

## Лекція 13

### Тема 4. Графи та їхні властивості

Теорія графів – одна з істотних частин математичного апарату інформатики та кібернетики. У термінах теорії графів можна сформулювати багато задач, пов'язаних із дискретними об'єктами. Такі задачі виникають у проектуванні інтегральних схем, схем управління, у дослідженні автоматів, в економіці й статистиці, теорії розкладів і дискретній оптимізації.

#### План лекції

##### Основні означення та властивості

##### Деякі спеціальні класи простих графів

##### Способи подання графів

- Матриця інцидентності

- Матриця суміжності

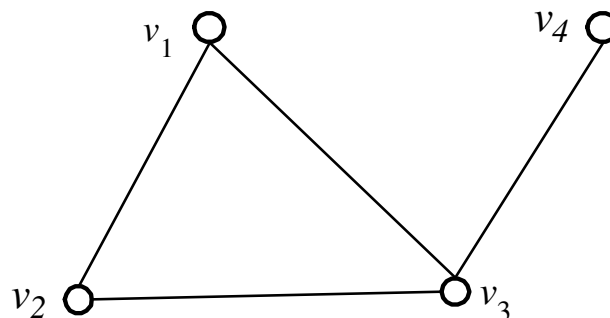
- Подання графа списком пар

- Подання графа списками суміжності

#### Основні означення та властивості

Термін „граф” уперше з'явився в книзі видатного угорського математика Д. Кеніга 1936р., хоча перші задачі теорії графів пов'язані ще з іменем Л. Ейлера (XVIII ст.).

*Простим графом* називають пару  $G=(V, E)$ , де  $V$  – непорожня скінченна множина елементів, названих *вершинами*,  $E$  – множина неупорядкованих пар різних елементів з  $V$ . Елементи множини  $E$  (неупорядковані пари різних вершин) називають *ребрами*.



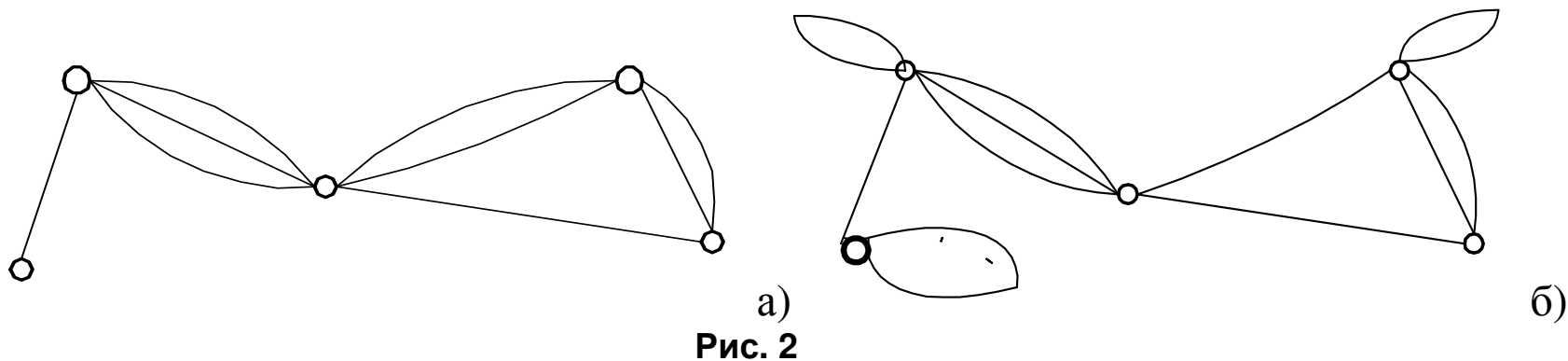
**Рис. 1**

**Приклад.** На рис. 1 зображено простий граф  $G$  з множиною вершин  $V=\{v_1, v_2, v_3, v_4\}$  і множиною ребер  $E=\{\{v_1, v_2\}, \{v_1, v_3\}, \{v_2, v_3\}, \{v_3, v_4\}\}$ .

Говорять, що ребро  $\{u, v\}$  з'єднує вершини  $u$  та  $v$ . Оскільки  $E$  – множина, то в простому графі пару вершин може з'єднувати не більше ніж одне ребро.

Іноді розглядають графи, у яких дві вершини можуть бути з'єднані більше ніж одним ребром. Так виникає поняття мультиграфа. *Мультиграфом* називають пару  $(V, E)$ , де  $V$  – скінченна непорожня множина вершин, а  $E$  – *сім'я* неупорядкованих пар різних елементів з множини  $V$ . Тут застосовано термін „сім'я” замість „множина”, бо елементи в  $E$  (ребра) можуть повторюватись. Ребра, що з'єднують одну й ту саму пару вершин, називають *кратними* (або *паралельними*) ребрами.

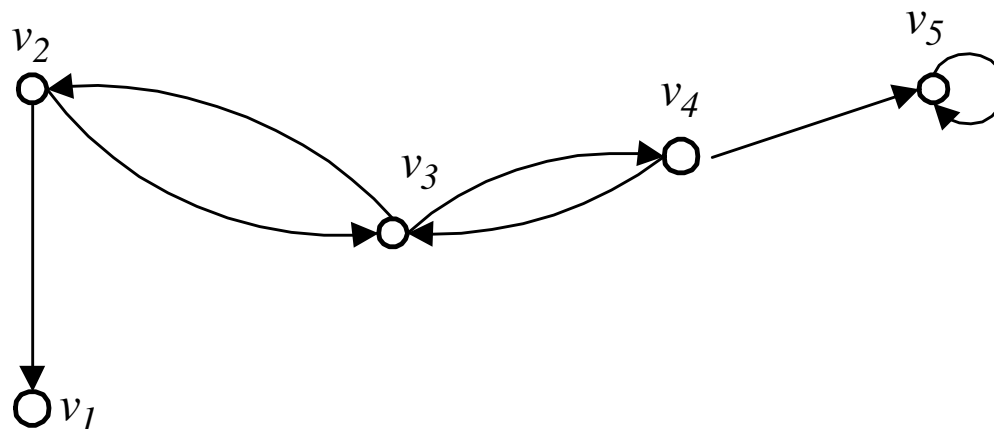
Подальше узагальнення полягає в тому, що окрім кратних ребер розглядають ще й *петлі*, тобто ребра, які з'єднують вершину саму із собою. *Псевдографом* називають пару  $(V, E)$ , де  $V$  – скінченна непорожня множина вершин, а  $E$  – сім'я невпорядкованих пар не обов'язково різних вершин.



