МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНА МЕТАЛУРГІЙНА АКАДЕМІЯ УКРАЇНИ

В. В. Кузьменко, Г. Г. Швачич, Г. І. Рижанкова, В. М. Пасинков

ОСНОВИ ДИСКРЕТНОЇ МАТЕМАТИКИ

РОЗДІЛ " ЕЛЕМЕНТИ ТЕОРІЇ ГРАФІВ"

Затверджено на засіданні Вченої ради академії
як конспект лекцій

Дніпропетровськ НМетАУ 2004

УДК 510.6

Кузьменко В. В., Швачич Г. Г, Рижанкова Г. І., Пасинков В. М. Основи дискретної математики. Розділ “Елементи теорії графів”: Конспект лекцій. – Дніпропетровськ: НМетАУ, 2004. – 38с.

Викладені основні поняття теорії графів в контексті сучасних концепцій дискретної математики та основ програмування. Виникнення теорії графів пов’язане з іменем Л. Ейлера. Переважно зусиллями Кіргофа та Келлі були одержані нові результати в теорії графів. Інтенсивний розвиток припадає на останні 50 – 60 років, він завдячує застосуванню графів у теорії автоматів, теорії проектування, економіці, хімії, біології і т.д.

Призначений для студентів спеціальності 7.080401 – інформаційні управляючі системи та технології.

Іл. 29. Бібліогр.: 17 найм.

Відповідальний за випуск Г. Г. Швачич, канд. техн. наук, проф.

Рецензенти: Б. І. Мороз., д-р техн. наук, проф. (Академія митної Служби України)

Д. Г. Зеленцов., канд. техн. наук, доц. (УДХТУ)

ТЕМА 1 ОЗНАЧЕННЯ ГРАФІВ, РІЗНОВИДИ ГРАФІВ

Означення неорієнтованого графа

Нехай **V** – деяка непуста множина. Позначимо $\mathbf{V^{(2)}}$ – множину всіх неупорядкованих різних двоелементних підмножин множини **V**, а $\mathbf{MV^{(2)}}$ – мультимножину множини $\mathbf{V^{(2)}}$, тобто в $\mathbf{MV^{(2)}}$ можуть входити однакові пари елементів із **V**, причому цих пар може бути скільки завгодно. Декартовий квадрат множини V позначимо $\mathbf{V^2}$.

Неорієнтованим мультиграфом G називається пара (V, E), де $E \subseteq MV^{(2)}$. Елементи множини V називаються вершинами, а елементи множини E – ребрами. Ребра позначаються парами (u, v), де u, v – вершина з V.

Мультиграф G = (V, E) називається **неорієнтованим графом**, якщо $E \subseteq V^{(2)}$. Всякий граф є мультиграфом, але не всякий мультиграф буде графом. Якщо G = (V, E) – мультиграф, то E може мати кілька ребер, які з'єднують одні і ті ж вершини u і v. Такі ребра називаються кратними ребрами. Отже, граф є мультиграфом, в якого кратність кожного ребра дорівнює одиниці.

Мультиграф називається **скінченим**, якщо множини V і E скінчені. Для скінченого графа, достатньо лише скінченності множин його вершин V, оскільки скінченність V дає скінченність $V^{(2)}$, тобто граф називається скінченим, якщо скінчена множина його вершин. Скінчений граф з n вершинами називається графом n-го порядку.

Інколи розглядають графи, які мають ребра (u, u). Таке ребро називається петлею, а мультиграф, який має петлі – псевдографом.

Дві вершини u і v графа G = (V, E) **суміжні**, якщо $(u, v) \in E$, і несуміжні – в протилежному випадку. Якщо $(u, v) \in E$, то вершини u, і v називаються кінцями ребра (u, v). Множина вершин графа, суміжних з деякою вершиною u, позначається Cm(u).

З наведених означень також випливає, що різниця між графом і мультиграфом полягає в тому, що дві вершини в графі можуть бути з'єднані не більш ніж одним ребром.

Два ребра називаються **суміжними**, якщо вони мають спільний кінець. Відношення суміжності як для вершин, так і для ребер є симетричним відношенням.

Вершина u і ребро e називаються **інцидентними**, якщо u є кінцем ребра e, і неінцидентним – в протилежному випадку.

Степенем n(u) **вершини графа** G називається число інцидентних їй ребер. Вершина степені 0 називається ізольованою, а вершина степені 1 – висячою, або кінцевою. Ребро, інцидентне кінцеві вершині, також називають кінцевим.

Лема. Сума степенів всіх вершин графа є парним числом.

Кожне ребро вносить у суму всіх вершин графа число 2, тобто

$$\sum_{v \in V} n(v) = 2|E|.$$

Наслідок. У будь-якому графі число вершин непарного степеня парне

Як показано на малюнку 1, графи зручно зображувати на площині або в просторі у вигляді діаграм, які складаються з точок і відрізків, що з'єднують точки. Точки ототожнюються з вершинами, а відрізки з ребрами.

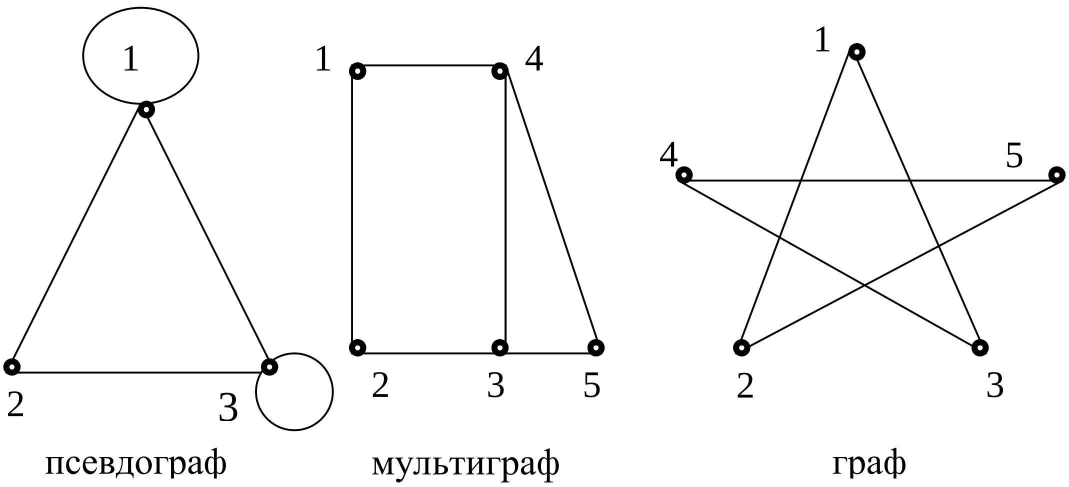
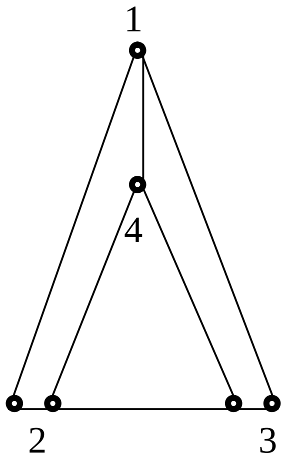


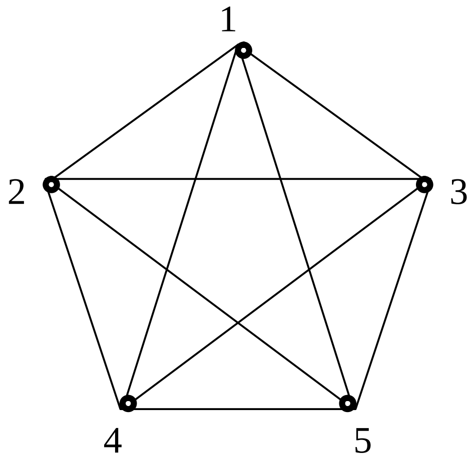
Рис.1

Різновиди графів

Граф, будь-які дві вершини котрого суміжні, називається повним графом. Отже, якщо G = (V, E) – повний граф, то $E = V^{(2)}$. Повний граф з n вершинами позначається K_n. На малюнку 2 зображені графи K_4 і K_5.



Граф K_4

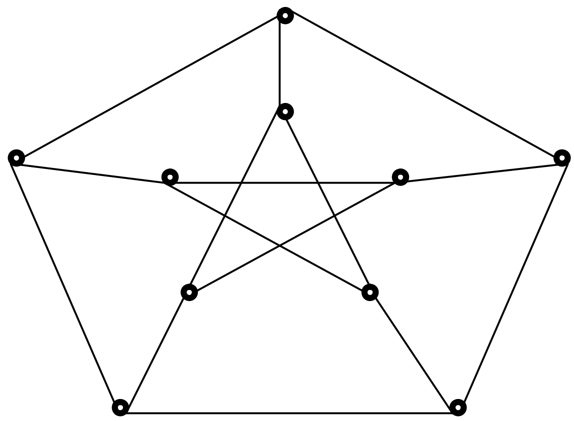


Граф K_5

Рис.2

Регулярні графи

Більш загальними, ніж повні є регулярні графи. Граф називається регулярним, або однорідним, якщо його вершини мають один і той же степінь. Якщо степінь кожної вершини дорівнює k, то граф називається регулярним графом степеня k. Отже, повний граф n-го порядку є регулярним графом степеня n – 1. Регулярні графи степеня 3 (малюнок 3) називають також кубічними, або тривалентними графами.



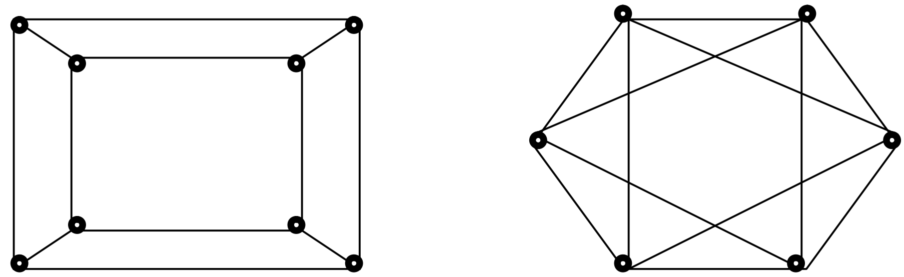
Граф Петерсона   Рис.3

Повністю незв'язані графи

Граф, множина ребер якого пуста, називається повністю незв'язаним або пустим. Незв'язаний граф з n вершинами позначається N_n. Кожний пустий граф є регулярним графом степеня 0.

