Н.А. Гук

# ПОСІБНИК ДО ВИВЧЕННЯ ДИСЦИПЛІНИ «ДИСКРЕТНА МАТЕМАТИКА»

2014

**Міністерство освіти і науки України**
**Дніпропетровський національний університет**
**ім. Олеся Гончара**

**Н.А. Гук**

# ПОСІБНИК ДО ВИВЧЕННЯ ДИСЦИПЛІНИ «ДИСКРЕТНА МАТЕМАТИКА»

**Дніпропетровськ**
**РВВ ДНУ**
**2014**

УДК 007:159.955
Г 21

Рецензенти : д-р фіз.-мат. наук, проф. О.М. Кісельова
д-р фіз.-мат. наук, проф. В.Б. Говоруха

Г 21 Гук, Н.А. Посібник до вивчення дисципліни «Дискретна математика» [Текст] / Н.А. Гук. – Д.: РВВ ДНУ, 2014. – 48 с.

У посібнику розглянуто основні відомості з курсу «Дискретна математика», а саме теорія множин і відношень, булеві функції, функціональна повнота систем булевих функцій, які набули широкого теоретичного і практичного застосування. Наведено приклади роботи із структурами даних для зображення множин, відношень, функцій у пам'яті обчислювальних машин та алгоритми для здійснення операцій над ними. Матеріал подано у доступній формі з великою кількістю прикладів, що дозволяє краще засвоювати матеріал. Кожен розділ містить задачі для самостійного розв'язування.

Для студентів першого курсу ДНУ галузі знань «Системний аналіз», «Прикладна математика», «Комп'ютерні науки», «Інформатика», які навчаються за спеціалізаціями «Системний аналіз», «Прикладна математика», «Комп'ютерна інженерія». Матеріалами посібника можуть також послуговуватися аспіранти і фахівці, які застосовують методи дискретної математики для розв'язування практичних задач.

# ПЕРЕДМОВА

Дискретна математика є відносно молодою наукою, поширений інтерес до якої в даний час пов'язаний із застосуванням засобів обчислювальної техніки та інформаційних технологій у всіх сферах людської діяльності. Дискретна математика вивчає дискретні об'єкти, їх властивості і зв'язки між ними за допомогою дискретних величин. Дискретні структури – фундаментальна основа програмної інженерії, їх широко використовують у дослідженнях з логіки, аналізу структур даних, економіки, квантової механіки, хімії, електро- та радіотехніки, лінгвістики, в теорії алгоритмів, теорії розкладів, теорії баз даних і знань, теорії ігор, теорії ймовірності, в програмуванні, під час розв'язування задач оптимізації, організації транспортних мереж і потоків та в інших розділах інформатики.

Сьогодні дискретна математика є важливою ланкою математичної освіти, фундаментом для вивчення практично всіх спеціальних курсів, які викладають для напрямів «Системний аналіз», «Комп'ютерні науки», «Програмна інженерія», «Інформатика». Дискретна математика включає такі вже сформовані розділи математики, як теорія множин, математична логіка, теорія алгебричних систем, комбінаторика, а також нові розділи, які найбільш інтенсивно стали розвиватися в середині XX сторіччя в рамках науково-технічного прогресу і масового використання ЕОМ: теорія графів, теорія алгоритмів, теорія кодування, теорія скінченних автоматів та ін.

Посібник знайомить студентів з найважливішими розділами дискретної математики і висвітлює такі поняття, як множина, функція, відображення, відношення, булеві функції, повнота системи булевих функцій. У ньому наведено приклади застосування структур даних для зображення множин, відношень, функцій у пам'яті обчислювальних машин та алгоритми для здійснення операцій над ними.

Розглянуті в посібнику поняття проілюстровано необхідною кількістю прикладів. Кожен розділ вміщує ретельно дібрані завдання для самостійної роботи.

# 1. ТЕОРІЯ МНОЖИН

## 1.1. Поняття множини

***Означення* 1.1.** ***Множиною*** є будь-яка визначена сукупність об'єктів, об'єднаних за певною ознакою чи властивістю. Об'єкти, з яких складена множина, мають назву елементів множини, елементи мають бути різними між собою.

***Означення* 1.2.** Якщо $a$ – один з об'єктів множини $A$, то говорять, що $a$ – *елемент множини* $A$ або $a$ *належить* множині $A$.

Множини позначають прописними літерами латинського алфавіту $A, B, C, \ldots$, а елементи множини – рядковими літерами латинського алфавіту $a, b, c, \ldots$ Якщо елемент $a$ належить множині $A$, то пишуть $a \in A$. Якщо $a$ не є елементом множини $A$, то пишуть $a \notin A$.

***Означення* 1.3.** Множину, що не містить елементів, називають *порожньою множиною* і позначають $\varnothing$. Порожня множина є підмножиною будь-якої множини.

Зазвичай здійснюючи конкретні міркування, елементи всіх множин вибирають з деякої однієї досить широкої множини, яка має назву *універсальної множини* або *універсума*.

Множину, яка містить скінченну кількість елементів, називають *скінченною,* у протилежному випадку – *нескінченною.*

***Означення* 1.4.** Кількість елементів у скінченній множині $A$ називають *потужністю* множини $A$ і позначають $|A|$.

## 1.2. Способи задання множин

**1. *Перерахування елементів.*** Множину задають списком елементів множини. Такий спосіб задання прийнятний тільки для опису скінченних множин. Список подають у фігурних дужках, наприклад, множину з перших п'яти парних додатних чисел зображують у вигляді

$$A = \{2,4,6,8,10\}.$$

**2. *Опис властивостей елементів множини.*** Множина складається з елементів, які мають певні властивості. Наприклад, для опису множини відмінників групи студентів використовують зображення

$$M = \{m | \quad m - \textit{відмінник групи}\}.$$

**3. *Породна процедура*.** Процедура описує спосіб генерування елементів множини із уже отриманих елементів множини або з інших об'єктів. Елементами множини є усі об'єкти, які можна побудувати за допомогою цієї процедури. Наприклад, множину усіх цілих чисел, що є ступенями двійки $M_{2^n}, n \in N$, де $N$ - множина натуральних чисел, можна побудувати за допомогою породної процедури, заданої двома правилами, названими рекурсивними:

$$\text{а)} 1 \in M_{2^n}; \qquad \text{б) якщо } m \in M_{2^n}, \text{ тоді } 2m \in M_{2^n}.$$

**4. *Характеристичний предикат.*** Множину задають списком елементів, властивості яких задовольняють умову, сформульовану у логічній функції-предикат.

Функція-предикат $P(x)$ – це логічна функція, яку описують у формі логічного твердження або процедури, вона набуває логічного значення 1, якщо логічна умова є істинна, або 0, якщо логічна умова є хибна. Множину $M$, яка складається з елементів $m$ таких, що задовольняють властивості $P(m)$, позначають у такий спосіб:

$$M = \{m| \quad P(m)\}.$$

Наприклад, розглянемо множину $M$ з натуральних чисел, менших за 5. Для опису такої множини побудуємо функцію-предикат $P(m): - m \in N, \quad m < 5$, яка набуває значення 1, якщо елемент $m$ належить множині натуральних чисел $N$ та виконується нерівність $m < 5$. Тоді елементи $m$, які задовольняють ці умови, належать множині $M$.

**Завдання для самостійної роботи**

1. Опишіть різними способами:
   - множину натуральних чисел, кратних 3 і таких, що не перевищують значення 100;
   - множину $\{x| \quad x - \textit{додатне непарне ціле число},\ \mathrm{x} < 35\}$;
   - множину $\{x| \quad x - \textit{ціле число}, \quad x^3 < 100\}$;
   - множину $\{x| \quad x - \textit{список студентів вашої групи}\}$;
   - множину $\{4, 8, 12, 16, 20, 24, 28, 32, 36, 40\}$.
2. Зобразіть множину $\{\textit{березень, квітень, травень}\}$, вдаючись до опису властивостей елементів множини.
3. Опишіть множину $\{1, 5, 25, 125, 625, 3125\}$ із застосуванням породної процедури.

## 1.3. Операції над множинами

***Означення* 1.5.** Множину $A$ називають ***підмножиною*** (або ***включенням***) множини $B$ $(A \subseteq B)$, якщо кожен елемент множини $A$ належить множині $B$, тобто якщо $x \in A$, то $x \in B$.

Якщо $A \subseteq B$ й $A \neq B$, то $A$ називають власною підмножиною множини $B$ й позначають $A \subset B$.

***Означення* 1.6.** Дві множини тотожні $(A = B)$, якщо вони є підмножинами один одного, тобто $A \subseteq B$ і $B \subseteq A$.

***Означення* 1.7.** *Об'єднанням* множин $A$ і $B$ називають множину із усіх тих і тільки тих елементів, які належать хоча б одній із множин $A$ або $B$. Об'єднання множин $A$ і $B$ позначають $A \cup B$. Це означення еквівалентне:

$$A \cup B = \{x| \quad x \in A \textit{ або } x \in B\}.$$

Наприклад, якщо $A = \{2, 3, 5, 6, 7\}$, $B = \{1, 2, 3, 7, 9\}$, то $A \cup B = \{1, 2, 3, 5, 6, 7, 9\}$.

***Означення* 1.8.** *Перерізом* множин $A$ і $B$ називають множину із усіх тих і тільки тих елементів, які належать як множині $A$, так і множині $B$. Переріз множин $A$ і $B$ позначають $A \cap B$. Це означення еквівалентне:

$$A \cap B = \{x \mid x \in A \ \ ma \ \ x \in B\}.$$

Наприклад, якщо $A = \{2, 3, 5, 6, 7\}$, $B = \{1, 2, 3, 7, 9\}$, то $A \cap B = \{2, 3, 7\}$.

***Означення* 1.9.** *Різницею* множин $A$ і $B$ називають множину із усіх тих і тільки тих елементів множини $A$, які не належать множині $B$. Різницю множин $A$ і $B$ позначають $A \setminus B$. Це означення еквівалентне:

$$A \setminus B = \{x \mid x \in A \ \ ma \ \ x \notin B\}.$$

Наприклад, якщо $A = \{2, 3, 5, 6, 7\}$, $B = \{1, 2, 3, 7, 9\}$, то $A \setminus B = \{5, 6\}$.

***Означення* 1.10.** Універсальною множиною (або універсумом) називають множину $U$, елементами якої є всі елементи, розглядувані в рамках конкретної постановки задачі. Універсум ще називають множиною всіх множин.

Наприклад, для елементарної арифметики універсумом є множина всіх раціональних чисел.

***Означення* 1.11.** *Доповненням* множини $A$ (позначають $\overline{A}$) називають множину із усіх тих і тільки тих елементів універсальної множини, які не належать множині $A$. Це означення еквівалентне:

$$\overline{A} = U \setminus A = \{x \mid x \in U \ \ ma \ \ x \notin A\}.$$

***Означення* 1.12.** *Симетричною різницею* множин $A$ і $B$ (позначають $A \Delta B$) називають множину з об'єднання елементів, які належать множині $A$ і не належать множині $B$, і елементів, які належать множині $B$ і не належать множині $A$. Це означення еквівалентне:

$$A \Delta B = (A \setminus B) \cup (B \setminus A).$$

Наприклад, якщо $A = \{2, 3, 5, 6, 7\}$, $B = \{1, 2, 3, 7, 9\}$, то $A \Delta B = \{1, 5, 6, 9\}$.

Кількість аргументів у операції визначає її арність, тому операції об'єднання, перерізу, різниці та симетричної різниці є бінарні, а операція доповнення – унарна.

### 1.4. Властивості операцій над множинами

***Теорема* 1.1.** Для будь-яких підмножин $A, B, C$ універсальної множини $U$ справедливим є таке:

*а) закони ідемпотентності* $A \cup A = A$; $A \cap A = A$;

*б) властивості комутативності* $A \cup B = B \cup A$; $A \cap B = B \cap A$;

*в) властивості асоціативності*

$$A \cup (B \cup C) = (A \cup B) \cup C; \qquad A \cap (B \cap C) = (A \cap B) \cap C;$$

*г) властивості дистрибутивності*

$$A \cup (B \cap C) = (A \cap B) \cup (A \cap C); \qquad A \cap (B \cup C) = (A \cap B) \cup (A \cup C);$$

*д) закон інволюції* $\overline{\left(\overline{A}\right)} = A$;

*е) закони де Моргана* $\overline{(A \cup B)} = \overline{A} \cap \overline{B}$; $\overline{(A \cap B)} = \overline{A} \cup \overline{B}$;

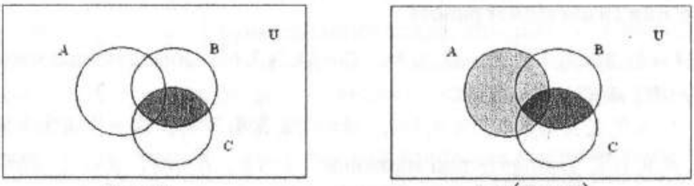
*є) властивості тотожності* $A \cup \varnothing = A$; $A \cap U = A$;

*ж) властивості доповнення* $A \cup \overline{A} = U$; $A \cap \overline{A} = \varnothing$.

Для доведення цих та інших тотожностей можна використовувати такі способи.

**1. Графічне зображення результатів виконання операцій за допомогою діаграм Ейлера-В'єна.**

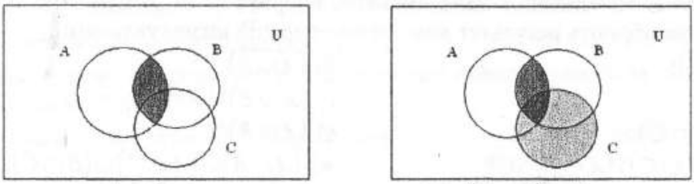
Діаграма Ейлера-В'єна – це зображення множини у вигляді геометричної фігури, наприклад кола. При цьому універсальну множину зображують у вигляді прямокутника. Доведемо, що $A\cap(B\cap C)=(A\cap B)\cap C$ (результати графічного зображення операцій лівої частини тотожності зображено на рис. 1.1, правої – на рис. 1.2):




$B\cap C$ $\qquad$ $A\cap(B\cap C)$

Рис. 1.1

Аналіз результатів виконання операцій над множинами свідчить, що графічні зображення лівої та правої частин тотожності збігаються, а отже, тотожність $A\cap(B\cap C)=(A\cap B)\cap C$ доведено.




$A\cap B$ $\qquad$ $(A\cap B)\cap C$

Рис. 1.2

**2. Формальні міркування**. До формальних міркувань вдаються з метою довести, що довільний елемент множини з лівої частини тотожності належить також і правій частині тотожності, та навпаки.

Наприклад, доведемо закон де Моргана $\overline{(A\cap B)}=\overline{A}\cup\overline{B}$. Нехай $x\in\overline{(A\cap B)}\Leftrightarrow x\notin(A\cap B)\Leftrightarrow x\notin A$, або $x\notin B\Leftrightarrow x\in\overline{A}$, або $x\in\overline{B}\Leftrightarrow x\in\overline{A}\cup\overline{B}$.

Доведемо тотожність $(A\cup B)\cap C=(A\cap C)\cup(B\cap C)$. Нехай $x\in(A\cup B)\cap C\Rightarrow x\in A\cup B$ та $x\in C\Leftrightarrow(x\in A$ або $x\in B)$ і $x\in C\Leftrightarrow(x\in A$ та $x\in C)$, або $(x\in B$ та $x\in C)\Leftrightarrow x\in A\cap C$, або $x\in B\cap C\Leftrightarrow x\in(A\cap C)\cup(B\cap C)$, тобто $(A\cup B)\cap C\subseteq(A\cap C)\cup(B\cap C)$ та $(A\cap C)\cup(B\cap C)\subseteq(A\cap B)\cap C$, тоді маємо $(A\cup B)\cap C=(A\cap C)\cup(B\cap C)$.

**3. Перетворення лівої та правої частин тотожності до виразів однакового вигляду.** Застосовуючи такий підхід, ліву та праву частини тотожності за допомогою перетворень зводимо до виразів однакового вигляду.

Наприклад, доведемо тотожність $((A\setminus B)\cup(A\cap B))\cup B=\overline{(\overline{A}\cup B)\setminus B}$.

Перетворимо ліву частину: $((A\setminus B)\cup(A\cap B))\cup B=$

$=((A\cap\overline{B})\cup(A\cap B))\cup B(A\cap(B\cup\overline{B}))\cup B=(A\cap U)\cup B=A\cup B$.

Відтак перетворимо праву частину: $\overline{(\overline{A}\cup B)\setminus B}=\overline{(\overline{A}\cup B)\cap\overline{B}}=$

$=(A\cap\overline{B})\cup B=(A\cup B)\cap(\overline{B}\cup B)=(A\cup B)\cap U=(A\cup B)$.

Отримаємо: $A\cup B=A\cup B$. Тотожність доведена.

**Завдання для самостійної роботи**

1. Дано $A=\{0,3,5,6,9\}$, $B=\{2,3,9\}$, $C=\{0,2,3,6\}$. Визначте такі множини: $B\setminus C$, $A\cap B$, $A\Delta C$, $(A\cup B)\setminus C$.

2. Дано $U=\{1,2,3,4,5,6,7,8,9,10\}$, $A=\{1,2,3,4,5,6\}$, $B=\{1,3,5,7,9\}$, $C=\{2,4,6,8,10\}$. Визначте такі множини: $A\cap B$, $B\cup C$, $A\setminus C$, $A\Delta B$, $\overline{C}$, $A\cup(B\cap C)$, $(A\cup B)\cap C$, $\overline{(A\cap B)}$, $\overline{A}\cap\overline{B}$.

3. Визначте, які з наведених тверджень істинні, а які хибні:

а) $A\cap\varnothing=A$; б) $A\cup\varnothing=A$; б) якщо $A\subseteq B$, то $A\cap B=A$;

в) якщо $A\cap B=A$, то $B\subseteq A$; г) якщо $A\subseteq B$, то $A\cup B=A$;

д) якщо $A\cup B=A$, то $B\subseteq A$.