**Приклад.** На рис. 2 зображено (а) – мультиграф і (б) – псевдограф.

Розглянуті три типи графів називають *неорієнтованими*. Псевдограф – це найзагальніший тип неорієнтованого графа, бо він може містити петлі й кратні ребра. Мультиграф – це неорієнтований граф, який може містити кратні ребра, але не може містити петель. Нарешті, простий граф – це неорієнтований граф без кратних ребер і без петель.

Розглядають також орієнтовані графи. *Орієнтованим графом* називають пару  $(V, E)$ , де  $V$  – скінченна непорожня множина вершин, а  $E$  – множина впорядкованих пар елементів множини  $V$ . Елементи множини  $E$  в орієнтованому графі називають *дугами* (або *орієнтованими ребрами*). Дугу  $(v, v)$  називають *петлею*.



**Рис. 3**

**Приклад.** На рис. 3 зображено орієнтований граф із множиною вершин  $V = \{v_1, v_2, v_3, v_4, v_5\}$  і множиною дуг  $E = \{(v_2, v_1), (v_2, v_3), (v_3, v_2), (v_3, v_4), (v_4, v_3), (v_4, v_5), (v_5, v_5)\}$ .

Зазначимо, що дуга – це впорядкована пара вершин (записують у круглих дужках), тому в графі на рис. дуги  $(v_2, v_3)$  та  $(v_3, v_2)$  – різні. На рисунках дуги позначають стрілками.

*Орієнтованим мультиграфом* називають пару  $(V, E)$ , де  $V$  – скінченна непорожня множина вершин, а  $E$  – сім'я впорядкованих пар елементів з  $V$ .

Отже, елементи (дуги) в  $E$  в разі орієнтованого мультиграфа можуть повторюватись, такі дуги називають *кратними*. Підкреслимо, що кратні дуги з'єднують одну пару вершин і однаково напрямлені.

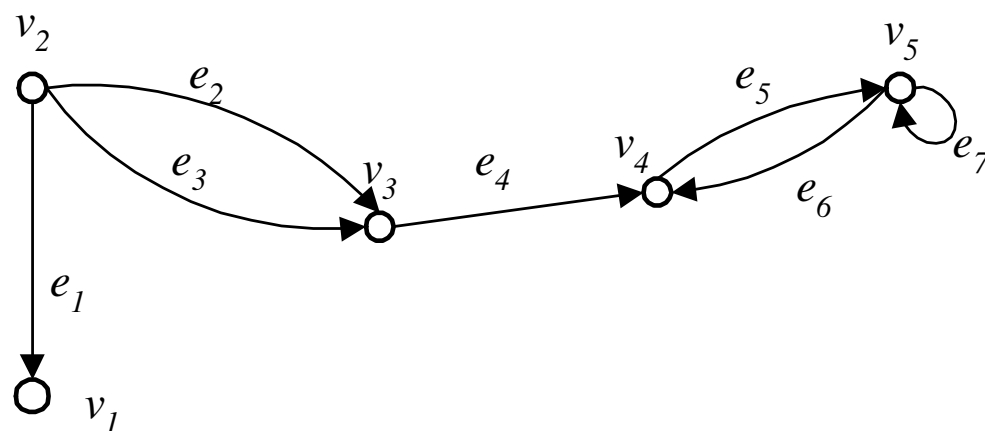


Рис. 4

**Приклад.** На рис. 4 наведено приклад орієнтованого мультиграфа. Дуги  $e_2$  та  $e_3$  – кратні, а дуги  $e_5, e_6$  – ні.

Надалі ми будемо використовувати термін „граф” для опису довільних графів – орієнтованих і неорієнтованих, із петлями та кратними ребрами чи без них. Термін „неорієнтований граф” або „псевдограф” – для довільного неорієнтованого графа, який може мати кратні ребра й петлі. Означення різних типів графів зведено в табл. 1.

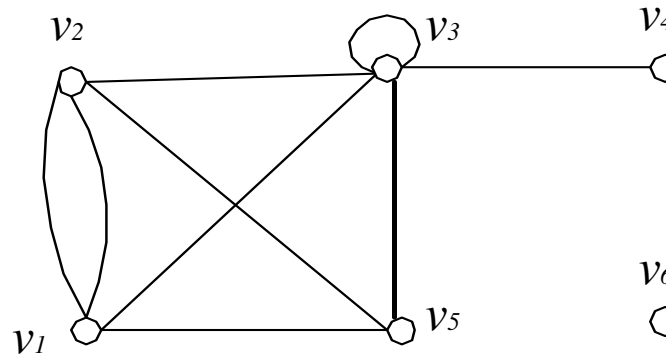
Таблиця 1

Тип графа	Ребра	Кратні ребра дозволені?	Петлі дозволені?
Простий граф	Неорієнтовані	Ні	Ні
Мультиграф	Неорієнтовані	Так	Ні
Псевдограф	Неорієнтовані	Так	Так
Орієнтований граф	Орієнтовані (дуги)	Ні	Так
Орієнтований мультиграф	Орієнтовані (дуги)	Так	Так

Дві вершини  $u$  та  $v$  в неорієнтованому графі  $G$  називають *суміжними*, якщо існує ребро  $\{u, v\}$ , тобто  $\{u, v\} \in E$ . Якщо  $e = \{u, v\}$  – ребро, то вершини  $u$  та  $v$  називають його *кінцями*. Два ребра називають *суміжними*, якщо вони мають спільний кінець. Вершину  $v$  та ребро  $e$  називають *інцидентними*, якщо вершина  $v$  – кінець ребра  $e$ . Зазначимо, що суміжність – це зв’язок між однорідними елементами графа, а інцидентність – зв’язок між його різнорідними елементами.

