Дискретна математика

Зима-Весна

Лекція 6. Теорія графів

Терновой Максим Юрійович
к.т.н., с.н.с., доцент кафедри інформаційно-телекомунікаційних мереж

Питання, які будуть розглянуті

1. Основні поняття теорії графів
2. Подання графа за допомогою матриці інцидентності
3. Подання графа за допомогою матриці суміжності графа
4. Визначення локальних степенів вершин графа. Повні графи
5. Ізоморфізм графів
6. Частини графа, суграфи й підграфи
7. Графи і бінарні відношення
8. Маршрути, шляхи, ланцюги та цикли
9. Зв'язність
10. Дерева
11. Кістякове дерево зв'язного графа
12. Ейлерові графи
13. Гамільтонові графи
14. Планарність графів
15. Задачі пошуку маршрутів в графі
16. Пошук відстані між вершинами графа
17. Мінімальні шляхи у зважених орієнтованих графах

1. Основні поняття теорії графів

Загальні поняття

Теорія графів – розділ дискретного аналізу, який має широке застосування в багатьох наукових дисциплінах завдяки тому, що поняття і інструментарій цієї теорії виявився дуже зручним для дослідження та інтерпретації різноманітних проблем у великої кількості наук: кібернетиці, технічній і економічній, теорії автоматів, теорії управління, теорії інформації тощо.

Загальні поняття

За допомогою інструментарію теорії графів розв'язується велика кількість економіко-математичних задач, зокрема наведена у вступі задача „про призначення", задача календарного планування, мережного планування тощо. Взагалі теорія графів має велике значення у телекомунікаціях.

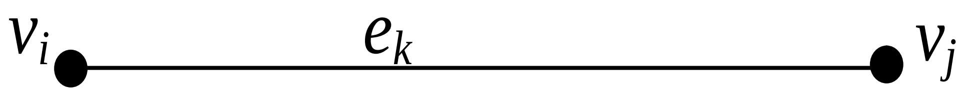
Граф

Означення 6.1. *Нехай V – довільна множина, E – деяка сукупність пар вигляду* (γ_i, γ_j) *, де* γ_i, γ_j $\in$ *V. Упорядкована пара G=(V,E), яка складається з множини V та сукупності E, називається* ***графом*** *із множиною вершин V і множиною ребер E.*

Іншими словами, графом називається пара двох множин: вершини *V* і ребер *E*, що зв’язують вершини. Термін сукупність означає можливість наявності однакових пар.

Зображення графів

При графічному зображенні графу елементи множини V зображають точками на площині, а ребра (v_i, v_j) – відрізками (прямолінійними або криволінійними), які з'єднують точки v_i і v_j. Ребра зазвичай позначаються e_k



Ребро e_k називається **інцидентним** по відношенню до вершин v_i і v_j .

Граф. Основні поняття

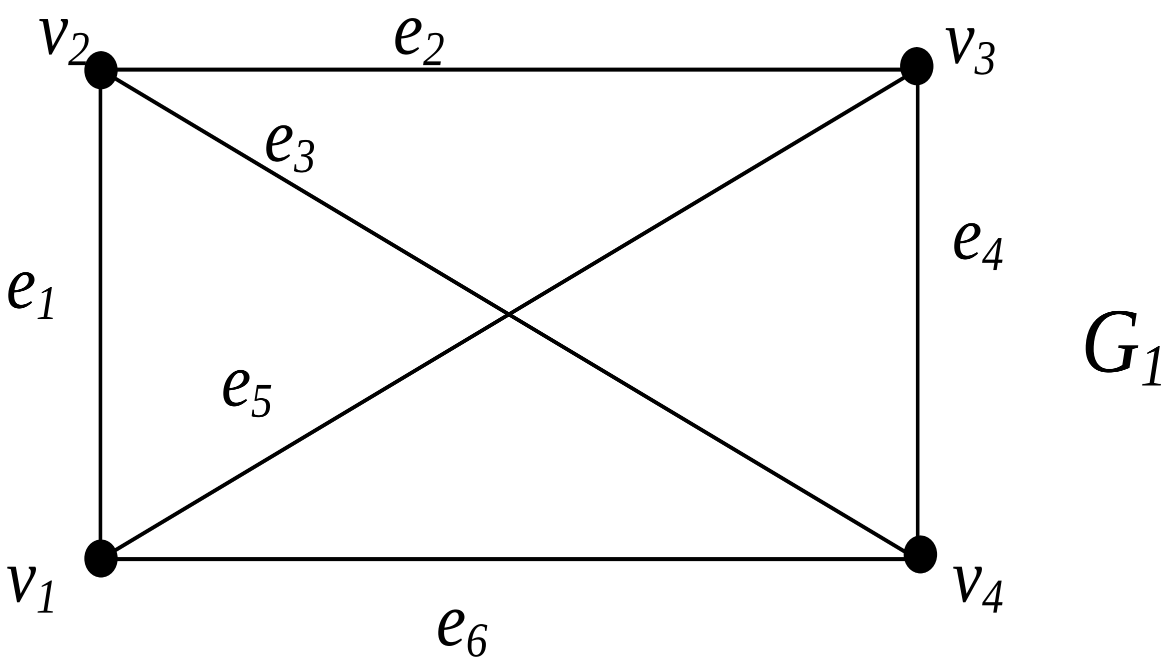
Граф називається **скінченим**, якщо множини його вершин і ребер є скінченими.

Множину вершин графа G позначають *V(G)*, а **множину ребер** – *E(G)*.

Кількість вершин графа – *n(G)*, а **кількість ребер графа** – *m(G)*.

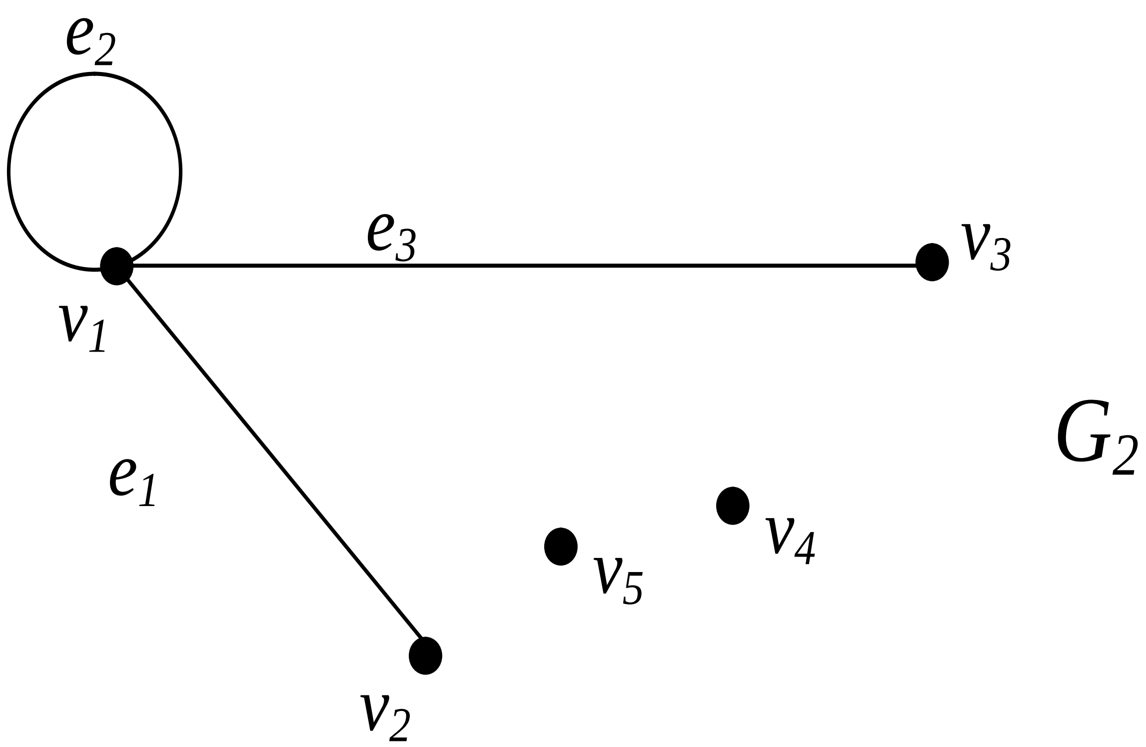
Означення 6.2. *Кількість вершин n(G) графа називають його* ***порядком***.

Граф. Приклад 1



Граф G_1 має множину вершин $V(G_1) = \{v_1, v_2, v_3, v_4\}$ і множину ребер $E(G_1) = \{e_1, e_2, e_3, e_4, e_5, e_6\}$. $n(G_1) = 4$, тобто порядок графа G_1 дорівнює 4, $m(G_1)=6$.

Граф. Приклад 2



Граф G_2 має множину вершин $V(G_2) = \{v_1, v_2, v_3, v_4, v_5\}$ і множину ребер $E(G_2) = \{e_1, e_2, e_3\}$. $n(G_2)=5$, $m(G_1)=3$. Це граф порядку 5.

Граф. Основні поняття

Означення 6.3. Кількість ребер графа, інцидентних деякій вершині γ, називається **локальним степенем**, або просто степенем вершини γ і позначається $\rho(\gamma)$. Якщо є ребро e в графі G, тобто $e = (\gamma, \omega) \in E(G)$, то можна сказати:

– вершини γ і ω **суміжні** в графі G;
– вершини γ і ω є **кінцями** ребра e;
– вершини γ та ω **інцидентні** ребру e;
– ребро e **інцидентне** вершині γ (ω).

Означення 6.4. *Два ребра називаються* ***суміжними***, *якщо обидва вони інцидентні одній вершині.*

Граф. Приклад 3

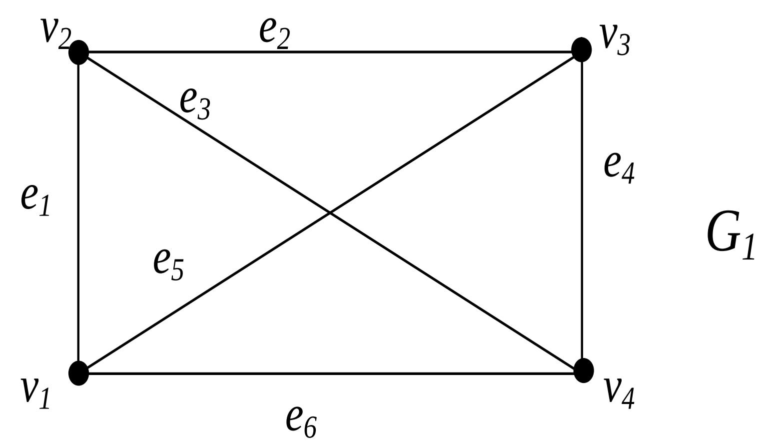
В графі G_1 суміжними є всі вершини $\gamma_1, \gamma_2, \gamma_3, \gamma_4$, суміжні трійки ребер

e_1, e_2, e_3;

e_1, e_5, e_6;

e_6, e_3, e_4;

e_2, e_3, e_4.



Розглянемо ребро e_2. Вершини γ_2 і γ_3 є кінцями ребра e_2.

Вершини γ_2 і γ_3 інцидентні ребру e_2. Ребро e_2 інціндентне вершинам γ_2 і γ_3.

Вершини $\gamma_1, \gamma_2, \gamma_3, \gamma_4$ всі мають однаковий локальний ступінь, який дорівнює 3.

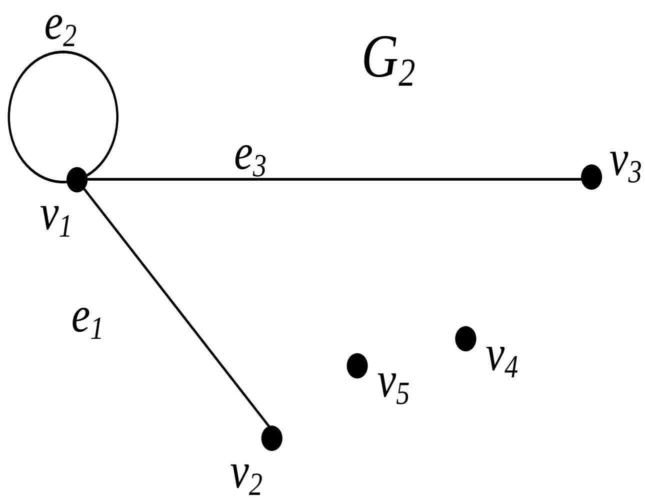
Граф. Приклад 4

В графі G_2 не всі вершини суміжні між собою.

Вершини γ_4 і γ_5 не суміжні жодній вершині графу. Їхній локальний ступінь $\rho(\gamma_4) = \rho(\gamma_5) = 0$.

Ребро e_3 інцідентне вершинам γ_1 і γ_3, а вершини γ_1 і γ_3 інцідентні ребру e_5. $\rho(\gamma_1) = 1$, $\rho(\gamma_3) = 1$, $\rho(\gamma_2) = 1$.

Ребра e_1, e_2, e_3 суміжні.



Граф. Основні поняття

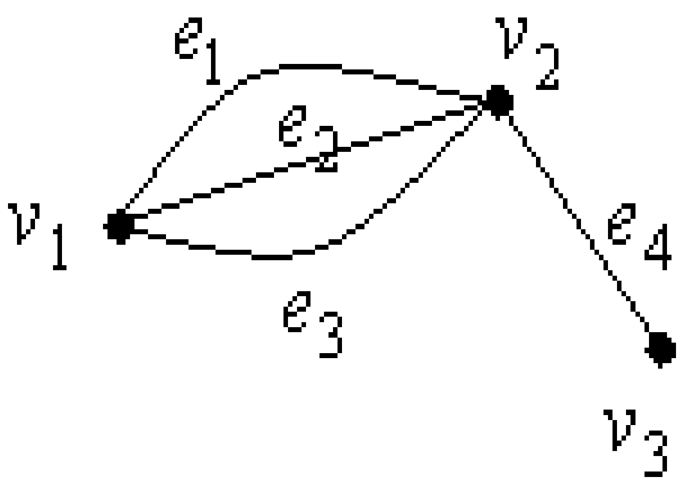
Означення 6.5. *Граф з порожньою множиною ребер називається* ***нуль-графом****. Якщо ж множина вершин V – порожня, то порожня і множина E. Такий граф називається* ***порожнім****.*

Лінії, що зображають ребра графа, можуть перетинатися, але точки перетину не є вершинами (перетин ребер e_3 і e_5 графу G_1 в прикладі 1).

Граф. Основні поняття

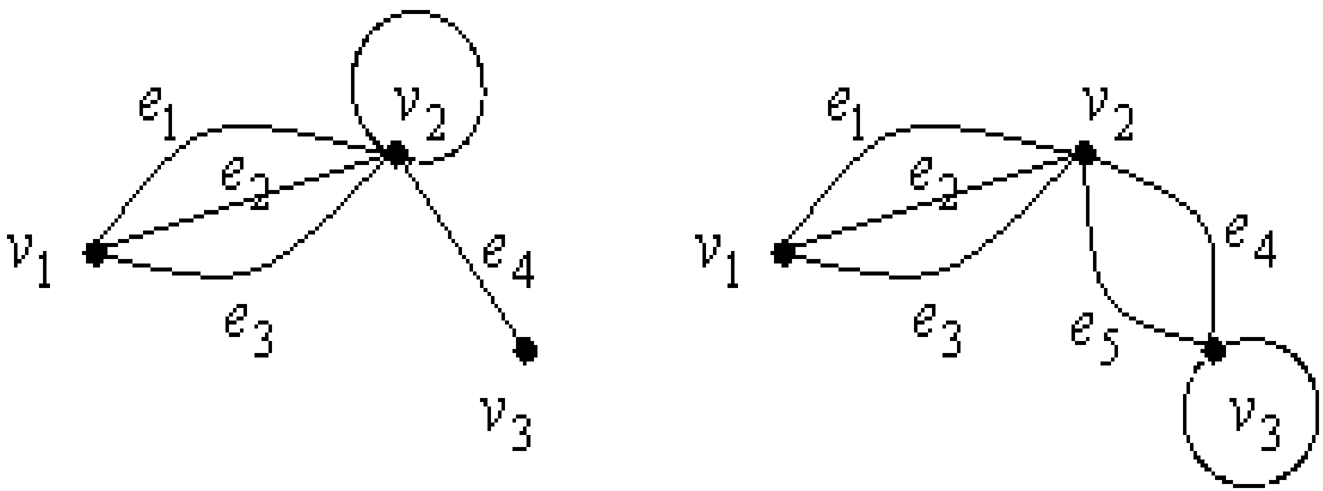
Різні ребра можуть бути **інцидентні** одній і тій самій парі вершин, такі ребра називаються ***кратними***.

Цей випадок відповідає наявності декількох однакових пар $(\gamma_i, \gamma_j) \in E(G)$.



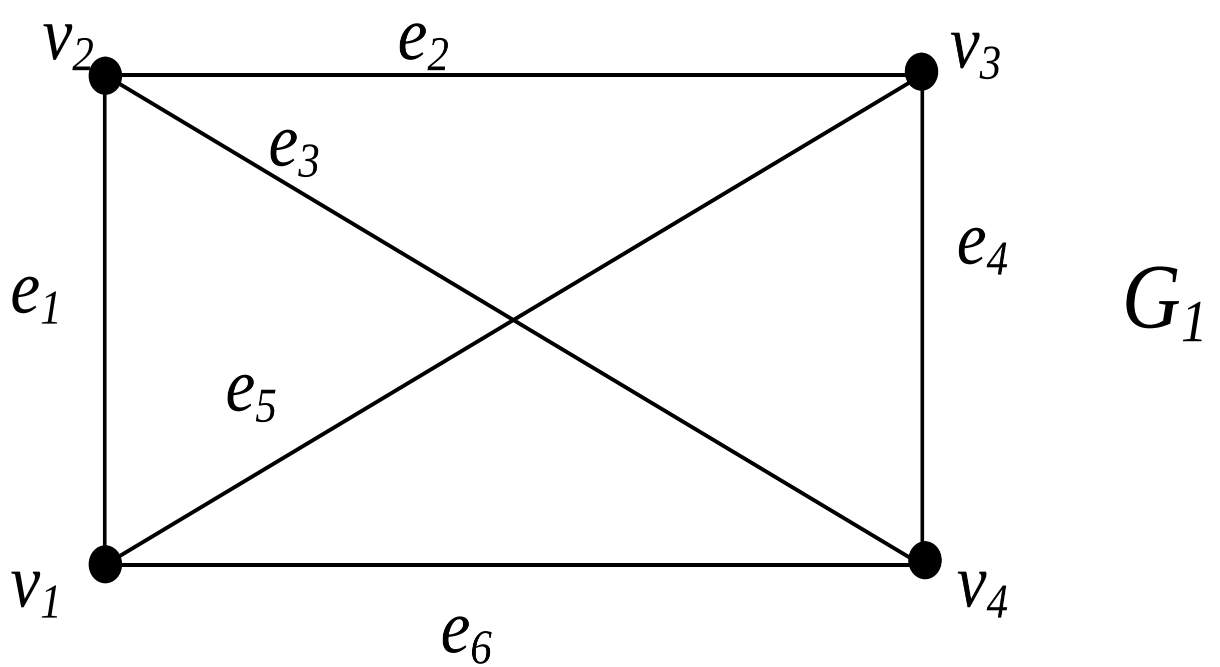
Граф. Основні поняття

Означення 6.6. *Граф, що містить кратні ребра, називається* ***мультиграфом****. Ребро може з'єднувати деяку вершину саму з собою, таке ребро називається* ***петлею****. Цей випадок відповідає наявності в множині пар вигляду (γ, γ). Граф з петлями та кратними ребрами називається* ***псевдографом****.*



Граф. Основні поняття

Означення 6.7. *Скінченний неорієнтований граф без петель і кратних ребер називається* ***звичайним***.

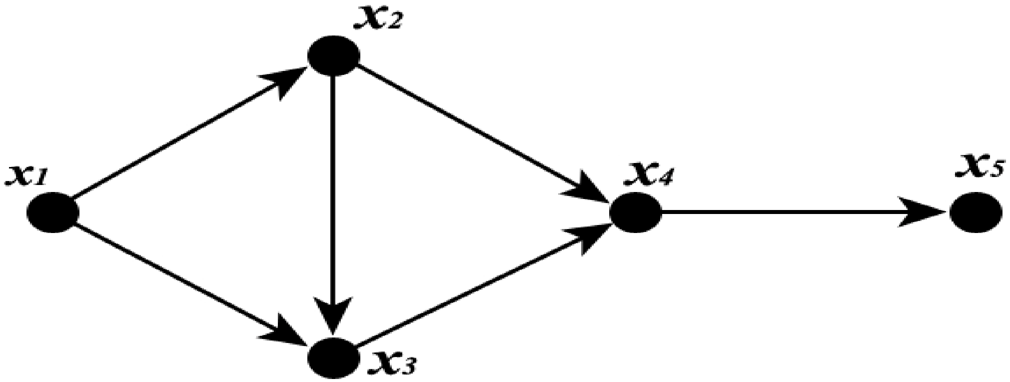


Орієнтовані графи. Основні поняття

Означення 6.8. *Якщо пари* $(\gamma, \omega) \in E$ *вважаються упорядкованими, то граф називається* ***орієнтованим*** *або* ***орграфом****. Інакше граф називається* ***неорієнтованим****. Ребра орієнтованого графа прийнято називати* ***дугами*** *та зображати направленими відрізками.*

При зображенні орієнтованих графів напрями ребер позначаються стрілками. Орієнтований граф також може мати кратні ребра, петлі, а також ребра, що з’єднують одні й ті самі вершини ребра, але у зворотних напрямах.

Орієнтовані графи. Приклад



$V=\{x_1;x_2;x_3;x_4;x_5\}$;

$E=\{(x_1;x_2);(x_2;x_4);(x_1;x_3);(x_2;x_3);(x_3;x_4);(x_4;x_5)\}$.

Наприклад: $l=(x_4;x_5)$ то x_4 – початок дуги, x_5 – кінець.

Зв`язок орієнтованих і неорієнтованих графів

Кожному неорієнтованому графу можна поставити у відповідність орієнтований граф з тією ж самою множиною вершин, в якій кожне ребро замінено двома орієнтованими ребрами, що є інцидентними тим же вершинам і мають зворотні напрями.

Мішані графи

Означення 6.9. *Граф, що має як ребра, так і дуги, називається* ***мішаним***.

Про дугу (γ, ω) кажуть, що вона виходить з вершини γ та входить у вершину ω. Іноді вершини γ і ω називають відповідно початком дуги та кінцем дуги (γ, ω).

Домовимося позначати орграфи літерою D або D з індексами.

2. Подання графа за допомогою матриці інцидентності

Матриця інцидентності

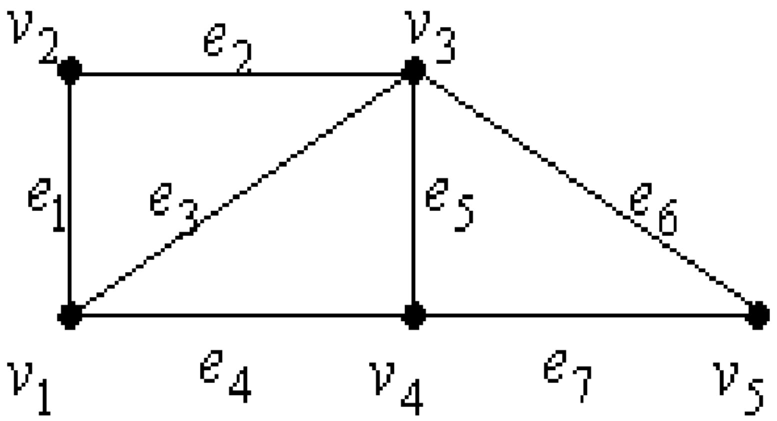
Задати граф означає задати множини його вершин і ребер, а також відношення інцидентності. Розглянемо звичайні графи. Нехай $v_1, v_2, \dots, v_n$ – вершини графа G; $e_1, e_2, \dots, e_m$ – його ребра.

Відношення інцидентності можна означити матрицею $E = \|\varepsilon_{ij}\|$, яка має m рядків і n стовпців де

$$\varepsilon_{ij} = \begin{cases} 1, & \text{якщо ребро } e_i \text{ інцидентне вершині } V_j \\ 0 & \text{в протилежному випадку} \end{cases}.$$

Стовпці матриць відповідають вершинам графа, а рядки – його ребрам.