4. Кожну з наведених нижче множин побудуйте за допомогою діаграми Ейлера-В'єна та зобразіть результат виконання операцій штрихуванням:

а) $\overline{(A\cap B)}$; б) $\overline{(A\cup B)}$;

в) $B\setminus\overline{A}$; г) $(A\cup B)\setminus(A\cap B)$;

д) $A\setminus(B\cap C)$; е) $(A\cap B)\Delta C$;

є) $(A\cup B\cup C)\setminus(A\cap B\cap C)$; ж) $(A\cap B)\cup(A\cap C)\cup(B\cap C)$.

## 1.5. Булеан

***Означення* 1.13.** Множину всіх підмножин з елементів множини $A$ називають *булеаном* $P(A)$.

Наприклад, булеан множини $A=\{a,b,c,d\}$ можна зобразити у такий спосіб: $P(A)=\{\varnothing,\{a\},\{b\},\{c\},\{d\},\{a,b\},\{a,c\},\{a,d\},\{b,c\},\{b,d\},\{c,d\},\{a,b,c\},$ $\{a,b,d\},\{a,c,d\},\{b,c,d\},\{a,b,c,d\}\}$, його потужність дорівнює $|P(A)|=16$.

## 1.6. Способи зображення множин в електронних обчислювальних машинах

Задати зображення будь-якого об'єкта предметної області в пам'яті електронної обчислювальної машини (ЕОМ) означає описати в термінах застосовуваної системи програмування структуру даних, яку використовують для зберігання інформації про об'єкт, та побудувати алгоритми, які реалізують операції над об'єктом.

Один і той же об'єкт можна зобразити різними способами, причому залежно від ситуації одне із зображень може бути більш ефективним, ніж інші. Вибір способу зображення об'єкта залежить від цілого ряду чинників: властивостей об'єкта, складу і відносної частоти використовування операцій над об'єктом.

Для скінченного універсуму, потужність якого не перевершує розрядності машинного слова, універсум та його підмножини зображують за допомогою бітової шкали (машинного слова, коду). Бітова шкала, застосовувана для зображення універсуму, має довжину, яка дорівнює вимірності універсаму, та складається з одиниць. Бітова шкала, яку застосовують для зображення підмножини універсуму, також має довжину, тотожну вимірності універсуму, та складається з одиниць, кількість яких визначають потужністю цієї підмножини. Якщо елемент універсуму належить підмножині, то в бітовій шкалі на відповідному місці розташовується одиниця, інакше – нуль. Відтак операції над підмножинами здійснюють за допомогою порозрядних логічних операцій.

Нехай заданий скінченний універсум $U=\{u_1,u_2,u_3,...,u_n\}$ з числом елементів $n$, що не перевершує розрядності ЕОМ, тоді для зображення підмножини $A_i$ застосовують бітову шкалу $C$, елементи якої будують за таким правилом:

$$C[j]=\begin{cases}1, & u_j \in A_i;\\ 0, & u_j \notin A_i\end{cases},\ j=\overline{1,n},\ i \in N.$$

Наприклад, якщо $U=\{1,2,3,4,5,6\}$, $A=\{2,3,6\}$, то множини $U$ і $A$ можна зобразити у вигляді бітових шкал таким чином:

$U=\{1,2,3,4,5,6\}$:

Таблиця 1.1

| $C[1]$ | $C[2]$ | $C[3]$ | $C[4]$ | $C[5]$ | $C[6]$ |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 |

$A=\{2,3,6\}$:

Таблиця 1.2

| $C[1]$ | $C[2]$ | $C[3]$ | $C[4]$ | $C[5]$ | $C[6]$ |
|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 0 | 1 |

## 1.7. Реалізація операцій над підмножинами універсуму, зображеними за допомогою бітових шкал

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 до множини*** $A$ $(\overline{A})$ зображують кодом, який будують інвертуванням коду множини $A$.

***Об'єднання множин*** $A$ і $B$ $(A \cup B)$ зображують кодом, який будують порозрядним логічним додаванням кодів множин $A$ і $B$.

***Переріз множин*** $A$ і $B$ $(A \cap B)$ зображують кодом, який будують порозрядним логічним множенням кодів множин $A$ і $B$.

Наприклад, для множин $A$, $B$ та $U$, описуваних у вигляді бітових шкал, результати виконання операцій над множинами можна зобразити у такий спосіб:

Таблиця 1.3

| | $C[1]$ | $C[2]$ | $C[3]$ | $C[4]$ | $C[5]$ | $C[6]$ |
|---|---|---|---|---|---|---|
| $U = \{1,2,3,4,5,6\}$ | 1 | 1 | 1 | 1 | 1 | 1 |
| $A = \{2,3,6\}$ | 0 | 1 | 1 | 0 | 0 | 1 |
| $B = \{1,5,6\}$ | 1 | 0 | 0 | 0 | 1 | 1 |
| $\overline{A}$ | 1 | 0 | 0 | 1 | 1 | 0 |
| $A \cup B$ | 1 | 1 | 1 | 0 | 1 | 1 |
| $A \cap B$ | 0 | 0 | 0 | 0 | 0 | 1 |

## 1.8. Генерація усіх підмножин універсуму

У процесі розв'язання практичних задач часом буває необхідно розглянути всі підмножини деякої множини, яка складається з $n$ елементів.

Для опису всіх підмножин множини будемо послуговуватися зображенням цілих чисел у вигляді кодів у двійковій системі числення. Число «0» зображує порожню множину, число 1 – підмножину з одного елемента, число $(2^n - 1)$ – підмножину, збіжну з $n$ - елементною множиною. Для зображення двійкового коду числа $2^n - 1$ застосовують послідовність, складену з $n$ одиниць.

Алгоритм генерації усіх підмножин деякої множини полягає у видаванні двійкових кодів цілих чисел, його зображують у такий спосіб:

*Вхід: $n$ – потужність множини, $n \geq 0$*
*Вихід: послідовність двійкових кодів підмножин*

```
for i from 0 to 2^n −1 do
    write двійковий код числа i
end for
```

Алгоритм генерує двійкові коди $2^n$ різних цілих чисел, тобто всі підмножини будуються тільки по одному разу. Зі збільшенням числа $i$ збільшується кількість розрядів, що необхідна для його зображення.

Недоліком такого способу генерації є відмінність складу кожної новоствореної множини від попередньої, наприклад, слідом за підмножиною з кодом 0111 буде створено підмножину з кодом 1000.

### 1.9. Алгоритм побудови бінарного коду Грея

Для усунення вищезазначеного недоліку застосовують алгоритм побудови бінарного коду Грея, за допомогою якого підмножини генеруються так, що кожна наступна підмножина утворюється з попередньо створеної множини шляхом видалення або додавання одного елемента. Під час розв'язання задач перебору цей підхід дає змогу використовувати результати, отримані на попередньому кроці, для аналізу наступного стану. Це суттєво скорочує час на виконання обчислень.

На вхід алгоритму подається число $n$, яке задає потужність множини $B$. На виході отримуємо послідовність кодів підмножин множини $B$. Для зберігання кодів підмножин використовуємо масив розмірності $n$, складений із нулів та одиниць. Алгоритм побудови бінарного коду Грея зображують у такий спосіб:

```
bool* B;                                  //Масив для зберігання кодів

int main()
{                                         // Головна функція
  int n;
  cout << "n = ";
  cin >> n;                               //Отримання потужності множини
  B = new bool[n];
  for(int i=0; i<n; i++) B[i] = 0;//Генерація коду порожньої множини
  PrintB(n);
  long long t = pow(2,n)-1;
  for(long long i = 1; i<=t; i++)
  {     int p = Q(i);
   B[p-1] = 1-B[p-1];     //Отримання коду наступної множини шляхом
                          додавання або видалення елемента на p-1 місце
       PrintB(n);                   //Генерування наступної множини
  }
  delete B;
}
void PrintB(int n)     //Функція виведення двійкового коду підмножини
{
       for(int i = n-1; i>=0; i--)        // n - потужність множини
       cout << B[i];
       cout << endl;
}
  int Q(int i)                          //Функція для визначення кількості
                                     двійок у розкладанні числа i на множ-
                                     ники та отримання індекса елемента в масиві
                                     B, якій замінюється
{
       int q = 1, j = i;
       while(j%2 == 0)          //Виконувати цикл доки j ділиться на 2
       {
       j/=2; q++;
       }
       return q;}
```

Результат тестування роботи алгоритму побудови бінарного коду Грея для множини, потужність якої дорівнює 3, наведено у табл. 1.4.

Таблиця 1.4

| $i$ | $p$ | $q$ | $B$ |
|---|---|---|---|
| 0 | | | 000 |
| 1 | 1 | 1 | 001 |
| 2 | 2 | 2 | 011 |
| 3 | 1 | 1 | 010 |
| 4 | 3 | 3 | 110 |
| 5 | 1 | 1 | 111 |
| 6 | 2 | 2 | 101 |
| 7 | 1 | 1 | 100 |

## 1.10. Зображення множин упорядкованими списками. Виконання операцій над множинами із застосуванням алгоритму типу злиття

У випадку, коли універсум дуже великий (або нескінченний), а підмножини універсуму мають невелику вимірність, для виконання операцій над підмножинами ефективним є зображення множини за допомогою списку елементів. Елемент списку задають записом з двома полями, перше з яких є інформаційне, друге – покажчик на наступний елемент. Список задають покажчиком на перший елемент та зображують у такий спосіб:

```
elem = record
i:= info;                          // інформаційне поле
n: =  ↑ elem                       // покажчик на наступний елемент
end record
```

Якщо елементи у списку впорядковані, то трудомісткість виконання операцій над множинами значно знижується.

Ефективна реалізація операцій над множинами, зображеними впорядкованими списками, заснована на алгоритмі злиття. В алгоритмі типу злиття дві множини, задані впорядкованими списками, переглядають паралельно. На кожному кроці рух відбувається у тій множині, в якій розглянутий елемент є менший. Розглянемо виконання операцій над множинами із застосуванням алгоритму злиття. На вхід алгоритму подаються дві множини $A$ і $B$ у вигляді списків, заданих за допомогою покажчиків на перший елемент списку.

### 1.10.1. Перевірка включення підмножини до множини

Алгоритм перевірки включення підмножини до множини дозволяє встановити, чи є одна множина підмножиною іншої множини. На виході алгоритму функція повертає логічне значення: істина (логічне значення 1), якщо $A \subset B$ , або хибність (логічне значення 0) у протилежному випадку. Розглянемо реалізацію алгоритму.

| Псевдокод | Пояснення |
|---|---|
| `pa := a`<br>`pb := b` | Покажчики на елементи списків встановлюємо на початок списків |
| `while pa ≠ nil and`<br>`pb ≠ nil do` | Ініціюємо цикл, ітерації якого виконуємо до тих пір, доки не досягнемо кінця списків |
| `if pa.i < pb.i then` | Здійснюємо порівняння елементів першого і другого списків. Якщо елемент першого списку менший за елемент другого списку, робимо висновок, що у множині $B$ не існує елементів множини $A$ |
| `return 0` | Функція повертає логічне значення 0, тобто $A \not\subset B$ |
| `else if pa.i > pb.i then`<br>`pb := pb.n` | Порівнюємо елементи першого і другого списків. Якщо елемент першого списку більший, ніж елемент другого списку, здійснюємо перехід до наступного елемента в списку $B$ |
| `else` | В цьому випадку елементи першого і другого списків є тотожні. |
| `pa := pa.n`<br>`pb := pb.n`<br>`end if`<br>`end if` | Перехід до наступного елемента здійснюємо в обох списках водночас |
| `return 1` | Якщо умову пункту 4 не виконано жодного разу, усі елементи множини $A$ належать множині $B$. У цьому випадку функція повертає логічне значення 1, тобто $A \subset B$ |

Кроки виконання алгоритму для випадку, коли задані множини $A = \{1,5,7\}$, $B = \{1,2,3,5,6,7\}$, зображено на рис. 1.3.

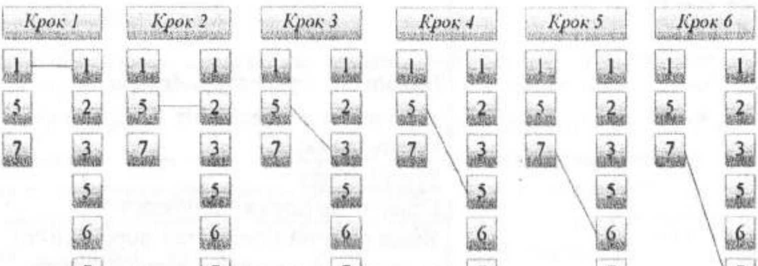



Рис. 1.3

### 1.10.2. Обчислення об'єднання множин

Алгоритм злиття можна застосувати до обчислення об’єднання множин. На вхід алгоритму подаємо дві множини $A$ і $B$ у вигляді впорядкованих списків, на виході отримуємо множину $C = A \cup B$ зі всіх тих і тільки тих елементів, які належать хоча б одній з множин. Розглянемо реалізацію алгоритму:

| Псевдокод | Пояснення |
|---|---|
| `pa := a`<br>`pb := b` | Покажчики на елементи списків встановлюємо на початок списків |
| `c := nil` | Вихідний список $C$ задаємо як порожній |
| `e := nil` | Покажчик на останній елемент вихідного списку $C$ |
| `while` $pa \neq nil$ `and`<br>$pb \neq nil$ `do` | Ініціюємо цикл, ітерації якого виконуємо до тих пір, доки не досягнемо кінця списків |
|   `if` $pa.i < pb.i$ `then`<br>    `d := pa.i`<br>    `pa := pa.n` | Якщо елемент першого списку менший, ніж елемент другого списку, то перехід до наступного елемента здійснюємо в першому списку. У змінної $d$ зберігаємо значення елемента для подальшого його додавання до списку $C$ |
|   `else if` $pa.i > pb.i$ `then`<br>      `d := pb.i`<br>      `pb := pb.n` | Аналогічну послідовність дій виконуємо для випадку, коли елемент першого списку більший,ніж елемент другого списку |
|     `else`<br>      `d := pa.i`<br>      `pa := pa.n`<br>      `pb := pb.n`<br>    `end if`<br>  `end if` | У випадку, коли елементи в обох списках є тотожні, здійснюємо перехід до наступних елементів в обох списках |
|   `Append(c, e, d)` | Виконуємо процедуру *Append*, за допомогою якої елемент $d$ додаємо в кінець списку $C$ |
| `end while` | Кінець циклу |
| `p := nil` | Створюємо порожній покажчик |
| `if` $pa \neq nil$ `then`<br>  `p := pa`<br>`end if` | Якщо перший список не є порожній, то покажчик $p$ вказує на перший список |
| `if` $pb \neq nil$ `then`<br>  `p := pb`<br>`end if` | Аналогічно, якщо другий список не є порожній, то покажчик $p$ вказує на другий список |

| | |
|---|---|
| `while p ≠ nil do`<br>`  Append(c, e, p.i)`<br>`  p := p.n` | Усі елементи, що залишилися в одному зі списків, додаємо до списку $C$ |
| `end while` | Кінець циклу |

Результати виконання алгоритму об'єднання двох множин $A = \{1,2,3\}$ та $B = \{2,5,6,9\}$, отримані за кроками, зображено на рис. 1.4.

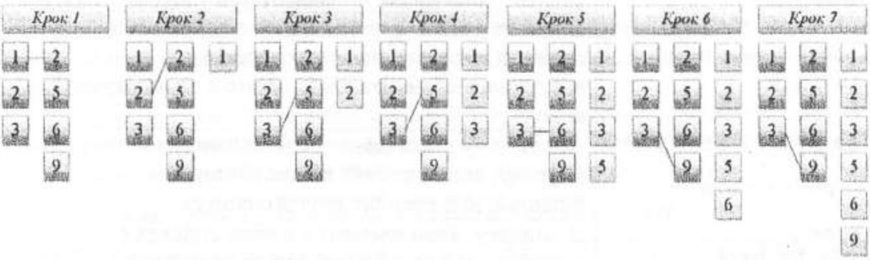



Рис. 1.4

Розглянемо більш детально виконання процедури *Append*, за допомогою якої елемент будемо додавати до результату об'єднання множин. На вхід процедури подаємо покажчики на перший і останній елементи списку, а також елемент $d$, який додається до списку $C$. Процедура передбачає додання елемента $d$ в кінець списку $C$.

| **Псевдокод** | **Пояснення** |
|---|---|
| `q.i := d`<br>`q.n := nil` | Створюємо список $q$, до якого додаємо елемент $d$ |
| `if c = nil then`<br>`  c := q` | Якщо список $C$ є порожній, то до списку $C$ надсилаємо елементи списку $q$ |
| `else`<br>`  c.n := q`<br>`end if` | У випадку, коли список $C$ не є порожній, покажчик встановлюємо на останній елемент списку $C$ та елементи списку $q$ додаємо до списку $C$ |
| `e := q` | Змінна $e$ зберігає покажчик на кінець списку для подальших викликів процедури |

### 1.10.3. Обчислення перерізу множин

Алгоритм злиття можна застосувати до обчислення перерізу множин. На вхід алгоритму подаємо дві множини $A$ і $B$ у вигляді впорядкованих списків, на виході отримуємо множину $C = A \cap B$ зі всіх тих і тільки тих елементів, які належать і множині $A$, і множині $B$. Розглянемо реалізацію алгоритму.

| Псевдокод | Пояснення |
|---|---|
| `pa := a`<br>`pb := b` | Покажчики на елементи списків встановлюємо на початок списків |
| `c := nil` | Вихідний список $C$ задаємо як порожній |
| `e := nil` | Покажчик на останній елемент вихідного списку $C$ |
| `while pa ≠ nil and`<br>`pb ≠ nil do` | Ініціюємо цикл, ітерації якого виконуємо до тих пір, доки не досягнемо кінця списків |
| `  if pa.i < pb.i then`<br>`    pa := pa.n` | Якщо елемент першого списку менший, ніж елемент другого списку, то перехід до наступного елемента здійснюємо в першому списку |
| `  else if pa.i > pb.i then`<br>`      pb := pb.n` | Аналогічну послідовність дій виконуємо для випадку, коли елемент першого списку більший, ніж елемент другого списку |
| `    else`<br>`      d := pa.i`<br>`      pa := pa.n`<br>`      pb := pb.n` | У випадку, коли елементи в обох списках є тотожні, здійснюємо перехід до наступних елементів в обох списках |
| `      Append(c, e, d)` | Виконуємо процедуру *Append*, за допомогою якої елемент $d$ додаємо в кінець списку $C$ |
| `    end if`<br>`  end if`<br>`end while` | Кінець циклу |

Результати виконання алгоритму перерізу двох множин $A = \{1,2,3\}$ та $B = \{2,5,6,9\}$, отримані за кроками, зображено на рис. 1.5.