*Степінь вершини* в неорієнтованому графі – це кількість ребер, інцидентних цій вершині, причому петлю враховують двічі. Степінь вершини  $v$  позначають  $\deg(v)$ . Якщо  $\deg(v)=0$ , то вершину  $v$  називають *ізолюваною*; якщо  $\deg(v)=1$  – *висячою*, або *кінцевою*.

**Приклад.** У неорієнтованому графі на рис. 5 степені вершин такі:  $\deg(v_1)=4$ ,  $\deg(v_2)=4$ ,  $\deg(v_3)=6$ ,  $\deg(v_4)=1$ ,  $\deg(v_5)=3$ ,  $\deg(v_6)=0$ . Отже, вершина  $v_6$  – ізольована, а  $v_4$  – висяча.



**Рис. 5**

Зв'язок між степенями вершин неорієнтованого графа та кількістю його ребер дає така теорема.

**Теорема 1.** Нехай  $G=(V, E)$  неорієнтований граф з  $m$  ребрами. Тоді

$$\sum_{v \in V} \deg(v) = 2m.$$

Зазначимо, що це твердження стосується будь-якого неорієнтованого графа, зокрема, з петлями й кратними ребрами.

**Доведення.** Кожне ребро додає по одиниці до степенів двох вершин, або двійку до степеня однієї вершини у випадку петлі. З цього випливає, що сума степенів вершин удвічі більша від кількості ребер. Теорему доведено.

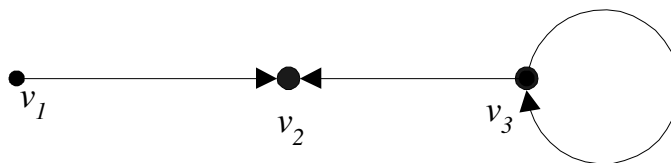
Зазначимо, що теорему називають „теоремою про рукостискання”, бо аналогічно як ребро має два кінці, так і під час рукостискання задіяні дві руки. З цієї теореми випливає, що сума степенів усіх вершин неорієнтованого графа – парне число. Цей простий факт має багато наслідків, один із яких сформульовано в наведеній нижче теоремі.

**Теорема 2.** Неорієнтований граф має парну кількість вершин непарного степеня.

Тепер розглянемо орієнтований мультиграф  $G=(V, E)$ . Якщо  $(u, v) \in E$ , то вершину  $u$  називають *початковою (ініціальною)*, а вершину  $v$  – *кінцевою (термінальною)* вершиною дуги  $e=(u, v)$ . Петля має початок і кінець в одній і тій самій вершині. Вершини орієнтованого графа називають *суміжними*, якщо одна з них – початкова, а інша – кінцева для якоїсь дуги. Дуги називають *суміжними*, якщо вони мають спільну вершину. Вершину  $u$  називають *інцидентною* дузі  $e$ , якщо  $u$  – початкова чи кінцева вершина цієї дуги.



Для орієнтованого графа означення степеня вершини інше. В орієнтованому мультиграфі *напівстепенем входу* вершини  $v$  називають кількість дуг, для яких вершина  $v$  кінцева; позначають  $\deg^-(v)$ . *Напівстепенем виходу* вершини  $v$  називають кількість дуг, для яких вершина  $v$  початкова; позначають  $\deg^+(v)$ .



**Рис. 6**

**Приклад.** Для графа, зображеного на рис. 6 напівстепені вершин такі:

$$\deg^-(v_1)=0, \deg^+(v_1)=1, \deg^-(v_2)=2, \deg^+(v_2)=0, \deg^-(v_3)=1, \deg^+(v_3)=2.$$

**Теорема 3.** Нехай  $G=(V, E)$  – орієнтований мультиграф, який має  $m$  дуг. Тоді

$$\sum_{v \in V} \deg^-(v) = \sum_{v \in V} \deg^+(v) = m.$$

**Доведення.** Оскільки кожна дуга має початкову й кінцеву вершини, то суми напівстепенів входу й виходу однакові. Кожна з цих сум, очевидно, дорівнює кількості дуг. Теорему доведено.

Неорієнтований граф  $H$  називають *підграфом* неорієнтованого графа  $G=(V, E)$ , якщо всі вершини графа  $H$  належать  $V$ , а всі його ребра належать  $E$ . Якщо графи  $H=(W, F)$  і  $G=(V, E)$  – прості та  $H$  – підграф графа  $G$ , то розглядають іще два окремих випадки. Підграф  $H$  називають *каркасним підграфом* (або *фактором*), якщо  $W=V$ . Якщо  $W \neq V$ , а  $F$  – множина всіх ребер із  $E$ , які мають кінці в  $W$ , то підграф  $H$  називають *породженим (індукованим) множиною  $W$*  і позначають як  $G(W)$ .

