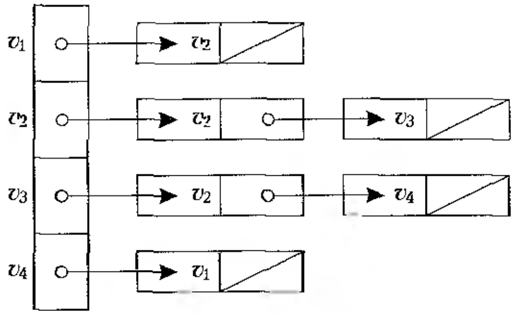


Рис. 3.18

Для орієнтованого графа сума довжин усіх списків суміжних вершин дорівнює загальній кількості дуг: дузі (u, v) відповідає елемент v зі списку $Adj[u]$. Для неорієнтованого графа ця сума дорівнює подвоєній кількості ребер, бо ребро $\{u, v\}$ породжує елемент у списку суміжних вершин як для вершини u, так і для вершини v.

3.4. Шляхи та цикли. Зв'язність

Шляхом довжиною r [52] із вершини u в вершину v в неорієнтованому графі називають послідовність ребер $e_1=\{x_0, x_1\}$, $e_2=\{x_1, x_2\}$, ..., $e_r=\{x_{r-1}, x_r\}$, де $x_0=u$, $x_r=v$, r – натуральне число. Отже, шлях довжиною r має r ребер, причому ребро враховують стільки разів, скільки воно міститься в шляху. Вершини u та v називають *крайніми*, а решту вершин шляху – *внутрішніми*.

Циклом у неорієнтованому графі називають шлях, який з'єднує вершину саму з собою, тобто $u=v$.

У простому графі шлях можна задати послідовністю вершин, через які він проходить: x_0, x_1, x_2, ..., x_{r-1}, x_r.

Шлях або цикл називають *простим*, якщо він не містить повторюваних ребер.

Говорять, що шлях із крайніми вершинами u та v *з'єднує* ці вершини. Шлях, що з'єднує вершини u та v, позначають як $\langle u, v\rangle$ та називають *$\langle u, v\rangle$-шляхом*.

Приклад 3.21. На рис. 3.19 зображено простий граф. У ньому a, d, c, f, e – простий шлях довжиною 4, оскільки пари $\{a, d\}$, $\{d, c\}$, $\{c, f\}$ і $\{f, e\}$ – ребра. Однак d, e, c, b – не шлях, бо пара $\{e, c\}$ – не ребро. Зазначимо, що b, c, f, e, b – цикл довжиною 4, позаяк $\{b, c\}$, $\{c, f\}$, $\{f, e\}$ й $\{e, b\}$ – ребра та цей шлях починається й закінчується в одній і тій самій вершині b. Шлях a, b, e, d, a, b, довжина якого дорівнює 5, не простий, тому що він двічі проходить через ребро $\{a, b\}$.

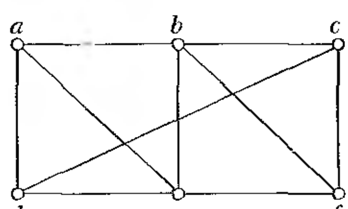


Рис. 3.19

Неорієнтований граф називають *зв'язним*, якщо будь-які дві його вершини з'єднані шляхом. Граф називають *незв'язним*, якщо він не є зв'язним. Незв'язний граф складається з двох або більше зв'язних підграфів, кожна пара з яких не має спільних вершин. Ці зв'язні підграфи називають *компонентами зв'язності* чи просто *компонентами* графа.

Приклад 3.22. Граф G на рис. 3.20 зв'язний; граф H – незв'язний, оскільки не існує шляху $\langle y, v\rangle$. Граф H має дві компоненти.

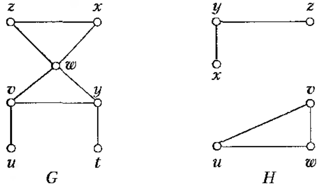


Рис. 3.20

Віддалю $d(u, v)$ між вершинами u та v називають довжину найкоротшого $\langle u, v\rangle$-шляху, а сам цей шлях називають *геодезичним*. Легко переконатись, що найкоротший шлях не містить повторюваних вершин (і ребер). Звідси випливає така теорема.

ТЕОРЕМА 3.4. Між кожною парою різних вершин зв'язного неорієнтованого графа існує простий шлях.

Для орієнтованого графа вводять поняття *орієнтованого шляху* (або просто *шляху*) з вершини u у вершину v. Це скінченна послідовність дуг $e_1 = (x_0, x_1)$, $e_2 = (x_1, x_2)$, ..., $e_r = (x_{r-1}, x_r)$, де $x_0 = u$, $x_r = v$. Вершини u та v, як і в неорієнтованому графі, називають *крайніми*, а решту вершин шляху – *внутрішніми*. *Довжиною* шляху називають кількість дуг, з яких він складається. *Орієнтованим циклом* називають орієнтований шлях, який з'єднує вершину саму із собою, тобто $u = v$. Орієнтований шлях або цикл називають *простим*, якщо жодна дуга не міститься в ньому більше одного разу.

