Міністерство освіти і науки України

Запорізький національний університет

О. Г. Спиця, А. Г. Кривохата

**ДИСКРЕТНІ СТРУКТУРИ**

**Навчальний посібник**

**для здобувачів ступеня вищої освіти бакалавра  
спеціальності 121 Інженерія програмного забезпечення   
освітньої програми Програмна інженерія**

Затверджено

вченою радою ЗНУ

Протокол № 6 від 21.12.2021

Запоріжжя

2022

УДК 512.64+514.12](075.8)

С727

Спиця О. Г., Кривохата А. Г. Дискретні структури : навчальний посібник для здобувачів ступеня вищої освіти бакалавра спеціальності 121 Інженерія програмного забезпечення освітньої програми Програмна інженерія. Запоріжжя : Запорізький національний університет, 2022. 107 с.

Навчальний посібник містить теоретичні відомості про основні поняття теорії множин, геометричну інтерпретацію множин, алгебру множин, відношення та операції над ними, властивості відношень, відношення порядку та еквівалентності, числення висловлень, метод математичної індукції, елементи комбінаторики, функції алгебри логіки, графи. У посібнику наведено приклади розв’язання основних типів задач.

Для здобувачів ступеня вищої освіти бакалавра спеціальності 121 Інженерія програмного забезпечення освітньої програми Програмна інженерія.

Рецензент

*С. М. Гребенюк,* доктор технічних наук, професор, завідувач кафедри фундаментальної та прикладної математики

Відповідальний за випуск

*І. В. Зіновєєв,* кандидат фізико-математичних наук, доцент, завідувач кафедри загальної математики

**ЗМІСТ**

|  |  |
| --- | --- |
| Вступ………………………………………………………………….…………… | 4 |
| 1 Теорія множин…………………………………………………………….….. | 6 |
| 1.1 Основні поняття теорії множин……………………………………... | 6 |
| 1.2 Потужність множин………………………………………………….. | 7 |
| 1.3 Способи задання множин……………………………………………. | 8 |
| 1.4 Алгебра підмножин………………………………………………….. | 9 |
| 1.5 Комп’ютерне подання множин та дій над ними…………………… | 12 |
| 1.6 Добуток Декарта……………………………………………………... | 14 |
| 2 Відношення…………………………………………....................................... | 16 |
| 2.1 Поняття відношення…………………………………………………. | 16 |
| 2.2 Бінарні відношення…………………………………………………... | 17 |
| 2.3 Способи задання відношень…………………………………………. | 18 |
| 2.4 Композиція відношень………………………………………………. | 21 |
| 2.5 Обернене відношення………………………………………………... | 23 |
| 2.6 Типи відношень……………………………………………………… | 25 |
| 2.7 Функціональні відношення……………………………………..…… | 27 |
| 2.8 Відношення порядку………………………………………...………. | 33 |
| 2.9 Відношення еквівалентності………………………………………… | 36 |
| 3 Математична логіка………………………………………………………….. | 38 |
| 3.1 Алгебра висловлень………………………………………………….. | 38 |
| 3.2 Основні закони алгебри висловлень………………………………… | 44 |
| 3.3 Логічний наслідок……………………………………………………. | 46 |
| 3.4 Нормальні форми логіки висловлювань……………………………. | 47 |
| 3.5 Математична індукція……………………………………………….. | 49 |
| 4 Комбінаторика……………………………………………………………….. | 52 |
| 4.1 Основні правила комбінаторного аналізу. Розміщення  та сполучення………………………………………………………... | 52 |
| 4.2 Обчислення кількості розміщень і сполучень……………………… | 53 |
| 4.3 Перестановки………………………………………………………… | 55 |
| 4.4 Біном Ньютона……………………………………………………….. | 55 |
| 4.5 Принцип включення-виключення…………………………………... | 59 |
| 5 Функції алгебри логіки. Булева алгебра ……………………………..…….. | 62 |
| 5.1 Способи задання булевих функцій………………………………..… | 62 |
| 5.2 Елементарні функції алгебри логіки……………………………..…. | 64 |
| 5.3 Основні властивості функцій алгебри логіки…………………….... | 67 |
| 5.4 Булева алгебра та її основні закони………………………….……… | 68 |
| 5.5 Нормальні форми булевих функцій………………………………… | 71 |
| 5.6 Мінімізація булевих функцій………………………………………... | 75 |
| 6 Теорія графів………………………………………………………………..... | 79 |
| 6.1 Графи та відношення………………………………………………… | 79 |
| 6.2 Елементи графів……………………………………………………… | 87 |
| 6.3 Цикломатика графів. Дерева………………………………………… | 98 |
| Використана література…….…………………………………………………..…. | 106 |

**ВСТУП**

Курс «Дискретні структури» належить до циклу професійної підготовки спеціальності.

*Метою* курсу «Дискретні структури» є формування теоретичних знань та практичних навичок для застосування об’єктів дискретних систем або структур у моделюванні, програмуванні та інформаційних технологіях. Основними *завданнями* вивчення навчальної дисципліни «Дискретні структури» є:

* вивчення основних принципів теорії множин;
* ознайомлення з основними поняттями теорії булевої алгебри;
* розгляд основних положень теорії графів;
* оволодіння основами комбінаторики;
* оволодіння методами доведення тверджень та тотожностей;
* розгляд основних методів побудови математичної моделі існуючого процесу;
* оволодіння методами застосування дискретної математики для розв’язування прикладних задач.

*Міждисциплінарні зв’язки*. Дисципліна «Дискретні структури» дає можливість закласти основу для подальшого вивчення курсів «Алгоритми та структури даних», «Емпіричні методи програмної інженерії», «Об’єктно-орієнтоване програмування».

У результаті вивчення навчальної дисципліни студент повинен

**знати:**

* основні поняття, факти та теореми лінійної алгебри;
* основні поняття, факти та теореми аналітичної геометрії;
* сфери застосування матриць та визначників;
* сфери застосування векторів, їх добутків, кривих та поверхонь І та ІІ порядків.

**Вміти:**

* застосовувати основні поняття, твердження та теореми до розв’язку задач;
* наводити приклади, які демонструють суттєвість теоретичних понять чи фактів, або спростовують хибні ствердження;
* застосовувати елементи алгебри до розв’язання задач геометрії, та використовувати матеріал попередніх тем при вивченні наступних;
* розв’язувати типові задачі кожної з вивчених тем.

*Змістове наповнення курсу, що викладається на лекційних і практичних заняттях та засвоюється студентом під час самостійної роботи*, забезпечує набуття компетентностей:

* здатність розв’язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій;
* здатність до абстрактного мислення, аналізу та синтезу;
* здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв’язання завдань інженерії програмного забезпечення;
* здатність до алгоритмічного та логічного мислення

*Програмні результати навчання.* У разі успішного завершення курсу студент зможе:

* аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки;
* знати і застосовувати відповідні математичні поняття, методи доменного, системного і об’єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.

Згідно з робочою програмою дисципліни «Дискретні структури» посібник охоплює теоретичний матеріал до наступних тем:

* Основні поняття теорії множин. Геометрична інтерпретація множин. Алгебра множин.
* [Відношення. Операції над відношеннями](https://moodle.znu.edu.ua/course/view.php?id=24). [Властивості відношень. Відношення порядку і еквівалентності](https://moodle.znu.edu.ua/mod/assign/view.php?id=18240).
* Числення висловлень. Математична індукція.
* Елементи комбінаторики.
* Функції алгебри логіки. Булева алгебра.
* Графи. Транспортні мережі. Мережеві графи.

В кожні темі окрім теоретичного матеріалу представлена велика кількість прикладів розв’язання задач з детальними коментарями.

1. ТЕОРІЯ МНОЖИН

У розділі 1 розглядаються основні поняття і означення сучасної дискретної математики: множина, елементи множини, операції над множинами та їх властивості, які становлять базовий словник для дискретної математики й є потрібні для розуміння всього подальшого матеріалу посібника.

* 1. Основні поняття теорії множин

Поняття множини є одне з фундаментальних невизначених понять сучасної математики і береться за основне, тобто за таке, що не зводиться до інших понять. Під множиною розуміють деяку сукупність різних поміж собою об’єктів, об’єднаних за певною ознакою, причому таких, що для кожного можна встановити, належить цей об’єкт даній сукупності чи ні. При цьому ніяких припущень що до природи об’єктів не робиться.

Об’єкти, з яких складено множину, називають її *елементами*.

Множини позначаються великими літерами латинської абетки: A, B, C,..., а об’єкти або елементи, які становлять множину, позначаються малими латинськими літерами: a, b, с, ..., або малими латинськими літерами з індексами.

**Приклад 1.1**

1) Множина N чисел натурального ряду 1, 2, 3, ...;

2) множина R дійсних чисел;

3) множина літер української абетки.

Твердження, що множина А складається з елементів умовно записується як . Порядок елементів множини не має значення.

Належність елемента до множини позначають символом ∈, тобто , , …, , або скорочено: . Якщо b не є елементом *А*, то пишуть: .

Множина, всі елементи якої є числами, називається *числовою*. Надалі ми будемо, насамперед, розглядати саме такі множини. Множина, елементами якої є інші множини, називається *класом* або *сімейством*.

Множина може мати скінчену кількість елементів, тобто бути *скінченою*, або бути *нескінченною*.

**Приклад 1.2**

1) множина непарних чисел (нескінчена);

2) множина всіх розв’язків рівняння (нескінчена);

3) множина студентів певного вищого навчального закладу (скінчена);

4) множина точок кола (нескінчена).

Існує множина, яка не містить жодного елемента. Така множина називається *порожньою* і позначається символом ∅.

**Приклад 1.3** Множина дійсних коренів рівняння є порожньою.

Не завжди відомо, чи існують елементи, які визначають деяку множину.

**Приклад 1.4** Множина виграшних квитків лотереї може стати визначеною тільки після тиражу.

Множина як об’єкт може бути елементом іншої множини.

**Приклад 1.5** У множині книг на полиці самі книги можуть розглядатися як множини сторінок.

Передбачається, що границі множини повинні бути чітко визначені. Саме задання множини явно або неявно обмежує сукупність об’єктів, які належать цієї множині. У будь-якій конкретній задачі доводиться мати справу тільки з фіксованою для цієї задачі, множиною.

Універсальною множиною (універсумом) називається множина, що містить всі елементи з деякою заданою властивістю. Позначається така множина через *U*.

Поняття «універсальної множини» залежить від задачі, яку розглядають. Прикладом універсальної множини може бути множина дійсних чисел, множина людей на планеті Земля тощо.

* 1. Потужність множин

Кардинальним числом (позначається або ) називається деякий об’єкт для позначення потужності будь-якої множини із сукупності множин.

Потужністю скінченної множини A називається кількість її елементів.

Кардинальне число є узагальненням поняття числа елементів скінченної множини на випадок нескінченної множини.

Стверджують, що множини та мають однакову потужність, якщо кожному елементу множини *А* можна поставити у відповідність єдиний елемент множини B, тобто можна встановити взаємнооднозначну відповідність поміж їхніми елементами . У цьому разі множини A та B називають рівнопотужними та позначають .

Кожні непорожні скінчені множини з n елементів є рівнопотужними множинами певного відрізку натурального ряду , де . У цьому разі їхня потужність дорівнює .

Потужність порожньої множини вважають рівною 0, тобто .

**Приклад 1.6**

1) Якщо , то .

Кожна множина, яка рівнопотужна множині натуральних чисел, називається зліченною. Її потужність позначається літерою (алеф нуль, алеф – перша літера єврейської абетки).

**Приклад 1.7** Множина непарних чисел P є рівнопотужна множині всіх натуральних чисел N, тому множина P є зліченою множиною і її потужність .

Потужність множини дійсних чисел проміжку називається *потужністю* континууму. Позначається через або .

**Приклад 1.8** Множини дійсних чисел проміжків та є рівнопотужні, оскільки по між ними можна встановити взаємнооднозначну відповідність.

* 1. Способи задання множин

При заданні множин слід визначити, які елементи до неї належать.

1. Множину можна задавати явним переліченням всіх її елементів: . Це є спосіб задання множини списком, який підходить тільки для задання множин з невеликою кількістю елементів. Позначення списку – у фігурних дужках.

**Приклад 1.9** Множина всіх студентів, присутніх в аудиторії (Петров, Сидоров, ...).

1. Узагальненням першого способу є задання елементів множини за допомогою певних елементів уже відомої множини, тобто так званою ***процедурою, що породжує***.

**Приклад 1.10** За відомою множиною цілих чисел визначимо множину степенів числа 3: .

1. Множину можна задавати ***за допомогою деякої характеристичної властивості***, якою володіє кожен з елементів множини, що розглядається, і не володіє кожен інший елемент, що не входить до цієї множини.

Характеристичну властивість запишемо у вигляді або , причому , якщо властивість є істинною. Такий спосіб ще називають ***предикатним***.

**Приклад 1.11**

а) Нехай задано множину натуральних чисел . Розглянемо сукупність елементів з множини *N*, які діляться на 3 (характеристична властивість). Дістанемо множину чисел, кратних до 3: . Задамо цю множину за допомогою характеристичної властивості ;

б) .

**Зауваження**. Переліченням елементів можна задати лише скінчені множини, а за допомогою характеристичної властивості можна задавати як скінчені так і нескінчені множини.

1. ***Рекурсивний спосіб***.

В якості прикладу розглянемо множину значень рекурсивної функції: , де . Нехай , й кожне наступне число залежить від двох попередніх наступним чином: , .

1. ***Вербальний*** (***словесний***).
   1. Алгебра підмножин

Тільки одного поняття множини ще недостатньо для вивчання існуючих дискретних структур. Необхідно ще ввести поняття частини множини і правил створювання нових множин із уже існуючих.

Множина *А*, всі елементи якої належать і до множини *В*, називається підмножиною (*частиною*) множини *В*.

Таке співвідношення поміж множинами називається включенням і позначається символом «⊆», тобто (А включене до *В* або В містить *А*). Вочевидь, що , якщо з належності елемента *х* до множини A випливає належність цього елемента і до множини *В*, тобто з .

Якщо множина A не міститься в множині В, використовують позначання .

Дві множини *А* та *В* називаються рівними (позначається ), якщо та . Це є визначення рівності двох множин за допомогою операції включення.

У літературі також зустрічається позначення . У цьому випадку під слід розуміти *строге включення* або *строга підмножина*, яке не припускає рівності, тобто якщо і . Якщо й та , то A називають власною підмножиною множини B. Нестроге включення допускає рівність (тоді A називається невласною підмножиною множини B).

Ми будемо використовувати позначання для нестрогого включення, яке допускає рівність .

**Приклад 1.12** Множина невід’ємних дійсних чисел , яка має спеціальне позначення , міститься у множині дійсних чисел , тобто .

Вважають, що порожня множина є невласною підмножиною кожної непорожньої множини *А*, тобто . Враховуючі, що А теж входить до *А*, то кожна непорожня множина А має принаймні дві різні підмножини та *А*.

**Зауваження 1** Нехай U – деяка фіксована множина. Розглянемо тільки такі множини А, В, С, ..., які є підмножинами множини *U*. У цьому випадку множина U буде універсальною множиною для всіх множин А, В, С, ... .

**Зауваження 2** Зі співвідношень й випливає, що , тобто відношення включення транзитивне (рис. 1.1).

Для графічного зображення множини використовують спеціальні конструкції – діаграми Ейлера-Венна, які зображують сукупність елементів, що утворюють множину, овалами, а універсум – прямокутником.

Відношення включення графічно зображено на рис. 1.1.

Скінчені власні підмножини певної множини можуть утворювати різноманітні сполучення з одного, двох, трьох тощо елементів цієї множини.

Множиною всіх підмножин (булеаном) певної основної множини Е називають множину, елементами якої є всі підмножини множини Е. Позначається булеан через . Він включає до свого складу також елементи та множину *Е*.

*U*

*С*

*В*

*А*

Рисунок 1.1 – Відношення включення

**Приклад 1.13** Якщо , то

.

**Зауваження 1** Порядок елементів у множині є несуттєвий.

**Зауваження 2** Якщо множина Е містить n елементів, то множина містить елементів, звідси й позначання множини як .

**Зауваження 3** Відношення належності ∈ та включення ⊆ – різні поняття. Наприклад, множина А може бути власною підмножиною множини А (), але вона не може бути власним елементом цієї множини ().

**Приклад 1.14** Якщо , то , а 2 та .

Зазвичай розглядають п’ять основних операцій над множинами: доповнення, об’єднання, перетин, різницю та симетричну різницю. Подамо їхні означення, припускаючи, що задано певний універсум U.

Елементи множини *U*, які не входять до *А*, утворюють доповнену множину до А (позначаються ). Або інакше, *доповнення* множини А до множини U, це множина .

За допомогою діаграми Ейлера-Венна доповнену множину можна зобразити геометрично (рис. 1.2), де A – затемнена частина.

Об’єднанням двох множин – А та В (позначається або ) – називається множина *С*, яка складається з усіх тих елементів, які належать хоча б до однієї з цих множин: .

**Зауваження**. Однакові елементи враховуються один раз.

Геометричну інтерпретацію об’єднання двох множин А та В подано на рис. 1.3, де – затемнена частина.

Підкреслимо, що до множині належать також і ті елементи, які водночас належать множинам А та В.

Перетином двох множин – А та В (позначається або ) – називається множина С, яка складається з усіх тих елементів, які належать множині А і множині В (водночас!): .

Геометричну інтерпретацію перетину подано на рис. 1.4, де – затемнена частина.

Різницею двох множин – А та В (позначається ) – називається множина .

Геометричну інтерпретацію різниці подано на рис. 1.5, де – затемнена частина.

|  |  |
| --- | --- |
| Симетричною різницею двох множин – А та *В* (позначається , або ) – називається множина  .  Геометричну інтерпретацію симетричної різниці подано на рис. 1.6, де – затемнена частина. |  |

**Приклад 1.15** Нехай , , тоді:

; ;

; .

Якщо визначити універсум , то ; .

**Зауваження**. Для скінченного числа множин в аналогічний спосіб визначаються операції об’єднання та перетину

, .

Властивості операцій над множинами

Нехай задано множини A, B, C та U (U – універсум). Тоді для операцій ∪, ∩, \, ¬ виконуються такі властивості:

1. Комутативність:

, ;

1. Асоціативність:

, ;

1. Дистрибутивність:

, ;

1. , ,

, ,

;

1. , ;
2. Ідемпотентність:

, ;

1. Доповнення:

, ;

1. , ;
2. , ;
3. Закони поглинання:

, ;

1. Закони де Моргана:

, ;

1. Подвійного доповнення (інволюція): ;
2. Вираз для різниці: .

Пари символів ∪ та ∩ у формулах 1 – 10 називають *двоїстими* між собою. Їх можна змінювати місцями, замінюючи при цьому U на ∅ й навпаки.

У справедливості властивостей 1 – 12 можна переконатися чи то геометрично, чи формальними міркуваннями щодо кожної рівності.

**Приклад 1.16** Спростити вираз .

*Розв’язання*.

.

* 1. Комп’ютерне подання множин та дій над ними

У комп’ютері можна подавати множини різними способами. Один зі способів – зберігати невпорядковані елементи множини. Проте в такому разі операції з множинами займатимуть багато часу через те, що потрібно щоразу переглядати елементи. Тому розглянемо інші способи.

Одним із найпоширеніших і найпростіших способів – *подання множин за допомогою бітових рядків*. Упорядкуємо довільним способом елементи універсальної множини. Нехай універсальна множина містить n елементів: . Тоді для зображення підмножини застосовують бітову шкалу , елементи якої будують за таким правилом:

.

Приклад 1.17 Якщо , , то множини і можна зобразити наступним чином:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  | 1 | 1 | 1 | 1 | 1 | 1 |
|  | 0 | 1 | 1 | 0 | 0 | 1 |

Тепер на комп’ютері легко виконати операції над множинами *А* та *В*. Операції над множинами, зображеними у вигляді бітових шкал, виконують через порозрядні логічні операції. У результаті отримаємо множину, яку також можна зобразити у вигляді бітової шкали.

*Доповнення* множини А до множини U () зображують кодом, який будують інвертуванням коду множини А.

Об’єднання двох множин А та В () зображують кодом, який будують порозрядним логічним додаванням кодів множин А та В.

Перетин двох множин А та В () зображують кодом, який будують порозрядним логічним множенням кодів множин А та В.

Різницю двох множин А та В () зображують кодом, який будують порозрядним логічним відніманням кодів множин А та В.

Симетричну різницю двох множин А та *В* () зображують кодом, який будують порозрядним логічним додаванням кодів множин та .

**Приклад 1.18** Для множин , та , описуваних у вигляді бітових шкал, результати виконання операцій над множинами можна зобразити у такий спосіб:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  | 1 | 1 | 1 | 1 | 1 | 1 |
|  | 0 | 1 | 1 | 0 | 0 | 1 |
|  | 1 | 0 | 1 | 0 | 1 | 0 |
|  | 1 | 0 | 0 | 1 | 1 | 0 |
|  | 1 | 1 | 1 | 0 | 1 | 1 |
|  | 0 | 0 | 1 | 0 | 0 | 0 |
|  | 0 | 1 | 0 | 0 | 0 | 1 |
|  | 1 | 0 | 0 | 0 | 1 | 0 |
|  | 1 | 1 | 0 | 0 | 1 | 1 |

Якщо універсальна множина має велику потужність, а її підмножини не дуже потужні, то подання за допомогою бітових рядків неефективне щодо витрат пам’яті. У такому разі доцільно використовувати інші структури даних – зазвичай зв’язані списки та хеш-таблиці [3]. У певних задачах потрібні спеціальні методи подання множин, які ґрунтуються на використанні дерев [3].

