

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

МАТЕМАТИЧНА ЛОГІКА

ТА ТЕОРІЯ АЛГОРИТМІВ

ПРАКТИКУМ

*Рекомендовано Методичною радою КПІ ім. Ігоря Сікорського
як навчальний посібник для студентів,
які навчаються за спеціальністю F1 «Прикладна математика»,
освітньою програмою «Машинне навчання та математичне моделювання»*

Київ
КПІ ім. Ігоря Сікорського
2026

Рецензенти: *Дичка І.А.*, д-р техн. наук., проф., професор кафедри програмного забезпечення комп'ютер-них систем ФПСМ

Відповідальний редактор: *Тарасенко-Клятченко О.В.*, канд. техн. наук, доц., доцент кафедри системного програмування і спеціалізованих комп'ютерних систем ФПСМ

*Гриф надано Методичною радою КПІ ім. Ігоря Сікорського
(протокол № 01 від 11.09.2026 р.)
за поданням Вченої ради факультету програмних систем
та прикладної математики
(протокол № 01 від 27.08.2026 р.)*

Електронне мережне навчальне видання

Темнікова Олена Леонідівна

МАТЕМАТИЧНА ЛОГІКА ТА ТЕОРІЯ АЛГОРИТМІВ

Математична логіка та теорія алгоритмів: Практикум [Електронний ресурс]: навч. посіб. для студ. спеціальності F1 «Прикладна математика», освітньої програми «Машинне навчання та математичне моделювання» / О.Л.Темнікова; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 3,57 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2026. – 207 с.

Навчальний посібник розроблено для оволодіння студентами, які навчаються за спеціальністю F1 «Прикладна математика», освітньою програмою «Машинне навчання та математичне моделювання» факультету програмних систем та прикладної математики КПІ ім. Ігоря Сікорського, практичними навичками з дисципліни «Математична логіка та теорія алгоритмів». Практикум містить основні теоретичні відомості та практичні прийоми для широкого спектра застосування і розв'язання задач з булевої алгебри; основні теоретичні відомості та практичні прийоми розв'язання задач з математичної логіки: основи формалізації та доведення логічних слідувань, вимоги до правил їх оформлення; поняття з основ теорії нетрадиційних логік та суджень для застосування методів багатозначних логік і нечітких множин, та з теорії алгоритмів, а саме з дослідження примітивно рекурсивних функцій, визначення основ арифметизації математики, побудови машин Тюрінга та нормальних алгоритмів Маркова, засвоєння основних принципів і методів функціонального програмування.

ЗМІСТ

ВСТУП.....	5
1. Булева алгебра	8
1.1. Булеві формули. Таблиці істинності. Диз'юнктивні та кон'юнктивні нормальні форми.....	8
1.2. Властивості булевих функцій. Алгебра Жегалкіна.	22
1.3. Функціональна повнота систем булевих функцій.....	31
Відповіді до розділу 1.....	35
2. Теорія висловлювань	39
2.1. Основні поняття.....	39
2.2. Формалізація і рішення логічних задач.....	53
2.3. Метод резолюцій.....	63
2.4. Формальні теорії. Числення висловлювань.....	74
Відповіді до розділу 2.....	80
3. Теорія предикатів першого порядку	98
3.1. Введення кванторів. Побудова таблиць істинності на обмеженій інтерпретації.....	98
3.2. Доведення логічного слідування в теорії предикатів 1-го порядку.....	107
3.3. Особливості застосування методу резолюцій в теорії предикатів 1-го порядку	115
3.4. Логічне програмування.....	117
Відповіді до розділу 3.....	121
4. Теорія алгоритмів.....	135
4.1. Примітивно рекурсивні та рекурсивні функції.....	135
4.2. Алгоритмічні схеми. Машина Тюрінга	141
4.3. Нормальний алгоритм Маркова.....	145
4.4. λ -числення та функції обробки списків.....	148
Відповіді до розділу 4.....	156

5. Некласичні логіки.....	171
5.1. Нечіткі множини. Операції над нечіткими множинами.....	171
5.2. Нечіткі та лінгвістичні змінні. Методи побудови функції приналежності. Нечіткий логічний висновок.....	174
5.3. Багатозначні логіки.....	188
5.4. Модальні логіки.....	193
Відповіді до розділу 5.....	196
РЕКОМЕНДОВАНА ЛІТЕРАТУРА.....	205

ВСТУП

Навчальне видання призначене для студентів, які вивчають дисципліну «Математична логіка та теорія алгоритмів» з циклу загальної підготовки навчального плану спеціальності F1 «Прикладна математика» і є базовим при навчанні за освітньою програмою «Машинне навчання та математичне моделювання», що спрямована для вирішення різних задач професійної діяльності. Метою викладання дисципліни є оволодіння основними поняттями і методами, що необхідні для формування світогляду на математичну логіку як на фундаментальну науку, що призначена для формалізації знань, розробки і використання нових інформаційних технологій, моделювання складних систем, систем штучного інтелекту. Крім того вивчення дисципліни надає змогу формування у студентів здатності до абстрактного мислення, знання основних алгоритмічних схем теорії алгоритмів та основних понять теорії обчислювання. Курс знайомить студентів з сучасними розділами неklasичних логік та прикладами їх практичного застосування.

Дисципліна «Математична логіка та теорія алгоритмів» розрахована на 30 академічних годин лекційних занять та на 30 годин занять практичних; вивчається на факультеті програмних систем та прикладної математики у 3 семестрі.

У даному практикумі розглядаються основні теоретичні відомості та практичні прийоми розв'язання задач з математичної логіки та теорії алгоритмів, вимоги до правил оформлення та доведення логічних слідувань, основи неklasичних логік, застосування різних алгоритмічних схем.

Навчальне видання складається з п'яти розділів за наступними темами (до кожної теми вказано список рекомендованої літератури):

Тема 1. Булева алгебра: булеві функції, властивості, повнота систем.

Література: 3, 6, 7, 8, 19, 20.

Тема 2. Алгебра висловлювань. Логічне слідування в алгебрі висловлювань. Формальні теорії.

Література: 1, 2, 6, 7, 9, 18, 19, 20.

Тема 3. Теорія предикатів першого порядку: інтерпретація формул логіки предикатів, формалізація речень природної мови, доведення логічних слідувань. Автоматичне доведення теорем: метод резолюцій, логічне програмування.

Література: 1, 2, 5, 6, 7, 9, 17, 18.

Тема 4. Алгоритмічні схеми: машини Тюринга, алгоритми Маркова, числення Черча.

Література: 4, 6, 7, 10, 13, 14, 20, 21.

Тема 5. Некласична логіка: нечіткі множини; багатозначна логіка, модальна логіка.

Література: 6, 7, 9, 11, 12, 15, 16, 19.

Кожній темі відповідає глава у практикумі; на початку наводяться короткі теоретичні відомості; перед списком завдань розташовано приклад, що містить методичні вказівки з виконання та оформлення розв'язання задач за даною темою; наприкінці є відповіді і розв'язок до деяких завдань.

Основною ідеєю викладання частини дисципліни – математичної логіки – є поняття аксіоматичного методу, його використання для побудови формальних систем, відображення формальної системи на моделі і використання цього математичного апарату для формалізації і дослідження проблемних областей. З цією метою при викладанні курсу використовується велика кількість змістових логічних задач. При вивченні математичної логіки розглядаються методи і результати, що стали основою однієї з парадигм програмування - логічного програмування. Курс містить поняття з основ теорії нетрадиційних логік та суджень для застосування методів багатозначних логік і нечітких множин для опису слабо структурованих проблемних областей. А починається опанування логіки з бази логічного моделювання: будевих функцій і булевої алгебри.

При вивченні розділу «Теорія алгоритмів» є викладення основних понять і методів, що формує погляд на теорію алгоритмів, як на фундаментальну

науку, яка використовується для визначення та розгляду філософії проблем обчислюваності та арифметизації математики. У даному практикумі розглядаються основні теоретичні відомості та практичні прийоми розв'язання задач з теорії алгоритмів, а саме з дослідження примітивно рекурсивних функцій, визначення основ арифметизації математики, побудови машин Тюрінга та нормальних алгоритмів Маркова, засвоєння основних принципів і методів функціонального програмування.

Після засвоєння кредитного модуля «Математична логіка та теорія алгоритмів» студенти мають продемонструвати такі результати навчання: уміння досліджувати системи булевих функцій на повноту, замкнутість, несуперечність; формалізувати знання про прикладну проблемну область мовою логіки висловлювань і логіки предикатів; формалізувати знання за допомогою нетрадиційних логік та міркувань; застосовувати методи багатозначних логік і нечітких множин для опису погано структурованих проблемних областей. Згідно з вимогами освітньо-професійної програми, студенти після засвоєння дисципліни «Математична логіка та теорія алгоритмів» також мають продемонструвати знання основних понять теорії обчислюваності та уміння і досвід побудови алгоритмічних схем: машин Тюрінга і нормальних алгоритмів Маркова; написання рекурсивних функцій, програм із застосуванням методів функціонального програмування.

Навчальне видання має стати в нагоді під час опрацювання матеріалів лекцій, підготовки до практичних занять, до заліку та при самостійній підготовці до ректорського контролю і вступу на навчання за програмою магістерської підготовки.

1. БУЛЕВА АЛГЕБРА

1.1. Булеві формули. Таблиці істинності. Диз'юнктивні та кон'юнктивні нормальні форми

Теоретичні відомості

Визначення 1.1. Будь-яку змінну, яка може приймати одне з двох можливих значень 0 або 1, назвемо булевою змінною.

Визначення 1.2. Функції, які визначені на множині $\{0, 1\}$ і приймають значення з цієї множини, називаються булевими функціями.

Булева функція називається n -місною, якщо число її змінних дорівнює n . Занумеруємо змінні булевої функції. Сукупність значень змінних булевої функції назвемо набором (або кортежем). Набір довжини n будемо називати n -набором. Тоді областю визначення n -місної булевої функції є сукупність всіх упорядкованих n -наборів, компонентами яких є значення булевих змінних 0 або 1.

Кожен n -набір можна розглядати як запис деякого числа в двійковій системі числення. Нехай n -набір має вигляд: $a_1, a_2, \dots, a_n$, де $a_i=0$ або 1. Тоді число, визначене як $a_1 \cdot 2^{n-1} + a_2 \cdot 2^{n-2} + \dots + a_n$, назвемо номером даного набору. Зрозуміло, що номером n -набору може бути будь-яке натуральне число $0 \leq K \leq 2^n - 1$.

Наприклад, для булевої функції $f(x, y, z)$ множина значень $x=1, y=0, z=1$ записується як набір 101. Булева функція може бути задана за допомогою таблиці – перерахування її значень на всіх наборах. Множину наборів прийнято записувати в лексикографічному порядку, тому кожен набір являє собою код двійкового числа. Відповідне йому десяткове число будемо називати номером набору. Наприклад, номер набору 101 дорівнює 5, номер набору 110 - 6.

Твердження 1.1. Кількість наборів булевої функції $f(x_1, \dots, x_n)$ від n змінних дорівнює 2^n . Кількість булевих функцій від n змінних дорівнює 2^{2^n} .

Існує 4 булеві функції від однієї змінної та 16 булевих функцій від двох змінних, тримісних - вже 256, а 4-х-місних - близько 65 тисяч.

Всі набори розташуємо в порядку зростання їх номерів, а поруч з кожним набором розмістимо значення булевої функції на цьому наборі. Отриману таким чином таблицю назовемо таблицею значень, або таблицею істинності булевої функції. У таблиці 1.1 наведено чотири булеві функції від однієї змінної:

f_0 – константа 0,

f_1 – тотожна функція $f(x) = x$,

f_2 – заперечення $f(x) = \neg x$,

f_3 – константа 1.

Таблиця 1.1.

x	f_0	f_1	f_2	f_3
0	0	0	1	1
1	0	1	0	1

В табл. 1.2 приведені функції від двох змінних.

Таблиця.1.2.

x	y	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Ці функції:

f_0 – константа 0

f_1 – $x \wedge y$ (або $x \& y$) – кон'юнкція

f_2 – Заперечення імплікації $x \rightarrow y$ $f(x,y) = \neg (x \rightarrow y)$

f_3 – Тотожна функція $f(x,y) = x$

- f_4 – Заперечення імплікації $y \rightarrow x$ $f(x,y) = \neg (y \rightarrow x)$
- f_5 – Тотожна функція $f(x,y) = y$
- f_6 – $x \oplus y$ – сума за *mod* 2 (операція Жегалкіна)
- f_7 – $x \vee y$ – диз'юнкція
- f_8 – заперечення диз'юнкції $f(x,y) = \neg (x \vee y) = (x \downarrow y)$ – стрілка Пірса (іноді можна зустріти назву функція Даггера або функція Вебба)
- f_9 – $x \equiv y$ (або $x \sim y$) – еквівалентність (еквіваленція)
- f_{10} – заперечення $f(y) = \neg y$
- f_{11} – $y \rightarrow x$ (або $y \supset x$) – імплікація
- f_{12} – заперечення $f(x) = \neg x$
- f_{13} – $x \rightarrow y$ (або $x \supset y$) – імплікація
- f_{14} – заперечення кон'юнкції $f(x,y) = \neg (x \wedge y) = (x \mid y)$ – штрих Шеффера
- f_{15} – константа 1

Твердження 1.2. Множина, що складається з двох значень 0 і 1, на якому визначені унарна операція заперечення $\neg$ згідно табл. 1.1 і бінарні операції диз'юнкції $\vee$ і кон'юнкції $\wedge$ відповідно до табл. 1.2, є булевою алгеброю.

Аксіоми і теореми булевої алгебри безпосередньо впливають з властивостей булевих решіток. Наведемо список аксіом і основних теорем булевої алгебри.

Аксіоми булевої алгебри:

(1.1) *комутативні закони:*

$$a \vee b = b \vee a;$$

$$a \wedge b = b \wedge a;$$

(1.2) *дистрибутивні закони:*

$$a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c);$$

$$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c);$$

(1.3) *асоціативні закони:*

$$a \vee (b \vee c) = (a \vee b) \vee c;$$

$$a \wedge (b \wedge c) = (a \wedge b) \wedge c;$$

(1.4) *властивості 0 й 1:*

$$a \vee 0 = a, a \wedge 1 = a;$$

(1.5) *закон виключення третього:*

$$a \vee \neg a = 1; \text{ та закон протиріччя: } a \wedge \neg a = 0.$$

Теорема:

(1.6) якщо для всіх a $a \vee b = a$, то $b = 0$;

якщо для всіх a $a \wedge b = a$, то $b = 1$;

(1.7) якщо $a \vee b = 1$ і $a \wedge b = 0$, то $b = \neg a$;

(1.8) $\neg \neg a = a$;

(1.9) $\neg 0 = 1, \neg 1 = 0$;

(1.10) *закон ідемпотентності:*

$$a \vee a = a;$$

$$a \wedge a = a;$$

(1.11) $a \vee 1 = 1, a \wedge 0 = 0$;

(1.12) *закони поглинання:*

$$a \vee (a \wedge b) = a;$$

$$a \wedge (a \vee b) = a;$$

(1.13) *закони де Моргана:*

$$\neg(a \vee b) = \neg a \wedge \neg b;$$

$$\neg(a \wedge b) = \neg a \vee \neg b;$$

(1.14) *закони склеювання:*

$$(a \vee b) \wedge (a \vee \neg b) = a;$$

$$(a \wedge b) \vee (a \wedge \neg b) = a;$$

(1.15) *закони поглинання доповнення:*

$$a \vee (\neg a \wedge b) = a \vee b;$$

$$a \wedge (\neg a \vee b) = a \wedge b.$$

Це найбільш проста модель булевої алгебри, разом з тим найбільш важлива для комп'ютерних наук. Інша поширена модель - модель обчислення висловлювань.

Якщо взяти деяку непорожню множину A , то її множина-ступінь $\rho(A)$ буде моделлю булевої алгебри, якщо домовитися про наступне (встановити ізоморфізм між поняттям та визначенням):

- елементи цієї булевої алгебри – різні підмножини множини A ;
- операція кон'юнкції визначається як перетин множин, диз'юнкції – об'єднання множин, операція заперечення – абсолютне доповнення;
- множина $\cup$ (універсум) і $\emptyset$ – відповідно одиниця й нуль алгебри.

Неважко переконатися, що всі аксіоми, наведені вище, будуть виконуватися.

Булеві формули

Визначення 1.3.

- Кожна змінна є формулою.
- Якщо x, y – формули, то формулами є $(\neg x)$, $(x \vee y)$, $(x \& y)$.
- Інших формул нема.

У зображенні формул прийняті наступні допущення: зовнішні дужки опускають; встановлюють пріоритети виконання операцій в наступному порядку:

$\neg$ – заперечення (найвищий пріоритет),

$\wedge$ – кон'юнкція,

$\vee$ – диз'юнкція,

$\rightarrow, \equiv$ – імплікація й еквіваленція (мають однаковий пріоритет, тим паче, що вони не є алгебраїчними операціями, а слугують для скорочення запису).

З урахуванням цих пріоритетів надлишкові дужки також опускаються.

Для кожної булевої формули можна побудувати її таблицю істинності, обчислюючи її значення на кожному наборі змінних. Звідси випливає, що кожній булевій формулі відповідає деяка булева функція, а формулу булевої

алгебри можна розглядати як представлення булевої функції від деяких змінних.

Кожна булева формула єдиним чином визначає булеву функцію. Єдиність випливає з того, що для кожної формули булевої алгебри можна побудувати єдину таблицю істинності. Тому в подальшому поняття «булева формула» і «булева функція» будемо вживати як рівнозначні.

Протилежне твердження про відповідність булевих формул і функцій, не справедливо, тобто кожній булевій функції можна поставити у відповідність безліч булевих формул.

Булеві формули називаються еквівалентними, або рівносильними, якщо їх таблиці збігаються. Якщо дві рівносильні формули з'єднати за допомогою операції еквівалентності, то отримана формула буде тотожно дорівнювати одиниці. Такі тотожно істинні формули називаються тавтологіями або загальнозначущими. Формули, тотожно рівні нулю, тобто тотожно хибні, називаються суперечливими або суперечностями. Аксиоми і теореми булевої алгебри задають рівносильні формули.

Розглянемо деякі рівносильні формули (див. Табл. 1.3).

Таблиця 1.3.

x	y	$x \rightarrow y$	$\neg x \vee y$	$y \rightarrow x$	$(x \rightarrow y)(y \rightarrow x)$	$x \equiv y$	$x \oplus y$	$\neg(x \oplus y)$
0	0	1	1	1	1	1	0	1
0	1	1	1	0	0	0	1	0
1	0	0	0	1	0	0	1	0
1	1	1	1	1	1	1	0	1

Ми бачимо, що таблиці значення формул $x \rightarrow y$ й $\neg x \vee y$ збігаються. Отже, ці формули еквівалентні:

$$(1.16) \quad x \rightarrow y = \neg x \vee y = \neg(x \& \neg y).$$

Еквівалентні також наступні формули:

$$(1.17) \quad x \equiv y = \neg(x \oplus y),$$

$$(1.18) \quad x \equiv y = (x \rightarrow y)(y \rightarrow x).$$

В (1.18) ми можемо замінити імплікацію на рівносильну їй формулу (1.16). В результаті отримаємо:

$$(1.19) x \equiv y = (x \rightarrow y)(y \rightarrow x) = (\neg x \vee y)(\neg y \vee x) \text{ або } (\neg x \& \neg y) \vee (x \& y).$$

$$(1.20) \text{ Тоді, } x \oplus y = \neg(x \equiv y) = (\neg x \& y) \vee (x \& \neg y) = (x \vee y) \& (\neg x \vee \neg y).$$

Ці рівносильності пояснюють, чому при визначенні булевої формули використовуються тільки операції $\neg$, $\vee$, $\wedge$ – операції імплікації, еквіваленції й складання по модулю 2 можуть бути введені за допомогою рівносильних формул (1.16) - (1.20).

За допомогою рівносильних формул можна проводити еквівалентні перетворення булевих формул. Наприклад, перетворимо булеву формулу $xy \rightarrow (\neg y \oplus z)$.

$$xy \rightarrow (\neg y \oplus z) =$$

$$= \neg(xy) \vee (\neg y \neg z \vee yz) =$$

$$= \neg x \vee \neg y \vee \neg y \neg z \vee yz =$$

$$= \neg x \vee \neg y \vee yz = \neg x \vee \neg y \vee z$$

$$x \rightarrow y = \neg x \vee y$$

$$x \oplus y = (\neg xy) \vee (x \neg y), \neg \neg y = y$$

закон де Моргана

закони поглинання

закон поглинання доповнення

Побудуємо також для вихідної формули і перетвореної таблицю істинності (див. Табл.1.4) і переконаємося в їх рівносильності.

Таблиця 1.4.

x	y	z	xy	$\neg y$	$\neg y \oplus z$	$xy \rightarrow (\neg y \oplus z)$	$\neg x$	$\neg x \vee \neg y$	$\neg x \vee \neg y \vee z$
0	0	0	0	1	1	1	1	1	1
0	0	1	0	1	0	1	1	1	1
0	1	0	0	0	0	1	1	1	1
0	1	1	0	0	1	1	1	1	1
1	0	0	0	1	1	1	0	1	1
1	0	1	0	1	0	1	0	1	1
1	1	0	1	0	0	0	0	0	0
1	1	1	1	0	1	1	0	0	1

Важливо вміти визначати, чи є формула тавтологією. Для цього можна скористатися одним з методів:

- побудувати таблицю істинності (формула повинна тотожно дорівнювати 1),
- виконати еквівалентні перетворення (і отримати 1),
- застосувати метод редукції (зведення до протиріччя – передбачається, що існує набір, на якому формула приймає значення 0, далі намагаються підібрати відповідні значення змінних або спростувати тезу).

Диз'юнктивні та кон'юнктивні нормальні форми

Оскільки одна і та сама булева функція може бути представлена різноманітними формулами, виникає питання про єдиність представлення булевих функцій, тобто знаходження такої універсальної форми, щоб кожна булева функція була представлена в ній. Такими формами в булевій алгебрі є *досконалі диз'юнктивна та кон'юнктивна нормальні форми*.

Визначення 1.4. Будь-яка булева формула, побудована за допомогою однієї тільки операції диз'юнкції над булевими змінними чи їхніми запереченнями, називається *елементарною диз'юнкцією* чи *диз'юнктом*. Наприклад, $x_1 \vee x_2 \vee \neg x_3 \vee x_4$ – елементарна диз'юнкція.

Визначення 1.5. Будь-яка булева формула, побудована за допомогою однієї тільки операції кон'юнкції над булевими змінними чи їхніми запереченнями, називається *елементарною кон'юнкцією*, чи *кон'юнктом*. Наприклад, $x_1 x_2 \neg x_3 \neg x_4$ – елементарна кон'юнкція.

Теорема 1.1. Для того, щоб елементарна диз'юнкція тотожно дорівнювала одиниці, необхідно та достатньо, щоб до неї входила деяка змінна разом із її запереченням.

Теорема 1.2. Для того, щоб елементарна кон'юнкція тотожно дорівнювала нулю, необхідно та достатньо, щоб у неї входила деяка змінна разом із її запереченням.

Справедливість теорем 1.1, 1.2 випливає з аксіоми (1.4) і визначення операцій диз'юнкції та кон'юнкції.

Визначення 1.6. Булева функція називається *конституентною одиниці* (мінтермом), якщо вона дорівнює одиниці тільки на одному наборі своїх аргументів.

Визначення 1.7. Булева функція називається *конституентною нуля* (макстермом), якщо вона дорівнює нулю тільки на одному наборі своїх аргументів.

Приклад. Серед булевих функцій двох аргументів кон'юнкція та стрілка Пірса є конституентами одиниці, а диз'юнкція, імплікація, штрих Шеффера є конституентами нуля.

Теорема 1.3. Не тотожно хибна елементарна кон'юнкція від n -змінних є конституентною одиниці від n -змінних.

Із теореми 1.3 випливає, що будь-яку конституенту одиниці можна представити у вигляді елементарної кон'юнкції. Для цього необхідно утворити кон'юнкцію всіх її аргументів та розставити заперечення над тими змінними, які рівні нулю на наборі, що перетворює функцію в одиницю.

Аналогічну теорему можна довести для елементарної диз'юнкції.

Теорема 1.4. Не тотожно істинна елементарна диз'юнкція від n -змінних є конституентною нуля від n -змінних.

Із цієї теореми випливає, що будь-яку конституенту нуля можна представити у вигляді елементарної диз'юнкції. Для цього необхідно утворити диз'юнкцію всіх змінних та розставити заперечення над тими змінними, які рівні одиниці на наборі, що перетворює функцію в нуль.

Приклад. Нехай функція $f(x, y, z, v)$ дорівнює одиниці на єдиному наборі 0110. Тоді вона є конституентною одиниці і її можна записати у вигляді: $f(x, y, z, v) = \neg xyz \neg v$.

Визначення 1.8. Диз'юнкція елементарних кон'юнкцій, що не містить двох однакових кон'юнкцій, називається *диз'юнктивною нормальною формою (ДНФ)*.

Визначення 1.9. Диз'юнктивна нормальна форма, всі кон'юнкції якої є конституенти одиниці, називається *досконалою диз'юнктивною нормальною формою (ДДНФ)*.

Визначення 1.10. Кон'юнкція елементарних диз'юнкцій, що не містить двох однакових диз'юнкцій, називається *кон'юнктивною нормальною формою (КНФ)*.

Визначення 1.11. Кон'юнктивна нормальна форма, всі диз'юнкції якої є конституенти нуля, називається *досконалою кон'юнктивною нормальною формою (ДКНФ)*.

Теорема 1.5. Будь-яку булеву функцію можна представити у вигляді ДДНФ і ДКНФ.

Приклад. Нехай булева функція $f(x, y, z)$ задана таблицею 1.5. Побудуємо ДДНФ і ДКНФ даної функції.

Таблиця 1.5.

x	0	0	0	0	1	1	1	1
y	0	0	1	1	0	0	1	1
z	0	1	0	1	0	1	0	1
$f(x, y, z)$	1	0	1	0	1	1	0	0

Для побудови ДДНФ булевої функції необхідно розглянути всі набори, на яких функція дорівнює одиниці, і утворити конституенти одиниці, що відповідають цим наборам. Отримані конституенти об'єднати знаками диз'юнкції. Отримаємо ДДНФ: $f(x, y, z) = \neg x \neg y \neg z \vee \neg x y \neg z \vee x \neg y \neg z \vee x \neg y z$. Двоїсте правило існує для побудови ДКНФ. Розглянемо всі набори, на яких булева функція дорівнює нулю. Утворимо всі конституенти нуля як елементарні диз'юнкції, в яких кожна змінна береться без заперечення, якщо вона дорівнює нулю, та з запереченням, якщо вона дорівнює одиниці на даному наборі. З'єднаємо всі конституенти нуля символами кон'юнкції, отримаємо ДКНФ:

$$f(x, y, z) = (x \vee y \vee \neg z)(x \vee \neg y \vee \neg z)(\neg x \vee \neg y \vee z)(\neg x \vee \neg y \vee \neg z).$$

Інший спосіб полягає в тому, що будь-яку булеву формулу можна привести до вигляду ДНФ (КНФ) та ДДНФ (ДКНФ) за допомогою еквівалентних перетворень, використовуючи співвідношення (1.1) – (1.20).

Приклад. Приведемо до ДДНФ формулу $(x \oplus y) \rightarrow xz$. Отримаємо:

$(x \oplus y) \rightarrow xz = \neg(x \oplus y) \vee xz = (x \equiv y) \vee xz = \neg x \neg y \vee xy \vee xz$. Отримана формула представлена в ДНФ, і для того, щоб отримати її представлення в ДДНФ, необхідно в кожному «доданку» диз'юнкції (в кожному кон'юнкті) присутність всіх трьох змінних, від яких залежить функція. Отже, необхідно «помножити» кожний «доданок» на 1, утворену як диз'юнкцію змінної, якої не вистачає в даному кон'юнкті, і її заперечення. Таким чином, $\neg x \neg y$ та xy «помножимо» на $1 = z \vee \neg z$, та xz – на $1 = y \vee \neg y$.

Тоді отримаємо: $\neg x \neg y (z \vee \neg z) \vee xy (z \vee \neg z) \vee xz (y \vee \neg y) = xyz \vee xy \neg z \vee \neg x \neg y z \vee \neg x \neg y \neg z \vee xyz \vee x \neg y z$. В силу ідемпотентності $xyz \vee xyz = xyz$, тому ДДНФ є $xyz \vee xy \neg z \vee \neg x \neg y z \vee \neg x \neg y \neg z \vee x \neg y z$. Аналогічно можна отримати ДКНФ.

Завдання 1.1.1. Перевірити еквівалентність формул A і B , використовуючи основні аксіоми та теореми булевої алгебри.

- 1) $A = xy \vee \neg x \neg z \vee x \neg z$, $B = x(\neg y \neg z) \vee (\neg x \vee \neg z)$;
- 2) $A = \neg x \neg y \neg z \vee \neg x \neg y z \vee \neg xy$, $B = \neg x$;
- 3) $A = xy \neg z \vee xy \vee z \neg z$, $B = xy$;
- 4) $A = x(y \vee z) \vee y(\neg x \neg z) \vee \neg z(\neg y \vee x)$, $B = x \vee y \vee \neg z$;
- 5) $A = (x \neg z) \vee (xy) \vee (y \neg z)$, $B = x \neg (yz) \vee \neg xz$;
- 6) $A = x \rightarrow (xy \rightarrow ((x \rightarrow y) \rightarrow y) z)$, $B = y \rightarrow (x \rightarrow z)$;
- 7) $A = \neg((x \rightarrow y) \vee ((x \rightarrow z)y))$, $B = (x \neg y)(\neg y \rightarrow x \neg z)$;
- 8) $A = (x \vee y)(y \vee z)(z \vee x)$, $B = xy \vee yz \vee zx$;
- 9) $A = ((x \vee y) \vee z) \rightarrow ((x \vee y)(x \vee z))$, $B = x \equiv z$;
- 10) $A = x(x \rightarrow y)$, $B = xy$;
- 11) $A = \neg(x \vee y) \vee (xy)$, $B = (x \vee \neg y)(\neg x \vee y)$;

$$12) A = x \oplus y, B = (\neg xy) \vee (\neg yx);$$

$$13) A = \neg(\neg xyz), B = x \vee \neg z \vee \neg y;$$

$$14) A = (x \vee y)(x \vee z)(y \vee w)(z \vee w), B = xw \vee yz;$$

$$15) A = ((x \oplus y) \rightarrow x \vee y)((\neg x \rightarrow y) \rightarrow (x \oplus y)), B = \neg(xy).$$

Завдання 1.1.2. Побудувати таблицю істинності булевої функції. Виразити булеву функцію у вигляді ДДНФ, ДКНФ. Спростити, використовуючи основні аксіоми та теореми булевої алгебри.

Приклад. $(x \rightarrow (y \vee z)) \rightarrow xz$.

Метод еквівалентних перетворень булевих формул.

$$\text{ДДНФ: } (x \rightarrow (y \vee z)) \rightarrow xz = \quad \text{використовуємо еквівалентність } x \rightarrow y \equiv \neg x \vee y$$

$$= \neg(\neg x \vee y \vee z) \vee xz = \quad \text{використовуємо закон де Моргана}$$

$$= x \neg y \neg z \vee xz \quad \text{- ця форма представлення булевої функції є ДНФ.}$$

Якщо виконувати спрощення, то далі використовуємо дистрибутивний закон

$$x \neg y \neg z \vee xz = x(\neg y \neg z \vee z) = \quad \text{закон поглинання доповнення } \neg y \neg z \vee z = \neg y \vee z$$

$$= x(\neg y \vee z) = x \neg y \vee xz.$$

Якщо метою є отримання ДДНФ алгебраїчним шляхом, то треба ввести 1 як

$$y \vee \neg y \quad (y \vee \neg y \equiv 1).$$

$$x \neg y \neg z \vee xz = (x \neg y \neg z) \vee (xz(y \vee \neg y)) = \quad \text{використовуємо закон дистрибутивності}$$

$$= x \neg y \neg z \vee xyz \vee x \neg yz.$$

Таблиця істинності.

x	0	0	0	0	1	1	1	1
y	0	0	1	1	0	0	1	1
z	0	1	0	1	0	1	0	1
f(x, y, z)	0	0	0	0	1	1	0	1
конституенти	$x \vee y \vee z$	$x \vee y \vee \neg z$	$x \vee \neg y \vee z$	$x \vee \neg y \vee \neg z$	$x \neg y \neg z$	$x \neg yz$	$\neg x \vee \neg y \vee z$	xyz

Для отримання ДКНФ: $(x \rightarrow (y \vee z)) \rightarrow xz = (\neg x \vee y \vee z) \rightarrow xz = \neg(\neg x \vee y \vee z) \vee xz =$
 $x \neg y \neg z \vee xz = (x \vee xz)(\neg y \vee xz)(\neg z \vee xz) = x(\neg y \vee x)(\neg y \vee z)(\neg z \vee x)(\neg z \vee z) =$
 $x(\neg y \vee z)(\neg z \vee x) = x(\neg y \vee z) = (x \vee y \neg y)(\neg y \vee z \vee x \neg x) = ((x \vee y) \vee z \neg z) \&$
 $\&((x \vee \neg y) \vee z \neg z)(\neg y \vee x \vee z)(\neg y \vee \neg x \vee z) = (x \vee y \vee z)(x \vee y \vee \neg z)(x \vee \neg y \vee z) \&$
 $\&(x \vee \neg y \vee \neg z)(\neg x \vee \neg y \vee z).$

Побудуємо ДДНФ та ДКНФ за допомогою таблиці істинності. Випишемо усі конституенти нулю та одиниці (див. таблицю). Диз'юнкція усіх конститuent одиниці дає ДДНФ: $f(x, y, z) = x \neg y \neg z \vee xyz \vee x \neg yz.$

Кон'юнкція усіх конститuent нулю дає ДКНФ:

$$f(x, y, z) = (x \vee y \vee z)(x \vee y \vee \neg z)(x \vee \neg y \vee z)(x \vee \neg y \vee \neg z)(\neg x \vee \neg y \vee z).$$

- 1) $(x \vee \neg y \vee z)(\neg x \neg y \neg z);$
- 2) $(\neg x \neg y \vee \neg y \neg z) \equiv (x \vee z \rightarrow y);$
- 3) $xy \vee x(z \vee y)z;$
- 4) $x \rightarrow (x \rightarrow z);$
- 5) $(x \rightarrow (x \rightarrow y)) \rightarrow z;$
- 6) $(x \equiv y) \equiv z;$
- 7) $(\neg x \rightarrow \neg y) \rightarrow (xy \rightarrow xz);$
- 8) $((x \rightarrow y) \rightarrow \neg x) \rightarrow (x \rightarrow yx);$
- 9) $\neg(x \vee y \rightarrow \neg x) \neg(xy \rightarrow \neg y);$
- 10) $(z \rightarrow x) \rightarrow (\neg(y \vee z) \rightarrow x);$
- 11) $\neg(xy \rightarrow x) \vee (x(y \vee x));$
- 12) $\neg(x(y \vee z)) \rightarrow (xy \vee z);$
- 13) $((x \rightarrow y) \rightarrow (z \rightarrow \neg x)) \rightarrow (\neg y \rightarrow \neg z);$
- 14) $(((((x \rightarrow y) \rightarrow \neg x) \rightarrow \neg y) \rightarrow \neg z) \rightarrow z);$
- 15) $(x \rightarrow (y \rightarrow z)) \rightarrow ((x \rightarrow \neg z) \rightarrow (x \rightarrow \neg y)).$

Завдання 1.1.3. Визначити, чи є булева формула тавтологією.

- 1) $(y \downarrow \neg(x \rightarrow z)) | ((\neg x \rightarrow \neg y) \rightarrow z);$
- 2) $(y \downarrow (\neg z \oplus y)) \rightarrow (((x \equiv y) \equiv z) \oplus y) | y);$
- 3) $(y \downarrow (x \rightarrow z)) | (((y \rightarrow x) \equiv \neg z) \oplus (((x \vee y) \rightarrow y) | x));$
- 4) $(x \oplus (x \rightarrow z)) | ((y \equiv z) \downarrow x)$
- 5) $(x \rightarrow (y \rightarrow z)) \rightarrow ((x \rightarrow \neg z) \rightarrow (x \rightarrow \neg y));$
- 6) $\neg(x \rightarrow (\neg x \equiv z)) (y \rightarrow \neg z) \vee \neg(x \rightarrow z);$
- 7) $(x(y \vee z)) \rightarrow ((x \equiv y) \vee (z \rightarrow x));$
- 8) $\neg(x \vee y \vee z) \rightarrow (\neg(xy) \equiv \neg(x \vee z));$
- 9) $(b \rightarrow a \vee c) \rightarrow ((b \rightarrow c) \rightarrow ((d \rightarrow c) \rightarrow (b \vee d \rightarrow c)));$
- 10) $(x \equiv \neg y) \vee (y \rightarrow xz) \vee (x \neg y \neg z).$

Завдання 1.1.4. Розв'язати рівняння булевої алгебри.

Вказівка. Для розв'язання рівнянь потрібно побудувати таблиці істинності для булевих формул з лівої і правої частин рівняння і визначити набори з однаковими значеннями функцій.

- 1) $(x \vee \neg yz)(xz \rightarrow z) = 0;$
- 2) $(x \vee y \neg z)(x \vee z) = 1;$
- 3) $[(xy \equiv z) \rightarrow y] = [x \neg y \vee \neg xz];$
- 4) $[xy \vee \neg yz] = [(x \oplus y) \rightarrow z];$
- 5) $[(xy \equiv z) \rightarrow y] = [(x \oplus y) \rightarrow z];$
- 6) $[y \equiv (\neg xz)] = 0.$

1.2. Властивості булевих функцій/ Алгебра Жегалкіна.

Теоретичні відомості

Визначення 1.12. Система елементів $\{0,1\}$, на якій визначені операції $\wedge$ (кон'юнкція) і $\oplus$ (сума за *mod* 2), для яких виконуються співвідношення:

$$(1.21) \quad x \oplus y = y \oplus x,$$

$$(1.22) \quad x (y \oplus z) = xy \oplus xz,$$

$$(1.23) \quad x \oplus x = 0,$$

$$(1.24) \quad x \oplus 0 = x,$$

а також співвідношення булевої алгебри, що відносяться до кон'юнкції і констант, називається *алгеброю Жегалкіна*.

З таблиці істинності операції суми за mod 2 слідує, що

$$(1.24) \quad \neg x = x \oplus 1.$$

Операцію диз'юнкції можна виразити через $\oplus$ і $\wedge$ так:

$$(1.25) \quad x \vee y = \neg(\neg x \neg y) = (x \oplus 1)(y \oplus 1) \oplus 1 = xy \oplus x \oplus y.$$

Визначення 1.13. Будь-яка формула алгебри Жегалкіна, що має вигляд суми (за mod 2) кон'юнкції булевих змінних, називається *поліномом Жегалкіна*.

Якщо в кожен член полінома Жегалкіна кожна змінна входить один раз і поліном не містить однакових членів, то такий поліном Жегалкіна називається *канонічним*.

Теорема 1.6. Будь-яка булева функція може бути єдиним способом предсталена у вигляді канонічного поліному Жегалкіна.

Правило для представлення булевої функції у вигляді полінома Жегалкіна: в булевої формулі, що задана у вигляді ДДНФ, досить замінити знак $\vee$ на знак $\oplus$, представити заперечення змінних як $\neg x = x \oplus 1$, розкрити дужки згідно із законом дистрибутивності (1.22) і привести подібні члени згідно (1.23, 1.24).

Приклад. Зведемо до канонічного поліному Жегалкіна булеву функцію: $f(x, y, z) = x\neg yz \vee x\neg y\neg z \vee xyz$. Оскільки функція знаходиться в ДДНФ, замінимо символи $\vee$ на $\oplus$, отримаємо:

$$\begin{aligned} f(x, y, z) &= x\neg yz \oplus x\neg y\neg z \oplus xyz = x(y \oplus 1)z \oplus x(y \oplus 1)(z \oplus 1) \oplus xyz = \\ &= xyz \oplus xz \oplus xyz \oplus xz \oplus xy \oplus x \oplus xyz = xyz \oplus xy \oplus x. \end{aligned}$$

Привести функцію до канонічного поліному Жегалкіна (КПЖ) можна і через суперпозицію відповідних поліномів функцій, які складають вихідну.

Наприклад, для $x \rightarrow y$ ДДНФ є $\neg x \neg y \vee \neg x y \vee x y$. Після перетворення ми отримаємо КПЖ $xy \oplus x \oplus 1$.

З іншої сторони, $x \rightarrow y = \neg x \vee y = (x \oplus 1) y \oplus (x \oplus 1) \oplus y = xy \oplus x \oplus 1$. Або $x \rightarrow y = \neg(x \neg y) = 1 \oplus x(1 \oplus y) = xy \oplus x \oplus 1$.

Тоді для функції з попереднього прикладу $(x \oplus y) \rightarrow xz$ по ДДНФ $xyz \vee xy \neg z \vee \neg x \neg yz \vee \neg x \neg y \neg z \vee x \neg yz$ побудуємо КПЖ $xyz \oplus xz \oplus x \oplus y \oplus 1$. Або, замінивши імплікацію на відповідний КПЖ ($\oplus$ і $\&$ допустимі операції в алгебрі Жегалкіна), отримаємо

$$(x \oplus y) \rightarrow xz = (x \oplus y)xz \oplus (x \oplus y) \oplus 1 = xyz \oplus xz \oplus x \oplus y \oplus 1.$$

Існують і інші способи побудови КПЖ: метод трикутника (трикутник Паскаля), перетворення Мебіуса.

Метод трикутника дозволяє перетворити таблицю істинності в поліном Жегалкіна шляхом побудови допоміжної трикутної таблиці згідно з такими правилами:

- Будується повна таблиця істинності.
- Будується допоміжна горизонтальна таблиця, в якій перший рядок збігається зі стовпцем значень функції в таблиці істинності.
- В кожному наступному рядку комірка отримується шляхом додавання за модулем 2 двох комірок попереднього рядка, які стоять в тому самому стовпчику та праворуч (перша комірка попереднього рядка додається до другої комірки, потім друга до третьої і т.д.).
- Рядки допоміжної таблиці нумеруються двійковими кодами в тому ж самому порядку, що й рядки таблиці істинності.
- Кожному двійковому коду ставиться у відповідність один із членів полінома Жегалкіна в залежності від позиції кодів-змінних, в яких стоять одиниці. Наприклад, комірки 111 відповідає член xyz , комірки 101 — член xz , комірки 010 — член y , комірки 000 — член $\text{const } 1$ і т. д.
- Якщо на першому місці будь-якого рядка стоїть одиниця, саме тоді відповідний член є в поліномі Жегалкіна.

Приклад на метод трикутника для функції $(x \oplus y) \rightarrow xz = 11000111$.

x	y	z									
0	0	0	1	1	1	0	0	0	1	1	1
0	0	1		0	1	0	0	1	0	0	
0	1	0	y	1	1	0	1	1	0		
0	1	1		0	1	1	0	1			
1	0	0	x	1	0	1	1				
1	0	1	xz	1	1	0					
1	1	0		0	1						
1	1	1	xyz	1							

Відповідь: $xyz \oplus xz \oplus x \oplus y \oplus 1$

Властивості булевих функцій

Визначення 1.14. Функція, яка виражена поліномом Жегалкіна вигляду $\sum \alpha_i x_i \oplus \gamma$, де $\alpha_i, \gamma \in 0$ або 1 , називається *лінійною* (L).

Всі функції однієї змінної лінійні. Лінійними функціями двох змінних є $x \oplus y$ і $x \equiv y$: $x \equiv y = \neg x \neg y \vee xy = (x \oplus 1)(y \oplus 1) \oplus xy = xy \oplus x \oplus y \oplus 1 \oplus xy = x \oplus y \oplus 1$.

Визначення 1.15. Нехай дана деяка булева формула A. Формула A* називається *двоїстою* формулі A, якщо вона виходить з A шляхом заміни операцій кон'юнкції на диз'юнкцію, диз'юнкції на кон'юнкцію, 1 на 0, 0 на 1 усюди, де вони входять.

Наприклад: $A = xy \vee \neg xy$; $A^* = (x \vee y)(\neg x \vee y)$.

Теорема 1.7. Якщо $A(x_1, \dots, x_n)$ и $A^*(x_1, \dots, x_n)$ – дві взаємнодвоїсті формули, то $\neg A(x_1, \dots, x_n) = A^*(\neg x_1, \dots, \neg x_n)$.

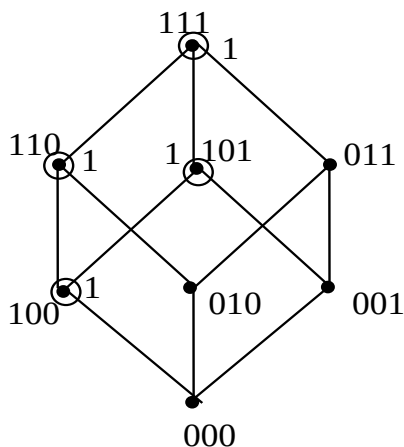
Визначення 1.16. Функція називається *самодвоїстою* (S), якщо вона двоїста сама собі, тобто $A(x_1, x_2, \dots, x_n) = \neg A(\neg x_1, \neg x_2, \dots, \neg x_n)$, або $\neg A(x_1, x_2, \dots, x_n) = A(\neg x_1, \neg x_2, \dots, \neg x_n)$.

Для самодвоїстої функції заперечення її аргументів призводить до заперечення самої функції, отже, самодвоїста функція на протилежних наборах приймає протилежні значення. Наприклад, функція $xy \vee xz \vee yz$ самодвоїста. Щоб показати це, візьмемо заперечення від кожної змінної і від всієї функції::

$$\begin{aligned} \neg(\neg x \neg y \vee \neg x \neg z \vee \neg y \neg z) &= (x \vee y)(x \vee z)(y \vee z) = (x \vee xy \vee xz \vee yz)(y \vee z) = \\ &= xy \vee xy \vee xyz \vee yz \vee xz \vee xyz \vee xz \vee yz = xy \vee xz \vee yz. \end{aligned}$$

Відношення порядку на наборах булевих змінних визначається так. Розглянемо два набори: $A = (\alpha_1, \dots, \alpha_i, \dots, \alpha_n)$ і $B = (\beta_1, \dots, \beta_i, \dots, \beta_n)$. Якщо для всіх i ($i = 1, \dots, n$) $\alpha_i \leq \beta_i$, і існує хоча б одне таке j , при якому $\alpha_j < \beta_j$, набір A передуює (менше) набору B . Це позначається: $A \leq B$. Якщо для деяких i $\alpha_i \leq \beta_i$ і існує таке j , що $\alpha_j > \beta_j$, то набори A і B *непорівнювані*. Наприклад: набори (0101) і (0010) непорівнювані, а набір (0101) передуює набору (0111).

Тоді булеву функцію можна задати на цій решітці, як показано на рис. 1.1. Причому серед булевих функцій можна виділити такі, які не зменшуються зі зростанням значень порівнянних наборів на булевої решітці. Такі функції називаються монотонними. Наприклад, на рис. 1.1. показана монотонна функція (00001111).



Визначення 1.17. Булева функція f називається *монотонною* (M), якщо для будь-яких двох наборів A і B з її області визначення таких, що $A \leq B$, $f(A) \leq f(B)$. Якщо хоча б для однієї пари наборів таких, що $A \leq B$, $f(A) > f(B)$, то функція немонотонна.

Рис. 1.1. Монотонна
функція.

Перевірити на монотонність булеву функцію можна в наступний спосіб:

Задана таблиця істинності булевої функції.

Крок 1. Порівнюємо значення функції на наборах, сусідніх першої змінної, тобто верхню половину стовпця значень функції (вектор φ_1) з нижньою половиною (вектор ψ_1). Якщо умова $\varphi_1 \leq \psi_1$ порушена, то функція не монотонна, йдемо на кінець.

Крок 2. Порівнюємо значення функції на наборах, сусідніх за другою змінною, тобто верхні чвертини стовпця значень функції (вектори φ'_2, φ''_2) із нижніми чвертинами (векторами ψ'_2, ψ''_2) у кожній половині. Якщо хоча б одна з умов $\varphi'_2 \leq \psi'_2$ або $\varphi''_2 \leq \psi''_2$ порушена, то функція не монотонна, йдемо на кінець.

Кроки 3-н. Аналогічно порівнюємо восьми, шістнадцяті частини, і так далі. Якщо жодна з умов, що перевіряються, не порушена, то функція монотонна.

Розглянемо дві булеві функції.

x	y	z	f(x,y,z)	g(x,y,z)	
0	0	0	0	0	φ'_2
0	0	1	0	1	
0	1	0	0	1	ψ'_2
0	1	1	1	0	φ_1

1	0	0	0	0	ψ_1
1	0	1	1	1	φ''_2
1	1	0	1	1	ψ''_2
1	1	1	1	1	

Перевіримо на монотонність першу функцію. Порівнюємо половини стовпця значень: $0001 \leq 0111$. Порівнюємо чвертини: $00 \leq 01$, $01 \leq 11$. Порівнюємо восьмі частини: $0 \leq 0$, $0 \leq 1$ та $0 \leq 1$, $1 \leq 1$. Отже, функція $f(x,y,z)$ – монотонна.

