Міністерство освіти і науки України
Рівненський державний гуманітарний університет
Кафедра інформатики та прикладної математики

Назарук М.В., Сінчук А.М.

Навчально-методичний посібник

МАТЕМАТИЧНА ЛОГІКА ТА ТЕОРІЯ АЛГОРИТМІВ

для самостійної та індивідуальної роботи студентів
спеціальностей 121 Інженерія програмного забезпечення та
122 Комп'ютерні науки

Рівне-2024

УДК 510.6:510.5(075)
Н 19

Друкується за рішенням Вченої ради Рівненського державного гуманітарного університету (протокол №7-2020 від 26.02.2020 р.)

Назарук М.В., Сінчук А.М. Математична логіка та теорія алгоритмів. Навчально-методичний посібник. – Рівне: РДГУ, 2020. – 118 с.

Рецензенти:
Гладка О.М., к.т.н., доцент кафедри комп'ютерних технологій та економічної кібернетики навчально-наукового інституту автоматики, кібернетики та обчислювальної техніки Національного університету водного водного господарства та природокористування;
Шевцова Н.В. к.т.н., доцент кафедри інформатики та прикладної математики Рівненського державного гуманітарного університету.

УДК 510.6:510.5(075)
Н 19

Навчально-методичний посібник містить короткий виклад основних теоретичних відомостей з математичної логіки та теорії алгоритмів; приклади розв'язання типових задач та задачі для самостійного розв'язування по кожній темі. Посібник призначений для самостійної та індивідуальної роботи студентів спеціальностей 121 Інженерія програмного забезпечення та 122 Комп'ютерні науки, які вивчають математичну логіку і теорію алгоритмів.

Зміст

Передмова

Навчально-методичний посібник розроблений у відповідності до програми ВНЗ і призначений для вивчення дисципліни «Математична логіка та теорія алгоритмів».

Посібник містить короткий виклад основних теоретичних відомостей з програмних розділів: «Алгебра висловлень», «Числення висловлень», «Аналіз змістовних міркувань на базі логіки висловлень. Логічні задачі», «Алгебра предикатів», «Логічні загальнозначущі формули алгебри предикатів. Числення предикатів», «Аналіз змістовних міркувань на базі логіки предикатів», «Випереджені нормальні форми і логічний висновок у логіці предикатів», «Квантори», «Багатозначна логіка», «Теорія алгоритмів. Основні положення та означення теорії алгоритмів», «Алгоритмічні моделі. Обчислювальні функції».

У навчально-методичному посібнику разом з коротким теоретичним викладом відомостей з математичної логіки також наведені типові приклади розв’язання задач та вправи для самостійного роботи по кожній темі.

Додатково необхідну теорію для розв’язання задач можна знайти в книгах із запропонованого в кінці списку літератури.

Навчально-методичний посібник призначений для самостійної та індивідуальної роботи студентів математичних, технічних та економічних спеціальностей, які вивчають математичну логіку, а також може бути використаний як план-конспект для проведення практичних занять з дисципліни «Математична логіка та теорія алгоритмів» у навчальних закладах різних профілів і рівнів.

ІСТОРИЧНИЙ НАРИС РОЗВИТКУ МАТЕМАТИЧНОЇ ЛОГІКИ

Логіка (грец. *Logika*) — наука про прийнятні способи міркувань. Слово “логіка” у його сучасному використанні багатозначне, хоча й не так різноманітне змістовними відтінками, як древньогрецьке *logos*, від якого воно походить. Традиційно з поняттям логіки зв’язують три основних аспекти: онтологічний – “логіка речей”, тобто необхідний зв’язок явищ об’єктивного світу (Демокріт); гносеологічний – “логіка знань”, тобто необхідний зв’язок понять, за допомогою яких пізнається “суть та істина” (Платон) і демонстраційний (довідний), або власне логічний, – “логіка доведень і заперечень”, тобто необ-хідний зв’язок думок (висловлень) у міркуваннях (висновках), примусова переконливість яких випли-ває тільки з форми цього зв’язку, не враховуючи, чи визна-чають ці думки “суть та істинність”, чи ні (Арістотель). Перші два аспекти відносяться до філософії й діалектичної логіки, останній же аспект утворює власне логіку, або сучасну логіку (яку слідом за І.Кантом іноді називають формальною логікою).

Виникнення логіки як науки бере початок з часів Арісто-теля (384-322 рр. до н.е.) – великого грецького мислителя, який вперше виділив і систематизував загальнозначущі (у деякому розумінні універсальні для людського мислення) способи міркувань.

Сучасна логіка розвинулася у точну науку, яка застосовує математичні методи. Вона стала, за словами П.С. Порецького (1846-1907), математичною логікою – логікою за предметом, математичною за методом.

Математичною логікою називають розділ математики, який вивчає математичні доведення і основи математики.

Отже математична логіка виникла на певному етапі розвитку формальної логіки, внаслідок застосування до неї математичної символіки та математичних методів. Предметом же вивчення формальної логіки є форми вірних умовиводів. Тому, перше завдання логіки – виявлення правильних і хибних способів міркувань. Найпростіші і найбільш уживані логічні правила під назвою фігур силогізмів були описані в 6а циклюван трактаті Арістотеля “Аналітики” (IV 6а . до н.е.), який до середини XIX століття був найбільш авторитетним твором з логіки.

Математична логіка із зовнішнього боку відрізняється від “звичайної” тим, що вона широко користується мовою математичних і логічних символів, виходячи з того, що вони можуть зовсім замінити слова звичайної мови і прийнятні у звичайних живих мовах способи об’єднання слів у реченнях. Ідея про те, що, записуючи всі вихідні припущення мовою спеціальних знаків, подібних до математичних, можна замінити міркування обчисленням, виникла дуже рано (Р.Луллій, 1235-1315 рр.). Але більш визначений і близький до запровадженого пізніше спосіб універсального логічного числення висунув і розвивав Г.Лейбніц (1646-1716 рр.). Лейбніц навіть надіявся, що в майбутньому філософи замість того, щоб марно сперечатися, будуть брати папір і обчислювати, хто з них правий.

Початок формуванню того апарату математичної логіки, який тепер ми називаємо логікою висловлень, поклав Дж.Буль (1815-1864). Логіко-математичні мови і їх змістовні теорії були потім значно розвинені у роботах Фреге (1848-1925рр.) У роботах Пеано (1858-1932 рр.) і, особливо, у фундаментальній монографії Рассела і Уайтхеда, виданій у 1910-1913 рр., були викладені великі розділи математики на мові математичної логіки.

У 20-х роках нашого століття з програмою 6а циклювання математики на базі математичної логіки виступив відомий математик Д.Гільберт (1862-1943 рр.). З цього часу і починається сучасний етап розвитку математичної логіки, який характеризується застосуваннями точних математичних методів при вивченні формальних аксіоматичних теорій.

Таким чином, у сучасної математичної логіки предметом дослідження часто є математичні теорії, такі як математичний аналіз, алгебра, елементарна геометрія, арифметика та 6а . У логіці математичні теорії вивчаються в цілому – і це одна з особливостей математичної логіки в порівнянні з іншими математичними дисциплінами.

Перш за все, математичну теорію уточнюють і описують на базі строгої логіко-математичної мови. Цей етап називається формалізацією теорії і складає важливу, хоча й попередню, частину дослідження теорії. Після формалізації до отриманої формальної аксіоматичної теорії вже можна застосовувати математичне дослідження, можна поставити точні проблеми, отримати математичні результати.

Які ж питання ставляться відносно теорії в цілому ?

По-перше, чи є теорія несуперечною? (тобто чи не виводиться в даній теорії одночасно деяке твердження і його заперечення). Так, за допомогою методу 7а циклювання Келі і Клейн показали, що геометрія Лобачевського несуперечна, якщо несуперечна звичайна евклідова геометрія.

Важливе враження на сучасників справило відкриття на початку нашого століття Кантором і Расселом парадоксів в теорії множин. Це відкриття говорило про те, що теорія множин з її широкими застосуваннями у “наївному” викладі є суперечливою теорією. Вивчення цього явища в значній мірі дало поштовх розвитку сучасних методів математичної логіки. Була сформульована аксіоматична теорія Цермело-Френкеля, в якій звичайні способи виводу парадоксів вже неможливі. Програма Гільберта 7а циклювання математики фінітними засобами також в значній мірі пов’язана з відкриттям парадоксів.

Відома друга теорема Геделя стверджує, що несуперечність достатньо багатої теорії не може бути встановлена засобами самої теорії. Дуже багато досліджень з некласичних, модельних і інтуіціоністських логік стимульовані цією ідеєю.

До нашого часу несуперечність таких теорій, як елементарна геометрія, арифметика, аналіз добре вивчена і досить надійно 7а циклюванн. Але ж існують аксіоматичні теорії (система Цермело-Френкеля, теорія Куайна), де питання несуперечності ще відкрито.

Другим питанням про теорію в цілому є питання про повноту тої чи іншої теорії. У багатьох математичних теоріях час від часу виникають конкретні проблеми, які не вдається ні довести, ні спростувати. Теорема Геделя про неповноту стверджує, що будь-яка досить багата теорія необхідно містить твердження, які не можна ні довести, ні спростувати в рамках теорії.

Але деякі важливі теорії виявляються повними. Наприклад, елементарна геометрія, теорія векторних просторів.

Нарешті, дуже важливо буває дослідити розв’язність тої чи іншої теорії. Так, Тарський у 1948р. побудував конкретний алгоритм, що дозволяв для будь-якого твердження елементарної геометрії з’ясувати, чи є це твердження істинним або хибним.

З іншого боку, логіки вміють доводити, що багато теорій, наприклад, арифметика, аналіз, теорія множин, нерозв’язні, тобто що не існує алгоритму, який би давав можливість для будь-якого твердження теорії з’ясувати, істинне воно чи хибне. Питання про існування тих чи інших алгоритмів займає важливе місце у дослідженнях логіків.

АЛГЕБРА ВИСЛОВЛЕНЬ

Теоретичні відомості

1. Висловлювання та операції над ними

Означення. Висловленням у логіці називається оповідне речення, відносно якого можна сказати істинне воно чи хибне.

Прості висловлювання породжують більш складні за допомогою логічних зв'язок або операцій.

Назва логічної операції	***Позначення***	***Мовний еквівалент***
Заперечення	$\neg A, \overline{A}$	„ невірно, що A”, „ не A”
Кон'юнкція	$A \& B, A \wedge B, AB$	„A і B”
Диз'юнкція	$A \vee B$	„A або B”
Імплікація	$A \supset B, A \to B, A \Rightarrow B$	„Якщо A то B”,” з A випливає B, ”A достатня для B”, „B необхідно для A”
Еквівалентність	$A \equiv B, A \leftrightarrow B, A \Leftrightarrow B$	„A тоді і тільки тоді, коли B”, „Для A необхідно і достатньо B”

2. Синтаксис формальної мови. Правила побудови формул

Алфавіт складається:

- ✓ з скінченого або зліченого числа літер, які позначаються маленькими латинськими літерами. Кожна літера приймає одне із 2-х значень: «так»/1 або «ні»/0. Літери в цьому випадку називають простими(елементарними, атомарними) висловлюваннями
- ✓ з логічних операцій(зв'язок)
- ✓ з допоміжних символів – дужок (,)

З простих (атомарних) висловлювань складають складні (молекулярні) висловлювання за допомогою логічних зв'язок(операцій).

Правила побудови формул.

У логіці висловлювань правильно побудованими формулами (ППФ) є наступні:

- ✓ атомарна формула є ППФ;
- ✓ якщо А – ППФ, то заперечення А – ППФ;
- ✓ А і В – ППФ, то і $A \wedge B, A \vee B, A \to B, A \leftrightarrow B$ - також ППФ;
- ✓ Формули, які не задовольняють умови 1-3 не є ППФ.

Зауваження. ППФ також називають словами.

3. Обчислення значень формул за допомогою таблиць або дерев істинності

Обчислення значень формул здійснюється за допомогою таблиць або дерев істинності.

Таблиця істинності для основних логічних операцій

a	B	$\bar{b}$	$a \wedge b$	$a \vee b$	$a \to b$	$a \leftrightarrow b$
0	0	1	0	0	1	1
1	0	1	0	1	0	0
0	1	0	0	1	1	0
1	1	0	1	1	1	1

Якщо відоме значення атомарних формул, то значення молекулярної формули визначають безпосередньо за значеннями логічних операцій. Якщо значення пропозиційних змінних невідоме, то будується таблиця істинності.

Послідовність виконання логічних операцій також зручно записувати у вигляді дерев істинності.

4. Інтерпретація формул. Методи перевірки тотожної істинності формул

Означення. Присвоєння значень 0 або 1 атомарним формулам, що входять в молекулярні називається інтерпретацією.

Означення. Відображення $h : A \to \{0,1\}$ називається інтерпретацією, якщо для будь-яких формул A і B числення висловлювань відображення h задовольняє наступні вимоги:

1. $h(\overline{A}) = \neg h(A)$;
2. $h(A \wedge B) = h(A) \wedge h(B)$.

Отже, якщо задати інтерпретацію $h(A \wedge B) = h(A) \wedge h(B)$, то за допомогою таблиці істинності або дерев істинності можна обчислити значення будь-якої складеної формули.

Означення. Кажуть, що формула A числення висловлювань істинна при деякій інтерпретації тоді, коли h(A)=1. У протилежному випадку кажуть, що h – хибна.

Означення. Якщо h(A)=1, то інтерпретацію h називають моделлю для формули A.

Означення. Формула A називається ***тавтологією*** (***суперечністю***), якщо вона приймає значення 1 (0), незалежно від інтерпретації. Якщо формула $A \to B$ - тавтологія, то кажуть, що з формули A логічно слідує формула B або формула B логічний (семантичний) наслідок формули A у численні висловлювань. Якщо $A \leftrightarrow B$ - тавтологія, то кажуть, що формули A і B логічно (семантично) еквівалентні числові висловлювання.

Таблиці та дерева істинності дають, але досить громіздку процедуру для перевірки значень формул.

Будемо називати цей спосіб *тривіальним*.

Алгебраїчний метод

Алгебраїчний метод ґрунтується на застосуванні тотожностей 11а цикл алгебри для спрощення формул числення висловлювань.

Зауваження. Множину S виконуваних формул числення висловлювань можна розглядати як універсальну алгебру з двома операціями: унарною – заперечення і бінарною – кон’юнкція.

Для відповідної перевірки обирають найбільш прості формули. Алгебраїчний метод дає можливість з нескінченої множини формул, які можна побудувати, виділити скінчену множину логічно різних формул (2^{2^n}).

Метод Куайна

Метод Куайна (узагальнення тривіального алгоритму). Нехай $\{p, q, \ldots, r\}$ - впорядкована множина висловлювань з формули А. Розглянемо перше висловлювання p і припишемо йому значення 1 або 0 (h(p)=1(0)). Підставимо це значення у формулу А і виконаємо обчислення. Отримаємо формулу А’, яка складається з $\{q, s, \ldots, r\}$ і т.д. На деякому кроці буде одержана формула $A^{(n)}$, яка є тавтологією або суперечністю. На цьому кроці робота алгоритму зупиняється.

Метод редукції

Метод редукції дає можливість виконати перевірку за допомогою зведення формули до суперечності. Він особливо зручний, коли у формулі багато імплікацій. Нехай $A = A_1 \to A_2$. Припустимо, що в деякі інтерпретації h(A)=0, тоді $h(A_1) = 1$ і $h(A_2) = 0$. Таким чином, перевірка значень формули А зводиться до перевірки значень формул A_1 та A_2.

Метод резолюцій

Будемо називати літерою атом або заперечення атома. *Диз’юнктом* називається диз’юнкція літер. Одиничним диз’юнктом назвемо 12а циклювання диз’юнкт. Якщо диз’юнкт не має жодної літери, то він називається порожнім і позначається 0. Літери r і $\bar{r}$ називають контрарними.

Правило 13а циклювання диз'юнктів. Якщо існує одиничний диз'юнкт L в множини диз'юнктів S, то S' одержано з S, викреслюванням тих диз'юнктів, що містять $\overline{L}$.

Якщо S' – порожня, то S – істинна. У протилежному випадку буде утворена S'' шляхом вилучення з S' всіх диз'юнктів такого типу. S'' – суперечність тоді і тільки тоді, коли S – суперечність.

Зауваження. Якщо $\overline{L}$ - одиничний диз'юнкт, то при його викреслені він перетворюється у порожній диз'юнкт.

Правило резолюції. Для будь-яких диз'юнктів C і C', якщо існує літера L в C, яка контрастна літері L' в C', то викресливши літери L і L' з C і C' відповідно, будуємо диз'юнкцію літер, що залишилися. Побудований диз'юнкт називається резольвентою C і C'.

Твердження. Нехай дано два диз'юнкти C_1 і C_2, тоді резольвента C диз'юнктів C_1 і C_2 є логічним наслідком C_1 і C_2.

Нехай S – множина диз'юнктів. Резолютивним виведенням диз'юнкта C з S називається скінчена послідовність диз'юнктів $C_1, C_2, ..., C_k$ така, що кожний диз'юнкт $C_i \in S$ або є резольвентою попередніх диз'юнктів. Виведення 0 з S називається доведенням суперечності S або спрощення S резолютивні. Кажуть, що диз'юнкція може бути виведена з S, якщо можливе виведення C з S.

<u>*Контрольні запитання*</u>

1. Коли формальна теорія вважається визначеною?
2. Що розуміється під синтаксичним аспектом вивчення формальної теорії?
3. Алфавіт і правила побудови формул у численні висловлювань.
4. Яка формула називається:

а) тавтологією;

б) суперечністю;

в) нейтральною;

г) виконуваною?

5. Методи перевірки тотожної істинності формул числення висловлювань:

а) тривіальний (таблиці та дерева істинності);

б) алгебраїчний метод;

в) метод Куайна;

г) метод редукції;

д) метод резолюцій.

<u>Приклади розв'язування типових задач</u>

<u>*Приклад 1.*</u> Знайти значення істинності формули

$a \leftrightarrow b \wedge \neg c \to b \vee \neg a \wedge c \vee a \wedge \neg b$ при $|a| = |b| = 1$, $|c| = 0$

Розв'язок:

Перепишемо формулу і позначимо внизу порядок виконання операцій

$u \equiv a \underset{10}{\leftrightarrow} b \underset{4}{\wedge} \underset{1}{\neg} c \underset{9}{\to} b \underset{7}{\vee} \underset{2}{\neg} a \underset{5}{\wedge} c \underset{8}{\vee} a \underset{6}{\wedge} \underset{3}{\neg} b$

Запишемо значення відповідних 14а циклюва при заданому значенні пропозиційних букв:

$|b \wedge \neg c| = |1 \wedge \neg 0| = 1$, $|\neg a \wedge c| = |\neg 1 \wedge 0| = 0$, $|a \wedge \neg b| = |1 \wedge \neg 1| = 0$,

$|b \vee \neg a \wedge c \vee a \wedge \neg b| = |1 \vee 0 \vee 0| = 1$, $|b \wedge \neg c \to b \vee \neg a \wedge c \vee a \wedge \neg b| = |1 \to 1| = 1$.

Отже, оскільки $|1 \to 1| = 1$, то $|u| = 1$

<u>*Приклад 2*</u>. Побудувати таблицю істинності і вказати тип формули

$a \to \neg(b \wedge c)$

Розв'язок:

Аналізуємо значення істинності розглядуваної формули при всіх можливих значеннях пропозиційних букв. Отримані результати зводимо в таблиці істинності:

a	b	c	$(b \wedge c)$	$\neg(b \wedge c)$	$a \to \neg(b \wedge c)$
0	0	0	0	1	1
0	0	1	0	1	1
0	1	0	0	1	1

0	1	1	1	0	1
1	0	0	0	1	1
1	0	1	0	1	1
1	1	0	0	1	1
1	1	1	1	0	0

Оскільки, формула приймає як значення істинності, так і значення хибності, то вона є нейтральною.

Приклад 3. Способом відшукання 15а циклювання15 встановити, що запропонована формула алгебри висловлень є логічно істинною

$(a \to b) \wedge (c \to d) \to (a \vee c \to b \vee d)$

Розв'язок:

Нехай $U \equiv (a \to b) \wedge (c \to d)$ та $V \equiv a \vee c \to b \vee d$. Тоді дану в умові задачі формулу можемо записати у вигляді $U \to V$ і вона не є логічно істинною тоді, коли $|U| = 1$ і $|V| = 0$.

Припустимо, що існує такий набір значень пропозиційних букв при якому $|U \to V| = 0$. Тоді $|(a \to b) \wedge (c \to d)| = 1$ і виходячи з означення кон'юнкції маємо $|(a \to b)| = 1$ та $|(c \to d)| = 1$.

З іншого боку $|a \vee c \to b \vee d| = 0$, а отже $|a \vee c| = 1$ та $|b \vee d| = 0$. Виходячи з означення диз'юнкції робимо висновок, що $|b| = 0$ та $|d| = 0$.

Із $|(a \to b)| = 1$ і $|b| = 0$ випливає, що $|a| = 0$, аналогічно із $|(c \to d)| = 1$ і $|d| = 0$ слідує, що $|c| = 0$, а це суперечить висновку, що $|a \vee c| = 1$.

Отже, прийшовши до суперечності, ми встановили, що досліджувана формула є тавтологією.

Приклад 4. Без побудови таблиці істинності встановити тип формули $a \wedge b \vee c \wedge d \to (a \vee b) \wedge (c \vee d)$

Розв'язок:

Припустимо, що формула не є тавтологією, тоді аналогічно попередньому прикладу $|a \wedge b \vee c \wedge d| = 1$ і $|(a \vee b) \wedge (c \vee d)| = 0$

Нехай $|(a \vee b)| = 0$, тоді $|a| = |b| = 0$, проте коли $|c| = |d| = 1$ виконується якраз співвідношення $|a \wedge b \vee c \wedge d| = 1$

Отже, наше припущення вірне і задана формула не є тавтологією. Але у випадку, коли $|a| = |b| = |c| = |d| = 1$ формула буде нейтральною.

Вправи для самостійного розв'язування

1. Знайти значення істинності формули:

1) $\neg(a \to c) \wedge (\neg b \vee (\neg c \to a))$ при $|a| = 1$, $|b| = 0$, $|c| = 1$;

2) $\neg(a \leftrightarrow b) \wedge c \to \neg a \vee (b \leftrightarrow c)$ при $|a| = 0$, $|b| = 1$, $|c| = 0$;

3) $(a \to \neg b) \wedge b \to \neg c \vee a \wedge b$ при $|a| = 1$, $|b| = 0$, $|c| = 1$.

2. Знайти значення істинності складного висловлення: «Якщо ми дістанемо путівки (П), то поїдемо до 16а циклюванн (Т) і ми або дістанемо путівки, або будемо відпочивати у родичів у селі (Р) тоді і тільки тоді, коли вони повернуться з відпустки (В)».

Значення істинності атомарних висловлень задано так $|\ddot{I}\ | = 0$, $|Đ| = 1$, $|\grave{O}| = 0$, $|\hat{A}| = 1$

3. Скласти таблиці істинності для формул алгебри висловлень:

1) $\neg a \vee b \leftrightarrow a \wedge \neg c$;

2) $(a \to b) \to ((b \to c) \to (c \to a))$;

3) $(a \leftrightarrow \neg b) \wedge (a \wedge b \vee \neg a \wedge \neg b)$;

4) $\neg b \vee \neg c \vee b \wedge c$;

5) $a \wedge b \vee \neg a \wedge \neg b \vee \neg a \wedge b \vee a \wedge \neg b$;

6) $(a \leftrightarrow b \wedge c) \leftrightarrow (a \leftrightarrow b) \wedge (a \leftrightarrow c)$.

4. Способом відшукання а циклювання встановити, що запропоновані формули алгебри висловлень – логічно істинні:

1) $(a \to b) \wedge (c \to d) \to (a \vee c \to b \vee d)$;

2) $a \vee b \to ((a \to c) \wedge (b \to d) \to c \vee d)$;

3) $(a \to (b \to c)) \to ((a \to b) \to (a \to c))$;

4) $(a \to (b \to c)) \to ((c \to d) \to (a \to (b \to d)))$.

1. Показати, що формула алгебри висловлень є не виконуваною:

$(a \to b) \wedge (a \to \neg b) \wedge (b \to c) \wedge (b \to \neg c) \wedge (c \to a) \wedge (c \to \neg a)$.

6. Способом відшукання а циклювання переконатися, що дані формули алгебри висловлень є суперечностями:

1) $\neg b \wedge a \wedge (a \to b)$;

2) $a \vee b \leftrightarrow \neg a \wedge (b \to \neg b)$;

3) $(a \to (b \to c)) \wedge (a \to b) \wedge a \wedge \neg c$.

7. Визначити типи вказаних формул:

1) $a \wedge c \vee b \wedge d \to (a \vee b) \wedge (c \vee d)$;

2) $(a \vee b \to c) \to (a \to c) \vee (b \to c)$;

3) $(a \wedge b \to c) \to (a \to c) \wedge (b \to c)$;

4) $(\neg a \vee b) \wedge (\neg b \vee c) \wedge a \wedge \neg c$;

5) $(a \to b) \wedge (c \to d) \to (a \wedge c \to b \wedge d)$;

6) $(a \wedge c \to b \wedge d) \to (a \to b) \wedge (c \to d)$;

7) $((a \leftrightarrow b) \to (c \leftrightarrow d)) \to (a \vee c \to b \vee d)$;

8) $(a \to b) \to ((c \leftrightarrow d) \to (a \vee c \leftrightarrow b \vee d))$.

8. Довести чи спростувати твердження:

1) Якщо формула U алгебри висловлень є тавтологією, то $\neg U$ є суперечністю.

2) Із двох формул U, $\neg U$ алгебри висловлень хоч одна логічно істинна.

3) Якщо формула U - виконувана формула алгебри висловлень, то $\neg U$ є не виконуваною.

4) Із двох формул U, $\neg U$ алгебри висловлень хоч одна є виконуваною.

5) Якщо $U \leftrightarrow V$ - виконувана формула, то U та V формули виконувані.

6) Якщо U та V - тавтології, то $V \leftrightarrow V$ - тавтологія. Чи правильне обернене твердження?

7) Якщо U та V - виконувані формули, то $U \wedge V$ теж виконувана формула. Чи вірне обернене твердження?

8) Якщо $U \leftrightarrow V$ - тавтологія, то U і V - тавтології. Чи вірне обернене твердження?

9) Якщо U та V - тавтології, то $U \vee V$ - тавтологія. Чи правильне обернене твердження?

9. Виразіть за допомогою логіки висловлювань речення природної мови:

а) „Якщо „Спартак" або „Баварія" програють, а „Динамо" виграє, то „Парі Сен Жермен" втратить перше місце і, крім того, Сергій програє парі".

Б) „Йде дощ або хтось не вимкнув душ".

В) „Якщо у вечері буде туман, то Микола або залишиться вдома, або буде вимушений скористатися таксі".

Г) „Анатолій залишиться і він або Сергій будуть чекати".

Д) „Анатолій залишиться і буде чекати або Сергій буде чекати".

Е) „Я поїду або на автобусі, або на таксі".

Є) „Ні Північ, ні Південь не здобули перемоги в громадянській війні".

Ж) „Пшениця збережеться тоді і тільки тоді, коли мінеральні добрива будуть розумно використані; якщо пшениця не збережеться, то фермери збанкрутують і залишать свої ферми".

З) „ Якщо я втомлений або голодний, то я не можу працювати".

Й) „Якщо Микола встане і піде до школи, то він буде задоволений, а якщо він не встане, то він не буде задоволений".