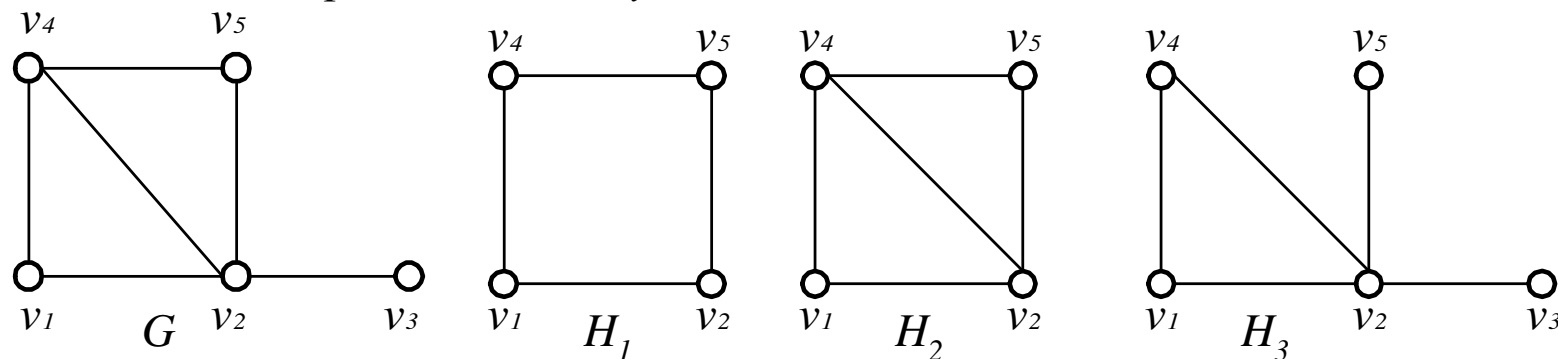


Рис. 7

**Приклад.** На рис. 7 зображено граф  $G$  та три його підграфів  $H_1$ ,  $H_2$ ,  $H_3$ , серед яких  $H_2$  – породжений, а  $H_3$  – каркасний.

Об'єднанням двох простих графів  $G_1=(V_1,E_1)$  та  $G_2=(V_2,E_2)$  називають простий граф  $G=(V, E)$  такий, що  $V=V_1\cup V_2$ ,  $E=E_1\cup E_2$ .

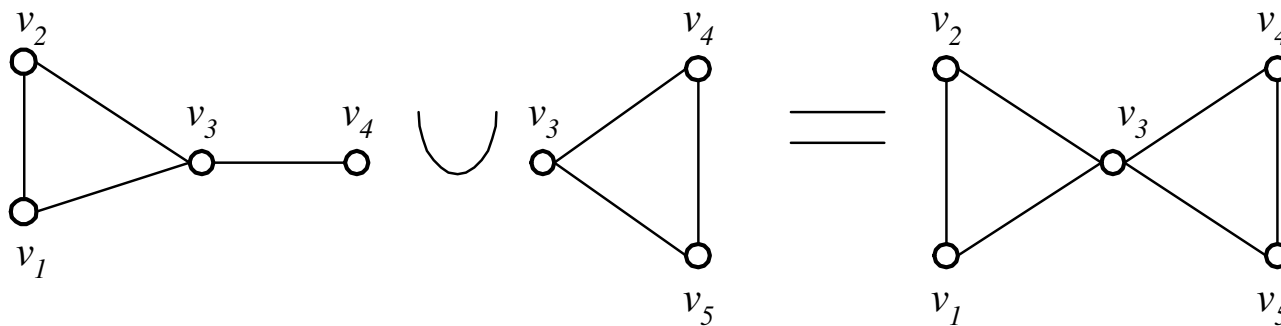


Рис. 8

**Приклад.** На рис. 8 наведено приклад об'єднання двох простих графів. Знаки „ $\cup$ ” та „ $=$ ” тут мають символічний зміст.

## Деякі спеціальні класи простих графів

Розглянемо деякі спеціальні класи простих графів, які часто використовують як приклади й широко застосовувані.

*Повний граф з  $n$  вершинами* (позначають як  $K_n$ ) – це граф, у якого будь-яку пару вершин з'єднано точно одним ребром.

Кількість ребер у графі  $K_n$  дорівнює  $C_n^2 = n(n-1)/2$ .

**Приклад.** На рис. 9 зображено графи  $K_n$  для  $n=1, 2, 3, 4, 5$ .

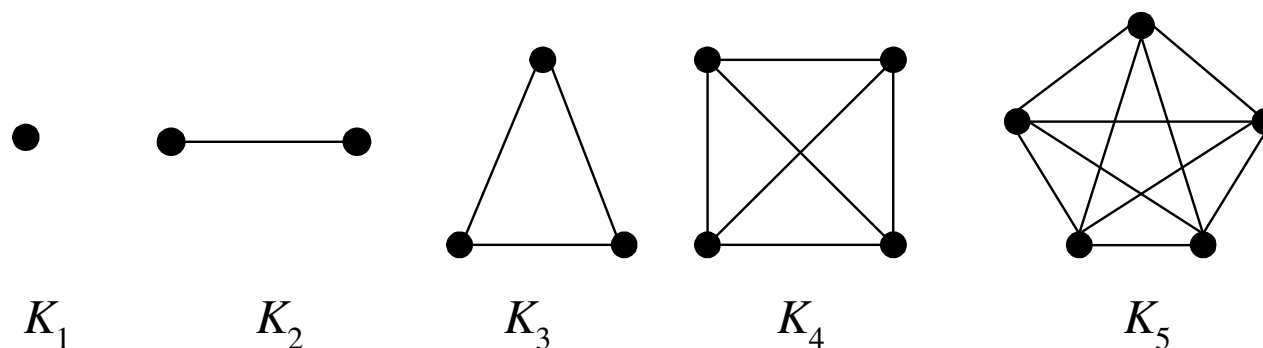
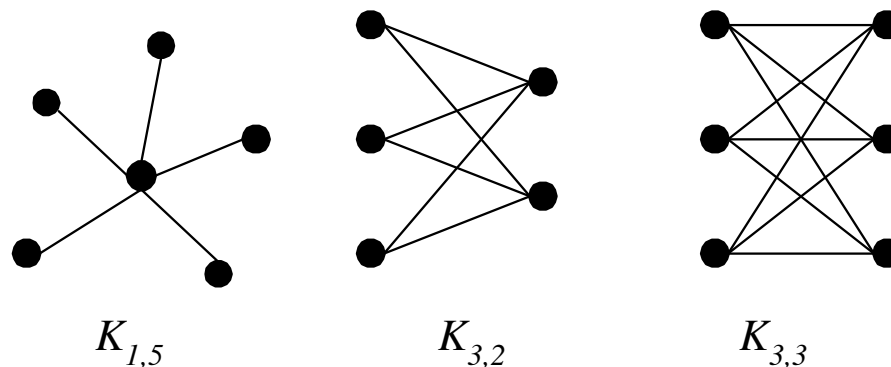