Перевіримо на монотонність функцію $g(x,y,z)$. Порівнюємо половини стовпця значень: $0110 \leq 0111$. Порівнюємо чвертини: $01 \leq 10$, $01 \leq 11$. Порівнюємо восьмі частини: $0 \leq 1$ але на другій позиції у першій чвертині 1, а у другій 0 – властивість $\leq$ не виконується, функція $g(x,y,z)$ не монотонна.

Визначення 1.18. Булева функція називається *функцією, що зберігає нуль* (T_0), якщо на нульовому наборі вона дорівнює нулю, тобто $f(0,\dots,0) = 0$.

Визначення 1.19. Булева функція називається *функцією, що зберігає одиницю* (T_1), якщо на одиничному наборі вона дорівнює одиниці, тобто $f(1,\dots,1) = 1$.

Властивості основних функцій наведено в табл.1.6.

Таблиця 1.6.

		КПЖ	T ₀	T ₁	M	S	L
f_0 –	константа 0	0	+	-	+	-	+
f_1 –	$x \wedge y$ – кон’юнкція	xy	+	+	+	-	-
f_3 –	тотожна функція $f(x,y) = x$	x	+	+	+	+	+
f_6 –	$x \oplus y$ – сума за <i>mod</i> 2 (операція Жегалкіна)	$x \oplus y$	+	-	-	-	+
f_7 –	$x \vee y$ – диз’юнкція	$xy \oplus x \oplus y$	+	+	+	-	-
f_8 –	$x \downarrow y$ – стрілка Пірса	$xy \oplus x \oplus y \oplus 1$	-	-	-	-	-
f_9 –	$x \equiv y$ (або $x \sim y$) – еквівалентність	$x \oplus y \oplus 1$	-	+	-	-	+
f_{12} –	заперечення $f(x) = \neg x$	$x \oplus 1$	-	-	-	+	+
f_{13} –	$x \rightarrow y$ – імплікація	$xy \oplus x \oplus 1$	-	+	-	-	-
f_{14} –	$x y$ – штрих Шеффера	$xy \oplus 1$	-	-	-	-	-
f_{15} –	константа 1	1	-	+	+	-	+

Завдання 1.2.1. Визначити властивості булевих функцій.

Приклад. $(x \rightarrow (y \vee z)) \rightarrow xz$.

Таблиця істинності.

x	y	z	$f(x, y, z)$	конституенти
0	0	0	0	$x \vee y \vee z$
0	0	1	0	$x \vee y \vee \neg z$
0	1	0	0	$x \vee \neg y \vee z$
0	1	1	0	$x \vee \neg y \vee \neg z$
1	0	0	1	$x \neg y \neg z$
1	0	1	1	$x \neg y z$
1	1	0	0	$\neg x \vee \neg y \vee z$
1	1	1	1	xyz

ДДНФ: $f(x, y, z) = x \neg y \neg z \vee xyz \vee x \neg y z$.

Перевіримо властивості функції.

1. Функція зберігає нуль, так як $f(0,0,0) = 0$.
2. Функція зберігає одиницю, так як $f(1,1,1) = 1$.

3. Самодвоїстість. За визначенням, функція є самодвоїстою, якщо $f(\neg x, \neg y, \neg z) = \neg f(x, y, z)$, тобто на протилежних наборах функція приймає протилежні значення. Набір 001 є протилежним до набору 110, але $f(0,0,1) = 0$ і $f(1,1,0) = 0$, отже, функція несамодвоїста. Взагалі, самодвоїста функція має однакову кількість 0 та 1 в таблиці істинності.

4. Монотонність. Функція монотонна тоді, коли на зростаючих порівняних наборах вона не спадає.

Набір 100 передуює набору 110, але $f(1,0,0) = 1 > f(1,1,0) = 0$, отже, функція немонотонна.

5. Лінійність. Щоб перевірити, чи є функція лінійною, побудуємо поліном Жегалкіна. Для цього у ДДНФ даної функції замінимо усі символи $\vee$ на $\oplus$ и виразимо заперечення як $\neg x = x \oplus 1$. Отримаємо поліном Жегалкіна:

$$f(x, y, z) = x\neg y\neg z \vee xyz \vee x\neg yz = (x(y \oplus 1)(z \oplus 1)) \oplus (xyz) \oplus (x(y \oplus 1)z).$$

Приведемо його до канонічного вигляду, використовуючи закон дистрибутивності: $x(y \oplus z) = xy \oplus xz$ і тотожності: $x \oplus x = 0$, $x \oplus 0 = x$. Тоді

$$f(x, y, z) = (x(y \oplus 1)(z \oplus 1)) \oplus (xyz) \oplus (xz(y \oplus 1)) = ((xy \oplus x)(z \oplus 1)) \oplus xyz \oplus \oplus (xyz \oplus xz) = xyz \oplus xy \oplus xz \oplus x \oplus xyz \oplus xyz \oplus xz = xyz \oplus xy \oplus x.$$

Функція нелінійна, так як у канонічному поліномі Жегалкіна є нелінійні елементи: xyz та xy . Даний спосіб побудови канонічного поліному Жегалкіна показує, що непарної кількості одиниць у таблиці істинності булева функція не буде лінійною, бо завжди залишиться елемент xyz .

Отримати поліном Жегалкіна можна й алгебраїчно через суперпозицію основних функцій і їх уявлення канонічним поліномом Жегалкіна:

$$\begin{aligned} (x \rightarrow y &= xy \oplus x \oplus 1; \quad x \vee y = xy \oplus x \oplus y). \\ (x \rightarrow (y \vee z)) \rightarrow xz &= (x \rightarrow (y \vee z)) xz \oplus (x \rightarrow (y \vee z)) \oplus 1 = \\ &= (x(yz \oplus y \oplus z) \oplus x \oplus 1)xz \oplus (x(yz \oplus y \oplus z) \oplus x \oplus 1) \oplus 1 = \end{aligned}$$

$$\begin{aligned}
&= (xyz \oplus xy \oplus xz \oplus x \oplus 1)xz \oplus xyz \oplus xy \oplus xz \oplus x \oplus 1 \oplus 1 = xz \oplus xyz \oplus xy \oplus xz \oplus \\
&x = \\
&= xyz \oplus xy \oplus x.
\end{aligned}$$

Або за спрощеною формулою:

$$x \neg y \vee xz = x(y \oplus 1)xz \oplus x(y \oplus 1) \oplus xz = xyz \oplus xz \oplus xy \oplus x \oplus xz = xyz \oplus xy \oplus x.$$

- 1) $xy \vee \neg x (y \vee xz) \neg (x \neg y \vee yz)$
- 2) $x (xy \vee \neg(yz) \vee \neg(y \vee \neg zt))$
- 3) $x \neg (\neg yz) \vee \neg (x \vee z)$
- 4) $\neg (\neg (\neg x \vee \neg yz) \vee \neg x \vee \neg yz) \vee (\neg x \vee \neg yz)(\neg z \vee y)$
- 5) $(xy \equiv z) \rightarrow y$
- 6) $f(x,y,z)=1$ на наборах 1,2,4,6
- 7) $f(x,y,z)=0$ на наборах 1,4,5
- 8) $(x \oplus y) \rightarrow z$
- 9) $(y \vee z) \equiv \neg x$
- 10) $(x \rightarrow y) \oplus (y \rightarrow z) \oplus (z \rightarrow x)$
- 11) $(x \equiv \neg y) \vee (y \rightarrow xz) \vee (x \neg y \neg z)$
- 12) $(x \rightarrow (x \rightarrow y)) \rightarrow z$
- 13) $(x \vee \neg yz)(xz \rightarrow z)$
- 14) $((y \vee z) \equiv \neg x)(xz \rightarrow \neg y)$
- 15) $(x \vee y \neg z)(x \vee z)$

Завдання 1.2.2. Побудувати поліном Жегалкіна і визначити властивості булевих функцій із завдання 1.1.2.

1.3. Функціональна повнота систем булевих функцій

Теоретичні відомості

Замкнуті класи булевих функцій

Нехай функція $f(x)$ представлена формулою $F(x)$, а функція g представлена формулою G . Тоді формула $F(G)$, яка виходить шляхом заміни кожного входження змінної x в формулі $F(x)$ на формулу G , представляє суперпозицію булевих функцій $f(g)$.

Визначення 1.20. Клас функцій називається *функціонально замкнутим*, якщо суперпозиція цих функцій належить даному класу.

Теорема 1.8. Суперпозиція лінійних функцій є функція лінійна.

Теорема 1.9. Суперпозиція монотонних функцій є функція монотонна. Отже, клас монотонних функцій є функціонально замкнутим.

Теорема 1.10. Клас функцій, що зберігають нуль, є *функціонально замкнутим*.

Теорема 1.11. Клас функцій, що зберігають одиницю, є функціонально замкнутим.

Теорема 1.12. Клас самодвоїстих функцій є функціонально замкнутим.

Визначення 1.21. Система булевих функцій називається *функціонально повною*, якщо будь-яка булева функція може бути представлена як суперпозиція функцій з цієї системи.

Позначимо: T_0 – клас функцій, що зберігають 0; T_1 – клас функцій, що зберігають 1; S – клас самодвоїстих функцій; M – клас монотонних функцій; L – клас лінійних функцій.

Теорема 1.13 (Поста). Для того, щоб система функцій була повна, необхідно і достатньо, щоб вона містила хоча б одну немонотонну, хоча б одну нелінійну, хоча б одну несамодвоїсту, хоча б одну, що не зберігає нуль, і хоча б одну, не зберігаючу одиницю, функцію.

Приклад. Системи функцій $\{\vee, \wedge, \neg\}$, $\{\oplus, \wedge, 0, 1\}$, $\{\rightarrow, \neg\}$ являються функціонально повними. Для прикладу перевіримо повноту системи функцій $\{\neg, \rightarrow\}$. Для досліджуваної системи складемо таблицю Поста – критерійну таблицю (табл.1.7). Якщо функція входить у функціонально замкнутий клас, то в таблиці Поста у відповідній чарунці ставиться знак '+', інакше – знак '-'.

Таблиця 1.7

	T_0	T_1	S	M	L
$\neg$	-	-	+	-	+
$\rightarrow$	-	+	-	-	-

Система функцій $\{\neg, \rightarrow\}$ повна, так як в кожному стовпці таблиці Поста є хоч би один знак '-'.

Повну систему називають неприводимою або базисом, якщо з неї не можна виключити жодної функції без того, щоб система не втратила властивості повноти.

Теорема 1.14. Будь-яка неприводима функціональна система булевих функцій складається з не більш ніж чотирьох функцій.

Функціонально повна система називається незалежною, якщо жодна з функцій системи не може бути суперпозицією інших функцій цієї системи. Базис – незалежний. Наприклад система $\{\oplus, \wedge, 0, 1\}$ є повною, але не є незалежною. Базисом буде система $\{\oplus, \wedge, 1\}$, так як $0=1 \oplus 1$ (табл.1.8).

Таблиця 1.8

	T_0	T_1	S	M	L
$\oplus$	+	-	-	-	+
$\wedge$	+	+	-	+	-
1	-	+	-	+	+
0	+	-	-	+	+

Тезу про повноту можна узагальнити. Якщо усі функції функціонально повної системи можна виразити через супрепозицію функцій іншої системи, то вона так само буде функціонально повною. Цей метод називається методом зведення. Наприклад, система алгебри Жегалкіна $\{\oplus, \wedge, 1\}$ є функціонально повною. Система $\{\equiv, \vee, 0\}$ також буде функціонально повною. Дійсно, можна виразити $x \oplus y = (x \equiv y) \equiv 0$, $x \wedge y = (x \vee y) \equiv (x \equiv y)$, $1 = 0 \equiv 0$. Та переконатися, побудувавши критеріальну таблицю (табл.1.9):

Таблиця 1.9

	T_0	T_1	S	M	L
$\equiv$	—	+	—	—	+
$\vee$	+	+	—	+	—
0	+	—	—	+	+

Завдання 1.3.1.

1) Довести, чи можна виразити за допомогою суперпозицій:

- a) $\neg$ через $\&$, $\vee$, $\rightarrow$, та $\equiv$;
- b) $\rightarrow$ через $\&$ і $\vee$;
- c) $\neg$ через $+$ і 1 ;
- d) $\neg$ через $\rightarrow$ і 0 ;
- e) $\neg$ через $\rightarrow$ і 1 ;

2) Виразити за допомогою суперпозицій, попередньо перевірив систему функцій на повноту:

- a) $\rightarrow$ через $\neg$ і $\&$;
- b) $\rightarrow$ через $\neg$ і $\vee$;
- c) $\&$ через $\neg$ і $\rightarrow$;
- d) $\vee$ через $\neg$ і $\rightarrow$;
- e) $\neg$, $\&$ та $\vee$ через $\{\rightarrow, 0\}$;

f) & через $\{+, 1, V\}$.

Завдання 1.3.2. Дослідити системи функцій на повноту та знайти базис:

	Функції системи I	Функції системи II	Функції системи III
1	$x \rightarrow (x \rightarrow z)$	$x \rightarrow (y \rightarrow z)$	$(x \rightarrow y) \rightarrow z$
2	$yx \vee \neg yz$	$xy \oplus xz \oplus yz$	$xy \vee xz \vee yz$
3	$(x \oplus y) \oplus z$	$(x \equiv y) \oplus z$	$(x \equiv y) \equiv z$
4	$\neg y \vee yz$	$\neg x \neg y \vee \neg y \neg z \vee \neg x \neg z$	$y \vee yz$
5	$x \neg y \vee \neg xz$		

Завдання 1.3.3. Виразити основні функції булевої алгебри $\{\neg, \&, V, 0, 1\}$ через функціонально повну систему, що складається з однієї функції $\{\downarrow\}$ або $\{\downarrow\}$.

Відповіді до розділу 1

1.1.1. Не еквівалентні формули: 1, 4, 5, 7, 9.

1.1.2.

№	спрощення	значення	КПЖ	T ₀	T ₁	S	M	L
1)	$\neg x \neg y \neg z$	10000000	$xyz \oplus xy \oplus xz \oplus$ $\oplus yz \oplus x \oplus y \oplus z \oplus 1$	-	-	-	-	-
2)	$\neg x \neg y \neg z \vee x \neg yz$	10000100	$xy \oplus yz \oplus x \oplus y \oplus z$ $\oplus 1$	-	-	-	-	-
3)	$xy \vee xz$	00000111	$xyz \oplus xy \oplus xz$	+	+	-	+	-
4)	$x \rightarrow z$	11110101	$xz \oplus x \oplus 1$	-	+	-	-	-
5)	$x \neg y \vee z$	01011101	$xyz \oplus xy \oplus xz \oplus$ $\oplus x \oplus z$	+	+	-	-	-
6)	$(x \equiv y) \equiv z$	01101001	$x \oplus y \oplus z$	+	+	+	-	+
7)	$\neg x \vee \neg y \vee z$	11111101	$xyz \oplus xy \oplus 1$	-	+	-	-	-
8)	$\neg x \vee y$	11110011	$xy \oplus x \oplus 1$	-	+	-	-	-
9)	xy	00000011	xy	+	+	-	+	-
10)	$x \vee y \vee z$	01111111	$xyz \oplus xy \oplus xz \oplus$ $\oplus yz \oplus x \oplus y \oplus z$	+	+	-	+	-
11)	x	00001111	x	+	+	+	+	+
12)	$xy \vee z$	01010111	$xyz \oplus xy \oplus z$	+	+	-	+	-
13)	$y \vee \neg z$	10111011	$yz \oplus z \oplus 1$	-	+	-	-	-
14)	z	01010101	z	+	+	+	+	+
15)	1	11111111	1	-	+	-	+	+

1.1.3. Не тавтологіями є формули 4, 6 і 10.

1.1.4. Підказка для формування відповіді:

$(x \vee \neg yz)(xz \rightarrow z)$	01001111
$(x \vee y \neg z)(x \vee z)$	00001111
$(xy \equiv z) \rightarrow y$	01110111
$y \equiv (\neg xz)$	10011100

$(x \oplus y) \rightarrow z$	11010111
$x \neg y \vee \neg xz$	01011100
$xy \vee \neg yz$	01000111

1.2.1.

№	значення	КПЖ	T_0	T_1	S	M	L
1)	00100011	$xyz+yz+y$	+	+	-	-	-
2)	0000000011111111	x	+	+	+	+	+
3)	10101011	$xyz+z+1$	-	+	-	-	-
4)	10110000	$xyz+xz+yz+x+z+1$	-	-	-	-	-
5)	01110111	$yz+y+z$	+	+	-	+	-
6)	01101010	$xy+x+y+z$	+	-	-	-	-
7)	10110011	$xyz+xy+zy+xz+x+z+1$	-	+	-	-	-
8)	11010111	$xz+yz+x+y+1$	-	+	-	-	-
9)	01111000	$yz+x+y+z$	+	-	-	-	-
10)	10000001	$xy+yz+xz+x+y+z+1$	-	+	-	-	-
11)	11111101	$xyz+xy+1$	-	+	-	-	-
12)	01011101	$xyz+xz+xy+x+z$	+	+	-	-	-
13)	01001111	$xyz+xz+yz+x+z$	+	+	-	-	-
14)	01111000	$yz+x+y+z$	+	-	-	-	-
15)	00001111	x	+	+	+	+	+

1.2.2. Див. відповідь для завдання 1.1.2.

1.3.1.

1)

- a) ні, всі T_1 ;
- b) ні, всі T_0, M ;
- c) так, всі лінійні;
- d) так, $\neg x = x \rightarrow 0$;

2)

- a) $x \rightarrow y = \neg (x \& \neg y)$
- b) $x \rightarrow y = \neg x \vee y$
- c) $x \& y = \neg (x \rightarrow \neg y)$
- d) $x \vee y = \neg x \rightarrow y$

е) ні, обидві T_1 .

$$\text{е) } \neg x = x \rightarrow 0, x \& y = \neg (x \rightarrow \neg y) = \\ = \neg (x \rightarrow (y \rightarrow 0)) = (x \rightarrow (y \rightarrow 0)) \rightarrow 0,$$

$$x \vee y = \neg x \rightarrow y = (x \rightarrow 0) \rightarrow y$$

$$\text{ф) } x \& y = \neg (\neg x \vee \neg y) = 1 + (1+x) \vee (1+y)$$

1.3.2.

I) (1 або 4) та 5;

	Функції системи I	T_0	T_1	S	M	L	Таблиця іст.
1	$x \rightarrow (x \rightarrow z)$	-	+	-	-	-	11110101
2	$yx \vee \neg yz$	+	+	-	-	-	01000111
3	$(x \oplus y) \oplus z$	+	+	+	-	+	01101001
4	$\neg y \vee yz$	-	+	-	-	-	11011101
5	$x \neg y \vee \neg xz$	+	-	-	-	-	01011100

II) 1 та (3 або 4);

	Функції системи II	T_0	T_1	S	M	L	Таблиця іст.
1	$x \rightarrow (y \rightarrow z)$	-	+	-	-	-	11111101
2	$xy \oplus xz \oplus yz$	+	+	+	+	-	00010111
3	$(x \equiv y) \oplus z$	-	-	+	-	+	10010110
4	$\neg x \neg y \vee \neg y \neg z \vee \neg x \neg z$	-	-	+	-	-	11101000

II) не повна, базис не існує.

	Функції системи III	T_0	T_1	S	M	L	Таблиця іст.
1	$(x \rightarrow y) \rightarrow z$	+	+	-	-	-	01011101
2	$xy \vee xz \vee yz$	+	+	+	+	-	00010111
3	$(x \equiv y) \equiv z$	+	+	+	-	+	01101001
4	$y \vee yz$	+	+	+	+	+	00110011

1.3.3.

Для штриха Шеффера:

$$\neg x = \neg (x \& x) = x | x$$

$$x \& y = \neg \neg (x \& y) = \neg (x | y) = (x | y) | (x | y);$$

$$x \vee y = \neg \neg (x \vee y) = \neg (\neg x \& \neg y) = (\neg x) | (\neg y) = (x | x) | (y | y);$$

$$0 = x \& \neg x = \neg \neg (x \& \neg x) = \neg (x | (\neg x)) = \neg (x | (x | x)) = (x | (x | x)) | (x | (x | x));$$

$$1 = x \vee \neg x = (x | x) | (\neg x | \neg x) = (x | x) | ((x | x) | (x | x)) \quad \text{АЛЕ}$$

$$1 = \neg 0 = \neg \neg (x | (\neg x)) = x | (x | x)$$

Для стрілки Пірса:

$$\neg x = \neg (x \vee x) = x \downarrow x$$

$$x \vee y = \neg \neg (x \vee y) = \neg (x \downarrow y) = (x \downarrow y) \downarrow (x \downarrow y);$$

$$x \& y = \neg \neg (x \& y) = \neg (\neg x \vee \neg y) = (\neg x) \downarrow (\neg y) = (x \downarrow x) \downarrow (y \downarrow y);$$

$$1 = x \vee \neg x = \neg \neg (x \vee \neg x) = \neg (x \downarrow (\neg x)) = \neg (x \downarrow (x \downarrow x)) = (x \downarrow (x \downarrow x)) \downarrow (x \downarrow (x \downarrow x));$$

$$0 = x \& \neg x = (x \downarrow x) | (\neg x \downarrow \neg x) = (x \downarrow x) \downarrow ((x \downarrow x) \downarrow (x \downarrow x))$$

$$\text{АЛЕ } 0 = \neg 1 = \neg \neg (x \downarrow (\neg x)) = x \downarrow (x \downarrow x)$$

2. ТЕОРІЯ ВИСЛОВЛЮВАНЬ

2.1. Основні поняття

Теоретичні відомості

Логіка, вивчаючи правильні міркування, оперує поняттями істинності і хибності. Правильне міркування або висловлювання покладається істинним, неправильне – хибним; для формального апарату представлення знань нам досить двозначної логіки.

Методологічні основи формальної логіки полягають у виконанні декількох принципів.

Принцип тотожності. Істинність фактів, що лежать в основі висловлювань і міркувань, встановлюється на підставі відомих законів, спостережень. Якщо істинність будь-якого факту встановлено, то вона не піддається сумніву і не змінюється в процесі міркування. Це означає також, що один і той же термін використовується завжди в одному і тому ж сенсі.

Принцип несуперечності означає, що, стверджуючи що-небудь, не можна заперечувати те ж саме. Один і той же факт (вислів) не може бути одночасно істинним і хибним.

Принцип виключення третього. Не можна одночасно відкидати висловлювання і його заперечення. Будь-яке висловлювання може бути або істинним, або хибним, – третього не дано.

Принцип достатньої підстави. Будь-яке висловлювання має бути обґрунтовано, тобто істинність твердження не можна приймати на віру. Якщо твердження виводиться з будь-яких суджень, даних, фактів – *підстав*, то їх повинно бути досить для встановлення істинності твердження.

Можна вказати ще на одну дуже важливу властивість - **монотонність достовірних міркувань**. Якщо істинність деякого висловлювання (*A*) вже встановлена, то додавання нових фактів (*B*) не змінює істинності *A*.

Визначення 2.1. *Просте висловлювання* – це просте розповідне речення, відносно якого можна однозначно сказати, істинне воно чи хибне.

Питальні або окличні речення висловлюваннями не є.

Логічні значення *Істинно (True)* чи *Хибно (False)* будемо позначати відповідно *T* и *F*.

Кожне просте висловлювання позначають символами латинського алфавіту (з індексами або без індексів), які називають *пропозиціональними символами*: $A, B, C, A_1, A_2 \dots$

Складні висловлювання складаються з простих за допомогою сполучників «ні», «і», «або», «якщо..., то...», «тоді і тільки тоді». Цим сполучникам відповідають логічні операції: унарна операція заперечення $\neg$ («ні»), бінарні операції кон'юнкції $\&$ («і»), диз'юнкція $\vee$ («або»), імплікація $\rightarrow$ («якщо..., то...»), еквівалентність $\equiv$ («тоді і тільки тоді»). Символи операцій називаються *пропозиціональними зв'язками*.

Приклади:

«Лондон – столиця Англії» $\Leftrightarrow A$. «Лондон не є столицею Англії» $\Leftrightarrow \neg A$. «Найбільше місто Англії, Лондон (A), є її столицею (B)» $\Leftrightarrow A \& B$.
«Населення Канади говорить англійською (A) або французькою мовою (B)» $\Leftrightarrow A \vee B$. «Якщо горобець – птаха (A), то у неї є крила (B)» $\Leftrightarrow A \rightarrow B$.
«Тварина є птахою (A) тоді і тільки тоді, коли у неї є крила (B)» $\Leftrightarrow A \equiv B$.

Істинність або хибність складних висловлювань залежить від істинності або хибності простих висловлювань, які входять до складних, а також тим способом, якими вони комбінуються, тобто зв'язкою, яку використовують для побудови складного висловлювання. Кожна логічна зв'язка визначається своєю *таблицею істинності* (табл. 2.1, 2.2).

Таблиця 2.1

A	$\neg A$
F	T
T	F

Таблиця 2.2

A	B	$A \& B$	$A \vee B$	$A \rightarrow B$	$A \equiv B$
F	F	F	F	T	T
F	T	F	T	T	F
T	F	F	T	F	F
T	T	T	T	T	T

За допомогою зв'язки **заперечення** $\neg$ із стверджувальних утворюються заперечні висловлювання.

Кон'юнкція $A \& B$, що відповідає сполучникам «і», «а», істинна в тому і тільки в тому випадку, якщо істинні обоє з висловлювань, що входять в неї. Наприклад, висловлювання «на вулиці йде дощ (A) із сильним вітром (B)» виражається формулою $A \& B$. Кон'юнкція комутативна, тому висловлювання «на вулиці сильний вітер (B) і дощ (A)», яке можна висловити формулою $B \& A$, еквівалентне першому.

За допомогою кон'юнкції формалізується принцип несуперечливості: $A \& \neg A \equiv F$ – **закон протиріччя**.

Диз'юнкція $A \vee B$, що відповідає сполучнику «або», є істинною в будь-якому випадку, коли істинне хоча б одне висловлювання, що входить в неї, і хибна тільки у випадку, коли обидва простих висловлювання хибні. Диз'юнкція формалізує **закон виключення третього**: $A \vee \neg A \equiv T$.

Імплікація $A \rightarrow B$ виражає логічний (часто – причинно-наслідковий) зв'язок між висловлюваннями A і B та формалізує звичайний умовивід, в якому з посилки (антецедента) A впливає висновок (консеквент) B . Таблиця істинності імплікації відображає правильні (достовірні) умовиводи.

Формальне висловлювання **принципу тотожності** можна записати так: $A \rightarrow A$ ($A \in A$). Формальне висловлювання принципу монотонності достовірних міркувань: $A \rightarrow (B \rightarrow A)$.

Еквівалентність $A \equiv B$ стверджує рівнозначність (рівносильність, тотожність) двох тверджень A і B ; вона істинна тоді і тільки тоді, коли істинностні значення A і B співпадають.

Еквівалентність $A \equiv B$ стверджує не тільки необхідність умови A для того, щоб було істинно B , але і достатність цієї умови, тобто $A \equiv B = (A \rightarrow B) \& (B \rightarrow A)$, где $A \rightarrow B$ виражає *необхідність* A , а $B \rightarrow A$ – *достатність* A .

За допомогою логічних зв'язок складні висловлювання можна записати у вигляді формули, яку називають *пропозиційальною формулою*.

Визначення 2.2.

- Кожна пропозиційальна буква є формула.
- Якщо A і B – формули, то формулами є: $(\neg A)$, $(A \& B)$, $(A \vee B)$.
- Інших формул немає.

При записі формул прийнято наступне: зовнішні дужки можна опускати; встановлено пріоритет операцій: $\neg$, $\&$, $\vee$, $\rightarrow$, $\equiv$. Логічні зв'язки для імплікації $\rightarrow$ та еквівалентності $\equiv$ вводяться для скорочення запису формул:

$$A \rightarrow B = \neg A \vee B = \neg(A \& \neg B), A \equiv B = (A \rightarrow B) \& (B \rightarrow A).$$

Побудувавши формулу алгебри висловлювань, ми відволікаємося від її змістовного сенсу і оперуємо тільки поняттями істинності та хибності.

Приписування пропозиційальним літерам їх істинностних значень називається *інтерпретацією формули*. Множина всіх інтерпретацій формули утворює її таблицю істинності. Якщо виконати відображення $0 \Leftrightarrow F$, $1 \Leftrightarrow T$, то кожній пропозиційальній зв'язці буде відповідати булева операція, а кожній формулі алгебри висловлювань – булева формула, з чого

впливає, що алгебра висловлювань є інтерпретацією булевої алгебри. В зв'язку з цим в ній зберігаються всі аксіоми і теореми булевої алгебри, зокрема й зображуваність формул алгебри висловлювань у вигляді ДДНФ і ДКНФ.

Визначення 2.3. Тотожно істинна формула називається *тавтологією*. Тотожно хибна формула називається *протиріччям*. Формула, яка приймає істинне значення хоча б на одній своїй інтерпретації, називається *здійсненою*. Формула, яка приймає на одних наборах істинні значення, а на якихось – хибні, називається *нейтральною*. Таким чином, формули розпадаються на три класи.



Тавтології є *виділеними* формулами логіки висловлювань; саме вони представляють найбільший інтерес. Тавтологія істинна при будь-яких значеннях простих висловлювань, що входять в неї; таблиця істинності тавтології в кожному рядку містить значення «істинне».

В формальній логіці поняття тавтології включає в себе всі тотожно істинні твердження. Вони формалізують правильні схеми міркувань.

Проблема можливості розв'язання в алгебрі висловлювань полягає в тому, щоб відшукати ефективну процедуру (алгоритм), за допомогою якої

для кожної формули логіки висловлювань можна встановити, чи є вона тавтологією, чи ні.

Очевидно, що така процедура для формул логіки висловлювань існує: це побудова таблиць істинності.

Приклади.

1. Як приклад розглянемо закон ствердження консеквента: $|= A \rightarrow (B \rightarrow A)$. (*принцип монотонності достовірних міркувань*). Оскільки кожен рядок таблиці істинності (табл. 2.3) цієї формули містить значення Т, формула є тавтологією. Також, $A \rightarrow (B \rightarrow A) = \neg A \vee \neg B \vee A = T \vee \neg B = T$.

Таблиця 2.3

A	B	$B \rightarrow A$	$A \rightarrow (B \rightarrow A)$
F	F	T	T
F	T	F	T
T	F	T	T
T	T	T	T

2. **Метод редукції (зведення до протиріччя)** – це спосіб скорочення переборів при складанні таблиці істинності. В якості приклада доведемо, що формула $(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$ – тавтологія.

Припустимо, що це не так, тобто існує така інтерпретація, на якій формула приймає хибне значення:

$$\overbrace{|(A \rightarrow (B \rightarrow C))|}^T \rightarrow \overbrace{(\underbrace{|(A \rightarrow B)|}_T \rightarrow \underbrace{|(A \rightarrow C)|}_F)}^F = F.$$

Маємо систему рівнянь :

- 1) $|A \rightarrow (B \rightarrow C)| = T,$
- 2) $|A \rightarrow B| = T,$

$$3) \quad |A \rightarrow C| = F.$$

З 3) випливає: $|A| = T$, $|C| = F$. Підставимо ці значення в 2): $|T \rightarrow B| = T$, звідки $|B| = T$. Підставимо знайдені значення $|A| = T$, $|C| = F$, $|B| = T$ в 1): $|T \rightarrow (T \rightarrow F)| = |T \rightarrow F| = F$. Отримане значення суперечить умові 1), а отже, не існує такої інтерпретації, на якій формула приймає хибне значення, тобто вона є тавтологією.

2. Перевіримо, чи є така формула тавтологією:

$$(A \rightarrow (B \rightarrow C)) \rightarrow (A \vee B \rightarrow C).$$

Припустимо, що $|(A \rightarrow (B \rightarrow C)) \rightarrow (A \vee B \rightarrow C)| = F$. Тоді

$$1) \quad |A \rightarrow (B \rightarrow C)| = T,$$

$$2) \quad |A \vee B| = T,$$

$$3) \quad |C| = F.$$

Із 2) випливає: а) $|A| = T$, $|B| = T$; б) $|A| = T$, $|B| = F$; в) $|A| = F$, $|B| = T$. Підставляючи значення $|A| = T$, $|B| = T$ в 1), отримаємо протиріччя: $|T \rightarrow (T \rightarrow F)| = F$. Але це ще не доводить, що формула є тавтологією. Розглянемо інші значення: б) $|A| = T$, $|B| = F$. Підставляючи ці значення в 1), маємо: $|T \rightarrow (F \rightarrow F)| = T$. Таким чином, при інтерпретації $|A| = T$, $|B| = F$, $|C| = F$ формула приймає хибне значення, а отже, вона не є тавтологією.

Логічний наслідок

Визначення 2.4. Якщо A і B – формули, то кажуть, що B логічно випливає з A , чи A логічно призводить B , якщо на всіх інтерпретаціях, де A приймає істинне значення, B також приймає істинне значення. Це позначають як $A \models B$ чи $A \Rightarrow B$.

Теорема 2.1. Логічний наслідок $A \models B$ виконан тоді і тільки тоді, коли формула $A \rightarrow B$ – тавтологія.

Визначення 2.5. Формула B логічно випливає з формул $A_1, A_2, \dots, A_n$, якщо на всіх тих інтерпретаціях, на яких $A_1, \dots, A_n$ приймають істинні значення одночасно, формула B також приймає істинне значення. Це позначається так: $A_1, \dots, A_n \models B$.

Теорема 2.2. $A_1, A_2, \dots, A_n \models B$ тоді і тільки тоді, коли $\models A_1 \& A_2 \& \dots \& A_n \rightarrow B$.

Визначення 2.5. $A \models B$ і $B \models A$, то формула A логічно еквівалентна формулі B . Це позначається як $A \Leftrightarrow B$, чи $A \equiv B$.

Якщо формула A логічно еквівалентна формулі B , то $A \equiv B$ – тавтологія.

Приклад. Перевіримо логічний наслідок: $A \rightarrow B, A \rightarrow \neg B \models \neg A$ (правило доведення до абсурду) і тавтологію: $\models (A \rightarrow B) \& (A \rightarrow \neg B) \rightarrow \neg A$. Позначимо тавтологію через E . Побудуємо таблицю істинності (табл. 2.4).

Таблиця 2.4

A	B	$A \rightarrow B$	$A \rightarrow \neg B$	$(A \rightarrow B) \& (A \rightarrow \neg B)$	$\neg A$	E
F	F	T	T	T	T	T
F	T	T	T	T	T	T
T	F	F	T	F	F	T
T	T	T	F	F	F	T

Як бачимо, на тих наборах, на яких посилки $A \rightarrow B, A \rightarrow \neg B$ приймають істинне значення одночасно, формула $\neg A$ також приймає істинне значення. Отже, логічний наслідок виконаний. З іншого боку, формула $(A \rightarrow B) \& (A \rightarrow \neg B) \rightarrow \neg A$ є тавтологією, що також є доказом дійсності логічного наслідку.

Логічні наслідки встановлюють *правила виведення* істинних висновків з істинних посилок. Кажуть, що логічне слідування (правило виведення)

зберігає істинність. Правила виведення, що зберігають істинність, називаються *достовірними (дедуктивним) міркуваннями*.

Метатеореми про тавтології

Теорема 2.3 (правило *modus ponens*). Якщо A – тавтологія і $A \rightarrow B$ – тавтологія, то B – тавтологія, тобто якщо $\models A$ і $\models A \rightarrow B$, то $\models B$.

Правило *modus ponens* (скорочено МР) встановлює логічне слідування $A, A \rightarrow B \models B$ і називається ще правилом *відділення*.

Правило МР висловлює елементарний акт дедукції. Імплікації $A \rightarrow B$, яка за визначенням має сенс «якщо A , то B », можна інтерпретувати як правило, в якому A є «причиною», а B – «наслідком». Тоді правило МР говорить про те, що наслідок B настає при виконанні умови A , тобто при істинності посилки.

Теорема 2.4 (правило підстановки). Якщо A – тавтологія, яка містить пропозиційні змінні $a_1, a_2, \dots, a_n$, то формула B , отримана з A заміною кожного входження a_i на деяку формулу A_i , також буде тавтологією.

Приклад. Формула $A \rightarrow (B \rightarrow A)$ – тавтологія. Замінімо A на $A \vee B$. Отримаємо нову тавтологію: $\models A \vee B \rightarrow (B \rightarrow A \vee B)$. Таким чином, кожну тавтологію можна розглядати як схему, з якої за допомогою підстановки можна отримати безліч тавтологій.

Теорема 2.5 (правило еквівалентної заміни). Якщо B можна отримати з A підстановкою формули B_1 замість одного або декількох входжень підформули A_1 в A , то $((A_1 \equiv B_1) \rightarrow (A \equiv B))$ є тавтологія, і, отже, якщо A_1 і B_1 логічно еквівалентні, то A і B також логічно еквівалентні.

Приклад. У тавтології $A \rightarrow (B \rightarrow A)$ замінімо підформулу $B \rightarrow A$ на еквівалентну їй формулу $\neg B \vee A$, отримаємо нову тавтологію $A \rightarrow \neg B \vee A$. Тавтологією також буде формула: $A \rightarrow \neg (B \& \neg A)$, так як $B \rightarrow A$

еквівалентна $\neg (B \& \neg A)$. Розглянуті метатеореми дають можливість отримувати нові тавтології і нові правила виведення з уже наявних.

Будь-яка система посилок має цінність тільки тоді, коли існує хоча б одна інтерпретація (тобто істиннісний розподіл літер, що входять в кожне висловлювання системи), на якій всі посилки істинні одночасно. Якщо такої інтерпретації не існує, то система посилок суперечлива, і з неї разом з деякою формулою A виводиться і її заперечення $\neg A$.

Приклад. Розглянемо систему висловлювань: $\{A \rightarrow B, B \rightarrow C \vee \neg A, \neg D \rightarrow A \& \neg C, D \rightarrow A\}$. Для перевірки системи на несуперечливість, потрібно прирівняти кожен гіпотезу до T . Якщо взяти значення для кожної пропозиційної літери, що входять до системи як «істина», то всі гіпотези будуть T - система несуперечлива.

Через алгебраїчні перетворення формули, що відповідає системі, можна отримати:

$$\begin{aligned} & (A \rightarrow B) \& (B \rightarrow C \vee \neg A) \& (\neg D \rightarrow A \& \neg C) \& (D \rightarrow A) = \\ & = (\neg A \vee B) \& (\neg B \vee C \vee \neg A) \& (D \vee (A \& \neg C)) \& (\neg D \vee A) = \\ & = (\neg A \vee (B \& (\neg B \vee C))) \& (D \vee A) \& (D \vee \neg C) \& (\neg D \vee A) = \\ & = (\neg A \vee (B \& C)) \& A \& (D \vee \neg C) = (B \& C) \& A \& (D \vee \neg C) = A \& B \& C \& \\ & (D \vee \neg C) = \\ & = A \& B \& C \& D. \end{aligned}$$

Для несуперечливості системи посилок отримана формула має доповнювати T . Це можливо тільки при входженні кожної пропозиційної літери із значенням T .

Завдання 2.1.1. Визначити, чи є наступні формули тавтологіями.

Приклад 1. Чи є формула $((A \rightarrow B) \rightarrow B) \rightarrow B$ тавтологією?

1 спосіб – побудова таблиці істинності.

A	B	$A \rightarrow B$	$(A \rightarrow B) \rightarrow B$	$((A \rightarrow B) \rightarrow B) \rightarrow B$
F	F	T	F	T
F	T	T	T	T
T	F	F	T	F
T	T	T	T	T

Формула не є тавтологією, тому що існує інтерпретація $|A| = T, |B| = F$, на якій вона набуває хибне значення.

2 спосіб. Дослідження формули методом редукції.

Припустимо, що існує набір, на якому формула набуває хибне значення:

$$|((A \rightarrow B) \rightarrow B) \rightarrow B| = F.$$

Тоді $|((A \rightarrow B) \rightarrow B)| = T, |B| = F$. Підставимо знайдене значення $|B| = F$ в першу рівність: $|((A \rightarrow F) \rightarrow F)| = T$. Розв'яжемо це рівняння відносно A : якщо $A = T$, то $|((T \rightarrow F) \rightarrow F)| = T$. Отже, при $|A| = T, |B| = F$ формула набуває значення, яке дорівнює F . Таким чином, вона не є тавтологією.

3 спосіб. Алгебраїчне перетворення.

$$\begin{aligned} ((A \rightarrow B) \rightarrow B) \rightarrow B &= \neg(\neg(\neg A \vee B) \vee B) \vee B = ((\neg A \vee B) \& \neg B) \vee B = \\ &= (\neg A \& \neg B) \vee B = \neg A \vee B. \end{aligned}$$

Як це корелюється з висновком, отриманим в попередньому пункті:

$|A| = T, |B| = F$? Так, звісно, формула не є тавтологією. Спрощення формули привело нас до розуміння, що формула отримує значення F тільки на одному наборі ($|A| = T, |B| = F$), тобто вона є конституентною нуля (макстермом), спрощення – її ДКНФ.

Приклад 2. Чи є формула $(A \rightarrow B) \& (B \rightarrow C) \& A \rightarrow C$ тавтологією?

Дослідження формули методом редукції.

Припустимо, що існує така інтерпретація, на якій формула набуває хибне значення.

Тоді $|A \rightarrow B| = T, |B \rightarrow C| = T, |A| = T, |C| = F$.

Тоді:

$$|B \rightarrow C| = |B \rightarrow F| = T \Rightarrow |B| = F;$$

$$|A \rightarrow B| = |A \rightarrow F| = T \Rightarrow |A| = F.$$

Звідси $|A| = F$, що суперечить третій рівності $|A| = T$. Отримане протиріччя демонструє, що формула не може набувати хибних значень. Отже, формула $(A \rightarrow B) \& (B \rightarrow C) \& A \rightarrow C$ є тавтологією.

Можемо побудувати і таблицю істинності і довести алгебраїчно.

$$\begin{aligned} (A \rightarrow B) \& (B \rightarrow C) \& A \rightarrow C &= \neg ((A \rightarrow B) \& (B \rightarrow C) \& A) \vee C = \\ &= \neg (\neg A \vee B) \vee \neg (\neg B \vee C) \vee \neg A \vee C = \textcolor{green}{A} \& \neg \textcolor{blue}{B} \vee \textcolor{blue}{B} \& \neg \textcolor{green}{C} \vee \neg \textcolor{green}{A} \vee \textcolor{blue}{C} = \\ &= \neg \textcolor{green}{B} \vee \textcolor{blue}{B} \vee \neg \textcolor{green}{A} \vee \textcolor{blue}{C} = \textbf{T} \vee \neg A \vee C = \textbf{T}. \end{aligned}$$

$$E = (A \rightarrow B) \& (B \rightarrow C) \& A \rightarrow C$$

A	B	C	$A \rightarrow B$	$B \rightarrow C$	$(A \rightarrow B) \& (B \rightarrow C) \& A$	E
F	F	F	T	T	F	T
F	F	T	T	T	F	T
F	T	F	T	F	F	T
F	T	T	T	T	F	T
T	F	F	F	T	F	T
T	F	T	F	T	F	T
T	T	F	T	F	F	T
T	T	T	T	T	T	T

Будь-який спосіб доводить, що формула є тавтологією.

- 1) $(A \rightarrow B) \rightarrow ((B \rightarrow C) \rightarrow (A \rightarrow C));$
- 2) $(A \rightarrow B) \rightarrow ((C \rightarrow B) \rightarrow (A \vee C \rightarrow B));$
- 3) $(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow C) \rightarrow (A \rightarrow B));$
- 4) $((A \rightarrow B) \rightarrow (A \rightarrow C)) \rightarrow ((A \rightarrow (B \rightarrow C));$
- 5) $(B \rightarrow A \vee C) \rightarrow ((B \rightarrow C) \rightarrow ((D \rightarrow C) \rightarrow (B \vee D \rightarrow C)));$
- 6) $(B \rightarrow C) \rightarrow (A \vee C \rightarrow (B \rightarrow C));$
- 7) $(A \rightarrow (B \rightarrow C)) \rightarrow (C \rightarrow A \vee B).$

Завдання 2.1.2. Перевірити, чи є формули тавтологіями, протиріччями або нейтральними. Які формули є здійсненими?

Для перевірки, до якого класу належить формула, можна звернутися до побудови таблиць істинності або до алгебраїчних перетворень. Спробуємо визначити клас формули через міркування – через метод редукції (*в логіці – пошук такого судження, з якого випливає судження, визнане нами за істинне; зведення складного до простішого в інтересах наукового аналізу*). В загальному випадку рекомендується намагатися визначити, чи є формула тавтологією, бо такі міркування нами вже відпрацьовані. Якщо формула виявиться не тавтологією, то підбиранням значень пропозиційних літер визначитися, чи буде вона здійсненою.

- 1) $(A \& B) \vee (C \& D) \rightarrow (A \vee B) \& (C \vee D);$
- 2) $A \& (A \& \neg B \vee \neg A \& B) \& B$
- 3) $(A \rightarrow B) \& (C \rightarrow D) \rightarrow (A \vee C \rightarrow B \vee D);$
- 4) $(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C));$
- 5) $\neg((A \vee B \rightarrow C) \rightarrow \neg(\neg A \vee C \rightarrow B \& \neg C));$
- 6) $(A \vee B \rightarrow C) \rightarrow (A \rightarrow C) \vee (B \rightarrow C);$
- 7) $(A \& B \rightarrow C) \rightarrow ((A \rightarrow C) \rightarrow (B \rightarrow C));$

- 8) $(A \rightarrow B) \& (C \rightarrow D) \rightarrow (A \& C \rightarrow B \& D);$
 9) $A \& (\neg A \vee \neg B) \& B;$
 10) $(A \& \neg B \vee \neg A \& B) \rightarrow B.$

Завдання 2.1.3. Дослідити систему посилок (гіпотез) на несуперечливість.

Приклад 1.

$$\{F \rightarrow E \vee L, E \vee \neg F \& U, U \rightarrow E \vee L, \neg F \rightarrow \neg L, E \rightarrow \neg E\}$$

Система посилок несуперечлива, якщо можна знайти таку інтерпретацію літер, при якій кожна посилка є істинною, тобто дорівнює T .

1. $|F \rightarrow E \vee L| = T;$
2. $|E \vee \neg F \& U| = T;$
3. $|U \rightarrow E \vee L| = T;$
4. $|\neg F \rightarrow \neg L| = T;$
5. $|E \rightarrow \neg E| = T.$

З 5-ої посилки слідує, що $|E| = F$. Тоді, з 2-ої: $|F| = F$ і $|U| = T$.

Для того, щоб 3-я посилка дорівнювала T , при вже отриманих значеннях пропозиційних літер, має бути $|L| = T$. Але при таких значеннях F та L 4-а посилка не буде дорівнювати T . Отже, дана система посилок є суперечливою.

Приклад 2.

$$\{A \rightarrow B, C \vee A, C \rightarrow E, \neg E, \neg E \rightarrow \neg C\}$$

Розглянемо систему:

1. $|A \rightarrow B| = T;$
2. $|C \vee A| = T;$
3. $|C \rightarrow E| = T;$

4. $|\neg E|=T$;

5. співпадає з ГЗ

З 4-ої посилки отримуємо $|E|=F$. Тоді при $|C|=F$, $|A|=T$, $|B|=T$ кожна посилка є істинною.

Система – несуперечлива.

1) $\{ A \vee B \rightarrow C \ \& \ D, \ D \vee E \rightarrow G, \ A \vee \neg G \}$;

2) $\{ (A \rightarrow B) \& (C \rightarrow D), (B \rightarrow D) \& (\neg C \rightarrow A), (E \rightarrow G) \& (G \rightarrow \neg D), \neg E \rightarrow E \}$;

3) $\{ (A \rightarrow B \& C) \& (D \rightarrow B \& E), (G \rightarrow \neg A) \& H \rightarrow I, (H \rightarrow I) \rightarrow G \& D, \neg(\neg C \rightarrow E) \}$;

4) $\{ (A \rightarrow B \& C) \& (D \rightarrow B \ \& \ E), (G \rightarrow \neg A), (H \rightarrow I) \rightarrow G \ \& \ D, \neg(\neg C \rightarrow E) \}$;

5) $\{ A \rightarrow C \ \& \ D, \ D \vee E \rightarrow C, \ A \vee C \rightarrow E \}$;

6) $\{ A \rightarrow B, \ B \rightarrow D, \ C \rightarrow A, \ (E \rightarrow C) \ \& \ (G \rightarrow \neg D) \}$;

2.2. Формалізація і рішення логічних задач

Теоретичні відомості

Мова алгебри висловлювань використовується для формалізації речень природної мови і доказів логічних слідувань.

Двозначна класична логіка корисна в повсякденному житті, навіть, якщо доводиться визнати, що в припущеннях, наприклад, про двозначність, є елемент свавілля. Для вирішення будь-якої інтелектуальної проблеми не слід нехтувати логікою. Наприклад, якщо розглядається якась ймовірнісна задача, то перш за все можна знайти наслідки, що впливають з набору гіпотез, без розгляду їх ймовірнісного (змістовного) сенсу. Посилання на модальну або інтуїціоністську логіку дають зрозуміти, що в деяких ситуаціях потрібна інша, не класична логіка

Зв'язкою «еквівалентність» замінюються речення природної мови наступної природи: *якщо - і - тільки - якщо, тоді і тільки тоді, рівносильно*. «Імплікація» замінює: *якщо - то, в разі А має місце В, А – тільки, якщо В, В - якщо А, А тягне В, для В досить А, для А необхідно В*. Заміна на «кон'юнкцію» - *А і В, не тільки А - а й В, В - хоча і А, В - незважаючи на А, як А - так і В, А разом з В, А - в той же час як В*.