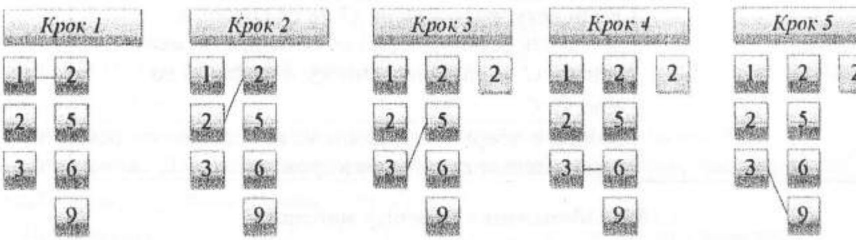



Рис. 1.5

# 2. ВІДНОШЕННЯ

## 2.1. Основні означення

***Означення* 2.1.** Упорядкованою множиною, кортежем або *упорядкованою послідовністю об'єктів називають* сукупність об'єктів, розташованих у певному порядку. При цьому упорядкована пара має такі властивості:

а) для будь-яких двох елементів $x$ і $y$ існує об'єкт, який називають упорядкованою парою та позначають як $\langle x, y\rangle$;

б) якщо $\langle x, y\rangle$ і $\langle u, v\rangle$ - упорядковані пари, то $\langle x, y\rangle = \langle u, v\rangle$ тоді і тільки тоді, коли $x = u$, $y = v$.

Зважаючи на це елемент $x$ будемо називати першою координатою, а елемент $y$ – другою координатою впорядкованої пари $\langle x, y\rangle$.

***Означення* 2.2.** *Бінарним* (або *двомісним*) відношенням $R$ називають підмножину впорядкованих пар, тобто множину, кожен елемент якої є впорядкованою парою.

Якщо $R$ – деяке відношення, то його зображують у такий спосіб:

$$\langle x, y\rangle \in R \text{ або } xRy.$$

***Означення* 2.3.** *Декартовим добутком (або прямим добутком)* множин $X$ і $Y$ називають множину упорядкованих пар $\langle x, y\rangle$, таких що перший елемент $x$ кожної пари належить множині $X$, а другий елемент $y$ – множині $Y$. Декартовий добуток множин позначають так:

$$X \times Y = \{\langle x, y\rangle | x \in X,\ y \in Y\}.$$

***Означення* 2.4.** *Областю визначення* бінарного відношення $R$ називають множину $D(R) = \{x| \quad \exists\, y \text{ такий, що } \langle x, y\rangle \in R\}$.

***Означення* 2.5.** *Областю значень* бінарного відношення $R$ називають множину $E(R) = \{y| \quad \exists\, x \text{ такий, що } \langle x, y\rangle \in R\}$.

Декартовий добуток не є, взагалі кажучи, комутативний та асоціативний, тобто $A \times B \neq B \times A$, $A \times (B \times C) \neq (A \times B) \times C$

***Означення* 2.6.** *Бінарним* відношенням $R$ називають підмножину пар $\langle x, y\rangle \in R$ прямого добутку $X \times Y$, тобто $R \subseteq X \times Y$.

Наприклад, якщо $A = \{1, 2, 3\}$, $B = \{2, 3, 4\}$, то декартовий добуток множин $A \times B$ дорівнює

$$A \times B = \{\langle 1,2\rangle, \langle 1,3\rangle, \langle 1,4\rangle, \langle 2,2\rangle, \langle 2,3\rangle, \langle 2,4\rangle, \langle 3,2\rangle, \langle 3,3\rangle, \langle 3,4\rangle\},$$

декартовий добуток множин $B \times A$ дорівнює

$$B \times A = \{\langle 2,1\rangle, \langle 2,2\rangle, \langle 2,3\rangle, \langle 3,1\rangle, \langle 3,2\rangle, \langle 3,3\rangle, \langle 4,1\rangle, \langle 4,2\rangle, \langle 4,3\rangle\}.$$

## 2.2. Способи задання відношень

Відношення, визначені на скінченних множинах, звичайно зображують у такий спосіб:

1. *Списком (перерахуванням)* упорядкованих пар, для яких це відношення виконується.

2. *Матрицею* – бінарному відношенню $R \subseteq X \times X$ відповідає квадратна матриця порядку $n$, кожен елемент $a_{ij}$ якої дорівнює 1, якщо між парою $x_i$ та $x_j$ відношення $R$ існує, і 0 у протилежному випадку, тобто

$$a_{ij} = \begin{cases} 1, & \text{якщо} \quad x_i R x_j \\ 0, & \text{у протилежному випадку.} \end{cases}$$

Наприклад, на множині $A = \{1, 2, 3, 4, 5\}$ задане бінарне відношення $R \subseteq A \times A$ , яке означає «бути строго більшим». В явному вигляді це відношення можна зобразити:

- описом властивостей елементів

$$R = \{\langle x, y \rangle \in A \times A \mid x > y\} ;$$

- списком пар

$$R = \{\langle 2,1 \rangle, \langle 3,1 \rangle, \langle 3,2 \rangle, \langle 4,1 \rangle, \langle 4,2 \rangle, \langle 4,3 \rangle, \langle 5,1 \rangle, \langle 5,2 \rangle, \langle 5,3 \rangle, \langle 5,4 \rangle\} ;$$

- матрицею відношення $R$

Таблиця 2.1

| $a_{ij}$ | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 |
| 2 | 1 | 0 | 0 | 0 | 0 |
| 3 | 1 | 1 | 0 | 0 | 0 |
| 4 | 1 | 1 | 1 | 0 | 0 |
| 5 | 1 | 1 | 1 | 1 | 0 |

## 2.3. Властивості бінарних відношень

***Означення* 2.7.** Відношення $R \subset A \times A$ називають *рефлексивним*, якщо для кожного $a \in A$ має місце $\langle a, a \rangle \in R$. Головна діагональ матриці такого відношення містить тільки одиниці. Наприклад, відношення "≤" на множині дійсних чисел рефлексивне.

***Означення* 2.8.** Відношення $R \subset A \times A$ називають *антирефлексивним*, якщо ні для якого $a \in A$ не виконується $\langle a, a \rangle \in R$, тобто якщо $\langle a, b \rangle \in R$, то $a \neq b$. Головна діагональ матриці такого відношення містить тільки нулі.

Наприклад, відношення "<" на множині дійсних чисел антирефлексивне.

***Означення* 2.9.** Відношення $R \subset A \times A$ називають *симетричним*, якщо для будь-якої пари $\langle a, b \rangle \in R$ пара $\langle b, a \rangle$ також належить $R$ . Матриця симетричного відношення симетрична відносно головної діагоналі, тобто для всіх індексів $i$ і $j$ виконується $a_{ij} = a_{ji}$.

Наприклад, відношення "=" на множині дійсних чисел симетричне.

***Означення* 2.10.** Відношення $R \subset A \times A$ називають *антисиметричним*, якщо $\langle a,b\rangle \in R$ і $\langle b,a\rangle \in R$ можливе лише у випадку, коли $a = b$, тобто ні для яких елементів $a$ і $b$, що відрізняються один від одного $(a \neq b)$, відношення $\langle a,b\rangle \in R$ і $\langle b,a\rangle \in R$ не існують одночасно. В матриці антисиметричного відношення відсутні одиниці, розташовані симетрично відносно головної діагоналі.

Наприклад, відношення "≤" на множині дійсних чисел антисиметричне.

***Означення* 2.11.** Відношення $R \subset A \times A$ називають *асиметричним*, якщо для будь-якої пари $\langle a,b\rangle \in R$ пара $\langle b,a\rangle \notin R$. Для матриці асиметричного відношення характерно, що всі елементи її головної діагоналі – нулі й немає жодної пари одиниць на місцях, розташованих симетрично відносно головної діагоналі.

Як приклади таких відношень можна навести відношення «бути батьком», «бути підлеглим» на множині людей, відношення строгого включення на множині всіх підмножин деякого універсуму. Очевидно, якщо відношення асиметричне, то воно й антирефлексивне.

***Означення* 2.12.** Відношення $R \subset A \times A$ називають *транзитивним*, якщо для будь-яких $a,b,c$ з умов існування $\langle a,b\rangle \in R$ і $\langle b,c\rangle \in R$ слідує $\langle a,c\rangle \in R$. Наприклад, відношення "≤", "=" на множині дійсних чисел транзитивні.

Наприклад, визначимо властивості відношення $R \subseteq A \times A$, якщо $R = \{\langle 1,1\rangle, \langle 1,2\rangle, \langle 1,4\rangle, \langle 2,1\rangle, \langle 2,2\rangle, \langle 2,4\rangle, \langle 3,3\rangle, \langle 3,5\rangle, \langle 4,4\rangle, \langle 5,3\rangle, \langle 5,5\rangle, \langle 6,6\rangle\}$ та $A = \{1,2,3,4,5,6\}$. Для дослідження властивостей відношення побудуємо матрицю відношення:

Таблиця 2.2

| $a_{ij}$ | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 1 | 0 | 0 |
| 2 | 1 | 1 | 0 | 1 | 0 | 0 |
| 3 | 0 | 0 | 1 | 0 | 1 | 0 |
| 4 | 1 | 1 | 0 | 1 | 0 | 0 |
| 5 | 0 | 0 | 1 | 0 | 1 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0 | 1 |

Відношення $R$ рефлексивне, тому що для кожного елемента $a \in A$, пара $\langle a,a\rangle \in R$. Головна діагональ матриці відношення $R$ містить одиниці.

Відношення $R$ не є антирефлексивне, тому що з наявності пари $\langle a,b\rangle \in R$ не слідує $a \neq b$, наприклад, $\langle 2,2\rangle \in R$, але $2 = 2$.

Відношення $R$ симетричне, оскільки матриця відношення $R$ симетрична відносно головної діагоналі.

Відношення $R$ не є антисиметричне, тому що $\langle 3,5\rangle \in R$ і $\langle 5,3\rangle \in R$, але $3 \neq 5$.

Для дослідження транзитивності зручно застосовувати таку схему (рис. 2.1).

Побудуємо три стовпці з шістьма точками в кожному, кількість точок у стовпці дорівнює кількості елементів у множині $A$. Елементи першого та другого стовпців, а також другого і третього стовпців з'єднуються стрілками, якщо відповідна пара належить відношенню $R$. На другій схемі з'єднуємо між собою стрілками усі пари елементів, для яких між елементами першого та третього стовпців першої схеми є шлях, побудований із двох стрілок. Якщо усі пари другої побудованої схеми збігаються з парами, з яких складається відношення $R$, то відношення $R$ є транзитивне.

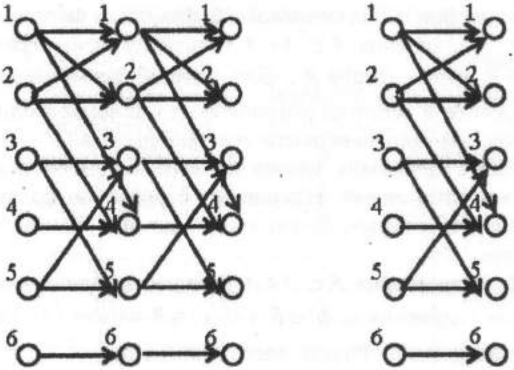



Рис.2.1

## 2.4. Операції над бінарними відношеннями

Для бінарних відношень, як і для множин, визначають теоретико-множинні операції :

1. *Об'єднання відношень* – $R_1 \bigcup R_2 = \{\langle a,b\rangle \mid \langle a,b\rangle \in R_1 \ \ \textit{або} \ \ \langle a,b\rangle \in R_2\}$.

2. *Переріз відношень* – $R_1 \bigcap R_2 = \{\langle a,b\rangle \mid \langle a,b\rangle \in R_1 \ \ i \ \ \langle a,b\rangle \in R_2\}$.

3. *Різниця відношень* – $R_1 \setminus R_2 = \{\langle a,b\rangle \mid \langle a,b\rangle \in R_1 \ \ i \ \ \langle a,b\rangle \notin R_2\}$.

4. *Доповнення до відношення* – $\overline{R} = (A \times B) \setminus R$.

***Означення* 2.13.** *Оберненим відношенням* до даного відношення $R$ називають множину $R^{-1} = \{\langle b,a\rangle \mid \langle a,b\rangle \in R\}$.

Наприклад, оберненим відношенням $R^{-1}$ до відношення $R = \{\langle 2,1\rangle, \langle 3,1\rangle, \langle 3,2\rangle, \langle 4,1\rangle, \langle 4,2\rangle, \langle 4,3\rangle, \langle 5,1\rangle, \langle 5,2\rangle, \langle 5,3\rangle, \langle 5,4\rangle\}$ є відношення $R^{-1} = \{\langle 1,2\rangle, \langle 1,3\rangle, \langle 2,3\rangle, \langle 1,4\rangle, \langle 2,4\rangle, \langle 3,4\rangle, \langle 1,5\rangle, \langle 2,5\rangle, \langle 3,5\rangle, \langle 4,5\rangle\}$.

***Означення* 2.14.** *Композицією* відношень $R_1$ і $R_2$ називають відношення

$$R_1 \circ R_2 = \{\langle a,b\rangle \mid \ \exists c \ \ \textit{такий, що} \langle a,c\rangle \in R_1 \ i \ \langle c,b\rangle \in R_2\}.$$

Зокрема, якщо відношення $R$ задано на множині $A$ $(R \subseteq A \times A)$, то композицію визначають як $R \circ R = R^2 = \{\langle a,b\rangle | \ \langle a,c\rangle \in R \ i \ \langle c,b\rangle \in R\}$.

Наприклад, задані відношення $R_1 = \{\langle 1,1\rangle, \langle 1,3\rangle, \langle 2,1\rangle, \langle 2,2\rangle, \langle 3,2\rangle\}$ і $R_2 = \{\langle 1,1\rangle, \langle 1,2\rangle, \langle 2,3\rangle, \langle 3,1\rangle, \langle 3,3\rangle\}$, необхідно знайти $R_1 \circ R_2$.

Для отримання результату виконання композиції відношень зручно застосовувати таку схему (рис. 2.2). На перший схемі рис. 2.2 побудовано три стовпці з трьох точок в кожному, кількість точок у кожному стовпці дорівнює кількості елементів у множині $A$. Елементи першого та другого стовпців з'єднані стрілками, якщо відповідна пара належить відношенню $R_1$, що є першим множником. Елементи другого і третього стовпців також з'єднані стрілками, якщо відповідна пара належить відношенню $R_2$ – другому множнику композиції $R_1 \circ R_2$.

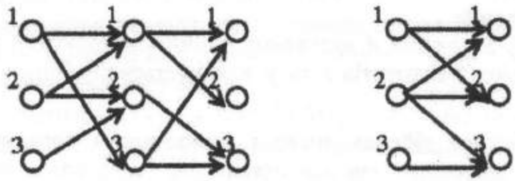



Рис. 2.2

До композиції відношень входять пари, складені з елементів першого та третього стовпців, між якими на схемі існує шлях, побудований з двох стрілок. Аналіз другої схеми (рис. 2.2) дозволяє отримати результат виконання операції композиції відношень $R_1$ і $R_2$ у такий спосіб:

$$R_1 \circ R_2 = \{\langle 1,1\rangle, \langle 1,2\rangle, (2,1), (2,2), \langle 2,3\rangle, \langle 3,3\rangle\}.$$

## 2.5. Відношення еквівалентності

***Означення* 2.15.** Бінарне відношення називають *еквівалентністю,* якщо воно рефлексивне, симетричне і транзитивне.

Відношення еквівалентності будемо позначати символом "≡".

Наприклад, відношення "*бути тотожними*", визначене на множині натуральних чисел, і "*бути подібними*", визначене на множині трикутників, є еквівалентністю.

Відзначимо, що кожному відношенню еквівалентності на множині $A$ відповідає єдине розбиття даної множини на підмножини, та будь-якому розбиттю множини $A$ на підмножини однозначно відповідає деяке відношення еквівалентності. При цьому елементи, які належать до однієї підмножини у розбитті, вважають такими, що знаходяться у відношенні еквівалентності, та навпаки, якщо елементи множини $A$ не зв'язані еквівалентністю, то вони входять до різних підмножин у розбитті. Підмножину елементів, еквівалентних елементу $a \in A$, будемо називати класом еквівалентності.

## 2.6. Відношення порядку

Відношення порядку дозволяє порівнювати між собою елементи однієї множини. Наприклад, відношення "*бути менше*" , "*бути більше*" , задані на множині $A$, дозволяють розташовувати елементи множини у відповідному порядку.

***Означення*** **2.16.** Бінарне відношення називають *відношенням порядку,* якщо воно антисиметричне і транзитивне. Відношення може бути також і рефлексивним, тоді його називають *відношенням нестрогого порядку.* Якщо відношення антирефлексивне, тоді його називають *відношенням строгого порядку.*

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

***Означення*** **2.17.** Множину $A$ називають лінійно (абсолютно) впорядкованою, якщо для будь-яких двох її елементів $x$ та $y$ виконується нерівність $x < y$ або $y < x$ ($x \le y$ або $y \le x$).

Наприклад, множина дійсних чисел з відношенням порядку "<" є лінійно впорядкована. Може виявитись, що для деяких пар $(x, y)$ жодне зі співвідношень $x < y$ або $y < x$ не виконується. Такі елементи $x$ та $y$ називають непорівнянними. У цьому випадку говоримо, що множина є частково впорядкована.

***Означення*** **2.18.** Мінімальним (максимальним) елементом множини $A$, на якій задано відношення нестрогого порядку, називають такий елемент $x \in A$, що для всякого елемента $y \in A$, що порівнюється з $x$, має місце нерівність $x \le y$ ($y \le x$).

***Означення*** **2.19.** Елемент $x \in A$ називають найменшим (найбільшим), якщо для кожного елемента $y \in A$ виконується $x \le y$ ($y \le x$).

З антисиметричності відношення порядку випливає, що будь-яка скінченна упорядкована множина містить не більше одного найбільшого (найменшого) елемента.