І) ”Якщо містер Джонс щасливий, то місіс Джонс нещаслива і якщо містер Джонс нещасливий, то місіс Джонс щаслива”.

10. Встановити тип формули за допомогою алгебраїчного методу та таблиць чи дерев істинності:

а) $a \wedge c \vee \neg b \wedge c \vee \neg a \wedge b \vee a \wedge \neg c$;

б) $\neg a \wedge b \wedge c \vee a \wedge \neg b \wedge c \vee a \wedge b \wedge c$;

в) $(\neg a \rightarrow b) \wedge c \leftrightarrow (a \leftrightarrow b)$;

г) $((\neg a \rightarrow b) \rightarrow b \wedge c) \wedge (b \rightarrow a)$.

11. Доведіть, що наведені нижче формули є тавтологіями:

а) $(p \rightarrow q) \vee (q \rightarrow p)$

б) $p \rightarrow (q \rightarrow (p \wedge q))$

в) $(p \rightarrow q) \rightarrow ((p \rightarrow (q \rightarrow r) \rightarrow (p \rightarrow r))$

г) $(\neg p \rightarrow \neg q) \rightarrow (q \rightarrow p)$

12. Перевірити суперечність множини диз’юнктів:

а) $\{\neg p, p \vee \neg q, p \vee q \vee r, \neg r\}$

б) $\{p \vee q, \neg p \vee q, \neg q\}$

в) $\{p \vee \neg q, q \vee r, \neg r \vee p, \neg p\}$

г) $\{p, \neg p \vee q \vee r, \neg q \vee c, \neg r \vee c, \neg c\}$

13. Скласти вихідну контактну схему, спростити її за допомогою алгебри висловлювань та скласти спрощену контактну схему для формул, наведених нижче:

а) $\neg(a \rightarrow b) \wedge \neg(a \wedge b) \vee b$

б) $a \wedge c \vee \neg b \wedge c \vee \neg a \wedge b \vee a \wedge \neg c$

в) $\neg a \wedge b \wedge c \vee a \wedge \neg b \wedge c \vee a \wedge b \wedge c$

г) $(\neg(b \vee c) \rightarrow b \wedge c \wedge d) \vee \neg b \wedge d$

ЧИСЛЕННЯ ВИСЛОВЛЕНЬ

<u>Теоретичні відомості</u>

Формальна теорія вважається визначеною коли задано: мова (деякий злічений алфавіт і правило побудови формули); система аксіом і правил виведення (логіка числення висловлювань).

У загальному сенсі аксіоми – це твердження про задані об’єкти і зв’язки, між ними сформовані у вигляді рядків деякої мови. Для числення висловлювань аксіоми – це найпростіше „істинне” висловлювання. Якщо є можливість перевірити чи є дана формула аксіомою, то теорію називають ефективно аксіоматизованою теорією.

Для числення висловлювань аксіомами будуть наступні формули:

$A_1 . a \to (b \to a);$

$A_2 .(a \to b) \to (a \to (b \to c) \to (a \to c));$

$A_3 .(a \wedge b) \to a;$

$A_4 .(a \wedge b) \to b;$

$A_5 .(a \to b) \to ((a \to c) \to a(b \wedge c));$

$A_6 . a \to (a \vee b);$

$A_7 . b \to (a \vee b);$

$A_8 .(a \to c) \to ((b \to c) \to ((a \vee b) \to c));$

$A_9 .(a \to \bar{b}) \to (b \to \bar{a});$

$A_{10}\, \bar{a} \to a.$

Правила виведення у загальному сенсі називається чітко сформульовані процедури, які дають змогу утворювати нові рядки ті, що існують. Рядки називаються теоремами даних теорій.

<u>Для</u> <u>числення</u> <u>висловлювань</u> <u>правилами</u> <u>виведення</u> є:

1. Правило підстановки. Якщо А вивідна формула, яка містить літеру p, то вивідною є формула A(B), яка здобувається з формули A(p), заміною всіх входжень літери в довільну формулу B.

2. Правило висновку (modus pones). Якщо А і $A \to B$ - вивідні формули, то вивідною є і формула В.

Зауваження. У поданому описі числення висловлювань аксіомами є формулами числення; формулами, що використовують правила виведення є метаформули або так звані схеми формул.

Означення. Схема формул – це вираз метамови, для позначення нескінченої множини всіх тих формул числення, які отримують після заміни метамови цієї схеми повними формулами числення.

У численні висловлювань правилами виведення є множина відношень $R_1, R_2, ..., R_m$ між формулами.

Означення. Якщо формула А і формули $A_1, A_2, ..., A_n$ знаходяться у деякому відношенні R, то А називається безпосереднім висновком з формул $A_1, A_2, ..., A_n$ сформульованим за правилом R.

Виведенням є будь-яка послідовність формул $A_1, A_2, ..., A_n$, в якій для будь-якого і A_i є, або аксіомою, або безпосереднім висновком з деяких попередніх формул.

Означення. Формула А називається теоремою формальної теорії, коли існує виведення, в якому останньою формулою є формула А.

Означення. Формула А називається висновком з множини формул Г, коли існує така послідовність формул $A_1, A_2, ..., A_n$, що $A_n = A$ і будь-яка з формул A_i є, або аксіомою або формулою з Г, або безпосереднім висновком деяких попередніх формул. Елементи Г називаються гіпотезами (посилками виведення).

Якщо Г – пуста, то А – аксіома.

Теорема дедукції

Теорема. (дедукції). Нехай Г – множина формул, а А і В – формули, тоді якщо Г, $A|\!=B$, то $Г|\!=A \to B$.

Зауваження. Теорема дедукції – це метамова числення висловлення, яка може розглядатися як правило побудови формул.

<u>Деякі</u> <u>застосування</u> <u>теореми</u> <u>дедукції</u>:

1. Правило силогізму: $A \to B, B \to C|\!=A \to C$

2. Правило виведення заперечення. $\dfrac{Г, A|\!=B, Г, A|\!=\overline{B}}{Г|\!=\overline{A}}$

3. а) $A \to (B \to C), B|\!=A \to C$

б) $|\!=\neg\overline{A} \to A$

в) $|\!=\rangle A \to A$

<u>Контрольні запитання</u>

1.Формалізація числення висловлювань.

2. Властивості числення висловлювань.

3. Дати означення формального доведення формули.

4. Сформулюйте аксіоми числення висловлень.

5. Сформулюйте правила виведення.

6. Дати означення вивідної формули в численні висловлень.

7. Сформулюйте та доведіть теорему дедукції.

<u>Зауваження. В якості системи аксіом ЧВ користуємось наступною:</u>

$I_1\ A \to (B \to A)$ $I_2\ (A \to (B \to C)) \to ((A \to B) \to (A \to C))$ $I_3 (\neg B \to \neg A) \to ((\neg B \to A) \to B)$

$II_1\ A \wedge B \to A$ $II_2\ A \wedge B \to B$ $II_3\ (A \to B) \to ((A \to C) \to (A \to B \wedge C))$

$III_1\ A \to A \vee B$ $III_2\ B \to A \vee B$ $III_3\ (A \to C) \to ((B \to C) \to (A \vee B \to C))$

$IV_1\ (A \to B) \to (\neg B \to \neg A)$ $IV_2\ A \to \neg\neg A$ $IV_3\ \neg\neg A \to A$

Правила виведення позначаємо:

1. Правило висновку: $\frac{A \to B, A}{B}(MP)$ (modus ponens)

2. Правило підстановки: $A(b) \to A(B)(S_b^B A)$ (*в* – пропозиційна буква, В – довільна формула числення висловлень)

Приклади розв'язування типових задач

Приклад 1. Вказати які з наступних формул являються аксіомами ЧВ:

а) $(A \to ((A \to A) \to A) \to ((A \to (A \to A)) \to (A \to A))$

Розв'язок:

Дана формула являється аксіомою A_2, якщо в ній в якості формули A взяти саму A, в якості B– $(A \to A)$, а в якості C – A.

б) $A \to ((\neg A \to B) \to A)$

Розв'язок:

Дана формула являється аксіомою A_1, якщо вважати $A \equiv A, B \equiv \neg A \to B$.

Приклад 2. Вказати відсутню формулу W так, щоб третю формулу з даних формул ЧВ можна було б отримати з першої і другої за правилом виведення:

$$A \to (C \to A), \left(A \to (C \to A)\right) \to \left(A \to \left(B \to (C \to A)\right)\right), W$$

Розв'язок:

Якщо в схемі правила виведення МР прийняти за A формулу $A \to (C \to A)$, то друга формула даної задачі може виступати в якості формули $A \to B$ правила МР, де B є формулою $A \to \left(B \to (C \to A)\right)$. За правилом МР воня являється результатом застосування цього правила до двох даних формул, тобто є шуканою формулою W.

Приклад 3. З'ясувати чи являється задана послідовність формул висновком з аксіом:

(1) $B \to (A \to B)$,

(2) $(B \to (A \to B)) \to (B \to (B \to (A \to B)))$,

(3) $B \to (B \to (A \to B))$.

Розв'язок: Задана послідовність формул являється висновком аксіом, тому що кожна із формул (1), (2) являє собою аксіому A_1, а формул (3) отримана з (1) і (2) шляхом застосування правила MP.

<u>*Приклад 4.*</u> Довести формальну теорему, використовуючи аксіоми ЧВ:

$A \to A$

Розв'язок:

1. $(A \to ((A \to A) \to A) \to ((A \to (A \to A)) \to (A \to A))$ (A_2)
2. $A \to ((A \to A) \to A)$ (A_1)
3. $(A \to (A \to A)) \to (A \to A)$ $(з\ 1\ і\ 2\ за\ MP)$
4. $A \to (A \to A)$ (A_1)
5. $A \to A$ $(з\ 3\ і\ 4\ за\ MP)$

<u>*Приклад 5.*</u> Користуючись теоремою дедукції довести, що

$A \to B, B \to C \vdash A \to C$

Розв'язок:

Доведемо спочатку, що $A \to B, B \to C, A \vdash C$.

Отже, маємо:

1. A (гіпотеза)
2. $A \to B$ (гіпотеза)
3. B (MP, 1, 2)
4. $B \to C$ (гіпотеза)
5. C (MP, 3,4)

Таким чином, ми довели $A \to B, B \to C, A \vdash C$. Застосувавши тепер до $A \to B, B \to C, A \vdash C$ теорему дедукції, отримаємо $A \to B, B \to C \vdash A \to C$, що і треба було довести.

Вправи для самостійного розв'язування

1. Вказати які з наступних формул являються аксіомами ЧВ:

1) $(B \to C) \to (A \to (B \to C))$;

2) $(\neg A \to B) \to (\neg B \to \neg\neg A)$;

3) $(\neg A \to \neg\neg B) \to ((\neg F \to \neg B) \to A)$;

4) $(B \to A) \to ((B \to \neg A) \to \neg B)$;

5) $(A \to B) \to (C \to (B \to A))$;

6) $(B \to (\neg A \to A)) \to ((B \to \neg A) \to (B \to A))$;

7) $(\neg\neg\neg A \to \neg A) \to ((\neg\neg\neg A \to A) \to \neg\neg A)$.

2. Вказати відсутню формулу W так, щоб третю формулу з даних формул ЧВ можна було б отримати з першої і другої за правилом виведення:

1) $A \to (B \to C),\ W,\ (A \to B) \to (A \to C)$;

2) $W,\ (\neg B \to \neg B) \to \big((\neg B \to B) \to B\big),\ (\neg B \to B) \to B$;

3) $A \to B,\ W,\ C \to (A \to B)$;

4) $B,\ B \to (A \to B),\ W$;

5) $W,\ (\neg A \to \neg B) \to \big((\neg A \to B) \to A\big),\ (\neg A \to B) \to A$;

6) $B \to A,\ W,\ (B \to \neg A) \to (B \to A)$;

7) $\neg B \to (A \to \neg B), \big(\neg B \to (A \to \neg B)\big) \to B, W$.

3. З'ясувати чи являється задана послідовність формул висновком з аксіом:

а) (1) $(\neg B \to \neg A) \to ((\neg B \to A) \to B)$,

(2) $\neg A \to (\neg B \to \neg A)$,

(3) $\neg A \to ((\neg B \to A) \to B)$.

Б) (1) $A \to (B \to A)$,

(2) $(A \to (B \to A)) \to ((A \to B) \to (A \to A))$,

(3) $(A \to B) \to (A \to A)$.

В) (1) $(A \to A) \to (B \to (A \to A))$,

(2) $(A \to A)$,

(3) $B \to (A \to A)$.

Г) (1) $\neg B \to \neg B$,

(2) $(\neg B \to \neg B) \to ((\neg B \to B) \to B)$,

(3) $(\neg B \to B) \to B$.

4. Довести формальні теореми, використовуючи аксіоми ЧВ:

1) $B \to (A \to A)$;

2) $(A \to B) \to (A \to A)$;

3) $B \to (B \to (A \to B))$;

4) $B \to (A \to (C \to A))$;

5) $A \to (B \to (C \to A))$;

6) $(\neg B \to B) \to B$;

7) $\neg B \to ((\neg A \to B) \to A)$.

5. Використовуючи теорему дедукції, довести, що наступні формули являються теоремами формального числення висловлень:

1) $\neg A \to (A \to B)$;

2) $(\neg B \to \neg A) \to (A \to B)$;

3) $A \to ((\neg B \to \neg A) \to B)$;

4) $(A \vee A) \to A$;

5) $A \to (\neg B \to \neg(A \to B))$;

6) $(A \to B) \to ((\neg A \to B) \to B)$;

7) $(A \to B) \to ((A \to \neg B) \to \neg A)$.

6. Користуючись теоремою дедукції довести:

1) $A \vdash (A \to B) \to B$; 2) $A \vdash A \vee B$; 3) $B \vdash A \vee B$;

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

5) $\neg B \to \neg A \vdash A \to B$;

6) $A \to B,\ \neg A \to B \vdash B$;

7) $A \vee B,\ A \to C,\ B \to C \vdash C$.

АНАЛІЗ ЗМІСТОВНИХ МІРКУВАНЬ НА БАЗІ ЛОГІКИ ВИСЛОВЛЕНЬ. ЛОГІЧНІ ЗАДАЧІ

Теоретичні відомості

Логічне слідування на базі алгебри висловлень

Означення. Говорять, що формула $\Omega(A_1, \dots, A_n)$, яка містить елементарні висловлення $A_1, \dots, A_n$, логічно слідує з формул $\Lambda_1(A_1, \dots, A_n)$, $\dots, \Lambda_m(A_1, \dots, A_n)$ (записують $\Lambda_1, \dots, \Lambda_m \vdash \Omega$), якщо формула Ω набуває значення істинності 1 при кожному такому розподілі значень істинності елементарних висловлень $A_1, \dots, A_n$, при якому всі формули $\Lambda_1, \dots, \Lambda_m$ набувають значення істинності 1. Формули $\Lambda_1, \dots, \Lambda_m$ при цьому називаються посилками або гіпотезами, а формула Ω – висновком.

Теорема. Формула $\Omega(A_1, \dots, A_n)$ логічно слідує з посилок $\Lambda_1(A_1, \dots, A_n), \dots, \Lambda_m(A_1, \dots, A_n)$ тоді і тільки тоді, коли формула $(\Lambda_1 \wedge \dots \wedge \Lambda_m) \to \Omega$ є тавтологією.

Доведення:

Необхідність. Нехай нам дано, що $\Lambda_1, \dots, \Lambda_m \vdash \Omega$. Припустимо, що формула $(\Lambda_1 \wedge \dots \wedge \Lambda_m) \to \Omega$ не є тавтологією. Це означає, що при деякому розподілі значень істинності елементарних висловлень $A_1, \dots, A_n$ $|(\Lambda_1 \wedge \dots \wedge \Lambda_m) \to \Omega| = 0$. З означення операції логічного слідування отримаємо, що це можливо лише у випадку, коли $|\Omega| = 0$ і $|\Lambda_1 \wedge \dots \wedge \Lambda_m| = 1$. На основі означення операції кон'юнкції звідси отримуємо, що всі Λ_i приймають істинне значення. Так як формула Ω логічно слідує з $\Lambda_1, \dots, \Lambda_m$, то Ω теж повинна приймати істинне значення. Отримана суперечність показує, що формула $(\Lambda_1 \wedge \dots \wedge \Lambda_m) \to \Omega$ в цьому випадку повинна бути тавтологією.

Необхідність доведена.

Достатність. Нехай нам дано, що формула $(\Lambda_1 \wedge \dots \wedge \Lambda_m) \to \Omega$ є тавтологією. Доведемо, що $|\Lambda_1 \wedge \dots \wedge \Lambda_m| = \Omega$. Розглянемо довільний розподіл значень істинності елементарних висловлень $A_1, \dots, A_n$, при якому всі Λ_i

набувають істинних значень. Тоді $|\Lambda_1 \wedge \dots .\wedge\ \Lambda_m| = 1$ на розглядуваному наборі значень $A_1, \dots, A_n$. Значення Ω на цьому наборі не може бути 0, бо тоді за означенням імплікації ми отримали б $|(\Lambda_1 \wedge \dots .\wedge\ \Lambda_m) \to \Omega| = 0$, що суперечить умові. Отже, на довільному наборі, де всі посилки Λ_i набувають значення істинності 1, Ω теж приймає значення 1. Це означає, що формула, Ω логічно слідує з посилок.

Теорему доведено.

Принципове значення даної теореми полягає в тому, що вона зводить питання про відношення логічного слідування між формулами до аналізу певної формули алгебри висловлень. Застосуємо розглянуті нами поняття до аналізу змістовних міркувань, що сформульовані в звичайній мові, можливо, збагаченій деякими символами. Для цього потрібно перекласти формулювання природньої мови на символічну мову логіки. Тим самим відбувається абстрагування від конкретного змісту висловлень і зберігається тільки логічна структура міркування.

Логічні задачі

Важко визначити, яку задачу слід назвати логічною. По суті кожна задача є такою, бо для її розв'язання потрібні ті або інші логічні міркування. Але все ж ми схильні одні задачі називати арифметичними, поскільки в них маємо справу з числовим матеріалом, інші – геометричними або алгебраїчними в залежності від того, чи йде в них мова про геометричні фігури або алгебраїчні вирази. Але є задачі, в яких ми не знаходимо ні геометричних фігур, ні чисел. У них йде мова про висловлення, що стосуються об'єктів довільної природи. Ці задачі називаються *логічними задачами*. Алгебра висловлень є одним з ефективних засобів розв'язування таких задач.

Контрольні запитання

1. Дати означення логічного слідування на базі алгебри висловлень.

2. Сформулювати критерій логічного слідування. Як застосовується поняття логічного слідування до аналізу правильності логічних міркувань?

3. Яка множина висловлень називається несуперечливою?

4. Назвіть методи доведення несуперечності множини висловлень.

Приклади розв'язування типових задач

Приклад 1. Перевірити чи є логічно правильними міркування: Коли в мене не працює холодильник, я викликаю інженера. Коли я викликаю інженера, до мене приходить його помічник. Помічник інженера до мене не прийшов. Отже, холодильник працює.

Розв'язок:

Множина Г у прикладі містить три речення, що складаються з таких атомів: А – холодильник не працює; В – я викликаю інженера; С – до мене приходить помічник інженера.

Запишемо це міркування мовою логіки висловлень:

$$\Gamma = \{A \rightarrow B, B \rightarrow C, \neg C\} \vdash \neg A$$

Припустимо, що існує інтерпретація *I* така, що всі речення з множини Г в ній істинні, а речення $\neg A$ – хибне. У такий спосіб ми або знайдемо таку інтерпретацію (тобто покажемо, що міркування не є логічним), або доведемо, що її не існує.

$(I, A \rightarrow B) = 1;$ (1)

$(I, B \rightarrow C) = 1;$ (2)

$(I, \neg C) = 1 \Rightarrow (I, C) = 0;$ (3)

$(I, \neg A) = 0 \Rightarrow (I, A) = 1.$ (4)

Із рівностей (1) та (4) випливає, що $(I, B) = 1$.

Але тоді не можуть бути справедливими одночасно рівність (2) (яка перетворюється на $(I, 1 \rightarrow C) = 1$) та рівність (3).

Таким чином, ми довели, інтерпретації з шуканими властивостями не існує, отже, міркування є істинним.

Приклад 2. Встановити чи коректне міркування:

1) Якщо в розкладі уроків на сьогодні є алгебра (A), то немає англійської мови ($\neg B$).

2) Урок історії є тільки тоді (C), коли є урок англійської мови (B).

3) Немає уроку географії ($\neg D$), якщо немає уроку історії ($\neg C$).

4) Урок алгебри є в розкладі на сьогодні (A).

Отже, в розкладі на сьогодні немає уроку географії ($\neg D$).

Розв'язок:

Перейдемо до символічних позначень. Посилки запишуться у вигляді:

2. $A \to \neg B$, 2) $C \to B$, 3) $\neg C \to \neg D$, 4) A, а висновок – у вигляді ($\neg D$).

Якщо наше міркування коректне, то формула

$$((A \to \neg B) \land (C \to B) \land (\neg C \to \neg D) \land A) \to \neg D \qquad (1)$$

Повинна бути тавтологією. Хибне значення формула може приймати лише в одному випадку: $|A \to \neg B|$=1, $|C \to B|$=1, $|\neg C \to \neg D|$=1, $|A|$=1, $|\neg D|$=0, тобто $|D|$=1.

При цьому $|\neg C \to \neg D|$=1 лише в одному випадку $|\neg C|$=0, тобто $|C|$=1. Тоді з $|C \to B|$=1 отримуємо $|B|$=1.

Але, якщо $|A|$=1, $|B|$=1, то $A \to \neg B$=0.

Отримана суперечність показує, що формула (1) є тавтологією, тобто наше міркування коректне.

Побудуємо формальне виведення висновку $\neg D$ з посилок 1)-4):

1) $A \to \neg B$, 2) $C \to B$, 3) $\neg C \to \neg D$, 4) A – посилки

5) $\neg B$ (MP (1, 4))

6) $\neg B \to \neg C$ (ПК (2))

7) $\neg C$ (MP (5,6))

8) $\neg D$ (MP (3,7)).

Приклад 3. Ігор, Андрій, Костя і Свшко провели між собою шаховий турнір, в якому кожний з них набрав різну кількість очок. Щодо розподілу місць між ними є такі дані:

1) Якщо Ігор не зайняв перше місце, то Сашко має останнє.

2) Або Костя зайняв перше місце, або Сашко третє.

3) Якщо Андрій не був на першому місці, то Костя був на другому.

Яке місце зайняв кожний з них?

Розв'язок: Введемо символічні позначення:

I_i – «Ігор зайняв i-те місце»; $А_i$ – «Андрій зайняв i-те місце»

$К_i$ – «Костя зайняв i-те місце»; С – «Сашко зайняв i-те місце», де *i=1,2,3,4*.

Тоді умова задачі записується наступним чином:

3. $\neg I_1 \to C_4$, 2) $K_1 \vee C_3$, $\neg A_1 \to K_2$.

Розв'язком задачі будуть такі набори значень істинності елементарних висловлень, при яких кожна з умов 1)-3), а значить, і їх кон'юнкція будуть істинними. Виконуючи рівносильні перетворення, зведемо кон'юнкцію цих умов до ДНФ:

$$(\neg I_1 \to C_4) \wedge (K_1 \vee C_3) \wedge (\neg A_1 \to K_2) \sim (I_1 \vee C_4) \wedge (K_1 \vee C_3) \wedge (A_1 \vee K_2) \sim$$

$$\sim ((I_1 \wedge K_1) \vee (I_1 \wedge C_3) \vee (C_4 \wedge K_1) \vee (C_4 \wedge C_3)) \wedge (A_1 \vee K_2) \sim$$

$$\sim ((I_1 \wedge C_3) \vee (C_4 \wedge K_1)) \wedge (A_1 \vee K_2)$$

$$\sim (I_1 \wedge C_3 \wedge A_1) \vee (I_1 \wedge C_3 \wedge K_2) \vee (C_4 \wedge K_1 \wedge A_1) \vee \sim$$

$$\sim \vee (C_4 \wedge K_1 \wedge K_2) \sim (I_1 \wedge C_3 \wedge K_2).$$

Висловлення $I_1 \wedge K_1$, $C_4 \wedge C_3$, $I_1 \wedge A_1$, $K_1 \wedge A_1$, $K_1 \wedge K_2$ – хибні, бо кожний з учасників набрав різну кількість очок.

Отже, кон'юнкція умов 1)-3) буде істинною лише у випадку $|I_1| = |C_3| = |K_2|=1$, тобто Ігор зайняв перше місце, Костя – друге, Сашко – третє, а Андрій – четверте.

<u>*Приклад 4.*</u> Показати, що наведена множина речень є сумісною:

$$\{A \to B \wedge C, A \vee B, B \wedge C\}.$$

Розв'язок:

Щоб показати сумісність множини речень, достатньо надати хоча б одну інтерпретацію, в якій всі ці реченя є істинними. Побудуймо таку інтерпретацію *I*, у якій

$(I, A \to B \wedge C) = 1;$ (1)

$(I, A \vee B) = 1;$ (2)

$(I, B \wedge C) = 1.$ (3)

Почнемо з речення (3), оскільки кон'юнкція накладає максимум обмежень на істинність речення. Із нього дістаємо:

$(I, B) = 1;$

$(I, C) = 1.$

Оскільки, диз'юнкція істинна, якщо хоча б один атом істинний, то речення (2) вже істинне, без додаткових обмежень на A.

Імплікація істинна, наприклад, у випадку, коли ліворуч у ній стоїть хибність, тому покладемо

$(I, A) = 0.$

Таким чином, ми побудували інтерпретацію, в якій всі речення заданої множини – істинні, а отже, довели, що множина – сумісна.

Зауважимо, що в загальному випадку така інтерпретація не єдина, але достатньо навести одну.

<u>Вправи для самостійного розв'язування</u>

1.Визначити чи коректне міркування

1) Василь пішов у кіно, якщо він виконав домашнє завдання. Василь пішов у кіно. Отже, Василь виконав домашнє завдання.

2) Якщо «Динамо» гратиме в основному складі і черговий матч не буде перенесено, то «Динамо» виграє черговий матч Якщо «Динамо» виграє черговий матч, то буде чемпіоном країни цього року. Якщо на цьому тижні міжнародного матчу не буде, то черговий матч чемпіонату країни не буде перенесено. Міжнародної зустрічі не буде і «Динамо» гратиме в основному складі. Отже, «Динамо» буде чемпіоном країни в цьому році.

3) Якщо капіталовкладення залишаться сталими, то зростуть урядові витрати, або виникне безробіття. Якщо урядові витрати не зростуть, то податки

будуть знижені. Якщо податки будуть знижені і капіталовкладення залишаться сталими, то безробіття не виникне. Отже, урядові витрати зростуть.

4) Якщо йде дощ, то або ми нікуди не підемо, або ми підемо в кіно. Якщо ми підемо в кіно, то в кіно є квитки. Квитків у кіно нема. Отже, якщо піде дощ, то ми нікуди не підемо.

5) Якщо я поїду автобусом, а автобус запізниться, то я пропущу важливу зустріч. Якщо я пропущу важливу зустріч, то це мене засмутить. Якщо я не отримаю цю роботу, то це мене засмутить і мені варто буде шукати іншу. Отже, якщо я поїду автобусом і автобус не запізниться, то я отримаю цю роботу.

6) Або Аня та Ваня одного віку, або Аня старша за Ваню. Якщо Аня та Ваня одного віку, то Оксана та Ваня не одного віку. Якщо Аня старша за Ваню, то Ваня старший за Віктора. Отже, або Оксана та Ваня не одного віку, або Ваня старший за Віктора.

