

## 5. ТЕОРІЯ ГРАФІВ

### 5.1. Основні поняття теорії графів

#### 5.1.1. Предмет теорії графів. Основні означення.

Виникнення теорії графів пов'язано із *задачею про Кенігсберзькі мости*. Схема міста Кенігсберга наведена на рис. 5.1. Потрібно обійти усі чотири ділянки суші, пройшовши по кожному мосту один раз.

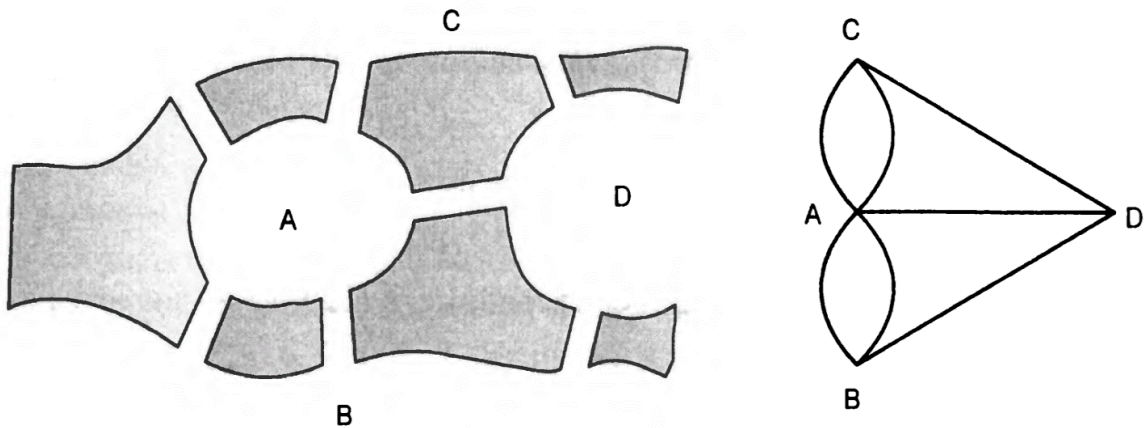


Рис. 5.1. Кенігсберзькі мости

Ще однією класичною задачею є *задача про три криниці*. Потрібно прокласти стежки від кожного будинку до кожної криниці таким чином, щоб стежки не перетиналися (див. рис. 5.2).

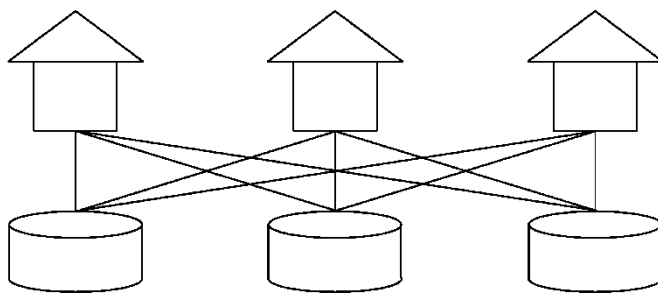


Рис. 5.2. Ілюстрація до задача про три криниці

*Мультимножина (набір)* — невпорядкована система об'єктів, *можливо із повтореннями*. При записі елементи мультимножини записуються у фігурних або трикутних дужках.

Приклад мультимножини:  $\{a, b, b, c, a, a\}$ .

Графом  $G$  називається пара  $(V, E)$ , де  $V$  — множина вершин графа,  $E$  — мультимножина ребер графа.

Якщо множини  $V$  та  $E$  — скінченні, то граф називається *скінченним*.

Елементи мультимножини  $E$  називаються *ребрами* графа. Ребро  $e$ , яке з'єднує вершини  $a$  та  $b$  позначається  $\langle a, b \rangle$  ( $e = \langle a, b \rangle \in E$ ,  $a, b \in V$ ), при цьому вершини  $a$  та  $b$  називаються *суміжними* та *інцидентними* ребру  $e$ . Виокремлюють три види ребер:

- ланки або неупорядковані ребра ( $e = \{a, b\} \in \bar{E}$ ). Набір ланок позначається  $\bar{E}$ ;
- дуги або впорядковані ребра ( $e = (a, b) \in \vec{E}$ ),  $\vec{E}$  — набір дуг;
- петлі — ребра вигляду  $e = (a, a)$ , початок і кінець якого співпадають.

Надалі будемо вважати, що усі графи є скінченними і петлі є різновидом дуг, тобто  $E = \bar{E} \cup \vec{E}$ .

Кількість вершин графа називається його *порядком*. Приклад діаграми графа наведено на рис. 5.3.

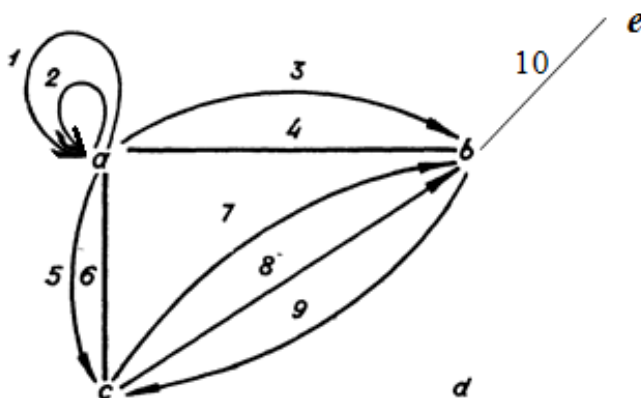


Рис. 5.3. Діаграма графа

Кількість ланок, інцидентних вершині  $v$ , будемо позначати  $\overline{\deg}(v)$ .

Кількість дуг, які входять у вершину  $v$ , будемо позначати  $\deg^+ v$ .

Кількість дуг, які виходять із вершини  $v$ , будемо позначати  $\deg^- v$ .

Величина  $\deg v$ , яка обчислюється за формулою

$$\deg v = \overline{\deg} v + \deg^+ v + \deg^- v,$$

називається *степеню* вершини  $v$ .

**Приклад 5.1.** Обчислимо степінь вершини  $a$  графа, діаграма якого зображена на рис. 5.3.

$$\overline{\deg}(a) = 2, \deg^+(a) = 2, \deg^-(a) = 4, \deg(a) = 2 + 2 + 4 = 8.$$

Вершина  $v$  називається *ізольованою*, якщо  $\deg(v) = 0$ .

Вершина  $v$  називається *вісячою*, якщо  $\deg(v) = 1$ .

Для графа, наведеного на рис. 5.3, вершина  $d$  є ізольованою, а вершина  $e$  — вісяча.

**Лема про рукостискання.** Для довільного графа  $(V, E)$

$$\sum_{v \in V} \deg(v) = 2|E|;$$

$$\sum_{v \in V} \deg^+(v) + \sum_{v \in V} \deg^-(v) = 2|\vec{E}|.$$

Граф  $G_1 = (E_1, V_1)$  називається *підграфом* графа  $G = (V, E)$ , якщо  $V_1 \subseteq V, E_1 \subseteq E$ .

**Приклад 5.2.** Граф, діаграма якого наведена на рис. 5.4, є підграфом графа з рис. 5.3.

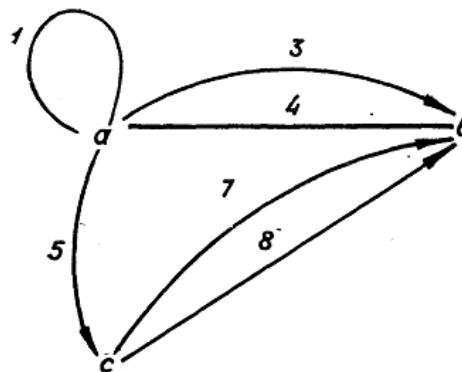


Рис. 5.4. Підграф графа, наведеного на рис. 5.3

*Ізоморфізм графів* — це бієкція (взаємно-однозначна відповідність) множин вершин графів, яка зберігає суміжність вершин. Тобто графи  $G_1 = (V_1, E_1)$  та  $G_2 = (V_2, E_2)$  *ізоморфні*, якщо бієкція  $f: V_1 \rightarrow V_2$  задовольняє умову:

$$\text{для довільних } a, b \in V_1 \quad \langle a, b \rangle \in E_1 \Leftrightarrow \langle f(a), f(b) \rangle \in E_2.$$

