МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
МИКОЛАЇВСЬКИЙ НАЦІОНАЛЬНИЙ АГРАРНИЙ УНІВЕРСИТЕТ

КАФЕДРА ЕКОНОМІЧНОЇ КІБЕРНЕТИКИ І МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ

МАТЕМАТИЧНА ЛОГІКА ТА ТЕОРІЯ АЛГОРИТМІВ

курс лекцій для здобувачів першого (бакалаврського) рівня
вищої освіти ОПП «Комп'ютерні науки»
за спеціальністю 122 «Комп'ютерні науки»
денної форми навчання

Миколаїв - 2021

УДК 510.6
М33

Друкується за рішенням науково-методичною комісією факультету менеджменту Миколаївського національного університету від 18.09.2021 року протокол № 9.

Укладачі:

О. В. Шебаніна – д-р екон. наук, професор, професор кафедри економічної кібернетики і математичного моделювання, Миколаївський національний аграрний університет;

В. П. Клочан – канд. екон. наук, доцент, завідувач кафедри економічної кібернетики і математичного моделювання, Миколаївський національний аграрний університет;

І. В. Клочан – д-р екон. наук, доцент, доцент кафедри економічної кібернетики і математичного моделювання, Миколаївський національний аграрний університет;

С. І. Тищенко – канд. пед. наук, доцент, доцент кафедри економічної кібернетики і математичного моделювання, Миколаївський національний аграрний університет;

А. М. Могильницька – канд. фіз.-мат. наук, доцент кафедри економічної кібернетики і математичного моделювання, Миколаївський національний аграрний університет;

В. О. Крайній – канд. екон. наук, доцент кафедри економічної кібернетики і математичного моделювання, Миколаївський національний аграрний університет;

І. І. Хилько – старший викладач кафедри економічної кібернетики і математичного моделювання, Миколаївський національний аграрний університет.

Рецензенти:

Стройко Т.В. - д-р. екон. наук, професор, завідувач кафедри економіки та менеджменту, Миколаївський національний університет імені В.О. Сухомлинського

Кравчук Л. С. - канд. екон. наук, доцент кафедри менеджменту та маркетингу, Миколаївський національний аграрний університет

ЗМІСТ

ВСТУП

Дисципліна «**Математична логіка**» є базовою нормативною дисципліною спеціальності 122 "Комп’ютерні науки", що читається у першому семестрі у обсязі 3 кредитів, у тому числі 60 годин аудиторних занять, з них 30 годин лекцій, 30 годин практичних занять і 30 годин самостійної роботи. Закінчується курс іспитом у 1 семестрі.

Метою і завданням навчальної дисципліни "Математична логіка" є засвоєння базових знань з основ математичної логіки. У загально світоглядному аспекті, поняття і методи математичної логіки необхідні для обґрунтування правильності тих чи інших способів отримання істинного знання. В прикладному аспекті, апарат математичної логіки необхідний для адекватного моделювання різноманітних предметних областей, створення сучасних програмних та інформаційних систем.

Предмет навчальної дисципліни "Математична логіка" включає в себе вивчення базових понять математичної логіки, розгляд семантичних моделей логіки та формально-аксіоматичних логічних систем.

Вимоги до знань та вмінь. Для засвоєння курсу необхідні знання основ елементарної математики, дискретної математики та алгебри. Здобувачі вищої освіти повинен знати основи теорії множин, булеві функції, основи загальної алгебри.

Місце в структурно-логічній схемі спеціальності. Нормативна навчальна дисципліна "Математична логіка" є складовою циклу професійної підготовки фахівців освітньо-кваліфікаційного рівня "Молодший бакалавр". Курс математичної логіки потрібен для подальшого вивчення нормативних дисциплін "Бази даних та інформаційні системи", "Інтелектуальні системи", "Теорія програмування", "Теорія обчислень", "Штучний інтелект, "Інформаційні технології", низки спецкурсів відповідного напряму.

Тема 1. Алгебра висловлень

1.1. Етапи становлення математичної логіки і теорії алгоритмів

Термін ”логіка” – наука про способи доведень й спростувань – походить від грецького $\lambda o \gamma o \xi$ (логос), що означає ”слово”, ”поняття”, ”смисл”. Поняття ”традиційна або формальна логіка” характеризує науку, яка бере свій початок від Аристотеля, котра вивчає форми й закони мислення, а також методи, за допомогою яких люди в дійсності роблять висновки, встановлюють зв’язок логічних форм з мовою. Поява логічних форм і категорій, формування законів мислення – це результат загальнолюдської практики. Довготривала практична діяльність людини в процесі пізнання оточуючої дійсності викристалізувала фігури й закони логіки. Логіка вивчає те загальне, що зв’язує міркування в їх русі до пізнання істини. Вона є наукою про закони і форми правильного міркування. Логіка вивчає форми міркувань, відволікаючись від їх конкретного змісту; установлює що із чого випливає.

Математична логіка, яку ще називають символічною або теоретичною логікою, виросла із логіки традиційної, але склала значне її розширення. З одного боки, ця наука застосувала математичні методи для вивчення загальних структур (форм) правильного мислення й тим самим сформувалася як розділ математики, з іншого – математична логіка зробила предметом свого вивчення процес доведення математичних теорем і самі математичні теореми. Математична логіка, являючись більш загальною й більш абстрактною, ніж традиційна формальна логіка, в той же час є більш конкретною, дає можливість розв’язувати велику кількість конкретних задач, нерозв’язних засобами традиційної формальної логіки. Апарат числень і формальних систем в математичній логіці значно більш досконалий, ніж апарат традиційної логіки. Тому він може застосовуватися для розв’язання таких складних задач, які недоступні для класичної логіки.

Засновником логіки як науки є Аристотель (384–322 рр. до н.е.). Він вперше розробив теорію дедукції, тобто теорію логічного виведення. Аристотель звернув увагу на те, що в міркуваннях ми із одних тверджень виводимо інші, виходячи не із конкретного змісту тверджень, а із визначеного зв’язку між їх формами й структурами. Протягом багатьох віків різними філософами удосконалювалася й змінювалася логіка Аристотеля. Це був перший (до-математичний) етап розвитку формальної логіки. другий етап пов’язаний з застосуванням в логіці математичних методів, початок якому поклав німецький філософ і математик Г. Лейбніц (1646–1716). Він намагався побудувати універсальну мову, за допомогою якої можна було б розв’язувати суперечки між людьми, а потім і взагалі всі ”ідеї замінити численнями”. Важливий період становлення математичної логіки починається з робіт англійського математика й логіка Джорджа Буля (1815–1864) ”Математичний аналіз логіки” (1847) та ”Дослідження законів мислення” (1854). Ним була побудована алгебра логіки. В цей період вона сформувалася як алгебра висловлювань й була значно розвинута в роботах шотландського логіка А. де Моргана (1806–1871), англійського логіка У. Джевонса (1835–1882), американського логіка Ч. Пірса (1839–1914), німецького алгебраїста і логіка Е. Шрьодера (1841–1902), російського математика й логіка П.С. Порецького (1846–1907). Утворення алгебри логіки стало заключним етапом у розвитку формальної логіки: алгебра логіки поставила й розв'язала в самому загальному вигляді ті задачі, які розглядалися в Аристотелевій логіці.

Значним поштовхом до нового періоду розвитку математичної логіки стала побудова в першій половині XIX ст. російським математиком М.І. Лобачевским (1792–1856) і незалежно від нього угорським математиком Я. Бояі (1802–1860) неевклідової геометрії. Крім того, створення аналізу нескінченно малих привело до необхідності обґрунтування числа як фундаментального поняття всієї математики. Завершили картину парадокси, виявлені в кінці XIX ст. в теорії множин: вони показали, що труднощі обґрунтування математики є труднощами логічного й методологічного характеру. В ході розвитку математичної логіки сформувалися три напрямки обґрунтування математики.

Засновником одного із напрямків – *логіцизму* – став німецький математик і логік Г. Фреге (1848–1925). Він намагався всю математику обґрунтувати через логіку. Ним і незалежно від нього Ч. Пірсом були введені в мову логіки предикати, предметні змінні й квантори, що дало можливість застосувати цю мову до питань обґрунтування математики. Задачу аксіоматичної побудови арифметики ставив перед собою італійський математик Дж. Пеано (1858–1932). Зведення чистої математики до логіки продовжили англійські математики Б. Рассел (1872–1970) і А. Уайтхед (1861–1947). Даний напрямок не увінчався успіхом, проте був створений багатий логічний апарат, без якого математична логіка не змогла б оформитися як повноцінна математична дисципліна.

Німецький математик Д. Гільберт (1862–1943) запропонував свій шлях подолання труднощів в основах математики, базуючись на застосуванні аксіоматичного методу, за допомогою якого всі математичні твердження записуються у вигляді логічних формул, деякі з яких виділяються в значенні аксіом, а інші із них виводяться. Цей напрямок отримав назву *формалізм*. Відкриття в 1930–1931рр. австрійським математиком К. Геделем (1906–1978) неповноти формалізованої арифметики показало обмеженість гільбертовської програми обґрунтування математики. Проте роботи Гільберта і його послідовників привели до глибокої розробки аксіоматичного методу й усвідомленню його фундаментальної ролі в математиці.

Представники напрямку, який виник на початку XX ст. у працях готландського математика Л. Брауєра (1881–1966) й одержав назву інтуїціонізму, запропонували відмовитися від розгляду нескінченних множин як завершених сукупностей, а також від закону виключення третього. Ними признавалися лише ті доведення, які конструктивно будували той чи інший об'єкт, і не признавалися чисті доведення існування.

XX ст. – час великого розвитку математичної логіки, формування багатьох її нових розділів: побудовані різні аксіоматичні теорії множин; на базі математичної логіки сформована теорія алгоритмів, за допомогою якої були вироблені декілька формалізацій поняття алгоритму. Методи теорії алгоритмів стали проникати в інші розділи математичної логіки і в суміжні математичні дисципліни. В інші розділи математики стала проникати й сама математична логіка. Прикладом може бути теорія моделей, котра утворилася на межі сучасної алгебри й логіки. Значний вклад у розвиток математичної логіки внесли математики І. І. Жегалкін, А.М. Колмогоров, П.С. Новіков, А.А. Макаров, А.І. Мальцев, С.А. Яновська . Ідеї й методи математичної логіки глибоко проникли в процес конструювання й створення ЕОМ, в програмування, інформатику, обчислювальну математику, структурну лінгвістику. Ці методи використовуються як при фізичному конструюванні й створенні комп'ютерів, так і при створенні математичного забезпечення до них. В основі багатьох мов програмування лежить теорія алгоритмів, теорія формальних систем, логіка предикатів. Наприклад, назва мови ПРОЛОГ означає скорочення від слів "ПРОграмування ЛОГічне". Крім того, синтез логіки й комп'ютерів привів до виникнення баз даних і експертних систем, що стало важливим етапом на шляху до створення штучного інтелекту.

1.2. Основні поняття алгебри висловлень.

Під висловленням прийнято розуміти твердження природною мовою, істинність якого можна установити, тобто іншими словами, твердження, про яке можна сказати, істинне воно чи хибне. Способи (правила) формального представлення висловлень, побудови нових висловлень з наявних за допомогою логічно витриманих перетворень, а також способи (методи) встановлення істинності висловлень вивчаються у логіці висловлень. Усі наукові знання (закони і явища фізики, хімії, біології й ін., математичні теореми і т.п.), події повсякденного життя, ситуації, що виникають в економіці і процесах керування, формулюються у виді висловлень. Наказові, питальні і безглузді пропозиції не є висловленнями.