Приклад



Матриця інцидентості

	v_1	v_2	v_3	v_4	v_5
e_1	1	1	0	0	0
e_2	0	1	1	0	0
e_3	1	0	1	0	0
e_4	1	0	0	1	0
e_5	0	0	1	1	0
e_6	0	0	1	0	1
e_7	0	0	0	1	1

Матриця інцидентності. Продовження

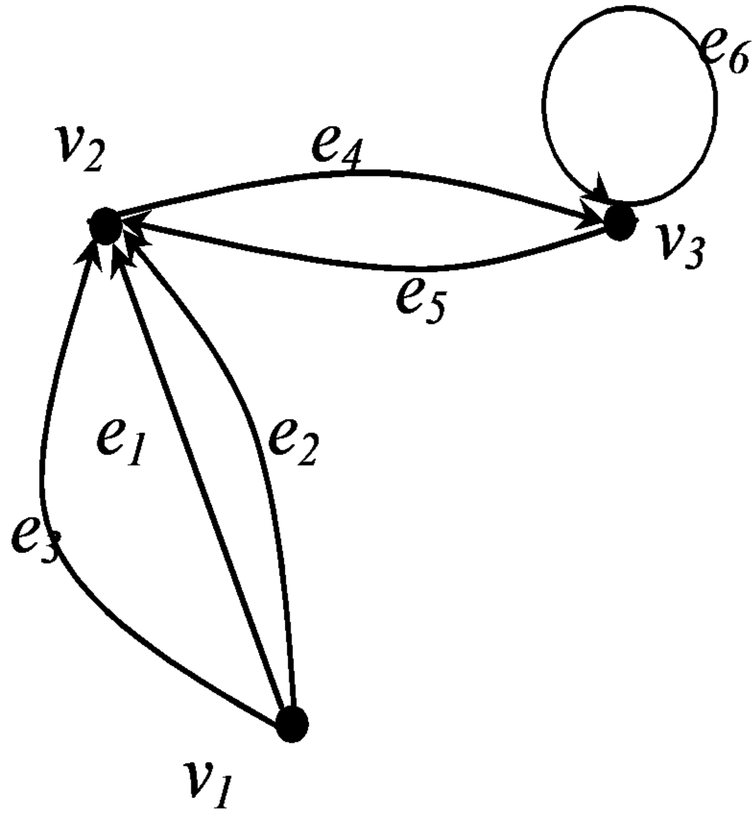
Сума у кожному рядку матриці інцидентності звичайного графу дорівнює 2, сума у кожному стовпці матриці інцидентності звичайного графу дорівнює локальному ступеню вершини, що відповідає даному стовпцю.

Якщо в неорієнтованому графі є петлі, тобто ребро, що є інцидентним одній і той самій вершині, то у відповідному рядку буде одна 1.

Матриця інцидентності орієнтованого графу

У матриці інцидентності $E = \|\varepsilon_{ij}\|$ орієнтованого графа G якщо вершина v_j – початок ребра e_i, то $\varepsilon_{ij} = -1$; якщо v_j – кінець e_i, то $\varepsilon_{ij} = 1$; якщо e_i – петля, а v_j – інцидентна їй вершина, тобто і початок і кінець одночасно, то $\varepsilon_{ij} = \alpha$, де α – будь-яке число, відмінне від 1, 0 і -1, в інших випадках $\varepsilon_{ij} = 0$.

Приклад



	v_1	v_2	v_3
e_1	-1	1	0
e_2	-1	1	0
e_3	-1	1	0
e_4	0	-1	1
e_5	0	1	-1
e_6	0	0	2

Матриця інцидентності

У кожному рядку матриці інцидентності для неорієнтованого або орієнтованого графа тільки два елементи відмінні від 0 (або один, якщо ребро є петлею), причому в орграфі сума елементів рядка, що не відповідає дузі, дорівнює 0.

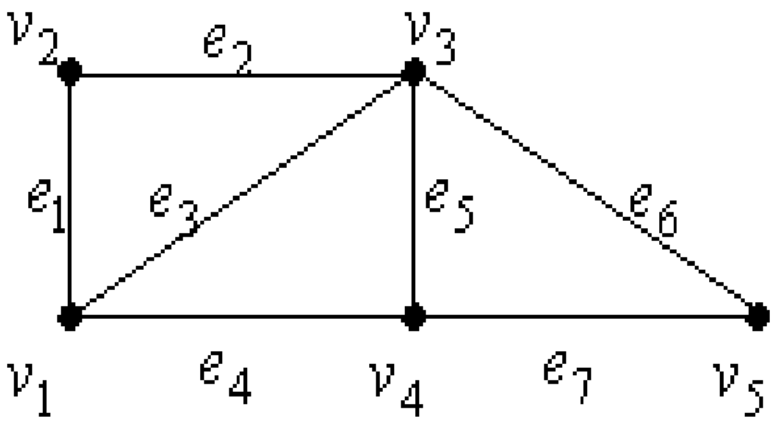
3. Подання графа за допомогою матриці суміжності графа

Матриця суміжності

Матриця суміжності графа - це квадратна матриця $\Delta = \|\delta_{ij}\|$, стовпцям і рядкам якої відповідають вершини графа.

Для неорієнтованого графа δ_{ij} дорівнює кількості ребер, інцидентних i-й і j-й вершинам, для орієнтованого графа цей елемент матриці суміжності відповідає кількості ребер з початком у i -й вершині й кінцем у j -й. Таким чином, матриця суміжності неорієнтованого графа симетрична ($\delta_{ij} = \delta_{ji}$), а орієнтованого – необов'язково.

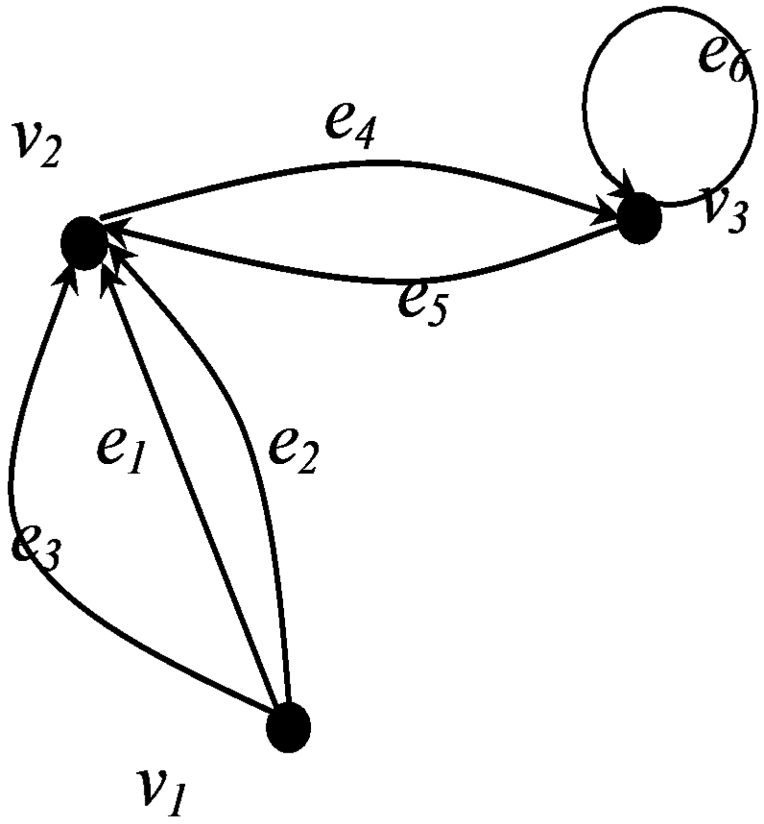
Матриця суміжності. Приклад



Матриця суміжності

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

Матриця суміжності. Приклад



	v_1	v_2	v_3
v_1	0	3	0
v_2	0	0	1
v_3	0	1	1

Матриця суміжності

Матриця суміжності повністю визначає відповідний неорієнтований або орієнтований граф. Число його вершин дорівнює розмірності матриці n, i -й і j -й вершинам графа інцидентні δ_{ij} ребер.

Матриця суміжності

Для неорієнтованого графа $\delta_{ij} = \delta_{ji}$, і всі його ребра визначаються верхнім правим трикутником матриці, розташованим над діагоналлю, включаючи останню. Кількість ребер графа дорівнює сумі δ_{ij} у цьому трикутнику, тобто $\sum_{i=1}^{n} \sum_{j=i}^{n} \delta_{ij}$. Ребра орієнтованого графа визначаються всіма елементами δ_{ij} матриці суміжності.

4. Визначення локальних степенів вершин графа. Повні графи

Визначення локальних степенів вершин графа

Якщо задані матриці суміжності Δ або інцидентності E графа, можна визначити локальні степені всіх його вершин.

Дійсно, в j -му стовпці матриці інцидентності, який відповідає вершині v_j, одиниці знаходяться на перетині з рядками, яким відповідають інцидентні цій вершині ребра, а інші елементи стовпця дорівнюють 0. Отже,

$$p(v_i) = \sum_{i=1}^{m} \varepsilon_{ij}$$
.

Визначення локальних степенів вершин графа

Елементи δ_{ij} матриці суміжності – це кількість ребер, інцидентних вершинам v_i і v_j.

Звідси

$$\rho(v_j) = \sum_{i=1}^{n} \delta_{ij}.$$

Визначення локальних степенів вершин орграфа

Для вершин орієнтованого графа визначаються два локальних степеня:

$\rho_1(v)$ ($\deg_{out}(v)$) – число ребер з початком у вершині v, або, інакше, кількість ребер, які виходять з v, і

$\rho_2(v)$ ($\deg_{in}(v)$)– кількість ребер, що входять у ребра v, тобто ребер, для яких ця вершина є кінцем. Петля дає внесок 1 у обидва ці степені.

Визначення локальних степенів вершин орграфа

Локальні степені вершин орієнтованого графа визначаються через коефіцієнти δ_{ij} його матриці суміжності: $p_1(v_i)=\sum_{j=1}^{n}\delta_{ij}, p_2(v_i)=\sum_{k=1}^{n}\delta_{ki}$.

Визначення локальних степенів вершин орграфа

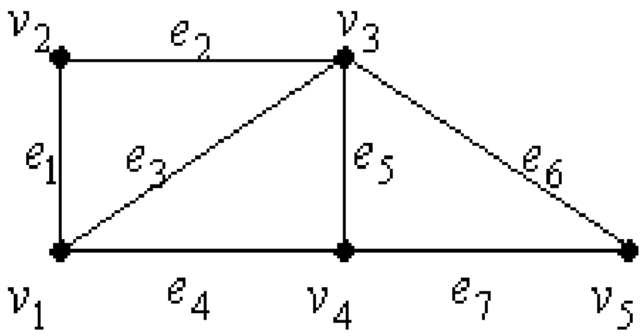
Вираз їх через коефіцієнти матриці інцидентності – значно складніше.

Оскільки кожне ребро орієнтованого графа G має один початок й один кінець, суми $\sum_{v \in G} \rho_1(v)$ та $\sum_{v \in G} \rho_2(v)$ дорівнюють кількості ребер цього графа, а отже, є рівними між собою:

$$\sum_{v \in V} \rho_1(v) = \sum_{v \in V} \rho_2(v) = m .$$

Визначення локальних степенів вершин орграфа. Приклад

Для графа, що заданий на рисунку, визначимо локальні ступені його вершин за матрицею інцидентності, підраховуючи суми одиниць в стовпцях.



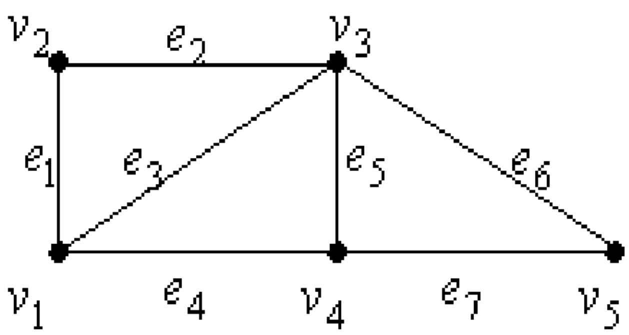
Матриця інцидентості

	v_1	v_2	v_3	v_4	v_5
e_1	1	1	0	0	0
e_2	0	1	1	0	0
e_3	1	0	1	0	0
e_4	1	0	0	1	0
e_5	0	0	1	1	0
e_6	0	0	1	0	1
e_7	0	0	0	1	1

$\rho(v_1)=3$, $\rho(v_2)=2$, $\rho(v_3)=4$, $\rho(v_4)=3$, $\rho(v_5)=2$.

Визначення локальних степенів вершин орграфа. Приклад

Для того ж самого графа визначимо локальні ступені його вершин за матрицею суміжності, підраховуючи суми одиниць в рядках (або в стовпцях) матриці:



Матриця суміжності

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

$\rho(v_1) = 3$, $\rho(v_2) = 2$, $\rho(v_3) = 4$, $\rho(v_4) = 3$, $\rho(v_5) = 2$.

Результати підрахунків за обома матрицями збігаються

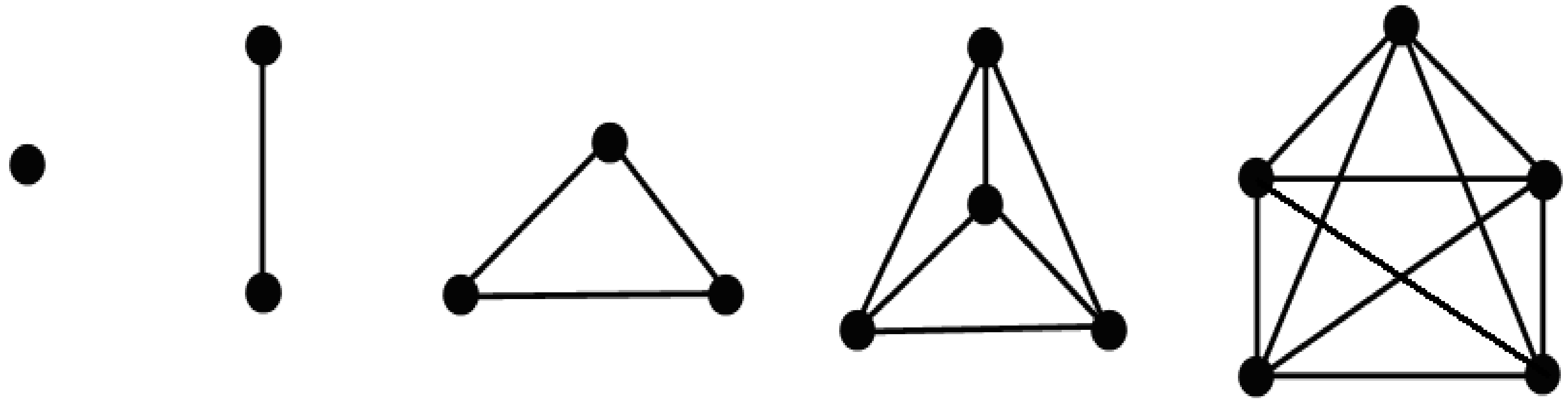
Повний граф

Означення 6.10. *Граф називається повним, якщо кожна пара його вершин сполучена ребром.*

У звичайного повного графа P_n (K_n)на n вершинах степені всіх вершин однакові і дорівнюють $(n\text{-}1)$.

Повний граф. Приклад

Повні графи K_1-K_5

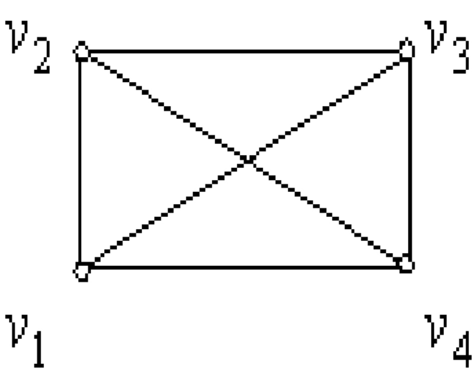


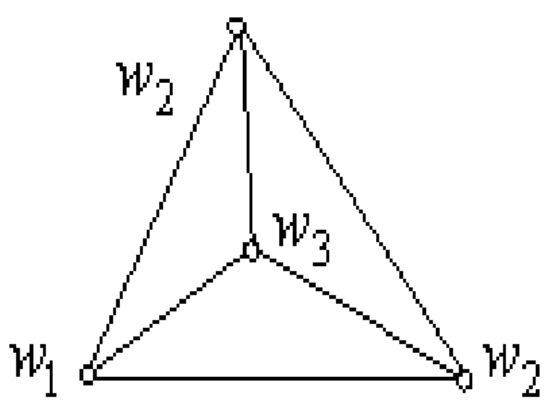
5. Ізоморфізм графів

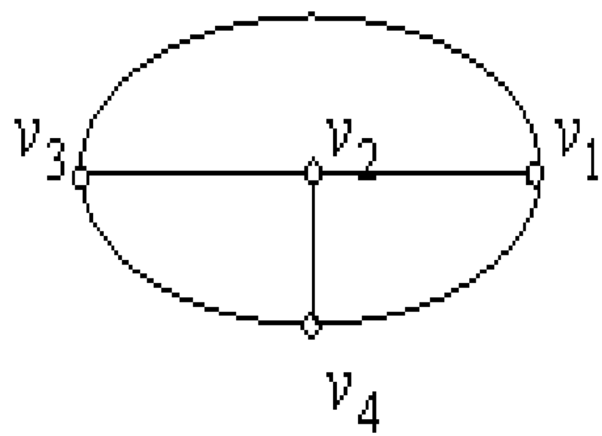
Способи представлення графу

Граф може бути представлений різними способами:
зображений на кресленні (рисунку),
заданий матрицею інцидентності,
матрицею суміжності.

Вигляд креслення залежить від форми ліній і взаємного розташування вершин. Іноді не так легко зрозуміти, чи однакові графи, зображені різними кресленнями:







Способи представлення графу

Вигляд матриць суміжності та інцидентності залежить від нумерації вершин і ребер графа.

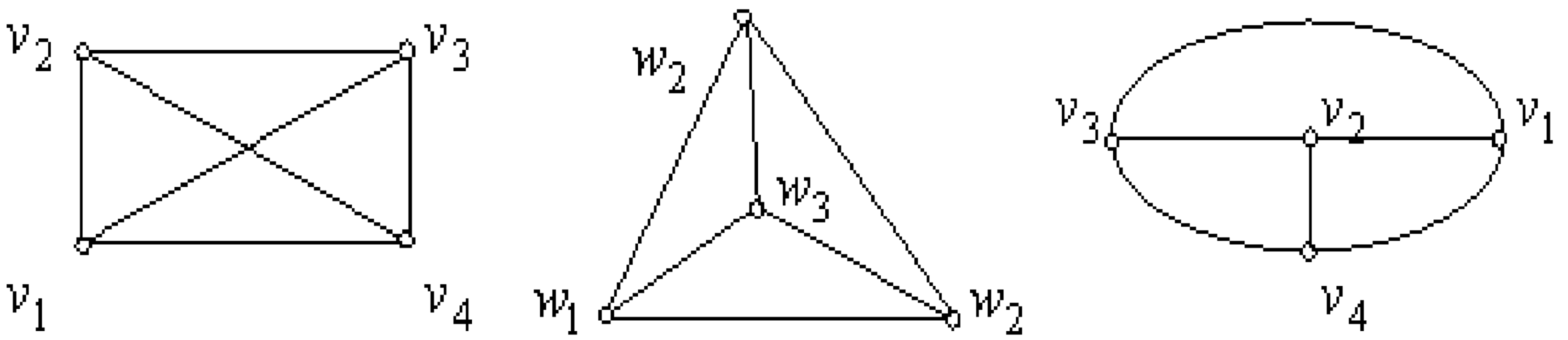
Строго кажучи, граф вважається повністю заданим, якщо нумерація його вершин зафіксовано.

Ізоморфізм графів

Означення 6.11. *Нехай існує бієкція* ϕ*, яка діє з множини вершин графа G на множину вершин графа H так, що для будь-яких вершин* v_1 *та* v_2 *графа G їхні образи* $\phi(v_1)$ *і* $\phi(v_2)$ *є суміжними в H тоді й тільки тоді, коли* v_1 *та* v_2 *– суміжні в G. Така бієкція називається* ***ізоморфізмом*** *графа G на граф H, а графи G і H називаються ізоморфними.*

Ізоморфізм графів

Іншими словами, графи, що мають однакову кількість вершин і ребер, є ізоморфними, якщо вони відрізняються тільки позначенням та нумерацією вершин і ребер. Таким чином, графи, зображені на рисунку, ізоморфні. Можна сказати, що це різні зображення одного графа на чотирьох вершинах.



Ізоморфізм графів

Щоб дізнатися, чи зображують дві матриці суміжності ізоморфні графи, можна, наприклад, здійснити всілякі перестановки рядків та стовпців першої матриці.

Якщо після однієї з цих перестановок виникне матриця, тотожно співпадаюча з другою, графи, які зображаються цими матрицями суміжності, будуть ізоморфними.

Проте, щоб пересвідчитися таким способом у тому, що графи не є ізоморфними, доведеться виконати всі n! перестановок рядків і стовпців, а це – досить трудомістка операція.

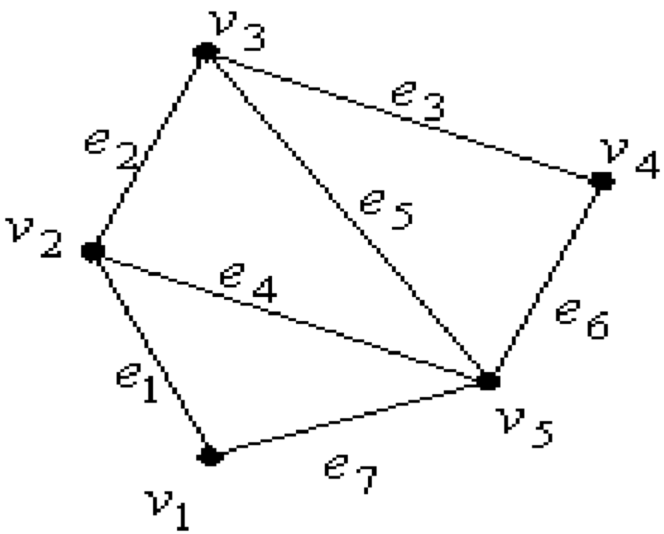
6. Частини графа, суграфи й підграфи

Частина графа

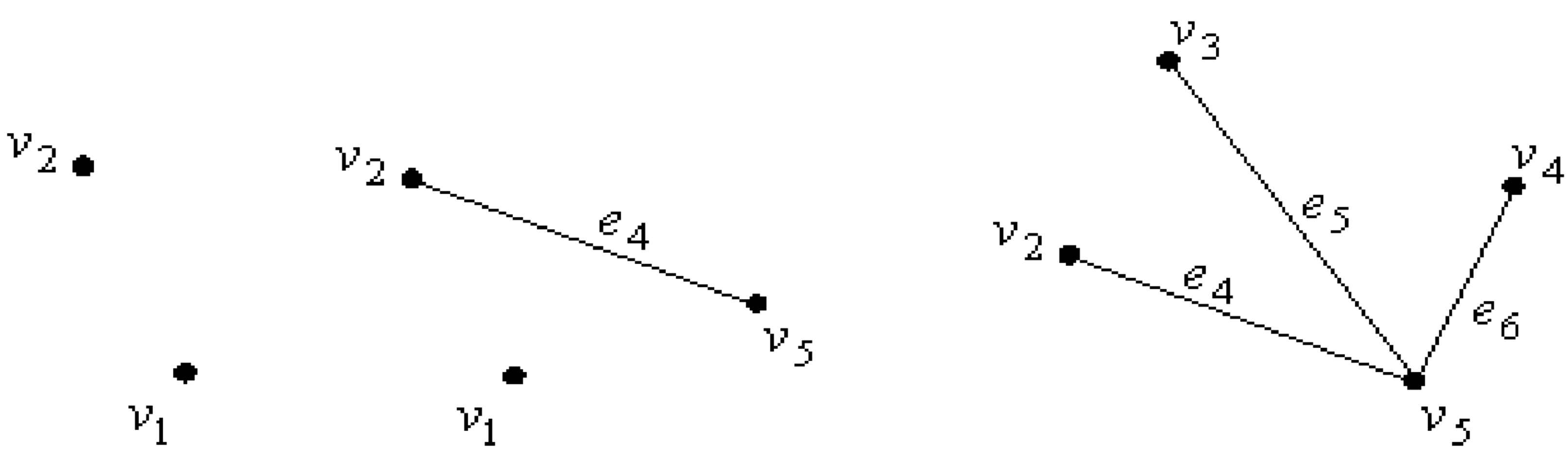
Означення 6.12. *Граф H називається* ***частиною*** *графа G(H $\subset$ G), якщо множина його вершин V(H) міститься в множині V(G), а множина E(H) ребер – в E(G). Якщо V(H) = V(G), то частина графа називається суграфом.*

Частина графа. Приклад

Для графа наведемо декілька частин:

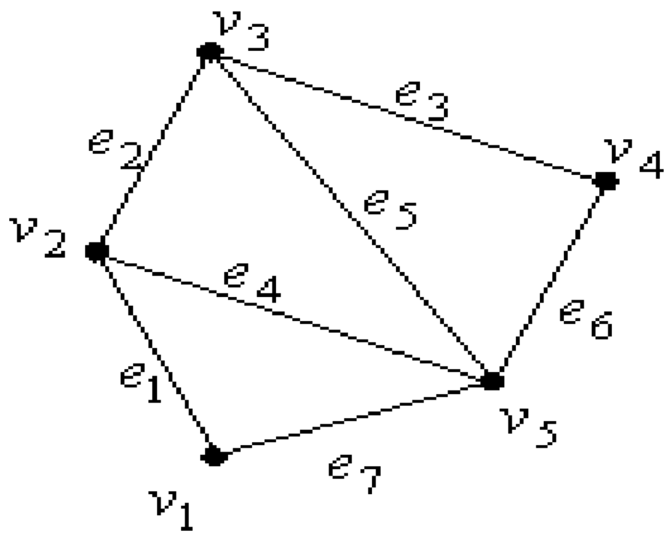


Частини графа

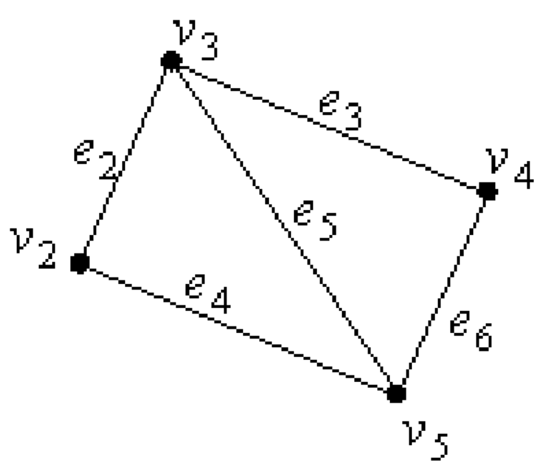


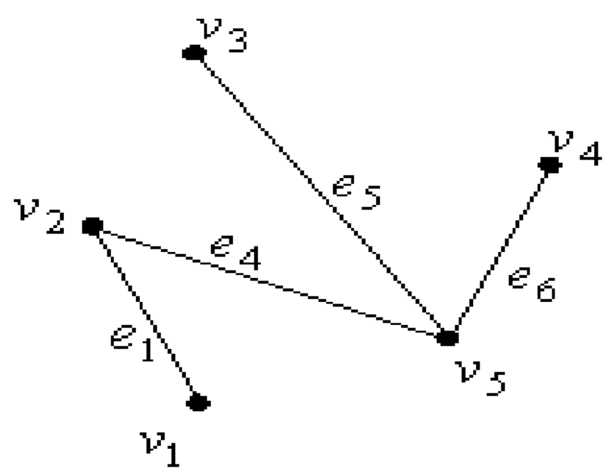
Частина графа. Приклад

Для графа наведемо декілька частин:



Частини графа





-суграф

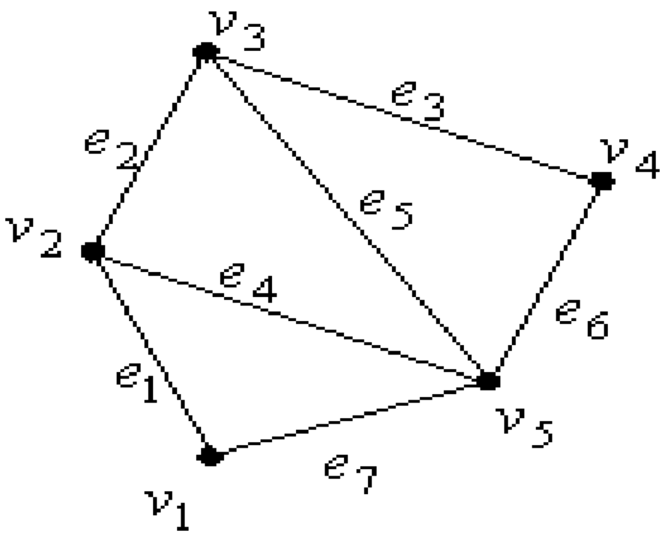
Частина графа

Наприклад, існує нульовий суграф, множина ребер якого є порожньою. Суграф H **покриває** вершини неорієнтованого графа G (або є **покривним**), якщо будь-яка вершина останнього – інцидентна хоча б одному ребру з H.

