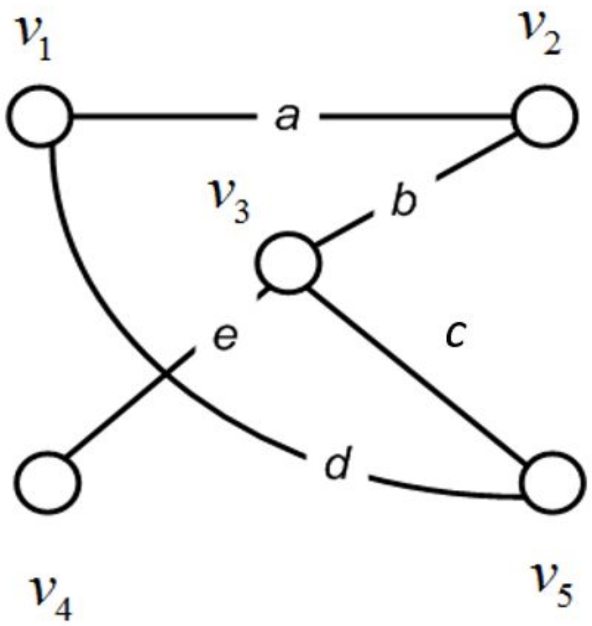
Перечислить вершины и рёбра графа.



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

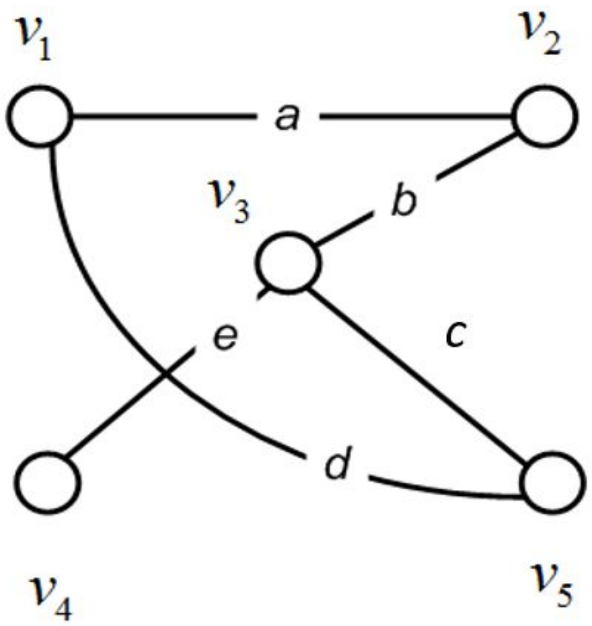
$$E = \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_1, v_5\}\{v_3, v_4\}\{v_3, v_5\}\}$$

Построить матрицу инцидентности



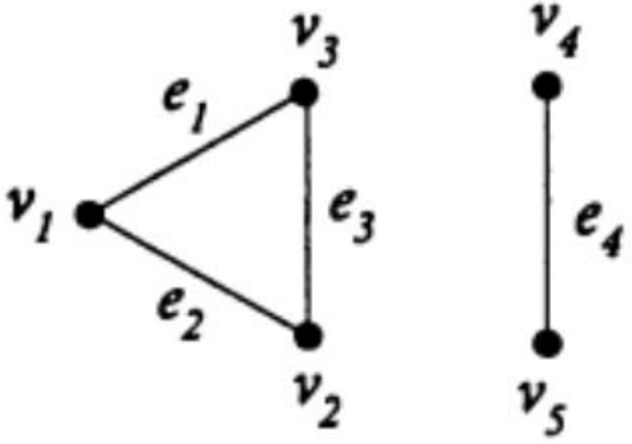
	a		*c*		
	1	0	0	0	0
	1	1	0	0	0
	0	1	1	0	1
	0	0	0	0	1
	0	0	1	1	0

Построить матрицу смежности



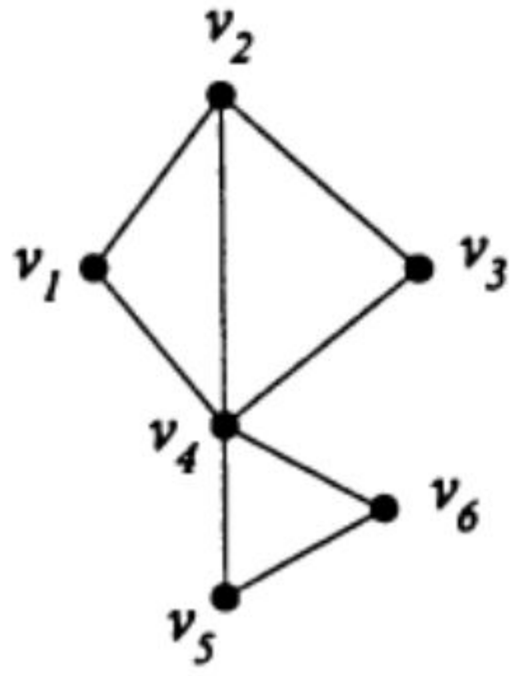
	0	1	0	0	1
	1	0	1	0	0
	0	1	0	1	1
	0	0	1	0	0
	1	0	1	0	0

