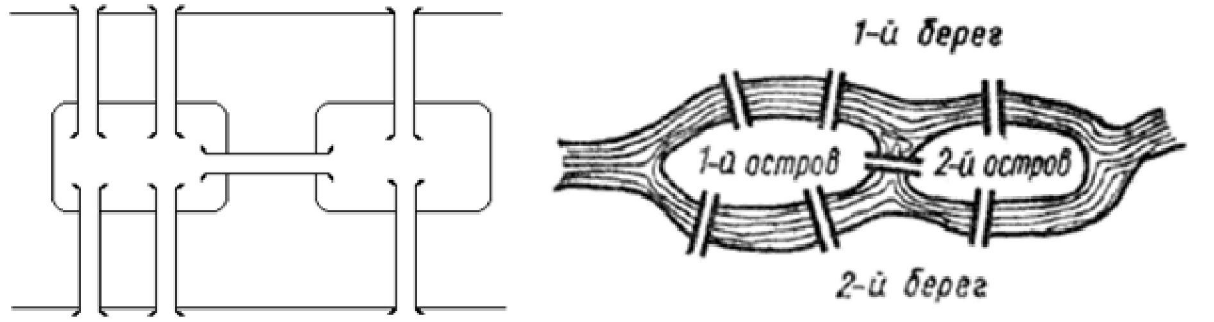
Теорія графів

- Історія теорії графів
- Основні поняття
- Представлення графів в комп'ютері
- Види графів
- Зв'язність
- Планарність
- Рід і товщина графу
- Незалежність і покриття
- Розфарбовування графа

Історія теорії графів

Задача про Кенігсбергскі мости

- На малюнку представлено схематичний план центральної частини міста Кенігсберг, що включає два береги річки Перголя, два острови на ній і сім мостів, що сполучають їх. Задача полягає в тому, щоб знайти маршрут, що проходить по усіх чотирьох ділянках суші по одному разу. При цьому через кожен з мостів можна проходити тільки по одному разу, а кінець і початок шляху повинні співпадати. Ця задача була розв’язана (показано, що розв’язок не існує) в 1736 році математиком Леонардом Ейлером.

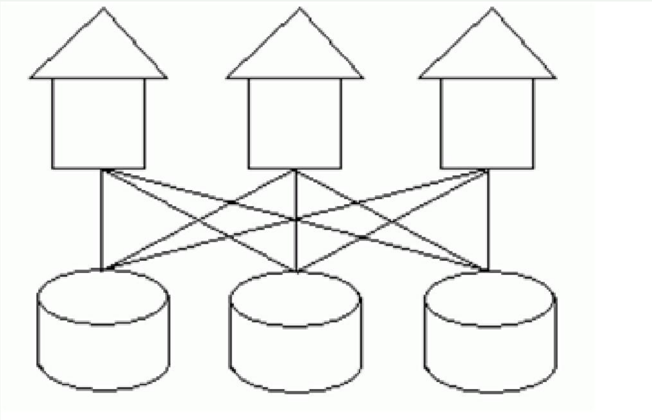


Задача про Кенігсбергські мости

Історія теорії графів

Задача про три домівки і три колодязі

- Є три домівки і три колодязі, якимсь чином розташовані на площині. Потрібно провести від кожної домівки до кожного колодязя стежину так, щоб стежини не перетиналися. Ця задача була розв’язана (показано, що розв’язок не існує) Куратовським в 1930 році.

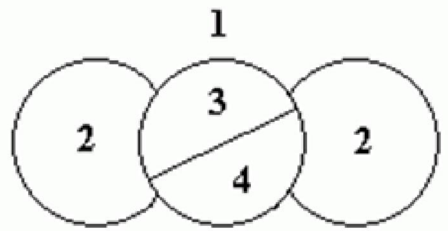


Задача про три домівки і три колодязі

Історія теорії графів

Задача про чотири фарби

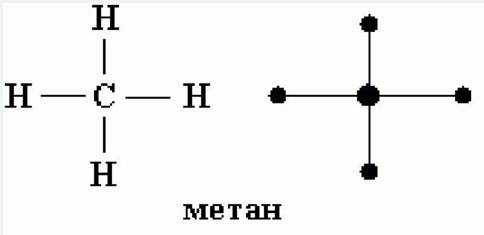
- Розбиття площини на частини, що не перетинаються, називається картою. Області на карті називаються сусідніми, якщо вони мають спільну границю. Задача полягає в розфарбовуванні карти так, щоб ніякі дві сусідні області на були зафарбовані одним кольором. З кінця позаминулого століття відома гіпотеза, що для цього досить чотирьох фарб. У 1976 році Аппель і Хейкен опублікували розв’язок цієї задачі, який базувався на переборі варіантів за допомогою комп'ютера.

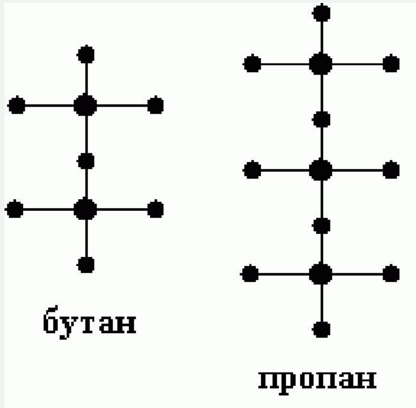


Задача про чотири фарби

Історія теорії графів

- У 1847 році Кірхгоф розробив ***теорію дерев*** для розв'язання систем лінійних алгебраїчних рівнянь, яка дозволила йому знайти струм в кожному провіднику в кожному контурі. Від ланцюгів він перейшов до графів. Кірхгоф показав, що не потрібно розглядати увесь граф, а розглядати прості цикли графа, які утворюються остовним графом.
- У 1957 році вчений Келлі розробив теорію перерахування дерев при спробі знайти ізомер вуглеводню.





Граф (від греч. γραφω - пишу)

- ***Графом G(V, E)*** називається сукупність двох множин –
  - непустої множини V (множини вершин) і
  - множини E - двоелементних підмножин множини V, що складається з різних елементів множини V (множини ребер).

$$G(V, E) = \langle V; E\rangle, \qquad V \neq \varnothing,$$

$$E = \{\ e=\{a,b\} \mid a,b\in V \ \&\ a\neq b \ \&\ e\neq\varnothing \ \&\ \forall e\in E \ |e|=2\ \}$$

Множина двоелементних підмножин визначає ***симетричне бінарне відношення*** на множині вершин:

$$E \subset V \times V, \qquad E = E^{-1}.$$

Граф

- Вершини, які не належать жодному ребру, називаються і***зольованими***.

- Ребро можна позначити

  не тільки *як множину* $\{v_1, v_2\}$,

  але й *як пару* (v_1, v_2).

Число вершин графа G позначається $\boldsymbol{p}$, а *число ребер* - $\boldsymbol{q}$:

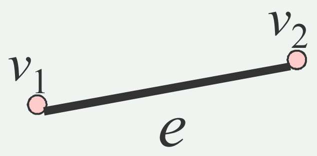
$$p = p(G) = |V|, \qquad q = q(G) = |E|.$$

- Якщо хочуть явно згадати числові характеристики графа, то кажуть

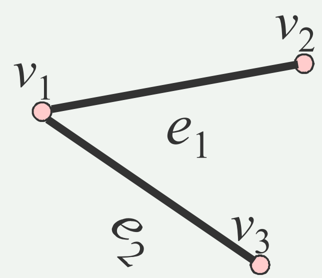
$$p, q \text{ - граф.}$$

Суміжність

- Нехай v_1, v_2 - вершини,
  $e=\{v_1,v_2\}$ – з'єднуючі їх ребра, тоді вершина v_1 і ребро e ***інцидентні***, ребро e і вершина v_2 також *інцидентні*.



- Два ребра, *інцидентні* одній вершині, називаються ***суміжними*** (ребро e_1 і e_2).
- Дві вершини, *інцидентні* одному ребру, також називаються ***суміжними*** (вершини v_1 і v_2, v_1 і v_3).



Суміжність

- Множина вершин, суміжних з вершиною v, називається ***множиною суміжності*** вершини v і позначається $\Gamma^{+}(v)$: $\Gamma^{+}(v)=\{ u\in V \mid (u,v)\in E \}$,

$$\Gamma^{*}(v) = \Gamma^{+}(v) + \{ v \}.$$

Якщо не обумовлено інше, то під Γ^{+} розуміється Γ^{*}.

Очевидно, що $u\in \Gamma(v) \Leftrightarrow v\in \Gamma(u)$.

- Якщо $A \subset V$ (A - деяка підмножина вершин графа), то $\Gamma(A)$ - множина всіх вершин, суміжних з вершинами з множини A: $\Gamma(A) = \{ u\in V \mid \exists v\in A \ \ u\in \Gamma(v) \} = \bigcup_{v\in A} \Gamma(v)$.

Діаграми

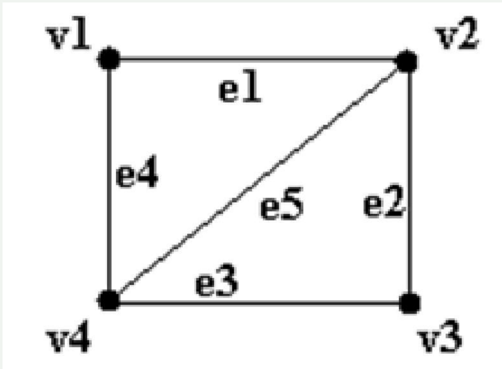
- Граф зображають ***діаграмою***:

  вершини - точками (або кружечками), ребра - лініями.

На рисунку представлено *діаграму графа*, який має 4 вершини і 5 ребер.

Вершини v_1 і v_2, v_2 і v_3, v_3 і v_4, v_4 і v_1, v_2 і v_4 - *суміжні*, а вершини v_1 і v_3 *несуміжні*.

Суміжні ребра: e_1 і e_2, e_2 і e_3, e_3 і e_4, e_4 і e_1, e_1 і e_5, e_2 і e_5, e_3 і e_5, e_4 і e_5.



Несуміжні ребра: e_1 і e_3, e_2 і e_4.

Множини суміжності вершини v_1:

$\Gamma^+(v_1) = \{v_2, v_4\}$

$\Gamma^*(v_1) = \{v_1, v_2, v_4\}$

Множина суміжності множини вершин $A=\{v_1, v_2\}$:

$\Gamma(A) = \{v_1, v_2, v_3, v_4\}$

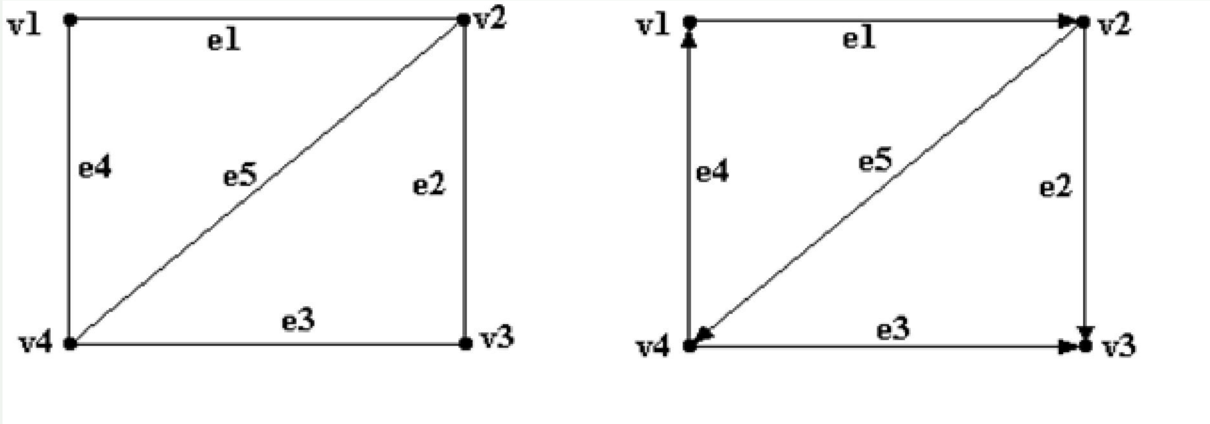
Способи представлення графів

Способи представлення графів в пам’яті комп’ютера відрізняються *об’ємом* займаної *пам’яті* та *швидкістю* виконання *операцій* над графами.

Представлення вибирається, виходячи з потреб конкретної задачі.

- Чотири найчастіше використовуваних представлення із вказівкою характеристики $n(p, q)$ – об’єму пам’яті для кожного представлення.
  1. **матриця суміжності**, $n(p, q) = O(p^2)$.
  2. **матриця інциденцій**, $n(p, q) = O(p*q)$.
  3. **списки суміжності**, $n(p, q) = O(p + 2q)$ [$n(p, q) = O(p + q)$].
  4. **масив ребер (або дуг)**, $n(p, q) = O(2q)$.

Представлення ілюструється на прикладах графу G і орграфу D, діаграми яких представлені нижче:



Діаграми графа (ліворуч) і орграфа (праворуч)

Матриця суміжності

- Представлення графу за допомогою квадратної булевої матриці *M* : ***array*** *[1..p, 1..p]* ***of*** *0..1*, що відображає суміжність вершин, називається ***матрицею суміжності***, де

$$M[i,j]=\begin{cases}1,\ \text{якщо вершина } v_i \text{ суміжна з вершиною } v_j\\ 0,\ \text{якщо вершини } v_i \text{ і } v_j \text{ не суміжні}\end{cases}$$

- Для матриці суміжності $n(p, q) = O(p^2)$.

Приклад:

$$G:\begin{vmatrix}0&1&0&1\\1&0&1&1\\0&1&0&1\\1&1&1&0\end{vmatrix}\qquad D:\begin{vmatrix}0&1&0&0\\0&0&1&1\\0&0&0&0\\1&0&1&0\end{vmatrix}$$

- Матриця суміжності графу *симетрична* відносно головної діагоналі, тому достатньо зберігати тільки верхню (або нижню) трикутну матрицю.