* 1. Добуток Декарта

*Кортеж* – це впорядкований набір елементів. Це не означення кортежу, бо не пояснено, що таке впорядкований набір. Уважатимемо поняття «кортеж» (вектор, рядок, ланцюжок), як і поняття множини, первісним, неозначуваним. Елементи, що утворюють кортеж, називають його *компонентами*.

На відміну від елементів множини, компоненти кортежу можуть повторюватись. Компоненти нумерують, кількість компонент називають *довжиною* (*розмірністю*) кортежу. Кортеж з n елементів будемо позначати як і будемо говорити, що він має довжину n, . Нескінченні кортежі не розглядатимемо.

Іноді дужки й навіть коми не пишуть, наприклад 011001. Кортежі довжиною 2 часто називають *парами*, довжиною 3 – *трійками*, довжиною n – n-ками («енками»).

Два кортежі *рівні*, якщо вони мають однакову довжину та відповідні їх компоненти рівні. Інакше кажучи, кортежі і рівні, якщо та , , ..., .

Нехай задано дві множини – A та B – певних елементів.

Множина впорядкованих пар елементів, з яких перший належить до А, а другий – до *В*, називається декартовим (прямим) добутком множин А та В і позначається як .

Всі елементи множини – кортежі довжини 2.

Введене поняття декартова добутку припускає узагальнення. *Декартовим добутком* множин називається множина наборів кортежів довжини n: .

Степенем множини A називають декартів добуток .

**Приклад 1.19** Точка М у прямокутній декартовій системі координат на площині задається впорядкованою парою дійсних чисел у такий спосіб: (, ). Тоді . Звідси й назва добутку – декартів.

**Приклад 1.20** Нехай ; . Тоді

,

.

Очевидно, що взагалі .

**Приклад 1.21** Якщо ; , то декартів добуток має вигляд

.

Визначимо , як пари елементів по одному з кожної множини А та В (пари елементів, що належать до декартова добутку, позначимо в таблиці точками, рис. 1.7).



Для скінченних множин потужність (кількість елементів) декартового добутку дорівнює добутку потужностей цих множин: .

**Приклад 1.22**

1) , де , – множина натуральних чисел.

2) , .

1. ВІДНОШЕННЯ

Поняття відношення є фундаментальним поняттям не тільки дискретної математики, але й в інших теоретичних та прикладних дисциплінах. Відношення визначається як будь-яка підмножина впорядкованих кортежів, побудованих з елементів абстрактних множин, і реалізує зв’язки між реальними об’єктами.

* 1. Поняття відношення

**Приклад 2.1** Приклади відношень

1. – зв’язок поміж елементом та множиною;
2. – зв’язок поміж множинами;
3. – нерівності;
4. = – рівність;
5. «бути братом»;
6. ділення без остачі.

Введемо поняття впорядкованої множини.

Множина називається впорядкованою, якщо кожному її елементу поставлено у відповідність число n (, n – номер цього елемента) та елементи множини розміщено в порядку зростання їхніх номерів.

За кількості елементів множину можна впорядкувати не в єдиний спосіб.

Відношення позначатимемо літерою R, тоді запис вказує на те, що поміж *х* та у (, ) існує зв’язок. В прикладі 5) поданому вище можна записати «*х* є брат у». Тут відношення R – «бути братом».

Відношення, яке визначене на одному об’єктові називається унарним, якщо ж його визначено поміж парами об’єктів, – називаються бінарним, поміж трьома об’єктами – тернарним і т. д.

Відношення повністю визначається парами , для яких воно виконується, тому кожне бінарне відношення можна розглядати як множину впорядкованих пар . При цьому порядок вибору елементів істотний. Перший елемент завжди вибирається з першої множини, другий – з другої.

*Рівність* впорядкованих пар визначається в такий спосіб: , якщо та .

**Приклад 2.2** Нехай , , а відношення R – «елемент *x* дільник елементу *у*», де , . Тоді відношення R визначається парами елементів множин А та В

,

тому *R* є підмножиною множини, що складається з усіх упорядкованих пар елементів по одному з кожної множини *А* та *В*.

**Зауваження**. Функція (іноді записують ) також є відношенням.

* 1. Бінарні відношення

Найпоширенішими з відношень є бінарні відношення.

Бінарним відношенням R на множинах A та B називається довільна підмножина множини декартова добутку . Якщо , то це записується як: .

Якщо , то і в цьому випадку стверджують, що бінарне відношення R задано на множині A.

Зображення відношення R () точками в таблиці називають графіком *відношення*; множину (), для яких існує таке (), що , називають областю визначення відношення *R*, а множину (), для яких існує таке *х*, що , – множиною значень.

**Зауваження**. Кожна підмножина R множини є бінарним відношенням.

**Приклад 2.3**

1. Позначимо в таблиці (рис. 2.1) точками елементи, які належать до підмножини декартова добутку множин A та *B* з прикладу 1.19 ():



Тоді R – бінарне відношення поміж множинами A та B.

1. Відношення нестрогого порядку () (рис. 2.2) є підмножиною декартова добутку , тобто всієї площини:

*О*

*y*

*x*

*y*=*x*



Рисунок 2.2 – Зображення відношення нестрогого порядку ()

1. Відношення виконується для пар (5, 3), (7, 1), (2, 2), але не виконується для пар (1, 7), (9, 11), (2, 5).
2. Якщо *X* − множина студентів ЗНУ, а *Y* − множина груп ЗНУ, то відношення множин *X* і *Y* − є множина .
3. Якщо *X* − множина товарів у магазині, а , то відношення множин *X* й *Y* − є множина .
   1. Способи задання відношень

Є багато різних способів задання відношень. Найбільш розповсюджені з них задання відношень у табличній формі, стрілками, перетином, переліком пар. Розглянемо кожний з цих способів.

1. Табличний спосіб задання відношень.

**Приклад 2.4** Нехай відношення R належить до декартова добутку , де множини та , і задане таблицею (рис. 2.3).



Табличний спосіб завжди можна розглядати як різновид матричного. Тому відношення R можна задати також матрицею суміжності, або відношення, рядки якої позначають елементами множини A, а стовпчики – елементами множини B і на перетині рядка зі стовпчиком стоїть 1 в разі , та 0 – у противному випадку:

.

Матриці відношення називають *булевими*, тому що їхніми елементами є лише числа 0 або 1.

Для розглянутого вище прикладу матриця відношення буде мати форму:

,

де компоненти матриці R: а – елементи множин А та В.

1. Відношення *R* можна також задавати у вигляді списку пар елементів декартова добутку *,* для яких дане відношення виконується:

.

1. Спосіб задання відношень стрілками.

Цей спосіб проілюструємо за допомогою відношення R з прикладу 2.4. При цьому використаємо два варіанти зображення бінарного відношення (рис. 2.4).



1. Завдання відношень перетином.

Нехай – кортеж довжини 2 (де ). Елемент a називається проекцією *елемента* с *на множину* А (або на першу вісь). Позначається як .

Нехай Е – підмножина декартова добутку множин А та В (). Множина елементів з *А*, які є проекцією елементів множини Е на А, називається проекцією множини Е *на множину* А. Позначається як .

**Приклад 2.5** Нехай ; , а відношення визначається переліком пар елементів:

.

Треба знайти: 1) ; 2) .

*Розв’язання*.

Накреслимо графік відношення R (рис. 2.5).

* 1. Розглянемо кортеж . Маємо

.

* 1. Відношення R задано на множинах A та B і визначається наступними кортежами: (), тому

.



Впроваджене поняття проекції кортежу довжини 2 можна узагальнити на кортежі довжини *n*.

*Проекцією кортежу* *на* i *-ту вісь* називають його і-ту компоненту: .

*Проекцією кортежу* *на осі* з номерами називають вектор довжини k з компонентами: .

*Проекцією множини* векторів на i -ту вісь називають множину проекцій усіх векторів з V на i -ту вісь: .

*Проекцією множини* векторів на осі з номерами називають множину проекцій усіх векторів з V на осі з номерами :

.

Перетином *множини* (відношення) R називається множина елементів , для яких .

**Приклад 2.6** Перетином множини R з прикладу 2.5 буде множина .

**Зауваження**. Проекція відокремлює елементи у множині *A*, а перетин – елементи у множині В.

Нехай задано відношення . Позначимо через перетин відношення R, тобто множину таких , що . Отже,

.

Множина перетинів відношення R () по всім називається фактор-множиною множини B за відношенням R (позначається через )

.

Фактор-множина повністю визначає відношення R.

**Приклад 2.7** Розглянемо відношення R з прикладу 2.5. Перетином , відношення R будуть відповідно множини

, , ,

, .

Під кожним елементом запишемо його перетин:

.

Другий рядок буде фактор-множиною множини *B* за відношенням *R*.

Нехай тепер , а *X* деяка підмножина множини А (). Позначимо об’єднання всіх перетинів за всіма через , тобто перетином множини R по множині X є множина

.

Вочевидь, що .

**Приклад 2.8** Нехай задано три множини – ; ; – й відомо, що ; . Тоді

.

* 1. Композиція відношень

Нехай задано три множини *A*, *B*, C й два відношення R та S поміж ними: , .

Композицією двох відношень R та S називається відношення SR (іноді позначають як ) яке задано на декартовому добутку та визначене як таке, що перетин SR по всіх збігається з перетином S по підмножині (), тобто

, (2.1)

або

.

Операцію композиції бінарних відношень іноді ще називають добутком *відношень*.

**Зауваження**. При визначенні композиції відношень використано символ , який називається квантором існування і читається «існує, знайдеться хоча б один». Окрім квантора існування ще є двоїстий до нього квантор , який називається квантором загальності, який читається «для будь-якого, для кожного, для всіх». Застосування кванторів спрощує формальні записи.

**Приклад 2.9** Розглянемо відношення R, яке визначене в прикладі 2.5, і відношення S, яке задане таблицею на рис. 2.6.



Тоді відношення SR визначається таблицею на рис. 2.7.



Відношення SR можна ще знайти інакше, якщо записати відношення S та R у вигляді підмножин відповідно декартових добутків та :

;

. (2.2)

Тоді

.

У правильності відповіді переконаємося за допомогою задання відношення стрілками (рис. 2.8).



Композицію двох відношень S та R можна знайти ще й у такий спосіб:

.

Отже, .

Перевіримо, чи виконується визначення (2.1), наприклад, для перетину . У лівій частині формули дістанемо .

Оскільки , то у правій частині формули (2.1) при , матимемо: . Ліва й права частини збігаються.

**Зауваження**. n – степенем відношення R на множині A називається його n -разова композиція з самим собою, тобто .

* 1. Обернене відношення

Визначимо ще одну додаткову унарну операцію над відношеннями, яка не має аналогів у загальному випадку серед теоретико-множинних операцій, що розглядалися раніше. Це операція оберненого відношення.

Оберненим відношенням щодо певного відношення R () називається таке відношення , яке задається на декартовому добутку і утворюється парами для яких .

З визначення оберненого відношення випливає, що має місце тоді й лише тоді, коли існує відношення .

**Приклад 2.10** Розглянемо відношення S, яке визначене в прикладі 2.9 та має стрілочне зображення (рис. 2.9).



Оберненим щодо відношення S буде відношення , стрілочне зображення якого має вигляд на рис. 2.10.



Відношення у вигляді підмножини запишеться як

.

З табличного подання відношення (рис. 2.11) бачимо, що елементи таблиці є симетричні до елементів таблиці S щодо прямої l.

Композиція відношень і обернене відношення мають властивості:

1. ;
2. Якщо , , то .

Рисунок 2.11

У теорії бінарних відношень важливу роль відіграють також відношення:

доповнення – , ;

тотожне (діагональ) – , ;

універсальне (повне) – , .

Якщо позначити через , та матриці відповідно відношень R, та то

, ,

де – матриця універсальної множини, всі елементи якої дорівнюють 1, чи то, інакше , де .

* 1. Типи відношень

Нехай на множині A задано відношення *R*.

1. Бінарне відношення R на множині А називається рефлексивним, якщо всякий елемент цієї множини знаходиться у відношенні R з самим собою, тобто для всіх (інакше для всіх ).

Наприклад, відношення нестрогої нерівності на множинах .

1. Відношення R на множині А називається антирефлексивним, якщо з випливає , або інакше, якщо ні для якого не виконується (тобто , що є одне й те саме, що ).

Наприклад, відношення строгої рівності на множинах або відношення «бути начальником», «бути братом», «бути молодшим» на множині людей.

Якщо умова рефлексивності виконується не для всіх елементів множини *А*, то говорять, що відношення R є *нерефлексивне*.

Наприклад, відношення, яке не є ні рефлексивним, ні антирефлексивним є відношення «бути симетричним відносно осі » не є ні рефлексивним, ні антирефлексивним (точка є симетричною самій собі, якщо вона лежить на осі , і не є симетричною самій собі в протилежному випадку).

1. Відношення R на множині А називається симетричним, якщо для кожної пари елементів а та b, які належать до А, з того, що , випливає (тобто для з ).

Наприклад, відношення рівності на множинах ; «бути симетричним відносно осі » та відношення «бути братом» на множині людей.

1. Бінарне відношення R на множині А називається антисиметричним, якщо для всіх а та b, які належать до *А*, з належності та до відношення R випливає , або інакше, ні для яких елементів а та b, що відрізняються один від одного (), відношення і не існують одночасно (тобто якщо та ).

Наприклад, відношення нестрогої нерівності на множинах : и .

Властивості симетричності і антисиметричності не є взаємно виключними.

1. Відношення називається *асиметричним*, якщо для будь-яких елементів або або , або інакше, якщо для будь-якої пари пара ().

Наприклад, відношення строгого включення в множині всіх підмножин деякого універсуму або відношення «бути батьком» в множині людей.

Очевидно, якщо відношення асиметричне, то воно й антирефлексивне.

1. Бінарне відношення R на множині А називається транзитивним, якщо для будь-яких трьох елементів а, b та *с*, які належать до множини *А*, з того, що та , випливає, що (тобто з того, що та ).

Наприклад, відношення =, , «жити в одному місті» або відношення «мати непорожній перетин» на системі множин не є транзитивним.

1. Бінарне відношення R на множині А називається повним, якщо для всіх елементів а та b, які належать до *А*, або , або , або (тобто, або , або , або ).

**Приклад 2.11**

* 1. Відношення, яке позначене знаком «=» – рефлексивне;
  2. відношення «бути сином» – антирефлексивне;
  3. відношення «жити в одному місті» – симетричне;
  4. відношення «бути начальником» – антисиметричне;
  5. відношення «бути братом» – транзитивне.

**Зауваження 1** При задаванні відношення R () матрицею:

* відношення є рефлексивне, якщо всі елементи головної діагоналі матриці дорівнюють 1 (тобто );
* відношення є антирефлексивне, якщо немає жодної одиниці на головній діагоналі (тобто );
* відношення є симетричне, якщо матриця є симетрична щодо головної діагоналі (тобто );
* відношення є антисиметричне, якщо немає жодної пари одиниць симетричної головної діагоналі (окрім одиниць на самій діагоналі) (тобто );
* для матриці асиметричного відношення характерно, що всі елементи її головної діагоналі – нулі й немає жодної пари одиниць на місцях, розташованих симетрично відносно головної діагоналі.

**Зауваження 2** Транзитивність бінарного відношення R на множині А перевіряється простим перебиранням всіх елементів множини А (на А повинно виконуватися включення ).

**Зауваження 3** Відношення є повне, якщо .

**Приклад 2.12** Нехай . До якого типу належить відношення

?



*Розв’язання*. Зобразимо відношення R за допомогою таблиці (рис. 2.11).

* Відношення R – рефлексивне, оскільки для кожного , маємо ();
* відношення R – симетричне, оскільки для всіх пар () маємо

|  |  |  |  |
| --- | --- | --- | --- |
| Випадок |  |  | ? |
| 1 | (1, 3) | (3, 1) | так |
| 2 | (1, 5) | (5, 1) | так |
| 3 | (3, 5) | (5, 3) | так |

* відношення R – транзитивне, оскільки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Випадок |  |  |  | ? |
| 1 | (1, 3) | (3, 1) | (1, 1) | так |
| 2 | (1, 3) | (3, 5) | (1, 5) | так |
| 3 | (3, 1) | (1, 3) | (3, 3) | так |
| 4 | (3, 1) | (1, 5) | (3, 5) | так |
| 5 | (5, 1) | (1, 3) | (5, 3) | так |
| 6 | (5, 1) | (1, 5) | (5, 5) | так |
| 7 | (5, 3) | (3, 1) | (5, 1) | так |
| 8 | (5, 3) | (3, 5) | (5, 5) | так |

* відношення R – не є антисиметричним, тому що, наприклад, з того, що й , не випливає 1=2.
  1. Функціональні **відношення**

Відношення R () називають функціональним, якщо для кожного перетин R по *х* містить не більше одного елемента (або один або жодного!, рис. 2.12).

У цьому випадку говорять, що відношення R діє з множини A у множину *B* і часто використовують позначення .

З точки зору теорії множин поняття числової функції є окремим випадком відношення, коли множини A та B є числові. Тому позначення функціональної залежності малими латинським буквами також застосовують в теорії множин і пишуть або , а відношення f називають *функцією*.



Рисунок 2.12 – Відношення, не функція

Функція f може бути задана не на всій множині *А*, а тільки на деякій її частині . В цьому випадку множину D називають областю визначення функції f, а підмножину , де називають областю значень функції f.

**Зауваження**. Image переводиться як зображення чи образ.

Елемент , де , називають образом *елемента* a, а сам елемент a – прообразом елемента b.

Якщо , то функція f називається всюди визначеною на А. У цьому разі .

**Приклад 2.13** Відношення , яке задано таблицею (рис. 2.13) є функціональним, але не всюди визначеним. Образом елемента є елемент , а прообразами елемента є елементи та .

**

**Зауваження**. Якщо відношення , обернене до функціонального відношення , є також функціональним, то відношення f буде взаємнооднозначним.

**Приклад 2.14** (функціонального й оберненого до нього відношення). Нехай та визначається таблицею 1 (рис. 2.14). Тоді відношення визначається таблицею 2 (рис. 2.14) і є функціональним, тому відношення f є взаємнооднозначним.

Рисунок 2.14 – Функціональне й обернене до нього відношення

Структура елементів для нас не є важливою, тому функції i в прикладі, який розглядається, зручно записувати як

, .

Якщо відношення водночас є функціональним та всюди визначеним на множині *А*, то воно називається відображенням множини А у множину B.

Наприклад, відношення , яке задано в прикладі 2.13, не є всюди визначеним, тому не є відображенням.

При стрілочному зображенні відображення f з кожної точки повинна виходити лише одна стрілка.

**Приклад 2.15** Довизначимо відношення прикладу 2.13, поклавши , тоді здобудемо функціональне відношення (рис. 2.15), яке вже є відображенням.

 або 

Рисунок 2.15 – Функціональне відношення , яке є відображенням

**Зауваження**. Нехай f є відображенням множини А на множину В. Перетин множини по є образом елемента x для функції f і позначається як . Елемент х називають аргументом, – значенням *функції*. Перетин множини В по є прообразом елемента у для функції f.

На рис. 2.16 та 2.17 графічно зображені образ елемента *х* та відображення f, яке діє з множини А у множину В.

Множина упорядкованих пар називається графіком відображення f.

Відображення f називається сюр’єктивним, або просто сюр’єкцією, якщо область значень f збігається з усією множиною В або , тобто якщо кожний елемент з множини В є образом хоча б одного елемента з множини А. У цьому разі f відображає А на В (рис. 2.17).

Відображення f називається ін’єктивним, або просто ін’єкцією, якщо відношення є функціональне (рис. 2.18), тобто різні елементи множини А переводяться в різні елементи множини В. У цьому разі кожний елемент з області значень f має єдиний прообраз, тобто з рівності випливає .

Відображення називається взаємнооднозначним, або бієктивним, або просто бієкцією, якщо воно є сюр’єктивне й ін’єктивне, інакше кажучи, якщо обернене відношення є відображенням (рис. 2.19).



Відображення називається оберненим відображенням до відображення .

Наступний рис. 2.20 ілюструє поняття сюр’єкції, ін’єкції та бієкції.



**Приклад 2.16** Нехай R – множина дійсних чисел, – множина дійсних додатних чисел, а функція .

1. Якщо , то функція задає відображення А у B (не сюр’єктивне, тому що від’ємні числа не є образами).
2. Якщо , то функція задає відображення А на В (сюр’єктивне).
3. Якщо , , то функція – ін’єктивне відображення, тому що воно є взаємнооднозначне: .

**Зауваження 1** Якщо функція є бієкцією, то функція також буде бієкцією і .

**Зауваження 2** Бієкція скінченної множини А на себе називається підстановкою. Якщо множина має n елементів, то можна розглядати множину n! всіх підстановок, пов’язаних з даною множиною А.

Елемент x називається нерухомою точкою відображення якщо .

**Зауваження**. Оскільки функція є окремим випадком відношення, це означає, що для функцій є також визначена композиція, яка в цьому разі називається суперпозицією функцій. Нехай f – функція, визначена на множині А зі значеннями в множині *В*, g – функція, визначена на множині В зі значеннями в множині C, тоді композиція є функція, яка діє з множини А в множину *С*. Таким чином суперпозиція функцій знову є функцією. З означення суперпозиції маємо .