<u>Приклад 1</u>. Наведемо приклади висловлень:

1. «Реєстрація фірми вимагає наявності її статуту»;
2. «В умовах підвищення конкуренції необхідний налагоджений механізм керування підприємством»;
3. «Ризик іманентно притаманний економічній діяльності»;
4. «Операції об'єднання, перетинання і доповнення є операціями з множинами»;
5. «Київ – столиця України»;
6. «Від перестановки місць доданків сума не змінюється»;
7. «Якщо йде дощ, то виходячи на вулицю, варто взяти парасолю».

Усі ці висловлення мають істинне значення. Можна привести приклади помилкових (хибних) висловлень:

8. «Київ не розташований на березі Дніпра»;

9. «Вашингтон – місто в Україні»;
10. «Студент – відмінник за результатами минулої сесії одержав у минулу сесію дві незадовільні оцінки;

Можна привести приклади висловлень, істинність яких оцінити неможливо, але вони в логіці висловлень не розглядаються. Крім того помітимо, що в ряді випадків істинність чи хибність висловлення залежить від того, яку конкретну реальність (систему, процес, явище) ми намагаємося з його допомогою описати. У такому випадку говорять, що дане висловлення істинне (чи хибне) у даній інтерпретації (контексті). Далі припускаємо, що контекст заданий і висловлення має визначене істинне значення. У логіці висловлень має значення не зміст висловлень, а їхнє істинне значення, причому істинне значення складених висловлень теж визначається не за змістом складеного висловлення, а за істинними значеннями складових його висловлень.

Означення 1. *Будемо називати висловлення простим (елементарним), якщо його неможливо розкласти на більш дрібні. Звичайно до них відносяться висловлення, що не містять логічних зв'язувань. Складним (складеним) називається висловлення, складене з простих за допомогою логічних зв'язувань.*

З приведених вище прикладів висловлень друге, четверте, шосте, сьоме, восьме і десяте є складними, всі інші – елементарні. У природній мові (при вербальному описі явища) роль зв'язувань при складанні складних пропозицій із простих грають наступні граматичні засоби: союзи «і», «або», «не»; слова «якщо ..., то», «або ... або» (у розділовому змісті), «тоді і тільки тоді, коли» і ін. У логіці висловлень логічні зв'язування, що використовуються для складання складних висловлень, зобов'язані бути визначеними точно.

Домовимося позначити висловлення заголовними буквами латинського алфавіту, наприклад $A =$«Київ – столиця України». Тому що логіка висловлень оперує не змістом, а істинностними значеннями висловлень, вона має справу тільки з позначеннями висловлювань, істинність яких визначена.

Якщо A – висловлення, то воно має істинне або хибне значення (у залежності від контексту). Тому A можна розглядати як логічну (булеву) перемінну. У такий спосіб на об'єкти, що розглядаються в логіці висловлень, можна поширити операції і закони булевої алгебри. Заради історичної правди відзначимо, що в історії науки було навпаки: спочатку виникла алгебра висловлень, у працях Джорджа Буля «Математичний аналіз логіки» і «Закони мислення « розглядалися саме висловлення.

З точки зору сучасного погляду на математику, логіка висловлювань містить у собі дві складові частини: алгебру висловлювань і числення висловлень. Почнемо з алгебри висловлювань.

Розглядатимемо алгебру висловлень як окремий випадок булевої алгебри. Для початку поширимо поняття операцій булевої логіки на об'єкти логіки висловлень. Розглянемо основні з цих операцій.

Означення 2. *Запереченням (інверсією) висловлення А називається висловлення істинне, коли висловлення А хибне, і хибне – у протилежному випадку. Позначення:* $\overline{A}$, $\neg A$ *(читається: «не А»).*

У природних мовах запереченню висловлення А відповідає складання з висловлення A нового висловлення «невірно, що A».

Означення 3. *Кон'юнкцією (операцією «І», логічним добутком) двох висловлень A і B називається висловлення істинне, коли обидва висловлення істинні, і хибними – у всіх інших випадках. Позначається: $A\&У$ або $A \wedge B$ (читається «A і B»).*

У природних мовах кон'юнкції відповідає з'єднання висловлень союзом «і».

Означення 4. *Диз'юнкцією (операцією „АБО", логічною сумою) двох висловлень A і B називається висловлення, хибне у випадку, коли обоє висловлення хибні, і істинне – у всіх інших випадках. Позначається як $A \vee B$, $A+У$ (читається «A або B»; розуміється як нероздільне «або»).*

У природних мовах диз'юнкції відповідає з'єднання висловлень союзом «або» у нерозділовому змісті.

Означення 5. *Імплікацією (логічним слідуванням) двох висловлень A і B називається висловлення, хибне, коли A істинно, а B хибне; у всіх інших випадках – істинне. Позначення: $A \to B$, $A \supset B$ (читається: «якщо A, то B», «з A випливає B»). При цьому висловлення A називається посилкою імплікації, а висловлення B – висновком.*

При міркуваннях у природних мовах складно погодитися відразу з тим, що твердження «з неправди випливає істинна» істинне, але ми змушені визнати, що з неправди може випливати усе, що завгодно, а з істини тільки істина. Якщо $A =$«Київ не стоїть на березі Дніпра», а $B =$«Київ – столиця України», то висловлення $C = A \to B$ має значення істина.

Означення 6. *Еквівалентністю (еквіваленцією, рівнозначністю) двох висловлень A і B називається висловлення істинне, коли істинні значення A і B збігаються, і хибне – у противному випадку. Позначається як $A \sim У$, $A \equiv B$, $A \leftrightarrow B$ (читається: «A еквівалентно B», «A рівнозначно B», «A тоді і тільки тоді, коли B»).*

Означення 7. *Нерівнозначністю (що виключає «АБО», додаванням по модулю 2) двох висловлень A і B називається висловлення, істинне, коли істинні значення A і B не збігаються, і хибне – у противному випадку. Позначення: $A \oplus B$, $A \wedge B$ і ін. (читається: «або A, або B», «чи A, чи B»; розуміється – у розділовому значені).*

У природних мовах нерівнозначності відповідає з'єднання висловлень союзом «або» у розділовому змісті.

Означення 8. *Операцією Шеффера (штрих Шеффера) над висловленнями A і B називається висловлення $A|B$ ($A \uparrow B$) (читається «A і B – несумісні»), які хибні тоді і тільки тоді, коли A і B обоє істинні.*

Означення 9. *Операцію Пірса для висловлень A і B називається висловлення $A \downarrow B$ (читається «Не A, не B»), що істинно тоді і тільки тоді, коли A і B обоє хибні.*

Означення 10. *Операція сильна диз'юнкція* $A \oplus B$ *(сума по модулю 2). Ця функція відповідає розділовому «або». Вона приймає значення хибність, якщо* A *і* B *одночасно істинні або хибні.*

Підкреслимо ще раз, що алгебра висловлень є окремим випадком алгебри логіки (булевої алгебри). Тому у алгебрі висловлень використовується такий зручний інструмент булевої алгебри, як таблиці істинності.

Наданим означенням відповідають таблиці істинності, які зведені до табл..1.1.

Таблиця 1.1

A	B	$A \wedge B$, $A \& B$, AB	$A \vee B$, $A+B$	$A \to B$	$A \sim B$, $A \leftrightarrow B$	$\overline{A}$, A'	$A \oplus B$	$A \mid B$, $A \uparrow B$	$A \downarrow B$
0	0	0	0	1	1	1	0	1	1
0	1	0	1	1	0	1	1	1	0
1	0	0	1	0	0	0	1	1	0
1	1	1	1	1	1	0	0	0	0

Означення 11. *Алфавітом називається будь-яка не порожня множина символів будь-якої природи. Словом у даному алфавіті називається будь-яка кінцева послідовність (можливо порожня) символів даного алфавіту. Слово* a *називається під словом слова* b*, якщо* $b_1 a\ b_2$ *для деяких слів* b_1 *і* b_2.

Алфавіт логіки висловлень містить наступні символи: букви A, B, C, …, X, Y, Z, … або букви з індексом – $A_1, A_2, A_3, ..., A_n$, що позначають висловлення; символи логічних операцій $\wedge (\&)$, $\vee$, $\neg$, $\Rightarrow (\supset)$, $\Leftrightarrow (\sim)$; символи дужок (,).

Надамо строго математичне означення формули в логіці висловлень, як ми робили це булевій алгебрі.

Означення 12. *Слово в алфавіті логіки висловлень називається формулою ,якщо воно задовольняє наступному визначенню:*

1) будь-яка змінна, що позначає елементарне висловлення, – формула;

2) якщо A *і* B *– формули, то* $(\neg A)$, $(A \& B)$, $(A \vee BQ)$, $(A \to B)$, $(A \sim B)$, $(A \oplus B)$ *– формули;*

3) інших формул немає.

Підкреслимо, що хоча ми вибрали для побудови логіки висловлень тільки шість логічних операцій, можливе застосування і інших систем функцій (згадаємо про повноту системи логічних функцій).

<u>**Приклад 2**</u>**.** Представимо логічними формулами наступні висловлення:

1. Висловлення «Йде дощ і сніг» складається з двох простих, з'єднаних зв'язкою «і»:

А – «Йде дощ»;

В – «Йде сніг».

«І» використана тут у з'єднувальному змісті, тому і логічна формула має вид:

$$A \wedge B.$$

Приклад .3. Складене (складне) висловлення «Кит – риба або ссавець» складається з двох простих:

А – «Кит – риба»;

В – «Кит – ссавець».

Висловлення A і B з'єднані зв'язуванням «або» (очевидно в розділовому змісті), тобто – $\oplus$. Таким чином, дане початкове складне висловлення представимо формулою:

$$A \oplus B.$$

Приклад 4. Складне висловлення «Якщо підвищується конкуренція, то варто налагодити механізм керування підприємством» включає два простих висловлення:

А – «Підвищується конкуренція»;

В – «Варто налагодити механізм керування підприємством».

У реченні «Якщо підвищується конкуренція, то варто налагодити механізм керування підприємством «висловлення A і B з'єднані зв'язуванням «якщо…,то…». Таким чином, дане початкове складне висловлення представимо формулою:

$$A \to B.$$

Приклад 5. Складне висловлення «Якщо підприємство в сфері промисловості та будівництва України має чисельність працюючих до 200 осіб, то це підприємство належить до категорії малих підприємств за Законом України «Про підприємство в Україні» містить два простих висловлення:

А – «Підприємство в сфері промисловості та будівництва України має чисельність працюючих до 200 осіб»,

В – «Підприємство належить до категорії малих підприємств за Законом України «Про підприємства в Україні».

Висловлення A і B зв'язані за допомогою «Якщо..., то...». Таким чином, дане початкове складне висловлення представимо формулою:

$$A \to B.$$

Приклад 6. Складне висловлення «Фізичне зношення основних засобів означає втрату основними засобами їх фізичних якостей під впливом хімічних, фізичних, механічних, біологічних процесів» містить два простих висловлення:

А – «Фізичне зношення основних засобів»,

В – «Втрата основними засобами їх фізичних якостей під впливом хімічних, фізичних, механічних, біологічних процесів».

Таким чином, дане початкове складне висловлення представимо формулою:

$$A \sim B.$$

Приклад 7. Складне висловлення «Якщо підприємство реалізує вироблену продукцію, то відшкодовує витрати на виробництво і отримує прибуток». Позначимо:

А – «Підприємство реалізує вироблену продукцію», B – «Підприємство відшкодовує витрати на виробництво»,

C – «Підприємство отримує прибуток».