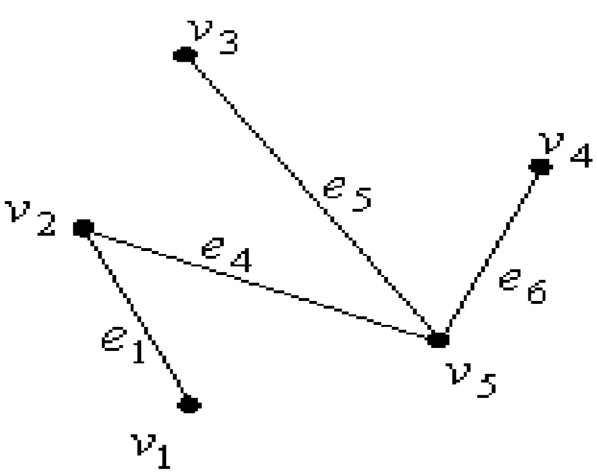
Таким чином, якщо в графі G існує ізольована вершина v, не інцидентна жодному ребру, покривного суграфа цього графа не існує.

Покривний суграф. Приклад

Для графа:



Покривний суграф



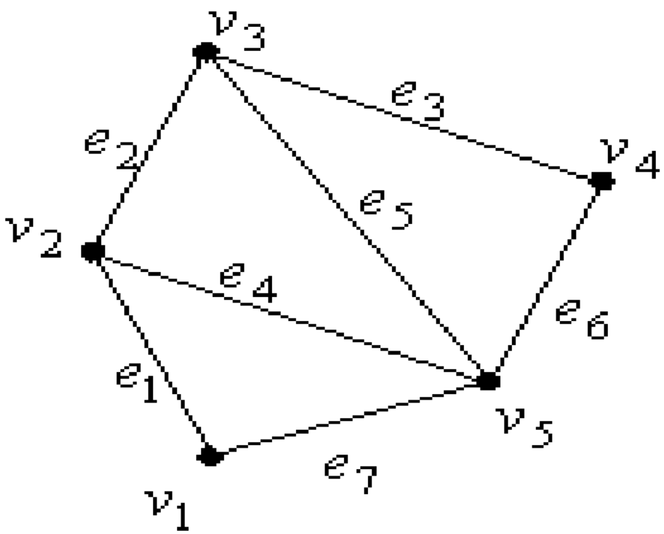
Підграф

Означення 6.13. ***Підграфом*** *графа G називається частина графа з множиною вершин $U \subset V(G)$, якщо її ребрами є всі ребра з $E(G)$, обидва кінці яких належать U.*

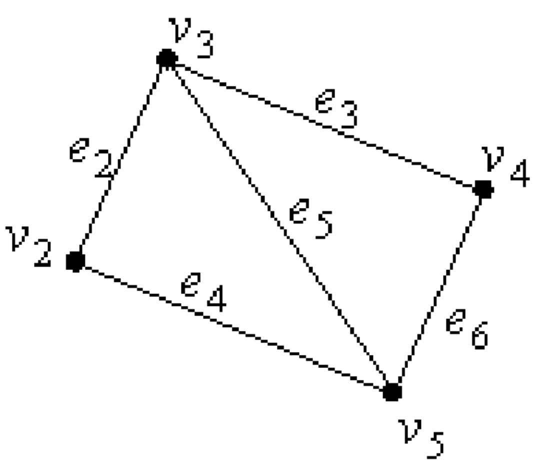
Зірчастий граф для вершини $v \in G$ складається з усіх ребер із початком або кінцем у вершині v і вершин, інцидентним цим ребрам (включаючи і вершину v).

Підграф. Приклад

Для графа:



Підграф:



Доповнення

Означення 6.14. *Доповнення* $\overline{H}$ *частини* H *визначається множиною всіх ребер графа G, які не належать H.*

Об'єднання $H_1 \cup H_2$ і переріз $H_1 \cap H_2$ частин H_1 і H_2 графу G визначаються природно:

$$V(H_1 \cup H_2) = V(H_1) \cup V(H_2);$$

$$E(H_1 \cup H_2) = E(H_1) \cup V(H_2);$$

$$V(H_1 \cap H_2) = V(H_1) \cap V(H_2);$$

$$E(H_1 \cap H_2) = E(H_1) \cap E(H_2).$$

Пряма сума

Дві частини H_1 та H_2 **не перерізаються по вершинах**, якщо вони не мають загальних вершин, а значить, і спільних ребер.

Об’єднання $H_1 \cup H_2$ частин, що не перерізаються по вершинах, називається **прямою сумою**.

Аналогічно визначається пряма сума будь-якого числа частин.

Частини H_1 й H_2 **не перерізаються по ребрах**, якщо $E(H_1) \cap E(H_2) = \varnothing$. Наприклад, для будь-якої частини H та її доповнення $\overline{H}$ сума $G = H \cup \overline{H}$ є прямою по ребрах.

7. Графи і бінарні відношення

Зв`язок графів з відношеннями

Між орієнтованими графами без кратних ребер з множиною вершин $V = \{v_1, \dots, v_n\}$ і бінарними відношеннями на множині V існує взаємно однозначна відповідність: відношенню R відповідає орієнтований граф $G(R)$, в якому ребро (v_i, v_j) існує тоді і тільки тоді, коли виконано співвідношення $v_i R v_j$.

Аналогічна взаємно однозначна відповідність є між симетричними бінарними відношеннями і неорієнтованими графами

Відповідність між операціями над відношеннями та графами

Розглянемо відповідність між операціями над відношеннями й операціями над графами. Кожне відношення R має заперечення $\overline{R}$, істинне тоді й тільки тоді, коли R – хибне. Наприклад, для відношення рівності $a=b$ запереченням є відношення нерівності $a \neq b$.

Відповідність між операціями над відношеннями та графами

Означення 6.15. *Граф* $G(\overline{R})$ *є* ***доповненням*** *графа G(R) відносно повного орієнтованого графа P(V) з множиною вершин V, на якій задано бінарне відношення R, що розглядається, і множиною дуг* $E(P(V)) = V \times V$.

Граф $G(R^{-1})$, де R^{-1} – відношення, обернене до *R*, різниться від графа *G(R)* тим, що напрямки всіх дуг замінено на зворотні.

Відповідність між операціями над відношеннями та графами

Відношення $R`$ **містить** відношення $R$ ($R` \supset R$), якщо вони визначені на одній і тій самій множині V та з $v_i R v_j$ виходить $v_i R` v_j$.

Відповідні графи $G(R)$ і $G(R`)$ мають одну й ту саму множину вершин $V$, а множина $E(R)$ ребер першого є підмножиною множини $E(R`)$ ребер другого. Таким чином, $G(R)$ є суграфом графа $G(R`)$, тобто $G(R`) \supset G(R)$.

Відповідність між операціями над відношеннями та графами

Для будь-яких бінарних відношень R_1 й R_2, заданих на одній і тій самій множині V, було визначено об'єднання $R_1 \cup R_2$ і переріз $R_1 \cap R_2$:

$$v_i(R_1 \cup R_2)v_j \Leftrightarrow v_i R_1 v_j \vee v_i R_2 v_j;$$

$$v_i(R_1 \cap R_2)v_j \Leftrightarrow v_i R_1 v_j \wedge v_i R_2 v_j.$$

Відповідні графи також є об'єднанням й перерізом:

$$G(R_1 \cup R_2) = G(R_1) \cup G(R_2);$$

$$G(R_1 \cap R_2) = G(R_1) \cap G(R_2).$$

Відповідність між операціями над відношеннями та графами

Деякі типи графів добре описуються на мові бінарних відношень.

Наприклад, нуль-граф $\varnothing(V)$, що не має ребер, відповідає нульовому (пустому) відношенню $v_i \varnothing v_j$, яке не містить жодної пари $(v_i, v_j) \in V \times V$; повному орієнтованому графу $D(V)$ відповідає універсальне (повне) відношення P.

Якщо R – рефлексивно, то $G(R)$ має петлі у всіх вершинах; якщо R – антирефлексивне, то $G(R)$ не має петель. Якщо R – транзитивне, то в графі $G(R)$ для кожної пари ребер (v_i, v_j) і (v_j, v_k) існує замикаюче ребро (v_i, v_k).

8. Маршрути, шляхи, ланцюги та цикли

Маршрут

Означення 6.16. *Нехай G – неорієнтований граф. **Маршрутом** M у графі G, що з'єднує вершину* v_1 *з вершиною* v_n*, називається така послідовність вершин і ребер, які чергуються,*

$$M = (v_1, e_1, v_2, e_2, \ldots, v_{n-1}, e_{n-1}, v_n), \qquad (6.1)$$

що починається у вершині v_1 *і закінчується в вершині* v_n*, і така, що кожні два сусідні ребра* e_{i-1} *і* e_i *мають спільну інциденту вершину* v_i.

Маршрут

Іншими словами, маршрутом, що з'єднує вершину v_1 з вершиною v_n, в неорієнтованому графі називається послідовність вершин і інцидентних їм ребер, яка починається в вершині v_1 і закінчується в вершині v_n.

Маршрут

Очевидно, що маршрут M можна задавати послідовністю його вершин $(v_1, v_2, \ldots, v_n)$ (в звичайному графі), а також послідовністю $(e_1, e_2, \ldots, e_n)$ ребер (для будь якого графа).

Одне і те саме ребро може зустрічатися в маршруті кілька разів. Вершина v_1 називається початком маршруту, v_n – кінець маршруту.

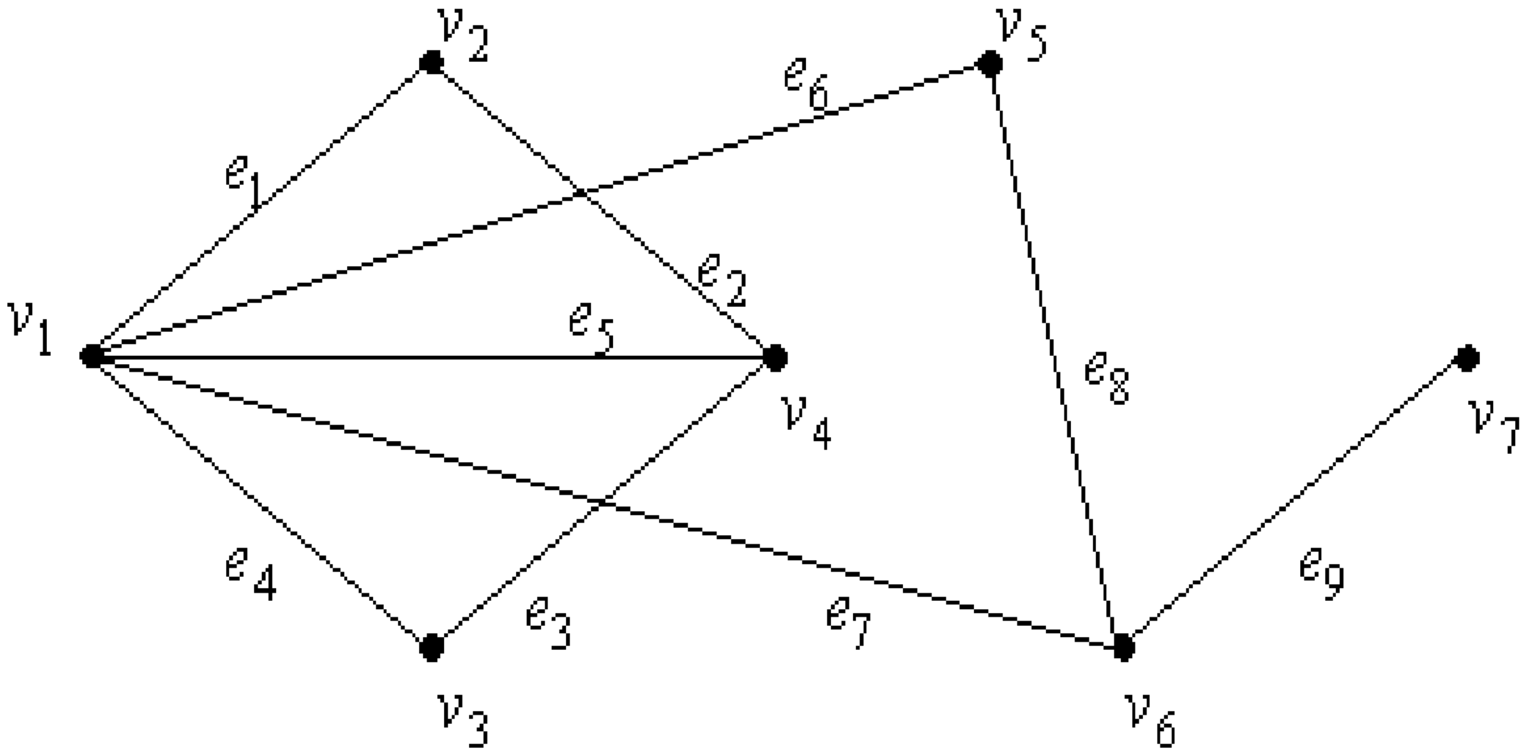
Внутрішні вершини

Означення 6.17. *Вершини, інцидентні ребрам маршруту, крім початкової і кінцевої, називаються* ***внутрішніми*** *або* ***проміжними***.

Оскільки різні ребра маршруту можуть бути інцидентними одній і тій самій вершині, початок або кінець маршруту може одночасно виявитися і внутрішньою вершиною

Внутрішні вершини. Приклад

Маршрут $(e_1, e_2, e_3, e_4, e_6, e_8)$



Довжина маршруту

Нехай маршрут М = (e_1, e_2, ... , e_n) має початок v_1 і кінець v_n.

Означення 6.18. *Число ребер маршруту називається його **довжиною**. Якщо* $v_1 = v_n$, *то маршрут називають **замкненим**.*

Ланцюги і цикли

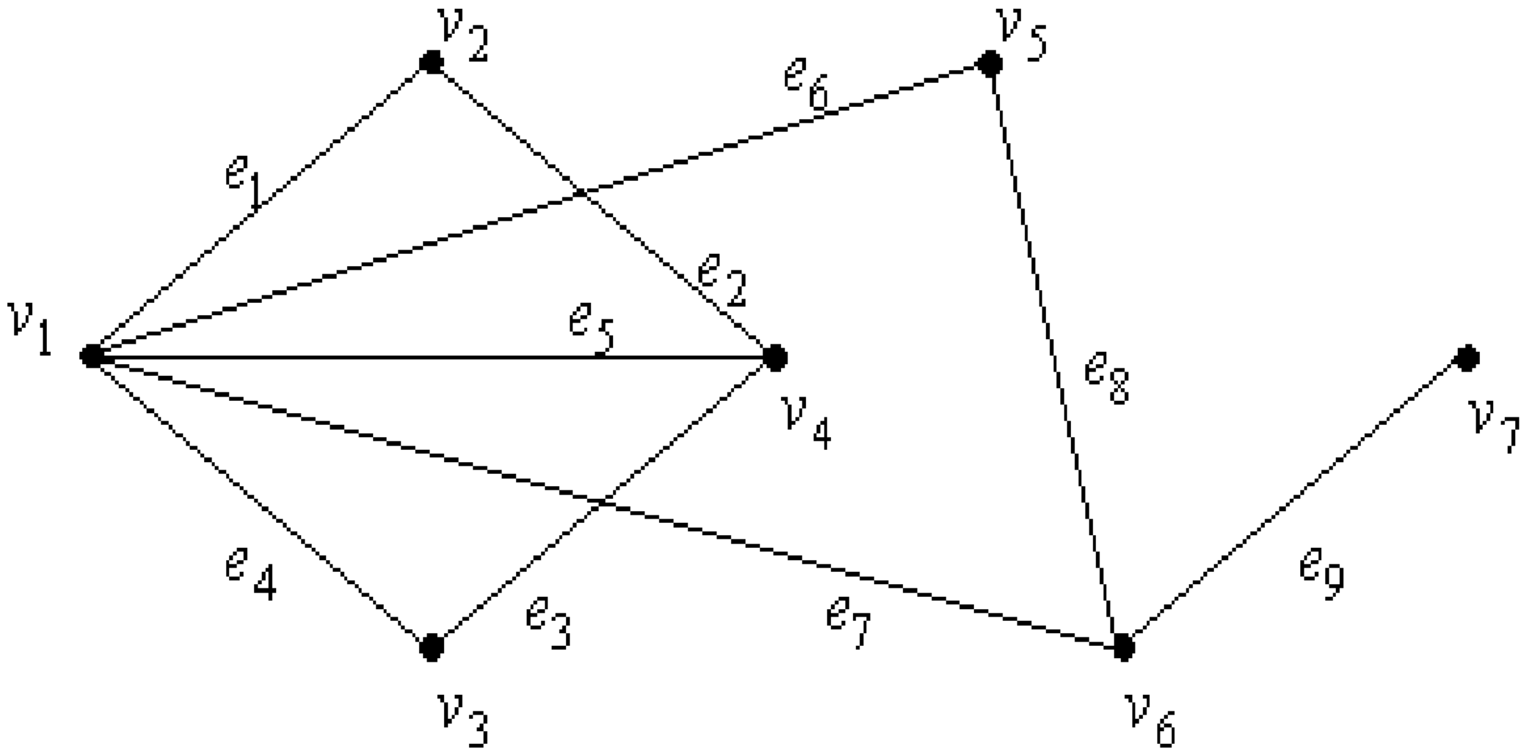
Означення 6.19. *Маршрут M називається* ***ланцюгом****, якщо кожне ребро зустрічається в ньому не більше ніж один раз, і* ***простим ланцюгом****, якщо будь-яка вершина (крім, можливо, початкової) зустрічається в ньому не більше, як один раз.*

Якщо ланцюг є замкненим, то його називають ***циклом****, а якщо простий ланцюг – замкнений, то це –* ***простий цикл****.*

Ланцюги і цикли. Приклад

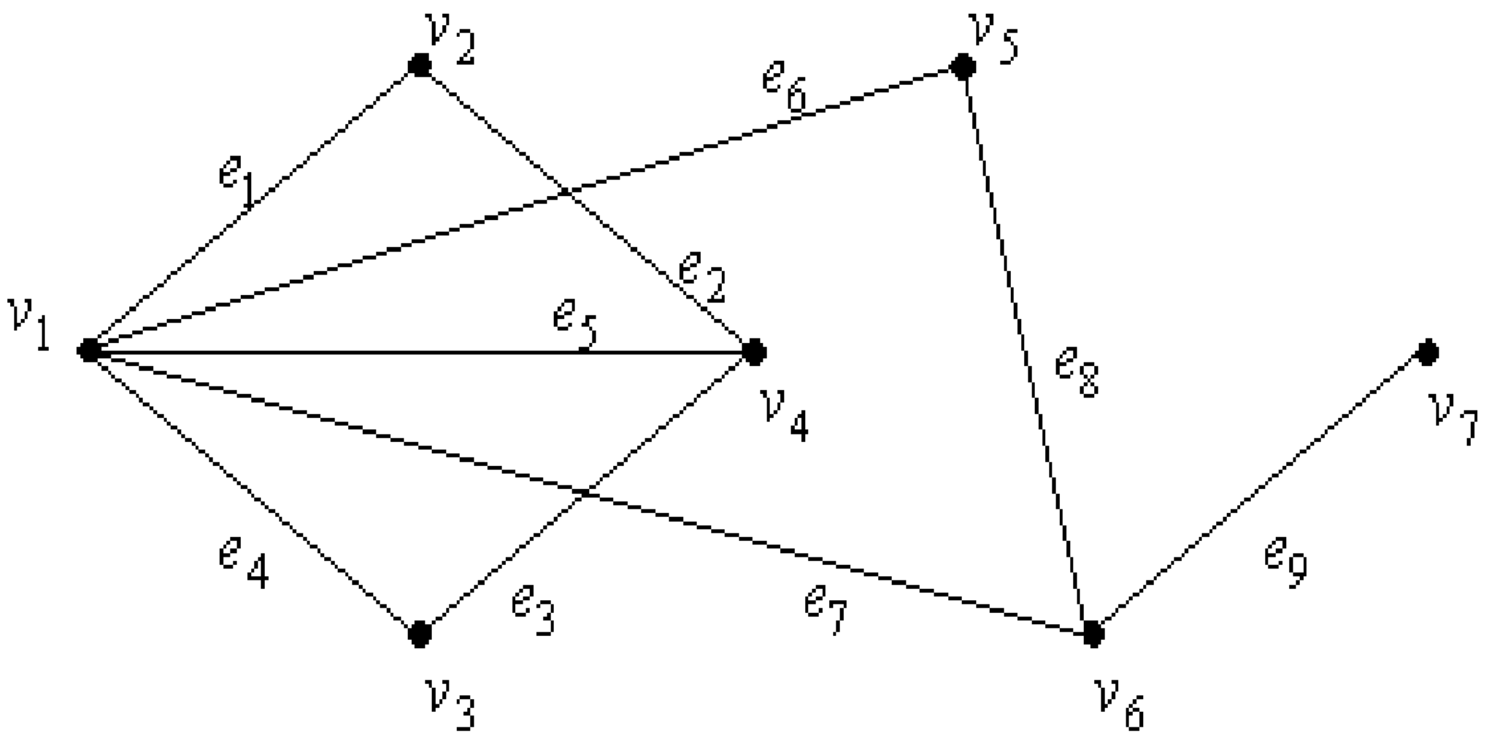
Маршрут $(e_1, e_2, e_3, e_4, e_6, e_8, e_7)$ є замкненим ланцюгом,

тобто циклом, але не є простим циклом, тому вершина v_1 зустрічається в ньому тричі.