## 2.7. Відношення домінування

У випадках, коли мова йде про множину людей, мають місце відношення домінування.

***Означення*** **2.20.** Бінарне відношення називають *відношенням домінування,* якщо воно антирефлексивне і антисиметричне. Говорять, що $x$ домінує над $y$ (позначається $x >> y$), коли $x$ у якомусь розумінні переважає $y$.

Наприклад, якщо всередині організації існує відношення підпорядкованості, то для двох членів організації один домінує над іншим, якщо він керує ним.

## 2.8. Функції

Для аналізу відношень $R \subseteq A \times B$, утворених множинами $A$ і $B$, які мають різну природу, вводять поняття відображення та функції.

***Означення* 2.21.** *Функцією* $f$ називають таке відношення $R$, ніякі два різних елементи якого не мають однакових перших координат. Тобто $f$ є функцією тоді й тільки тоді, коли вона задовольняє такі умови:

- елементами $f$ є упорядковані пари;
- якщо упорядковані пари $\langle a,b \rangle$ і $\langle a,c \rangle$ – елементи функції $f$, то $b = c$.

Отже, відношення $f$ на $A \times B$ називають функцією з множини $A$ в множину $B$ і позначають як $f : A \to B$.

Якщо $f : A \to B$ функція і $\langle a,b \rangle \in f$, то говорять, що $b = f(a)$.

***Означення* 2.22.** Множину $A$ називають *областю визначення* функції $f$ і позначають $D(f)$, а множину $B$ – *областю потенційних значень*. Якщо $I \subseteq A$, то множину $f(I) = \{b| \quad f(a) = b \text{ для деякого } a \in I\}$ називають *образом* множини $I$. Образ усієї множини $A$ називають *областю значень* функції $f$ і позначають $E(f)$.

Функціональне відношення можна розглядати як функцію. При цьому перша координата $a$ впорядкованої пари $\langle a,b \rangle \in f$ є прообразом (аргументом, змінною), а друга координата $b$ – образом (значенням функції). Якщо функціональне відношення $f \subset A \times B$ всюди визначене на $A$, то його називають відображенням множини $A$ на множину $B$ і записують $f : A \to B$. Очевидно, що різниця між відображенням та функцією зводиться до способу визначення цих відношень на множині $A$, причому відображення потрібно розглядати як окремий випадок функції. Більшість математиків не розрізняють поняття відображення і функції.

**Завдання для самостійної роботи**

4. Для множин $A = \{1, 2, 3\}$, $B = \{1, 3, 4\}$, $C = \{2, 4\}$ побудуйте декартові добутки $A \times B$, $A \times C$, $B \times C$, $B \times A$, $A \times B \times C$.
5. Побудуйте композицію відношень $R_1 \circ R_2$, якщо $R_1 = \{\langle 1,1 \rangle, \langle 1,2 \rangle, \langle 1,3 \rangle, \langle 2,3 \rangle, \langle 3,1 \rangle, \langle 3,3 \rangle\}, R_2 = \{\langle 1,1 \rangle, \langle 2,1 \rangle, \langle 2,2 \rangle, \langle 3,2 \rangle, \langle 3,3 \rangle\}$.
6. Подайте відношення $R = \{\langle 1,1 \rangle, \langle 1,2 \rangle, (2,1), (2,2), \langle 2,3 \rangle, \langle 3,3 \rangle\}$, задане на множині $A = \{1, 2, 3, 4\}$, у вигляді матриці відношення, побудуйте композицію відношень $R \circ R$.
7. Дослідіть властивості відношення:

- $R = \{\langle 1,1 \rangle, \langle 1,2 \rangle, \langle 1,3 \rangle, \langle 2,2 \rangle, \langle 2,4 \rangle, \langle 3,3 \rangle, \langle 4,1 \rangle, \langle 4,4 \rangle\}$, якщо $R \subseteq A \times A$ та $A = \{1,2,3,4\}$;
- $R = \{\langle x, y \rangle| \quad x \text{ брат } y\}$.

# 3. АЛГЕБРА ЛОГІКИ

## 3.1. Логічні функції. Основні означення

У процесі обчислення складних логічних висловлювань, проектування обчислювальної техніки, формалізації опису дій, які відбуваються в цифрових ЕОМ, застосовують математичний апарат алгебри логіки, або булевої алгебри, названої так за ім'ям її творця – англійського математика Джорджа Буля (1815–1864). Суттєвий внесок у створення алгебри логіки зробили також німецький і російський математики Г.В. Лейбніц (1646–1716) і П.С. Порецький (1846–1907).

Алгебра логіки була створена через необхідність точного опису математичних доведень складних висловлювань на підставі простих передумов, з метою зведення операцій над логічними висловлюваннями до формальних операцій над символами. Алгебра логіки оперує об'єктами (функціями або змінними функцій), які можуть набувати тільки двох різних значень – "хибність" та "істина", а це відповідає значенням двійкових змінних "0" і "1". Логічними змінними описують об'єкти з двома можливими станами. Для алгебри логіки такими змінними є висловлювання – будь-які твердження, стосовно яких можна сказати, що вони істинні чи хибні.

Нехай $B = \{0,1\}$ – бінарна множина, елементами якої є двійкові змінні 1 і 0, що не мають арифметичного сенсу та інтерпретуються як "істина", "хибність" або "так", "ні".

***Означення 3.1. Функцією алгебри логіки*** (або ***логічною функцією, булевою функцією***) від $n$ змінних $f(x_1, x_2, ..., x_n)$ називають $n$- вимірну логічну операцію на множині $B = \{0,1\}$. Тобто $f : B^n \to B$. Логічна функція – функція від логічних змінних може набувати тільки два логічні значення – 0 або 1.

Множину всіх логічних функцій будемо позначати як $P_2$, множину всіх логічних функцій $n$ змінних – $P_2(n)$.

***Означення 3.2. Логічною формулою*** називають формулу, складену з букв, знаків логічних операцій і дужок. При цьому буквами позначають логічні змінні. Кожна формула задає логічну функцію.

Довільну булеву функцію задають за допомогою таблиці значень або аналітичною формулою.

Оскільки аргументи логічних функцій можуть набувати лише двох значень, область визначення будь-якої логічної функції обмежена. Тому будь-яка функція алгебри логіки може бути задана таблицею її значень залежно від значень аргументів (таблицею істинності). У разі визначення логічної функції за допомогою таблиці істинності у таблиці ліворуч виписують усі можливі набори значень логічних змінних, а праворуч – значення функції, що відповідають цим наборам. Набір значень змінних, за якого $f(x_1, x_2, ..., x_n) = 1$, називають *одиничним набором функції*, набір значень змінних, за якого $f(x_1, x_2, ..., x_n) = 0$, називають *нульовим набором функції*.

Для логічної функції $f(x_1, x_2, ..., x_n)$ від $n$ змінних число всіх можливих наборів, що відрізняються один від одного, дорівнює $2^n$. Ця множина з $2^n$ наборів є *областю визначення* логічної функції. Число всіх різних функцій $n$ змінних дорівнює числу можливих розміщень нулів і одиниць у $n$ стовпцях таблиці з $2^n$ рядками, тобто $2^{2^n}$.

Кількість логічних функцій від $n$ змінних різко зростає зі збільшенням значення $n$ (табл. 3.1).

Таблиця 3.1

| Кількість змінних, $n$ | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Число логічних функцій | 4 | 16 | 256 | 65536 | $4{,}3 \cdot 10^9$ |

***Означення* 3.3.** Змінну $x_i$ логічної функції $f(x_1, x_2, ..., x_{i-1}, x_i, x_{i+1}, ..., x_n)$ називають ***несуттєвою*** (або ***фіктивною***), якщо $f(x_1, ..., x_{i-1}, 1, x_{i+1}, ..., x_n) = f(x_1, ..., x_{i-1}, 0, x_{i+1}, ..., x_n)$ за будь-яких значень інших змінних, тобто зміна $x_i$ в будь-якому наборі значень $x_1, ..., x_n$ не змінює значення функції.

Нехай змінна $x_i$ є фіктивна для функції $f(x_1, x_2, ..., x_{i-1}, x_i, x_{i+1}, ..., x_n)$. Вилучимо з таблиці істинності стовпець аргумент $x_i$. Відтак отримаємо нову таблицю для функції $n-1$ змінної, яку позначимо як $g(x_1, x_2, ..., x_{i-1}, x_{i+1}, ..., x_n)$. Отже, функцію $g$ отримали з функції $f$ шляхом вилучення фіктивної змінної.

Вилучення та додавання змінних є ефективний прийом, оскільки будь-яку скінченну сукупність функцій можна вважати залежною від однієї і тієї ж множини змінних.

Множину всіх логічних функцій однієї змінної наведено в табл. 3.2. Загальна кількість логічних функцій однієї змінної дорівнює $2^{2^1} = 4$.

Таблиця 3.2

| Значення змінної $x$ | $f_0(x)$ | $f_1(x)$ | $f_2(x)$ | $f_3(x)$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| Умовне позначення | 0 | $x$ | $\overline{x}$ | 1 |
| Найменування функції | Константа нуль | Змінна $x$ | Інверсія змінної $x$ | Константа одиниця |

Значення функцій $f_0(x)$, $f_3(x)$ є константами, вони не залежать від змінної $x$, отже, для них змінна $x$ є фіктивна змінна. Функція $f_1(x) = x$, тобто збігається зі змінною $x$, $f_2(x) = \overline{x}$, тобто є запереченням змінної $x$.

Розглянемо логічні функції двох аргументів, їх кількість дорівнює $2^{2^2} = 16$. Множину всіх логічних функцій двох змінних наведено в табл. 3.3. Як бачимо, із 16 функцій двох змінних 6 мають фіктивні змінні: у функціях $f_0$ і $f_{15}$ фіктивні змінні $x_1$ і $x_2$; у функціях $f_5$ і $f_{10}$ фіктивна змінна $x_1$; у функціях $f_3$ і $f_{12}$ фіктивна змінна $x_2$.

Таблиця 3.3

| Назва функції | Функція | Назва змінної | Значення змінних $x_1$, $x_2$ | | | | Умовне позначення функції | Виконувана функція |
|---|---|---|---|---|---|---|---|---|
| | | Змінна $x_1$ | 0 | 0 | 1 | 1 | | |
| | | Змінна $x_2$ | 0 | 1 | 0 | 1 | | |
| Константа нуль | $f_0(x_1, x_2)$ | | 0 | 0 | 0 | 0 | 0 | 0 |
| Кон'юнкція | $f_1(x_1, x_2)$ | | 0 | 0 | 0 | 1 | $x_1 \wedge x_2$<br>$x_1 \cdot x_2$<br>$x_1 \,\&\, x_2$ | $x_1 x_2$ |
| Заперечення імплікації | $f_2(x_1, x_2)$ | | 0 | 0 | 1 | 0 | $\overline{x_1 \to x_2}$ | $x_1 \overline{x_2}$ |
| Змінна $x_1$ | $f_3(x_1, x_2)$ | | 0 | 0 | 1 | 1 | $x_1$ | $x_1$ |
| Заперечення коімплікації | $f_4(x_1, x_2)$ | | 0 | 1 | 0 | 0 | $\overline{x_2 \to x_1}$ | $\overline{x_1} x_2$ |
| Змінна $x_2$ | $f_5(x_1, x_2)$ | | 0 | 1 | 0 | 1 | $x_2$ | $x_2$ |
| Додавання за модулем 2 | $f_6(x_1, x_2)$ | | 0 | 1 | 1 | 0 | $x_1 \oplus x_2$<br>$x_1 \Delta x_2$ | $\overline{x_1} x_2 \vee \overline{x_2} x_1$ |
| Диз'юнкція | $f_7(x_1, x_2)$ | | 0 | 1 | 1 | 1 | $x_1 \vee x_2$ | $x_1 \vee x_2$ |
| Стрілка Пірса (функція Вебба) | $f_8(x_1, x_2)$ | | 1 | 0 | 0 | 0 | $x_1 \downarrow x_2$ | $\overline{x_1 \vee x_2} = \overline{x_1}\,\overline{x_2}$ |
| Логічна еквівалентність | $f_9(x_1, x_2)$ | | 1 | 0 | 0 | 1 | $x_1 \equiv x_2$ | $\overline{x_1}\,\overline{x_2} \vee x_1 x_2$ |
| Інверсія змінної $x_2$ | $f_{10}(x_1, x_2)$ | | 1 | 0 | 1 | 0 | $\overline{x_2}$ | $\overline{x_2}$ |
| Коімплікація | $f_{11}(x_1, x_2)$ | | 1 | 0 | 1 | 1 | $x_2 \to x_1$ | $x_1 \vee \overline{x_2}$ |
| Інверсія змінної $x_1$ | $f_{12}(x_1, x_2)$ | | 1 | 1 | 0 | 0 | $\overline{x_1}$ | $\overline{x_1}$ |
| Імплікація | $f_{13}(x_1, x_2)$ | | 1 | 1 | 0 | 1 | $x_1 \to x_2$ | $\overline{x_1} \vee x_2$ |
| Штрих Шеффера | $f_{14}(x_1, x_2)$ | | 1 | 1 | 1 | 0 | $x_1 \vert x_2$ | $\overline{x_1 x_2} = \overline{x_1} \vee \overline{x_2}$ |
| Константа одиниця | $f_{15}(x_1, x_2)$ | | 1 | 1 | 1 | 1 | 1 | 1 |

## 3.2. Зображення булевих функцій формулами

Булеві функції (табл. 3.3) дозволяють будувати нові булеві функції за допомогою узагальненої операції, яка має назву операції суперпозиції. Операція суперпозиції полягає в підстановці замість аргументів функції інших булевих функцій (зокрема, аргументів).

***Означення 3.4. Суперпозицією функцій*** $f_1,...,f_n$ називають функцію $f$, отриману за допомогою підстановок цих функцій одна в одну і перейменування змінних.

Для запису формул будемо використовувати знаки операцій (функцій): $\wedge$, $\vee$, $\rightarrow$, $\downarrow$, $\oplus$, $|$, $\equiv$, а для визначення порядку виконання операцій (пріоритету операцій) – дужки; у разі їх відсутності у запису формули першими виконуватимемо операції заперечення (інверсії), потім – кон'юнкції, диз'юнкції, імплікації, еквівалентності, додавання за модулем 2, операції штрих Шеффера, стрілка Пірса. Операції, які мають однаковий пріоритет, виконуватимемо у порядку їх появи у запису формули, тобто зліва направо.

Наприклад, функція $f(x_1,x_2,x_3)=f_4(f_1(x_2,f_2(x_3,x_1)),f_3(x_2,x_1))$ являє собою суперпозицію функцій, з яких $f_1$ означає кон'юнкцію, $f_2$ – диз'юнкцію, $f_3$ – імплікацію, $f_4$ – додавання за модулем 2. Тоді функцію $f(x_1,x_2,x_3)$ можна зобразити формулою $f(x_1,x_2,x_3)=(x_2\wedge(x_3\vee x_1))\oplus(x_2\rightarrow x_1)$.

Побудовану функцію можна обчислити на можливих наборах змінних. Наприклад, обчислимо значення функції на наборі $(0,0,1)$, тобто коли змінні функції набувають таких значень: $x_1=0$, $x_2=0$, $x_3=1$. Для обчислення значення функції $f(x_1,x_2,x_3)$ на наборі $(0,0,1)$ підставимо в отриману формулу конкретні значення змінних $x_1$, $x_2$, $x_3$: $(x_2\wedge(x_3\vee x_1))\oplus(x_2\rightarrow x_1)= =(0\wedge(1\vee 0))\oplus(0\rightarrow 0)=1$.

На наборі змінних $(0,0,1)$ функція $f(x_1,x_2,x_3)$ набуває значення 1, тобто функція істинна.

***Означення 3.5. Еквівалентними,*** або ***рівносильними,*** називають формули, що подають одну і ту саму функцію. Еквівалентність формул в алгебрі логіки позначають символом "=".

Для того щоб встановити еквівалентність формул, необхідно побудувати їх таблиці істинності і порівняти значення функції на кожному наборі змінних.

Наприклад, доведемо еквівалентність формул $x_1\oplus x_2$ та $(x_1\wedge\bar{x}_2)\vee(\bar{x}_1\wedge x_2)$. Для цього складемо таблиці істинності наведених формул (табл. 3.4).

Таблиця 3.4

| $x_1$ | $x_2$ | $x_1\oplus x_2$ | $(x_1\wedge\bar{x}_2)\vee(\bar{x}_1\wedge x_2)$ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 |

Як бачимо, значення функцій $x_1 \oplus x_2$ і $(x_1 \wedge \bar{x}_2) \vee (\bar{x}_1 \wedge x_2)$, наведені у таблиці істинності, збігаються. Звідси робимо висновок, що формули еквівалентні.

***Теорема* 3.1.** Кожну булеву функцію можна зобразити у вигляді формули.

***Теорема* 3.2.** Кожну булеву функцію можна зобразити у вигляді формули, у якої наявні тільки операції заперечення (інверсії), кон'юнкції та диз'юнкції.

Наприклад, наведені в табл. 3.3 функції, які відрізняються від заперечення (інверсії), кон'юнкції, диз'юнкції, можна зобразити з використанням операцій заперечення, кон'юнкції та диз'юнкції у такий спосіб: імплікація – $x_1 \to x_2 = \overline{x_1} \vee x_2$; еквівалентність – $x_1 \equiv x_2 = (\overline{x_1} \vee x_2) \wedge (\overline{x_2} \vee x_1)$; штрих Шеффера – $x_1 | x_2 = \overline{x_1 \wedge x_2} = \overline{x_1} \vee \overline{x_2}$; стрілка Пірса – $x_1 \downarrow x_2 = \overline{x_1 \vee x_2} = \overline{x_1} \wedge \overline{x_2}$.