Рис. 9

Граф називають *порожнім*, якщо  $E=\emptyset$ , тобто такий граф не має ребер. Порожній граф з  $n$  вершинами позначають як  $O_n$ .

Простий граф  $G = (V, E)$  називають *двочастковим*, якщо множину його вершин  $V$  можна розбити на дві підмножини  $V_1$  і  $V_2$ , що не перетинаються ( $V_1 \cup V_2 = V$ ,  $V_1 \cap V_2 = \emptyset$ ), так, що кожне ребро з'єднує вершину з  $V_1$  і вершину з  $V_2$ . Іноді двочастковий граф позначають як  $G = (V_1 \cup V_2, E)$ . Двочастковий граф називають *повним двочастковим графом*, якщо кожному вершину з  $V_1$  з'єднано ребром із кожною вершиною з  $V_2$ . Повний двочастковий граф позначають як  $K_{m,n}$ , де  $m = |V_1|$ ,  $n = |V_2|$ . Граф  $K_{1,n}$  називають *зіркою*.

Граф  $K_{m,n}$  має  $n+m$  вершин та  $n \cdot m$  ребер.



**Рис. 10**

**Приклад 10.** На рис. 10 наведено повні двочасткові графи  $K_{1,5}$ ,  $K_{3,2}$  та  $K_{3,3}$ .

Циклом  $C_n$ ,  $n \geq 3$ , називають граф із множиною вершин  $V = \{v_1, v_2, \dots, v_n\}$  і множиною ребер  $E = \{\{v_1, v_2\}, \{v_2, v_3\}, \dots, \{v_{n-1}, v_n\}, \{v_n, v_1\}\}$ .

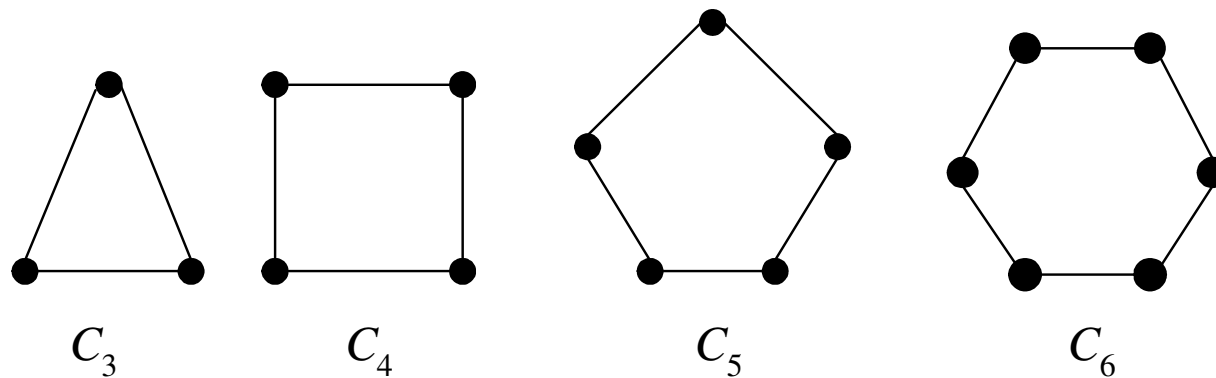
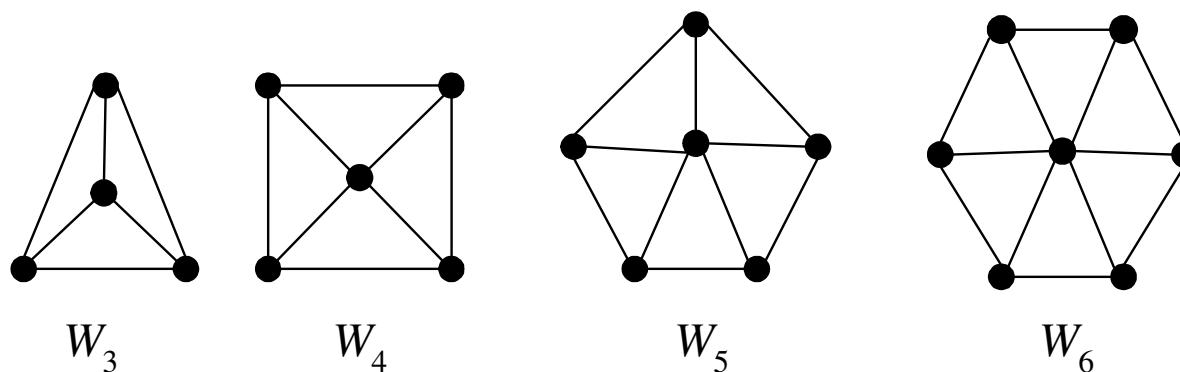


Рис. 11

**Приклад.** На рис. 11 зображено цикли  $C_3$ ,  $C_4$ ,  $C_5$  і  $C_6$ .

Колесом  $W_n$  називається граф, який одержують із циклу  $C_n$  додаванням ще однієї вершини, яку з'єднують з усіма  $n$  вершинами в  $C_n$  новими ребрами.