Приклад 2**.17** Нехай задано функції та . Тоді, .

**Зауваження**. Для відображень g і f справедлива формула .

Нехай задано множину .

Тотожним відображенням називається відображення, яке кожному елементові ставить у відповідність цей же самий елемент (позначається символом ). Таким чином, .

Якщо f та – відображення, визначені на множині А зі значеннями в цій же самій множині *А*, то відображення f називається відображенням на себе (*бієкцією на себе*) і мають місце рівності:

; . (2.3)

**Приклад 2.18** Нехай відображення f задано таблицею на рис. 2.21. Тоді відображення визначається таблицею на рис. 2.22.

Функції f і запишемо у вигляді: , .

Зображення відображень f і стрілками складається з циклів (рис. 2.23).



Рисунок 2.23

Перевіримо виконання умови (2.3):

,

.

Звідси випливає, що відображення є тотожним.

Знайдемо композицію відображень f та :

.

Функція () називається функцією n аргументів.

Така функція відображає кортеж у елемент .

* 1. **Відношення порядку**

Відношення, котрі зустрічаються на практиці, можуть мати водночас кілька однакових комбінацій властивостей, яким можна надати спеціальну назву й для яких можна вивчати окремо наслідки з цих комбінацій, притаманні всім відношенням з такою комбінацією властивостей. Розгляд розпочнемо з відношення порядку, яке дозволяє порівнювати поміж собою елементи однієї множини.

Бінарне відношення R, яке визначено на множині *А*, називається *відношенням* порядку, якщо воно є антисиметричне й транзитивне.

Бінарне відношення R на А називається *відношенням* нестрогого порядку, якщо воно є рефлексивне, антисиметричне й транзитивне.

Бінарне відношення R на А називається *відношенням* строгого порядку, якщо воно є антирефлексивне, антисиметричне та транзитивне.

Множину, на якій визначено відношення порядку (строгого або нестрогого), називають *упорядкованою*.

Якщо відношення порядку є повне, то воно називається *відношенням* повного, або лінійного *порядку*, а якщо воно не має властивості повноти, то називається *відношенням* часткового *порядку*. У цьому разі множина А із заданим на ньому відношенням R називається частково впорядкованою множиною (позначається , або просто *А*).

Множина, на якій визначено відношення повного порядку (або, якщо для будь-яких двох елементів множини та виконується нерівність або ( або )), називається лінійно (абсолютно) впорядкованою. Лінійно впорядковану множину називають також ланцюгом.

Зазвичай нестрогий порядок позначають через «». У цьому разі маємо нестрого впорядковану множину . Відношення строгого порядку зазвичай, позначають знаком «». Відношення порядку у загальному випадку позначають знаком «».

**Приклад 2.19** Нехай A – множина дійсних чисел, а відношення R на А є . Тут R – відношення нестрогого повного порядку, тому – нестрого впорядкована множина (лінійно впорядкована).

**Приклад 2.20** Нехай , а – булеан множини С, тобто множина всіх підмножин множини С. Тоді

.

Ця множина містить елементів.

Відношення R на множині визначимо як , якщо , де (R – відношення включення множин, тобто елемент , якщо ). Наприклад, , тому що , а , оскільки .

Можна перевірити, що відношення R визначене у такий спосіб, є рефлексивне, антисиметричне та транзитивне, тому на множині воно визначає нестрогий порядок, тобто відношення R на булеані є відношенням нестрогого часткового порядку.

Два елементи *а* та *b* частково впорядкованої множини називають *порівнянними*, якщо чи . Якщо *а* та *b* – такі елементи, що ні , ні , то їх називають *непорівнянними*.

Нехай на скінченній множині *А* задано якесь відношення часткового порядку *R*, задане за допомогою стрілок . Відношення *R* можна задати за допомогою такої процедури.

Починають із . Оскільки відношення часткового порядку рефлексивне, то в кожній вершині є петля. Потрібно вилучити всі ці петлі. Потім вилучають усі дуги , які є в ньому внаслідок транзитивності. Наприклад, якщо пари та належать відношенню, то в вилучають дугу . Більше того, якщо пари також належить відношенню *R*, то в вилучають дугу . Після цього всі вершини розмішують па площині так, щоб початкова вершина кожної дуги була нижче, ніж кінцева вершина. Тепер усувають усі стрілки, бо дуги спрямовано вгору, до кінцевих вершин.

Усі ці кроки задано коректно, і в разі скінченної множини *А* кількість таких кроків скінченна. Отримаємо діаграму Гассе (Н. Hasse), яка містить усю інформацію, потрібну для визначення відношення часткового порядку.

**Приклад 2.21** Нехай . Відношення *R* задамо так: тоді й лише тоді, коли *а* ділить . Отже,

.

Легко переконатись, що це відношення рефлексивне, антисиметричне й транзитивне, тому являє собою відношення часткового порядку на множині *А*.

Зобразимо діаграму Гассе для заданого відношення часткового порядку.

Почнемо з графічного зображення у вигляді стрілок (рис. 2.24, а). Вилучимо всі петлі (рис. 2.24, б), а потім – усі дуги, зумовлені властивістю транзитивності; це дуги (1, 4), (1, 6), (1, 8), (1, 12), (2, 8), (2, 12), (3, 12). Орієнтуємо всі дуги в напрямку знизу вверх і усунемо стрілки. Отримаємо діаграму Гассе (рис. 2.24, в).

Нехай A – підмножина впорядкованої множини E, на якій визначено відношення порядку «». Якщо існує такий елемент , що для кожного , то m називається нижньою межею множини *A*. Аналогічно, якщо існує елемент , що для кожного , то M називається верхньою межею множини A.

Якщо m та M належать до множини A, то m та M відповідно називаються мінімумом та максимумом множини A й позначаються символами

або ; або .

Верхня та нижня межі для кожної множини існують не завжди й не завжди є єдині.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| а | б | в |

Рисунок 2.24 – Діаграма Гассе

Якщо існує найбільша нижня межа множини *А*, то вона називається інфімумом і позначається , а якщо існує найменша верхня межа множини *А*, то вона називається супремумом і позначається .

Максимальні та мінімальні елементи легко визначити на діаграмі Гассе: це відповідно «верхні» й «нижні» її елементи (для «верхніх» елементів немає висхідних ребер, а для «нижніх» – низхідних).

**Приклад 2.22** На множині задано відношення часткового порядку . Знайдемо максимальні й мінімальні елементи множини . Діаграму Гассе для цієї множини зображено на рис. 2.25. Із неї доходимо висновку, що максимальні елементи – 12, 20 і 25, а мінімальні – 2 та 5. Цей приклад свідчить, що частково впорядкована множина може мати більше одного максимального чи мінімального елемента.



* 1. **Відношення еквівалентності**

Бінарне відношення R на множині А називається відношенням еквівалентності, якщо воно є одночасно рефлексивне, симетричне й транзитивне (позначається символами або ).

**Приклад 2.23**

1) рівність чисел та множин є відношенням еквівалентності;

2) у прикладі 2.12 відношення R задовольняє всім трьом наведеним властивостям, тому воно є відношенням еквівалентності.

Наприклад, класифікація об’єктів деякої множини A на непересічні підмножини елементів , де , , якщо вони мають однакові властивості, визначає відношення еквівалентності. В цьому випадку елементи однієї підмножини володіють однаковою властивістю та є еквівалентні до елементів тієї ж самої підмножини й не є еквівалентні до елементів решти підмножин , до того ж серед підмножин немає порожніх. Здобуті підмножини називаються класами еквівалентності множини А.

**Приклад 2.24** Нехай A – множина студентів одного міста. Визначимо на множині A відношення R – «*x* та y навчаються в одному вищому навчальному закладі», де . Відношення R буде відношенням еквівалентності, якщо жоден студент міста не навчається в кількох вищих навчальних закладах. У цьому разі класи еквівалентності становитимуть студенти одного вищого навчального закладу.

**Приклад 2.25** Відношення R у прикладі 1.12 розбиває множину А на класи [1], [2], [3], [4], [5], [6], де

.

Перевіримо: , оскільки ;

, оскільки ;

, оскільки .

;

;

;

.

Аналіз здобутих результатів засвідчує, що різними є лише три класи:

; ; .

Символом позначають множину всіх класів еквівалентності множини *A* за відношенням еквівалентності *R* та називають *фактор-множиною* множини *A* за відношенням еквівалентності *R*. У розглянутому прикладі фактор-множиною буде множина класів

.

Кожний елемент класу еквівалентності породжує цей же самий клас еквівалентності, отже представляє цей клас.

Системою представників певного відношення еквівалентності називається підмножина, яка містить по одному елементові з кожного класу еквівалентності.

**Приклад 2.26** На множині цілих чисел Z визначимо відношення за допомогою формули .

Відношення R є рефлексивне, тому що внаслідок рівності для .

Відношення R є симетричне, оскільки з належності випливає, що , тобто і, отже, .

Відношення R є транзитивне, оскільки з належності та до R, випливає, що , , тобто , оскільки , де – ціле число.

Отже, відношення R є рефлексивне, симетричне й транзитивне, тому воно є відношенням еквівалентності.

Розглянемо класи чисел

.

Здобудемо три різні класи еквівалентності стосовно відношення R прикладу, який розглядаємо:

,

,

.

Множина всіх класів еквівалентності

є фактор-множиною множини Z за відношенням еквівалентності R.

Елемент *x* належить до класу , якщо , де . (Пишуть ).

У загальному випадку, якщо Z – множина цілих чисел, а p – деяке число (), то вважатимемо , якщо , де , або, що є одне й те саме, .

**Приклад 2.27** Відношення паралельності прямих q на площині є відношенням еквівалентності, тому що:

* + 1. – рефлексивне;
    2. – симетричне;
    3. – транзитивне.

Кожний клас еквівалентності в множині прямих на площині – це множина паралельних прямих, яка повністю визначається напрямком однієї прямої.

**Приклад 2.28** Вважатимемо, що точка площини є еквівалентна точці цієї ж площини, якщо . У цьому разі класами еквівалентності будуть множини точок на площині з рівними абсцисами (тобто всі прямі, які є паралельні до осі *Оу*). У цьому випадку фактор-множиною буде множина всіх прямих на площині, які є паралельними до осі Оу.

Прикладами відношення еквівалентності є також рівність векторів, логічних тверджень тощо.

1. **МАТЕМАТИЧНА ЛОГІКА**
   1. Алгебра висловлень

*Логіка* – це наука про закони мислення та його форми. Вона як мистецтво суджень бере свій початок з далекої давнини. В логіку було впроваджено математичну символіку, і сьогодні вона використовує мову й методи математики. Звідси й назва – математична логіка. Основи математичної логіки було закладено в середині XIX сторіччя ірландським математиком Дж. Булем.

Останніми десятиліттями логіка набула широкого застосовування в техніці під час дослідження та розроблення електронних схем, обчислювальних машин, дискретних автоматів. Вона використовується також і в інших науках: економіці, біології, психології тощо.

Основним поняттям у логіці є висловлення, під яким розуміють думку, яка подається за допомогою твердження. Тобто *висловлення* – це певне твердження, яке може бути або істинним або хибним.

**Приклад 3.1** Наведемо приклади речень.

* + - * 1. Сніг білий.
        2. Одеса – столиця України.
        3. 4 є парне число.
        4. 2+3=6.
        5. *х*–4=0.
        6. Котра година?
        7. Читай уважно!

Перших чотири речення – висловлювання, решта три – ні, бо п’яте речення набуває істинного чи хибного значення залежно від значення змінної *х*, шосте та сьоме речення – не розповідні.

Значення «істина» чи «хибність», яких набуває висловлювання, називають його *значенням істинності*. Значення «істина» позначають буквою *Т* (від англ. «truth» чи цифрою 1), а «хибність» – буквою *F* (від «false» чи цифрою 0). Для позначення висловлювань використовують малі латинські букви з індексами чи без них. Символи, використовувані для позначення висловлювань, називають *атомарними формулами* чи *атомами*.

**Приклад 3.2** Наведемо приклади висловлювань.

1. *р*: «Сніг білий».

2. *q*: «Одеса – столиця України».

3. *r*: «4 є парне число».

4. *s*: «2+3=6».

Тут символи *р*, *q*, *r*, *s* – атомарні формули. Висловлення «Сніг білий», «4 є парне число» є істинними, а висловлення «Одеса – столиця України», «2+3=6» – хибними.

Поміж висловленнями використовуються різні логічні зв’язки: «якщо ..., то ...», «... або ...», «... і ...» тощо. За їхньої допомоги будуються інші нові висловлення. Складне висловлювання утворюють із наявних висловлювань за допомогою логічних операцій.

*Логічною операцією* називається операція, в якій операндами є висловлення, а операторами – логічні зв’язки.

*Алгебра логіки* (*алгебра висловлень*) являє собою науку про сукупність висловлень, над якими визначені логічні операції.

Висловлення, які характеризуються значеннями 0 чи 1, позначатимемо літерами *х*, *y*, z тощо (такі змінні називатимемо *бульовими*).

За допомогою алгебри логіки можна, наприклад, описувати роботу релейно-контактних схем. Для конкретики обмежимося розгляданням двополюсних схем, у яких поміж полюсами можуть існувати релейні контакти, з’єднані послідовно чи паралельно. При цьому стан контакту – 1 (0) означає, що він замкнений (розімкнений), тобто сигнал 1 (0) переводить електронний елемент у відкритий (закритий) стан.

Розглянемо спочатку схеми з одним контактом (рис. 3.1), на яких сам контакт та його стан позначено через х, а стан двополюсника позначатимемо літерою *y*.



Вочевидь, змінна *х* є незалежною, а змінна y – залежною бульовою змінною.

У разі першої схеми коло буде замкнене, якщо буде змінено стан контакту (замкнено), тобто змінна y набуде істинного значення () тоді й лише тоді, коли змінна *х* також набуде істинного значення (). У разі другої схеми, навпаки, змінна у набуде істинного значення (), коли змінна *х* збереже хибне значення, тобто стан контакту не зміниться ().

Перейдемо тепер до розглядання схеми «або» і схеми «і» (рис. 3.2).



Якщо контакти та з’єднані паралельно, то коло буде замкнене, тобто змінна у набуде істинного значення (), коли хоча б один з контактів та є замкненим, і розімкненим, тобто у набуватиме хибного значення, коли обидва контакти та є розімкненими. При послідовному з’єднанні контактів та коло буде замкнене (), коли обидві змінні та набуватимуть істинного значення (тобто , ), і розімкнене (), коли хоча б одна зі змінних та набуде хибного значення.

Опис більш складних релейно-контактних схем здійснюється за допомогою бульових функцій.

У логіці висловлювань використовують п’ять логічних операцій: *заперечення* (читають «не» та позначають знаком «¬»), *кон’юнкцію* (читають «і (та)» й позначають знаком «˄»), *диз’юнкцію* (читають «або (чи)» та позначають знаком «˅»), *імплікацію* (читають «якщо ..., то» та позначають знаком «→»), *еквівалентність* (читають «тоді й лише тоді» та позначають знаком «~»).

Наприклад, якщо А та В – твердження, то , , , , будуть, відповідно, запереченням твердження *А*, кон’юнкцією тверджень А та В тощо.

**Приклад 3.3** Наведемо приклади складних висловлювань.

1. Сніг білий, і небо теж біле.

2. Якщо погода хороша, то ми їдемо відпочивати.

У наведених прикладах логічні операції – це «і» та «якщо ..., то».

**Приклад 3.4** Розглянемо такі висловлювання: *р*: «Вологість велика», *q*: «Температура висока», *r*: «Ми почуваємо себе добре». Тоді речення «Якщо вологість велика та температура висока, то ми не почуваємо себе добре» можна записати складним висловлюванням

.

У логіці висловлювань атом *р* чи складне висловлювання називають *правильно побудованою формулою* або *формулою*. Вивчаючи формули, розглядають два аспекти – *синтаксис* і *семантику*.

Під *алфавітом* будемо розуміти кожну непорожню множину символів:

пропозиційних змінних – , , , , , ...;

логічних зв’язок – ¬, ˄, ˅, →, ~, ...;

технічних символів – (,) тощо.

*Словом* у певному алфавіті називається довільна скінчена послідовність символів (можливо, порожня). Слово а називається *підсловом* *b*, якщо для певних слів та . Слово називається *сполученням* (*конкатенацією*) слів a та b.

*Синтаксис* – це сукупність правил, які дають змогу будувати формули та розпізнавати правильні формули серед послідовностей символів. *Формулою алгебри висловлень* (*ФАВ*) називається слово, яке задовольняє такому означенню:

1. кожна пропозиційна змінна – формула;
2. якщо та – формули, то , , , , – формули;
3. слово є формулою, якщо воно побудоване лише з використанням скінченної кількості правил.

Наприклад, вирази (слова) , є формулами, а слова , , , , – підформули останньої формули.

Часто заперечення висловлювання позначають також . Такий спосіб запису заперечення не потребує дужок. Якщо не виникає непорозумінь, то зовнішні дужки у формулах можна випускати.

З метою економії дужок операції виконуються в такому порядку (пріоритет операцій): ¬, ˄, ˅, →, ~.

Раніше зазначалося, що висловлення може бути чи то істинне, чи хибне. *Інтерпретувати формулу* – означає приписати їй одне з двох значень істинності.

*Семантика* – набір правил інтерпретації формул, який має бути композиційним, тобто значення формули має бути функцією значень її складових.

Нехай та – формули. Тоді значення істинності формул , , , , так пов’язані зі значеннями істинності формул та .

1. Формула істинна, коли хибна, і хибна, коли формула істинна. Формулу читають «не » чи «це не так, що » та називають *запереченням* формули .
2. Формула істинна, якщо та водночас істинні. У всіх інших випадках формула хибна. Формулу читають « і » й називають *кон’юнкцією* формул та .
3. Формула хибна, якщо та водночас хибні. У всіх інших випадках істинна. Формулу читають « або » й називають *диз’юнкцією* формул та .
4. Формула хибна, якщо формула істинна, a – хибна. У всіх інших випадках вона істинна. Формулу називають *імплікацією*, атом – припущенням імплікації, a – її висновком. Оскільки імплікацію використовують у багатьох математичних міркуваннях, то існує багато термінологічних варіантів для формули . Ось деякі з них: «якщо , то », «з випливає », « лише тоді, коли », « достатнє для », «, якщо », « необхідне для ».
5. Формула істинна, якщо та мають однакові значення істинності. У всіх інших випадках формула хибна. Формулу читають « тоді й лише тоді, коли » чи « еквівалентне » та називають *еквівалентністю* формул та .

Семантику логічних операцій зручно задавати за допомогою таблиць, які містять значення істинності формул залежно від значень істинності їх атомів. Такі таблиці називають *таблицями істинності*. Семантику введених операцій у формі таблиць істинності наведено в табл. 3.1.

Таблиця 3.1 – Таблиця істинності основних формул

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
| 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 | 1 | 1 |

**Приклад 3.5** Знайдемо заперечення висловлювання «Сьогодні п’ятниця». Воно має вигляд «Це не так, що сьогодні п’ятниця». Це речення також можна сформулювати як «Сьогодні не п’ятниця» чи «П’ятниця не сьогодні». Зазначимо, що речення, пов’язані з часовою змінною, – не висловлювання доти, доки не визначено момент часу. Це стосується й змінних у реченнях, які характеризують місце чи особу. Ці речення – не висловлювання, якщо не зазначено відповідного місця чи конкретної особи.

**Приклад 3.6** Знайдемо кон’юнкцію висловлювань та , де – висловлювання «Сьогодні п’ятниця», a – «Сьогодні падає дощ». Кон’юнкція цих висловлювань – «Сьогодні п’ятниця, і сьогодні падає дощ». Воно істинне в дощову п’ятницю й хибне в інший день або в недощову п’ятницю.

**Приклад 3.7** Що являє собою диз’юнкція висловлювань та з прикладу 3.6? Диз’юнкція висловлювань та – висловлювання «Сьогодні п’ятниця чи сьогодні падає дощ». Воно істинне в будь-яку п’ятницю чи в будь-який дощовий день (зокрема, у дощову п’ятницю) і хибне тільки в недощові «не п’ятниці».

Логічна операція «диз’юнкція» відповідає одному з двох способів уживання слова «чи (або)» в українській мові. Диз’юнкція істинна, якщо істинне принаймні одне з двох висловлювань. Розглянемо речення «Лекції з логіки можуть відвідувати студенти, які прослухали курси математичного аналізу чи дискретної математики». Його зміст полягає в тому, що лекції можуть відвідувати як студенти, які прослухали обидва курси, так і ті, хто прослухав тільки один із них. Але є й інше, альтернативне «чи (або)». Розглянемо речення «Лекції з логіки мають відвідувати студенти, які прослухали тільки один із двох курсів – математичного аналізу чи дискретної математики». Зміст цього речення полягає в тому, що студенти, які прослухали обидва ці курси, уже не повинні слухати лекції з логіки. Аналогічно, якщо в меню зазначено «Закуску чи салат подають із першою стравою», то це майже завжди означає, що з першою стравою буде подано чи закуску, чи салат, а не обидві страви. В останніх двох реченнях використано альтернативне «чи (або)»; його позначають знаком «». Значення істинності цієї операції наведено в табл. 3.2.