4. Перевірити чи є несуперечливою множина висловлень:

1) Ліда складає на «добре» екзамен з алгебри тоді і тільки тоді, коли вона не пропустить останньої лекції. Ліда або пропустить останню лекцію, або не поїде на екскурсію до Києва. Якщо Ліда складе на «добре» екзамен з алгебри, то вона не поїде на екскурсію до Києва. Ліда складе на «добре» екзамен з алгебри або поїде на екскурсію до Києва.

2) Якщо $5 \in M_1$, то неправильно, що $5 \in M_2$.Неправильно, що $5 \in M_2$ тільки тоді, коли $3 \in M_3$ $5 \in M_2$або $3 \in M_3$. $3 \in M_3$ або $5 \in M_1$

3) Якщо курс цінних паперів росте чи відсоткова ставка знижується, то або падає курс акцій, або податки не підвищуються. Курс акцій знижується тоді й тільки тоді, коли росте курс цінних паперів і податки підвищуються. Якщо відсоткова ставка знижується, то або курс акцій не знижується, або курс цінних паперів не росте. Або податки підвищуються, або курс акцій падає і знижується відсоткова ставка.

4) Договір буде виконано тільки в тому випадку, якщо будинок буде добудований у жовтні. Якщо ми добудуємо будинок в жовтні, то ми зможемо

переїхати 1 листопада. Якщо ми не зможемо переїхати 1 листопада, то ми повинні будемо внести квартплату за листопад. Якщо договір не буде виконано, то ми повинні будемо внести квартплату за листопад. Ми не платитимемо квартплату за листопад.

3. Троє обвинувачених *X, Y, Z* дають такі свідчення *X* – «*Y* винен, а *Z* – ні»; *Y* – «Якщо *X* винен, то і *Z* теж»; а *Z* – «Я не винен, але хоч один з двох інших – винен».

1) Вважаючи, що *Y* і *Z* говорять правду, встановити, хто саме винен.

2) Якщо всі троє невинні, то хто з них сказав правду, а хто – неправду?

4. Задано 5 тверджень; треба визначити, яке з них є істинним, а яке хибним. Відомо, що:

1) серед цих 5 тверджень істинних є більше ніж хибних;

2) в списку цих 5 тверджень підряд слідують не більш ніж два твердження, які потребують однакової відповіді;

3) відповіді на перше і на п'яте питання — протилежні.

Якою має бути відповідь на друге питання, щоб правильні відповіді на всі поставлені питання визначались однозначно?

5. При дослідженні хвороби добуто таку інформацію:

1) симптом a спільно з симптомом b зустрічається тільки тоді, коли є симптом c;

2) наявність разом симптомів b і d тягне за собою хоч один із симптомів a, c;

3) якщо симптом b буває без a, то є також симптом c чи d;

4) за наявності симптому b без c випливає відсутність симптому a.

Спростити максимально цю інформацію.

АЛГЕБРА ПРЕДИКАТІВ

Теоретичні відомості

Поняття предикату узагальнює поняття висловлення. Логіка предикатів дає змогу формулювати співвідношення між елементами реального світу і виводити ці відношення у математиці.

У логіці предикатів прості речення підлягають подальшому аналізу. В них виділяють терми і предикати, які визначають властивості цих термів.

Означення. Терм – ім'я суб'єкту, про який говориться у реченні.

Наприклад, х – парне число: х – терм, парне число – предикат.

Про такі речення не можна сказати: істинне воно чи хибне. Проте, якщо в них замінити змінні на числа, то вони перетворяться на звичайні висловлення.

Означення. N-містним предикатом, визначеним на множині $M_1, M_2, ..., M_n$, називається речення, яке містить n змінних $x_1, x_2, ..., x_n$ і перетворюється у висловлення при підстановці х і М деякі елементи з даних множин.

Висловлення вважається нульовим предикатом.

Означення. $x_1, x_2, ..., x_n$ називаються предметними змінними. Елементи множини $M_1, M_2, ..., M_n$ називаються предметними сталими, а саме предикат називається логічною функцією від n змінних, яка визначена на множині $M_1, M_2, ..., M_n$ і приймає значення 0 і 1.

Позначаються предикати великими літерами з індексами або без, наприклад $P(x_1, x_2, ..., x_n)$. Довільне висловлення є 0-місним предикатом. Предикат можна вважати функцією п змінних, областю визначення якої є множина М, а множиною значень – логічні значення 1 (істина) та 0 (хиба).

Вираз, яким записується предикат – а циклювання форма. Нехай Р (х, у)= «х + у= 4» - двомісний предикат, визначений на множині N ЧN, тоді логічні значення відповідних висловлень записують, наприклад, як $|P(x, y)| = 1, |P(1,1)| = 0$ і т.д. При конструюванні предикатів часто використовують функціональні символи. Тут таким символом є «+» або «сума (x, y)», а P – предикатний символ «дорівнює 4».

Предикат $P(x_1, x_2, ..., x_n)$, заданий на множині M, називається тотожно істинним, якщо для будь-якого набору предметних констант $(a_1, a_2, ..., a_n)$ $\in$ M він перетворюється в істинне висловлення, тобто $|P(a_1, a_2, ..., a_n)| = 1$. Аналогічно формулюються означення тотожно хибного, виконуваного та спростовного предиката.

Оскільки значеннями предикатів є висловлення, то над предикатами можна виконувати ті ж логічні операції, що і над висловленнями: заперечення, кон'юнкцію, диз'юнкцію, імплікацію та еквіваленцію.

Запереченням предиката $P(x_1, x_2, ..., x_n)$, заданого на множині M, називається предикат $\overline{P(x_1, x_2, ..., x_n)}$, заданий на тій же множині, який перетворюється в хибне висловлення для будь-якого набору з множини істинності предиката $\overline{P(x_1, x_2, ..., x_n)}$ і в істинне для всіх інших наборів.

Якщо M_p - множина істинності (сукупність всіх наборів $(a_1, a_2, ..., a_n)$ $\in$ M, для кожного з яких $|P(a_1, a_2, ..., a_n)| = 1$) предиката P, то множина істинності предиката $\overline{P}$ буде $M_p = M \setminus M_p$.

Нехай деякий m-місний предикат $Q(y_1, y_2, ..., y_m)$ заданий на множині $L = L_1 \times L_2 \times ... \times L_m$, причому всі змінні x_i та y_i різні.

Кон'юнкцією двох предикатів P та Q називається (п+т) – місний предикат $P(x_1, x_2, ..., x_n) \wedge Q(y_1, y_2, ..., y_n)$, заданий на множині MЧL, 36а циклювання36ання36 в істинне висловлення для всіх тих і тільки тих значень змінних, при яких перетворюються в істинне висловлення обидва задані предикати.

Якщо предикати P та Q мають k спільних змінних, то місність кон'юнкції буде s = п + т – k, а загалом max $\{n, m\} \le s \le n + m$.

Означення диз'юнкції, імплікації та еквіваленції аналогічне. [2, 36а . 17]

Якщо M_P і M_Q - множини істинності предикатів P та Q, визначених на одній множині M, то множини істинності предикатів $P \wedge Q$, $P \vee Q$, $P \to Q$ та $P \leftrightarrow Q$ можна записати у вигляді:

$M_{P\wedge Q} = M_p \cap M_Q$; $M_{P\vee Q} = M_P \cup M_Q$; $M_{P\to Q} = M \setminus (M_P \setminus M_Q)$;

$M_{P\leftrightarrow Q} = (M_P \cap M_Q) \cup (M \setminus M_P \setminus M_Q))$;

Крім вказаних операцій над предикатами виконують кванторні операції (квантифікацію).

Зв «язуванням квантором загальності одномісного предиката P(x) називається операція, яка предикату P(x) ставить у відповідність висловлення $\forall x P(x)$ («для будь-якого x має місце P(x)»), яке істинне тоді і тільки тоді, коли предикат тотожно істинний. Отже,

$$|\forall x P(x)| = \begin{cases} 1, \text{якщо} \mathrm{P}(x) \text{тотожноістиннийпредикат;} \\ 0, \text{якщо} \mathrm{P}(x) \text{спростовнийпредикат} \end{cases}$$

Наприклад, висловлення $\forall x(x^2 - x + 1 > 0)$ на множині дійсних чисел істинне, а $\forall x(x^2 - 5x - 6 > 0)$ - хибне.

Зв'язуванням квантором існування одномісного предиката P(x) називається операція, яка предикату P(x) ставить у відповідність висловлення $\exists x P(x)$ («існує x, що має місце P(x)»), яке хибне тоді і тільки тоді, коли предикат тотожно хибний. Отже, $|\exists x P(x)|$ = P(x) – тотожно хибний предикат.

Наприклад, висловлення $\exists x(x^2 + 1 = 0)$ на множині дійсних чисел хибне, а $\exists x(x + 1 = 0)$ - істинне.

При формулюванні тверджень мовою предикатів часто зустрічаються речення чотирьох типів, які в арістотелевій логіці називаються категоричними судженнями і мають зміст та символічний запис:

A: а циклювання судження «всі S суть P» (всі елементи x, які мають властивість S, мають і властивість P) - $\forall x(S(x) \to P(x))$;

E: загальнозаперечувальне судження «будь-яке S не є P» (будь-який елемент x, який має властивість S, не має властивості P) - $\forall x(S(x) \to \overline{P(x)})$;

I: частково стверджувальне судження «деякі S суть P» (деякі елементи x, які мають властивість S, мають і властивість P) - $\exists x(S(x) \wedge P(x))$;

O: частково а циклювання, а судження «деякі S не є P» (деякі елементи x, які мають властивість S, не мають властивості P) - $\exists x(S(x) \wedge \overline{P(x)})$;

Комбінуючи речення A-O, можна записувати у символічній формі досить складні твердження.

Зауваження. Якщо предикат P(x) заданий на скінченній множині елементів $M = \{e_1, e_2, ..., e_k\}$, то операція зв'язування квантором загальності $\forall x P(x)$ (або існування) рівносильна кон'юнкції $P(c_1) \wedge P(c_2) \wedge ... \wedge P(c_k)$ (або відповідній диз'юнкції).

Зв'язування квантором загальності чи існування за деякою змінною x, n-місного предиката $P(x_1, x_2, ..., x_n)$ приводить до (n-1) – місного предиката $\forall x_i P(x_1, x_2, ..., x_n)$ чи $\exists x_i P(x_1, x_2, ..., x_n)$, який залежить від змінних $x_1, x_2, ..., x_{i-1}, x_{i+1}, ..., x_n$.При цьому висловлення $\forall x, P(a_1, a_2, ..., a_{i-1}, x_i, a_{i+1}, ...a_n)$ істинне тоді і тільки тоді, коли предикат $P(a_1, a_2, ..., a_{i-1}, x_i, a_{i+1}, ...a_n)$ тотожно істинний на множині M_i, а висловлення $\exists x_i P(a_1, a_2, ..., a_{i-1}, x_i, a_{i+1}, ...a_n)$ хибне тоді і тільки тоді, коли предикат $P(a_1, a_2, ..., a_{i-1}, x_i, a_{i+1}, ...a_n)$ тотожно хибний на M_i.

Далі можна по черзі зв'язувати різними кванторами інші змінні. Коли всі змінні будуть зв'язані, отримується висловлення. Наприклад, тримісний предикат $P(x, y, z) = "x^2 + y^2 \geq z^2"$, заданий на множині дійсних чисел, можна перетворити на двомісний $\exists x(x^2 + y^2 \geq z^2$, одномісний $\forall y \exists x(x^2 + y^2 \geq z^2)$, предикати або ж на істинне висловлення $\forall x \forall y \exists x(x^2 + y^2 \geq z^2)$, Якщо P – 0-місниЙ предикат (висловлення), то записи $\forall x P$ та $\exists x P$ означають те саме, що і P. [2, а 23]

Входження змінної в предикат називається зв'язаним, якщо вона є змінною квантора або знаходиться в області дії квантора за цією змінною. Інакше входження вільне. У складних предикатах область дії квантора виділяється дужками.

Предикати $P(x_1,....x_n)$ та $Q(x_1,...x_n)$, задані на одній множині М, називаються рівносильними (P Q), якщо $M_P = M_Q$ (один з них перетворюється в істинне висловлення на всіх тих наборах $(a_1,a_2,...a_n)\in M$, на яких і інший перетворюється в істинне висловлення). Предикат $Q(x_1,...x_n)$ називається логічним наслідком предиката $P(x_1,...x_n)$ (P Q), якщо $M_P \subseteq M_Q$.

Класифікація предикатів

Означення. Предикат $P^{(n)}$ визначений на множинні $M_1,M_2,...,M_n$ називається:

1. 39а цикл-істинним, якщо для будь-яких $a_i \in M_i$ $i=\overline{1,n}$ $P^{(n)}(a_1,...,a_n)\equiv 1$.
2. 39а цикл-хибним, якщо для будь-яких $a_i \in M_i$ $i=\overline{1,n}$ $P^{(n)}(a_1,...,a_n)\equiv 0$.
3. виконуваним, якщо існують $a_i \in M_i$ $i=\overline{1,n}$ $P^{(n)}(a_1,...,a_n)=1$.
4. спростовним, якщо існують $a_i \in M_i$ $i=\overline{1,n}$ $P^{(n)}(a_1,...,a_n)=0$.

Означення. Множиною істинності предикатів $P^{(n)}$, визначеного на множині $M_1,M_2,...,M_n$ називають сукупність всіх кортежів виду $(a_1,...,a_n)$, $a_i \in M_i$, де $P^{(n)}(a_1,...,a_n)=1$. Позначимо множину істинності через P^+.

Означення. Під інтерпретацією логіки предикатів розуміють систему $M_f=\{M,f\}$, що 39а циклюванн з кожним предикатом-символом певний n-місний предикат $P^{(n)}$.

Додатково в 39а цик 39а циклюва використовують 39а спеціальні 39а циклю, які називають *квантарами.*

Найпопулярнішими і найбільш часто вживаними виразами у математиці є фрази або формулювання типу «для 39а ц» і «існує». Поняття, що відповідає словам «для 39а ц», лежить в основі квантора 39а циклюван.

Нехай *P*(*x*) – предикат на множині *M*. Тоді *квантор 39а циклюван* – це 39а циклю, що ставить у 39а циклювання *P*(*x*) висловлення «для 39а ц *x* з *M* *P*(*x*) 39а цикл». Для позначення 39а ц 39а циклю використовують знак ∀, який і називають квантором а циклюван. Останнє висловлення у математичній а цик записують так: $\forall Xp(x)$ (читається: «для ц x P а x»).

Інший квантор, що є у певному смислі двоїстим до квантора а циклюван і називається <u>*квантором існування.*</u> Позначається знаком $\exists$. Якщо $Q(x)$ – деякий предикат на множині M, то висловлення «існує в множині M елемент x такий, що $Q(x)$ а цикл» записується у вигляді $\exists Xq(x)$ і читається скорочено «існує такий x, що Q а x» або «є такий x, що Q а x».

Наприклад. Розглянемо два бінарні предикати на множині натуральних чисел: предикат «x менше y» і предикат «x ділить y». Перший з них будемо записувати у традиційній формі – $x<y$, а другий у вигляді $x|\,y$. Тоді неважко переконатись, що висловлення $\forall x\exists y(x<y)$ і $\forall x\exists y(x|\,y)$ є істинними, а висловлення $\exists y\forall x(x<y)$ і - $\exists y\forall x(x|\,y)$ хибні. Істинними будуть, наприклад, висловлення $\forall x(0<x^2\text{-}x+1)$, $\exists x((x|1)\wedge(\neg(1<x)))$, $\forall x((x<1)\rightarrow(x<2))$, $\forall x(((2|\,x)\wedge(3|\,x))\rightarrow(6|\,x))$, а висловлення $\forall x(0<x^2\text{-}x+1)$, $\exists x((x|1)\wedge(\neg(1<x)))$, $\forall x((x<1)\rightarrow(x<2))$, $\forall x(((2|\,x)\wedge(3|\,x))\rightarrow(6|\,x))$ – хибні.

Важливу роль у 40а цик 40а циклюва 40а циклю поняття <u>*області 40а*</u> квантора, а якою розумітимемо той вираз, до якого відноситься цей квантор. Область 40а квантора позначається за допомогою дужок. Ліва дужка, що відповідає початку області 40а , записується безпосередньо 40а ци кванторної змінної даного квантора, а відповідна їй права дужка означає 40а циклюва області 40а цього квантора. Там, де це не викликає 40а циклювання40, дужки можна опускати і 40а цикл $\forall x(P(x))$ або $\exists x(P(x))$ писати 40а циклюва $\forall Xp(x)$ або $\exists Xp(x)$.

Наприклад. В 40а ц нижченаведених кванторних виразах область 40а квантора підкреслено.

$\exists x\underline{((3|x)\rightarrow(6|x))}$, $\exists x\underline{(3|x)}\rightarrow(6|x)$, $\forall x\underline{((x^2<9)\rightarrow(x<3))}$, $\forall x\underline{(x^2<9)}\rightarrow(x<3)$.

Перший і другий вирази, а також третій і четвертий відрізняються не лише областю дії квантора. Відмінність між ними є більш істотною, про що слід сказати окремо.

Розглянемо на універсальній множині R 41а ц 41а цикл чисел два вирази $x^2<10$ і $\exists x(x^2<10)$. Перший з них є предикатом, що залежить 41а змінної x. Замість x у нього можна підставляти різні дійсні значення і отримувати 41а ци висловлення (істинні чи хибні). Та сама предметна змінна x входить у другий вираз 41а цик. Якщо 41а цикл неї підставити будь-яке 41а цик значення, 41а циклюв беззмістовний вираз.

Нехай $P(x)$ – деякий предикат на M. Перехід 41а $P(x)$ до $\forall Xp(x)$ або $\exists Xp(x)$ називається *<u>зв'язуванням</u>* змінної x. Змінна x, на яку 41а циклю квантор, називається *<u>зв'язаною</u>*, у противному разі змінна x називається *<u>вільною.</u>*

Зв'язана змінна, як було продемонстровано вище, вже не є змінною у класичному розумінні цього поняття. Вона 41а циклю лише для 41а циклювання терма, 41а якого залежить відповідна пропозиційна форма. Вираз $\forall Xp(x)$ не залежить 41а x і при 41а циклюва P і M має певне значення. Звідси, зокрема, випливає, що можна здійснювати перейменування зв'язаної змінної (тобто перехід 41а $\forall Xp(x)$ до $\forall Tp(t)$) і воно не змінює значення 41а циклюва виразу.

<u>Контрольні запитання</u>

1. Що таке предикат?
2. Множина істинності предикату та її зображення за допомогою діаграм Ейлера-Венна.
3. Логічні операції над предикатами та їх теоретико-множинний зміст.
4. Квантори. Операції навішування кванторів.

<u>Приклади розв'язування типових задач</u>

<u>Приклад 1.</u> Нехай $D(x,y)$ – предикат ділення: $D(x,y)=$"x ділиться на y"; $S(x,y,z)$ – предикат суми: $S(x,y,z)=$"x – сума y і z"; $\Pi(x,y,z)$ – предикат добутку: $\Pi(x,y,z)=$"x – добуток y і z". Дати словесні формулювання наступних складних висловлень:

а) $П(a,b,c) \to (D(a,b) \wedge D(a,c))$;

б) $П(a,b,c) \leftrightarrow П(a,c,b)$;

в) $\bar{S}(a,b,c) \wedge \overline{П}(a,b,c)$.

Розв'язок:

а) "якщо a - добуток b і c, то a ділиться на b і a ділиться на c";

б) "від перестановки співмножників b і c добуток a не змінюється" (властивість комутативності арифметичної дії множення);

в) "число a не є ні сумою, ні добутком чисел b і c".

Приклад 2. Нехай змінна x визначена на великій кількості людей M, а $P_1(x)$ - предикат «*x* – має маму», $P_2(x)$ - предикат «*x* – має дочку». Дати словесне формулювання предикатних формул $\forall x P_1(x)$, $\exists x P_2(x)$.

Розв'язок:

$\forall x P_1(x)$ - "у кожної людини є мама"; $\exists x P_2(x)$ - "у деяких людей є дочка".

Приклад 3. Нехай $P(x,y)$ - предикат "x - валюта (грошова одиниця) країни y", де x належить множині грошових одиниць, а y належить множині країн. Розглянути всі варіанти навішування кванторів на обидві змінні, дати словесну інтерпретацію отриманих висловлень і визначити їхню істинність.

Розв'язок:

$\forall x P(x,y)$ - однозмінний предикат (змінне висловлення) "будь-яка грошова одиниця є валютою країни y". Дане змінне висловлення хибне.

$\forall y P(x,y)$- однозмінний предикат " будь-якої країни y грошова одиниця x є валютою ". Дане змінне висловлення хибне.

$\exists x P(x,y)$- однозмінний предикат "існує така грошова одиниця x, що є валютою країни y". Дане змінне висловлення істинно для будь-якого значення вільної змінної y.

$\exists y P(x,y)$- однозмінний предикат "існує така країна y, валютою якої є грошова одиниця x". Дане змінне висловлення істинно для будь-якого значення вільної змінної x.

Далі йдуть предикати нульової розмірності: $\forall x \forall y P(x,y)$, $\forall y \forall x P(x,y)$ - "валютою будь-якої країни є будь-яка грошова одиниця". Дане висловлення є хибним.

$\exists x \forall y P(x,y)$ - висловлення "існує така грошова одиниця x, що є валютою будь-якої країни". Дане висловлення є хибним.

$\forall y \exists x P(x,y)$ - висловлення "у будь-якої країни існує своя грошова одиниця". Дане висловлення істинно.

$\forall x \exists y P(x,y)$ - висловлення "для будь-якої грошової одиниці існує така країна, де вона є її валюта". Дане висловлення істинно.

$\exists y \forall x P(x,y)$ - висловлення "існує така країна, валютою якої є всі існуючі грошові одиниці". Дане висловлення хибне.

$\exists x \exists y P(x,y)$, $\exists y \exists x P(x,y)$ - висловлення "існує така грошова одиниця x і існує така країна y, що x є валютою країни y". Дані висловлення істинні.

Приклад 4. Зобразити таблицею одномісний предикат «Р є простим числом» на множині натуральних чисел $\{1,2,\ldots,20\}$.

Розв'язок:

Зобразимо розв'язок у вигляді таблиці, де $Q(P)=1$, якщо P – просте число і $Q(P)=0$ в протилежному випадку.

P	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$Q(P)$	0	1	1	0	1	0	1	0	0	0	1	0	1	0	0	0	1	0	1	0

Приклад 5. Нехай $P(x,y)$ заданий на множині $M=\{a,b,c\}$ таблицею.

x	a	A	A	B	B	b	C	c	c
y	a	B	C	A	B	c	A	b	c
$P(x,y)$	1	0	0	1	0	0	1	0	0

Визначити істинність наступних формул: $\forall xP(x,a)$, $\exists yP(b,y)$, $\exists x\forall yP(x,y)$.

Розв'язок:

Формула $\forall xP(x,a)$ - істинна, тому що для кожного x із множини $M=\{a,b,c\}$ прямує наступне:

$P(a,a)=P(b,a)=P(c,a)=1$.

Формула $\exists yP(b,y)$ - істинна, тому що існує такий набір змінних, при якому змінне висловлення $P(b,a)=1$.

Формула $\exists x\forall yP(x,y)$ - хибна, тому що не існує такого значення x, щоб для всіх y прямувало $P(x,y)=1$.

<u>*Приклад 6.*</u> Визначити, чи є дані висловлення формулами логіки предикатів. Якщо так, то визначити вільні і зв'язані змінні:

а) $\forall x\exists y\exists zP(x,y,z,u)$;

б) $\forall x\forall yP_1(x,y,z)\leftrightarrow\exists xP_2(x,y,z)$;

г) $\exists x\forall z(P_1(x,y,z)\rightarrow(P_2(x,z,y)\vee P_1(y,z,x)))$.

Розв'язок:

а) дане висловлення є формулою логіки предикатів. Змінні x,y,z - зв'язані, а змінна u - вільна.;

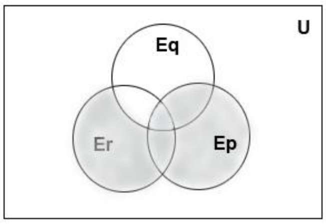
б) дане висловлення не є формулою логіки предикатів, тому що змінна y в першій частині формули $\forall x\forall yP_1(x,y,z)$ є зв'язаною, а в другій – $\exists xP_2(x,y,z)$ є вільною;

в) дане висловлення є формулою логіки предикатів, тому що змінні x,z зв'язані, а змінна y - вільна.

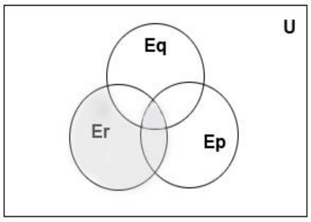
<u>*Приклад 7.*</u> Виразити множину істинності предиката через множини істинності відповідних елементарних предикатів. Зобразити розв'язок на кругах Ейлера $(P(x)\wedge\neg Q(x))\wedge R(x)\vee\neg R(x)\wedge\neg P(x)$

Розв'язок:

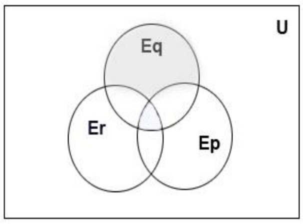
Позначимо відповідні множини істинності через E_p, E_Q, E_R. Тоді потрібно зобразити на діаграмі Ейлера множину $(E_p \vee \overline{E}_Q) \wedge E_R \vee \overline{E}_R \wedge \overline{E}_P$, яка і виражає множину істинності складного предиката через множини істинності елементарних предикатів.



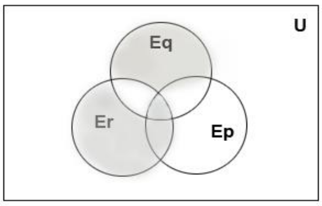
$E_p \vee \overline{E}_Q$



$(E_p \vee \overline{E}_Q) \wedge E_R$



$\overline{E}_R \wedge \overline{E}_P$



$(E_p \vee \overline{E}_Q) \wedge E_R \vee \overline{E}_R \wedge \overline{E}_P$

<u>*Приклад 8.*</u> Виразити множину істинності предиката $(P(x) \to Q(x)) \wedge (P(x) \to \neg Q(x))$ через множину E_p.

Розв'язок:

Виходячи з умови задачі, потрібно перетворити дану формулу алгебри предикатів так щоб її значення істинності не залежало від значення істинності предиката $Q(x)$

$$(P(x) \to Q(x)) \wedge (P(x) \to \neg Q(x)) \equiv (\neg P(x) \vee Q(x)) \wedge (\neg P(x) \vee \neg Q(x)) \equiv \neg P(x) \vee \neg P(x) \wedge Q(x) \vee \neg P(x) \wedge \neg Q(x) \vee Q(x) \wedge \neg Q(x) \equiv \neg P(x)$$

Отже, множиною істинності предиката буде множина $\overline{E}_P$.

<u>*Приклад 9.*</u> Чи є y вільною для x в формулі логіки предикатів:

1) $U(x) = \exists y F(x, y)$;

2) $U(x) = \forall y (F(x, y)) \vee P(x)$;

3) $U(x) = P(x) \to \forall y Q(y)$

Розв'язок:

В перших двох випадках y не являється вільною буквою для x, оскільки при підстановці замість на x суттєво змінюється зміст формул: при заміні вільні змінні стають зв'язними. В третьому випадку формула $P(x) \rightarrow \forall x Q(x) \equiv P(x) \rightarrow \forall y Q(y)$, оскільки отримана перейменуванням зв'язної змінної при якому жодна вільна змінна не стає зв'язною.