Для орієнтованого графа поняття зв'язності вводять по-різному, залежно від того, чи враховано напрямок.

Орієнтований граф називають *сильно зв'язним*, якщо для будь-яких його різних вершин u та v існують орієнтовані шляхи від u до v та від v до u. Отже, для сильної зв'язності орієнтованого графа має існувати послідовність дуг з урахуванням орієнтації від будь-якої вершини графа до будь-якої іншої.

Орієнтований граф може не бути сильно зв'язним, але може бути, так би мовити, „в одному цілому". У зв'язку з цим дамо таке означення. Орієнтований граф називають *слабко зв'язним*, якщо існує шлях між будь-якими двома різними вершинами у відповідному йому неорієнтованому графі (тобто без урахування напрямку дуг).

Зрозуміло, що сильно зв'язний граф водночас і слабко зв'язний.

Приклад 3.23. На рис. 3.21 зображено графи G й H. Граф G сильно зв'язний. Граф H слабко зв'язний; він не сильно зв'язний, бо не існує орієнтованого шляху від w_1 до w_2.

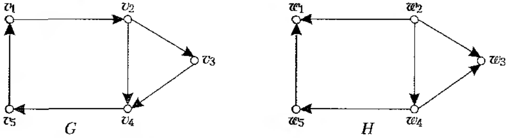


Рис. 3.21

Кількість різних шляхів між двома довільними вершинами графа можна підрахувати за допомогою матриці суміжності.

ТЕОРЕМА 3.5. Нехай G — граф (орієнтований або неорієнтований; можуть бути також кратні ребра й петлі), A — його матриця суміжності, яка відповідає заданій нумерації вершин $v_1, v_2, ..., v_n$. Тоді кількість різних шляхів довжиною r (r — натуральне) з вершини v_i у вершину v_j дорівнює (i, j)-му елементу матриці A^r.

Доведення (методом математичної індукції за r). Для $r = 1$ твердження теореми очевидне: кількість шляхів від v_i до v_j довжиною 1 дорівнює (i, j)-му елементу матриці A, бо цей елемент дорівнює кількості ребер (дуг), які з'єднують v_i та v_j.

Припустимо, що (i, j)-й елемент матриці A^r дорівнює кількості різних шляхів довжиною r від вершини v_i до вершини v_j. Це індуктивна гіпотеза. Оскільки $A^{r+1} = A^rA$, то (i, j)-й елемент матриці A^{r+1} дорівнює $\sum\limits_{k=1}^{n} b_{ik}a_{kj}$, де b_{ik} — (i, k)-й елемент матриці A^r, a_{kj} — (k, j)-й елемент матриці суміжності A.

За індуктивною гіпотезою b_{ik} дорівнює кількості шляхів довжиною r із вершини v_i у вершину v_k. Шлях довжиною $r+1$ із v_i у v_j складається зі шляху довжиною r із v_i до якоїсь проміжної вершини v_k та ребра (дуги) з v_k до v_j. За правилом добутку з комбінаторики кількість таких шляхів дорівнює добутку кількості b_{ik} шляхів довжиною r із v_i до v_k та кількості a_{kj} ребер (дуг) із v_k до v_j, тобто $b_{ik}a_{kj}$. Якщо ці добутки підсумувати для всіх можливих проміжних вершин v_k, то потрібний результат випливає з комбінаторного правила суми.

Розглянемо неорієнтовані графи K_n і C_n (див. рис. 3.9). Обидва ці графи зв'язні проте інтуїтивно зрозуміло, що для $n > 3$ граф K_n „сильніше зв'язаний", ніж граф C_n. Розглянемо два поняття, які характеризують міру зв'язності простого графа.

Числом вершинної зв'язності (або просто *числом зв'язності*) $\kappa(G)$ простого графа G називають найменшу кількість вершин, вилучення яких дає незв'язний або одновершинний граф. Зазначимо, що вершину вилучають разом із інцидентними їй ребрами. Наприклад, $\kappa(K_1) = 0$, $\kappa(K_n) = n - 1$, $\kappa(C_n) = 2$. Граф G, зображений на рис. 3.22, зв'язний, але його зв'язність можна порушити вилученням вершини u. Отже, $\kappa(G) = 1$. Якщо ж спробувати порушити зв'язність цього графа вилученням ребер (а не вершин), то потрібно вилучити не менше ніж три ребра.

Нехай G — простий граф з $n > 1$ вершинами. *Числом реберної зв'язності* $\lambda(G)$ графа G називають найменшу кількість ребер, вилучення яких дає незв'язний граф. Число реберної зв'язності одновершинного графа вважають таким, що дорівнює 0. Для графа G, зображеного на рис. 3.22, $\lambda(G) = 3$.