Переклад **А & В** найбільш простий, якщо в ньому не бере участь причина часу (про що йшлося вище). Багато смислових відтінків (незважаючи, хоча, але і) зникають при формалізації.

«Диз'юнкція» - А чи В, А - якщо не В, А чи В або обидва. Труднощі перекладу «диз'юнкції» полягають в двозначності певних термінів. Якщо в меню ресторану вказано «чай або кава - безкоштовно», то ми не здивуємося, якщо, зажадавши, і те й інше, отримаємо збільшений рахунок. У той же час, коли оголошення говорить, що «пожертвування приймаються в церкві або школі», то ми не думаємо, що принесене в церкву відкинуть через те, що ми вже жертвували в школі.

Якщо хочуть висловити виняткове «або», то повинні додати до «або», наприклад, фразу «але не те й інше разом» - формалізується $\neg (A \& B) \& (A \vee B)$.

Виразам «А, якщо не В», «А, крім випадку, коли В» надається сенс такого, що наявність В звільняє від відповідальності стверджувати істинність А. Тобто, якщо не $\neg B$, то А, а якщо В, то про А нічого не говориться ($\neg B \rightarrow A = \neg A \rightarrow B$).

У звичайній мові не вживаються дужки для вказівки сполучуваності частин складного речення. «Якщо Джонс присутній або якщо Вільямс висловиться за нашу пропозицію і Старк не стане заперечувати, то наша пропозиція буде прийнята» («стратити не можна помилувати» - на листі

необхідно розставляти потрібні акценти знаками пунктуації). Формалізація цього речення наступна:

$D \vee W \ \& \ \neg S \rightarrow P$ – формулі відсутні дужки, що не дозволяє однозначно сприймати інформацію відповідно з вихідним текстом (слід зазначити, що і правила побудови формул теж порушуються).

Приклад 2.1. Розглянемо логічне слідування: *Якщо не буде дощу, то ми поїдемо на пікнік. Якщо ми поїдемо на пікнік, то ми добре проведемо час. Дощу немає. Отже, ми добре проведемо час.*

Позначимо пропозиціональними буквами прості висловлювання: P - «не буде дощу»; S - «поїдемо на пікнік»; R - «добре проведемо час». Необхідно довести логічне слідування: $P \rightarrow S, S \rightarrow R, P \mid = R$.

Доведення.

1 спосіб. Довести тавтологію: $\mid = (P \rightarrow S) \ \& \ (S \rightarrow R) \ \& \ P \rightarrow R$. Для цього досить побудувати таблицю істинності.

2 спосіб. Доведення від протилежного (метод редукції).

Припустимо, що існує така інтерпретація, на якій всі посилки приймають істинне значення, а наслідок - хибне, тобто $\mid P \rightarrow S \mid = T, \mid S \rightarrow R \mid = T, \mid P \mid = T$. Припускаємо $\mid R \mid = F$. Тоді з $\mid S \rightarrow R \mid = \mid S \rightarrow F \mid = T$ випливає, що $\mid S \mid = F$; з $\mid P \rightarrow S \mid = \mid P \rightarrow F \mid = T$ випливає, що $\mid P \mid = F$, що суперечить третій посилці $\mid P \mid = T$. Це протиріччя доводить логічне слідування.

3 спосіб. Побудова формального логічного висновку.

Побудуємо логічний висновок, використовуючи відомі правила виведення. Логічний висновок записується зазвичай як пронумерована послідовність формул, праворуч від кожної формули записується коментар, який вказує, на якій підставі формула включена в цю послідовність. Посилки виведення зазвичай позначаються літерою Γ (гіпотеза).

1. $P \rightarrow S$ Г1
2. $S \rightarrow R$ Г2
3. P Г3
4. S правило МР (3,1)
5. R правило МР (4,2)

Остання формула в цьому висновку є логічним наслідком посилок Г1, Г2, Г3.

В силу того, що правило МР зберігає істинність, кожна формула, яка бере участь у виведенні, істинна при істинності посилок Г1, Г2, Г3. Тому, якщо з'єднати символом $\rightarrow$ формули, так, щоб посилка імплікації передувала висновку в цьому виведенні, то отримана формула також буде істинною, і, отже, вона є логічним наслідком посилок.

Тому з даного висновку ми можемо отримати логічне виведення:

$$P \rightarrow S, S \rightarrow R \mid = P \rightarrow R.$$

Це логічне слідування відповідає правилу виведення, яке називають правилом силогізму:

$$A \rightarrow B, B \rightarrow C \mid = A \rightarrow C.$$

Приклад 2.2. Через снігопад (A) машина потрапить в затор (B): $A \rightarrow B$. У затор машина не потрапила ($\neg B$). Отже, снігопаду не було ($\neg A$). Доведемо $A \rightarrow B, \neg B \mid = \neg A$.

Припустимо, що $\mid A \rightarrow B \mid = T, \mid \neg B \mid = T$ і $\mid \neg A \mid = F$, тоді $\mid A \rightarrow B \mid = \mid T \rightarrow F \mid = F$. Отримане протиріччя доводить логічне слідування.

Це логічне слідування відповідає правилу виведення **modus tollens (MT)**:

$$A \rightarrow B, \neg B \mid = \neg A.$$

Приклад 2.3. Решітка є модулярною (B) в тому випадку, якщо вона дистрибутивною (A): $A \rightarrow B$. Але ми знаємо, що якщо решітка немодулярна ($\neg B$), то вона і недистрибутивна ($\neg A$): $\neg B \rightarrow \neg A$.

Припустимо, що $|A \rightarrow B| = T$, $|\neg B \rightarrow \neg A| = F$, тоді $|\neg B| = T$, $|\neg A| = F$ і $|A \rightarrow B| = |T \rightarrow F| = F$. Отримане протиріччя доводить логічне слідування.

Це логічне слідування відповідає правилу виведення, яке називають правилом контрапозиції:

$$A \rightarrow B \mid = \neg B \rightarrow \neg A$$

Приклад 2.4. Якщо долар росте (B), то ціни не зменшуються ($\neg A$). Якщо долар падає ($\neg B$), то ціни теж не зменшуються ($\neg A$): $B \rightarrow \neg A$ і $\neg B \rightarrow \neg A$. Тобто, ціни не зменшуватися в будь-якому випадку. Згідно із законом контрапозиції ми можемо отримати з умов задачі: $A \rightarrow \neg B$ і $A \rightarrow B$.

$$A \rightarrow \neg B, A \rightarrow B \mid = \neg A.$$

Цей закон відображає правило приведення до абсурду. Якщо з посилки слідує одночасно істинно і хибно, то сама посилка – хибна.

Приклад 2.5. Я не зможу заснути (A) тільки в тому випадку, якщо вип'ю каву (B). Як формалізувати це речення? Здається, що «якщо» знаходиться при висловлюванні B , то має бути $B \rightarrow A$. Але сенс фрази полягає в наступному: «Якщо я не заснув, то з цього випливає тільки одне - я випив каву». Тобто, слід проводити формалізацію таким чином: $A \rightarrow B$. Вживання «тільки», «крім» тощо змінює, напрямок імплікації.

Випишемо розглянуті логічні закони:

$A \rightarrow B, A \mid = B$	<i>modus ponens</i>
$A \rightarrow B, B \rightarrow C \mid = A \rightarrow C$	<i>правило силогізму</i>
$A \rightarrow B \mid = \neg B \rightarrow \neg A$	<i>правило контрапозиції</i>

$A \rightarrow B, \neg B \models \neg A$	<i>modus tollens</i>
$A \rightarrow B, A \rightarrow \neg B \models \neg A$	<i>правило приведення до абсурду</i>

Приклад 2.6. Отримати можливі логічні наслідки з даної множини посилок: *Якщо в городі - бузина (A), то в Києві - дядько (B). Коні їдять овес (C) або в городі - бузина. Якщо коні їдять овес, то Дніпро впадає в Чорне море (E). Дніпро не впадає в Чорне море.*

Побудуємо дедуктивний формальний висновок:

1. $A \rightarrow B$ Г1
2. $C \vee A$ Г2
3. $C \rightarrow E$ Г3
4. $\neg E$ Г4
5. $\neg E \rightarrow \neg C$ правило контрапозиції (3)
(Якщо Дніпро не впадає в Чорне море, то коні не їдять овес)
6. $\neg C \rightarrow A$ еквівалентна заміна (2)
7. $\neg E \rightarrow A$ правило силогізму (5, 6)
(Якщо Дніпро не впадає в Чорне море, то в городі - бузина)
8. $\neg E \rightarrow B$ правило силогізму (1, 7)
(Якщо Дніпро не впадає в Чорне море, то в Києві дядько)
9. B МР (4,8)
(В Києві - дядько)

Таким чином, з даного набору посилок ми крок за кроком отримали всі можливі висновки.

В одному з оповідань про Шерлока Холмса склалася така ситуація: «Нам відомо, що злочинець не міг потрапити в кімнату ні через двері (A), ні через камін (B). Ми знаємо також, що він не міг сховатися в кімнаті (C),

оскільки в ній ховатися ніде. Як же тоді він проник сюди? - Через дах (D)! - Без сумніву. Він міг проникнути в цю кімнату тільки через дах.»

Це міркування можна формалізувати так: $A \vee B \vee C \vee D, \neg A, \neg B, \neg C \models D$, що рівносильно: $\neg A \rightarrow (\neg B \rightarrow (\neg C \rightarrow D)), \neg A, \neg B, \neg C \models D$. Триразове застосування правила МР доводить цей логічний висновок.

Задачі (за теоремою 2.2.) відповідає формула:

$(A \vee B \vee C \vee D) \& \neg A \& \neg B \& \neg C \rightarrow D$, яка має бути тавтологією у випадку, коли висновок з цієї множини посилок є дійсним.

Не слід забувати, що логічний наслідок виконується тільки тоді, коли з істинних висловлювань слідує істинний висновок. Тому повинна існувати хоча б одна інтерпретація, на якій всі посилки істинні одночасно. Якщо такої інтерпретації не існує, то система посилок суперечлива.

Приклад 2.7.

Які висновки можна зробити з наступних висловлювань:

" Повернувшись додому, Мегре подзвонив на набережну Орфевр:

- Є новини?

- Так, шеф. Надійшли повідомлення від інспекторів. Торранс встановив, що, якщо Франсуа був п'яний (F), то або Етьєн вбивця (E), або Франсуа бреше (L). Жуссе вважає, що або Етьєн вбивця, або Франсуа не був п'яний і вбивство сталося після опівночі (U). Інспектор Люка просив передати Вам, що якщо вбивство сталося після опівночі, то або Етьєн вбивця, або Франсуа бреше. Потім дзвонила ...

- Усе. Дякую. Цього достатньо. - Комісар поклав трубку. Він знав, що тверезий Франсуа ніколи не бреше. Тепер він знав все. "

Для побудови формального виводу необхідне знання деяких прийомів і теорем. Приведемо формальний вивід для вищевикладеної задачі:

1. $F \rightarrow E \vee L$ гіпотеза 1

- | | | |
|-----|-------------------------------|-------------------------------------|
| 2. | $E \vee \neg F \& U$ | гіпотеза 2 |
| 3. | $U \rightarrow E \vee L$ | гіпотеза 3 |
| 4. | $\neg F \rightarrow \neg L$ | гіпотеза 4 |
| 5. | $\neg E \rightarrow \neg F$ | гіпотеза 2 (видалення кон'юнкції) |
| 6. | $\neg E \rightarrow U$ | гіпотеза 2 видалення кон'юнкції) |
| 7. | $\neg E \rightarrow E \vee L$ | силогізм 3, 8 |
| 8. | $\neg E \rightarrow L$ | з 7 еквів.перетв. і ідемпотентність |
| 9. | $L \rightarrow F$ | контрапозиція гіпотези 4 |
| 10. | $\neg E \rightarrow F$ | силогізм 8, 11 |
| 11. | E | довед.до абсурда 5, 10 |

Ми бачимо, що гіпотеза 1 є зайвою, вона не бере участь в побудові формального виводу.

Якщо ми використаємо гіпотезу 1, наприклад, зробимо силогізм з контрапозицією гіпотези 4 ($L \rightarrow F$ та $F \rightarrow E \vee L$), то отримаємо $L \rightarrow E \vee L = \neg L \vee E \vee L = T \vee E = T$, що є даремним кроком.

Завдання 2.2.1. Перевірити логічне слідування.

Приклад. Джон або втомився, або він хворий. Якщо він втомився, то він роздратований. Він не роздратований. Отже, Джон – хворий.

Формалізуємо в логіці висловлювань гіпотези.

Нехай A – «Джон втомився»,

B – «Джон хворий»,

C – «Джон роздратований».

Побудуємо дедуктивний вивід, використовуючи наступні правила виводу:

$A, A \rightarrow B \models B$ – *modus ponens* (MP);

$A \rightarrow B, B \rightarrow C \models A \rightarrow C$ – правило силогізму (ПС);

$A \rightarrow B \models \neg B \rightarrow \neg A$ – правило контрапозиції (ПК);

$A \rightarrow B, A \rightarrow \neg B \models \neg A$ – правило доведення до абсурду (ПА);

$A \rightarrow B, \neg B \models \neg A$ – *modus tollens* (MT).

Побудуємо формальний вивід:

- | | |
|---------------------------|-------------------------|
| 1. $A \vee B$ | Γ_1 |
| 2. $A \rightarrow C$ | Γ_2 |
| 3. $\neg C$ | Γ_3 |
| 4. $\neg A$ | MT (2,3) |
| 5. $\neg A \rightarrow B$ | еквівалентна заміна (1) |
| 6. B | MP (4,5) |

Приклад 2. *Малі діти нерозумні. Той, хто може приборкувати крокодилів, заслуговує на повагу. Нерозумні люди не заслуговують на повагу. Отже, малі діти не приборкують крокодилів.*

Нехай A – «малі діти», B – «розумні», C – «заслуговують на повагу», D – «можуть приборкувати крокодилів». Побудуємо дедуктивний вивід, використовуючи наступні правила виводу:

Формальний вивід:

- | | |
|--------------------------------|---------------------------|
| 1. $A \rightarrow \neg B$ | Γ_1 |
| 2. $D \rightarrow C$ | Γ_2 |
| 3. $\neg B \rightarrow \neg C$ | Γ_3 |
| 4. $A \rightarrow \neg C$ | правило силогізму (1,3) |
| 5. $\neg C \rightarrow \neg D$ | правило контрапозиції (2) |
| 6. $A \rightarrow \neg D$ | правило силогізму (4,5) |

1) Якщо студент пропускає заняття (Z), він погано засвоює предмет (P). Якщо студент не навчається вдома (D), він також погано засвоює предмет. Якщо студент погано засвоїв предмет, він не складе залік (A). Студент не допускається до іспиту (I), якщо він не склав залік. Недопущений до іспиту студент має менше часу на відпочинок (R). Отже, якщо студент пропускає заняття або не навчається вдома, у нього менше часу на відпочинок.

2) Якщо Джонс не зустрічав цієї ночі Сміта (V), то або Сміт був убивцею (S), або Джонс бреше (D). Якщо Сміт був убивцею, то Джонс бачив Сміта цієї ночі ($\neg V$), і вбивство мало місце після опівночі (P). Якщо вбивство сталося після опівночі, то або Сміт був убивцею, або Джонс бреше. Джонс не зустрічав цієї ночі Сміта (V). Отже, якщо Джон говорить правду ($\neg D$), то Сміт - убивця.

3) Якщо Сміт переможе на виборах (V), він буде задоволений (Z), а якщо він буде задоволений, то він поганий борець в передвиборній кампанії (B). Але якщо він завалить вибори ($\neg V$), то втратить довіру партії (D). Якщо він поганий борець у передвиборній кампанії, йому слід вийти з партії (E). Він поганий борець, якщо втратить довіру партії. Сміт або переможе в передвиборній кампанії, або завалить її. Отже, йому треба вийти з партії в будь якому випадку.

4) Якщо Петров є членом нашої команди (P), він обов'язково хоробрий (H) і добре володіє технікою удару (T). Але він не входить у склад нашої команди. Отже, він не хоробрий.

5) Зарплата збільшується **тільки** (Z), якщо буде інфляція (I). Якщо буде інфляція, то збільшиться вартість життя (V). Якщо вартість життя зростає, то люди нещасливі ($\neg H$). Отже, при збільшенні зарплати люди будуть нещасними?

6) Якщо конгрес відмовляється прийняти нові закони ($\neg K$), то страйк не буде завершено ($\neg Z$), якщо тільки він не триває більше року (Y). Страйк

також буде завершено (Z), якщо президент фірми піде у відставку (P). Конгрес відмовляється діяти, страйк завершено, і президент фірми не йде у відставку. Отже, страйк тривав більше року.

Вказівка: 1 та 2 речення можна разом трактувати так, що, якщо страйк завершено, то або конгрес прийняв певні рішення або президент пішов у відставку або страйк тривав більше року.

7) Або Сергій і Борис – одного віку (A), або Сергій старше за Бориса (B). Якщо Сергій і Борис – одного віку, то Микола і Борис – різного (C). Якщо Сергій старше Бориса, то або Борис старше Віталія (D), або Микола і Борис – різного віку. Микола і Борис – однолітки. Отже, Борис старше Віталія.

8) Якщо Петро не бачив Миколу на вулиці ($\neg B$), то або Микола ходив в кіно (C), або Петро сказав правду (P). Якщо Микола не ходив в кіно, то Петро не бачив Миколу на вулиці, і Микола сказав правду (M). Якщо Микола сказав правду, то або він ходив в кіно, або Петро збрехав. Перевірити, чи ходив Микола в кіно.

2.3. Метод резолюцій

Теоретичні відомості

В основі методу резолюцій лежить процедура пошуку спростування, тобто замість доведення тавтології, що відповідає логічній задачі, доводиться те, що заперечення формули є протиріччям. Метод спростування для доказу логічного слідування полягає в наступному. Нехай виконується логічне слідування: $F1, F2 \models G$. Тоді $\models F1 \& F2 \rightarrow G$ є тавтологією, і, отже, $\models \neg(F1 \& F2 \rightarrow G) \equiv F$. Оскільки за визначенням посилки $F1, F2$ істинні, формула $F1 \& F2 \& \neg G$ може обернутися в хибну тільки, якщо $\models \neg G \equiv F$, тобто якщо $\models G \equiv T$. Тоді логічне слідування

виконано. В принципі процедура спростування формалізує метод редукції (так званий неформальний вивід).

Введемо декілька визначень.

Визначення 2.6. Диз'юнкція літер називається диз'юнктом, або clause (клаузою, клозом). Однолітерний диз'юнкт називається *одиничним диз'юнктом (фактом)*. Коли диз'юнкт не містить ніяких літер, його називають *пустим* диз'юнктом. Так як пустий диз'юнкт не містить літер, які могли б бути істинними за будь-яких інтерпретацій, то пустий диз'юнкт завжди хибний. Пустий диз'юнкт позначається символом $\square$.

Правило резолюцій Робінсона. Якщо для будь-яких двох диз'юнктів C_1 і C_2 існує літера $L_1 \in C_1$ і контрарна їй літера $L_2 \in C_2$ ($L_2 = \neg L_1$), то викресливши L_1 з C_1 і L_2 з C_2 і побудувавши диз'юнкт з решти літер, отримаємо так звану *резольвенту* C_1 і C_2 : $C'_1 \vee C'_2$, де $C'_1 = C_1 \setminus L_1$, $C'_2 = C_2 \setminus L_2$.

Теорема 2.6. Резольвента C є логічним слідуванням C_1 і C_2 , що містять контрарні літери L і $\neg L$: $L \vee C'_1, \neg L \vee C'_2 \models C'_1 \vee C'_2$.

Доведення. Припустимо, що $|L \vee C'_1| = T$, $|\neg L \vee C'_2| = T$, $|C'_1 \vee C'_2| = F$. Тоді $|C'_1| = F$, $|C'_2| = F$. Якщо $|L \vee C'_1| = T$, то $|L| = T$, але $|\neg L \vee C'_2| = T$, отже, $|L| = F$. Отримане протиріччя доводить теорему. $\diamond$

Правило резолюцій є узагальненням багатьох відомих нам правил виводу. Наприклад, правило силогізму: $A \rightarrow B, B \rightarrow C \models A \rightarrow C$ може бути переписано у вигляді: $\neg A \vee B, \neg B \vee C \models \neg A \vee C$, що відповідає правилу резолюцій. Правило МР: $A, A \rightarrow B \models B$ може бути переписано у вигляді: $A, \neg A \vee B \models B$, що також відповідає правилу резолюцій. Нарешті, закон протиріччя $A \& \neg A \equiv F$ рівнозначний правилу: $A, \neg A \models \square$, згідно якому, резольвента двох контрарних однолітерних диз'юнктів є пустий диз'юнкт.

Метод резолюцій стосовно вирішення задачі полягає в тому, щоб перевірити, чи містить множина диз'юнктів S , що складається з

перетворених в КНФ посилок і заперечення висновка, порожній диз'юнкт $\square$. Якщо S містить $\square$, то множина S нездійсненна, якщо ні, то треба перевірити, чи може він бути отриманий з даної множини диз'юнктив. Іншими словами, необхідно знайти множину основних прикладів, які спростовують вихідну множину диз'юнктив. Ця процедура полягає в побудові резолютивного виведення і ґрунтується на правилі резолюцій Робінсона.

Визначення 2.7. *Резолютивний вивід* з множини диз'юнктив S є послідовність $C_1, C_2, \dots, C_k$, така, що кожне C_i або належить S , або є резольвентою попередніх C_i . Якщо останній диз'юнкт $C_k = \square$, то множина диз'юнктив S є нездійсненною, а весь вивід називається спростуванням S . Якщо C_k не є пустим диз'юнктом і подальше застосування правила резолюцій неможливо, то множина S є здійсненою (але висновок в цьому разі не є наслідком з множини посилок, тобто, задача не розв'язується).

Приклад 2.8. Розглянемо задачу з прикладу 2.1. Необхідно перевірити логічне слідування в логіці висловлювань: $P \rightarrow S, S \rightarrow R, P \models R$. Складемо множину диз'юнктив S , для чого кожен формулу приведемо до КНФ, а від висновка R візьмемо заперечення. Отримаємо:

1. $\neg P \vee S$
2. $\neg S \vee R$
3. P
4. $\neg R$
5. $\neg S$ резольвента 4, 2
6. $\neg P$ резольвента 5, 1
7. $\square$ резольвента 3, 6

Приклад 2.9.

Повернемось до умови задачі 2.7. Припустимо, що вбивця - Етьєн. Можна формалізувати задачу і застосувати метод резолюції для з'ясування істинності нашого припущення.

1. $F \rightarrow E \vee L$ Умова задачі - список гіпотез
2. $E \vee \neg F \& U$
3. $U \rightarrow E \vee L$
4. $\neg F \rightarrow \neg L$

Приведемо гіпотези в КНФ

1. $\neg E$ заперечення виводу
2. $\neg F \vee E \vee L$ гіпотеза 1
3. $E \vee \neg F$ гіпотеза 2 (видалення кон'юнкції)
4. $E \vee U$ гіпотеза 2 (видалення кон'юнкції)
5. $\neg U \vee E \vee L$ гіпотеза 3
6. $F \vee \neg L$ гіпотеза 4
7. $\neg F$ резольвента 1,3
8. U резольвента 1,4
9. $\neg U \vee L$ резольвента 1,5
10. $\neg L$ резольвента 6,7
11. $\neg U$ резольвента 9,10
12. $\square$ резольвента 8,11

Правило резолюцій – дуже потужний і зручний засіб логічного доведення.

Гіпотеза 1 не задіяна в процедурі виведення, так само, як при формальному способі.

Приклад 2.10.

Побудуємо резолютивний вивід для задачі з прикладу 2.6. Але трохи змінимо умову, щоб розглянути приклад отримання заперечення висновку для іншої форми, ніж в попередній задачі.

Якщо в городі - бузина (A), то в Києві - дядько (B). Коні їдять овес (C) або в городі - бузина. Якщо коні їдять овес, то Дніпро впадає в Чорне море (E). ~~Дніпро не впадає в Чорне море.~~ (останню посилку викреслюємо).
Висновок для перевірки: Якщо Дніпро не впадає в Чорне море, то в Києві – дядько (формальний вивід зупинився би на п.8 – див вище розв’язок прикладу 2.6).

Формалізація:

1. $A \rightarrow B$	Г1	$(\neg A \vee B)$
2. $C \vee A$	Г2	так вже і є диз’юнкт
3. $C \rightarrow E$	Г3	$(\neg C \vee E)$
$\neg E \rightarrow B$	висновок	

Візьмемо заперечення висновку:

$$\neg(\neg E \rightarrow B) = \neg(\neg\neg E \vee B) = \neg(E \vee B) = \neg E \& \neg B.$$

Ми отримали два однолітерних негативних диз’юнкта: $\neg E$ та $\neg B$.

Запишемо множину диз’юнктів, яку будемо перевіряти на несуперечливість: $\{\neg A \vee B, C \vee A, \neg C \vee E, \neg E, \neg B\}$.

1. $\neg E$ заперечення висновку
2. $\neg B$ заперечення висновку
3. $\neg A \vee B$ Г1
4. $C \vee A$ Г
5. $\neg C \vee E$ Г3
6. $\neg C$ резольвента 1,5
7. A резольвента 4,6

8. B резолювента 3,7

9. $\square$ резолювента 2,8

Пустий диз'юнкт отримано, система, що складається з гіпотез-умов задачі і заперечення висновку, суперечлива; висновок – дійсний.

Системи з формалізованих гіпотез мають цінність, якщо вони не є суперечливими. Це означає, що існує така інтерпретація пропозиційних літер, при якій всі гіпотези-посилки можуть одночасно дорівнювати TRUE. Несуперечливість системи можна перевірити і методом резолюцій, застосовуючи правило резолюцій Робінсона для пошуку пустого диз'юнкта (якщо він є – система суперечлива).

Але системи можуть розглядатися не тільки в якості опису умов задачі. Використати метод резолюцій для визначення суперечності чи ні системи посилок можна для перевірки логічного слідування (також визначення чи є формула тавтологією). В такому випадку система містить умови (які є несуперечливі) і заперечення висновку, який перевіряється на логічне слідування з посилок. В такому разі суперечливість системи (наявність пустого диз'юнкту) означає, що заперечення висновку - хибне, тобто сам висновок є логічним наслідком з умови задачі.

Завдання 2.3.1. Визначити, чи є наступні формули тавтологіями методом резолюції.

Приклад 1. Чи є формула $((A \rightarrow B) \rightarrow B) \rightarrow B$ тавтологією?

Формула складається з антецедента $(A \rightarrow B) \rightarrow B$ та консеквента B .

Перетворимо антецедент в кон'юнктивну форму, тобто в кон'юнкцію диз'юнктів для запису множини диз'юнктів:

$$(A \rightarrow B) \rightarrow B = \neg(\neg A \vee B) \vee B = (A \& \neg B) \vee B = (A \vee B) \& (\neg B \vee B) =$$

$$= (A \vee B) \& T^{\text{true}} = A \vee B, \text{ та візьмемо заперечення висновку.}$$

Отримаємо множину $\{A \vee B, \neg B\}$. Пустий диз'юнкт отримати не можливо, тому формула не є тавтологією.

Приклад 2. Чи є формула $(A \rightarrow B) \& (B \rightarrow C) \& A \rightarrow C$ тавтологією?

Складемо множину диз'юнктив з посилок і заперечення висновку (це можливо тоді, коли зовнішня формула є імплікацією, тоді висновок – останній консеквент). Посилки $(A \rightarrow B)$, $(B \rightarrow C)$, A , та консеквент – C .

Тоді множина диз'юнктив буде наступною: $\{\neg A \vee B, \neg B \vee C, A, \neg C\}$.

Ми бачимо, що пустий диз'юнкт отримати можна, отже, система є суперечливою (за рахунок додавання заперечення висновку), тобто формула – тавтологія.

Зауваження *. $A \vee B$ та $\neg A \vee \neg B$ не дають пустий диз'юнкт. Вони не є запереченням один до одного.

Якщо діяти за правилом Робінсона, то, обравши за основу для контрарних літер будь-яку з двох A або B , наприклад, $\neg A$ та A , отримаємо в якості резольвенти диз'юнкт, що побудований з літер, що залишилися після видалення контрарних, $B \vee \neg B = \text{True}$, що робить його зовсім непотрібним, але не приводить до пустого диз'юнкта.

- 1) $(B \rightarrow C) \rightarrow (A \vee C \rightarrow (B \rightarrow C))$;
- 2) $(A \rightarrow (B \rightarrow C)) \rightarrow (A \vee B \rightarrow C)$;
- 3) $(A \rightarrow B) \rightarrow ((B \rightarrow C) \rightarrow (A \rightarrow C))$;
- 4) $(A \rightarrow B) \rightarrow ((C \rightarrow B) \rightarrow (A \vee C \rightarrow B))$;
- 5) $(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$.

Завдання 2.3.2. Дослідити систему посилок (гіпотез) на несуперечливість методом резолюції.

Приклад 1.

$\{F \rightarrow E \vee L, E \vee \neg F \& U, U \rightarrow E \vee L, \neg F \rightarrow \neg L, E \rightarrow \neg E\}$

Перетворимо кожну гіпотезу (посилки) у диз'юнкти і перевіримо на несуперечливість отриману множину диз'юнктив, використовуючи правило резолюції Робінсона по вилученню контрарних літер.

$$1. F \rightarrow E \vee L = \neg F \vee E \vee L \quad \Gamma 1$$

$$2. E \vee \neg F \& U = (E \vee \neg F) \& (E \vee U) :$$

$$2.a \quad E \vee \neg F$$

$$2.6 \quad E \vee U \quad \Gamma 2$$

$$3. U \rightarrow E \vee L = \neg U \vee E \vee L \quad \Gamma 3$$

$$4. \neg F \rightarrow \neg L = F \vee \neg L \quad \Gamma 4$$

$$5. E \rightarrow \neg E = \neg E \vee \neg E = \neg E \quad \Gamma 5$$

$$6. \neg F \vee L \quad \text{резольвента (1,5)}$$

$$7. \neg F \quad \text{резольвента (2.a,5)}$$

$$8. U \quad \text{резольвента (2.6,5)}$$

$$9. \neg U \vee L \quad \text{резольвента (3,5)}$$

$$10. \neg L \quad \text{резольвента (4,7)}$$

$$11. \neg U \quad \text{резольвента (9,10)}$$

$$12. \square \quad \text{резольвента (8,11)}$$

Отримано пустий диз'юнкт, отже система є суперечливою.

Приклад 2.

$\{A \rightarrow B, C \vee A, C \rightarrow E, \neg E, \neg E \rightarrow \neg C\}$

Метод резолюції:

$$1. \neg A \vee B \quad \Gamma 1$$

$$2. C \vee A \quad \Gamma 2$$

- | | |
|--------------------|---------------------|
| 3. $\neg C \vee E$ | ГЗ |
| 4. $\neg E$ | Г4 |
| 5. $E \vee \neg C$ | Г5 — співпадає з ГЗ |
| 6. $\neg C$ | резольвента (3,4) |
| 7. A | резольвента (2,6) |
| 8. B | резольвента (1,7) |

Пустий диз'юнкт отримати не можна, система не є суперечливою (викресленні диз'юнкти з унікальними літерами).

- 1) $\{ A \vee B \rightarrow C \ \& \ D, \ D \vee E \rightarrow G, A \vee \neg G ;$
- 2) $\{ (A \rightarrow B) \& (C \rightarrow D), (B \rightarrow D) \& (\neg C \rightarrow A), (E \rightarrow G) \& (G \rightarrow \neg D), \neg E \rightarrow E \};$
- 3) $\{ (A \rightarrow B \& C) \& (D \rightarrow B \& E), (G \rightarrow \neg A) \& H \rightarrow I, (H \rightarrow I) \rightarrow G \& D, \neg(\neg C \rightarrow E) \}$

Завдання 2.3.3. Перевірити логічне слідування методом резолюції.

Приклад 1. Джон або втомився, або він хворий. Якщо він втомився, то він роздратований. Він не роздратований. Отже, Джон – хворий.

Формалізуємо в логіці висловлювань гіпотези.

Нехай A – «Джон втомився», B – «Джон хворий», C – «Джон роздратований».

Формалізуємо умову:

- | | |
|----------------------|------------|
| 1. $A \vee B$ | Γ_1 |
| 2. $A \rightarrow C$ | Γ_2 |
| 3. $\neg C$ | Γ_3 |

Можна перевірити логічне слідування **методом резолюції**. Для цього перетворимо кожен гіпотезу (посилку) у диз'юнкт, візьмемо заперечення висновку і складемо множину диз'юнктів, яку перевіримо на несуперечливість за правилом резолюції Робінсона.

Складемо систему: $\{ A \vee B, \neg A \vee C, \neg C, \neg B \}$.

- | | |
|--------------------|----------------------|
| 1. $A \vee B$ | Γ_1 |
| 2. $\neg A \vee C$ | Γ_2 |
| 3. $\neg C$ | Γ_3 |
| 4. $\neg B$ | заперечення висновку |
| 5. A | резольвента 1, 4 |
| 6. C | резольвента 2, 5 |
| 7. $\square$ | резольвента 3, 6 |

Отримано пустий диз'юнкт, отже система, що складається з гіпотез і заперечення висновку, суперечлива; тобто логічне слідування є істинним.

Зауваження. Порівняйте з використанням методу резолюції в попередньому завданні – 2.3.2.. Метод резолюції методично перевіряє систему на суперечливість шляхом пошуку за правилом Робінсона пустого диз'юнкта (якщо знайде $\square$, то суперечлива; якщо ні – то несуперечлива). Глобальні висновки залежать від того, ЩО ми перевіряємо на суперечливість. Ми можемо перевіряти будь яку систему посилок (як в завданні 2.3.2.) – наприклад, систему правил бази знань, систему гіпотез для постановки задачі. В такому випадку наявність $\square$ робить неможливим подальше використання системи, тому що вона є суперечливою. Інша справа – перевірка логічного слідування з множини гіпотез (поточна задача) – множина диз'юнктів, що перевіряється на суперечливість, складається з диз'юнктів гіпотез та з диз'юнктів, отриманих від заперечення запропонованого висновку. Тоді наявність $\square$ – це добре, це означає, що система стала суперечливою при додаванні до неї заперечення висновку, отже, висновок – дійсний. А, якщо не отримано $\square$, то висновок не є слідуванням з системи гіпотез.

Приклад 2. *Малі діти нерозумні. Той, хто може приборкувати крокодилів, заслуговує на повагу. Нерозумні люди не заслуговують на повагу. Отже, малі діти не приборкують крокодилів.*

Нехай A – «малі діти», B – «розумні», C – «заслуговують на повагу», D – «можуть приборкувати крокодилів». Формалізуємо задачу:

1. $A \rightarrow \neg B$ Γ_1
2. $D \rightarrow C$ Γ_2
3. $\neg B \rightarrow \neg C$ Γ_3

Для перевірки висновка методом резолюції перетворимо кожну гіпотезу (посилку) у диз'юнкт, візьмемо заперечення висновку і складемо множину диз'юнктив, яку перевіримо на несуперечливість за правилом резолюції Робінсона.

Складемо систему: $\{ \neg A \vee \neg B, \neg D \vee C, B \vee \neg C, A, D \}$.

1. $\neg A \vee \neg B$ Γ_1
2. $\neg D \vee C$ Γ_2
3. $B \vee \neg C$ Γ_3
4. A **заперечення висновку:**
5. D $\neg (A \rightarrow \neg D) = \neg (\neg A \vee \neg D) = A \& D$
6. $\neg B$ *резольвента 1,4*
7. $\neg C$ *резольвента 3,6*
8. $\neg D$ *резольвента 2,7*
9. $\square$ *резольвента 5,8*

Отримано пустий диз'юнкт, отже система, що складається з гіпотез і заперечення висновку, суперечлива; тобто логічне слідування є істинним.

1–8) Перевірити логічний висновок методом резолюції задач із завдання 2.2.1.

9) Кульки синього кольору (В) легко проколоти (Р). Фірмові кульки (F) відрізняються особливою міцністю ($\neg P$). Кульки у формі "серце" (H) бувають тільки фірмовими. Кульки у формі "серце" - завжди синього або сріблястого кольору (S). Зараз у нас тут (Z) всі кульки мають форму "серце". Отже, зараз у нас тут кульки сріблясті.

10) Дерева, які ростуть в цьому саду (D), плодоносять (Р). Дерева, які плодоносять, дають хороший урожай (U). Дерева, що дають хороший урожай, отримують ретельний догляд (R). Жодне дерево в цьому саду не отримує ретельного догляду. Отже, в цьому саду не ростуть дерева ($\neg D$).

11) Якщо результат перегонів буде вирішений змовою (Z), або в будинках будуть орудувати шулери (S), то доходи від туризму впадуть (Т) і місто постраждає (М). Якщо доходи від туризму впадуть, то поліція буде задоволена (Р). Поліція ніколи не буває задоволена. Отже, змови на перегонах не буде.

Для задач 9-11 провести також формальний висновок.

2.4. Формальні теорії. Числення висловлювань

Теоретичні відомості

Теорія називається *змістовною*, якщо існує яка-небудь інтерпретація об'єктів, операцій, символів теорії. Алгебра висловлювань є змістовною теорією, оскільки кожен символ в ній інтерпретується як просте висловлювання, істинне або хибне.

Формальні теорії будуються за наступними правилами.

Задається зчисленна множина символів, що називається *алфавітом* теорії. З цієї множини символів будуються зчисленні послідовності, які

називаються *словами* або *виразами* даної теорії. З множини виразів виділяють *правильно побудовані вирази* – *формули*. З множини формул виділяють підмножину *аксіом*. Між формулами теорії визначають відношення – *правила виведення* теорії. Правила виведення дозволяють з множини аксіом виводити *теореми*. Таким чином, формальна теорія є аксіоматичною теорією.

Визначення 2.7. *Доведенням* теореми називають послідовність формул $A_1, \dots, A_n$, кожна з яких або є аксіомою, або отримана з попередніх за правилами виведення цієї теорії. Остання формула називається *теоремою* цієї теорії. Іншими словами, *теорема* - це формула, що виводиться з множини аксіом за правилами виведення, визначеними в цій теорії. Запис $\vdash_J A_n$ означає, що формула A_n виводиться в теорії J , тобто є теоремою теорії J .

Визначення 2.8. *Виведенням* формули A_n з множини гіпотез Γ називається послідовність формул $A_1, \dots, A_n$, кожна з яких є або аксіомою, або гіпотезою з множини Γ , або отриманою з попередніх по *правилах виведення*. Формула $B = A_n$ називається такою, що виводиться з множини гіпотез Γ . Означення $\Gamma \vdash_J A_n$ означає, що формула A_n виводиться з множини гіпотез Γ в теорії J .

Доведення відрізняється від виведення тим, що в ньому не використовуються гіпотези, тому теорему можна визначити як формулу, що виводиться з порожньої множини гіпотез.

Визначення 2.9.

1. *Символами алфавіту* теорії L є пропозиціональні літери $A, B, C, \dots$ з індексами або без індексів, пропозиціональні зв'язки $\neg, \rightarrow$, та допоміжні символи – дужки: $($ та $)$.
2. *Визначення формули.*

Будь-яка пропозиціональна літера є формула.

Якщо A та B є формулами, то формулами є $(\neg A)$, $(A \rightarrow B)$.

Інших формул немає.

3. У формальній теорії L визначена нескінченна кількість аксіом, що задаються трьома схемами аксіом:

A1: $A \rightarrow (B \rightarrow A)$;

A2: $(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$;

A3: $(\neg B \rightarrow \neg A) \rightarrow ((\neg B \rightarrow A) \rightarrow B)$.

Конкретні аксіоми виходять підстановкою формули замість пропозиціональної літери.

4. В теорії L визначено єдине правило виводу МР:

$A, A \rightarrow B \vdash B$.

Для скорочування запису формул вводяться метавизначення:

MB1: $\neg(A \rightarrow \neg B) = A \& B$.

MB2: $\neg A \rightarrow B = A \vee B$.

MB3: $(A \rightarrow B) \& (B \rightarrow A) = A \equiv B$.

Теорема 1 (L). $\vdash A \rightarrow A$ (закон тотожності - висловлювання є логічним наслідком самого себе)

Теорема 2 (L). $\vdash (A \rightarrow A) \rightarrow A$

B1. Правило силізму.

$A \rightarrow B, B \rightarrow C \vdash A \rightarrow C$

B2. Правило видалення середнього посилення.

$A \rightarrow (B \rightarrow C), B \vdash A \rightarrow C$

B3. Правило видалення крайнього посилення.

$(A \rightarrow B) \rightarrow C \vdash B \rightarrow C$.

Метатеорема про дедукцію (МТД).

Якщо з множини формул Γ і формули A виводима формула B , то з множини Γ виводима формула $A \rightarrow B$, тобто якщо $\Gamma, A \vdash B$, то $\Gamma \vdash A \rightarrow B$.

Обернена метатеорема про дедукцію.

Якщо існує вивід $\Gamma \vdash A \rightarrow B$, то формула B виводиться з Γ і A , т.е. якщо $\Gamma \vdash A \rightarrow B$, то $\Gamma, A \vdash B$

Теорема 3 (L) (зняття подвійного заперечення). $\vdash \neg\neg A \rightarrow A$

Теорема 4 (L) (введення подвійного заперечення). $\vdash A \rightarrow \neg\neg A$

Теорема 5 (L) (протиріччя/суперечність). $\vdash \neg A \rightarrow (A \rightarrow B)$

Теорема 6 (L) (контрапозиції). $\vdash (\neg A \rightarrow \neg B) \rightarrow (B \rightarrow A)$

Теорема 7 (L) (контрапозиції). $\vdash (B \rightarrow A) \rightarrow (\neg A \rightarrow \neg B)$

Теорема 8 (L). $\vdash A \rightarrow (\neg B \rightarrow \neg(A \rightarrow B))$

Теорема 9 (L). $\vdash (A \rightarrow B) \rightarrow ((\neg A \rightarrow B) \rightarrow B)$

Приклад 1. Метавизначення $\vdash (A \rightarrow B) \rightarrow \neg A \cup B$

ОМТД 2 рази і МВ 2 ($\neg A \rightarrow B = A \cup B$) $A \rightarrow B, \neg\neg A \vdash B$

- | | | |
|----|----------------------------|------------|
| 1. | $A \rightarrow B$ | $\Gamma 1$ |
| 2. | $\neg\neg A$ | $\Gamma 2$ |
| 3. | $\neg\neg A \rightarrow A$ | T3 |
| 4. | A | MP(2,3) |
| 5. | B | MP(1,4) |

Приклад 2. $\vdash \neg(A \cup B) \rightarrow \neg A \& \neg B$ – Закон де Моргана

ОМТД $\neg(A \cup B) \vdash \neg A \& \neg B$

МО1, МО2 $\neg(\neg A \rightarrow B) \vdash (\neg A \rightarrow \neg\neg B)$

Доведемо допоміжну лему

ЛЕМА $\vdash (\neg A \rightarrow \neg\neg B) \rightarrow (\neg A \rightarrow B)$

ОМТД $(\neg A \rightarrow \neg\neg B), A \vdash B$

- | | | |
|----|---------------------------------|------------|
| 1. | $\neg A \rightarrow \neg\neg B$ | $\Gamma 1$ |
| 2. | $\neg A$ | $\Gamma 2$ |

3. $\neg\neg B$ MP(1,2)
4. $\neg\neg B \rightarrow B$ T3
5. B MP(3,4)

Продовжимо доведення початкової теореми

1. $\neg(\neg A \rightarrow B)$ Г1
2. $\vdash (\neg A \rightarrow \neg\neg B) \rightarrow (\neg A \rightarrow B)$ Лема
3. $((\neg A \rightarrow \neg\neg B) \rightarrow (\neg A \rightarrow B)) \rightarrow (\neg(\neg A \rightarrow B) \rightarrow \neg(\neg A \rightarrow \neg\neg B))$ T7
4. $(\neg(\neg A \rightarrow B) \rightarrow \neg(\neg A \rightarrow \neg\neg B))$ MP(2,3)
5. $\neg(\neg A \rightarrow \neg\neg B)$ MP(1,4)

Приклад 3. $\vdash ((A \rightarrow B) \rightarrow A) \rightarrow A$ – Закон Пірса

ОМТД $(A \rightarrow B) \rightarrow A \vdash A$;

1. $(A \rightarrow B) \rightarrow A$ Г1
2. $\vdash \neg A \rightarrow (A \rightarrow B)$ T5
3. $\neg A \rightarrow A$ B1 (1,2)
4. $(\neg A \rightarrow A) \rightarrow A$ T2
5. A MP 3,4

Завдання 2.4.1. Довести теореми теорії L.

- 1) $\vdash ((A \rightarrow B) \rightarrow A) \rightarrow ((A \rightarrow B) \rightarrow B)$;
- 2) $\vdash (A \rightarrow C) \rightarrow (A \rightarrow B \vee C)$;
- 3) $\vdash (\neg A \rightarrow B) \rightarrow (\neg B \rightarrow A)$;
- 4) $\vdash (B \rightarrow C) \rightarrow (B \rightarrow (A \rightarrow C))$;
- 5) $\vdash (B \rightarrow A) \rightarrow (A \vee B \rightarrow A)$;
- 6) $\vdash (A \rightarrow (B \rightarrow C)) \rightarrow (B \rightarrow (A \rightarrow C))$;
- 7) $\vdash A \& B \rightarrow (C \rightarrow B)$;
- 8) $\vdash ((B \rightarrow A) \rightarrow C) \rightarrow (A \rightarrow C)$;
- 9) $\vdash (A \rightarrow B) \& (B \rightarrow C) \rightarrow (A \rightarrow C)$;

- 10) $\vdash (A \rightarrow B) \rightarrow ((B \rightarrow C) \rightarrow (A \rightarrow C));$
- 11) $\vdash (A \rightarrow B) \& (C \rightarrow D) \rightarrow (A \vee C \rightarrow B \vee D);$
- 12) $\vdash (A \vee B \rightarrow C) \rightarrow (A \rightarrow C) \vee (B \rightarrow C);$
- 13) $\vdash (A \rightarrow B) \& (C \rightarrow D) \rightarrow (A \& C \rightarrow B \& D);$
- 14) $\vdash \neg(A \& B) \rightarrow (A \& B \rightarrow B);$
- 15) $\vdash (B \rightarrow C) \rightarrow (A \vee C \rightarrow (B \rightarrow C));$
- 16) $\vdash (A \rightarrow C) \& (B \rightarrow C) \rightarrow (A \vee B \rightarrow C);$
- 17) $\vdash (A \rightarrow B \vee C) \rightarrow (\neg A \vee B \vee C);$
- 18) $\vdash (A \vee B \rightarrow C) \rightarrow (B \rightarrow C) \& (A \rightarrow C);$
- 19) $\vdash (A \rightarrow B \& C) \rightarrow (A \rightarrow B) \& (A \rightarrow C);$
- 20) $\vdash (A \rightarrow (B \rightarrow C)) \rightarrow (A \& B \rightarrow C).$

Відповіді до розділу 2

2.1.1. Тавтології 1,2,4,5,6; НІ – 3,7.

2.1.2.

- | | |
|----------------|-----------------|
| 1) нейтральна; | 6) тавтологія; |
| 2) протиріччя; | 7) нейтральна; |
| 3) тавтологія; | 8) тавтологія; |
| 4) тавтологія; | 9) протиріччя; |
| 5) протиріччя; | 10) нейтральна. |

2.1.3.

- | | |
|-------------------|-------------------|
| 1) несуперечлива; | 4) несуперечлива; |
| 2) суперечлива; | 5) несуперечлива; |
| 3) суперечлива; | 6) несуперечлива. |

2.1.3. 3)

$\{(A \rightarrow B \& C) \& (D \rightarrow B \& E), (G \rightarrow \neg A) \& H \rightarrow I, (H \rightarrow I) \rightarrow G \& D, \neg(\neg C \rightarrow E)\}$

Видалімо кон'юнкцію в першій гіпотезі; перетворювати посилки не потрібно.

Всі посилки повинні дорівнювати Т.

1 $A \rightarrow B \& C$

2 $D \rightarrow B \& E$

3 $(G \rightarrow \neg A) \& H \rightarrow I$

4 $(H \rightarrow I) \rightarrow G \& D$

Розглянемо останню посилку $\neg(\neg C \rightarrow E) = T$. Тоді $\neg C \rightarrow E = F$, і з цього слідує те, що записано в п. 5 і 6.

$$5 \quad |\neg C|=T \quad \Rightarrow |C|=F$$

$$6 \quad |E|=F$$

З того, що в 5 та 6 визначено значення С і Е, з 1 та 2 $\Rightarrow |A|=F, |D|=F$, а В може приймати будь-яке значення.

З 4 через хибне значення D консеквент вже буде хибний, незалежно від значення G. Тоді для істинності посилки $|H \rightarrow I|=F \Rightarrow |H|=T, |I|=F$.