Таблиця 3.2 – Таблиця істинності формули

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Імплікацію як логічну операцію називають також *умовним реченням*. Щоб зрозуміти, чому імплікація набуває таких значень істинності, слід сприймати її як зв’язок обов’язкового й очікуваного. Наприклад, розглянемо звернення, адресоване студентам: «Якщо ви виконаєте всі завдання, то отримаєте відмінну оцінку». Це означає, що в разі виконання студентами всіх завдань вони одержать відмінну оцінку. Якщо ж студенти не виконають усіх завдань, то вони можуть отримати оцінку «відмінно», а можуть і не отримати її залежно від інших обставин. Однак якщо студенти зробили всі завдання, а викладач не поставив оцінку «відмінно», то студенти відчуватимуть себе ображеними. Це відповідає ситуації, коли в імплікації припущення «Ви виконаєте всі завдання» істинне, а її висновок «Ви отримаєте відмінну оцінку» хибний.

Розуміння імплікації в логіці дещо відрізняється від його розуміння в природній мові. Наприклад, «Якщо буде сонячно, то ми підемо на пляж» – умовне речення, уживане у звичайній мові. Воно залишається істинним до того моменту, коли настане сонячний день, а ми не підемо на пляж. За означенням імплікації умовне речення «Якщо сьогодні п’ятниця, то 2+3=5» істинне, бо висновок імплікації істинний. При цьому значення істинності припущення в імплікації тут не має відношення до висновку. Імплікація «Якщо сьогодні п’ятниця, то 2+3=6» істинна щодня, крім п’ятниці, хоча висловлювання 2+3=6 хибне. Останні дві імплікації ми не вживаємо в природній мові (хіба що як жарт), оскільки у кожному з відповідних умовних речень немає змістовного зв’язку між припущенням і висновком.

Конструкція «якщо , то », використовувана у вигляді «if then » в алгоритмічних мовах, відрізняється за змістом від імплікації в логіці. Тут – висловлювання, a – програмний сегмент, який складається з одного чи багатьох операторів. Програмний сегмент виконується, якщо висловлювання істинне, і не виконується, якщо воно хибне.

Для знаходження значення істинності складного висловлювання потрібно надати значення істинності всім атомам, які містить відповідна формула. Набір значень істинності всіх атомів формули називають її *інтерпретацією*. Для обчислення значень істинності формули, яка зображає складне висловлювання, потрібно знаходити значення логічних операцій, визначених табл. 3.1. Послідовність обчислень задають парами дужок. Якщо формула має атомів, то є способів надати значення істинності її атомам, тобто така формула має інтерпретацій, а всі її значення можна звести в таблицю істинності з рядками. Формулу, яка містить атомів, називають - *місною*.

Таким чином, якщо значення істинності простих висловлень є відомі, то значення істинності складних висловлень може бути визначено за допомогою цих таблиць.

**Приклад 3.8** Покажемо істинність формули за будь-яких інтерпретацій (табл. 3.2).

Таблиця 3.2 – Таблиця істинності формули

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
| 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 | 1 | 1 |

Якщо існує хоча б одна інтерпретація, за якої формула набуває істинного значення, вона називається здійсненною, а якщо існує хоча б одна інтерпретація, за якої формула набуває хибного значення, вона називається спростовною.

Формула називається тавтологією (чи *тотожно-істинною*, чи *загальнозначущою*), якщо за будь-яких інтерпретацій її складових (змінних) вона набуває істинного значення. Позначення тавтології: .

Формула називається протиріччям (*тотожно-хибною*), якщо за будь-яких інтерпретацій вона набуває хибного значення.

**Приклад 3.9** Розглянемо формулу , що має інтерпретацій (табл. 3.3).

Таблиця 3.3 – Таблиця істинності формули

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

Ця формула істинна в усіх інтерпретаціях, тобто являє собою тавтологію.

**Приклад 3.10** Розглянемо формулу , що має інтерпретацій (табл. 3.4).

Таблиця 3.4 – Таблиця істинності формули

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
| 0 | 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 |

Ця формула хибна в усіх інтерпретаціях, тобто являє собою протиріччя.

*Областю істинності* (*областю хибності*) формули називається множина наборів значень змінних, за яких формула набуває істинного (хибного) значення.

Результати цих означень:

* формула є тавтологією тоді й лише тоді, коли не є спростовною;
* формула є тотожно-хибна тоді й лише тоді, коли не є здійсненною;
* формула є тавтологією тоді й лише тоді, коли є тотожно-хибною;
* формула є тотожно-хибна тоді й лише тоді, коли тавтологія;
* формула – тавтологія тоді й лише тоді, коли та набувають однакових значень за всіма наборами значень змінних.

* 1. Основні закони алгебри висловлень

Дві формули є *рівносильні* тоді й лише тоді, коли за будь-яких інтерпретацій їхніх змінних вони набувають однакових значень. Позначення: .

Теорема 3.1 Дві формули та є рівносильні тоді й лише тоді, коли .

**Наслідок**. Нехай – формула, в якій є певні входження формули , і нехай – результат заміни цього входження формули на формулу . Тоді:

якщо , то ;

якщо і то .

Теорема 3.2 Якщо і , то .

Подані теореми дозволяють здійснювати еквівалентні перетворювання формул і здобувати нові загальнозначущі формули.

Наприклад, тавтології можна здобути з рівносильності заміною знака = на знак ~. Скажімо, з рівносильності здобуваємо тавтологію . Доведення тавтології, наприклад,

,

можна виконати за допомогою перетворень:

.

Тотожно-істинні формули та формули рівносильності називаються *законами алгебри висловлень* (*властивостями*, *правилами*, *теоремами*). Існує нескінчена множина тавтологій та рівносильностей, а отже, і законів алгебри висловлень. Нижче наведено закони (з використанням пропозиційних змінних), які найчастіше зустрічаються на практиці:

, , , – закони сталих (констант);

– закон тотожності;

– закон подвійного заперечення;

– закон протиріччя;

– закон вилученого третього;

, – комутативність ˅ та ˄;

, – асоціативність ˅;

– – асоціативність ˄;

– перший дистрибутивний закон;

– другий дистрибутивний закон;

– перший закон поглинання;

– другий закон поглинання;

, – ідемпотентність;

– перший закон склеювання;

– другий закон склеювання;

, – закони де Моргана;

– правило твердження, modus ponens;

– правило спростування, modus tollens.

Закони асоціативності дають змогу записувати багатомісні диз’юнкції та кон’юнкції без дужок. За допомогою правил та можна усувати логічні операції імплікації й еквівалентності з формул. Ці правила можна використовувати також для введення імплікації й еквівалентності.

**Приклад 3.11** Застосувавши закони логіки висловлювань, доведемо еквівалентність формул і . Запишемо послідовність перетворень і назви використаних законів і правил:

.

* 1. Логічний наслідок

Формула алгебри висловлень є *логічним наслідком* з формули (позначається ), якщо є істинне на всіх наборах значень змінних, для яких є істинна. Наприклад, формула є логічним наслідком формули , тобто .

Теорема 3.3 Формула алгебри висловлень є логічним наслідком з формули тоді й лише тоді, коли формула є загальнозначущою, тобто

.

Логічний наслідок можна узагальнити на сукупності формул: формула алгебри висловлень є *логічним наслідком формул* та позначається як , якщо для довільного набору значень з істинності всіх , на цьому наборі випливає істинність . Наприклад, розглядаючи таблицю істинності, здобудемо три ілюстрації до наведеного означення:

– 6-й рядок;

– 6-й та 8-й рядки;

, – 1, 4 та 6-й рядки.

Таблиця 3.5 – Таблиця істинності для ілюстрації логічного наслідку формул сукупності

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |

Теорема 3.4 Формула алгебри висловлень є логічним наслідком формул тоді й лише тоді, коли формула є загальнозначущою, тобто .

Формули алгебри висловлень можна застосовувати для перевірки правильності логічних суджень, незважаючи на конкретний зміст висловлень.

Що ж стосується «здорового глузду», то він має виявлятися при використовуванні законів логіки висловлень у її конкретних додатках. Наприклад, висловлення «» – хибне. Однак воно стає істинним, якщо вважати, що число 100 записане у двійковій системі числення, а 10 – у десятковій.

**Приклад 3.12** Перевірити правильність наступного судження. Якщо замінити мікросхему (А), телевізор працюватиме (В) за умови, що напругу увімкнено (С). Мікросхему замінили, а напругу не увімкнули. Отже, телевізор не працюватиме.

*Розв’язання*. Це судження можна записати у вигляді

, .

Оскільки формули А, В, С не містять підформул, то можна перейти до відповідних пропозиційних змінних х, y, z.

Тоді даний логічний висновок набуде більш зручного вигляду:

, .

Судження буде слушним, якщо формула

є загальнозначуща. При викладках скористаємося основними законами алгебри висловлень:

.

Формула є загальнозначущою, отже, судження є правильне.

**Приклад 3.13** Я піду на лекцію () або залишуся в барі й вип’ю кави (). Я не піду на лекцію. Отже, я залишуся й вип’ю кави.

Запишемо логічне слідування:

.

Перевіримо загальнозначимість:

.

Судження є правильне.

* 1. Нормальні форми логіки висловлювань

*Літералом* називають атом або його заперечення. Приклади літералів – , , .

Літерал називають *позитивним*, якщо він не має знака заперечення, і *негативним*, якщо має. Пару літералів називають *контрарною*.

Говорять, що формулу записано в *кон’юнктивній* *нормальній формі* (КНФ), якщо вона має вигляд

,

де кожна з формул літерал або диз’юнкція літералів і всі формули різні.

**Приклад 3.14** Нехай , і – атоми. Тоді – формула, записана в КНФ. У ній і , тобто – диз’юнкція літералів , і , а – диз’юнкція літералів і .

Говорять, що формулу записано в *диз’юнктивній* *нормальній формі* (ДНФ), якщо вона має вигляд

,

де кожна з формул літерал або кон’юнкція літералів і всі формули різні.

**Приклад 3.15** Нехай , і – атоми. Тоді – формула, записана в ДНФ. У ній і , тобто – кон’юнкція літералів і , а – кон’юнкція літералів , і .

Довільну формулу можна перетворити в одну з нормальних форм, застосувавши закони логіки висловлювань. Для побудови нормальних форм потрібно виконати таку послідовність еквівалентних перетворень.

**Крок 1**. Застосувати правила та для усунення логічних операцій «» та «~».

**Крок 2**. Застосувати закон подвійного заперечення та закони де Моргана для перенесення знака заперечення безпосередньо до атомів.

**Крок 3**. Застосувати відповідні закони дистрибутивності для побудови нормальної форми. Щоб побудувати КНФ, потрібно використати дистрибутивний закон для диз’юнкції щодо кон’юнкції. Для побудови ДНФ слід застосувати дистрибутивний закон для кон’юнкції щодо диз’юнкції.

**Приклад 3.16** Побудувати ДНФ формули .

*Розв’язання*. Наведемо послідовність кроків та зазначимо застосовані закони логіки висловлювань.

.

Ми одержали ДНФ. Її можна спростити, якщо двічі використати закон поглинання: диз’юнктивний член поглинає члени і . Отже, – інша ДНФ заданої формули. Останні міркування свідчать, що ДНФ, загалом кажучи, не єдина.

**Приклад 3.17** Побудувати КНФ формули .

*Розв’язання*. Наведемо послідовність кроків та зазначимо застосовані закони логіки висловлювань.

.

Ми одержали шукану КНФ.

* 1. Математична індукція

У різних сферах своєї діяльності (науці, побуті чи виробництві) людина у своїх логічних висновках застосовує дедуктивний або індуктивний підходи, які ґрунтуються на поняттях дедукції й індукції. *Дедукція* (лат. deduction – висновок) являє собою перехід від загального до окремого, а індукція (лат. induction – наведення) – перехід від окремого до загального. Загальним для цих підходів є те, що вони доводять істинність чи хибність деяких тверджень, які до цього сприймались як гіпотези, тобто як *передбачувані припущення*.

Ці твердження поділяються на *загальні* й *окремі*. Твердження, що всі натуральні парні числа діляться на 2, є загальним, а твердження, що ціле число ділиться на 2, є окремим. Важливість дедукції полягає в тому, що вона дозволяє на основі загальних тверджень доводити окремі твердження. Дійсно, якщо парні числа діляться на 2, а число 6 парне, то звідси випливає, що число 6 повинне ділитися на 2.

Індукція на відміну від дедукції виходить з окремих тверджень і тому не завжди може переходити від них до загальних тверджень. Але все ж таки в багатьох випадках на її основі робляться близькі до достовірних вірогідні припущення. Наприклад, коли за деякими зразками визначаються властивості золота, то ці властивості людина поширює на все інше золото, яке є в світі. Такий підхід застосовується не лише щодо золота, але й щодо інших хімічних елементів, і поки що він себе виправдовував. Хоча повної гарантії такий підхід усе ж таки не дає, і тому він є неповним. Це ж саме стосується й багатьох законів природи. Їх людина не доводить, а виявляє. Але, крім такої неповної індукції, у математиці окремо використовується ще повна індукція.

*Звичайна* *індукція* являє собою індуктивний підхід до науки взагалі, а не тільки до математики і с неповною індукцією, оскільки не дає можливості одержувати завжди достовірну інформацію про властивості, які досліджуються.

Ця індукція хоча й широко використовується в математиці, але не завжди дає вірний і кінцевий висновок про істинність або хибність того чи іншого математичного твердження, оскільки вона створює його на основі певних окремих результатів, одержаних при дослідженні тієї чи іншої математичної властивості. Це твердження можна вважати дійсним лише в разі дослідження всіх без винятку можливих окремих результатів, що буває тільки тоді, коли їх кількість є скінченною. У випадку, коли кількість можливих результатів нескінченна, отримати висновок про істинність або хибність твердження щодо тієї чи іншої властивості за допомогою звичайної індукції неможливо в принципі.

Нехай маємо тричлен . Якщо підставити в нього замість нуль, то одержимо просте число 41; якщо одиницю, то – 43; якщо 2,3,4, 5, 6, 7, 8, 9, 10, то – числа 47, 53,61,71,83,87, 113, 131, 151 відповідно. Усі вони прості числа. Здасться, можна припустити, що в разі підстановки в цей тричлен будь-якого цілого додатного числа завжди в результаті будемо одержувати просте число. Однак це не так. Уже в разі зазначений тричлен ділиться на 41, а за умови будемо мати , тобто тричлен ділиться на 41. У цьому небезпека звичайної індукції, оскільки вона не гарантує позитивного результату в будь-якому випадку.

*Повна*, або *цілковита*, *індукція* – це математична індукція, яка дозволяє робити достовірні узагальнення на основі неповної індукції. Тобто для доведення теореми з допомогою математичної індукції потрібно, щоб результат цієї теореми з деякою вірогідністю вже був встановлений раніше у вигляді гіпотези. Потрібно лише далі для неї довести, що цей результат є достовірно загальним для задач даного типу. Тобто доведенню тієї чи іншої теореми, яка відображає розв’язок відповідної математичної задачі, повинна передувати гіпотеза, яку й потрібно довести або відкинути. Це означає, що математична індукція ґрунтується на звичайній індукції.

Математична індукція на протилежність звичайній гарантує стовідсоткову вірогідність, тобто достовірність одержаних з її допомогою результатів. Цей метод стосується тільки теорем, що відображають загальні властивості натуральних чисел 1, 2,..., та інших розділів математики, які спираються на натуральні числа. Наприклад, до таких розділів належить арифметика цілих чисел та теорія раціональних чисел, що ґрунтується на ній. Існують також розділи математики, які можуть бути інтерпретовані в термінах арифметики, наприклад, евклідова геометрія. Відповідно, у цих розділах також може бути використаний метод математичної індукції. Математична індукція, яка ще має назву *індукція за побудовою*, використовується також для доведення логічних формул.

Основою метода математичної індукції є її принцип, що поширюється на будь-які твердження , які стосуються чисел . Сформулюємо цей принцип у вигляді теореми.

**Теорема 3.5** Будь-яке твердження дійсне для будь-якого у випадку, якщо воно дійсне для , та із істинності цього твердження для будь-якого довільного випливає його істинність для .

Ця теорема розпадається на дві леми, перша з яких (**лема 1**) вимагає, щоб твердження було справедливим для , а друга (**лема 2**) – для , за умов, що твердження справедливе для . Лише в цьому випадку твердження буде справедливе для будь-якого .

*Доведення*. Припустимо, що умови лем 1 і 2 виконуються. Тоді твердження відповідно до леми 1 дійсне для . Відповідно до другої леми дійсне для . Але якщо дійсне для , то відповідно до тієї самої другої леми воно буде дійсне і для і далі для тощо необмежено для всіх можливих . *Теорему доведено*.

Із теореми випливає *метод математичної індукції*, що складається із виконання нижченаведених пунктів, які стосуються одержання твердження :

1. Висувається нова гіпотеза у вигляді твердження про деяку математичну властивість, що може бути як істинною, так і хибною.
2. Здійснюється перевірка гіпотези для . Якщо підтверджується, то відбувається перехід до наступного пункту 3. А якщо ні, то гіпотеза, що перевіряється, вважається неправильною й виконується перехід до пункту 1.
3. Здійснюється доведення гіпотези для за умови припущення, що істинне.
4. Якщо істинне, то доведення гіпотези для будь-якого натурального одержане. Якщо хибне, то здійснюється перехід до пункту 1.

Перший крок наведеного алгоритму являє собою звичайну (неповну) індукцію, яка реалізує індукційний підхід до науки в цілому.

Другий і третій кроки алгоритму є наслідком дії принципу математичної індукції та реалізують його практично. Принцип математичної індукції гарантує за умови, якщо другий і третій кроки методу виконані, що твердження дійсне. При цьому другий крок ґрунтується на лемі 1 і є основою (базою) метода математичної індукції, а третій – на лемі 2, яка визначає індукційний крок (перехід) методу.

Слід ще раз звернути увагу на те, що основою метода математичної індукції є принцип математичної індукції, який твердить, що якщо доведені леми 1 і 2 для твердження , то це твердження істинне для будь-якого . Тому завданням метода є доведення лем 1 і 2 для .

**Приклад 3.18** Доведіть, що при кожному натуральному стверджується рівність

.

*Розв’язання*.

Якщо , то .

Припустимо, що при , рівність правильна.

Якщо , то

,

,

.

Індуктивний перехід правильний, а тому рівність доведено.

Слід особливо підкреслити, що для правильного використання метода математичної індукції обов’язково потрібно доводити обидві леми 1 і 2, оскільки лема 1 є основою для проведення індукційних кроків у методі, що розглядається, а лема 2 дозволяє виконувати правильний перехід від випадку з до випадку з , який іде за ним. Якщо лема 1 не доведена, то тоді відсутня основа для проведення індукційних кроків. У результаті за допомогою леми 2 можна довести помилкову гіпотезу .

1. **КОМБІНАТОРИКА**

У комбінаторному аналізі (комбінаториці) вивчають об’єкти зі скінченної множини та їх властивості, а також визначають кількість об’єктів із певними властивостями. Розглядають також твердження (принципи), які використовують в різних задачах. На них ґрунтуються важливі методи математичного доведення, широко застосовувані в теорії скінченних автоматів та інших розділах.

* 1. **Основні правила комбінаторного аналізу. Розміщення та сполучення**

Почнемо з формулювання двох основних правил комбінаторики: правила суми та правила добутку.

***Правило суми***. Якщо об’єкт можна вибрати способами, а інший об’єкт – способами, то можна вибрати або , або способами.

**Приклад 4.1** Студент має вибрати тему курсової роботи зі списку, розміщеного на трьох аркушах. Аркуші містять відповідно 20, 15 і 17 тем. З якої кількості можливих тем студент робить свій вибір?

*Розв’язання*. За правилом суми кількість тем для вибору становить .

***Правило добутку***. Якщо об’єкт можна вибрати способами та після кожного такого вибору об’єкт можна вибрати способами, то пару об’єктів у зазначеному порядку можна вибрати способами. Це правило можна пояснити інакше. Нехай якусь процедуру можна виконати розв’язавши два завдання. Якщо є способів розв’язати перше завдання та способів розв’язати після цього друге завдання, то всю процедуру можна виконати способами.

**Приклад 4.2** В одній із версій мови БЕЙСІК ім’я змінної – це рядок з одного чи двох символів, якими можуть бути 26 букв латинського алфавіту та 10 цифр. Першим символом має бути буква. Крім того, не можна використовувати п’ять двосимвольних рядків, які зарезервовані для спеціального використання. Знайти, скільки різних імен змінних є в цій версії мови БЕЙСІК.

*Розв’язання*. Нехай – величина, яку потрібно обчислити, – кількість односимвольних імен, – двосимвольних. За правилом суми всього імен . Очевидно, що ; за правилом добутку . Отже, .

Розглянемо основні комбінаторні об’єкти – розміщення та сполучення, попередньо означивши важливе поняття вибірки.

Нехай задано скінченну непорожню множину і виконано таких кроків.

**Крок 1**. Із множини *А* вибирають якийсь елемент .

**Крок 2**. Із множини *А* чи з вибирають якийсь елемент .

**Крок** . Якщо – елементи, які вибрані на перших кроках (), то на цьому кроці вибирають якийсь елемент із множини чи . Тоді елементи утворюють *вибірку обсягом* , або -*вибірку*, із множини *А*.

Вибірку називають *впорядкованою*, якщо задано порядок її елементів, а ні – то *невпорядкованою*. Зрозуміло, що впорядкована -вибірка – це кортеж (вектор) з компонентами, і тому її позначають , . Невпорядковану - вибірку позначатимемо як , .