**Теорема 5.1.** Ізоморфізм графів є відношенням відношення еквівалентності (на множині графів).

**Приклад 5.3.** Графи, діаграми яких наведені на рис. 5.5, є ізоморфними.

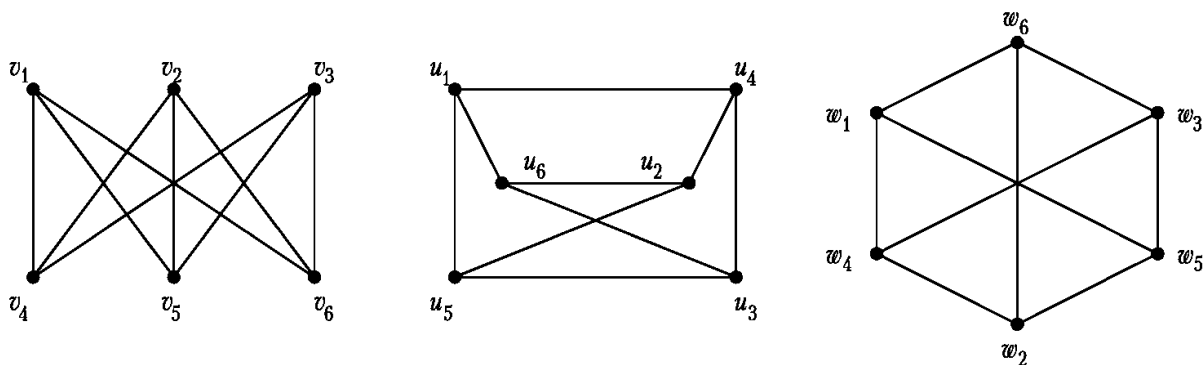


Рис. 5.5. Ізоморфні графи

Слід зазначити, що кількість вершин, ребер та степені вершин не визначають граф однозначно. На рис. 5.6. наведено діаграми двох неізоморфних графів, для яких усі відповідні параметри співпадають:



Рис. 5.6. Діаграми неізоморфних графів

### 5.1.2. Способи задання графів.

Основні способи задання графів:

а) *перелік елементів*: Множин  $E$  та  $V$  задаються переліком їх елементів.

**Приклад 5.4.** Задамо переліком елементів граф, діаграма якого наведена на рис. 5.3:

$$G = (V, E), V = \{a, b, c, d, e\}, E = \{e_1 = (a, a), e_2 = (a, a), e_3 = (a, b), e_4 = \{a, b\}, e_5 = (a, c), e_6 = \{a, c\}, e_7 = (c, b), e_8 = (c, b), e_9 = (b, c), e_{10} = \{b, e\}\};$$

б) графічна інтерпретація (діаграма);

с) *матриця інцидентності*.

Рядки матриці відповідають вершинам, стовпці — ребрам. Якщо вершина  $v_i$  інцидентна ребру  $e_j$ , то елемент  $b_{ij}$  матриці інцидентності  $B$  рівний:

- 1)  $-1$ , якщо  $e_j$  — дуга, яка виходить з вершини  $v_i$  і не є петлею;
- 2)  $1$  — у всіх інших випадках.

Якщо вершина не інцидентна ребру, то відповідний елемент матриці рівний 0.

Для графа з рис. 5.3 матриця інцидентності має вигляд

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

d) *матриця суміжності (сусідства)*

Матриця сусідства  $A$  — квадратна матриця порядку  $|V|$ , елементи визначаються наступним чином:

$$a_{ij} = \alpha \bar{a}_{ij} + \beta \vec{a}_{ij},$$

де  $\bar{a}_{ij}$  — кількість ланок, які з'єднують вершини  $v_i$  та  $v_j$ ,  $\vec{a}_{ij}$  — кількість дуг, які виходять із вершини  $v_i$  і входять у вершину  $v_j$ .

Для графа з рис. 5.3 матриця суміжності має вигляд

$$A = \begin{pmatrix} 2\beta & \alpha + \beta & \alpha + \beta & 0 & 0 \\ \alpha & 0 & \beta & 0 & \alpha \\ \alpha & 2\beta & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & \alpha & 0 & 0 & 0 \end{pmatrix}.$$

### 5.1.3. Основні види графів.

Виокремлюють наступні види графів:

- 1) порожній граф (або 0-граф) ( $V = \emptyset$ );
- 2) неорієнтований граф ( $E = \bar{E}$ );
- 3) звичайний граф (неорієнтований без кратних ребер);
- 4) орієнтований граф (усі ребра — дуги, тобто  $E = \vec{E}$ );

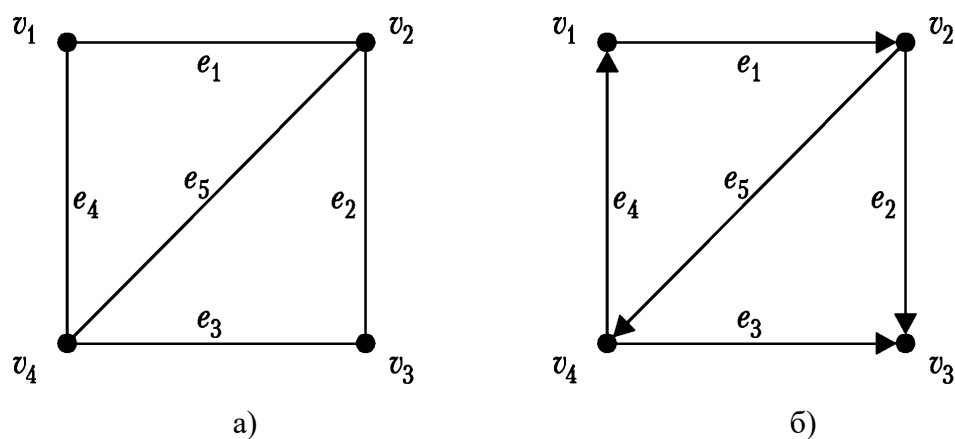


Рис. 5.7. Діаграми звичайного а) та орієнтованого б) графів

Неорієнтований граф  $G' = (V, E')$  називається *відповідним* до орієнтованого графа  $G = (V, E)$ , якщо  $E' = \{\{a, b\} \mid (a, b) \in E\}$ . Граф 5.7 а) є відповідним до графа 5.7 б).

- 5) мультиграф (допустимими є кратні ребра);
- 6) псевдограф (мультиграф, для якого допускаються петлі);
- 7) дводольний (біхроматичний) граф — множина вершини  $V$  розбивається на дві множини  $V_1, V_2$  ( $V_1 \cup V_2 = V, V_1 \cap V_2 = \emptyset$ ) так, що кожне ребро інцидентні одній вершині з  $V_1$  та одній вершині з  $V_2$ ;
- 8) граф Кеніга — звичайний дводольний граф (див. рис. 5.8);
- 9) повний граф — граф, який містить усі ребра для графів заданого типу:
  - а)  $K_n$  — повний звичайний граф;

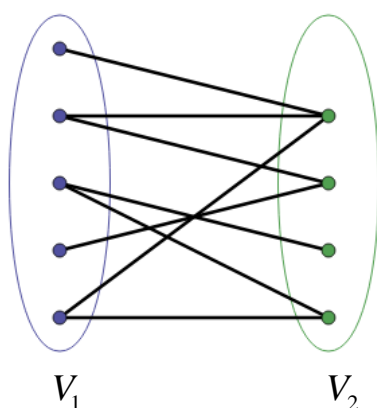
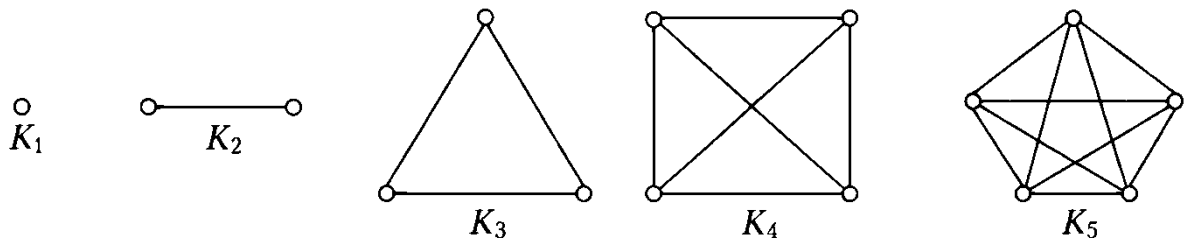


Рис. 5.8. Граф Кеніга

Рис. 5.9. Повні графи  $K_n$  для  $n = 1, 2, 3, 4, 5$ 

б)  $K_{n,m}$  — повний граф Кеніга ( $n$  та  $m$  — кількості елементів множин  $V_1$  та  $V_2$ );

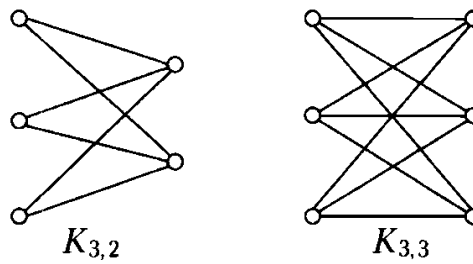
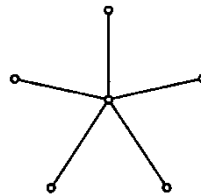


Рис. 5.10. Повні графи Кеніга

Граф  $K_{1,m}$  називається *зірковим*.