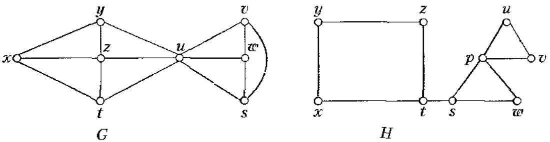


Рис. 3.22

Вершину u простого графа G називають *точкою з'єднання*, якщо граф G в разі її вилучення матиме більше компонент, ніж даний граф G. Зокрема, зв'язний граф G без вершини u стає незв'язним. Нагадаємо, що вершину u при цьому вилучають разом з інцидентними їй ребрами. Ребро графа G називають *мостом*, якщо його вилучення збільшує кількість компонент. Отже, точки з'єднання й мости — це своєрідні „вузькі місця" простого графа.

Приклад 3.24. Граф H, зображений на рис. 3.22, має три точки з'єднання t, s, p й один міст $\{t, s\}$.

Позначимо як $\delta(G)$ мінімальний степінь вершин графа G. Можна довести, що $\kappa(G) \leqslant \lambda(G) \leqslant \delta(G)$. Простий граф G називають *t-зв'язним*, якщо $\kappa(G) \geqslant t$, і *реберно t-зв'язним*, якщо $\lambda(G) \geqslant t$. Отже, відмінний від K_1 граф однозв'язний тоді й лише тоді, коли він зв'язний. Двозв'язні графи — це зв'язні графи без точок з'єднання, відмінні від графа K_1.

Приклад 3.25. Граф G, зображений на рис. 3.22, однозв'язний і реберно 3-зв'язний.

Очевидно, що кількість ребер у зв'язному простому графі з n вершинами не перевищує кількості ребер у графі K_n, тобто $n(n-1)/2$. Але скільки може бути ребер у простому графі з n вершинами й фіксованою кількістю компонент?

ТЕОРЕМА 3.6. Якщо простий граф G має n вершин і k компонент, то кількість m його ребер задовольняє нерівності

$$n-k \leqslant m \leqslant \frac{1}{2}(n-k)(n-k+1).$$

Доведення. Доведемо спочатку верхню оцінку. Нехай G — простий граф з n вершинами, k компонентами й максимальною для таких графів кількістю ребер $m_{\max}$. Очевидно, що кожна компонента графа G — повний граф. Нехай K_p, K_q — дві компоненти, $p \geqslant q > 1$, v — вершина з другої компоненти. Вилучимо з графа всі ребра, інцидентні вершині v, і з'єднаємо цю вершину ребром із кожною вершиною з першої компоненти. Кількість вершин і компонент при цьому не зміниться, а кількість ребер зросте на $p-(q-1)=p-q+1>1$, що неможливо, бо граф G має максимально можливу кількість ребер. Отже, лише одна компонента графа G являє собою повний граф із більшою ніж 1 кількістю вершин $n-(k-1)=n-k+1$. Отже, $m_{max} = (1/2)\ (n-k)(n-k+1)$.

Доведемо нижню оцінку математичною індукцією за кількістю ребер m. Для $m = 0$ твердження очевидне, оскільки тоді $k=n$ і, отже, $0 \leqslant 0$. Нехай тепер $m > 0$, і нижня оцінка справджується для графів із меншою кількістю ребер, ніж m. Припустимо, що граф G має найменшу можливу кількість ребер $m_{\min}$ серед усіх простих графів з n вершинами й k компонентами. Вилучивши довільне ребро, отримаємо граф з n вершинами, $k+1$ компонентою й $m_{\min}-1$ ребром. Для нього справджується припущення індукції: $n-(k-1) \leqslant m_{\min}-1$, звідки випливає нерівність $n-k \leqslant m_{\min}$.

Д. Кеніг сформулював простий критерій дводольності графа в термінах довжин простих циклів.

ТЕОРЕМА 3.7 (Кеніга, 1936 р.). Для того, щоб граф G був дводольним, необхідно й достатньо, щоб він не містив простих циклів із непарною довжиною.

Доведення. Необхідність. Нехай G – дводольний граф, C – один із його простих циклів довжиною k. Пройдемо всі ребра цього циклу, починаючи з вершини v. Зробивши k кроків, повернемось у вершину v. Оскільки кінці кожного ребра містяться в різних підмножинах вершин, то k – парне число.

Достатність. Нехай зв'язний граф $G=(V, E)$ з $n>1$ вершинами не має простих циклів із непарною довжиною та $v \in V$. Побудуємо розбиття $V=A \cup B$ ($A \cap B=\varnothing$) так. Довільну вершину $x \in V$ долучимо до множини A, якщо віддаль $d(x, v)$ парна, а ні, то до множини B. Залишилося довести, що породжені підграфи (див. підрозділ 3.1) $G(A)$ та $G(B)$ порожні. Припустимо, що це не так, тобто існують дві суміжні вершини u та w, які належать одній множині. Тоді жодна з цих вершин не збігається з v, бо $v \in A$, а всі вершини, суміжні з v, належать множині B. Нехай $U=\langle u, v\rangle$ та $W=\langle w, v\rangle$ – геодезичні шляхи, v_1 – остання (якщо починати від v) зі спільних вершин цих шляхів (рис. 3.23). Позначимо як X_U й Y_U відповідно частини шляху U від v до v_1 і від v_1 до u. Аналогічно, як X_W та Y_W позначимо відповідно частини шляху W від v до v_1 і від v_1 до w. Очевидно, що довжини шляхів X_U й X_W збігаються (тому ці шляхи геодезичні). Отже, довжини шляхів Y_U й Y_W мають один тип парності (позаяк вершини u та w належать одній множині, то їх віддалі від вершини v мають один тип парності). Але тоді об'єднання шляхів Y_U й Y_W та ребра $\{u, w\}$ являє собою простий цикл із непарною довжиною. Суперечність.