Упорядковані - вибірки з - елементної множини називають *розміщеннями* з елементів по , а невпорядковані – *сполученнями* з елементів по . Використовують також поняття - розміщення й - сполучення. Розглянемо два способи вибору елементів.

Згідно з першим способом вибору на кожному кроці вибирають елемент з усієї множини *А*. Отже, один і той самий елемент із множини *А* може зустрітись у вибірці декілька разів. Такі вибірки називаються *вибірками з повтореннями*.

У разі застосування другого способу вибраний елемент вилучають із множини *А*. Це означає, що на кожному -му кроці () вибирають елемент із множини , і вибірка не містить однакових елементів. Такі вибірки називають *вибірками без повторень*.

**Приклад 4.3** Задано множину , тобто .

*Розв’язання*. Наведемо розміщення без повторень із трьох елементів по два, тобто :

, , , , , ;

розміщення з повтореннями з трьох елементів по два:

, , , , , , , , ;

сполучення без повторень із трьох елементів по два:

, , ;

сполучення з повтореннями із трьох елементів по два:

, , , , , .

Зазначимо, що сполучення без повторень з елементів по – це просто - елементні підмножини множини з елементів; отже, їх можна записати так: , , . Сполучення з повтореннями – це, узагалі кажучи, не множина у звичайному розумінні: її елементи можуть повторюватись, тобто зустрічатися більше одного разу.

* 1. **Обчислення кількості розміщень і сполучень**

Кількість усіх розміщень без повторень з елементів по позначають як , де і – невід’ємні цілі числа, причому . Кількість різних розміщень із повтореннями з елементів по позначають як . Тут і – будь-які невід’ємні цілі числа. Кількість усіх сполучень без повторень з елементів по позначають як , де і – невід’ємні цілі числа, причому . Кількість усіх сполучень із повтореннями з елементів по позначимо як , де i – будь-які невід’ємні цілі числа. Числа називають *біноміальними коефіцієнтами*. Доведемо, що

, (4.1)

, (4.2)

, (4.3)

. (4.4)

Доведемо рівність (4.1). Розглянемо якесь розміщення без повторень з елементів по . Ми можемо взяти як будь-який з елементів, як – будь-який з елементів, що залишились, і продовжити цей процес. Отже, для залишається можливостей вибору. Використавши правило добутку, переконуємось у тому, що рівність (4.1) правильна.

Рівність (4.2) також справджується, бо в розміщенні з повтореннями для кожного елемента є незалежних можливостей вибору.

Доведемо рівність (4.3). Розглянемо якесь сполучення без повторень з елементів по . Виявимо, скільки можна отримати різних розміщень без повторень з елементів по із цього сполучення як з -елементної множини. За формулою (4.1) дістанемо . Очевидно, що в разі із двох сполучень без повторень і не можна одержати однакових розміщень без повторень з елементів по . Отже, , і рівність (4.3) доведено.

Нарешті, доведемо рівність (4.4). Замість -елементної множини розглянемо множину також з елементів. Кожну невпорядковану -вибірку з множини можна записати у вигляді , де , оскільки порядок елементів не суттєвий. Тоді – сполучення без повторень з елементів по . Розглянемо відображення множини всіх сполучень із повтореннями з елементів по на множину всіх сполучень без повторень з елементів по . . Два сполучення з повтореннями рівні, якщо вони складаються з однакових елементів і кратності цих елементів збігаються. Якщо , то й . Більше того, якщо – сполучення без повторень з елементів по , де , т – елемент множини сполучень із повтореннями з елементів по . Отже, – бієктивне відображення множини всіх сполучень із повтореннями з елементів по на множину всіх сполучень без повторень з елементів по . Рівність (4.4) доведено.

* 1. **Перестановки**

*Перестановка з елементів* – це особливий випадок розміщення без повторень з елементів, коли в розміщення входять усі елементи. Перестановки з елементів називають також -*перестановками*. Окремі -перестановки різняться лише порядком елементів. Кількість таких перестановок позначають як . Формулу для одержують із формули (4.1) для кількості розміщень без повторень:

. (4.5)

Розглянемо тепер задачу про перестановки елементів за умови, що не всі елементи різні (перестановки з повтореннями). Точніше, нехай є елементів різних типів, а число () – кількість елементів -го типу. Очевидно, що . Перестановки з елементів за такої умови називають *перестановками з повтореннями*. Кількість таких перестановок позначають як . Щоб знайти явний вираз для , візьмемо окрему перестановку та замінимо в ній усі однакові елементи різними. Тоді кількість різних перестановок, котрі можна отримати з узятої однієї перестановки, дорівнює . Якщо зробити це для кожної перестановки, то одержимо перестановок. Отже, , звідки

. (4.6)

**Приклад 4.4** Знайдемо кількість слів (рядків), які можна утворити, переставляючи букви слова PRODUCT. Оскільки жодна буква тут не повторюється, то можна утворити слів.

**Приклад 4.5** Знайдемо, скільки слів можна утворити, переставляючи букви слова SUCCESS. У цьому слові є повторні входження букв, тому скористаємося формулою для перестановок із повтореннями:

слів.

* 1. **Біном Ньютона**

*Біноміальними коефіцієнтами* називають числа – кількість сполучень з елементів по . Розглянемо деякі властивості біноміальних коефіцієнтів.

1. Нехай і – невід’ємні цілі числа, . Тоді . Справді,

.

1. *Рівність Паскаля*: .

Позначимо як к множину всіх сполучень з елементів по елементів; як – відповідно з елементів по ; – з елементів по . Кожному сполученню з , яке містить елемент , відповідає сполучення з . Якщо ж сполучення з не містить , то йому відповідає сполучення з . Отже, існує бієкція між множинами й . Оскільки очевидно, що , то , тобто .

Отримане рекурентне співвідношення дає змогу побудувати таблицю для чисел , яку називають *трикутником Паскаля* (табл. 4.1).

У рядку , трикутника Паскаля стоять коефіцієнти , до того ж кожен коефіцієнт, крім двох крайніх, які дорівнюють 1, дорівнює сумі двох коефіцієнтів з попереднього рядка, які стоять над ним (див. табл. 4.1).

Сума біноміальних коефіцієнтів одного рядка дорівнює і, крім того, сума біноміальних коефіцієнтів, які стоять на непарних місцях, дорівнює їх сумі на парних. У свою чергу, кожна з цих сум дорівнює .

Числовий трикутник відповідно до властивості симетрії біноміальних коефіцієнтів строго симетричний відносно його середньої лінії. Тому при використанні його для обчислення біноміальних коефіцієнтів достатньо побудувати будь-яку одну його половину.

Таблиця 4.1 – Трикутник Паскаля

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *n k* | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | … |
| 0 | 1 |  |  |  |  |  |  |  |  |  | … |
| 1 | 1 | 1 |  |  |  |  |  |  |  |  | … |
| 2 | 1 | 2 | 1 |  |  |  |  |  |  |  | … |
| 3 | 1 | 3 | 3 | 1 |  |  |  |  |  |  | … |
| 4 | 1 | 4 | 6 | 4 | 1 |  |  |  |  |  | … |
| 5 | 1 | 5 | 10 | 10 | 5 | 1 |  |  |  |  | … |
| 6 | 1 | 6 | 15 | 20 | 15 | 6 | 1 |  |  |  | … |
| 7 | 1 | 7 | 21 | 35 | 35 | 21 | 7 | 1 |  |  | … |
| 8 | 1 | 8 | 28 | 56 | 70 | 56 | 28 | 8 | 1 |  | … |
| 9 | 1 | 9 | 36 | 84 | 126 | 126 | 84 | 36 | 9 | 1 | … |
| … | … | … | … | … | … | … | … | … | … | … | … |

1. Послідовність дійсних чисел називають *унімодальною*, якщо існує такий натуральний номер , що ; , тобто:

* послідовність строго зростає на відрізку , ;
* послідовність строго спадає на відрізку , ;
* максимальне значення досягається не більше ніж у двох точках: і, можливо, .

Нагадаємо, що як позначають найбільше ціле число, яке менше чи дорівнює (*цілу частину* числа ); наприклад, , .

**Теорема 4.1** За фіксованого послідовність біноміальних коефіцієнтів (), , унімодальна, . У разі парного максимум досягається в точці , а в разі непарного – у двох точках: і . Вказівка для доведення: оцінити відношення двох сусідніх членів послідовності .

1. .

*Доведення*. На основі рівності Паскаля запишемо рівності:

, , …, ,

.

Оскільки , то, після підстановки виразу для у вираз для , отримаємо:

.

Після того, як підставимо коефіцієнти для у попередню йому рівність одержимо, що

.

Потім таку ж саму операцію виконаємо для решти коефіцієнтів безпосередньо до і в результаті отримаємо необхідну рівність, яка доводить дану теорему.

1. .

*Доведення*. На основі рівностей

и

маємо:

,

потім

і гак до одержання в кінцевому підсумку необхідного результату.

1. При ,

.

*Доведення*.

.

Оскільки відношення

,

то

.

**Наслідок 1** .

**Наслідок 2** .

1. При ,

.

*Доведення*. Відповідно до властивості 6

.

Унаслідок властивості 1, згідно з якою

,

.

1. *Рівність Вандермонда*. Нехай , , – невід’ємні цілі числа, причому . Тоді . Вказівка для доведення: скористатися правилом добутку.

**Теорема 4.2 (біноміальна)** Нехай та – змінні, – додатне ціле число. Тоді

.

*Доведення*. Дамо комбінаторне доведення цієї теореми [1]. Оскільки отримано внаслідок -кратного вибору і -кратного вибору з співмножників у виразі , то коефіцієнт при дорівнює кількості способів -кратного вибору з співмножників, тобто . Друга рівність випливає з того, що .

Легко переконатись, що .

**Приклад 4.6** Знайдемо розклад виразу . Скориставшись біноміальною теоремою, можемо записати:

.

Біноміальні коефіцієнти можна брати з трикутника Паскаля чи обчислювати за формулою (4.3).

**Приклад 4.7** Визначимо коефіцієнт при в розкладі . Очевидно, що цей коефіцієнт дорівнює

.

*Властивості бінома Ньютона:*

1. Кількість членів розкладу на 1 більше за показник .
2. Показники степеня числа зменшуються, а числа збільшуються від члена до члена на 1.
3. Сума показників і в кожному члені дорівнює .
4. .

Доведення. Виходячи з того, що, при , з одного боку

,

а з іншого

,

то

.

1. Сума біноміальних коефіцієнтів бінома Ньютона, які займають парні місця, дорівнює сумі біноміальних коефіцієнтів, які займають непарні місця, і кожна з них дорівнює .

*Доведення*. Припустимо, що для бінома Ньютона і . Тоді

,

і оскільки

,

то і

.

Доданки у наведеному вище виразі приймають у випадку, якщо парне, додатні значення, які дорівнюють , а у випадку, якщо непарне, від’ємне: , де ; ; – значення верхнього параметра коефіцієнтів, які стоять на парних місцях, а – значення верхнього параметра коефіцієнтів, які стоять на непарних місцях. Тоді

,

де – максимальне значення ; – максимальне значення .

З одержаної рівності випливає, що

.

Оскільки з виразу, який наведений вище, виходить, що вказані в ньому суми дорівнюють одна одній, а з властивості 4 при , що

,

то кожна з наведених сум повинна дорівнювати

.

Тобто

, .

* 1. **Принцип включення-виключення**

Цей принцип дає відповідь на запитання, як визначити кількість елементів у об’єднанні множин.

Для двох множин має місце формула

.

**Приклад 4.8** Знайдемо кількість додатних цілих чисел, що не перевищують 1000 та діляться на 7 або на 11. Позначимо як *А* множину чисел, що діляться на 7, *В* – множину чисел, що діляться на 11. Тоді

.

Для трьох множин формула для кількості елементів у їх об’єднанні ускладнюється:

.

**Приклад 4.9** Одну з мов (англійську, німецьку, іспанську) вивчає 231 студент, причому , , , , , , де як *Е*, *D*, *S* позначено множини студентів, які відповідно вивчають англійську, німецьку й іспанську мови. Скільки студентів вивчають усі три мови?

*Розв’язання*. Маємо

,

звідки випливає, що студентів.

**Теорема 4.3 (принцип включення-виключення)** Нехай – скінченні множини. Тоді

.

*Доведення*. Достатньо довести, що кожний елемент в об’єднанні множин ураховано в правій частині рівності точно один раз. Припустимо, що елемент *а* належить рівно множинам з , де . Тоді цей елемент ураховано разів у , разів у ; загалом його враховано разів під час сумування членів, які містять перетин множин . Отже, елемент *а* враховано точно разів у виразі в правій частині рівності. За властивістю біноміальних коефіцієнтів . Отже, , але , і тому . Це й означає, що кожний елемент об’єднання множин ураховано в правій частині рівності точно один раз.

Зазначимо, що формула включення-виключення містить доданків, по одному для кожної непорожньої підмножини з .

Наступна форма принципу включення-виключення може бути корисною для розв’язування задач, у яких потрібно знайти кількість елементів заданої множини *А*, які не мають жодної з властивостей .

Уведемо такі позначення:

– підмножина елементів, що мають властивість ;

– кількість елементів множини *А*, які водночас мають властивості ;

– кількість елементів множини *А*, які не мають жодної з властивостей ;

– кількість елементів у заданій множині *А*.

Тоді, очевидно,

.

За принципом включення-виключення можна записати

.

Остання формула подає принцип включення-виключення *в альтернативній формі*.

**Приклад 4.10** Нехай колода складається з карт, пронумерованих числами 1, ..., . Скількома способами можна розташувати карти в колоді так, що для жодного () карта с номером не займає -е місце?

*Розв’язання*. Маємо властивостей виду « -а карта займає в колоді -е місце». Число всіляких розташувань карт в колоді дорівнює . Число розташувань, при яких карта з номером займає місце (), дорівнює . Тоді

, .

Використовуючи принцип включення-виключення в альтернативній формі, отримаємо, що число розташувань, при яких жодна з властивостей не виконується, дорівнює

.

1. **ФУНКЦІЇ АЛГЕБРИ ЛОГІКИ. БУЛЕВА АЛГЕБРА**
   1. **Способи задання булевих функцій**

Булевою функцією незалежних змінних називається функція , , в якій кожна змінна і сама функція набувають власних значень з множини , тобто , ; .

Кортеж конкретних значень булевих змінних називається набором, або булевим вектором. Якщо незалежні змінні розміщено у прямому порядку, тобто у вигляді , то набір називається прямим, а якщо їх розміщено у зворотному порядку, тобто у вигляді , то набір називається зворотним.

Областю визначення булевої функції аргументів є сукупність булевих кортежів. Число різних булевих функцій є скінченне і дорівнює . За число булевих функцій дорівнює 4, а за – 16.

Існують такі *способи задання* булевих функцій.

1. Табличний. Функція задається у вигляді таблиці істинності. Наприклад, така таблиця

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 1 | 0 |

визначає функцію .

1. Графічний. Функція задається у вигляді -вимірного одиничного куба, у вершинах якого записано значення функції (у кружечках) та набори значень аргументів. Наприклад, функції, задані на рис. 5.1 та 5.2.

1. ***Координатний*** (картою Карно). У клітинках карти записуються значення функції (нулі зазвичай не вписують, їм відповідають порожні клітини). Значення змінної визначається відрізками (дужками) з позначенням цієї змінної. Наявність відрізка відповідає 1, а відсутність – 0. Наприклад, функція, задана на рис. 5.3.



Відрізки карти Карно мають відбивати всі можливі набори значень. Для цього можна скористатися лівою частиною таблиці істинності, в якій рекомендується попередньо виконати перестановки наборів в такий спосіб, щоб зменшити загальне число розривів у відрізках. Наприклад, на рис. 5.4 подано перестановки наборів функцій двох та трьох незалежних змінних.



1. ***Числовий***. Функція задається у вигляді цілих десяткових (вісімкових, шістнадцяткових) чисел, які є еквівалентами тих наборів значень аргументів, на яких функція набуває значення 1.

Наприклад,

,

де і набір значень аргументів 011 відповідає значенню функції 1 (див. рис. 5.2), і т. д.

1. ***Аналітичний***. Функція задається у вигляді формули. Наприклад:

.

* 1. **Елементарні функції алгебри логіки**

Булеві функції однієї та двох незалежних змінних прийнято називати елементарними *бульовими функціями*. Вони використовуються як логічні операції над булевими змінними при побудові булевих функцій багатьох незалежних змінних. Алгебра з такими логічними операціями називається алгеброю логіки, а булеві функції називаються ще функціями алгебри логіки.

Загальне число різних елементарних функцій (логічних операцій) дорівнює загальному числу функцій двох змінних, тобто (функції однієї змінної є окремим випадком функцій двох змінних).

Основними в алгебрі логіки є три логічні операції.

*Заперечення* (*інверсія*) – функція , яка набуває значення 1, коли , і значення 0 – за . Позначення: . Читається: «не ».

*Диз’юнкція* (*логічне додавання*) – функція , яка набуває значення 0 тоді й лише тоді, коли обидва аргументи дорівнюють нулю. Позначення: або . Читається: « плюс » або « або ».

*Кон’юнкція* (*логічне множення*) – функція , яка набуває значення 1 тоді й лише тоді, коли обидва аргументи дорівнюють одиниці. Позначається: або , Читається: « помножити на » або « та ».

Таблиці істинності наведених логічних операцій мають відповідно вигляд:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
| 0 | 1 |  | 0 | 0 | 0 |  | 0 | 0 | 0 |
| 1 | 0 |  | 0 | 1 | 1 |  | 0 | 1 | 0 |
|  |  |  | 1 | 0 | 1 |  | 1 | 0 | 0 |
|  |  |  | 1 | 1 | 1 |  | 1 | 1 | 1 |

Якщо операція містить один операнд, вона називається одномісною, або унарною, а якщо два, то – двомісною, або бінарною. Заперечення – це одномісна операція, а диз’юнкція та кон’юнкція – двомісні. При цьому вирази , , є прикладами логічних формул. Більш складні формули дістаємо за рахунок суперпозиції логічних формул, які, звичайно беруться у круглі дужки. Наприклад: .

Елементарні функції однієї змінної.

Таблиця 5.1 – Таблиця істинності функції однієї змінної

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 |

Функції та – константи 0 та 1 відповідно. Позначення: ; . Функція набуває тих самих значень, що й , тобто . Функція , тобто це є логічна операція заперечення.

Елементарні функції двох змінних.

Таблиця 5.2 – Таблиця істинності функції двох змінних

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| Операція | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 0 |  |  |  |  |  |  |  |  |  |  | + |  |  |  | 1 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Функції та – константи 0 та 1. Ці функції відрізняються від та формально. Функції … є унарні операції, а функції … – бінарні.

Функції та – це розглянуті вище операції диз’юнкції та кон’юнкції.

Функція – це додавання за модулем 2. Позначення:

.

Функція називається *еквівалентністю*. Позначення:

.

Функція – *імплікація*: .

– *заборона* (*заперечення імплікації*): .

– *стрілка Пірса* (*функція Вебба*), .

– *штрих Шеффера*, .

Решта функцій спеціальних назв не мають і, як буде показано далі, легко виражаються через вищенаведені функції.

Зауважимо, що ці функції є *інверсними*, тобто

, , …, .

Технічну реалізацію функцій однієї змінної наведено на рис. 5.5.



Технічну реалізацію деяких функцій двох змінних показано на рис. 5.6.



* 1. **Основні властивості функцій алгебри логіки**

Формулою алгебри логіки або логічним виразом називається скінчена послідовність булевих змінних та функцій, пов’язаних знаками логічних операцій та круглими дужками.

Функція алгебри логіки – це рівність, у лівій частині якої стоїть булева змінна, а у правій – логічний вираз. Отже, функція алгебри логіки визначається формулою.

Наприклад, – логічний вираз, – булева функція.

При обчислюванні логічних виразів дотримуються такого пріоритету операцій: насамперед обчислюються функції, потім заперечення, після чого логічне множення і, врешті, логічне додавання. Вирази, які стоять у дужках, обчислюються в першу чергу. Інші операції мають найменший пріоритет. Порядок їхнього виконання визначається круглими дужками.

Функції, які зводяться до залежності від меншого числа змінних, називаються виродженими, а функції, які суттєво залежать від усіх змінних, є невиродженими. Наприклад, серед функцій однієї змінної є дві вироджені функції. Це , , які можна розглядати як функції від нуля змінних.

Функції двох змінних містять ті самі константи і чотири функції однієї змінної – , , , .

Функція багатьох змінних називається *функцією, яка* зберігає константу 0, якщо :

.

Наприклад, функції ... мають цю властивість, а функції ... цієї властивості не мають (див. табл. 5.2).

Функція змінних називається *функцією, яка* зберігає константу 1, якщо :

.

Наприклад, функції , де , мають цю властивість, а функції – не мають.

Логічна функція називається двоїстою до функції , якщо має місце рівність . Операція побудови двоїстої функції складається з інвертування функції, всі змінні якої теж інвертовані. Для констант інверсію застосовують тільки до функціональних символів 0 або 1.

Наприклад, функція має властивість двоїстості до функції , тому що .

Логічна функція називається самодвоїстою, якщо :

.

Наприклад, функція є самодвоїстою, тому що (перевіряється за допомогою таблиці істинності).

Функція багатьох змінних називається монотонною, якщо для будь-якої пари наборів значень її аргументів та , які задовольняють нерівності , , виконується нерівність :

.