Найти все пары вершин, между которыми существует маршрут длины 2



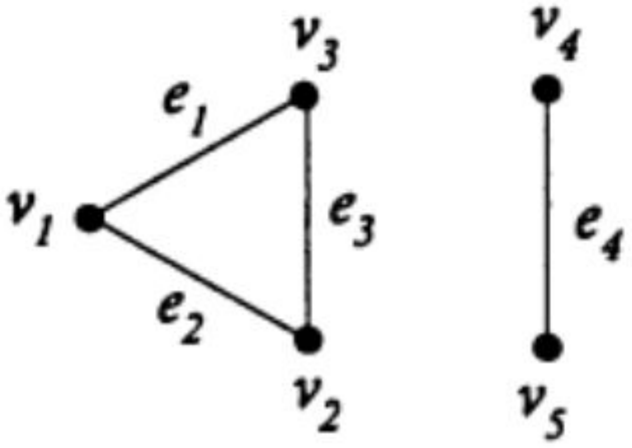
	1	1	1	0	0
	1	1	1	0	0
	1	1	1	0	0
	0	0	0	1	0
	0	0	0	0	1

Найти все пары вершин, между которыми существует маршрут длины 3



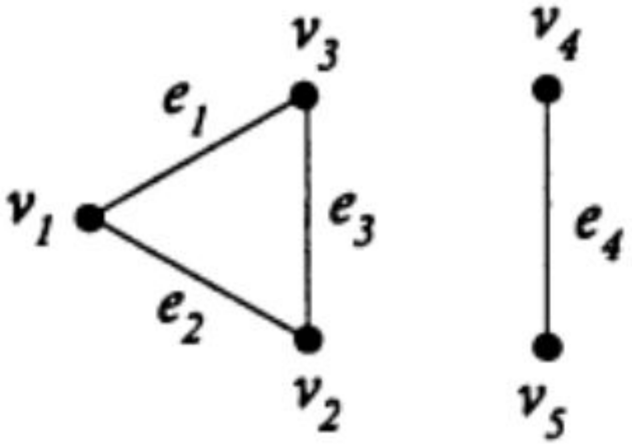
	1	1	1	1	1	1
	1	1	1	1	1	1
	1	1	1	1	1	1
	1	1	1	1	1	1
	1	1	1	1	1	1
	1	1	1	1	1	1

Найти все пары вершин, между которыми существует маршрут длины не меньше 3



	1	1	1	0	0
	1	1	1	0	0
	1	1	1	0	0
	0	0	0	1	1
	0	0	0	1	1

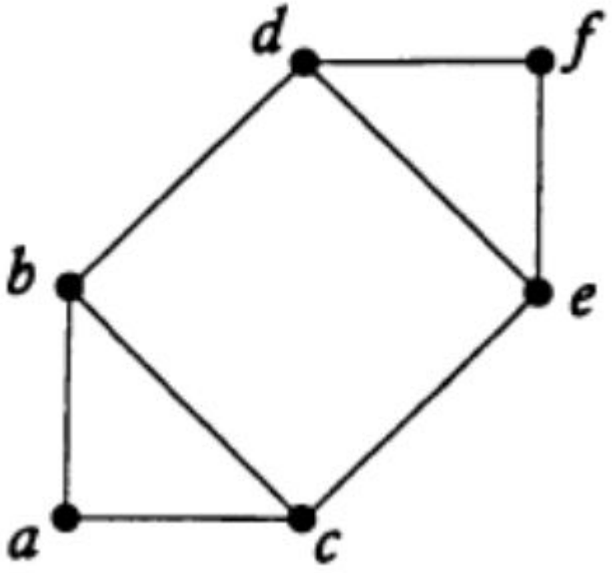
Определить количество маршрутов длины 2



	2	1	1	0	0
	1	2	1	0	0
	1	1	2	0	0
	0	0	0	1	0
	0	0	0	0	1

Определить длину маршрута

Определить минимальную длину маршрута, между любыми двумя вершинами



$A^{\odot 3} =$

	1	1	1	1	1	1
	1	1	1	1	1	1
	1	1	1	1	1	1
	1	1	1	1	1	1
	1	1	1	1	1	1
	1	1	1	1	1	1