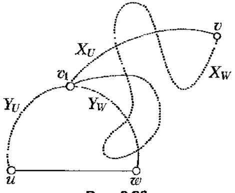


Рис. 3.23

Доведення теореми Кеніга підказує простий спосіб розпізнавання дводольності графа, що ґрунтується на простому алгоритмі, називаному пошуком ушир [15]. Множину вершин, суміжних із вершиною v, називають *оточенням* вершини v. ***Пошук ушир*** нумерує вершини графа. Починають із довільної вершини, надають їй номер 0. Кожній вершині з оточення вершини 0 приписують номер 1. Тепер розглядають почергово оточення всіх вершин із номером 1, і всім вершинам, що належать цим оточенням і ще не мають номера, надають номер 2. Розглядають оточення всіх вершин із номером 2 та продовжують процес нумерації, доки це можливо. Якщо даний граф $G=(V, E)$ зв'язний, то пошук ушир занумерує всі його вершини.

Далі розіб'ємо множину вершин V на дві підмножини — A та B. До множини A долучимо всі вершини з парними номерами (та 0), а до множини B — з непарними. Розглянемо породжені підграфи $G(A)$ та $G(B)$. Якщо обидва вони порожні (достатньо перевірити, що всі пари вершин з однаковими номерами не суміжні), то G — дводольний граф, а ні, то — не дводольний.

Існує й інший, більш розповсюджений варіант пошуку вшир. Він відрізняється тим, що всі вершини отримують різні номери (див. підрозділ 3.9).

3.5. Ізоморфізм графів

У теорії графів і її застосуваннях істотно, що існують об'єкти (вершини графа) і зв'язки між ними (ребра). Тому доцільно не розрізняти графи, котрі можна одержати один з одного зміною позначень вершин. Сформулюємо ці міркування у вигляді такого означення.

Нехай $G_1=(V_1, E_1)$ і $G_2=(V_2, E_2)$ — прості графи, а $\varphi\colon V_1 \to V_2$ — бієкція. Якщо для будь-яких вершин u та v графа G_1 їх образи $\varphi(u)$ та $\varphi(v)$ суміжні в G_2 тоді й лише тоді, коли u та v суміжні в G_1, то цю бієкцію називають ***ізоморфізмом*** графа G_1 на граф G_2, а графи G_1 і G_2 — ***ізоморфними*** (записують $G_1 \cong G_2$).

Отже, прості графи G_1 і G_2 ізоморфні, якщо існує така бієкція $\varphi\colon V_1 \to V_2$, що

$$\forall\, u, v \in V_1\ (\{u, v\} \in E_1 \text{ тоді й лише тоді, коли } \{\varphi(u), \varphi(v)\} \in E_2).$$

Приклад 3.26. Графи на рис. 3.24 ізоморфні; бієкцію φ можна задати так: $\varphi(x_1)=y_1$; $\varphi(x_2)=y_4$; $\varphi(x_3)=y_3$; $\varphi(x_4)=y_2$.

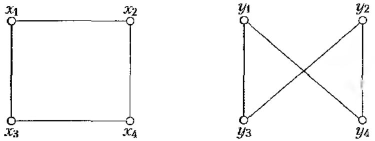


Рис. 3.24

Приклад 3.27. Усі три графи, зображені на рис. 3.25, ізоморфні. Довести це пропонуємо як вправу.

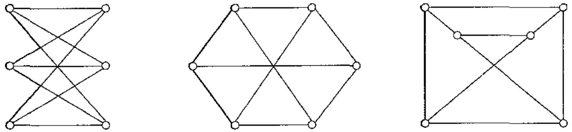


Рис. 3.25

Ізоморфні графи природно ототожнювати (їх можна зобразити одним рисунком). Вони могли б різнитися природою своїх елементів, але це не беруть до уваги, уводячи поняття „граф". Проте інколи все ж доводиться розрізняти ізоморфні графи, і тоді корисне поняття „позначеного графа". Граф з n вершинами називають *позначеним*, якщо його вершинам присвоєно якісь мітки, наприклад, числа 1, 2, ..., n. Ототожнимо кожну з вершин з її номером (і, отже, множину вершин – із множиною чисел $\{1, 2, ..., n\}$) й означимо рівність простих позначених графів G_1 і G_2 з однаковою кількістю вершин n так: $G_1 = G_2$ тоді й лише тоді, коли $E_1 = E_2$.