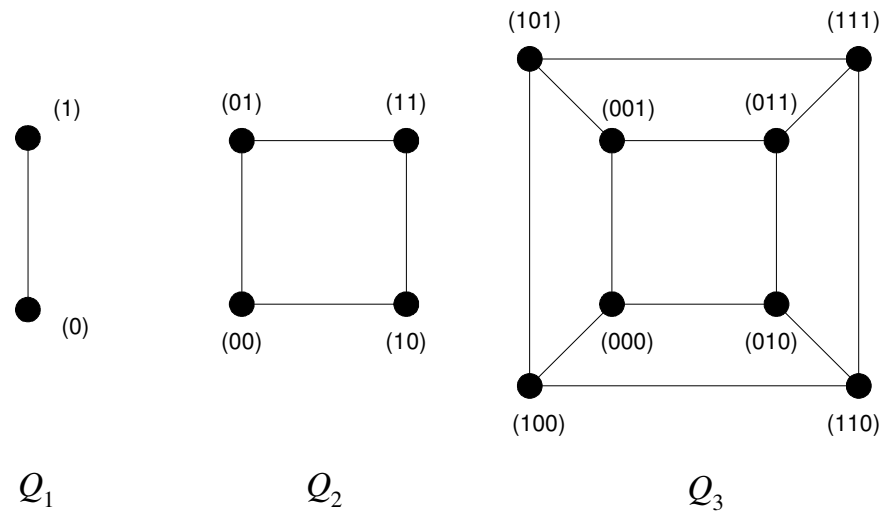


**Рис. 12**

**Приклад** На рис. 12 зображено колеса  $W_3$ ,  $W_4$ ,  $W_5$ ,  $W_6$ .

Граф, вершини якого відповідають усім  $2^n$  бітовим рядкам довжиною  $n$ , називають  $n$ -вимірним кубом і позначають  $Q_n$ . Дві вершини в  $Q_n$  з'єднано ребром тоді й лише тоді, коли відповідні бітові рядки відрізняються точно в одному біті.

**Приклад.** на рис. 13 зображено Графи  $Q_1$ ,  $Q_2$  та  $Q_3$ .



**Рис. 13**

Граф  $Q_{n+1}$  можна отримати з двох графів  $Q_n$ , з'єднавши ребрами їхні однаково позначені вершини. Після цього до бітових рядків у вершинах одного з графів  $Q_n$  зліва дописують 0, другого – дописують 1.

**Приклад.** Скільки ребер має граф  $Q_n$ ? Степінь кожної вершини графа  $Q_n$  дорівнює  $n$  (це випливає з означення) і тому  $\sum_{i=1}^n \deg(v_i) = n \cdot 2^n$ . Позначимо як  $m$  кількість ребер. За теоремою про рукостискання  $n \cdot 2^n = 2m$ . Звідси кількість ребер графа  $Q_n$  дорівнює  $n \cdot 2^{n-1}$ .



## Способи подання графів

Найзрозуміліший і корисний для людини спосіб подання (зображення) графів – це рисунок на площині у вигляді точок і ліній, які з'єднують ці точки. Проте цей спосіб подання абсолютно непридатний, якщо потрібно розв'язувати на комп'ютері задачі з графами.

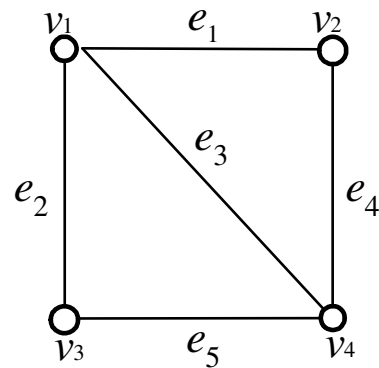


Рис. 14

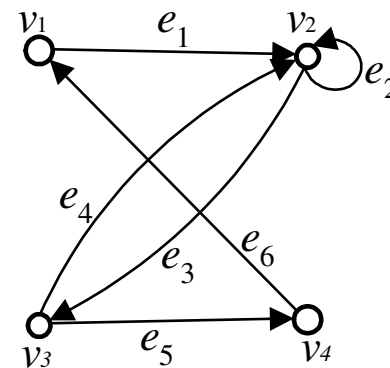


Рис. 15

Розглянемо декілька інших способів подання графів. Для спрощення розглядатимемо два найбільш важливих типи графів: простий граф (рис. 14) і орієнтований граф (рис. 15).

Матрицю, кожний елемент якої дорівнює 0 або 1, називають *булевою*.

### Матриця інцидентності

Нехай  $G=(V,E)$  – простий граф із множиною вершин  $V=\{v_1, v_2, \dots, v_n\}$  і множиною ребер  $E=\{e_1, e_2, \dots, e_m\}$ .