Матриця інциденцій

- Представлення графу за допомогою матриці *H :**array** [1..p, 1..q] **of** 0..1* (для орграфів *H :**array** [1..p, 1..q] **of** -1..1*), що відображає інцидентність вершин і ребер, називається ***матрицею інциденцій***, де

для неорієнтованого графу

$$H[i,j]=\begin{cases}1, \textit{ якщо вершина } v_i \textit{ інцидентна ребру } e_j\\ 0, \textit{ в противному випадку}\end{cases}$$

для орієнтованого графу

$$H[i,j]=\begin{cases}1, \textit{ якщо вершина } v_i \textit{ інцидентна ребру } e_j \textit{ і є його кінцем}\\ 0, \textit{ якщо вершина } v_i \textit{ неінцидентна ребру } e_j\\ -1, \textit{ якщо вершина } v_i \textit{ інцидентна ребру } e_j \textit{ і є його початком}\end{cases}$$

- Для матриці інциденцій *n(p, q) = O(p*q)*.

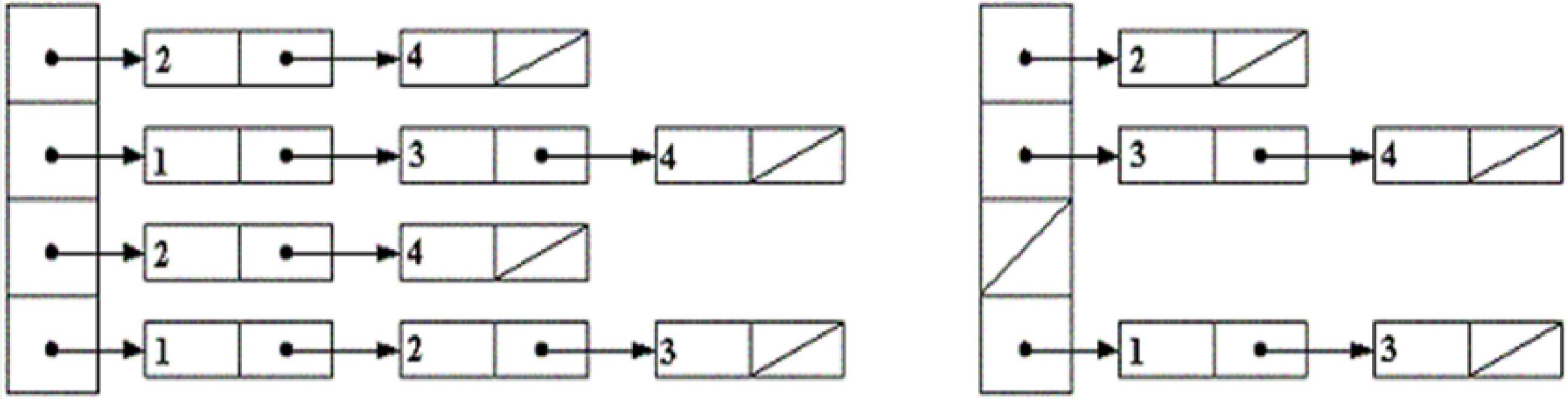
Приклад:

$$G:\ \begin{vmatrix}1&0&0&1&0\\1&1&0&0&1\\0&1&1&0&0\\0&0&1&1&1\end{vmatrix}\qquad D:\ \begin{vmatrix}-1&0&0&1&0\\1&-1&0&0&-1\\0&1&1&0&0\\0&0&-1&-1&1\end{vmatrix}$$

Списки суміжності

- Представлення графу за допомогою структури списку, що відображає *суміжність вершин* і складається з масиву вказівок Γ : ***array*** *[1..p]* ***of*** *^N* на списки суміжних вершин, де елемент списку представлено структурою
  N : ***record*** *v* : *1..p*; *n* : *^ N* ***end record***, називається ***списком суміжності***.
- У випадку представлення неорієнтованих графів списками суміжності $n(p, q) = O(p + 2q)$, а у випадку орієнтованих графів $n(p, q) = O(p + q)$.

Приклад:



Списки суміжності для графу G (ліворуч) і орграфу D (праворуч)

Масив дуг (ребер)

- Представлення графу за допомогою масиву структур *E* : ***array*** *[1..q]* ***of record*** *b, e : 1..p* ***end record***, що відображає список пар суміжних вершин, називається ***масивом ребер*** (або, для орграфів, ***масивом дуг***).
- Для масиву ребер (або дуг) $n(p, q) = O(2q)$.

Приклад:

b	e	b	e
1	2	1	2
1	4	2	3
2	3	2	4
2	4	4	1
3	4	4	3

Представлення графу і орграфу за допомогою масиву ребер (дуг)

Види графів

- Граф, який складається з *однієї* вершини, називається ***тривіальним***.
- Граф, що складається з одних *ізольованих* вершин, називається ***нуль-графом*** (**O_p**).
- Граф, в якому <u>кожна пара вершин з'єднана</u> ребром, називається ***повним*** (**K_p**).

Повний граф з *p* вершинами має максимально можливе число ребер: $q(K_p) = p\ (p - 1)\ /\ 2$

- Якщо елементами множини *E* є *впорядковані* пари (на ребрах задано напрям або порядок), то граф *G*(*V*,*E*) називають ***орієнтованим*** (***орграфом***), де *V*- множина ***вузлів*** графу, а *E* - множина його ***дуг***.

Види графів

- Якщо елементом множини *E* може бути пара *однакових* елементів *V*, то такий елемент множини *E* називається ***петлею***, а граф називається *графом з петлями* (***псевдографом***).

- Якщо *E* є не множиною, а *набором*, що містить кілька однакових елементів, то ці елементи називаються *кратними ребрами*, а граф називається ***мультиграфом***.

- Якщо елементами множини *E* не обов'язково є 2-елементні, а будь-які підмножини множини V, то такі елементи множини *E* називаються *гіпердугами*, а граф називається ***гіперграфом***.

Зведена таблиця графів

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

Види графів

- Якщо задана функція $f : V \to M$ і/або $f : E \to M$,
то множина M називається множиною *міток*,
а граф називається ***поміченим*** (або *навантаженим*)

В якості множини поміток використовуються *букви* або *цілі числа*.

- Якщо функція f ін'єктивна, тобто різні вершини (ребра) мають різні мітки, то граф називають ***нумерованим***.

- Якщо у графа помічені ребра - це ***реберно-помічений*** граф.

Вираз "граф $G(V, E)$" означає неорієнтований непомічений граф без петель і кратних ребер з множиною вершин V і множиною ребер E.

Ізоморфізм графів

- Говорять, що два графа $G_1(V_1, E_1)$ і $G_2(V_2, E_2)$ ***ізоморфні*** (позначається $G_1 \sim G_2$), якщо існує бієкція h: $V_1 \rightarrow V_2$, що зберігає суміжність $e_1=(u, v) \in E_1$ $e_2=(h(u), h(v)) \in E_2$

ТЕОРЕМА. *Ізоморфізм графів є відношення еквівалентності.*

Доведення:

Дійсно, ізоморфізм має всі необхідні властивості:

[Рефлексивність.] G ~ G, де потрібна бієкція є тотожною функцією;

[Симетричність.] Якщо $G_1 \sim G_2$ з бієкцією h, то $G_2 \sim G_1$ з бієкцією h^{-1};

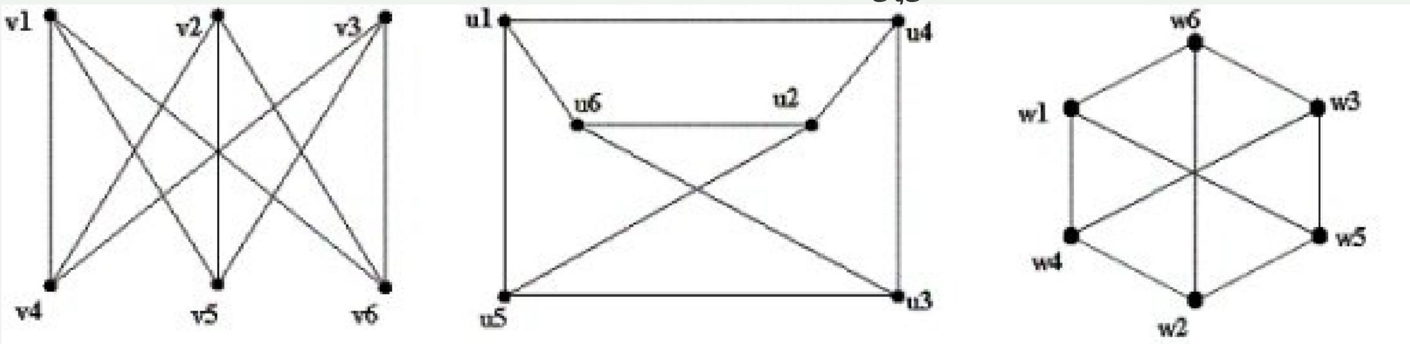
[Транзитивність.] Якщо $G_1 \sim G_2$ з бієкцією h і $G_2 \sim G_3$ з бієкцією g, то $G_1 \sim G_3$ з бієкцією $g \circ h$.

Ізоморфізм графів

Графи розглядаються з *точністю до ізоморфізму*, тобто розглядаються класи еквівалентності по відношенню ізоморфізму.

Приклад:

Три зовні розрізнюванні діаграми, приведені на рисунку, є діаграмами одного і того ж графу **$K_{3,3}$**.



Діаграми ізоморфних графів

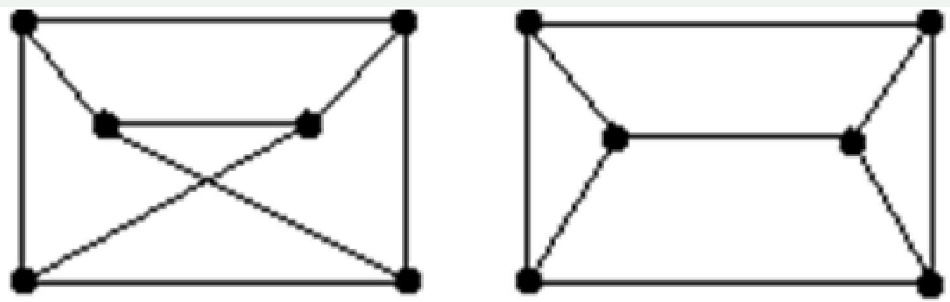
- Числова характеристика, однакова для всіх ізоморфних графів, називається ***інваріантом*** графа.

Так, *p(G)* і *q(G)* - інваріанти графа *G*. Невідомо жодного набору інваріантів, які визначають граф з точністю до ізоморфізму.

Ізоморфізм графів

Приклад:

Кількість вершин, ребер і кількість суміжних вершин для кожної вершини не визначають граф. На рисунку представлені діаграми графів, у яких вказані інваріанти співпадають, але графи при цьому не ізоморфні.



Діаграми неізоморфних графів із співпадаючими інваріантами

Частинні графи. Підграфи

➢ ***Підграфом*** графу $G(V,E)$ називається граф $G_X(X,E_X)$, у який входить лише
- частина вершин (вузлів) графу G, що утворюють множину $X=V'\subset V$,
- разом з ребрами (дугами) $E_X=E'\subset E$, що з'єднують ці вершини.

➢ Граф $G_X(X,E_X)$ називається ***власним*** підграфом графу G, якщо $X\subset V$ & $E_X\subset E$ & $(V' \neq V \vee E' \neq E)$.

❖ Підграф $G_X(X,E_X)$ називається ***остовним*** підграфом графу $G(V, E)$, якщо G_X містить всі вершини графу G: $X = V$.

Остовний підграф (або ***фактор***) $G'(V, E')$ графу $G(V, E)$ визначається підмножиною ребер E'.

❖ Підграф $G_X(X,E_X)$ називається ***правильним*** підграфом графу $G(V, E)$, якщо G_X містить усі ребра G:

$$\forall u, v \in X \quad \{u, v\} \in E \Rightarrow \{u, v\} \in E_X.$$

Правильний підграф $G'(V', E')$ графу $G(V, E)$ визначається підмножиною вершин V'.

➢ ***Частинним графом*** графа $G(V,E)$ називається граф G_Y, який містить тільки частину ребер (дуг) графа G: $G_Y(V_Y, Y)$, де $Y\subseteq E$.

Валентність (степінь)

- Кількість ребер, інцидентних вершині *v*, називається ***степенем*** (***валентністю***) вершини *v* і позначається $d(v)$[1] :

$$\forall v \in V \qquad 0 \le d(v) \le p - 1, \qquad d(v) = |\, \Gamma^{+}(v)\, |$$

(якщо не обумовлено особо, то петля враховується двічі при підрахунку $d(v)$)

  - Степінь $d(v)$ вершини *v* - це кількість суміжних з нею вершин.
  - Кількість вершин, не суміжних з *v*, позначимо $\bar{d}(v)$.

Ясно, що $\forall v \in V \quad d(v) + \bar{d}(v) = p$.

Степінь ***ізольованої*** вершини рівна нулю (тобто $d(v) = 0$).

- Якщо степінь вершини рівна одиниці (тобто $d(v) = 1$), то вершина називається ***кінцевою*** або ***висячею***.

- Позначимо ***мінімальну*** степінь вершини графа G – $\delta(G)$, а ***максимальну*** – через $\Delta(G)$:

$$\delta(G(V, E)) = \min d(v), \qquad \Delta(G(V, E)) = \max d(v).$$

[1] Використовують також позначення *deg v*.

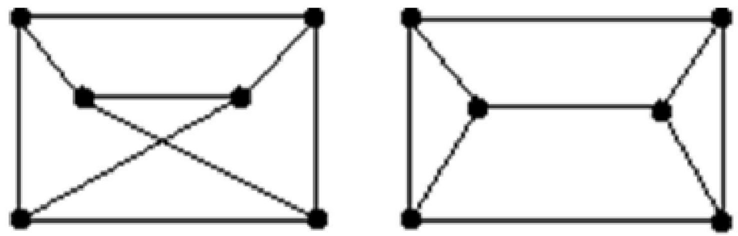
Валентність (степінь)

- Якщо степені всіх вершин графу рівні *k,* то граф називається ***регулярним*** графом степені *k:* $\delta(G) = \Delta(G) = k$.
- **Степінь регулярності** являється *інваріантом* графа і позначається *r(G)*.

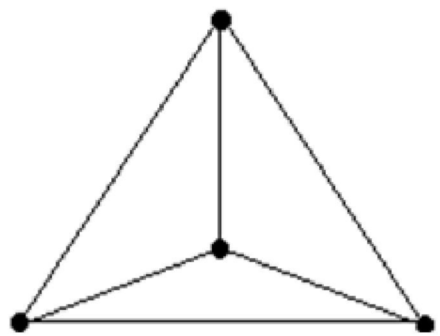
Для нерегулярних графів *r(G)* не визначено.