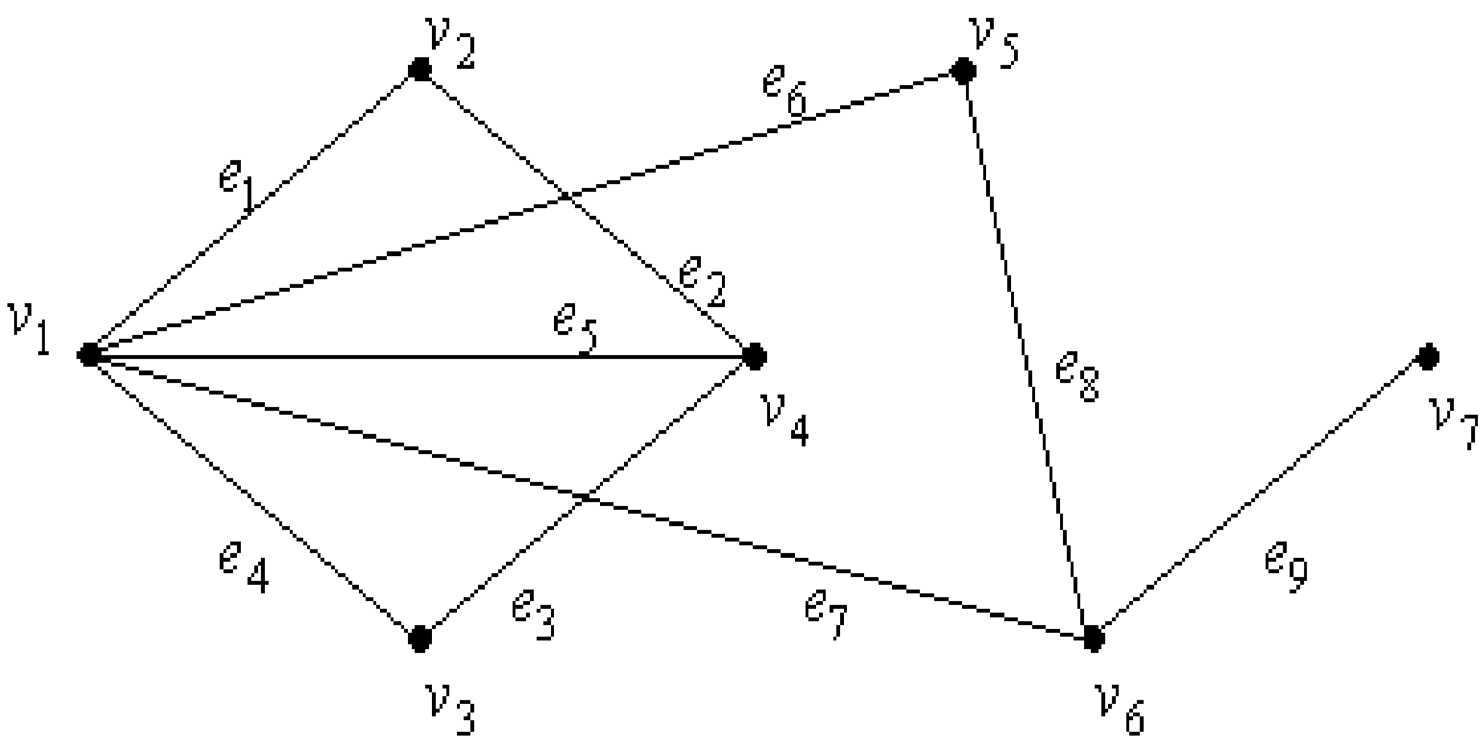
Ланцюги і цикли. Приклад

Простим циклом є цикл (e_1, e_2, e_3, e_4), причому послідовності (e_1, e_2, e_3, e_4), (e_2, e_3, e_4, e_1), (e_3, e_4, e_1, e_2) і (e_4, e_1, e_2, e_3) зображують один і той самий цикл.



Ланцюги і цикли. Приклад

Часто вважається, що можна міняти порядок ребер циклу на зворотний, тобто, наприклад, послідовність (e_4, e_3, e_2, e_1) зображує той же цикл



Маршрути в орієнтованих графах

Визначення маршруту легко перенести з графа на орієнтований граф.

Маршрут в останньому називатимемо **шляхом**. Відповідно можна перенести також визначення ланцюга, простого ланцюга та циклу.

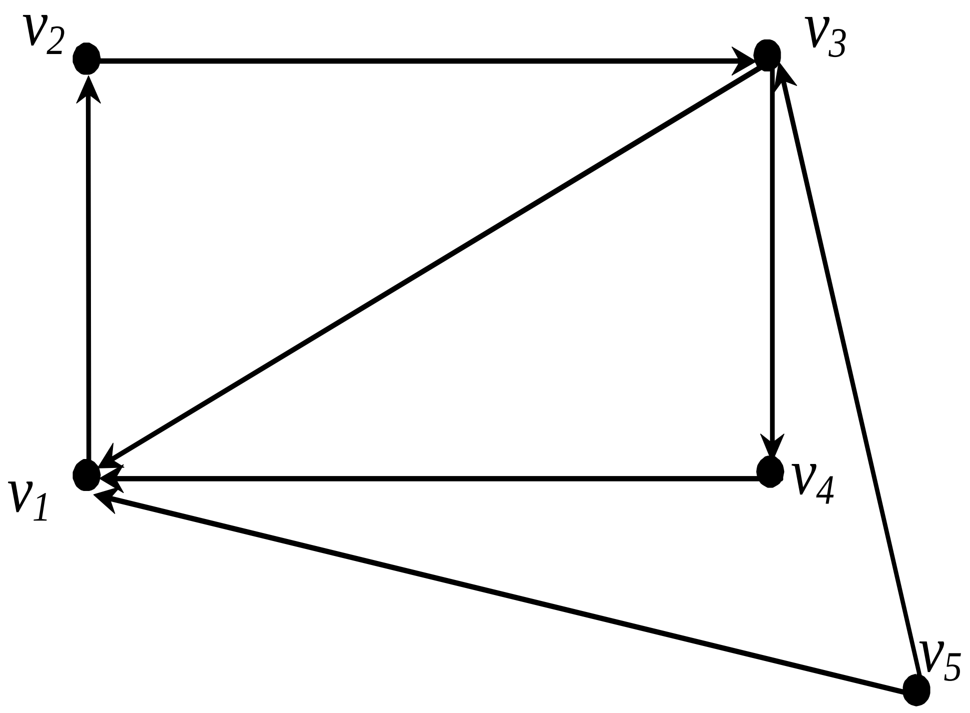
Простий цикл в орієнтованому графі ще називається **контуром**.

Визначення кількості шляхів

Твердження 6.1. k-й ступінь матриці суміжності графа G визначає наявність шляхів завдовжки k: елемент a_{ij} матриці $|G|^k$ дорівнює кількості шляхів довжини k, які мають початок у вершині v_i і кінець у вершині v_j (це справджується і для $i = j$, в цьому випадку шлях є циклом).

Визначення кількості шляхів. Приклад

Розглянемо граф G:



Визначення кількості шляхів. Приклад

Побудуємо матрицю суміжності графа G і знайдемо матрицю, яка є її другим степенем, по якій визначимо шляхи завдовжки 2:

$$
\begin{array}{c|ccccc}
 & v_1 & v_2 & v_3 & v_4 & v_5 \\
\hline
v_1 & 0 & 1 & 0 & 0 & 0 \\
v_2 & 0 & 0 & 1 & 0 & 0 \\
v_3 & 1 & 0 & 0 & 1 & 0 \\
v_4 & 1 & 0 & 0 & 0 & 0 \\
v_5 & 1 & 0 & 1 & 0 & 0
\end{array}
\times
\begin{array}{c|ccccc}
 & v_1 & v_2 & v_3 & v_4 & v_5 \\
\hline
v_1 & 0 & 1 & 0 & 0 & 0 \\
v_2 & 0 & 0 & 1 & 0 & 0 \\
v_3 & 1 & 0 & 0 & 1 & 0 \\
v_4 & 1 & 0 & 0 & 0 & 0 \\
v_5 & 1 & 0 & 1 & 0 & 0
\end{array}
=
\begin{array}{c|ccccc}
 & v_1 & v_2 & v_3 & v_4 & v_5 \\
\hline
v_1 & 0 & 0 & 1 & 0 & 0 \\
v_2 & 1 & 0 & 0 & 1 & 0 \\
v_3 & 1 & 1 & 0 & 0 & 0 \\
v_4 & 0 & 1 & 0 & 0 & 0 \\
v_5 & 1 & 1 & 0 & 1 & 0
\end{array}
$$

Визначення кількості шляхів. Приклад

Це, наприклад, шляхи з v_1 у v_3, з v_3 у v_2, з v_5 у v_1 і т.д. На головній діагоналі цієї матриці жодної одиниці.

Це свідчить про те, що в графі G немає жодного циклу завдовжки 2.

Визначення кількості шляхів. Приклад

Знайдемо третій ступінь матриці суміжності графа G, і за якою визначимо шляхи завдовжки 3.

$$
\begin{array}{c|ccccc}
 & v_1 & v_2 & v_3 & v_4 & v_5 \\
\hline
v_1 & 0 & 0 & 1 & 0 & 0 \\
v_2 & 1 & 0 & 0 & 1 & 0 \\
v_3 & 1 & 1 & 0 & 0 & 0 \\
v_4 & 0 & 1 & 0 & 0 & 0 \\
v_5 & 1 & 1 & 0 & 1 & 0
\end{array}
\times
\begin{array}{c|ccccc}
 & v_1 & v_2 & v_3 & v_4 & v_5 \\
\hline
v_1 & 0 & 1 & 0 & 0 & 0 \\
v_2 & 0 & 0 & 1 & 0 & 0 \\
v_3 & 1 & 0 & 0 & 1 & 0 \\
v_4 & 1 & 0 & 0 & 0 & 0 \\
v_5 & 1 & 0 & 1 & 0 & 0
\end{array}
=
\begin{array}{c|ccccc}
 & v_1 & v_2 & v_3 & v_4 & v_5 \\
\hline
v_1 & 1 & 0 & 0 & 1 & 0 \\
v_2 & 1 & 1 & 0 & 0 & 0 \\
v_3 & 0 & 1 & 1 & 0 & 0 \\
v_4 & 0 & 0 & 1 & 0 & 0 \\
v_5 & 1 & 1 & 1 & 0 & 0
\end{array}
$$

Визначення кількості шляхів. Приклад

Прикладами шляхів завдовжки 3 є шляхи з v_1 у v_4, з v_2 у v_1, з v_5 у v_1 (через v_4), з v_5 у v_2 (через v_1) і т.д. У матриці є одиниці на головній діагоналі. Ще елементи a_{11}, a_{22}, a_{33}, тобто є цикл завдовжки 3, який містить вершини v_1, v_2, v_3.

Визначення кількості шляхів. Приклад

Знайдемо четвертий ступінь матриці суміжності графа G, і визначимо шляхи завдовжки 4.

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

×

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

=

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

Визначення кількості шляхів. Приклад

Це шляхи, наприклад, з v_1 у v_2 (v_1, v_2, v_3, v_1, v_2), з v_5 у v_3 (v_5, v_3, v_1, v_2, v_3) і т.д.

Наявність одиниць на головній діагоналі свідчить про те, що є цикл завдовжки 4.

Цей цикл містить вершини v_1, v_2, v_3, v_4.

9. Зв'язність

Зв'язний граф

Означення 6.19. *Дві вершини v і w називаються* ***зв'язаними****, якщо існує маршрут з кінцями v та w. Граф називається* ***зв'язним****, якщо будь-яка пара його вершин є зв'язаною. Якщо граф не є зв'язним, то він називається* ***незв'язним***.

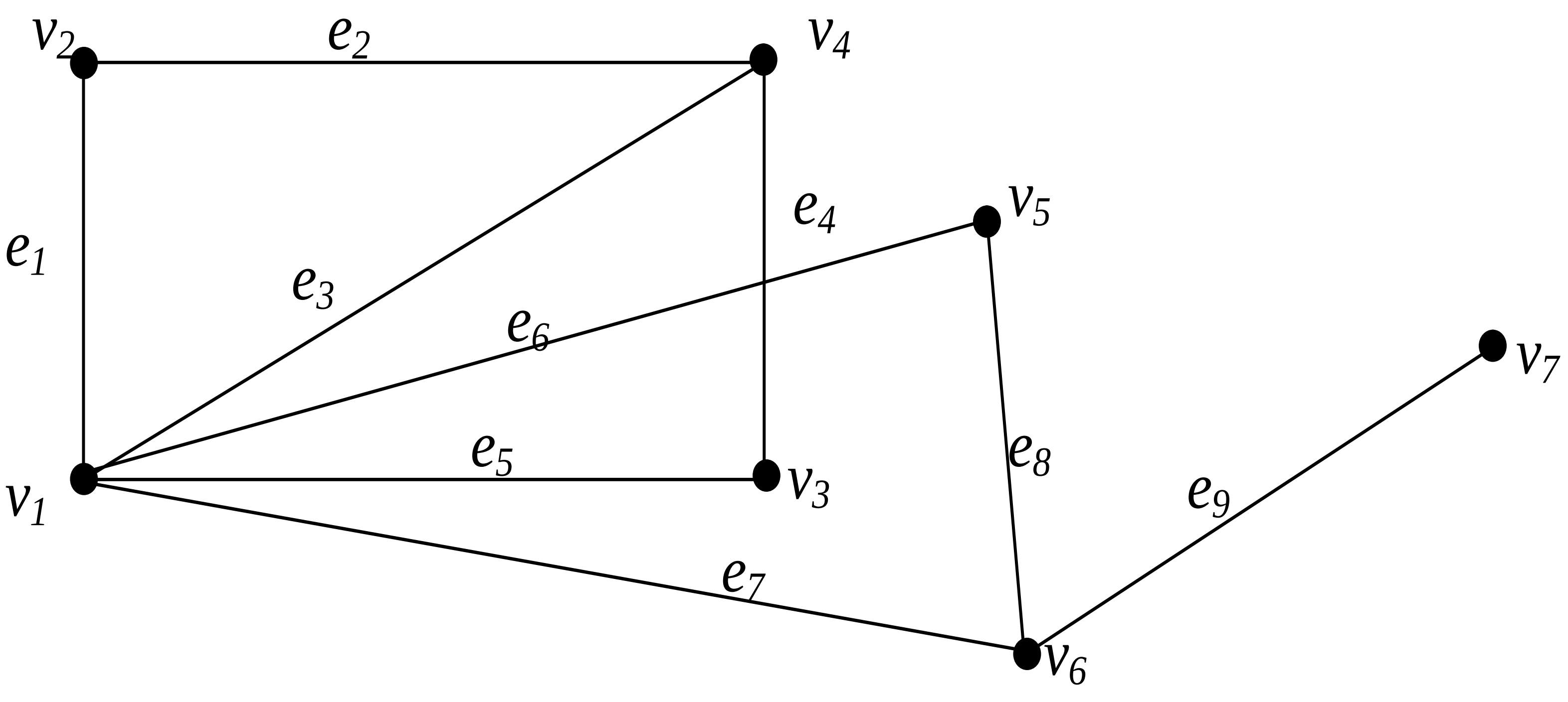
Означення 6.20. ***Зв'язністю*** *графа називається мінімальна кількість вершин, вилучення яких приводить до утворення незв'язного графа.*

Щільність графа

Означення 6.21. *Кількість вершин у максимальному повному підграфі графа G називається* ***щільністю*** $\alpha(G)$ *графа G. Кількість вершин у максимальному нульовому підграфі графа G називається* ***нещільністю*** $\varepsilon(G)$ *графа G.*

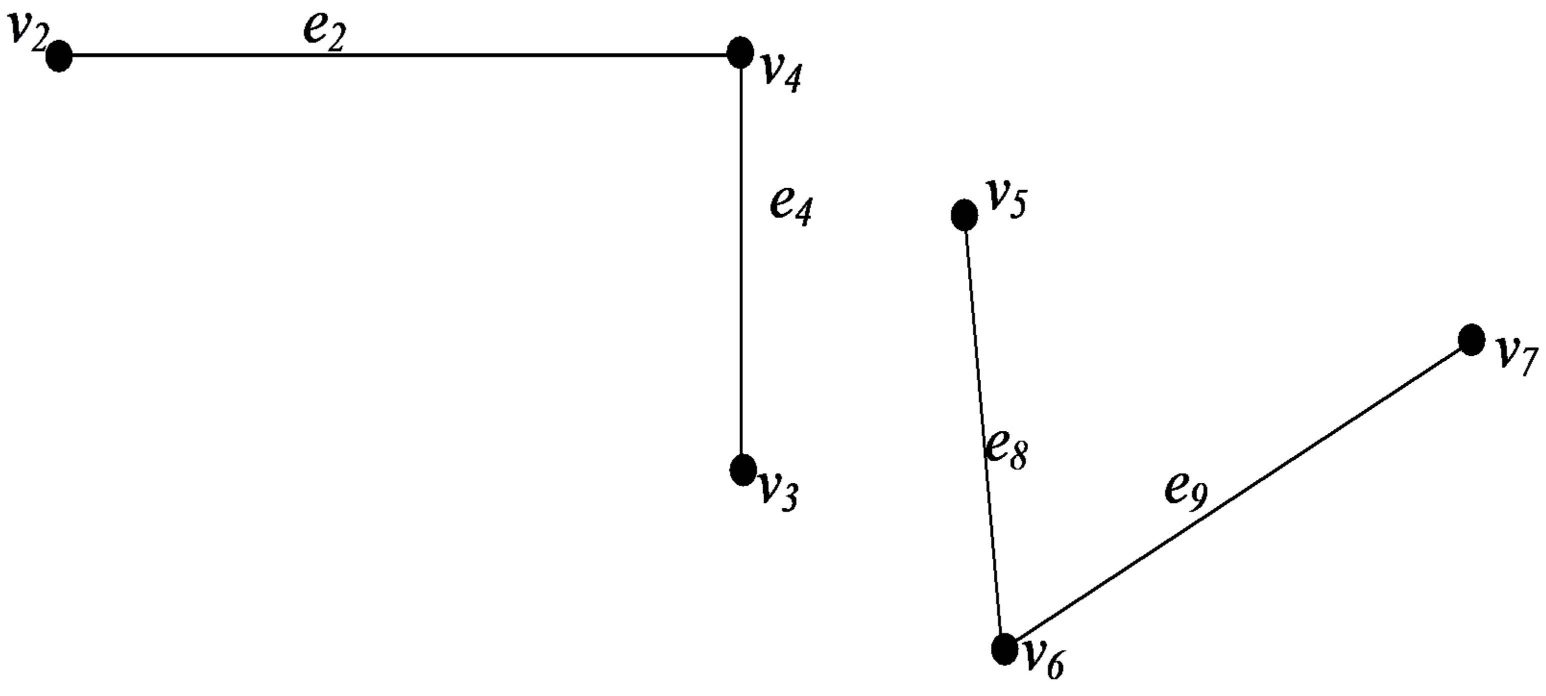
Приклад

Граф, наведений на рис., є зв'язним, тому що будь-яка пара його вершин зв'язана. Але яка його зв'язність? Тобто, якою є мінімальна кількість вершин його графа, влучення яких приводить до утворення незв'язного графа?



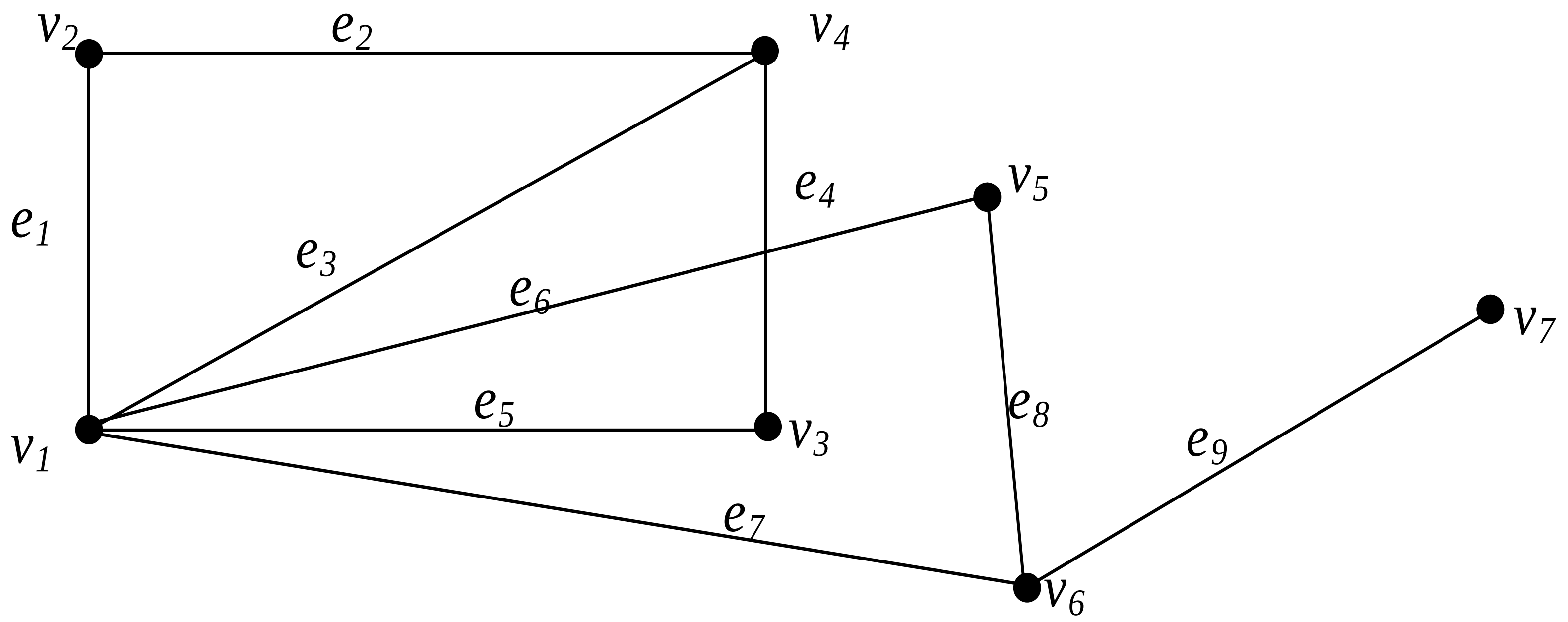
Приклад (продовження)

Очевидно, що зв’язність дорівнює одиниці, тому що вилучення однієї вершини v_1 приводить до утворення незв’язного графу.



Приклад (продовження)

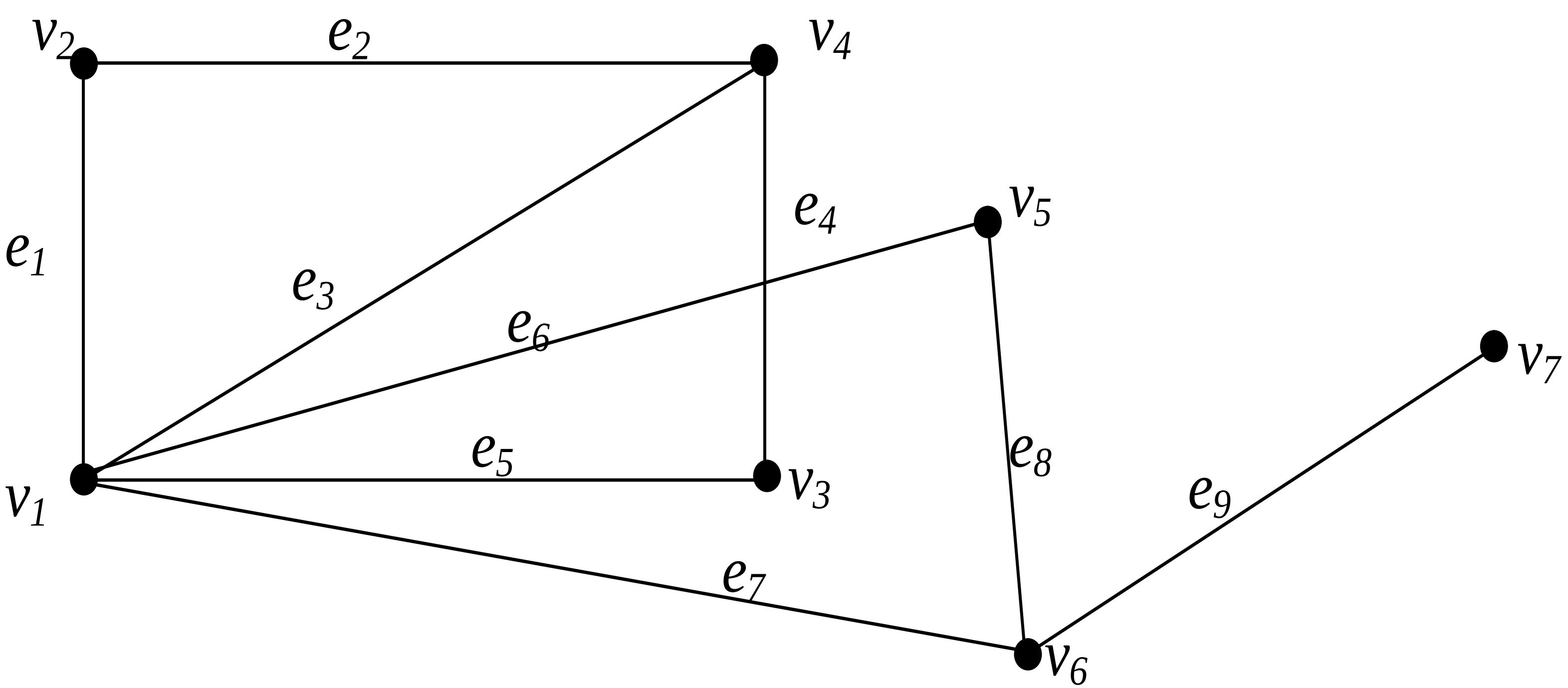
Визначимо щільність того ж самого графа. Для цього треба визначити максимальний повний підграф графа. В графі 7 вершин, але сам граф не повний, у нього немає повних підграфів на 6, 5 і 4 вершинах, але є декілька під графів на трьох вершинах. Отже, щільність графа дорівнює 3



Приклад (продовження)

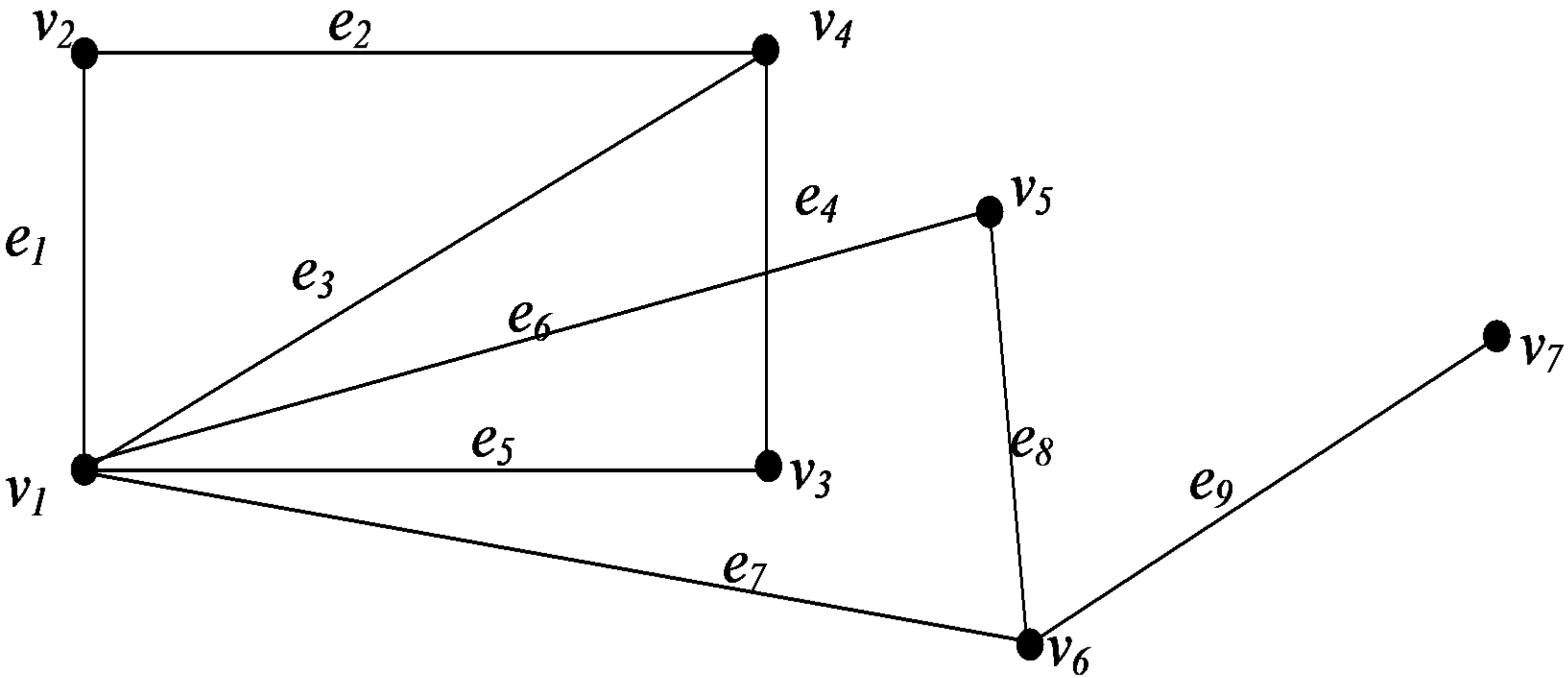
Визначимо нещільність того ж самого графа. Для цього треба визначити максимальний нульовий підграф.