Таким чином, дане початкове складне висловлення представимо формулою:

$$A \to (B \wedge C).$$

Приклад 8. «Якщо соціологічні дослідження показують, що споживач віддає перевагу зручності і різноманіттю вибору, то фірмі варто зробити упор на удосконалення товару або збільшення різноманіття нових форм».

Складене висловлення складається з наступних простих:

X – «Соціологічні дослідження показують, що споживач віддає перевагу зручності»,

Y – «Соціологічні дослідження показують, що споживач віддає перевагу різноманіттю вибору»,

Z – «Фірмі варто зробити упор на удосконалення товару»,

U – «Фірмі варто зробити упор на збільшення різноманіття нових форм».

Таким чином, дане початкове складне висловлення представимо формулою:

$$(X \,\&\, Y) \to (Z \vee U).$$

У логіці висловлень вивчається побудова складних висловлень, які виражені формулами, не залежно від змісту простих висловлень, що їх складають. Тому висловлення, яких подано одною і тією самою формулою, логічно нерозрізнені. Наприклад, складні висловлення з прикладів 4.4, 4.5. Істинне значення цих і будь-яких інших складних висловлень, описуваних даною логічною формулою $(A \to B)$, буде визначатися тільки тим, істинні чи хибні висловлення A, B. Оскільки кожне з цих висловлень може бути або істинним, або хибним, тобто може мати одне з двох значень, то неважко бачити, що дана формула (складене висловлення), що включає два символи простих висловлень, має $2^2 = 4$ *логічних інтерпретацій* (наборів значень простих висловлень, що складають дану формулу), що розрізняються. Формула, яка реалізує складне висловлення $(X \,\&\, Y) \to (Z \vee U)$ з прикладу 4.8, що включає чотири символи простих висловлень, має $4^2 = 16$ логічних інтерпретацій, що розрізняються. При цьому *змістовних інтерпретацій* цієї формули, мабуть, нескінченна множина.

Приклад 9. «Якщо прогноз стверджує, що можна дістати великий прибуток на випуску нових товарів, то при розробці стратегії розвитку фірмі варто зробити упор на маркетинг і мережу розподілу, а також доцільно відкрити більш великі магазини і розширити торгову мережу» [12].

Позначимо прості висловлення:

A – «Прогноз стверджує»,

B – «Можна дістати великий прибуток на випуску нових товарів»,

C – «При розробці стратегії фірмі варто зробити упор на маркетинг», D – «При розробці стратегії фірмі варто зробити упор на мережу розподілу»,

E – «Фірмі доцільно відкрити більш великі магазини»,

F – «Фірмі доцільно розширити торгову мережу»,

Таким чином, дане початкове складне висловлення представимо формулою:

$$(A \sim B) \to (C \wedge D \wedge E \wedge F).$$

Приклад 10. «Якщо фірма продовжує випуск існуючого продукту й орієнтована на існуючий ринок, то для неї доцільна стратегія «малого корабля» або економії витрат. Така стратегія приваблива, якщо інтенсивний маркетинг – стратегічний господарський фактор, але слабка сторона організації. Якщо інтенсивний маркетинг є стратегічним господарським фактором і сильною стороною фірми, то фірмі варто дотримуватися стратегії захоплення нових ринків для існуючого продукту» [12].

Введемо позначення простих висловлень, що містяться в першому реченні:

A – «Фірма продовжує випуск існуючого продукту»,

B – «Фірма орієнтована на існуючий ринок»,

C – «Для фірми доцільна (приваблива) стратегія «Малого корабля»,

D – «Для фірми доцільна (приваблива) стратегія економії витрат».

З урахуванням уведених позначень логічна формула для першого речення має вигляд:

$$(A \,\&\, B) \to (C \sim D)\,.$$

Друге речення містить нові прості висловлення:

K – «Інтенсивний маркетинг є стратегічним господарським фактором організації»,

L – «Інтенсивний маркетинг є слабкою стороною організації».

Логічна формула, що представляє друге речення:

$$(K \,\&\, L) \to (C \sim D)\,.$$

У третьому реченні містяться нові прості висловлення:

M – «Інтенсивний маркетинг є сильною стороною організації»,

N – «Фірмі варто дотримуватися стратегії захоплення нових ринків для існуючого продукту».

Логічна формула для третього речення:

$$(K \,\&\, M) \to N\,.$$

Остаточний текст записується наступною логічною формулою:

$$((A \,\&\, B) \to (C \sim D)) \,\&\, ((K \,\&\, L) \to (C \sim D)) \,\&\, ((K \,\&\, M) \to N)\,.$$

1.3. Основні задачі алгебри висловлень.

В зв’язку з тим, що алгебра висловлень є окремим випадком алгебри логіки, яку ми розглянули раніше, весь апарат алгебри логіки застосовний до алгебри висловлень. Всі розділи алгебри логіки можуть бути інтерпретовані як розділи алгебри висловлень. Нагадуємо, що алгебра логіки оперувала тільки змінними і формулами, незалежно від змісту змінних, які входили до формул. Так само і алгебра висловлень має справу з формулами логіки висловлень незалежно від змісту конкретних висловлень, які входять до цих формул. Саме цим окресленні межі логіки висловлень. Тобто в логіці висловлень є проблеми:

1. встановлення (визначення) істинності формули логіки висловлень;
2. доведення тотожності формул;
3. встановлення рівносильності формул логіки висловлень;
4. встановлення розв'язуваності формул;
5. скорочення формул логіки висловлень.

Всі ці проблеми (задачі) розв'язуються методами алгебри логіки, які ми достатньо повно розглядали в I розділі. Зупинимось на деяких з них, які є найбільш цікавими у алгебрі логіці висловлень.

Приклад 11. Встановити істинність формули $AB\overline{C}+C$, якщо A і B істинні, а C хибне.

Таблиця 1.2

A	B	C	$\overline{C}$	AB	$AB\overline{C}$	$AB\overline{C}+C$
1	1	0	1	1	1	1

Найбільш наочне розв'язати цю задачу за допомогою таблиці істинності (див. табл.1.2).

Приклад 12. Встановити істинність формули $A\overline{B}\to\overline{C}$, якщо A і C істинні, а В хибне.

Розв'язуємо так само за допомогою таблиці істинності (табл.1.3).

Таблиця 1.3

A	B	C	$\overline{C}$	$A\overline{B}$	$A\overline{B}\to\overline{C}$
1	0	1	0	1	0

Наведемо приклад інтерпретації щодо даної формули. Нехай висловлення:
А – «на вулиці сяє сонце»,
В – «йде дощ»,
С – «на вулиці добра погода».
Тоді висловлення:
$\overline{B}$ – «не йде дощ»,
$\overline{C}$ – «на вулиці недобра погода».

Надана формула реалізує складне висловлення «якщо на вулиці сяє сонце і не йде дощ, то на вулиці недобра погода», яке при наданих значеннях істинності простих висловлень А, В, С має значення хибність.

Приклад 13. Встановити істинність формули $(AB)\to(C\sim D)$, з прикладу 10, якщо всі висловлення А, В, С і D істинні.

Таблиця 1.4

A	B	C	D	AB	$C\sim D$	$(AB)\to(C\sim D)$
1	1	1	1	1	1	1

Тобто складне висловлення «якщо фірма продовжує випуск існуючого продукту й орієнтована на існуючий ринок, то для неї доцільна стратегія «малого корабля» або економії витрат» має значення істинне.

<u>Приклад 14</u>. Встановити істинність формули $(KL) \to (C \sim D)$, з прикладу 10, якщо всі висловлення K, L, C і D істинні.

Таблиця 1.5

K	L	C	D	KL	$C \sim D$	$(KL) \to (C \sim D)$
1	1	1	1	1	1	1

Тобто складне висловлення «стратегія «малого корабля» або економії витрат приваблива, якщо інтенсивний маркетинг – стратегічний господарський фактор, але слабка сторона організації» має значення істинне.

<u>Приклад 15</u>. Встановити істинність формули $(KM) \to N$, з прикладу 10, якщо всі висловлення K, M і N істинні.

Таблиця 1.6

K	M	N	KM	$(KM) \to N$
1	1	1	1	1

Тобто складне висловлення «якщо інтенсивний маркетинг – стратегічний господарський фактор, але слабка сторона організації. Якщо інтенсивний маркетинг є стратегічним господарським фактором і сильною стороною фірми, то фірмі варто дотримуватися стратегії захоплення нових ринків для існуючого продукту» має істинне значення.

<u>Приклад 16</u>. Встановити істинність формули

$F = ((AB) \to (C \sim D)) \cup ((KL) \to (C \sim D)) \,\&\, ((KM) \to N)$, з прикладу 10.

Ми вже встановили істинність всіх складових цієї формули (табл.1.4÷1.6) і остаточно маємо табл..1.7.:

Таблиця 1.7

$(AB) \to (C \sim D)$	$(KL) \to (C \sim D)$	$(KM) \to N$	F
1	1	1	1

Тобто складне висловлення з прикладу 10 «Якщо фірма продовжує випуск існуючого продукту й орієнтована на існуючий ринок, то для неї доцільна стратегія «малого корабля» або економії витрат. Така стратегія приваблива, якщо інтенсивний маркетинг – стратегічний господарський фактор, але слабка сторона організації. Якщо інтенсивний маркетинг є стратегічним господарським фактором і сильною стороною фірми, то фірмі варто дотримуватися стратегії захоплення нових ринків для існуючого продукту» має істинне значення за умови істинного значення кожного простого висловлення, яке до нього входить.

Тобто можна сказати, що якщо експерт не помилився щодо визначення істинності простих висловлень А, В, С, D, K, L, M і N, то його рекомендація (складне висловлення) заслуговує на повагу, тому що має значення істина.

Таким чином на прикладах ми розглянули клас задач визначення істинності формул логіки висловлень. До того же ми захопили дуже важливий клас задач інтерпретації формул логіки висловлень. Треба звернути увагу на те, що одна й та сама формула має багато інтерпретацій. Формула $(AB) \to (C \sim D)$ з прикладу 13 і формула $(KL) \to (C \sim D)$ з прикладу 14 збігаються з точністю до позначок висловлень, вони мають однакові таблиці істинності, але, як видно з прикладів 11– 16, інтерпретації цих формул різні. Можне ще навести багато інтерпретацій формули $(AB) \to (C \sim D)$.

Так само, як алгебрі логіки, в алгебрі висловлень існують поняття диз'юнктивної нормальної форми (ДНФ), кон'юнктивної нормальної форми (КНФ), довершеної диз'юнктивної нормальної форми (ДДНФ), довершеної кон'юнктивної нормальної форми (ДКНФ). ДДНФ і ДКНФ частіше за все використовуються для скорочення формул.