## 3.3. Закони алгебри логіки

До основних законів алгебри логіки відносять:

1. *Закон ідемпотентності:* $x \wedge x = x$; $x \vee x = x$.
2. *Закон подвійного заперечення:* $\bar{\bar{x}} = x$.
3. *Закон комутативності:* $x \wedge y = y \wedge x$; $x \vee y = y \vee x$.
4. *Закон асоціативності (закон сполучення):*
   для кон'юнкції – $x \wedge (y \wedge z) = (x \wedge y) \wedge z = x \wedge y \wedge z$;
   для диз'юнкції – $x \vee (y \vee z) = (x \vee y) \vee z = x \vee y \vee z$.
5. *Закон дистрибутивності (закон розподілу):*
   для диз'юнкції – $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$;
   для кон'юнкції – $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$.
6. *Закон де Моргана (правило інверсії):* $\overline{x \vee y} = \bar{x} \wedge \bar{y}$; $\overline{x \wedge y} = \bar{x} \vee \bar{y}$.
7. *Закон протиріччя:* $x \wedge \bar{x} = 0$.
8. *Закон виключення третього:* $x \vee \bar{x} = 1$.
9. *Закон поглинання:* $(x \wedge y) \vee x = x$.
10. *Закон склеювання:* $(x \wedge y) \vee (x \wedge \bar{y}) = x$.
11. *Закон узагальненого склеювання:*
$$(x \wedge y) \vee (y \wedge z) \vee (\bar{x} \wedge z) = (x \wedge y) \vee (\bar{x} \wedge z).$$

Довести ці тотожності можна, скориставшись даними таблиці істинності.

За допомогою цих законів можна виконувати різні тотожні перетворення булевих функцій, які становлять величезний інтерес у задачах зі спрощення формул. Оскільки формули переважно являють собою суперпозицію інших формул і функцій, то можна говорити про входження до формули інших формул, які мають назву *підформул*. Виконуючи тотожні перетворення, будь-які підформули можна замінювати на еквівалентні їм.

Наприклад, спростимо формулу $(y \vee z) \vee (x \wedge \bar{z})$, застосовуючи закони алгебри логіки:

$(y \vee z) \vee (x \wedge \bar{z}) = \ y \vee (z \vee (x \wedge \bar{z})) =$ (за законом асоціативності)

$= \ y \vee ((z \vee x) \wedge (z \vee \bar{z})) =$ (за законом дистрибутивності)

$= \ y \vee ((z \vee x) \wedge 1) =$ (за законом виключення третього)

$= \ y \vee (z \vee x)$ (враховуючи властивості констант).

**Завдання для самостійної роботи**

1. Нехай $f_1$ означає стрілку Пірса, $f_2$ – диз'юнкцію, $f_3$ – еквівалентність, $f_4$ – додавання за модулем 2. Зобразіть функцію формулою і обчисліть значення функції на зазначеному наборі:

а) $f_2(f_4(x_1, f_3(x_2, x_1)), f_1(x_3, x_1))$; $(0,1,1)$;

б) $f_2(f_1(x_1, x_2), f_3(x_2, f_1(x_3, x_2)))$ ; $(1,0,1)$;

в) $f_1(f_2(f_3(x_1, x_2), f_1(x_1, x_3)), x_2)$; $(0,0,1)$;

2. Скориставшись законами булевої алгебри, максимально спростіть вирази. За допомогою таблиць істинності порівняйте початковий вираз зі спрощеним:

а) $(x \vee (\bar{k} \wedge y)) \wedge ((\bar{x} \wedge (\bar{y} \vee k)) \vee z) \vee \bar{z} \vee (x \vee (y \wedge \bar{k}))$;

б) $((x \vee z) \wedge (x \vee k)) \wedge (((z \vee (z \wedge y)) \wedge \bar{z}) \vee \bar{x})$;

в) $(\bar{y} \vee k) \wedge ((\bar{k} \wedge z) \vee (x \wedge z) \vee (\bar{k} \wedge \bar{z}) \vee (x \wedge \bar{z})) \wedge (y \vee k)$;

г) $(x \vee \bar{z}) \wedge (\bar{x} \vee \bar{y}) \wedge (\bar{y} \vee z) \wedge (\bar{x} \vee y) \wedge (y \vee z)$;

д) $(x \wedge z) \vee ((y \vee \bar{k}) \wedge (\bar{x} \vee \bar{k}) \wedge (\bar{x} \vee k) \wedge (k \vee y)) \vee (x \wedge \bar{z})$;

е) $((\bar{y} \vee \bar{z}) \wedge (x \vee y)) \vee (k \wedge \bar{z}) \vee (((\bar{y} \wedge \bar{x}) \vee z) \wedge (x \vee y))$.

## 3.4. Розкладання булевої функції за змінними. Досконала нормальна форма

***Теорема* 3.3.** Будь-яку логічну функцію можна зобразити формулою

$$f(x_1, x_2, \dots, x_n) = \bigvee_{\sigma_1 \sigma_2 \dots \sigma_m} x_1^{\sigma_1} x_2^{\sigma_2} \dots x_m^{\sigma_m} \ f(\sigma_1, \sigma_2, \dots, \sigma_m, x_{m+1}, x_{m+2}, \dots, x_n), \quad (3.1)$$

де $m \le n$, а диз'юнкцію обчислено на всіх $2^m$ наборах значень змінних $x_1$, $x_2$, ..., $x_m$. Зображення (3.1) має назву розкладання булевої функції $f(x_1, x_2, \dots, x_n)$ за змінними $x_1$, $x_2$, ..., $x_m$.

Найбільш важливим випадком є розкладання булевої функції за всіма її змінними. Тоді усі змінні в правій частині (3.1) набувають фіксованих значень, а функції $f(\sigma_1, \sigma_2, \dots, \sigma_n)$ в кон'юнкціях правої частини (3.1) дорівнюють 0 або 1.

У цьому випадку отримуємо :

$$f(x_1, x_2, \dots, x_n) = \vee \ x_1^{\sigma_1} x_2^{\sigma_2} \dots x_n^{\sigma_n} ; \ f(\sigma_1, \sigma_2, \dots, \sigma_n) = 1, \quad (3.2)$$

тут диз'юнкція береться за всіма наборами $(\sigma_1, \sigma_2, \dots, \sigma_n)$, на яких функція $f(\sigma_1, \sigma_2, \dots, \sigma_n) = 1$.

Розкладання функції за всіма її змінними $x_1$, $x_2$, ..., $x_n$, зображене в вигляді (3.2), називають *досконалою диз'юнктивною нормальною формою (ДДНФ)*.

Якщо функцію подати таблицею істинності, то ДДНФ функції $f$ буде містити стільки диз'юнкцій, скільки разів функція $f$ у рядках таблиці істинності набуває значень 1, тобто кожному одиничному набору $(\sigma_1, \sigma_2, ..., \sigma_n)$ відповідає кон'юнкція всіх змінних, у яку змінна $x_i$ потрапляє із запереченням, якщо $\sigma_i = 0$, і без заперечення, якщо $\sigma_i = 1$.

Існує взаємно однозначна відповідність між таблицею функції $f$ і її ДДНФ.

***Теорема* 3.4.** Для будь-якої булевої функції $n$ змінних (окрім константи 0) існує єдина досконала диз'юнктивна нормальна форма.

Наприклад, побудуємо ДДНФ функції, заданої таблицею:

Таблиця 3.5

| Значення змінної $x_1$ | Значення змінної $x_2$ | Значення змінної $x_3$ | Значення функції $f$ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 |

Знайдемо набори змінних, на яких функція $f(x_1, x_2, x_3)$ набуває значення 1. Відповідно до (3.2) ДДНФ даної функції має вигляд

$$f(x_1, x_2, x_3) = \bar{x}_1\bar{x}_2x_3 \vee \bar{x}_1x_2\bar{x}_3 \vee x_1\bar{x}_2x_3.$$

***Зауваження* 1.** Єдина функція, що не має ДДНФ – це константа 0, тому що в її таблиці істинності не існує жодного одиничного набору.

***Наслідок* 1.** Будь-яку логічну функцію можна зобразити формулою

$$f(x_1, x_2, ..., x_n) = \wedge\left(\bigvee_{i=1}^{n} x_i^{\overline{\sigma_i}}\right);\ f(\sigma_1, \sigma_2, ..., \sigma_n) = 0, \tag{3.3}$$

тут кон'юнкція береться за всіма наборами $(\sigma_1, \sigma_2, ..., \sigma_n)$, на яких функція $f(x_1, x_2, ..., x_n) = 0$.

Таке зображення функції називають *досконалою кон'юнктивною нормальною формою (ДКНФ)*. ДКНФ функції $f$ містить стільки кон'юнкцій, скільки нулів у таблиці істинності для функції $f$; кожному нульовому набору $(\sigma_1, \sigma_2, ..., \sigma_n)$ відповідає диз'юнкція всіх змінних, у яку змінна $x_i$ потрапляє із запереченням, якщо $\sigma_i = 1$, і без заперечення, якщо $\sigma_i = 0$.

Існує взаємно однозначна відповідність між таблицею функції $f$ і її ДКНФ, отже, ДКНФ для всякої логічної функції єдина.

Наприклад, для функції (табл. 3.5) ДКНФ матиме такий вигляд:

$f(x_1,x_2,x_3)=(x_1\vee x_2\vee x_3)(x_1\vee \bar{x}_2\vee \bar{x}_3)(\bar{x}_1\vee x_2\vee x_3)(\bar{x}_1\vee \bar{x}_2\vee x_3)(\bar{x}_1\vee \bar{x}_2\vee \bar{x}_3)$.

***Зауваження* 2.** Єдина функція, що не має ДКНФ – це константа 1, тому що в її таблиці істинності не існує жодного нульового набору.

Наприклад, побудуємо ДДНФ і ДКНФ для логічної функції трьох змінних, зображеної формулою $f(x_1,x_2,x_3)=((x_1\vee x_2)\bar{x}_3)\to(x_1\equiv x_3)$. Почнемо із побудови таблиці істинності формули:

Таблиця 3.6

| $x_1$ | $x_2$ | $x_3$ | $\bar{x}_3$ | $x_1\vee x_2$ | $(x_1\vee x_2)\bar{x}_3$ | $x_1\equiv x_3$ | $((x_1\vee x_2)\bar{x}_3)\to(x_1\equiv x_3)$ |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 |

Тепер ДДНФ функції зобразимо у такий спосіб:
$f(x_1,x_2,x_3)=\bar{x}_1\bar{x}_2\bar{x}_3\vee\bar{x}_1\bar{x}_2x_3\vee\bar{x}_1x_2\bar{x}_3\vee\bar{x}_1x_2x_3\vee x_1\bar{x}_2x_3\vee x_1x_2x_3$.

А відтак ДКНФ функції набуде вигляду
$f(x_1,x_2,x_3)=(\bar{x}_1\vee x_2\vee x_3)(\bar{x}_1\vee\bar{x}_2\vee x_3)$.

**Завдання для самостійної роботи**

1. Побудуйте ДДНФ і ДКНФ для функцій трьох змінних $f_1,\ldots,f_7$, заданих таблицями істинності:

Таблиця 3.7

| $x_1$ | $x_2$ | $x_3$ | $f_1$ | $f_2$ | $f_3$ | $f_4$ | $f_5$ | $f_6$ | $f_7$ |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 |

2. Логічну функцію зобразіть булевою формулою у вигляді ДДНФ і ДКНФ:

а) $f(x_1,x_2,x_3)=(\bar{x}_1\oplus x_2)(x_3\oplus 1)$; б) $f(x_1,x_2,x_3)=x_1\mid\overline{(x_3\vee x_2)}$;

в) $f(x_1,x_2,x_3)=(x_1\downarrow x_2)(x_3\oplus x_1)$; г) $f(x_1,x_2,x_3)=x_3\oplus(x_1\bar{x}_2)$.

## 3.5. Еквівалентні перетворення

***Еквівалентні перетворення*** – перетворення, що використовують закони алгебри логіки, наведені у п. 3.3. Еквівалентні перетворення застосовують для спрощення формул і доведення еквівалентності формул. У разі проведення еквівалентних перетворень необхідно дотримуватися *правила підстановки* і *правила заміни. Підставляючи* формулу замість змінної, усі входження змінної в початкове співвідношення мають бути одночасно замінені цією формулою.
*Правило заміни* дозволяє замінювати підформулу у формулі на еквівалентну.

***Означення* 3.6.** *Елементарною кон'юнкцією* називають кон'юнкцію змінних або їх заперечень, у якій кожна змінна зустрічається не більше одного разу.

***Означення* 3.7.** *Диз'юнктивною нормальною формою (ДНФ)* називають формулу, що має вигляд диз'юнкції елементарних кон'юнкцій.
Тобто ДДНФ – це ДНФ, кожна елементарна кон'юнкція якої містить усі змінні із запереченням або без нього.

***Зведення булевої функції до ДНФ*** полягає в тому, що:

а) усі заперечення необхідно встановити над змінними, застосовуючи подвійне заперечення і закони де Моргана;

б) розкрити дужки, використовуючи закони асоціативності і дистрибутивності;

в) видалити зайві кон'юнкції і повторення змінних у кон'юнкціях за допомогою законів ідемпотентності, протиріччя і виключення третього;

г) видалити константи, використовуючи властивості констант.

Процедура зведення ДНФ до ДДНФ полягає у застосуванні закону склеювання в оберненому напрямку для кон'юнкцій, які містять не всі змінні. Інший же спосіб одержання ДДНФ включає такі етапи: за ДНФ булевої функції будуємо таблицю істинності, на підставі якої ми можемо записати ДДНФ або ДКНФ.

Наприклад, побудуємо ДНФ функції $f(x,y,z)=x\bar{y}\vee z(\bar{x}\vee y\bar{z})$:

$$f(x,y,z)=x\bar{y}\vee z(\bar{x}\vee y\bar{z})=x\bar{y}\vee(\bar{x}z\vee yz\bar{z})=x\bar{y}\vee(\bar{x}z\vee y0)=x\bar{y}\vee(\bar{x}z\vee 0)=x\bar{y}\vee\bar{x}z.$$

Побудуємо ДНФ функції $f(x,y,z)=y\vee x(\bar{y}\vee\bar{x}\bar{z})\overline{(y(\bar{x}\vee z)\vee xy)}$:

$$f(x,y,z)=y\vee x(\bar{y}\vee\bar{x}\bar{z})\overline{(y(\bar{x}\vee z)\vee xy)}=y\vee(x\bar{y}\vee x\bar{x}\bar{z})\overline{(y(\bar{x}\vee z))}\,\overline{xy}=$$

$$=y\vee(x\bar{y}\vee 0z)(\bar{y}\vee\overline{(\bar{x}\vee z)})\cdot(\bar{x}\vee\bar{y})=y\vee x\bar{y}(\bar{y}\vee\bar{\bar{x}}\bar{z})(\bar{x}\vee\bar{y})=$$

$$=y\vee x\bar{y}(\bar{y}\vee x\bar{z})(\bar{x}\vee\bar{y})=y\vee x\bar{y}(\bar{x}\bar{y}\vee x\bar{x}\bar{z}\vee\bar{y}\bar{y}\vee x\bar{y}\bar{z})=$$

$$=y\vee x\bar{y}(\bar{x}\bar{y}\vee 0z\vee\bar{y}\vee x\bar{y}\bar{z})=y\vee x\bar{y}(\bar{x}\bar{y}\vee\bar{y}\vee x\bar{y}\bar{z})=y\vee(x\bar{x}\bar{y}\bar{y}\vee x\bar{y}\bar{y}\vee xx\bar{y}\bar{y}\bar{z})$$

$$=y\vee(x\bar{y}\vee x\bar{y}\bar{z})=y\vee x\bar{y}(1\vee\bar{z})=y\vee x\bar{y}1=y\vee x\bar{y}=x\vee y.$$

Розглянемо приклад спрощення формули, вдаючись до еквівалентних перетворень. Нехай задана логічна функція $f(x,y,z)=x\vee y\bar{z}\vee x\bar{y}\bar{z}$. Результат спрощення формули матиме такий вигляд:

$$f(x,y,z)=x\vee y\bar{z}\vee x\bar{y}\bar{z}=x\vee\bar{z}(y\vee x\bar{y})=x\vee\bar{z}(x\vee y)=x\vee x\bar{z}\vee y\bar{z}=x(1\vee\bar{z})\vee y\bar{z}=$$

$$=x1\vee y\bar{z}=x\vee y\bar{z}.$$

Побудуємо ДДНФ функції з використанням закону оберненого склеювання:

$f(x,y,z) = x \vee y\bar{z} = x(y \vee \bar{y})(z \vee \bar{z}) \vee (x \vee \bar{x})y\bar{z} = \quad = xyz \vee x\bar{y}z \vee xy\bar{z} \vee x\bar{y}\bar{z} \vee xy\bar{z} \vee \bar{x}y\bar{z} =$
$\bar{x}y\bar{z} \vee x\bar{y}\,\bar{z} \vee x\bar{y}z \vee xy\bar{z} \vee xyz$.

***Означення* 3.8.** *Елементарною диз'юнкцією* називають диз'юнкцію змінних або їх заперечень, у якій кожна змінна зустрічається не більше одного разу.

***Означення* 3.9.** *Кон'юнктивною нормальною формою (КНФ)* називають формулу, що має вигляд кон'юнкції елементарних диз'юнкцій.
Тобто ДКНФ – це КНФ, кожна елементарна диз'юнкція якої включає всі змінні із запереченням або без нього.

***Зведення булевої функції до КНФ*** полягає в такому:

а) необхідно застосувати до формули правило подвійного заперечення і звести до ДНФ;

б) за допомогою законів де Моргана позбавитися подвійного заперечення і перетворити заперечення елементарних кон'юнкцій на елементарні диз'юнкції.