<u>*Приклад 10.*</u> Які з наведених речень є предикатами (для предикатів вказати місність):

a) «х паралельна прямій l (х – довільна пряма на площині);

b) «х є притокою у» (х, у – назви всіх можливих рік);

c) «деякі парні числа діляться на натуральне число у»;

d) «всі непарні числа діляться на 2»;

e) «х або Україна і Росія» (х – назва країни);

f) $"x^2 + \sqrt{y} + \sin y"$ (х, у – дійсні числа);

g) $"x^2 + 5 = \sqrt{5} + \sin y"$ (х, у – дійсні числа);

h) «для довільного х0 існує п, що ху = 1».

Розв'язок:

а) l-місний предикат, який перетворюється в істинне висловлення на множині всіх прямих, які паралельні до фіксованої прямої l. В) 2 – місний предикат, який може перетворюватися або в істинне висловлення (наприклад, Десна є притокою Дніпра), або в хибне (Десна є притокою Волги). С) 1-місний предикат, який перетворюється завжди в істинне висловлення (тотожно істинний), оскільки існують парні числа виду 2у, 4у, 6у, які діляться на у. d) 0-місний предикат, який є хибним висловленням, e), f) Не предикати, оскільки при довільній підстановці замість змінних конкретних значень речення не можуть бути висловленнями. G) 2-місний предикат, який перетворюється в хибне висловлення при будь-яких дійсних х та у (тотожно хибний). Н) 0-місний предикат, який є істинним висловленням.

Приклад 11. Знайти множину істинності предикатів:

a) $P(x) = "\sin x \geq 1" x \in R$;

b) $P(x) = "x^2 + 6x - 16 \leq 0" (x \in R)$;

c) P(x) = «відрізок АВ видно з точки х під прямим кутом»;) P(x) = «точка х рівновіддалена від точокА і В»;

e) $P(x, y) = "x \geq 0 \leftrightarrow y \leq 0" (x, y, R)$.

Розв'язок: а) Множина істинності даного предиката – це множина розв'язків нерівності $\sin x \geq 1$, яка перетворюється у істинне висловлення лише тоді, коли $\sin x = \sin x =$1, отже $M_p = \left\{ \frac{\pi}{2} + 2\pi k, k \in z \right\}$. В) M_p = [-8; 2]. С) Множина істинності – множина точок кола, яке побудоване на відрізку АВ як на діаметрі, крім самих точок А та В. d) Множина істинності – всі точки серединного перпендикуляра для відрізка АВ. Е) Множина істинності – всі точки координатної площини, що знаходяться у другому (утворюється еквіваленція двох хибних висловлень) та четвертому (еквіваленція двох істинних висловлень) квадрантах та на координатних осях.

Приклад 12. Прочитати висловлення та визначити, які з них істинні, а які хибні, вважаючи, що всі змінні належать множині дійсних чисел:

а) $\forall x \exists y (x + y = 3)$;) $\exists y \forall x (x + y = 3)$;) $\forall x \exists y (x + y = 3) \rightarrow (2 = 3)$;) $\forall x ((x > 1) \vee (x < 2) \leftrightarrow (x = x))$;

Розв'язок: а) Навішування квантора загальності по х приведе до істинного висловлення, якщо одномісний предикат $A(x) = "\exists y (x + y = 3)"$ на множині дійсних чисел тотожно істинний. При довільному $x_0 \in R$ висловлення $\exists y (x + y = 3)$ істинне, оскільки предикат $B(y) = "x_0 + y = 3"$ виконуваний (він перетворюється в істинне висловлення при $y = 3 - x_0$). Отже, предикат А(х) тотожно істинний, тому висловлення $\forall x A(x) = \forall x \exists y (x + y = 3)$, яке читається так: «Для довільного дійсного х існує дійсне у, що х + у = 3», істинне. В) Висловлення «Існує дійсне у, що його сума з довільним дійсним х дорівнює 3» очевидно хибне. Дійсно, предикат $B(y) = "\forall x (x + y = 3)"$ тотожно хибний, бо для довільного $y_0 \in R$ предикат $A(x) = "x + y_0 = 3"$ спростовний (перетворюється в хибне висловлення, наприклад, при $x = 1 - y_0$). C) Висловлення «Якщо сума двох довільних дійсних чисел дорівнює 3, то 2= 3» істинне, оскільки є імплікацією двох хибних висловлень. D) Висловлення «Довільне дійсне число x дорівнює самому собі тоді і тільки тоді, коли воно або більше за 1, або менше за 2» є істинним, бо предикат $A(x) = "(x > 1) \vee (x < 2) \leftrightarrow (x = x)"$ є тотожно істинним як еквіваленція двох предикатів $"(x > 1) \vee (x < 2)"$ та» (x = x)», які перетворюються в істинне висловлення при довільних x $\in$ R.

<u>*Приклад 13.*</u> Дано предикати A(x) та B(x). Записати реченням висловленняC та B:

5. A(x) = «x-студент», B(x) - «x – склав іспити»,

$C = \exists x(A(x) \wedge \overline{B(x)}), D = \forall x(A(x) \rightarrow B(x))$;

b) A(x) = «x: - гриб», B(x)= «x – їстівний»,

$C = \exists x(A(x) \wedge \overline{B(x)}), D = \forall x(A(x) \rightarrow \overline{B(x)})$;

c) A(x)= «x – наука», B(x) - «x – гуманітарна»,

$C = \overline{\forall x(A(x) \rightarrow B(x))}, D = \exists x(A(x) \wedge B(x))$.

► **a)** Висловлення C – частково 48а циклювання48а (O): «Є студенти, які не склали іспити»; D – загальиоствердживальне (A): «Всі студенти склали іспити». **B**). Висловлення C – частково 48а циклювання48а (O): «Є неїстівні гриби»; D – загальнозаперечувальне (E): «Всі гриби неїстівні».**c**) Висловлення C – заперечення 48а циклювання48ання4848ьного: «Не всі науки гуманітарні», а це за змістом рівносильне твердженню «Існують негуманітарні науки», яке символічно записується як $\exists x A(x) \wedge \overline{B(x)}$ D – частково стверджувальне (I): «Існують гуманітарні науки».

<u>*Приклад 14.*</u> Записати речення у символічній формі, ввівши доречні у кожному випадку предикати:

a) Мухтар – собака;

b) деякі журналісти були в космосі;

c) якщо число ділиться на 12, то воно ділиться на 2,4 і 6;

d) громадяни Бельгії обов'язково володіють або німецькою, або французькою мовою;

e) деякі студенти здали всі екзамени;

f) кожен студент не здав принаймні один екзамен;

g) кожна людина знає англійську мову, або має друга, який знає англійську мову.

A) Якщо A(x) = «x – собака» - предикат, визначений на множині всіх тварин, а т – позначення клички Мухтар, то символічний записданого речення: A(т). b) Визначимо на множині всіх людей предикати A(x) = «x – журналіст» та B(x) = «x – був у космосі». Зі змісту речення зрозуміло, що «дослівний» переклад такий: існують люди, які одночасно є журналістами і були у космосі. Тому можна записати: $\exists x(A(x) \wedge B(x))$. Зауважимо, що запис $\exists x(A(x) \to B(x))$ не відповідає даному реченню, оскільки перетворюється в істинне висловлення навіть для тих x, які не є журналістами. Якби предикат B(x) був визначений на множині всіх журналістів, то речення можна було б записати коротко: $\exists x B(x)$ .c) Задамо на множині натуральних чисел предикати A(x) = «x – ділиться на 12», B(x) = «x – ділиться на 12», C(x) = «x – ділиться на 4», D(x) = «x – ділиться на 6». Оскільки у реченні йдеться про довільне число, то застосуємо квантор загальності: $\forall x(A(x) \to (B(x) \wedge C(x) \wedge D(x)))$. D) Після введення на множині 49а циклюва предикатів A(x)= «x-громадянин Бельгії», B(x)= «x – володіє німецькою», C(x)= «x – володіє французькою мовою» дане речення можна символічно записати: $\forall x(A(x) \to (B(x) \vee C(x)))$.e) Для опису відношень між різними об'єктами (студенти та екзамени) використовуються також і багатомісні предикати. Нехай A(x) =» x – студент», B(y)= «y – екзамен», C (x, y)= «x здав екзамен y». Тоді дане речення можна перефразувати «існують студенти x, що який би не був екзамену, вони його здали» і записати: $\exists x \forall y(A(x) \wedge B(y) \to C(x, y))$. F) Дане речення за змістом протилежне до попереднього, тому використаємо ті самі предикати. Потрібно записати, що «для кожного студента x існує екзамен y і він його не здав»:

$\forall x \exists y (A(x) \wedge B(y) \wedge \overline{C(x,y)})$.g) Введемо предикати A(x) = «x – знає англійську мову» та B (x, y) - «x друг y». Речення «для кожної людини x або вона знає англійську, або існує друг y, що знає англійську» символічно записується: $\forall x [A(x) \vee \exists y (B(x,y) \wedge A(y))]$.

Вправи для самостійного розв'язування

6. Розглянути всі можливі варіанти навішення кванторів на предикат $P(x,y)$ - "книга x з бібліотеки y". Дати словесні формулювання отриманих висловлень і визначити їхню істинність.

7. Здійсніть «переклад» речень природної мови на мову логіки предикатів:

1) Усі судді – юристи.
2) Деякі юристи – шахраї.
3) Жодний суддя не є шахраєм.
4) Деякі судді старі за віком, проте жваві.
5) Суддя Джонс не старий і не жвавий.
6) Усі жінки-юристи обожнюють якогось суддю.
7) Деякі юристи обожнюють жінок.
8) Деякі шахраї не обожнюють жодного юриста.
9) Суддя Джонс не обожнює жодного шахрая.
10) Існують як юристи, так і шахраї, що обожнюють суддю Джонса.
11) Кожна людина прагне знайти свою половину.
12) Деякі люди мають витончений смак.
13) Для всякого потенційного поля ротор дорівнює нулю.
14) Сума довжин двох сторін трикутника завжди менше довжини його третьої сторони.
15) Всі сторони ромба рівні.

16) Будь-яка людина має батька.

8. Записати символами логіки предикатів твердження:

1) Кожна диференційована в точці x_0 функція f є неперервною в точці x_0.

2) Всі числа кратні 48 кратні 8 і кратні 6, але не всяке число, кратне 6, є кратним 48.

3) Не існує числа, яке кратне 6 і разом з тим кратне 3.

4) Деякі вписані в коло 4-кутники є квадратами.

5) Жодна необмежена на $[a, b]$ функція не є неперервною на $[a, b]$.

6) В множині всіх натуральних чисел існує найменше число.

7) В множині всіх натуральних чисел існує найбільшого число.

9. Застосовуючи символи логіки предикатів, записати означення:

1) Об'єднання двох множин A і B.

2) Перерізу двох множин A і B.

3) Різниці двох множин A і B.

4) Симетричної різниці двох множин A і B.

5) Декартового добутку двох множин A і B.

10.Нехай $S(x, y, z)$ і $\Pi(x, y, z)$ - предикати суми і добутку відповідно, певні:

- на множині натуральних чисел з нулем N_0;

- на множині всіх цілих чисел Z.

6. З'ясувати, який зміст мають наступні формули, і на якої (з перерахованих) множині вони істинні:

а) $\exists y \forall x S(x, y, z)$;

б) $\exists y \forall x \Pi(x, y, z)$;

в) $\forall x \forall z \exists y S(x, y, z)$;

г) $\forall x \forall z \exists y \Pi(x, y, z)$.

7. Визначити, чи є дані вираження формулами логіки предикатів. Якщо так, то визначити вільні і зв'язані змінні:

а) $\forall x \exists u \exists z P(x, y, z, u, v)$;

б) $\exists y P_1(x, y, z) \wedge \forall y P_2(x, y, z)$;

в) $\exists x \forall z P_3(x, y, z) \rightarrow \exists z P_1(x, y, z)$;

г) $\forall x \forall u (P_5(x, y) \vee P_6(z, u))$;

д) $\exists y P_2(x, y, z) \leftrightarrow \forall x \forall y P_6(x, y)$;

е) $\forall x \forall y P_1(x, y, z) \vee \exists x \exists z P_1(x, y, z)$;

ж) $\exists y \forall x \forall z P(x, y, z)$.

8. Зобразити таблицею з двома входами предикат:

1) заданий рівнянням $2x + 3y = 17$, де x належать множині $M = \{1,2,3,4,5,6,7,8\}$;

2) заданий нерівністю $|x + y| \leq 2$, де x, y належать множині $M = \{-2, -7, 0, 1, 2\}$.

9. Зобразити таблицею предикат $P(x) \rightarrow Q(x)$, де через $P(x)$ позначено «$x \vdots 3$», а через $Q(x)$ «$x \vdots 6$» на множині $\{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15\}$.

10. Нехай предикат $P(x, y)$ задано на множині $M = \{a, b, c\}$ таблицею (по варіантах):

а)	x	a	a	a	b	b	b	c	c	c
	y	a	b	c	a	b	c	a	b	c
	$P(x, y)$	1	0	1	0	1	0	1	0	1
б)	x	a	a	a	b	b	b	c	c	c
	y	a	b	c	a	b	c	a	b	c
	$P(x, y)$	0	0	1	0	0	1	0	0	1
в)	x	a	a	a	b	b	b	c	c	c

	y	a	b	c	a	b	c	a	b	c
	$P(x,y)$	1	1	1	0	0	0	1	1	1
г)	x	a	a	a	b	b	b	c	c	c
	y	a	b	c	a	b	c	a	b	c
	$P(x,y)$	0	1	0	0	1	1	1	0	0

Визначити істинність наступних формул:

$\forall xP(x,a)$, $\exists xP(x,a)$, $\forall xP(x,b)$, $\exists xP(x,b)$, $\forall yP(a,y)$, $\exists yP(a,y)$, $\forall yP(b,y)$, $\exists yP(b,y)$, $\forall x\forall yP(x,y)$, $\exists x\exists yP(x,y)$, $\forall x\exists yP(x,y)$, $\exists x\forall yP(x,y)$, $\forall y\exists xP(x,y)$, $\exists y\forall xP(x,y)$.

11. Виразити множину істинності предиката через множини істинності елементарних предикатів. Розв’язок зобразити на кругах Ейлера-Венна:

1) $\neg(P(x)\vee Q(x)) \leftrightarrow \neg P(x)\vee\neg Q(x)$;
2) $(\neg P(x) \to Q(x))\vee P(x)\wedge Q(x)$;
3) $P(x)\vee Q(x)\wedge R(x) \leftrightarrow (P(x)\vee Q(x))\wedge(P(x)\vee R(x))$;
4) $\neg(P(x)\wedge Q(x)) \leftrightarrow \neg P(x)\vee\neg Q(x)$;
5) $(P(x)\vee Q(x))\wedge R(x) \leftrightarrow P(x)\wedge R(x)\vee Q(x) \wedge R(x)$;
6) $(P(x) \leftrightarrow Q(x))\vee R(x) \to (Q(x) \leftrightarrow P(x)\vee R(x)$
7) $(P(x)\vee\neg Q(x))\wedge R(x)\vee\neg R(x)\wedge\neg P(x)$
8) $(P(x)\to R(x))\wedge(Q(x)\to\neg R(x))\wedge(P(x)\to\neg R(x))$
9) $\neg(P(x)\vee Q(x))\leftrightarrow\neg P(x)\vee\neg Q(x)$

12. Зобразити таблично всі індивідуальні двомісні предикати, означені на множині *M,* яка містить рівно один елемент. Скільки буде таких відмінних між собою 53а циклюва?

13. Скільки є різних індивідуальних тримісних предикатів означених на:

1) одноелементній множині;
2) двоелементній множині.

Відповідь 53а циклюванн.

14. Чи y є вільною змінною для x у формулі логіки предикатів $a(x)$?

1) $a(x) = \exists yF(x,y)$;
2) $a(x) = \forall y(F(x,y))\vee P(x)$;

3) $a(x) = P(x) \to \forall y Q(y)$;

4) $a(x) = Q(y) \to \forall z P(x, y, z)$.

ЛОГІЧНІ ЗАГАЛЬНОЗНАЧУЩІ ФОРМУЛИ АЛГЕБРИ ПРЕДИКАТІВ. ЧИСЛЕННЯ ПРЕДИКАТІВ

Теоретичні відомості

Для формул логіки предикатів зберігаються всі рівносильності та правила перетворень логіки висловлень. Крім того, можна довести наступні правила:

1. Перенесення квантора через заперечення. Нехай A вільну змінну x. $(\overline{\forall x})A(x) \equiv (\exists x)A(x))$ $\overline{(\exists x)A(x)} \equiv (\forall x)\overline{A(x)}$.

2. Винесення квантора за дужки. Нехай A – формула, що містить вільну змінну x, а формула B не містить і обидві формули задовольняють пункт 3 побудови правильних формул, тоді рівносильними будуть такі співвідношення:

$$\overline{(\forall x)}(A(x) \wedge B) \equiv (\forall x)A(x) \wedge B; \quad (\exists x)(A(x) \vee B) \equiv (\exists x A(x)) \vee B;$$

$$(\forall x)(A(x) \vee B) \equiv (\forall x A(x)) \vee B; \quad (\exists x)(A(x) \wedge B) \equiv (\exists x A(x)) \wedge B.$$

Зауваження. Якщо формула B також містить вільну змінну x, то справедливі:

$$(\forall x)(A(x) \wedge B(x)) \equiv (\forall x A(x)) \wedge (\forall x B(x)); \ (\exists x)(A(x) \vee B(x)) \equiv (\exists x A(x)) \vee (\exists x B(x)).$$

3. Комутативність однойменних кванторів.

$$(\forall x)(\forall y)A(x, y) \equiv (\forall y)(\forall x)A(x, y); \quad (\exists x)(\exists y)A(x, y) \equiv (\exists y)(\exists x)A(x, y).$$

Перейменування зв’язних змінних. Замінюючи зв’язну змінну формули A іншою, що не входить в цю формулу, у кванторі і всюди в області його дії одержуємо формулу 54а циклювання54ання54 A

Рівносильність формул логіки предикатів

Нехай формули ф і д мають однакову множину змінних.

Означення. Формули ф і д називаються в заданій інтерпретації, якщо на будь-якому наборі вільних змінних, вони набувають однакових значень.

Означення. Формули ф і д є рівносильними на множинні М, якщо вони рівносильні у всіх інтерпретаціях, заданих на М.

Означення. Формули ф і д є рівносильними в логіці предикатів, якщо вони рівносильні на всіх множинах.

Означення. Формули, в яких застосовується лише символи $\wedge, \vee, \neg$, при чому $\neg$ зустрічається лише перед символами предикатів, називають зведеними.

Твердження. Для будь-якої формули існує рівносильна їй зведена формула, причому множини вільних і зв'язних змінних цих формул збігаються.

Означення. Зведена формула називається нормальною, якщо вона не містить символів кванторів або всі квантори записані попереду.

Твердження. Для будь-якої зведеної формули логіки предикатів існує рівносильна їй нормальна формула цієї ж довжини.

Інтерпретація формул логіки предикатів

Розглянемо деяку інтерпретацію з множиною М.

Означення. Формула А називається здійснюваною в заданій інтерпретації, якщо існує набір $(a_1,...,a_n)$, $a_i \in M_i$ предметних сталих для значень вільних змінних формули $A(x_1, x_2,..., x_n)$ такий, що $A(x_1, x_2,..., x_n) = 1$.

Означення. Формула А є істинною, якщо для будь-якого набору $(a_1,...,a_n)$, $a_i \in M_i$, $A(x_1, x_2,..., x_n) = 1$.

Означення. Формула А є логічно-загальнозначущою в логіці предикатів, якщо вона істина в кожній інтерпретації.

Твердження. Формула $(\forall x)A(x) \rightarrow A(y)$, де у не входить $A(x)$ - логічно-загальнозначуща.

Якщо формула А – 55а цикл-істинна формула логіки висловлювань, а $(x_{i1}, x_{i2},..., x_{in})$ - список її змінних, то підставляючи замість кожної змінної x_i формулу логіки предикатів B_i без порушень правил (1-4) з означення правильно складених формул, одержимо логічно-загальнозначущу формулу логіки предикатів.

Задача розпізнавання загальнозначущої формули логіки предикатів суттєво складніша ніж розпізнавання тавтологій висловлень. Як і в логіці висловлень вона називається проблемою розв'язності і полягає в наступному: вказати ефективний алгоритм розпізнавання логічно-загальнозначущих формул логіки предикатів. У загальному ця проблема не розв'язана.

Формалізація числення предикатів

Числення предикатів будується за класичною схемою побудови формальних теорій, тобто аксіоматичним шляхом. Відмітимо, що аксіоми поділяються на два класи: логічні і власні аксіоми.

Побудова аксіоматичної теорії.

I.Мова.

1.Алфавіт числення предикатів складається з предметних змінних $x_1, x_2, ..., x_n$, констант $(a_1,...,a_n)$, предикатних букв $P_1^1, P_2^2, ..., P_k^i$ і функціональних букв $f_1^1, ,... f_k^i$, а також знаки логічних операцій, кванторів і розділових знаків. Верхні індекси предикатних і функціональних букв вказують на число аргументів, а нижнє використовується для звичайної нумерації.

2. Поняття формули визначається в два етапи. Спочатку вводять поняття терма

а) предметні змінні і предметні константи – терми;

б) якщо $f^{(n)}$ - функціональна буква, а $t_1,...,t_n$ - терми, то $f^{(n)}(t_1,...,t_n)$ також терм;

в) інших термів, крім утворених за правилами а і б, немає.

Далі формулюють означення формули:

а) якщо $P^{(n)}$ - предикатна буква, а $t_1,...,t_n$ - терми, то формула $P^{(n)}(t_1,...,t_n)$ називається

елементарною. Усі входження змінних у формулу $P^{(n)}(t_1,...,t_n)$ називаються вільними;

б) якщо Λ_1 і Λ_2 - формули, то $\Lambda_1 \wedge \Lambda_2, \Lambda_1 \vee \Lambda_2, \Lambda_1 \to \Lambda_2, \overline{\Lambda_1}$ також формули. Усі входження вільних змінних у Λ_1 і Λ_2 є вільними і в усіх наведених формулах;

в) якщо $\Lambda(x)$ - формула, що містить вільне входження змінної х, то $\forall x \Lambda(x)$ і $\exists x \Lambda(x)$ також формули. У цих формулах всі входження змінної х називаються зв'язаними.

Входження решти змінних у формулу називаються вільними;

г) інших формул, ніж побудованих за правилами а, б, в, немає.

II. Аксіоми числення предикатів

Аксіоми числення предикатів утворюють дві групи аксіом:

а) першу групу складають аксіоми довільного числення висловлювань. Як правило ці аксіоми є схемами аксіом;

б) у другу групу входять так звані предикатні аксіоми:

$P1. \forall x A(x) \to A(y)$; $P2. A(y) \to \exists x A(x)$.

У цих аксіомах існує формула, яка містить вільні входження х, причому жодне з них не знаходиться в області дії квантора по у.

Зауваження. Вище наведені аксіоми є логічними аксіомами числення предикатів. Власні аксіоми не можуть бути сформульовані у загальному вигляді, оскільки вони залежать від предметної області М і змінюються від теорії до теорії.

Означення. Якщо числення предикатів не має власних аксіом, то воно називається численням предикатів першого роду.

III. Правилами виведення:

а) правило висновку $\dfrac{A, A \to B}{B}$;

б) правило узагальнення $Gen \dfrac{B \to A(x_i)}{B \to (\forall x_i) A(x_i)}$;

в) правило існування $\mathrm{Re} \dfrac{A(x_i) \to B}{(\exists x_i) A(x_i) \to B}$.

В обох останніх формулах формула $A(x_i)$ містить вільні входження x_i, а формула В їх не містить.

Поняття виведення формули, теореми, виведення формули з множини гіпотез, визначаються у численні предикатів аналогічно численню висловлень. Мають місце наступні теореми.

Теорема. Будь-яка вивідна формула (теорема) числення предикатів є 58а цикл-істинною (логічно-загальнозначущою) формулою.

Теорема. Будь-яка тотожно-істинна предикатна формула є вивідною (теоремою) по численню предикатів.

Теорема. Предикатні формули А і В – рівносильні тоді і тільки тоді, коли формула $A \leftrightarrow B$ є вивідною в численні предикатів.

<u>Контрольні запитання</u>

1. Означення формули алгебри предикатів.
2. Означення формули алгебри предикатів, яка логічно істинна на множині М.
3. Означення логічно загальнозначущої формули алгебри предикатів.
4. Означення рівносильності формул алгебри предикатів: на множині М і в цілому.
5. Формалізація числення предикатів.
6. Властивості числення предикатів.
7. Дати означення формального доведення формули.
8. Сформулюйте аксіоми числення предикатів.
9. Сформулюйте правила виведення.
10. Дати означення вивідної формули в численні висловлень.

<u>***Приклади розв'язування типових задач***</u>

<u>Приклад 1.</u> Чи правильними є доведення?

Нехай a – довільний елемент U. Тоді з огляду на довільність U маємо:

1) $P(a) \vdash \forall x P(x)$;
2) $\forall x P(x) \vdash \forall x P(x) \vee \forall x Q(x)$;
3) $P(a) \vdash \forall x P(x) \vee \forall x Q(x)$:
4) $Q(a) \vdash \forall x Q(x)$;
5) $\forall x Q(x) \vdash \forall x P(x) \vee \forall x Q(x)$;
6) $Q(a) \vdash \forall x P(x) \vee \forall x Q(x)$;
7) $P(a) \vee Q(a) \vdash \forall x P(x) \vee \forall x Q(x)$;
8) $\forall x(P(x) \vee Q(x)) \vdash P(a) \vee Q(a)$;
9) $\forall x\big(P(x) \vee Q(x)\big) \vdash \forall x P(x) \vee \forall x Q(x)$.

Розв'язок:

Очевидно, що доведення є невірним, оскільки невірним є отриманий результат. Це доводить наступна інтерпретація: U=N, *P(x)* – «*x* – парне число», *Q(x)* – «*x* – непарне число».

Проаналізувавши наведену інтерпретацію, можна зробити висновок, що помилка має місце на 4-му кроці наведеного ланцюжка. Адже, для *Q(a) a* не є довільним елементом універсальної множини, а елементом для якого виконується *P(a)*.

<u>*Приклад 2.*</u> Довести тотожність $\forall x(A(x) \wedge B(x)) = \forall x A(x) \wedge \forall x B(x)$.

Розв'язок:

Нехай для визначеності предметна область предикатів складається із двох елементів a і b. Тоді: $\forall x(A(x) \wedge B(x)) = (A(a) \wedge B(a)) \wedge (A(b) \wedge B(b)) =$
$= (A(a) \wedge A(b)) \wedge (B(a) \wedge B(b)) = \forall x A(x) \wedge \forall x B(x)$.

Друга тотожність доводиться аналогічно.

<u>*Приклад 3.*</u> Довести тотожність $\exists x(A \wedge B(x)) = A \wedge \exists x B(x)$.

Розв'язок:

Доведемо другу тотожність, $\exists x(A \wedge B(x)) = (A \wedge B(a)) \vee (A \wedge B(b)) =$
$= A \wedge (A \vee B(a)) \wedge (A \vee B(b)) \wedge (B(a) \vee B(b)) = A \wedge (B(a) \vee B(b)) = A \wedge \exists x B(x)$.

<u>Приклад 4.</u> Довести тотожність $\exists x(A(x) \to B(x)) = \forall x A(x) \to \exists x B(x)$.