Приклад:

На рисунку приведені діаграми двох регулярних, але неізоморфних графів степені 3, а також діаграма регулярного графу степені 3.



Діаграми неізоморфних графів із співпадаючими інваріантами



Діаграма регулярного графу степені 3

Теорема Ейлера

ТЕОРЕМА (Ейлера). *Сума степенів вершин графа рівна подвоєній кількості ребер:* $\Sigma\, d(v) = 2q$.

Доведення:

При підрахунку суми степенів вершин кожне ребро враховується двічі: для одного кінця ребра і для другого.

НАСЛІДОК 1. *Число вершин непарної степені парне.*

Доведення:

По теоремі Ейлера сума степенів всіх вершин - парне число. Сума степенів вершин парної степені парна, а значить, сума степенів вершин непарної степені також парна, тобто їх число *парне*.

НАСЛІДОК 2. *Сума півстепенів вузлів орграфу рівна подвоєній кількості дуг:* $\Sigma\, d^{+}(v) + \Sigma\, d^{-}(v) = 2q$.

Доведення:

Сума півстепенів вузлів орграфу рівна сумі степенів вершин графу, отриманого з орграфу відкиданням орієнтації дуг.

Маршрути і ланцюги

- ***Маршрутом*** ***M(v_0,v_k)*** у графі називається чергуючуюся послідовність вершин і ребер v_0, e_1, v_1, e_2, v_2,..., e_k, v_k, в якій будь-які два сусідніх елементи інцидентні.
  - якщо $v_0 = v_k$, то маршрут ***замкнений***, інакше ***відкритий***.
- якщо всі ребра різні, то маршрут називається ***ланцюгом***.
  - якщо всі вершини (а значить, і ребра) різні, то маршрут називається ***простим ланцюгом***.
  - у ланцюгу v_0, e_1,..., e_k, v_k вершини v_0 і v_k називаються ***кінцями*** ланцюга. Говорять, що ланцюг з кінцями *u*, *v* ***з'єднує*** вершини *v* і *u*. *Ланцюг*, що з'єднує вершини *u* і *v*, позначається **<*u*, *v*>.**

Очевидно, що якщо є ланцюг, що з'єднує вершини *u* і *v*, то існує і простий ланцюг, що з'єднує ці вершини .

Цикли

- *Замкнутий ланцюг* називається ***циклом***.

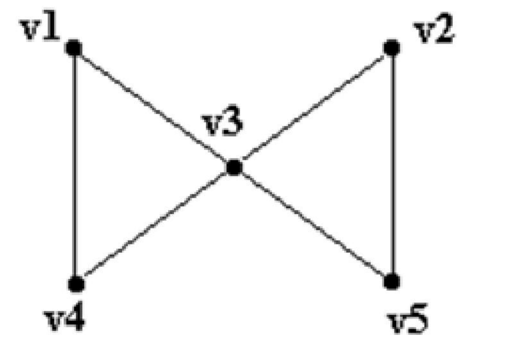
Число циклів у графі *G* позначається *z(G)*.

  - *Замкнений простий ланцюг* називається ***простим циклом***.

- Граф без циклів називається ***ациклічним***.
- Для орієнтованих графів *ланцюг* називається ***шляхом***, а *цикл* - ***контуром***.
- Граф, який складається з простого циклу з *k* вершинами, позначається C_k.

Приклад: C_3 - трикутник.

Маршрути, ланцюги, цикли



Маршрути, ланцюги, цикли

Приклад:

v_1, v_3, v_1, v_4 - маршрут, але не ланцюг;

$v_1, v_3, v_5, v_2, v_3, v_4$ - ланцюг, але не простий ланцюг;

v_1, v_4, v_3, v_2, v_5 - простий ланцюг;

$v_1, v_3, v_5, v_2, v_3, v_4$ - цикл, але не простий цикл;

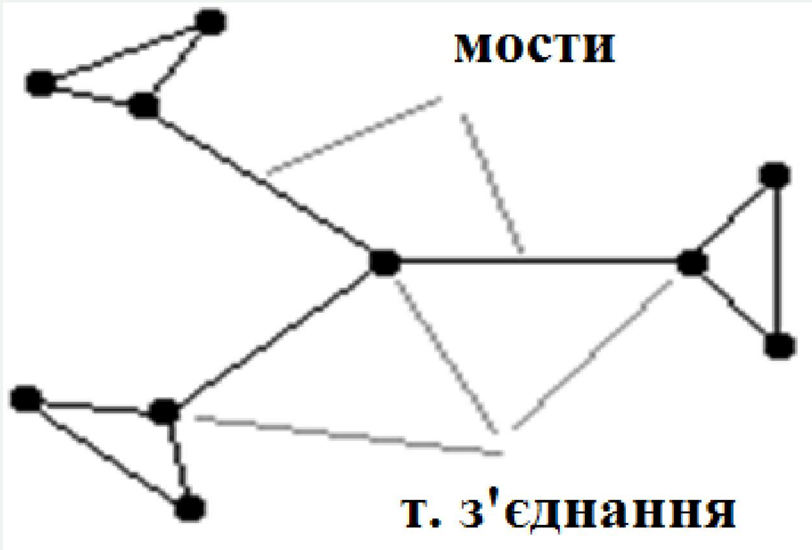
v_1, v_3, v_4, v_1 - простий цикл.

Зв’язність

- Дві вершини в графі ***зв’язані***, якщо існує з’єднуючий їх (простий) ланцюг.
- Граф, у якому *всі вершини зв’язані*, називається ***зв’язним***.
- Відношення зв’язності вершин являється *еквівалентністю*.
- *Класи еквівалентності* по відношенню зв’язаності називаються ***компонентами зв’язності*** графу. Число компонент зв'язності графу *G* позначається *k*(G).
- Граф *G* є зв’язним тоді і тільки тоді, коли *k*(G) = 1.

Зв'язність

- Якщо $k(G) > 1$, то G – ***незв’язний*** граф.
- Граф, який складається тільки з ізольованих вершин (в якому $k(G)=p(G)$ і $r(G) = 0$), називається ***повністю незв’язним***.
- ***Точка з’єднання / міст*** – це вершина / ребро, видалення якої / якого приводить до порушення зв’язності компонент даного графу.



Зв'язність

ТЕОРЕМА. *Якщо граф G має p вершин і k компонент зв'язності, то максимально можлива кількість ребер в ньому*

$$N(p, k) = \tfrac{1}{2}(p - k + 1)(p - k)$$

Зв'язність характеризується:

1. ***Числом вершинної зв'язності*** *(числом зв'язності)* $\chi(G)$ - найменшою кількістю вершин, видалення яких приводить до незв'язного або тривіального графу.

   Так, $\chi(K_1) = 0$; $\chi(K_p) = p - 1$; $\chi(C_p) = 2$.

2. ***Числом реберної зв'язності*** $\lambda(G)$ - мінімальною кількістю ребер, видалення яких приведе до незв'язного графу.

Зв'язність

В загальному випадку $\boldsymbol{\chi(G) \le \lambda(G) \le deg_{min}(G)}$, де $deg_{min}(G)$ - мінімальний степінь вершин в графе.

- **Випадок $\boldsymbol{\lambda(G) \le deg_{min}(G)}$:**
  - Якщо граф тривіальний, то в ньому немає ребер, а значить: $\lambda(G) = deg(G) = 0$.
  - Якщо G – зв'язний граф, то перетворити його у незв'язний граф можна таким чином: знайти вершини з мінімальним степенем і видалити інцидентні їм ребра.
- **Випадок $\boldsymbol{\chi(G) \le \lambda(G)}$:**
  - Для незв'язних графів $\chi = \lambda = 0$;
  - Для графа з мостом $\chi = \lambda = 1$.
  - В загальному випадку $\chi \le \lambda$, так як видалення вершини веде за собою видалення всіх інцидентних ребер.

Відстань між вершинами, яруси і діаметр графа

- ***Довжиною маршруту*** називається кількість ребер у ньому (з повторами). Якщо маршрут $M = v_0, e_1, \ldots, e_k, v_k$, то довжина M рівна k (позначається $|M| = k$).
- ***Відстанню*** між вершинами u і v (позначається $d(u, v)$) називається довжина найкоротшого ланцюга $\langle u, v \rangle$, а сам найкоротший ланцюг називається ***геодезичним***.
- Множина вершин, які знаходяться на однаковій відстані n від вершини v (позначення $D(v, n)$), називається ***ярусом***: $D(v, n) = \{u \in V \mid d(v, u) = n\}$

Очевидно, що всякий граф однозначно розбивається на яруси відносно даної вершини.

- ***Діаметром*** графа G (позначається $D(G)$) називається довжина найдовшого ланцюга.
- ***Обхват*** графу - це довжина найкоротшого простого циклу.
- ***Оточення*** графу - довжина максимального простого циклу.

Ексцентриситет і центр

- ***Ексцентриситетом*** $e(v)$ вершини v у зв'язному графі $G(V, E)$ називається максимальна відстань від вершини v до інших вершин графу G: $e(v) = max\ d(v, u)$.

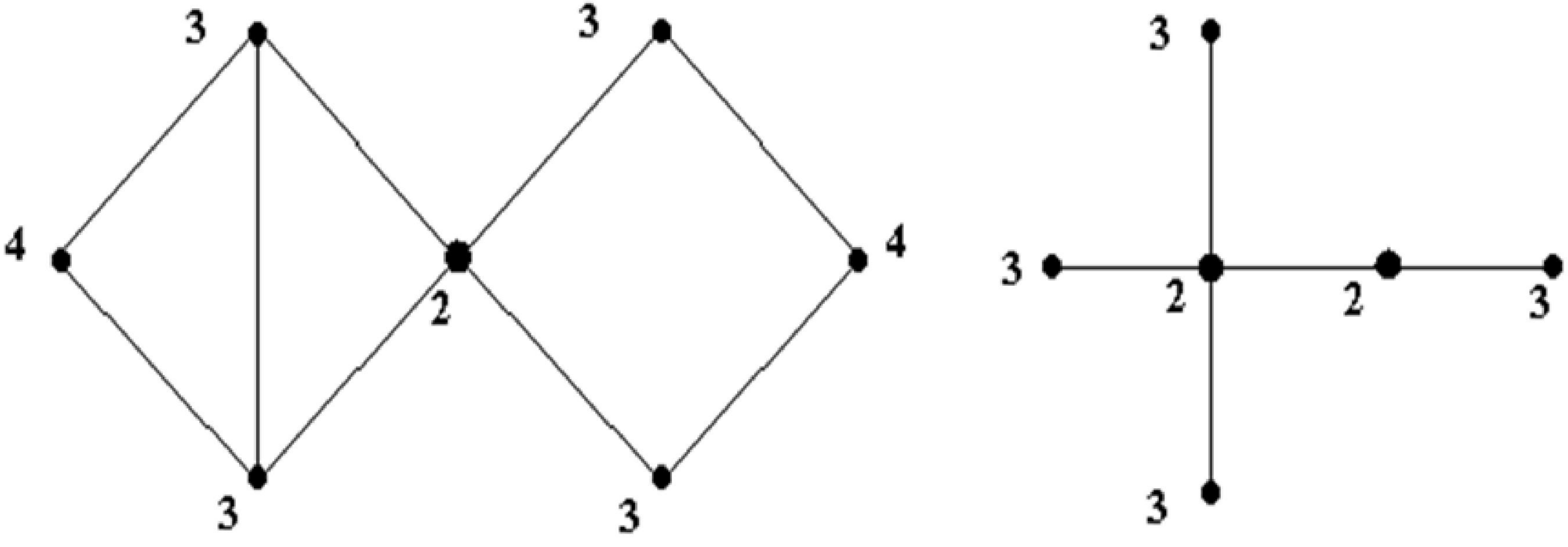
Найбільш ексцентричні вершини - це кінці діаметру.

- ***Радіусом*** $R(G)$ графу G називається найменший з ексцентриситетів вершин: $R(G) = min\ e(v)$.
- Вершина v називається ***центральною***, якщо її ексцентриситет співпадає з радіусом графа, $e(v) = R(G)$.
- Множина центральних вершин називається ***центром*** і позначається $C(G)$: $C(G) = \{ v \in V \mid e(v) = R(G)\}$.

Ексцентриситет і центр

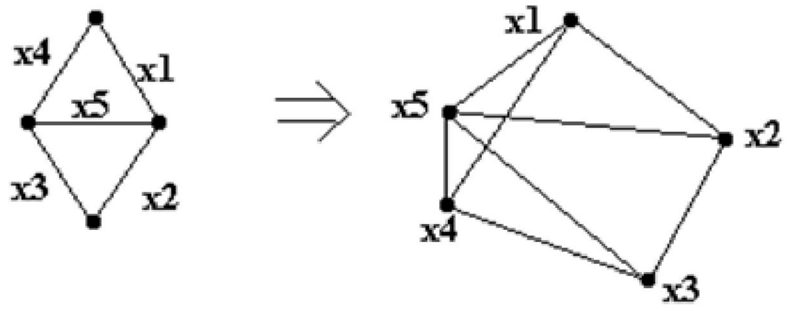
Приклад.

На рисунку вказані ексцентриситети вершин і центри двох графів. Вершини, які утворюють центр, виділені.