Наприклад, функція є монотонною (див. табл. 5.2).

Функція багатьох змінних називається лінійною, якщо її можна подати у вигляді многочлена , де , :

.

Наприклад, функція є лінійною, тому що .

Система функцій алгебри логіки називається повною, якщо кожна інша функція алгебри логіки може бути виражена за допомогою суперпозицій цих функцій. При цьому стверджують, що повна система функцій утворює *базис* у логічному просторі.

Мінімальним базисом є такий базис, вилучення з якого будь-якої функції порушує його повноту.

Теорема 5.1 (Поста-Яблонського) Для того щоб система функцій була повною, необхідно і достатньо, щоб вона містила в собі хоча б одну функцію: незберігаючу константу 0, незберігаючу константу 1, несамодвоїсту, немонотонну й нелінійну.

З теореми випливає, що таких функцій має бути п’ять. Але, через те що деякі функції мають одразу кілька потрібних властивостей, базис може складатися з меншого числа функцій.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Властивості функції** | **Функції** | | | | | | | | | |
| 0 | 1 | ¬ | + | • | / |  |  |  |  |
| Незберігаюча 0 |  | \* | \* |  |  | \* | \* | \* |  |  |
| Незберігаюча 1 | \* |  | \* |  |  | \* | \* |  | \* | \* |
| Несамодвоїста | \* | \* |  | \* | \* | \* | \* | \* | \* | \* |
| Немонотонна |  |  | \* |  |  | \* | \* | \* | \* | \* |
| Нелінійна |  |  |  | \* | \* | \* | \* | \* |  | \* |

З таблиці видно, що повними системами функцій будуть: {¬, +, •}, {¬, +}, {¬, •}, {/}, {}, {0, } тощо. Так, наприклад, алгебра Буля побудована на системі функцій {¬, +, •}, а алгебра Жегалкіна використовує базис {1, •, }.

* 1. **Булева алгебра та її основні закони**

Булевою алгеброю називається множина логічних функцій з операціями диз’юнкція, кон’юнкція та заперечення, – тобто алгебра, базисом якої є система функцій {¬, +, •}.

Операції булевої алгебри звичайно називають *булевими операціями*, а функції – *булевими функціями*.

Розглянемо тепер ***основні закони булевих операцій***:

1. комутативний (для диз’юнкції та кон’юнкції):

, ;

1. асоціативний:

, ;

1. дистрибутивний:

– перший дистрибутивний закон;

– другий дистрибутивний закон;

1. ідемпотентний:

, ;

1. інверсний (формули де Моргана):

, ;

1. закон вилученого третього (для диз’юнкції) і закон суперечності (для кон’юнкції):

, .

У булевій алгебрі мають місце такі властивості:

; ; ; ; ; ; .

Решта функцій двох змінних логіки виражаються через базис булевої алгебри в такий спосіб:

; ; ;

; ; .

У справедливості цих формул легко переконатися за допомогою таблиці істинності.

Закони булевої алгебри та її властивості надають можливість виконувати перетворювання логічних виразів з метою побудови найбільш простих (компактних) формул.

**Приклад 5.1** Спростити: .

*Розв’язання*.

.

Алгеброю Жегалкіна називається множина логічних функцій з операціями кон’юнкція, додавання за модулем 2 і константа 1, тобто алгебра, базисом якої є система функцій {1, •, }.

Подамо ***основні закони*** цієї алгебри:

* 1. комутативний:

, ;

* 1. асоціативний:

, ;

* 1. дистрибутивний:

;

* 1. ідемпотентний:

;

* 1. закон приведення подібних членів:

.

В алгебрі Жегалкіна мають місце такі властивості:

; ; .

Решта операцій алгебри логіки виражаються через базис цієї алгебри в такий спосіб:

; ; ;

; ;

; .

Функція алгебри Жегалкіна, подана у вигляді суми за модулем 2 добутків незалежних змінних, називається *канонічним многочленом*, або поліномом Жегалкіна.

Наприклад, – поліном Жегалкіна.

Лінійна функція

,

де , , є окремим випадком полінома Жегалкіна.

Можна довести, що для кожної функції алгебри логіки існує єдиний поліном Жегалкіна.

Якщо булеву функцію задано у вигляді ДДНФ, то для здобуття многочлена Жегалкіна треба: знак «+» замінити знаком «», заперечення замінити на вираз , розкрити дужки і зробити всі можливі спрощення.

Для зображення будь-якої логічної функції у вигляді полінома Жегалкіна застосовують еквівалентні перетворення ДНФ або ДДНФ, метод невизначених коефіцієнтів, карти Карно, метод трикутника.

**Приклад 5.8** Побудувати поліном Жегалкіна для функції за допомогою еквівалентних перетворень.

*Розв’язання*.

.

**Приклад 5.9** Побудувати поліном Жегалкіна для функції .

*Розв’язання*.

*1 спосіб (за допомогою метода невизначених коефіцієнтів)*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | | |
|  |  |  |  | Рівняння для визначення невідомого коефіцієнта |
| 0 | 0 | 0 | 1 |  |
| 0 | 0 | 1 | 1 |  |
| 0 | 1 | 0 | 0 |  |
| 0 | 1 | 1 | 1 |  |
| 1 | 0 | 0 | 0 |  |
| 1 | 0 | 1 | 0 |  |
| 1 | 1 | 0 | 1 |  |
| 1 | 1 | 1 | 0 |  |

Розв’язання системи алгебраїчних рівнянь, побудованої в останньому стовпчику таблиці, дозволяє отримати значення всіх невідомих коефіцієнтів:

, , , , , , , .

Після визначення коефіцієнтів та підстановки їх у функцію отримаємо зображення булевої функції у вигляді полінома Жегалкіна:

.

*2 спосіб (за допомогою еквівалентних перетворень)*

.

При розробці багатьох схем електронних пристроїв та вузлів дискретної автоматики використовуються логічні елементи, які реалізують функцію Вебба та функцію Шеффера.

Як було зазначено раніш, кожну з цих функцій можна використовувати як базис алгебри логічних функцій.

Функція Вебба:

; ; ;

; ; .

Функція Шеффера:

; ; ;

; ; .

Отже, як і в булевій алгебрі, кожну функцію чи операцію можна розкласти і в алгебрі Вебба, і в алгебрі Шеффера.

* 1. **Нормальні форми булевих функцій**

Елементарною диз’юнкцією (кон’юнкцією) називається диз’юнкція (кон’юнкція) скінченного числа булевих змінних, у якій кожна змінна зустрічається не більше одного разу в прямому чи інверсному вигляді. Наприклад:

, – елементарні диз’юнкції;

, – елементарні кон’юнкції.

Диз’юнктивною нормальною формою (кон’юнктивною нормальною формою) називається формула, яка містить диз’юнкцію (кон’юнкцію) скінченного числа різних елементарних кон’юнкцій (диз’юнкцій).

Позначення: ДНФ, КНФ.

Наприклад: , – ДНФ;

, – КНФ.

ДНФ (КНФ) називається досконалою і позначається ДДНФ (ДКНФ), якщо в кожній її елементарній кон’юнкції (диз’юнкції) подано всі змінні.

Наприклад: – ДДНФ;

– ДКНФ.

Для того щоб ***привести формулу до ДДНФ***, потрібно:

* за допомогою законів та властивостей булевої алгебри привести її до ДНФ;
* якщо в елементарній кон’юнкції не міститься змінної із загальної кількості змінних, які входять до даної формули, додати до цієї кон’юнкції співмножник і розкрити дужки;
* з однакових елементарних кон’юнкцій вилучити всі, окрім однієї

або

* скласти таблицю істинності висловлення;
* для кожного набору значень змінних , на якому функція дорівнює 1, виписати кон’юнкції всіх змінних; над тими змінними, які на цьому наборі дорівнюють 0, ставляться заперечення;
* всі такі кон’юнкції з’єднати знаком диз’юнкції.

**Приклад 5.2**

1 спосіб.

.

2 спосіб.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  | ДДНФ | |
| 0 | 0 | 0 | 1 | 0 | 1 |  |  |
| 1 | 0 | 0 | 1 | 0 | 1 |  |
| 0 | 1 | 0 | 1 | 0 | 1 |  |
| 0 | 0 | 1 | 0 | 0 | 0 |  |
| 1 | 1 | 0 | 1 | 1 | 1 |  |
| 1 | 0 | 1 | 0 | 0 | 0 |  |
| 0 | 1 | 1 | 0 | 0 | 0 |  |
| 1 | 1 | 1 | 0 | 1 | 1 |  |

Для того щоб ***привести формулу до ДКНФ***, доцільно спочатку привести її до ДНФ, а потім від ДНФ перейти до КНФ в такий спосіб.

Нехай ДНФ має вигляд

,

де – елементарні кон’юнкції, .

Формулу приведемо до ДНФ , де – елементарні кон’юнкції. Тоді

.

Застосовуючи правило де Моргана, перетворимо елементарні кон’юнкції на елементарні диз’юнкції , де . Отже, дістанемо КНФ

.

І, врешті, використовуючи закон суперечності та другий дистрибутивний закон, зробимо перехід від КНФ до ДКНФ.

Або

* скласти таблицю істинності висловлення;
* для кожного набору значень змінних , на якому функція дорівнює 0, виписати диз’юнкції всіх змінних; над тими змінними, які на цьому наборі дорівнюють 1, ставляться заперечення;
* всі такі диз’юнкції з’єднати знаком кон’юнкції.

**Приклад 5.3**

1 спосіб.

.

2 спосіб.

Введемо наступні позначення: , , , , , , , та складемо таблицю істинності.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  | ДКНФ | |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 |  |  |
| 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |  |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |  |
| 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |  |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |  |
| 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |  |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |  |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |  |

**Приклад 5.4**

.

Елементарна диз’юнкція (кон’юнкція), яка містить усі змінні, називається конституентою нуля (одиниці). Наприклад, якщо загальна кількість змінних , то – конституента нуля, a – конституента одиниці.

Вочевидь, що конституента нуля перетворюється на нуль лише за одного набору значень змінних. У нашому прикладі конституенті нуля відповідає набір (1, 0, 0). Аналогічно, конституента одиниці перетворюється на одиницю також лише за одного набору. Наприклад, конституенті одиниці відповідає набір (1, 0, 1).

Оскільки для заданої булевої функції її ДДНФ являє собою диз’юнкцію конституент одиниці, а її ДКНФ – це кон’юнкція конституент нуля, то дана функція перетворюється на одиницю чи нуль лише за відповідних цим конституентам наборів значень змінних. Справедливе є і зворотне твердження.

Це дозволяє за заданою таблицею істинності булевої функції одразу записати її досконалі нормальні форми і, навпаки, за заданою ДНФ – скласти таблицю істинності.

Досконалі форми для функції позначають:

для ДДНФ – ;

для ДКНФ – ,

де символ або позначає, що диз’юнкція або кон’юнкція виконуються за відповідними конституентами.

**Приклад 5.5** Знайти досконалі нормальні форми для функції Вебба.

*Розв’язання*.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

– ДДНФ;

– ДКНФ.

**Приклад 5.6** Перетворити функцію на ДДНФ.

*Розв’язання*.

– ДДНФ.

**Приклад 5.7** Для функції, заданої власною ДКНФ , скласти таблицю істинності.

*Розв’язання*.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

* 1. **Мінімізація булевих функцій**

Одна й та сама функція алгебри логіки може бути подана в певному базисі по-різному. Тому, наприклад, при побудові економних схем цифрових автоматів виникає проблема подання логічних функцій у мінімальній формі.

Мінімальною ДНФ (КНФ) *булевої функції* називається така ДНФ (КНФ), котра містить найменше число елементарних кон’юнкцій (диз’юнкцій) та змінних у них стосовно решти ДНФ (КНФ), які представляють дану функцію.

В інженерній практиці найчастіше мінімізується число змінних (число літер) у ДНФ (КНФ).

Нині розроблено чималу кількість методів (способів, прийомів) мінімізації в класі нормальних форм. Нижче розглянемо лише один з них – метод Квайна-Мак-Класкі мінімізації ДДНФ, який ґрунтується на систематичному застосовуванні операцій склеювання та поглинання:

, ,

де – елементарна кон’юнкція.

Бульова функція називається імплікантою функції , якщо вона перетворюється на одиницю при наборі змінних, на якому сама функція також дорівнює одиниці. Коротше кажучи, якщо , то й .

З означення випливає, що кожна конституента одиниці, яка входить до складу ДДНФ, або їхня диз’юнкція є імплікантою певної булевої функції.

Імпліканта g називається простою, якщо жодна її частина не може бути імплікантою функції *f*.

**Приклад 5.10** Для функції знайти всі імпліканти.

*Розв’язання*.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 |

; ; ;

; ;

; .

Імпліканти та є простими, решта – ні.

Можна довести, що кожна булева функція є еквівалентна до диз’юнкції власних простих імплікант.

Булеву функцію, зображену за допомогою простих імплікант, називатимемо скороченою *ДНФ*. Пошук мінімальної ДНФ здійснюється серед скорочених ДНФ шляхом їхнього простого перебирання. У розглянутому прикладі скорочена ДНФ має вигляд

.

Оскільки інших скорочених ДНФ немає, то ця форма й буде мінімальною.

***Метод Квайна-Мак-Класкі*** виконується в три етапи:

1. знаходження простих імплікант;
2. пошук скорочених ДНФ;
3. вибір з цих форм мінімальної.

Без обмеження спільності розглянемо його на конкретному прикладі. Нехай треба мінімізувати логічну функцію, задану таблицею істинності

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

**Перший етап**: знаходження простих імплікант. На першому кроці цього етапу слід виписати з таблиці істинності конституенти одиниці, розміщуючи їх за групами (див. 1-й крок в таблиці). Номер групи N відповідає кількості одиниць у конституенті; N може набувати значення від 0 до *n*, де n – загальна кількість змінних.

На другому кроці цього етапу виконаємо поелементне порівняння конституент (початкових імплікант) сусідніх груп, тобто здійснимо склеювання. Конституента 1-ї групи (0100) склеюється за змінною з конституентою 2-ї групи (0101) і за змінною – з конституентою 2-ї групи (1100). Конституента 2-ї групи (0011) склеюється за змінною з конституентою 3-ї групи (0111) і за змінною – з конституентою (1011) цієї ж групи тощо.

Результат склеювання, тобто загальну частину конституент, запишемо в наступний стовпець, роблячи прочерк «–» на місці вилученої змінної (2-й крок в таблиці). Конституенти, які брали участь в операції склеювання, позначимо символом «\*».

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1-й крок | | | | | |  | 2-й крок | | | | | |  | 3-й крок | | | | | |
| № гр. | \* |  |  |  |  |  | № гр. | \* |  |  |  |  |  | № гр. | \* |  |  |  |  |
| 1 | \* | 0 | 1 | 0 | 0 |  | 1 | \* | 0 | 1 | 0 | – |  | 1 |  | – | 1 | 0 | – |
| 2 | \* | 0 | 0 | 1 | 1 |  |  | \* | – | 1 | 0 | 0 |  | Прості імпліканти | | | | | |
|  | \* | 0 | 1 | 0 | 1 |  | 2 |  | 0 | – | 1 | 1 |  |  | – | 1 | 0 | – |  |
|  | \* | 1 | 0 | 0 | 1 |  |  |  | – | 0 | 1 | 1 |  |  | 0 | – | 1 | 1 |  |
|  | \* | 1 | 1 | 0 | 0 |  |  |  | 0 | 1 | – | 1 |  |  | – | 0 | 1 | 1 |  |
| 3 | \* | 0 | 1 | 1 | 1 |  |  | \* | – | 1 | 0 | 1 |  |  | 0 | 1 | – | 1 |  |
|  | \* | 1 | 0 | 1 | 1 |  |  |  | 1 | 0 | – | 1 |  |  | 1 | 0 | – | 1 |  |
|  | \* | 1 | 1 | 0 | 1 |  |  |  | 1 | – | 0 | 1 |  |  | 1 | – | 0 | 1 |  |
|  |  |  |  |  |  |  |  | \* | 1 | 1 | 0 | – |  |  |  |  |  |  |  |

Якщо початкова імпліканта (1-й крок) мала n змінних (розрядів), то кожна імпліканта 2-го кроку має змінну. Імпліканти 2-го кроку знову піддаються операції склеювання. При цьому склеюванню підлягають імпліканти сусідніх груп, в яких в одній і тій самій позиції стоїть символ «–». Після цього кроку дістаємо імпліканти, які містять змінних і т.і., допоки подальше склеювання стає неможливим.

Виписавши тепер з усіх кроків непозначені символом «\*» імпліканти, дістанемо сукупність простих імплікант.

**Другий етап**: пошук скорочених ДНФ. З цією метою складемо імплікантну таблицю.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Конституенти  Прості  імпліканти |  |  |  |  |  |  |  |  |
|  | \* |  | \* |  | \* |  |  | \* |
|  |  | \* |  |  |  | \* |  |  |
|  |  | \* |  |  |  |  | \* |  |
|  |  |  | \* |  |  | \* |  |  |
|  |  |  |  | \* |  |  | \* |  |
|  |  |  |  | \* |  |  |  | \* |

Кожен рядок цієї таблиці відповідає простій імпліканті, а кожен стовпець – початковій імпліканті (конституенті). Якщо проста імпліканта поглинає (накриває) конституенту одиниці, тобто є її частиною, то відповідна клітина матриці позначається символом «\*». Потім відшукаємо стовпці імплікантної таблиці, які мають лише по одній позначці. Такі позначки обводимо кружечком. Відповідні цим позначкам прості імпліканти називаються *базисними* і становлять так зване ядро бульової функції, яке неодмінно входить до скороченої ДНФ.

Після цього розглянемо різні варіанти вибору сукупності простих імплікант, які спільно накриють позначками інші клітини рядка імплікантної таблиці. Ці імпліканти разом з ядром утворять скорочену ДНФ.

З таблиці видно, що скороченими ДНФ для заданої функції f будуть:

* 1. ;
  2. ;
  3. ;
  4. ;
  5. .

**Третій етап**: вибір мінімальної форми. Серед цих скорочених ДНФ обирається та, яка задовольняє критерію мінімальності. При цьому враховуються економічні та технічні чинники її реалізації в конкретному цифровому пристрої. У нашому прикладі мінімальна ДНФ має вигляд

.

Якщо винести за дужки , здобудемо більш простий вираз:

,

який містить менше число змінних (літер). Така форма подання функції називається *дужковою*.

Технічну реалізацію цих форм даної функції подано на рис. 5.7.



**Зауваження**. Метод Квайна-Мак-Класкі можна використовувати і для здобуття мінімальної КНФ. Для цього слід розглянути значення функції і конституенти одиниці, які відповідають цим значенням. В наслідок дістанемо .

Потім треба виконати мінімізацію відповідно до вищевикладеного методу. Застосувавши формули де Моргана до дістаної мінімальної ДНФ для функції , знайдемо мінімальну КНФ для функції *f*.

1. ТЕОРІЯ ГРАФІВ
   1. Графи та відношення

Теорія графів – потужний апарат для розв’язування прикладних завдань найрізноманітніших галузей науки й техніки, до яких належать, наприклад: аналіз та синтез електричних кіл та систем, проектування мереж зв’язку та дослідження скінченних автоматів, мережеве планування й керування, вибір оптимальних маршрутів та потоків у мережах, моделювання життєдіяльності й нервової системи в живих організмах тощо.

Початок теорії графів як математичної дисципліни було покладено Ейлером у його відомих міркуваннях щодо кенігсберзьких мостів. Однак ця стаття Ейлера, опублікована 1736 року була єдиною упродовж майже 100 років. Інтерес до проблем теорії графів відродився близько середини ХІХ сторіччя і був зосереджений переважно в Англії. Існувало чимало причин для такого пожвавлення вивчення графів. Природничі науки вплинули на це завдяки дослідженням електричних мереж, моделей кристалів та структур молекул. Розвинення формальної логіки призвело до вивчення бінарних відношень у формі графів.

Величезна кількість популярних головоломок формулювалися безпосередньо в термінах графів. Найвідоміше з-посеред цих задач – проблема чотирьох фарб – уперше поставлена перед математиками де Морганом приблизно 1850 року (задача щодо визначення кількості припустимих фарб для розфарбування розбиття будь-якої площини так, щоб ніякі суміжні області не були однакового кольору). Жодна інша проблема теорії графів не породжувала стільки численних, часто дотепних робіт.

У разі потреби подавання в наочній формі системи взаємопов’язаних об’єктів звертаються до такої побудови: на площині чи у просторі обирають кілька точок і певні пари з цих точок поєднують лініями. Об’єкт, здобутий у наслідок такої побудови, називається *графом*.

За приклади графів можуть слугувати блок-схема алгоритму, з’єднання в електричній схемі, мережа шляхів поміж населеними пунктами.

Одну й ту саму систему об’єктів та зв’язків між ними можна відобразити по-різному, застосовуючи наведену вище побудову: у різні способи розміщувати точки, за їхні з’єднувальні лінії брати ті чи інші криві тощо.

Більш того, можна взагалі не зображати, а зазначити систему зв’язків об’єктів у якій завгодно іншій формі, наприклад у словесній. Це міркування засвідчує, що потрібне визначення графа як певного формального об’єкта, який можна подавати наочно у всілякі способи.