Підставляємо отримані значення в 3: $|(G \rightarrow \neg A) \& H \rightarrow I| = |(G \rightarrow \neg A) \& T \rightarrow F| = T$

Для виконання умови істинності посилки 3 треба мати $|G \rightarrow \neg A| = F$.

Але у нас вже відомо значення А і $|\neg A|=T$, що робить одночасну здійсненність системи неможливою Система – суперечлива.

2.2.1.

1) ФОРМАЛІЗАЦІЯ і формальний висновок

$$1. Z \rightarrow P \quad \Gamma 1$$

$$2. D \rightarrow P \quad \Gamma 2$$

$$3. P \rightarrow A \quad \Gamma 3$$

$$4. A \rightarrow I \quad \Gamma 4$$

$$5. I \rightarrow R \quad \Gamma 5$$

Висновок: $Z \vee D \rightarrow R$

$$6. P \rightarrow I \quad \text{силогізм 3, 4}$$

$$7. P \rightarrow R \quad \text{силогізм 6, 5}$$

$$8. (Z \rightarrow P) \& (D \rightarrow P) = (\neg Z \vee P) \& (\neg D \vee P) = (\neg Z \& \neg D) \vee P = \neg (Z \vee D) \vee P = (Z \vee D) \rightarrow P \quad \text{введення \& та екв.перетворення 1 і 2}$$

$$9. Z \vee D \rightarrow R \quad \text{силогізм 8, 7}$$

2) ФОРМАЛІЗАЦІЯ і формальний висновок

1. $V \rightarrow S \vee D$
2. $S \rightarrow \neg V \ \& \ P$
3. $P \rightarrow S \vee D$
4. V

Висновок : $\neg D \rightarrow S$

5. $S \vee D = D \vee S = \neg D \rightarrow S$ МР 1, 4 та екв.перетворення

Гіпотези 2 та 3 – зайві для отримання даного висновку.

3) ФОРМАЛІЗАЦІЯ і формальний висновок

1. $V \rightarrow Z$
2. $Z \rightarrow B$
3. $\neg V \rightarrow D$
4. $B \rightarrow E$
5. $D \rightarrow B$
6. $V \vee \neg V$

Висновок : E

7. $V \rightarrow B$ силогізм 1, 2
8. $V \rightarrow E$ силогізм 7, 4
9. $\neg V \rightarrow B$ силогізм 3, 5
10. $\neg V \rightarrow E$ силогізм 9, 4
11. $\neg E \rightarrow \neg V$ контрапозиція 8
12. $\neg E \rightarrow V$ контрапозиція 10
13. $\neg \neg E = E$ доведення до абсурду і зняття подвійного заперечення 11,12

4) ФОРМАЛІЗАЦІЯ і формальний висновок (не виконується)

1. $P \rightarrow H \ \& \ T = \neg P \vee (H \ \& \ T) = (\neg P \vee H) \ \& \ (\neg P \vee T) = (P \rightarrow H) \ (P \rightarrow T)$
2. $\neg P$

Висновок : $\neg H$

3. $P \rightarrow H$ видалення & з 1

4. $P \rightarrow T$ видалення & з 1

5) ФОРМАЛІЗАЦІЯ і формальний висновок

1. $Z \rightarrow I$ (тільки)

2. $I \rightarrow V$

3. $V \rightarrow \neg H$

Висновок: $Z \rightarrow \neg H$

4. $Z \rightarrow V$ силогізм 1, 2

5. $Z \rightarrow \neg H$ силогізм 4, 3

6) ФОРМАЛІЗАЦІЯ і формальний висновок

1. $Z \rightarrow K \vee Y \vee P$ якщо страйк завершено, то або конгрес прийняв певні рішення або президент пішов у відставку або страйк тривав більше року

2. $\neg K$

3. $\neg P$

4. Z

Висновок: Y

5. $K \vee Y \vee P$ МР 1, 4

6. $\neg K \rightarrow (\neg P \rightarrow Y)$ еквівал.перетворення 5

7. $\neg P \rightarrow Y$ МР 2, 6

8. Y МР 3, 7

7) ФОРМАЛІЗАЦІЯ і формальний висновок

1. $A \vee B$

2. $A \rightarrow C$

3. $B \rightarrow D \vee C$

4. $\neg C$

Висновок: D

5. $\neg A \rightarrow B$ екв.перетв.1

6. $\neg A$ МТ 2, 4

7. $\neg A \rightarrow D \vee C$ сил. 3, 5

8. $D \vee C$ МР 6, 7

9. $\neg D \rightarrow C$ екв.перетв.8

10. $\neg C \rightarrow D$ контрапозиція 9

11. D МР 4, 10

8) ФОРМАЛІЗАЦІЯ і формальний висновок

1. $\neg B \rightarrow (C \vee P)$ Г1

2. $\neg C \rightarrow (\neg B \& M) = C \vee (\neg B \& M) = (C \vee \neg B) \& (C \vee M) =$
 $= (\neg C \rightarrow \neg B) \& (\neg C \rightarrow M)$ Г2

3. $M \rightarrow (C \vee \neg P)$ Г3

Висновок: C

4. $\neg C \rightarrow \neg B$ екв. та видалення & 2

5. $\neg C \rightarrow M$ екв. та видалення & 2

6. $\neg C \rightarrow C \vee P = C \vee C \vee P = C \vee P = \neg C \rightarrow P$ сил. 4, 1, екв. та ідемпот.

7. $\neg C \rightarrow C \vee \neg P = C \vee C \vee \neg P =$
 $= C \vee \neg P = \neg C \rightarrow \neg P$ сил. 5, 3, екв. та ідемпот.

8. $\neg \neg C = C$ абсурд 6,7 та зняття подвійного заперечення

2.3.1.

1) $(B \rightarrow C) \rightarrow (A \vee C \rightarrow (B \rightarrow C))$:

Якщо ми допускаємо, що формула – тавтологія, то можемо «розібрати» її за правилами ОМТД: $B \rightarrow C, A \vee C, B \mid - C$

Отримаємо множину диз'юнктив з посилок та заперечення висновку:

$$\{ \neg B \vee C, A \vee C, B, \neg C \}.$$

Ми бачимо, що пустий диз'юнкт отримати можна (навіть без другої посилки, вона – зайва), отже, система є суперечливою (за рахунок додавання заперечення висновку), тобто формула – тавтологія.

2) $(A \rightarrow (B \rightarrow C)) \rightarrow (A \vee B \rightarrow C)$:

Розберемо формулу так, як би вона була тавтологією (відокремлюємо комами підформули, рухаючись зліва направо по черзі, починаючи з самої зовнішньої операції):

$$1) (A \rightarrow (B \rightarrow C)) \rightarrow (A \vee B \rightarrow C);$$

$$2) A \rightarrow (B \rightarrow C), (A \vee B \rightarrow C);$$

$$3) A \rightarrow (B \rightarrow C), A \vee B$$

та висновок, що перевіряється в нашому випадку C .

Таким чином перетворюючи першу посилку та взявши заперечення висновку отримаємо множину диз'юнктив $\{ \neg A \vee \neg B \vee C, A \vee B, \neg C \}$.

Спробуємо застосувати правило Робінсона і отримати резольвенти:

$$1. \neg C - \text{заперечення висновку}$$

$$2. \neg A \vee \neg B \vee C - \text{посилка (гіпотеза) 1}$$

3. $A \vee B$ – посилка (гіпотеза) 2

4. $\neg A \vee \neg B$ – резольвента 1,2

5. $B \vee \neg B$ – резольвента 3,4 *

Пустий диз'юнкт отримати не можна, тобто, формула не є тавтологією.

3) $(A \rightarrow B) \rightarrow ((B \rightarrow C) \rightarrow (A \rightarrow C))$:

$\{ A \rightarrow B, B \rightarrow C, A, \neg C \}$

$\{ \neg A \vee B, \neg B \vee C, A, \neg C \}$

1 $\neg A \vee B$ Г1

2 $\neg B \vee C$ Г2

3 A Г3

4 $\neg C$ запереч. висн.

5 B резольвента 1,3

6 C резольвента 2,5

7 $\square$ резольвента 4,6

4) $(A \rightarrow B) \rightarrow ((C \rightarrow B) \rightarrow (A \vee C \rightarrow B))$:

1 $A \rightarrow B = \neg A \vee B$ Г1

2 $C \rightarrow B = \neg C \vee B$ Г2

3 $A \vee C$ Г3

4 $\neg B$ заперечення висновку

5 $\neg A$ резольвента 1, 4

6 $\neg C$ резольвента 2, 4

7 C резольвента 3, 5

8 $\square$ резольвента 6, 7

5) $(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$:

$$A \rightarrow (B \rightarrow C), A \rightarrow B \models A \rightarrow C$$

$$\neg(A \rightarrow C) = \neg(\neg A \vee C) = A \& \neg C$$

$$\{\neg A \vee \neg B \vee C, \neg A \vee B, A, \neg C\}$$

Пустий диз'юнкт отримати можна, формула – тавтологія.

2.3.2.

2) $\{A \vee B \rightarrow C \& D, D \vee E \rightarrow G, A \vee \neg G\}$

1 ~~$A \vee G$~~

2 ~~$A \vee D$~~

3 ~~$B \vee G$~~

4 ~~$B \vee D$~~

5 ~~$D \vee G$~~

6 ~~$E \vee G$~~

7 $A \vee \neg G$

8 ~~$A \vee G$~~ резольвента 2,5

9 ~~$A \vee A$~~ або ~~$G \vee G$~~ (резольвента 7,8) = тавтології

Пустий диз'юнкт не отримано, система не є суперечливою.

3) $\{(A \rightarrow B) \& (C \rightarrow D), (B \rightarrow D) \& (\neg C \rightarrow A), (E \rightarrow G) \& (G \rightarrow \neg D), \neg E \rightarrow E\}$

1 $\neg A \vee B$

2 $\neg C \vee D$

3 $\neg B \vee D$

4 $C \vee A$

5 $\neg E \vee G$

6 $\neg G \vee \neg D$

7 E

8	G	резольвента 7,5
9	¬D	резольвента 8,6
10	¬C	резольвента 9,2
11	¬B	резольвента 9,3
12	¬A	резольвента 11,1
13	C	резольвента 12,4
14	□	резольвента 10,13

Система – суперечлива, вона не має цінності для розв’язку задачі.

3) $\{(A \rightarrow B \& C) \& (D \rightarrow B \& E), (G \rightarrow \neg A) \& H \rightarrow I, (H \rightarrow I) \rightarrow G \& D, \neg(\neg C \rightarrow E)\}$

Перетворення:

$$1 \ A \rightarrow B = \neg A \vee B$$

$$2 \ A \rightarrow C = \neg A \vee C$$

$$3 \ D \rightarrow B = \neg D \vee B$$

$$4 \ D \rightarrow E = \neg D \vee E$$

$$5 \ (G \rightarrow \neg A) \& H \rightarrow I = \neg((\neg G \vee \neg A) \& H) \vee I = \neg(\neg G \vee \neg A) \vee \neg H \vee I = \\ = (G \& A) \vee \neg H \vee I = (G \vee \neg H \vee I) \& (A \vee \neg H \vee I)$$

$$6 \ (H \rightarrow I) \rightarrow G \& D = \neg(\neg H \vee I) \vee G \& D = H \& \neg I \vee G \& D = \\ = (H \vee G) \& (H \vee D) \& (\neg I \vee G) \& (\neg I \vee D)$$

$$7 \ \neg(\neg C \rightarrow E) = \neg(C \vee E) = \neg C \& \neg E$$

РЕЗОЛЮЦІЯ

$$1 \ \neg A \vee B$$

$$7 \ H \vee G$$

$$2 \ \neg A \vee C$$

$$8 \ H \vee D$$

$$3 \ \neg D \vee B$$

$$9 \ \neg I \vee G$$

$$4 \ \neg D \vee E$$

$$10 \ \neg I \vee D$$

$$5 \ G \vee \neg H \vee I$$

$$11 \ \neg C$$

$$6 \ A \vee \neg H \vee I$$

$$12 \ \neg E$$

Викреслено унікальні літери, але порожній диз'юнкт отримати можна – система суперечлива.

13 $\neg A$	резольвента 2, 11
14 $\neg H \vee I$	резольвента 6, 13
15 $\neg D$	резольвента 4, 12
16 H	резольвента 8, 15
17 $\neg I$	резольвента 8, 10
19 $\neg H$	резольвента. 14, 17
20 $\square$	резольвента 16, 19

2.3.3.

1) 1. $\neg Z \vee P$	$\Gamma 1$
2. $\neg D \vee P$	$\Gamma 2$
3. $\neg P \vee A$	$\Gamma 3$
4. $\neg A \vee I$	$\Gamma 4$
5. $\neg I \vee R$	$\Gamma 5$

Отримаємо заперечення висновку

$$\neg (Z \vee D \rightarrow R) = \neg (\neg (Z \vee D) \vee R) = (Z \vee D) \& \neg R$$

6. $Z \vee D$	запереч.висновку
7. $\neg R$	запереч.висновку
8. $\neg I$	резольвента 7, 5
9. $\neg A$	резольвента 4, 8
10. $\neg P$	резольвента 3, 9
11. $\neg Z$	резольвента 1, 10
12. $\neg D$	резольвента 2, 10
13. D	резольвента 6, 11
14. $\square$	резольвента 12, 13

2) 1. $\neg V \vee S \vee D$

2. V

$\neg (\neg D \rightarrow S) = \neg (D \vee S) = \neg D \wedge \neg S$ заперечення висновку

3. $\neg D$

4. $\neg S$

Далі, використовуючи правило Робінсона, можна отримати пустий диз'юнкт без застосування гіпотез 2 і 3.

3) 1. $\neg E$

2. $\neg V \vee Z$

3. $\neg Z \vee B$

4. $V \vee D$

5. $\neg B \vee E$

6. $\neg D \vee B$

7. ~~$V \vee \neg V$~~

8. $\neg B$ резольвента 1, 5

9. $\neg Z$ резольвента 3, 8

10. $\neg V$ резольвента 2, 9

11. D резольвента 4, 10

12. B резольвента 6, 11

13. $\square$ резольвента 8, 12

4) 1. H

2. $\neg P$

3. $\neg P \vee H$

4. $\neg P \vee T$

Жодної резольвенти отримати не можна.

5) 1. Z заперч. висн.

2. H заперч. висн.

3. $\neg Z \vee I$ Г1

- 4. $\neg I \vee V$ Г2
- 5. $\neg V \vee \neg H$ Г3
- 6. I резольвента 1, 3
- 7. V резольвента 4, 6
- 8. $\neg H$ резольвента 5, 7
- 9. $\square$ резольвента 2, 8

- 6)** 1. $\neg Y$ запереч.висновку
- 2. $\neg Z \vee K \vee Y \vee P$ Г1
 - 3. $\neg K$ Г2
 - 4. $\neg P$ Г3
 - 5. Z Г4
 - 6. $K \vee Y \vee P$ резольвента 2, 5
 - 7. $Y \vee P$ резольвента 3, 6
 - 8. P резольвента 1, 7
 - 9. $\square$ резольвента 4, 8

- 7)** 1. $\neg D$ заперечення висновку
- 2. $A \vee B$ Г1
 - 3. $\neg A \vee C$ Г2
 - 4. $\neg B \vee D \vee C$ Г3
 - 5. $\neg C$ Г4
 - 6. $\neg B \vee C$ резольвента 1, 4
 - 7. $A \vee C$ резольвента 2, 6
 - 8. C резольвента 3, 7
 - 9. $\square$ резольвента 5, 8

- 8)** 1. $\neg C$ заперечення висновку
- 2. $B \vee C \vee P$ Г1
 - 3. $C \vee \neg B$ Г2

4. $C \vee M$	$\Gamma 2$
5. $\neg M \vee C \vee \neg P$	$\Gamma 3$
6. $B \vee P$	резольвента 1, 2
7. $\neg B$	резольвента 1, 3
8. M	резольвента 1, 4
9. $\neg M \vee \neg P$	резольвента 1, 5
10. P	резольвента 6, 7
11. $\neg M$	резольвента 10, 9
12. $\square$	резольвента 8, 11

- 9) 1. $B \rightarrow P$
 2. $F \rightarrow \neg P$
 3. $H \rightarrow F$
 4. $H \rightarrow B \vee S$
 5. $Z \rightarrow H$ ВИСНОВОК: $Z \rightarrow S$

Формальний висновок:

6. $Z \rightarrow F$ сил. 5, 3
 7. $Z \rightarrow \neg P$ сил. 6, 2
 8. $\neg P \rightarrow \neg B$ контр. 1
 9. $Z \rightarrow \neg B$ сил. 7, 8
 10. $Z \rightarrow B \vee S$ сил. 4, 5
 11. $\neg Z \vee B \vee S = \neg B \rightarrow (Z \rightarrow S)$ екв.перетв. 10
 12. $Z \rightarrow (Z \rightarrow S) = \neg Z \vee \neg Z \vee S = \neg Z \vee S = Z \rightarrow S$

сил. 9, 10+екв.перетв.

Резолюція [заперечення висновку $\neg (Z \rightarrow S) = \neg (\neg Z \vee S) = Z \& \neg S$]:

1. $\neg B \vee P$ $\Gamma 1$
 2. $\neg F \vee \neg P$ $\Gamma 2$
 3. $\neg H \vee F$ $\Gamma 3$
 4. $\neg H \vee B \vee S$ $\Gamma 4$
 5. $\neg Z \vee H$ $\Gamma 5$

- 6. Z запереч.висновку
- 7. $\neg S$ запереч.висновку
- 8. H резольвента 5, 6
- 9. F резольвента 3, 8
- 10. $B \vee S$ резольвента 4,8
- 11. B резольвента 7,10
- 12. P резольвента 1, 11
- 13. $\neg F$ резольвента 2, 12
- 14. $\square$ резольвента 9, 13

10) Формалізація:

- 1. $D \rightarrow P$
- 2. $P \rightarrow U$
- 3. $U \rightarrow R$
- 4. $D \rightarrow \neg R$

Висновок: $\neg D$

Вказівка для формального виведення: З гіпотез 1-3 отримаємо через силогізми $D \rightarrow R$. Разом з гіпотезою 4 через правило доведення до абсурду отримаємо висновок.

11) Формалізація:

- 1. $Z \vee S \rightarrow T \& M$
- 2. $T \rightarrow P$
- 3. $\neg P$

Висновок: $\neg Z$

Вказівка для формального виведення: перша гіпотеза через еквівалентні перетворення розбивається на чотири дволітерні диз'юнкти.

$$\begin{aligned}
 Z \vee S \rightarrow T \& M &= \neg (Z \vee S) \vee (T \& M) = (\neg Z \& \neg S) \vee (T \& M) = \\
 &= (\neg Z \vee T) \& (\neg Z \vee M) \& (\neg S \vee T) \& (\neg S \vee M) = \\
 &= (Z \rightarrow T) \& (Z \rightarrow M) \& (S \rightarrow T) \& (S \rightarrow M).
 \end{aligned}$$

2.4.1.

1) $\vdash ((A \rightarrow B) \rightarrow A) \rightarrow ((A \rightarrow B) \rightarrow B);$

$(A \rightarrow B) \rightarrow A, A \rightarrow B \vdash B$ 2 рази ОМТД

1 $(A \rightarrow B) \rightarrow A$ Γ_1

2 $A \rightarrow B$ Γ_2

3 A МР 1,2

4 B МР 2,3

2) $\vdash (A \rightarrow C) \rightarrow (A \rightarrow B \vee C);$

$A \rightarrow C, A \vdash B \vee C$ 2 рази ОМТД

$A \rightarrow C, A \vdash \neg B \rightarrow C$ МВ

$A \rightarrow C, A, \neg B \vdash C$ ОМТД

1 $A \rightarrow C$ Γ_1

2 A Γ_2

3 $\neg B$ Γ_3

4 C МР 1,2

3) $\vdash (\neg A \rightarrow B) \rightarrow (\neg B \rightarrow A);$

$\neg A \rightarrow B, \neg B \vdash A$ 2 РАЗИ ОМТД

1 $\neg A \rightarrow B$ Γ_1

2 $\neg B$ Γ_2

3 $(\neg A \rightarrow B) \rightarrow (\neg B \rightarrow \neg \neg A)$ Т7

4 $\neg B \rightarrow \neg \neg A$ МР 1,3

5 $\neg \neg A \rightarrow A$ Т3

6 $\neg B \rightarrow A$ В1(сил) 4,5 **або** 6 $\neg \neg A$ МР 2,4

7 A МР 2,6 7 A МР 5,6

4) $\vdash (B \rightarrow C) \rightarrow (B \rightarrow (A \rightarrow C));$

$B \rightarrow C, B, A \vdash C$ 3 рази ОМТД

1 $B \rightarrow C$ Γ_1

2 B Γ_2

3 A Γ3

4 C MP 1, 2

5) $\vdash (B \rightarrow A) \rightarrow (A \vee B \rightarrow A)$;

$B \rightarrow A, A \vee B \vdash A$ 2 рази ОМТД

1 $B \rightarrow A$ Γ1

2 $A \vee B$ Γ2

3 $\neg A \rightarrow B$ MB 2

4 $\neg A \rightarrow A$ B1(сил) 1,3

5 $(\neg A \rightarrow A) \rightarrow A$ T2

6 A MP 4,5

або

4 $(B \rightarrow A) \rightarrow (\neg A \rightarrow \neg B)$ T7

5 $\neg A \rightarrow \neg B$ MP 1,4

6 $(\neg A \rightarrow \neg B) \rightarrow ((\neg A \rightarrow B) \rightarrow A)$ A3

7 $(\neg A \rightarrow B) \rightarrow A$ MP 5, 6

8 A MP 3, 7

11) $\vdash (A \rightarrow B) \& (C \rightarrow D) \rightarrow (A \vee C \rightarrow B \vee D)$;

$A \rightarrow B, C \rightarrow D, A \vee C \vdash B \vee D$ 2 рази ОМТД, видалення &

$A \rightarrow B, C \rightarrow D, \neg A \rightarrow C \vdash \neg B \rightarrow D$ MB

$A \rightarrow B, C \rightarrow D, \neg A \rightarrow C, \neg B \vdash D$ ОМТД

1 $A \rightarrow B$ Γ1

2 $C \rightarrow D$ Γ2

3 $\neg A \rightarrow C$ Γ3

4 $\neg B$ Γ4

5 $\neg A \rightarrow D$ B1 (сил) 2,3

6 $(A \rightarrow B) \rightarrow (\neg B \rightarrow \neg A)$ T7

7 $\neg B \rightarrow \neg A$ MP 1, 6

або два MP 4-7 та 8-5 // або B1 7-5 та MP 4-8

8 $\neg A$ MP 4, 7 // 8 $\neg B \rightarrow D$ B1 7, 5
 9 D MP 5, 8 // 9 D MP 4, 8

12) $\vdash (A \vee B \rightarrow C) \rightarrow (A \rightarrow C) \vee (B \rightarrow C);$

$A \vee B \rightarrow C \vdash (A \rightarrow C) \vee (B \rightarrow C)$ ОМТД

$(\neg A \rightarrow B) \rightarrow C \vdash \neg (A \rightarrow C) \rightarrow (B \rightarrow C)$ МВ

$(\neg A \rightarrow B) \rightarrow C, \neg (A \rightarrow C) \vdash (B \rightarrow C)$ ОМТД

1 $(\neg A \rightarrow B) \rightarrow C$ Г1

2 $\neg (A \rightarrow C)$ Г2

3 $B \rightarrow C$ ВЗ (крайн) 1

13) $\vdash (A \rightarrow B) \& (C \rightarrow D) \rightarrow (A \& C \rightarrow B \& D);$

$A \rightarrow B, C \rightarrow D, A, C \vdash B \& D$ 2 рази ОМТД

1 $A \rightarrow B$ Г1

2 $C \rightarrow D$ Г2

3 A Г3

4 C Г4

5 B MP 1,3

6 D MP 2,4

7 $B \& D$ введення & 5,6

16) $\vdash (A \rightarrow C) \& (B \rightarrow C) \rightarrow (A \vee B \rightarrow C);$

$A \rightarrow C, B \rightarrow C, A \vee B \vdash C$ 2 рази ОМТД

1 $A \rightarrow C$ Г1

2 $B \rightarrow C$ Г2

3 $A \vee B$ Г3

4 $\neg A \rightarrow B$ МВ 3

5 $\neg A \rightarrow C$ В1 4, 2

6 $(\neg A \rightarrow C) \rightarrow ((A \rightarrow C) \rightarrow C)$ Т9

7 $(A \rightarrow C) \rightarrow C$ MP 5, 6

8 C MP 1, 7

17) $\vdash (A \rightarrow B \vee C) \rightarrow (\neg A \vee B \vee C);$

$A \rightarrow B \vee C \vdash \neg A \vee B \vee C$ ОМТД

1 $A \rightarrow B \vee C$ Г1

2 $\neg \neg A \rightarrow A$ Т3

3 $\neg \neg A \rightarrow B \vee C$ В1 1, 2

4 $\neg A \vee B \vee C$ МВ 3

Або $A \rightarrow B \vee C \vdash \neg A \vee B \vee C$ ОМТД

$A \rightarrow (\neg B \rightarrow C) \vdash \neg \neg A \rightarrow (\neg B \rightarrow C)$ МВ

$A \rightarrow (\neg B \rightarrow C), \neg \neg A, \neg B \vdash C$ 2 рази ОМТД

1 $A \rightarrow (\neg B \rightarrow C)$ Г1

2 $\neg \neg A$ Г2

3 $\neg B$ Г3

4 $\neg \neg A \rightarrow A$ Т3

5 $\neg \neg A \rightarrow (\neg B \rightarrow C)$ В1 1, 4

6 $\neg B \rightarrow C$ МР2, 5

7 C МР 3,6

18) $\vdash (A \vee B \rightarrow C) \rightarrow (B \rightarrow C) \& (A \rightarrow C);$

Доводимо окремо, як у 18, для $B \rightarrow C$ та $A \rightarrow C$.

Для $B \rightarrow C$ – через метавизначення та видалення крайньої посилки

Доведемо для:

$A \vee B \rightarrow C \vdash A \rightarrow C$

1 $A \vee B \rightarrow C$ Г1

2 $(\neg A \rightarrow B) \rightarrow C$ МВ 1

3 $(\neg B \rightarrow A) \rightarrow (\neg A \rightarrow B)$ правило контрапоз.

(або можна використати комутативний закон –

треба його теж довести $A \vee B \rightarrow B \vee A$)

4 $(\neg B \rightarrow A) \rightarrow C$ В1 2, 3

5 $A \rightarrow C$ В3 4

3. ТЕОРІЯ ПРЕДИКАТИВ ПЕРШОГО ПОРЯДКУ

3.1. Введення кванторів. Побудова таблиць істинності на обмеженій інтерпретації

Теоретичні відомості

Одномісним предикатом $P(x)$, визначеним на множині M , називається вираз, який після підстановки в нього замість x предмета з області визначення M перетворюється в висловлювання. Область визначення предиката називається *предметною областю*. Елементи з області визначення називаються *предметними постійними (предметами)*. Змінна, від якої залежить предикат, називається *предметною змінною*.

Одномісні предикати традиційно служать для формалізації понять.

У загальному випадку n -місний предикат визначає n -місне відношення.

Предикат можна розглядати як функцію, що визначена на деякій множині об'єктів, та може приймати два значення, T і F , тобто як булеву функцію. Тому над предикатами визначені всі булеві операції: $\neg$ (заперечення), $\&$ (кон'юнкція), $\vee$ (диз'юнкція), $\rightarrow$ (імплікація), $\equiv$ (еквівалентність), а також дві нові операції – операції навішування кванторів: $\forall$ – загальності і $\exists$ – існування.

Якщо $P(x)$ визначає деяку властивість на множині M , то формула $\forall x P(x)$ позначає висловлювання: «для будь-якого предмета $x \in M$ властивість $P(x)$ виконується», або «всі x мають властивість $P(x)$ ». Значення формули $|\forall x P(x)| = T$ (істинно), якщо властивість P виконано для всіх об'єктів з M , і $|\forall x P(x)| = F$ (хибно), якщо існує хоча б один елемент $x=a$, $a \in M$, для якого властивість P не виконана, тобто $|P(a)|=F$.

Якщо $M = \{a_1, a_2, \dots, a_n\}$ - кінцева область визначення предиката $P(x)$, то формули з кванторами можуть бути виражені через кон'юнкцію і диз'юнкцію:

$$\forall x P(x) = P(a_1) \& P(a_2) \& \dots \& P(a_n), \quad \exists x P(x) = P(a_1) \vee P(a_2) \vee \dots \vee P(a_n).$$

Таким чином, квантор загальності є узагальненням кон'юнкції, а квантор існування – узагальненням диз'юнкції на нескінченній області визначення.

Квантори $\forall$ і $\exists$ пов'язані один з одним за принципом подвійності (по законам де Моргана):

$$\neg \forall x P(x) \equiv \exists x \neg P(x), \neg \exists x P(x) \equiv \forall x \neg P(x).$$

Основними символами мови логіки предикатів є:

- пропозиціональні символи $\neg$ і $\rightarrow$,
- квантори загальності $\forall$ і існування $\exists$,
- допоміжні символи (i) ,
- предметні змінні $x_1, x_2, \dots, x_n, \dots$,
- предметні постійні $a_1, a_2, \dots, a_n, \dots$,
- функціональні символи $f_1^1, f_1^2, \dots, f_k^j, \dots$,
- предикатні символи $P_1^1, P_1^2, \dots, P_k^j, \dots$

Нижній індекс предикатного або функціонального символу – це номер, який служить для розрізнення однойменних символів з однаковим числом аргументів, верхній індекс вказує число аргументів.

Визначимо поняття *терма* і *формули*.

Визначення терма.

1. Кожна предметна змінна є *терм*.
2. Кожна предметна постійна є *терм*.
3. Функціональний символ $f(t_1, \dots, t_n)$, де $t_1, \dots, t_n$ – терми, є терм.
4. Інших термів немає.

Визначення формули.

1. $P_i^n(t_1, \dots, t_n)$, де P_i^n – предикатний символ, $t_1, \dots, t_n$ – терми, є *атомарна (елементарна) формула*.
2. Якщо A і B – формули і x – предметна змінна, то формулами є $(\neg A)$, $(A \rightarrow B)$, $(\forall x A)$, $(\exists x A)$.
3. Інших формул немає.

Вирази $A \& B$, $A \vee B$, $A \equiv B$ визначаються так само, як в обчисленні L .

Пріоритет операцій повинен бути таким: $\neg$, $\forall$ і $\exists$, $\&$, $\vee$, $\rightarrow$, $\equiv$.

Формула, на яку поширюється дія квантора, називається *областю дії квантора*. Змінні, за якими навішується квантор і потрапляють в його область дії, називаються *пов'язаними змінними*. Змінні, що лежать поза областю дії квантора, називаються *вільними*.

Формула, що не містить вільних змінних, називається *замкненою*. Замкнені формули є висловлюваннями.

Область дії квантора обмежується дужками, якщо вона містить більше одного предиката.

На рис. 3.1 наведені приклади формул логіки предикатів і вказані вільні і зв'язані змінні.

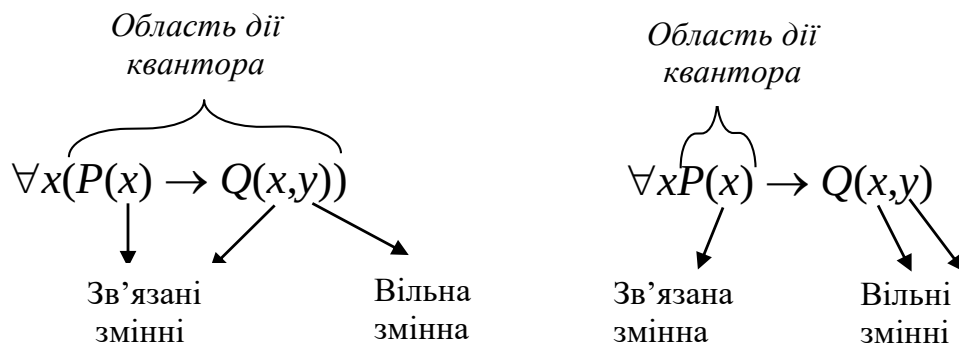


Рис. 3.1. Вільні і зв'язані змінні.

Завдання 3.1.1. Записати висловлювання у вигляді формул логіки предикатів.

Приклад.

Нехай $x \in \{\text{числа}\}$, на цій області визначення задані наступні предикати:

$N(x)$: x – натуральне число, $C(x)$: x – ціле число, $B(x)$: x – парне число, $D(x, y)$: x ділиться на y , $P(x)$: x – просте число, $G(x)$: x – додатне число.

- 1 Будь-яке натуральне число - ціле.

$$\forall x (N(x) \rightarrow C(x))$$

- 2 Будь-яке ціле число - або парне або ні

$$\forall x (C(x) \rightarrow B(x) \vee \neg B(x))$$

- 3 Для будь-якого цілого числа знайдеться інше ціле таке, що ділить перше.
 $\forall x (C(x) \rightarrow \exists y (C(y) \& D(x,y)))$
- 4 Існують прості парні числа.
 $\exists x (P(x) \& B(x))$
- 5 Будь-яке просте ціле число - натуральне.
 $\forall x (P(x) \& C(x) \rightarrow N(x))$ або* $\forall x (P(x) \rightarrow (C(x) \rightarrow N(x)))$
- 6 Всі парні числа діляться на два.
 $\forall x (B(x) \rightarrow D(x,2))$
- 7 Жодне натуральне число не є від'ємним.
 $\forall x (N(x) \rightarrow G(x))$ або** $\neg \exists x (N(x) \& \neg G(x))$
- 8 15 ділиться на 3 і 5.
 $D(15,5) \& D(15,3)$
- 9 Всяке натуральне число, що ділиться на 12, ділиться на 3 і 4.
 $\forall x (N(x) \& D(x,12) \rightarrow D(x,3) \& D(x,4))$ або
 $\forall x (N(x) \rightarrow (D(x,12) \rightarrow D(x,3) \& D(x,4)))$
- 10 Ділення на нуль - неможливо!
 $\forall x \neg D(x,0)$ або** $\neg \exists x D(x,0)$

Вказівка. 2,3,4,5,12,15 – константи з області визначення.

* впливає з еквівалентності представлень формул :

$$A \& B \rightarrow C = \neg (A \& B) \vee C = \neg A \vee \neg B \vee C = A \rightarrow (B \rightarrow C) .$$

**впливає з законів де Моргана загальних і для кванторів.

Предикати можна розглядати як функції, визначені на певній предметній області, що набувають значення Т або F. Тому до них можуть застосовуватися усі логічні операції, а також ще додаткові – навішування кванторів. Правила формалізації речень природної мови з предикатами наступні: якщо вживається «для всіх», «кожний», «будь-який», то навішується квантор загальності ($\forall$), якщо - «деякий», «знайдеться», то квантор існування

($\exists$). Якщо в предикатах речення є спільна змінна, пов'язана із квантором загальності, то між предикатами ставиться логічна зв'язка імплікація ($\rightarrow$), якщо існування - то кон'юнкція ($\&$). Для позначення окремих об'єктів з області визначення (Том - один зі студентів) прийнято використовувати константи. Речення з константами не мають вигляд законів (те, що повинне виконуватися для всіх), то, також як і у випадку, якщо мова йде про «деяких», зв'язкою між предикатами служить кон'юнкція.

Природне сполучення «і» формалізується логічною операцією кон'юнкцією ($\&$), «або» - диз'юнкцією ($\vee$). При оцінюванні рішення враховуються варіативність, яка спирається на переставні закони - комутативності та контрапозиції.

Нехай $x \in \{\text{люди}\}$, $y \in \{\text{речі, які можна читати і писати}\}$.

На цих областях визначення задані предикати:

$P(x)$: x – професор,

$S(x)$: x – студент,

$V(x)$: x – поет,

$R(x, y)$: x полюбляє читати y ,

$W(x, y)$: x пише y ,

$N(y)$: y – роман,

$K(y)$: y – конспект,

$C(y)$: y – вірші,

$U(y)$: y – підручник,

$L(y)$: y – лист,

$H(y)$: y – шпаргалка.

- 1) Деякі романи написані у віршах.
- 2) Жоден підручник не написаний у віршах.
- 3) Усі конспекти – підручники.
- 4) Деякі вірші – листи.

- 5) *«Евгеній Онегин» – це роман у віршах.*
- 6) *Ніхто не любить писати листи.*
- 7) *Деякі люди пишуть вірші.*
- 8) *Усі поети любляють читати вірші.*
- 9) *Усі студенти любляють читати підручники.*
- 10) *Усі студенти пишуть конспекти.*
- 11) *Деякі студенти пишуть тільки шпаргалки.*
- 12) *Ніхто зі студентів не пише підручники.*
- 13) *Деякі професори, а також студенти, пишуть вірші.*
- 14) *Деякі не люблять читати ніяких підручників.*
- 15) *Студент Том любить читати підручники.*
- 16) *Професор Джонс – поет.*
- 17) *Конспекти пишуть тільки студенти.*
- 18) *Кожен професор написав хоч один підручник.*
- 19) *Студенти не пишуть листів.*
- 20) *Кожен любить читати будь-які вірші.*

Завдання 3.1.2. Побудувати таблиці істинності на області інтерпретації $D=\{1,2\}$.

Приклад. $E = \forall x \exists y (P(x) \vee Q(y) \rightarrow R)$.

Предикат $P(x)$ та формули від нього з кванторами (для $Q(y)$ - так само) на області інтерпретації $D = \{1, 2\}$ набувають наступні значення:

x	P_1	P_2	P_3	P_4
1	F	F	T	T
2	F	T	F	T
$\exists x P(x)$	F	T	T	T
$\forall x P(x)$	F	F	F	T

Вказівка. R – замкнута формула, а саме висловлення, що набуває значення T і F .

Оскільки предикат $P(x)$ набуває 4 значення, предикат $Q(y)$ – 4 значення, формула R – 2 значення, і у формулі E немає вільних змінних, її таблиця істинності буде складатися з $4 \times 4 \times 2 = 32$ рядків. Очевидно, що якщо $|R| = T$, то $|E| = T$, тому залишається обчислити значення формули на тих 16 інтерпретаціях, що залишилися при $|R| = F$.

Нехай $|R| = F$. Розглянемо обчислення значення формули на інтерпретації $P = P_1$ і $Q = Q_2$.

$$\forall x \left\{ \begin{array}{l} x=1 \exists y \left\{ \begin{array}{l} y=1 P_1(1) \vee Q_2(1) \rightarrow R \\ y=2 P_1(1) \vee Q_2(2) \rightarrow R \end{array} \right. \\ x=2 \exists y \left\{ \begin{array}{l} y=1 P_1(2) \vee Q_2(1) \rightarrow R \\ y=2 P_1(2) \vee Q_2(2) \rightarrow R \end{array} \right. \end{array} \right. = \forall x \left\{ \begin{array}{l} x=1 \exists y \left\{ \begin{array}{l} F \vee F \rightarrow F = T \\ F \vee T \rightarrow F = F \end{array} \right. = T \\ x=2 \exists y \left\{ \begin{array}{l} F \vee F \rightarrow F = T \\ F \vee T \rightarrow F = F \end{array} \right. = T \end{array} \right. = T$$

Нехай $P = P_2$ и $Q = Q_1$.

Тоді

$$\forall x \left\{ \begin{array}{l} x=1 \exists y \left\{ \begin{array}{l} y=1 P_2(1) \vee Q_1(1) \rightarrow R \\ y=2 P_2(1) \vee Q_1(2) \rightarrow R \end{array} \right. \\ x=2 \exists y \left\{ \begin{array}{l} y=1 P_2(2) \vee Q_1(1) \rightarrow R \\ y=2 P_2(2) \vee Q_1(2) \rightarrow R \end{array} \right. \end{array} \right. = \forall x \left\{ \begin{array}{l} x=1 \exists y \left\{ \begin{array}{l} F \vee F \rightarrow F = T \\ F \vee F \rightarrow F = T \end{array} \right. = T \\ x=2 \exists y \left\{ \begin{array}{l} T \vee F \rightarrow F = F \\ T \vee F \rightarrow F = F \end{array} \right. = F \end{array} \right. = \forall x \left\{ \begin{array}{l} T \\ F \end{array} \right. = F$$

Аналогічно знаходяться інші значення формули E , які наведені нижче у таблиці.

P	P ₁	P ₁	P ₁	P ₁	P ₂	P ₂	P ₂	P ₂	P ₃	P ₃	P ₃	P ₃	P ₄	P ₄	P ₄	P ₄
Q	Q ₁	Q ₂	Q ₃	Q ₄	Q ₁	Q ₂	Q ₃	Q ₄	Q ₁	Q ₂	Q ₃	Q ₄	Q ₁	Q ₂	Q ₃	Q ₄
E	T	T	T	F	F	F	F	F	F	F	F	F	F	F	F	F

Завдання.

- 1) $\exists x \forall y (P(x) \vee Q(y) \rightarrow R)$;
- 2) $\forall x (\exists y (R \rightarrow P(x) \vee Q(y)))$;
- 3) $\exists x (R \rightarrow \forall y (P(x) \rightarrow Q(y)))$;
- 4) $\forall x (R \rightarrow \forall y (Q(y) \rightarrow P(x)))$;
- 5) $\exists x (P(x) \rightarrow \exists y (R \rightarrow Q(y)))$;
- 6) $\exists x (P(x) \rightarrow \forall y (Q(y) \rightarrow P(x)))$;
- 7) $\exists x (P(x) \& \exists y (Q(y) \rightarrow R))$;
- 8) $\forall x (P(x) \vee \exists y (Q(y) \rightarrow R) \rightarrow S)$;
- 9) $\exists x (P(x) \& \forall y Q(y)) \rightarrow \forall x P(x)$;
- 10) $\exists x \forall y (P(x) \vee Q(y) \rightarrow Q(y))$.

Завдання 3.1.3. Перевірити логічну загальнозначущість наступних формул.

Вказівка. Формула логічно загальнозначуща, якщо вона набуває істинне значення на всіх можливих інтерпретаціях. Є достатнім знайти хоча б одну інтерпретацію, на якій формула набуває хибне значення, щоб довести, що вона не загальнозначуща. Якщо такої інтерпретації не вдається знайти, та в процесі пошуку ми приходимо до протиріччя, то можна зробити висновок, що формула є логічно загальнозначущою.

Приклад 1. Дана формула: $\forall x(P(x) \rightarrow Q(x)) \vee \forall x \neg Q(x)$.

Для перевірки загальнозначущості використовуємо метод знаходження контрприкладу. Допустимо, що формула $\forall x(P(x) \rightarrow Q(x)) \vee \forall x \neg Q(x)$ не є логічно загальнозначущою. Тоді існує така множина M , і такі інтерпретації $P^*(x)$, $Q^*(x)$ предикатів $P(x)$, $Q(x)$ відповідно, що $|\forall x(P^*(x) \rightarrow Q^*(x)) \vee \forall x \neg Q^*(x)| = F$. Звідси випливає, що

$$|\forall x(P^*(x) \rightarrow Q^*(x))| = F \quad (1)$$

$$|\forall x \neg Q^*(x)| = F \quad (2)$$

Формула (2) набуває хибне значення, якщо на області інтерпретації існує хоча б одне значення $a \in M$, при якому $|\neg Q^*(a)| = F$, а саме $|Q^*(a)| = T$. Для того, щоб формула (1) набула значення F , є достатнім, щоб існувало хоча б одне значення x , наприклад, $x = c$, таке, що $|P^*(c)| = T$, а $|Q^*(c)| = F$ (c не обов'язково співпадає з a). Отже, ми знайшли контрприклад: є достатнім на деякій n -елементній множині M ($n \geq 2$) знайти таку інтерпретацію, при якій $|P^*(c)| = T$, $|Q^*(c)| = F$ и $|Q^*(a)| = T$.

Таким чином, формула $\forall x(P(x) \rightarrow Q(x)) \vee \neg Q(x)$ не є логічно загальнозначущою.

Приклад 2. $\exists x(P(x) \vee Q(x)) \rightarrow \exists x P(x) \vee \exists x Q(x)$.

Припустимо, що існує множина M і інтерпретація на ній $P^*(x)$, $Q^*(x)$, така, що формула на цій інтерпретації приймає значення F :

$|\exists x(P^*(x) \vee Q^*(x)) \rightarrow \exists xP^*(x) \vee \exists xQ^*(x)| = F$. Тоді:

$$|\exists xP^*(x) \vee \exists xQ^*(x)| = F, \quad (1)$$

$$|\exists x(P^*(x) \vee Q^*(x))| = T. \quad (2).$$

Формула (1) хибна, якщо $|\exists xP^*(x)| = F$ и $|\exists xQ^*(x)| = F$, а звідси витікає, що

$$|P^*(x)| = F \text{ для кожного } x \in M \quad (3)$$

$$\text{і } |Q^*(x)| = F \text{ для кожного } x \in M. \quad (4)$$

З (2) випливає, що існує таке $a \in M$, що $|P^*(a) \vee Q^*(a)| = T$. Можливі три варіанти: а) $|P^*(a)| = T$, $|Q^*(a)| = T$; б) $|P^*(a)| = T$, $|Q^*(a)| = F$; в) $|P^*(a)| = F$, $|Q^*(a)| = T$.

Варіант а) знаходиться у протиріччі з умовами (3), (4). Однак, це протиріччя ще не доводить загальнозначущість формули – слід перевірити усі варіанти.

Варіант б). Якщо $|P^*(a)| = T$, то це є протиріччям до умов (3).

Варіант в). Якщо $|Q^*(a)| = T$, то це є протиріччям до умов (4).

Отже, не існує такої інтерпретації, на якій формула $\exists x(P(x) \vee Q(x)) \rightarrow \exists xP(x) \vee \exists xQ(x)$ набуває хибне значення, а саме, вона логічно загальнозначуща.

$$1) \forall x(P(x) \& Q(x)) \equiv \forall xP(x) \& \forall xQ(x);$$

$$2) \exists x(P(x) \vee Q(x)) \equiv \exists xP(x) \vee \exists xQ(x);$$

$$3) \forall xP(x) \vee \forall xQ(x) \rightarrow \forall x(P(x) \vee Q(x));$$

$$4) \forall x(P(x) \vee Q(x)) \rightarrow \forall xP(x) \vee \forall xQ(x);$$

$$5) \exists x(P(x) \& Q(x)) \rightarrow \exists xP(x) \& \exists xQ(x);$$

$$6) \exists xP(x) \& \exists xQ(x) \rightarrow \exists x(P(x) \& Q(x));$$

$$7) (\exists xP(x) \rightarrow \exists xQ(x)) \rightarrow \exists x(P(x) \rightarrow Q(x));$$

$$8) \exists x(P(x) \rightarrow Q(x)) \rightarrow (\exists xP(x) \rightarrow \exists xQ(x));$$

$$9) \forall x(P(x) \equiv Q(x)) \rightarrow (\forall xP(x) \equiv \forall xQ(x));$$

$$10) (\exists x P(x) \equiv \exists x Q(x)) \rightarrow \exists x (P(x) \equiv Q(x));$$

$$11) (\forall x P(x) \rightarrow \forall x Q(x)) \rightarrow \forall x (P(x) \rightarrow Q(x));$$

$$12) \forall x (P(x) \rightarrow Q(x)) \rightarrow (\forall x P(x) \rightarrow \forall x Q(x)).$$

3.2. Доведення логічного слідування в теорії предикатів 1-го порядку

Теоретичні відомості

Основні правила виведення логіки предикатів

1. Правило універсальної конкретизації (УК):

$$\forall x A(x) \mid = A(y), \text{ якщо } y \text{ вільно для } x \text{ в } A(x).$$

2. Правило екзистенціальної конкретизації (ЕК):

$$\exists x A(x) \mid = A(b), \text{ де } b \in D.$$

3. Правило екзистенціального узагальнення (ЕУ):

$$A(y) \mid = \exists x A(x), \text{ де } x \text{ вільно для } y \text{ в } A(y).$$

4. Правило загальності:

$$C \rightarrow A(x) \mid = C \rightarrow \forall x (A(x)).$$

5. Правило існування:

$$A(x) \rightarrow C \mid = \exists x A(x) \rightarrow C.$$

6. Правило узагальнення *Gen* (від англійського слова Generalization):

Якщо $\Gamma \mid = A(x)$, то $\Gamma \mid = \forall x A(x)$, якщо x не входить вільно ні в одну з формул Γ .

Обчислення предикатів, яке не містить предметних констант, функціональних символів, предикатів і власних аксіом, називається чистим. Обчислення предикатів, що містить предметні константи та/або функціональні символи та/або предикатні символи і власні аксіоми, що зв'язують їх, називаються прикладними. Обчислення предикатів, в якому квантори можуть пов'язувати тільки предметні змінні, але не можуть пов'язувати функціональні символи або предикати, називається обчисленням першого порядку.

Символами **теорії K** служать ті ж самі символи логіки предикатів: пропозиціональні зв'язки $\rightarrow, \neg, \forall, \exists$, допоміжні символи $(,)$, множини предметних змінних: $x_1, x_2, \dots$, предметних постійних: $a_1, a_2, \dots$, функціональні символи: $f_i^n, i=1, \dots, k, n=0, \dots, m$, предикатні символи: $P_i^n, i=1, \dots, k, n=0, \dots, m$. Визначення терма, формули і пропозиціональних зв'язок $\&, \vee, \equiv$ залишаються в силі для теорії першого порядку.