Кожний підграф на одній вершині є нульовим.



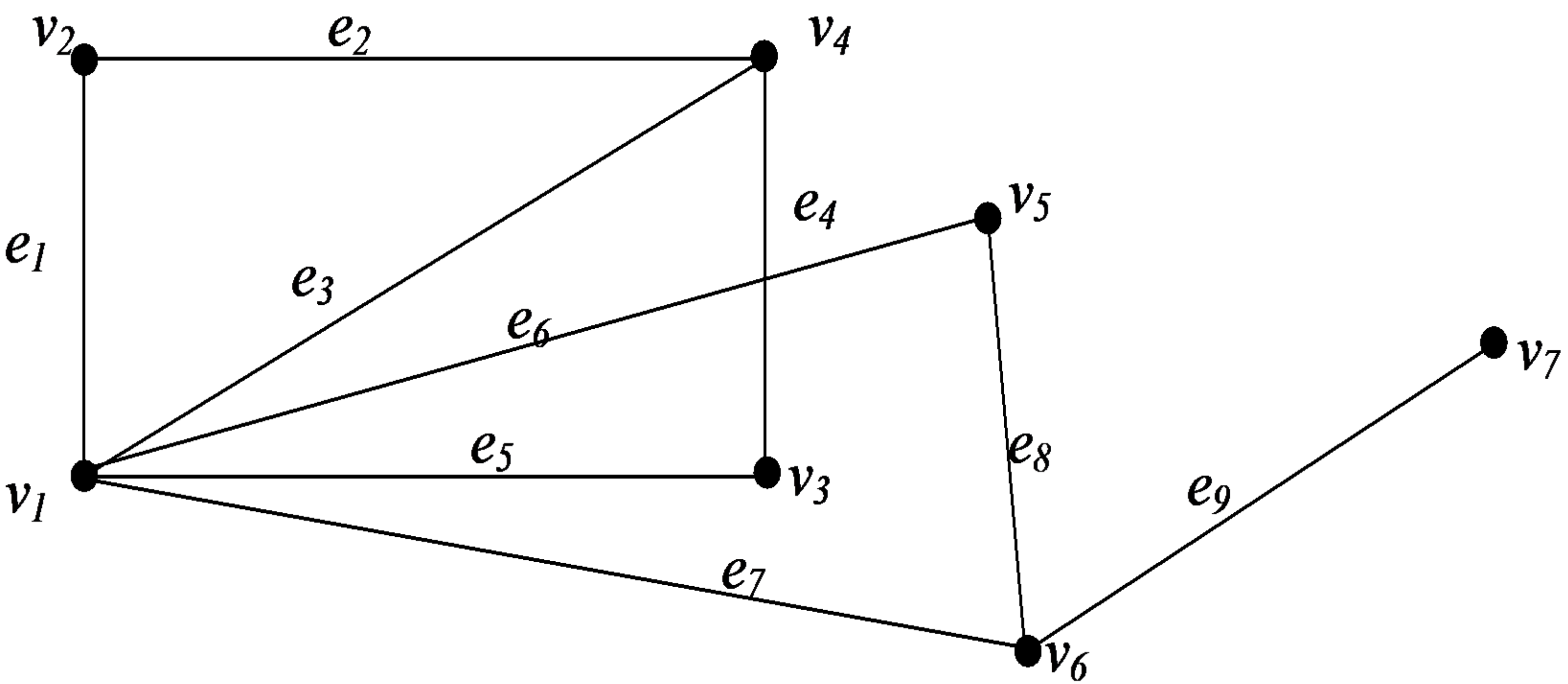
Приклад (продовження)

Є нульові підграфи на двох вершинах, наприклад (v_1, v_7), (v_2, v_6), (v_2, v_7), (v_3, v_5), (v_3, v_6), (v_3, v_7), (v_4, v_5), (v_4, v_6), (v_4, v_7), (v_5, v_7).



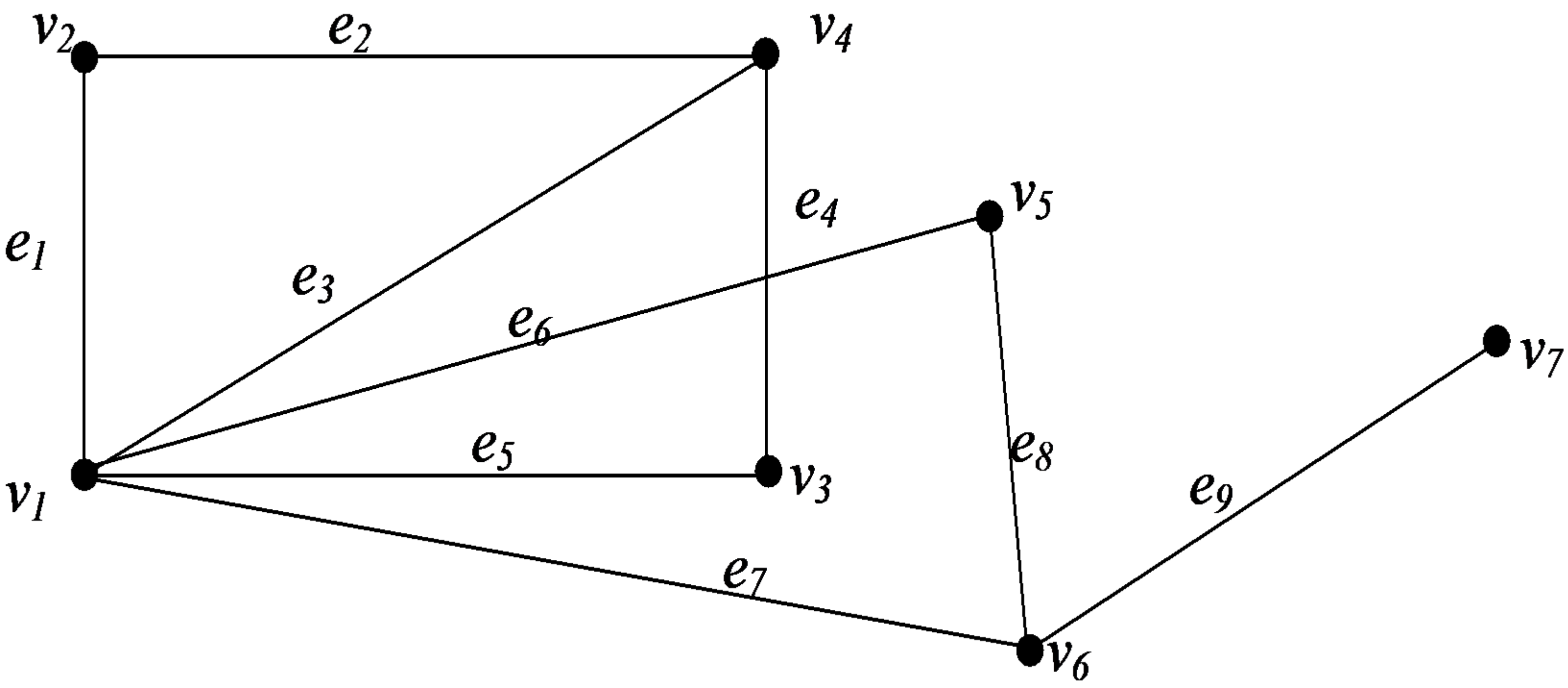
Приклад (продовження)

Є нульові підграфи на трьох вершинах: (v_2, v_3, v_5), (v_2, v_3, v_6), (v_2, v_3, v_7), (v_3, v_5, v_7), (v_4, v_5, v_7).



Приклад (продовження)

І є нульовий підграф на чотирьох вершинах (v_2, v_3, v_5, v_7). Це максимальний нульовий підграф. Тому нещільність графу з рис. дорівнює чотирьом.



Відстань між вершинами

Означення 6.22. *Довжина найменшого ланцюга між вершинами* v *і* w *звичайного графа* G *називається* ***відстанню*** $d(v,w)$ *між цими вершинами.*

Вона задовольняє аксіоми метрики:

$d(v,w) \ge 0$;

$d(v,w) = 0 \Leftrightarrow v = w$;

$d(v,w) = d(w,v)$;

$d(v,w) + d(w,u) \ge d(v,u)$.

Діаметр

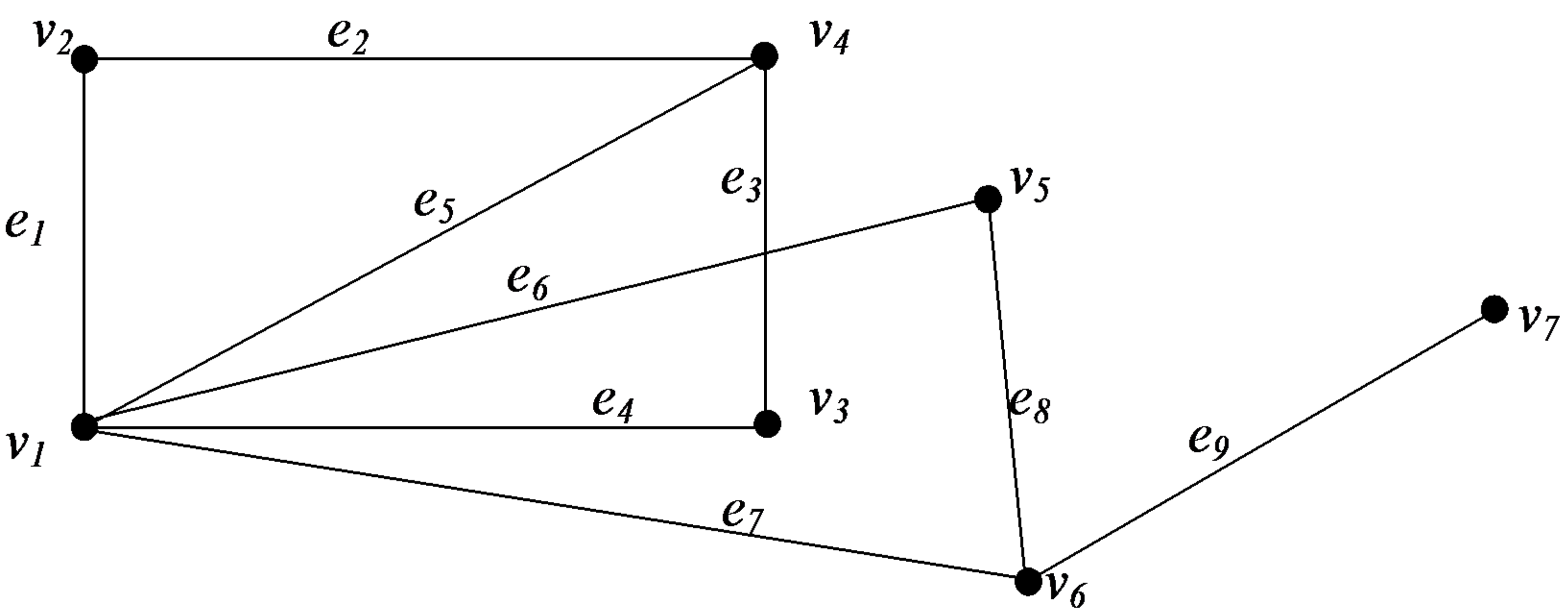
Означення 6.23. *Діаметром графа G називається*

величина $d(G)=\max\limits_{v,w\in V} d(v,w)$.

Іншими словами, діаметром графа G називається максимальна відстань між двома вершинами графа G.

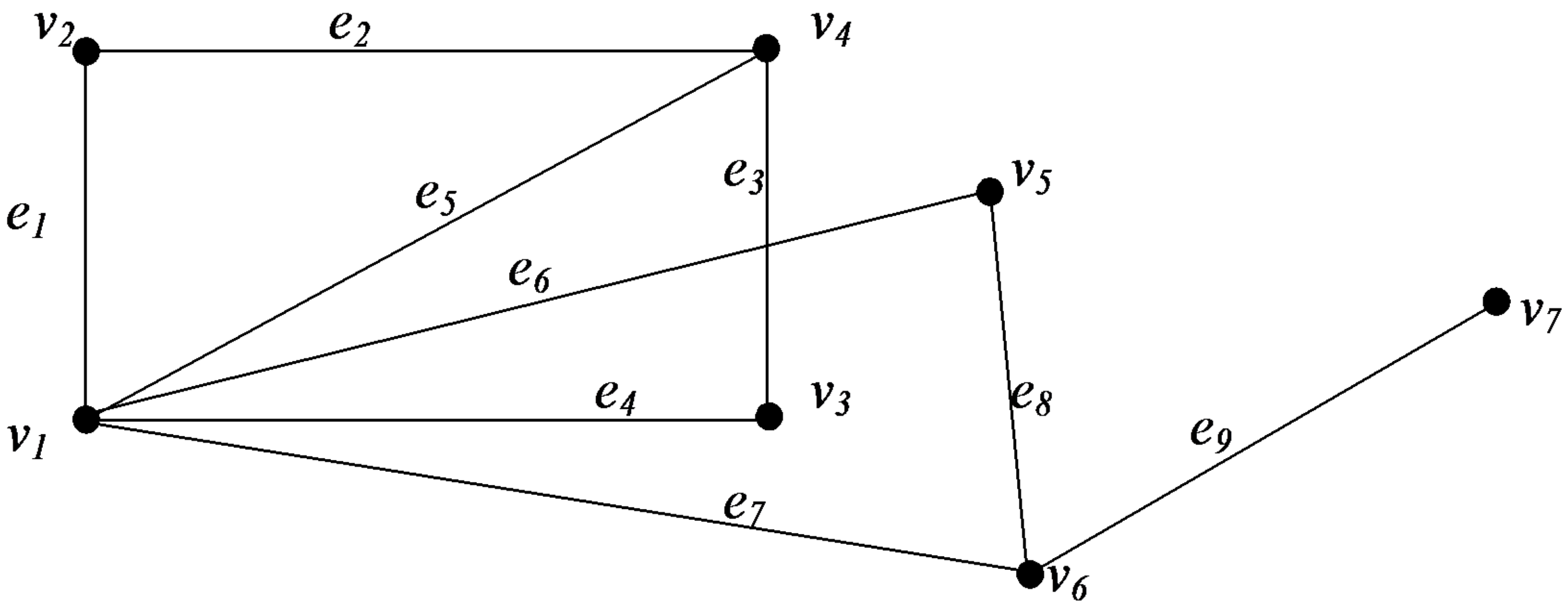
Приклад

Визначимо відстані між деякими вершинами графа з рис. $d(v_1,v_2)=1$, це ребро e_5, але є більші прості ланцюги, що зв'язують v_i та v_4, це ланцюги (e_1,e_2) і (e_4,e_3). $d(v_1,v_7)=2$, $d(v_3,v_5)=2$, $d(v_3,v_6)=2$, $d(v_2,v_6)=2$, $d(v_2,v_7)=3$.



Приклад (продовження)

В звичайному графі G (рис.) 7 вершин, тому всього існує C_7^2 пар вершин, для яких слід визначити відстані. Ми визначили декілька, але очевидно, що відстані більш за 3 в цьому графі немає. Тому діаметр $d(G)=3$



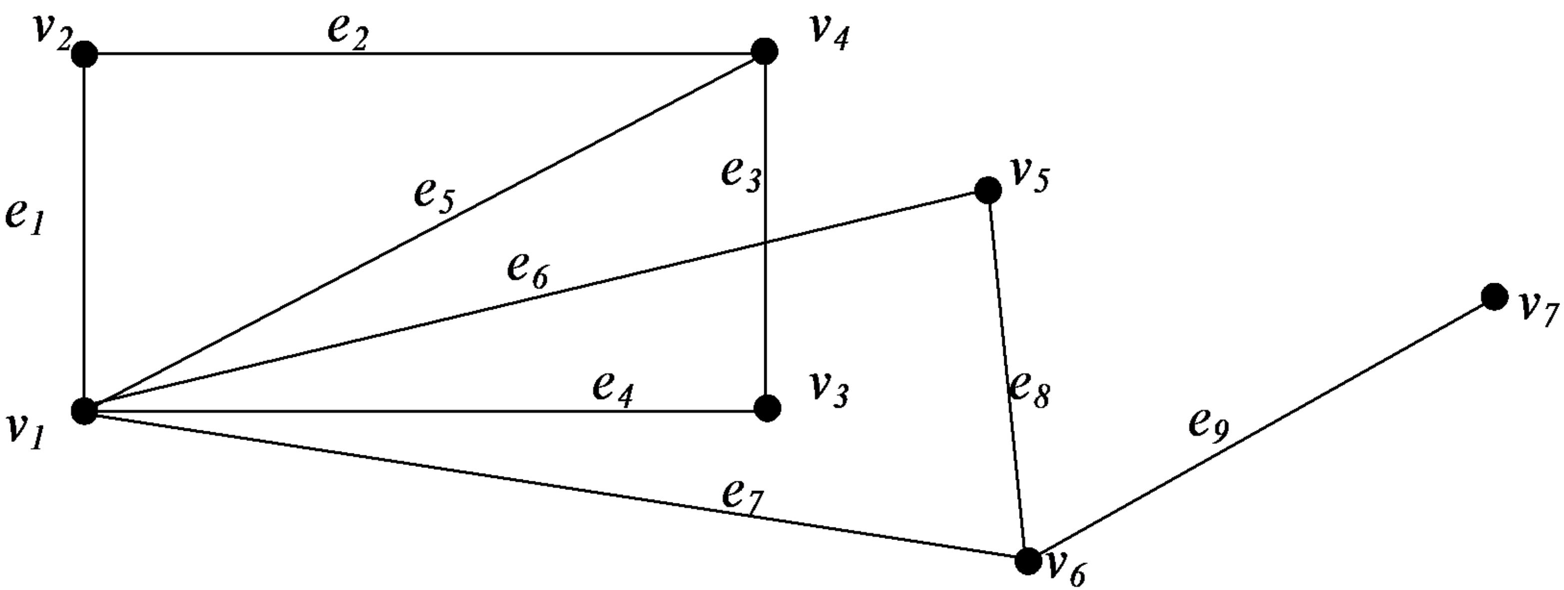
.

Центр графа

Означення 6.24. ***Центром*** *C_0 графа G називається вершина графа G, для якої максимальна з відстаней до інших вершин є мінімальною. Радіусом* r(G) *графа G називається максимальна відстань від центра C_0 графа G до його вершин.*

Приклад

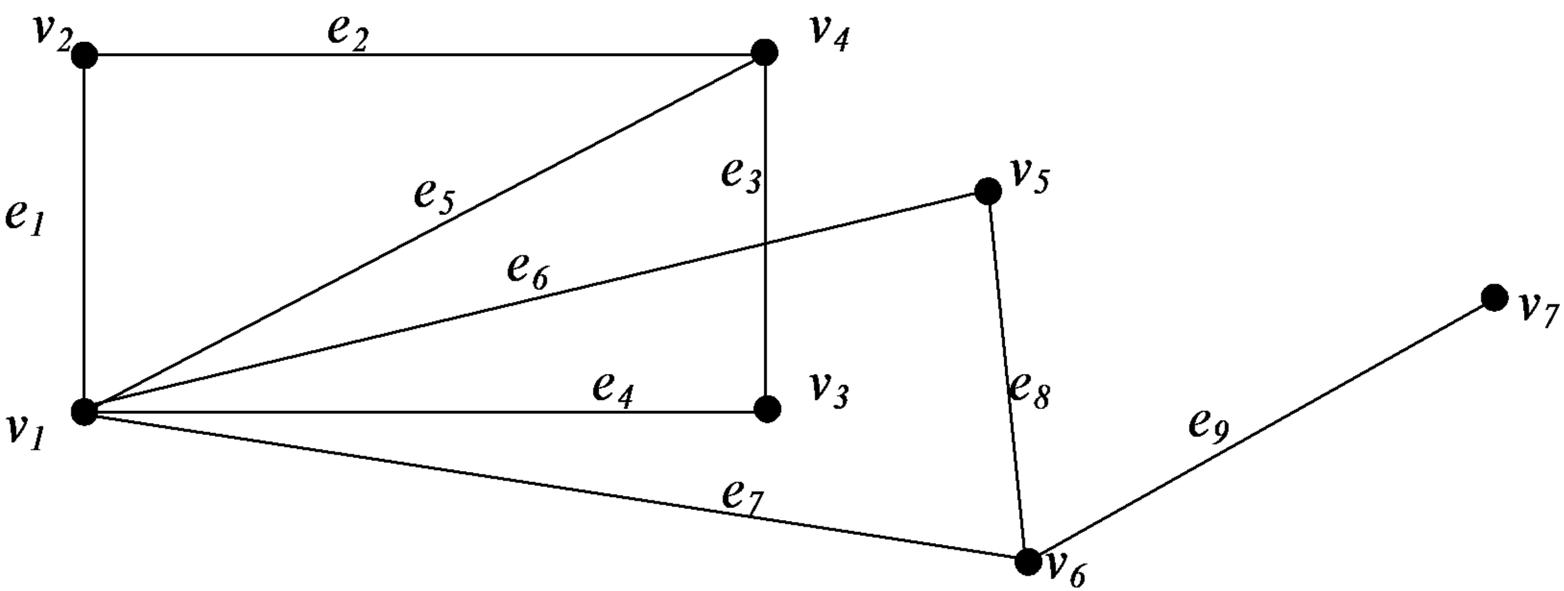
Для графа з рис. визначимо центр C_0 і радіус $r(G)$. Для цього визначимо відстані $d(v_1,v_2)=1$, $d(v_1,v_3)=1$, $d(v_1,v_4)=1$, $d(v_1,v_5)=1$, $d(v_1,v_6)=1$, $d(v_1,v_7)=2$. Максимальна відстань від v_1 до інших вершин дорівнює 2(і є відстанню від v_1до v_7).



Приклад(продовження)

$d(v_2, v_1) = 1$, $d(v_2, v_3) = 2$, $d(v_2, v_4) = 1$, $d(v_2, v_5) = 2$, $d(v_2, v_6) = 2$, $d(v_2, v_7) = 3$.

Максимальна відстань від v_2 до інших вершин дорівнює 3.



Приклад(продовження)

$d(v_3, v_1) = 1$, $d(v_3, v_2) = 2$, $d(v_3, v_4) = 1$, $d(v_3, v_5) = 2$, $d(v_3, v_6) = 2$, $d(v_3, v_7) = 3$.

Максимальна відстань від v_3 до інших вершин дорівнює 3.

$d(v_4, v_1) = 1$, $d(v_4, v_2) = 1$, $d(v_4, v_3) = 1$, $d(v_4, v_5) = 2$, $d(v_4, v_6) = 2$, $d(v_4, v_7) = 3$.

Максимальна відстань від v_4 до інших вершин дорівнює 3.

Приклад(продовження)

$d(v_5,v_1)=1$, $d(v_5,v_2)=2$, $d(v_5,v_3)=2$, $d(v_5,v_4)=2$, $d(v_5,v_6)=1$, $d(v_5,v_7)=2$.

