Міністерство освіти і науки України
Чернівецький національний університет
імені Юрія Федьковича

Тетяна КАРАВАНОВА

Теорія алгоритмів

Частина 2. Обчислювальні алгоритми

Навчальний посібник

Чернівці
Чернівецький національний університет
імені Юрія Федьковича
2022

УДК 519.16(072)
К 21

Друкується за ухвалою Вченої ради
Чернівецького національного університету імені Юрія Федьковича
(протокол №11 від 31.10.2022 р.)

Рецензенти:

Міца О.В., доктор технічних наук, доцент, зав. кафедри інформаційних управляючих систем та технологій ДВНЗ «Ужгородський національний університет»;

Чернікова Л.А., кандидат педагогічних наук, доцент, проректор з навчально-методичної роботи комунального закладу «Запорізький обласний інститут післядипломної педагогічної освіти» Запорізької обласної ради.

Караванова Т.П.

К 21 Теорія алгоритмів. Частина 2. Обчислювальні алгоритми : навч. посіб. / Т.П. Караванова. Чернівці : Чернівец. нац. ун-т ім. Ю. Федьковича. 2022. 288 с.

Навчальний посібник містить достатньо повний об'єм матеріалу, необхідного для оволодіння навичками побудови оптимізаційних алгоритмів різної складності. Даний навчальний посібник може бути корисним як при вивченні курсу «Теорія алгоритмів», так і під час підготовки студентів до інтелектуальних змагань з програмування.

Для студентів вищих навчальних закладів спеціальностей «Комп'ютерні науки», «Системний аналіз», «Прикладна математика» та інших технічних спеціальнсотей.

УДК 519.16(072)

Від автора

У першій частині посібника «Терорія алгоритмів», який доповнений назвою «Необчислювальні алгоритми», розглядалися основні положення теорії алгоритмів, базові алгоритмічні конфігурації, структури даних, що лежать в основі побудови будь-яких інших алгоритмів, пошукові алгоритми та алгоритми сортування, які за своєю методикою використовують лише операції порівняння та обміну.

Частіше за все розв'язання складних алгоритмічних задач, метою яких є пошук кращого з усіх можливих варіантів розв'язків, може бути реалізовано за допомогою повнопереборних алгоритмів. Однак існують різні методи, які дозволяють оптимізувати такі задачі і розв'язати їх за більш короткий час. Методи розв'язування таких задач представлені в алгоритмізації різними розділами: теорії графів, динамічного програмування, жадібних алгоритмів, обчислювальної геометрії тощо.

Метою створення даного навчального посібника є вироблення навичок грамотного тестування розроблених алгоритмів, умінь точного оцінювання ефективностіїх роботи, вибору найбільш підходящих для кожної конкретної задачі оптимізаційних методів.

Достатній досвід роботи автора дозволяє мати надію, що даний навчальний посібник стане корисним студентам не тільки під час підготовки до занять з курсу «Теорія алгоритмів», але й у подальшій професійній роботі, під час підготовки до участі в інтелектуальних змаганнях з програмування.

Тетяна Караванова

Алгоритми на графах

Теорія графів це один із розділів математики, що дозволяє формалізувати взаємозв'язки між різноманітними видами інформації, організувати їх абстрактне представлення.

Існує клас задач, який носить назву класу NP-задач, для знаходження оптимальних розв'язків яких необхідно перебрати всі можливі варіанти і серед них визначити найкращі відповідно до поставлених умов. Зрозуміло, що за часовою складністю такий підхід до розв'язання подібних задач значно програє.

Матеріал розділів цієї частини посібника має на меті ознайомити з оптимізаційними методами розв'язування алгоритмічних задач, тобто з такими, які дозволяють знайти розв'язок задачі за найменшу кількість ітерацій. Це можливо здійснити за рахунок відкидання на кожному проміжному кроці виконання алгоритму «неперспективних» варіантів, тобто таких, які не приведуть до шуканого результату.

Знаючи різні методи розв'язування задач і коректно використовуючи їх при побудові алгоритмів можна значно скоротити час виконання таких алгоритмів.

Переходячи до теорії графів, слід нагадати, що попередньо ми побічно познайомилися з графами. Це був пошук у мережі, де сама мережа була схематично зображена графом, у структурах даних працювали з деревом, яке теж є різновидом графа.

Можна також задатися питанням щодо змісту прикладних задач, які можуть базуватися на теорії графів. Не вдаючись у тлумачення мети розв'язання цих задач, про які йтиметься у цьому розділі, наведемо лише умови їх вирішення. Це можуть бути такі прості і зрозумілі приклади: представлення шляхів між окремими населеними пунктами, фінансові взаємовідносини між різними банками (грошові позики або борги), аудиторії та предмети, які можуть у них викладатися, у навчальному закладі, програми навігатори, що використовуються на мобільних пристроях тощо.