Платонові графи

Платоновими графами називаються графи, утворені вершинами і ребрами п’яти правильних многогранників – платонових тіл: тетраедра, куба, октаедра, додекаедра та ікосаедра. Графи, зображені на малюнку 4, відповідають тетраедру, кубу і октаедру.



Граф, що відповідає кубу

Граф, що відповідає октаедру

Рис.4

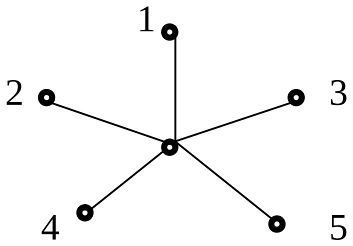
Двочастинні графи

Граф називається двочастинним, якщо існує таке розбиття множин його вершин на два класи, при якому кінці кожного ребра лежать у різних класах. Означення двочастинного графа можна подати іншим чином – в термінах розфарбування його вершин двома кольорами, наприклад червоним і синім. При цьому граф називається двочастинним, якщо його вершину можна пофарбувати синім або червоним кольором, так щоб кожне його ребро мало один кінець червоний, а другий – синій.

Якщо в двочастинному графі будь-які дві вершини з різних класів суміжні, то такий граф називається повним двочастинним графом. Повний двочастинний граф, в якого один клас має m вершин, а другий – n вершин, позначається K_{mn}.

Повний двочастинний граф (малюнок 5) K_{1n} називається зірковим графом.

Аналогічно можна ввести k-частинні графи. Граф називається k-частинним, якщо існує таке розбиття множин його вершин на k класів, при якому будь-яке ребро графа з’єднує дві вершини з різних класів.



Повний двочастинний граф K_5

Рис.5

Орієнтовані графи

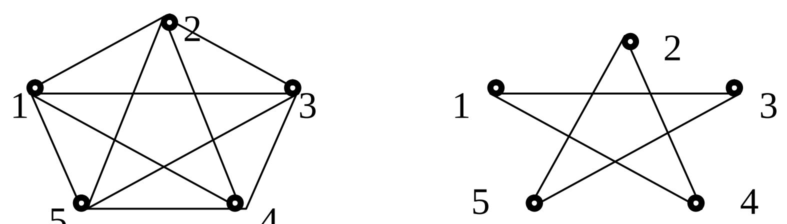
Граф G = (V, E) називається орієнтованим графом (орграфом) якщо $E \subseteq V^2$, тобто вершини всіх його ребер упорядковані. Якщо $(u, v) \in E$, то вершину u називають початковою вершиною, а v – кінцевою вершиною ребра. Орграфи зображуються так як і графи, з тією лише різницею, що їх ребра позначаються стрілками, які ведуть з початкової вершини в кінцеву.

Граф G = (V, E) називається орієнтованим мультиграфом, якщо $E \subseteq MV^2$ і кожне його ребро упорядковане, тобто вказано, яка вершина перша, а яка друга. Отже, в орграфах ребра (u, v) і (v, u) різні

Ізоморфізм графів. Підграфи

Нехай G = (V, E), H = (V_1, E_1) – графи і h: $V \to V_1$ – взаємооднозначна відповідність (тобто $|V| = |V_1|$). Відображення h називається ізоморфізмом графів G і H, якщо для будь-яких вершин u і v графа G їх образи h(u) і h(v) суміжні в графі H тоді і тільки тоді, коли u і v суміжні в графі G. Якщо таке відображення h існує, то графи G і H називаються ізоморфними. Відношення ізоморфізму графів є відношенням еквівалентності. Ізоморфні графи як правило не розрізняються.

Граф H = $(V^{|}, E^{|})$ (малюнок 6), називається підграфом графа G = (V, E), якщо $V^{|} \subseteq V$ і $E^{|} \subseteq E$. Якщо H підграф графа G, то H знаходиться в графі G. Підграф H графа G називається остовним підграфом, коли $V^{|} = V$.



Граф G

Підграф H остовний для G

Рис.6

ТЕМА 2 ОПЕРАЦІЇ НАД ГРАФАМИ

Операція вилучення ребра

Нехай $G = (V, E)$ – граф і $\mathbf{e} \in E$ деяке його ребро. Граф G_1 одержано з графа G внаслідок операції вилучення ребра e якщо $G_1 = (V, E \setminus \{e\})$. Кінці ребра e не вилучаються з множини V.

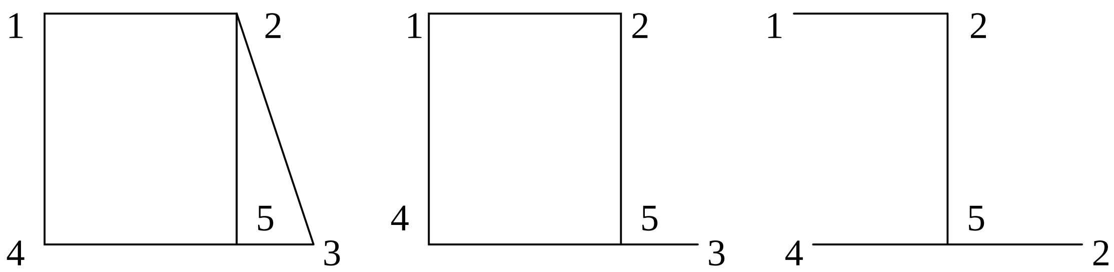
Неважко показати, що для довільних ребер **e** і $\mathbf{e_1}$ графа G виконується така тотожність: $(G - e) - e_1 = (G - e_1) - e$.

Оскільки виконується тотожність $(A \setminus B) \setminus C = A \setminus (B \cup C)$, то маємо

$$G_1 = G - e = (V, E \setminus \{e\}),$$

$(G - e) - e_1 = G_1 - e_1 = (V, (E \setminus \{e\} \setminus \{e_1\}) = (V, E \setminus (\{e\} \cup \{e_1\})) = (V, E \setminus (\{e_1\} \cup \{e\})) = (V, (E \setminus \{e_1\}) \setminus \{e\}) = (G - e_1) - e$.

Отже, якщо виконується підряд кілька операцій вилучення ребра (малюнок 7), то результат не залежить від порядку вилучення ребер з графа.



a) граф G; b) граф $G \setminus \{2, 3\}$; c) граф $(G \setminus \{(2, 3)\}) \setminus \{(1, 4)\}$.

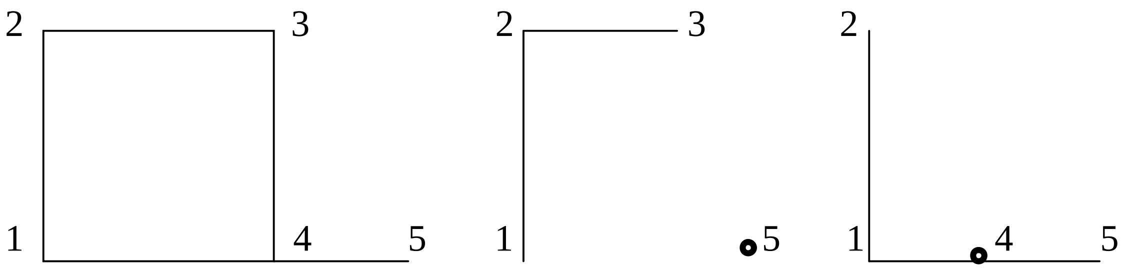
Вилучення ребра

Рис.7

Операція вилучення вершини

Нехай $G = (V, E)$ і $v \in V$. Граф $G_1 = G - v$ одержали з графа G внаслідок операції вилучення вершини v, якщо вершина v вилучена з V, а з E вилучені всі ребра, інцидентні з вершиною v.

Операція вилучення вершини (малюнок 8) не залежить від порядку, в якому вилучаються вершини з графа



a) граф G; b) граф G – {4} c) граф G – {3}

Вилучення вершини

Рис.8

Операції вилучення ребра, вершини і перехід до підграфа – це операції, за допомогою яких можна з початкового графа одержати графи з меншим числом вершин і ребер.

Операція введення ребра

Якщо u, v $\in$ V і (u, v) $\notin$ E в графі G = (V, E), то граф G + e = (V, E $\cup$ $\cup$\{e\}), де e = (u, v).

Внаслідок комутативності операції об'єднання множин можна стверджувати, що послідовність операцій введення ребер у граф G не залежить від порядку, у якому ці ребра вводяться в граф G. Справедлива тотожність

$\forall e, e_1 \in E \qquad ((G + e) + e_1 = (G + e_1) + e)$.

Операція введення вершини в ребро

Нехай (u, v) – деяке ребро графа G. Введенням вершини w в ребро (u, v) називається операція внаслідок якої одержимо два ребра (u, w) і (w, v), а ребро (u, v) при цьому вилучається з графа G.

Операція об'єднання графів

Граф F називається об'єднанням графів G = (V, E) і H = (V_1 E_1), якщо F = ($V \cup V_1$, $E \cup E_1$). Граф F позначається $G \cup H$. Об'єднання графів F = =$G\cup H$ називається диз'юнктивним, якщо $V \cap V_1 = \varnothing$.

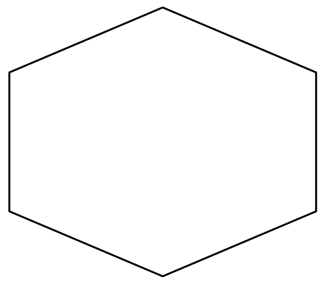
Безпосередньо з означення операції випливає, що ($\forall$ G, H) ($G \cup H$ = =$H \cup G$).

Операція диз'юнктивного об'єднання графів дає можливість ввести до розгляду ще один важливий тип графів.

Граф називається **зв'язним**, якщо його не можна подати у вигляді диз'юнктивного об'єднання двох підграфів, і **незв'язним** – у протилежному випадку.

Отже, всякий незв'язний граф можна зобразити у вигляді диз'юнктивного об'єднання скінченого числа зв'язних підграфів. Кожний з таких зв'язних підграфів називається **компонентом зв'язності**.

Зв'язний регулярний граф степеня 2 називається **циклічним графом** (малюнок 9). Циклічний граф з n вершинами позначається C_n.