Максимальна відстань від v_5 до інших вершин дорівнює 2.

$d(v_6,v_1)=1$, $d(v_6,v_2)=2$, $d(v_6,v_3)=2$, $d(v_6,v_4)=2$, $d(v_6,v_5)=1$, $d(v_6,v_7)=2$.

Максимальна відстань від v_6 до інших вершин дорівнює 2.

$d(v_7,v_1)=2$, $d(v_7,v_2)=3$, $d(v_7,v_3)=3$, $d(v_7,v_4)=3$, $d(v_7,v_5)=2$, $d(v_7,v_6)=1$.

Максимальна відстань від v_7 до інших вершин дорівнює 3.

Приклад(продовження)

Таким чином мінімальною із максимальних з вершин є відстань 2, і на звання центру претендують 3 вершини v_1, v_5 і v_6.

Тому радіус дорівнює максимальної відстані від цих вершин до інших вершин, а це

$$r(G) = 2$$

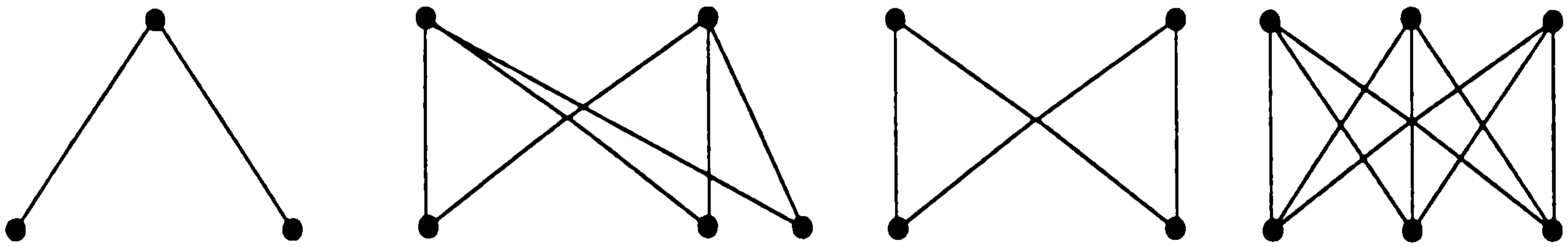
Дводольний граф

Означення 6.25. ***Дводольний граф*** $G_{m,n} = (V, E)$ *– це граф, множину вершин якого можна розбити на дві підмножини V_1 і V_2 ($V = V_1 \cup V_2, V_1 \cap V_2 = \varnothing, |V_1| = m$, $|V_2| = n$) таким чином, що кожне ребро графа з'єднує вершини з різних підмножин.*

Дводольний граф називається ***повним*** *$K_{m,n}$, якщо для кожної пари вершин $v \in V_1$ та $w \in V_2$ існує ребро $(v, w) \in E$*

Приклад

Графи $K_{1,2}$, $K_{2,3}$, $K_{2,2}$, $K_{3,3}$



Сильнозв'язний та однобічно зв'язний граф

Означення 6.26. *Орієнтований граф називається* ***сильнозв'язним (зв'язним)****, якщо для будь-яких двох його вершин* v *і* w *існує шлях в обох напрямках.*

Орієнтований граф називається ***зв'язним (слабозв'язним)****, якщо відповідний йому неорієнтований граф є зв'язним.*

Означення 6.27. *Орієнтований граф називається* ***однобічно зв'язним****, якщо для будь-яких двох його вершин* v_i *та* v_j *існує шлях хоча би в одному напрямку.*

Компонента зв'язності

Означення 6.28. ***Компонентою зв'язності*** *графа G називається його зв'язний підграф, який не є підграфом жодного іншого зв'язного підграфа графа G.*

Приклад

На рис. зображено граф із чотирма компонентами зв’язності.

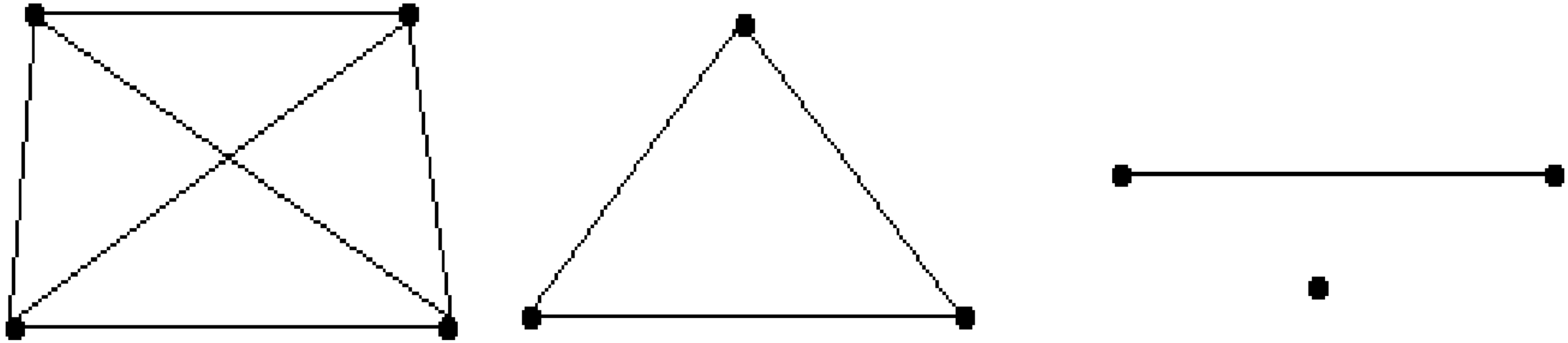


Рис.

Приклад

На рис. наведений орієнтований граф, що має три компоненти зв'язності.

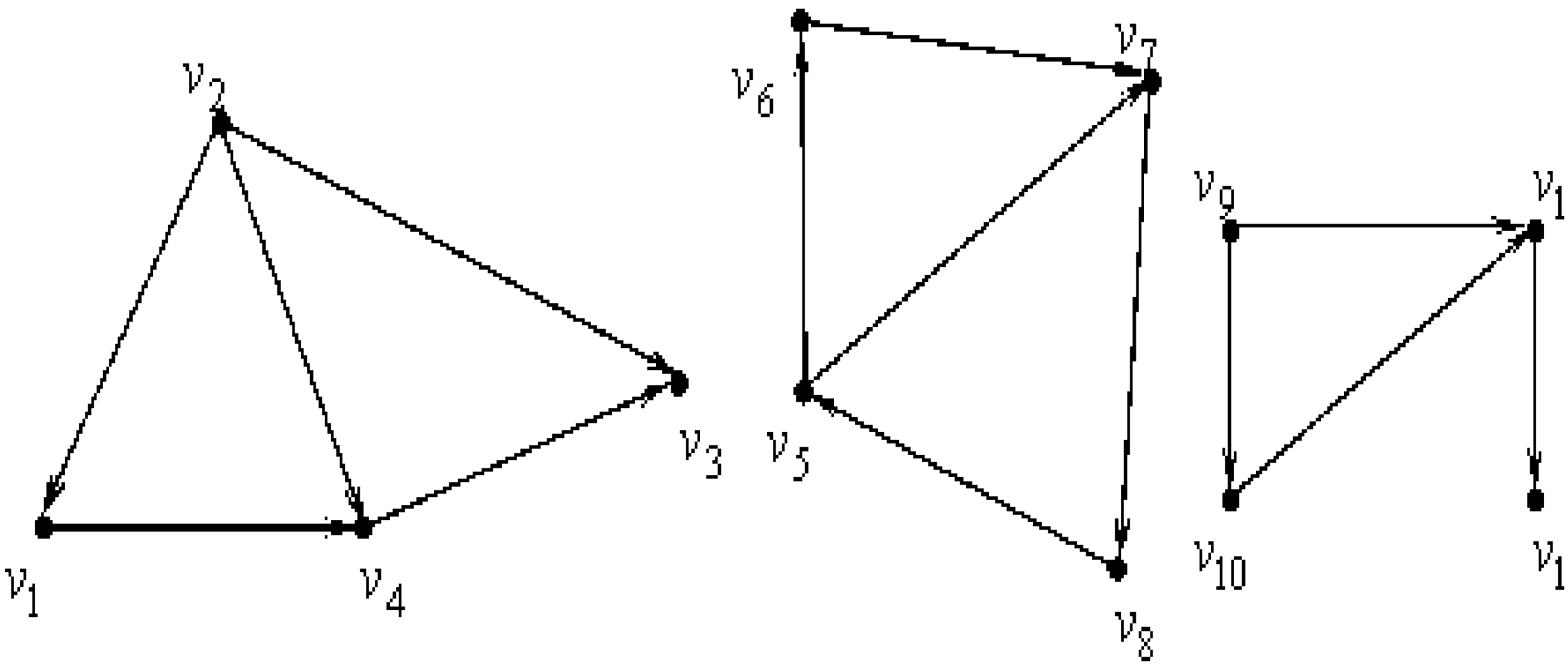


Рис.

Відношення зв'язності

Виходячи з означення зв’язності двох вершин, можна говорити про бінарне відношення зв’язності (позначимо його σ), задане на множині $V(G)$.

Відношення зв'язності неорієнтованого графу

Твердження 6.1. *Якщо припустити, що вершина зв'язана сама з собою, тобто існує маршрут із* v *у* v*, то бінарне відношення зв'язності* σ *має наступні властивості:*

1. рефлексивність, тобто $v \sigma v \ \forall v \in V$;

2. симетричність, тобто $v_1 \sigma v_2 \Leftrightarrow v_2 \sigma v_1 \ \forall v_1, v_2 \in V$

іншими словами, якщо існує маршрут із вершини v_1 *у вершину* v_2*, то існує маршрут і в зворотному напрямку, тобто з вершини* v_2 *у вершину* v_1;

Відношення зв'язності неорієнтованого графу

3. транзитивність, тобто

$v_1 \sigma v_2 \wedge v_2 \sigma v_1 \Rightarrow v_1 \sigma v_3 \; \forall v_1, v_2, v_3 \in V$, *іншими словами, якщо існує маршрут із вершини* v_1 *у вершину* v_2, *а з вершини* v_2 *у вершину* v_3, *то існує маршрут із вершини* v_1 *у вершину* v_3.

Таким чином відношення зв'язності є відношенням еквівалентності, яке визначене на множині V.

Відношення зв'язності неорієнтованого графу

Очевидно, що:

1. $v_1 \sigma v_2$ тоді й тільки тоді, коли вершини v_1 і v_2 належать одній компоненті зв'язності графа *G*;

2. для будь-якого класу еквівалентності $V_1 \in \{[v]_\sigma\}$ граф G_1, породжений множиною V_1, є компонентою зв'язності графа *G*.

Вилучення вершин

Під операцією вилучення вершини з графа розумітимемо операцію вилучення деякої вершини разом із ребрами, які інцидентні їй.

Означення 6.29. *Вершина графа, вилучення якої збільшує кількість компонент зв'язності, називається* ***відокремлю вальною*** *врешиною або точкою зчленювання*

(по. рус. разрезающая вершина, точка сочленения)

Вилучення вершин. Приклад

Для графа, зображеного на рис., точками зчленування (відокремлювальними вершинами) будуть вершини v_3, v_4, v_7, v_{10}.

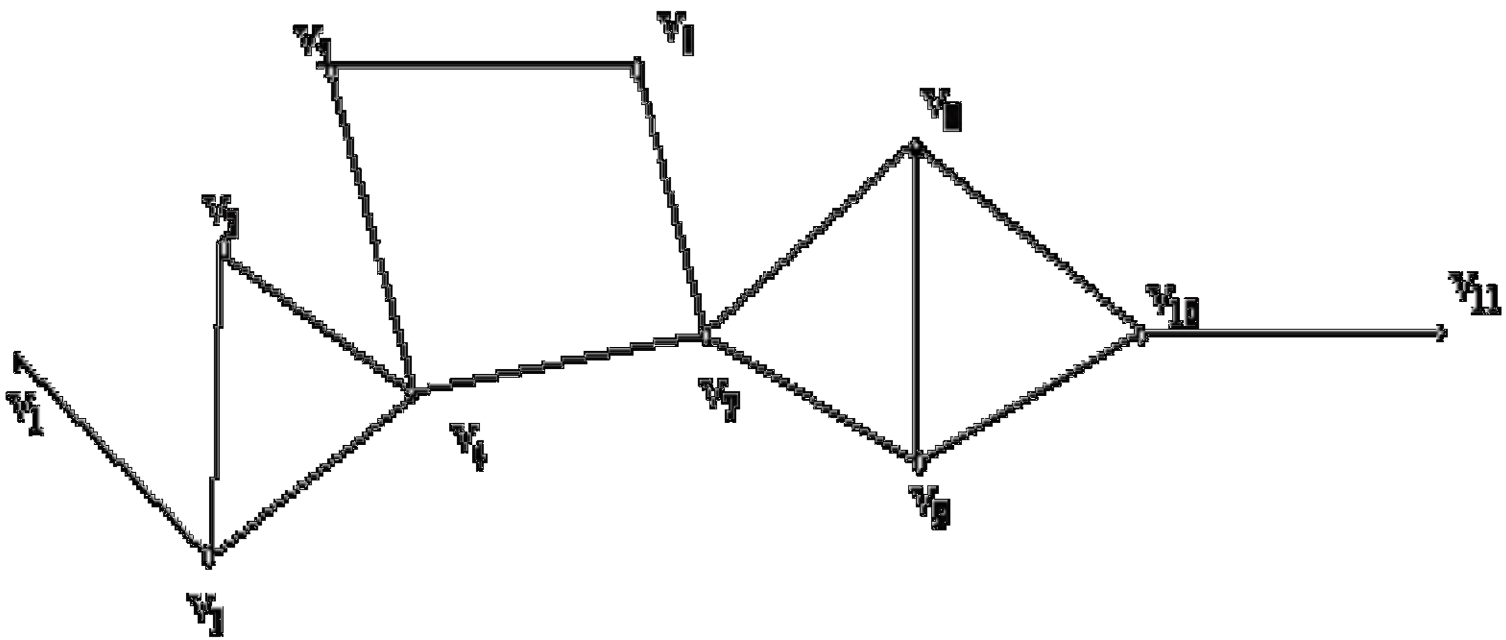


Рис.

10. Дерева

Дерева

Означення 6.30. Звичайний граф G називається **деревом**, якщо він є зв'язним і не має циклів, а граф G, всі компоненти зв’язності якого є деревами - **лісом**.

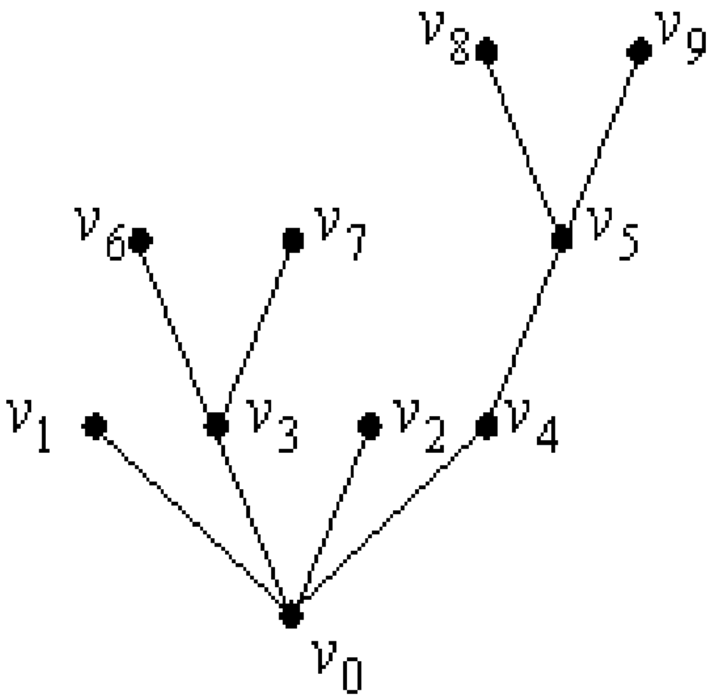
Означення 6.31. *Вершина v графа G називається* ***кінцевою*** *(такою що висить), якщо її локальний ступінь дорівнює 1.*

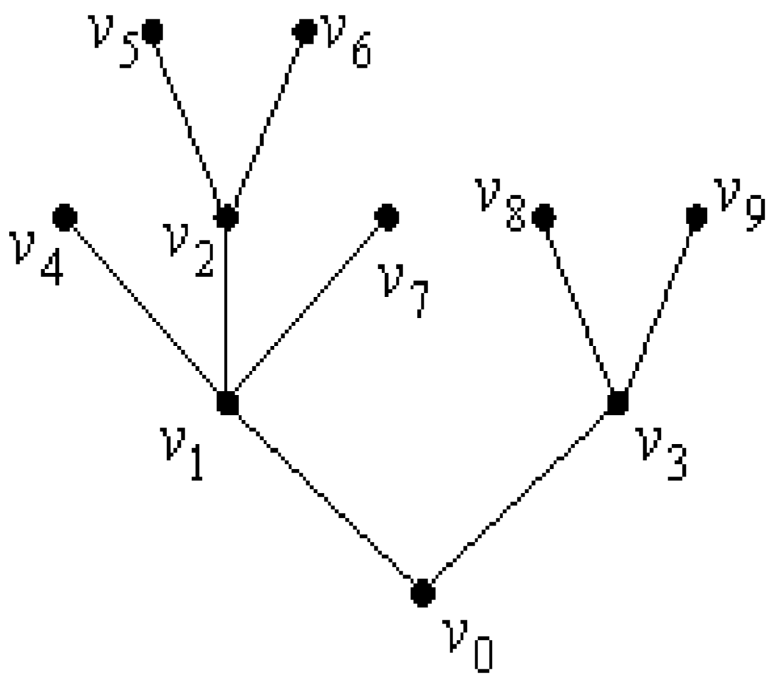
Ребро, яке інцідентне кінцевій вершини, називається ***кінцевим***.

Очевидно, якщо дерево має більш за одну вершину, то воно має хоча б одне кінцеве ребро, якщо дерево має більш двох вершин, то серед них є некінцеві.

Приклад

Графи, зображені на рис., є деревами.





Еквівалентні твердження

1) граф G - дерево;

2) граф G є зв'язним і не має простих циклів;

3) граф G є зв'язним, і $n(G) = m(G) + 1$;

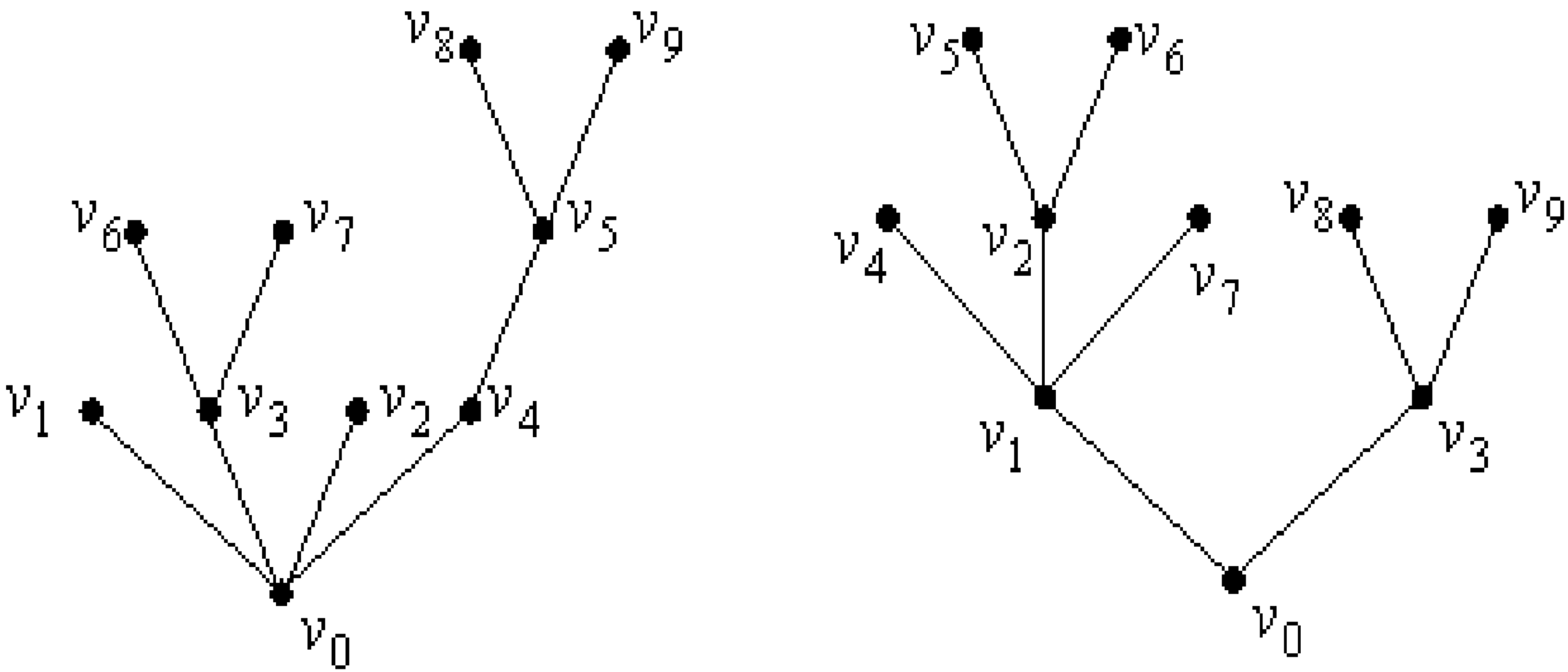
4) для будь-яких двох різних вершини графа G існує єдиний (і притому простий) ланцюг;

5) граф G не містить циклів, але, додаючи до нього будь-яке нове ребро, дістаємо рівно один цикл.

Корінь дерева

В дереві G вибирають вершину v_0, яку називають **коренем** дерева G.

На рис. наведено приклади двох ізоморфних графів – дерев, які відрізняються за вибором кореня.



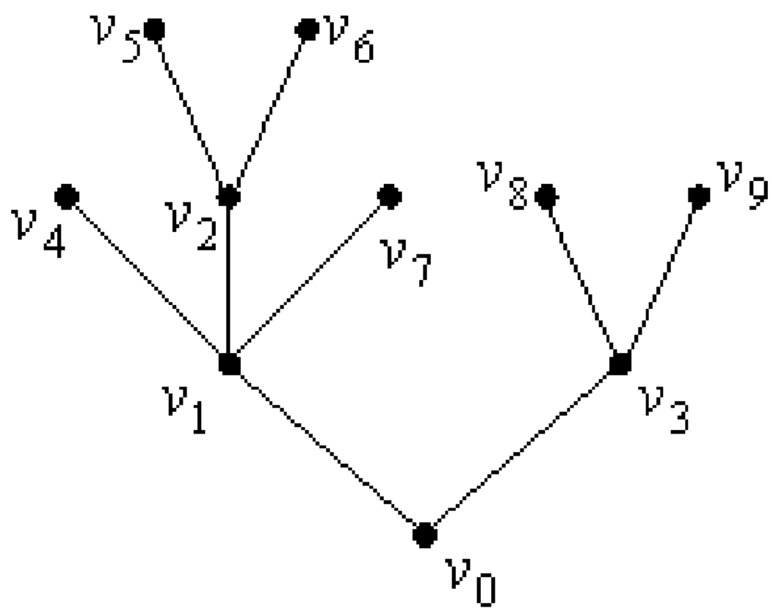
Гілки та вершини типу n

Нехай v – деяка вершина дерева G з коренем v_0, V_1 – множина вершин, які зв'язані з коренем v_0 ланцюгами, що містять вершину v. Ця множина породжує підграф $G_1(V)$, який називається **гілкою** вершини v у дереві з коренем v_0.

Означення 6.32. *Нехай є дерево G.* ***Вершинами типа 1*** *називаються кінцеві вершини. Якщо з дерева G вилучити всі кінцеві вершини разом з інцідентними ребрами, то в частині графа, що залишилася, є кінцеві вершини. Вони називаються* ***вершинами типа 2*** *дерева G. Аналогічно визначаються* ***вершини типа 3, 4, …***

Приклад

В наведеному на рис. графі гілкою вершини v_2 дерева з коренем v_0 є частина графа –дерево на вершинах $(v_1, v_4, v_5, v_6, v_7)$. В початковому графі вершини $v_4, v_5, v_6, v_7, v_8, v_9$ – типу 1. v_2, v_3 – типу 2. v_1, v_0 – типу 3.



Цикломатичне число

Означення 6.33. *Нехай $G = (V, E)$ – звичайний граф. Його* ***цикломатичним числом*** *називається число*

$$\gamma(G) = n_\sigma + m - n,$$

де n_σ – кількість зв'язних компонент графа; m – кількість його ребер, а n – кількість вершин.

Цикломатичне число звичайного графа є невід'ємним.

Цикломатичне число дерева дорівнює нулю, цикломатичне число лісу – сумі цикломатичних чисел своїх зв'язних компонент-дерев, тобто також дорівнює нулю.

11. Кістякове дерево зв'язного графу

Кістякове дерево

Означення 6.34. ***Кістяковим деревом*** *зв'язного графа G називається будь-яка його частина, що містить усі вершини графа G і є деревом.*

Нехай G - зв'язний граф. Тоді кістякове дерево графа G (якщо воно існує) має містити $n(G)-1$ ребер, і є результатом вилучення з G рівно $m(G)-(n(G)-1)=m(G)-n(G)+1$ ребер.

Покажемо існування кістякового дерева для довільного зв'язного псевдографа $G=(V,X)$, описавши алгоритм його вибору.