Стверджуватимемо, що задано скінченний неорієнтований граф, якщо задано такі два об’єкти:

* 1. скінчена непорожня множина ; елементи цієї множини називають вершинами графа;
  2. деяка множина невпорядкованих пар елементів з *X*; ця множина позначається *U*, її елементи називають ребрами.

Той факт, що граф означується парою множин Х та *U*, записують у вигляді .

За наочного подавання графа вершини зображуються точками, ребра – лініями, які з’єднують точки.

**Приклад 6.1** , де ; . Наочно цей граф зображено на рис. 6.l.



**Зауваження**. Якщо – ребро графа, то стверджують, що ребро з’єднує вершини та .

Поряд із наведеним визначенням графа можливі й інші визначення графа.

Іноді виникає потреба розглядати графи, в яких одну й ту саму пару вершин з’єднує кілька ребер. Такі графи називаються мультиграфами (рис. 6.2).

Можливі також графи, в яких певні ребра можуть мати збіжні кінці. Такі ребра називають петлями (рис. 6.3).

У більшості додатків теорії графів можна відкидати петлі й замінювати кратні ребра на одне ребром. Тому надалі подане вище визначення буде головним і словом «граф» позначатимемо скінченний неорієнтований граф без петель і кратних ребер (його ще називають простим, або звичайним) (рис. 6.4).

Граф з петлями і кратними ребрами називається псевдографом (рис. 6.5).

Поняття орієнтованого графа (орграфа) виникає, якщо ребрам графа надати напрямок (тобто орієнтацію) в такий спосіб, що один з кінців ребра буде початком, а інший – кінцем.

Стверджуватимемо, що задано *орієнтований граф*, якщо зазначено два об’єкти:

* + 1. непорожня скінчена множина X – вершини графа;
    2. множина *U*, утворена з впорядкованих пар вершин.

Елементи множини U називають дугами. Дуга орієнтованого графа зображується відрізком із зазначенням напрямку (стрілкою) (рис. 6.6).



|  |  |
| --- | --- |
| **Приклад 6.2** Орієнтований граф , де ; . Граф G зображено на рис. 6.7.  **Зауваження**. Якщо – дуга орграфа, то стверджують, що дуга виходить з вершини і закінчується у вершині . |  |

Стверджуватимемо, що задано *орієнтований мультиграф*, якщо зазначено два об’єкти:

* + 1. непорожня множина X – вершини графа;
    2. множина *U*, утворена з упорядкованих пар вершин.

Отже, елементи (дуги) в *U* в разі орієнтованого мультиграфа можуть повторюватись; такі дуги називають *кратними*. Зауважимо, що кратні дуги з’єднують одну пару вершин і однаково напрямлені.

Надалі ми будемо використовувати термін «граф» для опису довільних графів – орієнтованих і неорієнтованих, із петлями та кратними ребрами чи без них, а термін «неорієнтований граф» або «псевдо граф» – для довільного неорієнтованого графа, який може мати кратні ребра й петлі. Означення різних типів графів зведено в табл. 6.1.

Таблиця 6.1 – Означення різних типів графів

|  |  |  |  |
| --- | --- | --- | --- |
| **Тип графа** | **Ребра** | **Чи дозволені кратні ребра?** | **Чи дозволені петлі?** |
| Простий граф | неорієнтовані | ні | ні |
| Мультиграф | неорієнтовані | так | ні |
| Псевдограф | неорієнтовані | так | так |
| Орієнтований граф | орієнтовані (дуги) | ні | так |
| Орієнтований мультиграф | орієнтовані (дуги) | так | так |

Нехай задано граф .

Про ребро цього графа стверджують, що воно з’єднує вершини x та *у*.

Дві вершини називаються суміжними, якщо вони є кінцями одного ребра (відрізка або стрілки).

Про ребро та вершину *x* стверджують, що вони є інцидентні. Те ж саме можна сказати й про ребро та вершину *у*.

Далі позначатимемо *кількість вершин* графа – літерою *n*, а *кількість ребер* графа – літерою *m*: , . Це основні числові характеристики графа.

Кількість ребер, інцидентних до певної вершини *x*, називається степенем цієї вершини і позначається , або .

Вершина, в якої степінь дорівнює 0, називається ізольованою (вершина x рис. 6.8). Вершини, які мають степінь 1, називаються висячими, або кінцевими (вершина x рис. 6.9).

Справедливими є два такі простих твердження.

Теорема 6.1 Сума степенів усіх вершин графа дорівнює подвоєній кількості ребер.

*Доведення*. Кожне ребро двічі входить до суми, звідки й випливає твердження.

Теорема 6.2 У кожному графові число вершин, які мають непарний степінь, є парне.

*Доведення*. Нехай – множина вершин непарного степеня; – множина вершин парного степеня. Зазначимо, що

, ,

Отже

.

Тоді

.

Вочевидь, що є парна як сума парних чисел: – парна відповідно до теореми 1. Отже, – парна, що й треба було довести.

Для орієнтованих графів замість степеня вершини *х* вводять поняття півстепенів: *додатні* й *від’ємні* *півстепені* вершини *х*:

– число дуг, які входять до вершини *x*;

– число дуг, які виходять з вершини *x*.

Граф, який не має ребер (), називається порожнім. Усі вершини порожнього графа є ізольовані.

Граф, в якому кожна пара вершин з’єднана ребром, називається повним.

Повний -вершинний граф позначається ; для кожної його вершини *х* маємо (рис. 6.10).



Нехай задано граф .

Граф називається підграфом графа , якщо та .

Якщо вилучити з графа певні ребра та вершини, дістанемо підграфи вихідного графа (рис. 6.11).

Граф називається кістяковим підграфом графа , якщо та .

Кістяковий підграф здобудемо, якщо в графі G вилучимо частину ребер, не зачіпаючи вершин.

Відокремимо в графі G певну підмножину вершин .



Нехай означує множину ребер графа *G*, обидва кінці яких належать до множини А. Підграф називають *підграфом*, *породженим множиною вершин А*.

Способи задання графів

* + 1. Скінченний граф може бути задано переліком його елементів, тобто за визначенням (елементи позначаються латинськими літерами з індексами або просто натуральними числами).

Приклад 6.3 Граф, де , можна задати переліком елементів наступним чином:

.

Такий метод не є наочний, що утруднює виявлення характеристик графа.

* + 1. Геометричне задання графа.

Кожен граф може бути задано геометрично у тривимірному просторі, але не завжди його можна зобразити на площині так, щоб ребра перетинались тільки в вершинах. Граф, який може бути зображено на площині, називається планарним (рис. 6.12). Не є планарним повний граф з п’ятьма вершинами (рис. 6.13).

* + 1. ***Матричне задання графа***.

Не завжди зручно задавати граф у тому вигляді, як це зазначено вище. Наприклад, при опрацюванні графа на комп’ютері його зручно зображати в матричній формі.

1. Розглянемо – орграф, де

, .

Скінченний орієнтований граф задається *матрицями суміжності* та *інцидентності*.

*Матрицею* суміжності орграфа G називається квадратна матриця , в якої:

*Матрицею* інцидентності (або *матрицею* інциденцій) орграфа G називається матриця , в якої елементи:

**Приклад 6.4** Розглянемо орграф *G*, який задано геометрично (рис. 6.14). Для нього матриця суміжності матиме вигляд

.

|  |  |
| --- | --- |
| Матриця інцидентності матиме вигляд  . |  |

1. Розглянемо – скінченний неорієнтований граф, де

, .

Скінченний орієнтований граф задається *матрицями суміжності* та *інцидентності*.

*Матрицею* суміжності графа G називається квадратна матриця , в якої:

*Матрицею* інцидентності графа G називається матриця , в якої елементи:

|  |  |
| --- | --- |
| **Приклад 6.5** Розглянемо граф , який задано геометрично (рис. 6.15). Для нього матриця суміжності матиме вигляд  . |  |

Матриця інцидентності матиме вигляд

.

**Зауваження**. Матрицю суміжності можна визначити і для псевдографів. Тоді в разі орієнтованого (неорієнтованого) псевдографа , де – кратність дуги (ребра ) у цьому псевдографі.

Визначення матриці інцидентності без змін переносяться і на довільні мультиграфи (орієнтовані й неорієнтовані) і навіть на неорієнтовані псевдографи.

|  |  |
| --- | --- |
| **Приклад 6.6** Нехай задано геометрично орієнтований псевдограф G (рис. 6.16).  Тоді матриця суміжності матиме вигляд  . |  |

**Зауваження**. Матриця суміжності для звичайних графів і матриця інцидентності для будь-яких графів задає граф однозначно.

Нескладно з’ясувати, що матриця є симетричною для кожного неорієнтованого графа *G*. Матриця , де G – орграф, у загальному випадку не є симетричною.

За допомогою матриць зручно задавати графи (орграфи) для опрацювання на комп’ютері. Однак слід зазначити, що за великої кількості вершин матриця суміжності стає громіздкою. Те саме можна сказати і про матрицю інцидентності, причому її розміри залежать, окрім того, й від кількості ребер (дуг) графа.

При визначенні будь-якого математичного поняття домовляються, які об’єкти вважати за однакові й які треба розрізнювати.

Ізоморфні об’єкти – це такі об’єкти, які в подальшій теорії не розрізнюються і розглядаються як один об’єкт.

Наприклад, два графи та , зображені на рис. 6.17, відрізняються лише позначанням вершин і способом розміщення на площині.



Якщо в графі перепозначити вершини за схемою а – 1; b – 2; c – 3; d – 4, то множина вершин і ребер у графах та збігатиметься й матимемо той самий граф.

Графи та називаються ізоморфними, якщо поміж множинами їхніх вершин можна встановити взаємнооднозначну відповідність, за якої кожні дві вершини є суміжні в одному з графів тоді й лише тоді, коли відповідні їм вершини є суміжні в іншому графі.

**Приклад 6.7** Графи та , які зображені на рис. 6.18, є ізоморфні.



Якщо на певній множині М задано бінарне відношення , то R можна зобразити як граф . Такий граф не має кратних ребер, тому що , а при перелічуванні елементів множини М кожний з них зазначається лише один раз.

Правильним є й обернене твердження: кожен граф без кратних ребер задає певне бінарне відношення на множині його вершин. Якщо відношення є рефлексивне, то граф має в кожній вершині петлю; якщо R – симетричне, то – неорієнтований граф.

* 1. Елементи графів

Нехай – скінченний неорієнтований граф. Скінчена послідовність вершин та ребер графа

,

в якій кожне ребро є ребро, яке з’єднує вершини та , називається маршрутом на графі *G*.

Говорять, що цей маршрут з’єднує вершини та . Число називають довжиною маршруту.

Отже, довжина маршруту – це кількість ребер, які входять до маршруту. Маршрут називають замкненим, якщо .

Маршрут, в якому всі ребра є різні, називають ланцюгом. Замкнений ланцюг називають циклом.

Ланцюг називають простим, якщо всі його вершини є різні.

Простий цикл – це цикл, у якому всі вершини, окрім першої та останньої, є різні.

**Приклад 6.8** У графі, поданому на рисунку



– маршрут;

– ланцюг;

– простий ланцюг.

Орієнтовані маршрути на орграфі визначують в аналогічний спосіб, з тією різницею, що початкова вершина дуги маршруту має збігатися з кінцевою вершиною попередньої дуги.

Інакше кажучи, рух за маршрутом припускається лише в напрямках, зазначених стрілками.

Маршрут, який не містить повторних дуг, називається *шляхом*, а той, що не містить повторних вершин, – *простим шляхом*. Замкнений шлях називається *контуром*, а простий замкнений шлях – *простим контуром*.

Граф без циклів називається *ациклічним* (орграф – *безконтурним*), в іншому разі граф називається *циклічним* (орграф – *контурним*).

Умовимося вважати, що кожна вершина з’єднується сама з собою маршрутом довжини 0 і що цей маршрут є простим циклом. Такий цикл називають *нульовим* (якщо сказано просто цикл, то мається на увазі, що він не є нульовий).

Теорема 6.1 Нехай задано маршрут *S*. Тоді, якщо він не є замкнений, то містить простий ланцюг з одними й тими самими кінцями.

*Доведення*.

1) Нехай – маршрут, який з’єднує вершини та , де . Якщо він є простим ланцюгом, то твердження теореми доведено.

2) Припустимо, що маршрут S не є простим ланцюгом. Отже, знайдеться вершина (), яка входить до цього маршрута, принаймні двічі:

.

Тоді – замкнений підмаршрут маршрута *S*. Вилучимо його з маршрута S. Внаслідок цього дістанемо більш короткий маршрут з тими самими кінцями:

.

Якщо дістаний маршрут не є простим ланцюгом, повторивши попередні міркування, дістанемо новий маршрут .

Оскільки S – скінченний маршрут, то через скінчену кількість кроків m дістанемось певного маршруту , який не містить однакових вершин. Тоді – простий ланцюг.

Теорема 6.2 Кожний замкнений маршрут С містить простий цикл.

*Доведення*.

Якщо замкнений маршрут не є простим циклом, то знайдеться вершина , яка входить до даного маршрута, принаймні двічі, чи вершина , яка входить до маршрута принаймні тричі. Тоді з маршрута С можна виокремити більш короткий замкнений маршрут або . Якщо цей маршрут не є простим циклом, повторимо попередні міркування. Оскільки вхідний маршрут є скінченний, то через певну кількість кроків m дістанемо маршрут , який є простим циклом.

**Наслідок**. Якщо ланцюг не є простим, то він містить простий цикл.

Нехай граф – неорієнтований.

Вершина a називається *зв’язаною* з вершиною *b*, якщо існує маршрут, який з’єднує ці вершини. Стверджують, що вершина b досяжна з вершини *a*.

Граф, будь-яка пара вершин якого є зв’язана, називається *зв’язним*.

Якщо в довільному графі G вершина a зв’язана з *b*, а вершина b зв’язана з *с*, то, вочевидь, що a є зв’язана з *с*. Відношення зв’язності для вершин є відношенням еквівалентності. Отже, існує таке розкладання множини вершин *X*:

1. , за , що всі вершини b однієї множини Х; є зв’язані між собою, а вершини з різних множин не є зв’язані.
2. , за . Тоді, відповідно до (1) та (2) матиме пряме розкладання.
3. , де , , ... , – зв’язні підграфи, що не перетинаються.

Ці підграфи називаються зв’язними компонентами графа G, або компонентами зв’язності графа G.

Число р – ще одна числова характеристика графа.

Для зв’язного графа ; якщо граф є незв’язний, то .

Теорема 6.3 Кожен неорієнтований граф розкладається єдиним чином на пряму суму (3) власних зв’язних компонент.

**Зауваження**. Якщо певний граф не є зв’язним і розкладається на декілька компонентів, то вивчення цього незв’язного графа можна звести до досліджування окремих його компонентів, які є зв’язні. Тому у переважній більшості випадків має сенс припускати, що заданий граф є зв’язний.

Приклад 6.9 Граф, зображений на рисунку, має три компоненти зв’язності.



Через те що кількість компонентів зв’язності дорівнює кількості зв’язних підграфів графа, наведений граф – тризв’язний (число зв’язності ).

Зв’язність для орієнтованих графів (орграфів) визначається так само, як і для неорієнтованих, тобто без урахування напрямків дуг. Специфічними для орграфів є поняття сильної, однобічної та слабкої зв’язності.

Орграф називається сильно зв’язним, якщо для кожної пари його вершин а та b існує шлях з вершини а до вершини b.

Орграф називається однобічно зв’язним, якщо для кожної пари його вершин принаймні одна є досяжна з іншої.

Орграф називається слабко зв’язним, якщо зв’язним є асоційований з ним псевдограф (рис. 6.19 та 6.20).

**Приклад 6.10** Три компоненти , , сильної зв’язності орграфа *G* зображено на рисунку



Теорема 6.4 Нехай А – матриця суміжності графа . Тоді – число маршрутів довжини від вершини до вершини .

Зв’язний граф може бути розділено на незв’язані поміж собою підграфи вилучанням з нього певних вершин і/або ребер. При вилученні вершин вилучаються і всі інцидентні до них ребра, а при вилученні ребер інцидентні до них вершини зберігаються.

Вершина, вилучення якої перетворює зв’язний граф на незв’язний, називається точкою з’єднання (рис. 6.21).

Ребро, вилучання якого перетворює зв’язний граф на незв’язний, називається мостом (рис. 3.22).

За наявності моста є хоча б дві точки з’єднання.

Граф називають нероздільним, якщо він є зв’язний і не має точок з’єднання.

Граф, який має хоча б одну точку з’єднання, є роздільним і називається сепарабельним. Він розбивається на блоки, кожний з яких являє собою максимально нероздільні підграфи.

Нехай задано зв’язний граф . *Відстанню поміж двома вершинами* *х* та у графа G називається довжина найкоротшого ланцюга, який зв’язує ці вершини.

Відстань між вершинами х та у графа G позначається через або , якщо зрозуміло, про який саме граф йдеться.

Нескладно перевірити, що впроваджене в такий спосіб позначення відстані – задовольняє звичайним аксіомам метрики:

1. ; ;
2. ;
3. .

Діаметром графа називається величина , де максимум береться за всіма можливими парами вершин графа.

Визначимо для кожної вершини x графа G величину , тобто відстань від вершини *x* до найвіддаленішої від *x* вершини графа. Мінімум цієї величини за всіма вершинами графа називається радіусом графа G:

.

Вершина , в якій досягається цей мінімум, називається центральною.

*Периферійною точкою графа* називається вершина, від якої максимальна з відстаней до інших вершин була б максимальна.

*Алгоритм знаходження відстаней від даної вершини*

*до інших вершин графа :*

1. позначаємо через ;
2. позначаємо індексом всі вершини, суміжні з вершиною , виписуємо множину всіх цих вершин з їхніми позначками;
3. кожну вершину, що не належить множині і суміжну з кожною з вершин, що належать множині , позначаємо індексом ; виписуємо множину всіх цих вершин з їхніми позначками …;

) повторюємо описану процедуру доти, поки множина непомічених вершин не виявиться порожньою.

**Приклад 6.11** Визначити відстань від вершини 7 (для зручності запису позначимо вершини графа арабськими цифрами) до всіх вершин графа , зображеного на рис. 6.23.



*Розв’язання.* Згідно алгоритму відстань від вершини 7 будемо шукати в такий спосіб:

1) ;

2) ;

3) .

Більше непомічених вершин немає. Тобто відстані від вершини 7 до кожної з вершин графа такі

; .

Для визначення центра і радіуса графа необхідно побудувати для нього *матрицю відстаней* , кожен елемент якої описує відстань між вершинами і графа , тобто . Очевидно, що матриця відстаней симетрична щодо головної діагоналі (елементи якої дорівнюють нулю, тому що ).

**Приклад 6.12** Визначити центр, периферійні вершини, радіус і діаметр графа , зображеного на рис.  6.24.



*Розв’язання.* Матриця відстаней графа має вигляд.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |  |
| 1 | 0 | 1 | 1 | 2 | 3 | 4 | 4 | 4 | 4 | **4** |
| 2 | 1 | 0 | 1 | 1 | 2 | 3 | 3 | 3 | 3 | **3** |
| 3 | 1 | 1 | 0 | 1 | 2 | 3 | 3 | 3 | 3 | **3** |
| 4 | 2 | 1 | 1 | 0 | 1 | 2 | 2 | 2 | 2 | **2** |
| 5 | 3 | 2 | 2 | 1 | 0 | 1 | 1 | 1 | 1 | **3** |
| 6 | 4 | 3 | 3 | 2 | 1 | 0 | 1 | 1 | 2 | **4** |
| 7 | 4 | 3 | 3 | 2 | 1 | 1 | 0 | 1 | 1 | **4** |
| 8 | 4 | 3 | 3 | 2 | 1 | 1 | 1 | 0 | 1 | **4** |
| 9 | 4 | 3 | 3 | 2 | 1 | 2 | 1 | 1 | 0 | **4** |

Отже, згідно з означенням, центром графа є вершина 4; периферійні вершини – 1, 6, 7, 8, 9. Радіус графа , а діаметр графа .

Приклад 6.14 Знайти діаметр, радіус, центр і периферійні вершини для графа, зображеного на рис.  6.25.



*Розв’язання*.

Для розв’язання цієї задачі зручно попередньо обчислити так звану матрицю відстаней поміж вершинами графа. У даному разі це буде матриця розміром , елемент якої, що стоїть на місці , дорівнює відстані від вершини до вершини :

.

За визначенням, діаметр графа дорівнює найбільшому елементові матриці відстаней. Отже, . Периферійною точкою графа називається вершина, від якої максимальна з відстаней до інших вершин була б максимальна, отже вершини 1, 2, 4, 6, 7, 8, 9 – периферійні.

Для знаходження радіуса відшукаємо в кожному рядку найбільше число; ці числа виписано праворуч від матриці відстаней. Найменші з них дають значення радіуса .

Вершини 3-тя та 5-та є центральними.

При обчислюванні відстаней поміж вершинами треба розв’язати таку задачу: у зв’язному графі G задано дві вершини – a та *b*; знайти ланцюг найменшої довжини, який зв’язує a з *b*.

Алгоритм розв’язання задачі полягає у послідовному присвоєнні вершинам графа позначок, які дорівнюють кількості ребер найкоротшого ланцюга поміж вершиною, що розглядається, і вершиною a.

Опис алгоритму.

Позначимо вершину *a* як 0. Усі вершини, суміжні з вершиною *a*, позначимо як 1. Непомічені вершини, суміжні з вершинами, які мають позначку 1, позначимо 2; суміжні з ними – 3 т. д., допоки не буде позначено вершину *b*.