Аксіоми теорії K розбиваються на *логічні аксіоми* і *власні*.

Логічні аксіоми. Хоч би якими були формули A, B, C теорії K , такі формули є логічними аксіомами теорії K .

$$A1 \ A \rightarrow (B \rightarrow A).$$

$$A2 \ (A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C)).$$

$$A3 \ (\neg B \rightarrow \neg A) \rightarrow ((\neg B \rightarrow A) \rightarrow B).$$

$$A4 \ \forall x \ A(x) \rightarrow A(y), \text{ якщо } y \text{ вільно для } x \text{ у формулі } A(x).$$

$$A5 \ \forall x (A \rightarrow B(x)) \rightarrow (A \rightarrow \forall x \ B(x)), \text{ якщо } A \text{ не містить вільних входжень } x.$$

Власні аксіоми формулюються окремо для кожної конкретної змістовної предметної області.

Теорія першого порядку, що не містить власних аксіом, називається *обчисленням предикатів першого порядку*.

Правилами виведення у всякій теорії першого порядку є:

1) *modus ponens* (MP): з A і $A \rightarrow B$ слідує B ,

2) правило *узагальнення* Gen: з $\Gamma \vdash A(x)$, слідує $\Gamma \vdash \forall x A(x)$, якщо x не входить вільно ні в одну з формул Γ .

Метатеореми про дедукції. Якщо існує висновок формули B з множини гіпотез Γ і формули A , і в цьому висновку ні при якому застосуванні правила Gen до формул, що залежать від A , не зв'язується квантором ніяка вільна змінна формули A , то $\Gamma \vdash A \rightarrow B$.

Формалізація речень природної мови

Мова логіки предикатів традиційно служить для формалізації висловлювань природної мови. Вважається, що «мова предикатів» є вродженою і лежить в основі засвоєння рідної мови і форм пізнавальної активності.

Загальні судження висловлюють включення (або невключення) класу предметів в інший клас, наприклад: «Всі судді - юристи».

Приватні судження висловлюють часткову приналежність (чи неналежність) предметів одного класу до іншого класу, наприклад, «Деякі вчені є викладачами».

Суб'єкт і предикат називаються термінами суджень. Зв'язка в судженні відображає зв'язок, який існує між предметом думки (суб'єктом) і відповідним властивістю. Вона вказує на наявність або відсутність у предмета судження властивості, ознаки, про яку йде мова в предикаті.

Залежно від обсягу суб'єкта судження діляться на загальні (всі) і часткові (деякі), в тому числі і поодинокі. Загальне судження - судження, в якому по кожному мислимому суб'єкту елемента множини стверджується або заперечується певна ознака. Часткове судження - судження, в якому міститься знання про наявність або відсутність ознаки у частини предметів мисливих в суб'єкті, а про наявність або відсутність цієї ознаки для інших предметів, ми можемо нічого не знати (одиничне судження - окремий випадок часткового, судження про суб'єкт, у якого є єдине поняття, наприклад, *Київ - столиця України*, але в логіці, поодинокі судження особливим чином не виділяються).

За якістю, тобто за характером зв'язки, судження можна розділити на позитивні і негативні. У стверджувальних - обсяг суб'єкта включається до обсягу предиката. Тобто, у судженнях констатується наявність ознаки у певного предмета або множини. А в негативних - обсяг виключається, у судженнях констатується відсутність ознаки у предметів, мисливих в суб'єкті судження. В цілому, негативні судження несуть менше інформації в порівнянні зі стверджувальними. Тому в логіці частіше зустрічаються

судження, що відповідають правилу визначення понять, згідно яким визначення повинні бути стверджувальними.

Якщо за основу поділу суджень на види брати і кількісну і якісну характеристику, то можна виділити чотири основні схеми.

Загально стреджувальні - наявність ознаки для кожного предмета. Всі S суть P : $\forall x (S(x) \rightarrow P(x))$.

Загально негативне - констатується відсутність ознаки в кожному предметі, що мислиться в суб'єкті судження.

Жодне S не є P : $\forall x (S(x) \rightarrow \neg P(x))$.

Частково стреджувальне - наявність ознаки для частини предметів. Жодне S не є P : $\forall x (S(x) \rightarrow \neg P(x))$.

Частково негативне - відсутність ознаки для частини предметів. Деякі S не є P - $\exists x (S(x) \& \neg P(x))$.

В даному розділі розглянемо два способи доказів логічних слідувань: неформальний спосіб, заснований на доказі від супротивного (метод редукції), та формальний метод в численні предикатів (формальна теорія **K**). Надалі буде розглянуто більш ефективний метод, що дозволяє автоматизувати процес доказів логічного слідування (логічної загальнозначущості) – метод резолюції та його реалізація в логічному програмуванні.

Завдання 3.2.1. Перевірити логічне слідування в логіці предикатів 1-го порядку.

Приклад. Область визначення - люди коледжу. 1. Усі випускники Ітона в коледжі грають в крикет. 2. Ніхто, крім викладачів, не обідає за верхнім столом. 3. Жоден з тих, хто грає в крикет, не вміє веслувати. 4. Всі мої друзі в цьому коледжі - випускники Ітона. 5. Всі викладачі – чудові веслувальники. Отже, жоден мій друг не обідає за верхнім столом.

Процес доведення логічного слідування починається з формалізації поставленого завдання засобами логіки предикатів 1-го порядку.

Введемо предикати на заданій області визначення: $V(x)$ - x випускник Ітона; $K(x)$ - x грає в крикет; $P(x)$ - x викладач; $O(x)$ - x обідає за верхнім столом; $G(x)$ - x вміє веслувати; $D(x)$ - x мій друг.

Формалізуємо посилки: 1. $\forall x (V(x) \rightarrow K(x))$; 2. $\forall x (O(x) \rightarrow P(x))$; 3. $\forall x (K(x) \rightarrow \neg G(x))$; 4. $\forall x (D(x) \rightarrow V(x))$; 5. $\forall x (P(x) \rightarrow G(x))$; і висновок, який треба перевірити: $\forall x (D(x) \rightarrow \neg O(x))$.

Доказ логічного слідування можна провести методом від протилежного (метод редукції) або за допомогою побудови формального виводу в теорії K - теорії числення предикатів першого порядку.

Доказ **методом редукції** полягає в наступному:

- передбачається існування інтерпретації, на якій всі посилки (гіпотези) набувають істинне значення, а слідування - хибне;
- знаходяться значення кожного простого висловлювання, що входить до посилок і висновка;
- при отриманні суперечності доводиться відсутність такої інтерпретації.

Припустимо, що всі посилки істинні, а висновок хибний. У такому випадку справедливий наступний запис:

1. $|\forall x (V(x) \rightarrow K(x))| = T$;
2. $|\forall x (O(x) \rightarrow P(x))| = T$;
3. $|\forall x (K(x) \rightarrow \neg G(x))| = T$;
4. $|\forall x (D(x) \rightarrow V(x))| = T$;
5. $|\forall x (P(x) \rightarrow G(x))| = T$;
- В. $|\forall x (D(x) \rightarrow \neg O(x))| = F$.

Тоді, якщо висновок хибний, то існує інтерпретація, на якій це висловлювання набуває помилкове значення.

Нехай існує така константа a з області визначення, що справедливо наступне:

$$|D(a) \rightarrow \neg O(a)|=F \Rightarrow |D(a)|=T, |O(a)|=T.$$

Оскільки посилки правильні для всіх x з області визначення, то вони також правильні і для константи a з цієї ж області визначення, тому справедлива заміна x на a .

Після підстановки в посилку 2 вже відомих значень висловлювань отримаємо:

$$|T \rightarrow P(a)|=T \Rightarrow |P(a)|=T$$

Після підстановки в посилку 4 вже відомих значень висловлювань отримаємо:

$$|T \rightarrow V(a)|=T \Rightarrow |V(a)|=T$$

Після підстановки в посилку 1 вже відомих значень висловлювань отримаємо:

$$|T \rightarrow K(a)|=T \Rightarrow |K(a)|=T$$

Після підстановки в посилку 3 вже відомих значень висловлювань отримаємо:

$$|T \rightarrow \neg G(a)|=T \Rightarrow |G(a)|=F$$

Після підстановки в посилку 5 вже відомих значень висловлювань отримаємо: $|T \rightarrow G(a)|=T \Rightarrow |G(a)|=T \neq F$.

Отримане протиріччя доводить неправильність припущення, з чого випливає, що висновок істинний, а не хибний.

При побудові формального виводу для перевірки логічного слідування в численні предикатів першого порядку використовуються наступні правила:

Правило універсальної конкретизації (УК) -

$$\forall x A(x) \Rightarrow A(y), \text{ якщо } y \text{ вільно для } x \text{ в } A(x).$$

Правило узагальнення (Gen) -

Якщо $\Gamma \vdash A(x)$, то $\Gamma \vdash \forall x A(x)$, якщо x не входить вільно ні в одну з формул Γ .

Також в теорії К числення предикатів першого порядку використовуються правило виведення МР і теореми теорії L, які включені в множину теорем теорії К (див. Розділ 2).

Формальний висновок в теорії К

1. $\forall x (V(x) \rightarrow K(x))$ Г1
2. $\forall x (O(x) \rightarrow P(x))$ Г2
3. $\forall x (K(x) \rightarrow \neg G(x))$ Г3
4. $\forall x (D(x) \rightarrow V(x))$ Г4
5. $\forall x (P(x) \rightarrow G(x))$ Г5
6. $V(y) \rightarrow K(y)$ УК(1)
7. $O(y) \rightarrow P(y)$ УК(2)
8. $K(y) \rightarrow \neg G(y)$ УК(3)
9. $D(y) \rightarrow V(y)$ УК(4)
10. $P(y) \rightarrow G(y)$ УК(5)
11. $V(y) \rightarrow \neg G(y)$ ПС(6, 8)
12. $\neg G(y) \rightarrow \neg P(y)$ ПК(10)
13. $V(y) \rightarrow \neg P(y)$ ПС(11, 12)
14. $D(y) \rightarrow \neg P(y)$ ПС(9, 13)
15. $\neg P(y) \rightarrow \neg O(y)$ ПК(7)
16. $D(y) \rightarrow \neg O(y)$ ПС(14, 15)
17. $\forall x (D(x) \rightarrow \neg O(x))$ Gen(16)

1) *Область визначення: люди і твори.* 1. Жоден студент $S(x)$ не пише ($W(x,y)$ – x пише y) листи $L(y)$. 2. Деякі студенти мають написані конспекти $K(y)$. Отже, деякі конспекти – це не листи.

2) *Область визначення: люди.* 1. Кожен, хто сильний і розумний, досягне успіху в кар'єрі. 2. Джон - сильний. 3. Тільки розумні люди навчаються в університетах. 4. Джон - студент університету. *Висновок: Джон досягне успіху в кар'єрі.*

3) Область визначення: люди (x) і книжки (y). 1. Деякі студенти ($P(x)$) **не люблять читати** ($L(x,y)$) жодних підручників ($D(x)$). 2. Всі студенти люблять читати журнали ($Q(x)$). *Висновок: підручники - це не журнали.*

4) Область визначення: небісні тіла. 1. Кожне небесне тіло є або зіркою, або кометою, або планетою. 2. Венера - небесне тіло, що не є зіркою. 3. У комет, розташованих недалеко від сонця, є хвости. 4. Венера недалеко від Сонця, але у неї немає хвоста. *Висновок: Венера - планета.*

5) Область визначення: тварини. 1. Тварини, які не брикаються ($K(x) - x$ не брикається), завжди незворушні ($N(x) - x$ незворушливий). 2. У віслюка ($O(x) - x$ віслук) немає рогів ($R(x) - x$ не має рогів). 3. Буйвол ($B(x) - x$ буйвол) завжди може перекинути вас через огорожу ($G(x) - x$ може перекинути вас через огорожу). 4. Тварин, які брикаються, не легко проковтнути ($P(x) - x$ не легко проковтнути). 5. Тварина, яка не має рогів, не може перекинути вас через огорожу. 6. Всі тварини, **крім** буйвола, легко переходять у лють (не є незворушливими) = незворушливий – тільки буйвол. *Висновок: Віслюка проковтнути не легко.*

6) Область визначення: тварини. 1. **Єдині** тварини у цьому будинку ($H(x) - x$ в цьому будинку) –кішки ($C(x) - x$ кішка). 2. Будь-яку тварину можна приручити ($D(x) - x$ можна приручити), якщо вона любить дивитися на місяць ($L(x) - x$ любить дивитися на місяць). 3. Якщо тварина викликає у мене огиду ($O(x) - x$ викликає у мене огиду), я намагаюся триматися від неї подалі ($F(x) - x$ я тримаюся подалі від x). 4. Жодна тварина не плотоядна ($T(x) - x$ не плотоядна), якщо вона не бродить ночами ($N(x) - x$ не бродить ночами). 5. Жодна кішка не пропустить нагоди зловити мишу ($M(x) - x$ ловить мишей). 6. Я не пускаю до себе до кабінету тварин ($R(x) - x$ я не пускаю x до кабінету), **крім тих**, що знаходяться у цьому будинку. 7. Кенгуру ($k-const$) не піддається прирученню. 8. **Лише** плотоядні тварини ловлять мишей. 9. Тварини, яких я не пускаю до себе в кабінет, викликають у мене огиду. 10. Тварини, які блукають ночами, люблять дивитися на місяць. *Висновок: Я завжди намагаюся триматися подалі від кенгуру.*

7) *Область визначення: люди.* 1. Святий Франциск любить кожного, хто любить когось. 2. Кожен любить когось. *Висновок: Святий Франциск любить кожного* ($L(x)$ - x любить когось; $F(x)$ - святий Франциск любить x)

3.3. Особливості застосування методу резолюцій в теорії предикатів 1-го порядку

Завдання 3.3.1. Довести логічне слідування методом резолюцій.

Приклад. Доведемо виконуваність логічного слідування для завдання з попереднього розділу методом автоматичного доведення теорем - методом резолюцій.

Основна ідея застосування методу резолюцій для перевірки логічного слідування полягає в тому, щоб перевірити, чи містить множина S (поле знань задачі) пустий диз'юнкт $\square$. Якщо S його містить, то множина S нездійсненна, якщо ні, то треба перевірити чи може він бути отриманий з даної множини диз'юнктив. Іншими словами, необхідно знайти множину основних прикладів, що спростовують вихідну множину диз'юнктив. Ця процедура ґрунтується на правилі резолюції.

Правило резолюції Робінсона: для будь-яких двох диз'юнктив C^1 і C^2 якщо існує літера $L^1 \in C^1$ і контрарна їй літера $L^2 \in C^2$ ($L^2 = \neg L^1$), то, викреслюючи L^1 з C^1 і L^2 з C^2 і побудувавши диз'юнкт з решти літер, отримаємо резольвенту C^1 і C^2 ($C^1 \vee C^2$).

Резольвента C є логічне слідування C^1 и C^2 , що містять контрарні літери L и $\neg L$.

Застосування методу резолюції в логіці першого порядку ускладнюється тим, що диз'юнкти містять змінні, які можуть не збігатися в двох однакових літерах. У таких випадках здійснюються підстановки, заміни та уніфікації.

Складемо спочатку множину диз'юнктив даного логічного слідування, привівши посилки і заперечення висновку до упередженої нормальної форми (УНФ) і, потім, до стандартної скулемовської форми (ССФ):

1. $\forall x (V(x) \rightarrow K(x))$
 $\forall x (\neg V(x) \vee K(x))$
 $\neg V(x) \vee K(x)$
2. $\forall x (O(x) \rightarrow P(x))$
 $\forall x (\neg O(x) \vee P(x))$
 $\neg P(x) \vee O(x)$
3. $\forall x (K(x) \rightarrow \neg G(x))$
 $\forall x (\neg K(x) \vee \neg G(x))$
 $\neg K(x) \vee \neg G(x)$
4. $\forall x (D(x) \rightarrow V(x))$
 $\forall x (\neg D(x) \vee V(x))$
 $\neg D(x) \vee V(x)$
5. $\forall x (P(x) \rightarrow G(x))$
 $\forall x (\neg P(x) \vee G(x))$
 $\neg P(x) \vee G(x)$
- В. $\neg(\forall x(D(x) \rightarrow \neg O(x)))$
 $\exists x \neg(\neg D(x) \vee \neg O(x))$
 $\exists x (D(x) \& O(x))$
 $D(a)$
 $O(a)$

Отримана множина диз'юнктив: $\{\neg V(x) \vee K(x), \neg P(x) \vee O(x), \neg K(x) \vee \neg G(x), \neg D(x) \vee V(x), \neg P(x) \vee G(x), D(a), O(a)\}$

Тепер побудуємо висновок, послідовно застосовуючи правило резолюцій:

1. $D(a)$
2. $O(a)$

3. $\neg V(x) \vee K(x)$
4. $\neg O(x) \vee P(x)$
5. $\neg K(x) \vee \neg G(x)$
6. $\neg D(x) \vee V(x)$
7. $\neg P(x) \vee G(x)$
8. $V(a) \{a/x\}$ рез. (1, 6)
9. $K(a) \{a/x\}$ рез. (1, 8)
10. $\neg G(a) \{a/x\}$ рез. (5, 9)
11. $\neg P(a) \{a/x\}$ рез. (7, 10)
12. $\neg O(a) \{a/x\}$ рез. (4, 11)
13. $\square$ рез. (2, 12)

Пустий диз'юнкт отриманий, отже, висновок - істинний.

Умови завдань для формалізації і доведення – завдання 3.2.1.

3.4. Логічне програмування

Завдання 3.4.1. Скласти логічну програму і перевірити логічне слідування.

Приклад. Побудова логічної програми включає в себе опис задачі на мові клауз і побудову дерева логічного висновку.

Логічна програма складається із сукупності фактів, які утворюють базу знань. Завдання формулюється як питання про деякі факти або відносини, визначені у базі знань. Розв'язання завдання - це перевірка логічного слідування (з фактів і правил, визначених у базі знань). Виведення логічного слідування здійснюється методом резолюцій, який реалізується виконанням самої програми. Сам метод резолюцій прихований від користувача.

Логічна програма включає в себе перелік предикатів (після слова predicates), що вводяться при формалізації; сукупність фактів (що записуються після слова clauses) і правил (після фактів). Вирази, що описують правила, називаються

«клозами» або «клаузами» (clause). Їх синтаксис відрізняється від традиційного синтаксису числення предикатів. Клауза є різновидом запису диз'юнкта, в якій позитивні та негативні літери розділені символами :-, причому, позитивні літери пишуться зліва, а негативні - справа.

Перевірка логічного слідування відбувається у вигляді запиту до бази знань і має синтаксис: Goal? - <вираз>. Знак «-» перед виразом в питанні означає, що при пошуку відповіді від цього виразу береться заперечення, а потім виконується резолютивний висновок на множині фактів, правил і заперечення виразу, що записаною в питанні. Якщо ця множина виявиться суперечливою, тобто буде отриманий пустий диз'юнкт, то логічне слідування виконано, і програма відповість «так» (yes), або програма видасть результат уніфікацій (підстановки), який привів до отримання $\square$;

в іншому випадку вона відповість «ні» (no).

Запишемо множину диз'юнктів завдання з розділу 4.1 у вигляді клауз.

Predicates: $V(x)$, $K(x)$, $P(x)$, $O(x)$, $G(x)$, $D(x)$.

Clauses:

$D(a)$

$K(x) :- V(x)$

$P(x) :- O(x)$

$:- K(x), G(x)$

$V(x) :- D(x)$

$G(x) :- P(x)$

Goal ? – not $O(a)$

Дерево виконання логічної програми представлено нижче.

Ініціюється робота з заперечення виразу, що знаходиться у розділі Goal. Потім проглядається список клауз і вибирається перша зустрінена клауза або факт, що містять літери, контрарні до наявної. Після уніфікації, якщо вона є необхідною, отримуємо резольвенту, яка тепер задає пошук контрарної літери

в списку клауз. Процес продовжується до отримання пустого диз'юнкту або до вичерпання можливого перебору.

$$\begin{array}{c}
 O(a) \\
 | \\
 P(x):- O(x) \\
 \{a/x\} \\
 | \\
 P(a) \\
 | \\
 G(x):- P(x) \\
 \{a/x\} \\
 | \\
 G(a):- \\
 | \\
 :- K(x), G(x) \\
 \{a/x\} \\
 | \\
 :- K(a) \\
 | \\
 K(x):- V(x) \\
 \{a/x\} \\
 | \\
 :- V(a) \\
 | \\
 V(x):- D(x) \\
 \{a/x\} \\
 |
 \end{array}$$

$\therefore D(a)$

|

$D(a)$

|

□

У даному прикладі отримано пустий диз'юнкт, отже, множина диз'юнктів, що складена з посилок і заперечення висновку, є суперечливою. А це означає, що логічне слідування виконується.

Для складення логічної програми продовжуємо використовувати умови завдань 3.2.1. і визначену множину диз'юнктів в завданні 3.3.1.

Відповіді до розділу 3.

3.1.1.

- 1) $\exists y(C(y) \& N(y));$
- 2) $\forall y(U(y) \rightarrow \neg C(y));$
- 3) $\forall y(K(y) \rightarrow U(y));$
- 4) $\exists x(C(x) \& L(y));$
- 5) $N(e) \& C(e);$
- 6) $\forall x \forall y(L(y) \rightarrow \neg W(x,y));$
- 7) $\exists x \forall y(C(y) \rightarrow W(x,y));$
- 8) $\forall x(V(x) \rightarrow \forall y(C(y) \rightarrow R(x,y));$
- 9) $\forall x(S(x) \rightarrow \forall y(U(y) \rightarrow R(x,y));$
- 10) $\forall x(S(x) \rightarrow \forall y(K(y) \rightarrow W(x,y));$
- 11) $\exists x(S(x) \& \forall y(W(x,y) \rightarrow H(y));$
- 12) $\forall x(S(x) \rightarrow \forall y(U(y) \rightarrow \neg W(x,y));$
- 13) $\exists x((S(x) \vee P(x)) \& \forall y(C(y) \rightarrow W(x,y));$
- 14) $\exists x \forall y(U(y) \rightarrow \neg R(x,y));$
- 15) $S(t) \& \forall y(U(y) \rightarrow R(x,y));$
- 16) $P(d) \& V(d);$
- 17) $\forall y(K(x) \rightarrow \forall x(S(x) \rightarrow W(x,y));$
- 18) $\forall x(P(x) \rightarrow \exists y(U(y) \& W(x,y)));$
- 19) $\forall x(S(x) \rightarrow \forall y(L(y) \rightarrow \neg W(x,y));$
- 20) $\forall x \exists y(C(y) \& R(x,y)).$

3.1.2.

P	P_1	P_1	P_1	P_1	P_2	P_2	P_2	P_2	P_3	P_3	P_3	P_3	P_4	P_4	P_4	P_4
Q	Q_1	Q_2	Q_3	Q_4	Q_1	Q_2	Q_3	Q_4	Q_1	Q_2	Q_3	Q_4	Q_1	Q_2	Q_3	Q_4
1	при значенні R=F															
	T	F	F	F	T	F	F	F	T	F	F	F	F	F	F	F
2	при значенні R=T															
	F	T	T	T	F	T	T	T	F	T	T	T	T	T	T	T
3	при значенні R=T															
	T	T	T	T	T	T	T	T	T	T	T	T	F	F	F	T
4	при значенні R=T															
	T	F	F	F	T	F	F	F	T	F	F	F	T	T	T	T
5	при значенні R=T															
	T	T	T	T	T	T	T	T	T	T	T	T	F	T	T	T
6	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T
7	при значенні R=F															
	F	F	F	F	T	T	T	F	T	T	T	F	T	T	T	F
7	при значенні R=T															
	F	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T
8	при значеннях S=F та R=F															
	F	F	F	T	F	F	F	F	F	F	F	F	F	F	F	F
9	T	T	T	T	T	T	T	F	T	T	T	F	T	T	T	T
10	T	T	T	T	T	T	T	T	T	T	T	T	F	F	F	T

3.1.3.

4), 6), 8), 10, 11) - формула не логічно загальнозначуща.

3.2.1, 3.3.1, 3.4.1

1)

Область визначення: люди і твори

1. Жоден студент $S(x)$ не пише ($W(x,y)$ – x пише y) листи $L(y)$. 2. Деякі студенти мають написані конспекти $K(y)$. *Отже, деякі конспекти – це не листи.*

1) $\forall x(S(x) \rightarrow \forall y(L(y) \rightarrow \neg W(x, y)))$;

2) $\exists x(S(x) \& \exists y(W(x,y) \& K(y)))$;

Висновок: $\exists y(K(y) \& \neg L(y))$

3) $S(z) \rightarrow \forall y(L(y) \rightarrow \neg W(z, y))$ UK1

4) $S(a) \& \exists y(W(a,y) \& K(y))$ EK2

5) $S(a)$ видалення &4

6) $\exists y(W(a,y) \& K(y))$ видалення &4

7) $W(a,b) \& K(b)$ EK6

8) $\forall y(L(y) \rightarrow \neg W(a, y))$ GMP 5, 3

9) $L(t) \rightarrow \neg W(a, t)$ UK8

10) $W(a,b)$ видалення &7

11) $K(b)$ видалення &7

12) $\neg L(b)$ GMT 9, 10

13) $K(b) \& \neg L(b)$ введення & 11, 12

14) $\exists y(K(y) \& \neg L(y))$ **EY 13**

2)

Область визначення: люди. 1. Кожен, хто сильний ($P(x)$) і розумний ($R(x)$), досягне успіху в кар'єрі ($S(x)$). 2. Джон (j) - сильний. 3. Тільки розумні люди навчаються в університетах ($U(x)$). 4. Джон - студент університету. **Висновок:** *Джон досягне успіху в кар'єрі ($S(j)$).*

1. $\forall x(P(x) \& R(x) \rightarrow S(x))$ r1

2. $P(j)$ r2

3. $\forall x(U(x) \rightarrow R(x))$	г3
4. $U(j)$	г4
5. $P(y) \& R(y) \rightarrow S(y)$	УК1
6. $U(y) \rightarrow R(y)$	УК3
7. $R(j)$	GMP 4,6
8. $P(j) \& R(j)$	& 2,7
9. $S(j)$	GMP 5,8

3)

Область визначення: люди (x) і книжки (y). 1. Деякі студенти (P(x)) не люблять читати (L(x,y)) жодних підручників (D(x)). 2. Всі студенти люблять читати журнали (Q(x)). Висновок: підручники - це не журнали.

$$\Gamma_1: \exists x(P(x) \& \forall y(D(y) \rightarrow L(x, y)))$$

$$\Gamma_2: \forall x(P(x) \rightarrow \forall y(Q(y) \rightarrow \neg L(x, y)))$$

$$\text{Висновок } G: \forall y(D(y) \rightarrow \neg Q(y))$$

Для доведення логічного слідування можна застосувати метод від супротивного, формальне виведення і метод резолюцій.

Метод від супротивного.

Припустимо, що при істинності посилок $|F_1| = T$, $|F_2| = T$ висновок приймає хибне значення: $|G| = F$.

З $| \forall y(D(y) \rightarrow \neg Q(y)) | = F$ слідує, що існує таке $y = a$, що $|D(a) \rightarrow \neg Q(a)| = F$.

Тоді $|D(a)| = T$, $| \neg Q(a) | = F$, т.е. $|Q(a)| = T$.

З $| \exists x(P(x) \& (D(a) \rightarrow L(x, a))) | = T$ слідує, що існує таке $x = b$, що $|P(b) \& (D(a) \rightarrow L(b, a))| = T$. Тоді $|P(b)| = T$ та $|D(a) \rightarrow L(b, a)| = T$, а так як $|D(a)| = T$, то $|L(b, a)| = T$.

Посилка $F_2: | \forall x(P(x) \rightarrow (Q(a) \rightarrow \neg L(x, a))) | = T$ для всіх x , в тому числі для $x=b$, отже, $|P(b) \rightarrow (Q(a) \rightarrow \neg L(b, a))| = T$, а так як $|P(b)| = T$, $|Q(a)| = T$, то з $|Q(a) \rightarrow \neg L(b, a)| = T$ слідує $| \neg L(b, a) | = T$.

Таким чином, отримуємо, що істинні обидва твердження: $|T \rightarrow L(b, a)| = T$

$$i \quad | T \rightarrow \neg L(b, a) | = T, \text{ т.е. } | L(b, a) | = T \quad i | \neg L(b, a) | = T.$$

Отримане протиріччя доводить логічне слідування.

Формальний виведення

(вказіть праворуч, які правила було застосовано, самотійно):

1. $\exists x(P(x) \& \forall y(D(y) \rightarrow L(x, y)))$
2. $\forall x(P(x) \rightarrow \forall y(Q(y) \rightarrow \neg L(x, y)))$
3. $P(b) \& \forall y(D(y) \rightarrow L(b, y))$
4. $P(b)$
5. $\forall y(D(y) \rightarrow L(b, y))$
6. $P(w) \rightarrow \forall y(Q(y) \rightarrow \neg L(w, y))$
7. $\forall y(Q(y) \rightarrow \neg L(b, y))$
8. $Q(z) \rightarrow \neg L(b, z)$
9. $D(z) \rightarrow L(b, z)$
10. $L(b, z) \rightarrow \neg Q(z)$
11. $D(z) \rightarrow \neg Q(z)$
12. $\forall y(D(y) \rightarrow \neg Q(y))$

Метод резолюцій.

Знайдемо попереджені нормальні форми (ПНФ) для посилок.

$$\begin{aligned} \Gamma 1: \exists x(P(x) \& \forall y(D(y) \rightarrow L(x, y))) &= \exists x(P(x) \& \forall y(\neg D(y) \vee L(x, y))) = \\ &= \exists x(\forall y(\neg D(y) \vee L(x, y)) \& P(x)) = \exists x \forall y((\neg D(y) \vee L(x, y)) \& P(x)) - \text{ПНФ.} \end{aligned}$$

$$\begin{aligned} \Gamma 2: \forall x(P(x) \rightarrow \forall y(Q(y) \rightarrow \neg L(x, y))) &= \forall x(\neg P(x) \vee \forall y(\neg Q(y) \vee \neg L(x, y))) = \\ &= \forall x(\forall y(\neg Q(y) \vee \neg L(x, y)) \vee \neg P(x)) = \forall x \forall y(\neg Q(y) \vee \neg L(x, y) \vee \neg P(x)) - \text{ПНФ.} \end{aligned}$$

Знайдемо скулемовську стандартну форму (ССФ) посилок.

$\Gamma 1$: Елімінуємо квантор $\exists$ в формулі $\exists x \forall y((\neg D(y) \vee L(x, y)) \& P(x))$.

Візьмемо $x = a$, отримаємо ССФ посилок $\Gamma 1$:

$$\forall y((\neg D(y) \vee L(a, y)) \& P(a),$$

$\Gamma 2$: $\forall x \forall y(\neg Q(y) \vee \neg L(x, y) \vee \neg P(x))$ знаходиться в ССФ.

Візьмемо заперечення від висновка G :

$$\neg \forall y (D(y) \rightarrow \neg Q(y)) = \exists y \neg (\neg D(y) \vee \neg Q(y))$$

$= \exists y (D(y) \& Q(y))$. Це УНФ. Елімінуємо квантор $\exists$, взяв $y = b$: $D(b) \& Q(b)$ – ССФ.

Отримаємо множину диз'юнктив:

$$S = \{\neg D(y) \vee L(a, y), P(a), \neg Q(y) \vee \neg L(x, y) \vee \neg P(x), D(b), Q(b)\}.$$

Побудуємо резолютивний вивід:

1. $\neg D(y) \vee L(a, y)$
2. $P(a)$
3. $\neg Q(y) \vee \neg L(x, y) \vee \neg P(x)$
4. $D(b)$
5. $Q(b)$
6. $\neg L(x, b) \vee \neg P(x)$ $\{b/y\}$, резольвента 5, 3
7. $\neg L(a, b)$ $\{a/x\}$, резольвента 2, 6
8. $\neg D(b)$ $\{b/y\}$, резольвента 1, 7
9. $\square$ резольвента 4, 8

Побудова логічної програми.

Запишемо множину диз'юнктив у вигляді клауз.

Predicates $P(x), D(y), Q(y), L(x, y)$,

Clauses

$P(a)$

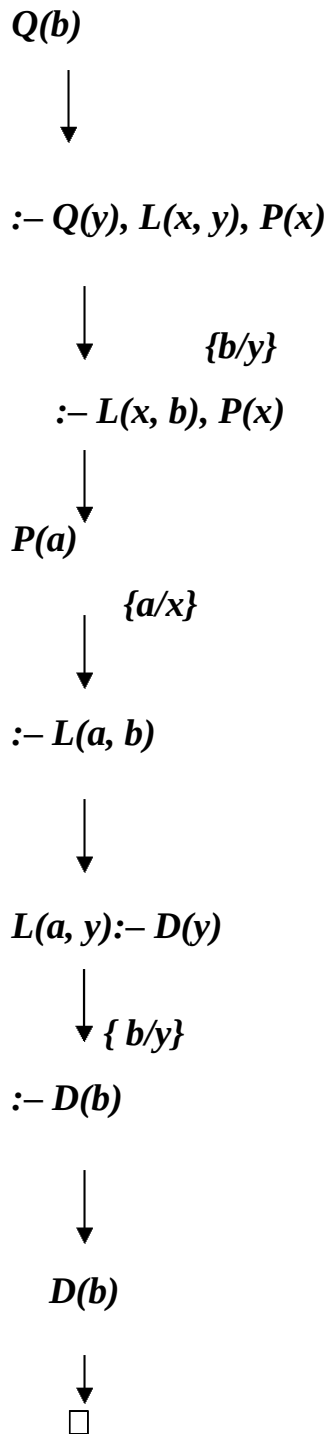
$D(b)$

$:- Q(y), L(x, y), P(x)$

$L(a, y) :- D(y)$

Goal: ? – not $Q(b)$

Логічне дерево виконання логічної програми:



4)

Область визначення : небесні тіла.

1. Кожне небесне тіло є або зіркою, або кометою, або планетою. 2. Венера - небесне тіло, що не є зіркою. 3. У комет, розташованих недалеко від сонця, є хвости. 4. Венера недалеко від Сонця, але у неї немає хвоста.

Висновок: Венера - планета.

$S(x)$ – x зірка,

$C(x)$ – x комета,

$P(x)$ – x планета,

$F(x)$ – x недалеко від сонця,

$T(x)$ – x має хвіст,

V – const Венера.

1. $\forall x (S(x) \vee C(x) \vee P(x))$ г1

2. $\neg S(V)$ г2

3. $\forall x (C(x) \& F(x) \rightarrow T(x))$ г3

4. $F(V) \& \neg T(V)$ г4

Висновок: $P(V)$ – потрібно довести, в умові він відсутній.

5. $S(y) \vee C(y) \vee P(y)$ УК1

6. $C(y) \& F(y) \rightarrow T(y)$ УК3

7. $F(V)$ видалення & 4

8. $\neg T(V)$ видалення & 4

9. $F(y) \rightarrow (C(y) \rightarrow T(y))$ екв.6

10. $C(V) \rightarrow T(V)$ GMP 7, 9

11. $\neg C(V)$ МТ 8, 10

12. $\neg S(y) \rightarrow (\neg C(y) \rightarrow P(y))$ екв.5

13. $\neg C(V) \rightarrow P(V)$ GMP 2, 12

14. $P(V)$ МР 11, 13

Переведення гіпотез в ПНФ та ССФ

1, 2, 4 – ПНФ=ССФ

3. $\forall x (C(x) \& F(x) \rightarrow T(x)) = \forall x (\neg C(x) \vee \neg F(x) \vee T(x))$

Заперечення висновку

$\neg P(V)$ (ПНФ=ССФ)

Множина диз'юнктив

$\{\neg P(V), \neg S(V), F(V), \neg T(V), S(x) \vee C(x) \vee P(x), \neg C(x) \vee \neg F(x) \vee T(x)\}$

Резолюція

1. $\neg P(V)$ заперечення висновку
2. $\neg S(V)$
3. $F(V)$
4. $\neg T(V)$
5. $S(x) \vee C(x) \vee P(x)$
6. $\neg C(x) \vee \neg F(x) \vee T(x)$ *nn.2-6 гіпотези*
7. $S(V) \vee C(V)$ $\{V/x\}$ *рез. 1, 5*
8. $C(V)$ *рез. 2, 7*
9. $\neg F(V) \vee T(V)$ $\{V/x\}$ *рез. 6, 8*
10. $T(V)$ *рез. 3, 9*
11. $\square$ *рез. 4, 10*

Логічна програма

Predicates: $P(x), F(x), T(x), S(x), C(x)$

Clauses: $F(V)$
 $:- T(V)$
 $:- S(V)$
 $T(x) :- C(x), F(x)$
 $S(x), C(x), P(x)$

Goal ? – $P(V)$.

$:- P(V)$
|
 $S(x), C(x), P(x)$
| $\{V/x\}$
 $S(V), C(V)$
|

$$\begin{array}{c}
:- S(V) \\
| \\
C(V) \\
| \\
T(x) :- C(x), F(x) \\
| \quad \{V/x\} \\
T(V) :- F(V) \\
| \\
F(V) \\
| \\
T(V) \\
| \\
:-T(V) \\
| \\
\Box
\end{array}$$

5)

Область визначення: тварини.

1. Тварини, які не брикаються ($K(x)$ – x не брикається), *завжди* незворушні ($N(x)$ – x незворушливий).

2. У віслюка ($O(x)$ – x віслюк) немає рогів ($R(x)$ – x не має рогів).

3. Буйвол ($B(x)$ – x буйвол) *завжди* може перекинути вас через огорожу ($G(x)$ – x може перекинути вас через огорожу).

4. Тварин, які брикаються, не легко проковтнути ($P(x)$ – x не легко проковтнути).

5. Тварина, яка не має рогів, не може перекинути вас через огорожу.

6. Всі тварини, **крім** буйвола, легко переходять у лють (не є незворушливими) = незворушливий – тільки буйвол.

Висновок: Віслюка проковтнути не легко.

Формалізація:

1. $\forall x (K(x) \rightarrow N(x))$
2. $\forall x (O(x) \rightarrow R(x))$
3. $\forall x (B(x) \rightarrow G(x))$
4. $\forall x (\neg K(x) \rightarrow P(x))$
5. $\forall x (R(x) \rightarrow \neg G(x))$
6. $\forall x (N(x) \rightarrow B(x))$

Висновок $\forall x (O(x) \rightarrow P(x))$

- | | |
|--|---------------|
| 7 $K(y) \rightarrow N(y)$ | УК 1 |
| 8 $O(y) \rightarrow R(y)$ | УК2 |
| 9 $B(y) \rightarrow G(y)$ | УК3 |
| 10 $\neg K(y) \rightarrow P(y)$ | УК4 |
| 11 $R(y) \rightarrow \neg G(y)$ | УК5 |
| 12 $N(y) \rightarrow B(y)$ | УК 6 |
| 13. $O(y) \rightarrow \neg G(y)$ | сил 8, 11 |
| 14 $K(y) \rightarrow B(y)$ | сил 7, 12 |
| 15 $K(y) \rightarrow G(y)$ | сил 14, 9 |
| 16 $\neg P(y) \rightarrow K(y)$ | контр.10 |
| 17 $\neg P(y) \rightarrow G(y)$ | сил 15, 16 |
| 18 $\neg G(y) \rightarrow P(y)$ | контр.17 |
| 19 $O(y) \rightarrow P(y)$ | сил 13, 18 |
| 20 $\forall x (O(x) \rightarrow P(x))$ | Gen 19 |

*При формалізації цілком допустимо взяти «віслюка» як константу з області визначення («буйвол» не може бути константою через 6-те речення – **крім**)*

Тоді задача виглядає наступним чином і розв'язок схожий на наступний приклад.

1. $\forall x (K(x) \rightarrow N(x))$
2. $R(o)$

3. $\forall x (B(x) \rightarrow G(x))$
4. $\forall x (\neg K(x) \rightarrow P(x))$
5. $\forall x (R(x) \rightarrow \neg G(x))$
6. $\forall x (N(x) \rightarrow B(x))$

P(o)

6)

Область визначення: тварини

1. **Єдині** тварини у цьому будинку ($H(x)$ – x в цьому будинку) – кішки ($C(x)$ – x кішка).
2. Будь-яку тварину можна приручити ($D(x)$ – x можна приручити), якщо вона любить дивитися на місяць ($L(x)$ – x любить дивитися на місяць).
3. Якщо тварина викликає у мене огиду ($O(x)$ – x викликає у мене огиду), я намагаюся триматися від неї подалі ($F(x)$ – я тримаюся подалі від x).
4. Жодна тварина не плотоядна ($T(x)$ – x не плотоядна), якщо вона не бродить ночами ($N(x)$ – x не бродить ночами).
5. Жодна кішка не пропустить нагоди зловити мишу ($M(x)$ – x ловить мишей).
6. Я не пускаю до себе до кабінету тварин ($R(x)$ – я не пускаю x до кабінету), **крім тих**, що знаходяться у цьому будинку.
7. Кенгуру (***k-const***) не піддається прирученню.
8. **Лише** плотоядні тварини ловлять мишей.
9. Тварини, яких я не пускаю до себе в кабінет, викликають у мене огиду.
10. Тварини, які блукають ночами, люблять дивитися на місяць.

*Висновок: Я **завжди** намагаюся триматися подалі від кенгуру.*

1. $\forall x (H(x) \rightarrow C(x))$
2. $\forall x (L(x) \rightarrow D(x))$
3. $\forall x (O(x) \rightarrow F(x))$
4. $\forall x (N(x) \rightarrow T(x))$

5. $\forall x (C(x) \rightarrow M(x))$
6. $\forall x (\neg R(x) \rightarrow H(x))$
7. $\neg D(k)$
8. $\forall x (M(x) \rightarrow \neg T(x))$
9. $\forall x (R(x) \rightarrow O(x))$
10. $\forall x (\neg N(x) \rightarrow L(x))$

Висновок: F(k)

- | | |
|-----------------------------------|-------------|
| 11 $(H(y) \rightarrow C(y))$ | YK 1 |
| 12 $(L(y) \rightarrow D(y))$ | YK2 |
| 13 $(O(y) \rightarrow F(y))$ | YK3 |
| 14 $(N(y) \rightarrow T(y))$ | YK4 |
| 15 $(C(y) \rightarrow M(y))$ | YK5 |
| 16 $(\neg R(y) \rightarrow H(y))$ | YK6 |
| 17 $(M(y) \rightarrow \neg T(y))$ | YK8 |
| 18 $(R(y) \rightarrow O(y))$ | YK9 |
| 19 $(\neg N(y) \rightarrow L(y))$ | YK 10 |
| 20 $\neg L(k)$ | GMT 7, 12 |
| 21 $N(k)$ | GMT 20, 19 |
| 22 $T(k)$ | GMP 21, 14 |
| 23 $\neg M(k)$ | GMT 22, 17 |
| 24 $\neg C(k)$ | GMT 23, 15 |
| 25 $\neg H(k)$ | GMT 24, 11 |
| 26 $R(k)$ | GMT 25, 16 |
| 27 $O(k)$ | GMP 26, 18 |
| 28 $F(k)$ | GMP 27, `13 |

7) Область визначення: люди. 1. Святий Франциск любить кожного, хто любить когось. 2. Кожен любить когось. Висновок: Святий Франциск любить кожного.

$L(x)$ - x любить когось;

$F(x)$ - святий Франциск любить x ;

1) $\forall x(L(x) \rightarrow F(x))$, 2) $\forall x(L(x))$. Висновок $\forall xF(x)$

1. $\forall x(L(x) \rightarrow F(x))$ Г1

2. $\forall x(L(x))$ Г2

3. $L(y) \rightarrow F(y)$ УК 1

4. $L(y)$ УК 2

5. $F(y)$ МР 3,4

6. $\forall xF(x)$ Gen 5

Варіант розв'язку при іншій формалізації:

$L(x,y)$ - x любить y ;

F - святий Франциск.

1) $\forall x(\exists y L(x,y) \rightarrow L(F,x))$, 2) $\forall x \exists y L(x, y)$. Висновок $\forall x L(F,x)$

1. $\forall x(\exists y L(x,y) \rightarrow L(F,x))$ Г1

2. $\forall x \exists y L(x, y)$ Г2

3. $\exists y L(z,y) \rightarrow L(F, z)$ УК 1

4. $\exists y L(z, y)$ УК 2

5. $L(F, z)$ МР 3,4

6. $\forall x L(F, x)$ Gen 5

4. ТЕОРІЯ АЛГОРИТМІВ

4.1. Примітивно рекурсивні та рекурсивні функції

Теоретичні відомості

1. Наступні функції називаються *вихідними функціями*.

(I) **Нуль-функція**: $Z(x) = 0$ для кожного x .

(II) **Додаток одиниці** (*наступний за*): $N(x) = x + 1$ для кожного x .

(III) **Проектуючі функції**: $U_i^n(x_1, \dots, x_n) = x_i$ при всіх $x_1, \dots, x_n$ ($i = 1, \dots, n$; $n = 1, 2, \dots$).

2. Наступні два правила: підстановка і примітивна рекурсія, – служать для отримання нових функцій, виходячи з уже наявних.

(IV) **Підстановка**. Кажуть, що функція f отримана за допомогою підстановки з функцій $g(y_1, \dots, y_m)$, $h_1(x_1, \dots, x_n)$, ..., $h_m(x_1, \dots, x_n)$, якщо $f(x_1, \dots, x_n) = g(h_1(x_1, \dots, x_n), \dots, h_m(x_1, \dots, x_n))$.

(V) **Схема примітивної рекурсії**. Функція f отримана з функцій g і h за допомогою рекурсії, якщо

$$f(x_1, \dots, x_n, 0) = g(x_1, \dots, x_n),$$

$$f(x_1, \dots, x_n, y + 1) = h(x_1, \dots, x_n, y, f(x_1, \dots, x_n, y)).$$

3. Функція f називається *примітивно рекурсивною*, якщо вона може бути отримана з вихідних функцій за допомогою кінцевого числа підстановок (IV) і рекурсій (V), тобто якщо існує така кінцева послідовність функцій $f_1, \dots, f_n$, що $f_n = f$ і для кожного i , $0 \leq i \leq n$, функція f_i або вихідна, або може бути отримана з деяких попередніх їй в цій послідовності функцій за допомогою застосування правила (IV) (підстановки) або правила (V) (рекурсії).

4. Теорема. Наступні функції є примітивно рекурсивними.

(a) $x + y$ (додавання).

Доведення.

За правилом рекурсії (V):

$$x + 0 = x, \text{ тобто } f(x, 0) = U_1^1(x) = x.$$

$$x + (y + 1) = N(x + y), \text{ тобто } f(x, y + 1) = N(f(x, y)).$$

(b) $x \cdot y$ (множення).

Доведення.

$$x \cdot 0 = 0, \text{ тобто } g(x, 0) = Z(x).$$

$x \cdot (y + 1) = (x \cdot y) + x$, тобто $g(x, y + 1) = f(x, g(x, y))$, де f є функція додавання.

с) x^y (піднесення до степеня).

Доведення.

$$x^0 = 1, x^{y+1} = (x^y) \cdot x.$$

(d) $\delta(x) = x - 1$, якщо $x > 0$ і $\delta(x) = 0$, якщо $x = 0$ (віднімання одиниці).

Доведення.

$$\delta(0) = 0, \delta(y + 1) = y.$$

(e) $x \div y = x - y$, якщо $x \geq y$ і $x \div y = 0$, якщо $x < y$ (обмежена різниця).

Доведення.

$$x \div 0 = x, x \div (y + 1) = \delta(x \div y).$$

(f) $|x - y| = X - y$, якщо $x \geq y$, $|x - y| = y - x$, якщо $x < y$ (модуль різниці).

Доведення.

$$|x - y| = (x \div y) + (y \div x) \text{ (підстановка).}$$

(g) $sg(x) = 0$, якщо $x = 0$, $sg(x) = 1$, якщо $x \neq 0$ (Сігнум x).

Доведення.

$$sg(0) = 0, sg(y + 1) = 1.$$

(h) $unsg(x) = 1$, якщо $x = 0$, $unsg(x) = 0$, якщо $x \neq 0$;

Доведення.

$$unsg(x) = 1 \div sg(x).$$

(i) $x!$.

Доведення.

$$0! = 1, (y + 1)! = (Y!) \cdot (y + 1).$$

(j) $\min(x, y) = \text{найменшому з чисел } x \text{ і } y$;

Доведення.

$$\min(x, y) = x \div (x \div y).$$

$$(k) \min(x_1, \dots, x_n).$$

Доведення.

Припустимо, що функція $\min(x_1, \dots, x_n)$ – примітивно рекурсивна. Для $\min(x_1, \dots, x_{n+1})$

$$\text{маємо } \min(x_1, \dots, x_{n+1}) = \min(\min(x_1, \dots, x_n), x_{n+1}).$$

$$(l) \max(x, y) = \text{найбільшому з чисел } x \text{ і } y.$$

Доведення.

$$\max(x, y) = y + (x \div y).$$

$$(m) \max(x_1, \dots, x_n);$$

Доведення.

$$\max(x_1, \dots, x_{n+1}) = \max(\max(x_1, \dots, x_n), x_{n+1}).$$

(n) $rm(x, y)$ = залишку від ділення y на x , якщо $x \neq 0$, і y , якщо $x = 0$.