Основні поняття теорії графів

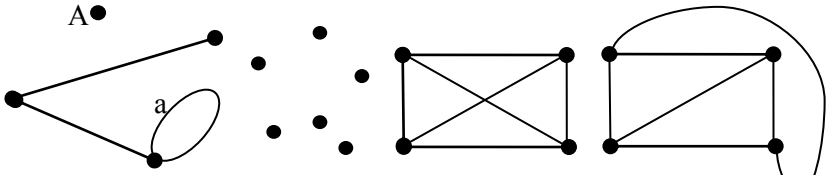
Графом називається сукупність точок (вершин) таліній (ребер), що їх з'єднують.

Можна також ще сказати, що *графом називається сукупність точок і ліній, в якій кінці кожної лінії належать множині точок.*

Якщо ребро з'єднує дві вершини, то кажуть, що воно ***інцидентно*** цим вершинам, а вершини, які з'єднані таким ребром називаються ***суміжними***. Якщо кінці ребра належать одній вершині, то таке ребро називається ***петлею***. На малюнку мал.1а) ребро *а* є петлею.

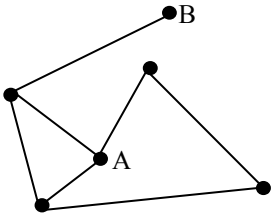
Вершини, які не належать кінцям жодного з ребер у графі, називаються ***ізольованими***. Прикладом ізольованої вершини на малюнку 1а) є вершина *А*. Граф, який складається лише з ізольованих вершин, називається ***нуль-графом*** (мал.1б). До речі, а чи може в графі існувати ребро без вершин? Ні, лінія, хоча б один із кінців якої не належить множині вершин графа, не є ребром.

Граф, у якому будь-яка пара вершин з'єднана ребрами, називається ***повним*** (мал.1 в)).

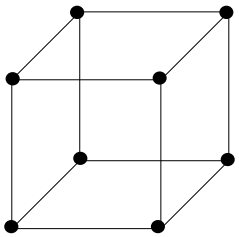


Мал. 1 а) Мал. 1б) Мал. 1 в)

Якщо всі вершини і ребра графу знаходяться в одній площині, то він називається ***плоским*** (мал.2а)), у протилежному випадку **—** ***просторовим*** (мал.2б)).



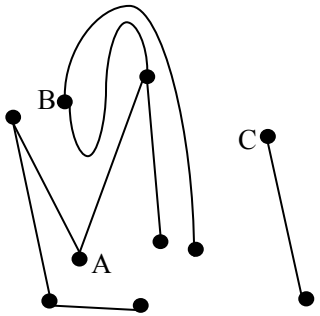
Мал. 2 а)



Мал. 2 б)

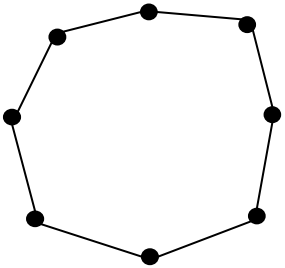
Степенем вершини будемо називати число ребер, яким належить ця вершина. Або ще можна сказати, що степенем вершини називається кількість ребер, інцидентних даній вершині. Позначається степінь вершини a як $P(a)$. Наприклад, вершина A на мал.2а) має степінь 3, а вершина B — степінь 1: $P(A)=3$; $P(B)=1$.

Нехай задано граф з N вершинами, які позначені $a_1,a_2,...,a_N$. Кажуть, що існує ***шлях*** від вершини a_i до вершини a_j, якщо існує послідовність ребер, яка з'єднує вершини a_i та a_j. З малюнку 3 видно, що існує шлях між вершинами A та B, оскільки існує така послідовність ребер, що їх з'єднує. У свою чергу кажуть, що вершини a_i та a_j є ***зв'язними***, якщо існує шлях від a_i до a_j. Це означає, що вище названі вершини A та B є зв'язними. А от вершини A та C, а також B та C з малюнку 3 — не зв'язні, оскільки не існує жодного шляху, яким можна було б дістатися від однієї із вказаних вершин до іншої.