Алгоритм знаходження кістякового дерева

Крок 1. Вважаємо $i=1$. Вибираємо в G довільну вершину v_1, що утворює частину G_1 псевдографа G, причому G_1 є деревом.

Крок 2. Якщо $i=n$, де $n=n(G)$, то задачу розв'язано, і G_i - шукане кістякове дерево псевдографа G. В іншому випадку переходимо до кроку 3.

Алгоритм знаходження кістякового дерева(продовження)

Крок 3. Нехай уже побудовано дерево G_i, що є частиною псевдографа G і містить деякі вершини $v_1,\ldots,v_i$, де $1 \le i \le n-1$. Будуємо граф G_{i+1}, додаючи до графа G_i нову вершину $v_{i+1} \in V$, суміжну в G з деякою вершиною v_j графа G_i, та нове ребро, інцідентне вершинам v_i і v_{i+1} (внаслідок зв'язаності G та тієї обставини, що $i < n$, вершина v_{i+1} обов'язково знайдеться). Очевидно, граф G_{i+1} також є деревом. Прирівнюємо $i := i+1$ і переходимо до кроку 2.

Алгоритм знаходження кістякового дерева зваженого графу

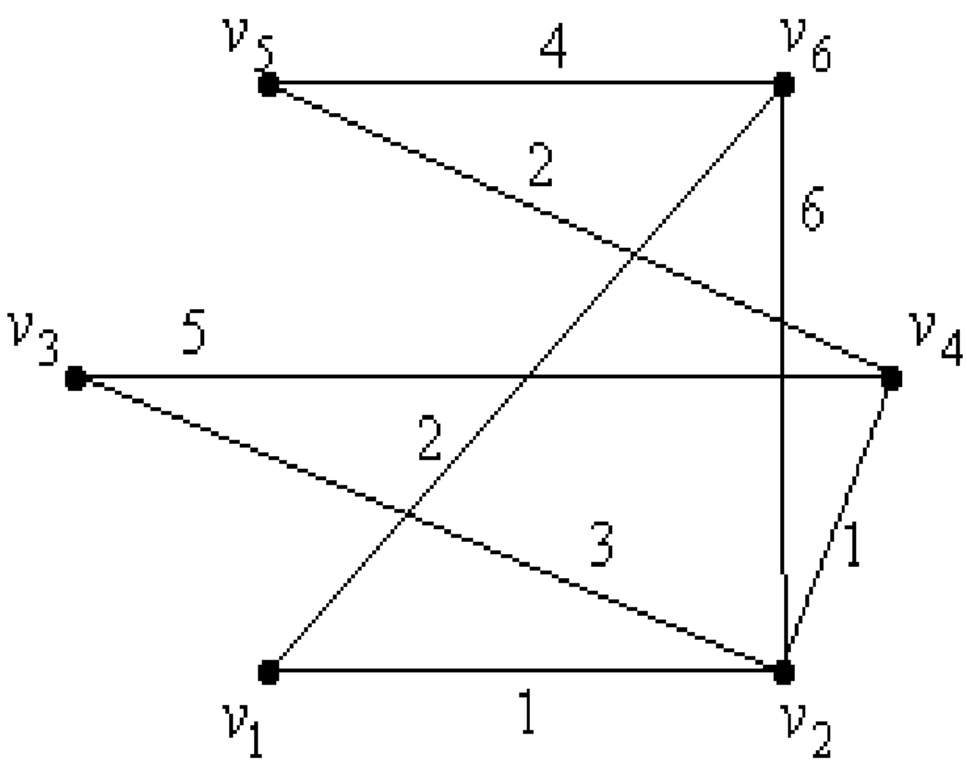
Крок 1. Виберемо в графі G ребро мінімальної ваги. Разом з інцидентними йому вершинами воно утворить підграф G_2 графа G. Покладемо $i = 2$.

Крок 2. Якщо $i = n$, де $n = n(G)$, то задачу розв'язано, і G_2 – шукане мінімальне кістякове дерево графа G. В іншому випадку переходимо до кроку 3.

Крок 3. Будуємо граф G_{i+1}, додаючи до графа G_i нове ребро мінімальної ваги, вибране серед усіх ребер графа G, кожне з яких інцидентне якій-небудь вершині графа G_i і одночасно інцидентне якій-небудь вершині графа G, що не міститься в G_i. Разом із цим ребром включаємо в G_{i+1} інцидентну йому вершину, яка не міститься в G_i. Надамо $i := i + 1$ і переходимо до кроку 2.

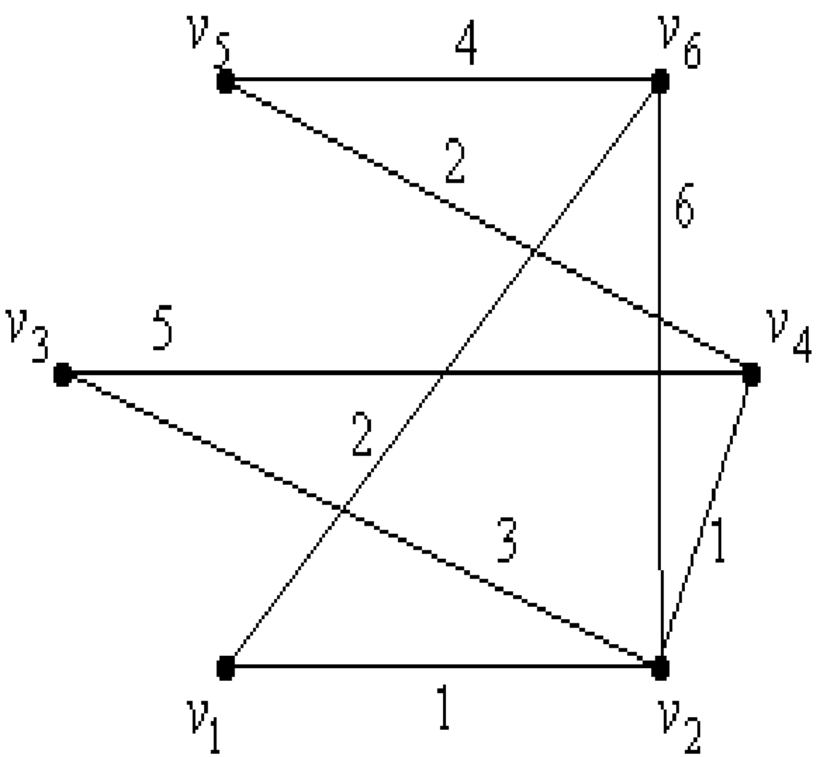
Приклад

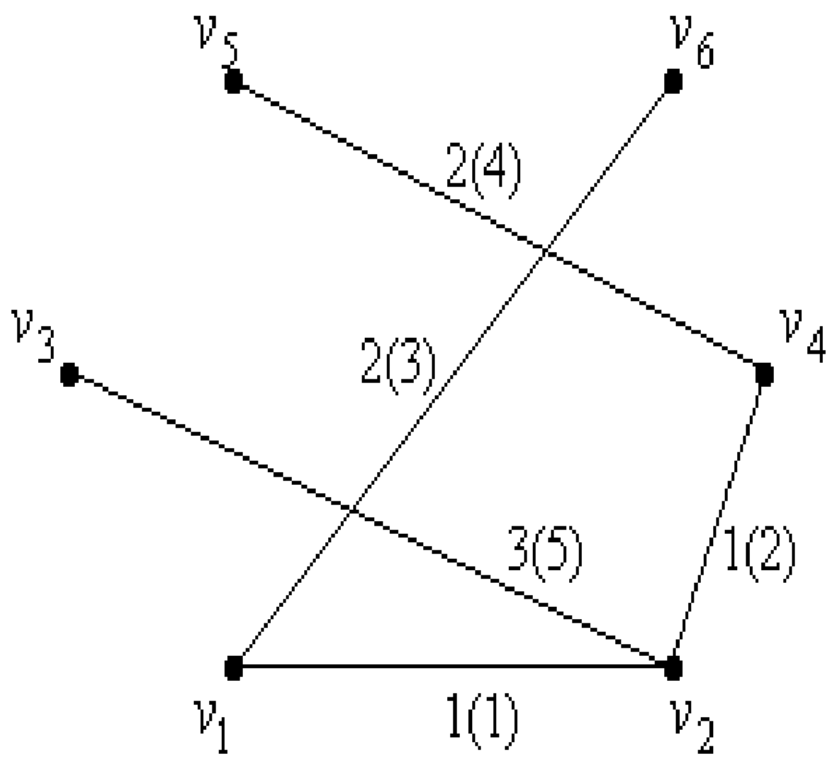
Нехай є звичайний граф G. Знайти кістякове дерево графа G з мінімальною сумою довжин ребер.



Приклад (продовження)

Згідно наведеному алгоритму пошуку кістякового дерева з мінімальною сумою довжин ребер знаходимо кістякове дерево. В дужках позначено номер чергового ребра, яке додаємо до дерева.





Алгоритм знаходження кістякового дерева(продовження)

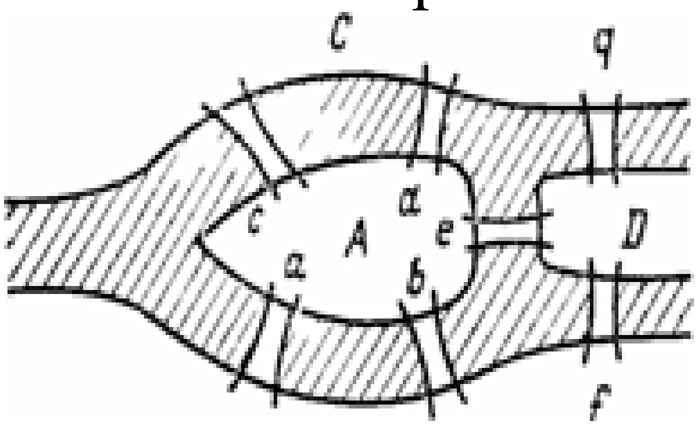
Крок 3. Нехай уже побудовано дерево G_i, що є частиною псевдографа G і містить деякі вершини $v_1,\dots,v_i$, де $1 \le i \le n-1$. Будуємо граф G_{i+1}, додаючи до графа G_i нову вершину $v_{i+1} \in V$, суміжну в G з деякою вершиною v_j графа G_i, та нове ребро, інцідентне вершинам v_i і v_{i+1} (внаслідок зв'язаності G та тієї обставини, що $i < n$, вершина v_{i+1} обов'язково знайдеться). Очевидно, граф G_{i+1} також є деревом. Прирівнюємо $i := i+1$ і переходимо до кроку 2.

12. Ейлерові графи

Постановка задачі

Народженням теорії графів ми зобов'язані постановці й розв’язанню цієї задачі Л. Ейлером.

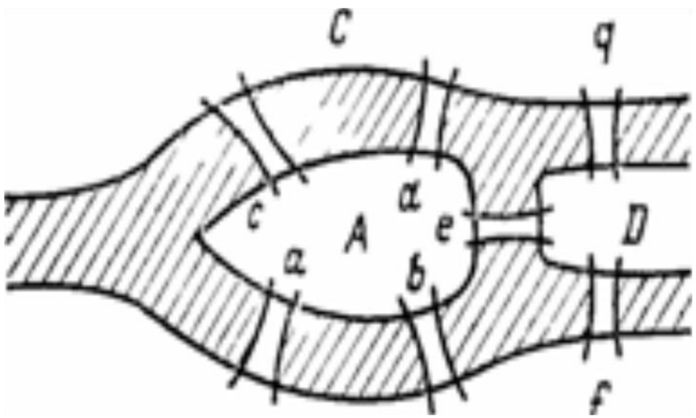
Розташування мостів у м. Кенігсберзі в часи його життя показано на рис.

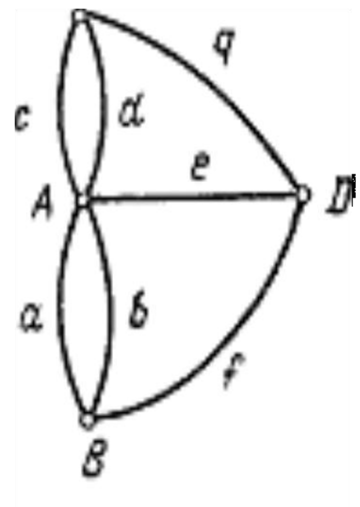


Задача, яку він розв’язував, формулюється так: якщо вийти з деякої частини міста, чи можна пройти кожен міст один раз і повернутися в початкову частину міста.

Постановка задачі

Можна побудувати граф задачі, в якому кожній частині міста відповідає вершина, а кожному мосту – ребро, інцидентне вершинам, що стосуються тих частин міста, що з’єднуються цим мостом (б).





а)

б)

Пояснення

Очевидно, обходу мостів відповідає послідовність ребер графа задачі, в якій два сусідніх ребра мають спільну вершину, тобто маршрут.

Оскільки в кінці обходу треба повернутися в початкову частину міста і на кожному мості побувати по одному разу, цей маршрут є циклом, що містить усі ребра графа.

Ейлерів цикл

Означення 6.36. *Цикли, що містять усі ребра графа, називаються* ***ейлеревими****, а графи, які мають цей цикл, -* ***ейлеревими***.

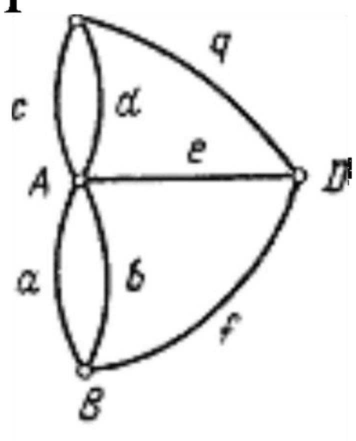
Іншими словами Ейлерові графи - це такі графи, які можна зобразити одним розчерком пера, причому процес цього зображення починається й закінчується в одній і тій самій вершині.

Л. Ейлер розв’язав поставлену задачу і сформулював так звану теорему Ейлера.

Ейлерів цикл

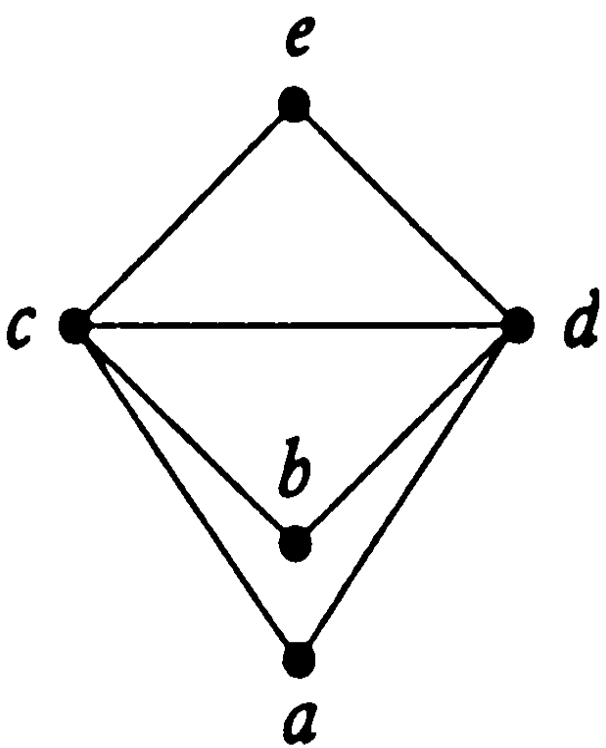
Теорема 6.1 (теорема Ейлера). *Скінчений неорієнтований граф G є ейлеревим тоді й тільки тоді, коли він зв'язний і степені всіх його вершин парні.*

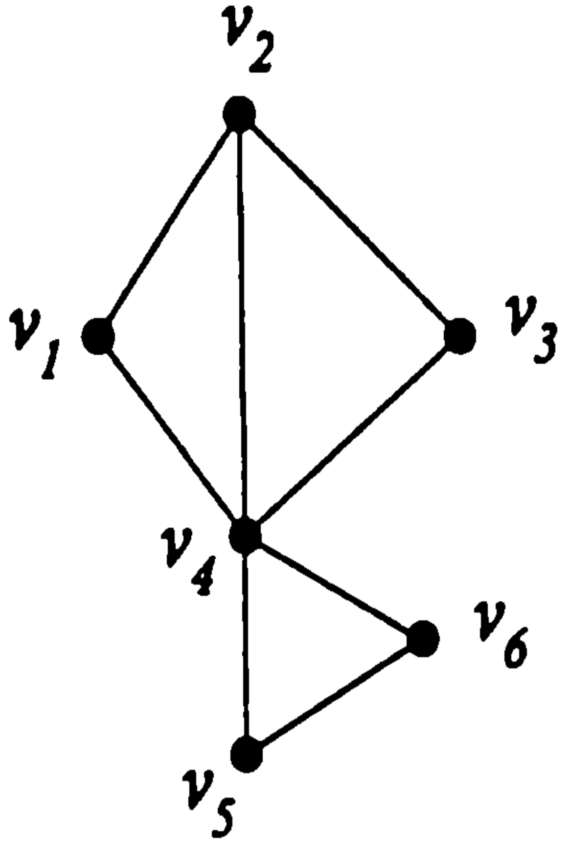
За теоремою Ейлера, сформульована їм задача не має розв’язку, тому що граф, що наведений на рис. не є ейлеревим.



Приклад

Граф на лівому рисунку має ейлерів цикл, а граф на правому не має.





Ейлерів шлях

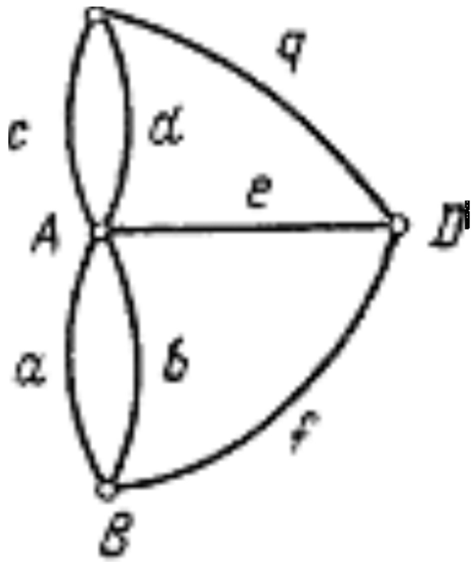
Означення 6.37. Ланцюги, *що містять усі ребра графа, називаються* ***ейлеревими (маршрутами, шляхами)****, а графи, які мають цей цикл, -* ***напівейлеревими***.

Ейлерів ланцюг, який не є ейлеревим циклом називається ***власним ейлеревим ланцюгом (маршрутом, шляхом)***.

Теорема. *Граф має власний ейлерів ланцюг тоді і тільки тоді, коли він зв'язний і рівно дві його вершини мають непарну ступінь.*

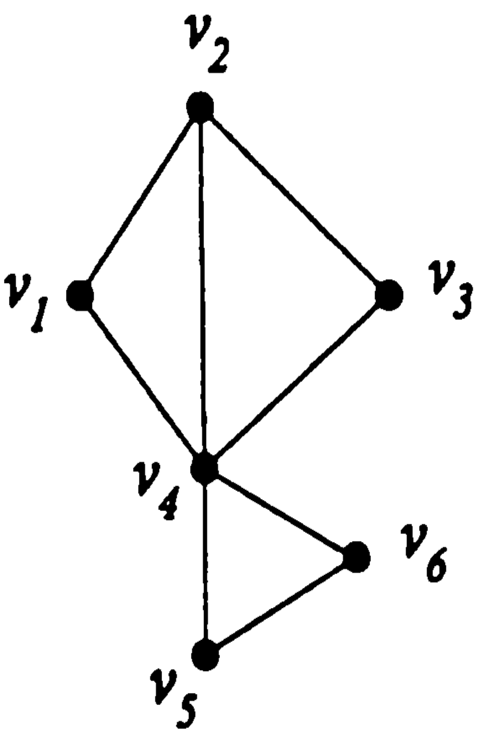
Приклад

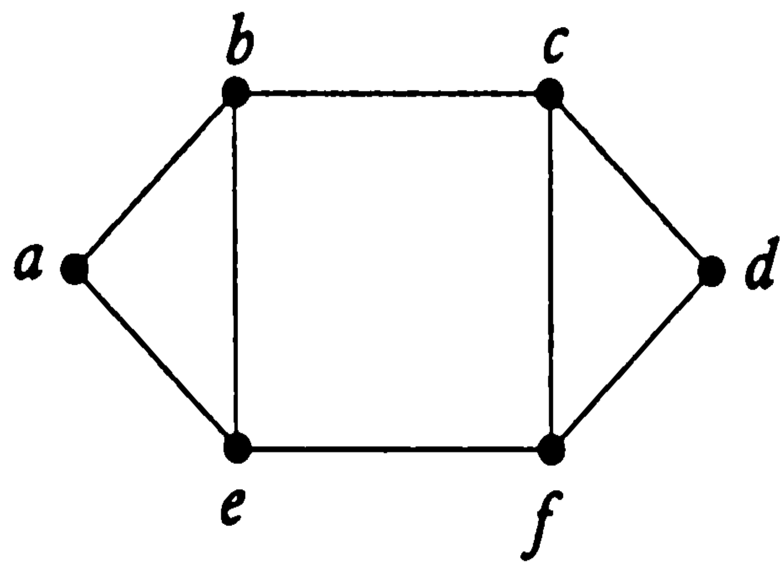
За теоремою граф, що наведений на рис., не має ейлеревого ланцюгу, оскільки 4 його вершини мають непарну степінь.



Приклад

Граф на лівому рисунку має власний ейлерів ланцюг, а граф на правому не має.





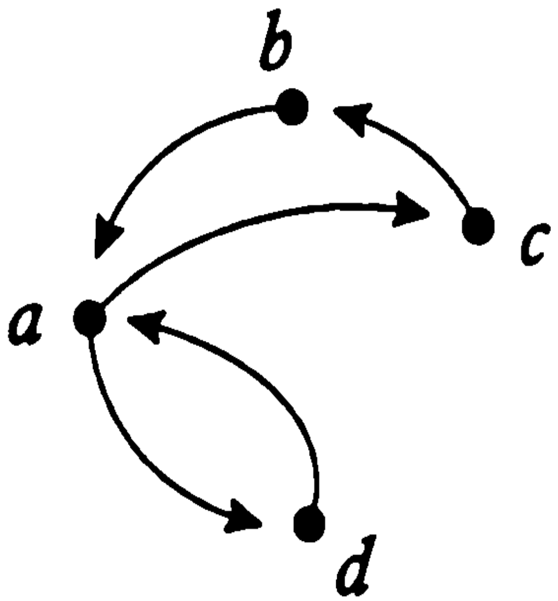
Ейлерів цикл

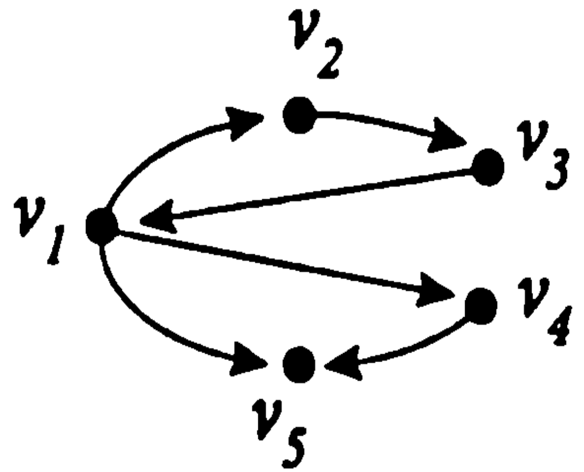
Означення 6.38. *Нехай є орієнтований граф G. Орієнтований цикл, який включає всі ребра і вершини графа G, називається ейлеревим циклом.*

Теорема. *Орієнтований граф має ейлерів цикл тоді і тільки тоді, коли він зв'язний і степінь входу для кожної вершини дорівнює степені її виходу.*

Приклад

Орграф на лівому рисунку має ейлерів шлях, так як степінь входу для кожної вершини дорівнює степені виходу, а на правому не має.





13. Гамільтонові графи

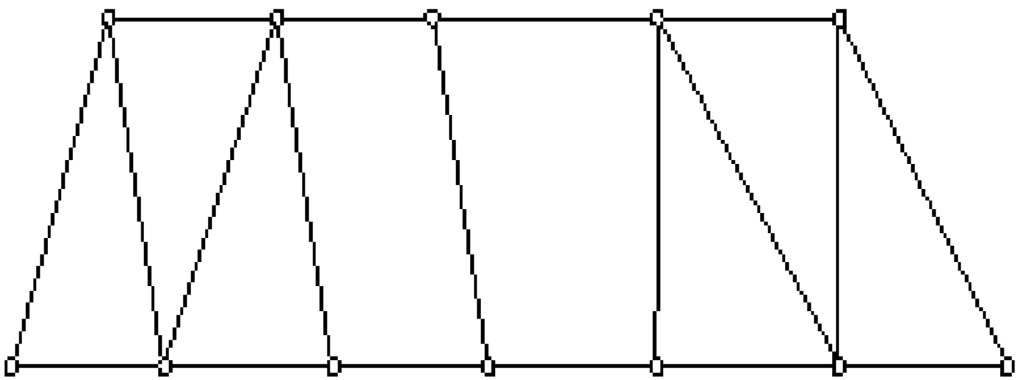
Гамільтонові цикли

Означення 6.40. ***Гамільтоновим*** *циклом називається простий цикл, що проходить через усі вершини графа. Граф, який містить гамільтонів цикл, називається* ***гамільтоновим***.