Ексцентриситети вершин і центри графів

Суміжнісні графи



Побудова суміжнісного графу

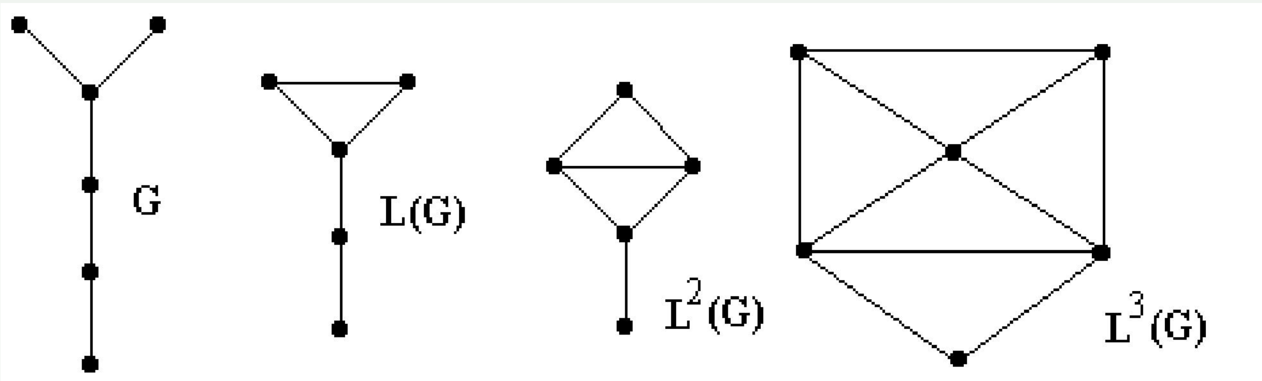
- Суміжнісний граф позначається *L(G)*.
- У суміжнісному графі кількість вершин рівна кількості ребер у вихідному графі - *q*. Кількість ребер у суміжнісному графі рівна

$$\frac{1}{2}\left[\sum_{i=1}\deg_i^2(V_i)\right]-q$$

- $L(G)=\left(q,\frac{1}{2}\left[\sum_{i=1}\deg_i^2(V_i)\right]-q\right)$

Властивості суміжнісних графів

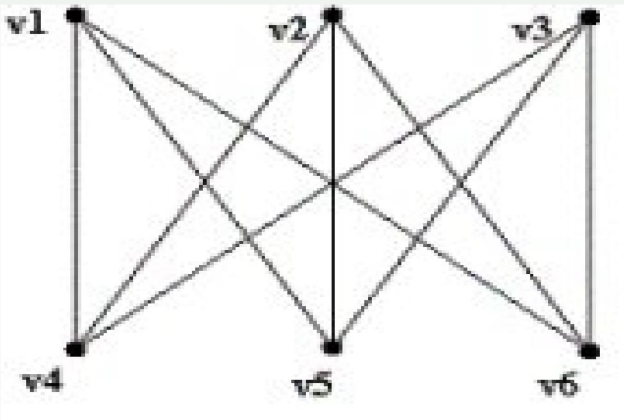
- Зв'язний граф G ізоморфний своєму суміжнісному графу, якщо він - простий цикл довжиною 3.
- Якщо є ейлерів граф, то суміжнісний йому граф - ейлерів і гамільтоновий.
- Якщо є гамільтоновий граф, то суміжнісний йому граф - теж гамільтоновий.
- Якщо G - граф з p вершинами ($p > 1$) і він не є простим ланцюгом, то $L^n(G)$ - гамільтоновий граф для всіх $n \geq p - 3$.



Дводольні графи

- ***Дводольний*** граф (або *біграф*, або *парний граф*) - це граф $G(V, E)$, такий, що
  - множина V розбита на дві неперетинні множини V_1 і V_2,
  - всяке ребро з E інцидентне вершині з V_1 і вершині з V_2 (тобто з'єднує вершину з V_1 з вершиною з V_2).
- Множини V_1 і V_2 називаються ***долями*** дводольного графу.
- Якщо дводольний граф містить всі ребра, які з'єднують множини V_1 і V_2, то він називається ***повним дводольним графом***.

Якщо $|V_1| = m$ і $|V_2| = n$, то повний дводольний граф – $K_{m,n}$.



Приклад.

На рисунку приведена діаграма графа $K_{3,\,3}$

Дводольні графи

- **ТЕОРЕМА**. *Граф є дводольним тоді і тільки тоді, коли всі його прості цикли мають парну довжину.*

Доведення.

[Необхідність (від супротивного).]

Нехай $G(V_1, V_2; E)$ - дводольный граф, і $v_1, v_2, \ldots, v_{2k+1}, v_1$ - простий цикл непарної довжини. Нехай $v_1 \in V_1$, тоді $v_2 \in V_2, \ldots, v_{2k+1} \in V_1$. Маємо: $v_1, v_{2k+1} \in V_1$ і $(v_1, v_{2k+1}) \in E$, що протирічить дводольності.

[Достатність.] Не обмежуючи загальності, можна вважати, що G – зв'язний граф, оскільки кожну компоненту зв'язності можна розглядати окремо. Розіб'ємо множину V на V_1 і V_2.

Далі від супротивного. Нехай є дві вершини в одній долі, з'єднані ребром. Нехай для визначеності $u, w \in V_2$ і $(u, w) \in E$. Розглянемо геодезичні $<v, u>$ і $<v, w>$. Тоді довжини $|<v, u>|$ і $|<v, w>|$ непарні. Ці геодезичні мають спільні вершини (як мінімум, вершину v). Розглянемо найбільш віддалену від v спільну вершину геодезичних $<v, u>$ і $<v, w>$ і позначимо її v' (може статися так, що $v = v'$). Маємо: $|<v', u>| + |<v', w>| = |<v, u>| + |<v, w>| - 2|<v, v'>|$ - парне, і $v',\ldots,u, w,\ldots,v'$ - простий цикл непарної довжини, що протирічить умові.

Двудольні графи

- ***Досконалою паросполукою*** з однієї множини в іншу у дводольному графі G називається <u>взаємно однозначна відповідність</u> між даними множинами, які мають таку властивість: відповідні вершини у вихідному графі суміжні.

- **ТЕОРЕМА** (про весілля). *Нехай деякий граф $G(V_1, V_2)$ - дводольний і для кожної підмножини $A \subset V_1$, позначимо через $\varphi(A) \subset V_2$ множину вершин, які суміжні хоча б з однією вершиною у множині A. Тоді досконала паросполука існує тоді і тільки тоді, коли*

$$|A| \leq |\varphi(A)|$$

Дерева

- Граф без циклів називається ***ациклічним*** (або ***лісом***).
- ***Дерево*** - це зв'язний ациклічний граф. (позначається T_n, де n - кількість вершин)

Дерево можна побудувати шляхом додавання ребер в його вершинах. Найпростіше дерево складається з 2 вершин, з’єднаних ребром. При додаванні чергового ребра, додається ще одна вершина.

- Граф G - ***дерево***, якщо це зв’язний граф і $p=q+1$, де p, q - кількість вершин і ребер відповідно.
- Граф G - ***дерево***, якщо це ациклічний граф такий, що якщо між двома його вершинами провести ребро, у ньому отримується рівно 1 простий цикл.
- Граф G - ***дерево***, якщо будь-які 2 його вершини з’єднані простим ланцюгом.

Дерева

- При введенні в графі операції видалення ребра, причому такої операції, яка не приводить до порушення зв'язності графу, можна отримати при її послідовному застосуванні (поки можливо) ***остовне (каркасне) дерево***.
- Якщо маємо граф з характеристиками (p, q) - граф, то для отримання дерева потрібно видалити $q - p + 1$ ребро. Дане число називається ***циклічним рангом*** графа або ***цикломатичним числом*** $v(G) = q - p + 1$.
- Число $v^*(G) = p - 1$ ребер будь-якого остову графу G називається ***коциклічним рангом*** графа G.

$v(T_n) = 0$, $v(C_n) = 1$ - простий цикл,

$v(G^k) = q - p + k$, $v^*(G^k) = p - k$,

де G^k - незв'язний граф, що складається з k-компонент зв'язності.

Дерева

ТЕОРЕМА. *Центр вільного дерева складається з однієї вершини або з двох суміжних вершин:*

$$z(G) = 0 \quad \& \quad k(G) = 1 \Rightarrow C(G) = K_1 \vee C(G) = K_2.$$

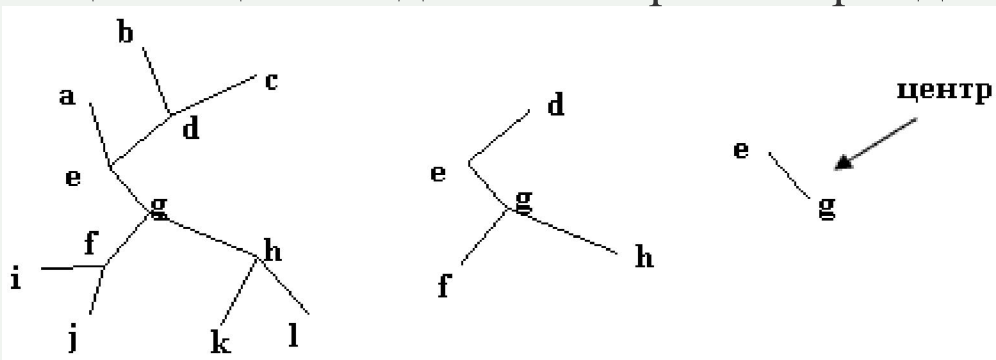
Доведення.

Для дерев K_1 і K_2 твердження теореми очевидно.

Нехай тепер $G(V, E)$ - деяке вільне дерево, відмінне від K_1 і K_2. Розглянемо граф $G'(V', E')$, отриманий з G видаленням всіх висячих вершин. При цьому, якщо G' - дерево, поскільки ацикличність і зв'язність при видаленні висячих вершин зберігається.

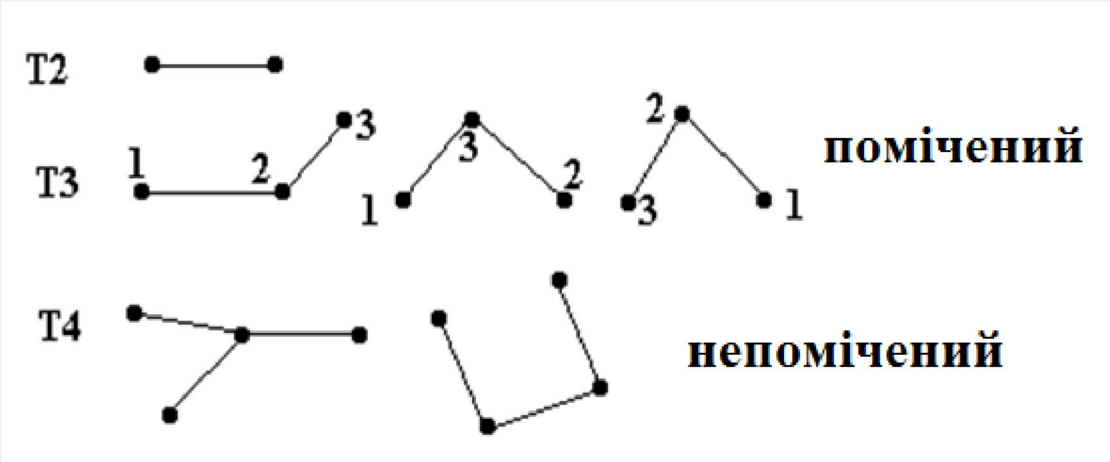
Далі, якщо ексцентриситет $e_G(v) = d(v, u)$, то u - висяча вершина в дереві G. Тому $\forall v \in V' \quad e_G(v) = e_{G'}(v) + 1$ і при видаленні висячих вершин ексцентриситети тих, що залишились, зменшуються на 1. Значить, при видаленні висячих вершин центр не змінюється, $C(G) = C(G')$.

Поскільки дерево G скінченне, то видаляючи на кожному кроці всі висячі вершини, в кінці кінців за декілька кроків прийдемо до K_1 або K_2.



Перерахування дерев

- Теорія перерахування дерев займається розробкою методів підрахунку неізоморфних графів, які мають дані властивості.
- Основне питання теорії перерахування дерев - скільки існує дерев T_n неізоморфних одне одному.
- Дана задача була розв'язана Келлі. Він довів, що всього може бути $\boldsymbol{n^{n-2}}$ помічених неізоморфних дерев. Дана теорія застосовується для розв'язання задач при створенні найкоротшого зв'язаного ланцюга.



Остовне дерево мінімальної ваги

Приклад.

Необхідно побудувати мережу залізничних доріг, що пов'язують n міст таким чином, щоб з кожного міста можна було потрапити в будь-яке інше місто і кількість рельсів при цьому повинно бути мінімальним.

<u>Формальна постановка задачі.</u>

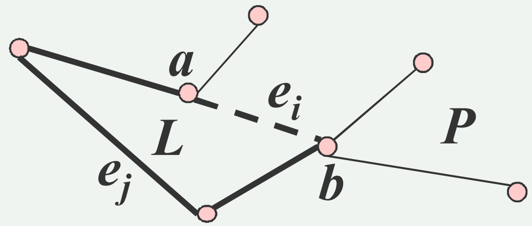
Є n міст a_1, a_2,...,a_n, які потрібно з'єднати мережею доріг. Для кожної пари міст (a_i, a_j) відома вартість будівництва дороги $d(a_i, a_j)$. Потрібно знайти найдешевший варіант будівництва. Тобто, потрібно знайти на зваженому повному графі з n вершин остовне дерево найменшої довжини.

Побудова остову мінімальної ваги

- Алгоритм Г. Штейнгауза:
  - Вибрати довільне місто і з’єднати його з найближчим. Повторити цю дію для всіх останніх міст.
  - Якщо замість єдиного дерева отримаємо ліс, то необхідно вибрати одне з дерев лісу і з’єднати його найкоротшим ребром з іншим деревом. Повторити таке зв’язування дерев, поки не буде отримано одне дерево.
- Алгоритм Краскала
  - З’єднати 2 вершини графа найбільш коротким ребром.
  - Послідовно додавати саме коротке ребро з тих, що залишилися так, щоб не утворювалися цикли.

❑ Остовне дерево графа, побудоване по данному алгоритму називається ***економічним***.

Економічне дерево



ТЕОРЕМА. *Економічне дерево має мінімальну довжину в графі.*

Доведення.

Нехай **P** – остовне дерево мінімальної довжини;
Q – економічне дерево.

Нехай ребра e_1, e_2, ..., e_{n-1} занумеровані в порядку їх приєднання при побудові економічного дерева **Q**, тобто $d(e_k) \le d(e_{k+1})$.

Якщо **P** ≠ **Q**, то є хоча б одне ребро $e_i \notin$ **P**, що з’єднує деякі вершини a і b в графі **Q**.

Нехай $L(a,b)$ – ланцюг графа **P**, що з'єднує у ньому ці вершини a і b.

Якщо ребро e_i додати до графу **P**, отримується цикл. А так як граф **Q** циклів не має, то в отриманому циклі повинно входити одне ребро не з графу **Q** – ребро e_j. Видаливши його з **P**, отримаємо дерево **P'** з тим же числом вершин, що і в графі **P**, довжина якого рівна $d(\mathbf{P'})=d(\mathbf{P})+d(e_i)-d(e_j)$.

Так як граф **P** має найменшу довжину, то $d(e_j) \le d(e_i)$, але e_i – ребро найменшої довжини, при додаванні якого до ребер e_1, e_2, ..., e_{i-1} по алгоритму побудови економічного дерева **Q** не утворюється циклів.