Наприклад, побудуємо КНФ функції $f(x,y,z) = xz \vee y\bar{z} \vee x$:

$$f(x,y,z) = xz \vee y\bar{z} \vee x = \overline{\overline{xz \vee y\bar{z} \vee x}} = \overline{\overline{xz} \cdot \overline{y\bar{z}} \cdot \bar{x}} = \overline{(\bar{x} \vee \bar{z})(\bar{y} \vee z)\bar{x}} =$$
$$= \overline{(\bar{x}\,\bar{y} \vee \bar{y}\,\bar{z} \vee \bar{x}z \vee z\bar{z})\bar{x}} = \overline{(\bar{x}\,\bar{y} \vee \bar{y}\,\bar{z} \vee \bar{x}z)\bar{x}} = \overline{\bar{x}\,\bar{x}\,\bar{y} \vee \bar{x}\,\bar{y}\,\bar{z} \vee \bar{x}\,\bar{x}z} = \overline{\bar{x}\,\bar{y} \vee \bar{x}\,\bar{y}\,\bar{z} \vee \bar{x}z} =$$
$$= \overline{\bar{x}\,\bar{y}(1 \vee \bar{z}) \vee \bar{x}z} = \overline{\bar{x}\,\bar{y} \vee \bar{x}z} = (x \vee y)(x \vee \bar{z}).$$

**Завдання для самостійної роботи**

1. Побудуйте ДНФ функції:

а) $f(x,y,z) = \overline{(x \vee \bar{y}) \vee \overline{x\bar{z}} \vee y}$; б) $f(x,y,z) = \overline{xy} \vee \overline{x\bar{z} \vee y} \vee \overline{x\bar{y}z}$;

в) $f(x,y,z) = \overline{\overline{x \vee \bar{z}} \vee (x\bar{y} \vee z)} \vee \bar{x}yz$; г) $f(x,y,z) = \overline{\overline{xyz} \vee \overline{(\bar{x} \vee y)x\bar{z}}} \vee y$;

д) $f(x,y,z) = \overline{\bar{y}z \vee \bar{x}} \vee \overline{x\bar{y} \vee \overline{(y \vee \bar{z})}}$; е) $f(x,y,z) = \overline{\overline{xy} \vee \overline{(\bar{x}z \vee y)}} \vee \bar{x}yz$;

є) $f(x,y,z) = \overline{x\bar{y} \vee \overline{(yz \vee \bar{x})}} \vee \overline{xy}$; ж) $f(x,y,z) = x \vee \overline{y \vee \overline{(x\bar{z} \vee \bar{x}y)} \vee z}$.

2. Спростіть формулу, використовуючи: а) еквівалентні перетворення; б) обернене склеювання. Одержіть ДДНФ функції за допомогою табличного подання функції $f(x,y,z)$.

а) $f(x,y,z) = \bar{x}z \vee xyz \vee \overline{xz}$; б) $f(x,y,z) = yz \vee \bar{x}z \vee xyz$;

в) $f(x,y,z) = x\bar{y} \vee x\bar{z} \vee yz \vee \bar{x}yz$; г) $f(x,y,z) = \bar{y}z \vee x\bar{y} \vee xz \vee xy\bar{z}$;

д) $f(x,y,z) = xy \vee yz \vee xz \vee x\bar{y}$; е) $f(x,y,z) = \bar{x}y \vee xy\bar{z} \vee xz \vee yz$;

є) $f(x,y,z) = xz \vee \bar{x}y \vee y\bar{z} \vee \bar{x}y\bar{z}$; ж) $f(x,y,z) = xy \vee y\bar{z} \vee x\bar{y}z$.

3. Одержіть КНФ функції $f(x,y,z)$ за правилом зведення ДНФ до КНФ:

а) $f(x,y,z) = xy \vee x\bar{z} \vee z$; б) $f(x,y,z) = xy \vee \bar{y}z \vee \overline{x\bar{z}}$;

в) $f(x,y,z) = xy \vee \overline{y\bar{z}} \vee \overline{x\bar{y}}$; г) $f(x,y,z) = x\bar{y} \vee \bar{x}yz$;

д) $f(x,y,z) = \bar{x} \vee yz \vee \bar{x}y\bar{z}$; е) $f(x,y,z) = xy \vee \bar{y}z$.

# 4. ФУНКЦІОНАЛЬНА ПОВНОТА СИСТЕМ БУЛЕВИХ ФУНКЦІЙ

## 4.1. Поняття замкненого класу. Властивості замикання

***Означення* 4.1.** Множину логічних функцій M називають замкненим класом, якщо:

1) для будь-якої функції $f \in M$ будь-яка заміна змінних не виводить функцію з множини $M$;
2) будь-яка суперпозиція функцій $f_1, f_2, \dots, f_n \in M$ також належить множині $M$.

***Означення* 4.2.** Замиканням системи логічних функцій $F$ (позначають $[F]$) називають множину всіх функцій, що реалізуються формулами над F.

***Властивості замикання:***

1. $[[F]] = [F]$;
2. $F \subset [F]$;
3. $F_1 \subset F_2 \Rightarrow [F_1] \subset [F_2]$;
4. $[F_1 \cup F_2] \subset [F_1] \cup [F_2]$.

***Означення* 4.3.** Клас функцій $F$ називають замкненим, якщо клас функцій $F$ збігається з його замиканням, тобто $[F] = F$.

*Існує 5 найбільш важливих замкнених класів*: клас функцій, що зберігають значення 0; клас функцій, що зберігають значення 1; клас самодвоїстих функцій; клас монотонних функцій; клас лінійних функцій.

Розглянемо більш детально кожен з цих класів.

## 4.2. Клас функцій, що зберігають значення 0

***Означення* 4.4.** Функцію, яка на наборі змінних зі всіх нулів набуває значення 0, називають функцією, що зберігає значення 0.

Клас функцій, що зберігає значення 0, позначають $T_0$ та зображують у такий спосіб:

$$T_0 := \{ f \mid \; f(0\,0 \dots 0) = 0 \}.$$

До класу $T_0$, наприклад, належать функції $0,\ x,\ x \wedge y,\ x \vee y$ та не належать функції $1,\ \bar{x}\ ,\ x \equiv y$.

***Теорема* 4.1.** Клас функцій, що зберігають значення 0, замкнений.

***Доведення.*** Для доведення замкненості класу $T_0$ покажемо, що суперпозиція функцій з класу $T_0$ також належать до $T_0$.

Нехай функції $f_0, f_1, \dots, f_k \in T_0$. Побудуємо суперпозицію з цих функцій у вигляді $\Phi(x_1, x_2, \dots, x_n) = f_0(f_1(x_1 \dots x_n), \dots, f_k(x_1 \dots x_n))$ та покажемо, що $\Phi(x_1, x_2, \dots, x_n)$ також належить до класу $T_0$.

Розглянемо $\Phi(0,0,\dots,0)$: $\Phi(0,0,\dots,0) = f_0(f_1(0 \dots 0) f_2(0 \dots 0) \dots f_k(0 \dots 0)) = f_0(0 \dots 0) = 0$. Тобто відповідно до означення 4.4 $\Phi(x_1, x_2, \dots, x_n) \in T_0$. Теорему доведено.

## 4.3. Клас функцій, що зберігають значення 1

*Означення* **4.5.** Функцію, яка на наборі змінних з усіх одиниць, набуває значення 1, називають функцією, що зберігає значення 1.

Клас функцій, що зберігає значення 1, позначають $T_1$ та зображують у такий спосіб:

$$T_1 := \{f \mid \; f(11...1) = 1\}.$$

До класу $T_1$, наприклад, належать функції 1, $x$, $x \wedge y$, $x \vee y$, $x \to y$, $x \equiv y$ та не належать функції 0, $\bar{x}$, $x \oplus y$, $x \downarrow y$.

***Теорема* 4.2.** Клас функцій, що зберігають значення 1, замкнений.

***Доведення.*** Для доведення замкненості класу $T_1$ покажемо, що суперпозиція функцій з класу $T_1$ також належать до $T_1$.

Нехай функції $f_0, f_1, ..., f_k \in T_1$. Побудуємо суперпозицію з цих функцій у вигляді $\Phi(x_1, x_2, ..., x_n) = f_0(f_1(x_1 ... x_n), ...., f_k(x_1 ... x_n))$ та покажемо, що $\Phi(x_1, x_2, ..., x_n)$ також належить до класу $T_1$.

Розглянемо $\Phi(11...1)$ : $\Phi(11...1) = f_0(f_1(11...1) f_2(11...1) ... f_k(11...1)) =$ $= f_0(11...1) = 1$.

Тобто відповідно до означення 4.5 $\Phi(x_1, x_2, ..., x_n) \in T_1$. Теорему доведено.

## 4.4. Поняття двоїстої функції

*Означення* **4.6.** Функцію $f^*(x_1, ..., x_n)$ називають *двоїстою* до функції $f(x_1, ..., x_n)$, якщо справедливе співвідношення

$$f^*(x_1, ..., x_n) = \bar{f}(\bar{x}_1, ..., \bar{x}_n).$$

Операція побудови двоїстої функції складається з інвертування функції, всі змінні якої теж інвертовані. Для констант інверсію застосовують тільки до функціональних символів 0 або 1.

*Означення* **4.7.** Набір $(\bar{x}_1, \bar{x}_2 ..., \bar{x}_n)$ називають *протилежним* набору $(x_1, x_2 ..., x_n)$ і позначають $\bar{\bar{x}}$.

Функція, яка є двоїста до заданої, набуває на протилежних наборах $\bar{x}$ значення, протилежні значенням $f(x)$. Отже, таблицю істинності двоїстої функції $f^*(x_1, ..., x_n)$ можна отримати з таблиці для функції $f(x_1, ..., x_n)$ шляхом перевертання стовпця значень функції (ця дія відповідає інверсії змінних) та інвертування значень функції (ця дія відповідає інвертуванню функції).

Наприклад, знайдемо функцію, двоїсту до $x_1 \wedge x_2$. Спочатку побудуємо таблицю істинності для заданої функції. Останній рядок табл. 4.1 побудовано шляхом перевертання стовпця значень функції $x_1 \wedge x_2$ (попередній рядок табл. 4.1) та інвертування значень функції.

Відзначимо, що останній рядок таблиці відповідає значенням функції диз'юнкції, тому є справедливий такий ланцюжок тотожностей:

$(x_1 \wedge x_2)^* = \overline{\bar{x}_1 \wedge \bar{x}_2} = x_1 \vee x_2$, відповідно до якого, диз'юнкція виявляється функцією, двоїстою до кон'юнкції.

Таблиця 4.1

| | | | | |
|---|---|---|---|---|
| $x_1$ | 0 | 0 | 1 | 1 |
| $x_2$ | 0 | 1 | 0 | 1 |
| $x_1 \wedge x_2$ | 0 | 0 | 0 | 1 |
| $(x_1 \wedge x_2)^* = \overline{\bar{x}_1 \wedge \bar{x}_2}$ | 0 | 1 | 1 | 1 |

Зауважимо, що відношення двоїстості між функціями симетричне, тобто якщо $f^*$ двоїста до $f$, то $f$, навпаки, двоїста до $f^*$:

$$\bar{f}(\bar{x}_1,\ldots,\bar{x}_n) = \bar{\bar{f}}^*(\bar{\bar{x}}_1,\ldots,\bar{\bar{x}}_n) = f^*(x_1,\ldots,x_n).$$

***Принцип двоїстості.*** Якщо у формулі $F$, що описує функцію $f$, всі символи функції замінити на символи відповідних двоїстих функцій, то отримана формула $F^*$ буде описувати функцію $f^*$, яка є двоїста до вихідної функції $f$.

***Принцип двоїстості в булевій алгебрі.*** Якщо у формулі $F$, що описує функцію $f$, всі кон'юнкції замінити на диз'юнкції, а диз'юнкції – на кон'юнкції, константи 1 на 0, а константи 0 на 1, то одержимо формулу $F^*$, що буде описувати функцію $f^*$, двоїсту до вихідної функції $f$.

Справедливе твердження: якщо функції тотожні між собою $(f_1 = f_2)$, то і двоїсті їм функції також тотожні між собою $(f_1^* = f_2^*)$.

Наприклад, побудуємо формулу, що описує функцію, двоїсту до $x_1 \to x_2$:

$$f = x_1 \to x_2 = \bar{x}_1 \vee x_2 = \overline{x_1 \wedge \bar{x}_2}\text{, тоді } f^* = \overline{\bar{\bar{x}}_1 \wedge \bar{x}_2} = \bar{x}_1 \wedge x_2.$$

Побудуємо функцію, двоїсту до функції $f(x_1,x_2) = x_1x_2 \vee \bar{x}_1\bar{x}_2$:

$$f^*(x_1,x_2) = ((x_1 \wedge x_2) \vee (\bar{x}_1 \wedge \bar{x}_2))^* = \overline{(\bar{x}_1 \wedge \bar{x}_2) \vee (\bar{\bar{x}}_1 \wedge \bar{\bar{x}}_2)} =$$
$$= \left(\overline{\bar{x}_1 \wedge \bar{x}_2}\right) \wedge \left(\overline{x_1 \wedge x_2}\right) = (x_1 \vee x_2) \wedge (\bar{x}_1 \vee \bar{x}_2).$$

**Завдання для самостійної роботи**

1. Для функції $f(x_1,x_2,x_3)$ знайдіть двоїсту:

а) $f(x_1,x_2,x_3) = x_1x_2\bar{x}_3 \vee \overline{x_1 \vee x_2x_3}$;

б) $f(x_1,x_2,x_3) = (x_2 \vee x_3) \vee (x_2\bar{x}_3 \vee x_1x_3)$;

в) $f(x_1,x_2,x_3) = x_1\bar{x}_2 \vee x_2x_3 \vee x_1\bar{x}_2x_3$;

г) $f(x_1,x_2,x_3) = (x_1 \vee \bar{x}_2)(x_1\bar{x}_3 \vee x_2x_3)$.

## 4.5. Клас самодвоїстих функцій

***Означення* 4.8.** Функцію, двоїсту до самої себе, називають *самодвоїстою*, тобто якщо виконується умова $f(x_1,...,x_n)=f^*(x_1,...,x_n)$, то функція $f(x_1,...,x_n)$ є *самодвоїста*.

Клас самодвоїстих функцій позначають $T^*$ та зображують у такий спосіб:

$$T^* := \{f| \quad f(x_1\, x_2 ... x_n) = f^*(x_1\, x_2 ... x_n)\}.$$

До класу $T^*$ , наприклад, належать функції $x,\ \bar{x},\ x \oplus y \oplus 1$.

***Теорема* 4.3.** Клас $T^*$ самодвоїстих функцій є замкнений.

***Доведення.*** Для доведення замкненості класу $T^*$ покажемо, що суперпозиція функцій з класу $T^*$ також належать до $T^*$.

Нехай функції $f_0, f_1, ..., f_k \in T^*$. Побудуємо суперпозицію з цих функцій у вигляді $\Phi(x_1, x_2, ..., x_n) = f_0(f_1(x_1 ... x_n), ...., f_k(x_1 ... x_n))$ та покажемо, що $\Phi^*(x_1, x_2, ..., x_n) = \Phi(x_1, x_2, ..., x_n)$.

$$\Phi^*(x_1,x_2,...,x_n) = f_0^*(f_1^*(x_1,x_2,...,x_n) f_2^*(x_1,x_2,...,x_n) ... f_k^*(x_1,x_2,...,x_n)) =$$
$$= f_0(x_1,x_2,...,x_n) = \Phi(x_1,x_2,...,x_n).$$

Тобто відповідно до означення 4.8 $\Phi(x_1, x_2, ..., x_n) \in T^*$. Теорему доведено.

## 4.6. Клас монотонних функцій

***Означення* 4.9.** Для двох наборів $\alpha = (\alpha_1 \ldots \alpha_n)$ та $\beta = (\beta_1 \ldots \beta_n)$ виконується відношення передування, якщо $\alpha_1 \le \beta_1, \ldots, \alpha_n \le \beta_n$, тобто значення кожної компоненти набору $\alpha$ не більше значення відповідної компоненти в наборі $\beta$.

Наприклад, набори $\alpha = (0,1,0,1)$ та $\beta = (1,1,0,1)$ знаходяться у відношенні передування. Набори (0,1,0) та (1,0,0) не знаходяться у відношенні передування, тобто вони непорівнянні.

***Означення* 4.10.** Функцію $f(x_1 \ldots x_n)$ називають монотонною, якщо для будь-яких двох наборів $\alpha$ і $\beta$, що знаходяться у відношенні передування, виконується нерівність $f(\alpha) \le f(\beta)$.

Клас монотонних функцій позначають $T_M$ та зображують у такий спосіб:

$$T_M := \{f| \quad f(\alpha_1\, \alpha_2 ... \alpha_n) \le f(\beta_1\, \beta_2 ... \beta_n), \text{ якщо } \quad \alpha \le \beta\}.$$

Серед булевих функцій однієї та двох змінних кон'юнкція, диз'юнкція, константи 0 та 1 – монотонні, а функції заперечення, імплікації, еквівалентності, штрих Шеффера, стрілка Пірса – немонотонні. Наприклад, імплікація на наборі (0,0) дорівнює 1, а на наборі (1,0) – 0; оскільки набір (0,0) передує набору (1,0), а для значень функції виконується нерівність $f(0,0) \ge f(1,0)$, то властивість монотонності для імплікації не виконується.

***Теорема* 4.4.** Клас монотонних функцій $T_M$ є замкнений.

***Доведення.*** Нехай функції $f_0, f_1, ..., f_k \in T_M$. Побудуємо суперпозицію з цих функцій у вигляді $\Phi(x_1, x_2, ..., x_n) = f_0(f_1(x_1 ... x_n), ...., f_k(x_1 ... x_n))$ та покажемо, що для наборів $\alpha$ та $\beta$, таких що $\alpha \le \beta$, $\Phi(\alpha_1, \alpha_2, ..., \alpha_n) \le (\beta_1, \beta_2, ..., \beta_n)$. Оскільки функції $f_0, f_1, ..., f_n \in T_M$ монотонні, то $f_1(\alpha) \le f_1(\beta), ......, f_k(\alpha) \le f_k(\beta)$, тому набори значень для суперпозиції $\Phi(x_1, x_2, ..., x_n)$ впорядковані. Зважаючи на монотонність функції $f_0$, маємо: $\Phi(\alpha_1, \alpha_2, ..., \alpha_n) \le \Phi(\beta_1, \beta_2, ..., \beta_n)$. Теорему доведено.

## 4.7. Алгебра Жегалкіна. Способи побудови полінома Жегалкіна

***Означення* 4.11.** Алгебра булевих функцій, утворених за допомогою функцій кон'юнкції та додавання за модулем 2, змінних та констант 0, 1, має назву алгебри Жегалкіна. В алгебрі Жегалкіна виконуються співвідношення:

1) $x \oplus y = y \oplus x$;
2) $x \wedge (y \oplus z) = x \wedge y \oplus x \wedge z$;
3) $x \oplus x = 0$;
4) $x \oplus 0 = x$.

З таблиці істинності для булевої функції додавання за модулем 2 виходить, що $\overline{x} = x \oplus 1$.