Припустімо, що вершина b набула оцінки *k*. Повертаємося від b до *a*, відшукуючи послідовно: суміжну з b вершину , яка має оцінку , суміжну з вершину , яка має оцінку , і т.д. доти, допоки з певної вершини з оцінкою 1 не дістанемось вершини a.

Ланцюг – шуканий, він має довжину *k*.

На прикладі графа, зображеного на рис. 6.25 покажемо оцінки, яких набувають вершини графа у перебігу роботи алгоритму.

Через те, що вершина b набула оцінки 7, довжина найкоротшого ланцюга від a до b дорівнює 7. Цей ланцюг виділений на рис. 6.26.



**Зауваження**. Описаний алгоритм іноді називають *хвильовим*: процес розміщення оцінок нагадує поширення збурювання, яке виникає у вершині a і рухається зі швидкістю одне ребро за одиницю часу.

Якщо існує цикл у скінченному графі, в якому кожне ребро графа брало участь один раз, то такий цикл називається ейлеровим циклом, а граф, який містить такий цикл, – ейлеровим графом.

Теорема 6.5 Скінченний граф G є ейлеревим графом тоді й лише тоді, коли:

* + 1. G – зв’язний;
    2. усі його вершини мають парні степені (при підрахунку степеня вершини, будь-яку інцидентну їй петлю вважати двічі).

**Приклад 6.15** Чи мають графи, зображені на рис. 6.27 (а, б), ейлерів цикл.

Рисунок 6.27

*Розв’язання*. Щоб відповістити на поставлене запитання, порахуємо степені вершин графа:

а) ; ; ; ; .

Степені всіх вершин графа, зображеного на рис. 6.27 а), парні, отже, граф, має ейлерів цикл;

б) ; ; ; ; ; .

Степені вершин 4 і 5 графа, зображеного на рис. 6.27 б), непарні, отже, граф не має ейлерів цикл.

Шлях, що включає кожне ребро графа тільки один раз, називається е*йлеровим шляхом*. У тому випадку, якщо ейлерів шлях не є ейлеровим циклом, він називається *власним ейлеровим шляхом*.

**Теорема 6.6** Граф має власний ейлерів шлях тоді і тільки тоді, коли він зв’язний і рівно дві його вершини мають непарний степінь.

**Приклад 6.16** Граф, зображений на рис. 6.27 б), не має ейлерового циклу, але має власний ейлерів шлях, тому що дві його вершини мають непарний степінь.

Гамільтоновим циклом називається простий цикл, що проходить через усі вершини графа.

*Шляхом Гамільтона* (або *гамільтоновим ланцюгом*) називається простий ланцюг, що проходить через всі вершини графа, з початком і кінцем у різних вершинах і .

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

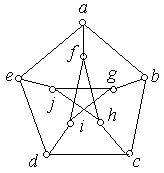
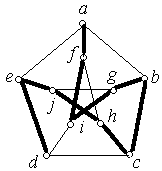
Гамільтонів цикл названо іменем ірландського математика Вільямса Гамільтона, який 1859 року вперше почав вивчати ці питання.

**Зауваження**. Гамільтонів цикл існує далеко не в кожному графі. Більш того, через кожні дві вершини розглянутого графа може проходити простий цикл, а гамільтонів цикл при цьому може бути відсутній:

|  |  |  |
| --- | --- | --- |
|  | | – граф з гамільтоновим циклом; |
|  | – граф, в якому немає гамільтонового циклу. | |

Щоб пройти через вершини *A*, *B*, C, гамільтонів цикл має містити всі ребра, які лежать на сторонах трикутника. Але тоді він не проходить через розміщену в центрі трикутника вершину *O*.

**Приклад 6.17** Граф Петерсона, зображений на рис. 6.28 а), має шлях Гамільтона (рис. 6.28 б)), але не має циклу Гамільтона.

а) б)

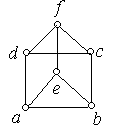
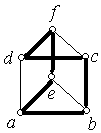
Рисунок 6.28

**Теорема 6.6** Якщо граф має ребро розрізу, то він не може мати цикл Гамільтона. Якщо компоненти графа, отримані шляхом видалення ребра розрізу, мають цикл Гамільтона, то граф має шлях Гамільтона.

**Теорема 6.7** Якщо ‑ зв’язний граф з вершинами і для кожної пари несуміжних вершин , , сума степенів вершин задовольняє умові , тоді граф має цикл Гамільтона.

**Наслідок.** Якщо ‑ зв’язний граф з вершинами і для кожної вершини виконується умова , то граф має цикл Гамільтона.

**Приклад 6.18** Знайдіть цикл Гамільтона, якщо він існує, для графа , зображеного на рис. 6.29, а.

а) б)

Рисунок 6.29

*Розв’язання*. Граф ‑ зв’язний, кількість вершин графа – . Степінь кожної з вершин дорівнює 3, тобто кожна з вершин графа задовольняє умові . Отже, даний граф є гамільтонів, тобто існує цикл Гамільтона. Знайти його можемо тільки методом перебирання. Результати прямого перебирання – цикл (рис. 6.29 ,б).

Розповсюджена інтерпретація задачі щодо гамільтонових циклів полягає в такому. Обід накрито на круглому столі. З-посеред гостей декотрі є друзями. За яких умов можна розсадити всіх в такий спосіб, щоби по обидва боки кожного з присутніх сиділи саме його друзі?

Так звана *задача про комівояжера* (її називають ще задачею про бродячого торговця), є задачею, яка належить до гамільтонових ланцюгів. Район, який має відвідати комівояжер, має певну кількість міст. Відстані поміж ними є відомі. Треба відшукати найкоротший шлях, який проходив би через усі пункти і повертався до вхідного. Ця задача має низку додатків у досліджуванні операцій, наприклад у питаннях щодо найбільш ефективного використовування рухомого складу чи обладнання.

У задачі про комівояжера міста можна подати як вершини графа *G*, в якому кожній парі вершин приписується відстань . Якщо дві вершини не є з’єднані, то можна покласти, що .

Завдання полягає в тому, щоб знайти такий гамільтонів цикл С, для якого сума – мінімальна.

Оскільки зазвичай йдеться лише про скінчену кількість вершин, задачу може бути розв’язано шляхом перебирання. Однак жодного ефективного алгоритму, окрім перебору, не відомо. Є тільки певні окремі схеми для окремих випадків. Один доволі важливий приклад визначення найкоротшої повітряної лінії, яка сполучає всі столиці штатів у США, обчислили до кінця Дагірі, Далкерсон та Джонсон.

Незважаючи на подібність у визначеннях для ейлерових та гамільтонових циклів, у цих поняттях є мало спільного.

Критерій існування ейлерових циклів було встановлено просто; для гамільтонових циклів жодне загальне правило невідоме. Більш того, іноді навіть для конкретних графів буває надто складно вирішити, чи можна знайти такий цикл.

* 1. Цикломатика графів. Дерева

Цикломатика – це вивчення циклів у графі.

З усієї сукупності циклів даного графа можна відокремити цілковито певну кількість незалежних (базисних) циклів, а решту здобути з базисних циклів за допомогою спеціальної операції додавання. Приклад такого додавання на рис. 6.30.



Нехай задано граф . Ребро графа, через яке проходить хоча б один цикл, назвемо цикловим ребром. Ребро, яке не входить до жодного циклу, називатимемо перешийком.

**Приклад 6.19** У графі, зображеному на рис. 6.31, ребра та – перешийки, решта ребер – циклові.



Рисунок 6.31 – Граф з перешийками та цикловими ребрами

Лема. При вилученні зі зв’язного графа циклового ребра він залишається зв’язним. При вилученні зі зв’язного графа перешийка граф розпадається на дві компоненти зв’язності.

*Доведення*.

Якщо ребро , яке вилучається, є циклове, то після його вилучення вершини a та b, як і раніше, можна з’єднати в ланцюг – залишком циклу, до якого входило ребро . Звідси випливає, що і кожні дві вершини графа після вилучення ребра залишаються зв’язаними ланцюгом.

Й навпаки, якщо – перешийок, то після його вилучення вершини a та b не можна зв’язати ланцюгом, інакше цей ланцюг разом з ребром утворить цикл у вихідному графі. З іншого боку, кожна вершина залишається зв’язаною чи то з вершиною *a*, чи то з вершиною *b*.

**Наслідок**. При вилученні з графа циклового ребра кількість зв’язних компонентів графа не змінюється. При вилученні перешийка кількість зв’язних компонентів графа збільшується на одиницю.

Нехай задано граф , , ; – кількість компонент зв’язності графа. Величина називається цикломатичним числом графа.

Теорема **6.8** Для кожного графа цикломатичне число є невід’ємне, тобто .

*Доведення*.

Вилучаємо з графа по одному ребру, і відстежуємо змінювання величини . Параметри вихідного графа позначимо m, n і . Ті самі величини після вилучення ребра позначимо через , та .

У процесі вилучення ребер можливі два випадки:

а) вилучуване ребро – циклове. Тоді , , ;

.

б) вилучуване ребро – перешийок. У такому разі

, , .

Тоді .

Отже, при вилученні ребра величина або не змінюється, або зменшується на одиницю. Після вилучення всіх ребер дістанемо порожній граф, у якому , , , тобто .

Отже, у вхідного графа .

**Зауваження**. Доведення теореми свідчить про зв’язок цикломатичного числа з наявністю циклів у графі.

Якщо , то у графі є принаймні один цикл. При вилученні циклового ребра деякі цикли розриваються – і це призводить до зменшення . Якщо продовжувати вилучення ребер, то, зрештою, розриваються всі цикли – і стає рівним 0. Після цього вже не змінюється, тому що всі ребра стали перешийками.

**Приклад 6.20** Нехай треба зв’язати кілька населених пунктів мережею доріг або телефонною мережею; взагалі яким-небудь чином зв’язати один пункт з одним. Пропонований проект подано на рис. 6.32, а).



Надалі виникла потреба здешевити проект, внаслідок чого певні зв’язки було вилучено (рис. 6.32 б, в).

У графі на рис. 6.32, в вже жодного зв’язку вилучати не можна, в противному разі граф перестане бути зв’язним і поставлене завдання не буде розв’язано.

Такого роду графи називаються деревами.

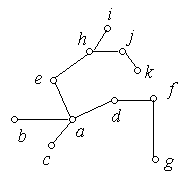
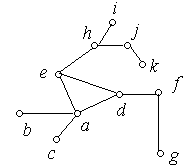
*Неорієнтованим деревом* (або просто *деревом*) називається неорієнтований зв’язний граф без циклів, отже без петель і кратних ребер, що має не менш двох вершин.

Дерево є мінімальним зв’язним графом у тому розумінні, що видалення хоча б одного ребра приводить до того, що граф виявляється незв’язним. Отже, маємо інше означення. Деревом називається зв’язний граф без циклів.

Дерево не може мати ані кратних ребер, ані петель, ані ізольованих вершин. Кожний ланцюг у дереві є простий, через те що в іншому разі дерево містило б цикл, що є неможливе.

При додаванні до дерева ребра утворюється цикл, а при вилученні хоча б одного ребра дерево розпадається на компоненти, кожна з яких є або деревом, або ізольованою вершиною.

**Приклад 6.21** Чи є графи, зображені на рис.6.33 а), б), деревами?

а) б)

Рисунок 6.33

*Розв’язання*. Граф на рис. 6.33 а) є деревом, тому що він зв’язний і не містить циклів. Граф на рис. 6.33 б) не є деревом, тому що містить цикл .

Дерева називаються істотно різними, якщо вони не є ізоморфні.

На рис. 6.34 показано усі можливі дерева з шістьма вершинами – неізоморфні.



У поданій нижче теоремі перелічуються властивості дерев, кожна з яких повністю характеризує дерево.

Теорема **6.9** Еквівалентними є такі означення дерева:

а) дерево – це зв’язний граф без циклів;

б) дерево – це зв’язний граф, у якому кожне ребро є перешийком;

в) дерево – це зв’язний граф, цикломатичне число якого дорівнює нулю;

г) дерево – це граф, у якому для кожних двох вершин існує лише один з’єднувальний ланцюг.

Наведені твердження нескладно виводяться одне з іншого за схемою а) => б) => в) => г) => а).

У властивості в) маємо: або , тобто у кожному дереві кількість ребер є на одиницю менша за кількість вершин.

**Зауваження**. Рис. 6.35 певним чином пояснює назву «дерево». Відокремимо в дереві певну вершину а (корінь). Оскільки ребра, які прилягають до неї, – перешийки, то після вилучання кожного з них від дерева відокремлюється компонента зв’язності графа. На рисунку вони позначені , , ..., . Кожна з цих компонентів також є деревом.



Вилучання з довільного зв’язного графа всіх циклових ребер дає в наслідок дерево таке, що:

1. , тобто множина вершин дерева T збігається з множиною вершин графа G;
2. , тобто кожне ребро дерева є водночас ребром графа *G*. Кожне дерево *T*, яке задовольняє умовам 1) та 2) називається кістяковим *деревом* графа G (рис. 6.36).

**Зауваження**. Через те, що вилучання циклових ребер можна провадити у різні способи, то один і той самий граф має багато кістякових дерев.

Нехай у графі G виокремлено кістякове дерево T. Ребра графа, які не ввійшли до *T*, називатимемо хордами.

Теорема **6.10** Якими б не були кістякове дерево і хорда u цього дерева, у графі G існує єдиний цикл, який має хорду u і не має інших хорд.

*Доведення*.

Нехай . У дереві T є єдиний ланцюг, який з’єднує вершини а та b. Долучаючи до цього ланцюга ребро *u*, здобуваємо потрібний цикл.



*Лісом* називається незв’язний неорієнтований граф без циклів. Зв’язані компоненти лісу є деревами.

**Приклад 6.22** На рис. 6.37 наведений приклад лісу, що містить три дерева.

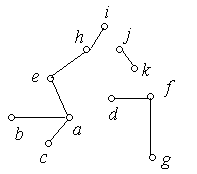


Рисунок 6.37

Добре зрозумілим прикладом дерева є: будь-яке генеалогічне дерево; організаційна структура будь-якого підприємства, організації.

*Орієнтованим деревом* називається вільний від петель орієнтований граф, співвіднесений граф якого є деревом; тому якщо існує шлях від вершини до вершини , то він єдиний.

Помітимо, що якщо в орієнтованому дереві є ребро , то немає ребра , інакше шлях був би циклом.

**Приклад 6.23** На рис. 6.38 наведено приклад орієнтованого дерева:

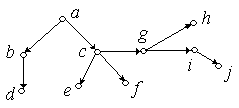


Рисунок 6.38

Вершини дерева степеня 1 називаються *листям*, інші вершини – *внутрішніми вершинами*.

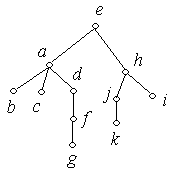
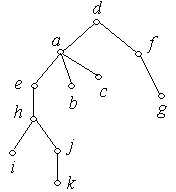
Наприклад, у дерева, зображеного на рис. 6.38 а), листя це вершини . Інші вершини є внутрішніми.

Довільний незв’язний граф, який не містить циклів, називається лісом.

Еквівалентне визначення лісу: граф *G*, усі компоненти зв’язності якого є деревами, називається лісом.

Дерева визначаються такою властивістю: для будь-яких двох вершин дерева шлях з вершини до вершини єдиний. І, навпаки, якщо для будь-яких двох вершин графа існує єдиний шлях з вершини до вершини , тоді граф ‑ дерево.

Припустимо, що дерево являє собою якийсь фізичний об’єкт, рухливий у вершинах. Його можна підвісити за кожну з вершин. Так, наприклад, якщо дерево, зображене на рис. 6.33 а) підвісити за вершину , або , то воно буде виглядати, як показано на рис. 6.39 а) або на рис. 6.39 б).

а) б)

Рисунок 6.39

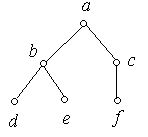
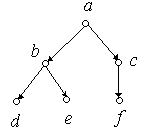
Обрана нами вершина називається *коренем* дерева і розташовується в самій верхній його частині. Для дерева на рис. 6.39 а) коренем є вершина , а для дерева на рис. 6.30 б) – вершина .

Дерево, корінь якого визначений, називається *кореневим деревом*.

Аналогічно визначається і *орієнтоване кореневе дерево*. При цьому варто пам’ятати, що всі його ребра орієнтуються від кореня.

При заміні кореневого неорієнтованого дерева на кореневе орієнтоване дерево , говорять, що є породженим кореневим деревом .

**Приклад 6.24** На рис. 6.40 а) зображене неорієнтоване кореневе дерево, а на рис. 6.40 б) – породжене їм орієнтоване кореневе дерево.

а) б)

Рисунок 6.40

Якщо корінь дерева обраний, то можна ввести ще ряд визначень.

*Рівнем вершини* називається довжина єдиного шляху від кореня дерева до вершини .

*Висотою дерева* називається довжина самого довгого шляху від кореня дерева до листя.

**Приклад 6.25** Для кореневого дерева, зображеного на рис. 6.41, визначити рівень вершини , висоту дерева.

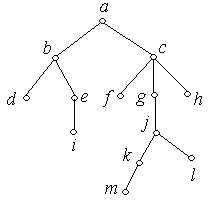


Рисунок 6.41

*Розв’язання*. Рівень вершини дорівнює двом; а висота дерева з коренем у вершині дорівнює максимальному шляху від кореня до вершини ‑ ‑дорівнює п’яти.

Задача про найкоротшу з’єднувальну мережу.

Нехай – навантажений граф, у якого задано вагу кожного ребра . Треба побудувати кістякове дерево T графа *G*, сума ваги ребер якого є мінімальна.

Цій задачі можна дати таку інтерпретацію: пунктів на місцевості треба зв’язати мережею доріг чи то ліній телефонного зв’язку. Для кожної пари пунктів і та j задано вартість їхнього з’єднання – це і є «вага» ребра у даному разі.

Треба побудувати з’єднувальну мережу мінімальної вартості. Така мережа буде деревом, і при цьому серед усіх кістякових дерев вона матиме мінімальну суму довжин ребер, які входять до неї.

**Алгоритм розв’язання поставленої задачі** є надто простий. Дерево будується покроково; на кожному кроці долучається одне ребро.

Правило для вибору цього ребра є таке: серед ще не обраних ребер береться найкоротше, яке не утворює циклу з вже обраними ребрами.

**Приклад 6.26** У матриці C елемент, що стоїть в *і*-му рядку та j-му стовпчику, зазначає в умовних одиницях витрати, потрібні для того, щоб зв’язати пункт і з пунктом j:

.

Задача полягає в тому, щоб з найменшими витратами з’єднати всі пункти один з одним.

Застосування сформульованого вище алгоритму реалізовується так.

У матриці C відшукується мінімальний ненульовий елемент. Він вилучається з матриці, а відповідне йому ребро долучається до мережі, якщо при цьому не утвориться циклів.

Ці дії повторюються.

Отже, на перших п’яти кроках роботи алгоритму буде обрано ребра , , , та . Вони становитимуть частину мережі, подану на рис. 6.42, a). З ребер, що залишилися, мінімальну довжину має ребро , але до мережі воно не долучається, тому що утворює цикл з вже обраними ребрами.



На наступних етапах алгоритму до дерева буде долучено ребра , та . Побудовано дерево подано на рис. 6.42, б).

ВИКОРИСТАНА ЛІТЕРАТУРА

1. Дискретна математика. Навчальний посібник. Ужгород: ПП «АУТДОРШАРК», 2021. 124 с. URL : [http://files.znu.edu.ua/files/Bibliobooks/ Inshi70/0051143.pdf](http://files.znu.edu.ua/files/Bibliobooks/%20Inshi70/0051143.pdf)
2. Дискретна математика : навчальний посібник для здобувачів ступеня вищої освіти бакалавра освітньо-професійних програм «Середня освіта (Математика)», «Середня освіта (Інформатика)», «Математика», «Комп’ютерна математика», «Комп’ютерне моделювання», «Інформаційні системи та технології» / П. Г. Стєганцева та ін. Запоріжжя : ЗНУ, 2021. 177 с. URL : <http://files.znu.edu.ua/files/metodychky/2021/01/0045777.pdf>
3. Комп’ютерна дискретна математика (Частина 1): Розрахункова робота : навч. посіб. для здобувачів ступеня бакалавра за освітньою програмою “Інженерія програмного забезпечення розподілених систем” спеціальності 121 “Інженерія програмного забезпечення” / Л. І. Кублій; КПІ ім. Ігоря Сікорського. Київ: КПІ ім. Ігоря Сікорського, 2020. 165 с. URL : <http://files.znu.edu.ua/> files/Bibliobooks/Inshi70/0051140.pdf
4. Шевченко Г.В., Шкапа В.В. Дискретна математика. Навчально-методичнтй посібник. Київ : ДУТ, 2018. 158 с. URL : <http://files.znu.edu.ua/files/> Bibliobooks/Inshi59/0043304.pdf

Навчальне видання

(*українською мовою*)

Спиця Оксана Геннадіївна

Кривохата Анастасія Григорівна

**ДИСКРЕТНІ СТРУКТУРИ**

**Навчальний посібник**

**для здобувачів ступеня вищої освіти бакалавра  
спеціальності 121 Інженерія програмного забезпечення   
освітньої програми Програмна інженерія**

Рецензент *С. М. Гребенюк*

Відповідальний за випуск *І. В. Зіновєєв*

Коректор *О. Г. Спиця*