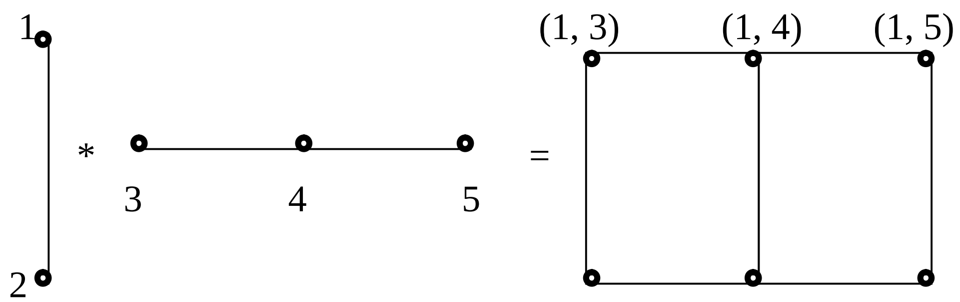


Циклічний граф

Рис.9

Добуток графів

Добутком графів G = (V, E) і H = (V_1, E_1) (малюнок 10), називається граф F = G * H, у якого V = V * V_1, а E визначається таким чином: вершини (u, u_1) і (v, v_1) суміжні в F тоді і тільки тоді, коли u = v, а u_1 і v_1 суміжні в H або u_1 = v_1, а u і v суміжні в G.



Добуток графів

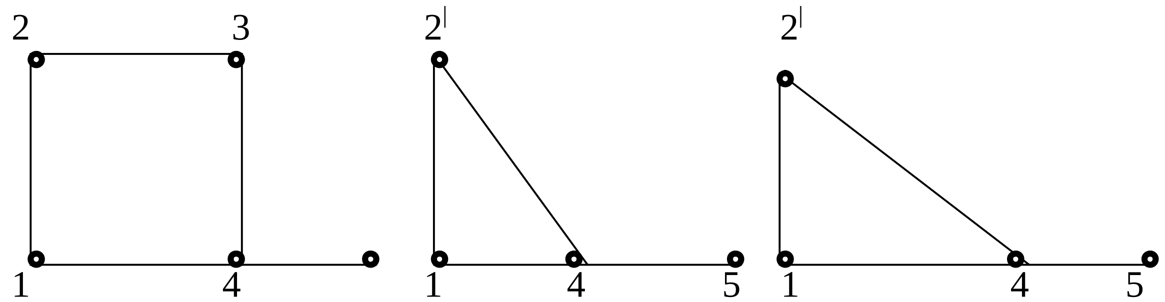
Рис.10

Ототожнення (злиття) вершин

Якщо G = (V, E) – граф, u, v – дві його вершини і Cm(u) = $\{u_1, \ldots u_k\}$, Cm(v) = $\{v_1, \ldots v_l\}$, то граф H = G – u – v, одержаний приєднанням нової вершини $u^|$ до множини вершин H і множини ребер $(u^|, u_i)$, $(u^|, v_j)$ (i = 1, 2, …k, j = 1, 2, … l) до множини ребер H, називається графом, одержаним із G ототожненням вершин u і v.

Операція стягування ребра

Операція стягування ребра (u, v) в графі G = (V, E) (малюнок 11), означає ототожнення вершин u і v в графі G. Операція стягування ребра дозволяє ввести таке поняття. Граф G називається графом, який стягується до графа H, якщо H можна одержати з G за допомогою деякої послідовності операцій стягування ребра. легко помітити, наприклад, що граф Петерсена стягується до графа K_5 де n < 5. Очевидно також що всякий непустий зв’язний граф стягується до K_3. Наприклад, простий ланцюг P_n не стягується до K_3. Логічно ввести параметр $\lambda(G)$ – максимум порядків повних графів, до яких стягується граф G. Параметр $\lambda(G)$ називається числом Хадвігера графа G.

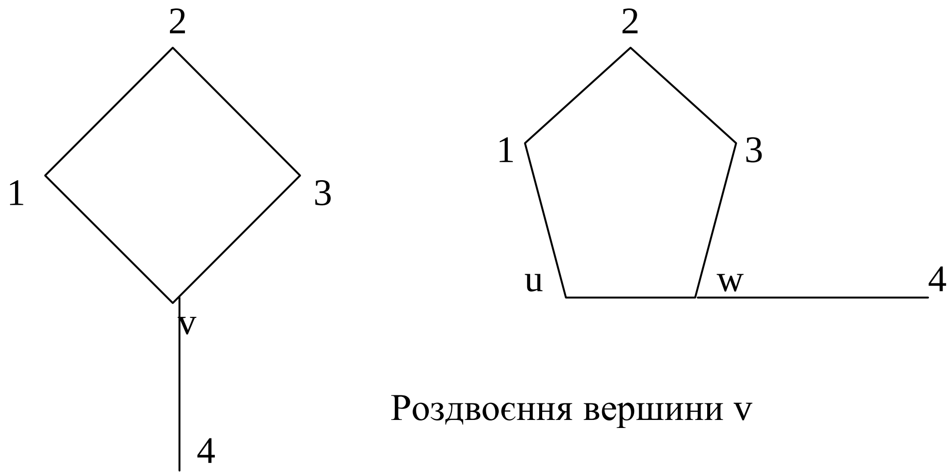


а) граф G;   b) ототожнення вершин 2 і 3 в G;   c) стягування ребра (2, 3) в G.

Рис.11

Операція роздвоєння (розщеплення) вершин

Нехай V – деяка з вершин графа G. Розіб’ємо множину суміжних з нею вершин на дві частини M і P, а потім виконаємо таке перетворення графа G: вилучимо вершину v разом з інцидентними їй ребрами і введемо дві нові вершини, вершини u і w разом з ребром, яке з’єднає ці вершини, вершину u з’єднаємо ребром з кожною вершиною множини M, а вершину w – з кожною вершиною множини P. Одержаний граф позначимо G^{-} і будемо вважати, що він одержаний з графа G внаслідок роздвоювання (розщеплення) вершини v (малюнок 12).



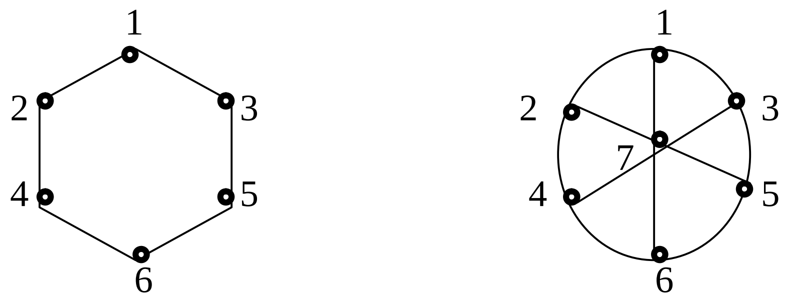
Роздвоєння вершини v

Рис.12

Операція з'єднання графів

Нехай $G = (V, E)$ і $G_1 = (V_1, E_1)$ – два графи у яких множини вершин V і V_1 не перетинаються, тобто $V \cap V_1 = \varnothing$. Операція з'єднання графів G і G_1 полягає в тому, що множини V і V_1 об'єднуються, а потім з'єднуються ребрами кожна вершина графа G з кожною вершиною G_1. Якщо E^u означає множину ребер, яка одержана об'єднанням множин E і E_1 разом з утвореними новими ребрами, то $G = G + G_1 = (V \cup V_1, E^u)$.

Операція з'єднання графів (малюнок 13), може бути виражена у вигляді добутку (суперпозиції) операції об'єднання графів G і G_1 та послідовності операцій введення ребра.



a) граф C_6; b) граф W_7

З'єднання графів

Рис 13.

Тема 3 Властивості графів

Маршрути, цикли, зв'язність

Маршрутом у заданому графі G = (V, E) називається скінчена послідовність його ребер, яка має вигляд (v_0, v_1), (v_1, v_2), … (v_{k-1}, v_k). Число K ребер маршруту називається довжиною цього маршруту.

Маршрут називається ланцюгом, якщо всі його ребра різні, і простим ланцюгом, якщо всі його вершини, крім можливо першої і останньої – різні

Циклом, називається циклічний ланцюг, а простим циклом – простий циклічний ланцюг.

Граф називається **антициклічним, або лісом**, якщо в ньому відсутні цикли. Безпосередньо з означення маршруту і циклу випливають такі твердження:

- будь-який маршрут, котрий з'єднує дві вершини графа, має простий ланцюг, що з'єднує ці вершини;
- будь-який цикл в графі завжди має простий цикл.

Граф називається зв'язним, якщо дві його вершини зв'язані маршрутом. Зв'язний підграф H графа G називається **максимальним**, якщо H не міститься в жодному зв'язному підграфі графа G.

Граф зв'язний тоді і тільки тоді, коли його не можна представити у вигляді диз'юнктивного об'єднання двох графів. Максимальний зв'язний підграф називається **компонентом зв'язності**.

Властивості регулярних графів

Нехай G – регулярний граф степеня k. Степінь регулярного графа позначається deg(G). Очевидним наслідком означення регулярного графа є такі твердження:

- повний граф є регулярним графом;
- не існує регулярного графа G = (V, E) з n вершинами степеня k, у якого k і n непарні.

Дійсно, якщо n і k – непарні числа, то їх добуток k*n теж непарне число, але n*k = 2|E|.

Якщо $G = (V \cup V_1, E)$ – непустий регулярний двочастинний граф, тоді $|V| = |V_1|$, де V, V_1 – класи розбиття множин вершин графа G.

Властивості двочастинних та зв'язних графів

Існує простий критерій двочастинності графа, який виражається в термінах довжини циклів. Граф $G = (V, E)$ є двочастинним тоді і тільки тоді, коли він не має циклів непарної довжини.

Визначаючи спосіб розпізнавання двочастинності графа, будемо приписувати номери 0 та 1 вершинам графа G:

- починаючи з довільної вершину u графа G, приписуємо їй номер 0;
- кожній вершині з множини Cm(u) приписуємо номер 1;
- для всіх вершин, суміжних з вершинами множини Cm(u) приписуємо номер 0;
- для всіх вершин, яким приписали номер 0, знаходимо всі суміжні з ними вершини і приписуємо їм номер 1 і т. д.

Після того, як всі вершини будуть перенумеровані таким чином, будуємо дві множини V_0 і V_1, до яких входять всі вершини з номерами відповідно 0 і 1. Якщо графи $G_1 = (V_0, E_0)$ і $G_2 = (V_1, E_1)$ пусті, то граф G двочастинний, а якщо ні (тобто E_0 або E_1 непусті), то граф G не є двочастинним.

Визначимо тепер одну з властивостей зв'язних графів. Будь який граф $G = (V, E)$ єдиним способом подається у вигляді диз'юнктивного об'єднання своїх компонентів зв'язності.

Якщо граф G – граф з n вершинами і k компонентами зв'язності, то число його ребер задовольняє нерівності

$$n - k \leq m\ (n - k) * (n - k + 1)/2.$$

Розрізом графа G називається така множина ребер, що розділяє граф G. Розріз графа G, який складається лише з одного ребра, називається мостом.