Приклад 3.28. На рис. 3.26 зображено три різні позначені графи.

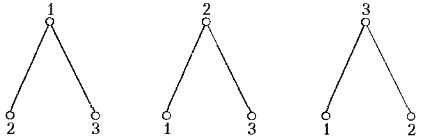


Рис. 3.26

Щоб наголосити, що графи розрізняють лише з точністю до ізоморфізму, говорять про абстрактний граф. Останній має різні матриці суміжності залежно від нумерації вершин. З'ясуємо, як пов'язані між собою ці матриці. Нехай G_1 і G_2 – прості позначені графи з n вершинами, і $G_1 \cong G_2$. Тоді ці графи різняться лише нумерацією вершин, тобто існує підстановка s на множині вершин, яка *зберігає суміжність*: вершини u та v тоді й лише тоді суміжні в графі G_1, коли їх образи $s(u)$ та $s(v)$ суміжні в G_2. Нехай $A = [a_{ij}]$ і $B = [b_{ij}]$ – матриці суміжності відповідно графів G_1 та G_2. Тоді $b_{s(i)s(j)} = a_{ij}$ $(i, j = 1, 2, ..., n)$. Тим самим доведено таку теорему.

ТЕОРЕМА 3.8. Прості графи ізоморфні тоді й лише тоді, коли їх матриці суміжності можна отримати одну з одної однаковими перестановками рядків і стовпців.

Виявити ізоморфізм дуже складно. Теоретично алгоритм перевірки пари простих графів на ізоморфізм існує – його сформульовано в попередній теоремі. Проте він незручний, бо для його виконання може бути потрібно до $n!$ перестановок і перевірок.

Часто неважко довести, що два прості графи не ізоморфні, якщо порушується властивість, інваріантна щодо ізоморфізму, наприклад:

- кількість вершин;
- кількість ребер;
- кількість вершин конкретного степеня (вершині $v \in V_1$, $\deg(v) = d$, має відповідати вершина $u = \varphi(v) \in V_2$, $\deg(u) = d$).

Є й інші інваріанти, але порушення інваріантності – це лише достатня умова неізоморфності графів. Не існує набору інваріантів для виявлення ізоморфізму.

Приклад 3.29. Графи на рис. 3.27 неізоморфні. Вони мають по п'ять вершин і по шість ребер, проте граф G_2 має вершину степеня 1, якої не має граф G_1.

Рис. 3.27

Приклад 3.30. На рис. 3.28 зображено графи G_1 і G_2. Обидва вони мають по вісім вершин і по десять ребер. Вони також мають по чотири вершини степеня 2 і по чотири вершини степеня 3. *Однак ці графи не ізоморфні. Справді, позаяк* $\deg(x_1) = 2$ *в* G_1, *то вершині* x_1 має відповідати одна із чотирьох вершин y_2, y_4, y_6, y_8 у графі G_2. Зазначені вершини мають у графі G_2 степінь 2. Проте кожна з цих чотирьох вершин суміжна з іншою вершиною степеня 2, що не виконується для вершини x_1 у графі G_1.

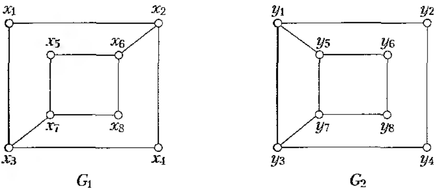
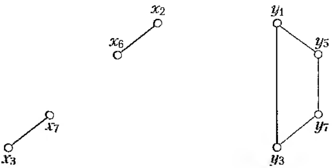


Рис. 3.28

Те, що графи, зображені на рис. 3.28, не ізоморфні, можна довести й інакше. На рис. 3.29 зображено підграфи графів G_1 і G_2, породжені вершинами степеня 3. Якщо графи G_1 і G_2 ізоморфні, то й зазначені підграфи мають бути ізоморфними. Проте підграфи з рис. 3.29 не ізоморфні.



Підграф G_1 Підграф G_2

Рис. 3.29

Ми розглянули поняття ізоморфізму для простого графа. Для неорієнтованих мультиграфів і псевдографів, а також орієнтованих графів природно вводять поняття ізоморфізму як бієкції між множинами вершин, яка зберігає суміжність, кратності ребер, петлі та спрямування дуг. Теорема 3.8 залишається правильною для мультиграфів, псевдографів і орієнтованих графів.

Розглянемо тепер один результат, пов'язаний з оцінкою складності алгоритмів на графах, які задано матрицею суміжності. Нехай P — якась властивість простого графа: $P(G)=1$ або $P(G)=0$ залежно від того, чи має граф G цю властивість. Припустимо, що властивість P задовольняє такі умови.