Рис. 5.11. Зірковий граф  $K_{1,5}$ 

10) *Однорідні* (регулярні) графи степені  $d$  — звичайні графи, степені усіх вершин яких рівні  $d$  ( $\forall v \in V \deg v = d$ ). Зв'язний однорідний граф степені 2 називається *циклічним графом*. Циклічний граф  $n$ -го порядку позначається  $C_n$ .

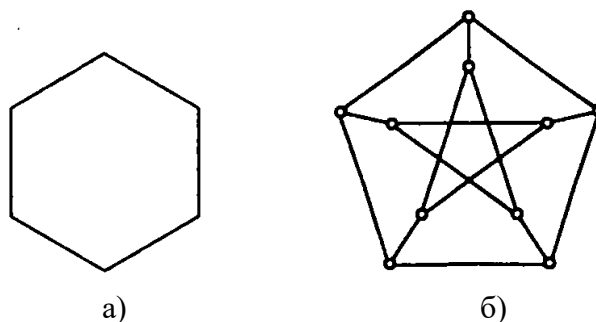
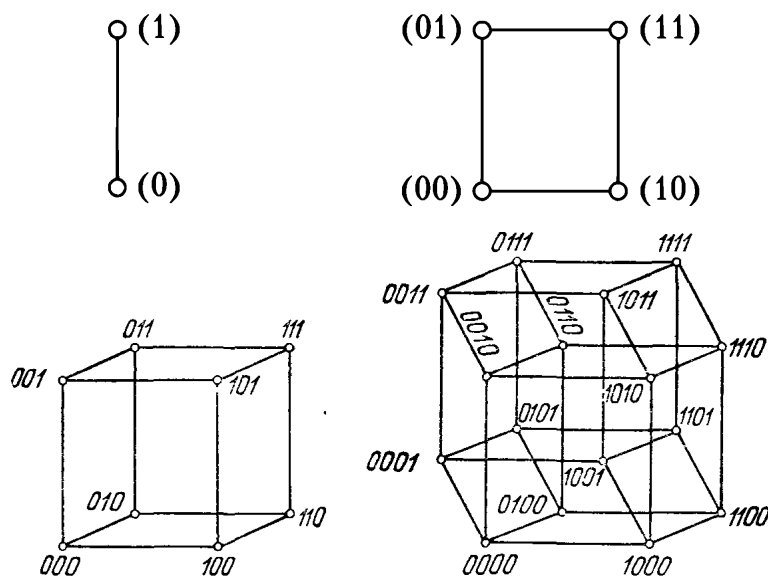


Рис. 5.12. Регулярні графи: а) граф  $C_6$ ; б) кубічний граф Петерсена

**Теорема 5.2.** Нехай  $n, d \in \mathbb{N}$  — натуральні числа,  $0 \leq d \leq n-1$ . Тоді існує регулярний граф  $n$ -го порядку степеня  $d$ .

- 11)  $n$ -вимірний куб — звичайний граф, вершинами якого є  $n$ -вимірні булеві вектори, вершини  $u$  та  $v$  суміжні тоді і тільки тоді, коли  $d(u, v) = 1$  (відстань Хеммінга між вершинами рівна 1).

Рис. 5.13.  $n$ -вимірний куб

- 12) *Турнір* (повний орієнтований граф без петель та кратних дуг).

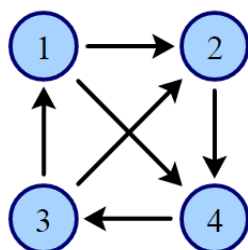


Рис. 5.14. Діаграма турніру

- 13) *Помічені та нумеровані графи.*

Якщо задана функція  $f: V \rightarrow M$  та (або)  $g: E \rightarrow M$ , то множина  $M$  називається множиною *міток*, а граф  $G$  називається *поміченим*. Якщо функції  $f$  та  $g$  однозначні (ін'єктивні) і  $M \subset \mathbb{N}$ , то граф називається *(про)нумерованим*.

Прикладом нумерованого графа є граф, зображений на малюнку 5.14.



## 5.2. Маршрути у графі

Нехай  $G = (V, E)$  — заданий граф,  $a = v_{i_0}$ ,  $b = v_{i_n}$ . Послідовність вершин та ребер

$$M(a, b) = ae_{j_1}v_{i_1}e_{j_2}\dots v_{i_{n-1}}e_{j_n}b,$$

де  $e_{j_k}$  — ребро, інцидентне вершинам  $v_{i_{k-1}}$  та  $v_{i_k}$  називається *маршрутом довжини  $n$* , який з'єднує вершини  $a$  та  $b$ . Вершини  $a$  та  $b$  називаються *кінцями маршруту*.

Якщо усі ребра маршруту є дугами ( $e_{j_k} = (v_{i_{k-1}}, v_{i_k})$ ,  $j = 1, \dots, n$ ), то маршрут називається *шляхом*, який з'єднує вершину  $a$  з вершиною  $b$ .

Маршрут називається *простим*, якщо вершини у ньому не повторюються (крім, можливо першої та останньої).

Якщо усі ребра у маршруті  $M(a, b)$  є різними, то маршрут називається *ланцюгом*, який з'єднує вершини  $a$  та  $b$  і позначається  $L(a, b)$ .

Замкнений ланцюг називається *циклом*. Орієнтований цикл називається *контуром*.

Наприклад,  $M(a, b) = b4a1a3b9c8b10e10b4a$  — маршрут у графі, діаграма якого наведена на рис. 5.3. Цей маршрут не є ланцюгом, оскільки ребра 4 та 10 повторюються.  $L(a, b) = b4a2a3b9c6a$  — ланцюг, який з'єднує вершини  $a$  та  $b$ . Цей ланцюг не є простим, оскільки вершини повторюються.  $a5c7b4a$  — приклад простого циклу,  $a5c8b$  — простий шлях.

**Теорема 5.3.** Довільний маршрут, який з'єднує у графі вершини  $a$  та  $b$ , містить у собі простий ланцюг.

**Теорема 5.4.** Довільний цикл містить у собі простий цикл. Довільний цикл непарної довжини містить у собі простий цикл непарної довжини.

**Теорема 5.5 (Кеніга).** Звичайний граф дводольний тоді і тільки тоді, коли для нього не має простих циклів непарної довжини.

### 5.2.1. Метричні характеристики графів

*Відстань між вершинами  $u$  та  $v$*  неорієнтованого графа — це довжина найкоротшого ланцюга, який з'єднує ці вершини. Відстань позначається  $d(u, v)$ .

Найкоротша *геодезична* між двома вершинами — найкоротший ланцюг, який їх з'єднує.

*Діаметр графа* — максимальна відстань між вершинами:

$$D(G) = \max \{d(u, v) \mid u, v \in V\}.$$

*Ексцентриситет*  $\varepsilon(v)$  вершини  $v$  — відстань до найбільш віддаленої від неї вершини:

$$\varepsilon(v) = \max \{d(v, u) \mid u \in V\}.$$

*Радіус графа* — мінімальний ексцентриситет:

$$R(G) = \min \{\varepsilon(v) \mid v \in V\}.$$

*Центр графа* — множина вершин графа, які мають мінімальний ексцентриситет:

$$C(G) = \{v \mid \varepsilon(v) = R(G)\}.$$

**Приклад. 5.5.** Вказати ексцентриситети вершин, центр, радіус та діаметр графа, діаграма якого наведена на рис. 5.15.

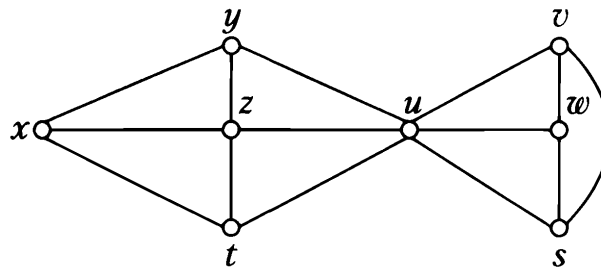


Рис. 5.15.

$$\varepsilon(x) = \varepsilon(v) = \varepsilon(w) = \varepsilon(s) = 3, \quad \varepsilon(y) = \varepsilon(z) = \varepsilon(t) = \varepsilon(u) = 2.$$

$$D(G) = 3, \quad R(G) = 2, \quad C(G) = \{y, z, t, u\}.$$

### 5.3. Зв'язність. Компоненти зв'язності

У параграфах 5.3.1 та 5.3.2 розглядаються лише неорієнтовані графи.

#### 5.3.1. Відношення зв'язності у неорієнтованому графі.

Вершини  $a, b \in V$  неорієнтованого графа  $G = (V, E)$  називаються *зв'язними*, якщо можна побудувати ланцюг, який їх з'єднує. При цьому вершина вважається зв'язною із самою собою.

Відношення зв'язності є відношенням еквівалентності на множині вершин. *Компоненти зв'язності* графа — класи еквівалентності за відношенням зв'язності.

Кількість компонент зв'язності позначається  $\kappa(G)$ .