*Матрицею інцидентності* графа  $G$ , яка відповідає заданій нумерації вершин і ребер, називають булеву  $n \times m$  матрицю  $M$  з елементами  $m_{ij}$  ( $i=1, \dots, n, j=1, \dots, m$ ), де

$$m_{ij} = \begin{cases} 1, & \text{якщо вершина } v_i \text{ та ребро } e_j \text{ інцидентні,} \\ 0 & \text{в протилежному випадку.} \end{cases}$$

**Приклад.** Для графа, зображеного на рис. 14, матриця інцидентності має вигляд

$$\begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{bmatrix} e_1 & e_2 & e_3 & e_4 & e_5 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}.$$

Отже, для простого графа в матриці інцидентності в кожному стовпці точно дві одиниці, і немає однакових стовпців. Матрицю інцидентності можна використовувати й для подання мультиграфа. Тоді з'являться однакові стовпці (вони відповідають кратним ребрам). Для подання псевдографа петлю  $e_j$  у вершині  $v_i$  зображають значенням  $m_{ij} = 2$  (у цьому разі матриця інцидентності, очевидно, не булева).

За допомогою матриці інцидентності можна подавати й орієнтовані графи. Для таких графів вона також не булева. Нехай  $G=(V,E)$  – орієнтований граф із множиною вершин  $V=\{v_1, v_2, \dots, v_n\}$  і множиною дуг  $E=\{e_1, e_2, \dots, e_m\}$ .

Матрицею інцидентності орієнтованого графа  $G$ , яка відповідає заданій нумерації вершин і дуг, називають  $n \times m$  матрицю  $M$  з елементами  $m_{ij}$  ( $i=1, \dots, n, j=1, \dots, m$ ), де

$$m_{ij} = \begin{cases} 1, & \text{якщо дуга } e_j \text{ виходить з вершини } v_i, \\ -1, & \text{якщо дуга } e_j \text{ входить у вершину } v_i, \\ 2, & \text{якщо дуга } e_j \text{ — це петля у вершині } v_i, \\ 0 & \text{в інших випадках.} \end{cases}$$

**Приклад 5.** Для графа, зображеного на рис. 15, матриця інцидентності має вигляд

$$\begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{bmatrix} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \\ 1 & 0 & 0 & 0 & 0 & -1 \\ -1 & 2 & 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix}.$$

З алгоритмічної точки зору матриця інцидентності не є добрим вибором для комп'ютерних застосувань. По-перше, вона вимагає  $nm$  комірок пам'яті, більшість із яких зайнята нулями. По-друге, незручний доступ до інформації. Щоб отримати відповідь на елементарні запитання (наприклад, чи існує дуга  $(v_i, v_j)$ , до яких вершин ведуть дуги з  $v_i$ ), у найгіршому випадку потрібно перебрати всі стовпці матриці, тобто виконати  $m$  кроків.

### Матриця суміжності.

Нехай  $G=(V, E)$  – простий граф,  $|V|=n$ . Припустимо, що вершини графа  $G$  занумеровані:  $v_1, v_2, \dots, v_n$ . Матрицею суміжності графа  $G$  (яка відповідає даній нумерації вершин) називають булеву  $n \times n$  матрицю  $A$  з елементами  $a_{ij}$  ( $i, j=1, \dots, n$ ), де

$$a_{ij} = \begin{cases} 1, & \text{якщо } \{v_i, v_j\} \in E, \\ 0 & \text{в протилежному випадку.} \end{cases}$$

**Приклад 6.** Матриця суміжності для графа, зображеного на рис. 14, має вигляд

$$\begin{array}{ccccc} & v_1 & v_2 & v_3 & v_4 \\ \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} & \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \end{array}.$$

Цілком очевидно, що для неорієнтованого графа  $a_{ij}=a_{ji}$ , тобто матриця суміжності неорієнтованого графа симетрична. Більше того, позаяк у простому графі немає петель, то для нього в матриці суміжності  $a_{ii}=0$  ( $i=1, \dots, n$ ).

Матрицю суміжності можна використовувати також для подання псевдографа. Тоді це не булева матриця: елемент  $a_{ij}$  дорівнює кількості ребер, що з'єднують  $v_i$  та  $v_j$ . Петлю у вершині  $v_i$  подають значенням діагонального елемента  $a_{ii}=1$ .

Для подання орієнтованих графів також використовують матрицю суміжності. Це булева  $n \times n$  матриця  $A$  з елементами  $a_{ij}$  ( $i, j=1, \dots, n$ ), де

$$a_{ij} = \begin{cases} 1, & \text{якщо } (v_i, v_j) \in E, \\ 0 & \text{в протилежному випадку.} \end{cases}$$

**Приклад 7.** Матриця суміжності для графа, зображеного на рис. 15, має вигляд

$$\begin{array}{ccccc} & v_1 & v_2 & v_3 & v_4 \\ \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} & \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} \end{array}.$$

Зазначимо, що матриця суміжності орієнтованого графа, загалом кажучи, несиметрична.

Матрицю суміжності можна використовувати й для подання орієнтованого мультиграфа. У такому разі це не булева матриця: елемент  $a_{ij}$  дорівнює кількості дуг, які мають  $v_i$  початковою вершиною, а  $v_j$  – кінцевою.

Великою перевагою матриці суміжності як способу подання графа є швидкий доступ до інформації: за один крок можна одержати відповідь на запитання, чи існує ребро (дуга) з  $v_i$  у  $v_j$ . Недоліком є те, що незалежно від кількості ребер обсяг пам'яті становить  $n^2$  комірок. Як іще один аргумент проти використання матриці суміжності можна зазначити, що деякі алгоритми, які в разі використання матриці суміжності мають оцінку складності  $O(n^2)$ , для іншого способу подання графа мають кращу  $O$ -оцінку.

Нарешті, розглянемо ще два способи подання графів у пам'яті комп'ютера. Будь-яку скінченну послідовність елементів довільної природи будемо називати *списком*, а кількість елементів списку – його *довжиною*.