При додаванні ребра e_j до цих ребер також не утворюються цикли, тому $d(e_j)=d(e_i)$, а значить **P'** має також, як і **P**, найменшу довжину і одним спільним ребром більше з економічним деревом **Q**, ніж **P**.

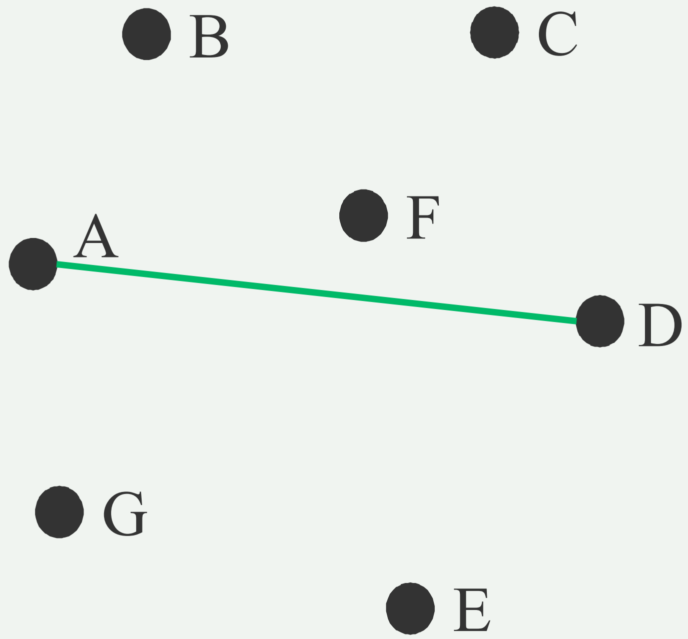
Повторюючи цю операцію кілька раз, одержимо дерево найменшої довжини, що співпадає з економічним деревом.

Приклад побудови остовного дерева (1/7)

	A	**B**	**C**	**D**	**E**	**F**	**G**
A		7		5			
B	7		8	9	7		
C		8			5		
D	5	9			15	6	
E		7	5	15		8	9
F				6	8		11
G					9	11	

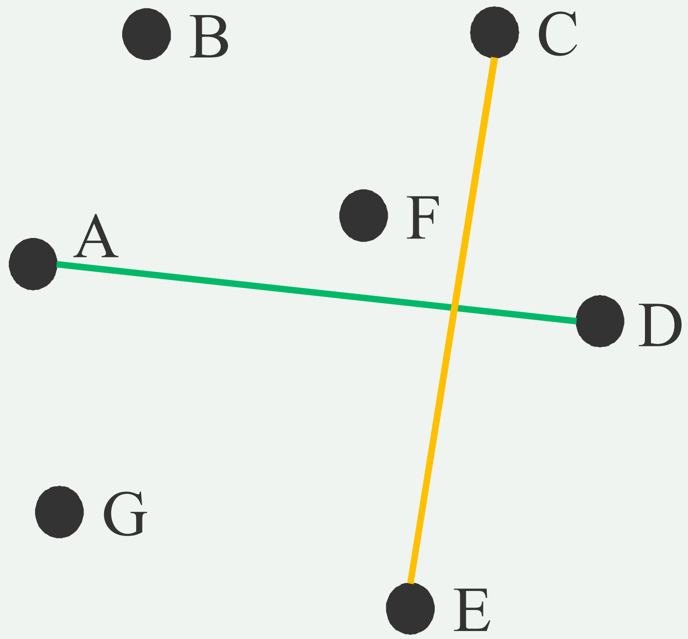
Приклад побудови остовного дерева (2/7)

	A	**B**	**C**	**D**	**E**	**F**	**G**
A		7		5			
B	7		8	9	7		
C		8			5		
D	5	9			15	6	
E		7	5	15		8	9
F				6	8		11
G					9	11	



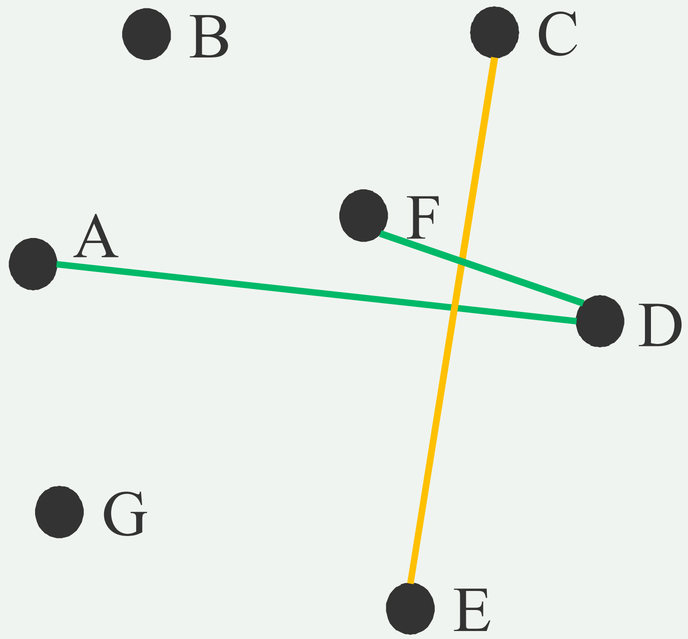
Приклад побудови остовного дерева (3/7)

	A	**B**	**C**	**D**	**E**	**F**	**G**
A		7		5			
B	7		8	9	7		
C		8			5		
D	5	9			15	6	
E		7	5	15		8	9
F				6	8		11
G					9	11	



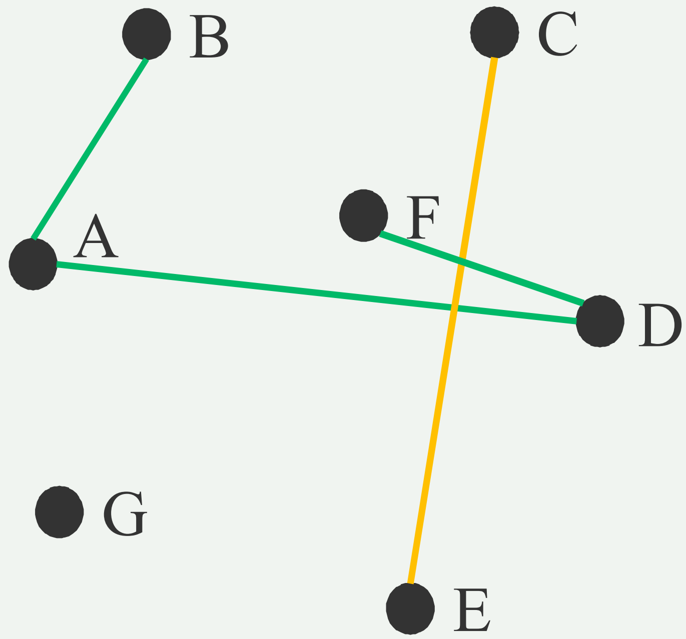
Приклад побудови остовного дерева (4/7)

	A	B	C	D	E	F	G
A		7		5			
B	7		8	9	7		
C		8			5		
D	5	9			15	6	
E		7	5	15		8	9
F				6	8		11
G					9	11	



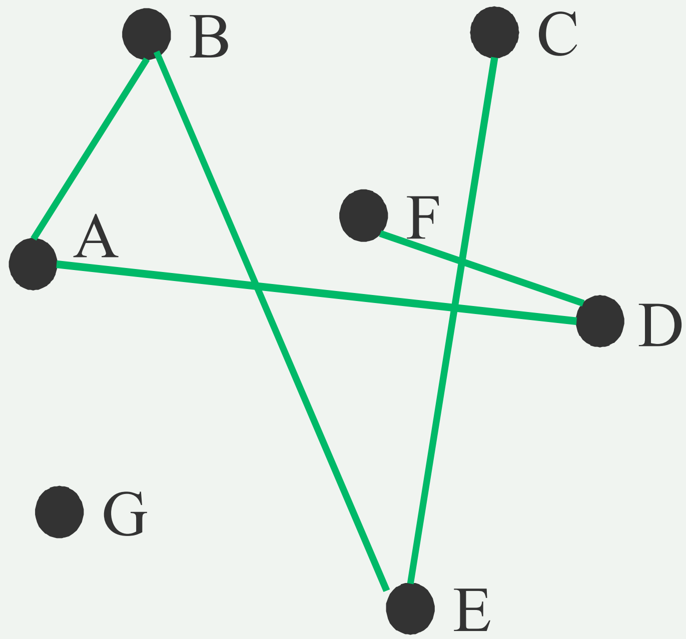
Приклад побудови остовного дерева (5/7)

	A	B	C	D	E	F	G
A		7		5			
B	7		8	9	7		
C		8			5		
D	5	9			15	6	
E		7	5	15		8	9
F				6	8		11
G					9	11	



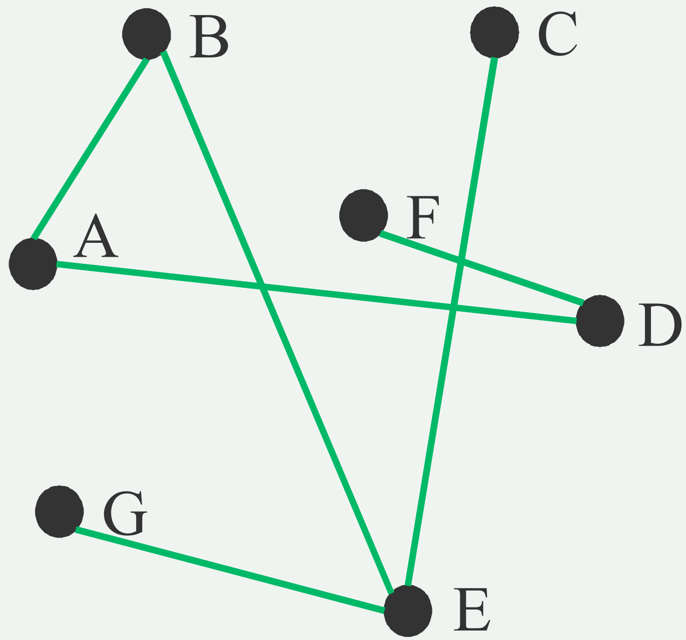
Приклад побудови остовного дерева (6/7)

	A	B	C	D	E	F	G
A		7		5			
B	7		8	9	7		
C		8			5		
D	5	9			15	6	
E		7	5	15		8	9
F				6	8		11
G					9	11	



Приклад побудови остовного дерева (7/7)

	A	B	C	D	E	F	G
A		7		5			
B	7		8	9	7		
C		8			5		
D	5	9			15	6	
E		7	5	15		8	9
F				6	8		11
G					9	11	



- Сумарна довжина дерева - 32

Ейлерові графи

- ***Ейлерів граф*** - це зв'язний граф, якщо у ньому існує замкнутий ланцюг, що проходить рівно один раз через кожне ребро.
- Якщо зняти обмеження на замкненість, то отримаємо ***напів-Ейлерів граф***.

ЛЕМА. *Якщо степінь кожної вершини графа не менше 2, то він містить цикл.*

Лема є очевидною, так як побудова такого маршруту можливе, якщо немає висячих вершин (степінь яких менше 2).

Ейлерові графи

ТЕОРЕМА. *Для зв'язного графа G наступні твердження еквівалентні:*

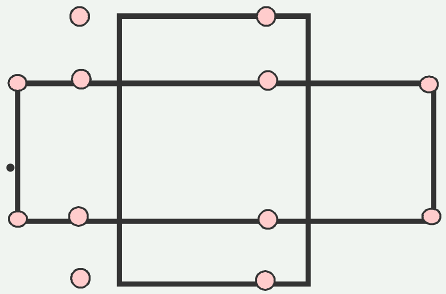
1. *G - ейлерів граф;*
2. *кожна вершина графа G має парну степінь;*
3. *множину ребер графа можна розбити на прості цикли.*

Доведення.

1. Потрібно показати, що з 1) випливає 2).

Так як за визначенням Ейлерова графа - це граф, у якому є маршрут, що містить рівно 1 раз кожне ребро графа, то, розглядаючи даний маршрут, відповідну вершину будемо проходити кілька раз (при обході графа, як мінімум, по 1 разу входимо у вершину і 1 раз виходимо з неї, тобто її степінь кратна 2, тобто парна);

Ейлерові графи



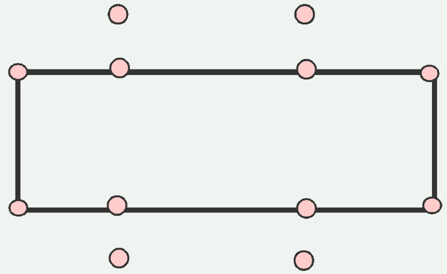
2. Потрібно довести, що з 2) випливає 3).

Так як *G* - зв'язний і нетривіальний,

то степінь кожної вершини не менше двох, а значить, (згідно леми) в графі *G* міститься цикл.

Видалимо цей цикл з графу.

Так як все вершини мають парний степінь, то отриманий граф має ті ж самі властивості.



В цьому графі всі вершини мають

- парну степінь (також 0),
- або є окремими вершинами,
- або існує компонента зв'язності з парним степенем.

Значить, у графі є цикл.

Видалимо цикли з графу, поки увесь граф не буде представлений множиною тривіальних графів;

Ейлерові графи

3. Потрібно довести, що з 3) випливає 1).

Припустимо z_1 - деякий цикл, який належить G.

Якщо G складається тільки з z_1, то еквівалентність 3) $\Rightarrow$ 1) доведена, граф буде Ейлеровим.

В противному випадку, якщо є ще цикл z_2, то ці два простих цикли мають одну спільну вершину. І як би ми не будували маршрут, два рази будемо проходити через цю вершину, тобто це Ейлерів граф.

НАСЛІДОК. *Нехай G - зв'язний граф, у якому не більше двох вершин мають непарні степені. Тоді у графі G існує незамкнений ланцюг, що містить всі ребра, а сам граф – напів-Ейлерів.*

Приклад.

Існує план виставки. Потрібно так розмістити покажчики, щоб відвідувач побував у кожному залі 1 раз і відвідав їх усі.