1. $P(G)=P(G')$, якщо графи G та G' ізоморфні.
2. $P(G)=0$ для довільного порожнього графа й $P(G)=1$ для довільного повного графа з достатньо великою кількістю вершин.
3. Додавання ребра не порушує властивості P, тобто $P(G)\leqslant P(G')$ для довільних простих графів $G=(V, E)$ та $G'=(V, E')$ таких, що $E\subset E'$.

Приклад такої властивості P — існування простого циклу в графі, який має принаймні три вершини.

ТЕОРЕМА 3.9. Якщо P — властивість простого графа G, що задовольняє умови 1–3, то будь-який алгоритм, котрий перевіряє властивість P (тобто обчислює значення $P(G)$ для графа G), виходячи з матриці суміжності, виконує в найгіршому випадку $\Omega(n^2)$ кроків, де n — кількість вершин графа.

3.6. Ейлерів цикл у графі

Початок теорії графів як розділу математики пов'язують із задачею про кенігсберзькі мости. Сім мостів міста Кенігсберга (нині – Калінінград у Росії) було розміщено на річці Прегель так, як зображено на рис. 3.30. Чи можна, починаючи з якоїсь точки міста, пройти через усі мости точно по одному разу й повернутись у початкову точку? Швейцарський математик Л. Ейлер розв'язав цю задачу. Його розв'язання, опубліковане 1736 р., було першим явним застосуванням теорії графів. Ейлер поставив у відповідність плану міста мультиграф G, вершини якого відповідають чотирьом частинам A, B, C, D міста, а ребра — мостам. Цей мультиграф зображено на рис. 3.31.

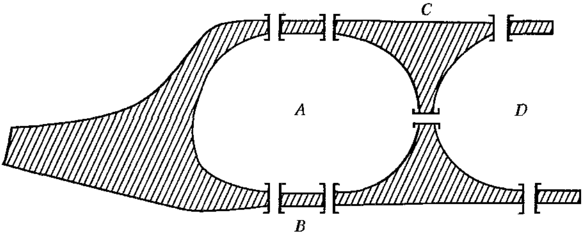


Рис. 3.30

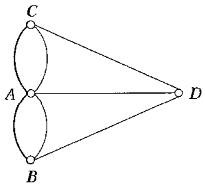


Рис. 3.31

Отже, задачу про кенігсберзькі мости мовою теорії графів можна сформулювати так: чи існує в мультиграфі G простий цикл, який містить усі ребра цього мультиграфа? Ейлер довів нерозв'язність задачі про кенігсберзькі мости. Нагадаємо, що в простому циклі ребра не повторюються, а вершини можуть повторюватись.

Ейлеровим циклом у зв'язному мультиграфі G називають простий цикл, який містить усі ребра графа, ***ейлеровим шляхом*** — простий шлях, що містить усі ребра графа.

Приклад 3.31. На рис. 3.32 проілюстровано концепцію ейлерових циклів і шляхів. Граф G_1 має ейлерів цикл, наприклад, a, e, c, d, e, b, a; граф G_3 не має ейлерового циклу, але має ейлерів шлях: a, c, d, e, b, d, a, b; граф G_2 не має ні ейлерового циклу, ні ейлерового шляху.

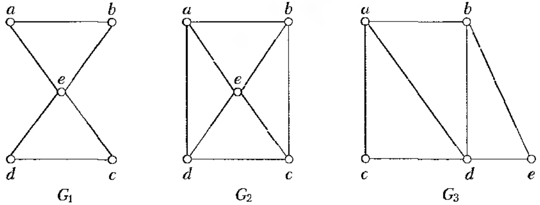


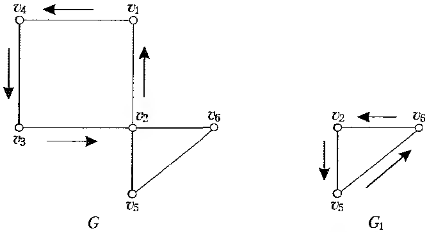
Рис. 3.32

Існує простий критерій (необхідна й достатня умова) для виявлення того, чи має граф ейлерів цикл.

ТЕОРЕМА 3.10. Зв'язний мультиграф G має ейлерів цикл тоді й лише тоді, коли степені всіх його вершин парні.

Доведення. Необхідність. Нехай у графі G існує ейлерів цикл. Тоді він проходить через кожну вершину графа та входить до неї по одному ребру, а виходить по іншому. Це означає, що кожна вершина інцидентна парній кількості ребер ейлерового циклу. Оскільки такий цикл містить усі ребра графа G, то звідси випливає парність степенів усіх його вершин.