Операцію диз'юнкції можна виразити через $\oplus$ та $\wedge$ у такий спосіб:

$$x \vee y = \overline{\overline{x} \wedge \overline{y}} = (x \oplus 1) \wedge (y \oplus 1) \oplus 1 = (x \wedge y) \oplus x \oplus y \oplus 1 \oplus 1 = (x \wedge y) \oplus x \oplus y.$$

***Означення* 4.12.** Довільну формулу алгебри Жегалкіна, яка має вигляд суми (за модулем 2) кон'юнкцій булевих змінних, називають поліномом Жегалкіна.

У 1927 році І.І. Жегалкін запропонував зручне зображення булевих функції у вигляді полінома. В зарубіжній літературі таке зображення зазвичай називають алгебричною нормальною формою (АНФ). Поліном Жегалкіна називають канонічним, якщо у кожний доданок полінома Жегалкіна кожна змінна входить один раз та якщо поліном не містить однакових доданків.

***Теорема* 4.5.** Довільну булеву функцію зображують у вигляді канонічного полінома Жегалкіна єдиним чином.

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

Розглянемо, наприклад, побудову полінома Жегалкіна для функції $f(x_1, x_2, x_3) = \overline{x_1}\,\overline{x_2} \vee x_2(x_1 \oplus x_3)$ за допомогою методу невизначених коефіцієнтів:

Таблиця 4.2

| Зображення функції у вигляді полінома Жегалкіна $f(x_1,x_2,x_3)=c_0\oplus c_1x_1\oplus c_2x_2\oplus c_3x_3\oplus c_{12}x_1x_2\oplus c_{23}x_2x_3\oplus c_{13}x_1x_3\oplus c_{123}x_1x_2x_3$ | | | | |
|---|---|---|---|---|
| $x_1$ | $x_2$ | $x_3$ | $f$ | Відповідне рівняння для визначення коефіцієнта |
| 0 | 0 | 0 | 1 | $c_0=1$ |
| 0 | 0 | 1 | 1 | $c_0+c_3=1$ |
| 0 | 1 | 0 | 0 | $c_0+c_2=0$ |
| 0 | 1 | 1 | 1 | $c_0+c_2+c_3+c_{23}=1$ |
| 1 | 0 | 0 | 0 | $c_0+c_1=0$ |
| 1 | 0 | 1 | 0 | $c_0+c_1+c_3+c_{13}=0$ |
| 1 | 1 | 0 | 1 | $c_0+c_1+c_2+c_{12}=1$ |
| 1 | 1 | 1 | 0 | $c_0+c_1+c_2+c_3+c_{12}+c_{13}+c_{23}+c_{123}=0$ |

Розв'язання системи алгебричних рівнянь, побудованої в останньому стовпчику табл. 4.2, дозволяє отримати значення всіх невідомих коефіцієнтів:

$$\begin{cases} \underline{c_0=1} \\ c_0+c_3=1 \\ c_0+c_2=0 \\ c_0+c_2+c_3+c_{23}=1 \\ c_0+c_1=0 \\ c_0+c_1+c_3+c_{13}=0 \\ c_0+c_1+c_2+c_{12}=1 \\ c_0+c_1+c_2+c_3+ \\ +c_{12}+c_{13}+c_{23}+c_{123}=0 \end{cases} \Rightarrow \begin{cases} \underline{c_0=1} \\ 1+c_3=1 \quad \underline{c_3=0} \\ 1+c_2=0 \quad \underline{c_2=1} \\ 1+c_2+c_3+c_{23}=1 \\ 1+c_1=0 \quad \underline{c_1=1} \\ 1+c_1+c_3+c_{13}=0 \\ 1+c_1+c_2+c_{12}=1 \\ 1+c_1+c_2+c_3+ \\ +c_{12}+c_{13}+c_{23}+c_{123}=0 \end{cases} \Rightarrow$$

$$\begin{cases} \underline{c_0=1} \\ \underline{c_3=0} \\ \underline{c_2=1} \\ 1+1+0+c_{23}=1 \quad \underline{c_{23}=1} \\ \underline{c_1=1} \\ 1+1+0+c_{13}=0 \quad \underline{c_{13}=0} \\ 1+1+1+c_{12}=1 \quad \underline{c_{12}=0} \\ 1+c_1+c_2+c_3+c_{12}+c_{13}+c_{23}+c_{123}=0 \end{cases} \Rightarrow \begin{cases} \underline{c_0=1} \\ \underline{c_3=0} \\ \underline{c_2=1} \\ \underline{c_{23}=1} \\ \underline{c_1=1} \\ \underline{c_{13}=0} \\ \underline{c_{12}=0} \\ 1+1+1+0+ \\ +0+0+1+c_{123}=0 \quad \underline{c_{123}=0.} \end{cases}$$

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

$$f(x_1, x_2, x_3) = 1 \oplus x_1 \oplus x_2 \oplus x_2 x_3.$$

Побудувати поліном Жегалкіна можна також за допомогою еквівалентних перетворень, для цього необхідно замінити усі операції, які відрізняються від додавання за модулем 2 та кон'юнкції, на еквівалентні. Наприклад, за допомогою еквівалентних перетворень функцію $f(x_1, x_2, x_3) = \overline{x_1 x_2} \vee x_2(x_1 \oplus x_3)$ можна зобразити у вигляді полінома Жегалкіна у такий спосіб:

$$f(x_1,x_2,x_3) = \overline{x_1}\,\overline{x_2} \vee x_2(x_1 \oplus x_3) = \overline{x_1}\underbrace{\overline{x_2}x_2}_{0}(x_1 \oplus x_3) \oplus \overline{x_1}\,\overline{x_2} \oplus x_2(x_1 \oplus x_3) =$$
$$= 0 \oplus (x_1 \oplus 1)(x_2 \oplus 1) \oplus x_2 x_1 \oplus x_2 x_3 = x_2 x_1 \oplus x_2 \oplus x_1 \oplus 1 \oplus x_2 x_1 \oplus x_2 x_3 =$$
$$= 1 \oplus x_1 \oplus x_2 \oplus x_2 x_3 \oplus \underbrace{x_2 x_1 \oplus x_2 x_1}_{0} = 1 \oplus x_1 \oplus x_2 \oplus x_2 x_3.$$

## 4.8. Клас лінійних функцій

***Означення* 4.13.** Функцію, зображену поліномом Жегалкіна вигляду $(\alpha_1 \wedge x_1) \oplus (\alpha_2 \wedge x_2) \oplus ... \oplus (\alpha_n \wedge x_n) \oplus \gamma$, де константи $\alpha_1, ..., \alpha_n, \gamma$ дорівнюють 0 або 1, називають **лінійною.**

Клас лінійних функцій позначають $T_L$ та зображують у такий спосіб:

$$T_L := \{f \mid \ f(x_1\ x_2 ... x_n) = \alpha_1 \wedge x_1 \oplus \alpha_2 \wedge x_2 \oplus ... \oplus \alpha_n \wedge x_n \oplus \gamma\}.$$

Наприклад, усі функції однієї змінної лінійні, функції двох змінних $x \oplus y$ та $x \equiv y$ є також лінійні.

***Теорема* 4.6.** Клас лінійних функцій $T_L$ є замкненим.

***Доведення.*** Нехай функції $f_0, f_1, ..., f_k \in T_L$. Побудуємо суперпозицію з цих функцій у вигляді $\Phi(x_1, x_2, ..., x_n) = f_0(f_1(x_1 \ldots x_n), ...., f_k(x_1 \ldots x_n))$ та покажемо, що вона є також лінійною функцією.

Оскільки
$$f_0(x_1\ x_2 ... x_n) = \alpha_1^0 \wedge x_1 \oplus \alpha_2^0 \wedge x_2 \oplus ... \oplus \alpha_n^0 \wedge x_n \oplus \gamma^0;$$
$$f_1(x_1\ x_2 ... x_n) = \alpha_1^1 \wedge x_1 \oplus \alpha_2^1 \wedge x_2 \oplus ... \oplus \alpha_n^1 \wedge x_n \oplus \gamma^1;$$
$$\vdots$$
$$f_k(x_1\ x_2 ... x_n) = \alpha_1^k \wedge x_1 \oplus \alpha_2^k \wedge x_2 \oplus ... \oplus \alpha_n^k \wedge x_n \oplus \gamma^k,$$

то
$$\Phi(x_1\ x_2 ... x_n) = \gamma^0 \oplus \alpha_1^0 \wedge (\alpha_1^1 \wedge x_1 \oplus \alpha_2^1 \wedge x_2 \oplus ... \oplus \alpha_n^1 \wedge x_n \oplus \gamma^1) \oplus$$
$$\oplus \alpha_2^0 \wedge (\alpha_1^2 \wedge x_1 \oplus \alpha_2^2 \wedge x_2 \oplus ... \oplus \alpha_n^2 \wedge x_n \oplus \gamma^2) \oplus ...$$
$$... \oplus \alpha_k^0 \wedge (\alpha_1^k \wedge x_1 \oplus \alpha_2^k \wedge x_2 \oplus ... \oplus \alpha_n^k \wedge x_n \oplus \gamma^k) = C_1 \wedge x_1 \oplus C_2 \wedge x_2 \oplus ... \oplus C_n \wedge x_n \oplus C_0.$$

Отже, $\Phi(x_1, x_2, ..., x_n) \in T_L$. Теорему доведено.

## 4.9. Функціональна повнота систем булевих функцій

***Означення* 4.14.** Систему булевих функцій називають функціонально повною, якщо будь-яку булеву функцію можна зобразити у вигляді формули, що складається тільки з функцій цієї системи.

Існує ряд функціонально повних систем булевих функцій, наприклад $\{\wedge,\vee,^{-}\}$, $\{\wedge,^{-}\}$, $\{\vee,^{-}\}$, $\{\ |\ \}$, $\{\downarrow\}$, $\{\wedge,\oplus,1\}$, $\{\rightarrow,^{-}\}$, $\{\vee,^{-},\oplus\}$, $\{\wedge,^{-},\oplus\}$.

Найбільш важливим и використовуваним є базис $\{\wedge,\vee,^{-}\}$. Система функцій булевої алгебри $F_0=\{\wedge,\vee,^{-}\}$ функціонально повна, оскільки за теоремою 3.2 будь-яку булеву функцію можна зобразити за допомогою кон'юнкції, диз'юнкції і заперечення, тобто будь-яку логічну функцію можна подати у вигляді формули, що складається тільки з функцій цієї системи.

Розглянемо систему $F_1=\{\wedge,^{-}\}$, яку називають *кон'юнктивною системою Буля*. Скориставшись законами де Моргана і подвійного заперечення, з її функцій можна виразити диз'юнкцію: $x\vee y=\overline{\bar{x}\wedge\bar{y}}$.

Розглянемо систему $F_2=\{\vee,^{-}\}$, яку називають *диз'юнктивною системою Буля*. Застосовуючи закони де Моргана і подвійного заперечення, з її функцій можна виразити кон'юнкцію: $x\wedge y=\overline{\bar{x}\vee\bar{y}}$.

З повноти системи $F_0$ і можливості подання однієї з функцій через інші робимо висновок про повноту систем $F_1$ і $F_2$. У цьому сенсі система $F_0$ виявляється надлишковою, тому що вона зберігає властивості повноти навіть за вилучення з неї диз'юнкції або кон'юнкції.

Наприклад, зобразимо булеву формулу $z(x\vee y)\vee\bar{x}y$ у базисах $F_1=\{\wedge,^{-}\}$ і $F_2=\{\vee,^{-}\}$ у вигляді

$$z(x\vee y)\vee\bar{x}y=z\cdot\overline{\bar{x}\,\bar{y}}\vee\bar{x}y=\overline{\overline{z\cdot\overline{\bar{x}\,\bar{y}}}\cdot\overline{\bar{x}y}};$$

$$z(x\vee y)\vee\bar{x}y=\overline{\bar{z}\vee\overline{(x\vee y)}}\vee\overline{\bar{\bar{x}}\vee\bar{y}}=\overline{\bar{z}\vee\overline{(x\vee y)}}\vee\overline{x\vee\bar{y}}.$$

З наведеного прикладу бачимо, що в базисі $F_0=\{\wedge,\vee,^{-}\}$ логічні функції мають більш прості формули. Застосування ненадлишкових базисів під час запису формул призводить до більш громіздкого вигляду виразів, що описують логічні функції. Тому в кожному конкретному випадку доводиться вибирати між кількістю використовуваних операцій і громіздкістю одержуваних виразів.

Розглянемо систему $F_3=\{\ |\ \}$, яку називають *системою Шеффера*. Ця система функціонально повна, оскільки зводиться до системи $F_1$ в такий спосіб: заперечення зображують у вигляді $\bar{x}=x\,|\,x$, кон'юнкцію – $xy=\overline{x\,|\,y}$.

Наприклад, зобразимо булеву формулу $z(x\vee y)\vee\bar{x}y$ в базисі $F_3=\{\ |\ \}$ у такий спосіб:

$$z(x\vee y)\vee\bar{x}y=\overline{\overline{z\cdot\overline{\bar{x}\,\bar{y}}}\cdot\overline{\bar{x}y}}=\overline{\overline{z\cdot\overline{\overline{(x|x)}\cdot\overline{(y|y)}}}\cdot\overline{\overline{(x|x)}\cdot y}}=\overline{\overline{z\cdot\overline{\overline{((x|x)|(y|y))}}}\cdot\overline{\overline{((x|x)|y)}}}=$$

$$\overline{\overline{z\cdot((x|x)|(y|y))}\cdot((x|x)|y)}=\overline{\overline{z|((x|x)|(y|y))}\cdot((x|x)|y)}=$$

$\overline{z\mid((x\mid x)\mid(y\mid y))\cdot((x\mid x)\mid y)}=\overline{(z\mid((x\mid x)\mid(y\mid y)))\mid((x\mid x)\mid y)}=$
$=(z\mid((x\mid x)\mid(y\mid y)))((x\mid x)\mid y)$.

Розглянемо систему $F_4=\{\downarrow\}$, яку називають *системою Пірса*. Ця система функціонально повна, тому що зводиться до системи $F_2$ у такий спосіб: заперечення зображують у вигляді $\bar{x}=x\downarrow x$; диз'юнкцію – $x\vee y=\overline{x\downarrow y}$.

Наприклад, зобразимо булеву формулу $z(x\vee y)\vee\bar{x}y$ в базисі $F_4=\{\downarrow\}$:

$$z(x\vee y)\vee\bar{x}y=\overline{\bar{z}\vee\overline{(x\vee y)}}\vee\overline{x\vee\bar{y}}=\overline{(z\downarrow z)\vee\overline{(x\downarrow y)}}\vee\overline{x\vee(y\downarrow y)}=$$
$$=\overline{(z\downarrow z)\vee(x\downarrow y)}\vee\overline{\overline{x\downarrow(y\downarrow y)}}=\overline{\overline{(z\downarrow z)\downarrow(x\downarrow y)}}\vee(x\downarrow(y\downarrow y))=$$
$$=((z\downarrow z)\downarrow(x\downarrow y))\vee(x\downarrow(y\downarrow y))=\overline{\overline{((z\downarrow z)\downarrow(x\downarrow y))\downarrow(x\downarrow(y\downarrow y))}}=$$
$$=(((z\downarrow z)\downarrow(x\downarrow y))\downarrow(x\downarrow(y\downarrow y)))\downarrow(((z\downarrow z)\downarrow(x\downarrow y))\downarrow(x\downarrow(y\downarrow y))).$$

Розглянемо систему $F_5=\{\wedge,\oplus,1\}$, яку називають *системою Жегалкіна*. Ця система функціонально повна, тому що зводиться до системи $F_1$ у такий спосіб: заперечення зображують у вигляді $\bar{x}=x\oplus 1$; диз'юнкцію $x\vee y$ згідно із законами де Моргана та виразами для заперечень $\bar{x}=x\oplus 1$, $\bar{y}=y\oplus 1$ зображують у вигляді $x\vee y=(x\oplus 1)(y\oplus 1)\oplus 1=xy\oplus y\oplus x\oplus 1\oplus 1=xy\oplus x\oplus y$.

Наприклад, зобразимо булеву формулу $z(x\vee y)\vee\bar{x}y$ в базисі $F_5=\{\wedge,\oplus,1\}$. Для розв'язання скористаємося вже відомими співвідношеннями, а саме $x\cdot x=x$, $x\oplus x=0$, тоді

$$z(x\vee y)\vee\bar{x}y=z(xy\oplus x\oplus y)\vee(x\oplus 1)y=yz(xy\oplus x\oplus y)(x\oplus 1)\oplus z(xy\oplus x\oplus y)\oplus$$
$$\oplus y(x\oplus 1)=(xyyz\oplus xyz\oplus yyz)(x\oplus 1)\oplus xyz\oplus xz\oplus yz\oplus xy\oplus y=$$
$$=xxyyz\oplus xxyz\oplus xyyz\oplus xyyz\oplus xyz\oplus yyz\oplus xyz\oplus xz\oplus yz\oplus xy\oplus y=$$
$$=xyz\oplus xyz\oplus xyz\oplus xyz\oplus xyz\oplus yz\oplus xyz\oplus xz\oplus yz\oplus xy\oplus y=xy\oplus xz\oplus y.$$

Розглянемо систему $F_6=\{\to,\bar{\ }\}$, яку називають *імплікаційною системою*. Ця система функціонально повна, оскільки її можна звести до системи $F_2$ у такий спосіб: $x\to y=\bar{x}\vee y$.

Наприклад, зобразимо булеву формулу $z(x\vee y)\vee\bar{x}y$ в базисі $F_6=\{\to,\bar{\ }\}$:

$$z(x\vee y)\vee\bar{x}y=\overline{z(x\vee y)}\to\bar{x}y=\overline{\bar{z}\vee\overline{(x\vee y)}}\to\overline{\bar{\bar{x}}\vee\bar{y}}=\bar{z}\vee\overline{(x\vee y)}\to\overline{x\vee\bar{y}}=$$
$$=\left(z\to\overline{(x\vee y)}\right)\to\overline{(\bar{x}\to\bar{y})}=\left(z\to\overline{(\bar{x}\to y)}\right)\to\overline{(\bar{x}\to\bar{y})}.$$

**Завдання для самостійної роботи**

1. Зобразіть логічні функції, задані булевими формулами, у базисах $F_1,F_3,F_5$:
а) $x\bar{y}\vee(\bar{x}\vee z)y$; б) $y\vee xz\vee\bar{x}y$; в) $\overline{xy}\vee xy\bar{z}$; г) $x(\bar{y}\vee z)\vee\bar{z}\vee\bar{x}$;
д) $y\vee\bar{x}(\bar{y}\vee z)$; е) $\bar{y}z\vee\bar{x}y\vee xz$; є) $(x\vee\bar{y})z\vee x\bar{z}$; ж) $\bar{x}z\vee x\bar{y}z$.