Розв'язок:

$$\exists x(A(x) \to B(x)) = \exists x\left(\overline{A}(x) \vee B(x)\right) = \left(\overline{A}(a) \vee B(a)\right) \vee \left(\overline{A}(b) \vee B(b)\right) =$$

$$= \left(\overline{A}(a) \vee \overline{A}(b)\right) \vee (B(a) \vee B(b)) = \exists x\overline{A}(x) \vee \exists x B(x) = \overline{\forall} x A(x) \vee \exists x B(x) = \forall x A(x) \to \exists x B(x).$$

<u>Приклад 5.</u> Довести тотожність $\forall x \forall y P(x, y) = \forall y \forall x P(x, y)$.

Розв'язок:

$$\forall x \forall y P(x, y) = \forall x(P(x, a) \wedge P(x, b)) = (P(a, a) \wedge P(b, a)) \wedge (P(a, b) \wedge P(b, b)) =$$

$$= (P(a, a) \wedge P(a, b)) \wedge (P(b, a) \wedge P(b, b)) = \forall y(P(a, y) \wedge P(b, y)) = \forall y \forall x P(x, y)$$

<u>Приклад 6.</u>

Довести 60а циклювання60ання60 наступної формули логіки предикатів $\forall x(P(x) \to Q(x)) \to (\forall x P(x) \to \forall x Q(x))$

Розв'язок:

$$\forall x\big(P(x) \to Q(x)\big) \to \big(\forall x P(x) \to \forall x Q(x)\big) \equiv$$

$$\overline{\forall x\left(\overline{P(x)} \vee Q(x)\right)} \vee (\overline{\forall x P(x)} \vee \forall x \ \equiv \exists x(P(x) \wedge \overline{Q(x)}) \vee (\exists x \overline{P(x)} \vee \forall x Q(x)) \equiv$$

$$\equiv \exists x(\overline{P(x)} \vee (P(x) \wedge \overline{Q(x)}) \vee \forall x Q(x) \equiv \exists x(\overline{P(x)} \vee \overline{Q(x)}) \vee \forall x Q(x)) \equiv$$

$$\equiv \exists x \overline{P(x)} \vee \exists x \overline{Q(x)} \vee \forall x Q(x) \equiv \exists x \overline{P(x)} \vee \overline{\forall x Q(x)} \vee \forall x Q(x) \equiv 1 \vee \exists x \overline{P(x)} \equiv 1$$

<u>Вправи для самостійного розв'язування</u>

1. Методом побудови довільної інтерпретації над довільною 60а циклюванн множиною довести рівносильності:

1) $\forall x \forall y F(x, y) = \forall y \forall x F(x, y)$;

2) $\exists x \exists y F(x, y) = \exists y \exists x F(x, y)$.

11.Показати, що формули логіки предикатів: $\forall x \exists y P(x, y)$ і $\exists y \forall x P(x, y)$ рівносильні на одноелементній множині, але не є рівносильними на двоелементній множині, отже не є рівносильними.

12.Чи є істинним твердження: «Якщо у формулі логіки предикатів *а* поміняти місцями два одноіменних квантори, то одержана при цьому формула буде рівносильна *a*»? Якщо це твердження істинне, то доведіть його, а якщо воно хибне, то спростуйте , побудувавши відповідний 61а циклювання.

4. Побудувавши відповідну інтерпретацію (61а циклювання), довести, що формули логіки предикатів не є рівносильними:

1) $\exists x\big(P(x) \wedge Q(x)\big)$ i $\exists x P(x) \wedge \exists x Q(x)$;

2) $\forall x\big(P(x) \vee Q(x)\big)$ i $\forall x P(x) \vee \forall x Q(x)$;

3) $\exists x P(x) \to \exists x Q(x)$ i $\exists x\big(P(x) \to Q(x)\big)$.

5. Довести чи спростувати твердження про рівносильність (логічну еквівалентність) таких пар формул логіки предикатів:

1) $\forall x\big(P(x) \to Q(y)\big)$ i $\forall x(P(x)) \to Q(y)$;

2) $\exists x(P(x) \to Q(y))$ i $\exists x(P(x)) \to Q(y)$;

3) $\forall x(P(x)) \to P(y)$ i $\forall x(P(x) \to P(y))$;

4) $Q(x) \to \exists x P(x)$ i $\exists x\big(Q(y) \to P(x)\big)$;

5) $\exists x P(x) \wedge Q(y)$ i $\exists x(P(x) \wedge Q(y))$;

6) $\exists x(P(x)) \to Q(y)$ $i\, \forall x(P(x) \to Q(y))$;

7) $\forall x\big(P(x)\big) \to Q(y)$ i $\exists x(P(x) \to Q(y))$.

6. Довести рівносильності:

а) $\exists x(A(x) \vee B(x)) = \exists x A(x) \vee \exists x B(x)$;

б) $\forall x(A \vee B(x)) = A \vee \forall x B(x)$;

в) $\exists x \exists y P(x, y) = \exists y \exists x P(x, y)$.

7. Довести, що формули логіки предикатів є логічно загальнозначущими, а обернена імплікація – ні:

1) $\exists y \forall x F(x, y) \to \forall x \exists y F(x, y)$;

2) $\forall x(P(x) \to Q(x)) \to (\exists x P(x) \to \exists x Q(x))$;

3) $\forall x(q \to P(x)) \leftrightarrow (q \to \forall x P(x))$;

4) $\forall x P(x) \vee \forall x Q(x) \to \forall x(P(x) \vee Q(x))$;

5) $\forall x(P(x) \to Q(x)) \to (\forall x P(x) \to \forall x Q(x))$;

6) $(\exists x P(x) \to \exists x Q(x)) \to \exists x(P(x) \to Q(x))$.

8. Довести чи спростувати твердження про те, що запропоновані формули є тотожно істинними:

1) $\exists x P(x) \wedge \exists x(P(x) \to Q(x)) \to \exists x Q(x)$;

2) $\exists x P(x) \wedge \forall x(P(x) \to Q(x)) \to \exists x Q(x)$;

3) $\forall x P(x) \wedge \exists x(P(x) \to Q(x)) \to \exists x Q(x)$;

4) $\forall x(P(x) \to (Q(x) \to R(x))) \to \forall x(P(x) \vee Q(x) \to R(x))$;

5) $\forall x(P(x) \vee Q(x)) \to R(x) \to \forall x(P(x) \to (Q(x) \to R(x)))$.

АНАЛІЗ ЗМІСТОВНИХ МІРКУВАНЬ НА БАЗІ ЛОГІКИ ПРЕДИКАТІВ

Теоретичні відомості

Означення. Говорять, що формула логіки предикатів Ω, логічно слідує з формул $\Lambda_1, \ldots, \Lambda_m$ (записують $\Lambda_1, \ldots, \Lambda_m \vdash \Omega$), якщо при будь-якій інтерпретації над довільною областю *M* формула Ω набуває значення істинності 1 при кожному заміщенні вільних предметних змінних предметними константами з *M*, при якому формули $\Lambda_1, \ldots, \Lambda_m$ набувають значення істинності 1. Як і в алгебрі висловлень, формули $\Lambda_1, \ldots, \Lambda_m$ називаються посилками або гіпотезами, а формула Ω – висновком.

Теорема. Логічне слідування $\Lambda_1, \ldots, \Lambda_m \vdash \Omega$ має місце тоді і тільки тоді, коли формула $(\Lambda_1 \wedge \ldots \wedge \Lambda_m) \longrightarrow \Omega$ логічно загальнозначущою.

Якщо формули не містять кванторів, то можна користуватися правилами виведення, встановленими для числення висловлень. Крім цих правил будемо користуватися спеціальними правилами, які застосовуються до формул, що містять квантори. У формулюванні цих правил істотну роль відіграють певні обмеження.

Змінна y називається вільною для x у формулі $\Omega(x)$, якщо жодне вільне входження *x* в $\Omega(x)$ не належить до області дії квантора, який зв'язує *y*.

Спеціальні правила виведення логіки предикатів:

1) $\forall x \Omega(x) \vdash \Omega(y)$, якщо y є вільною для x в $\Omega(x)$ (правило опускання квантора загальності, скорочено ПО ∀).

2) $\exists x \Omega(x) \vdash \Omega(y)$, де a – деяка предметна константа (правило опускання квантора існування ПО ∃).

3) Якщо $\Lambda_1, \ldots, \Lambda_m \vdash \Omega(x)$, то $\Lambda_1, \ldots, \Lambda_m \vdash \forall x \Omega(\mathrm{x})$ при умові, що x не є вільною змінною в жодній з формул $\Lambda_1, \ldots, \Lambda_m$ (правило універсального узагальнення, скорочено ПУУ).

4) $\Omega(y) \vdash \exists x \Omega(x)$, якщо y є вільною для x в $\Omega(x)$ (правило екзистенціального узагальнення, скорочено ПЕУ).

Контрольні запитання

1. В чому полягають операції навішування кванторів?
2. Чи можуть бути зведені операції навішування кванторів до введених раніше операції над предикатами?
3. Дати означення формули алгебри предикатів.
4. Навести приклади формул із вільними та зв'язними змінними.
5. Які формули алгебри предикатів називаються замкнутими? Чи являються такі формули висловленнями?

Приклади розв'язування типових задач

Приклад 1. Записати символами логіки предикатів твердження:

а) кожне число, кратне 10, кратне 5 і кратне 2;

б) кожен квадрат є ромбом, але неправильно, що будь-який ромб є квадратом.

Розв'язок:

а) Відомий предикат для позначення подільності націло – «⁝». Отже, вказане твердження для універсальної множини натуральних чисел записується так :

$$\forall x(x ⁝ 10 \rightarrow x ⁝ 5 \wedge x ⁝ 2)$$

б) Візьмемо за універсальну множину – множину опуклих 64а циклювання. Введемо предикати *R(x)* – «*x* є ромбом», *K(x)* – «*x* є квадратом». Тоді формула матиме вигляд

$$\forall x(K(x) \rightarrow R(x)) \wedge \neg(\forall x(R(x) \rightarrow K(x)))$$

Приклад 2. Перевірити, чи є коректним міркування:

1) Жодна необмежена послідовність не є збіжною.

2) Всі фундаментальні послідовності збіжні.

Отже, жодна необмежена послідовність не є фундаментальною.

Розв'язок:

Введемо наступні предикати:

A(x) – «послідовність *x* обмежена»,

B(x) – «послідовність *x* збіжна»,

C(x) – «послідовність *x* фундаментальна».

Тоді посилки нашого міркування запишуться у вигляді:

13.$\forall x(\neg A(x) \to \neg B(x))$, 2) $\forall x(C(x) \to B(x))$, а висновок – у вигляді $\forall x(\neg A(x) \to \neg \mathrm{C}(x))$. Нам потрібно перевірити, чи має місце логічне слідування:

$\forall x\big(\neg A(x) \to \neg B(x)\big), \forall x(C(x) \to B(x)) \vdash \forall x(\neg A(x) \to \neg \mathrm{C}(x))$.

Неважко переконатися, що має місце така формальна вивідність:

1) $\forall x(\neg A(x) \to \neg B(x))$ – посилка,

2) $\forall x(C(x) \to B(x))$ – посилка,

3) $\neg A(y) \to \neg B(y)$ (ПО $\forall$(1)),

4) $C(y) \to B(y)$ (ПО $\forall$(2)),

5) $\neg B(x) \to \neg C(y)$ (ПК (4)),

6) $\neg A(y) \to \neg \mathrm{C}(y)$ (ПС (3,5)),

7) $\forall y(\neg A(y) \to \neg C(y))$ (ПУУ (6)),

8) $\forall x(\neg A(x) \to \neg C(x))$ (перейменування зв'язної змінної в (7)).

Отже, дане міркування коректне, очікуваний висновок логічно слідує з прийнятих посилок.

<u>*Приклад 3.*</u> Записати символами логіки предикатів твердження та перевірити яке із тверджень сильніше:

1) Існує таке *x*, що *P(x)* – хибне.

2) «Існує таке *x*, що *P(x)»* – хибне.

Розв'язок:

1) $\exists x(\neg P(x))$;

2) $\neg(\exists x P(x))$.

2-ге твердження є сильнішим, оскільки, із нього завжди слідуватиме перше. Справді, очевидно, що неможливим є випадок $|\neg(\exists x P(x))| = 1$ і $|\exists x(\neg P(x))| = 0$. З іншого боку простий приклад показує, що із першого твердження не завжди слідує друге.

Нехай $\Omega \coloneqq N$ і *P(x)* – *«*x⋮3*»*. Тоді в цій інтерпретації отримаємо, $|\neg(\exists x P(x))| = 0$, $|\exists x(\neg P(x))| = 1$.

<u>Приклад 4.</u> Універсальна множина є множина R. Які з наведених висловлень істинні, а які хибні. У випадку хибного висловлення навести 66а циклювання66.

1) $\forall x \forall y \exists z(x \wedge y = z)$;
2) $\forall x(x^0 = 1)$;
3) $\forall x(\frac{2x-6}{x-3} = 2)$.

Розв'язок:

Перше висловлення є істинним, тому що операція множення замкнена на множині дійсних чисел.

Висловлення 2) та 3) – хибні. У випадку 2) твердження під знаком квантора загальності – хибне, при $x = 0$, а 3) – при $x = 3$.

<u>Вправи для самостійного розв'язування</u>

14.Порівняти символічні записи тверджень:

1) Кожен студент3-го курсу передплачує хоча б одну газету.
2) Існує хоча б одна газета, яку передплачує кожний студент 3-го курсу.

Яке з цих тверджень сильніше.

15.Зобразити на діаграмах Венна запропоновані твердження, записавши їх перед тим символічною мовою алгебри предикатів:

1) Усі правильні трикутники є рівнобедреними, тоді як деякі рівнобедрені трикутники не є правильними.

2) Жодне 67а циклювання67 число не є раціональним, а деякі дійсні числа не є ні раціональними, ні 67а циклювання67ан.

3) Не всі комплексні числа – уявні, але всі уявні числа є комплексними.

3. Визначити, які з тверджень є логічно еквівалентні:

1) Неправильно, що всі числа, кратні 4, є точними квадратами.

2) Всі числа, кратні 4, не є точними квадратами.

3) Не всі числа, кратні 4, є точними квадратами.

4) Існує число, кратне 4, яке не є точним квадратом.

5) Деякі числа, не кратні 4, не є точними квадратами.

4. Визначити, які з тверджень є логічно еквівалентні:

1) Неравильно, що існує послідовність (x_n), для якої існує найбільший член і не існує точної верхньої межі.

2) Кожна послідовність, для якої існує найбільший член, має точну верхню межу.

3) Не існує послідовності (x_n), у якої є найбільший член і немає точної верхньої межі.

4) Існує послідовність (x_n), яка має найбільший член і має точну верхню межу.

5) Жодна послідовність (x_n), яка не має точної верхньої межі, не має найбільшого члена.

5. Перевірити чи випливають на базі логіки предикатів з даних припущень зроблені з них висновки, тобто чи є правильними (коректними) проведені міркування. Якщо міркування правильне, то побудувати відповідний дедуктивний ланцюжок (від припущень до висновку, якщо ж проведене міркування – не коректне, то побудувати відповідний 67а циклювання).

1) Деякі неперервні на $[a, b]$, функції – 67а циклювання67а на $[a, b]$. Всі 67а циклювання67а на $[a, b]$ функції – 67а циклюван на $[a, b]$. Отже всі неперервні на $[a, b]$ функції є 67а циклювання на $[a, b]$.

2) Деякі вписані в коло 4-кутники є прямокутниками. Кожний прямокутник є паралелограмом. Отже, деякі вписані в коло 4-кутники є паралелограмами.

3) Кожна збіжна послідовність – обмежена. Кожна обмежена послідовність має точну верхню і точно нижню межі. Отже, кожна збіжна послідовність має точну верхню і нижню межі.

4) Кожне число, кратне 153, кратне 17 і кратне 9. Кожне число – кратне 9 тільки тоді, коли сума цифр запису цього числа в десятковій системі кратна 9. Сума цифр запису числа 23878 – не кратна 9. Отже, 23878 – не кратне 153.

5) Кожний дріб є виразом. Всяке дробове число є числом. Жодне число не є виразом. Отже, жодне дробове число не є дробом.

6) Кожна зліченна множина є нескінченною. Кожна незліченна множина є нескінченною. Дана множина E –скінченна. Жодна скінченна множина не є нескінченною. Отже, існує множина, яка не є зліченною і не є незліченною.

16.Оцінити істинність чи хибність наступних висловлень. У разі хибності висловлення – навести відповідний 68а циклювання:

1) $\forall x(\sqrt{x^2} = x)$;

2) $\forall x(\sqrt{x^2} = x) \to \forall y(y^2 > y)$;

3) $\forall x \forall y(\frac{x}{y} > 1 \to x > y)$;

4) $\forall x((x > \pi) \wedge (x < e) \leftrightarrow \neg(x = x))$.

17.Порівняти область дії квантора і значення істинності виразів

$\forall x(x > 2) \to 1 > 2$ і $\forall x(x > 2 \to 1 > 2)$.

8. Універсальна множина – множина всіх натуральних чисел N. Проаналізувати область дії кванторів і значення істинності висловлень:

1) $\forall x \forall y(\exists z(z > x \wedge z < y) \leftrightarrow x < y)$;

2) $\forall x \forall y \exists z((z > x \wedge z < y) \leftrightarrow x < y)$.

9. Оцінити істинність математичних тверджень:

1) $\forall x \forall y \exists z(x < z \wedge z < y \vee y < z \wedge z < x)$;

2) $\forall x \forall y \exists z(x < y \rightarrow x < z \wedge z < y)$

У випадку, коли універсальна множина є: множина всіх натуральних чисел; множина всіх цілих чисел; множина всіх раціональних чисел; множина всіх дійсних чисел.

18.Визначити чи є істинними на універсальній множині дійсних чисел такі математичні твердження:

1) $\forall x \forall y \exists z(z = \sqrt{x^2+y^2})$;

2) $\forall x \forall y \exists z(z = \ln(x^2+y^2))$;

3) $\forall x \forall y(\ln(\mathrm{xy}) = \ln x + \ln y)$;

4) $\forall x \exists y(xy) = 1$;

5) $\forall x \exists y(x^2 + y^2 = 1)$;

6) $\forall x \forall y(x^2 + y^2 = 0 \rightarrow x = 0 \wedge y = 0)$

ВИПЕРЕДЖЕНІ НОРМАЛЬНІ ФОРМИ І ЛОГІЧНИЙ ВИСНОВОК У ЛОГІЦІ ПРЕДИКАТІВ

Теоретичні відомості

Елементарна формула, правильно побудовані формули, область дії квантора, інтерпретація формул логіки предикатів, загальнозначущі та суперечливі формули, логічний наслідок.

Використовуючи поняття предиката, квантора і терма, можна визначити поняття формули у логіці предикатів.

Якщо P — n-місний предикат і t_1, …, t_n— терми, то $P(t_1, \ldots, t_n)$ називається ***атомом*** або ***елементарною формулою*** логіки предикатів. Наприклад: ДІЛИТЬСЯ(x, 13), ДІЛИТЬСЯ(x, y), БІЛЬШЕ(плюс(x, 1), x), ДОРІВНЮВАТИ(x, 1), СКЛАДАТИ(студенти, сесії).

Правильно побудованими формулами логіки предикатів називаються формули, які можна рекурсивно визначити таким чином:

I.Атом є формулою.

2. Якщо F і G— формули, то $(\neg F)$, $(F \vee G)$, $(F \wedge G)$, $(F \rightarrow G)$, $(F – G)$ також є формулами.

3. Якщо F — формула, а x — вільна змінна, то $(\forall x)\ F$ і $(\exists x)F$ теж формули.

4. Ніяких формул, крім породжених вказаними вище правилами, не існує.

Частина формули, на яку поширюється дія квантора, називається ***областю дії квантора.***

Оскільки дія квантора може поширюватися не на всю формулу, а тільки на її частину, то змінна може бути зв’язаною в одній частині формули і вільною в другій. У цьому випадку вважають, що змінна є і зв’язаною, і вільною одночасно.

В логіці висловлень інтерпретація формули полягає у приписуванні атомам істиннісних значень. У логіці предикатів поняття інтерпретації формули декілька поширюється: необхідно вказати предметну область (область значень предметних змінних) і значення констант, а також функціональних символів і предикатів, що зустрічаються у формулі.

Інтерпретація формули F логіки предикатів складається з елементів 71а циклюванн предметної області D, значень всіх констант, функціональних символів і предикатів, що зустрічаються в F. Вказані значення задаються таким чином:

1. Кожній константі ставиться у відповідність деякий елемент з D.

2. Кожному n-місному функціональному символу ставиться у відповідність відображення з D^n у D. Тут $D^n = (x_1, x_2, \ldots, x_n)$, де $x_1, \ldots, x_n \in D$.

3. Кожному n-місному предикату ставиться у відповідність відображення з D^n в $\{I, X\}$.

Для кожної інтерпретації на області D формула може одержати істиннісне значення I або X згідно з такими правилами:

1. Якщо задані значення формул F і G, то 71а циклюван значення формул $(\neg F)$, $(F \vee G)$, $(F \wedge G)$, $(F \rightarrow G)$, $(F \sim G)$ одержуються за допомогою таблиць істинності відповідних логічних операцій.

2. Формула $(\forall x)F$ одержує значення I, якщо F одержує значення I для кожного x з D, у протилежному випадку вона одержує значення X.

3. Формула $(\exists x)F$ одержує значення I, якщо F одержує значення I хоча б для одного x з D, у протилежному випадку вона одержує значення X.

4. Формула, що містить вільні змінні, не може одержати істиннісне значення.

Колізія змінних, заміна зв'язаної змінної, комутативні і дистрибутивні властивості кванторів, закон де Моргана для кванторів.

Всі закони і тотожності, які справедливі у логіці висловлень, залишаються справедливими і у логіці предикатів. Крім того, у логіці предикатів існують додаткові законі, що призначені для еквівалентного перетворення формул, що містять квантори та змінні.

Слід зауважити, що перенесення квантора на початок формули може змінити зміст предикатного висловлення. Так, наприклад, $\forall x\, F(x) \wedge \forall x\, G(x)$ не еквівалентне $\forall x\,(F(x) \wedge G(x))$. У загальному випадку слід перейменувати зв'язані змінні, щоб запобігти *колізії*— ситуації, коли у формулі одна й та ж змінна знаходиться в області дії протилежних кванторів. Наприклад, у формулі $\forall x\,(F(x)) \rightarrow \exists x\,(Q(x))$ змінна x одночасно знаходиться в області дії кванторів $\forall$ і $\exists$.

Розглянемо висловлення, що використовує предикат рівності ДОРІВНЮЄ двох чисел: $\forall x\ \exists y$ ДОРІВНЮЄ(x + 1, y). Дане висловлення означає, що для будь-якого числа x існує число y, яке більше його на одиницю. Наведене висловлення є істинним. Однак, якщо поміняти порядок розташування кванторів на протилежний, то одержимо таке висловлення: $\exists y\ \forall x$ ДОРІВНЮЄ(x + 1, y). Одержане висловлення означає, що існує таке число y (одне!), яке на одиницю більше будь-якого числа x. Це висловлення не відповідає попередньому і є хибним.

Для еквівалентних перетворень предикатних висловлень з кванторами необхідно використовувати наведені нижче закони. Перш ніж безпосередньо перейти до розглядання законів дій з кванторами, введемо такі позначення: .F(x) і $H(x)$ — одномісні предикати, $P(x, y)$ — двомісний предикат.

1. ***Заміна зв'язаної змінної:***

$(\exists x)\ F(x) = (\exists y)\ F(y)$; $(\forall x)\ F(x) = (\forall y)\ F(y)$.

Введення нового позначення зв'язаної змінної (тобто перейменування зв'язаної змінної) не змінює зміст формули логіки предикатів, якщо виконується така умова: ніяка вільна змінна у будь-якій частині формули не повинна після перейменування бути зв'язаною. Іншими словами, для нового позначення зв'язаної змінної слід обирати букву, яка відсутня у формулі. Наприклад: $(\forall x)\ (\exists y)\ P(x, y) = (\forall x)\ (\exists z)\ P(x, z)$.

У даному прикладі здійснено операцію перейменування зв'язаної змінної y.

2. ***Комутативні властивості кванторів:***

$(\forall x)\ (\forall y)\ P(x, y) = (\forall y)\ (\forall x)\ P(x, y)$;

$(\exists x)\ (\exists y)\ P(x, y) = (\exists y)\ (\exists x)\ P(x, y)$.

Змінювати місцями можна тільки однойменні квантори.

$(\forall x)\ (\exists y)\ P(x, y) \neq (\exists y)\ (\forall x)\ P(x, y)$.

3. ***Дистрибутивні властивості кванторів:***

$(\forall x)F(x) \vee G = (\forall x)(F(x) \vee G)$;

$(\exists x)F(x) \vee G = (\exists x)(F(x) \vee G)$;

$(\forall x)F(x) \wedge G = (\forall x)(F(x) \wedge G)$;

$(\exists x)F(x) \wedge G = (\exists x)(F(x) \wedge G)$,

де G — формула логіки предикатів, яка не містить x;

$(\forall x)F(x) \wedge (\forall x)H(x) = (\forall x)(F(x) \wedge H(x))$;

$(\exists x)F(x) \vee (\exists x)H(x) = (\exists x)(F(x) \vee H(x))$.

Сформульований дистрибутивний закон справедливий тільки для квантора загальності $\forall$ при кон'юнкції $\wedge$ і квантора існування $\exists$ при диз'юнкції $\vee$, оскільки інші комбінації призводять до нерівностей:

$(\forall x)F(x) \vee (\forall x)H(x) \neq (\forall x)(F(x) \vee H(x))$;

$(\exists x)F(x) \wedge (\exists x)H(x) \neq (\exists x)(F(x) \wedge H(x))$.

Для подолання цього обмеження дистрибутивного закону, слід використовувати заміну зв'язаної змінної:

$(\forall x)F(x) \vee (\forall x)H(x) = (\forall x)F(x) \vee (\forall y)H(y) = (\forall x)(\forall y)\ (F(x) \vee H(y))$;

$(\exists x)F(x) \wedge (\exists x)H(x) = (\exists x)F(x) \wedge (\exists y)H(y) = (\exists x)(\exists y)\ (F(x) \wedge H(y))$.

Таким чином, у загальному випадку дистрибутивні властивості кванторів можна записати такою схемою:

$(Q_1x)F(x) \vee (Q_2y)\ H(y) = (Q_1x)(Q_2y)\ (F(x) \vee H(y))$;

$Q_1x)F(x) \wedge (Q_2y)\ H(y) = (Q_1x)(Q_2y)\ (F(x) \wedge H(y))$,

Де Q_1, Q_2 — будь-який з кванторів $\exists$ або $\forall$.

4. ***Закон де Моргана для кванторів:***

$\neg((\forall x)F(x)) = (\exists x)\neg F(x)$;

$\neg(\exists x)F(x)) = (\forall x)\neg F(x)$.

Випереджена нормальна форма, алгоритм зведення до випередженої нормальної форми, правила видалення/введення квантора загальності/існування

В логіці висловлень було введено дві нормальні форми: кон'юнктивна і диз'юнктивна. В логіці предикатів вводиться третя нормальна форма, що називається випередженою нормальною формою.

Формула F в логіці предикатів знаходиться у ***випередженій нормальній формі (ВНФ)*** тоді і тільки тоді, коли вона може бути зображена у вигляді $(Q_1x_1)\ldots(Q_nx_n)(M)$, де кожне (Q_ix_i), $i = 1, \ldots, n$, є або $(\forall x)$, або $(\exists x)$, а M— формула, що не містить кванторів. Причому $(Q_1x_1)\ldots(Q_nx_n)$ називається ***префіксом***, а M — ***матрицею формули*** F.

Для перетворення виразів довільної форми у ВНФ необхідно по черзі виконати такі етапи перетворення.