Наприклад, граф  $G$ , зображений на рис. 5.16, має три компоненти зв'язності ( $G_1$ ,  $G_2$  та  $G_3$ ).

*Зв'язний граф* — це граф, усі вершини якого зв'язні між собою ( $\kappa(G) = 1$ ).

### 5.3.2. Класифікація ребер та вершин графів з точки зору зв'язності

*Міст* — ребро графа, видалення якого збільшує кількість компонент зв'язності. Усі інші ребра — *циклові*.

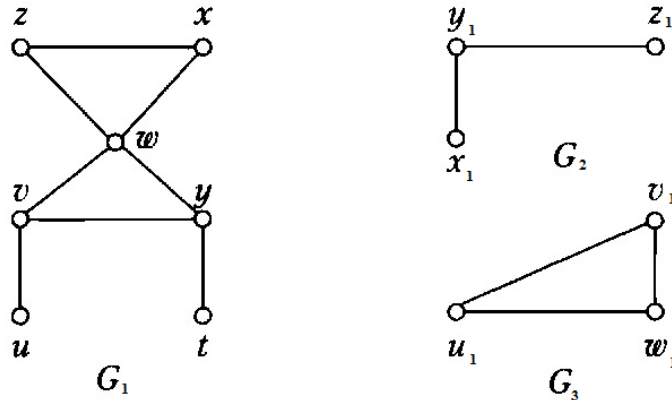


Рис. 5.16. Граф із трьома компонентами зв'язності

*Точка зчеплення* (шарнір) — вершина графа, видалення якої збільшує кількість компонент зв'язності.

**Приклад 5.6.** На рис. 5.17 ребро  $x$  — міст, усі інші ребра — циклові. Вершини  $u$  та  $v$  — точки зчеплення.

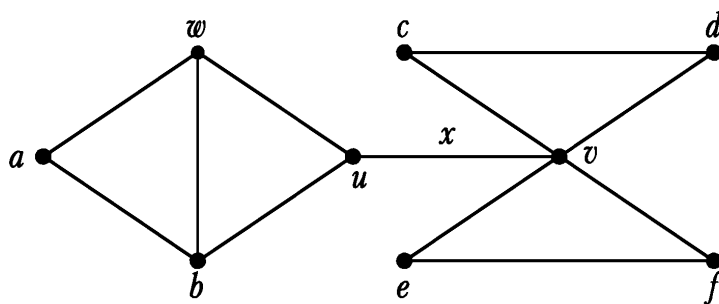


Рис. 5.17. Ребра та мости

**Теорема 5.6.** Кожний простий цикл містить циклове ребро.

**Теорема 5.7.** Кількість ребер звичайного графа задовольняє нерівність

$$|V| - \kappa(G) \leq |E| \leq (|V| - \kappa(G))(|V| - \kappa(G) + 1) / 2.$$

**Наслідок.** Якщо  $|E| > (|V| - 1)(|V| - 2) / 2$ , то граф  $G = (V, E)$  — зв'язний.

### 5.3.3. Зв'язність у орієнтованих графах

Вершина  $b$  називається *досяжною* із вершини  $a$  орієнтованого графа  $G$ , якщо існує шлях, який з'єднує  $a$  з  $b$ .

Відношення досяжності не є відношенням еквівалентності.

Вершини  $a, b \in V$  орієнтованого графу  $G = (V, E)$  називаються *сильно зв'язними*, якщо вершина  $a$  є досяжною із  $b$  і навпаки.

Відношення сильної зв'язності є відношенням еквівалентності на множині вершин. На рис. 5.18 зображено *компоненти сильної зв'язності*.

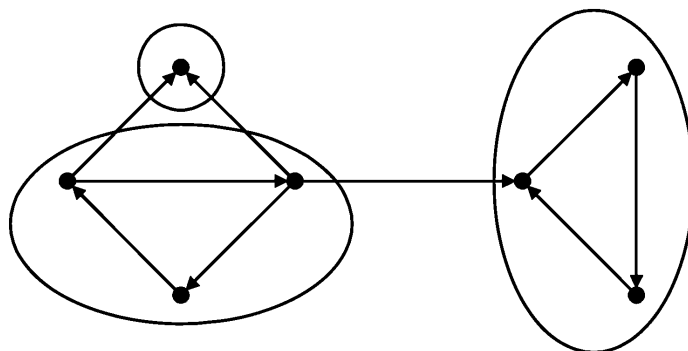


Рис. 5.18. Компоненти сильної зв'язності

### 5.3.4. Обхід графів

*Обхід графів* — систематичний перелік вершин графа. У більшості випадків вказується початкова вершина, з якої треба починати обхід.

#### Алгоритм обходу вершин графа

Вважаємо, що на початку усі вершини не відмічені.

Заносимо у список  $L$  початкову вершину.

while  $L \neq \emptyset$

{

    видаляємо із списку  $L$  першу вершину  $v$ ;

    позначаємо вершину  $v$  як відвідану;

    додаємо до  $L$  усі не відвідані вершини, у які можна потрапити із  $v$ ;

}

Якщо у алгоритмі обходу нові вершини додаються у початок списку, то такий обхід називається *пошуком у глибину*.

Якщо у алгоритмі обходу нові вершини додаються у кінець списку, то такий обхід називається *пошуком у ширину*.

На рис. 5.19 продемонстровано обхід графа із застосуванням пошуку у глибину та ширину, починаючи із вершини  $b$ :

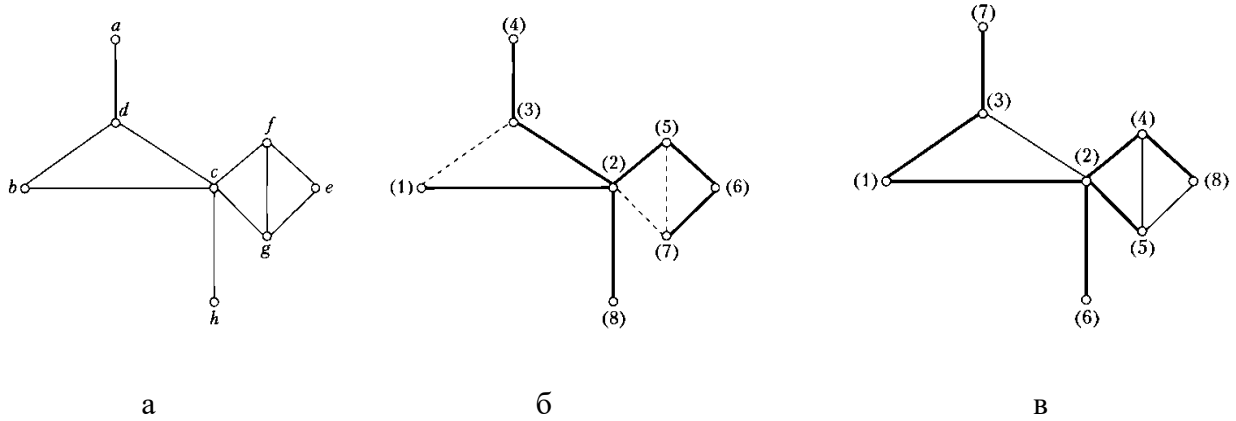


Рис. 5.19. Обхід графа у глибину (б) та ширину (в)

**Приклад 5.7.** Провести обхід вершин звичайного графа, діаграма якого наведена на рис. 5.20, починаючи із вершини  $a$  (попередньо позначити усі інші вершини).

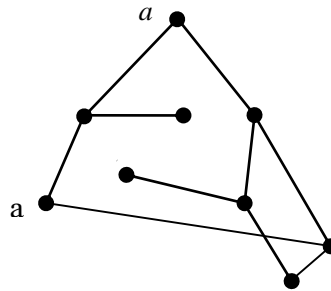


Рис. 5.20.

## 5.4. Ейлерові та гамільтонові графи

У наступних трьох параграфах усі графи (якщо це спеціально не обумовлено в умові) вважаються неорієнтованими.

### 5.4.1. Графи Ейлера

*Граф Ейлера* (ейлерів граф) — це зв'язний граф, для якого існує цикл, який містить усі ребра.

Прикладом графа Ейлера є граф  $G_1$  на рис. 5.21. Графи  $G_2$  та  $G_3$  не є графами Ейлера.

**Теорема 5.8.** Зв'язний граф є ейлеровим тоді і тільки тоді, коли степені усіх вершин є парними.

**Наслідок.** Зв'язний граф є ейлеровим тоді і тільки тоді, коли множину його ребер можна розбити на цикли, що не перетинаються.