Нехай множина ребер довільного графа G, є множиною, що розділяє граф G. Якщо таку множину його ребер, вилучити з графа G викликає збільшення числа його компонентів зв'язності. Якщо граф G зв'язний, то вилучення множини ребер, що розділяє граф G веде до незв'язного графа.

Метричні характеристики зв'язних графів

Нехай G = (V, E) – зв'язний граф, а u і v – дві його різні вершини. **Відстанню між вершинами** u і v називається довжина найкоротшого маршруту, який з'єднує вершини u і v, і позначається d(u, v). Покладемо також, що d (u, u) = 0. Очевидно, що для введеної таким чином відстані виконуються аксіоми:

1. $d(u, v) \geq 0$;
2. $d(u, v) = 0$ тоді і тільки тоді, коли $u = v$;
3. $d(u, v) = d(u, v)$;
4. $d(u, v) + d(u, w) \geq d(u, w)$ (нерівність трикутника).

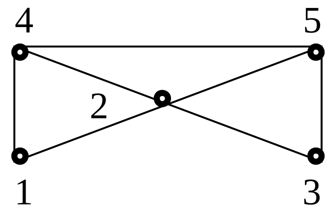
Поняття відстані між вершинами дає можливість визначити поняття степеня графа. Нехай G – зв'язний граф, k – натуральне число. Граф G^k – k-й степінь графа G, має ту ж множину вершин, що й G, а нерівні між собою вершини u і v суміжні в графі G^k тоді і тільки тоді, коли $d(u, v) \leq k$. Безпосередньо з цього випливає така властивість графа G^k.

Якщо $k \geq |V| - 1$, де V – множина вершин графа G, то G^k – повний граф.

Нехай u – деяка фіксована вершина графа G = (V, E). Величина $e(u) = \max d(u, v)$, $v \in V$, називається ексцентриситетом вершини u. Максимальний серед всіх ексцентриситетів вершин графа G називається діаметром графа G і позначається d(G). Отже, $d(G) = \max e(u)$, $u \in V$.

Вершина v графа G називається периферійною, якщо e(v) = d(G). Простий ланцюг довжини d(G), відстань між початковою і кінцевою вершинами якого дорівнює d(G), називається діаметральним ланцюгом.

У графі G (малюнок 14) маємо d(1, 2) = 1, d(1, 3) = 2 = d(1, 5), e(1) = 2, e(2) = 1, d(G) = 2. Всі вершини, крім вершини 2, периферійні, а (1, 2, 3) – діаметральний ланцюг.



Граф G

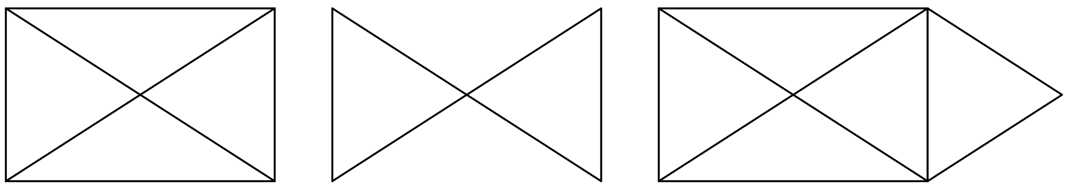
Рис.14

Властивості ейлерових та гамільтонових графів

Одним з важливіших різновидів зв’язних графів є ейлерові та гамільтонові графи.

Зв’язний граф G називається ейлеровим (малюнок 15), якщо існує замкнутий ланцюг, який включає кожне його ребро. Такий ланцюг називається **ейлеровим ланцюгом**.

Зв’язний граф G називається напівейлеровим (малюнок 15), якщо в ньому існує ланцюг, який включає кожне його ребро. Таким чином, всякий ейлерів граф буде напівейлеровим.



а) неейлерів гра; b) ейлерів граф c) напівейлерів граф

Рис.15

Зв’язний граф G є ейлеровим тоді і тільки тоді, коли кожна вершина G має парний степінь.

Нехай G = (V, E) – ейлерів граф. Тоді наступна процедура завжди можлива і веде до ейлерового ланцюга в графі G: виходячи з будь-якої вершини $u \in V$, йдемо по ребрах графа G довільним чином згідно з такими правилами:

1. стираємо ребра, які пройдені, і стираємо ізольовані вершини, які при цьому виникають;
2. на кожному етапі йдемо по мосту лише тоді, коли немає інших можливостей.

Проблема існування замкнутого ланцюга, який проходить через кожне ребро зв’язного графа G, аналогічно може бути сформульована і для вершин. Тобто, чи існує замкнутий ланцюг у зв’язному графі G, який проходить рівно один раз через кожну вершину графа G. Очевидно, що такий ланцюг має бути циклом. Якщо такий цикл у графа G існує, то він називається **гамільтоновим ланцюгом**, а граф G – **гамільтоновим графом**.

Якщо в графі G = (V, E) порядку n зафіксувати одну з вершин і обхід графа завжди починати з неї, то всякому гамільтоновому циклу буде відповідати перестановка елементів множини V.

Тема 4 Матриці і графи

Всяке бінарне відношення можна задати у вигляді матриці, яка складається з нулів та одиниць і називається **матрицею суміжностей**. Взаємно однозначна відповідність між бінарними відношеннями і графами на деякій скінченій множині дає можливість стверджувати, що всякий граф можна подавати у вигляді матриці суміжностей. Якщо G = (V, E) – скінчений граф порядку n, то йому відповідає квадратна матриця A (G) розмірності n * n, яка має вигляд

$$a_{ij} = \begin{cases} 1, \text{ коли вершини } v_i \text{ і } v_j \text{ суміжні,} \\ \\ 0 \text{ – в протилежному випадку.} \end{cases}$$

Внаслідок симетричності відношення суміжності матриця суміжностей графа має бути симетричною, а число одиниць в і–му рядку – дорівнювати степені вершини u_i.

Завдання графа за допомогою матриць суміжностей дозволяє сформулювати простий критерій ізоморфності графів.

Графи ізоморфні тоді і тільки тоді, коли їх матриці суміжностей можна одержати одна з другої однаковими перестановками рядків і стовпчиків.

Перенумеруємо вершини графів G = (V, E) і H = (V_1, E_1) (|V| = |V_1| = n) цілими числами від 1 до n.

Якщо A(G) = B(H), то все доведено. У протилежному випадку графи G і H відрізняються лише нумерацією вершин. Значить існує така підстановка s на множині V, яка зберігає суміжність, тобто якщо $(u, v) \in E$, то $(s(u), s(v) \in E_1)$. Тоді маємо $b_{s(i)s(j)} = a_{ij}$, i, j = 1, 2, … n.

Якщо A – матриця суміжностей графа G, який має порядок n, то елемент b_{ij} матриці A^k є числом маршрутів довжини k, які з’єднують вершини u_i та u_j в графі G.

В графі порядку n маршрут, що з’єднує вершини v і u, існує тоді і тільки тоді, коли елемент матриці $\boldsymbol{C} = \sum_{\boldsymbol{k}=1}^{\boldsymbol{n}} \boldsymbol{A}^{\boldsymbol{k}}$, що відповідає цим вершинам не дорівнює нулю.

Граф зв'язний тоді і тільки тоді, коли кожний елемент матриці графа дорівнює одиниці.

Аналогічно визначається матриця суміжностей орграфа G = (V, E):

$$a_{ij} = \begin{cases} 1, \text{ якщо } (u_i, u_j) \in E, \\ 0 - \text{в протилежному випадку.} \end{cases}$$

Матриці суміжностей неорієнтованого графа і орграфа різняться тим, що перша завжди симетрична, а друга не обов'язково симетрична. Очевидно що всяка квадратна матриця, елементи якої 0 і 1, буде матрицею суміжностей деякого орієнтованого графа.

Нехай маємо матрицю

$$\begin{vmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{vmatrix}$$

Цій матриці відповідає орграф зображений на малюнку 16.

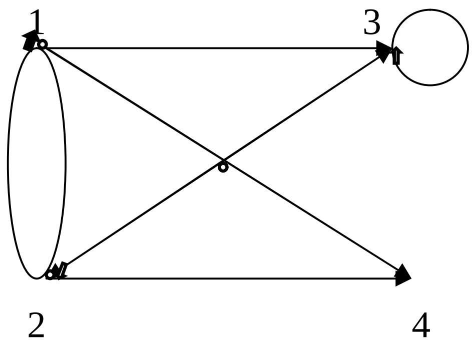


Рис.16

Рангом графа G називається ранг його матриці суміжностей, який позначається **rank(G)**. Ранги матриць суміжностей ізоморфних графів рівні між собою.

Матриця I(G), називається **матрицею інцидентності** графа G = (V, E), де V = {1, 2, … n}, E = (e_1, … e_m), якщо вона відповідає таким умовам:

$$i_{ki} = \begin{cases} 1, \text{ якщо вершина } k \text{ і ребро } e_1 \text{ інцидентні,} \\ 0 - \text{в протилежному випадку.} \end{cases}$$

Графи (орграфи) ізоморфні тоді і тільки тоді, коли їх матриці інцидентності можна одержати одну з другої довільними перестановками рядків і стовпчиків.

Граф G можна перетворити в орграф, якщо кожному ребру надати одну з двох можливих орієнтацій. Одержаний орграф називається **орієнтацією графа** G.

ТЕМА 5 РОЗФАРБУВАННЯ ГРАФІВ

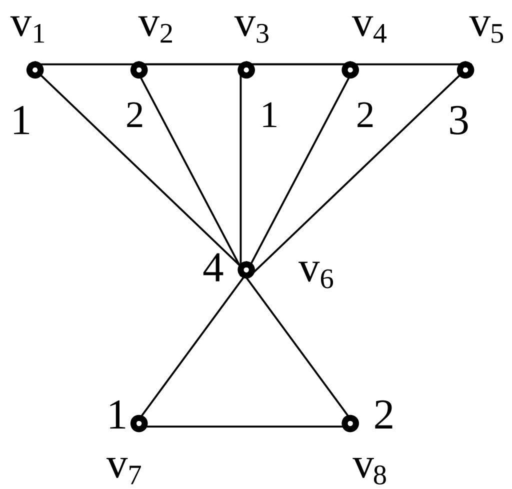
Правильне розфарбування

Нехай G = (V, E) – скінчений граф, а k– деяке натуральне число. Довільна функція f: $V \to N_k$ де $N_k = \{1, 2, \dots k\}$, називається **вершинним k-розфарбуванням**, або просто k-розфарбуванням графа G. Розфарбування називається правильним, якщо $(\forall(u, v) \in E)\ (f(u) \neq f(v))$.