<u>Приклад 17</u>. Скоротити формулу

$$F = \overline{K}MN + \overline{K}M\overline{N} + K\overline{M}N + K\overline{M}\,\overline{N} + \overline{K}\,\overline{M}N + \overline{K}\,\overline{M}\,\overline{N} + KMN$$

Використовуючи закони булевої логіки (табл..1.10) маємо:

$$F = \overline{K}MN + \overline{K}M\overline{N} + K\overline{M}N + K\overline{M}\,\overline{N} + \overline{K}\,\overline{M}N + \overline{K}\,\overline{M}\,\overline{N} + KMN =$$

$$\underbrace{\overline{K}MN + \overline{K}MN}_{\overline{K}MN} + \overline{K}M\overline{N} + K\overline{M}N + K\overline{M}N + K\overline{M}\,\overline{N} + \overline{K}\,\overline{M}N + \overline{K}\,\overline{M}N + \overline{K}\,\overline{M}\,\overline{N} + KMN =$$

$$= \overline{K}M(\underbrace{N + \overline{N}}_{=1}) + K\overline{M}(\underbrace{N + \overline{N}}_{=1}) + \overline{K}\,\overline{M}(\underbrace{N + \overline{N}}_{=1}) + MN(\underbrace{K + \overline{K}}_{=1}) + \overline{M}N(\underbrace{K + \overline{K}}_{=1}) =$$

$$= \overline{K}M + K\overline{M} + \overline{K}\,\overline{M} + \overline{K}\,\overline{M} + MN + \overline{M}N = \overline{K}(M + \overline{M}) + \overline{M}(K + \overline{K}) + N(M + \overline{M}) =$$

$$= \overline{K} + \overline{M} + N$$

Якщо ми застосуємо послідовно закон де Моргана, а потім співвідношення $\overline{a} + b = a \to b$, то отримаємо

$$F = \overline{K} + \overline{M} + N = \overline{K}\,\overline{M} + N = KM \to N .$$

Таким чином ми отримали одну із формул із п.9 прикладу 2. якщо ви побажаєте повернутися до початкових інтерпретацій висловлень K, M, N то ви можете отримати інтерпретацію початкової формули приклада 4 й отримати інтерпретацію кінцевої скороченої формули.

Нагадуємо, що в алгебрі логіки ми розглянули повні системи логічних функцій, тобто такі набори логічних функцій, за допомогою яких можна представити будь-яку функцію. Прикладами таких систем є: $\{\wedge, \vee, \neg\}$, $\{\wedge, \neg\}$, $\{\vee, \neg\}$.

Вигляд ДДНФ і ДКНФ є прикладом представлення будь-якої функції (за винятком тотожньо рівної 0 або 1) за допомогою повної системи $\{\wedge, \vee, \neg\}$. згадаємо, як ми отримували, наприклад, ДДНФ функції за „одиницями”, а ДКНФ за „нулями” таблиці істинності.

<u>Приклад 18</u>. Отримали ДДНФ і ДКНФ функції, яка реалізована формулою $F = KM \to N$.

Побудуємо таблицю істинності наданої функції (табл.1.8)

Таблиця 1.8

K	M	N	KM	$KM \to N$	Елементарні	
					кон’юнкції	диз’юнкції
0	0	0	0	1	$\overline{K}\overline{M}\overline{N}$	$K + M + N$
0	0	1	0	1	$\overline{K}\overline{M}N$	$K + M + \overline{N}$
0	1	0	0	1	$\overline{K}M\overline{N}$	$K + \overline{M} + N$
0	1	1	0	1	$\overline{K}MN$	$K + \overline{M} + \overline{N}$
1	0	0	0	1	$K\overline{M}\overline{N}$	$\overline{K} + M + N$
1	0	1	0	1	$K\overline{M}N$	$\overline{K} + M + \overline{N}$
1	1	0	1	0	$KM\overline{N}$	$\overline{K} + \overline{M} + N$
1	1	1	1	1	KMN	$\overline{K} + \overline{M} + \overline{N}$

Будуємо ДДНФ за „одиницями” за правилом, яке наведено в X.

$$KM \to N = \overline{K}\overline{M}\overline{N} + \overline{K}\overline{M}N + \overline{K}M\overline{N} + \overline{K}MN + K\overline{M}\overline{N} + K\overline{M}N + KMN.$$

Ця формула збігається з формулою з прикладу 4. Тобто ми перевірили, ч правильно ми зробило скорочення (спрощення) формули в прикладі 4.

Будуємо ДКНФ за „нулями” функції:

$$KM \to N = \overline{K} + \overline{M} + N.$$

Отримана ДКНФ містить одну елементарну диз’юнкцію $\overline{K} + \overline{M} + N$.

В алгебрі висловлень існує ще один клас задач. Це задачі на побудову таблиць істинності, в яких треба по заданій таблиці істинності знайти одне або кілька висловлень, що мають цю таблицю. Цей тип задач завжди має рішення, причому таке, у якому використовуються три операції – кон’юнкція, диз’юнкція, заперечення.

Метод рішення полягає в тому, що кожному рядку таблиці істинності ставляться у відповідність істинні кон’юнкції, що містять кожну змінну чи її заперечення – відповідно тому, чи стоїть під цією змінною в даному рядку значення 1 чи 0. Нагадаємо, що такі кон’юнкції називаються елементарними кон’юнкціями.

<u>Приклад 19</u>. Знайти складне висловлення, що складається з трьох простих висловлень, таблиця істинності якого:

Таблиця 1.9

A	B	C	$f(A,B,C)$	Елементарні кон'юнкції
1	1	1	0	$A \wedge B \wedge C$
1	1	0	1	$A \wedge B \wedge \overline{C}$
1	0	1	0	$A \wedge \overline{B} \wedge C$
1	0	0	0	$A \wedge \overline{B} \wedge \overline{C}$
0	1	1	0	$\overline{A} \wedge B \wedge C$
0	1	0	0	$\overline{A} \wedge B \wedge \overline{C}$
0	0	1	0	$\overline{A} \wedge \overline{B} \wedge C$
0	0	0	1	$\overline{A} \wedge \overline{B} \wedge \overline{C}$

Щоб знайти висловлення з заданої таблиці істинності, потрібно взяти диз'юнкцію елементарних кон'юнкцій з тих рядків, яким у заданій таблиці істинності відповідає значення 1. У даному прикладі шукане висловлення буде: $(A \wedge B \wedge \overline{C}) \vee (\overline{A} \wedge \overline{B} \wedge \overline{C})$.

Очевидно, що цей метод рішення може бути перенесений на випадок n змінних.

Вправа 1.

1. Виходячи з визначення логічної формули, визначити, чи є формулами наступні вираження:

1) $(A \vee B) \to (\overline{A} \wedge \overline{B}) \to$
2) $(A \sim B) \to \overline{A}$
3) $(A \to (A \oplus B)) \sim (D \to \neg C)$
4) $\neg(A \to \neg B)\neg$
5) $(A \vee)B$
6) $(A \oplus B) \sim\to C$
7) $\sim C \oplus D$
8) $(A \sim \oplus D)$
9) $(A \to B) \to C$
10) $\neg(A \to B) \oplus C$
11) $((A \oplus B) \to C$
12) $\to (A \to C)$

2. Застосувавши еквівалентні перетворення, спростіть приведені формули:

1) $(A \to A) \to A$;
2) $(A \wedge B) \vee (A \wedge \overline{B})$;
3) $(\neg(\overline{A} \wedge B) \vee (A \to B)) \wedge A$;
4) $(A \to B) \wedge (B \to \overline{A})$;
5) $A \sim B \to C \to \overline{A}$;
6) $C \oplus A \wedge (B \uparrow \overline{A})$;
7) $C \uparrow A \to (\overline{B} \oplus A)$;
8) $A \vee \overline{A} \to (B \oplus \overline{B}) \wedge C$;
9) $B \downarrow C(B \wedge \overline{A} \leftarrow C)$;
10) $A \to B \downarrow C \wedge \overline{A}$;
11) $A \downarrow B \wedge (C \leftarrow \overline{B})$;
12) $B \to (\overline{C} \to A \downarrow \overline{B})$;

3. Записати логічними формулами наступні складні висловлення:

1) „Для характеристики ефективності виробництва окремих виробів розраховують не тільки рентабельність підприємства, а й рентабельність продукції";

2) „Щоб успішно діяти в ринковій системі, підприємства України змушені створювати на базі відділів реалізації нові структури, які займатимуться маркетингом";

3) „Для ринкової економіки характерними ознаками реалізації є: зв’язок, не визначений державним розподілом, між виробниками і споживачами продукції, вільний вибір партнерів, наявність конкуренції між партнерами”;

4) „Для здійснення нормальної виробничої діяльності підприємства повинно у кожний момент мати потрібну кількість виробничих запасів, незавершене виробництво, затрати на майбутній період”;

5) „Якщо темпи росту ринку продукту корпорації високі і розмір контрольованої нею частки ринку також високий, то відповідно до матриці портфельного аналізу цей продукт відноситься до категорії „зірка”; він дає великий доход, але вимагає значних вкладень”;

6) „Аналіз і практика показують, що чим більше економіка захищається від інфляції через індексацію, тим не стабільнішою стає економіка; чим більше суспільство прагне ізолювати своїх членів від інфляції, тим не стабільнішим стає їх становище. Країни, які повністю індексували свою економіку, не мали великого успіху, але й не мали великих коливань”;

7) „Якщо частка підприємця на ринку певного товару перевищує 35%, то за Законом України „Про обмеження монополізму та недопущення недобросовісної конкуренції у підприємницькій діяльності” його становище визнається монопольним. Рішенням Антимонопольного комітету України може визнаватися монопольним становище підприємства, частка якого на ринку певного товару менша 35%”;

8) „Якщо стратегічна господарська одиниця корпорації – лідер у непривабливій (можливо, старої) галузі, її стратегією може бути максимізація прибутку на уже вкладений капітал, але не вкладення нового”;

9) „У ситуації, де життєво необхідне розширення фірми або де ключові патенти або ключові ресурси знаходяться в руках інших компаній, а даній фірмі бракує технічних знань, кращою стратегією для неї є придбання (підприємств)”.

Тема 2. Числення висловлень

2.1. Числення висловлювань

В попередній лекції було уведено означення формальної теорії та розглянуто загальні властивості формальних теорій, а також правила виведення та властивості виведення у формальних теоріях, включаючи виведення з гіпотезами. В цій лекції розглядається числення висловлювань – конкретний приклад формальної теорії. Можна сказати, що це числення є базовим та, в деякому сенсі, найпростішим, тому знайомство з формальними теоріями прийнято починати саме з нього.

Для того, щоб означити числення висловлювань, ми маємо задати його складові як формальної системи: алфавіт, формули, аксіоми та правила виведення.

Означення 2.1. Числення висловлювань – це формальна теорія L, в якій:

1) Алфавіт включає пропозиційні літери: A, B, C,... з індексами або без; пропозиційні зв'язки: $\neg$ (заперечення) та $\rightarrow$ (імплікація); допоміжні символи: (та);
2) Визначення формули числення L:
   - Довільна прорпозиційна літера є формулою.
   - Якщо A та B формули, то формулами також є $(\neg A)$ та $(A\rightarrow B)$.
   - Інших формул в численні L не існує.
3) У численні L визначена нескінчена множина аксіом, які будуються за допомогою трьох **схем аксіом**:

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

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

   A3. $(\neg B\rightarrow\neg A) \rightarrow ((\neg B\rightarrow A) \rightarrow B)$.
4) У численні L визначено єдине правило виведення MP: $A, A\rightarrow B \models B$.

У пункті 3 визначення числення висловлювань йде мова про схеми аксіом. Це означає, що для отримання конкретної аксіоми, ми маємо взяти одну з трьох схем (A1, A2, A3) та замість пропозиційних літер, які входять до неї, підставити певні формули (якими також є й атомарні формули, тобто пропозиційні літери). До того ж, замість однієї й тієї самої пропозиційної літери аксіоми ми маємо підставляти одну й ту саму формулу. Наприклад, зі схеми A1 отримуються такі аксіоми:

- $A\rightarrow (A\rightarrow A)$;
- $(A\rightarrow B)\rightarrow (B\rightarrow(A\rightarrow B))$;
- $A\rightarrow (\neg A\rightarrow A)$.