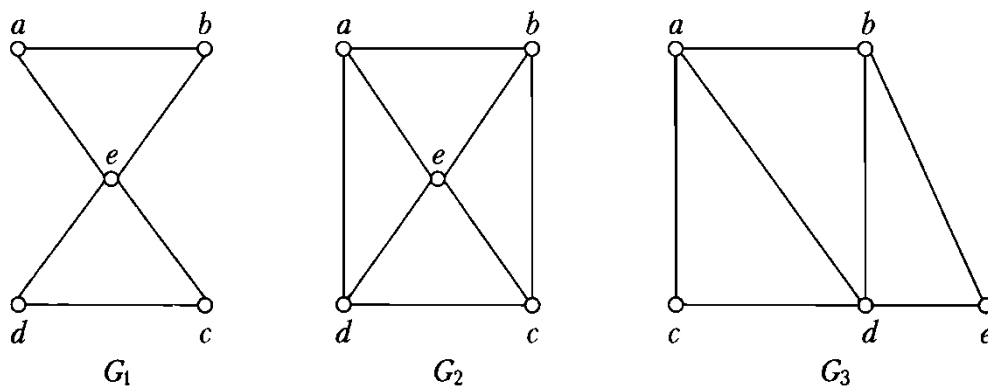


Рис. 5.21.

Ланцюг, який містить усі ребра графа називається *ланцюгом Ейлера*. Граф, для якого існує ланцюг Ейлера називається *напівейлеровим графом*.

Прикладом напівейлерового графа є граф  $G_3$ .

**Теорема 5.9.** Зв'язний граф є напівейлеровим тоді і тільки тоді, коли не більше двох його вершин мають непарну степінь.

На рис. 5.22 зображено граф із задачі про кенігсберзькі мости. Оскільки степені усіх чотирьох вершин непарні, то він не є ні ейлеровим, ні напівейлеровим.

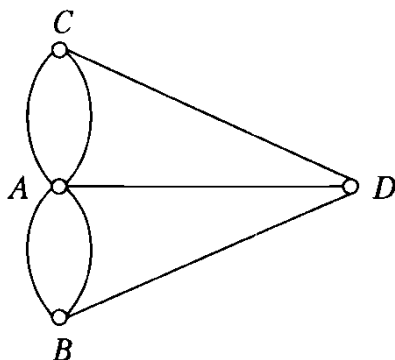


Рис. 5.22. Граф для задачі про кенігсберзькі мости

**Алгоритм Флері** знаходження ейлерового ланцюга:

- 1) Починаємо з вершини з непарною степеню (або з довільної вершини у випадку парності степенів усіх вершин).
- 2) стираємо (викреслюємо) пройдені ребра та ізольовані вершини, які виникають в процесі руху.
- 3) на кожному кроці вибираємо міст в якості наступного ребра тільки тоді, коли немає циклових ребер, інцидентних поточній вершині.

### 5.4.2. Гамільтонові графи.

Зв'язний граф називається *гамільтоновим*, якщо існує цикл, який проходить через кожну вершину графа рівно один раз.

Прикладом графа Гамільтона є граф, наведений на рис 5.23 (на діаграмі ребра циклу виділені товстішими лініями).

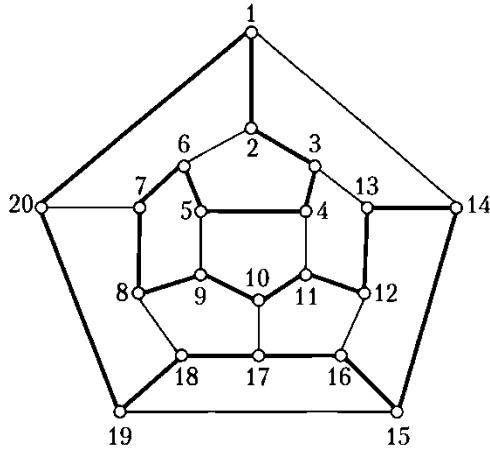


Рис. 5.23. Гамільтонів шлях

**Теорема 5.23 (теорема Дірака).** Якщо у графі  $G = (V, E)$  з  $n \geq 3$  вершинами степінь кожної вершини не менша за  $n/2$ , то граф  $G$  є гамільтоновим.

### 5.5. Планарні графи

Граф допускає *вкладення* (вкладається) на деякій поверхні, якщо його можна зобразити на цій поверхні так, щоб ребра графа не перетиналися.

Граф називається *планарним*, якщо його можна вкласти на площину.

На рис. 5.24 зображено діаграму планарного графа  $K_4$  та його вкладення.

**Теорема 5.11.** Граф  $G$  вкладається на сфері тоді і тільки тоді, коли він є планарним.

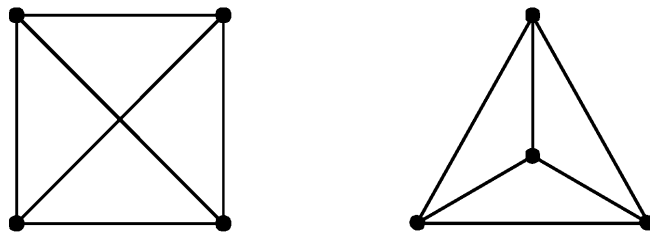


Рис. 5.24. Вкладення графа  $K_4$

Частина площина, обмежена ребрами планарного графа, називається *гранню*. Множина граней планарного графа позначається  $F$ . У цю множину включаться також і зовнішня частина площини. Для графа, зображеного на рис. 5.25,  $F = \{r_1, r_2, r_3, r_4\}$ .

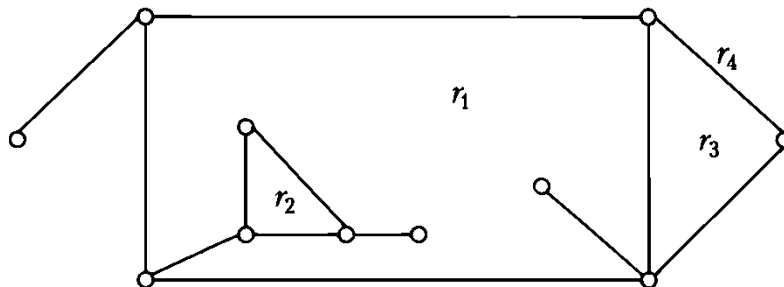


Рис 5.25. Грані планарного графа

**Теорема 5.12 (формула Ейлера).** Для планарного графа

$$|V| - |E| + |F| = \kappa(G) + 1.$$

**Наслідок 1.** Для довільного випуклого многогранника

$$B + \Gamma - P = 2,$$

де  $B$  — кількість вершин многогранника,  $\Gamma$  — кількість граней,  $P$  — кількість ребер.

**Наслідок 2.** У будь-якому планарному графі без кратних ребер та петель  $|E| \leq 3|V| - 6$ .

**Наслідок 3.** Графи  $K_5$  та  $K_{3,3}$  не є планарними.

**Наслідок 4.** У будь-якому планарному графі без кратних ребер та петель існує вершина, степінь якої не більша за 5.

## 5.6. Дерева

### 5.6.1. Ліс

Нехай  $G = (V, E)$  — заданий неорієнтований граф. Величина

$$\lambda(G) = |E| - |V| + \kappa(G).$$

називається *цикломатичним* числом графа  $G$ .

**Теорема 5.13.** Для довільного графа  $G$

$$\lambda(G) \geq 0 \text{ і } \lambda(G) = 0 \Leftrightarrow \text{граф } G \text{ не містить циклів.}$$

*Ліс* — це граф без циклів. Приклад лісу зображено на рис. 5.26.



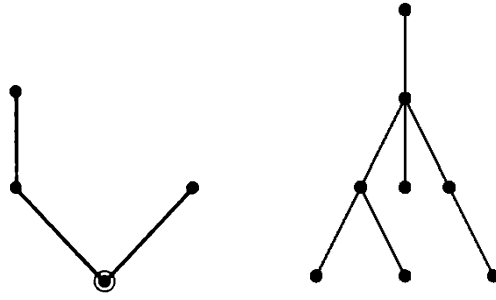


Рис. 5.26. Ліс

**Теорема 5.14.** Наступні твердження про граф  $G$  є еквівалентними:

- 1)  $G$  — ліс;
- 2)  $G$  не містить простих циклів;
- 3) Всі ланцюги в  $G$  — прості;
- 4)  $\lambda(G) = 0$ .

### 5.6.2. Неорієнтовані дерева

*Дерево* — зв'язний граф без циклів. Приклад дерева зображено на рис. 5.27.

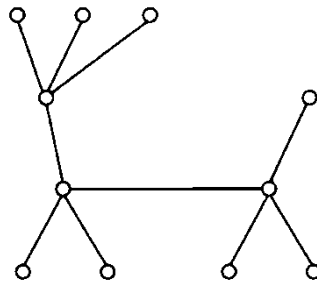


Рис. 5.27. Дерево

**Теорема 5.15.** Наступні твердження про граф  $G$  є еквівалентними:

- 1)  $G$  — дерево;
- 2) Будь-які дві вершини в  $G$  з'єднані рівно одним простим ланцюгом;
- 3)  $G$  — зв'язний граф, кожне ребро якого — міст;
- 4)  $|V| = |E| + 1$  і  $\kappa(G) = 1$ ;
- 5)  $\lambda(G) = 0$ , але після додавання довільного нового ребра  $\lambda(G) = 1$ .