Доведення.

$$rm(x, 0) = 0, rm(x, y+1) = N(rm(x, y)) \cdot sg(|x - N(rm(x, y))|).$$

$$(o) qt(x, y) = \text{частці від ділення } y \text{ на } x.$$

Доведення.

$$qt(x, 0) = 0, qt(x, y+1) = qt(x, y) + unsq(|x - N(rm(x, y))|).$$

5. (VI) μ -оператор (оператор мінімізації). Нехай функція $g(x_1, \dots, x_n, y)$ така, що для будь-яких $x_1, \dots, x_n$ існує по крайній мірі одне значення y , при якому $g(x_1, \dots, x_n, y) = 0$. Позначимо через $\mu_y(g(x_1, \dots, x_n, y) = 0)$ найменше значення y , при якому $g(x_1, \dots, x_n, y) = 0$. Тоді функція f отримана з функції g з допомогою μ -оператора, якщо $f(x_1, \dots, x_n) = \mu_y(g(x_1, \dots, x_n, y) = 0)$.

6. Функція f називається *рекурсивною*, якщо вона може бути отримана з вихідних функцій за допомогою кінцевого числа застосувань підстановки (IV), рекурсії (V) і μ -оператора (VI).

Це останнє визначення відрізняється від визначення примітивно рекурсивної функції лише додатковим дозволом застосовувати μ -оператор. Тому будь-яка примітивно рекурсивна функція є також і рекурсивною функцією. Зворотнє, втім, не завжди вірно. (В літературі замість терміна «рекурсивний» іноді вживається термін «частково рекурсивний»).

7. Відношення $R(x_1, \dots, x_n)$ називається *примітивно рекурсивним* (рекурсивним) відношенням, якщо примітивно рекурсивною (відповідно рекурсивною) є його характеристична функція $C_R(x_1, \dots, x_n)$.

8. Теорема. Відношення, які можна отримати з примітивно рекурсивних (або рекурсивних) за допомогою пропозиційних зв'язок і обмежених кванторів, також примітивно рекурсивні (відповідно, рекурсивні).

Доведення.

Нехай відношення $P(x_1, \dots, x_n)$ і $Q(x_1, \dots, x_n)$ примітивно рекурсивні (рекурсивні). Це означає, що примітивно рекурсивними (рекурсивними) є їхні характеристичні функції C_P і C_Q . Тоді примітивно рекурсивними (рекурсивними) є і відношення: $\neg P(x_1, \dots, x_n)$ (його характеристична функція: $C_{\neg P} = 1 \div C_P$), $P(x_1, \dots, x_n) \vee Q(x_1, \dots, x_n)$ (характеристична функція: $C_{P \vee Q} = sg(C_P + C_Q)$), $P(x_1, \dots, x_n) \& Q(x_1, \dots, x_n)$ (характеристична функція: $C_{P \& Q} = C_P \cdot C_Q$). Звідси неважко отримати характеристичні функції для формул: $R_1 = \forall y_{y < z} P(x_1, \dots, x_n, y)$ і $R_2 = \exists y_{y < z} P(x_1, \dots, x_n, y)$ із допомогою обмеженої суми і обмеженого добутку: $C_{R_2} = sg(\sum_{y < z} C_P(x_1, \dots, x_n, y))$ і $C_{R_1} = \prod_{y < z} C_P(x_1, \dots, x_n, y)$ ¹.

¹ Оскільки квантор загальності - узагальнення кон'юнкції, а квантор існування - узагальнення операції диз'юнкції, для отримання нових характеристичних функцій можна скористатися операторами максимуму і мінімуму.

Приклади.

1. Для функції $sg(x)$ зворотну функцію можна визначити як $sg^{-1}(x) = \mu x (sg(x)=1)$. Тоді $sg^{-1}(x)=x$, якщо $x=0$, 1; і $sg^{-1}(x)$ не визначене, якщо $x > 1$.

2. Визначимо функцію $y = [z/x]$ (частка від ділення z на x) як функцію, зворотну множенню: $z = y \cdot x$. Тоді $y = \mu_{y \leq z} (|y \cdot x - z| = 0)$, тобто це буде найменше значення y , при якому існує рішення рівняння $|y \cdot x - z| = 0$. Функція в повному обсязі не визначена, тому що не кожні два числа діляться один на одного без залишку.

3. Функція $y = [\sqrt{x}]$ (ціла частина $\sqrt{x}$) може бути визначена за допомогою обмеженого μ -оператора як $y = \mu_{y \leq x} (y^2 = x)$. Ця функція також є частково визначною функцією.

4. Відношення $x_1 = x_2$ примітивно рекурсивно, так як характеристична функція його збігається з функцією $unsg(|x_1 - x_2|)$.

5. Примітивно рекурсивна функція $unsg(x_1 \div x_2)$ служить характеристичною функцією відношення $x_1 < x_2$, яке, таким чином, примітивно рекурсивно.

Геделева нумерація

Кожному символу u довільної теорії першого порядку K наступним чином поставимо у відповідність позитивне число $g(u)$, зване *геделевим номером* символу u :

$$g(()) = 3;$$

$$g(.) = 7;$$

$$g(0) = 5;$$

$$g(\neg) = 9;$$

$$g(\rightarrow) = 11;$$

$$g(x_k) = 5 + 8k \text{ для } k = 1, 2, \dots;$$

$$g(a_k) = 7 + 8k \text{ для } k = 1, 2, \dots;$$

$$g(f_k^n) = 9 + 8(2^n 3^k) \text{ для } k, n \geq 1;$$

$$g(A_k^n) = 11 + 8(2^n 3^k) \text{ для } k, n \geq 1;$$

при цьому покладемо $g(\forall x_i) = g((x_i))$.

Таким чином, різним символам поставлені у відповідність різні геделеві номери, що є натуральними числами. Наприклад, $g(x_2)=21$, $g(a_4)=39$, $g(f_1^2)=105$, $g(A_2^1)=155$.

Нехай дано вираз $u_0 u_1 \dots u_r$, що представляє, наприклад, формулу теорії першого порядку. Геделевим номером $g(u_0 u_1 \dots u_r)$ цього виразу визначимо як $2^{g(u_0)} 3^{g(u_1)} \dots p_r^{g(u_r)}$, де $p_i \in i$ -е просте число і $p_0=2$.

Наприклад, $g(A_1 \rightarrow (A_2 \rightarrow A_1)) = 2^{g(A_1)} 3^{g(\rightarrow)} 5^{g(A_2)} 11^{g(\rightarrow)} 13^{g(A_1)} 17^{g(A_1)}$.

Зауважимо, що, в силу єдиності розкладання натуральних чисел в добутки степенів простих чисел, різні вирази отримують при цьому різні геделеві номери. Крім того, геделеві номери виразів є парними числами, і тому відмінні від геделевих номерів символів.

Нарешті, геделев номер довільної послідовності $e_0, \dots, e_r$ виразів (наприклад, виведення формули e_r в теорії K) визначимо наступним чином: $g(e_0, \dots, e_r) = 2^{g(e_0)} \dots 3^{g(e_1)} \dots p_r^{g(e_r)}$. Як і раніше, різні послідовності виразів мають різні геделеві номери, а так як ці останні парні і, крім того, мають парний показник ступеня при 2, то вони відмінні і від геделевих номерів символів, і від геделевих номерів виразів.

Таким чином, функція g взаємно однозначно відображає множину всіх символів, виразів і кінцевих послідовностей виразів в множину цілих додатних чисел. Множина значень функції g не збігається, проте, з множиною всіх натуральних чисел, так як не всі натуральні числа є геделевими номерами.

Завдання 4.1.1

1) Застосувати операцію примитивної рекурсії до функцій $g(x_1)$ і $h(x_1, x_2, x_3)$. Функцію $f(x_1, x_2) = R(g, h)$ записати в “аналітичній формі”.

1) $g(x_1)=x_1$ $h(x_1, x_2, x_3)=x_1 + x_2$

2) $g(x_1)=x_1$ $h(x_1, x_2, x_3)=x_1 + x_3$

3) $g(x_1)=x_1$ $h(x_1, x_2, x_3)=x_2 + x_3$

4) $g(x)=2^x$ $h(x_1, x_2, x_3)=(x_3)^{x_1}$ (вважаємо $0^0=1$)

2) Нехай $g_1(y)$, $g_2(y)$, $g_3(x,y)$ – примитивно рекурсивні функції. Показати, що $f(x,y)$, що визначається наступною схемою, теж примитивна:

$$\begin{cases} f(0,y) = g_1(y), \\ f(x+1,0) = g_2(x), \\ f(x+1,y+1) = g_3(x,y) \end{cases}$$

3) Побудувати об'єкти, що відповідають геделевим номерам:

- 1944
- 816293376
- 47
- 585

4) Знайти геделев номер виразу (записати формулу):

- $f_1^1(a_1)$;
- $\exists x_1 A_1^1(x_1)$;
- $\forall x_1 A_1^2(x_1, x_2)$;
- $A_1^2(a_1, x_2) \rightarrow A_1^1(x_1)$.

4.2. Алгоритмічні схеми. Машина Тюрінга

Теоретичні відомості

Машина Тюрінга складається з трьох частин.

1) Нескінченна в обидві сторони стрічка, розбита на клітинки, в кожній з яких може перебувати один (і тільки один) символ зовнішнього алфавіту $A = \{ \alpha, \beta, \gamma, \dots \}$. Інформація записується на стрічці у вигляді слова в алфавіті A .

2) Голівки, що зчитує, яка в один дискретний момент часу оглядає одну клітинку і може перебувати в одному з внутрішніх станів $q_i \in Q$, де Q - алфавіт внутрішніх станів. При цьому вона розпізнає записаний в чарунці символ зовнішнього алфавіту, записує замість нього інший символ (можливо, той же самий), переходить в інший стан (можливо – залишається в поточному), після чого зсувається вправо, вліво або залишається на місці (П, Л, Н).

3) Наступна частина машини Тюрінга - це програма, що складається з команд виду:

$$\alpha q_i \rightarrow \beta q_j \xi, \quad \xi \in \{Л, П, Н\}, \quad q_i, q_j \in Q, \quad \alpha, \beta \in A.$$

В результаті виконання цієї команди символ α на стрічці буде замінений символом β , головка змінить своє положення в залежності від ξ і внутрішній стан зміниться з q_i на q_j .

Послідовність команд задає алгоритм переробки слова. На вхід машини Тюрінга подається вихідне слово W в алфавіті A і задається початковий стан q_0 : початкова конфігурація. Робота машини Тюрінга відбувається по *тактах*, або по *кроках*. При роботі машини Тюрінга можливе розширення зовнішнього алфавіту A допоміжними символами, які після переробки слова замінюються символами алфавіту A або витираються. У число символів алфавіту A обов'язково входить порожній символ Λ . Вважається, що спочатку все чарунки стрічки заповнені порожніми символами.

Приклад.

Визначити кінцеве слово після роботи програми для машини Тюрінга в абетці $A = \{ |, *, \Lambda \}$ (Λ - позначення пробілу), якщо початкове слово: $|||*$. В початковому стані q_0 машина знаходиться на крайньому лівому символі.

Програма для машини Тюрінга :

$A \setminus Q$	q_0	q_1	q_2
	Λ q_1 П	q_1 П	q_2 Л
*	Λ !	* q_1 П	* q_2 Л
Λ	q_2 Л	q_0 П	Λ q_0 П

Машина Тюрінга є пристроєм, що переробляє знаки. Задача машини Тюрінга - переробляти вхідне слово в деяке вихідне слово. Послідовність команд в таблиці задає алгоритм переробки слова. На вхід машини Тюрінга подається слово й задається початковий стан q_0 :

$q_0|||* \rightarrow \Lambda q_1||* \rightarrow |q_1|* \rightarrow ||q_1* \rightarrow ||*q_1\Lambda \rightarrow ||*|q_0\Lambda \rightarrow ||*|q_2| \rightarrow ||*q_2|| \rightarrow ||q_2*||$
 $\rightarrow |q_2|*|| \rightarrow q_2||*|| \rightarrow q_2 \Lambda||*|| \rightarrow q_0||*|| \rightarrow \Lambda q_1|*|| \rightarrow q_1|*|| \rightarrow |q_1*|| \rightarrow |*q_1|| \rightarrow$
 $|*|q_1| \rightarrow |*||q_1\Lambda \rightarrow |*|||q_0\Lambda \rightarrow |*|||q_2| \rightarrow |*||q_2|| \rightarrow |*|q_2||| \rightarrow |*q_2||| \rightarrow |q_2*||| \rightarrow$
 $q_2|*||| \rightarrow q_2 \Lambda|*||| \rightarrow q_0|*||| \rightarrow \Lambda q_1|*||| \rightarrow *q_1||| \rightarrow *|q_1||| \rightarrow *||q_1| \rightarrow$
 $*|||q_1\Lambda \rightarrow *|||q_0\Lambda \rightarrow *|||q_2| \rightarrow *|||q_2|| \rightarrow *||q_2||| \rightarrow *|q_2||| \rightarrow *q_2||| \rightarrow$
 $q_2*||| \rightarrow q_2\Lambda*||| \rightarrow q_0*||| \rightarrow \Lambda|||$

Завдання 4.2.1.

Визначити кінцеве слово після роботи програми для машини Тюрінга в абетці $A = \{ |, *, \Lambda \}$, якщо задано початкове слово. В початковому стані q_0 машина знаходиться на крайньому лівому символі.

1) Визначити кінцеве слово після роботи програми для машини Тюрінга, якщо початкове слово: $|*$.

$A \setminus Q$	q_0	q_1	q_2
$ $	$\Lambda q_1 \Pi$	$ q_1 \Pi$	$ q_2 \text{Л}$
$*$	$\Lambda !$	$* q_1 \Pi$	$* q_2 \text{Л}$
Λ	$ q_2 \text{Л}$	$ q_0 \Pi$	$\Lambda q_0 \Pi$

2) Визначити кінцеве слово після роботи програми для машини Тюрінга, якщо початкове слово: $||*||$.

$A \setminus Q$	q_0	q_1	q_2	q_3	q_4	q_5	Q_6
$ $	$\Lambda q_1 \Pi$	$ q_1 \Pi$	$\Lambda q_3 \text{Л}$	$ q_3 \text{Л}$	$2 q_6 \text{Л}$	$ q_5 \text{Л}$	$ q_6 \Pi$
$*$	$\Lambda q_4 \Pi$	$* q_1 \Pi$	$1 q_5$	$* q_3 \text{Л}$			
Λ		$\Lambda q_2 \text{Л}$		$\Lambda q_0 \Pi$	$0!$	$\Lambda!$	$\Lambda!$

Завдання 4.2.2. Написати програму для машини Тюрінга, навести список контрольних прикладів (верифікація всіх можливих випадків), розгорнуто розписати один з прикладів. Вважається, що вхідні дані не містять помилок, коректні й такі, що обумовлені в задачі.

1) Порівняти два слова в унарній системі числення:

алфавіт $\{ |, *, 1, 2, 0, \Lambda/\lambda \}$;

вхідне слово $||*||$; вихідне 0 – однакові, $1/2$ – перше/друге більше;

2) Перевірити, чи відсортовано слово в алфавіті $\{a, b\}$; відповіді – так/У/1, ні/Н/0; допрацювати алгоритм та реалізувати функцію впорядкування слова;

3) Обчислення функції $f(x)=2x$ в унарній системі числення:

алфавіт $\{ |, *, \Lambda/\lambda \}$; вхідне слово $||$; вихідне $||*||$;

4) Обчислення функції $f(x)=x/2$ з остачею в унарній системі числення;
алфавіт $\{ |, *, A/\lambda \}$;

вхідне слово $|||||$ або $||||||$; вихідне $|||||*|||$ або $||||||*|||*$ з остачею;

5) Переробити двійкове число у від'ємне (додатковий код). Реалізувати два алгоритми: (інверсія +1) або (останні нулі і остання 1 залишаються, а потім, все інвертувати до початку, рухаючись праворуч наліво); алфавіт $\{0,1\}$;

6) Перевірити входження заданого символа в слово в алфавіті $\{a, b, c, *\}$; відповіді – так/Y/1, ні/N/0; символ відокремлюється *, наприклад, $b*abacb$ – так;

7) Підрахувати кількість букв в слові:

алфавіт $\{a, b, |, A/\lambda\}$; вхідне слово abb , вихідне $abb|||$;

8) Підрахувати кількість заданої букви в слові:

алфавіт $\{a, b, |, *, A/\lambda\}$; вхідне слово $b*baabaabb$, вихідне $b*baabaabb|||$;

9) Заміна першого входження/всіх входжень заданої букви на іншу:

алфавіт $\{a, b, *, A/\lambda\}$; вхідне слово $b*a*abaabb$, вихідне $aaaabb/aaaaa$;

10) Заміна букви із зазначеного місця на задану:

алфавіт $\{a, b, |, *, A/\lambda\}$; вхідне слово $b|||baabaabb$, вихідне $babbbaabb$;

11) Скопіювати слово: вхід $babbc$, вихід $babbc* babbc$;

12) Прочитати слово в зворотному порядку: вхід $baaba$, вихід $baaba*aabab$;

13) Порівняти два двійкових числа однакової розрядності, розділовий знак *: вхідне слово $1010*1001$; вихідне 0 – однакові, 1/2 – перше/друге більше;

14) Логічний (циклічний) зсув вправо (вліво) двійкового слова на задану кількість розрядів (відзначається *); приклад вхідного слова для реалізації функції циклічного зсуву вправо на три розряди $***010100$, вихідне – 100010 ; для логічного зсуву вліво на два розряди - $1001**$, вихідне – 0100 ; кількість розрядів може відмічатися як на початку слова, так і наприкінці;

15) Поразрядна диз'юнкція (перетин) двох двійкових чисел однакової розрядності; розділовий знак *; вихідне слово – тільки результат: вхід $1001*1100$, результат поразрядної диз'юнкції – 1101 , перетину - 1000 .

4.3. Нормальний алгоритм Маркова

Теоретичні відомості

Нормальний алгоритм Маркова (НАМ) - це кінцевий впорядкований набір підстановок виду $P_i \rightarrow (.) Q_i, i = 1, 2, \dots, n$. $P_i \rightarrow Q_i$ - звичайна підстановка, яка означає, що крайнє ліве входження підслова P_i в W замінюється на Q_i , $P_i \rightarrow . Q_i$ - кінцева підстановка, тобто після її виконання робота алгоритму завершується.

Початкове слово $W \in A^*$ переробляється за допомогою алгоритму наступним чином. Починаючи з першої підстановки, алгоритм шукає перше ліве входження підслова P_i в слові W . Якщо воно знайдено, то воно замінюється на Q_i , після чого список підстановок проглядається з самого початку. Якщо ж входження P_i не було знайдено, то вибирається наступна підстановка. Якщо була виконана заключна підстановка (з крапкою), то процес переробки слова завершується, і тоді говорять, що алгоритм застосуваний до даного слова. Може виявитися, що процес не завершується, тобто жодна із заключних підстановок не застосовувалася в процесі переробки слова. У цьому випадку говорять, що алгоритм не застосуваний до даного слова.

Приклади.

Визначити кінцеве слово після роботи нормального алгоритму Маркова (НАМ) в абетці $A = \{ |, \Lambda, * \}$ (Λ - позначення пробілу) :

1. $a | | - | a$
2. $a \Lambda - ! \text{ stop}$
3. $a | \Lambda - * | ! \text{ stop}$
4. $\Lambda | - a |$

Застосуємо НАМ до вхідного слова $|||||$:

$||||| \rightarrow (4) a ||||| \rightarrow (1) |a||| \rightarrow (1) ||a| \rightarrow (1) |||a \rightarrow (2) |||$

Завдання 4.3.1. Визначити кінцеве слово після роботи нормального алгоритму Маркова (НАМ) в абетці $A = \{ |, \Lambda, * \}$:

1) початкове слово $||$, НАМ:

1. $a \mid - \mid a$
2. $a \Lambda - ! \text{ stop}$
3. $a \mid \Lambda - * \mid ! \text{ stop}$
4. $\Lambda \mid - a \mid$

2)) початкове слово $|$, НАМ:

1. $a \mid - \mid a$
2. $a \Lambda - ! \text{ stop}$
3. $\Lambda \mid - a \mid$

3) початкове слово: $|||$, НАМ:

1. $a \mid \mid - \mid a$
2. $a \mid \mid \Lambda - * \mid \mid ! \text{ stop}$
3. $a \mid \Lambda - * \mid ! \text{ stop}$
4. $a \Lambda - ! \text{ stop}$
5. $\Lambda \mid - a \mid$

Завдання 4.3.2. Скласти НАМ, навести список контрольних прикладів (верифікація всіх можливих випадків), розгорнуто розписати один з прикладів. Вважається, що вхідні дані не містять помилок, коректні й такі, що обумовлені в задачі.

1) Порівняти два слова в унарній системі числення:

алфавіт $\{ |, *, 1, 2, 0, \Lambda/\lambda \}$;

вхідне слово $||*|||$; вихідне 0 – однакові, $1/2$ – перше/друге більше;

2) Обчислення функції $f(x)=2x$ в унарній системі числення:

алфавіт $\{ |, \Lambda/\lambda \}$; вхідне слово $||$; вихідне $||||$;

3) Обчислення функції $f(x)=x/2$ з остачею в унарній системі числення;

алфавіт $\{ |, *, \Lambda/\lambda \}$;

вхідне слово $|||||$ або $||||||$; вихідне $|||$ або $|||*$ з остачею;

4) Переробити двійкове число у від'ємне (додатковий код). Реалізувати два алгоритми: (інверсія +1) або (останні нулі і остання 1 залишаються, а потім, все інвертувати до початку, рухаючись праворуч наліво); *алфавіт* {0,1};

5) Перевірити входження заданого символу в слово в алфавіті {a, ..., z, *}; відповіді – так/У/1, ні/Н/0; символ відокремлюється *, наприклад, *b*abccb* – так, *c*abaab* – ні;

6) Підрахувати кількість букв в слові;

алфавіт {a, ..., z, |, /λ}; вхідне слово *cbb*, вихідне *cbb|||*;

7) Підрахувати кількість заданої букви в слові:

алфавіт {a, ..., z, |, *, /λ}; вхідне слово *b*baabccbb*, вихідне *b*baabccbb|||*;

8) Заміна першого входження/всіх входжень заданої букви на іншу:

алфавіт {a, ..., z, *, /λ}; вхідне слово *b*a*abccbb*, вихідне *aaccbb/aaccbaa*;

9) Заміна букви з зазначеного місця на задану:

алфавіт {a, ..., z, |, *, /λ}; вхідне слово *b|||baabaccb*, вихідне *babbccb*;

10) Скопіювати слово: вхід *babbc**, вихід *babbc* babbc*;

11) Прочитати слово в зворотному порядку: вхід *babaa**, вихід *babaa*aabab*;

12) Порівняти два двійкових числа однакової розрядності, розділовий знак *: вхідне слово *1010*1001*; вихідне 0 – однакові, 1/2 – перше/друге більше;

13) Логічний (циклічний) зсув вправо (вліво) двійкового слова на задану кількість розрядів (відзначається *); приклад вхідного слова для реалізації функції циклічного зсуву вправо на три розряди ****010100*, вихідне – *100010*; для логічного зсуву вліво на два розряди - *1001***, вихідне – *0100*; кількість розрядів може відмічатися як на початку слова, так і наприкінці;

14) Поразрядна диз'юнкція (перетин) двох двійкових чисел однакової розрядності; розділовий знак *; вихідне слово – тільки результат: вхід *1001*1100*, результат поразрядної диз'юнкції – *1101*, перетину – *1000*;

15) Вилучити літери, що повторюються в слові з *алфавіту* {a, ..., z}: вхід *babzaacc*, вихід *bzas*.

4.4. λ -числення та функції обробки списків

Теоретичні відомості

Лямбда-числення (λ -числення) – розділ дискретної математики, що вивчає процес обчислення, як математичний процес; є теоретичним механізмом для багатьох прикладних наук, в першу чергу, для функціонального програмування. Основна методика вивчення дослідження *застосовності* (калька з англійського - *application*) операндів до функції, при цьому останні представляються у вигляді λ -термів:

$\lambda x. (x + y)$ - функція складання двох своїх операндів.

Ось запис функції, яка отримує на вхід число і обчислює значення, більше його на одиницю: $(\lambda x. x + 1)$; застосування функцій до аргументів записується ось так: $(\lambda x. x + 1) 2$.

Ми можемо спрощувати записаний нами вираз, підставляючи замість змінних їх значення (бета-спрощення, бета-редукція):

$(\lambda x. x + 1) 2$ – застосовуємо бета-редукцію, підстановку значень аргументів $\rightarrow (2 + 1)$ – застосовуємо дельта-редукцію, тобто, обчислення функції від аргументів $\rightarrow 3$

Розглянемо функцію від двох аргументів – суму x і y :

$(\lambda x \ y. x + y)$

$(\lambda x \ y. x + y) 1$ - застосування бета-редукції, замінивши x на $1 \rightarrow (\lambda y. 1 + y)$

Отримали іншу функцію, від меншої кількості змінних.

Запишемо: $(\lambda x \ y. x + y)$ еквівалентно $(\lambda x. (\lambda y. x + y))$.

Саме цей прийом називається *каррінг* (currying) - звуження простору змінних.

Основні особливості функціонального програмування

1. Виділяється базовий набір функцій.
2. Нові функції конструюються з базових в допомогою λ -абстрактора, операцій аплікації (застосування функції до аргументу), композиції функцій і рекурсії.

3. Функції обчислюються шляхом *підстановки аргументів замість змінних*, зазначених у λ -абстракторі.

4. Кожен вираз єдиним чином визначається значенням його складових частин (тобто не залежить від контексту, відсутній побічний ефект, коли значення аргументу змінюється після обчислення функції).

5. Складні структури даних можуть виступати як одиниці даних.

6. У функціональному програмуванні відсутній оператор присвоювання. Функції визначаються за допомогою λ -абстрактора, при цьому вони можуть бути іменованими чи неіменованими. Значення функції обчислюються за допомогою підстановки (результатом не обов'язково є число, це може бути нова функція), наприклад,

$g = \lambda x . 2 * x + 3$ (функції присвоєно ім'я g).

$g(10)$ – обчислюється значення g : $2 * 10 + 3 = 23$.

Та ж функція, але неіменована:

$(\lambda x . 2 * x + 3)(10)$ обчислює те ж саме значення.

Підстановка функції замість аргументу: $(\lambda x . 2 * x + 3)(5 * y)$ дає нову функцію: $2 * 5 * y + 3$. Ця операція називається *аплікацією* функцій.

7. У функціональному програмуванні використовується умовний вираз виду:

$if <вираз> then <вираз> else <вираз>;$

$if <вираз> then <вираз>.$

Наприклад:

$h = \lambda x . if (x > 10) then g(x) else f(x)$

$f = \lambda x . x + y$

$g = \lambda x . 2 * x + 3$

Тоді $h(5)$ – це функція $f(5)$, тобто $5 + y$.

8. Можна визначити композицію функцій, використовуючи λ -абстрактор по іменах функцій: $\lambda f . (\lambda g (\lambda x . F(g(x))))$.

Тоді при підстановці в це вираження визначених раніше функцій отримаємо:

$(\lambda f. (\lambda g. (\lambda x. F(g(x)))))(f, g)$ – отримаємо нову функцію $\lambda x. 2 * x + 3 + y$.

9. *Оператор циклу відсутній в строго функціональній мові. Замість нього застосовується рекурсія.*

Рекурсія буває двох видів: висхідна і спадна.

Розглянемо їх на прикладі функції факторіалу:

$$F(x) = x! = \begin{cases} 1, & \text{якщо } x = 0, \\ x * f(x-1), & \text{якщо } x > 0. \end{cases}$$

Функціональна програма:

$Factorial = \lambda x. \text{if } x = 0 \text{ then } 1 \text{ else } x * Factorial(x-1).$

Процес обчислення:

$Factorial(3) = 3 * Factorial(2) = 3 * 2 * Factorial(1) = 3 * 2 * 1 * Factorial(0) = 3 * 2 * 1 * 1 = 6.$

Це *спадна рекурсія*. Проміжні значення заносяться в стек, а коли буде обчислено термінальне значення (умова закінчення), то результати вибираються з стека і обчислюється остаточний результат.

Висхідна рекурсія використовує накопичувальний параметр. Вона визначається за допомогою двох функцій. Одна функція задає початкове значення накопичувального параметра:

$F1 = \lambda x. fact(x, 1),$

інша функція рекурсивно обчислює факторіал:

$fact(x, y) = \lambda x. \lambda y. \text{if } x = 0 \text{ then } y \text{ else } fact(x-1, y * x).$

Для обчислення $3!$ потрібно викликати функцію $F1(3)$, $F1$ присвоїть накопичувальному параметру y значення 1 і викличе функцію $fact(x, y)$, передавши в неї значення $x = 3, y = 1$:

$F1(3) = fact(3, 1) = fact(2, 1 * 3) = fact(1, 3 * 2) = fact(0, 6 * 1) = 6.$

2) *Рекурсивна* програма для реалізації арифметичної функції повернення n -го члена геометричної прогресії з першим членом b та знаменником q :

$\text{geo}(b, q, n) := \text{if } n=1 \text{ then } b \text{ else } \text{geo}(b*q, q, n-1)$ – спадна рекурсія;

$\text{geo}(3, 2, 5) = \text{geo}(3*2, 2, 4) = \text{geo}(6*2, 2, 3) = \text{geo}(12*2, 2, 2) = \text{geo}(24*2, 2, 1) = 48$.

Висхідна рекурсія для даної функції не є доречною.

Функції обробки списків

Елементарними функціями обробки списків є такі:

1. Функція ***Car*** (X) вибирає зі списку **1-й** елемент.

Аргумент X може бути тільки списком, а результат - 1-й елемент – може бути атомом або списком. Функція ***Car*** (X) для аргументу-атома не визначена.

2. Функція ***Cdr*** (X) від списку є список, отриманий відкиданням 1-го члена. Функція ***Cdr*** (X) повертає вказівник на другий елемент списку, тобто на **хвіст списку**. ***Cdr*** (X) для атома не визначена.

Car і ***Cdr*** – це функції-селектори. Використовуючи аплікацію цієї функції до себе: ***Cdr*** ($\dots$ (***Cdr*** (X))), можна отримати вказівник на другий, третій і т.д. елементи списку. Таким чином можна отримати доступ до будь-якого елемента списку. Наприклад, вираз ***Car*** (***Cdr*** (***Cdr*** (X))) матиме значенням 3-й елемент списку.

3. Функція ***Cons*** (X, Y) – конструктор для створення списків. Він додає аргумент X в голову списку Y . Аргумент X може бути списком або атомом, Y - обов'язково список.

Результат - список, отриманий з Y додаванням X в якості першого елемента. Таким чином, довжина результату дорівнює $n + 1$, якщо Y - список довжини n . Основний концепт: ***Cons*** (***Car*** (X), ***Cdr*** (X)) = X .

4. Функція ***Atom*** (X) – предикат. Він повертає T , якщо X є атомом, і F , якщо X – список. Служить для розпізнавання списків і атомів.

Наприклад, у функції $f(x) := \text{if } \text{atom}(x) \text{ then } \text{Nil} \text{ else } \text{car}(x)$, – $f(x)$ видає 1-й елемент x , якщо x - список, і видає Nil , якщо x - атом.

Таким чином, якщо ***Car*** (x) - частково-визначена функція, то $f(x)$ - повсюди визначена функція.

5. Функція **EQ** (x, y) – предикат, який порівнює атоми. Він дорівнює T , якщо два атома рівні, F – якщо атоми різні або один з аргументів є списком. Якщо X, Y – обидва списки, то результат не визначений.

6. Функція **NULL** (X) – предикат, який повертає T , якщо аргумент-список порожній, і F , якщо список не порожній. Для атома – не визначений.

Тільки для числових атомів можна використовувати операції: $+$, $-$, $*$, div , mod , і предикати $x \leq y$, $x = y$ і *m.i.*.

Приклади.

Написати рекурсивну програму для реалізації функції:

1) *Вставити список Y в список X, починаючи із заданої позиції n*

$Insert(X, Y, n) := \text{if } n=1 \text{ then } \text{concat}(Y, X) \text{ else } \text{cons}(\text{car}(X), \text{Insert}(\text{cdr}(X), Y, n-1))$,
де $\text{concat}(Y, X)$ – функція конкатенації двох списків¹.

Приклад, $Insert((a\ b\ c\ d), (1\ 2\ 3), 3) = \text{cons}(\text{car}((a\ b\ c\ d)), \text{Insert}(\text{cdr}((a\ b\ c\ d)), (1\ 2\ 3), n-1)) = \text{cons}(a, \text{Insert}(\text{cdr}((b\ c\ d)), (1\ 2\ 3), 2)) = \text{cons}(a, \text{cons}(\text{car}((b\ c\ d)), \text{Insert}(\text{cdr}((b\ c\ d)), (1\ 2\ 3), n-1))) = \text{cons}(a, \text{cons}(b, \text{Insert}(\text{cdr}((c\ d)), (1\ 2\ 3), 1))) = \text{cons}(a, \text{cons}(b, \text{concat}((1\ 2\ 3), (c\ d)))) = \text{cons}(a, \text{cons}(b, (1\ 2\ 3\ c\ d))) = \text{cons}(a, (b\ 1\ 2\ 3\ c\ d)) = (a\ b\ 1\ 2\ 3\ c\ d)$.

¹ $\text{concat}(X, Y) := \text{if } X = \text{Nil} \text{ then } Y \text{ else } \text{cons}(\text{car}(X), \text{concat}(\text{cdr}(X), Y))$,
 $\text{concat}((1\ 2\ 3), (a\ b)) = \text{cons}(1, \text{concat}((2\ 3), (a\ b))) = \text{cons}(1, \text{cons}(2, \text{concat}((3), (a\ b)))) = \text{cons}(1, \text{cons}(2, \text{cons}(3, \text{concat}(), (a\ b)))) = \text{cons}(1, \text{cons}(2, \text{cons}(3, (a\ b)))) = (1\ 2\ 3\ a\ b)$;

2) *Знайти максимум двох двійкових чисел однакової розрядності*

$\text{MAXL}(X, Y) := \text{if } X = \text{Nil} \text{ then Nil else if } \text{car}(X) < \text{car}(Y) \text{ then } Y \text{ else } \text{cons}(\text{car}(X), \text{MAXL}(\text{cdr}(X), \text{cdr}(Y)))$;

Приклад:

$\text{MAXL}(1101, 1110) = \text{cons}(\text{car}(1101), \text{MAXL}(\text{cdr}(1101), \text{cdr}(1110))) = \text{cons}(1, \text{MAXL}(101, (110))) = \text{cons}(1, \text{cons}(\text{car}(101), \text{MAXL}(\text{cdr}(101), \text{cdr}(110)))) = \text{cons}(1, \text{cons}(1, \text{MAXL}((01), (10)))) = \text{cons}(1, \text{cons}(1, Y)) = \text{cons}(1, \text{cons}(1, (10))) = (1\ 1\ 1\ 0)$;

$\text{MAXL}(110, 110) = \text{cons}(\text{car}(110), \text{MAXL}(\text{cdr}(110), \text{cdr}(110))) =$

$=\text{cons}(1, \text{MAXL}(10), (10)) = \text{cons}(1, \text{cons}(\text{car}(10), \text{MAXL}(\text{cdr}(10), \text{cdr}(10)))) =$
 $=\text{cons}(1, \text{cons}(1, \text{MAXL}((0), (0)))) = \text{cons}(1, \text{cons}(1, \text{cons}(0, \text{MAXL}((0), (0))))) =$
 $\text{cons}(1, \text{cons}(1, \text{cons}(0, ()))) = (110) - \text{однакові}.$

Завдання 4.4.1. Написати *рекурсивну* програму для реалізації *арифметичних* функцій у вигляді *спадної* та *висхідної* рекурсії:

- 1) Ціле від ділення двох чисел x на y – функція $\text{div}(x, y)$;
- 2) Остача від ділення двох чисел x на y – функція $\text{mod}(x, y)$;
- 3) Сума n членів арифметичної прогресії – функція $\text{suma}(a, d, n)$, де a – це перший член прогресії, d – різниця між попереднім та наступним членами прогресії;
- 4) Добуток двох натуральних чисел через функції додавання та віднімання – функція $\text{mult}(x, y)$ через $\text{plus}(x, y)$ та $\text{minus}(x, y)$;
- 5) Степенева функція x^n – функція $\text{deg}(x, n)$.

Завдання 4.4.2. Написати *рекурсивну* програму для реалізації функції f обробки списку X . Вважати вхідні дані коректними і такими, що не містять помилок. Вид рекурсії – *спадна* або *висхідна* – можна обрати самостійно:

- 1) Знайти максимальний елемент списку – функція $f(X)$;
- 2) Визначити скільки елементів в списку більше заданого числа a – функція $f(X, a)$;
- 3) Визначити добуток додатних чисел списку – функція $f(X)$;
- 4) Визначити суму квадратів елементів числового списку – функція $f(X)$;

В завданнях 5-8 множини змодельовані списком, тобто елементи списку не повторюються:

- 5) Знайти перетин множин – функція $f(X, Y)$;
- 6) Знайти об'єднання множин – функція $f(X, Y)$;
- 7) Знайти відносну різницю множин $X \setminus Y$ – функція $f(X, Y)$;
- 8) Знайти симетричну різницю множин – функція $f(X, Y)$;
- 9) Визначити номер позиції першого входження елемента a в список X – функція $f(X, a)$;

- 10) Визначити кількість входження елемента a в список X – функція $f(X, a)$;
- 11) Вставити елемент a в список X на задану позицію n – функція $f(X, a, n)$;
- 12) Замінити в списку X елемент на певній позиції n на заданий a – функція $f(X, n, a)$;
- 13) Вилучити елемент зі списку X із заданої позиції n – функція $f(X, n)$;
- 14) Вилучити перше входження елемента a зі списку X – функція $f(X, a)$;
- 15) Вилучити всі входження елемента a зі списку X – функція $f(X, a)$;
- 16) Замінити в списку X всі входження елемента a на заданий b – функція $f(X, a, b)$;
- 17) Вставити елемент a у впорядкований за зростанням непорожній список X ; врахувати, що елемент може бути доданим у якості першого або останнього; список може складатися з одного елемента – функція $f(X, a)$;
- 18) Впорядкувати список за спаданням – функція $f(X)$;

До завдань 5-8: побудувати з довільного списку список-множину – написати функцію вилучення однакових елементів списку.

В завданнях 19-22 розглядаються двійкові числа однакової розрядності, що моделюються списками з 0 та 1; нулі на початку числа не вилучаються, наприклад, 001 – трьохрозрядне число 1. А як написати функцію, яка буде порівнювати кількість розрядів двох двійкових чисел і приводити числа до однакової розрядності?

- 19) Порівняти два двійкових числа однакової розрядності – функція $f(X, Y)$;
- 20) Логічний зсув вліво/вправо на задану n кількість розрядів (логічний зсув: число доповнюється нулями – наприклад, зсув вліво на 2 розряди числа 01011 $\rightarrow$ 01100) – функція $f(X, a)$;
- 21) Циклічний зсув вліво/вправо на задану n кількість (циклічний зсув: число доповнюється «виштовхненими» елементами – наприклад,

циклічний зсув вправо на 2 розряди числа 01001 $\rightarrow$ 01010) – функція $f(X, a)$;

- 22) Побітова сума за модулем 2 (без перенесення 1 в старший розряд) двох двійкових чисел однакової розрядності – функція $f(X, Y)$. Написати також функцію суми за модулем 2 двох двійкових чисел однакової розрядності з перенесенням 1 в старші розряди.

Відповіді до розділу 4.

4.1.1.

1) $x_1 + x_2 - 1$:

$$f(x_1, x_2) = h(x_1, x_2 - 1, f(x_1, x_2 - 1)) = x_1 + x_2 - 1$$

2) $x_1(x_2 + 1)$:

$$\begin{aligned} f(x_1, x_2) &= h(x_1, x_2 - 1, f(x_1, x_2 - 1)) = x_1 + f(x_1, x_2 - 1) = \\ &= x_1 + h(x_1, x_2 - 2, f(x_1, x_2 - 2)) = x_1 + x_1 + f(x_1, x_2 - 2) = \dots \\ &\dots = x_1 + \dots + x_1 + f(x_1, 0) = x_1 + \dots + x_1 + x_1 = x_1(x_2 + 1) \end{aligned}$$

3) $x_1 + \sum_{i=0}^{x_2-1} (x_2 - i)$ (сума від 0 до $x_2 - 1$):

$$\begin{aligned} f(x_1, x_2) &= h(x_1, x_2 - 1, f(x_1, x_2 - 1)) = x_2 + f(x_1, x_2 - 1) = \\ &= x_2 + h(x_1, x_2 - 2, f(x_1, x_2 - 2)) = x_2 + x_2 - 1 + f(x_1, x_2 - 2) = \dots \\ &\dots = x_2 + (x_2 - 1) + \dots + (x_2 - x_2 + 1) + f(x_1, 0) = \dots \end{aligned}$$

4) $2^{(x_1 * x_2)}$

4.1.2.

Вказівки: 2-а та 3-я формули визначають схему рекурсії з підстановкою $(x-1)$ замість першого аргументу, а для випадка, коли $x-1=0$, використовується перша формула

4.1.3. Побудувати об'єкти, що відповідають геделевим номерам:

- 1944 відповідає $()$
- 816293376 відповідає $\neg x_1 (2^9 3^{13})$
- 47 відповідає a_5
- 585 відповідає f_2^3

4.1.4. Знайти геделев номер виразу (записати формулу):

Для виразу $\exists x_1 A_1^1(x_1)$ – вказівки: $\exists x_1 A_1^1(x_1) = \neg \forall x_1 \neg A_1^1(x_1) = \neg(x_1) \neg A_1^1(x_1)$

4.2.1. 1) $\parallel$; 2) 2 ;

4.2.2.

1) Порівняти два слова в унарній системі числення:

|||*|| → 1 |||*||| → 2 |||*||| → 0

	q0	q1	q2	q3	q4
	λq1R	q1R	λq3L	q3L	2!
*	λq4R	*q1R	1!	*q3L	
λ		λq2L		λq0R	0!

2) Перевірити, чи відсортовано слово в алфавіті {a, b}:

aaa → Yes bbb → Yes aabb → Yes

	q0	q1
a	λq0R	N!
b	λq1R	λq1R
λ	Y!	Y!

Впорядкувати слово:

aaa → aaa bbb → bbb aabb → aabb

	q0	q1	q2	q3
a	aq0R	bq2L	aq3R	
b	bq1R	bq1R	bq2L	aq1R або a q0 R
λ	!	!	λq3R	

3) Обчислення функції $f(x)=2x$ в унарній системі числення:

|||* → |||*|||

	q0	q1	q2	q3
	wq1R	q1R		q3L
*	!	*q1R		*q3L
λ		q2R	q3L	
w				q0R

4) Обчислення функції $f(x)=x/2$ з остачею в унарній системі числення:

||||| $\rightarrow$ |||||*||

|||||| $\rightarrow$ |||||*|||*

	q0	q1	q2	q3	q4	q5	q6
	q0R	q1L	q3R	wq4R	q4R	q5R	
*		*q1L	!	*q5R	*q4R		
λ	*q1L	λ q2R	!		q1L	*q2R	!
w		q2R					

5) Переробити двійкове число у від'ємне (додатковий код).

Інверсія +1 (не виходимо за розрядну сітку):

	q0	q1
0	1q0R	1!
1	0q0R	0q1L
λ	λ q1L	!

Останні нулі і остання 1 залишаються, а потім, все інвертується до початку, рухаючись праворуч наліво:

	q0	q1	q2
0	0q0R	0q1L	1q2L
1	1q0R	1q2L	0q2L
λ	λ q1L	!	!

6) Перевірити входження заданого символу в слово в алфавіті {a, b, c, *}:
 $c^* abcb$ $\rightarrow$ Yes $c^* abaab$ $\rightarrow$ No

	q0	q1	q2	q3
a	λ q1R	Y!	aq2R	aq3R
b	λ q2R	bq1R	Y!	bq3R
c	λ q3R	cq1R	cq2R	Y!
*		λ q1R	λ q2R	λ q3R
λ		N!	N!	N!

7) Підрахувати кількість букв в слові ($aabba \rightarrow aabba|||||$):

	q0	q1	q2	q3	q4
a	*q1R	aq1R	aq2L	aq3R	aq4L
b	*q3R	bq1R	bq2L	bq3R	bq4L
	!	q1R	q2L	q3R	q4L
*			aq0R		bq0R
λ		q2L		q4L	

8) Підрахувати кількість заданої букви в слові:

b*abab → b*abab||

	q0	q1	q2	q3	q4	q5	q6
a	aq1R	\$q2R	aq2R	aq3L	aq4R	aq5R	aq6L
b	bq4R	bq1R	bq2R	bq3L	\$q5R	bq5R	bq6L
		!	q2R	q3L	!	q5R	q6L
*		*q1R			*q4R		
\$				aq1R			bq1R
λ			q3L			q6L	

10) Заміна букви із зазначеного місця на задану:

b|||abab → abbb

	q0	q1	q2	q3	q4	q5	q6	q7
a	aq0R	Aq2L		λ q4R	aq6L	aq7L	a!	b!
b	bq0R	Bq2L		λ q5R	bq6L	bq7L	a!	b!
	*q1R	q1R	q2L					
*			*q0R	*q3L	λ q4R	λ q5R		
A	Aq3L	Aq1R	Aq2L		aq4R	aq5R		
B	Bq3L	Bq1R	Bq2L		bq4R	bq5R		
λ								

13) Порівняти два двійкових числа однакової розрядності, розділовий знак *: вхідне слово 1010*1001; вихідне 0 – однакові, 1/2 – перше/друге більше:

11101*	10110	→	1
10101*	10111	→	2
10101*	10101	→	0

	q0	q1	q2	q3	q4	q5
0	$\lambda q1R$	$0q1R$	$0q2R$	$*q5L$	$1!$	$0q5L$
1	$\lambda q2R$	$1q1R$	$1q2R$	$2!$	$*q5L$	$1q5L$
*	$0!$	$*q3R$	$*q4R$	$*q3R$	$*q4R$	$*q5L$
λ						$\lambda q0R$

4.3.1. 1) |; 2) ||; 3) |;

4.3.2.

4) Переробити двійкове число у від'ємне (додатковий код).

110 $\rightarrow$ 010

1101 $\rightarrow$ 0011

Інверсія +1 без виходу за розрядну сітку:

1	$a0 \rightarrow 1a$
2	$a1 \rightarrow 0a$
3	$a \rightarrow b$
4	$0b \rightarrow. 1$
5	$1b \rightarrow b0$
6	$\lambda b \rightarrow. \lambda$
7	$0 \rightarrow a0$
8	$1 \rightarrow b1$

Останні нулі і остання 1 залишаються, а потім, все інвертувати до початку, рухаючись праворуч наліво ($x \in \{0, 1\}$):

1	$0a \rightarrow a0$
2	$1a \rightarrow b1$
3	$0b \rightarrow 1b$
4	$1b \rightarrow 0b$
5	$b \rightarrow. \lambda$
6	$x\lambda \rightarrow xa$

5) Перевірити входження заданого символу в слово ($\chi, \gamma \in \{a, \dots, z\}$):

c* abcbc $\rightarrow$ Yes

c* abaab $\rightarrow$ No

1	$\chi^* \chi$	$\rightarrow.$	yes
2	$\chi^* \gamma$	$\rightarrow$	χ^*
3	χ^*	$\rightarrow.$	no

6) Підрахувати кількість букв в слові:

1	$^* \chi$	$\rightarrow$	$\chi ^*$
2	$ \chi$	$\rightarrow$	$\chi $
3	*	$\rightarrow.$	λ
4	χ	$\rightarrow$	$^* \chi$

$aab \rightarrow (3) ^*aab \rightarrow (1) a|^*ab \rightarrow (1) a|a|^*b \rightarrow (1) a|a|b|^* \rightarrow (2) aa||b|^* \rightarrow (2) aa|b||^* \rightarrow (2) aab|||^* \rightarrow .() aab|||$

8) Заміна всіх входжень заданої букви на іншу (χ , γ та τ – літери з алфавіту $\{a, \dots, z\}$):

1	$\chi^* \gamma^* \chi$	$\rightarrow$	$\gamma \chi^* \gamma^*$
2	$\chi^* \gamma^* \tau$	$\rightarrow$	$\tau \chi^* \gamma^*$
3	$\chi^* \gamma^*$	$\rightarrow.$	λ

$c^*z^*abcbdz \rightarrow (2) ac^*z^*bcbdz \rightarrow (2) abc^*z^*cbdz \rightarrow (1) abzc^*z^*bdz \rightarrow (2) abzbc^*z^*dz \rightarrow (2) abzbdc^*z^*zc \rightarrow (2) abzbdzc^*z^*c \rightarrow (1) abzbdzzc^*z^* \rightarrow .(3) abzbdzz$
 $c^*z^*abbdz \rightarrow (2) ac^*z^*bbdz \rightarrow (2) abc^*z^*bdz \rightarrow (2) abbc^*z^*dz \rightarrow (2) abbdzc^*z^*z \rightarrow (2) abbdzc^*z^* \rightarrow .(3) abbdz$

15) Вилучити літери, що повторюються в слові з алфавіту $\{a, \dots, z\}$:

1	$\beta \chi \chi$	$\rightarrow$	$\beta \chi$
2	$\beta \chi \gamma$	$\rightarrow$	$\gamma \beta \chi$
3	$\beta \chi$	$\rightarrow$	λ
4	$\alpha \chi \gamma$	$\rightarrow$	$\chi \alpha \beta \chi \gamma$
5	$\chi \alpha$	$\rightarrow.$	λ
6	χ	$\rightarrow$	$\alpha \chi$

$abbdacad \rightarrow (6) \alpha abbdacad \rightarrow (4) a\alpha bbdacad \rightarrow (2) a\alpha b\beta abdacad \rightarrow (2)$
 $a\alpha b\beta\beta adacac \rightarrow (2) a\alpha bbd\beta aacac \rightarrow (1) a\alpha bbd\beta acac \rightarrow (2) \quad a\alpha bbd\beta\beta aad \rightarrow (1)$
 $a\alpha bbd\beta\beta ad \rightarrow (2) a\alpha bbd\beta\beta a \rightarrow (3) a\alpha bbd\beta\beta \rightarrow (4) ab\alpha\beta\beta bddcd \rightarrow (1) ab\alpha\beta\beta bddcd \rightarrow (2)$
 $ab\alpha d\beta\beta cd \rightarrow (2) ab\alpha d\beta\beta d \rightarrow (2) \quad ab\alpha d\beta\beta \rightarrow (3) ab\alpha d\beta\beta \rightarrow (4) ab\alpha d\beta\beta \rightarrow (4) ab\alpha d\beta\beta \rightarrow (4)$
 $\rightarrow (2) ab\alpha d\beta\beta dd \rightarrow (1) ab\alpha d\beta\beta d \rightarrow (3) ab\alpha d\beta\beta \rightarrow (4) ab\alpha d\beta\beta \rightarrow (3) ab\alpha d\beta\beta \rightarrow (5)$
 $abdc$

4.4.1.