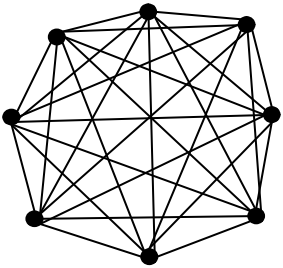


Мал. 3

Надалі називатимемо граф ***зв'язним***, якщо будь-яка пара його вершин зв'язна. Зрозуміло, що повний граф завжди є зв'язним, але не всякий зв'язний граф є повним. Прикладом може бути будь-який багатокутник: він зв'язний, але не повний (мал.4а)). Але якщо у цьому многокутнику провести всі діагоналі, то отримаємо повний зв'язний граф (мал.4б)).

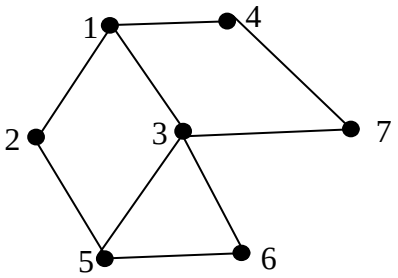


Мал. 4 а)



Мал. 4 б)

Введемо ще один термін: ***циклом*** **називається шлях, в якому збігаються початкова і кінцева вершини.** На малюнку 5 зображено граф, який для кожної вершини має подекілька циклів. Наприклад, для вершини **1** існує 6 циклів: (1-2, 2-5, 5-3, 3-1), (1-3, 3-7, 7-4, 4-1), (1-3, 3-5, 5-6, 6-3, 3-1), (1-2, 2-5, 5-6, 6-3, 3-1), (1-2, 2-5, 5-6, 6-3, 3-7, 7-4, 4-1), (1-3, 3-5, 5-6, 6-3, 3-7, 7-4, 4-1) .



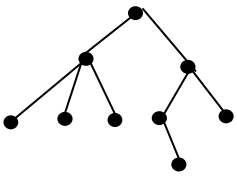
Мал. 5

Якщо цикл через кожну вершину проходиться лише один раз, то такий цикл називається ***простим***. Як видно з малюнку 5 для вершини**1** існує чотири простих цикли: (1-2, 2-5, 5-3, 3-1), (1-3, 3-7, 7-4, 4-1), (1-2, 2-5, 5-6, 6-3, 3-1), (1-2, 2-5, 5-6, 6-3, 3-7, 7-4, 4-1).

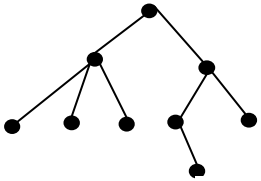
Довжиною шляху **(циклу) називається кількість ребер цього шляху (циклу).** На малюнку 5 для кожного вказаного циклу вершини **1** неважко порахувати довжину: 4, 4, 5, 5, 7, 7.

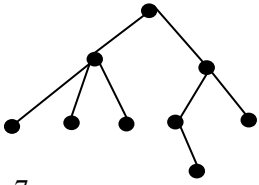
Граф, який не має жодного циклу, називається ***деревом*** (мал.6). **Декілька дерев, які не мають спільних вершин, називаються** ***лісом*** (мал.7). Попередньо вже розглядалися

дерева, як структури даних, але при цьому обмежувалися випадком, коли кількість нащадків кожної вершини була однаковою.



Мал. 6

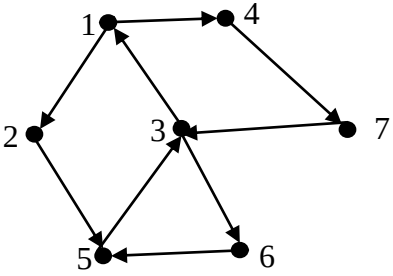




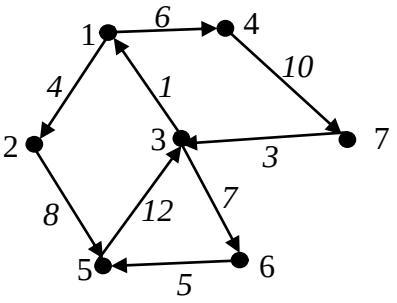
Мал. 7

Розглядаючи дерево як граф надалі будемо розв'язувати більш загальні задачі.

Досі йшлося про графи, у яких ребра не мають орієнтації, тобто на малюнку 5 ребра 1-2 та 2-1 абсолютно рівнозначні. Проте не у всіх задачах така ситуація може не мати значення. Наприклад, якщо мова йтиметься про географічну відстань між двома містами, то у цьому випадку нас, можливо, і не буде цікавити напрямок руху між ними. Але якщо метою задачі є визначення економії палива на цьому відрізку шляху, а один населений пункт розміщений на горі, а другий — внизині, то ребра 1-2 та 2-1 будуть «різновартісними» і для них необхідно стрілкою вказувати напрямок. Такий граф, у якому для всіх ребер вказано відповідний напрямок, називатимемо ***орієнтованим*** або ***орграфом*** (мал. 8).