Слід звернути увагу на те, що в численні висловлювань використовуються тільки символи зв'язок імплікації та заперечення. Як і в алгебрі висловлювань, це робиться для зменшення кількості операцій. Інші зв'язки ми можемо виразити за допомогою імплікації та заперечення:

- $A\wedge B$ означає $\neg(A\rightarrow\neg B)$;

- $A \vee B$ означає $\neg A \to B$;
- $A \sim B$ означає $\neg((A \to B) \to \neg(B \to A))$.

Тепер ми можемо розглянути приклади виведення теорем у теорії L. Доведемо теорему $A \to A$. Оскільки єдиним правилом виведення є MP (правило **modus ponens**), то нам потрібно взяти таку аксіому, щоб формула $A \to A$ була у кінці формули. Для цього підходять перші дві схеми аксіом. Третя не підходить, тому що в ній зустрічається зв'язка заперечення, яка не присутня у теоремі, яку ми доводимо. У схемі A1 формула $A \to A$ з'являється вкінці, якщо замінити літеру B на A. Але для того, щоб вивести формулу $A \to A$ з $A \to (A \to A)$ за правилом MP нам необхідна наявність вже виведеної формули A. Тож перша схема не підходить.

Розглянемо схему A2 и змінимо літери B та C на формули $A \to A$ та A відповідно. Отримаємо аксіому:

$(A \to ((A \to A) \to A)) \to ((A \to (A \to A)) \to (A \to A))$.

Як ми бачимо, в кінці цієї аксіоми зустрічається потрібна нам формула $A \to A$. Але для її виведення нам потрібні тепер вивести дві формули: $A \to ((A \to A) \to A)$ та $A \to (A \to A)$. Обидві формули ми отримуємо з першої схеми за підстановкою замість літери B формул $A \to A$ та A відповідно.

Підсумуємо наші міркування, записавши їх у вигляді наступного виводу, наводячи для кожного пункту схему аксіом або правило виведення із засновками, які застосовувались для отримання цього пункту.

Теорема L1. $\models A \to A$

1. $(A \to ((A \to A) \to A)) \to ((A \to (A \to A)) \to (A \to A))$ A2
2. $A \to ((A \to A) \to A)$ A1
3. $A \to (A \to A)$ A1
4. $(A \to (A \to A)) \to (A \to A)$ MP (1,2)
5. $A \to A$ MP (3,4)

Наведемо приклад виведення ще двох теорем теорії L.

Теорема L2. $A \models B \to A$

1. A гіпотеза
2. $A \to (B \to A)$ A1
3. $B \to A$ MP (1,2)

Теорема L3. $\models (\neg A \to A) \to A$

1. $(\neg A \to \neg A) \to ((\neg A \to A) \to A)$ A3
2. $\neg A \to \neg A$ L1
3. $(\neg A \to A) \to A$ MP (1,2)

В останньому виводі ми використали вже виведену теорему L1. Це дозволяє нам робити третя властивість виведень з гіпотезами.

2.2. Теорема дедукції

У математичних міркуваннях часто якесь твердження B доводиться у припущенні правильності якогось іншого твердження A, після чого встановлюють, що правильним є твердження “якщо А, то В”. У численні висловлювань цей метод обґрунтовується такою теоремою.

Теорема 2.1 (теорема дедукції Ербрана). Нехай Г – множина формул, А і В – формули й Г, А $\vDash$ В. Тоді Г $\vDash$ А$\rightarrow$В.

Доведення. Нехай $B_1,...,B_n$ є виведенням В з Г та А. Доведення проведемо методом індукції за n-довжиною виведення. При n=1 формула В збігається з B_1. Згідно з означенням виведення можливі є три випадки:

- B_1 – аксіома:
  1. $\vDash B_1$
  2. $\vDash B_1 \rightarrow (A \rightarrow B_1)$ A1
  3. $\vDash A \rightarrow B_1$ за правилом МР
  4. $\Gamma \vDash A \rightarrow B_1$ за першою властивістю виведення з гіпотезами.
- B_1 – формула з множини Г – доведення аналогічно попередньому пунктові.
- B_1 збігається з А. Але ми вже довели теорему L1: $\vDash$ А$\rightarrow$А і за першою властивістю виведення з гіпотезами маємо Г $\vDash$ А$\rightarrow$А.

Припустимо тепер, що коли довжина виведення В з Г, А менша від n, твердження теореми є правильним. Доведемо його для випадку, коли довжина виведення дорівнює n. При цьому можливі чотири випадки:

- B_n – аксіома – доводиться аналогічно коли n=1;
- B_n – формула з множини Г – доводиться аналогічно коли n=1;
- B_n збігається А – доводиться аналогічно коли n=1;
- B_n виводиться за МР з попередніх формул, тобто у послідовності $B_1,...,B_n$ є формули: B_m та $B_l = B_m \rightarrow B_n$, $l<n$, $m<n$. Тоді за припущенням індукції маємо:
  1. $\Gamma \vDash A \rightarrow B_m$
  2. $\Gamma \vDash A \rightarrow B_l$, тобто $\Gamma \vDash A \rightarrow (B_m \rightarrow B_n)$
  3. $\Gamma \vDash (A \rightarrow (B_m \rightarrow B_n)) \rightarrow ((A \rightarrow B_m) \rightarrow (A \rightarrow B_n))$ A2
  4. $\Gamma \vDash (A \rightarrow B_m) \rightarrow (A \rightarrow B_n)$ МР (2,3)
  5. $\Gamma \vDash A \rightarrow B_n$ МР (1,4)

Теорему доведено. ►

Справедлива обернена зворотна теорема дедукції.

Теорема 2.2 (зворотна теорема дедукції). Якщо існує вивід Г $\vDash$ А$\rightarrow$В, то формула В виводиться з Г та А, тобто якщо Г $\vDash$ А$\rightarrow$В, то Г, А $\vDash$ В.

Доведення. Нехай вивід формули А$\rightarrow$В має вигляд: $B_1,...,B_{n-1}$, А$\rightarrow$В, де $B_1,...,B_{n-1}$ – формули з множини Г. Тоді вивід формули В з Г та А буде мати вигляд: $B_1,...,B_{n-1}$, А$\rightarrow$В, А, В, так як В слідує з А$\rightarrow$В та А по правилу МР. ►

Теорема дедукції має наступні наслідки.

Наслідок 1 (правило силогізму). А$\rightarrow$В, В$\rightarrow$С $\vDash$ А$\rightarrow$С.

Побудуємо виведення.

1. А$\rightarrow$В гіпотеза
2. В$\rightarrow$С гіпотеза

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

Тоді отримали $A\to B$, $B\to C$, $A \models C$. За теоремою дедукції маємо $A\to B$, $B\to C \models A\to C$.

<u>Наслідок 2</u> (правило видалення середньої посилки). $A\to(B\to C)$, $B \models A\to C$.
Після двократного застосування правила MP дістаємо $A\to(B\to C)$, B, $A \models C$. Звідси за теоремою про дедукцію маємо $A\to(B\to C)$, $B \models A\to C$.

2.3. Приклади виведень у теорії L

Застосування теореми дедукції та її наслідків дуже спрощує побудову виведень у теорії L. Наведемо декілька прикладів таких виведень.

Теорема L4. $\models \neg\neg A\to A$

1. $(\neg A \to \neg\neg A) \to ((\neg A \to \neg A) \to A)$ A3
2. $\neg A \to \neg A$ L1
3. $(\neg A \to \neg\neg A) \to A$ наслідок 2 до 1,2
4. $\neg\neg A \to (\neg A \to \neg\neg A)$ A1
5. $\neg\neg A\to A$ наслідок 1 до 3,4

Теорема L5. $\models A\to\neg\neg A$

1. $(\neg\neg\neg A\to\neg A) \to ((\neg\neg\neg A \to A) \to \neg\neg A)$ A3
2. $\neg\neg\neg A\to\neg A$ L4
3. $(\neg\neg\neg A \to A) \to \neg\neg A$ MP (2,3)
4. $A \to (\neg\neg\neg A \to A)$ A1
5. $A \to \neg\neg A$ наслідок 1 до 3,4

Теорема L6. $\models \neg A\to (A\to B) \Leftrightarrow \neg A, A \models B$.

1. $\neg A$ гіпотеза 1
2. A гіпотеза 2
3. $(\neg B\to\neg A) \to ((\neg B\to A) \to B)$ A3
4. $\neg A \to (\neg B\to\neg A)$ A1
5. $A \to (\neg B\to A)$ A1
6. $\neg B\to\neg A$ MP (1,4)
7. $\neg B\to A$ MP (2,5)
8. $(\neg B\to A) \to B$ MP (3,5)
9. B MP (7,9)

Теорема L7. $\models (\neg A\to\neg B) \to (B\to A) \Leftrightarrow \neg A\to\neg B \models B\to A$.

1. $\neg A\to\neg B$ гіпотеза
2. $(\neg A\to\neg B) \to ((\neg A\to B) \to A)$ A3
3. $(\neg A\to B) \to A$ MP (1,2)
4. $B \to (\neg A\to B)$ A1
5. $B\to A$ наслідок 1 до 3,4

Теорема L8. $\models (B\to A) \to (\neg A\to\neg B) \Leftrightarrow B\to A \models \neg A\to\neg B$.

1. $B\to A$ гіпотеза

2. $\neg\neg B \to B$ L4
3. $A \to \neg\neg A$ L5
4. $\neg\neg B \to A$ наслідок 1 з 1,2
5. $\neg\neg B \to \neg\neg A$ наслідок 1 з 3,4
6. $(\neg\neg B \to \neg\neg A) \to (\neg A \to \neg B)$ L7
7. $\neg A \to \neg B$ MP (5,6)

Теорема L9. $\models A \to (\neg B \to \neg(A \to B)) \Leftrightarrow A \models \neg B \to \neg(A \to B)$.

1. A гіпотеза
2. $((A \to B) \to B) \to (\neg B \to \neg(A \to B))$ L8
3. $A, A \to B \models B$ правило MP
4. $A \models (A \to B) \to B$ теорема дедукції до 3
5. $(A \to B) \to B$ MP (1,4)
6. $\neg B \to \neg(A \to B)$ MP (2,5)

Теорема L10. $\models (A \to B) \to ((\neg A \to B) \to B) \Leftrightarrow A \to B, \neg A \to B \models B$.

1. $A \to B$ гіпотеза 1
2. $\neg A \to B$ гіпотеза 2
3. $(A \to B) \to (\neg B \to \neg A)$ L8
4. $(\neg A \to B) \to (\neg B \to \neg\neg A)$ L8
5. $\neg B \to \neg A$ MP (1,3)
6. $\neg B \to \neg\neg A$ MP (2,4)
7. $(\neg B \to \neg\neg A) \to ((\neg B \to \neg A) \to B)$ A3
8. $(\neg B \to \neg A) \to B$ MP (6,7)
9. B MP (5,8)

2.4. Інші формалізації логіки висловлювань

Теорія L не є єдиною можливою формалізацією числення висловлювань. Її основна перевага – лаконічність при збереженні певної наочності. Дійсно, у теорії L всього лиш дві зв'язки, три схеми аксіом та одне правило. Відомі також і інші формалізації числення висловлювань, запропоновані різними авторами.

Теорія L_1 (Гілберт, Акерман). Основні зв'язки: $\vee$, $\neg$ ($A \to B = \neg A \vee B$).