Достатність. Припустимо тепер, що всі вершини графа G мають парний степінь. Почнемо шлях P_1 із довільної вершини v_1 і продовжимо його, наскільки це можливо, вибираючи щоразу нове ребро. Позаяк степені всіх вершин парні, то, увійшовши в будь-яку вершину, відмінну від v_1, ми завжди маємо можливість вийти з неї через іще не пройдене ребро. Тому шлях P_1 можна продовжити, додавши це ребро. Отже, побудова шляху P_1 завершиться у вершині v_1, тобто P_1 обов'язково виявиться циклом. Якщо з'ясується, що шлях P_1 містить усі ребра графа G, то це ейлерів цикл. У протилежному випадку вилучимо з G всі ребра циклу P_1 і всі вершини, інцидентні лише вилученим ребрам. Отримаємо якийсь зв'язний граф G_1. Оскільки P_1 та G мають вершини лише парних степенів, то, очевидно, і граф G_1 матиме цю властивість. Окрім того, позаяк граф G зв'язний, то графи P_1 і G_1 мають принаймні одну спільну вершину v_2. Тепер із вершини v_2 побудуємо цикл P_2 в графі G_1 аналогічно до того, як ми будували цикл P_1 у графі G. Цикл P_2 вставимо в цикл P_1 на місце вершини v_2. Одержимо цикл P_3. Описані побудови показано на рис. 3.33.



$P_1 = v_1, v_4, v_3, v_2, v_1$ $\qquad$ $P_2 = v_2, v_5, v_6, v_2$

Рис. 3.33

Цикл $P_3 = v_1, v_4, v_3, P_2, v_1 = v_1, v_4, v_3, v_2, v_5, v_6, v_2, v_1$ ейлерів. Якби він виявився не ейлеровим, то потрібно продовжити аналогічні побудови й отримати ще більший цикл. Цей процес закінчиться побудовою ейлерового циклу. Зазначимо, що доведення достатності має конструктивний характер: подано алгоритм побудови ейлерового циклу.

Існує й інший алгоритм побудови ейлерового циклу, який дає змогу побудувати цей цикл одразу. Це алгоритм Флері (Fleury).

Алгоритм Флері побудови ейлерового циклу

Робота алгоритму полягає в нумерації ребер у процесі побудови ейлерового циклу.

Крок 1. Початок. Починаємо з довільної вершини u та присвоюємо довільному ребру $\{u, v\}$ номер 1. Викреслюємо ребро $\{u, v\}$ й переходимо у вершину v.

Крок 2. Ітерація. Нехай v — вершина, у яку ми перейшли на попередньому кроці, k — останній присвоєний номер. Вибираємо довільне ребро, інцидентне вершині v, причому міст вибираємо лише тоді, коли немає інших можливостей. Присвоюємо вибраному ребру номер $(k+1)$ і викреслюємо його.

Крок 3. Закінчення. Цей процес закінчуємо, коли всі ребра графа викреслено та пронумеровано — ці номери задають послідовність ребер в ейлеровому циклі.

Повертаючись до задачі про кенігсберзькі мости, виявляємо, що мультиграф, зображений на рис. 3.31, має всі вершини непарного степеня. Отже, цей мультиграф не має ейлерового циклу, тому неможливо пройти кожний міст по одному разу й повернутись у початкову точку шляху.

ТЕОРЕМА 3.11. Зв'язний мультиграф має ейлерів шлях, але не має ейлерового циклу тоді й лише тоді, коли він має точно дві вершини непарного степеня.

Зазначимо, що будь-який ейлерів шлях починається в одній із цих двох вершин непарного степеня, а закінчується в іншій. Оскільки мультиграф для кенігсберзьких мостів має чотири вершини з непарними степенями, можна дійти висновку про неможливість пройти кожний міст по одному разу, навіть якщо не потрібно повертатись у початкову точку.

Граф, який має ейлерів цикл, часто називають ***ейлеровим***.

Ейлеровим циклом у слабко зв'язному орієнтованому мультиграфі називають орієнтований простий цикл, який містить усі ребра графа.

ТЕОРЕМА 3.12. Орієнтований слабко зв'язний мультиграф має ейлерів цикл тоді й лише тоді, коли напівстепінь входу кожної вершини дорівнює її напівстепеню виходу.

3.7. Гамільтонів цикл у графі

Шлях $x_0, x_1, ..., x_{n-1}, x_n$ у зв'язному графі $G=(V, E)$ називають ***гамільтоновим шляхом***, якщо $V=\{x_0, x_1, ..., x_{n-1}, x_n\}$ і $x_i \neq x_j$ для $0 \leqslant i < j \leqslant n$. Цикл $x_0, x_1, ..., x_{n-1}, x_n, x_0$ (тут $n > 1$) у графі $G=(V, E)$ називають ***гамільтоновим циклом***, якщо $x_0, x_1, ..., x_{n-1}, x_n$ — гамільтонів шлях. Інакше кажучи, гамільтонів цикл і гамільтонів шлях проходять через кожну вершину графа точно один раз (можливо, окрім першої й останньої вершин). Зазначимо, що гамільтонові цикл і шлях, узагалі кажучи, не містять усіх ребер графа.

Термін „гамільтонів" у цих означеннях походить від імені відомого ірландського математика У. Гамільтона (W. Hamilton), який 1857 р. запропонував гру „Навколосвітня подорож". Кожній із двадцяти вершин додекаедра (правильного дванадцятигранника, грані якого — п'ятикутники) приписано назву одного з великих міст світу. Потрібно, розпочавши з довільного міста, відвідати решту 19 міст точно один раз і повернутись у початкове місто. Перехід дозволено ребрами додекаедра [52].