Мал. 8



Мал. 9

Для неорієнтованого ребра графа порядок, в якому вказуються вершини, які ним з'єднуються, не важливий. Для орієнтованого ребра першою вказується вершина, з якої воно виходить. В орграфі також існує поняття орієнтованого шляху: це послідовність ребер між двома вершинами з урахуванням їх орієнтації. Шлях в орграфі, який є циклом, називається орієнтованим циклом. До речі, порівняно із неорієнтованим графом, зображеним на малюнку 5, в орієнтованому графі на малюнку 8 для вершини **1** існує вже тільки два орієнтованих цикли: (1-2, 2-5, 5-3, 3-1) та (1-4, 4-7, 7-3, 3-1). Вони ж одночасно є і простими. Зв'язність орієнтованого графа визначається без урахування орієнтації його ребер. Для орієнтованих графів специфічним є і визначення степенів вершин. Для кожної з них окремо визначається ***вхідна степінь,*** що дорівнює кількості ребер, які входять у вершину, і ***вихідна степінь***що визначається кількістю ребер, які виходять із неї. Сума вхідної і вихідної степенів вершини орієнтованого графа називається степенем цієї вершини.

Якщо ж у графі вказана ще й «вага» кожного ребра, то такий граф називається ***зваженим.*** Тобто існують неорієнтовані зважені графи та орієнтовані зважені графи (мал. 9). Прикладом може бути задача, коли задається напрямок руху дорогами між населеними пунктами та вартість перевезення ними деякого вантажу.

Слід ввести поняття ще двох специфічних графів, з якими нам доведеться мати справу при розв'язанні задач. Це ***ейлерів*** та ***гамільтонів*** графи.

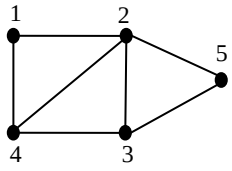
Граф називається ***ейлеровим***, якщо у ньому можна повернутися в ту саму вершину, з якої ми вийшли, обійшовши всі ребра тільки по одному разу. Існують також відповідно поняття ейлерових шляхів та циклів у графі.

Граф називається ***гамільтоновим***, якщо у ньому можна повернутися в ту саму вершину, з якої ми вийшли, обійшовши всі вершини тільки по одному разу. Зрозуміло, що можна говорити про існування гамільтонових шляхів та циклів у графах.

Способи представлення графів

Розглянуто графічне представлення різноманітних графів, яке є досить наочним. Але для складання алгоритму щодо розв'язання задачі, що базується на графах, не важливі геометричні деталі малюнка. Вважається, що граф однозначно заданий, коли задані множини його вершин та ребер і вказані всі зв'язки цих елементів графа, тобто відомо які вершини якими ребрами з'єднані.

Існує два найпоширеніших способи представлення графів. Для прикладу розглянемо граф, зображений на малюнку 10 і продемонструємо на ньому ці способи.



Мал. 10 а)

№ вершин	*1*	*2*	*3*	*4*	*5*	
1	0	1	0	1	0	1⇒2⇒4
2	1	0	1	1	1	2⇒1⇒5⇒3⇒4
3	0	1	0	1	1	3⇒2⇒4⇒5
4	1	1	1	0	0	4⇒1⇒3⇒2
5	0	1	1	0	0	5⇒3⇒2

Мал. 10 б)

Мал. 10 в)

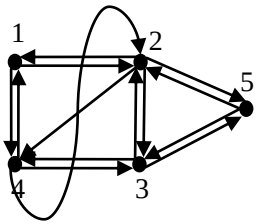
Перший спосіб передбачає задання ***матриці суміжності***, яка для графа з n вершинами представляється двовимірним масивом $n*n$.

Якщо матриця суміжності представляє неорієнтований незважений граф (мал.10а)), то за наявності суміжності вершин i та j відповідні елементи матриці дорівнюють 1, тобто $a[i,j]=a[j,i]=1$, а за відсутності ребра між вершинами i та j — 0, тобто $a[i,j]=a[j,i]=0$ (мал. 10 б)).

Які висновки можна зробити, проаналізувавши цю матрицю суміжності, зображену на мал.10б)? По-перше, для неорієнтованого графа вона є симетричною. По-друге, значення діагональних елементів дорівнюють 0, оскільки у нашому графі відсутні петлі.

Якою ж буде матриця суміжності для інших графів? Якщо матриця суміжності описує незважений орграф, то можлива

відсутність симетричності (мал.12а),б)). У випадку, коли в орграфі всі суміжні вершини мають двоорієнтовані ребра (мал.11а)), то матриця суміжності такого графу (мал.11б)) збігається з матрицею суміжності для графу, зображеного на малюнку 10а).