Схеми аксіом:

A1. $A \vee A \to A$

A2. $A \to A \vee B$

A3. $A \vee B \to B \vee A$

A4. $(B \to C) \to (A \vee B \to A \vee C)$

Правило виведення: MP.

Теорія L_2 (Росер). Основні зв'язки: $\wedge$, $\neg$ ($A \to B = \neg(A \wedge \neg B)$).

Схеми аксіом:

A1. $A \to (A \wedge A)$

A2. $(A \wedge B) \to A$

A3. $(C \to B) \to (\neg(A \wedge B) \to \neg(C \wedge A))$

Правило виведення: MP.

Теорія L_4 (Кліні). Основні зв'язки: $\wedge$, $\vee$, $\neg$, $\to$.
Схеми аксіом:

A1. $A\to (B\to A)$;
A2. $(A\to(B\to C)) \to ((A\to B) \to (A\to C))$;
A3. $(A\wedge B)\to A$
A4. $(A\wedge B)\to B$
A5. $A \to (B \to (A\wedge B))$
A6. $A\to A\vee B$
A7. $B\to A\vee B$
A8. $(A\to C) \to ((B\to C) \to ((A\vee B) \to C))$
A9. $(A\to B) \to ((A\to\neg B) \to \neg A)$.
A10. $\neg\neg A\to A$

Правило виведення: MP.

2.5. Модельні властивості теорії L

У минулій лекції нами були розглянуті властивості, якими можуть володіти формальні теорії. Розглянемо виконуваність цих властивостей для теорії L. В якості інтерпретації формальної теорії L оберемо алгебру висловлювань. Поставимо у відповідність кожній літері теорії L пропозиційну літеру, кожній формулі L – формулу логіки висловлювань, кожній теоремі – тавтологію логіки висловлювань. Неважко пересвідчитись, що схеми аксіом теорії L є тавтологіями алгебри висловлювань. Відповідно, логіка висловлювань є моделлю теорії L.

Теорема 2.3 (повнота теорії L). Формула A виводиться у теорії L тоді і тільки тоді, коли A – тавтологія.

Доведення. В один бік доведення випливає з того факту, що аксіоми A1 – A3 є тавтологіями, в чому неважко пересвідчитись, побудувавши для них таблиці істинності. А застосування правила MP до тавтологій зберігає логічне слідування, тобто отримані формули також будуть тавтологіями (див. теорему 14.3). Отже, всяка теорема теорії L – тавтологія.

Доведення в протилежний бік в рамках даного курсу не розглядається. Додатково з ним можна ознайомитись у [2, 4, 7, 9, 11]. ►

Теорема 2.4. Теорія L несуперечлива.

Доведення. Кожна теорема теорії L є тавтологією логіки висловлювань. Заперечення формули, яка є тавтологією, відповідно, не є тавтологією. Отже, для жодної формули A неможливо, щоби A та $\neg A$ були теоремами теорії L. ►

Теорема 2.5. Теорія L неповна у широкому сенсі.

Доведення. Дійсно, не всяка формула або її заперечення є теоремами теорії L. Якщо взяти нейтральну формулу логіки висловлювань, то її заперечення також є нейтральною формулою, тобто ані сама формула, ані її заперечення не є тавтологіями алгебри висловлювань. Тому ці формули не є теоремами числення L. ►

Теорема 2.6. Теорія L повна у вузькому сенсі.

Доведення. Для доведення теореми треба показати, що теорія L стає суперечливою при додаванні до її системи аксіом довільної формули цієї теорії, яка не доводиться.

Дійсно, теорія L має три схеми аксіом A1, A2, A3 та правило виведення MP. Побудуємо нову теорію L', додавши до системи аксіом L формулу A, яка не є тавтологією логіки висловлювань. Тоді формула A приймає хоча б одне хибне значення на деякій інтерпретації. Значить, якщо A представлена кон'юнктивною нормальною формою, то ця форма має містити хоча б одну елементарну диз'юнкцію δ, яка не містить жодну змінну разом із її запереченням: $\delta = B'_1 \vee \ldots \vee B'_n$, де $B'_i = \neg B_i$, якщо $|B_i|$=T, та $B'_i = B_i$, якщо $|B_i|$=F. У елементарній диз'юнкції δ замінимо кожне входження пропозиційної літери B_i на B, якщо $|B_i|$=F, та на $\neg$B, якщо $|B_i|$=T. Отримуємо: $\delta' = B \vee \neg\neg B \vee \ldots \vee B \vee \neg\neg B = B$. Зробимо іншу заміну: зімнемо B_i на B, якщо $|B_i|$=T, та на $\neg$B, якщо $|B_i|$=F. Тоді, $\delta'' = \neg B \vee \ldots \vee \neg B = \neg B$. У новій теорії L'.

У новій теорії L' формула A – аксіома, тобто $\models$ A. Оскільки A може бути представлена у ДКНФ, то за правилом видалення $\wedge$ довільна елементарна диз'юнкція δ кон'юнктивної нормальної форми також може бути доведена. А через те, що диз'нкція δ може бути представлена у вигляді як δ', так і δ'', то в L' виконується і $\models \delta'$, і $\models \delta''$, а це означає, що $\models$ B і $\models \neg$B Тобто теорія L' суперечлива. ►

<u>Теорема 2.7.</u> Теорія L розв'язувана.

Доведення. Це справедливо за тим міркуванням, що для кожної теореми теорії існує відповідна тавтологія, а для довільної тавтології можна побудувати таблицю істинності. ►

<u>Теорема 2.8.</u> Схеми аксіом A1, A2, A3 у теорії L незалежні.

Доведення. Для кожної зі схем аксіом ми маємо показати, що їй властива певна особливість, яку не мають інші схеми аксіом. Або ж навпаки, якщо дві з трьох схем мають певну особливість, то її не має схема, яка залишилась. До того ж, ця особливість, якщо вона виконується для певних схем, має зберігатись при використанні до цих схем правила виведення MP.

Доведемо незалежність A1. Для цього розглянемо наступні таблиці істинності для зв'язок заперечення та імплікації у трьохзначній логіці.

A	¬A
0	1
1	1
2	0

Табл. 2.1,а.

A	B	A→B
0	0	0
0	1	2
0	2	2
1	0	2
1	1	2
1	2	0
2	0	0
2	1	0
2	2	0

Табл. 2.1,б.

Для довільних розподілень значень 0, 1, 2 для літер, які входять у формулу А, ці таблиці дозволяють знайти відповідне значення формули А. Якщо формула А завжди приймає значення 0, то вона називається виділеною. Правило МР зберігає цю властивість. В цьому можна пересвідчитись, звернувши увагу на перший рядок таблиці 2.1,б. Також неважко пересвідчитись в тому, що довільна аксіома, яка отримується за схемами А2 або А3, також буде виділеною. Відповідно, виділеною також буде і довільна формула, яка отримується з А2 та А3 за допомогою МР. Але формула $A\to(B\to A)$, яка є частковим випадком А1, не є виділеною, тому що вона приймає значення 2, коли А приймає значення 1 та В приймає значення 2.
Доведення незалежності А2 та А3 є аналогічним, тільки розглядаються інші варіанти таблиць 2.1. Для додаткової інформації можна звернутись до [7]. ►

2.6. Інші методи перевірки тотожної істинності формул логіки висловлювань

Таким чином, на даний момент ми знаємо принаймні три способи перевірки для довільної формули логіки висловлювань чи є вона тавтологією, тобто чи є вона тотожньо істинною. Перший, тривіальний, полягає в побудові таблиці істинності для цієї формули. Ми можемо це зробити через те, що кількість літер, що входять у довільну формулу є скінченою (n), а кількість можливих кортежів значень, що їх можуть приймати ці літери, відповідно, дорівнюватиме 2^n, що також є скінченим числом. Отже, за скінчену кількість кроків ми можемо побудувати таблицю істинності і, якщо в кожному рядку буде стояти значення Т, то цим буде доведено тотожна істинність обраної формули.
Другий метод відноситься до булевої алгебри. Тому він, відповідно, й називається – алгебраїчний. Він полягає у зведенні довільної формули логіки висловлювань до ДНФ або КНФ. Якщо під час такого зведення формула перетвориться на 1, тобто Т, то це й буде означати її тотожну істинність.
Третій метод був розглянутий у цій лекції і полягає у побудові для обраної формули виводу у формальній теорії L. Через те, що ця теорія є аксіоматичною, то й метод має відповідну назву – аксіоматичний. Якщо у теорії L побудований вивід для певної формули з використанням лише трьох схем аксіом А1, А2, А3, то за властивістю повноти теорії отримуємо, що ця формула буде тавтологією, тобто тотожньо істинною.
Перший метод є найпростішим, але й водночас найгроміздкішим. Якщо ж порівнювати другий метод із третім, то з'ясується, що для певних формул найліпшим, тобто економнішим за обчислювальними витратами, буде другий, а для інших - третій.
Нижче ми розглянемо ще два методи: метод Квайна та метод редукції. Для них так само не можна сказати, який з них у порівнянні з рештою є кращим: для формул різного вигляду найкращими будуть різні методи.
Отже, **метод Квайна** полягає в наступному. Нехай $\{A_1, A_2,\ldots, A_n\}$ – упорядкована множина пропозиційних літер, що зустрічаються у формулі $P(A_1,$

$A_2, \ldots, A_n$). Візьмемо першу з літер – A_1 і припишемо їй, наприклад, значення T (F). Підставимо це значення у формулу P і виконаємо обчислення, які можуть виникнути в результаті такої підстановки. Після виконання обчислень одержимо деяку формулу $P'(A_2, \ldots, A_n)$, до якої знову застосовується описана процедура, тобто вибираємо літера A_2, приписується їй значення T (F), виконується обчислення і т.д. Може трапитися так, що на деякому кроці буде отримана формула P'', яка є тавтологією або суперечністю незалежно від значень висловлювань, які входять до складу формули P''. Отже, на цьому кроці роботу алгоритму можна зупинити. Таким чином, метод Квайна в деяких випадках приводить до розгляду значно меншої кількості інтерпретацій, ніж тривіальний алгоритм побудови таблиць істинності.

Наприклад, розглянемо формулу $P = (((A \wedge B) \to C) \wedge (A \to B)) \to (A \to C)$. Множина літер {A,B,C}. Вибираємо літеру A. При цьому можливі два випадки:

1) A=T. Тоді

$P = (((T \wedge B) \to C) \wedge (T \to B)) \to (T \to C) = ((B \to C) \wedge B) \to C = P'$.

Тепер вибираємо B і розглядаємо знову можливі випадки:

1.1) B=T. Тоді $P' = ((T \to C) \wedge T) \to C = (C \wedge T) \to C = C \to C$ – тавтологія.

1.2) B=F. Тоді $P' = ((F \to C) \wedge F) \to C = (T \wedge F) \to C = F \to C = T$.

2) A=F. Тоді:

$P = (((F \wedge B) \to C) \wedge (F \to B)) \to (F \to C) = ((F \to C) \wedge T) \to T = (T \wedge T) \to T = T \to T = T$.

Отже, дана формула є тавтологією.

Метод редукції дає можливість виконувати перевірку формул логіки висловлювань шляхом зведення до абсурду. Він особливо зручний, коли в записі формули зустрічається багато імплікацій.