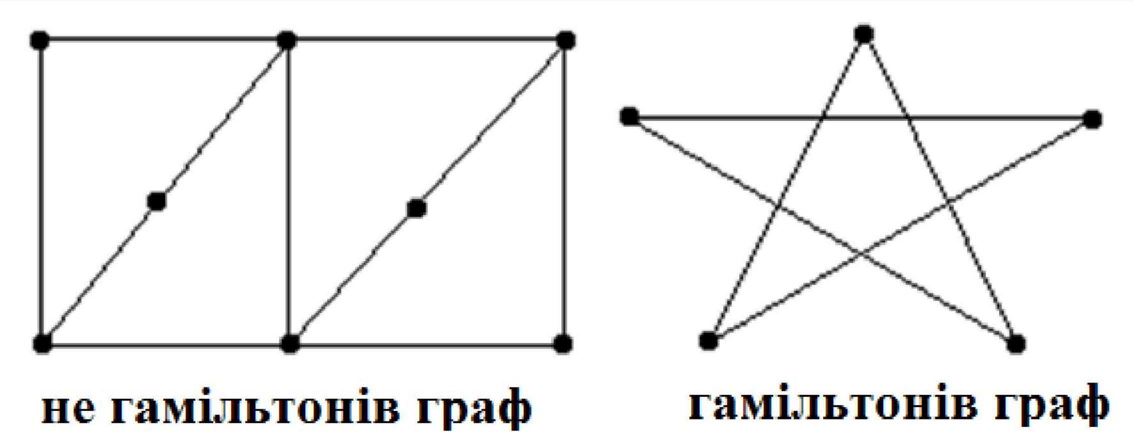
Алгоритм побудови Ейлерового циклу

- Вибираємо довільну вершину.

- Йдемо по ребрам довільним чином, дотримуючись таких принципів:
  - видаляємо ребра по мірі проходження;
  - видаляємо ізольовані вершини;
  - на кожному етапі йдемо по мосту лише тоді, коли немає іншої можливості.

Гамільтонові графи

- ***Гамільтоновий граф*** - це граф, у якому є простий цикл, що містить кожну вершину цього графу.
- Цей *цикл* називають ***гамільтоновим***.
- Простий *ланцюг*, що містить кожну вершину графа, також називають ***гамільтоновим***.



Не існує теореми, яка дає необхідну і достатню умову визначення гамільтонового графу.

Гамільтонові графи

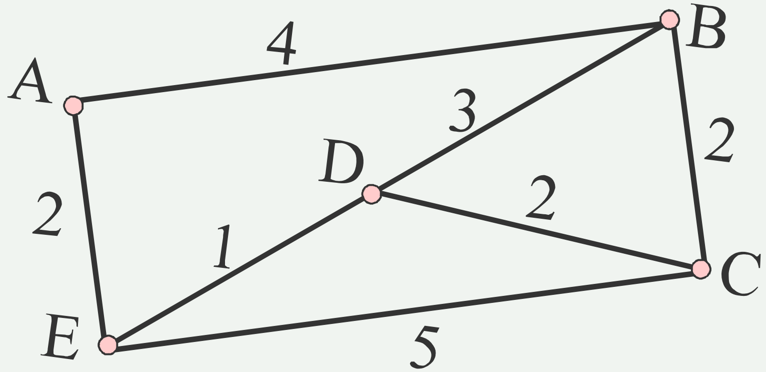
- <u>Достатні умови</u> існування в графі гамільтонового циклу.

1. Граф зі степеневою послідовністю $d_1 \le d_2 \le \ldots \le d_n$ є гамільтоновим, якщо для довільного k, що задовольняє нерівностям $1 \le k < n/2$, істинна імплікація $(d_k \le k) \Rightarrow (d_{n-k} \ge n - k)$;
2. Якщо в G з $p>3$ для будь-якої вершини степінь не менше $p/2$, то це - гамільтоновий граф;
3. Якщо для будь-якої парь несуміжних вершин сума їх степенів більша або рівна p, то це гамільтоновий граф.

Задача комівояжера

- Дано *n* міст, відстані між якими відомі. Комівояжер повинен відвідати всі *n* міст по одному разу, повернувшись у те, з якого відправився. Потрібно знайти такий маршрут руху, при якому сумарний пройдений шлях буде мінімальним.

Очевидно, що це задача відшукання найкоротшого гамільтонового циклу у навантаженому повному графі.



Метод повного перебору

Схема розв’язання задачі комівояжера:

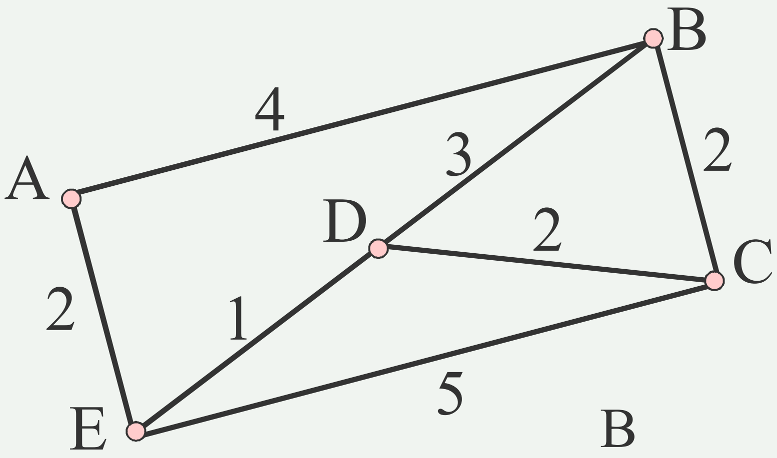
- Згенерувати всі *n!* можливих перестановок вершин повного графу.
- Підрахувати для кожної перестановки довжину маршруту.
- Вибрати з них найкоротший.

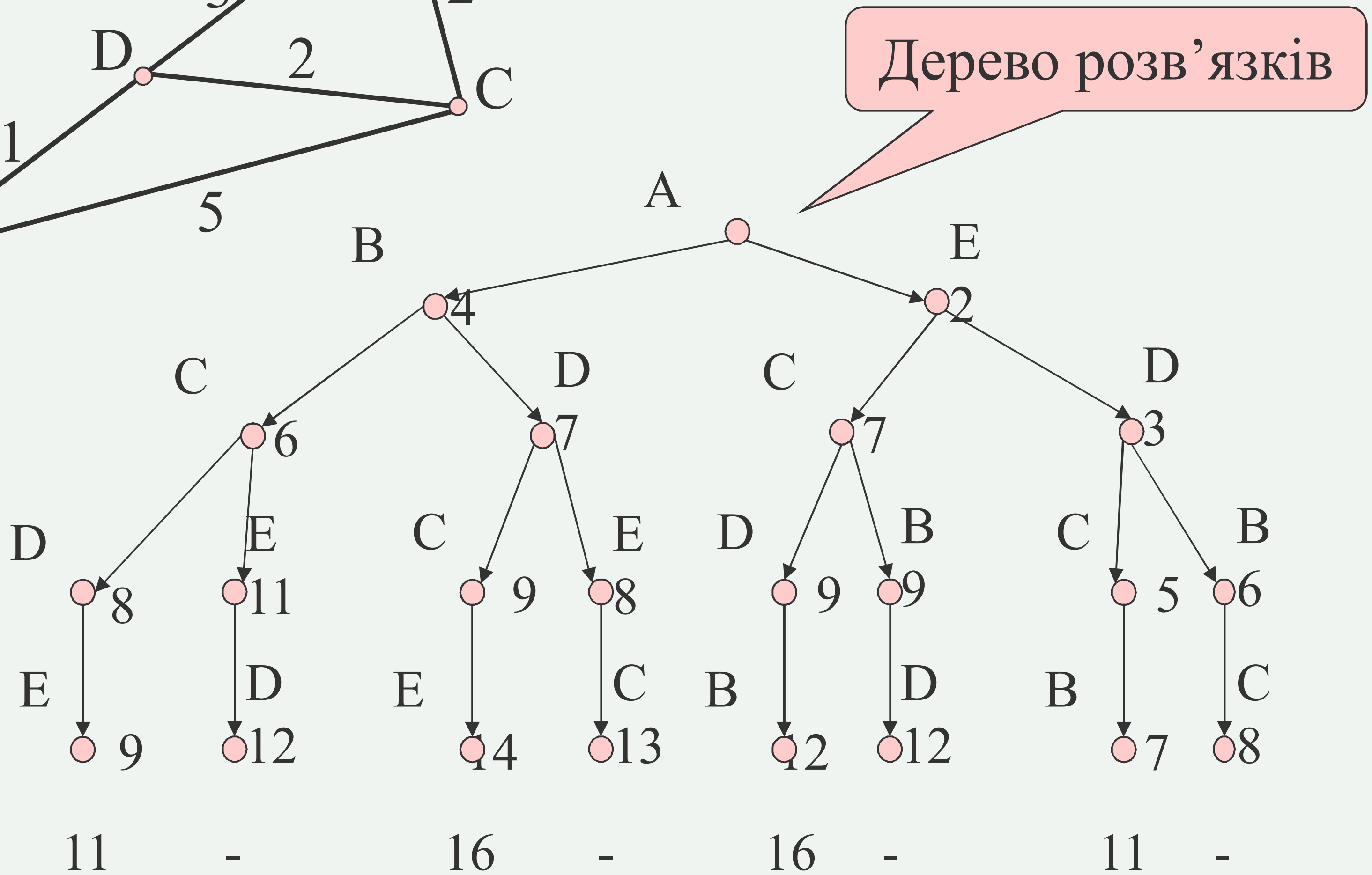
Очевидно, що таке обчислення потребує не менше *O(n!)* кроків.

N! - швидко зростаюча функція. Тобто, розв’язання задачі комівояжера методом *повного перебору* практично нездійсненно навіть для порівняно невеликих *n*.

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
n!	1	2	6	24	120	720	5040	40320	362880	3628800	39916800	479001600	6227020800	87178291200	1,30767E+12

Метод повного перебору





Метод віток і границь

1. Вибрати початкову вершину А графу і присвоїти їй оцінку довжини шляху 0.
2. Провести все вітки з вершини з мінімальною оцінкою.
3. Кожній з утворених вершин присвоїти відповідну оцінку.
4. Чи є серед утворених вершин вершина *A* (початкова вершина)

   Так: 5. Обчислити довжину отриманого циклу L_{m}.

   6. Відкинути всі маршрути, оцінки яких $\geq L_{\mathrm{m}}$.

   Ні: 7. Чи є на графі невід'єднані вершини

   Так: 2.

   Ні: 8. Виписати утворений маршрут L_{m}.

Задача про найкоротший шлях

Дано неорієнтований зважений граф *G(V,E)*. Кожному ребру графа приписано число $d(e) \geq 0$, яке називається довжиною ребра. При цьому будь-який ланцюг $\mu = <v_0, v_k>$ характеризується довжиною: $$d(\mu) = \sum_{e \in \mu} d(e).$$

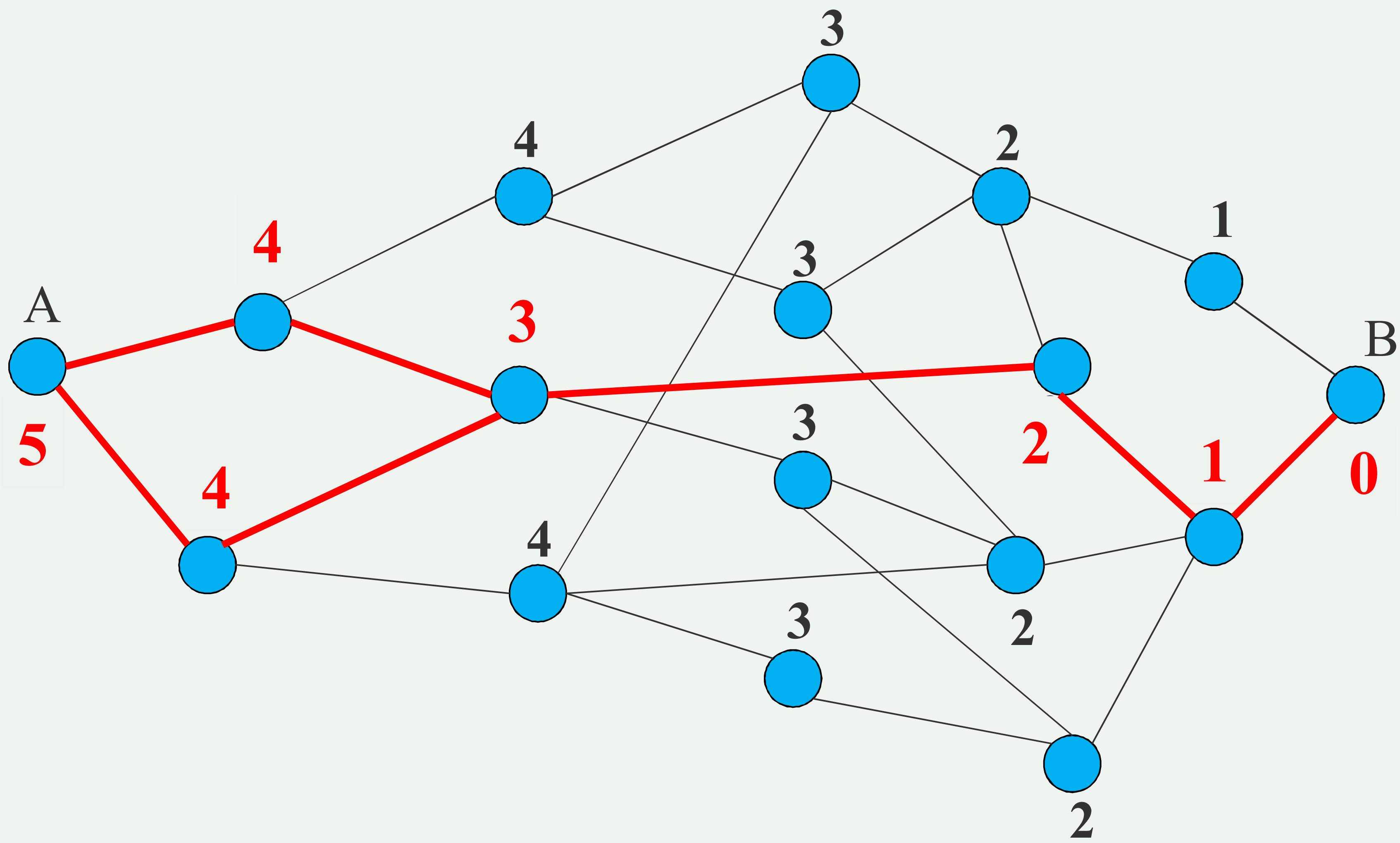
Потрібно знайти для 2 довільних вершин *a* і *b* графу *G* шлях $<a,b>$ такий, щоб його довжина була найменшою.