1. Виключити логічні зв'язки еквіваленції (~) та імплікації ($\rightarrow$), виразивши їх через операції диз'юнкції, кон'юнкції і заперечення за допомогою таких законів: $F \rightarrow G = \neg F \vee G$; $F \sim G = (\neg F \vee G) \wedge (\neg G \vee F) = \neg F \wedge \neg G \vee F \wedge G$.

2. Опустити знаки операцій заперечення безпосередньо на предикати, використовуючи закон подвійного заперечення $\neg(\neg F) = F$ і закони де Моргана $\neg(F \vee G) = \neg F \wedge \neg G$, $\neg(F \wedge G) = \neg F \vee \neg G$, у тому числі для кванторів:
$\neg((\forall x)F(x)) = (\exists x)(\neg F(x))$; $\neg((\exists x)F(x)) = (\forall x)(\neg F(x))$.

3. Якщо необхідно, перейменувати зв'язані змінні.

4. Винести квантори на початок формули, використовуючи відповідні закони, для одержання випередженої нормальної форми.

<u>Приклади розв'язування типових задач</u>

<u>Приклад 1.</u> Визначити, які змінні є зв'язаними, а які — вільними у таких формулах:

1. $A(x, y)$.
2. $\exists y\ (B(x) \rightarrow \forall x\ A(x, y))$.
3. $\exists x\ (B(x) \rightarrow \forall x\ A(x, y))$.

Розв'язок. Обидві змінні у формулі 1 є вільними. У формулі 2 змінна y є зв'язаною, а змінна x – і зв'язаною, і вільною (змінна x вільна у предикаті $B(x)$ і зв'язана у предикаті $A(x, y)$). 15 формулі 3 змінна x є зв'язаною, а змінна y — вільною.

<u>*Приклад 2.*</u> Звести формулу $(\forall x)F(x) \to (\exists x)Q(x)$ до ВНФ.

Розв'язок. Спочатку виключимо імплікацію, потім опустимо знак операції заперечення безпосередньо на предикат і винесемо квантор на початок:

$$(\forall x)F(x) \to (\exists x)H(x) = \neg((\forall x)F(x)) \vee (\exists x)H(x) = (\exists x)(\neg F(x)) \vee (\exists x)H(x) = (\exists x)(\neg F(x) \vee H(x))$$

<u>*Приклад 3.*</u> Одержати ВНФ для формули

$$G \equiv (\forall x)(\forall y)((\forall z)P(x, y) \wedge P(y, z)) \to (\exists z)R(x, y, z).$$

Розв'язок. Скористаємося наведеним вище алгоритмом.

$$G = (\forall x)(\forall y)(\neg((\forall z)P(x, z) \wedge P(y, z)) \vee (\exists z)R(x, y, z)) =$$
$$= (\forall x)(\forall y)((\exists z)(\neg P(x, y) \vee \neg P(y, z)) \vee (\exists u)R(x, y, u)) =$$
$$= (\forall x)(\forall y)(\exists z)(\exists u)(\neg P(x, y) \vee \neg P(y, z) \vee R(x, y, u)).$$

Контрольні запитання

1. Дайте визначення випередженої нормальної форми.

2. За допомогою яких законів можна опустити знаки операцій заперечення безпосередньо на предикати?

3. Сформулюйте алгоритм перетворення виразів довільної форми у ВНФ.

4. Назвіть правила висновку, які можна використовувати для проведення дедуктивних умовиводів з висловленнями логіки предикатів.

5. В чому полягає правило видалення квантора загальності?

6. Як і навіщо використовується правило введення квантора загальності?

7. Поясніть відмінність у трактуванні елемента предметної області у правилі видалення квантора існування від прийнятої у правилі введення квантора загальності.

8. Яке правило введення квантора існування?

<u>***Вправи для самостійного розв'язування***</u>

1. Звести до ВНФ такі вирази:

а) $(\forall y)(F_1(y) \vee \neg(\exists x)P(x, y))$;

б) $\neg(\exists x)((\forall y)A(x,y)\wedge(\exists y)(\forall z)(C(z)\rightarrow B(x,y)))$;

в) $(\exists x)(\forall y)B(x,y)\sim(\exists x)A(x)$;

г) $\neg(\forall y)(\exists x)(A(x)\rightarrow B(y))$;

д) $(\forall x)F_1(x)\rightarrow\neg(\forall x)(F_2(y)\vee(\forall y)P(x,y))$.

2. Визначте, чи є формула $(\exists x)$ $(P(x) \wedge Q(x))$ логічним наслідком формул $(\exists x)$ $P(x)$ і $(\exists x)$ $Q(x)$.

3. Застосовуючи дедуктивні правила логіки предикатів, наведіть висновки з таких засновків:

3.1.Якщо хтось з тих людей — автор цих пліток, то він глупий і безпринципний. Але ніхто з тих людей не глупий і не позбавлений принципів.

3.2.Якщо всі ці люди не хоробрі або на них не можна покластися, то вони не належать до нашої компанії. Але вони належать до нашої компанії.

3.3.Якщо хтось з підозрілих здійснив всі ці нерозкриті крадіжки, то він був ретельно підготовлений і мав співучасника. Якщо б всі крадіжки були підготовлені ретельно, то, якщо б був співучасник, вкрадено було б набагато більше. Але останнє не має місця.

3.4.Якщо один з нас піде завтра на перше заняття, то він повинен буде підвестися рано, а якщо ми підемо сьогодні ввечері у кіно, то він ляже пізно спати. Якщо будь-який з нас ляже пізно спати, а підведеться рано, то буде задовольнятися п’ятьма годинами сну. Але ми не можемо задовольнятися п’ятьма годинами сну.

3.5.В бюджеті виникне дефіцит, якщо і тільки якщо не підвищать деякі мита. Державні витрати на всі соціальні нестатки скоротяться, якщо і тільки якщо у бюджеті буде дефіцит. Деякі мита підвищать.

Якщо всі ціни одночасно підвищуються, то підвищується і заробітна плата. Всі ціни високі або застосовується регулювання цін. Якщо застосовується регулювання цін, то немає інфляції. Спостерігається інфляція.

КВАНТОРИ

<u>Теоретичні відомості</u>

При визначенні істиннісного значення предиката неабиякий інтерес становить питання: чи є він істинним при будь-якому значенні предметної змінної або чи існує хоча б одне значення змінної, при якому цей предикат істинний. Наприклад, твердження *«Всі прості числа мають два дільника»* можна формалізувати за допомогою предиката МАТИ_ДВА_ДІЛЬНИКА(x), який є істинним для всіх x у предметній області простих чисел. Твердження *«Існують натуральні числа, які не діляться на 2»* означає, що предикат ДІЛИТЬСЯ_НА_2(x) істинний не для всіх x у предметній області натуральних чисел.

Нехай $P(x)$ — предикат, визначений на M. Висловлення «для всіх $x \in M$, $P(x)$ істинне» позначається $\forall x\ P(x)$. Знак $\forall$ називається ***квантором загальності.***

Квантори керують областю значення змінної, наступної за символом квантора. Якщо застосовується квантор загальності, то ми говоримо, що висловлення істинне для всіх х з деякої множини.

Висловлення «існує таке $x \in M$, що $P(x)$ істинне» позначається $\exists x\ P(x)$, де знак $\exists$ називається ***квантором існування***.

Квантор існування застосовується, коли треба вказати, що існує хоча б одне значення змінної, для якого істинне це висловлення.

В логіці предикатів (або першого порядку) існує таке обмеження: не можна застосовувати квантори до предикатів. Наприклад, не можна записати $\forall P\ P(x)$. Однак такі операції здійсненні у логіках більш високих порядків.

Перехід від $P(x)$ до $\forall x\ P(x)$ або $\exists x\ P(x)$ називається ***зв'язуванням*** змінної x, а сама змінна x у цьому випадку — ***зв'язаною.***

Змінна, не зв'язана ніяким квантором, називається ***вільною***.

Від того, чи є змінна зв'язаною або вільною, залежить значення предиката. Вільна змінна — це предметна змінна, яка може приймати різні значення з множини M, і значення предиката $P\ (x)$ залежить від значення змінної x. Навпаки, вираз $\forall x\ P(x)$ не залежить від змінної x і при заданих P і M має визначене значення; тут x — зв'язана змінна.

Зв'язані змінні зустрічаються не тільки у логіці. Наприклад, у виразах $\sum_{x=1}^{10} f(x)$ або $\int_a^b f(x)dx$ змінна x зв'язана і дані вирази при фіксованих a, b і f мають визначені значення, не залежні від будь-якого значення x.

<u>Приклади розв'язування типових задач</u>

<u>Приклад 1.</u> Записати у вигляді предикатів з кванторами такі висловлення: *«Всі студенти складають іспити», «Деякі студенти складають іспити на відмінно».*

Розв'язок. Введемо предикати: P — «складати іспити» і Q—«складати іспити на відмінно». Предметна область даних предикатів являє множину студентів. Тоді вихідні вирази набудуть вигляду:

$\forall(x)\ P(x)$ і $\exists(x)\ Q(x)$.

<u>Приклад 2.</u> Розглядаючи як предметну область множину дійсних чисел, записати у вигляді виразу логіки предикатів математичні твердження:

«Для всіх x правильно, що $(x-1)^2 = x^2 - 2x + 1$»;

«Існує число, квадрат якого дорівнює 4».

Розв'язок. Введемо предикат ДОРІВНЮЄ(x, y), який істинний у тому випадку, якщо значення змінної x дорівнює значенню y. Тоді, використовуючи квантори, можна записати:

$\forall x$ ДОРІВНЮЄ($(x-1)^2$, (x^2-2x+1)); $\exists x$ ДОРІВНЮЄ(x^2, 4).

Застосування кванторів до багатомісних предикатів зменшує кількість вільних змінних, від яких залежить цей предикат. Нехай $A(x, y)$ — деякий двомісний предикат, визначений на довільній множині M. Квантор загальності і квантор існування можна застосувати до неї як для змінної x, так і для змінної y: $\forall x A(x,y)$; $\forall y A(x,y)$; $\exists x A(x,y)$; $\exists y A(x,y)$.

Всі чотири наведені вирази є записами одномісних предикатів від відповідної вільної змінної. Так, $\forall x A(x,y)$ — одномісний предикат від змінної y: $\forall x A(x,y) = F(y)$. Предикат F істинний точно для таких елементів $b \in M$, для яких предикат $A(x, b)$ істинний на всіх значеннях аргументу x. Якщо зобразити множину значень істинності предиката $A(x, y)$ у вигляді матриці, то предикат $F(y) = \forall x\, A(x, y)$ істинний для таких $y = b$, для яких стовпчик аргументу $y = b$ містить виключно букву I. Для ілюстрації нижче наведено матрицю предиката (таблиця 1), що визначений на множині M з п'яти елементів a_1, a_2, a_3, a_4, a_5, і матриці одномісних предикатів (таблиці 2-5), що одержані з вихідного за допомогою застосування кванторів.

Таблиця 1. Предикат $A(x, y)$

$X \backslash Y$	a_1	a_2	a_3	a_4	a_5
a_1	*I*	*I*	*X*	*I*	*X*
a_2	*X*	*I*	*X*	*I*	*X*
a_3	*I*	*I*	*X*	*I*	*I*
a_4	*X*	*I*	*X*	*I*	*X*
a_5	*I*	*I*	*X*	*I*	*I*

Таблиця 2. Предикат $\forall x\, A(x, y)$

y	$\forall x\, A(x, y)$
a_1	*X*
a_2	*I*
a_3	*X*
a_4	*I*
a_5	*X*

Таблиця 3. Предикат $\exists x\, A(x, y)$

y	$\exists x\, A(x, y)$
a_1	*I*
a_2	*I*
a_3	*X*
a_4	*I*
a_5	*I*

Таблиця 4. Предикат $\forall y\, A(x, y)$

x	$\forall y\, A(x, y)$
a_1	*X*
a_2	*X*
a_3	*X*
a_4	*X*
a_5	*X*

Таблиця 5. Предикат $\exists y\, A(x, y)$

x	$\exists y\, A(x, y)$
a_1	*I*
a_2	*I*
a_3	*I*
a_4	*I*
a_5	*I*

Подібним чином $\forall y\, A(x, y) = C(x)$ — одномісний предикат від x, що істинний для таких $a \in M$, для яких рядок аргументу $x = a$ містить тільки букву

I. Предикат $\exists x\ A(x, y) = =H(y)$ істинний для таких $a \in M$, для яких відповідний стовпчик $y = b$ містить, щонайменше, один раз букву I. Нарешті, предикат $\exists y\ A(x, y)$ істинний для таких $x = a \in M$, для яких у відповідному рядку $x = a$ зустрічається буква I.

Таким чином, застосування квантора за однією із змінних двомісного предиката перетворює його на одномісний. У випадку трьохмісних предикатів застосування квантора призводить до двомісного предикату. Аналогічно, для n-місних предикатів застосування квантора за будь-якою змінною перетворює предикат на $(n - 1)$-місний, тобто зменшує його порядок на одиницю.

Квантор загальності можна інтерпретувати як узагальнення кон’юнкції, а квантор існування — як узагальнення диз’юнкції. Насправді, якщо область визначення M предиката P скінченна, наприклад, $M = \{a_1, a_2, \ldots, a_n\}$, то висловлення $\forall x\ P(x)$ еквівалентне кон’юнкції $P(a_1) \wedge P(a_2) \wedge \ldots \wedge P(a_n)$, а висловлення $\exists x\ P(x)$— диз’юнкції $P(a_1) \vee P(a_2) \vee \ldots \vee P(a_n)$.

Як приклад розглянемо предикат $P(x)$, який означає «x — непарне число» і визначений на області $M = \{a, b, c\}$. Висловлення $\forall x\ P(x)$ означає: «a — непарне число, і b — непарне число, і c— непарне число»; а висловлення $\exists x\ P(x)$ означає те ж, що і диз’юнкція «a — непарне число, або b— непарне число, або c— непарне число».

Контрольні запитання

1. Що розуміють під квантором загальності?
2. Дайте визначення поняттю квантор існування.
3. Які змінні називаються зв’язаними, а які — вільними?
4. Поясніть на прикладах призначення зв’язаних змінних.
5. До яких наслідків призводить застосування квантора за однією із змінних n-місного предиката?
6. Скількома різними способами може бути застосований квантор існування до n-місного предикату?
7. Коли предикат $\forall x\ A(x, y)$ приймає значення «Істина»?

<u>Вправи для самостійного розв'язування</u>

1. Запишіть такі висловлення, використовуючи знаки кванторів:

а) існує число x таке, що $x + 1 = 5$;

б) яким би не було число y, $y + 0 = y$;

в) будь-яке число або додатне, або від'ємне, або дорівнює нулю.

2. Вкажіть вільні та зв'язані входження кожної із змінних у таких формулах:

а) $\forall x\, P(x, y) \wedge \forall y\, Q(y)$;

б) $\forall x\, (P(x) \to P(.y))$;

в) $\forall x\, (P(x) \to Q(y)) \vee \exists y\, R\, (x, y)$;

г) $\forall x\, [(P(x) \to Q(y)) \vee \exists y\, R\, (x, y)]$.

3. Нехай x і y— будь-які люди, $Q(x, y)$ означає «x батько y». Наведені висловлення сформулюйте природною мовою, визначивши їх значення істинності:

а) $\forall x\, \exists y\, Q(x, y)$;

б) $\exists x\, \forall y\, Q(x, y)$;

в) $\forall y\, \exists x\, Q(x, y)$;

г) $\exists y\, \forall x\, Q(x, y)$;

д) $\forall x\, \forall y\, Q(x, y)$;

е) $\exists x\, \exists y\, Q(x, y)$.

4. Нехай $N(x)$ — «x — натуральне число», $C(x)$ — «x — ціле число», $P(x)$ — «x — просте число», $E(x)$ — «x — парне число», $O(x)$— «x— непарне число», $D(x, y)$— «y ділиться на x». Сформулюйте природною мовою наведені висловлення, встановивши їх значення істинності:

а) $P(z)$;

б) $E(2) \wedge P(2)$;

в) $\forall x\, (D(2, x) \to E(x))$;

г) $\exists x\, (E(x) \wedge D(x, 6))$;

д) $\forall x\, [P(x) \to \exists y\, (E(y) \wedge D(x, y))]$;

е) $\forall x\, (N(x) \to C(x))$;

ж) $\exists x\ (N(x) \to C(x))$;

з) $\forall x\ (C(x) \to N(x))$;

і) $\forall x\ \forall y\ [O(x) \to (P(y) \to D(x, y))]$;

к) $\forall x\ [C(x) \to (E(x) \vee \neg E(x))]$;

л) $\exists x\ \forall y\ [(C(x) \wedge C(y)) \to D(x, y)]$;

м) $\forall x\ \forall y\ [(E(x) \wedge O(x)) \to \neg D(x, y)$

5. Предикат $P(x, y)$ задано в предметній області $D = \{a, b\}$ такою матрицею:

x	a	a	b	b
y	a	b	a	b
$P(x, y)$	0	1	1	1

Яка з нижченаведених формул визначає цей предикат?

а) $\forall x\ P(x, a)$;

б) $\forall y\ P(a, y)$;

в) $\exists y\ \forall x\ P(x, y)$;

г) $\forall y\ \forall x\ P(x, y)$;

д) $\forall y\ \forall x\ \neg P(x, y)$.

БАГАТОЗНАЧНА ЛОГІКА

Виникнення багатозначних логік, значення істинності висловлення, алфавіт багатозначної логіки, унарні і бінарні функції, повна система функцій багатозначної *логіки*

Вперше багатозначна логіка з'явилася через заперечення аристотелева закону виключеного третього. Відповідно до цього закону диз'юнктивне висловлення $p \vee \neg p$ є тавтологія, а атомарне висловлення p у аристотелевій логіці завжди або істинне, або хибне. Оскільки в аристотелевій логіці будь-яке висловлення може приймати тільки одне з двох значень істинності (істину або хибність), вона одержала назву ***двозначної логіки***. В 1921 році Я. Лукашевич у маленькій статті на двох сторінках розглядає трьохзначну логіку, тобто таку логіку, в якій будь-яке висловлення p може приймати одне з трьох можливих значень істинності. Незалежно від Лукашевича Е. Пост аналізує ***m-значну*** логіку, в якій висловлення p може приймати одне з m можливих значень істинності, де m — будь-яке ціле число, більше за 1. У випадку, коли m більше за 2, логіку називають ***багатозначною***. В 1930 році Лукашевич і Тарський приступають до подальшого вивчення ***m-значної*** логіки. В 1932 році поняття ***m-значної*** логіки узагальнюється Г. Рейхенбахом, що розглядає нескінченнозначну логіку, в якій для висловлення p існує нескінченна множина значень істинності.

Видатний вчений А. Гейтінг приблизно у то й же час побудував двозначну символічну логіку, виходячи з потреб інтуїціониської математичної школи. Ця логіка, на відміну від аристотелевої, не приймає беззаперечно законів виключеного третього і подвійного заперечення. Внаслідок цього закони створеної зі спеціальними цілями логіки Гейтінга, як і закони багатозначних логік, відрізняються від законів Аристотеля. Тому такі логіки називають неаристотелевими. Символічна двозначна логіка, побудована Гейтінгом у роботі «Принципи математики», належить до неаристотелевих логік, відрізняючись від аристотелевої іншою інтерпретацією імплікації.

Подібно неевклідовим геометріям неаристотелеві логіки також знайшли собі застосування. Нескінченнозначна логіка була задумана Г. Рейхенбахом як фундамент математичної теорії ймовірності. А у 1933 році Т. Швицький помітив, що багатозначні логіки можуть бути використовувані у сучасній квантовій фізиці. Багато аспектів такого використання були досліджені Г. Біркгофом і Г. Рейхенбахом. Використання інтуїціоністами логіки Гейтінга також свідчить про математичні цінності нових логік.

Для спрощення розгляду основних положень теорії багатозначних логік обмежимося трьохзначною логікою і скористаємося методом таблиць істинності. В першу чергу, відтворимо таблицю істинності для операції кон'юнкції (таблиця 1).

Таблиця 1. Таблиця істинності кон'юнкції

<table>
<tr><td colspan="2" rowspan="2">$\wedge$</td><td colspan="2">q</td></tr>
<tr><td>I</td><td>X</td></tr>
<tr><td rowspan="2">p</td><td>I</td><td>I</td><td>X</td></tr>
<tr><td>X</td><td>X</td><td>X</td></tr>
</table>

Таблиця 1 побудована таким чином: у лівому стовпці знаходяться можливі значення істинності для висловлення *p,* а у верхньому рядку — можливі значення істинності для висловлення *q*. Знаючи значення істинності вказаних висловлень, можна знайти значення їх кон'юнкції у комірці, що стоїть на перетині рядку, що відповідає значенню істинності *p*, і стовпця, відповідного значенню істинності *q*. Оскільки, за визначенням, кон'юнкція істинна в тому і тільки в тому випадку, коли обидва висловлення *p* і *q* істинні, значення *I* стоїть у лівій верхній комірці таблиці, а врешті — *X*. Зазначимо, що таблиця заповнюється на підставі одного лише визначення операції кон'юнкції.

Тепер перейдемо до трьохзначної логіки і позначимо три можливі значення істинності висловлення через *I*, «?» і *X*. Знов складемо таблицю істинності операції кон'юнкції (таблиця 2). Оскільки кон'юнкція істинна, коли обидва висловлення *p* і *q* істинні, ліва верхня комірка таблиці повинна містити значення *I*, крім того, *I* не може знаходитися ні в якій іншій комірці таблиці. Таким чином, залишається вісім комірок, в кожній з яких може бути записано або значення *X*, або значення «?». Всього одержується $2^8 = 256$ способів заповнення таблиці. Звідси маємо, що у трьохзначній логіці існує 256 різних визначень кон'юнкції.

Таблиця 2. Шаблон таблиці істинності кон'юнкції у трьохзначній логіці

<table>
<tr><td colspan="2" rowspan="2">∧</td><td colspan="3">q</td></tr>
<tr><td>I</td><td>?</td><td>X</td></tr>
<tr><td rowspan="3">p</td><td>I</td><td>I</td><td></td><td></td></tr>
<tr><td>?</td><td></td><td></td><td></td></tr>
<tr><td>X</td><td></td><td></td><td></td></tr>
</table>

В таблицях 3 і 4 наведено дві з 256 можливих таблиць істинності, операції кон'юнкції у трьохзначній логіці.

Таблиця 3. Таблиця істинності кон'юнкції у трьохзначній логіці (Лукашевич)

<table>
<tr><td colspan="2" rowspan="2">∧</td><td colspan="3">q</td></tr>
<tr><td>I</td><td>?</td><td>X</td></tr>
<tr><td rowspan="3">p</td><td>I</td><td>I</td><td>?</td><td>X</td></tr>
<tr><td>?</td><td>?</td><td>?</td><td>X</td></tr>
<tr><td>X</td><td>X</td><td>X</td><td>X</td></tr>
</table>

Таблиця 4. Таблиця істинності кон'юнкції у трьохзначній логіці (Бочар)

<table>
<tr><td colspan="2" rowspan="2">∧</td><td colspan="3">q</td></tr>
<tr><td>I</td><td>?</td><td>X</td></tr>
<tr><td rowspan="3">p</td><td>I</td><td>I</td><td>?</td><td>X</td></tr>
<tr><td>?</td><td>?</td><td>?</td><td>?</td></tr>
<tr><td>X</td><td>X</td><td>?</td><td>X</td></tr>
</table>

Таблиця істинності 3 була запропонована Лукашевичем, Постом і Россером. Вона будується відповідно до угоди, за якою значення «?» більш хибне, ніж *I*, але менш хибне, ніж *X*, а значення кон'юнкції співпадає із значенням істинності більш хибного із складових висловлень.

Відмінне від описаного визначення кон'юнкції дає Бочар (таблиця 4). За Бочаром символ «?» означає нерозв'язність, а кон'юнкція висловлень p і q вважається нерозв'язною у випадку нерозв'язності хоча б одного із складових висловлень.

Розглянемо таблицю істинності операції заперечення. У випадку заперечення єдине обмеження полягає у тому, що $\neg p$ не може бути істинним при істинності висловлення p, і $\neg p$ не може бути хибним у випадку хибності p. Вказане обмеження повністю визначає таблицю істинності для заперечення у двозначній логіці і припускає 12 можливих способів визначення заперечення у трьохзначній. Нижче наведено дві з 12 можливих для заперечення таблиць істинності операції заперечення. Таблиця істинності 5, що запропонована Постом, заснована на угоді, згідно з якою операції заперечення привласнюється наступне за порядком значення аргументу. Таблиця істинності 6 була запропонована Бочаром, Лукашевичем і Россером, які виходили з того, що $\neg(\neg p)$ повинне бути еквівалентне p.

Таблиці істинності інших логічних операцій можуть бути побудовані на підставі їх визначення за допомогою операцій кон'юнкції та заперечення. Оскільки кон'юнкція та заперечення незалежні, а решта операцій може бути через них виражена, то існує взагалі 256 × 12 = 3072 різних трьохзначних логік. Таким чином, кількість різних можливих структур багатозначної логіки надзвичайно велика.

Таблиця 5. Таблиця істинності заперечення у трьохзначній логіці (Пост)

p	$\neg p$
I	?
?	*X*
X	*I*

Таблиця 6. Таблиця істинності заперечення у трьохзначній логіці (Бочар)

p	$\neg p$
I	*X*
?	?
X	*I*

В деяких випадках при побудові багатозначних логік використовується таке поняття значення істинності.

Дійсне число $T(p)$ з інтервалу [0, 1], яке ставиться у відповідність висловленню p, називають ***значенням істинності*** висловлення p.

Значення істинності $T(p)$ можна розуміти як ймовірність того, що висловлення p істинне, причому два висловлення, що мають одне й те ж значення істинності, можна вважати логічно еквівалентними. Значення істинності $T(p) = 1$ означає істинність, а значення $T(p) = 0$ — хибність висловлення p. Заперечення $\neg p$ визначається за допомогою значення істинності таким чином: $T(\neg p) = 1 - T(p)$. В трьохзначній логіці, наприклад, як значення істинності можна прийняти числа 0, 1/2 і 1. Якщо значення істинності $T(p) = 1/2$, то значення істинності $\neg p$ також дорівнює 1/2, і p виявляється логічно еквівалентним своєму запереченню. Таке висловлення може бути назване сумнівним, причому заперечення цього висловлення також виявляється сумнівним. Зазначений підхід свідчить про наявність тісного зв'язку між багатозначними логіками і теорією ймовірностей.

Багатозначна логіка розглядає однорідні логічні функції, що визначені на множині $(0, 1, \ldots, k - 1)$, яка складається з k елементів. В силу однорідності сама функція *k-значної* логіки від n змінних приймає значення з тієї ж скінченної множини.

Функції k-значної логіки визначені і приймають значення, що входять до деякої множини $B_k = \{0, 1, \ldots, k - 1\}$, що називається ***алфавітом*** цієї ***логіки***.

Функцію k-значної логіки однозначно визначає її таблиця значень (істинності). Множину всіх функцій *k-значної* логіки позначають P_k. Кількість функцій P_k, що залежать від n змінних, дорівнює k^{k^n}. Як і у двозначній логіці, висловлення зображуються у вигляді формул k-значної логіки. Елементарні функції зображують узагальнення аналогічних функцій двозначної логіки. Розглянемо основні функції k-значної логіки.

Унарні функції

Циклічне заперечення:

$\neg x = x + 1 (\mathrm{mod}\ k)$,

де $y \bmod k$ — залишок від ділення y на k.