**Наслідок.** У нетривіальному дереві є принаймні дві висячі вершини.

### 5.6.3. Нумеровані дерева

Для компактного подання нумерованих дерев використовується *код Прюфера*, який для дерева з  $n$  вершинами містить  $n - 2$  числа.

При побудові коду на кожному кроці видаляється висяча вершина із найменшим номером і номер вершини, з якою вона була пов'язана, дописується у кінець коду.

Код Прюфера для дерева на рис. 39: 7, 9, 1, 7, 2, 2, 7, 1, 2, 5

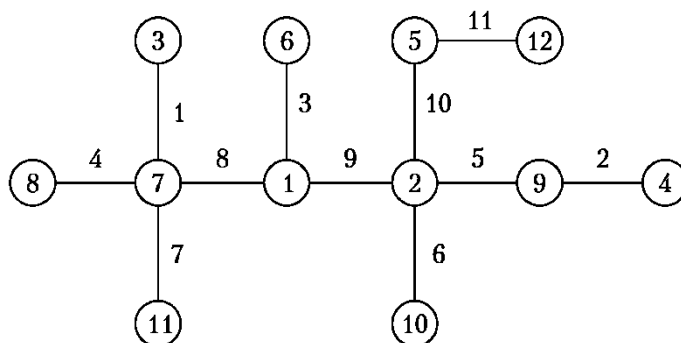


Рис. 5.28. Приклад побудови коду Прюфера для нумерованого дерева

При декодуванні на  $i$ -му ( $i = 1, 2, \dots, n - 2$ ) кроці із списку ще невикористаних вершин вибирається (без повернення) вершина з найменшим номером, який не зустрічається у ще не обробленій частині коду Прюфера. Ця вершина з'єднується з  $i$ -ою вершиною у кодї Прюфера. У кінці з'єднуються останні дві невикористані вершини.

З існування та унікальності коду Прюфера випливає наступна теорема:

**Теорема 5.16 (теорема Келї).** Кількість нумерованих дерев із  $n$  вершинами рівна  $n^{n-2}$ .

### 5.6.4. Кореневі дерева.

У *корневих деревах* у множині вершин виділяється *корінь*. Орієнтація вершин корневих дерев відбувається у напрямку від кореня. Якщо вершини  $v$  та  $u$  — суміжні і відстань від  $v$  до кореня дерева більша за відстань від  $u$  до кореня, то вершина  $v$  називається *дочірньою* вершиною для  $u$ , а вершина  $u$  — *батьківською* для  $v$ . Тому кореневі дерева можна вважати орієнтованими графами.

**Теорема 5.17.** Кожна вершина крім кореня має рівно одну батьківську вершину.

*Листи* — це вершини кореневого дерева, які не мають дочірніх вершин.

Множина вершин, які розташовані на однаковій відстані від кореня називається *ярусом* дерева.

Дерева б)-в) на рис. 5.29 — кореневі дерева з коренем  $a$  та  $c$  відповідно, які отримуються із звичайного дерева а). На рис. 5.29 в)  $a, e$  — вершини 1-го ярусу,  $b, d$  — 2-го,  $f, g$  — 3-го.

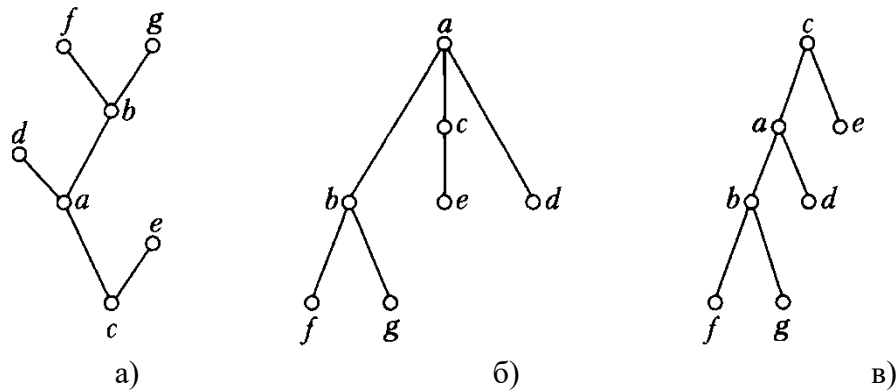


Рис. 5.29. Приклад корневих дерев з коренями  $a$  та  $c$  відповідно

### 5.6.5. Орієнтовані дерева

Поняття *орієнтованого дерева* є синонімом до поняття кореневого дерева. Означення орієнтованого дерева:

- 1) Існує єдина вершина  $r$  (корінь), для якої  $\deg^+(r) = 0$ .
- 2) Для всіх інших вершин  $v \in T$   $\deg^+(v) = 1$ .
- 3) Кожна вершина досяжна із кореня.

**Зауваження.** При зображенні орієнтованих дерев вважають, що дуги спрямовані зверху вниз. Тому на діаграмах часто не зображають стрілки.

Еквівалентне означення ордерера  $T$  з використанням піддерев.

- 1) Є єдиний елемент  $r$  — корінь.
- 2) Усі інші вершини містяться у  $k$  ( $k \geq 0$ ) підблоках, які називаються піддеревами.

$$T = (r, \{T_1, \dots, T_k\}).$$

Для упорядкованих дерев також вказується відносний порядок піддерев  $T_1, \dots, T_k$ .

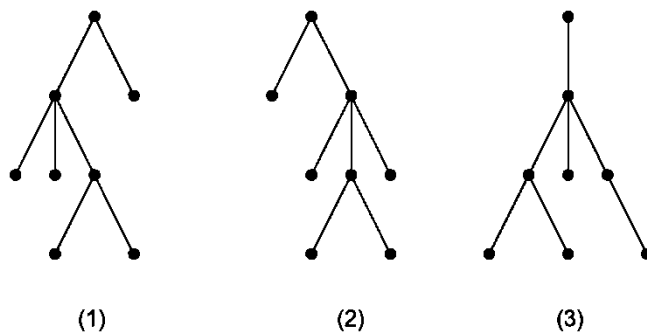


Рис. 5.30. Діаграми орієнтованих дерев

Наведені на рис. 5.30 дерева є ізоморфними як звичайні дерева та не ізоморфними, як впорядковані дерева. Як орієнтовані дерева (1) та (2) ізоморфні, але (2) та (3) та (1) та (3) не є ізоморфними.

### 5.6.6. Бінарні дерева

Означення *бінарного дерева*:

- 1) Є одна вершина — *корінь дерева*.
- 2) Усі інші вершини належать одному із піддерев (лівому чи правому), які не перетинаються.

За допомогою бінарних дерев можна зобразити довільне упорядковане дерево. При переході до бінарних дерев для кожної вершини ліве ребро з'єднує її із старшим сином (у початковому дереві), праве ребро — із наступним (молодшим) братом у початковому дереві. На рис. 5.31 наведено упорядковане дерево а) та відповідне йому бінарне дерево б).

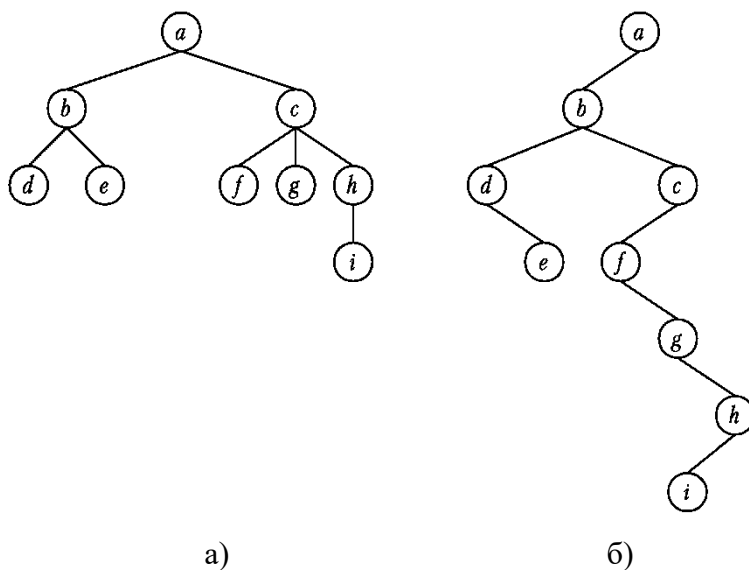


Рис. 5.31. Зображення упорядкованого дерева а) за допомогою бінарного б)

Бінарне дерево не є окремим випадком упорядкованого дерева. Наведені на рис. 5.32 бінарні дерева не є ізоморфними, хоча як упорядковані дерева вони ізоморфні. Справа у тому, що корінь має лише піддерево, і тому дерева на обох діаграмах не відрізняються, якщо розглядати їх як упорядковані.