Граф, для якого існує правильне k-розфарбування, називається розфарбованим графом. При визначенні розфарбування графа множину N_k можна замінити довільною k-елементною множиною.

Правильне розфарбування графа можна трактувати як розфарбування кожної його вершини в один з k кольорів, таким чином, щоб суміжні вершини були розфарбовані в різні кольори. Оскільки функція f не обов'язково взаємно однозначна, то при k-розфарбуванні фактично може бути використано менш ніж k кольорів. Отже, правильне k-розфарбування можна розглядати як розбиття множини вершин V графа G на класи $V_1 \cup \cup V_2 \cup \dots \cup V_i = V$, де $i \leq k$, $V_i \neq \varnothing$, i = 1, 2, …, l. Кожний клас V_i – незалежна множина, а самі класи називаються колірними класами.

Мінімальне число k, при якому існує правильне k-розфарбування графа G, називається **хроматичним числом** цього графа і позначається $X_p(G)$. Якщо $X_p(G) = k$, то граф G називається k-хроматичним. Правильне k-розфарбування графа G при $k = X_p(G)$ називається мінімальним (малюнок 17).



На малюнку 17 натуральними числами 1, 2, 3, 4 позначені фарби відповідних вершин

Рис.17

Практичні задачі, що призводять до розфарбування

Задача складання розкладу занять. Нехай потрібно прочитати кілька лекцій за найкоротший проміжок часу. Читання лекції займає одну годину, але деякі лекції не можуть читатися одночасно (наприклад, коли їх читає один лектор). Побудуємо граф G = (V, E), де V – множина, яка відповідає множині лекцій, причому дві вершини суміжні тоді і тільки тоді, коли відповідні лекції не можуть читатися одночасно. Очевидно, що всяке правильне розфарбування цього графа визначає допустимий розклад: лекції, які відповідають вершинам графа, що становлять один колірний клас, читаються одночасно. Навпаки, всякий допустимий розклад визначає правильне розфарбування графа G. Оптимальний розклад відповідає мінімальним розфарбуванням, а число годин, необхідне для того, щоб прочитати всі лекції, дорівнює $X_p(G)$.

Хроматичні числа деяких графів

Для деяких простих графів неважко знайти хроматичні числа. Наприклад, $X_p(K_n) = n$, $X_p(K_n – e) = n – 1$, $X_p(K_{nm}) = 2$, $X_p(C_{2n}) = 2$, $X_p(C_{2n-1}) = 3$.

Твердження про хроматичні числа графів:

- граф є 1-хроматичним тоді і тільки тоді, коли він пустий, а 2-хроматичним – коли він двочастинний і непустий, 2-хроматичні графи, називають біхроматичними;
- непустий граф є біхроматичним тоді і тільки тоді, коли він не має циклів непарної довжини;
- якщо **r** означає максимальний степінь вершин графа G = (V, E), то для всякого графа G виконується нерівність $X_p(G) \le r + 1$;
- всякий кубічний граф розфарбовується чотирма фарбами;
- якщо G зв’язний неповний граф і $r \ge 3$, то $X_p(G) \le 3$.

Хоча наведені твердження і дають певну інформацію про хроматичні числа графів, але їх оцінки досить неточні. Наприклад, зірковий граф $K_{1,n}$, розфарбовується n фарбами, насправді є біхроматичним.

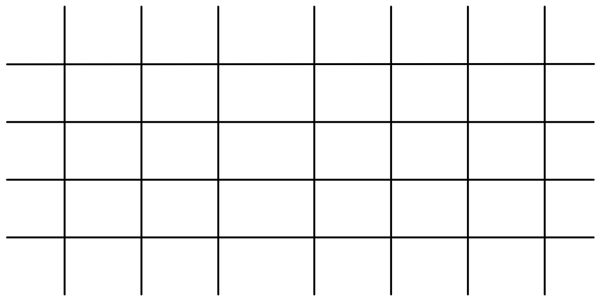
ТЕМА 6 НЕСКІНЧЕНІ ГРАФИ

Нескінченим (неорієнтованим) графом називається пара G = (V, E) де V – нескінчена множина вершин і E – нескінчена множина ребер. Якщо обидві множини V і E злічені, то граф називається **зліченим**. Слід зазначити, що ці визначення не стосуються випадку, коли V – скінчена множина, а E – нескінчена множина, або коли E – скінчена множина, а V – нескінчена множина. Граф G слід вважати скінченим, якщо він має в першому випадку скінчене число вершин і нескінчене число ребер (або петель), а в другому випадку – скінчене число ребер і нескінчене число ізольованих вершин.

Означення таких понять, як суміжні вершини, інцидентні ребра, ізоморфні графи, підграф графа, зв'язний граф, компонента зв'язності повністю переносяться на нескінчені графи.

Степенем вершин у нескінченого графа називається потужність множини ребер, інцидентних вершин v. Отже степінь вершин в нескінченому графі може бути як скінченим, так і нескінченим.

Нескінчений граф називається **локально скінченим**, якщо степінь кожної його вершини скінчений. Прикладом локально скінченого графа може бути квадратна решітка (малюнок 18), складена з цілих чисел.



Локально скінчений граф Рис.18

Нескінчений граф називається **локально зліченим**, якщо кожна вершина цього графа має злічений степінь.

Для нескінчених графів дійсні наступні твердження:

- будь який зв'язний локально злічений нескінчений граф є зліченим;
- всякий зв'язний локально скінчений нескінченний граф є зліченим.

На скінчені графи переноситься і поняття маршруту, причому воно може мати такі різновиди:

- **скінчений маршрут** у нескінченому графі G визначається так, як у випадку скінчених графів;
- **нескінченим в одну сторону маршрутом** у графі G, який починається вершиною v_0, називається нескінчена послідовність ребер (v_0, v_1), (v_1, v_2), (v_2, v_3), …;
- **нескінченим в обидві сторони маршрутом** у графі G називається нескінчена послідовність ребер …, (v_{-2}, v_{-1}), (v_{-1}, v_0), (v_0, v_1), (v_1, v_2), …

Нескінчені в одну і обидві сторони ланцюги і прості ланцюги визначаються так, як і поняття довжини ланцюга і відстані між вершинами. Умовні існування нескінчених ланцюгів у графах дає теорема, яка в теорії графів відома під назвою леми Кьоніга.

Лема Кьоніга. Нехай G – зв'язний локально скінчений нескінчений граф. Тоді для всякої його вершини v існує нескінчений в один бік простий ланцюг, який починається у вершині v.

Доведення. Якщо u довільна вершина графа G, відмінна від вершини v, то існує деякий простий ланцюг, який з'єднує вершини v і u. Звідси випливає, що в G повинно бути нескінченно багато простих ланцюгів з початком у вершині v. Оскільки степінь вершини v скінчений, то нескінчена множина таких простих шляхів повинна починатися з одного і того ж ребра. Якщо таким ребром є ребро (u, v_1), то повторивши цю процедуру для вершини v_1, одержимо нову вершину v_2 і відповідне їй ребро (v_1, v_2). Продовжуючи цей процес далі, одержимо нескінчений в одну сторону простий ланцюг v, v_1, v_2, …

Важливе значення леми Кьоніга полягає в тому, що вона дає можливість одержувати результати про нескінчені графи з відповідних результатів про скінчені графи.

Зв'язний злічений граф G називається **ейлеровим**, якщо в ньому існує нескінчений в обидві боки ланцюг, який містить кожне ребро графа G. Такий ланцюг називається двостороннім ейлеровим ланцюгом.

Злічений граф називається **напівейлеровим**, якщо в ньому існує нескінченний в один, або в обидва боки ланцюг, який містить в собі кожне ребро графа G.

ТЕМА 7 ДЕРЕВА ТА ЇХ ВЛАСТИВОСТІ

Означення дерева. Властивості дерев

Зв’язний ациклічний граф називається (неорієнтованим) деревом. Дерево називається кореневим, якщо в ньому виділена вершина, яка називається коренем. Остовним деревом графа G називається остовний підграф графа G, який є деревом.

У відношенні дерев діють наступні твердження:

- граф є деревом тоді і тільки тоді, коли будь-які дві його вершини зв’язані лише одним ланцюгом;
- якщо T – дерево і u – його кінцева вершина, то граф T – u – дерево;
- всяке непусте дерево має щонайменше дві кінцеві вершини і одне кінцеве ребро;
- ребро зв’язаного графа називається суттєвим, якщо його вилучення веде до порушення зв’язності цього графа;
- якщо $T = (V, E)$ – дерево і вершина $v \in V$, то граф $T^{|} = (V \cup \{v\}, E \cup \cup\{(u, v)\})$, де u – довільна вершина із V, теж є деревом.

Якщо граф T має n вершин, тоді еквівалентні такі твердження:

- T є деревом;
- T – не має циклів і має $n - 1$ ребро;
- T – зв’язний граф і має $n - 1$ ребро;
- T – зв’язний граф, і кожне його ребро є мостом;
- Будь-які дві вершини графа T з’єднані рівно одним простим ланцюгом;
- T – не має циклів, але введення нового ребра в T сприяє виникненню рівно одного циклу;

Якщо T має цикл, то будь які дві вершини цього циклу з’єднані не менше, ніж двома простими ланцюгами. Добавляючи до графа T деяке ребро e, одержимо цикл, оскільки вершини інцидентні ребру e уже зв’язані в T простим ланцюгом.

Якщо G – ліс з n вершинами і k компонентами, тоді G має $n - k$ ребер.

Теорема Келі. Число різних дерев, які можна побудувати на n вершинах, дорівнює n^{n-2}.

Доведення. Нехай $V = \{v_1, v_2, \dots v_n\}$ і $T = (V, E)$ – дерево. В T є кінцеві вершини. Нехай v_1 – перша з них і (u_1, v_1) – відповідне їй кінцеве ребро. Вилучимо з T це ребро і вершину v_1, позначимо в множині V вершину u_1 і занесемо її в список вершин L, який спочатку є пустим. Одержаний граф знову є деревом. Застосуємо до нього ту ж процедуру, одержимо нове дерево і список $L = [u_1, u_2]$ і т. д. Цю процедуру застосуємо доти, доки після вилучення ребра (u_{n-2}, v_{n-2}) не залишиться єдине ребро (u_{n-1}, v_{n-1}). Тоді список $L = [u_1, u_2, \dots u_{n-2}]$ однозначно визначиться деревом T, і двома різними деревами T і $T^{|}$ з n вершинами яким відповідають різні списки. Кожна вершина u з'являється в списку Ln(u) – 1 раз.