Таким чином, ця елементарна функція зображує узагальнення заперечення у розумінні «циклічного» зміщення значень:

$$\begin{cases} x+1, \text{ при } x \neq k-1 \\ 0, \text{ при } x = k-1 \end{cases}$$

Заперечення Лукашевича:

$N_x = k - 1 - x.$

Наведена елементарна функція N_x є іншим узагальненням операції заперечення у розумінні «дзеркального» відображення значень.

Узагальнене заперечення:

$$I^{\sigma}(x) = \begin{cases} k-1, \text{ при } x = \sigma \\ 0, \text{ при } x \neq \sigma, \text{ де } x, \sigma \in \{0,1,\ldots,k-1\} \end{cases}$$

Елементарна функція $I^{\sigma}(x)$ при $\sigma \neq k - 1$ є узагальненням деяких властивостей заперечення.

Характеристична функція:

$$J^{\sigma}(x) = \begin{cases} 1, \text{ при } x = \sigma \\ 0, \text{ при } x \neq \sigma, \text{ де } x, \sigma \in \{0,1,\ldots,k-1\} \end{cases}$$

Функція $J^{\sigma}(x)$ — характеристична функція значення а при $\sigma \neq k - 1$ зображує узагальнення операції заперечення.

Бінарні функції

1. Узагальнення кон’юнкції: $\min(x_i, x_j)$.
2. Інше узагальнення кон’юнкції: $x_i\, x_j\ (\mathrm{mod}\ k)$.
3. Узагальнення диз’юнкції: $\max(x_i, x_j)$.

Система функцій $f_1 \ldots f_n$ називається ***повною,*** якщо будь-яка функція з P_k може бути зображена у вигляді формули, що складається з цих функцій.

В k-значній логіці залишаються справедливими деякі закони двозначної, а саме: асоціативності, комутативності, дистрибутивності і т. д. Подібно до функцій двозначної логіки k-значні логічні функції можуть бути задані у вигляді таблиці. Кількість стовпців у таблиці дорівнює k^n, де n — кількість змінних, що входять до функції, а кількість функцій визначається числом k^{k^n}, яке швидко зростає із збільшенням k.

В k-значній логіці існує k констант $f_0 = 0$, $f_1 = 1$, …, $f_{k-1} = k - 1$. Серед функцій однієї змінної найбільш часто використовуваними є такі:

Характеристичні функції i-го порядку — $f_0(x)$, $f_1(x)$, $f_2(x)$, що визначені в таблиці 7.

Таблиця 7. Функції однієї змінної у трьохзначній логіці

	X			
$F(x)$	x	0	1	2
	$f_0(x)$	2	0	0
	$f_1(x)$	0	2	0
	$f_2(x)$	0	0	2
	N_x	2	1	0
	$\neg x$	1	2	0

2) Заперечення Лукашевича $N_x = k - 1 - x$.

3) Функція циклічного заперечення $\neg x = x + 1$ (mod). Таблиці істинності вказаних функцій у трьохзначній логіці будуть мати вигляд, що зображений у таблиці 5.17. Серед функцій двох змінних найбільш важливе значення мають такі:

1) *значна* диз’юнкція — $x_1 \vee x_2 = \max(x_1, x_2)$.

2) *значна* кон’юнкція — $x_1 \wedge x_2 = \min(x_1, x_2)$.

3) Функція Шеффера — Веббах — $x_1|x_2 = x_1 \vee x_2 + 1(\mathrm{mod}\ k)$.

4) Додавання за модулем $k - x_1 + x_2\ (\mathrm{mod}\ k)$.

5) Множення за модулем $k - x_1 * x_2\ (\mathrm{mod}\ k)$.

Значення даних функцій при $k = 4$ зображено в таблиці 8.

Таблиця 8. Функції двох змінних у чотиризначній логіці

x_1	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3
x_2	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
$x_1 \vee x_2$	0	1	2	3	1	1	2	3	2	2	2	3	3	3	3	3
$x_1 \wedge x_2$	0	0	0	0	0	1	1	1	0	1	2	2	0	1	2	3
$x_1 \vee x_2$+ l(mod k)	1	2	3	0	2	2	3	0	3	3	3	0	0	0	0	0
$x_1 + x_2$(mod k)	0	1	2	3	1	2	3	0	2	3	0	1	3	0	1	2
$x_1 * x_2$ (mod k)	0	0	0	0	0	1	2	3	0	2	0	2	0	3	2	1

Скориставшись поняттям характеристичних функцій для двозначного випадку, нескладно записати ДДНФ і ДКНФ у багатозначній логіці. Нагадаємо, що основну роль у ДДНФ відіграють елементарні кон’юнкції $x_1^{\sigma_1},\ldots,x_n^{\sigma_n}$, які відмінні від нуля лише на одному наборі $(\sigma_1,\ldots,\sigma_n)$. При цьому всі вони одержані з кон’юнкції, що відповідає одиничному набору, підстановкою функції від однієї змінної x^{σ}.

Контрольні запитання

1. Що розуміють під багатозначною логікою?
2. Які існують різновиди багатозначних логік?
3. Скільки різних визначень кон’юнкції існує у чотиризначній логіці?
4. Складіть таблицю істинності кон’юнкції у трьохзначній логіці.
5. В чому полягає єдине обмеження операції заперечення у багатозначній логіці?
6. Дайте визначення поняттю значення істинності висловлення.
7. Сформулюйте визначення алфавіту k-значної логіки.
8. Скільки існує різних функцій k-значної логіки від n змінних?
9. Запишіть формули основних унарних функцій k- значної логіки.
10. Назвіть найважливіші бінарні функції k-значної логіки.
11. Яка система функцій й-значної логіки називається повною?
12. Назвіть найбільш часто використовувані функції однієї змінної.
13. Складіть таблицю основних функцій двох змінних у чотиризначній логіці.

Вправи для самостійного розв’язування

1. Складіть таблицю істинності функції узагальненого заперечення у трьохзначній логіці.

2. Побудуйте таблицю істинності імплікації у трьохзначній логіці, виходячи з припущення, що істина не може імплікувати хибність. Чи можливо у даній логіці єдиним чином виразити імплікацію через заперечення та кон’юнкцію?

3. За аналогією з двохзначною логікою доведіть справедливість дистрибутивного закону у k-значній логіці.

4. Побудуйте таблиці істинності функцій однієї змінної у чотиризначній логіці.

5. Запишіть формули ДДНФ і ДКНФ у k-значній логіці.

ТЕОРІЯ АЛГОРИТМІВ. ОСНОВНІ ПОЛОЖЕННЯ ТА ОЗНАЧЕННЯ ТЕОРІЇ АЛГОРИТМІВ

Теоретичні відомості

Поняття алгоритму належить до фундаментальних понять математики, кібернетики та інформатики. Під алгоритмом розуміють чіткі інструкції про виконання в певній послідовності деякої системи операцій для рішення задач певного класу.

Звичайно, запропоноване визначення швидше роз'яснює слово "алгоритм", а не дає точного математичного визначення поняття алгоритм. Незважаючи на це, в науці довгий час використовувалися саме інтуїтивні визначення цього поняття про що свідчать і наступні приклади.

Алгоритм – чітко визначений опис способу рішення задачі у вигляді кінцевої послідовності дій.

Алгоритм – правило, що сформульоване на деякій мові та визначає процес переробки вихідних даних в необхідні результати.

Алгоритм – сукупність правил, що визначає ефективну процедуру рішення будь-якої задачі деякого заданого класу задач.

Алгоритм – вказівки, що однозначно визначають процес перетворення вихідної інформації у вигляді послідовності елементарних дискретних кроків, які дозволяють за кінцеву їх кількість отримати необхідний результат.

Основними властивостей алгоритму відносять наступні характеристики:

Детермінованість-визначення кроків алгоритму, тобто після кожного кроку або зазначається, який крок слід роботи далі, або дається команда на зупинку. Ця властивість означає, що застосування алгоритму до тих самих даних має призвести до одного і того самого результату.

Масовість – алгоритм може бути використаний для розв'язання цілого класу задач одного типу.

Результативність – виконання алгоритму має або закінчитися результатом, або інформацією про те, чому не може бути одержаний результат.

Зрозумілість – алгоритм має бути зрозумілим конкретному виконавцю, який повинен виконати кожну команду алгоритму у суворій послідовності з її призначенням.

Дискретність – можливість розбиття алгоритму на скінчену кількість етапів, причому результати попереднього етапу є вхідними для наступного.

Серед вимог, що подані до алгоритму, необхідно вирізняти наступні:

• будь-який алгоритм застосовується для початкових даних і видає результат; маючи за мету в подальшому уточнити поняття алгоритму, необхідно уточнити також і поняття даних, тобто зазначити, які вимоги задовольняють об'єкти, щоб з ними могли працювати алгоритми;

• дані та алгоритм розміщуються в пам'яті, яка вважається однорідною та дискретною, іноді навіть нескінченною;

• у якості виконавця може виступати людина, різні технічні пристрої.

Слід зазначити, що навіть при виконанні всіх зазначених вимог не всяка послідовність дій є алгоритмом.

Контрольні запитання

1. Дайте визначення поняття алгоритм.
2. Назвати основні властивості алгоритму.
3. Вимоги до алгоритмів.

АЛГОРИТМІЧНІ МОДЕЛІ. ОБЧИСЛЮВАЛЬНІ ФУНКЦІЇ

Алгоритмом прийнято називати систему обчислень, що для деякого класу задач Z із умови A дозволяє за допомогою однозначно визначеної послідовності операцій, що здійснюються “механічно”, отримати результат B. Для доведення, подамо всі початкові умови задачі Z у вигляді послідовності: $\alpha_0, \alpha_1, \alpha_2, \alpha_3, \ldots, \alpha_n$.

Якщо ж позначити через F множину номерів n тих умов αn, які алгоритм може перетворити в рішення, то результат роботи алгоритму, що здійснює перетворення $\alpha_n \to \beta_m$ однозначно визначається заданою на F числовою функцією m=f(n).

Таким чином, довільний алгоритм можна звести до обчислення значення деякої числової функції. І, навпаки, якщо для функції f існує алгоритм, який призводить до стандартного запису значення функції m=f(n), то функція f називається алгоритмічно обчислювальною або просто обчислювальною.

Як було зазначено раніше, у теорії алгоритмів, зокрема і в теорії обчислювальних функцій, прийнято конструктивний, фінітний підхід, основною рисою якого є те, що вся множина об’єктів (у нашому випадку – ункцій) будується зі скінченого числа початкових об’єктів за допомогою простих операцій та операторів, ефективна виконуваність яких досить очевидна.

При цьому слід зазначити, що у теорії обчислювальних функцій визначають множину натуральних чисел N=0,1,2,3,… і розглядають тільки числові функції, розуміючи під ними функції n змінних (n-місні функції), аргументи і значення яких належать N.

Виходячи з вищесказаного, надамо означення обчислювальної функції:

Обчислювальна функція – це числова функція $f(x_1, x_2, x_3,.., x_n)$, значення якої можна обчислювати за допомогою деякого алгоритму на підставі відомих значень аргументу.

Найпростіші функції Розглянемо клас числових функцій, що використовуються в якості базису для побудови обчислювальних функцій. O(x)=0 (нуль-функція) S(x)=x+1 (функція наступності, але не додавання одиниці) I n m (x_1, x_2, x_3,.., x_m, … x_n) = x_m (функція проекції або введення фіктивних змінних або вибору аргументу).

Оператори, що застосовуються до найпростіших функцій В якості операторів, застосування яких до базисних функцій призводить до утворення нових функцій оберемо наступні три оператори:

• оператор суперпозиції;

• оператор примітивної рекурсії;

• оператор мінімізації або найменшого кореня.

Оператор суперпозиції (Sn m) полягає у підстановці одних арифметичних функцій замість аргументів інших функцій. Нехай задана m-місна функція F(x_1, x_2, x_3,.., x_m) та m-на кількість n-місних функцій f_1(x_1, x_2, x_3,.., x_n) , f_2(x_1, x_2, x_3,.., x_n), f_3(x_1, x_2, x_3,.., x_n),…,f_m(x_1, x_2, x_3,.., x_n). Тоді говорять, що n-місна функція ϕ (x_1, x_2, x_3,.., x_n) утворилася в результаті підстановки у функцію F замість її аргументів m функцій f_1, f_2, f_3,…, f_m. Така підстановка називається суперпозицією Sn m. Sn m(F, f_1, f_2, …, f_m) = F (f_1(x_1, x_2, .., x_n) , f_2(x_1, x_2, .., x_n),…,f_m(x_1, x_2, .., x_n)) = ϕ (x_1, x_2, .., x_n)

Оператор примітивної рекурсії (Rn). Дозволяє будувати n+1-місну арифметичну функцію f(x_1, x_2, x_3,.., x_n, y) з двох заданих функцій, одна з яких є n-місною ϕ(x_1,x_2,..,x_n), а інша – n+2-місна функція ψ(x_1,x_2,..,x_n,y,z) за наступною схемою: f(x_1, x_2, x_3,.., x_n,0)=ϕ(x_1,x_2,..,x_n) f(x_1, x_2, x_3,.., x_n,y+1)= ψ(x_1,x_2,..,x_n,y, f(x_1, x_2, x_3,.., x_n, y)) Таким чином, f(x_1, x_2, x_3,.., x_n, y) = R_n (ϕ,ψ). Для правильного розуміння операції примітивної рекурсії необхідно зазначити, що будь-яку функцію від меншої кількості аргументів можна розглядати як функцію від більшої кількості аргументів. Зокрема, 95а циклювання95анн (n=0) є одномісними і відповідно: f(x)=a f(x,y+1)=ψ(x,y, f(y)), де *a* – константа.

Схеми примітивної рекурсії визначають функцію f рекурсивно не тільки через інші функції ϕ та ψ, а й через значення f у попередніх точках – значення f у точці (y+1) залежить від значення f у точці (y).

Оператор мінімізації (μy) Розглянемо обчислювальну n-місну числову функцію $f(x_1, x_2, x_3,.., x_n)$. Зафіксуємо деякі значення $x_1, x_2, x_3,.., x_{n-1}$ її перших (n-1) аргументів та розглянемо рівняння $f(x_1, x_2, x_3,.., x_{n-1}, y) = xn$ Значення виразу для заданої функції f залежить від значення x1, x2, x3,.., xn-1, xn, тому вираз дає часткову функцію змінних параметрів x1, x2, x3,.., xn-1, xn, що визначає μ-оператор.

Функція називається **примітивно-рекурсивною**, якщо вона бути утворена з найпростіших функцій за допомогою скінченого числа застосувань операторів суперпозиції Sn m та примітивної рекурсії R_n.

Функція називається **частково-рекурсивною**, якщо вона бути утворена з найпростіших функцій за допомогою скінченого числа застосувань операторів суперпозиції Sn m , примітивної рекурсії Rn та мінімізації μ_y.

Кожна стандартно задана частково-рекурсивна функція є обчислювальною за певною процедурою, яка відповідає інтуїтивному уявленню алгоритму, а з іншого боку – які б досі не будувалися класи точно визначених алгоритмів, завжди з‘ясовувалося, що числові функції, які обчислювалися за алгоритмами цих класів, були частково-рекурсивними. Тому загальноприйнятою є така наукова гіпотеза (теза Черча):

ТЕЗА ЧЕРЧА – клас алгоритмічно обчислювальних часткових числових функцій збігається з класом усіх частково-рекурсивних функцій. У формулювання цієї тези входить інтуїтивне поняття обчислювальності, тому його не можна ні спростувати, ні довести. Це факт, на користь якого свідчить багаторічна математична практика.

<u>Приклади розв’язування типових задач</u>

<u>Приклад 1</u>. Здійснюючи операцію суперпозиції функцій f(x)=0 та g(x)=x+1 отримуємо h(x)=g(f(x))=0+1=1.

<u>Приклад 2</u>. Побудуйте 2-місну функцію f(x,y)=x+y з елементарних функцій за допомогою оператора примітивної рекурсії. Функція f(x,y)=x+y визначається функцією проекції I1 1 (x) = x та функцією слідування S(x,y,z)=z+1. Таким

чином, f(x,0)= I1 1 (x) = x f(x,1)= S(x,0,x)=x+1 f(x,2)= S(x,1,x+1)=x+2 f(x,y-1)= S(x,y-2,x+y-2)=x+y-1 f(x,y)= S(x,y-1,x+y-1)=x+y або f(x,0)= I1 1 (x) = x f(x,y+1)= f(x,y)+1=S(x+y) Таким чином, f(x,y)= x+y=R1 (I1 1 (x), g(x,y,z)), де g(x,y,z)=S(z)=z+1

Приклад 3. Побудувати 2-місну функцію f(x,y) =x*y з елементарних функцій за допомогою оператора примітивної рекурсії. Fxy(3,2)=6 fxy(3,2)=Fxy(3,1)+3=6 fxy(3,1)=Fxy(3,0)+3=3 fxy(3,0)=0=O(x)

Приклад 4. Побудувати 2-місну функцію f(x,y) =xy з елементарних функцій за допомогою оператора примітивної рекурсії. Fx y (3,2)=9 fx y (3,2)=fx y (3,1)*3=9 fx y (3,1)=fx y (3,0)*3=3 fx y (3,0)=1=S(O(x)).

Приклад 5. Отримаємо за допомогою μ-оператора функцію d(x,y): d(x,y)=μz[y+z=x]=μz[S((I3 2 (x,y,z), I3 3 (x,y,z),)]= I3 1 (x,y,z). Обчислимо функція d(7,2). Для цього необхідно задати y значення 2 та встановити змінній Z послідовно значення 0,1,2.., кожного разу обчислюючи суму y+z. Як тільки вона дорівнюватиме 7, то відповідне значення прийняти за значення d(7,2).

Контрольні запитання

1. Інтуїтивне поняття алгоритму.
2. Приклади алгоритмів.
3. Властивості алгоритмів.
4. Що таке рекурсія.
5. Елементарні арифметичні функції.
6. Операції.
7. Підстановки (суперпозиції).
8. Примітивної рекурсії.
9. Мінімізації.

АЛГОРИТМІЧНІ МОДЕЛІ НА ОСНОВІ ДЕТЕРМІНОВАНИХ ПРИСТРОЇВ

Основні положення та означення фінітного комбінаторного процесу Поста Фінітний комбінаторний процес Поста (тут та надалі – машина Поста) складається із стрічки та пристрою керування (каретки).

Стрічка нескінчена та складається з комірок однакового розміру (див. 1).

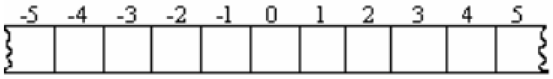


Рис. 1. Нескінчена стрічка

Послідовність, у якій розташовані комірки, відповідає послідовності, у якій розташовані натуральні числа. В кожній комірці стрічки може бути записано або символ мітки j або нічого (відповідна комірка називається або поміченою або пустою). Інформація про заповнення стрічки називається її станом.

Каретка може переміщуватися вздовж стрічки праворуч та ліворуч. Коли ж вона нерухома, то стоїть проти однієї з комірок.

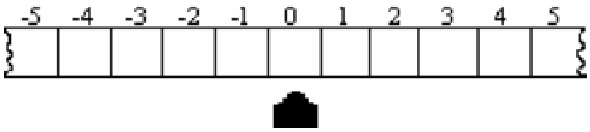


Рис. 2. Нескінчена стрічка і пристрій керування

Інформація про стан стрічки та місце розташування каретки називається станом машини Поста.

Робота машини Поста відбувається у дискретному часі. За одиницю часу каретка може виконати одну з операцій:

• зміститися праворуч;

• зміститися ліворуч;

• встановити мітку;

• знищити мітку;

• визначити наявність мітки.

Кожна програма машини Поста складається з команд.

Кожна команда програми машини Поста складається з номеру команди, операції та переходу. Наприклад, команда зміщення праворуч $i \rightarrow j$; команда зміщення ліворуч $i \leftarrow j$; команда встановлення мітки i ⮟ j; команда знищення мітки i ε j; команда зупинки і стоп або i ! (знак оклику).

Відповідно, програма машини Поста – це скінчений перелік команд, що має наступні властивості:

• на n-му місці записується команда з номером N;

• передача керування повинна відбуватися тільки до існуючого номеру команди.

Необхідні умови роботи машини Поста:

• визначеність стану машини Поста (99а циклювання99ання каретки і міток);

• наявність програми машини Поста.

Зауваження щодо роботи машини Поста:

• виконання команди встановлення/знищення мітки не призводить до переміщення каретки і можливе тільки за умови пустої / відміченої комірки;

• виконання команди передачі керування з верхнім та нижнім індексом не змінює стан машини Поста. Верхній перехід відбувається у випадку, коли комірка, яку визначає каретка, пуста, і навпаки.

Результат виконання програми машини Поста:

• в ході виконання програми машина Поста зустрічається із командою зупинки, що призводить до результативної зупинки.

• в ході виконання програми машина Поста зустрічається із не коректною командою, що призводить до без результативної зупинки.

• в ході виконання програми машина Поста не зустрічається ні з однією з вищевказаних команд , що призводить до “а циклювання”.

Зауваження: різні програми, що опрацьовують один і той же стан, можуть призводити до всіх трьох результатів, і навпаки, одна і та ж програма може давати різні результати для різних початкових станів.

Формальний опис машини Тьюрінга Машина Тьюрінга (МТ) складається з: • нескінченної стрічки, що поділена на комірки. В кожній комірці може бути записаний один із символів кінцевого алфавіту A={a_0, a_1, a_2, …, a_m}, що називається зовнішнім алфавітом (ЗА). Для кожної машини Тьюрінга можна задати власний ЗА. • керуючого пристрою, що може знаходитися в одному із внутрішнього станів Q={q_1, q_2, …, q_n}. Кількість елементів Q визначає об'єм "внутрішньої пам'яті" машини Тьюрінга. У множині Q вирізняють два спеціальні стани: початковий q_1 та кінцевий q_z (або ! – знак оклику), де z – не числовий індекс, а мнемонічна ознака кінця. Таким чином, машина Тьюрінга починає роботу в стані q_1 та, потрапивши в qz зупиняється. • каретки, що переміщуючись вздовж стрічки, може: • записувати в комірку символ зовнішнього алфавіту; • зміщуватися на комірку праворуч/ліворуч чи залишатися на місці Функціонування машини Тьюрінга можна описати так: в залежності від внутрішнього стану машини Тьюрінга (q_i) та символу зовнішнього алфавіту на стрічці (a_j) відбувається запис нового символу зовнішнього алфавіту (a/ j), зміщення каретки (d) та перехід до нового внутрішнього стану (q/ i).

Функціональна схема машини Тьюрінга Враховуючи те, що для кожної пари qi aj необхідна команда вигляду (1), програму машини Тьюрінга зручно записувати у вигляді прямокутної таблиці, стовпчики якої відповідають знакам внутрішніх станів машини Тьюрінга, а рядки – знакам зовнішнього алфавіту (див. таблиця 1.).

Таблиця 1. Функціональна схема машини Тьюрінга

A \ Q	q_1	q_2	…	q_n
a_0	$q'_i a'_j d$			
a_1				
…				
a_m				

На перетині стовпчиків та рядків записана трійка знаків. Таку таблицю називають функціональною схемою машини Тьюрінга. Конфігурація машини Тьюрінга Повним станом машини Тьюрінга або конфігурацією машини Тьюрінга, за якою можна однозначно визначити поведінку машини, називається сукупність внутрішнього стану, стану стрічки та положення каретки. Конфігурацію машини Тьюрінга надалі будемо записувати у вигляді a_1 q_i a_2 ($_2$), де q_i – поточний внутрішній стан, a_1 – слово ліворуч від каретки, a_2 – слово, що утворене символом, який визначається кареткою, та символами праворуч від нього. Наприклад, конфігурація К з внутрішнім станом, словом abcde на стрічці та кареткою під символом С, записується як abq_icde.

Стандартною початковою конфігурацією назвемо конфігурацію вигляду q_1a, тобто конфігурація, що містить початковий стан, в якому каретка розглядає крайній лівий символ слова, записаного на стрічці. Якщо машина Тьюрінга, почавши роботу з деяким словом, записаним на стрічці, прийде в заключний стан, то вона називається застосованою для цього слова. Результатом її роботи вважається слово, записане на стрічці в заключний момент. Якщо ж машина в жодний момент не прийде в заключний стан, то вона називається незастосовною до слова, і результат її роботи не визначений.

Універсальна машина Тьюрінга На сьогодні ми дотримувались точки зору, за якою різні алгоритми виконувалися в різних машинах Тьюрінга, що різнилися, звичайно, своїми функціональними схемами.

Але виникає природне запитання:

а чи можна побудувати універсальну машину Тьюрінга, яка б була здатна виконувати будь-який алгоритм, тобто була здатна виконувати роботу будьякої машини Тьюрінга?

Для відповіді на це запитання уявимо ситуацію, коли ми задали деякому виконавцю (людині) завдання, яке б полягало у демонстрації роботи машини Тьюрінга по опрацюванню деякої початкової інформації.

Звичайно, додатковою умовою буде повідомлення функціональної схеми даної машини Тьюрінга.

Ми неодноразово, складаючи програми для машини Тьюрінга, самі виступали в якості виконавців, відтворюючи алгоритм машини.

Складемо словесний опис цього алгоритму:

Вказівка1: визначити на стрічці комірку, під якою підписана буква.

Вказівка2: знайти у функціональній схемі стовпчик, який позначений буквою, що написана під початковою коміркою.

Вказівка3: у стовпчику, що визначений у вказівці 2, знайти трійку символів, що розташована на перехресті із рядком, який позначений тим символом, що вписаний у комірку.

Вказівка4: змінити символ у комірці на перший символ трійки.

Вказівка5: якщо в трійці другим символом є символ !, то зупинити роботу.

Якщо ж в трійці другим символом є символ E, то зміни символ, що підписаний під визначеною коміркою, третім символом трійки.

Вказівка7: якщо в трійці другим символом є символ L/R, то зітри символ під коміркою та ліворуч/праворуч запиши третій символ із трійки.

Вказівка 8: перейти до вказівки 1.

Але на місці виконавця може бути і сама машина Тьюрінга. А це означає, що замість словесного алгоритму за допомогою 7 вказівок, алгоритм відтворення може бути заданий у вигляді функціональної схеми.

Вихідними даними до нашого алгоритму буде функціональна схема та початкова конфігурація деякої машини Тьюрінга.

Але при цьому є два зауваження:

• безпосередня подача функціональної схеми деякої машини Тьюрінга та відповідної конфігурації на стрічку універсальної машини Тьюрінга в якості вихідної інформації неможлива! Адже функціональна схема задається двомірною таблицею, конфігурація у вигляді одномірної таблиці з підписаним нижче символом стану;

• універсальна машина Тьюрінга може оперувати тільки фіксованим скінченим зовнішнім алфавітом. Але при цьому вона повинна бути пристосована до прийому в якості функціональної схеми та конфігурації з будь-якою кількістю будь-яких символів.

Таким чином постає завдання: знайти спосіб "одномірного" подання інформації та кінцевого алгоритму.

Наведені приклади алгоритмів свідчать про те, що багато відомих алгоритмів задаються машиною Тьюрінга як точним математичним об'єктом.

Виникає питання, а чи будь-який алгоритм (у змістовному розумінні) може бути заданий деякою МТ. Відповідь на це дає гіпотеза Тьюрінга: для будь-якого алгоритму існує МТ, що його реалізує.

Очевидно, що гіпотеза Тьюрінга не є математичкою теоремою. Її обґрунтування аналогічне обґрунтуванню принципу нормалізації Маркова та гіпотези Черча. До того ж, всі розглянуті нами уточнення поняття алгоритму (нормальні алгоритми, частково рекурсивні функції та машини Тьюрінга) еквівалентні між собою.