Рис. 5.32. Зображення різних бінарних дерев

**Теорема 5.18.** Кількість різних бінарних дерев із  $n$  вершинами рівна  $\frac{1}{n+1} C_{2n}^n$ .

**Обходи бінарних дерев:**

- *прямий* (префіксний): корінь, ліве піддерево, праве піддерево;
- *внутрішній* (інфіксний, симетричний): ліве піддерево, корінь, праве піддерево;
- *зворотний* (постфіксний, кінцевий): ліве піддерево, праве піддерево, корінь.

Проведемо обходи вершин дерева, наведеного на рис. 5.33.

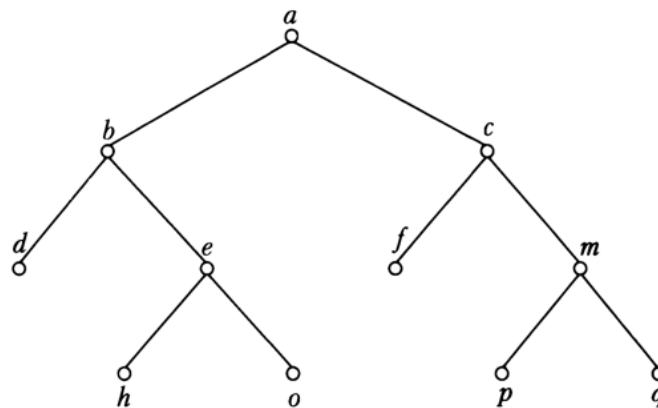


Рис. 5.33. Обхід бінарного дерева

Прямий:  $a, b, d, e, h, o, c, f, m, p, q$ .

Внутрішній:  $d, b, h, e, o, a, f, c, p, m, q$ .

Зворотний:  $d, h, o, e, b, f, p, q, m, c, a$ .

Дерева арифметичних виразів (листи відповідають числам або змінним, внутрішні вершини — операціям). Запис виразу з використанням прямого обхо-

ду вершин називається *польською нотацією* (префіксним записом), запис з використанням внутрішнього обходу — *інфіксним записом*, запис з використанням зворотного обходу вершин — *оберненою польською нотацією* (ОПН) виразу.

**Приклад 5.8.** Записати вираз  $(a + b / c) * (d - e * f)$  у префіксній, інфіксній формах та ОПН.

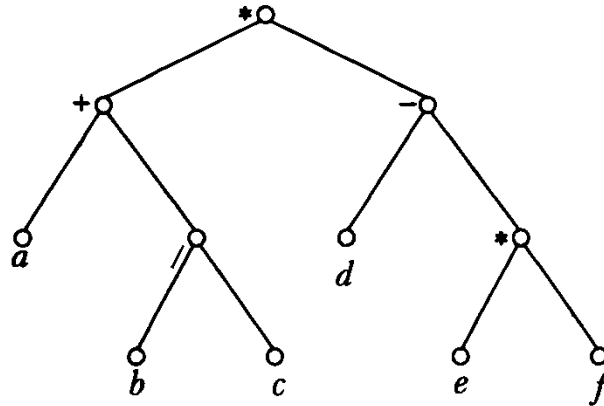


Рис. 5.34. Дерево арифметичного виразу

Дерево виразу наведено на рис. 5.34. З трьох використанням вищенаведених обходів отримуємо наступні записи виразу:

префіксний:  $* + / b c - d * e f$  ;

інфіксний:  $a + b / c * d - e * f$  ;

ОПН:  $abc / + d e f * - *$  .

Перевагою префіксного запису та ОПН є те, що вони не вимагають дужок для однозначного відтворення виразу. Інфіксний запис неоднозначний і тому потребує дужок.

#### Алгоритм Дейкстри побудови ОПН:

1) Поки вхідний рядок не закінчився:

Читаємо черговий символ.

Якщо символ є числом, то

додаємо його у вихідний рядок.

Якщо символ є відкриваючою дужкою, то

заносимо його у стек.

Якщо символ є закриваючою дужкою, то

до тих пір, поки верхнім елементом стека не буде відкриваюча дужка, виштовхуємо елементи зі стеку і вихідний рядок. При цьому відкриваюча дужка видаляється зі стеку.

Якщо на верхівці стеку виявився знак (символ) функції, то

виштовхуємо його зі стеку.

Якщо відкриваюча дужка не знайдена,

то початковий вираз не є коректним.

Якщо символ виявився оператором  $o_1$ , то

якщо на верхівці стеку знаходиться оператор  $o_2$ , який має пріоритет  $\geq$  за  $o_1$ , то

виштовхуємо  $o_2$  зі стеку.

заносимо  $o_1$  у стек.

- 2) якщо стек непорожній, то виштовхуємо його вміст у кінець вихідного рядка.

Обчислення виразів, записаних у ОПН також засновано на використанні стека.

### Алгоритм:

- 1) Обробка вхідного символу:
  - Якщо вхідний символ є операндом, то занести його у стек.
  - Якщо вхідним є символ операції, то відповідна операція виконується над потрібною кількістю операндів, які виймаються зі стеку. Результат операції заноситься у верхівку стека.
- 2) Якщо вхідний рядок оброблений не до кінця, перейти до кроку 1.
- 3) Вибрати кінцевий результат обчислень зі стеку.

## 5.7. Задачі до п'ятого розділу

1. Зобразити діаграму графа, який заданий за допомогою:

а) матриці інцидентності

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

б) матриці сусідства

$$A = \begin{pmatrix} \beta & 2\beta + \alpha & \beta + \alpha & 0 & 0 & \beta \\ \alpha + \beta & 0 & 0 & 0 & 3\alpha & 0 \\ \alpha & 0 & 2\beta & 0 & 0 & 0 \\ 2\beta & \beta & \beta & 0 & \beta & 2\alpha \\ 0 & 3\alpha & 0 & 0 & 0 & \alpha \\ 0 & 0 & 0 & 2\alpha & \alpha & 2\beta \end{pmatrix}$$

2. Діаграма графа  $G$  наведена на рис. 5.35,  $G'$  — відповідний йому неорієнтований граф. Позначити вершини та ребра та виконати наступні завдання:

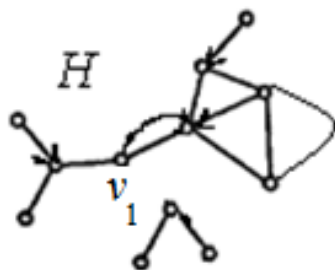


Рис. 5.35

- а) Вказати матриці інцидентності та суміжності.
  - б) Знайти степені усіх вершин підграфа  $H$ .
  - в) Вказати компоненти слабкої та сильної зв'язності та визначити цикломатичне число графа  $G'$ .
  - г) Знайти діаметр, радіус та центр графа неорієнтованого графа  $H'$ , який відповідає підграфу  $H$ .
  - д) Вказати усі точки зчеплення та мости графа  $G'$ .
  - е) вказати компоненти графа  $G'$ , які є ейлеревими та гамільтоновими.
3. Чи існує ейлерів шлях у графах, зображених на рис. 5.36.

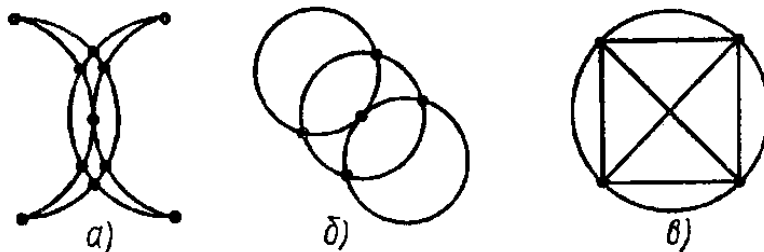


Рис. 5.36

Якщо так, то вказати його.

4. Діаграма графа наведена на рис. 5.37. Виконати наступні завдання:
- а) визначити ексцентриситети вершин, діаметр, радіус і центр графа;
  - б) вказати усі геодезичні між вершинами  $c$  та  $h$ ;
  - в) перевірити, чи є граф ейлеревим, напівейлеровим, гамільтоновим або планарним;
  - г) провести обхід вершин з використанням пошуку у глибину та ширину, починаючи з вершини  $e$ .



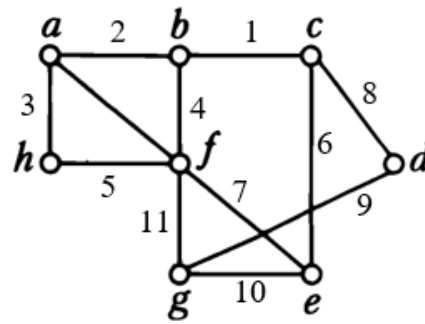


Рис. 5.37

5. Для впорядкованого дерева, діаграма якого зображена на рис. 5.38, виконати наступні завдання:
- вказати код Прюфера;
  - знайти центр дерева;
  - провести обхід вершин, починаючи із вершини 1;
  - зобразити кореневе дерево з коренем у вершині 1;
  - зобразити впорядковане дерево за допомогою бінарного дерева.