2. Зобразіть логічні функції, задані булевими формулами, у базисах $F_2,F_4,F_6$:
а) $x\vee\bar{y}(x\vee z)$; б) $yz\vee\bar{x}yz$; в) $y(\bar{x}\vee\bar{z})\vee xz$; г) $xyz\vee\bar{x}\vee y$;
д) $\bar{x}\vee\bar{y}\vee x\bar{y}z$; е) $yz\vee\bar{x}(y\vee z)$; є) $x\bar{y}\vee y\bar{z}\vee\bar{x}z$; ж) $y\bar{z}\vee x\vee\bar{x}z$.

## 4.10. Теореми про функціональну повноту систем булевих функцій

Визначення необхідних і достатніх умов функціональної повноти конкретної системи булевих функцій є основною проблемою. Опишемо властивості функцій, які дозволяють виразити основні булеві операції – диз'юнкцію, кон'юнкцію і заперечення за допомогою цих функцій. Розглянемо декілька допоміжних тверджень.

***Лема про немонотонну функцію.*** З будь-якої немонотонної функції алгебри логіки $f(x_1,...,x_n)$ за допомогою підстановки констант можна отримати заперечення $\varphi(x) \equiv \bar{x}$.

***Доведення.*** Нехай $f \notin T_M$. Тоді існують набори $\tilde{\alpha} = (\alpha_1,...,\alpha_n)$ та $\tilde{\beta} = (\beta_1,...,\beta_n)$, такі що $\tilde{\alpha} < \tilde{\beta}$ (тобто $\forall j \quad \alpha_j \le \beta_j$), а $f(\tilde{\alpha}) > f(\tilde{\beta})$. Укажемо на ті розряди $i_1,...,i_k$ наборів $\tilde{\alpha}$ та $\tilde{\beta}$, в яких вони відрізняються. Зрозуміло, що в наборі $\tilde{\alpha}$ в цих розрядах розташовані 0, а в наборі $\tilde{\beta}$ – 1.

Розглянемо послідовність наборів $\tilde{\alpha}_0, \tilde{\alpha}_1, \tilde{\alpha}_2, ..., \tilde{\alpha}_k$, таких що $\tilde{\alpha} = \tilde{\alpha}_0 < \tilde{\alpha}_1 < \tilde{\alpha}_2 < ... < \tilde{\alpha}_k = \tilde{\beta}$, набор $\tilde{\alpha}_{i+1}$ будують з набору $\tilde{\alpha}_i$ заміною одного з нулів, розміщеного в одному з розрядів $i_1,...,i_k$, на одиницю. Оскільки $f(\tilde{\alpha}) = 1$, а $f(\tilde{\beta}) = 0$, серед наборів $\tilde{\alpha}_0, \tilde{\alpha}_1, \tilde{\alpha}_2, ..., \tilde{\alpha}_k$ знайдуться два сусідні $\tilde{\alpha}_i$ та $\tilde{\alpha}_{i+1}$, такі що $f(\tilde{\alpha}_i) = 1$ та $f(\tilde{\alpha}_{i+1}) = 0$. Нехай вони відрізняються у $r$ -му розряді: $\tilde{\alpha}_i = (\alpha_1, \alpha_2, ..., \alpha_{r-1}, 0, \alpha_{r+1}, ..., \alpha_n)$, $\tilde{\alpha}_{i+1} = (\alpha_1, \alpha_2, ..., \alpha_{r-1}, 1, \alpha_{r+1}, ..., \alpha_n)$. Відтак задамо функцію $\varphi(x)$ : $\varphi(x) = (\alpha_1, \alpha_2, ..., \alpha_{r-1}, x, \alpha_{r+1}, ..., \alpha_n)$. Дійсно, тоді $\varphi(0) = f(\tilde{\alpha}_i) = 1$, $\varphi(1) = f(\tilde{\alpha}_{i+1}) = 0$ та $\varphi(x) \equiv \bar{x}$. ***Лему доведено.***

***Лема про нелінійну функцію.*** З будь-якої нелінійної функції алгебри логіки $f(x_1,...,x_n)$ за допомогою підстановки констант та використання заперечень можна отримати диз'юнкцію та кон'юнкцію.

***Доведення.*** Нехай $f(x_1,...,x_n) \notin T_L$. Розглянемо поліном Жегалкіна цієї функції. З нелінійності функції випливає, що в поліномі будуть наявні кон'юнкції вигляду $x_{i_1} \cdot x_{i_2} \cdot ... x_{i_n}$. Не обмежуючи загальність міркувань, оберемо найкоротшу з них: $K = x_1 \cdot x_2 \cdot ... x_k$. Покладемо $x_3, x_4, ..., x_k = 1$, а для всіх $x_j \notin K$, $x_j = 0$. Підстановкою цих констант у поліном Жегалкіна обернемо кон'юнкцію $K = x_1 \cdot x_2 \cdot ... x_k$ у кон'юнкцію $x_1 \cdot x_2$, а інші кон'юнкції – в нуль. Тоді поліном Жегалкіна цієї функції набуває вигляду
$f(x_1,...x_n) = x_1 \cdot x_2 \cdot P_1(x_3,...,x_n) \oplus x_1 \cdot P_2(x_3,...,x_n) \oplus x_2 \cdot P_3(x_3,...,x_n) \oplus x_1 \cdot P_4(x_3,...,x_n)$, причому $P_1(x_3,...,x_n) \ne 0$.

Інакше кажучи, $\exists\ a_3, a_4, ..., a_n \in E_2 = \{0,1\}$ такі, що $P_1(a_3, a_4, ..., a_n) = 1$. Уведемо допоміжну функцію $f'(x_1, x_2, a_3, a_4, .. a_n) = x_1 x_2 \cdot 1 \oplus x_1 \cdot b \oplus x_2 \cdot c \oplus d$. Функцію $f(x \oplus c, y \oplus b, a_3, a_4, ..., a_n)$ можна зобразити у такий спосіб:

$f(x\oplus c, y\oplus b, a_3, a_4,..., a_n) = (x\oplus c)(y\oplus b)\oplus(x\oplus c)b\oplus(y\oplus b)c\oplus d =$
$= xy\oplus x\cdot b\oplus y\cdot c\oplus b\cdot c\oplus x\cdot b\oplus b\cdot c\oplus y\cdot c\oplus b\cdot c\oplus d = xy\oplus(bc\oplus d) =$
$$= \begin{cases} xy, bc\oplus d = 0 \\ \overline{xy}, bc\oplus d = 1 \end{cases}.$$

***Лему доведено.***

Одну з центральних теорем математичної логіки, яка описує необхідні та достатні умови функціональної повноти системи булевих функцій, сформулював американський математик Еміль Пост в 1941 році.

***Теорема Поста:*** Для того щоб система функцій була функціонально повною, необхідно і достатньо, щоб вона містила хоча б одну немонотонну функцію, хоча б одну нелінійну функцію, хоча б одну несамодвоїсту функцію, хоча б одну функцію, що не зберігає значення нуль, та хоча б одну функцію, що не зберігає значення одиниця.

***Доведення.***

***Необхідність***. Необхідність умови теореми випливає із замкненості та неповноти класів монотонних, лінійних, самодвоїстих функцій та функцій, які зберігають значення 0 та 1. Доведено (за теоремами 4.1 – 4.4, 4.6), що функція, яка не належить даному замкненому класові, не може бути побудована шляхом суперпозиції функцій цього класу.

***Достатність.*** Для доведення достатності покажемо, що за допомогою функцій, які не належать деяким з класів $T_0$, $T_1$, $T^*$, $T_M$, $T_L$, можна побудувати повну систему функцій. Такою повною системою є, наприклад, система, що складається з заперечення та кон'юнкції. Дійсно, довільна булева функція може бути зображена у вигляді ДДНФ, тобто як суперпозиція заперечення, диз'юнкції та кон'юнкції. Відповідно система $\{\bar{\ },\wedge,\vee\}$ є функціонально повна. З цієї системи можна виключити $\vee$, оскільки вона може бути зображена як суперпозиція кон'юнкції та заперечення: $x\vee y = \overline{\bar{x}\wedge\bar{y}}$.

Спочатку побудуємо константи. Почнемо з константи 1. Нехай $\varphi(x) = f_0(x,...,x)$ , де $f_0$ – функція, що не зберігає значення нуль. Тоді $\varphi(0) = f_0(0,...,0)\neq 0$, тобто $\varphi(0) = 1$. Можливі два варіанти:

1) $\varphi(1) = 1$. Тоді формула $\varphi$ реалізує константу 1.
2) $\varphi(1) = 0$. Тоді формула $\varphi$ реалізує заперечення.

Розглянемо несамодвоїсту функцію $f^*$. За означенням несамодвоїстої функції виконується нерівність $f^*(\alpha_1,...,\alpha_n)\neq\overline{f^*(\overline{\alpha_1},...,\overline{\alpha_n})}$, тобто $f^*(\alpha_1,...,\alpha_n) = f^*(\overline{\alpha_1},...,\overline{\alpha_n})$. Нехай тепер $\psi(x) = f^*(x^{\alpha_1},...,x^{\alpha_n})$ . Тоді $\psi(0) = f^*(0^{\alpha_1},...,0^{\alpha_n}) = f^*(\overline{\alpha_1},...,\overline{\alpha_n}) = f^*(\alpha_1,...,\alpha_n) =$
$= f^*(1^{\alpha_1},...,1^{\alpha_n}) = \psi(1)$.

Таким чином, $\psi(0) = \psi(1)$, звідки $\psi = 1$ або $\psi = 0$. Якщо $\psi = 1$, то константу 1 побудовано. В іншому випадку $\psi$ реалізує 0. А тому $\varphi(\psi(x)) = 1$.

Константу 0 будуємо аналогічно, тільки замість функції $f_0$ розглядаємо функцію $f_1$, яка не зберігає значення 1.

За допомогою немонотонної функції підстановкою в неї констант можна побудувати заперечення. Дійсно, нехай $f_M$ – немонотонна функція. Тоді існують набори $\alpha$ та $\beta$ такі, що $\alpha$ передує $\beta$, тобто $\alpha \le \beta$, а $f_M(\alpha)=1$, $f_M(\beta)=0$. Оскільки $\alpha \le \beta$, то у наборі $\alpha$ є декілька, наприклад, $k$ елементів, які дорівнюють 0, тоді як у наборі $\beta$ ті ж самі елементи дорівнюють 1. Візьмемо набір $\alpha$ та замінимо в ньому перший такий нульовий елемент на 1, отримаємо набір $\alpha^1 : \alpha \le \alpha^1$, який відрізняється від набору $\alpha$ тільки одним елементом (такі набори мають назву сусідніх). Повторюючи цю операцію $k$ разів, отримаємо послідовність $\alpha \le \alpha^1 \le \ldots \le \alpha^{k-1} \le \beta$, в якій кожні два сусідні набори відрізняються один від одного тільки одним елементом. В цьому ланцюжку знайдуться два такі набори $\alpha^i, \alpha^{i+1}$, що $f_M(\alpha^i)=1$ та $f_M(\alpha^{i+1})=0$. Нехай ці набори відрізняються $j$-м елементом (значення змінної $x_j$ ), а решта елементів в них однакові. Тоді отримаємо функцію $f_m(\alpha_1^i,\ldots,\alpha_{j-1}^i,x,\alpha_{j+1}^i,\ldots,\alpha_n^i)=g(x_j)$, яка залежить тільки від змінної $x$. Відтак $g(0)=g(\alpha_j^i)=f(\alpha^i)=1$, $g(1)=g(\alpha_j^{i+1})=f(\alpha^{i+1})=0$. Звідси маємо, що $g(x)=\bar{x}$.

Побудуємо кон'юнкцію за допомогою підстановки у нелінійну функцію констант та використання заперечення. Дійсно, нехай $f_L$ – нелінійна функція. Тоді в її поліномі Жегалкіна існує нелінійний доданок, який містить кон'юнкцію принаймні двох змінних. Нехай, для визначеності, це змінні $x_1$ та $x_2$. А отже, функцію $f_L$ можна зобразити у такий спосіб:

$f_L=(x_1 \wedge x_2 \wedge f_a(x_3,\ldots,x_n))(x_1 \wedge f_b(x_3,\ldots,x_n)) \oplus (x_2 \wedge f_c(x_3,\ldots,x_n)) \oplus f_d(x_3,\ldots,x_n)$,

до того ж $f_a(x_3,\ldots,x_n) \ne 0$.

Відповідно $\exists \alpha_1,\ldots,\alpha_n \quad f(\alpha_3,\ldots,\alpha_n)=1$.

Нехай $b=f_b(x_3,\ldots,x_n)$, $c=f_c(x_3,\ldots,x_n)$, $d=f_d(x_3,\ldots,x_n)$,

$L(x_1,x_2)=f_L(x_1,x_2,\alpha_3,\ldots,\alpha_n)=(x_1 \wedge x_2) \oplus (x_1 \wedge b) \oplus (x_2 \wedge c) \oplus d$.

Тоді нехай $\psi(x_1,x_2)=\varphi(x_1 \oplus c, x_2 \oplus b) \oplus (b \wedge c) \oplus d$.

У результаті

$$\psi(x_1,x_2)=(x_1 \oplus c) \wedge (x_2 \oplus b) \oplus b \wedge (x_1 \oplus c) \oplus c \wedge (x_2 \oplus b) \oplus d \oplus (b \wedge c) \oplus d =$$
$$=(x_1 \wedge x_2) \oplus (c \wedge x_2) \oplus (b \wedge x_1) \oplus (b \wedge c) \oplus (b \wedge x_1) \oplus (b \wedge c) \oplus (c \wedge x_2) \oplus (b \wedge c) \oplus d \oplus$$
$$\oplus (b \wedge c) \oplus d = x_1 \wedge x_2.$$

Функцію $x \oplus \alpha$ можна виразити, оскільки $x \oplus 1 = \bar{x}$, $x \oplus 0 = x$. Теорему доведено.

**Завдання для самостійної роботи**

Дослідіть функціональну повноту систем булевих функцій:

а) $\{\oplus,\vee,1\}$; б) $\{\oplus,\wedge,\equiv\}$; в) $\{\to,0\}$;

г) $\{\to,\oplus\}$; д) $\{\equiv,\wedge,0\}$; е) $\{\equiv,\vee,0\}$.

# СПИСОК РЕКОМЕНДОВАННОЇ ЛІТЕРАТУРИ

**Гаврилов, Г.П.** Задачи и упражнения по курсу дискретной математики [Текст] / Г.П. Гаврилов, А.А. Сапоженко. – М.: Наука. – 2007. – 408 с.

**Гончарова, Г.А.** Элементы дискретной математики [Текст]: учеб. пособие/ Г.А. Гончарова, А.А. Мочалин. – М.: Форум: ИНФРА-М. – 2007. – 128 с.

**Иванов, Б.Н.** Дискретная математика. Алгоритмы и программы. Расширенный курс [Текст] / Б.Н. Иванов. – М: Известия. – 2011. – 512 с.

**Спирина, М.С.** Дискретная математика [Текст]: учеб. / М.С. Спирина. –М.: Академия. – 2009. – 368 с.

**Яблонский, С.В.** Введение в дискретную математику [Текст] / С.В. Яблонский. – М.: Наука. – 2006. – 384 с.

**Кук, Д.** Компьютерная математика [Текст] / Д. Кук, Г. Бейз. – М.: Наука. Главная редакция физико-математической литературы. – 1990. – 384 с.

**Кузнецов, О.П.** Дискретная математика для инженера. [Текст] / О.П. Кузнецов, Г.М. Адельсон-Вельский. – 2-е изд., перераб. и доп. – М.: Физматгиз. – 1988. – 408 с.

**Новиков, Ф.А.** Дискретная математика для программистов [Текст]: учебник для вузов / Ф.А. Новиков. – 2-е изд. – СПб.: Питер. – 2007. – 364 с.

**Романовский, И.В.** Дискретный анализ [Текст] / И.В. Романовский. – Невский Диалект: БХВ-Петербург. – 2003. – 320 с.

**Марченков, С.С.** Булевы функции [Текст] / С.С. Марченков. – М.: Физматлит. – 2002. – 68 с.

**Лавров, И.А.** Задачи по теории множеств, математической логике и теории алгоритмов [Текст] / И.А. Лавров, Л.Л. Максимова. – М.: Физматлит. – 2004. – 256 с.

**Бондаренко, М.Ф.** Комп'ютерна дискретна математика [Текст]: підруч. / М.Ф. Бондаренко, Н.В. Білоус, А.Г. Руткас. – Х.: СМІТ. – 2004. – 480 с.

**Швай, О.Л.** Практикум з дискретної математики [Текст]: навч. посіб. для студ. вищ. навч. закл. / О.Л. Швай. – Луцьк: Волин. нац. ун-т ім. Лесі Українки. – 2011. – 236 с.

**Капітонова, Ю.В.** Основи дискретної математики [Текст] / Ю.В. Капітонова [та ін.]. – К.: Наук. думка. – 2002. – 579 с.

# ЗМІСТ

Темплан 2014, поз. 17

Навчальне видання

Наталія Анатоліївна Гук

**Посібник до вивчення дисципліни «Дискретна математика»**

Редактор В.О. Насєкан
Техредактор Т.І. Севост'янова
Коректор Т.А. Белиба

Підписано до друку 20.02.2014 Формат 60x84/16 Папір друкарський. Друк плоский
Ум. друк. арк. 2,8. Ум. фарбовідб. 2,8. Обл.-вид. арк. 3,4. Тираж 100 пр.
Зам. № 941

РВВ ДНУ, просп. Гагаріна, 72, м. Дніпропетровськ, 49010.
Друкарня «Ліра», пл. Десантників, 1, м. Дніпропетровськ, 49038.
Свідоцтво про внесення до державного реєстру
серія ДП №14 від 13.07.2000 р.