Нехай формула P має вигляд імплікації, наприклад, $P = A \to B$. Припустимо, що в деякій інтерпретації I формула P приймає значення F. Тоді у відповідності з таблицею істинності для імплікації маємо A=T та B=F. Таким чином, перевірка формули P зводиться до перевірки формул A та B. Після цього даний процес застосовується до формул A та B і т.д.

Наприклад, маємо формулу $P = ((A \wedge B) \to C) \to (A \to (B \to C))$. Нехай для деякої інтерпретації I маємо P=F. Тоді $(A \wedge B) \to C = T$, а $A \to (B \to C) = F$. Застосуємо цю процедуру до другої з формул. Отримуємо A=T та $B \to C = F$. Звідси знаходимо, що A=T, B=T, C=F. Але при отриманих значеннях $(A \wedge B) \to C = F$, що суперечить припущенню. Отже, формула P тотожно істинна.

Тема 3. Вивідність із формул. Метатеорема дедукції

3.1. Основні схеми логічно правильних умовиводів.

Означення 3.1. *Процес одержання нових знань, виражених висловленнями, з інших знань, також виражених висловленнями, називається умовиводом. Вхідні висловлення називаються посилками (гіпотезами, умовами), а одержувані висловлення – висновком (наслідком)*

Означення 3.2. *Умовивід називається правильним, якщо з кон'юнкції істинних посилок випливає істинний висновок, тобто з того, що всі посилки істинні випливає істинність висновку.*

Нехай $A_1, A_2, ..., A_m$ – посилки, B висновок. Тоді для визначення правильності умовиводу за схемою $\frac{\mathrm{A}_1, A_2, ..., A_m}{B}$, тобто твердження про те, що із самих посилок $A_1, A_2, ..., A_m$ випливає висновок B, потрібно встановити тотожну істинність формули $(A_1 \wedge A_2 \wedge .... \wedge A_m) \to B$.

Приведемо приклади найбільше використовуваних *схем логічно правильних* умовиводів:

1. Правило висновку – стверджуючий модус (modus ponens)

„Якщо істинне, що з висловлення A випливає висловлення B і висловлення A істинно, то істинно й висловлення B”. Позначається:

$$\frac{A \to B, A}{B}.$$

Доведемо modus ponens. Задля цього доведемо тотожну істинність формули $((A \to B) \wedge A) \to B$ за допомогою таблиці істинності:

AB	$A \to B$	$(A \to B) \wedge A)$	$((A \to B) \wedge A) \to B$
00	1	0	1
01	1	0	1
10	0	0	1
11	1	1	1

Правило modus ponens доведено.

Приклад 3.1. Нехай є такі висловлення:

А – „Йде дощ”;

В – „На вулиці асфальт мокрий”.

Правило modus ponens для цих висловлень має такий вигляд: „Якщо йде дощ, то на вулиці асфальт мокрий. Йде дощ. Отже, на вулиці асфальт мокрий”.

2. Правило заперечення – негативний модус (modus tollens): „Якщо з A треба B істинно, але висловлення B хибне (істинне $\overline{B}$), то хибне A (істинне $\overline{A}$)”:

$$\frac{A \to B, \overline{B}}{\overline{A}}.$$

Доведемо modus tollens. Задля цього доведемо тотожну істинність формули $((A \to B) \wedge \overline{B}) \to \overline{A}$ за допомогою таблиці істинності:

AB	$A \to B$	$\overline{B}$	$(A \to B) \wedge \overline{B}$	$\overline{A}$	$((A \to B) \wedge \overline{B}) \to \overline{A}$
00	1	1	1	1	1
01	1	0	0	1	1
10	0	1	0	0	1
11	1	0	0	0	1

Правило modus tollens доведено.

Приклад 3.2. Для тих самих висловлень з прикладу 4.8. правило modus tollens

$$\frac{A \to B, \overline{B}}{\overline{A}}.$$

має таку інтерпретацію: „Якщо йде дощ, то на вулиці асфальт мокрий. На вулиці асфальт не мокрий (сухий). Отже, не йде дощ (дощу нема)”.

3. Правила ствердження-заперечення (modus ponendo-tollens): „Якщо справедливо або висловлення A, або висловлення B (у розділовому змісті) і істинно одне з них, то інше хибне”:

$$\frac{A \oplus B, A}{\overline{B}}; \frac{A \oplus B, B}{\overline{A}}.$$

Доведемо одне з правил modus ponendo-tollens $((A \oplus B) \wedge A) \to \overline{B}$ (друге доводиться за аналогією) за допомогою таблиці істиності:

AB	$A \oplus B$	$(A \oplus B) \wedge A$	$\overline{B}$	$((A \oplus B) \wedge A) \to \overline{B}$
00	0	0	1	1
01	1	0	0	1
10	1	1	1	1
11	0	0	0	1

Правило modus ponendo-tollens доведено.

Приклад 3.23. Нехай є такі висловлення:

A – „зараз літо”,

B – „зараз осінь”.

Тоді правило ствердження-заперечення (modus ponendo-tollens), але цієї інтерпретації має вигляд:

1) „Зараз або літо, або осінь. Зараз літо. Отже, зараз не осінь”.

2) „Зараз або літо, або осінь. Зараз осінь. Отже, зараз не літо”.

4. Правила заперечення-твердження(modus tollen-ponens):

а) „Якщо істинно або A, або B (у розділовому змісті) і хибне одне з них, то істинне інше”:

$$\frac{A \oplus B, \overline{A}}{B}; \frac{A \oplus B, \overline{B}}{A};$$

б) „Якщо істинно A або B (у нерозділовому змісті) і хибне одне з них, то істинне інше":

$$\frac{A \vee B, \overline{A}}{B}; \frac{A \vee B, \overline{B}}{A}.$$

Доведемо два з чотирьох правил modus tollen-ponens $\frac{A \oplus B, \overline{A}}{B}$ і $\frac{A \cup B, \overline{A}}{B}$ за допомогою таблиць істиності:

AB	$A \oplus B$	$\overline{A}$	$(A \oplus B) \wedge \overline{A}$	$((A \oplus B) \wedge \overline{A}) \to B$
00	0	1	0	1
01	1	1	1	1
10	1	0	0	1
11	0	0	0	1

AB	$A \vee B$	$\overline{A}$	$(A \vee B) \wedge \overline{A}$	$((A \vee B) \wedge \overline{A}) \to B$
00	0	1	0	1
01	1	1	1	1
10	1	0	0	1
11	1	0	0	1

Правило modus ponendo-tollens доведено.

Приклад 3.24. Для тих самих висловлень з прикладу 4.22 правило а) заперечення-ствердження (modus tollen-ponens)

$$1)\ \frac{A \oplus B, \overline{A}}{B},\ 2)\ \frac{A \oplus B, \overline{B}}{A}$$

має вигляд:

1) „Зараз або літо, або осінь. Зараз не літо. Отже, зараз осінь”, або

2) „Зараз або літо, або осінь. Зараз не осінь. Отже, зараз літо”.

Приклад 3.25. Позначимо елементарні висловлення:

А – „Йде сніг”,

В – „Йде дощ”.

Тоді правило modus tollen-ponens в нерозділовому змісті:

$$\frac{A \vee B, \overline{A}}{B}; \frac{A \vee B, \overline{B}}{A}.$$

має вигляд:

1) „Йде дощ або сніг. Дощу нема. Отже, йде сніг”, або

2) „Йде дощ або сніг. Снігу нема. Отже, йде дощ”.

5. Правило транзитивності (спрощене правило силогізму): „Якщо з A випливає B, а з B випливає C, то з A випливає C”:

$$\frac{A \to B, B \to C}{A \to C}.$$

Доведемо правило силогізму за допомогою таблиці істинності

ABC	$A \to B$	$B \to C$	$(A \to B) \wedge (B \to C)$	$A \to C$	$(A \to B) \wedge (B \to C) \to (A \to C)$
000	1	1	1	1	1
001	1	1	1	1	1
010	1	0	0	1	1
011	1	1	1	1	1
100	0	1	0	0	1
101	0	1	0	1	1
110	1	0	0	0	1
111	1	1	1	1	1

Приклад 3.26. Позначимо елементарні висловлення:
A − „Наступає кінець фінансового року”;
B − „Треба готувати фінансовий звіт”;
C − „Головний бухгалтер має заповнити форми фінансового звіту”.

Тоді правило транзитивності $\dfrac{A \to B, B \to C}{A \to C}$ має вигляд:

„Якщо наступає кінець фінансового року, то треба готувати фінансовий звіт. Якщо треба готувати фінансовий звіт, то головний бухгалтер фірми має заповнити форми фінансового звіту. Отже, якщо наступає кінець фінансового року, то головний бухгалтер має заповнити форми фінансового звіту.”

6. Закон протиріччя:

„Якщо з A впливає B і $\overline{B}$, то хибне A (істинне $\overline{A}$)”:

$$\frac{A \to B, A \to \overline{B}}{\overline{A}}.$$

Доведемо закон протиріччя за допомогою таблиці істинності:

AB	$A \to B$	$\overline{B}$	$A \to \overline{B}$	$(A \to B) \wedge (A \to \overline{B})$	$\overline{A}$	$((A \to B) \wedge (A \to \overline{B})) \to \overline{A}$
00	1	1	1	1	1	1
01	1	0	1	1	1	1
10	0	1	1	0	0	1
11	1	0	0	0	0	1

7. Правило контрапозиції:

„Якщо істинне, що з A випливає B, то істинне, що з $\overline{B}$ випливає $\overline{A}$”;

$$\frac{A \to B}{\overline{B} \to \overline{A}}$$

Доведемо правило контрапозиції за допомогою таблиці істинності:

AB	$A \to B$	$\overline{B}$	$\overline{A}$	$\overline{B} \to \overline{A}$	$(A \to B) \to (\overline{B} \to \overline{A})$
00	1	1	1	1	1
01	1	0	1	1	1
10	0	1	0	0	1
11	1	0	0	1	1

<u>Приклад 3.27</u>. Використаємо елементарні висловлення A і B з прикладу 4.26. Тоді правило контрапозиції $\dfrac{A \to B}{\overline{B} \to \overline{A}}$ має вигляд:

„Якщо наступає кінець фінансового року, то треба готувати фінансовий звіт. Отже, якщо не треба готувати фінансовий звіт, то кінець фінансового року не наступає”.

8. Правило складної контрапозиції:

„Якщо істинне, що з A і B вплиavає C, то з A и $\overline{C}$ випливає $\overline{B}$”:

$$\frac{(A \wedge B) \to C}{(A \wedge \overline{C}) \to \overline{B}}$$

Доведемо правило складної контрапозиції за допомогою таблиці істинності:

ABC	$A \wedge B$	$A \wedge B \to C$	$\overline{C}$	$\overline{B}$	$A \wedge \overline{C}$	$(A \wedge \overline{C}) \to \overline{B}$	$((A \wedge B) \to C) \to ((A \wedge \overline{C}) \to \overline{B})$
000	0	1	1	1	0	1	1
001	0	1	0	1	0	1	1
010	0	1	1	0	0	1	1
011	0	1	0	0	0	1	1
100	0	1	1	1	1	1	1
101	0	1	0	1	0	1	1
110	1	0	1	0	1	0	1
111	1	1	0	0	0	1	1

9. Правило перерізу:

„Якщо істинне, що з A випливає B, а з B і C треба D, то з A і C випливає D”:

$$\frac{A \to B, (B \wedge C) \to D}{(A \wedge C) \to D}$$

A	B	C	AB	$\to C$	$A\overline{C}$	$\to \overline{B}$	$\to$
0	0	0	0	1	0	1	1
0	0	1	0	1	0	1	1
0	1	0	0	1	0	1	1
0	1	1	0	1	0	1	1
1	0	0	0	1	1	1	1
1	0	1	0	1	0	1	1
1	1	0	1	0	1	0	1
1	1	1	1	1	0	1	1

Наведемо без доведення ще декілька правильних умовиводів.