1) Ціле від ділення двох чисел x на y :

↓ **спадна рекурсія:**

$$\text{div}(x, y) := \text{if } x < y \text{ then } 0 \text{ else } 1 + \text{div}(x - y, y),$$
$$\text{div}(7, 3) = 1 + \text{div}(4, 3) = 1 + 1 + \text{div}(1, 3) = 1 + 1 + 0 = 2;$$

↑ висхідна рекурсія:

$$\operatorname{div} V(x, y) := dv(x, y, 0),$$

div (x, y, z):= if x<y then z else div(x-y, y, z+1),

$$\text{div}V(7,3) = \text{dv}(7,3,0) = \text{dv}(4,3,1) = \text{dv}(1,3,1+1) = 2;$$

2) Остача від ділення двох чисел x на y :

$$\text{mod}(x, y) := \text{if } x < y \text{ then } x \text{ else } \text{mod}(x-y, y),$$
$$\text{mod } (7, 3) = \text{mod } (4, 3) = \text{mod } (1, 3) = 1;$$

висхідна рекурсія для даної функції не є доречною;

3) Сума n членів арифметичної прогресії – функція $\text{suma}(a, d, n)$, де a – це перший член прогресії, d – різниця між попереднім та наступним членами прогресії;

$$\text{sumM } (a, d, n) := \text{sum } (a, d, n, a),$$
$$\text{sum}(a, d, n, z) := \text{if } n=1 \text{ then } z \text{ else } d+a+\text{sum}(a+d, d, n-1, z),$$
$$\begin{aligned} \text{sum}(3, 2, 5, 3) &= 2 + 3 + \text{sum}(3+2, 2, 4, 3) = 5 + 2 + 5 + \text{sum}(5+2, 2, 3, 3) = \\ &= 12 + 2 + 7 + \text{sum}(7+2, 2, 2, 3) = 21 + 2 + 9 + \text{sum}(11, 2, 1, 3) = 32 + 3 = 35; \end{aligned}$$

спадна рекурсія для даної функції не є доречною;

4) Добуток двох натуральних чисел через функції додавання `plus (x, y)` – “+” та віднімання `minus (x, y)` – “-”:

$$\text{mult}(x, y) := \text{if } y=1 \text{ then } x \text{ else plus}(x, \text{mult}(x, \text{minus}(y, 1))) = \\ \text{if } y=1 \text{ then } x \text{ else } x + \text{mult}(x, y-1),$$
$$\text{mult}(2, 3) = 2 + \text{mult}(2, 2) = 2 + 2 + \text{mult}(2, 1) = 2 + 2 + 2 = 6;$$

5) Степенева функція x^n :

↓ **спадна рекурсія:**

$\text{deg}(x, n) := \text{if } n=0 \text{ then } 1 \text{ else } x * \text{deg}(x, n-1),$

$\text{deg}(5, 3) = 5 * \text{deg}(5, 2) = 5 * 5 * \text{deg}(5, 1) = 25 * 5 * \text{deg}(5, 0) = 125 * 1 = 125;$

↑ **висхідна рекурсія:**

$\text{degG}(x, n) := \text{dg}(x, n, 1),$

$\text{dg}(x, n, z) := \text{if } n=0 \text{ then } z \text{ else } \text{dg}(x, n-1, z*x),$

$\text{degG}(5, 3) = \text{dg}(5, 3, 1) = \text{dg}(5, 2, 1*5) = \text{dg}(5, 1, 5*5) = \text{dg}(5, 0, 25*5) = 125;$

4.4.2.

1) Знайти максимальний елемент списку:

$\text{maxX}(X) := \text{max}(\text{cdr}(X), \text{car}(X)),$

$\text{max}(Y, a) := \text{if } Y = \text{Nil} \text{ then } a \text{ else if } \text{car}(Y) > a \text{ then } \text{max}(\text{cdr}(Y), \text{car}(Y)) \text{ else } \text{max}(\text{cdr}(Y), a),$

$\text{maxX}(2\ 5\ 7\ 3\ 4) = \text{max}(5\ 7\ 3\ 4, 2) = \text{max}(7\ 3\ 4, 5) = \text{max}(3\ 4, 7) = \text{max}(4, 7) = \text{max}(\text{Nil}, 7) = 7;$

2) Визначити скільки елементів в списку більше заданого числа a:

↑ **висхідна рекурсія:**

$\text{more}(X, a) := \text{mr}(X, a, 0),$

$\text{mr}(X, a, b) := \text{if } X = \text{Nil} \text{ then } b \text{ else if } \text{car}(X) > a \text{ then } \text{mr}(\text{cdr}(X), a, b+1) \text{ else } \text{mr}(\text{cdr}(X), a, b),$

$\text{more}(6\ 3\ 4\ 5\ 2, 3) = \text{mr}(6\ 3\ 4\ 5\ 2, 3, 0) = \text{mr}(3\ 4\ 5\ 2, 3, 1) = \text{mr}(4\ 5\ 2, 3, 1) = \text{mr}(5\ 2, 3, 2) = \text{mr}(2, 3, 3) = \text{mr}(\text{nil}, 3, 3) = 3;$

3) Визначити добуток додатних чисел списку:

↓ **спадна рекурсія:**

$\text{multpos}(X) := \text{if } X = \text{Nil} \text{ then } 1 \text{ else if } \text{car}(X) > 0 \text{ then } \text{multpos}(\text{cdr}(X)) * \text{car}(X) \text{ else } \text{multpos}(\text{cdr}(X)),$

$\text{multpos}(-1\ 3\ 0\ -4\ 5) = \text{multpos}(3\ 0\ -4\ 5) = 3 * \text{multpos}(0\ -4\ 5) = 3 * \text{multpos}(-4\ 5) = 3 * \text{multpos}(5) = 3 * 5 * \text{multpos}() = 15 * 1 = 15;$

4) Визначити суму квадратів елементів числового списку:

↓ **спадна рекурсія:**

$\text{sumsquares}(X) := \text{if } X = \text{Nil} \text{ then } 0 \text{ else } \text{car}(x) * \text{car}(X) + \text{sumsquares}(\text{cdr}(x)),$

sumsquares (-1 3 0 -4 5) = (-1)*(-1)+ sumsquares (3 0 -4 5) = 1+ 3*3 +
 +sumsquares (0 -4 5) = 10 + 0*0 + sumsquares (-4 5) = 10 + (-4)*(-4) +
 +sumsquares (5) = 26 + 5*5+ sumsquares (Nil) = 51 + 0 = 51;

↑ **висхідна рекурсія:**

sumsq (X) := smsq (X, 0),

smsq (X, s) := if X=Nil then s else smsq (cdr (X), car (X) * car(X) +s),

sumsq (-1 3 0 -4 5) = smsq (-1 3 0 -4 5, 0) = smsq (3 0 -4 5, (-1)*(-1)+0) =
 smsq (0 -4 5, 3*3+1) = smsq (-4 5, 0*0+10) = smsq (5, (-4)*(-4)+10) =
 = smsq (Nil, 5*5+26) = 51;

Приклади 5-8 оперують зі списками, які моделюють множини, тобто не містять елементів, що дублюються. Для приведення довільного списку до вигляду множин, треба вилючити всі повтори елементів:

set (X) := if X= Nil then Nil else if in (car (X), cdr (X)) then set (cdr (X)) else cons (car (X), set (cdr (X))),

set (s d s a a f g s c f) = set (d s a a f g s c f) = cons (d, set (s a a f g s c f)) =
 cons (d, set (a a f g s c f)) = cons (d, set (a f g s c f)) = cons (d, cons (a, set (f g
 s c f))) = cons (d, cons (a, set (g s c f))) = cons (d, cons (a, cons (g, set (s c f)))) =
 = cons (d, cons (a, cons (g, cons (s, set (c f))))) = cons (d, cons (a, cons (g, cons
 (s, cons (c, set (f))))) = cons (d, cons (a, cons (g, cons (s, cons (c, cons (f, set
 (Nil))))) = = cons (d, cons (a, cons (g, cons (s, cons (c, cons (f, Nil))))) =
 = (d a g s c f).

В програмах використовується функція

in (a, X) := if X=Nil then F else if a=car(X) then T else in (a, cdr (X)),

що повертає T або F в залежності від того, входить елемент a в список X або ні:

in (b, (a b c d)) = in (b, (b c d)) = T,

in (b, (a c d)) = in (b, (c d)) = in (b, (d)) = in (b, Nil) = F.

5) Знайти перетин множин:

↓ **спадна рекурсія:**

intersection (X, Y) := if X=Nil then Nil else if in (car (X), Y) then cons(car(X),

intersection (cdr (X), Y)) else intersection (cdr (X), Y),

intersection ((a d g e r), (a f c g b)) =cons(a, intersection ((d g e r),(a f c g b)))=

= cons (a, intersection ((g e r), (a f c g b))) = cons (a, cons (g, intersection ((e r), (a f c g b)))) = cons (a, cons (g, intersection ((r), (a f c g b)))) = cons (a, cons (g, intersection ((), (a f c g b)))) = cons (a, cons (g, Nil)) = (a g);

↑ **висхідна рекурсія:**

intersect (X, Y) := intset (X, Y, Nil),

intset (X, Y, Z) := if X=Nil then Z else if in (car (X), Y) then intrset (cdr (X), Y, cons(car(X), Z)) else intset (cdr (X), Y, Z),

intersect ((a d g e r), (a f c g b)) = intset ((a d g e r), (a f c g b), Nil) = intset ((d g e r), (a f c g b), cons (a, Nil)) = intset ((g e r), (a f c g b), (a)) = intset ((e r), (a f c g b), cons (g, (a))) = intset ((r), (a f c g b), (g, a)) = intset ((), (a f c g b), (g a)) = (g a);

6) Знайти об'єднання множин:

union (X, Y) := if X=Nil then Y else if in (car (X), Y) union (cdr (X), Y) then else union (cdr (X), cons(car(X), Y)),

union ((a d g e r), (a f c g b)) = union ((d g e r), (a f c g b)) = union ((g e r), cons (d, (a f c g b))) = union ((e r), (d a f c g b)) = union ((r), cons (e, (d a f c g b))) = union ((), cons (r, (e d a f c g b))) = (r e d a f c g b);

7) Знайти відносну різницю множин $X \setminus Y$:

relative (X, Y) := if X=Nil then Nil else if in (car (X), Y) union (cdr (X), Y) then else cons(car(X), union (cdr (X), Y)),

relative ((a d g e r), (a f c g b)) = relative ((d g e r), (a f c g b)) = cons (d, relative ((g e r), (a f c g b))) = cons (d, relative ((e r), (a f c g b))) = cons (e, cons (d, relative ((r), (a f c g b)))) = cons (r, cons (e, cons (d, relative ((), (a f c g b)))))) = cons (r, cons (e, cons (d, Nil))) = (r e d);

8) Знайти симетричну різницю множин $X \oplus Y$:

symmetric (X, Y) := union (relative (X, Y), relative (Y, X)),

or symmetric (X, Y) := relative (union (X, Y), intersection (X, Y));

9) Визначити номер позиції першого входження елемента a в список X:

number1(X, a) := if not (in (a, X)) then 0 else if car(X)=a then 1 else 1 + number1 (cdr (X), a),

number1((1 4 6 3 7 3), 3) = 1 + number1((4 6 3 7 3), 3) = 1 + 1 + number1((6 3 7 3), 3) = 2 + 1 + number1((3 7 3), 3) = 3 + 1 = 4;

10) Визначити кількість входження елемента a в список X:

quant (X, a) := if X=Nil then 0 else if car(X)=a then 1 + quant (cdr (X), a) else
quant (cdr (X), a),
quant ((1 4 6 3 7 3), 3) = quant ((4 6 3 7 3), 3) = quant ((6 3 7 3), 3) = quant ((3
7 3), 3) = 1 + quant ((7 3), 3) = 1 + quant ((3), 3) = 1 + 1 + quant ((), 3) =
= 2 + 0 = 2;

11) Вставити елемент a в список X на задану позицію n:

! if X=Nil then Nil [помилка у вхідних даних – номер більше за довжину
списку більше ніж на 1 – поверне старий список; якщо тільки на 1, то
елемент додається останнім в список]:

insertN (X, a, n) := if X=Nil then Nil else if n=1 then cons (a, X) else
cons(car(X), insertN(cdr (X), a, n-1),

insertN ((s d t w e), g, 3) = cons (s, insertN ((d t w e), g, 2)) = cons (s, cons
(d, insertN ((t w e), g, 1))) = cons (s, cons (d, cons (g, (t w e)))) = (s d g t w e),
insertN ((s d t), g, 4) = cons (s, insertN ((d t), g, 3)) = cons (s, cons (d, insertN
((t), g, 2))) = cons (s, cons (d, cons (t, insertN ((), g, 1)))) = cons (s, cons (d,
cons (t, cons (g, ()))))) = (s d t g),

insertN ((s d t), g, 5) = cons (s, insertN ((d t), g, 4)) = cons (s, cons (d, insertN((t),
g, 3))) = cons (s, cons (d, cons (t, insertN ((), g, 2)))) = cons (s, cons (d, cons(t,
Nil))) = (s d t);

12) Замінити в списку X елемент на певній позиції n на заданий a
(номер позиції не може бути більше за довжину списку):

replaseN (X, n, a) := if n=1 then cons (a, cdr(X)) else cons(car(X),
replaseN(cdr(X), a, n-1),

replaseN ((s d t w e), g, 3) = cons (s, replaseN ((d t w e), g, 2)) = cons (s, cons
(d, replaseN ((t w e), g, 1))) = cons (s, cons (d, cons (g, (w e)))) = (s d g w e);

13) Вилучити елемент зі списку X із заданої позиції n (номер позиції
не може бути більше за довжину списку):

↓ **спадна рекурсія** частіше більш лаконічна і зрозуміла:

deleteN (X, n) := if n=1 then cdr (X) else cons (car (X), deleteN (cdr (X), n-1)),
deleteN ((s d r t g), 3) = cons (s, deleteN ((d r t g), 2)) = cons (s, cons (d, deleteN
((r t g), 1))) = cons (s, cons (d, (t g))) = (s d t g);

↑ **висхідна рекурсія** потребує більше додаткових функцій:

delete (X, n) := delN (X, n, Nil),

$\text{cons} (r, \text{cons} (c, \text{cons} (e, \text{replacE} ((), f, c)))))) = \text{cons} (s, \text{cons} (d, \text{cons} (c, \text{cons} (r, \text{cons} (c, \text{cons} (e, \text{Nil})))))) = (s\ d\ c\ r\ c\ e),$
 $\text{replacE} ((s\ d\ r\ e), f, c) = \text{cons} (s, \text{replacE} ((d\ r\ e), f, c)) = \text{cons} (s, \text{cons} (d, \text{replacE} ((r\ e), f, c))) = \text{cons} (s, \text{cons} (d, \text{cons} (r, \text{replacE} ((e), f, c)))) = \text{cons} (s, \text{cons} (d, \text{cons} (r, \text{cons} (e, \text{replacE} ((), f, c)))) = \text{cons} (s, \text{cons} (d, \text{cons} (r, \text{cons} (e, \text{Nil})))) = (s\ d\ r\ e);$

17) Вставити елемент *a* у впорядкований за зростанням непорожній список *X*:

$\text{insertE} (X, a) := \text{if } X = \text{Nil} \text{ then } (a) \text{ else if } \text{car} (X) < a \text{ then } \text{cons} (\text{car} (X), \text{insert} (\text{cdr} (X), a)) \text{ else } \text{cons} (a, X),$

$\text{insertE} ((2\ 4\ 6\ 7\ 8), 5) = \text{cons} (2, \text{insertE} ((4\ 6\ 7\ 8), 5)) = \text{cons} (2, \text{cons} (4, \text{insertE} ((6\ 7\ 8), 5))) = \text{cons} (2, \text{cons} (4, \text{cons} (5, (6\ 7\ 8)))) = (2\ 4\ 5\ 6\ 7\ 8),$

$\text{insertE} ((2\ 4\ 5\ 7\ 8), 5) = \text{cons} (2, \text{insertE} ((4\ 5\ 7\ 8), 5)) = \text{cons} (2, \text{cons} (4, \text{insertE} ((5\ 7\ 8), 5))) = \text{cons} (2, \text{cons} (4, \text{cons} (5, (5\ 7\ 8)))) = (2\ 4\ 5\ 5\ 7\ 8),$

$\text{insertE} ((2\ 4), 5) = \text{cons} (2, \text{insertE} ((4), 5)) = \text{cons} (2, \text{cons} (4, \text{insertE} ((), 5))) = \text{cons} (2, \text{cons} (4, (5))) = (2\ 4\ 5);$

18) Впорядкувати список за спаданням (суперпозиція функцій з завдання 1 – пошук максимального елемента списку, та 14 - вилучення першого входження елемента зі списку):

$\text{desc} (X) := \text{if } X = \text{Nil} \text{ then } \text{Nil} \text{ else } \text{cons} (\text{MaxX} (X), \text{desc} (\text{delete1} (X, \text{MaxX} (X))))),$

$\text{desc} (2\ 4\ 6\ 4\ 3\ 7) = \text{cons} (7, \text{desc} (2\ 4\ 6\ 4\ 3)) = \text{cons} (7, \text{cons} (6, \text{desc} (2\ 4\ 3\ 4))) = \text{cons} (7, \text{cons} (6, \text{cons} (4, \text{desc} (2\ 3\ 4)))) = \text{cons} (7, \text{cons} (6, \text{cons} (4, \text{cons} (4, \text{desc} (2\ 3))))) = \text{cons} (7, \text{cons} (6, \text{cons} (4, \text{cons} (4, \text{cons} (3, \text{desc} (3))))) = \text{cons} (7, \text{cons} (6, \text{cons} (6, \text{cons} (4, \text{cons} (4, \text{cons} (3, \text{cons} (2, \text{desc} ())))))) = \text{cons} (7, \text{cons} (6, \text{cons} (4, \text{cons} (4, \text{cons} (3, \text{cons} (2, \text{Nil})))))) = (7\ 6\ 4\ 4\ 3\ 2);$

Впорядкувати список за спаданням можна аналогічно, використовуючи функцію знаходження мінімального елемента списку;

19) Порівняти два двійкових числа однакової розрядності (повертає результат 0 – однакові, 1 – *X* більше, 2 – *X* менше):

$\text{compare} (X, Y) := \text{if } X = \text{Nil} \text{ then } 0 \text{ else if } \text{car} (X) > \text{car} (Y) \text{ then } 1 \text{ if } \text{car} (X) < \text{car} (Y) \text{ then } 2 \text{ else } \text{compare} (\text{cdr} (X), \text{cdr} (Y)),$

$\text{compare} (110, 110) = \text{compare} (10, 10) = \text{compare} (0, 0) = \text{compare} (\text{Nil}, \text{Nil}) = 0,$

compare (100, 110) = compare (00, 10) = 2;

Через те, що X та Y мають однакову розрядність не має сенсу перевіряти на Nil обидва списки, вони зменшуються симетрично;

20) Логічний зсув вліво на задану n кількість розрядів двійкового числа, змодельованого списком з 0 та 1:

displaceLL (X, n) := dispLL (X, n, Nil),

dispLL (X, n, Z) := if X=Nil then concat (X, Z) else dispLL (cdr (X), n-1, cons (0, Z)),

displaceLL (01110, 2) = dispLL (01110, 2, Nil) = dispLL (1110, 1, cons (0, Nil)) = dispLL (110, 0, cons (0, (0))) = concat (110, 00) = 11000;

21) Циклічний зсув вправо на задану n кількість розрядів двійкового числа, змодельованого списком з 0 та 1:

displaceCR (X, n) := dispCR (reverse (X), n, Nil),

dispCR (X, n, Z) := if n=0 then concat (Z, reverse (X)) else dispCR (cdr (X), n-1, cons (car (X), Z)),

displaceCR (01110, 2) = dispCR (reverse (01110), 2, Nil) = dispCR (01110, 2, Nil) = dispCR (1110, 1, cons (0, Nil)) = dispCR (110, 0, cons (1, (0))) = concat (10, reverse (110)) = concat (10, 011) = 10011;

22) Побітова сума за модулем 2 (без перенесення 1 в старший розряд) двох двійкових чисел однакової розрядності:

LOm2 (X, Y) := if X==Nil then Nil else if car(X)=car(Y) then cons (0, LOm2 (cdr (X), cdr (Y))) else cons (1, LOm2 (cdr (X), cdr (Y))),

LOm2 (110, 101) = cons (0, LOm2 (10, 01)) = cons (0, cons (1, LOm2 (0, 1))) = cons (0, cons (1, cons (1, LOm2 ((), ()))) = cons (0, cons (1, cons (1, Nil))) = (011);

Сума за модулем 2 (з перенесенням 1 в старший розряд) двох двійкових чисел однакової розрядності:

Om2 (X,Y) := m2 (reverse (X), reverse (Y), Nil, 0),

m2 (X, Y, Z, f) := if X=Nil then

if f=1 then cons (f, Z)

else Z

else if car(X)=car(Y) then

if car(X)=f then m2 (cdr (X), cdr (Y), cons (f, Z), f)

else m2(cdr(X),cdr (Y),cons(f, Z),not (f))

else m2 (cdr (X), cdr (Y), cons (not (f), Z), f),

$Om2(101, 110) = m2(101, 011, Nil, 0) = m2(01, 11, cons(1, ()), 0) = m2(1, 1, cons(1, (1), 0)) = m2((), (), cons(0, (1\ 1), 1)) = cons(1, (011)) = (1011),$

$Om2(1100, 1011) = m2(0011, 1101, Nil, 0) = m2(011, 101, cons(1, ()), 0) = m2(11, 01, cons(1, (1)), 0) = m2(1, 1, cons(1, (11)), 0) = m2((), (), cons(0, (111)), 1) = cons(1, (0111)) = (10111),$

$Om2(101, 001) = m2(101, 100, Nil, 0) = m2(01, 00, cons(0, ()), 1) = m2(1, 0, cons(1, (0)), 0) = m2((), (), cons(1, (10)), 0) = (110).$

Як привести двійкові числа, що моделюються списками, до однакової розрядності?

samewordlength (X, Y) := if length(X)<length(Y) then function (append (X, length(Y)–length(X)), Y) else if length(X)>length(Y) then function (X, append (Y, length(X)–length(Y))) else function (X, Y),

append (Z, n) := if n=0 then Z else cons (0, append (Z, n–1)),

$samewordlength(101, 11010) = function(append(101, 5-3), 11010) = function(append(101, 2), 11010) = function(cons(0, append(101, 1)), 11010) = function(cons(0, cons(0, append(101, 0))), 11010) = function(cons(0, cons(0, 101))), 11010) = function(00101, 11010),$

де function (X, Y) – функція будь-якої логічної операції над двійковими числами, що оперує побітово з числами однакової розрядності, наприклад, LОm2 (X, Y) з завдання 22.

5. НЕКЛАСИЧНІ ЛОГІКИ

5.1. Нечіткі множини. Операції над нечіткими множинами

Теоретичні відомості

Нечітку множину A визначають через універсальну множину X і функцію належності $\mu_A(x)$, що набуває значення з відрізка $[0, 1]$. Тому, нечітка множина A - це сукупність пар виду

$$A = \{ (x, \mu_A(x)), x \in X \}.$$

Основними операціями над нечіткими множинами, що використовуються у моделюванні, є наступні:

1. Операція об'єднання: об'єднанням нечітких множин A і B у X називають нечітку множину $A \cup B$ з функцією належності, що має вигляд

$$\mu_{A \cup B}(x) = \max\{\mu_A(x), \mu_B(x)\}, x \in X.$$

2. Операція перетину (перерізу): перетином нечітких множин A і B у X називають нечітку множину $A \cap B$ з функцією належності, що має вигляд

$$\mu_{A \cap B}(x) = \min\{\mu_A(x), \mu_B(x)\}, x \in X.$$

3. Операція доповнення: доповненням A в X називають множину $\bar{A}$ (A') з функцією належності

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x), x \in X.$$

4. Відстань Хеммінга

$$d_h = \sum_i |\mu_A(x_i) - \mu_B(x_i)|$$

Відстань Евкліда

$$d_e = \sqrt{\sum_i (\mu_A(x_i) - \mu_B(x_i))^2}$$

Для підрахунку відносної відстані, треба розділити отримане значення відповідної відстані на кількість елементів в універсальній множині

Приклад. Задана універсальна множина $X = \{D, E, F, G\}$ і нечіткі підмножини A і B : $A = \{(D|0,5), (E|0,6), (F|0,9)\}$, $B = \{(D|0,1), (E|0,8), (G|0,3)\}$.

Для заданого приклада знайдемо $A \cap B$, $A \cup B$, A' , B' , відстань Хеммінга, відстань Евкліда та відповідні відносні відстані:

перетин - $A \cap B = \{(D|0,1), (E|0,6), (F|0), (G|0)\};$

об'єднання - $A \cup B = \{(D|0,5), (E|0,8), (F|0,9), (G|0,3)\};$

доповнення - $A' = \{(D|0,5), (E|0,4), (F|0,1), (G|1)\}$

та $B' = \{(D|0,9), (E|0,2), (F|1), (G|0,7)\}.$

Відстань Хемінга дорівнює

$$d_h = |0,5-0,1| + |0,6-0,8| + |0,9-0| + |0-0,3| = 0,4 + 0,2 + 0,9 + 0,3 = 1,8.$$

Для підрахунку відносної відстані, треба розділити отримане значення на кількість елементів в універсальній множині: $(1,8/4) = 0,45.$

Відстань Евкліда: $d_e = 1,1$; відносна відстань Евкліда: $0,275.$

Вказівка. Довести самостійно $d_e \leq d_h$.

Завдання 5.1.1. Для заданої універсальної множини і нечітких підмножин A і B знайдіть $A \cap B$, $A \cup B$, A' , B' , відстань Хеммінга, відстань Евкліда та відповідні відносні відстані.

На універсальній множині $X = \{D, E, F, G\}$ задано:

1) $A = \{(D|1), (E|0), (F|0,2), (G|0,6)\},$

$$B = \{(D|0,8), (E|1), (F|0,5), (G|0,6)\}.$$

2) $A = \{(D|0,1), (E|0), (F|0,2), (G|0,7)\},$

$$B = \{(D|0,5), (E|0,3), (F|0,2), (G|0,6)\}.$$

3) $A = \{(D|0,5), (E|0), (F|0,4), (G|0,7)\},$

$$B = \{(D|0,1), (E|0,3), (F|0,4), (G|0,6)\}.$$

4) $A = \{(D|0,2), (E|0,3), (F|0,4), (G|0,8)\},$

$$B = \{(D|0,1), (E|0,5), (F|0,5), (G|0,6)\}.$$

5) $A = \{(D|0,2), (E|0,3), (F|0,4), (G|0,8)\},$

$$B = \{(D|0,6), (E|0,2), (F|0,3), (G|0)\}.$$

Завдання 5.1.2. Перевірити виконуванисть основних логічних законів для нечітких підмножин A , B і C .

Приклад 1. Доведемо $A \cup U = U$. Доведення проводиться за визначеннями.

Дві нечіткі множини дорівнюють одна одній, якщо співпадають функції належності кожного елемента універсальної множини, що входять до нечітких підмножин. Для доведення рівності необхідно показати, що функції належності в нечітких підмножинах праворуч і ліворуч від знака « $\Rightarrow$ » - однакові.

Праворуч знаходиться множина U , функції належності в якій для кожного елемента є «1» (для пустої множини $\emptyset$ за визначенням функції належності кожного елемента дорівнюють «0»). Ліворуч знаходиться нечітка підмножина, яка є результатом виконання операції об'єднання нечітких підмножин A і U .

За визначенням операції об'єднання функції належності елементів є $\mu_{A \cup U}(x) = \max\{\mu_A(x), \mu_U(x)\}$. Значення функцій належності елементів з A набувають значення з відрізка $[0, 1]$, отже, максимальне значення між числом з $[0, 1]$ і 1 є 1, що доводить рівність $A \cup U = U$.

Приклад 2. Доведемо $A \cup A^c \neq U$.

Для нечітких підмножин не виконується закон виключення третього. Покажемо це.

Ліворуч знаходиться нечітка множина з функцією належності, що визначається для кожного елемента як $\max\{\mu_A(x), \mu_{A^c}(x)\}$, де $\mu_A(x)$ число з $[0, 1]$, а $\mu_{A^c}(x) = 1 - \mu_A(x)$. В загальному випадку $\max\{\mu_A(x), \mu_{A^c}(x)\} \neq 1$; наприклад, якщо $\mu_A(x) = 0,3$, то $\max\{0,3; 1-0,3\} = \max\{0,3; 0,7\} = 0,7$.

У випадку, коли $\mu_A(x) = 1$, або $\mu_A(x) = 0$, рівність $A \cup A^c = U$ виконується, що співпадає з результатами теорії чітких множин (чіткі множини є частковим випадком нечітких множин, коли $\mu_A(x_i)$ належать $\{0, 1\}$), і показує, що теорія нечітких множин є розширенням теорії чітких множин.

В теорії нечітких множин також не виконується закон протиріччя, і, як наслідок, не виконуються закони поглинання доповнень та склеювання.

Приклад 3. Доведемо $A \cap (A \cup B) = A$.

Для доведення даної тотожності слід розглянути три випадки:

- 1) $\mu_A(x_i) = \mu_B(x_i)$;
- 2) $\mu_A(x_i) < \mu_B(x_i)$;
- 3) $\mu_A(x_i) > \mu_B(x_i)$.

Розглянемо, наприклад, випадок $\mu_A(x_i) < \mu_B(x_i)$. Тоді для елемента x_i функція належності для нечіткої підмножини ліворуч буде визначена як

$$\min\{\mu_A(x_i), \max\{\mu_A(x_i), \mu_B(x_i)\}\} = \min\{\mu_A(x_i), \mu_B(x_i)\} = \mu_A(x_i).$$

Аналогічно визначаємо функції належності елементів x_i для кожного з випадків нечіткої підмножини ліворуч. Результатом $A \cap (A \cup B)$ буде нечітка підмножина A .

- 1) $\emptyset \subset A \cap A^c$;
- 2) $\emptyset = A \cap \emptyset$;
- 3) $A = A \cap U$;
- 4) $A \cup \emptyset = A$;
- 5) $A \cap A^c \neq \emptyset$;
- 6) $A \cup (A \cap B) = A$;
- 7) $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$;
- 8) $A \cup A = A$;
- 9) $(A \cap B) \cup (A^c \cap B) \neq B$;
- 10) $A \cap (A^c \cup B) \neq A \cap B$.

5.2. Нечіткі та лінгвістичні змінні. Методи побудови функції приналежності. Нечіткий логічний висновок.

Теоретичні відомості

Поняття змінної є основоположним математичним поняттям. Нечіткі і лінгвістичні змінні використовуються при природно-мовному описі різних

об'єктів і явищ, при формалізації процесів прийняття рішень в важкоформалізуємих ситуаціях, в описі складних явищ.

Нечіткою змінною називається

$$\langle \alpha, X, C_\alpha \rangle,$$

де α - найменування нечіткої змінної; $X = \{x\}$ - область її визначення (базова множина); $C_\alpha = \{ \mu_A(x) / x \}$ - нечітка множина на X , що описує обмежені, але можливі значення нечіткої змінної α .

У більш загальному випадку значеннями таких змінних можуть бути слова або речення природної або формальної мови, і тоді відповідні змінні називають *лінгвістичними*. Так, наприклад, нечітка змінна *висота* могла б набувати таких значень: *високий, низький, доволі високий, дуже високий, більш-менш високий, не високий, не дуже високий* тощо. Ці значення є речення, що утворені поняттям *високий* та запереченням *ні*, співками *і, але*, а також словами типу *дуже, досить, цілком, більш-менш*.

Лінгвістична змінна є змінною вищого порядку, ніж нечітка змінна, в тому сенсі, що значеннями лінгвістичної змінної є нечіткі змінні. Лінгвістичні змінні призначені в основному для аналізу складних або погано визначених явищ. Використання словесних описів типу тих, якими оперує людина, робить можливим аналіз систем настільки складних, що вони недосяжні звичайному математичному аналізу.

Лінгвістичною змінною називається

$$\langle \beta, T_\beta, X, G, M \rangle,$$

де β - найменування лінгвістичної змінної;

T_β - множина значень (терм-множина), що представляють собою найменування нечітких змінних, областю визначення (базовою множиною) кожної з яких є X ;

G - синтаксичне правило, породжує найменування $\alpha \in T_\beta$ вербальних значень лінгвістичної змінної β ;

M - семантичне правило, яке ставить у відповідність кожній нечіткій змінній $\alpha \in T_\beta$, нечітку множину C_α , тобто сенс (семантику) нечіткої змінної α .

Процедура утворення нових термів G здійснюється за допомогою логічних зв'язок: *ні*, *і* та *або* і модифікаторів типу «дуже», «злегка» т.п.. Процедурю завдання нечітких змінних M , наприклад, "мала швидкість», «середня швидкість», «висока швидкість», а так само відповідних нечітких множин для термів з $G(T)$ виконується відповідно до правил трансляції нечітких зв'язок і модифікаторів. Правило M може бути виконане по одній з типових операцій над нечіткими множинами.

При завданні лінгвістичної змінної (ЛЗ) її значення, тобто, нечіткі множини (НМ), повинні задовольняти наступним вимогам.

1. Упорядкованість. Нечіткі множини повинні бути впорядковані (розташовуватися за базовою шкалою) відповідно до порядку завдання якісних значень для лінгвістичної змінної.
2. Обмеженість. Область визначення ЛЗ має бути чітко позначена (визначені мінімальні та максимальні значення на базовій шкалі). На межах універсальної множини U , де визначено ЛЗ, значення функцій приналежності її мінімального та максимального НМ мають бути одиничними.

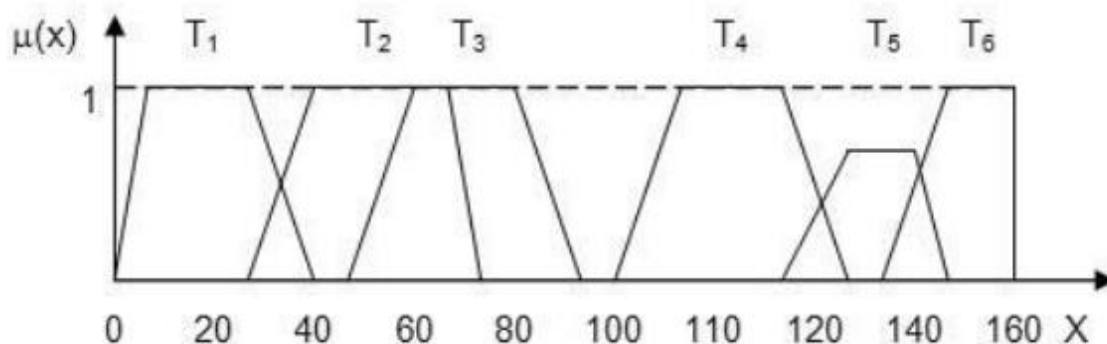
На малюнку нижче наведено приклад завдання ЛП з порушенням вимог.

T_1 має неправильну функцію приналежності, а T_6 - правильну.

3. Узгодженість. Повинне дотримуватися природне розмежування понять, коли одна й та сама точка універсальної множини U не може одночасно належати $\mu(x) = 1$ двом і більше НМ (ця умова узгодженості порушується парою $T_2 - T_3$).

4. Повнота. Кожне значення в області визначення ЛП має описуватися хоча б одним НМ (дана умова повноти порушується між парою $T_3 - T_4$).

5. Нормальність. Кожне поняття ЛП повинно мати хоча б один еталонний чи типовий об'єкт, тобто, у будь-якій точці функція приналежності НМ має бути одиничною (це правило порушується T5).



Приклад завдання ЛП із порушенням вимог

Існує кілька точок зору на змістовну інтерпретацію функцій приналежності, при яких переважуючою оцінкою степеня приналежності є фіксування конкретних значень в інтервалі $[0,1]$. Таке фіксування носить суб'єктивний характер і проводиться в результаті експертного опитування.

Для методів, заснованих на експертних оцінках, важливим є характер вимірювань: первинний, тобто безпосередній, або похідний, тобто опосередкований, а також тип шкали, якою користується експерт і яка потім визначає вид операцій з обробки експертної інформації. Інший особливістю можна назвати те, що оцінювані об'єкти володіють двома типами властивостей:

- 1) властивості, які можна безпосередньо виміряти, наприклад: довжину, висоту, зріст, вагу, обсяг і т.п.,
- 2) властивості, які не мають кількісної міри, а є якісними, наприклад: краса, зручність, корисність і т.п.

У другому випадку, як правило, вдаються до попарного порівняння об'єктів, що володіють якісними властивостями і визначають місце (ранг) по відношенню до розглядаємого поняття A .

Існуючі методи побудови функції приналежності можна розділити на групи: *прямі і непрямі*.

Прямі методи полягають в тому, що експерт безпосередньо задає ступінь приналежності кожного $x \in X$ поняттю A чи визначає вид функції $\mu_A(x)$. При цьому значення функції узгоджуються з уподобаннями експерта по відношенню x_i на множині X наступним чином:

- для будь-яких $x_1, x_2 \in X$ $\mu_A(x_1) < \mu_A(x_2)$ тоді і тільки тоді, коли x_2 переважніше x_1 тобто в більшій мірі (з більшою ймовірністю, з більшою можливістю, з більшою необхідністю) характеризується поняттям A ;

- для будь-яких $x_1, x_2 \in X$ $\mu_A(x_1) = \mu_A(x_2)$ тоді і тільки тоді, коли x_1 і x_2 байдужі щодо поняття A .

Як правило, прямі методи використовуються для опису об'єктів, властивості яких можна безпосередньо виміряти. У цьому випадку здійснюють безпосереднє завдання значень степені приналежності. До прямих методів можна віднести методи, засновані на ймовірнісному трактуванні функції приналежності $\mu_A(x) = p(x/A)$, тобто ймовірність того, що об'єкт $x \in X$ буде віднесений до нечіткої множини A .

У *непрямих методах* значення функції приналежності вибираються виходячі з заздалегідь сформульованих умов. Інформація, що надходить від експертів, є лише вихідною для реалізації встановленого в даному методі алгоритму обробки цієї інформації з метою отримання $\mu_A(x)$. Опосередковані методи є єдино можливими в тих випадках, коли властивості, через які визначається поняття A , не мають кількісної міри, наприклад, "краса". У таких випадках використовують рангові вимірювання при попарному порівнянні об'єктів. Непрямі методи більш трудомісткі, ніж прямі, але їхня перевага - в стійкості по відношенню до спотворень у відповіді, а отже, в більшій об'єктивності отримуваної функції приналежності.

Одним з можливих способів прийняття рішень при невизначеності інформації є застосування нечіткого логічного висновку.

Нечітким логічним висновком називається отримання висновку у вигляді нечіткої множини, відповідного поточним значенням входів, з використанням нечіткої бази знань і нечітких операцій: $B_t = A_t \circ (A \rightarrow B)$

Основу нечіткого логічного виведення становить композиційне правило Заде та нечітка імплікація.

Використовуючи нечітку імплікацію $A \rightarrow B$ отримаємо відношення на $A \times B$: $\mu_{R_{A \times B}} = \{\mu_{\langle a_1, b_1 \rangle}; \mu_{\langle a_1, b_2 \rangle}; \dots, \mu_{\langle a_n, b_m \rangle}\}$, де кожне значення отримано як мінімум за правилом Мамдані (існують і інші схеми: Цукамото, Ларсена): $\mu_{\langle a_i, b_j \rangle} = \min(\mu_a, \mu_b)$.

Композиційне правило виведення Заде формулюється так: якщо відомо нечітке відношення між входною (x) і вихідною (y) змінними, то при нечіткому значенні входної змінної $x = \tilde{A}$, нечітке значення вихідної змінної визначається наступним чином: $y = \tilde{A} \circ R$, де $\circ$ - композиція.

Формально це можна записати у такій формі:

якщо $x \in A$, то $y \in B$, інакше $y \in C$

$x \in A'$

$y \in D$

Завдання 5.2.1. Застосувати композиційне правило виведення з урахуванням нечіткої імплікації за правилом Мамдані:

Rule: *if X is A, then Y is B*

Fact: *X is A'*

Conclusion: *Y is B'*

Приклад:

Задано: $A = \{x_1/0,5; x_2/1; x_3/0,6\};$

$X = \{x_1, x_2, x_3\}$

$B = \{y_1/1; y_2/0,4\};$

$Y = \{y_1, y_2\}$

та $A' = \{x_1/0,6; x_2/0,9; x_3/0,7\}.$

Використовуючи нечітку імплікації отримаємо відношення на $X \times Y$:

$\mu_R = \{ \langle x_1, y_1 \rangle / 0,5; \langle x_1, y_2 \rangle / 0,4; \langle x_2, y_1 \rangle / 1; \langle x_2, y_2 \rangle / 0,4; \langle x_3, y_1 \rangle / 0,6; \langle x_3, y_2 \rangle / 0,4 \}$, де кожне значення отримано як $\min$ за правилом Мамдані:

$\langle x_1; y_1 \rangle$ - $\min A/x_1$ и B/y_1 , отримаємо $0,5 = \min(0,5; 1)$

$\langle x_2; y_2 \rangle$ - $\min A/x_2$ и B/y_2 , отримаємо $0,4 = \min(1; 0,4)$ і т.п.

Запишемо функції приналежності відношення (результат нечіткої імплікації) у вигляді матриці:

μ_R	y_1	y_2
x_1	0,5	0,4
x_2	1	0,4
x_3	0,6	0,4

Далі визначаємо власне нечіткий висновок:

A'		y_1	y_2
	x_1	0,5	0,4
$x_1/0,6; \quad x_2/0,9; \quad x_3/0,7$	x_2	1	0,4
	x_3	0,6	0,4

$$\mu_{B'}(y_1) = \sup_{x \in X} \min[\mu_{A'}(x_i), R(x_i, y_1)] =$$

$$= \max [\min(0,6;0,5), \min(0,9;1), \min(0,7;0,6)] = \max(0,6;0,9;0,7) = 0,9$$

$$\mu_{B'}(y_2) = \sup_{x \in X} \min[\mu_{A'}(x_i), R(x_i, y_2)] =$$

$$= \max [\min(0,6;0,4), \min(0,9;0,4), \min(0,7;0,4)] = \max (0,4; 0,4; 0,4) = 0,4$$

Відповідь: $B' = \{y_1/0,9; y_2/0,4\}$

Завдання:

№	A	B	A'
1	$\{x_1/0,3; x_2/0,1; x_3/0,5\}$	$\{y_1/0,3; y_2/0,4; y_3/0,6\}$	$\{x_1/0,3; x_2/0,2; x_3/0,4\}$
2	$\{x_1/0,2; x_2/0,7\}$	$\{y_1/0,5; y_2/0,7; y_3/0,9\}$	$\{x_1/0,5; x_2/1\}$
3	$\{x_1/0,4; x_2/1; x_3/0,6\}$	$\{y_1/0,2; y_2/0,4; y_3/0,3; y_4/0,1\}$	$\{x_1/0,7; x_2/0,1; x_3/0,6\}$

Завдання 5.2.2.

Запропоновано декілька правил, які переводять нечітке умовне висловлювання «якщо $x \in A$, то $y \in B$, інакше $y \in C$ » в нечітке відношення $U \bullet V$.

Нехай x, y - імена об'єктів; A, A', B, C, D - нечіткі поняття, представлені нечіткими множинами, визначеними на U, U, V, V, V відповідно.

Тут A, B, C - нечіткі множини в U, V, V , задані у вигляді

$$A = \int \mu_A(u) / u; B = \int \mu_B(v) / v; C = \int \mu_C(v) / v.$$

Тоді маємо:

A. Максиміне правило Rm' :

$$Rm' = (A \times B) \cup (\sim A \times C) = \int (\mu_A(u) \wedge \mu_B(v)) \vee ((1 - \mu_A(u)) \wedge \mu_C(v)) / (u, v).$$

B. Арифметичне правило Ra' (Лукасевича):

$$\begin{aligned} Ra' &= (\sim A \times V + U \times B) \cap (A \times V + U \times C) = \\ &= \int 1 \wedge (1 - \mu_A(u) + \mu_B(v)) \wedge (\mu_A(u) + \mu_C(v)) / (u, v). \end{aligned}$$

B. Розмите бінарне правило:

$$\begin{aligned} Rb' &= (\sim A \times V \cup U \times B) \cap (A \times V \cup U \times C) = \\ &= \int ((1 - \mu_A(u)) \vee \mu_B(v)) \wedge ((\mu_A(u) \vee \mu_C(v)) / (u, v). \end{aligned}$$

Г. Правило Танака-Мідзумото (Геделева логічна система):

$$\begin{aligned} Rgg' &= (A \times V \Rightarrow U \times B) \cap (\sim A \times V \Rightarrow U \times C) = \\ &= \int (\mu_A(u) \rightarrow \mu_B(v)) \wedge ((1 - \mu_A(u)) \rightarrow \mu_C(v)) / (u, v) \end{aligned}$$

$$\text{де } \mu_A(u) \rightarrow \mu_B(v) = \begin{cases} 1, & \text{при } \mu_A \leq \mu_B \\ \mu_B, & \text{при } \mu_A > \mu_B \end{cases}$$

Таким чином, повертаючись до початкової постановці завдання наслідок D можна отримати наступним чином:

$$Dm = A' \bullet Rm',$$

$$Da = A' \bullet Ra',$$

$$Db = A' \bullet Rb',$$

$$Dgg = A' \bullet Rgg'.$$

Нехай є наступні посилки:

якщо x - маленьке, то y - велике, інакше y - маленьке ;

якщо $x \in A$, то $y \in B$, інакше $y \in C$

x - не дуже маленьке .

$x \in A'$. Знайти значення y' .

Множина $U = 1 + 2 + 3 = \{1, 2, 3\}$ - прийнята форма запису через +.

$$\text{маленьке} = 1/1 + 0.4/2 = 1/1 + 0.4/2 + 0/3;$$

$$\text{велике} = 0.5/2 + 1/3 = 0/1 + 0.5/2 + 1/3.$$

Квантифікатор **дуже** може інтерпретуватися за допомогою операції концентрації, тобто зведення в квадрат: **дуже $x = \int \mu^2 x(u)/u$:**

$$\text{тоді термін дуже маленьке} = 1/1 + 0.16/2 = 1/1 + 0.16/2 + 0/3;$$

а квантифікатор **не** може інтерпретуватися за допомогою операції заперечення: **не $x = \int (1 - \mu x(u))/u$:**

$$\text{тоді не дуже маленьке} = 0.84/2 + 1/3 = 0/1 + 0.84/2 + 1/3.$$

В умові

$A = \text{маленьке}, B = \text{велика}, C = \text{маленьке}.$

Тобто, $A = 1-0,4-0$; $B = 0-0,5-1$; $C = 1-0,4-0$.

A x - не дуже маленьке - $A' = 0-0,84-1$

Для формули $D = A' \bullet R'$, де це мах-мін операція, а R' - одна з матриць нечіткого відношення, отриманих нижче за певним правилом нечіткого виведення.