## Подання графа списком пар (списком ребер)

Економнішим щодо пам'яті, особливо якщо  $m$  (кількість ребер) значно менша, ніж  $n^2$  ( $n$  – кількість вершин), є метод подання графа списком пар, які відповідають його ребрам (або дугам). Пара  $[u, v]$  відповідає ребру  $\{u, v\}$ , якщо граф неорієнтований, і дузі  $(u, v)$ , якщо граф орієнтований.

Для графів, зображених на рис. 14 та 15, списки пар подані відповідно на рис. 16, а та 16, б.

$v_1$	$v_2$
$v_1$	$v_3$
$v_1$	$v_4$
$v_2$	$v_4$
$v_3$	$v_4$

а)

$v_1$	$v_2$
$v_2$	$v_2$
$v_2$	$v_3$
$v_3$	$v_2$
$v_3$	$v_4$
$v_4$	$v_1$

б)

Рис. 16

Очевидно, що обсяг пам'яті в разі використання цього способу подання дорівнює  $2m$  – це найекономніший щодо пам'яті спосіб. Недолік – велика (порядку  $m$ ) кількість кроків для знаходження множини вершин, до яких ідуть ребра або дуги із заданої вершини. Ситуацію можна значно поліпшити, упорядкувавши множину пар лексикографічно та застосувавши двійковий пошук.

## Подання графа списками суміжності.

Орієнтований граф  $G$  (без кратних дуг, але, можливо, з петлями) можна подати, указавши скінченну непорожню множину вершин  $V$  і відповідність  $\Gamma$ , котра вказує, як зв'язані між собою вершини. Відповідність  $\Gamma$  – багатозначне відображення множини  $V$  у  $V$ , а граф у такому разі позначають парою  $G=(V, \Gamma)$ . У літературі часто означають (орієнтований) граф саме в таких поняттях.

**Приклад.** Для орієнтованого графа, зображеного на рис. 15, відповідність  $\Gamma$  задано табл. 1.

Цей спосіб подання можна використовувати й для неорієнтованих простих графів, якщо кожне ребро умовно замінити двома протилежно напрямленими дугами.

**Приклад.** Для простого графа, зображеного на рис. 14, відповідність  $\Gamma$  задано табл. 2.

Таблиця 1

$v$	$\Gamma(v)$ (термінальні вершини)
$v_1$	$v_2$
$v_2$	$v_2, v_3$
$v_3$	$v_2, v_4$
$v_4$	$v_1$

Таблиця 2

$v$	$\Gamma(v)$ (суміжні з $v$ вершини)
$v_1$	$v_2, v_3, v_4$
$v_2$	$v_1, v_4$
$v_3$	$v_1, v_4$
$v_4$	$v_1, v_2, v_3$



Якщо відображення  $\Gamma$  діє не на одну вершину, а на множину вершин  $A = \{x_1, x_2, \dots, x_p\}$ , то під  $\Gamma(A)$  розуміють об'єднання множин

$$\Gamma(A) = \Gamma(x_1) \cup \Gamma(x_2) \cup \dots \cup \Gamma(x_p).$$

Розглянемо спосіб комп'ютерного подання графа *списками суміжності*. Для цього використовують масив  $Adj$  із  $n = |V|$  списків – по одному на кожну вершину. Для кожної вершини  $u \in V$  список  $Adj[u]$  містить у довільному порядку (вказівники на) всі вершини множини  $\Gamma(u)$ .

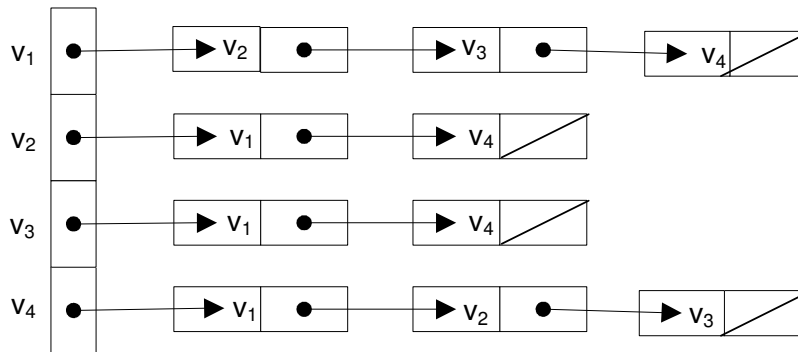


Рис. 17

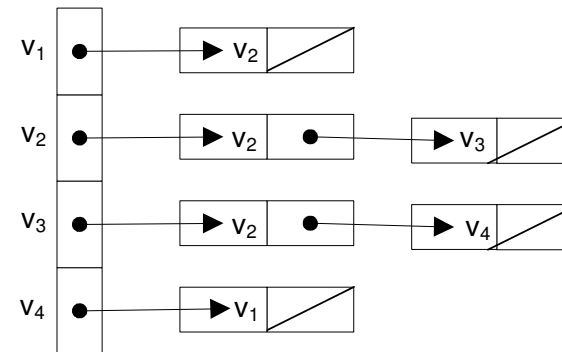


Рис. 18

**Приклад.** На рис. 17 за допомогою списків суміжності подано неорієнтований граф з рис. 14. На рис. 18 показано аналогічне подання для орієнтованого графа з рис. 15.

Для орієнтованого графа сума довжин усіх списків суміжних вершин дорівнює загальній кількості дуг: дузі  $(u, v)$  відповідає елемент  $v$  зі списку  $Adj[u]$ . Для неорієнтованого графа ця сума дорівнює подвоєній кількості ребер, бо ребро  $\{u, v\}$  породжує елемент у списку суміжних вершин як для вершини  $u$ , так і для вершини  $v$ . В обох випадках об'єм необхідної пам'яті складає  $O(\max(|V|, |E|)) = O(\max(|V| + |E|))$ .