Найкоротший шлях в графі з ребрами одиничної довжини

<u>Правило</u>: кожній вершині v_i приписується індекс w_i, рівний довжині найкоротшого шляху з даної вершини у кінцеву.

<u>Порядок приписування індексів</u>:

1. Кінцевій вершині *b* приписується індекс $w_b=0$.
2. Всім вершинам, суміжним із кінцевою вершиною *b*, приписується індекс, рівний 1.
3. Всім вершинам, суміжним із поміченими вершинами, приписується індекс на 1 більший, ніж у суміжній поміченій вершині.
4. Процес розмітки (п.3) продовжується до тих пір, поки не буде помічена початкова вершина *a*. Її індекс w_a буде рівний шуканій довжині найкоротшого шляху.
5. Найкоротший шлях визначається при русі з початкової вершини у напрямку зменшення значень індексів.


3
4
2
1
4
3
A
3
B
5
3
2
1
0
4
4
2
3
2

Найкоротший шлях в графі з ребрами довільної довжини

Порядок приписування індексів:

1. Кожна вершина помічається індексом наступним чином:
   - Кінцевій вершині b приписується індекс $w_b = 0$;
   - Всім останнім $v_i – w_i = \infty$.
2. Шукаємо ребро (v_i,v_j), таке, щоб $(w_j - w_i) > d(v_i,v_j)$ і замінюємо індекс w_j індексом $w_j' = w_i + d(v_i,v_j) < w_j$.
3. Процес розмітки (п.3) продовжується до тих пір, поки залишається хоча б 1 ребро (v_i,v_j), для якого можна зменшити w_i.
4. Найкоротший шлях визначається при русі з початкової вершини у напрямку зменшення значень індексів, причому $(w_i - w_j) = d(v_i,v_j)$.

Укладання графу

- ***Жордановою кривою*** називають неперервну спрямлену лінію, яке не має самоперетинів.
- *Граф G* ***укладається*** *у простір* ***S***, якщо існує така *бієкція* вершин і ребер графу *G* відповідно в точки і жорданові криві цього простору, яка зберігає *інцидентність* ребер і вершин графу *G*. Зображений таким чином граф називають ***укладкою графу G у простір S***.

Теорема про укладання графу

ТЕОРЕМА. Кожен граф укладається у тривимірний (евклідовий) простір.

Доведення.

Розмістимо усе вершини графа в різних точках осі OX.

З усіх площин, що проходять через цю вісь, виберемо $|E_G|$ різних площин.

Кожне ребро $e \in E_G$ зображуємо в окремій площині півколом, що сполучає вершини цього ребра.

Таким чином, отримуємо укладання графа в просторі Евкліда, оскільки усі ребра лежать в різних площинах і не перетинаються.

Плоскі графи

- ***Плоским*** називається граф, який розміщений на площині або сфері без перетинів.
- ***Планарним*** називається граф, який ізоморфний плоскому. Це граф, який укладається на площині, тобто має *плоску укладку*.
- ***Гранню*** графу називається максимальна по включенню множина точок площини, кожна пара яких може бути з'єднана жордановою кривою, що не перетинає ребра графа.
- ***Границею грані*** є множина вершин і ребер, приналежних цій грані (або циклу, що її утворює).
- Всякий плоский граф має одну єдину необмежену грань. Ця грань називається ***зовнішньою*** гранню. Останні грані графу називаються ***внутрішніми***.

Властивості планарних графів

1. Всякий планарний граф дозволяє таке плоске укладання, в якій будь-яка обрана вершина (ребро) буде належати зовнішній грані.
2. Граф G, отриманий шляхом злиття 2 вершин (ребер), приналежних різним планарним графам, є планарним. При цьому вершина (ребро) злиття є точкою (ребром) з'єднання графу G.
3. Будь-які 2 вершини, приналежні границі деякої грані плоского графу, можна з'єднати простим ланцюгом довільної довжини так, що вибрана грань розіб'ється на 2 грані.
4. Для будь-якого плоского графу кожна точка площини, що не лежить на ребрі, входить тільки в одну грань.
5. Для будь-якого плоского графу кожна точка ребра, яка не є вершиною, входить тільки в одну грань, якщо це ребро є мостом, і точно в 2 грані, якщо воно не є мостом.

Планарність

ТЕОРЕМА. *Якщо G - зв'язний плоский граф, що має p-вершин і q-ребер і f-граней, то* $\boldsymbol{p + f - q = 2}$.

Доведення.

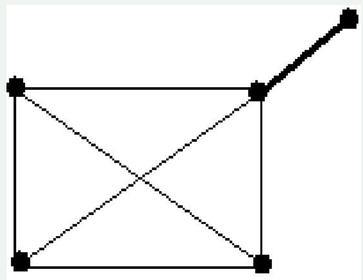
Візьмемо тривіальний граф: $p = 1, q = 0, f = 1$.
Перевіряємо $1 + 1 - 0 = 2$.

Візьмемо граф $G(2,1)$. Перевіряємо: $2 + 1 - 1 = 2$.

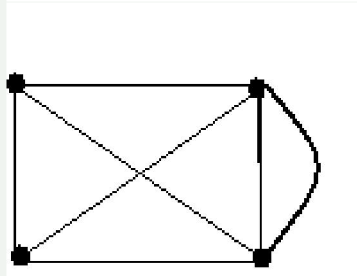
Допустима теорема справедлива для $q = q'$.

Додаємо нове ребро в цей граф, у підсумку отримаємо 2 випадки:

1. Дане ребро не утворює нової грані,
   значить $(p + 1) + f - (q' + 1) = p + f - q$.



2. Дане ребро утворює нову грань,
   значить, не утворюється нова вершина:
   $p + f_{+1} - q_{+1} = 2$.



Ця теорема називається ***формулою Ейлера*** для багатогранників.

Планарність

НАСЛІДКИ.

1. Нехай *G* планарний і у нього *p*-вершин, *q*-ребер, *f*-граней і *k*-компонент зв'язності. Тоді

$$p + f - q - k = 1;$$

2. Якщо *G* - зв'язний планарний граф, що має хоча б 1 цикл непарної довжини, то $q \le 3p - 6;$
   - Для дводольних графів в цьому випадку: $p \le 2p - 4.$
3. Число граней будь-якої плоскої укладки зв’язного планарного (*p,q*)-графу стале і рівне

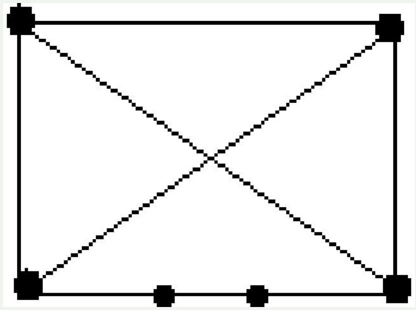
$$q - p + 2.$$

   - Тобто число граней планарного графу не залежить від способу його укладки на площині.

Планарність

ТЕОРЕМА (Понтрягіна - Куратовского).
Граф планарний тоді і тільки тоді, коли він не містить підграфу, який гомеоморфний K_5 і $K_{3,3}$.

- Графи G_1 і G_2 - ***гомеоморфні***, якщо вони обидва можуть бути отримані з одного і того ж графу включенням в його ребра нових вершин степені 2.



Розв'язання задач на планарність необхідні в автоматизованому проектуванні, коли необхідно, наприклад, виконати трасування друкованої плати.

Питання про трасування друкованої плати - це задача отримання розбиття графу на *n*-планарних підграфів.

Алгоритм плоского укладання графу. Основні визначення

Нехай побудовано деяке плоске укладання підграфу $G^1(V^1,E^1)$ графу $G(V,E)$.

- ***Сегментом S відносно G^1*** називають підграф графу G одного з двох видів:
  1. Ребро $e=(v_i,v_j) \in V: v_i,v_j \in V^1, e \notin E^1$;
  2. Зв'язну компоненту графу $(G - G^1)$, доповнену
     - всіма ребрами графу G, інцидентними вершинам взятої компоненти *і*
     - кінцями цих ребер.
- Вершину a сегмента S відносно G^1 називають ***контактною***, якщо $a \in V^1$.
- ***Допустимою гранню*** для сегменту S відносно G^1 називають грань графу G^1, що містить всі контактні вершини сегменту S. Позначимо $\Gamma(S)$ – множину допустимих граней для S (воно може бути порожнім).

Алгоритм плоского укладання графу. Основні визначення

- Простий ланцюг сегменту S, що з'єднує 2 різні контактні вершини, і який не містить інших контактних вершин, називають ***α-ланцюгом***. Всякий *α-ланцюг* сегменту може бути укладений в довільну грань, допустиму для цього сегменту.
- Два сегменти S_1 і S_2 відносно G^1 називають ***конфліктними***, якщо
  1. $Q = \Gamma(S_1) \cap \Gamma(S_2) \neq \varnothing$,
  2. існує 2 *α-ланцюга* з різних сегментів S_1 і S_2, які неможливо без перетинів укласти одночасно в жодну грань $\Gamma \in Q$.

  Інші сегменти не конфліктують.

Алгоритм плоского укладання графу. Основні визначення

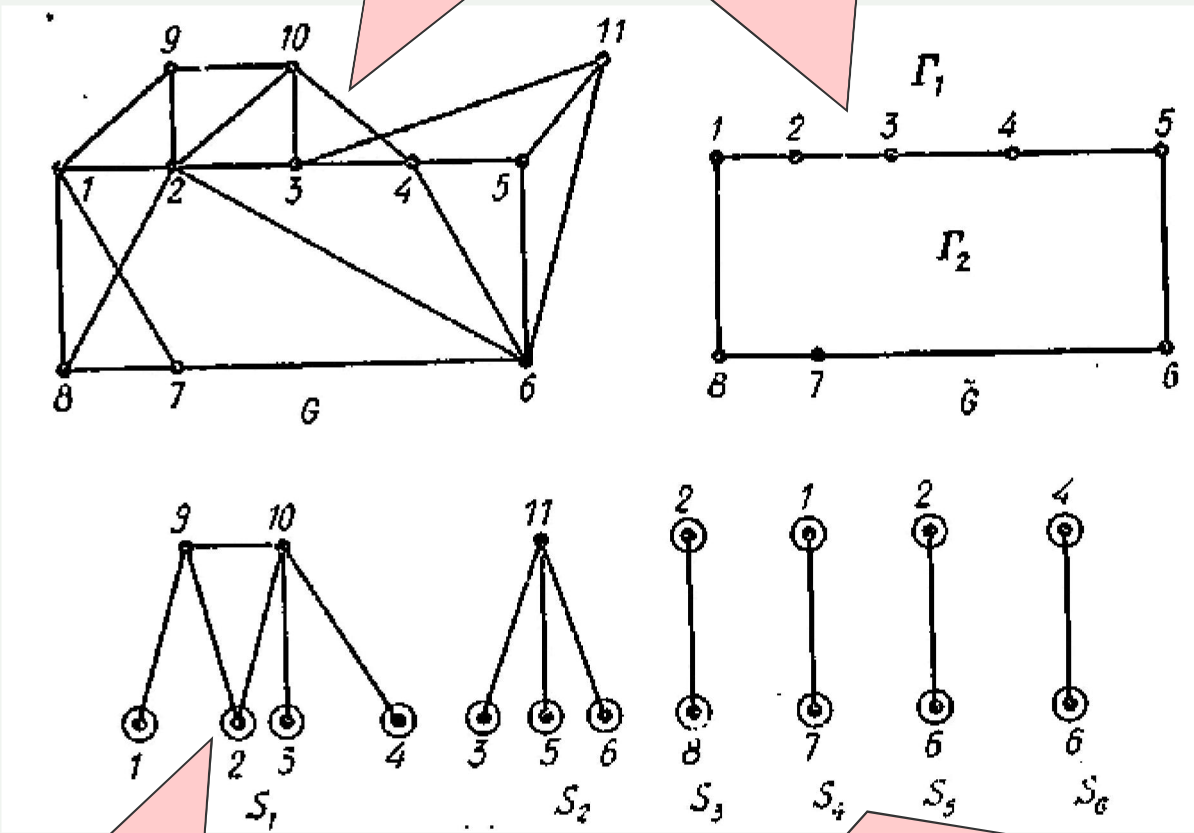
Допустимі грані: $\Gamma(S_i) = \{\Gamma_1, \Gamma_2\}$

α-ланцюга S_1: (1,9,10,4), (1,9,2), (2,9,10,4), (2,10,4), (2,10,3), (3,10,4)

Конфліктні сегменти: S_1 і S_2, S_3 і S_4; S_2 і S_6

Граф $G(V,E)$

Укладений підграф графу G - $G^1(V^1,E^1)$



S_i - сегменти відносно G^1

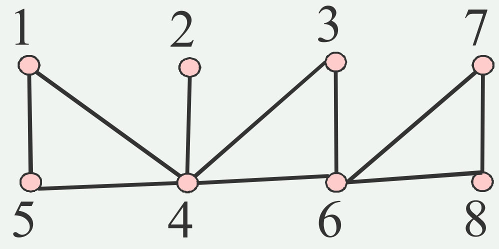
Контактні вершини

γ-алгоритм плоского укладання графу

1. Виберемо деякий простий цикл C з графу G і укладемо його на площині; нехай $G^1 = C$.
2. Знайдемо грані G^1 і сегменти відносно G^1. Якщо множина сегментів пуста, то побудоване плоске укладання графу G є кінцевим.
3. Для кожного сегменту S визначаємо множину $\Gamma(S)$.
4. Якщо існує сегмент S, для якого $\Gamma(S) = \varnothing$, то граф не планарний – закінчуємо; інакше – п.5.
5. Якщо існує сегмент S, для якого $|\,\Gamma(S)\,| = 1$, то п.7.; інакше – п.6.
6. Для деякого сегменту S ($|\,\Gamma(S)\,| > 1$) вибрати довільну допустиму грань Γ.
7. Помістимо довільний *α-ланцюг* сегменту S у вибрану грань Γ, замінимо G^1 на (G^1 + *α-ланцюг*), і перейдемо до п.2.

Внутрішня стійкість

- Підмножина вершин графу $G(V,E)$ називається ***незалежною*** (*внутрішньо стійкою*), якщо жодні 2 вершини з цієї множини не суміжні.