1) Відношення для максимінного правила:

$$Rm' = (A \times B) \cup (\sim A \times C) = \int_{U \times V} (\mu_A(u) \wedge \mu_B(v)) \vee ((1 - \mu_A(u)) \wedge \mu_C(v)) / (u, v)$$

Приклад обчислення значень елементів матриці Rm' :

$$(u_1, v_1) = (1 \wedge 0) \vee (0 \wedge 1) = 0, \quad (u_1, v_2) = (1 \wedge 0.5) \vee (0 \wedge 0.4) = 0.5,$$

$$(u_1, v_3) = (1 \wedge 1) \vee (0 \wedge 0) = 1,$$

$$(u_2, v_1) = (0.4 \wedge 0) \vee (0.6 \wedge 1) = 0.6, \quad (u_2, v_2) = (0.4 \wedge 0.5) \vee (0.6 \wedge 0.4) = 0.4,$$

$$\begin{aligned}
 (u_2, v_3) &= (0.4 \wedge 1) \vee (0.6 \wedge 0) = 0.4, \\
 (u_3, v_1) &= (0 \wedge 0) \vee (1 \wedge 1) = 1, \quad (u_3, v_2) = (0 \wedge 0.5) \vee (1 \wedge 0.4) = 0.4, \\
 (u_3, v_3) &= (0 \wedge 1) \vee (1 \wedge 0) = 0,
 \end{aligned}$$

$$Rm' = \begin{vmatrix} 0 & 0.5 & 1 \\ 0.6 & 0.4 & 0.4 \\ 1 & 0.4 & 0 \end{vmatrix}$$

Тоді значення y' може бути визначене в такий спосіб:

$$\begin{aligned}
 y' &= \text{не дуже маленьке} \bullet Rm' = \\
 &= \begin{vmatrix} 0 & 0.84 & 1 \end{vmatrix} \bullet \begin{vmatrix} 0 & 0.5 & 1 \\ 0.6 & 0.4 & 0.4 \\ 1 & 0.4 & 0 \end{vmatrix} = \begin{vmatrix} 1 & 0.4 & 0.4 \end{vmatrix},
 \end{aligned}$$

тобто $y' = 1/1 + 0.4/2 + 0.4/3$, що може бути інтерпретовано (з деякою натяжкою) як *досить-таки маленьке*.

2) Далі розглянемо для зазначених вище посилок арифметичне правило Ra' :

$$Ra' = (\sim A \times V + U \times B) \cap (A \times V + U \times C) = \int_{U \times V} 1 \wedge (1 - \mu_A(u) + \mu_B(v)) \wedge ((\mu_A(u) + \mu_C(v)) / (u, v))$$

=

$$\begin{vmatrix} 0 & 0.5 & 1 \\ 0.6 & 0.8 & 0.4 \\ 1 & 0.4 & 0 \end{vmatrix}$$

Тоді значення y' при використанні арифметичного правила може бути визначене в такий спосіб

$$y' = \text{не дуже маленьке} \bullet Ra' = \begin{vmatrix} 0 & 0.84 & 1 \end{vmatrix} \bullet \begin{vmatrix} 0 & 0.5 & 1 \\ 0.6 & 0.8 & 0.4 \\ 1 & 0.4 & 0 \end{vmatrix} = \begin{vmatrix} 1 & 0.8 & 0.4 \end{vmatrix},$$

тобто $y' = 1/1 + 0.8/2 + 0.4/3$.

3) Висновок з використанням розмитого бінарного правила наведено нижче:

$$Rb' = (\sim A \times V \cup U \times B) \cap (A \times V \cup U \times C) = \int_{U \times V} ((1 - \mu_A(u)) \vee \mu_B(v)) \wedge ((\mu_A(u) \vee \mu_C(v)) / (u, v) =$$

$$\begin{vmatrix} 0 & 0.5 & 1 \\ 0.4 & 0.4 & 0.4 \\ 1 & 0.4 & 0 \end{vmatrix}.$$

Тоді значення y' , може бути визначене в такий спосіб

$$y' = \text{не дуже маленьке} \bullet Rb' = \begin{vmatrix} 0 & 0.84 & 1 \end{vmatrix} \bullet \begin{vmatrix} 0 & 0.5 & 1 \\ 0.4 & 0.4 & 0.4 \\ 1 & 0.4 & 0 \end{vmatrix} = \begin{vmatrix} 1 & 0.4 & 0.4 \end{vmatrix},$$

тобто $y' = 1/1 + 0.4/2 + 0.4/3$, що може бути також інтерпретовано як **досить таки маленьке**.

4) Останній нечіткий висновок проведемо з використанням правила Танака-Мідзумото

$$Rgg' = (A \times V \Rightarrow U \times B) \cap (\sim A \times V \Rightarrow U \times C) = \int_{U \times V} (\mu_A(u) \rightarrow \mu_B(v)) \wedge ((1 - \mu_A(u)) \rightarrow \mu_C(v)) / (u, v)$$

$$\text{Где } \mu_A(u) \rightarrow \mu_B(v) = \begin{cases} 1, & \text{если } \mu_A \leq \mu_B \\ \mu_B, & \text{если } \mu_A > \mu_B \end{cases}$$

$$Rgg' = \begin{vmatrix} 0 & 0.5 & 1 \\ 0 & 0.4 & 0 \\ 1 & 0.4 & 0 \end{vmatrix}$$

$$y' = \text{не дуже маленьке} \bullet Rgg' = \begin{vmatrix} 0 & 0.84 & 1 \end{vmatrix} \bullet \begin{vmatrix} 0 & 0.5 & 1 \\ 0 & 0.4 & 0 \\ 1 & 0.4 & 0 \end{vmatrix} = \begin{vmatrix} 1 & 0.4 & 0 \end{vmatrix},$$

тобто $y' = 1/1 + 0.4/2 + 0/3$, що інтерпретується як **маленьке**.

Розглянуті приклади показують, що використання відношень Rm' , Ra' , Rb' для нечіткого висновку не дають наслідків, які збігалися б з нашими інтуїтивними уявленнями. Кращі результати дає відношення Rgg' .

Виконати: Маємо наступне правило:

якщо x - маленьке, то y - велике, інакше y - маленьке.

Знайти значення y' , використовуючи послідовно всі чотири правила нечіткого виведення (максиміне, арифметичне, розмите бінарне і правило Танака-Міндзумото).

Друга посилка: x - *дуже велике*.

Значення змінних **маленьке** = $\{1/1, 2 / 0.7, 3 / 0.2, 4/0\}$

велике = $\{1/0, 2 / 0.25, 3 / 0.75, 4/1\}$

A = маленьке, B = велике, C = маленьке, B^2 = дуже велике.

Зверніть увагу, множинами U (для A) та V (для B і C) є $\{1, 2, 3, 4\}$.

Завдання 5.2.3.

Лінгвістичні невизначеності це - модифікатори, які можна розглядати як унарні оператори над нечіткими підмножинами. Вони можуть бути представлені, наприклад, операціями концентрації, розтягування, операторами збільшення нечіткості тощо. Так, невизначеності «дуже» часто ставлять у відповідність операцію концентрації CON, а невизначеності «більш-менш» іноді ставлять у відповідність операцію розтягування DIL.

Приклад. Нехай $U = \{1,2,3,4\}$ і $A = \text{"малий"} = ((1/1), (2/0,6), (3/0,2), (4/0))$.

Тоді "дуже малий" = $A^2 = ((1/1), (2/0,36), (3/0,04), (4/0))$ – застосування CON;

а "більш-менш" = $\sqrt{A} = A^{0.5} = ((1/1), (2/0,78), (3/0,44), (4/0))$ – застосування DIL

Якщо взяти "не малий", то потрібно застосувати формулу $(1-\mu(x))$:

"не малий A " = $((1/0), (2/0,4), (3/0,8), (4/1))$.

Задана множина $U = \{a, b, c, d\}$.

Задані нечіткі лінгвістичні підмножини «маленьке» і «велике»:

«маленьке» = $\{a/1, b/0.7, c/0.2, d/0\}$; **«велике»** = $\{a/0, b/0.25, c/0.75, d/1\}$.

Застосувати квантіфікатори: **дуже, приблизно, не, не дуже...**

*Квантіфікатор **дуже** може інтерпретуватися за допомогою операції концентрації, тобто зведення в квадрат; квантіфікатор **не** може*

інтерпретуватися за допомогою операції заперечення: **не** $x = \lceil (1 - \mu_x(u)) / u \rceil$;

приблизно – оператор розтягування.

Дуже маленький

Дуже великий

Не маленький

Не великий

Приблизно маленький

Приблизно великий

Не дуже маленький

Не дуже великий

Завдання 5.2.4.

Побудова функцій приналежності на основі парних порівнянь

Вихідною інформацією для побудови функцій приналежності є експертні парні порівняння. Для кожної пари елементів універсальної множини експерт оцінює перевагу одного елемента над іншим по відношенню до властивості нечіткої множини. Зручно представляти наступний матрицею:

$$A = \begin{matrix} & \begin{matrix} u_1 & u_2 & \dots & u_n \end{matrix} \\ \begin{matrix} u_1 \\ u_2 \\ \dots \\ u_n \end{matrix} & \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} \end{matrix}$$

де a_{ij} рівень переваги елемента u_i над u_j ($i, j = \overline{1, n}$), який визначається за дев'ятибальною шкалою Сааті:

1 - якщо відсутня перевага елемента u_i над елементом u_j ;

3 - якщо є слабка перевага u_i над u_j ;

5 - якщо є суттєва перевага u_i над u_j ;

7 - якщо є явна перевага u_i над u_j ;

9 - якщо є абсолютна перевага u_i над u_j ;

2, 4, 6, 8 - проміжні порівняльні оцінки.

Матриця парних порівнянь є діагональною ($a_{ij} = 1, i = \overline{1, n}$) і обернено симетричною ($a_{ij} = \frac{1}{a_{ji}}, i, j = \overline{1, n}$).

При узгоджених парних порівняннях процедура побудови функцій приналежності значно спрощується. З узгодженими думками експерта матриця парних порівнянь має наступні властивості:

- вона діагональна, тобто $a_{ii}=1, i=1..n$;
- вона обернено симетрична, тобто елементи, які симетричні відносно головної діагоналі, пов'язані залежністю $a_{ij}=1/a_{ji}, i,j=1..n$;
- вона транзитивна, тобто $a_{ik}a_{kj}=a_{ij}, i,j,k=1..n$.

Наявність цих властивостей дозволяє визначити всі елементи матриці парних порівнянь, якщо відомі $(n-1)$ недіагональних елементів. Наприклад, якщо відомий k -тий рядок, тобто елементи $a_{kj}, i,j=1..n$, то довільний елемент a_{ij} визначається так:

$$a_{ij} = \frac{a_{kj}}{a_{ki}} \quad i,j,k = \overline{1,n}.$$

Після визначення всіх елементів матриці парних порівнянь степені приналежності нечіткої множини обчислюються за формулою:

$$\mu(u_i) = \frac{1}{a_{1i} + a_{2i} + \dots + a_{ni}}$$

Формули не потребують виконання складних обчислювальних процедур, пов'язаних із пошуком власного вектору матриці A .

Приклад. Побудувати функцію приналежності нечіткої множини "високий чоловік" на універсальній множині $\{170, 175, 180, 185, 190, 195\}$, якщо відомі такі експертні парні порівняння:

- абсолютна перевага 195 над 170;
- явна перевага 195 над 175;
- значна перевага 195 над 180;
- слабка перевага 195 над 185;
- відсутність переваги 195 над 190.

Приведеним експертним висловлюванням відповідає така матриця парних порівнянь:

	170	175	180	185	190	195
170	1	7/9	5/9	1/3	1/9	1/9
175	9/7	1	5/7	3/7	1/7	1/7
180	9/5	7/5	1	3/5	1/5	1/5
185	3	7/3	5/3	1	1/3	1/3
190	9	7	5	3	1	1
195	9	7	5	3	1	1

	170	175	180	185	190	195
μ високий чоловік (u_i) (субнормальна нечітка множина)	0.0399	0.0513	0.0718	0.1196	0.3588	0.3588
μ високий чоловік (u_i) (нормальна нечітка множина)	0.1111	0.1429	0.2000	0.3333	1.0000	1.0000

Побудувати функцію приналежності нечіткої множини "чоловік середнього зросту" та "чоловік невисокий" методом узгоджених парних порівнянь.

5.3. Багатозначні логіки

Теоретичні відомості

Тризначна логіка була історично першою багатозначною логікою і є найпростішим розширенням двозначної логіки. Перелік істиннісних значень тризначної логіки, крім «істинно» і «хибно», включає також третє значення.

Як третє логічне значення висловлювання **Лукасевичем** було введено значення, що виражається словами «ймовірно», «нейтрально». Про кожне висловлювання у системі Лукасевича можна сказати: воно або істинно (1), або хибно (0), або нейтрально (S). Це стало можливим завдяки тому, що Лукасевич одним із перших висунув тезу про можливість побудови логічних обчислень, в яких не діє принцип несуперечності. На підставі тризначної логіки Лукасевич побудував систему модальної логіки, в якій поряд із дослідженням логічних операцій над асерторичними висловлюваннями (ствердженнями та запереченнями) досліджуються так звані модальні висловлювання (сильні та слабкі твердження та заперечення).

Основними функціями їм були взяті заперечення та імплікація, а похідними – кон'юнкція та диз'юнкція. Тавтологія у логіці Лукасевича набуває значення «1», тобто це – виділене істинне значення. Заперечення та імплікація визначаються таблицями та рівностями, наведеними нижче:

x	заперечення
1	0
S	S
0	1

$x \rightarrow y$	1	S	0
1	1	S	0
S	1	1	S
0	1	1	1

Через те, що в загальному випадку в багатозначних логіках число значень істинності аргументів та функцій може бути нескінченним, то прийняті позначення та обчислення через формули:

$[Nx] = 1 - [x]$ - заперечення

$[Cxy] = 1$, якщо $[x] \leq [y]$ - імплікація

$[Cxy] = 1 - [x] + [y]$, якщо $[x] > [y]$

Тобто,

$[Cxy] = \min(1, 1 - [x] + [y])$

Визначення похідних функцій:

$[Kxy] = \min([x], [y])$ - кон'юнкція

$[Axy] = \max([x], [y])$ - диз'юнкція

Вочевидь, що у цих визначеннях не є законами логіки (тавтологіями) закони двозначної логіки: виключення третього, протиріччя та його заперечення. Тому система Лукасевича не є запереченням двозначної логіки. У його логіці правило зняття подвійного заперечення, правила де Моргана та правило контрапозиції є тавтологіями. А правило приведення до абсурду – ні.

Гейтінг створив тризначну пропозиційну логіку, ґрунтуючись на твердженні, що істинним є лише $x \rightarrow \neg \neg x$. Імплікація та заперечення відрізняються від визначень цих операцій у Лукасевича.

імплікація

x	заперечення
1	0
S	0
0	1

$x \rightarrow y$	1	S	0
1	1	S	0
S	1	1	0
0	1	1	1

Через те, що в загальному випадку в багатозначних логіках число значень істинності аргументів та функцій може бути нескінченним, то прийняті позначення та обчислення через формули:

$$[Cxy]=1, \text{ якщо } [x] \leq [y]$$

$$[Cxy]=[y], \text{ якщо } [x] > [y]$$

Тавтологія набуває значення 1, кон'юнкція-диз'юнкція обчислюється як мінімум і максимум. У логіці Гейтинга закони несуперечності, контрапозиції, де Моргана та виключення четвертого є тавтологіями. Але закон заперечення третього та його заперечення – не є тавтологіями.

Завдання 5.3.1. Перевірити закон зняття подвійного заперечення у тризначній логіці Лукасевича та у тризначній логіці Гейтинга.

Закон зняття подвійного заперечення формулюється наступним чином:

$$\neg\neg A \rightarrow A.$$

Необхідно перевірити, чи є дана формула в логіці Лукасевича та в логіці Гейтинга тавтологією.

Приклад 1. Розглянемо тризначну логіку Лукасевича.

Побудуємо таблицю істинності:

A	$\neg A$	$\neg\neg A$	$\neg\neg A \rightarrow A$
1	0	1	1
S	S	S	1
0	1	0	1

Отже, закон зняття подвійного заперечення у логіці Лукасевича виконується, тобто формула є тавтологією.

Приклад 2.

Побудуємо таблицю істинності для формули $\neg\neg A \rightarrow A$ в системі Гейтинга:

A	$\neg A$	$\neg\neg A$	$\neg\neg A \rightarrow A$
1	0	1	1
S	0	1	S
0	1	0	1

Отже, закон зняття подвійного заперечення у логіці Гейтинга не виконується, для значення висловлювання S формула не дорівнює «1».

1) Перевірити закон навішування подвійного заперечення:

- а) у тризначній логіці Лукасевича;
- б) у тризначній логіці Гейтинга.

2) Перевірити закон силогізму:

- а) у тризначній логіці Лукасевича;
- б) у тризначній логіці Гейтинга.

3) Перевірити прямий закон контрапозиції:

- а) у тризначній логіці Лукасевича;
- б) у тризначній логіці Гейтинга.

4) Перевірити закон оберненої контрапозиції:

- а) у тризначній логіці Лукасевича;
- б) у тризначній логіці Гейтинга.

5) Перевірити закон доведення до абсурду:

- а) у тризначній логіці Лукасевича;
- б) у тризначній логіці Гейтинга.

6) Перевірити закон де Моргана:

- а) у тризначній логіці Лукасевича;
- б) у тризначній логіці Гейтинга.

Завдання 5.3.2.

У тризначній логіці Поста (для циклічного заперечення) побудувати таблицю істинності для імплікації.

K -значна логіка Поста Е.Л. є узагальненням окремого випадку двозначної логіки ($k=2$). Пост ввів N^1_x - циклічне заперечення, N^2_x - симетричне заперечення. Циклічне заперечення визначається рівностями:

$$[N^1_x] = [x] + 1 \text{ при } [x] \leq k-1$$

$$[N^1_x] = 1$$

Симетричне заперечення за Постом визначається:

$$[N^2_x] = k - [x] + 1.$$

Вочевидь, що при $k=2$ циклічне і симетричне заперечення збігаються з запереченням двозначної логіки і між собою. Операції кон'юнкції і диз'юнкції визначаються як мінімум і максимум значення аргументів.

Якщо значеннями істинності є 1,2,3, то з n -значної системи Поста виокремлюється тризначна логіка P3.

Як приклад розглянемо тризначну систему Поста P3.

x	N^1_x	N^2_x
1	2	3
2	3	2
3	1	1

Імплікацію можна визначити за формулою : $X \rightarrow Y = \neg X \vee Y$. Побудуємо таблицю істинності формули $\neg X \vee Y$ для циклічного заперечення.

X	Y	$\neg X$	$\neg X \vee Y$
1	1	3	3
1	2	3	3
1	3	3	3
2	1	2	2
2	2	2	2
2	3	2	3
3	1	1	1
3	2	1	2
3	3	1	3

1) У тризначній логіці Поста (для симетричного заперечення) побудувати таблицю істинності для імплікації.

2) У тризначній логіці Поста (для циклічного заперечення) побудувати таблицю істинності для еквівалентності.

3) У тризначній логіці Поста (для симетричного заперечення) побудувати таблицю істинності для еквівалентності.

4) У тризначній логіці Поста (для симетричного заперечення) побудувати таблицю істинності для формули $X \vee \neg X$.

5) У тризначній логіці Поста (для циклічного заперечення) побудувати таблицю істинності для формули $X \vee \neg X$.

6) У тризначній логіці Поста (для симетричного заперечення) побудувати таблицю істинності для формули $X \wedge \neg X$.

7) У тризначній логіці Поста (для циклічного заперечення) побудувати таблицю істинності для формули $X \wedge \neg X$.

5.4. Модальні логіки

Теоретичні відомості

У природній мові висловлювання можуть набувати різні значення істинності в залежності від контексту. При цьому вживаються такі слова, як

«можливо», «іноді», «повинно», «необхідно» тощо. Такі слова називаються *модальностями*.

Формальна мова, що використовує поняття "можливо" і "необхідно" в якості кванторів, називається логікою можливого, або *алетичною логікою*.

У модальній логіці на рівні числення висловлювань використовується відома система пропозиціональних зв'язок, але до цих зв'язок додаються модальні оператори $\Box$ - *необхідно* і $\Diamond$ - *можливо*.

Наприклад, $\Box F$ читається: «необхідно, щоб F » або " F необхідно". Це означає, що F необхідно істинно, або істинно у всіх можливих світах.

$\Diamond F$ читається: «можливо, що F » або " F можливо". Це означає, що F можливо істинно, або істинно в деякому можливому світі.

Тимчасова логіка вводить модальності "іноді" і "завжди" і їх заперечення "часто" і "ніколи" разом з квантифікаторами "минулого" і "в майбутньому". Тимчасові оператори при цьому використовуються наступні:

G - завжди (в майбутньому); GA істинно, якщо A залишається завжди істинним;

H - завжди (в минулому); HA істинно, якщо A завжди було істинним;

F - іноді (в майбутньому); FA - істинно, якщо A іноді буде істинним;

P - іноді (в минулому); PA істинно, якщо A іноді було істинним;

U - до тих пір, поки; $U(A, B)$ істинно, якщо A істинно, починаючи з поточного моменту до тих пір, поки B не стане істинним в майбутньому.

Всі ці модальності вводяться як оператори.

Приклад.

1. Можливо, що Том виграє в лотерею телевізор:

$\Diamond$ Виграти (Том, лотерея, телевізор) .

2. Неможливо, щоб Сміт в майбутньому завжди у все вигравав:

$\neg(\Diamond(F\forall y\exists z\text{Виграти}(\text{Сміт}, y, z)))$.

3. Необхідно, щоб у кожної людини було якесь хобі:

$\Box \forall x\exists y \text{МаєХобі}(x,y)$.

Завдання 5.4.1. Формалізувати речення з використанням модальних операторів.

- 1) Можливо, що Джон складе іспит на відмінно.
- 2) Кожний студент мусить вести конспект всіх лекцій.
- 3) Можливо існувала снігова людина.
- 4) Неможливо, щоб людині постійно не щастило.
- 5) Будь-яка людина вірить в можливу удачу.
- 6) Не можна завжди вигравати.
- 7) Можливо на Марсі є життя.
- 8) Можливо в минулому Землю відвідували прибульці.
- 9) Щоб влаштуватися на престижну роботу, треба мати диплом.
- 10) Треба займатися якимось видом спорту, щоб запобігти можливим майбутнім хворобам.

Відповіді до розділу 5

5.2.1.

- 1) $B' = \{y1/0,3; y2/0,4; y3/0,4\}$
- 2) $B' = \{y1/0,5; y2/0,7; y3/0,7\}$
- 3) $B' = \{y1/0,2; y2/0,4; y3/0,3; y4/0,1\}$

5.2.2.

Маємо наступне правило: якщо x - маленьке, то y - велике, інакше y - маленьке. Знайти значення y , використовуючи послідовно всі чотири правила нечіткого виведення (максиміне, арифметичне, розмите бінарне і правило Танака-Міндзумото).

Друга посилка: x - дуже велике.

Значення змінних:

$$\text{маленьке} = \frac{1}{1} + \frac{0,7}{2} + \frac{0,2}{3},$$

$$\text{велике} = \frac{0,25}{2} + \frac{0,75}{3} + \frac{1}{4}$$

Квантіфікатор **дуже** може інтерпретуватися за допомогою операції концентрації, тобто зведення в квадрат: $\text{дуже } x = \int \mu^2 x(u) / u$.

$$\text{Тоді термін дуже великий} = \frac{0,06}{2} + \frac{0,56}{3} + \frac{1}{4}.$$

Відношення для максимінного правила:

$$Rm' = (A \times B) \cup (\sim A \times C) = \int_{U \times V} (\mu_A(u) \wedge \mu_B(v)) \vee ((1 - \mu_A(u)) \wedge \mu_C(v)) / (u, v)$$

Тут A = маленьке, B = велике, C = маленьке, B^2 = дуже велике.

$$(u_1, v_1) = (1 \wedge 0) \vee (0 \wedge 1) = 0$$

$$(u_1, v_2) = (1 \wedge 0,25) \vee (0 \wedge 0,7) = 0,25$$

$$(u_1, v_3) = (1 \wedge 0,75) \vee (0 \wedge 0,2) = 0,75$$

$$(u_1, v_4) = (1 \wedge 1) \vee (0 \wedge 0) = 1$$

$$(u_2, v_1) = (0,7 \wedge 0) \vee (0,35 \wedge 1) = 0,35$$

$$(u_2, v_2) = (0,7 \wedge 0,25) \vee (0,35 \wedge 0,7) = 0,35$$

$$(u_2, v_3) = (0,7 \wedge 0,75) \vee (0,35 \wedge 0,2) = 0,7$$

$$\begin{aligned}
(u_2, v_4) &= (0,7 \wedge 1) \vee (0,35 \wedge 0) = 0,7 \\
(u_3, v_1) &= (0,2 \wedge 0) \vee (0,85 \wedge 1) = 0,85 \\
(u_3, v_2) &= (0,2 \wedge 0,25) \vee (0,85 \wedge 0,7) = 0,7 \\
(u_3, v_3) &= (0,2 \wedge 0,75) \vee (0,85 \wedge 0,2) = 0,2 \\
(u_3, v_4) &= (0,2 \wedge 1) \vee (0,85 \wedge 0) = 0,2 \\
(u_4, v_1) &= (0 \wedge 0) \vee (1 \wedge 1) = 1 \\
(u_4, v_2) &= (0 \wedge 0,25) \vee (1 \wedge 0,7) = 0,7 \\
(u_4, v_3) &= (0 \wedge 0,75) \vee (1 \wedge 0,2) = 0,2 \\
(u_4, v_4) &= (0 \wedge 1) \vee (1 \wedge 0) = 0
\end{aligned}$$

$$Rm' = \begin{vmatrix} 0 & 0.25 & 0.75 & 1 \\ 0.35 & 0.35 & 0.7 & 0.7 \\ 0.85 & 0.7 & 0.2 & 0.2 \\ 1 & 0.7 & 0.2 & 0 \end{vmatrix}$$

Тоді значення у' може бути визначене в такий спосіб:

$$y' = \text{дуже велике} \bullet Rm' = \begin{vmatrix} 0 & 0.06 & 0.56 & 1 \end{vmatrix} \bullet \begin{vmatrix} 0 & 0.25 & 0.75 & 1 \\ 0.35 & 0.35 & 0.7 & 0.7 \\ 0.85 & 0.7 & 0.2 & 0.2 \\ 1 & 0.7 & 0.2 & 0 \end{vmatrix} = \begin{vmatrix} 1 & 0.7 & 0.2 & 0.2 \end{vmatrix},$$

тобто $y' = \frac{1}{1} + \frac{0.7}{2} + \frac{0.2}{3} + \frac{0.2}{4}$, що може бути інтерпретовано (з деякою натяжкою) як досить-таки мале.

Далі розглянемо для зазначених вище посилок арифметичне правило Ra' :

$$Ra' = (\sim A \times V + U \times B) \cap (A \times V + U \times C) =$$

$$\int_{U \times V} 1 \wedge (1 - \mu_A(u) + \mu_B(v)) \wedge ((\mu_A(u) + \mu_C(v)) / (u, v))$$

$$\begin{aligned}
(u_1, v_1) &= 1 \wedge (1 - 1 + 0) \wedge (1 + 1) = 0 \\
(u_1, v_2) &= 1 \wedge (1 - 1 + 0,25) \wedge (1 + 0,7) = 0,25 \\
(u_1, v_3) &= 1 \wedge (1 - 1 + 0,75) \wedge (1 + 0,2) = 0,75 \\
(u_1, v_4) &= 1 \wedge (1 - 1 + 1) \wedge (1 + 0) = 1 \\
(u_2, v_1) &= 1 \wedge (1 - 0,7 + 0) \wedge (0,7 + 1) = 0,3 \\
(u_2, v_2) &= 1 \wedge (1 - 0,7 + 0,25) \wedge (0,7 + 0,7) = 0,45 \\
(u_2, v_3) &= 1 \wedge (1 - 0,7 + 0,75) \wedge (0,7 + 0,2) = 0,9
\end{aligned}$$

$$\begin{aligned}
(u_2, v_4) &= 1 \wedge (1 - 0,7 + 1) \wedge (0,7 + 0) = 0,7 \\
(u_3, v_1) &= 1 \wedge (1 - 0,2 + 0) \wedge (0,2 + 1) = 0,8 \\
(u_3, v_2) &= 1 \wedge (1 - 0,2 + 0,25) \wedge (0,2 + 0,7) = 0,9 \\
(u_3, v_3) &= 1 \wedge (1 - 0,2 + 0,75) \wedge (0,2 + 0,2) = 0,4 \\
(u_3, v_4) &= 1 \wedge (1 - 0,2 + 1) \wedge (0,2 + 0) = 0,2 \\
(u_4, v_1) &= 1 \wedge (1 - 0 + 0) \wedge (0 + 1) = 1 \\
(u_4, v_2) &= 1 \wedge (1 - 0 + 0,25) \wedge (0 + 0,7) = 0,7 \\
(u_4, v_3) &= 1 \wedge (1 - 0 + 0,75) \wedge (0 + 0,2) = 0,2 \\
(u_4, v_4) &= 1 \wedge (1 - 0 + 1) \wedge (0 + 0) = 0
\end{aligned}$$

Тоді значення у 'при використанні арифметичного правила може бути визначене в такий спосіб

$$y' = \text{дуже велике} \quad \bullet \quad Ra' = \begin{vmatrix} 0 & 0.25 & 0.75 & 1 \\ 0.3 & 0.45 & 0.9 & 0.7 \\ 0.8 & 0.9 & 0.4 & 0.2 \\ 1 & 0.7 & 0.2 & 0 \end{vmatrix} \bullet = [1 \quad 0.7 \quad 0.4 \quad 0.2]$$

$$\text{тобто } y' = \frac{1}{1} + \frac{0.7}{2} + \frac{0.4}{3} + \frac{0.2}{4}.$$

Висновок з використанням розмитого бінарного правила наведено нижче:

$$Rb' = (\sim A \times V \cup U \times B) \cap (A \times V \cup U \times C) =$$

$$\int_{U \times V} ((1 - \mu_A(u)) \vee \mu_B(v)) \wedge ((\mu_A(u) \vee \mu_C(v)) / (u, v)$$

$$\begin{aligned}
(u_1, v_1) &= (1 - 1 \vee 0) \wedge (1 \vee 1) = 0 \\
(u_1, v_2) &= (1 - 1 \vee 0,25) \wedge (1 \vee 0,7) = 0,25 \\
(u_1, v_3) &= (1 - 1 \vee 0,75) \wedge (1 \vee 0,2) = 0,75 \\
(u_1, v_4) &= (1 - 1 \vee 1) \wedge (1 \vee 0) = 1 \\
(u_2, v_1) &= (1 - 0,7 \vee 0) \wedge (0,7 \vee 1) = 0,3 \\
(u_2, v_2) &= (1 - 0,7 \vee 0,25) \wedge (0,7 \vee 0,7) = 0,3 \\
(u_2, v_3) &= (1 - 0,7 \vee 0,75) \wedge (0,7 \vee 0,2) = 0,7
\end{aligned}$$

$$\begin{aligned}
(u_2, v_4) &= (1 - 0,7 \vee 1) \wedge (0,7 \vee 0) = 0,7 \\
(u_3, v_1) &= (1 - 0,2 \vee 0) \wedge (0,2 \vee 1) = 0,8 \\
(u_3, v_2) &= (1 - 0,2 \vee 0,25) \wedge (0,2 \vee 0,7) = 0,7 \\
(u_3, v_3) &= (1 - 0,2 \vee 0,75) \wedge (0,2 \vee 0,2) = 0,2 \\
(u_3, v_4) &= (1 - 0,2 \vee 1) \wedge (0,2 \vee 0) = 0,2 \\
(u_4, v_1) &= (1 - 0 \vee 0) \wedge (0 \vee 1) = 1 \\
(u_4, v_2) &= (1 - 0 \vee 0,25) \wedge (0 \vee 0,7) = 0,25 \\
(u_4, v_3) &= (1 - 0 \vee 0,75) \wedge (0 \vee 0,2) = 0,2 \\
(u_4, v_4) &= (1 - 0 \vee 1) \wedge (0 \vee 0) = 0
\end{aligned}$$

Тоді значення y' , може бути визначене в такий спосіб

$$y' = \text{дуже велике} \quad \bullet \quad Rb' = \begin{bmatrix} 0 & 0,06 & 0,56 & 1 \end{bmatrix} \bullet \begin{bmatrix} 0 & 0,25 & 0,75 & 1 \\ 0,3 & 0,3 & 0,7 & 0,7 \\ 0,8 & 0,7 & 0,2 & 0,2 \\ 1 & 0,25 & 0,2 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0,7 & 0,2 & 0,2 \end{bmatrix}$$

тобто $y' = \frac{1}{1} + \frac{0,7}{2} + \frac{0,2}{3} + \frac{0,2}{4}$, що може бути також інтерпретовано як досить таки маленьке.

Останній нечіткий висновок проведемо з використанням правила Танака-Мідзумото

$$R_{gg}' = (A \times V \Rightarrow U \times B) \cap (\sim A \times V \Rightarrow U \times C) = \int_{U \times V} (\mu_A(u) \rightarrow \mu_B(v)) \wedge ((1 - \mu_A(u)) \rightarrow \mu_C(v)) / (u, v)$$

$$\text{Где } \mu_A(u) \rightarrow \mu_B(v) = \begin{cases} 1, & \text{если } \mu_A \leq \mu_B \\ \mu_B, & \text{если } \mu_A > \mu_B \end{cases}$$

$$\begin{aligned}
(u_1, v_1) &= (1 \rightarrow 0) \wedge ((1 - 1) \rightarrow 1) = 0 \\
(u_1, v_2) &= (1 \rightarrow 0,25) \wedge ((1 - 1) \rightarrow 0,7) = 0,25 \\
(u_1, v_3) &= (1 \rightarrow 0,75) \wedge ((1 - 1) \rightarrow 0,2) = 0,75 \\
(u_1, v_4) &= (1 \rightarrow 1) \wedge ((1 - 1) \rightarrow 0) = 1 \\
(u_2, v_1) &= (0,7 \rightarrow 0) \wedge ((1 - 0,7) \rightarrow 1) = 0 \\
(u_2, v_2) &= (0,7 \rightarrow 0,25) \wedge ((1 - 0,7) \rightarrow 0,7) = 0,25 \\
(u_2, v_3) &= (0,7 \rightarrow 0,75) \wedge ((1 - 0,7) \rightarrow 0,2) = 0,2
\end{aligned}$$

$$\begin{aligned}
(u_2, v_4) &= (0,7 \rightarrow 1) \wedge ((1 - 0,7) \rightarrow 0) = 0 \\
(u_3, v_1) &= (0,2 \rightarrow 0) \wedge ((1 - 0,2) \rightarrow 1) = 1 \\
(u_3, v_2) &= (0,2 \rightarrow 0,25) \wedge ((1 - 0,2) \rightarrow 0,7) = 0,7 \\
(u_3, v_3) &= (0,2 \rightarrow 0,75) \wedge ((1 - 0,2) \rightarrow 0,2) = 0,2 \\
(u_3, v_4) &= (0,2 \rightarrow 1) \wedge ((1 - 0,2) \rightarrow 0) = 0 \\
(u_4, v_1) &= (0 \rightarrow 0) \wedge ((1 - 0) \rightarrow 1) = 1 \\
(u_4, v_2) &= (0 \rightarrow 0,25) \wedge ((1 - 0) \rightarrow 0,7) = 0,7 \\
(u_4, v_3) &= (0 \rightarrow 0,75) \wedge ((1 - 0) \rightarrow 0,2) = 0,2 \\
(u_4, v_4) &= (0 \rightarrow 1) \wedge ((1 - 0) \rightarrow 0) = 0
\end{aligned}$$

$$R_{gg'} = \begin{vmatrix} 0 & 0.25 & 0.75 & 1 \\ 0 & 0.25 & 0.2 & 0 \\ 1 & 0.7 & 0.2 & 0 \\ 1 & 0.7 & 0.2 & 0 \end{vmatrix}$$

$$y' = \text{дуже велике} \quad \bullet \quad R_{gg'} = \begin{vmatrix} 0 & 0.06 & 0.56 & 1 \\ 0 & 0.25 & 0.75 & 1 \\ 0 & 0.25 & 0.2 & 0 \\ 1 & 0.7 & 0.2 & 0 \\ 1 & 0.7 & 0.2 & 0 \end{vmatrix} = \begin{vmatrix} 0 & 0.25 & 0.75 & 1 \\ 0 & 0.25 & 0.2 & 0 \\ 1 & 0.7 & 0.2 & 0 \\ 1 & 0.7 & 0.2 & 0 \end{vmatrix}$$

тобто $y' = \frac{1}{1} + \frac{0.7}{2} + \frac{0.2}{3} + \frac{0}{4}$, що інтерпретовано як маленьке.

5.2.3.

	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>
<i>Дуже маленький</i>	1	0,49	0,04	0
<i>Не маленький</i>	0	0,3	0,8	1
<i>Приблизно маленький</i>	1	0,8	0,4	0
<i>Не дуже маленький</i>	0	0,51	0,96	1
<i>Дуже великий</i>	0	0,625	0,56	1
<i>Не великий</i>	1	0,75	0,25	0
<i>Приблизно великий</i>	0	0,5	0,86	1
<i>Не дуже великий</i>	1	0,375	0,44	0

5.3.1.

1) Закон формулюється таким чином: $A \rightarrow \neg\neg A$

а) виконується (тобто є тавтологією);

б) виконується;

2) Закон формулюється таким чином: $(A \rightarrow B) \rightarrow ((B \rightarrow C) \rightarrow (A \rightarrow C))$

- а) не виконується
(не є тавтологією; на наборі 1S0 формула дорівнює S);
- б) виконується;
- 3) Закон формулюється таким чином: $(A \rightarrow B) \rightarrow (\neg B \rightarrow \neg A)$
- а) виконується;
- б) виконується;
- 4) Закон формулюється таким чином: $(\neg A \rightarrow \neg B) \rightarrow (B \rightarrow A)$
- а) виконується;
- б) не виконується
(не є тавтологією; на наборі S1 формула дорівнює 0);
- 5) Закон формулюється таким чином: $(A \rightarrow B) \rightarrow ((A \rightarrow \neg B) \rightarrow \neg A)$
- а) не виконується
(не є тавтологією; на наборі SS формула дорівнює S);
- б) виконується;
- 6) Один з варіантів закону формулюється таким чином:
 $\neg(A \& B) = \neg A \vee \neg B$
- а) виконується;
- б) виконується.

Побудова таблиць істинності:

$\neg(A \& B) = \neg A \vee \neg B$ – закон де Моргана

A	B	A&B	ЛУКАСЕВИЧ				ГЕЙТИНГ			
			$\neg A$	$\neg B$	$\neg A \vee \neg B$	$\neg(A \& B)$	$\neg A$	$\neg B$	$\neg A \vee \neg B$	$\neg(A \& B)$
0	S	0	1	S	1	1	1	0	1	1
S	0	0	S	1	1	1	0	1	1	1
S	S	S	S	S	S	S	0	0	0	0
S	1	S	S	0	S	S	0	0	0	0
1	S	S	0	S	S	S	0	0	0	0

$(A \rightarrow B) \rightarrow^2 ((A \rightarrow \neg B) \rightarrow^1 \neg A)$ – правило доведення до абсурду

		ЛУКАСЕВИЧ						ГЕЙТИНГ					
A	B	$A \rightarrow B$	$\neg B$	$A \rightarrow \neg B$	$\neg A$	1	2	$A \rightarrow B$	$\neg B$	$A \rightarrow \neg B$	$\neg A$	1	2
0	S	1	S	1	1	1	1	1	0	1	1	1	1
S	0	S	1	1	S	S	1	0	1	1	0	0	1
S	S	1	S	1	S	S	S	1	0	0	0	1	1
S	1	1	0	S	S	1	1	1	0	0	0	1	1
1	S	S	S	S	0	S	1	S	0	0	0	1	1

$(A \rightarrow B) \rightarrow^2 (\neg B \rightarrow^1 \neg A)$ – контрапозиція

		ЛУКАСЕВИЧ					ГЕЙТИНГ				
A	B	$A \rightarrow B$	$\neg B$	$\neg A$	1	2	$A \rightarrow B$	$\neg B$	$\neg A$	1	2
0	S	1	S	1	1	1	1	0	1	1	1
S	0	S	1	S	S	1	0	1	0	0	1
S	S	1	S	S	1	1	1	0	0	1	1
S	1	1	0	S	1	1	1	0	0	1	1
1	S	S	S	0	S	1	S	0	0	1	1

$(\neg A \rightarrow^1 \neg B) \rightarrow^2 (B \rightarrow A)$ – зворотна контрапозиція

		ЛУКАСЕВИЧ					ГЕЙТИНГ				
A	B	$\neg A$	$\neg B$	1	$B \rightarrow A$	2	$\neg A$	$\neg B$	1	$B \rightarrow A$	2
0	S	1	S	S	S	1	1	0	0	0	1
S	0	S	1	1	1	1	0	1	1	1	1
S	S	S	S	1	1	1	0	0	1	1	1
S	1	S	0	S	S	1	0	0	1	S	S
1	S	0	S	1	1	1	0	0	1	1	1

$(A \rightarrow B) \rightarrow^2 ((B \rightarrow C) \rightarrow^1 (A \rightarrow C))$ – силісгізм

19	ЛУКАСЕВИЧ							ГЕЙТИНГ				
A	B	C	$A \rightarrow B$	$B \rightarrow C$	$A \rightarrow C$	1	2	$A \rightarrow B$	$B \rightarrow C$	$A \rightarrow C$	1	2
0	0	S	1	1	1	1	1	1	1	1	1	1
0	S	0	1	S	1	1	1	1	0	1	1	1
0	S	S	1	1	1	1	1	1	1	1	1	1
0	S	1	1	1	1	1	1	1	1	1	1	1
0	1	S	1	S	1	1	1	1	S	1	1	1
S	0	0	S	1	S	S	1	0	1	0	0	1
S	0	S	S	1	1	1	1	0	1	1	1	1
S	0	1	S	1	1	1	1	0	1	1	1	1
S	S	0	1	S	S	1	1	1	0	0	1	1
S	S	S	1	1	1	1	1	1	1	1	1	1
S	S	1	1	1	1	1	1	1	1	1	1	1
S	1	0	1	0	S	1	1	1	0	0	1	1
S	1	S	1	S	1	1	1	1	S	1	1	1
S	1	1	1	1	1	1	1	1	1	1	1	1
1	0	S	0	1	S	1	1	0	1	S	S	1
1	S	0	S	S	0	S	1	S	0	0	1	1
1	S	S	S	1	S	S	1	S	1	S	S	1
1	S	1	S	1	1	1	1	S	1	1	1	1
1	1	S	1	S	S	1	1	1	S	S	1	1

ВИСНОВКИ

ЗАКОН	ЛУКАСЕВИЧ	ГЕЙТИНГ
<i>де Моргана</i>	+	+
<i>абсурда</i>	-	+
<i>контрапозиції</i>	+	+
<i>зворотна контрапозиція</i>	+	-
<i>силісгізм</i>	+	+

5.3. 2.

Вказівка: еквівалентність розглядати як $(X \equiv Y) = (X \rightarrow Y) \& (Y \rightarrow X)$.

5.4.1.

1) Можливо, що Джон складе іспит на відмінно.

$\Diamond$ СкластиІспит (Джон, відмінно);

2) Кожний студент мусить вести конспект всіх лекцій.

$\Box (\forall x \forall y \text{ ПишеКонспект } (x, y))$;

3) Можливо існувала снігова людина.

$\Diamond P$ Існує (снігова людина);

4) Неможливо, щоб людині постійно не щастило.

$\neg \Diamond (\exists x \neg \text{Щастить}(x))$;

5) Будь-яка людина вірить в можливу удачу.

$\Diamond \exists z \forall x \text{ Вірить } (x, z)$;

6) Не можна завжди вигравати.

$\neg \Diamond (G (\forall x \forall y \text{ Вигравати}(x, y)))$;

7) Можливо на Марсі є життя.

$\Diamond (\exists x \text{ Проживає}(x, \text{Марс}))$;

8) Можливо в минулому Землю відвідували прибульці.

$\Diamond (P (\exists x (\text{Прибульці } (x) \& \text{Відвідувати } (\text{Земля}, x))))$;

9) Щоб влаштуватися на престижну роботу, треба мати диплом.

$\forall x (\Box \text{МаєДиплом}(x) \rightarrow G (\exists y (\text{ПрестижРаб}(y) \& \text{Влаштувався}(x, y))))$;

10) Треба займатися якимось видом спорту, щоб запобігти можливим майбутнім хворобам.

$\forall x (\Box (H (\exists y \text{ Займається}(x, y))) \rightarrow F (\neg \Diamond \exists z \text{ Хворіє}(x, z)))$.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Базова література

1. S. C. Kleene. Mathematical Logic. John Wiley & Sons. Dover reprint, 2002. ISBN 0-486-42533-9.
2. Robert R Stoll. Sets Logic and Axiomatic Theories Edition Paperback – January 1, 1961.
3. Бардачов Ю.М. Дискретна математика / Ю.М.Бардачов, Н.А.Соколова, В.Є.Ходаков – К.: Вища школа, 2002. –287 с.
4. Заяць В.М. Функційне програмування: навч. посіб. для студ. вищ. навч. закл., що навчаються за спец. "Програмне забезпечення автомат. систем" / В. М. Заяць; Міністерство освіти і науки України, Нац. ун-т "Львівська політехніка". - Львів : Бескид Біт, 2003. - 160 с.
5. Нікольський Ю.В., Пасічник В.В., Щербина Ю.М. Системи штучного інтелекту : Навчальний посібник. За ред. В.В.Пасічника. – 2-ге вид., випр. та доп. - Львів : Магнолія -2006, 2013
6. Таран Т. А. Основи дискретної математики. Навчальний посібник. К.: Просвіта. – 2003. – с.288.
7. Математична логіка та теорія алгоритмів: Конспект лекцій [Електронний ресурс]: навч. посіб. для студ. спеціальності 113 «Прикладна математика», освітньої програми «Наука про дані та математичне моделювання» / О. Л. Темнікова ; КПІ ім. Ігоря Сікорського. – Мережеве електронне ви-дання (1 файл: 3,92 Мбайт). – Київ : ПТФ Просвіта, 2025. – 252 с; ISBN 978-617-7010-33-2
8. ДИСКРЕТНА МАТЕМАТИКА. Частина 2. Булева алгебра. ПРАКТИКУМ [Електронний ресурс]: навч. посіб. для студ. спеціальності 113 «Прикладна математика», освітньої програми «Наука про дані та математичне моделювання» / О. Л. Темнікова, Д. Ю. Тавров; КПІ ім. Ігоря Сікорського. — Електронні текстові дані (1 файл: 3,18 Мбайт). — Київ : КПІ ім. Ігоря Сікорського, 2024. — 185 с.

9. Темнікова О.Л. Математична логіка. Практикум [Електронний ресурс]: навч. посіб. для студ. спеціальності 113 «Прикладна математика», освітньої програми «Наука про дані та математичне моделювання»; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 1,37 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2020. – 76 с.
10. Темнікова О.Л. Теорія алгоритмів. Алгоритмічні схеми. Практикум [Електронний ресурс]: навч. посіб. для студ. спеціальності 113 «Прикладна математика», освітньої програми «Наука про дані та математичне моделювання» / О.Л.Темнікова ; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 1,54 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2022. – 43 с.

Допоміжна література

11. Zadeh, L. A. Fuzzy algorithms. Information and Control 12 (2), 1968. p. 94. doi:10.1016/S0019-9958(68)90211-8.
12. Zadeh, L. A. Fuzzy logic = computing with words. IEEE Transactions on Fuzzy Systems 4 (2), 1996. p. 103. doi:10.1109/91.493904.
13. Бадаєв Ю.І. Функціональне програмування: навч. посіб. / Ю. І. Бадаєв, О. Є. Круш, А. Л. Гурін, С. В. Ричік. - Київ: НТУУ "КПІ", 2012. – 136 с.
14. Бородкіна І.Л. Теорія алгоритмів: навчальний посібник для студентів вищих навчальних закладів / І.Л. Бородкіна, Г.О. Бородкін; Міністерство освіти і науки України; Національний університет біоресурсів і природокористування України. - Київ : Центр учбової літератури, 2020.— 184 с.
15. Желдак Т. А. Нечіткі множини в системах управління та прийняття рішень: навч. пос. /Т. А. Желдак, Л. С. Коряшкіна, С. А. Ус; за редакцією С. А. Ус. – Дніпро: НТУ ДП, 2020. – 386 с.
16. Івахів О. В. Основи побудови систем керування з нечіткою логікою: навч. пос. /О. Івахів, М. Наконечний. – Львів: Растр-7, 2017. – 129 с.
17. Керрол Л. Історія з вузликами. - М.: Наука. 1973. – 408 с.

18. Конверський А.Є. Логіка: (традиційна і сучасна): підручник для студ вищ. навч. закладів / А.Є. Конверський ; Київський національний університет ім. Тараса Шевченка. - Київ: ЦНЛ, 2004. – 536 с.
19. Кофман А. Введення в теорію нечітких множин. - М.: Радіо та зв'язок. 1982. – 432 с.
20. Лупенко С.А. Комп'ютерна логіка :підручник /С.А. Лупенко; Міністерство освіти і науки України. - Львів: Видавництво "Магнолія 2006", 2021. – 573 с.
21. Матвієнко М.П. Математична логіка та теорія алгоритмів: навчальний посібник для студентів ВНЗ / М.П. Матвієнко, С.П. Шаповалов; МОН України, Сумський державний університет. - Київ: Ліра-К, 2015. – 211 с.