Навпаки, кожен список L визначає дерево T за допомогою зворотної побудови. Якщо задано список L, то знаходимо першу позначену вершину v_1 в множині V, таку, що $v_1 \in L$. Ця вершина визначає ребро (u_1, v_1). Далі витримаємо позначку вершини u_1 в множині V, вилучимо u_1 із L і продовжуємо побудову для нового списку L, який складається з n – 3 елементів. Одержаний внаслідок такої побудови граф є деревом.

Оскільки різних елементів в списку L може бути не більш n^{n-2}, то цим доводиться справедливість теореми.

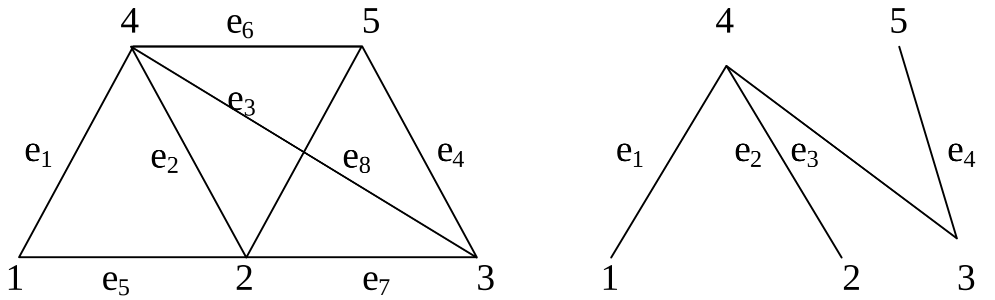
Теорема Келі має багато різних інтерпретацій. Зупинимося на одній з них, яка відома під назвою задачі про з'єднання n міст залізницями так, щоб не будувати зайвих ліній; скількома способами можна побудувати таку систему доріг?

Теорема Келі дає розв'язок цієї задачі. Для цього кожному дереву T, побудованому на n вершинах, ставиться у відповідність сума $\boldsymbol{C}(\boldsymbol{T}) = \sum_{(\boldsymbol{a},\boldsymbol{b}) \in \boldsymbol{E}} \boldsymbol{c}(\boldsymbol{a},\boldsymbol{b})$. виправши серед всіх сум найменшу, одержимо розв'язок задачі. Для цього, як випливає з теореми Келі, необхідно переглянути не більш ніж n^{n-2} сум.

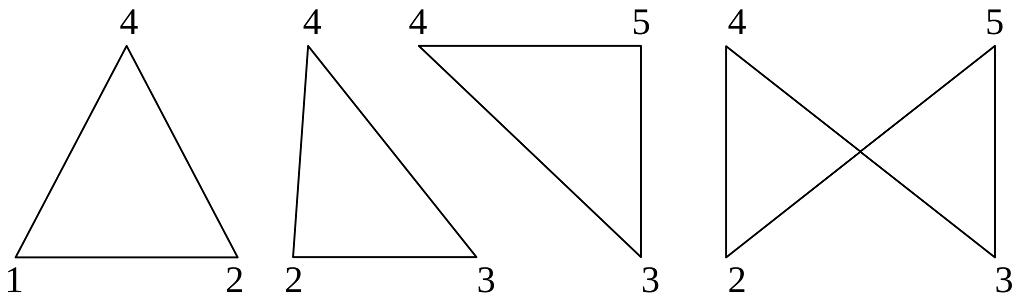
Застосовуючи процедуру доведення теореми Келі до кожного компонента графа G, одержимо граф, який називається **остовним лісом**. Число ребер, які при цьому вилучаються, називаються **цикломатичним числом** або **циклічним рангом графа** G і позначається C(G). Цикломатичне число є мірою зв'язності графа.

Фундаментальна система циклів графа

З поняттям остовного лісу T графа G тісно пов'язане поняття фундаментальної системи циклів, яка асоціюється з T. Нехай T – остовний ліс графа G. Якщо ввести довільне ребро в граф G, яке не входить в T, то одержимо єдиний цикл. Множина всіх циклів, які одержані в такий спосіб (шляхом введення окремо кожного ребра в граф G, яке не входить до T), називається фундаментальною системою циклів, асоційованою з T. У кожному випадку, коли нас цікавить, який остовний ліс розглядається, будемо говорити про фундаментальну систему циклів графа G (малюнок 19).



Граф G — Остовне дерево графа G



Фундаментальна система циклів, асоційована з остовним деревом графа G

Рис.19

Для фундаментальної системи циклів дійсні такі твердження:

- якщо G = (V, E) довільний скінчений граф, то число ребер G, які необхідно вилучити для одержання остовного лісу T, не залежить від порядку їх вилучення;
- граф G являє собою ліс тоді і тільки тоді, коли C(G) = 0;
- всякий ациклічний підграф довільного скінченого графа G = (V, E) є підграфом остовного дерева графа G;
- всяке дерево порядку $n \geq 2$ має щонайменше дві кінцеві вершини.

ТЕМА 8 АЛГОРИТМИ ПОШУКУ НАЙКОРОТШОГО ШЛЯХУ

Кожній дузі $v_{(i,j)}$ графа G поставимо у відповідність число a(x, y). Якщо в графі G відсутня деяка дуга $v_{(i,y)}$, визначимо її як a(x, y) = ∞. Назвемо число a(x, y), довжиною дуги $v_{(i,j)}$. Довжину шляху в графі визначимо як суму довжин окремих дуг, які складають цей шлях.

Для будь-яких двох вершин **s** і **t** графа G можуть існувати декілька шляхів, що з'єднують вершину **s** с вершиною **t**. Завжди відшукується найкоротший шлях, що з'єднує **s** і **t**.

Алгоритм Дейкстри дозволяє визначити найкоротший шлях між двома виділеними вершинами **s** і **t** при позитивних довжинах дуг. Головна ідея, що лежить в основі алгоритму Дейкстри гранично проста. Припустимо, що нам відомі **m** вершин, найближчих до вершини **s.** Близькість будь-якої вершини **x** до вершини **s** визначається довжиною найкоротшого шляху, що веде з **s** у **x**. Нехай відомі найкоротші шляхи, що з'єднують вершину **s** із виділеними **m** вершинами.

Спочатку розфарбовуються найближчі до s вершини. Будуються шляхи для кожній із нерозфарбованих вершин, які безпосередньо з'єднують, розфарбовану вершину з нерозфарбованою. Потім вибирається з усіх шляхів найкоротший. Його називають умовно найкоротшим.

Алгоритм Дейкстри для пошуку найкоротшого шляху

Крок 1. Перед початком виконання алгоритму усі вершини і дуги нерозфарбовані. Кожній вершині в ході виконання алгоритму, присвоюється число d(x), рівне довжині найкоротшого шляху з **s** у **x,** який включає тільки розфарбовані вершини.

Покладемо d(x) = 0 і d(x) = ∞ для всіх значень **x**, відмінних від **s**. Розфарбуємо вершину **s** і покладемо **y** = **s** (**y** – остання з розфарбованих вершин).

Крок 2. Для кожної нерозфарбованої вершини x таким способом перерахуємо довжину d(x):

$$d(x) = \min \{d(x), d(y) + a(y, x)\} \quad (1)$$

Якщо d(x) = ∞ для всіх нерозфарбованих вершин **x**, закінчимо процедуру алгоритму, у вихідному графі відсутні шляхи з вершини **s** у нерозфарбовані вершини. У протилежному випадку розфарбуємо ту з вершин **x**, для якої довжина дуги на даному кроку досягався мінімум відповідно до виразу (1). Покладемо y = x.

Крок 3. Якщо y = t, закінчимо процедуру, найкоротший шлях із вершини **s** у вершину t знайдений (це єдиний шлях із **s** у **t**, складений із розфарбованих дуг). У протилежному випадку перейдемо до другого кроку.

Щоразу, коли офарблюється деяка вершина, (не рахуючи вершини s), офарблюється і деяка дуга, що заходить у дану вершину. Таким чином, на будь-якому етапі алгоритму в кожну вершину заходить не більш ніж одна розфарбована дуга. Розфарбовані дуги не можуть утворювати у вихідному графі цикл, тому що в алгоритмі не може офарблюватися дуга, кінцеві вершини котрої вже розфарбовані.

Розфарбовані дуги у вихідному графі утворять орієнтоване дерево з коренем у вершині **s**. Таке дерево називається орієнтованим деревом найкоротших шляхів. Єдиний шлях від вершини **s** до будь-якого вершині **x**, який належить дереву найкоротших шляхів, є найкоротшим шляхом між зазначеними вершинами.

Якщо найкоротшому шляху з вершини **s** у вершину **x**, у дереві найкоротших шляхів належить вершина **y**, то частина цього шляху, укладена між x і y, є найкоротшим шляхом між цими вершинами. Якщо між **x** і **y** існує більш короткий шлях, то шлях між вершинами **s** і **x** не може бути найкоротшим.

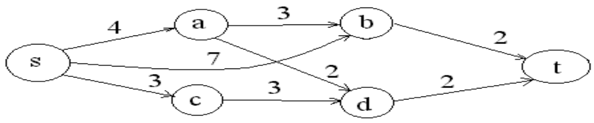
Оскільки на всіх етапах алгоритму Дейкстри розфарбовані дуги утворять у вихідному графі орієнтоване дерево, то алгоритм можна розглядати як процедуру нарощування орієнтованого дерева з коренем у вершині **s**. Коли в процедурі нарощування досягається вершина **t**, алгоритм може бути припинений.

При визначенні найкоротших шляхів із вершини **s** в усі вершини вихідного графа, процедуру нарощування дерева варто продовжити доти, поки усі вершини графа не будуть включені в орієнтоване дерево найкоротших шляхів. При цьому, для вихідного графа буде отримане орієнтоване дерево, за умови, що в графі існує хоча б одне таке дерево.

Для того, щоб описаний алгоритм дозволяв одержувати дерево найкоротших шляхів від вершини **s** до всіх інших вершин, його третій крок повинний бути скоригований у такий спосіб: якщо усі вершини стають розфарбованими, закінчити процедуру (для будь-якої вершини x є єдиний шлях із **s** у **x**, який складається з пофарбованих дуг, цей шлях є найкоротшим шляхом між відповідними вершинами); у противному випадку перейти до кроку 2.

Використаємо алгоритм Дейкстри до орієнтованого графа, зображеному на малюнку 20, для винаходження в ньому найкоротшого шляху між вершинами **s** і **t.**

Перед виконанням кроку 2 алгоритму Дейкстри пофарбована тільки вершина **s**. Крім того, d(x) = 0 і d(x) = ∞ для усіх вершинах, що не збігаються з **s**.