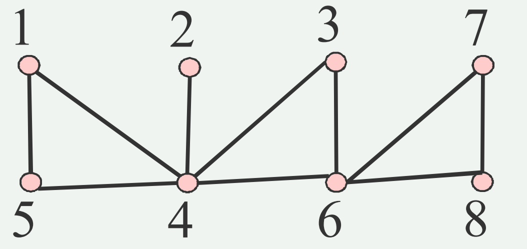
$\{1, 2, 3\}, \{1, 2\}, \{2, 3, 7\}, \{2, 5, 6\}$

Тобто якщо $S \subseteq V$ і S незалежні в G, то підграф $G(S)$ – пустий. Якщо при цьому $S' \subset S$, то S' – також незалежна множина.

- Незалежна множина ***максимальна***, якщо вона не є власною підмножиною деякої іншої незалежної множини. $\{4, 7\}$
- *Найбільш потужна* незалежна множина називається ***найбільшою***. $\{1, 2, 3, 7\}, \{2, 3, 5, 8\}$
- Число вершин в найбільш незалежній множині графу G називається ***числом незалежності*** $\alpha_0(G)$ (*числом внутрішньої стійкості*) графу. $\alpha_0(G) = 4$.

Зовнішня стійкість

- Підмножина V' вершин графу $G(V,E)$ називається ***домінуючою*** (*зовнішньо стійкою*), якщо кожна вершина з множини $V \setminus V'$ суміжна з деякою вершиною з V'.



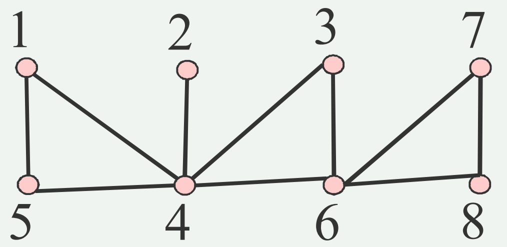
$\{1, 2, 3, 6\}, \{1, 2, 6\}, \{4, 8\}, \{4, 6\}$

- Домінуюча множина ***мінімальна***, якщо ніяка її власна підмножина не є домінуючою. $\{4, 7\}$
- Домінуюча множина з *найменшою потужністю* називається ***найменшою***. $\{4, 6\}, \{4, 7\}, \{4, 8\}$
- Підмножина вершин графу, яка є як *незалежно*, так і *домінуючою,* називається ***ядром*** графу.

$\{1, 2, 3, 7\}, \{1, 2, 3, 8\}, \{2, 3, 5, 7\}, \{2, 3, 5, 8\}, \{4, 7\}$

Покриття

- Вершина і ребро графу ***покривають*** одне одного, якщо вони інцидентні.



Ребро $e = (1,4)$ покриває вершини 1 і 4,
а вершини $1,4$ покриває ребро e.

- Підмножина $V' \subseteq V$ вершин графу $G(V,E)$ називається ***покриттям*** (*вершинним покриттям*, *опорою*) графу G, якщо кожне ребро графу G інцидентне хоча б одній вершині множини V'.

{1, 2, 3, 5, 6, 8}, {4, 5, 6, 8}, {4, 5, 6, 7}

- Покриття графу ***мінімальне***, якщо воно не містить покриття з меншим числом вершин. {1, 2, 3, 5, 6, 8}
- Покриття графу називають ***найменшим***, якщо число вершин у ньому найменше серед всіх покриттів графу.

{4, 5, 6, 8}, {4, 5, 6, 7}

- Число вершин в найменшому покритті називається $\beta_0(G)$ ***числом покриття*** (*числом вершинного покриття*) графу. $\beta_0(G) = 4$

Незалежність і покриття

ТЕОРЕМА. *Підмножина V' вершин графу $G(V,E)$ є найменшим, мінімальним покриттям тоді і тільки тоді, коли $\overline{V'} = V \backslash V'$ – найбільша (максимальна) незалежна множина.*

Значить, $\alpha_0(G) + \beta_0(G) = |\, G \,|$.

ТЕОРЕМА. *Для будь-якого графу G справедлива нерівність:*

$$\alpha_0(G) \geq \sum_{v \in V} \frac{1}{1 + \deg(v)}$$

НАСЛІДОК. *Для будь-якого графу з n вершинами*

$$\alpha_0(G) \geq n \,/\, (1 + d),$$

де d – середнє арифметичне степенів вершин графу.

Побудова незалежної множини

- Алгоритм побудови незалежної множини M, такої що $$|M| \geq \sum_{v \in V} \frac{1}{1 + \deg(v)}.$$

1. Вибрати в графі вершину з мінімальним степенем і занести її в множину M.
2. Видалити з графу G вибрану в п.1 вершину і всі суміжні з нею вершини.
3. Продовжувати процес виконання п.1 і п.2 до тих пір, поки в графі G не залишиться вершин.

Даний алгоритм забезпечує лише наближений розв’язок.

Розфарбовування графу

- ***Розфарбовування графу*** - це таке приписування кольорів вершинам графу, щоб 2 суміжні вершини не були одного кольору.
- ***Однокольоровий клас*** - це множина всіх вершин одного кольору.
- ***Хроматичне число*** - це мінімальне число n, для якого граф має n-розфарбування. Позначення: $\boldsymbol{\chi}$.
- Граф називається n-розфарбовуваним, якщо його хроматичне число $\chi(G) \leq n$.
- Граф n-хроматичний, якщо $\chi(G) = n$.
- Будь-який граф має n-розфарбування, якщо $\chi(G) \leq n \leq p$.
- Для повних графів $\chi = p$, для дводольних $\chi = 2$, для графів-циклів $\chi = 2$, якщо довжина парна і 3, якщо довжина непарна. Для дерева $\chi = 2$.

Розфарбовування графу

- **ТЕОРЕМА.** *χ будь-якого графу задовольняє нерівності $\chi(G) \le 1 + \alpha$, де α - максимальна степінь вершини в графе.*

- **ТЕОРЕМА.**

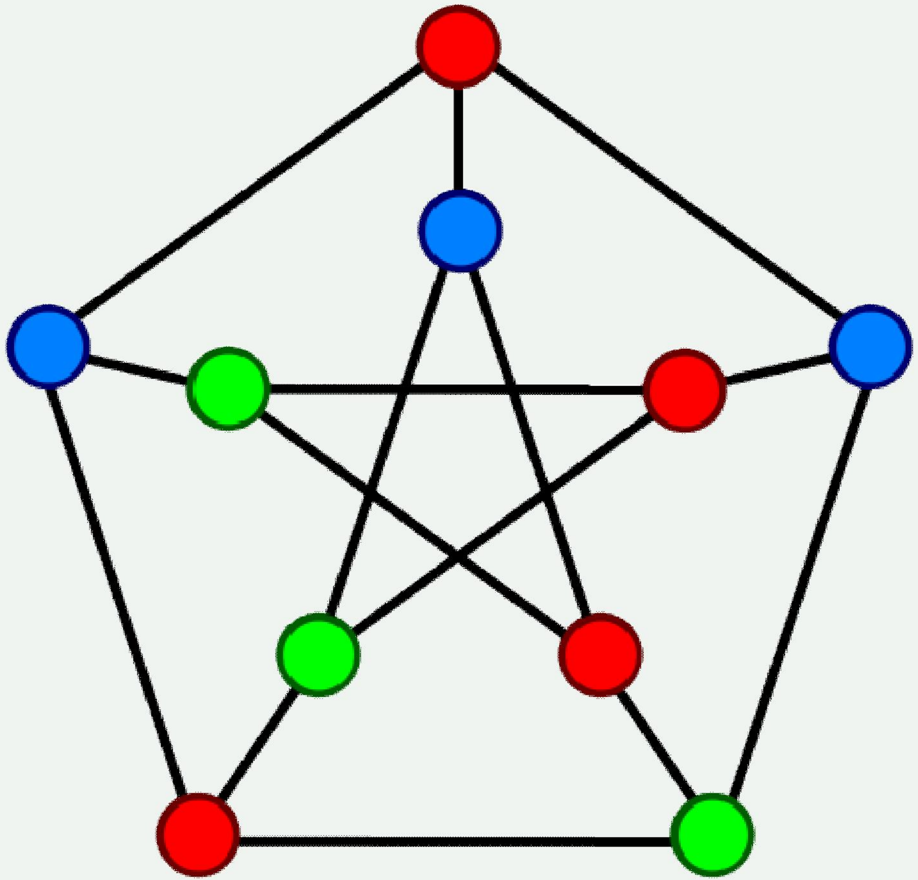
1. *$\chi = \alpha$, якщо граф G не містить в якості компоненти повний граф з кількістю вершин $\alpha + 1$;*
2. *Для $\alpha = 2$, G не містить цикл непарної довжини.*

- Ці теореми доцільно застосовувати, коли степені вершин в графі приблизно рівні. Інакше отримується груба оцінка.

- **ТЕОРЕМА** (о чотирьох фарбах). *Для будь-якого нетривіального зв'язного планарного графу $\chi = 2..4$*

- **НАСЛІДОК.** *Будь-який планарний граф, що не містить цикл непарної довжини має $\chi \le 3$.*
  Для будь-якої карти достатньо чотирьох фарб, щоб її розфарбувати.

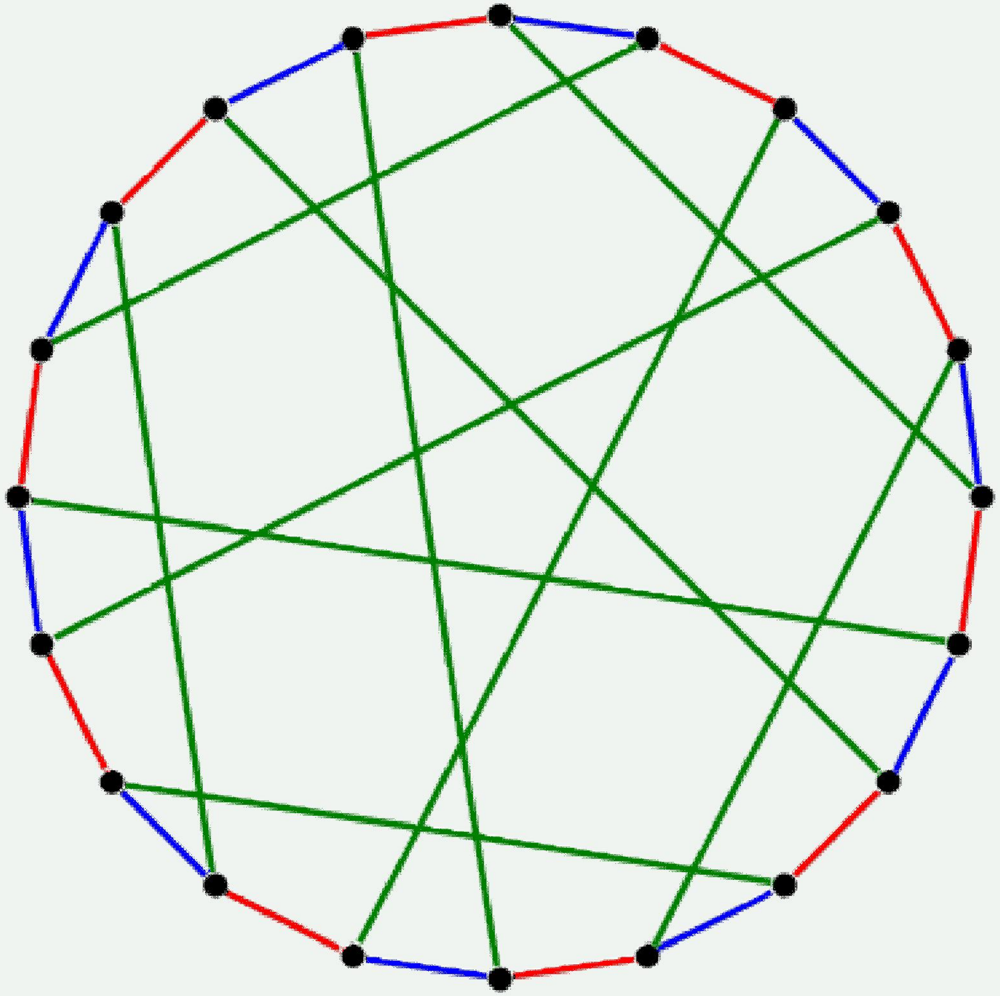
Приклад 3-розфарбування графу Петерсена



Реберне розфарбування

- Граф G - n-реберно розфарбовуваний, якщо необхідно n-фарб, щоб розфарбувати ребра графу таким чином, щоб будь-які 2 інцидентні одній вершині ребра не були одного кольору.
- Якщо G n-реберно розфарбовуваний, то n - його **хроматичний клас** $\chi_l(G)$.
- Для хроматичного класу справедлива нерівність $\alpha \le \chi_l(G) \le \alpha + 1$.
- Для простих циклів $\chi_l = 2$, якщо довжина парна і 3, якщо непарна.
- Для дводольного графу $K_{m,n}$ $\chi_l(K_{m,n}) = max(m, n)$.
- Для повного графу K_p $\chi_l(K_p) = p$, якщо p - непарне і $p \neq 1$; і $p - 1$, якщо p - парне.

Приклад реберного розфарбування



Хроматичний многочлен

- Зв'яжемо з кожним поміченим графом деяку функцію. Розфарбуванням графу G t-кольорами назвемо розфарбування, що використовує не більше t кольорів.

- Два розфарбування *різні*, якщо хоча б одній вершині присвоєні різні кольори.

- Нехай $P(G, h)$ - кількість способів, якими можна розфарбувати граф G n-фарбами. Якщо $\chi > n$, то $P = 0$. Мінімальне n, для якого $P \neq 0$ є χ даного графу.

- Для K_3 $P(K_3, n) = n(n - 1)(n - 2)$.

- Для будь-якого повного графу $P(K_p, n) = c_n^p = n(n - 1)(n - 2)...(n - p + 1)$ - це хроматичний многочлен повного графу.

Властивості хроматичного многочлену

1. Степінь хроматичного многочлену рівна p.
2. Коефіцієнт при K_p рівний 1.
3. Коефіцієнт при $n_{p-1} = -q$, коефіцієнти чергуються по знаку. Мінімальний показник степені n рівний числу компонент в графі G.

- Хроматичний многочлен достатньо повно характеризує граф G, але немає необхідних і достатніх умов визначення того, що многочлен є хроматичним, тобто для нього існує деякий граф.