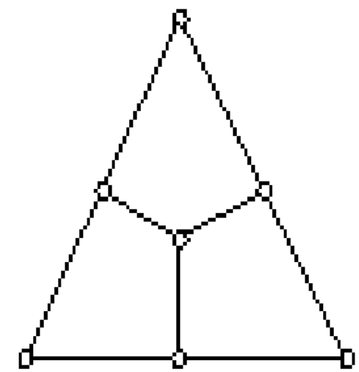
Такий цикл існує далеко не в кожному графі. Більш того, через будь-які дві вершини графа, який розглядається, може пройти простий цикл (у цьому випадку граф *G* називається циклічно зв'язним), а гамільтонів цикл при цьому може бути відсутнім.

Приклад

На рис. а) зображено граф із гамільтоновим циклом, а на рис. б) - циклічно зв'язний граф, в якому немає гамільтонова циклу.



а)

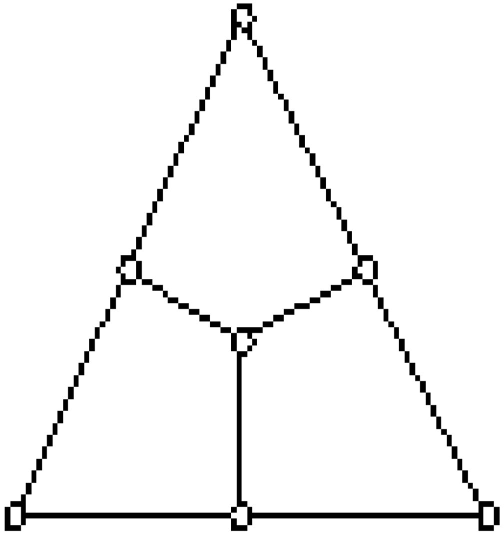


б)

Гамільтонів ланцюг

Іноді можна побудувати простий ланцюг, що проходить через усі вершини графа з початком і кінцем у різних заданих вершинах $v', v'' \in G$.

Такий ланцюг також називається **гамільтоновим**. У графі на рис. є гамільтонів ланцюг.



Достатня умова існування гамільтонового циклу

Теорема. *Якщо зв'язний граф G(V,E) з n вершинами, де* $n \geq 3$, *і для кожної пари різних несуміжних вершин* $u, v \in V, \deg(u) + \deg(v) \geq n$, *тоді граф G має гамільтонів цикл.*

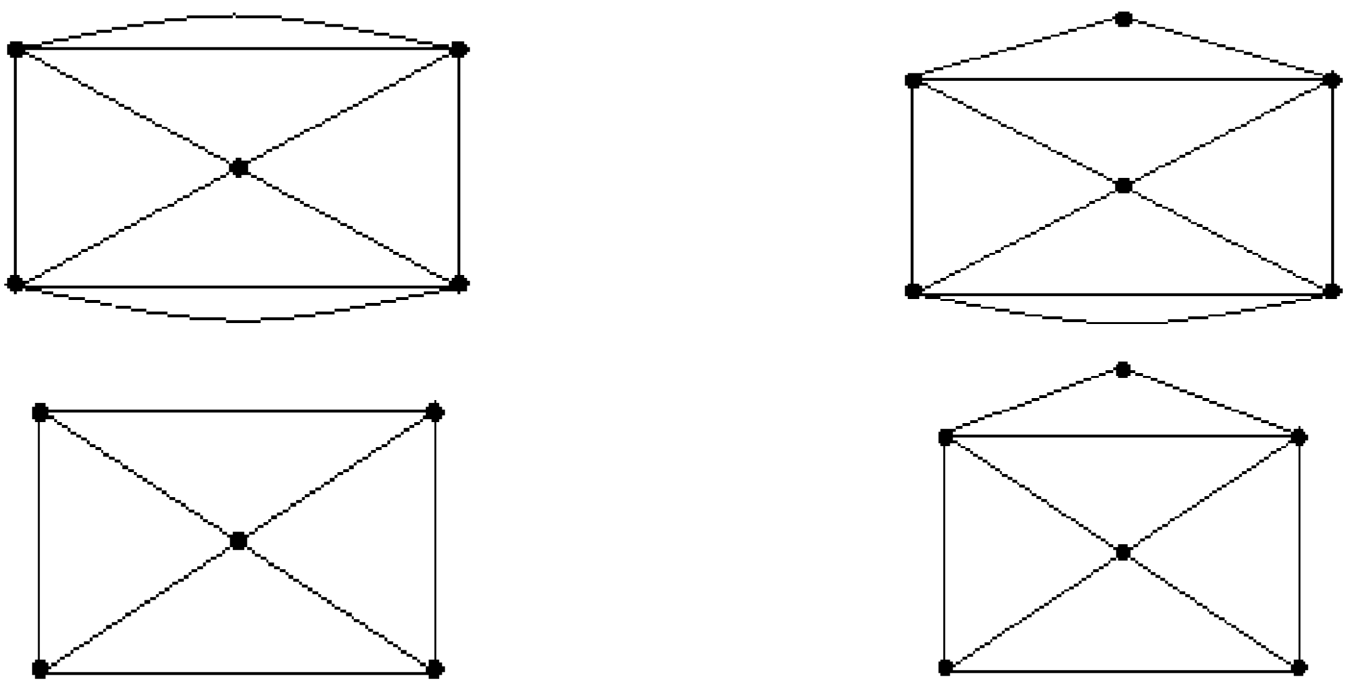
Наслідок. *Якщо зв'язний граф G(V,E) з n вершинами, де* $n \geq 3$, *і якщо для кожної вершини* $v \in V$ *виконується умова* , $\deg(v) \geq n/2$, *то граф G має гамільтонів цикл.*

Задача коммівояжера

Із поняттям гамільтонових циклів тісно пов’язана так звана задача комівояжера: в зваженому графі *G* визначити гамільтонів цикл мінімальної ваги (іншими словами, комерсант повинен здійснити поїздку по містах і повернутися назад, побувавши в кожному місті рівно один раз; при цьому вартість поїздки повинна бути мінімальною).

Порівняння гамільтонових і ейлерових графів

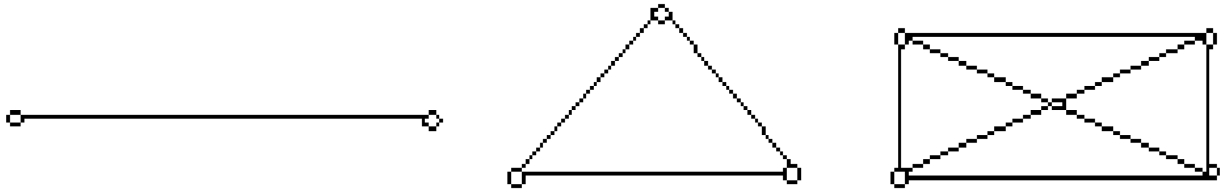
На перший погляд, поняття гамільтонова циклу подібне до поняття ейлерева цикла. Однак це не так. В першому рядку наведені ейлерові графи, в другому рядку - неейлерові, в першому стовпці наведені гамільтонові графи, в другому - негамільтонові.



14. Планарність графів

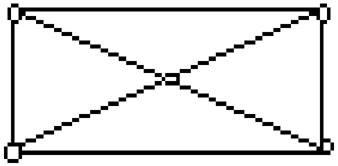
Планарні графи

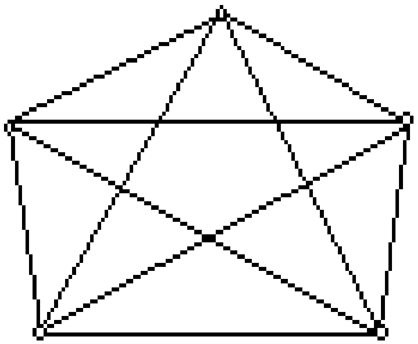
Означення 6.41. *Граф G називається* ***планарним*** *(плоским), якщо існує ізоморфний йому граф, який може бути зображений на площині без перетину ребер*.



Планарні графи

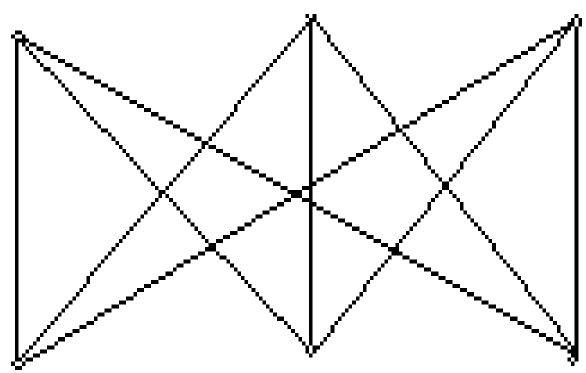
Повний граф на п’яти вершинах P_5 (K_5) не є планарним, а граф P_4 (K_4) - це повний плоский граф.





Планарні графи

Розглянемо повний дводольний граф $K_{3,3}$, який є математичною моделлю відомої задачі про три будинки і три колодязі, що формулюється так. Є три будинки і три колодязі. Сусіди ворогують, не хочуть зустрічатися, але хочуть користуватися всіма трьома колодязями. Чи можна прокласти стежки від кожного будинку до кожного колодязя так, щоб вони не перетиналися? Ця задача нерозв'язна, тому що граф $K_{3,3}$ - неплоский.



Умова планарності графів

Графи P_5 і $K_{3,3}$ відіграють фундаментальну роль у теорії планарності. Їх називають **графами Понтрягіна-Куратовського**.

Твердження. *Граф G є планарним (плоским) тоді і тільки тоді, коли він не містить підграфів Понтрягіна-Куратовського.*

15. Задачі пошуку маршрутів в графі

Постановка задачі

При розв'язанні широкого кола прикладних задач нерідко виникає необхідність знайти маршрут, що зв'язує задані вершини в графі G.

Задача зводиться до пошуку маршруту у зв'язному графі $G = (V, E)$, який з'єднує задані вершини $v, w \in V$, де $v \neq w$.

Постановка задачі:

Дано:

1) $G = (V, E)$ - зв'язний граф.

2) Вершини $v, w \in V$, де $v \neq w$, у графі G.

Знайти: Маршрут, що зв'язує задані вершини $v, w \in V$, де $v \neq w$, у графі G.

Алгоритм Террі

У зв'язному графі G завжди можна знайти маршрут, що зв'язує дві задані вершини v та w, якщо, виходячи з вершини v і здійснюючи послідовний перехід від кожної досягнутої вершини до суміжної з нею, керуватися такими правилами:

Алгоритм Террі

1) йдучи по довільному ребру, кожний раз відмічати напрямок, в якому воно було пройдене;

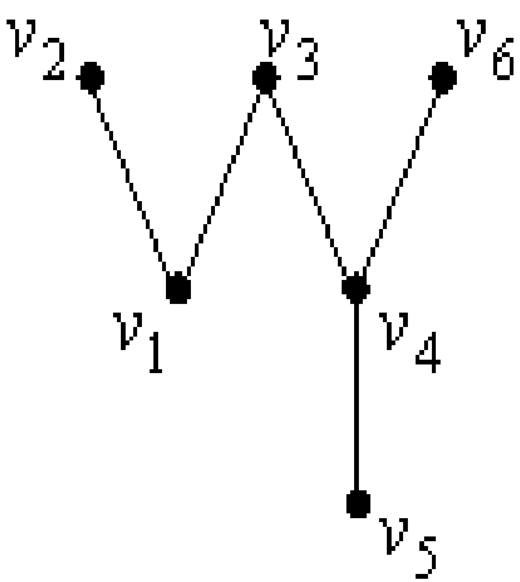
2) виходячи з деякої вершини v_1, завжди рухатися тільки по тому ребру, яке не було пройдене або було пройдене у зворотному напрямку;

3) для кожної вершини v_1, відмінної від v, відмічати те ребро, яке першим заходить у v_1, якщо вершина v_1 зустрічається вперше;

4) виходячи з деякої вершини v_1, відмінної від v, по першому ребру, яке заходить у v_1, рухатися лише тоді, коли немає інших можливостей.

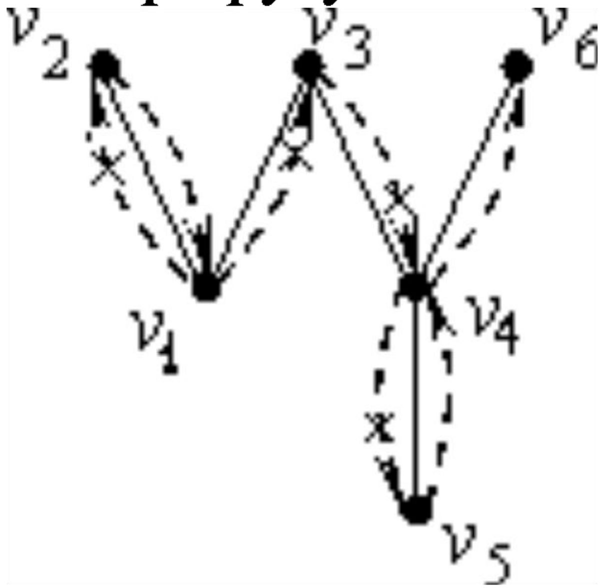
Алгоритм Террі. Приклад

Використовуючи алгоритм Террі, знайти маршрут, що зв'язує v_1 і v_6 у графі G, зображеному на рис. Граф G - це схема лабіринту, де v_6 - вихід із нього, а v_1 - розвилка, з якої починається пошук виходу.



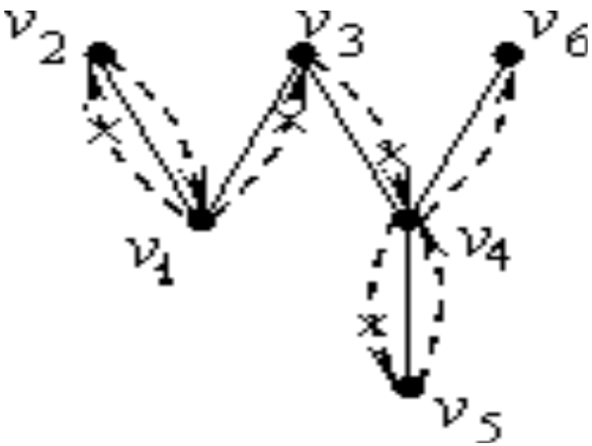
Алгоритм Террі. Приклад

На рис. показано один із можливих варіантів руху по графу G згідно з алгоритмом Террі. Штриховими дугами зображено схему руху по графу G. Знаками помічено перші ребра, які заходять у вершини (помітка робиться ближче до тієї вершини, в яку ребро заходить). Ця схема руху відповідає маршруту $v_1v_2v_1v_3v_4v_3v_5v_6$.



Алгоритм Террі. Приклад

Зазначимо, що після того, як із вершини v_1 зайшли у вершину v_3, внаслідок правила 4 не можна повернутися у v_1, оскільки існують інші можливості, а (v_1, v_3) є першим ребром, що заходить у v_3. Далі, після того, як із вершини v_4 зайшли у вершину v_5, внаслідок правила 4 треба рухатися знов до вершини v_4, і далі в наслідок правила 2 рухатися до вершини v_6.



16. Пошук відстані між вершинами графа

Довжина шляху та відстань між вершинами орграфа

Розглянемо деякі властивості мінімальних шляхів (маршрутів). Раніше було наведено означення довжини маршруту та відстані між вершинами звичайного графа G. Аналогічно визначаються поняття довжини шляху і відстані між вершинами орієнтованого графа D.

Образ і прообраз вершини

Означення 6.42. *Назвемо* ***образом вершини*** x *в орієнтованому графі D множину кінців дуг, початком яких є вершина* x, *і позначимо його D(x), а множину початків дуг, кінцем яких є вершина* x, *назвемо* ***прообразом вершини*** x *і позначимо його* $D^{-1}(x)$.

Алгоритм пошуку відстані та мінімального шляху (1)

Нехай $D = (V, E)$ - орієнтований граф із n вершинами $(n \geq 2)$ і v, w - задані вершини з V, де $v \neq w$. Опишемо алгоритм пошуку відстані та відповідного їй мінімального шляху з v у w в орієнтованому графі D.

Крок 1. Позначаємо вершину v індексом 0, а вершини, що належать образу вершини v, індексом 1. Вважаємо $k = 1$.

Крок 2. Множину вершин з індексом k позначаємо $FW_k(v)$. Якщо $FW_k(v) = \varnothing$ або виконується $k = n-1$ і $w \notin FW_k(v)$, то вершина w не досяжна з v, і робота алгоритму на цьому завершується. В іншому випадку переходимо до кроку 3.

Алгоритм пошуку відстані та мінімального шляху(2)

Крок 3. Якщо $w \notin FW_k(v)$, то переходимо до кроку 4. В іншому випадку існує шлях з v у w довжини *k*, причому цей шлях є мінімальним. Послідовність вершин

$v, w_1, w_2, \ldots, w_{k-1}, w$,

де

$$
\begin{aligned}
& w_{k-1} \in FW_{k-1}(v) \cap D^{-1}(w); \\
& w_{k-2} \in FW_{k-2}(v) \cap D^{-1}(w_{k-1}); \\
& \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
& w_1 \in FW_1(v) \cap D^{-1}(w_2).
\end{aligned}
\qquad (6.2)
$$

і є шуканий мінімальний шляхом з v у w. На цьому робота алгоритму закінчується.

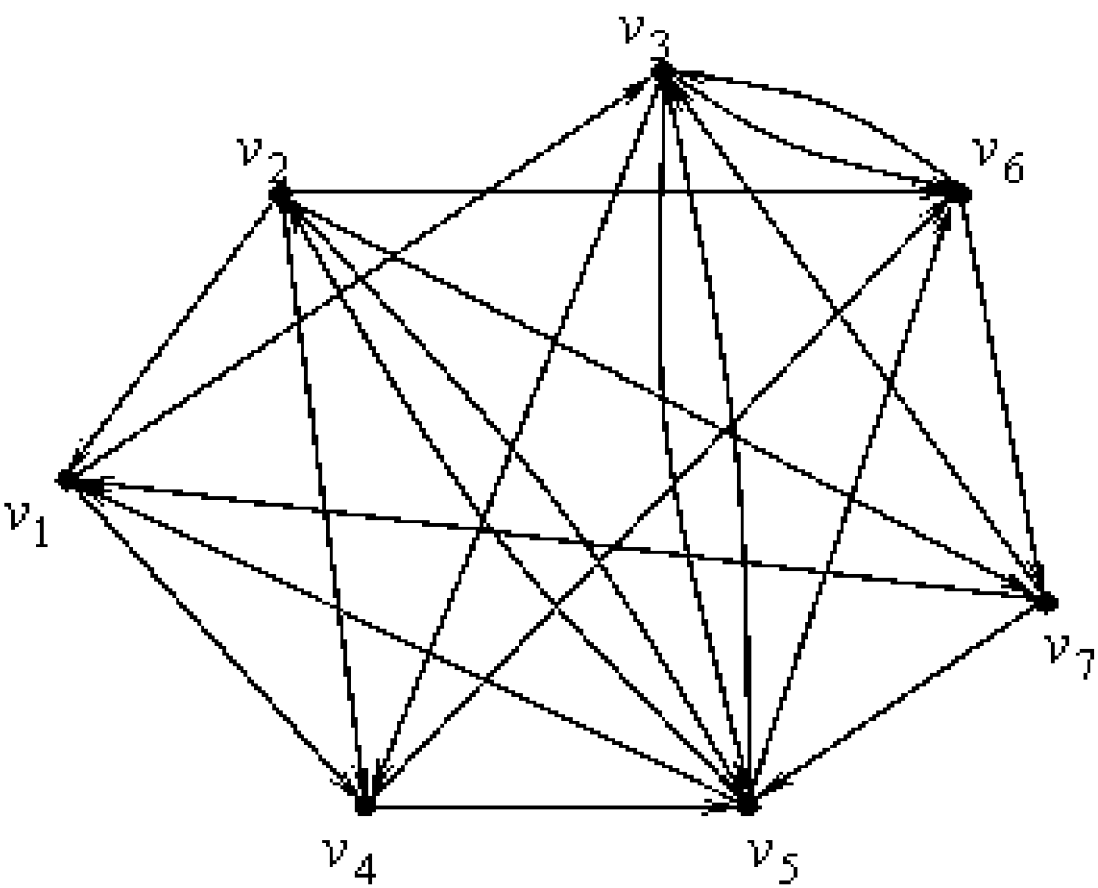
Алгоритм пошуку відстані та мінімального шляху(3)

Крок 4. Позначаємо індексом $k+1$ всі непозначені вершини, що належать образу множини вершин з індексом k. Множину вершин з індексом $k+1$ позначаємо $FW_{k+1}(v)$. Нехай $k:=k+1$ і переходимо до кроку 2.

Вершини $w_1,...,w_{k-1}$ із (6.2), взагалі кажучи, можуть бути визначені неоднозначно. Ця неоднозначність відповідає випадкам, коли існує декілька різних мінімальних шляхів з v у w в орграфі D.

Алгоритм пошуку відстані та мінімального шляху. Приклад(1)

Використовуючи алгоритм, визначити мінімальний шлях із v_1 у v_7 в орграфі *D*, заданому на рис.



Алгоритм пошуку відстані та мінімального шляху. Приклад(2)

Матриця суміжності цього графа така:

	v_1	v_2	v_3	v_4	v_5	v_6	v_7
v_1	0	0	1	1	0	0	0
v_2	1	0	0	1	1	1	1
v_3	0	0	0	1	1	1	0
v_4	0	0	0	0	1	1	0
v_5	1	1	1	0	0	1	0
v_6	0	0	1	0	0	0	1
v_7	1	0	1	0	1	0	0

Алгоритм пошуку відстані та мінімального шляху. Приклад(3)

Діючи згідно з алгоритмом, послідовно знаходимо

$$FW_1(v_1) = \{v_3, v_4\}$$

$$FW_2(v_1) = D(FW_1(v_1)) - \{v_1, v_3, v_4\} = \{v_4, v_5, v_6\} - \{v_1, v_3, v_4\} = \{v_5, v_6\}$$

$$FW_3(v_1) = D(FW_2(v_1)) - \{v_1, v_3, v_4, v_5, v_6\} = \{v_1, v_2, v_3, v_6, v_7\} -$$

$$- \{v_1, v_3, v_4, v_5, v_6\} = \{v_2, v_7\}$$

Таким чином, $v_7 \in FW_3(v_1)$, а отже, існує шлях із v_1 у v_7 завдовжки 3, і цей шлях є мінімальним.

Алгоритм пошуку відстані та мінімального шляху. Приклад(4)

Визначимо цей мінімальний шлях із v_1 в v_7.

$$FW_2(v_1) \cap D^{-1}(v_7) = \{v_5, v_6\} \cap \{v_2, v_6\} = \{v_6\},$$

$$FW_2(v_1) \cap D^{-1}(v_6) = \{v_3, v_4\} \cap \{v_2, v_3, v_4, v_5\} = \{v_3, v_4\}.$$

Це означає, що існує 2 шляхи завдовжки 3 з вершини v_1 в v_7, а саме (v_1, v_3, v_7) і (v_1, v_4, v_7).

17. Мінімальні шляхи у зважених орієнтованих графах

Зважений орграф

Орієнтований граф $D = (V, E)$ є зваженим, якщо на множині дуг E визначено вагову функцію $l := E \to R$.

Таким чином, у зваженому орієнтованому графі D кожній дузі $e \in E$ поставлено у відповідність вагу l.

Значення $l(e)$ будемо називати **вагою дуги** e.

Вага шляху

Для будь-якого шляху M зваженого орієнтованого графа D позначимо через $l(M)$ суму ваг дуг, що входять у M, при цьому кожна дуга враховується стільки разів, скільки вона входить у шлях.

Величину $l(M)$ будемо називати **вагою шляху** M у зваженому орієнтованому графі D.

Якщо ваги дуг дорівнюють 1, то $l(M)$ виражає введену раніше довжину шляху M у не незваженому орієнтованому графі.

Отже, будь-який не незважений орієнтований граф можна вважати зваженим із вагами дуг, що дорівнюють 1.

Вага шляху

Означення 6.43. *Шлях у зваженому орієнтованому графі D з вершини v у вершину w, де $v \neq w$, називається* ***мінімальним****, якщо він має найменшу вагу серед усіх шляхів з v в w.*

Властивості мінімальних шляхів у зваженому орієнтованому графі $D = (V, E)$

1) будь-який мінімальний шлях (маршрут) є простим ланцюгом;

2) якщо $v_1, v_2, ..., v_k$ - мінімальний шлях, то для будь-яких проміжних вершин v_i, v_j, шлях $v_i, v_{i+1}, ..., v_j$ також є мінімальним;

3) якщо $v, ..., u, w$ - мінімальний шлях (маршрут) серед шляхів з v у w, що містять не більше як $k+1$ дуг, то $v..u$ - мінімальний шлях серед шляхів з v у u, що з'єднують не більше за k дуг.

Алгоритми знаходження мінімального шляху на графі

Дивись розділ 14.5 в книжці «Дискретная Математика и Комбинаторика» Джеймс Андерсон

Питання, які були розглянуті

1. Основні поняття теорії графів
2. Подання графа за допомогою матриці інцидентності
3. Подання графа за допомогою матриці суміжності графа
4. Визначення локальних степенів вершин графа. Повні графи
5. Ізоморфізм графів
6. Частини графа, суграфи й підграфи
7. Графи і бінарні відношення
8. Маршрути, шляхи, ланцюги та цикли
9. Зв'язність
10. Дерева
11. Кістякове дерево зв'язного графа
12. Ейлерові графи
13. Гамільтонові графи
14. Планарність графів
15. Задачі пошуку маршрутів в графі
16. Пошук відстані між вершинами графа
17. Мінімальні шляхи у зважених орієнтованих графах