Орієнтований граф

Рис.20

Крок 2. (y = s):

d(a) = min {d(a), d(s) + a(s, a)} = min {∞, 0 + 4} = 4,

d(b) = min {d(b), d(s) + a(s, b)} = min {∞, 0 + 7} = 7,

d(c) = min {d(c), d(s) + a(s, c)} = min {∞, 0 + 3} = 3,

d(d) = min {d(d), d(s) + a(s, d)} = min {∞, 0 + ∞} = ∞,

d(t) = min {d(t), d(s) + a(s, t)} = min {∞, 0 + ∞} = ∞.

Оскільки розмір d(c) = 3 є мінімальним з розмірів d(a), d(b), d(d), d(c) і d(t), то вершина c офарблюється (малюнок 21). Так само офарблюється і дуга (s, c), яка визначає вершину d(s). Поточне дерево з найкоротших шляхів складається з дуги s, c.

Крок 3. Оскільки вершина t залишається нерозфарбованою, здійснюється перехід до кроку 2.

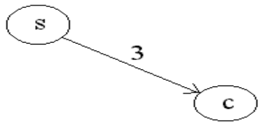
Крок 2. (y = c):

d(a) = min {d(a), d(c) + a(c, a)} = min {4, 3 + ∞} = 4,

$d(b) = \min\{d(b), d(c) + a(c, b)\} = \min\{7, 3 + \infty\} = 7,$

$d(d) = \min\{d(d), d(c) + a(c, d)\} = \min\{\infty, 3 + 3\} = 6,$

$d(t) = \min\{d(t), d(c) + a(c, t)\} = \min\{\infty, 3 + \infty\} = \infty.$



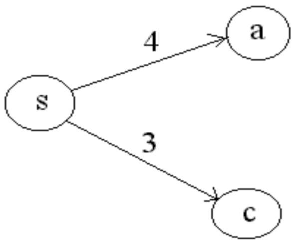
Орієнтоване дерево найкоротших шляхів, що росте

Рис..21

Оскільки розмір d(a) = 4 є мінімальним з розмірів d(a), d(b), d(d), d(c) і d(t), то вершина a офарблюється (малюнок 22). Так само офарблюється і дуга (s, a), яка визначає вершину d(a). Поточне дерево з найкоротших шляхів тепер складається з дуг (s, c) і (s, a).

Крок 3. Оскільки вершина t залишається нерозфарбованою, здійснюється перехід до кроку 2.

Крок 2 (y = a).



Орієнтоване дерево найкоротших шляхів, що росте

Рис.22

$d(b) = \min\{d(b), d(a) + a(a, b)\} = \min\{7, 4 + 3\} = 7,$

$d(d) = \min\{d(d), d(a) + a(a, d)\} = \min\{6, 4 + 2\} = 6,$

$d(t) = \min\{d(t), d(a) + a(a, t)\} = \min\{\infty, 4 + \infty\} = \infty.$

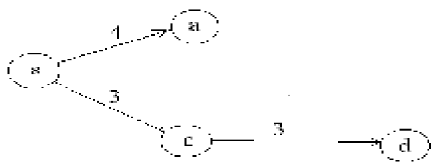
Оскільки розмір d(d) = 6 є мінімальним з розмірів d(b), d(d) і d(t), то вершина d офарблюється (малюнок 23). Можна вважати, що розмір d(d) визначають як дуга (c, d), так і дуга (a, d). Тому можна офарбити будь-яку з цих дуг. Офарбимо, наприклад дугу (c, d). Поточне дерево найкоротших шляхів складається тепер із дуг (s, c), (s, a) і (s, d).

Крок 3. Оскільки вершина t залишається неофарбленню, здійснюється перехід до кроку 2.

Крок 2. (y = d):

$d(b) = \min \{d(b), d(d) + a(d, b)\} = \min \{7, 6 + \infty\} = 7,$

$d(t) = \min \{d(t), d(d) + a(d, t)\} = \min \{\infty, 6 + 2\} = 8.$



Орієнтоване дерево найкоротших шляхів, що росте

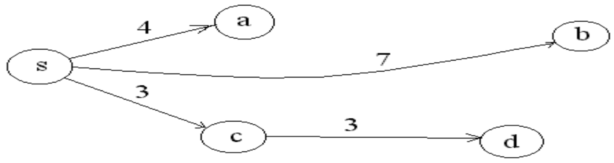
Рис.23

Оскільки розмір d(b) = 7 = min {d(b), d(t)} менше розміру d(t) то вершина b офарблюється (малюнок 24). Так само офарблюється і дуга (s, b), що визначає розмір d(b). Поточне дерево найкоротших шляхів тепер складається з дуг (s, c), (s, a), (c, d) і (s, b).

Крок 3. Оскільки вершина t залишається неофарбленню, здійснюється перехід до кроку 2.

Крок 2. (y = b):

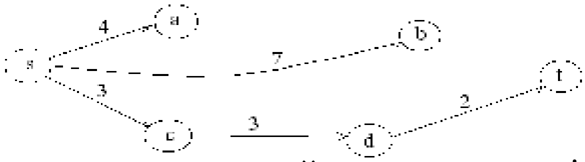
$d(t) = \min \{d(t), d(b) + a(b, t)\} = \min \{8, 7 + 2\} = 8.$



Орієнтоване дерево найкоротших шляхів, що росте

Рис.24

Вершина t офарблюється (малюнок 25). Разом із ній офарблюється дуга (d, t), що визначає довжину d(t). Остаточно побудоване дерево найкоротших шляхів складається з дуг (s, c), (s, a), (c, d) (s, b) і (d, t).



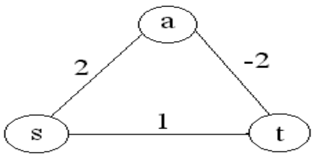
Орієнтоване дерево найкоротших шляхів, що росте

Рис.25

Найкоротший шлях, що з'єднує вершину **s** із вершиною **t**, складають-ся з дуг (s, c), (s, d), (d, t) і має довжину 3 + 3 + 2 = 8. Це не єдиний найко-ротший шлях між вершинами **s** і **t**. Шлях, що складається з дуг (s, a), (a, d), (d, t), має довжину 4 + 2 + 2 + = 8 і також є найкоротшим шляхом між вер-шинами s і e. найкоротший шлях буде єдиним тільки в тому випадку, якщо в процедурі алгоритму жодного разу не виникає неоднозначність у виборі дуги, що офарблюється.

Алгоритм Дейкстри, якщо деякі з довжин дуг негативні

Для приклада роздивимося неорієнтований граф (малюнок 26), у якому найкоротшим шляхом, між вершинами s і t, є шлях, що складається з дуг (s, a) і (a, t). довжина цього шляху дорівнює 2 − 2= 0.



Неорієнтований граф

Рис.26

Алгоритм Дейкстри може бути обгрунтований на випадок, коли деякі з дуг, мають негативні довжини. Ідея відповідного узагальнення належить Форду. Необхідна модифікація алгоритму Дейкстри складається в такому:

- На кроку 2 алгоритму перерахунок розмірів d(x) за допомогою співвідношення (1) провадиться для усіх вершини, а не тільки для неофарбованих. Отже, числа d(x) можуть зменшатся як для неофарбованих, так і для розфарбованих вершин.
- Якщо для деякої пофарбованої вершини x відбувається зменшення розміру d(x), то з цієї вершини і інцидентної їй пофарбованої дуги, починається фарбування.
- Процедура алгоритму закінчується тільки тоді, коли усі вершини пофарбовані, і коли після виконання кроку 2 жодне з чисел d(x) не змінюється.

Обгрунтування модифікованого алгоритму Дейкстри. Алгоритм Форда

Обгрунтування проведемо за схемою доказу від противного. При цьому, будемо мати на увазі, що процедура алгоритму Форда може бути скінчена тільки після того, як для всіх **x** і **y** починає виконуватися співвідношення:

$$d(x) + a(x, y) \geq d(y) \qquad (2)$$

Інакше з вершини **y** було б обов'язково зняте фарбування на ітерації, що безпосередньо випливає за ітерацією, на якій була пофарбована вершина **x**.

Припустимо, що після закінчення процедури алгоритму Форда довжина d(y) для деякої вершини **y** не збігається з довжиною найкоротшого шляху до цієї вершини з вершини **s** Відзначимо, якщо довільна довжина d(z) для довільної довжини z кінцева, то вона являє собою довжину деякого шляху з'єднуючого вершини s і z. Якщо таких шляхів декілька, то в якості x береться та вершина, найкоротший шлях до який від вершини s включає найменше число дуг. У силу вибору вершин **y** і **x** довжина d(x) повинна збігатися з довжиною найкоротшого шляху з **s** у **x**, відкіля d(y) > d(x) + a(x, y). Проте, останнє співвідношення суперечить співвідношенню (2). Визначимо алгоритм Форда до графа, зображеному на мал.26.

Крок 1. Офарблюється вершина s. Покладається d(s) = 0, d(a) = ∞ и d(t) = ∞.

Крок 2. ($y = s$).

$d(a) = \min\{d(a), d(s) + a(s, a)\} = \min\{\infty, 0 + 2\} = 2$,

$d(t) = \min\{d(s), d(a) + a(a, s)\} = \min\{0, 2 + 8\} = 1$.

Оскільки d(t) менше ніж d(a), вершина t офарблюється. Разом із нею офарблюється і дуга (s, t), що і складає поточне дерево найкоротших шляхів.

Крок 3. Оскільки пофарбовані не усі вершини, робиться перехід до кроку 2.

Крок 2. ($y = t$).

Оскільки з вершини t не виходить жодної дуги, то числа d(x) залишаються незмінними. Тому офарблюється вершина a і разом із ній, дуга (s, a). Тепер дерево найкоротших шляхів складається з дуг (s, t) і (s, a).

Крок 3. здійснюється повернення до кроку 2 із тим, щоб спробувати зменшити числа d(x).

Крок 2. ($y = a$). Визначаються:

$d(t) = \min\{d(t), d(a) + a(a, t)\} = \min\{1, 2 - 2\} = 0$,

$d(s) = \min\{d(s), d(a) + a(a, s)\} = \min\{0, 2 + ?\} = 0$.

Оскільки розмір d(t) зменшується від 1 до 0, із вершини t то дуга (s, t) офарбовується. Тепер дерево найкоротших шляхів складається тільки з дуги (s, a).

У вихідному графі залишається нефарбованою тільки одна вершина − вершина t. Ця вершина офарбовується з дугою (a, t), оскільки на даному кроку фігурує a. Дерево найкоротших шляхів тепер включає дуги (s, a) і (a, t).

Крок 3. здійснюється чергове повернення до кроку 2.