Приклад 3.32. Ту саму задачу можна зобразити й на площині (рис. 3.34). Вона зводиться до відшукання в графі гамільтонового циклу. Один із можливих розв'язків показано потовщеними лініями.

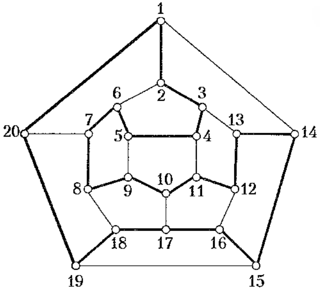


Рис. 3.34

Не всі зв'язні графи мають гамільтонів цикл хоча б тому, що такий граф має бути двозв'язним (тобто граф, який має точки з'єднання, не може мати гамільтонового циклу). Приклад графа, зображеного на рис. 3.35, свідчить, що двозв'язності недостатньо для наявності гамільтонового циклу. Незважаючи на зовнішню подібність формулювань задач про існування ейлерового й гамільтонового циклів, ці задачі принципово різні. Використовуючи результати попереднього підрозділа, легко виявити, чи має граф ейлерів цикл, і, якщо має, то побудувати його.

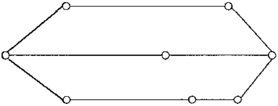


Рис. 3.35

Ситуація для гамільтонового циклу істотно інша. Відповісти на питання, чи має граф гамільтонів цикл, зазвичай, дуже важко. Вивчення достатніх умов наявності в графі гамільтонового циклу — один із важливих напрямів у теорії графів. Інтуїтивно зрозуміло, що граф із багатьма ребрами, достатньо рівномірно розподіленими, з великою ймовірністю має гамільтонів цикл. Доведемо одну з теорем такого типу.

ТЕОРЕМА 3.13 (Г. Дірака, 1952 р.). Якщо для кожної вершини v зв'язного простого графа з $n \geqslant 3$ вершинами виконується нерівність $\deg(v) \geqslant n/2$, то цей граф має гамільтонів цикл.

Доведення. Додамо до графа G k нових вершин і з'єднаємо ребром кожну з них із кожною вершиною з G. Отриманий граф з $n+k$ вершинами позначимо як G'.

Уважатимемо, що k – найменша кількість вершин, потрібних для того, щоб у графі G з'явився гамільтонів цикл. Доведемо, що припущення $k \geqslant 1$ призводить до суперечності.

Нехай $v, p, w, ..., v$ – гамільтонів цикл у графі G', де v та w – вершини з G, а p – одна з нових вершин. Тоді вершина w несуміжна з v, інакше ми могли б не використовувати вершину p, що суперечить мінімальності числа k. Більше того, вершина w', суміжна з вершиною w, не може в гамільтоновому циклі безпосередньо йти за вершиною v', суміжною з v. Справді, якщо є гамільтонів цикл $v, p, w, ..., v', w', ..., v$, то ми можемо замінити його на $v, v', ..., w, w', ..., v$, повернувши частину циклу між w та v'. Це знову суперечить мінімальності числа k (рис. 3.36).

Суміжні

$v, p, w, ..., v', w', ..., v$

Суміжні

Рис. 3.36

Отже, кількість вершин графа G', не суміжних з w, не менша від кількості вершин, суміжних з v (тобто дорівнює принаймні $n/2+k$). Натомість очевидно, що кількість вершин графа G', суміжних з w, також дорівнює принаймні $n/2+k$. Але жодна з вершин графа G' не може бути водночасно суміжною й несуміжною з вершиною w, тому загальна кількість вершин графа G' дорівнює щонайменше $n+2k$. Але це суперечить тому, що кількість вершин графа G' дорівнює $n+k$.

Як знайти гамільтонів цикл або переконатись, що його немає? Очевидний алгоритм, який можна застосувати, – це повний перебір усіх можливостей, тобто $n!$ перестановок усіх вершин графа й перевірок. Зрозуміло, що такий спосіб не практичний. Розгляд реального алгоритму бектрекінг відкладемо до наступного розділу (розділ 4.5, приклад 4.15).

Задача знаходження гамільтонового циклу NP-повна (див. розділ 10). Для таких задач невідомий ефективний (тобто з поліноміальною складністю) алгоритм їх розв'язання, і є вагомі підстави вважати, що його не існує.

Граф, який містить гамільтонів цикл, часто називають *гамільтоновим графом*.

3.8. Зважені графи й алгоритми пошуку найкоротших шляхів

У реальних задачах на графах часто потрібно брати до уваги додаткову інформацію – фактичну віддаль між окремими пунктами, вартість проїзду, час проїзду тощо. Для цього використовують поняття зваженого графа.

Зваженим називають простий граф, кожному ребру e якого приписано дійсне число $w(e)$. Це число називають *вагою ребра e*. Аналогічно означають *зважений*