Мал. 11 а)

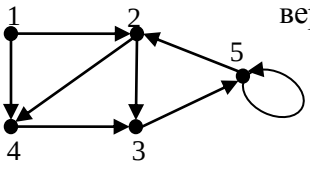
№ вершин	*1*	*2*	*3*	*4*	*5*
1	0	1	0	1	0
2	1	0	1	1	1
3	0	1	0	1	1
4	1	1	1	0	0
5	0	1	1	0	0

Мал. 11 б)

1⇒2⇒4
2⇒1⇒5⇒3⇒4
3⇒2⇒4⇒5
4⇒1⇒3⇒2
5⇒3⇒2

Мал. 11 в)

Особливістю незваженого орграфу, у якому присутні ребра-петлі, є те, що для відповідних вершин діагональні елементи матриці суміжності не дорівнюють 0 (мал.12 а),б)). Аналогічною буде ситуація і для неорієнтованих графів.



Мал. 12 а)

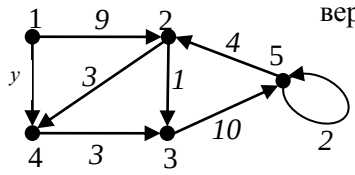
№ вершин	*1*	*2*	*3*	*4*	*5*
1	0	1	0	1	0
2	0	0	1	1	0
3	0	0	0	0	1
4	0	0	1	0	0
5	0	1	0	0	1

Мал. 12 б)

1⇒2⇒4
2⇒3⇒4
3⇒5
4⇒3
5⇒2⇒5

Мал. 12 в)

Розглянемо тепер випадок зваженого графу. Елементи матриці суміжності *a*[*i*,*j*], що відповідає такому графу, будуть збігатися зі значеннями «ваги» ребер між вершинами *i* та *j* (мал.13а),б)).



Мал. 13 а)

№ вершин	1	2	3	4	5
1	0	9	0	5	0
2	0	0	1	3	0
3	0	0	0	0	10
4	0	0	3	0	0
5	0	4	0	0	2

Мал. 13 б)

$1 \Rightarrow 2_9 \Rightarrow 4_5$

$2 \Rightarrow 3_1 \Rightarrow 4_3$

$3 \Rightarrow 5_{10}$

$4 \Rightarrow 3_3$

$5 \Rightarrow 2_4 \Rightarrow 5_2$

Мал. 13 в)

Ще одним способом представлення графа є ***списки суміжних вершин***. У цьому разі для кожної вершини створюється список суміжних вершин, який у довільному порядку містить всі суміжні з нею вершини. Графам, зображеним на мал. 10 а), 11 а), 12 а), 13 а) відповідають списки суміжних вершин, зображені на мал. 10 в), 11 в), 12 в), 13 в).

Який спосіб представлення заданого графу краще за все використовувати? На це питання можна відповісти тільки стосовно кожного конкретного алгоритму. Звичайно, що обробляти елементи матриці суміжності зрозуміліше і простіше. В ній одразу ж можна дізнатися, чи є дві задані вершини суміжними, яку вагу має ребро, з'єднані дві задані вершини чи ні тощо.

Однак, наприклад, для неорієнтованих графів, що не мають петель, матриця суміжності містить зайву інформацію, оскільки вона є симетричною. Якщо ж у якості опису графа обрати список суміжних вершин, то значно покращується компактність преставлення інформації, проте при цьому ускладнюється її обробка. Саме тому на самому початку було наголошено, що спосіб представлення графу у кожному окремому випадку обирається автором алгоритму згідно умови задачі та сформульованої мети її розв'язання.

Зміст

Навчальне видання

Караванова Тетяна Петрівна

ТЕОРІЯ АЛГОРИТМІВ
ЧАСТИНА 2. ОБЧИСЛЮВАЛЬНІ АЛГОРИТМИ

Навчальний посібник

Відповідальний за випуск – ***І. М. Черевко***

Літературний редактор – ***О. В. Лупул***
Технічний редактор – ***Т. П. Караванова***

Підписано до друку 14.09.2022. Формат 60x84/16.
Папір офсетний. Друк різографічний. Умов.-друк. арк.15,8.
Обл.-вид. арк. 16,9. Тираж 00. Зам. Н-000п.
Видавництво та друкарня Чернівецького національного університету.
58012, Чернівці, вул. Коцюбинського, 2.
e-mail: ruta@chnu.edu.ua

Свідоцтво суб'єкта видавничої справи ДК № 891 від 08.04.2002.