Шаг2. ($y = t$).

З вершини t не виходить не однієї дуги, розміри d(x) не змінюються, нефарбованих вершин у вихідному графі немає.

Крок 3. Всі вершини графа надаються пофарбованими, на попередньому кроку алгоритму, не одну з розмірів d(x) не вдалося зменшити. Процедура алгоритму закінчується. Найкоротший шлях із вершини **s** у вершину **t** складається з дуг (s, a) і (a, t) і має довжину $2 - 2 = 0$.

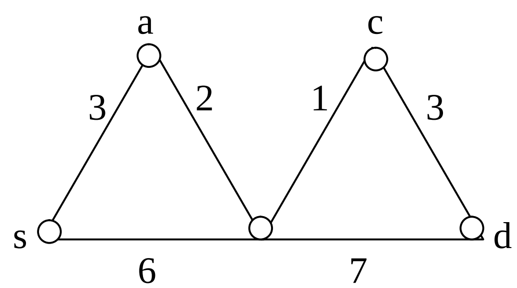
Алгоритм Форда не вирішує задач пошуку найкоротшого шляху при наявності у вихідному графі контуру, що має негативну довжину.

Задача листоноші як алгоритм пошуку найкоротшого шляху

Будь-який листоноша, перед тим як відправитись в шлях, повинен підібрати на пошті листи, що стосуються його ділянки, після цього він повинний доставити їх адресатам, які знаходяться уздовж маршруту його проходження, і з'явитися на пошту, щоб повернути залишену кореспонденцію. Кожний листоноша, бажає витрачати менше сил, або хоча б пройти маршрут найкоротшим шляхом. Задача листоноші полягає в тому, щоб пройти усі вулиці маршруту і повернуться в його початкову точку, скорочуючи при цьому довжину пройденого шляху.

Задача листоноші може бути переформульована в термінах теорії графів. Для цього побудуємо граф G = (V, E), у котрому ребро відповідає вулиці в маршруті листоноші, а кожна вершина – стику двох вулиць. Ця задача являє собою задачу пошуку найкоротшого маршруту, який включає кожне ребро, принаймні, один раз, і закінчується у початку прямування.

Нехай s − початкова вершина маршруту, $v_{(i, j)} > 0$ – довжина кожного ребра. У графі, поданому на малюнку 27, є декілька шляхів, по котрим листоноша може обминути всі ребра і повернуться у вершину s.



Граф, в якому існує ейлерів маршрут

Рис.27

Сформульованим вимогам задовольняє, наприклад, кожний із нижче приведених чотирьох маршрутів:

Маршрут 1: (s, a), (a, b), (b, c), (c, d), (d, b), (b, s),

Маршрут 2: (s, a), (a, b), (b, d), (d, c), (c, b), (b, s),

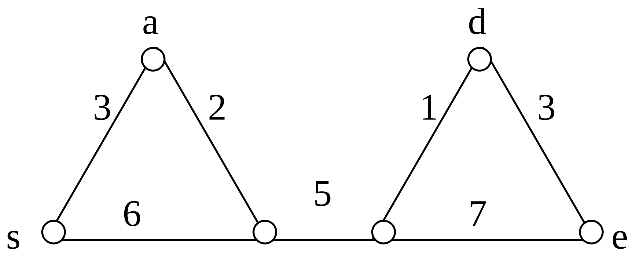
Маршрут 3: (s, b), (b, c), (c, d), (d, b), (b, a), (a, s),

Маршрут 4: (s, b), (b, d), (d, c), (c, b), (b, s), (a, s).

У кожному із чотирьох зазначених маршрутів кожне ребро входить тільки один раз. Таким чином, загальна довжина кожного маршруту дорівнює 3 + 2 + 1 + 3 + 7 + 6 = 22. Кращих маршрутів для листоноші не існує.

Маршрут, у котрому кожне ребро обходиться один разів, називається ейлеровим маршрутом.

Роздивимося граф, поданий на малюнку 28. Очевидно, що на ньому відсутні шляхи листоноші, у котрих ребро (b, c) входило б тільки один разі, тобто відсутній ейлерів маршрут.



Граф, в котрому відсутній ейлерів маршрут

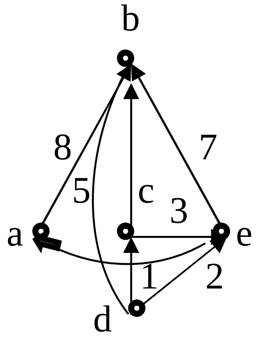
Рис.28

Можливим оптимальним маршрутом (тобто маршрутом найменшої довжини) на даному графі є шлях (s, a), (a, b), (b, c) (із, d), (d, e), (e, c), (із, b), (b, s). Загальна довжина цього маршруту дорівнює 3 + 2 + 5 + 1 + 3 +7 + 5 + 6 = 32.

Спробуємо встановити, чи існують які-небудь інші оптимальні варіанти маршруту на цьому графі? Очевидно, що в незв'язному графі не існує маршруту листоноші (не говорячи вже, про існування ейлерового маршруту). Тому надалі, будемо припускати також, що кількість приходів листоноші в деяку вершину повинно бути рівно кількості відходів його з цієї вершини. При цьому, якщо листоноша не проходить більш одного разу по ребру, інцидентному деякій вершині, то ця вершина повинна бути інцидентна парному числу ребер. Загальна кількість ребер, інцидентних вершині x, назвемо степенем вершини x, і будемо позначати через d(x).

Якщо в графі G усі вершини мають парну степінь, то граф називається парним. **У орієнтованому графі число дуг, які входять у вершину x, називається внутрішнім степенем вершини x; цей степінь позначається через $d^-(x)$. Число дуг, що виходять із вершини x, називається зовнішнім степенем $b^+(x)$ вершини x.**

На малюнку 29, $d^-(a) = 2$, $d^+(a) = 1$. **Якщо в графі для усіх вершин $d^+(a) = d^-(a)$, то такий граф називається симетричним.**



Задача листоноші на орієнтованому графі

Рис.29

Припустимо, що ми знаємо оптимальний шлях листоноші на графі G, який починається і закінчується у вершині s. Яким буде оптимальний маршрут листоноші, якщо в якості початкової вибрати деяку іншу вершину, наприклад вершину t? Відповідь на це питання знаходиться з таких розумінь. Деякий оптимальний маршрут R, що починається у вершині s, у кінцевому рахунку, колись уперше проходить через вершину t. Назвемо цю частину маршруту R_1, а частину , що залишилася , позначимо через R_2. Зауважимо, що R_1починается у вершині s і закінчується у вершині t. У свою чергу R_2 починається у вершині t і закінчується у вершині s. Сформулюємо новий маршрут R', що складається з R_2 і такого за ним R_1. Маршрут R' починається у вершині t, закінчується також у вершині t і має таку ж сумарну довжину, як і маршрут R. Отже, R' повинний бути оптимальним маршрутом, що починається з вершини t.

Таким чином, справедливе таке твердження. Сумарна довжина оптимального маршруту листоноші не залежить від того, яка з вершин даного маршруту буде обрана в якості початкової.

Література

1. Акимов О. Е. Дискретная математика. Логика, группы, графы. – М.: Лаборатория базовых знаний, 2001. – 376с.
2. Басакер Р., Саати Т. Конечные графы и сети. – М.: Наука, 1967. – 368с.
3. Белов В. В., Воробьев Е. М., Шаталов В. Е. Теория графов. – М.: Высшая школа, 1976. – 392с.
4. Березина Л. Ю. Графы и их применение. – М.: Просвещение, 1979. – 144с.
5. Берж К. Теория графов и её применение. – М. ИЛ. 1962. – 319с.
6. Донец Г. А. Шор Н. З. Алгебраический подход к проблеме раскраски плоских графов. – Киев.: Наукова думка, 1982. – 143с.
7. Евстегнеев В. А. Применение теории графов в программировании. – М.: Наука, 1985. – 352с.
8. Зыков А. А. Основы теории графов. – М.: Наука, 1987. – 381с.
9. Зыков А. А. Теория конечных графов. – Новосибирск.: Наука, 1969. – 543с.
10. Капітонова Ю. В., Кривий С. Л., Летичевський О. А., Луцький Г. М., Печурін М. К. Основи дискретної математики. / Підручник. – Київ: Наукова думка, 2002. – 578с.
11. Майника Э. Алгоритмы оптимизации на сетях и графах. – М.: Мир, 1884. – 454с.
12. Мальцев А.И. Алгоритмы и рекурсивные функции, – М.: Наука, 1986. – 368с.
13. Оре О. Теория графов. – М.: Наука, 1980 – 336с.
14. Сваами М., Тсуласираман К. Графы сети и алгоритмы. – М.: Мир, 1984. – 454с.
15. Татт У. Теория графов. – М.: Мир, 1988. – 349с.
16. Уилсон Р. Введение в теорию графов. – М.: Мир, 1977. – 207с.
17. Харри Ф. Теория графов. – М.: Мир, 1973. – 300с.

ЗМІСТ

ТЕМА 1 ОЗНАЧЕННЯ ГРАФІВ, РІЗНОВИДИ ГРАФІВ ………………….. 3
ТЕМА 2 ОПЕРАЦІЇ НАД ГРАФАМИ ……………………………………. 8
ТЕМА 3 ВЛАСТИВОСТІ ГРАФІВ ……………………………………….. 13
ТЕМА 4 МАТРИЦІ І ГРАФИ …………………………………………… 17
ТЕМА 5 РОЗФАРБУВАННЯ ГРАФІВ ………………………………… 19
ТЕМА 6 НЕСКІНЧЕНІ ГРАФИ …………………………………………. 21
ТЕМА 7 ДЕРЕВА ТА ЇХ ВЛАСТИВОСТІ ……………………………… 23
ТЕМА 8 АЛГОРИТМИ ПОШУКУ НАЙКОРОТШОГО ШЛЯХУ ………….. 26
ЛІТЕРАТУРА ……………………………………………………….... 37

Навчальне видання

Кузьменко Вячеслав Віталійович
Швачич Геннадій Григорович
Рижанкова Галина Іванівна
Пасинков Володимир Миколайович

Основи дискретної математики
розділ “Елементи теорії графів”

Конспект лекцій

Тем. План 2004, поз. 225

Підписано до друку 28. 09.04. Формат 60x84 1/16. Папір друк. Друк плоский. Облік.-вид. арк. 2,23 Умов.-друк. арк. 2,25. Тираж 100 пр. Замовлення № .

Національна металургійна академія України
49600, Дніпропетровськ – 5, пр. Гагаріна, 4

Редакційно – видавничий відділ НМетАУ