Важливе теоретичне значення має факт існування в кожній алгоритмічній системі, в якій поняття алгоритму є математично точним, так званого універсального алгоритму, який може виконувати роботу будь-якого конкретного алгоритму.

Зокрема, можна побудувати універсальну машину Тьюрінга (УМТ), яка може виконувати роботу будь-якої конкретної МТ. Для цього на вхід УМТ потрібно подати два слова, одне з яких є кодом ФС конкретної МТ, а друге - код
її вхідного слова в деякому стандартному, наприклад, двійковому алфавіті. Із існування УМТ слідує теоретичний висновок (30-і роки ХХст.) про можливість побудови програмного автомата, який би виконував роботу будьякого алгоритму. Такими автоматами є сучасні комп'ютери.

На закінчення відмітимо, що тепер, при наявності точного поняття алгоритму питання про неіснування алгоритму розв'язку тієї чи іншої задачі масової проблеми) стає математичною теоремою. Доведення теореми про алгоритмічну нерозв'язність певного класу задач означає, що не існує і ніким, і ніякими засобами не можна побудувати єдиного алгоритму, яким розв'язуються всі задачі даного класу.

На сьогодні уже відомо багато алгоритмічно нерозв'язних проблем, зокрема, такими є:

1) проблема розпізнавання самозастосовності алгоритмів;
2) проблема розпізнавання анулювання для будь-якого алгоритму;
3) проблема розпізнавання застосовності алгоритму до того чи іншого слова;
4) комбінаторна проблема Е.Поста;
5) проблема тотожності слів для півгруп (із скінченним числом твірних елементів і скінченним числом визначальних співвідношень);
6) проблема тотожності слів для груп;
7) проблема представлення для матриць;
8) 10-а проблема Гільберта та ін.

Алгоритмічна нерозв'язність всіх вказаних проблем доводиться, виходячи із припущення про справедливість гіпотез Черча, Тьюрінга та Маркова. Існування алгоритмічно нерозв'язних проблем означає, що при пошуку алгоритму розв'язку тієї чи іншої проблеми треба мати на увазі. що такого алгоритму можливо взагалі не існує. Тому поряд із спробами побудови такого алгоритму треба одночасно прагнути довести його неіснування. Алгоритмічна нерозв'язність задач того чи іншого класу зовсім не означає неможливості розв'язати будь-яку конкретну задачу із цього класу. Мова йде про неможливість розв'язування всіх задач даного класу одним і тим же методом. Цілком можливо, що існують алгоритми для розв'язування окремих підкласів таких задач. Тому в таких випадках проблема формулюється стосовно більш вузького класу задач. Наприклад, якщо проблема тотожності слів, як відомо, є алгоритмічно нерозв'язною для класу всіх скінченно породжених груп, то ця проблема тотожності може ставитись для окремих підкласів груп, зокрема, скінченно породжених абелевих, нільпотентних, циклічних, локально-скінченних і т.д. груп. Головне тут полягає в тому, щоб відшукати найбільш широкий клас груп, для якого проблема тотожності розв'язується позитивно. Пошук максимально широких класів задач, для яких та чи інша проблема розв'язується позитивно, є однією із важливих задач розвитку сучасної математики.

<u>Приклади розв’язування типових задач</u>

1. Написати програму для машини Тьюрінга для збільшення числа n на 1. Число записане у вигляді послідовності одиниць. Каретка знаходиться десь зліва від числа (див. рис. 3).

λ	λ	λ	1	1	1	λ	λ

q_1

Рис. 3. Конфігурація машини Тьюрінга для задачі збільшення числа *n* на 1

Пояснення до рішення задачі: на першому такті визначається символ у комірці. Якщо це знак λ, то відбувається зміщення праворуч (символ в комірці та стан машини Тьюрінга залишаються незмінними). У випадку знаходження лівої позиції заданого числа (каретка зустріла в комірці перший символ 1), відбувається зміщення каретки ліворуч та перехід до стану q2 (символ в комірці залишається без змін). Зустрівшись зі символом λ у стані q2 , машина Тьюрінга змінює його на символ 1 та припиняє роботу (переходить до стану qz).

2. Побудувати МТ (скласти її функціональну схему), яка реалізує функцію слідування S(n)=n+1, де n - натуральне число, записане в алфавіті {I}.

Зовнішнім алфавітом цієї машини буде алфавіт $A=\{I\}$. Нехай початкова конфігурація машини має вигляд: $\Lambda g_0 I^n \Lambda$ тобто в початковому стані g_0 ЧЕ машини оглядає комірку, в якій записана перша зліва одиниця вхідного числа.

Змістовно можна запропонувати один з таких алгоритмів: одиниця залишається без зміни, ЧЕ зсувається вправо до тих пір, поки не зустріне першу справа пусту клітку, в яку потрібно записати одиницю. Результат отримано, машину треба зупинити.

Відповідна функціональна схема має вигляд: машину треба зупинити.

Q \ A	g_0
I	$I\Pi g_0$
Λ	IН!

Будемо вважати, що МТ має стандартний початок роботи, якщо перед початком роботи ЧЕ машини оглядає першу зліва пусту клітку, що передує вхідному слову, і стандартний кінець роботи - після закінчення роботи машина повертається в стандартний початок. Тоді функціональна схема, що реалізує функцію слідування, запишеться так:

Q \ A	g_0	g_1	g_2
I		$I\Pi g_1$	IЛ g_2
Λ	Πg_1	IЛ g_2	ΛН!

3. Побудувати МТ, що подвоює натуральні числа, записані в алфавіті $\{I\}$. Нехай початкова конфігурація така: $\Lambda I^{n-1} g_0 I \Lambda$ Змістовно кожну паличку будемо замінювати по черзі на зірочку, дописуючи в кінці слова ще одну зірочку. В результаті ми одержимо 2n зірочок. На останньому етапі кожну зірочку замінимо на паличку і зупинимось. ФС шуканої МГ матиме такий вигляд:

Q \ A	g_0	g_1	g_2
I	$*\Pi g_1$		IН g_0
$*$	$I\Pi g_0$	$*\Pi g_1$	$*$Л g_2
Λ		$*$Л g_2	!Н

Контрольні запитання

1. Дайте визначення машини Тьюрінга.
2. Назвіть основні складові елементи МТ?

3. Як визначається команда МТ.

4. Що таке функціональна схема МТ?

5. Що таке конфігурація МТ?

6. Що таке стандартна початкова конфігурація МТ? Фінальна конфігурація МТ?

7. Як змінюється конфігурація МТ при виконанні команди МТ відповідного типу?

8. Дайте визначення еквівалентних МТ.

9. Що такє МТ-обчислювана функція?

10.Сформулюйте гіпотезу Тьюрінга?

<u>Вправи для самостійного розв'язування</u>

1. Перевірити, що МТ, яка працює за наведеною нижче ФС, залишає незмінним усяке парне число $2n$, а всяке непарне число $2n+1$ перетворює в число $2n$. Числа на стрічці записуються в алфавіті $\{\mathrm{I}\}$. Початкова конфігурація: $g_0\Lambda I^n\Lambda$. Функціональна схема:

A \ Q	g_0	g_1	g_2
I	$\text{IП}\,g_1$	$\text{IП}\,g_0$	$\Lambda\,\text{Н!}$
Λ	$\Lambda\,\text{Н!}$	$\Lambda\,\text{Л}\,g_2$	

2. Побудувати МТ, яка перетворює кожне натуральне число n, записане в алфавіті $\{\mathrm{I}\}$, в число $n+3$.

3. Побудувати МТ, яка б перетворювала кожне натуральне число n, записане в алфавіті $\{\mathrm{I}\}$, в число $n+k$, де k - фіксоване натуральне число, записане в тому ж алфавіті.

4. Побудувати МТ, яка додавала б раціональні дроби з однаковими знаменниками.

5. Нехай в алфавіті А = $\{a,b\}$ задані слова P і Q . Скласти ФС МТ, яка слово $\alpha = P * Q$ переробляє в слово P.

6. Побудувати МТ, яка перетворює слово $1\underbrace{100\ldots01}_{k\ \text{разів}}$ в слово $\underbrace{100\ldots01}_{k\ \text{разів}}1$ разів (перенесення крайньої лівої одиниці в кінець слова).

ТЕОРІЯ НОРМАЛЬНИХ АЛГОРИТМІВ МАРКОВА

Основні положення та означення теорії нормальних алгоритмів Теорія нормальних алгоритмів була розроблена радянським математиком А.А. Марковим (1903 р. – 1979 р.) наприкінці 40-х – на початку 50-х років XX ст. і є ще однією алгоритмічною моделлю для уточнення поняття алгоритму.

Алгорифмом (запропоноване Марковим поняття для позначення алгоритму) в теорії нормальних алгорифмів є правила по перетворенню слів в довільному алфавіті. Нормальні алгорифми оперують словами скінченої довжини, перетворюючи їх одне в одне за допомогою підстановок, тобто змін однієї частини слова на іншу.

Цей процес нагадує процедуру виконання завдання типу: “Шляхом поступової зміни літер слово “слон” перетворити на “муху”.

Поняття алфавіту нормального алгоритму Алфавітом (як і випадку будь-якої формальної системи) називається будьяка скінчена множина деяких символів.

Будь-яка скінчена послідовність N літер деякого алфавіту – це слова завдовжки N у цьому алфавіту. Наприклад, в алфавіту A з трьох літер {a,b,c} словами є послідовності a,b,c,ab,bac,aacbaccc.

Порожнє слово, що не містить жодного символу, позначається як λ.

Якщо слово α є частиною слова β, то кажуть, що слово α входить у слово β.

Наприклад, у слові d = *abcbcbabaa* є 4 підслова a, 2 підслова *bcb*, одне слово cba.

Поняття про підстановку.

Підстановкою називається операція над словами, що задається за допомогою впорядкованої пари (α, β) та полягає у наступному: у довільному слові S знаходять перше входження слова α та, не змінюючи інших частин слова S, змінюємо в ньому це входження словом β.

Отримане слово називають результатом підстановки застосування підстановки (α, β) до слова S.

Якщо ж першого входження α в слово β немає (і відповідно немає ні одного входження α в S), то вважається, що підстановка (α, β) не застосована до слова S.

Для позначення підстановки (α, β) надалі будемо використовувати запис $\alpha \to \beta$, який називається формулою підстановки (α, β). Деякі підстановки будемо вважати заключними ($\alpha \to * \beta$ або $\alpha \to \beta$!).

Схеми орієнтованих підстановок Впорядкований скінчений перелік формул підстановок в алфавіті А називається схемою орієнтованих підстановок або схемою нормальних алгорифмів. Дана схема і визначає алгоритм перетворення слів, який називається нормальним алгорифмом Маркова.

$$\begin{cases} \alpha_1 \to \beta_1 // ! \\ \alpha_2 \to \beta_2 // ! \\ \dots\dots\dots\dots \\ \alpha_n \to \beta_n // ! \end{cases}$$

Означення нормального алгоритму Маркова

Нехай задано алфавіт А і зафіксовано впорядковану (задану в певному порядку) систему орієнтованих підстановок P. Виходячи з довільного слова α в алфавіті А розглядаються підстановки в тому порядку, в якому їх задано.

Перша підстановка, що зустрілася, з лівою частиною, яка є підсловом α, використовується для перетворення α, в яке замість першого входження лівої частини підстановки підставляється її права частина, внаслідок чого утворюється слово α_1. Далі, виходячи з слова α_1, процес повторюється, поки він не зупиниться.

Ознаками зупинки процесу перетворення слова α є два випадки:

• коли утворюється таке слово αn, що жодне з лівих частин допустимих підстановок не є його словами;

• коли при утворенні слова αn використано останню підстановку (підстановку із знаком !).

<u>Приклади розв’язування типових задач</u>

Приклад 1. Нехай задано алфавіт A={*а,б,в,г,...я*} і систему орієнтованих підстановок P: {я→у, л→у, с→м, в→б, р→т, т→р!, о→х, н→а} застосуємо алгорифм до слова “слон” та прослідкуємо перетворення: слон→суон→муон→мухн→муха застосуємо алгорифм до слова “ветер” та прослідкуємо перетворення: ветер→бетер→бетет→берет.

Приклад 2. Нехай задано алфавіт A={1,+} і систему орієнтованих підстановок P={+→λ, 1→1}.

Слово 1+11+1111+1 алгорифм перетворює наступним чином:

1+11+1111+1

111+1111+1

1111111+1

11111111

Еквівалентним є алгорифм із P={1+→+1,+1→1, 1→1}.

Приклад 3. Нехай задано алфавіт A={1,0} і систему орієнтованих підстановок P={1→λ, 0→0}.

Слово 0101 алгорифм перетворює в слово 00.

Приклад 4. Нехай задано алфавіт A={0,1,2,3,4,5,6,7,8,9} і систему орієнтованих підстановок P { 0b→1!, 1b→2!, 2b→3!, 3b→4!, 4b→5!, 5b→6!, 6b→7!, 7b→8!, 8b→9!, 9b→b0, b→1!, a0→0a, a1→1a, a2→2a, a3→3a, a4→4a, a5→5a, a6→6a, a7→7a, a8→8a, a9→9a, 0a→0b, 1a→1b, 2a→2b, 3a→3b, 4a→4b, 5a→5b, 6a→6b, 7a→7b, 8a→8b, 9a→9b, λ→a}.

Застосуємо алгорифм до слова “499” та прослідкуємо перетворення: 499→a499→4a99→49a9→499a→499b→49b0→4b00→500 Як видно із прикладу, даний алгорифм виконує збільшення числа на 1.

<u>Контрольні запитання</u>

1. Алгоритмічна система нормальні алгоритми Маркова (НАМ).
2. Поняття функції обчислюваної за Марковим.
3. Побудова НАМ для нуль – функції, функції безпосереднього слідування, селекторної функції.

4. Поняття еквівалентних відносно алфавіту А нормальних алгоритмів Маркова
5. Композиція НАМ. Приклад.
6. Розгалуження двох НАМ під керуванням третього. Приклад.
7. З'єднання НА. Приклад.
8. Повторення одного НА під керування іншого. Приклад.

<u>Вправи для самостійного розв'язування</u>

1. Нормальний алгоритм U_1 в алфавіті A = {a,b} заданий схемою:

   $$\begin{cases} bab \to aa \\ aa \to b \\ bb \to a \end{cases}$$

   Знайти результат дії цього алгоритму на слово $\alpha = abaaabb$.
2. В алфавіті A = {a,b, c} нормальний алгоритм U2 заданий такою системою підстановок:

   $$\begin{cases} ba \to ab \\ ca \to ac \\ cb \to bc \end{cases}$$

   Знайти U_2 (acbcab).
3. Нехай натуральні числа записуються в алфавіті {I}. Вхідне слово має вигляд m∗ n=III∗ IIII, тобто записане в розширеному алфавіті {I, ∗ }. Нормальний алгоритм U3 заданий схемою:

   $$\begin{cases} * I \to I * \\ * \to \wedge \end{cases}$$

   Знайти U_3 (III∗IIII)
4. Побудувати нормальний алгоритм U_4, який перетворює будь-яке слово α в алфавіті A {*a*,*b*, *c*} в слово *abc*, тобто $U_4(\alpha) = abc$.

АЛГОРИТМІЧНО НЕРОЗВ'ЯЗУВАНІ ПРОБЛЕМИ

Алгоритмічно нерозв'язувані проблеми – не невдача, а науковий факт.

Знання можливої алгоритмічної нерозв'язуваності має бути таким самим елементом наукової культури, як для фізика знання про неможливість створення вічного двигуна.

Якщо ж важливо мати справу з розв'язуваною задачею, то потрібно чітко уявляти дві обставини:

• відсутність загального алгоритму, який вирішує певну проблему не означає, що в кожному конкретному випадку не можна досягти успіху;

• поява нерозв'язаності – це, зазвичай, результат надмірної загальності задачі.

Теорема М.Райса.

Теорема Райса є однією з найбільш загальних теорем теорії алгоритмів, що пояснює природу багатьох проблем в практиці програмування.

Для пояснення теореми дамо деякі визначення. Нехай існує деяка множина А натуральних чисел.

Характерною функцією множини А будемо називати функцію

$$\chi_A(x) = \begin{cases} 1, & якщо x \in A \\ 0, & якщо x \notin A \end{cases}$$

Множина А називається рекурсивною, якщо її характерна функція рекурсивна.

Наведемо змістовне формулювання теореми Райса.

Нехай Q – деяка властивість одномісних частково-рекурсивних функцій.

Властивість Q називається нетривіальною, якщо є функції, яким характерна дана властивість Q, і яким вона не властива.

Враховуючи те, що частково-рекурсивні функції можна задати програмою їх обчислення, виникає питання: чи можливо за програмою визначити, чи має відповідна функція певну нетривіальну властивість?

У відповідності до тез Черча і Тьюрінга задача є алгоритмічно розв'язуваною тоді і тільки тоді, коли існує деяка машина Тьюрінга M_0, що вирішує дану задачу. Нехай необхідно визначити, чи характерна функції $\phi_м(x)$, що реалізується машиною М, властивістю Q. На стрічці машини M_0 повинна бути записана інформація про програму машини М. Цю інформацію задамо у вигляді шифру Ш(М). Результатом роботи машини М0 повинна бути відповідь "так" чи "ні", в залежності від того, характерна функції фм(х) властивість Q чи ні. Першому випадку домовимося співставляти символ 1, другому – 0. Таким чином, машина Тьюрінга M_0 розпізнає властивість Q одномісних частковорекурсивних функцій, якщо конфігурацію q_1Ш(М) вона перетворює в $q_1 1$, якщо функції $\phi_м(x)$ характерна властивість Q, та в $q_1 0$ в іншому випадку.

Із сказаного робимо наступний висновок: якою б не було нетривіальна властивість Q одномісних частково-рекурсивних функцій, задача розпізнання цієї властивості алгоритмічно нерозв'язувана, тобто не існує машини M_0, яка вирішує дану задачу (теорема Райса). Для доказу даного висновку покажемо, о з існування машини M_0, яка розпізнає властивість Q, витікає рекурсивність множини номерів Fq функцій, яким властивість Q також характерна.

За наявності машини M_0 обчислення функції (n) F_q ϕN може бути здійснено наступним чином. Спочатку запис числа n переводиться в шифр Ш(M_n) машини з номером n, потім застосовується машина M_0, яка видає 1, якщо функція (n) Fq ϕN =$\phi_м(x)$ має властивість Q, і 0 – в іншому випадку. В результаті значення (n) Fq ϕN обчислюється в алфавіті {0,1}. Потім 0 перетворюється в | , в 1 – в ||.

З обчислювальності функції (n) Fq ϕN витікає її рекурсивність. Але в силу нетривіальності властивості Q множина Fq не пуста і відмінна від сукупності всіх одномісних частково-рекурсивних функцій, ось чому у відповідності до теореми Райса функція (n) Fq ϕN не може бути рекурсивна. Отримане протиріччя доводить теорему.

Аналогічні результати можуть бути встановлені і для інших програм, за допомогою яких можна подати частково-рекурсивні функції, зокрема і для програм алгоритмічних мов програмування. Звідси витікає нерозв'язуванність багатьох програм, що пов'язані з програмуванням.

Наприклад, якщо є деяка програма, то за її допомогою неможливо визначити функцію, яку дана програма реалізує. За двома програмами неможливо встановити, чи реалізують вони одну і ту ж функцію, а це призводить до нерозв'язуванності багатьох задач, що пов'язані з еквівалентними перетвореннями і мінімізацією програм.

Приклади алгоритмічно нерозв'язуваних проблем.

Проблема самозастосованості. Проблема самозастосованості одна з найактуальніших алгоритмічно нерозв'язуваних проблем.

Зміст проблеми в наступному. Будемо розглядати машини Тьюрінга, у зовнішньому алфавіті яких присутні, поряд з іншими, символи 1 та 0.Нехай на стрічку машини М записаний її шифр Ш(М) і машина запущена в початковому стані q1. Якщо після деякого скінченого числа кроків машина М прийде в заключний стан, то таку машину будемо називати самозастосованою, в іншому випадку – несамозастосовано. Проблема самозастосованості полягає в тому, щоб на основі машини М визначити, чи є вона самозастосованою. Машина

Тьюрінга М0 вирішує проблему самозастосованості, якщо для будь-якої машини М конфігурацію q_1Ш(М) вона перетворює в $q_1$1, якщо М самозастосована, і в q10 – якщо несамозастосована.

Звідси, проблема самозастосованості алгоритмічно нерозв'язувана, тобто не існує машини Тьюрінга, щоб вирішувала дану задачу.

Для доведення уявимо протилежне – машина М, що вирішує проблему самозастосованості існує. На її основі побудуємо нову машину М. Для цього стан q_0 зробимо не заключними, та введемо новий заключний стан q_0 / та додамо до програми M_0 дві нові команди: q01→q01E q00→q / 00E.

Машина М застосована до шифрів несамозастосованих машин і не застосована до шифрів самозастосованих машин. Дійсно, якщо деяка машина М несамозастосована, то на початку М, працюючи так як M_0, перейде в конфігурацію $q_0$0, а потім зупиниться у відповідності до команди $q_0$0→q/00E.

Якщо ж М само застосована, то М перейде в конфігурацію $q_0 1$ і ця конфігурація буде повторюватися нескінчено у відповідності до команди $q_0 1 \rightarrow q_0 1E$.

Сама машина М/ є або само застосованою або несамозастосованою. У першому випадку вона застосована до власного шифру, тобто до шифру само застосованих машин, що неможливо з побудови М. В іншому випадку вона не застосована до власного шифру, тобто до шифру несамозастосованих машин, о також неможливо. Зазначене протиріччя виникло із припущення існування машини M_0, що вирішує проблему самозастосованості.

Проблема зупинки

Серед загальних вимог до алгоритмів відзначалася вимога результативності.

Найрадикальнішим формулюванням тут була б вимога, щоб за будь-яким алгоритмом А і даними α можна було б визначити, чи призведе робота А з початковими аргументами α до результату чи ні? Іншими словами, треба побудувати такий алгоритм В, щоб В(А,α)=I, якщо А(α) дає результат, та В(А,α)=Х, в іншому випадку.

З урахуванням тези Тьюрінга цю задачу можна сформулювати як задачу про побудову машини Тьюрінга: побудувати машину T_0 таку, що для будь-якої машини Тьюрінга Т і будь-яких початкових даних для машини $T_0(\xi T, \alpha)$=I, якщо машина Т(α) зупиняється, та $T_0(\xi т, \alpha)$=X, якщо вона не зупиняється.

Ця задача називається проблемою зупинки, а її формулювання нагадує задачу побудови універсальної машини Тьюрінга.

Контрольні запитання

1. Поняття про алгоритмічно розв'язувані проблеми та алгоритмічно не розв'язувані проблеми
2. Теорема М.Райса.
3. Проблема самозастосованості.
4. Проблема зупинки.

Література

1. Бардачов Ю.М. Дискретна математика : Підручник для студентів вищ. техніч. навч. Закладів / Ю.М.Бардачов, Н.А.Соколова, В.Є.Ходаков. – К.: Вища школа, 2002. – 286 с.

2. Борисенко О.А. Лекції з дискретної математики (множини і логіка): Навч. посібник для студ. вищ. навч. Закладів / О.А.Борисенко. – [3-є вид., випр. і доп.]. – Суми: Університет. кн., 2002. – 175 с.

3. Капітонова Ю.В. Основи дискретної математики / Ю.В.Капітонова, С.Л. Кривий, О.А.Летичевський та ін. – К. : Наук. думка, 2002. – 579 с

4. Хромой Я.В. Математична логика: Посібник / Я.В.Хромой. – Київ: Вища школа, 1983. – 208 с.

5. Андрійчук В.І. Вступ до дискретної математики : Навч. посіб. для студ. вищ. навч. Закладів / В.І.Андрійчук, М.Я.Комарницький, Ю.Б.Іщук. – К. : Центр навч. літ., 2004. – 254с.

6. Лиман Ф.М. Математична логіка і теорія алгоритмів: Навч.посібник для студ.фіз.-мат.спец. пед.ін-тів / Ф.М. Лиман. – К.: 1994. – 176с.

7. Лісова Т.В. Математична логіка та теорія алгоритмів. [практикум] - Ніжин: Вид-ць ПП Лисенко М.М., 2011 - 116 с.

8. Буренніков Ю.А., Дерібо О.В. Тестовий контроль знань студентів, як засіб підвищення ефективності навчального процесу // Вісник ВПІ. - 1994. - №2. - С. 81-84.

9. Дубів О.В., Нелюбов В.О. Методичні рекомендації по розробці тестових завдань для автоматизованого контролю знань студентів. - Ужгород: ЗакДУ, 2007. - 28 с.

10. Душенков В.Д., Панасик Т.І. Деякі аспекти методики складання тестових завдань // Організація навчально-виховного процесу. - 2006. - №8. - с. 104-109.

11.Завало С. Т, Костарчук В.М., Хацет Б. І, Алгебра і теорія чисел. ч. 2, - К.:Вища школа, 1976. - 408 с.

12. Завало С.Т., Левіщенко С.С., Пилаєв В.В., Рокицький І.О. Алгебра і теорія чисел. Практикум. Ч. 1. - К.: Вища школа, 1983. - 232 с.

13. Кліменко В.М., Дупляк В.Д., Шиліна О.П. Об’єктивний контроль знань студентів // Вісник ВПІ. - 1994. - №2. - С. 87-88.

14. Коломієць М.П., Молодова Л.В. Словник іншомовних слів. - К.: Освіта, 1998. - 190 с.

15. Лосєва Н.М. Тестування в умовах багатоступеневої підготовки фахівців у вищій школі // Освіта і управління. - 2002. - №4. - с. 150 - 156.

16. Лузіна М.О., Голуб Г.Г., Возна А.М. Система комплексної діагностики знань студентів. Навчальний посібник. - Львів: Львівський банківський інститут НБУ, 2002. - 38 с.

17. Малихін А. Тестовий контроль і підвищення якості освіти у вищій педагогічній школі // Рідна школа. - 2006. - Червень. - С. 9-11.

18. Методика навчання і наукових досліджень у вищій школі. Навч.посіб / За ред. С.У. Гончаренко, П.М. Олійника. - К.: Вища школа, 2003. - 323 с.

19. Окунев Л.Я., Краткий курс теории чисел, Учебное пособие для пединститутов, М., 1956. - 240 с.

20. Практикум педагогічної майстерності: Навчальний посібник /Кол. автор.: Сергеєва Л.М., Молчанова А.О., Пащенко О.В. та ін./ За ред. В.В. Олійника - К.: ТОВ «Етіс Плюс», 2008. - 184 с.

21. Сметанський М.І. Контроль за навчально-пізнавальною діяльністю студентів: проблеми, шляхи розв’язання // Вища школа. - 2004. - №4. - С. 63 - 68.

22. Сушкевич А.К., Теорія чисел. Видавництво Харківського Державного Університета Імені А.М. Горького, Х., 1954.

23. Фадеев Д.К., Сочинський И.С. Сборник задач по высшей алгебре. - М.: Наука, 1977. - 288 с.

24. Євладенко В.М., Халецька З.П., Нарадовий В.В. Математична логіка та теорія алгоритмів. - К.:Код, 2009.-116 с.

25. Головко Н.М. Приклади застосування методу резолюцій // Наукові записки. - Випуск 6. - Кіровоград: РВВ КДПУ ім. В. Винниченка, 2013. - с. 12-15
26. Шкільняк С.С. Математична логіка: приклади і задачі. / С.С. Шкільняк– Київ: ВПЦ "Київський університет", 2002. – 56 с.
27. Яворський Б.І. Теорія алгоритмів / Конспект лекцій. / Б.І. Яворський - Тернопіль: ТДТУ імені Івана Пулюя, 2000. - 32 с.