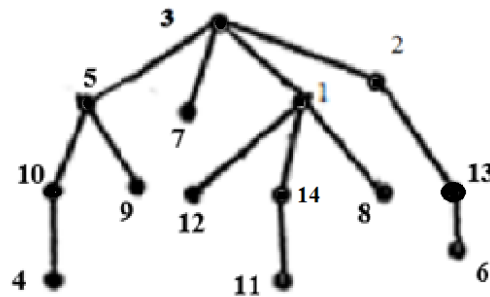


Рис. 5.38

- Зобразити дерево із коренем 5, яке має наступний код Прюфера: 4, 1, 2, 1, 1, 9, 1, 3, 4.
- Записати у префіксній та постфіксній формі наступний вираз
 
$$((a+1)(b+c)-2)*(3-b)/(b-a*c)+4.$$
- Обчислити значення виразу  $1, 3, 2, ^, 6, -, +, 2, 3, ^, 5, -, *, 3, 4, *, -$  та записати його у звичайному (інфіксному) вигляді.
- Записати усі неізоморфні графи 4-го порядку, які є лісом.
- Вказати усі неізоморфні звичайні графи, у яких:
  - Одна вершина має степінь 2, дві — 3, дві — 4.
  - Три вершини мають степінь 2, дві — 3, одна — 4.

11. Довести, що серед довільних 6 людей є 3 попарно знайомі або три 3 попарно незнайомі.

12. Серед пар графів, зображених на рис. 5.39 та 5.40, вказати пари ізоморфних.

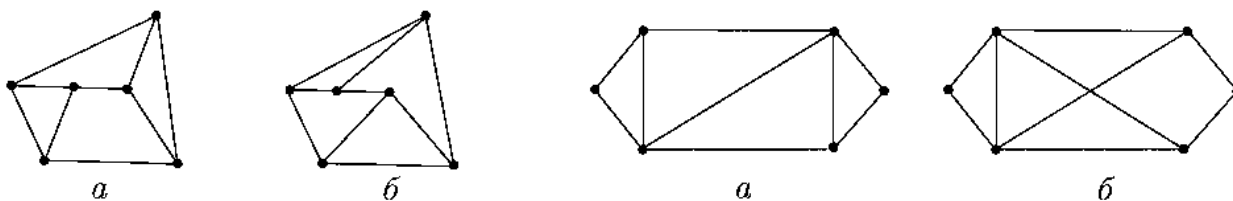


Рис. 5.39

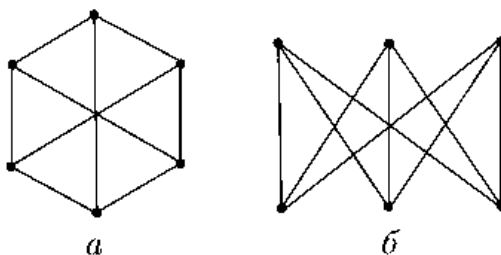


Рис. 5.40.

13. Для графа, зображеного на рис. 12, знайти цикли з:

- а) 4 ребер;
- б) 5 ребер;
- в) 11 ребер.

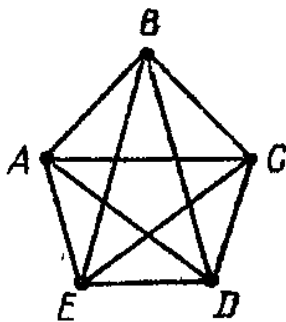


Рис. 5.41

**РЕКОМЕНДОВАНА ЛІТЕРАТУРА**

1. Андерсон Д. Дискретная математика и комбинаторика. СПб.: Вильямс, 2003. 958 с.
2. Андрійчук В. І., Комарницький М. Я., Іщук Ю. Б. Вступ до дискретної математики. — К.: Центр навчальної літератури, 2004. — 254 с.
3. Бардачов Ю. М., Соколова Н. А., Ходаков В. Є. Дискретна математика. — К.: Вища школа, 2002. — 287 с.
4. Бондаренко М.Ф., Білоус Н.В., Руткас А.Г. Комп'ютерна дискретна математика. — Харків: "Компанія Сміт", 2004. — 480 с.
5. Вітенько І. В. Математична логіка: Курс лекцій. — Ужгород: УжДУ, 1971. — 224 с.
6. Гаврилов Г. П., Сапоженко А. А. Задачи и упражнения по дискретной математике: Учеб. пособие. — 3-е изд., перераб. — М.: ФИЗМАТЛИТ, 2005. — 416 с.
7. Дрозд Ю. Дискретна математика. — К.: Київський університет імені Т. Шевченка, 2004. — 70 с.
8. Капітонова Ю. В., Кривий С. Л., Летичевський О. А., Луцький Г. М. Основи дискретної математики. — К.: Наукова думка, 2002. — 580 с.
9. Комплект конкурсних завдань до університетських студентських олімпіад зі спеціальності 122 «Комп'ютерні науки» / Розробник: В. М. Коцовський. — Ужгород: Видавництво УжНУ "Говерла", 2017. — 13 с.
10. Коцовський В. М. Дискретна математика та теорія алгоритмів. Частина І: Конспект лекцій для студентів спеціальностей: 6.122 — "Комп'ютерні науки", 6.121 — "Інженерія програмного забезпечення" / В. М. Коцовський. — Ужгород: Видавництво УжНУ "Говерла", 2019. — 52 с.
11. Коцовський В. М. Дискретна математика та теорія алгоритмів. Частина ІІ: Конспект лекцій для студентів спеціальностей: 6.122 — "Комп'ютерні науки", 6.121 — "Інженерія програмного забезпечення" / В. М. Коцовський. — Ужгород: Видавництво УжНУ "Говерла", 2019. — 52 с.
12. Коцовський В. М. Дискретна математика та теорія алгоритмів: Методичні матеріали до практичних робіт / В. М. Коцовський. — Ужгород: Видавництво УжНУ "Говерла", 2019. — 35 с.
13. Коцовський В. М. Методи та системи штучного інтелекту: методичний посібник для студентів напрямів підготовки: 6.050101 — «Комп'ютерні науки», 6.050103 — «Програмна інженерія» факультету інформаційних технологій УжНУ / В. М. Коцовський. — Ужгород: Видавництво УжНУ «Говерла», 2017. — 56 с.

14. Коцовський В. М. Теорія паралельних обчислень. Частина I: Методичний посібник / В. М. Коцовський. — Ужгород: Видавництво УжНУ "Говерла", 2019. — 51 с.
15. Коцовський В. М. Теорія паралельних обчислень. Частина II: Методичний посібник / В. М. Коцовський. — Ужгород: Видавництво УжНУ "Говерла", 2019. — 52 с.
16. Коцовський В. М. Технології розподілених систем та паралельних обчислень. Частина I: Методичний посібник — Ужгород: Видавництво УжНУ "Говерла", 2017. — 51 с.
17. Коцовський В. М. Технології розподілених систем та паралельних обчислень. Частина II: Методичний посібник. — Ужгород: Видавництво УжНУ "Говерла", 2017. — 76 с.
18. Кузнецов О. П., Адельсон-Вельский Г. М. Дискретная математика для инженера. М.: Энергоатомиздат, 1988. — 480 с.
19. Лавров И. А., Максимова Л. Л. Задачи по теории множеств, математической логике и теории алгоритмов. — 5-е изд., исправл. — М.: ФИЗМАТЛИТ, 2004. — 256 с.
20. Латонин Л. А., Макаренков Ю. А., Николаева В. В., Столяр А. А. Математическая логика: Учеб. пособие. — Мн.: Выш. шк., 1991. — 269 с.
21. Нефедов В. Н., Осипова В. А. Курс дискретной математики. — М.: Изд-во МАИ, 1992. — 264 с.
22. Нікольський Ю. В., Пасічник В. В., Щербина Ю. М. Дискретна математика. — К.: Видавнича група ВНУ, 2007. — 368 с.
23. Новиков Ф. А. Дискретная математика: Учебник для вузов. 2-е изд. Стандарт третьего поколения. — СПб.: Питер, 2013. — 432 с.
24. Цейтлін Г. Є. Елементи теорії булевих функцій. — К: Техніка, 1973. — 76 с.
25. Яблонский С. В., Лупанов О. Б. Дискретная математика и математические вопросы кибернетики. — М.: Наука, 1974. — 312 с.
26. Яблонский С. В. Введение в дискретную математику: Учеб. пособие для вузов / Под ред. В. А. Садовниченко. — 4-е изд., стер. — М.: Высшая школа, 2003. — 384 с.
27. Ядренко М. Й., Оленко А. Я. Дискретна математика. навчально-методичний посібник. — К.: Київський університет ім. Т. Шевченка, 1995. — 83 с.