10. Правило імпортації (об’єднання посилок):

$$\frac{A \to (B \to C)}{(A \vee B) \to C}.$$

11. Правило експортації (роз’єднання посилок):

$$\frac{(A \vee B) \to C}{A \to (B \to C)}.$$

12. Правила дилем:

а) $\dfrac{A \to C, B \to C, A \vee B}{C}$; б) $\dfrac{A \to B, A \to C, \overline{B} \vee \overline{C}}{\overline{A}}$;

в) $\dfrac{A \to B, C \to D, A \vee C}{B \vee D}$; г) $\dfrac{A \to B, C \to D, \overline{B} \vee \overline{D}}{\neg A \vee \neg C}$.

Прикладами умовиводів, що не є правильними, можуть служити:

а) $\dfrac{A \to B, B}{A}$; б) $\dfrac{A \to B, \neg A}{\neg B}$; в) $\dfrac{A \vee B, A}{\neg B}$; і ін.

Для того щоб перевірити, чи є даний умовивід логічно правильним, варто відновити схему міркування й визначити, чи є вона схемою логічно правильних міркувань. Однак така перевірка ускладнюється тим, що схем логічно правильних умовиводів нескінченна множина. Для перевірки правильності міркувань може бути використаний метод доведення від протилежного (закон протиріччя – правило 6), однак така перевірка може виявитися трудомісткою. На прикладі алгебри логіки можна показати прості способи перевірки правильності міркувань.

Щоб довести, що даний умовивід не є правильним, необхідно представити його логічною формулою й установити що формула не є тотожно істинною.

<u>Приклад 3.28</u>. Встановимо, що умовивід $\dfrac{A \to B, B}{A}$ не є правильним. Для умовиводу $\dfrac{A \to B, B}{A}$ логічною формулою є $((A \to B) \wedge B) \to A$.

За допомогою таблиці істинності (табл.3.13) встановимо, що:

Таблиця 3.13

A	B	C	$A \to B$	$(A \to B) \wedge B$	$((A \to B) \wedge B) \to A$
0	0	0	1	0	1
0	0	1	1	0	1
0	1	0	1	1	0
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	0	0	1
1	1	0	1	1	1
1	1	1	1	1	1

<u>Приклад 3.29</u>. Встановимо, що умовивід $\dfrac{A \to B, \overline{A}}{\overline{B}}$ не є правильним. Для умовиводу $\dfrac{A \to B, \overline{A}}{\overline{B}}$ логічною формулою є $((A \to B) \cap \overline{A}) \to \overline{B}$

Таблиця 3.14

A	B	C	$A \to B$	$(A \to B) \wedge \overline{A}$	$((A \to B) \wedge \overline{A}) \to \overline{B}$
0	0	0	1	1	1
0	0	1	1	1	1
0	1	0	1	1	0
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	0	0	1
1	1	0	1	0	1
1	1	1	1	0	1

За допомогою таблиці істинності (табл.3.14) встановили, що наданий умовивід не є правильним.

Приклад 3.30. Встановити, до якої схеми відноситься умовивід:

Петренко В. – студент 1–го курсу даної форми навчання або КНЕУ, або КПІ. Він не студент КПІ, отже, він – студент КНЕУ.

Позначимо елементарні висловлення:

А – «Петренко В. – студент 1–го курсу денної форми навчання КНЕУ.»

У – «Петренко В. – студент 1–го курсу денної форми навчання КПІ.»

Схема даного умовиводу $\dfrac{A \oplus B, \overline{A}}{A}$, і вона є схемою правильного умовиводу заперечення-твердження (modus tollen-ponens).

Приклад 3.31. Встановити, до якої схеми відноситься умовивід:

„Якщо студент був відсутній на заняття, він не виконав лабораторну роботу. Він не виконав лабораторну роботу. Отже, він був відсутній на занятті”.

Позначимо елементарні висловлення:

А – „Студент був відсутній на занятті”,

В – „Студент не виконав лабораторну роботу”.

Схема даного умовиводу:

Чи є дані міркування логічно правильними?

Схема даного міркування

$$\frac{A \to B, B}{A},$$

ставиться до схеми (а) неправильних міркувань, отже, це міркування невірно.

Приклад 3.32. Встановити, до якої схеми відноситься умовивід:

„Ця людина студент або підприємець. Він студент. Отже, не підприємець”.

Позначимо елементарні висловлення:

А – „Ця людина студент”,

В – „Ця людина підприємець”.

Тоді $\overline{B}$ – „Ця людина не підприємець”.

Зважаючи на те, що в першому реченні союз „або” використаний у нерозділовому змісті, схема даного умовиводу має вигляд:

$$\frac{A \vee B, A}{\overline{B}}.$$

Схема відповідає схемі (в) неправильних умовиводів, тому даний умовивід також є неправильним.

Приклад 3.33. Встановити, до якої схеми відноситься умовивід:

„Ця людина живе у Києві або Запоріжжі. Він живе в Києві. Отже, він не живе в Запоріжжі”.

Позначимо елементарні висловлення:

А – „Ця людина живе у Києві”,

В – „Ця людина живе у Запоріжжі”.

Тоді $\overline{A}$ – „Ця людина не живе у Києві”.

Умовивід правильний, тому що його схема є схемою правильних умовиводів (див. правило 4, а):

$$\frac{A \oplus B, \overline{A}}{B}.$$

Приклад 3.34. Встановити, до якої схеми відноситься умовивід:

„Сьогодні понеділок або вівторок. Сьогодні вівторок. Отже сьогодні не понеділок”.

Позначимо елементарні висловлення:

А – „Сьогодні понеділок”,

В – „Сьогодні вівторок”.

Тоді $\overline{A}$ – „Сьогодні не понеділок”.

Союз „або” тут використаний у розділовому змісті, тому міркування правильне, тому що ставиться до схем правильних міркувань (див. правило 3):

$$\frac{A \oplus B, B}{\overline{A}}.$$

Приклад 3.35. „Якщо фірма з продажу комп’ютерної техніки запрошує на роботу провідного фахівця в галузі інформаційних технологій, то вона планує розширення спектру послуг з задоволення потреб населення з SOFT і HARDWARE. Фірма з продажу комп’ютерної техніки запросила на роботу провідного фахівця в галузі інформаційних технологій. Отже, фірма планує розширення спектру послуг з задоволення потреб населення з SOFT і HARDWARE.” Встановити правильність даного умовиводу.

Позначимо елементарні висловлення:

А – „Фірма з продажу комп’ютерної техніки запрошує на роботу провідного фахівця в галузі інформаційних технологій”,

В – „Фірма планує розширення спектру послуг з задоволення потреб населення з SOFT і HARDWARE”.

З урахуванням позначень умовивід приймає вигляд:

$$\frac{A \to B, A}{B}.$$

Ця схема є схемою правильного умовиводу (див. правило 1 modus ponens).

Приклад 3.36. Записати логічною формулою наступний умовивід:

„Якщо фірма запрошує на роботу провідного фахівця в області новітньої технології, то вона вважає її привабливої й розвертає роботи зі зміни технології виробництва свого традиційного продукту або починає розробку нового продукту. Фірма запросила на роботу провідного фахівця в області новітньої технології. Отже, вона розпочинає роботи зі зміни технології виробництва продукту, що випускає, або розробці нового продукту”.

Встановити правильність даного умовиводу.

Позначимо елементарні висловлення:

A – „Фірма запрошує на роботу провідного фахівця в області новітньої технології”.

B – „Фірма вважає дану новітню технологію привабливою”.

C – „Фірма розвертає роботу зі зміни технології виробництва свого традиційного продукту”.

D – „Фірма починає розробку нового продукту”.

З урахуванням прийнятих позначень умовивід прийме вигляд: „Якщо *A,* то B й (C або D). A. Отже, C або D”.

Використовуючи логічні зв’язування, одержимо остаточно:

$$((A \to (B \wedge (C \vee D))) \wedge A) \to (C \vee D).$$

Для перевірки правильності умовиводу відновимо схему умовиводу:

$$\frac{A \to (B \wedge (C \vee D)), A}{C \vee D},$$

й зрівняємо її зі схемою правила 1

Таким чином, даний умовивід вірний при істинності *B.*

Вправа 3.2.

1. Приведіть приклади, що ілюструють кожну з дванадцяти схем правильних умовиводів.

2. Приведіть приклади, що ілюструють схеми неправильних умовиводів.

3. До якої схеми умовиводів належать наступні умовиводи:

1) «В 1950-1970 роки характерними рисами економіки Західної Германії були високі темпи зростання і великими прибутками, тому німецькі підприємства не мали досвіду подолання ситуацій, пов’язаних з загрозою їх існування. В останні десятиріччя XX сторіччя темпи зростання зупинилися і прибутки значно зменшилися. Отже, виникла необхідність поліпшення інструментарію планування і управління.»

2) «Поведінка ринку цінних паперів виявляє наступну тенденцію: умови, у яких ціни зростають, змінюються умовами, у яких ціни спадають. Зараз ціни не спадають. Отже, в наступному періоді ціни зростатимуть.»

3) «Стратегічний контролінг має допомагати підприємству ефективного використовувати його переваги і створювати нові потенціали успішної діяльності в перспективі. Служба стратегічного контролінгу є внутрішнім консультантом менеджерів і власників підприємства при побудові, стратегії,

стратегічних цілей і завдань. Підприємство створило службу стратегічного маркетингу. Отже, підприємство має велику перспективу.».

4. Довести, що правила імпортації (об’єднання посилок), правило експортації (роз’єднання посилок) і правило дилем є правильними схемами умовиводів.

КОНТРОЛЬНІ ЗАПИТАННЯ ДО МОДУЛЯ **№1.**

1. Що називається висловленням?
2. Наведіть приклади елементарних і складних висловлень.
3. Що називається інверсією висловлення A?
4. Що називається кон’юнкцією двох висловлень A і B?
5. Що називається імплікацією двох висловлень A і B?
6. Що називається еквівалентністю двох висловлень A і B?
7. Що називається нерівнозначністю двох висловлень A і B?
8. Дайте визначення алфавіту.
9. Що називається словом у деякому алфавіті?
10. Дайте визначення формули логіки висловлень.
11. Перелічите основні задачі алгебри висловлень.
12. Як можна перевірити істинність формули алгебри висловлень?
13. Що називається аксіоматичною теорією? Як вона будується?
14. Яка формула називається тавтологією? Приведіть приклади.
15. Яка формула називається протиріччям? Приведіть приклади.
16. Яка формула називається здійсненою? Приведіть приклади.
17. Наведіть приклади аксіом числення висловлень.
18. Як формулюється modus ponens?
19. Як формулюється modus tollens?
20. Як формулюється modus ponendo-tollens?
21. Як формулюється modus tollen-ponens?
22. Як формулюється спрощене правило силогізму?
23. Як формулюється закон протиріччя?
24. Як формулюється закон контрапозиції?
25. Як формулюється закон складної контрапозиції?
26. Як формулюється правило перерізу?
27. Як формулюється правило імпортації?
28. Як формулюється правило ектрапортації?
29. Навести приклади правильних і неправильних умовиводів.

Тема 4. Предикати та операції над ними

В попередньому розділі ми розглянули логіку висловлень і в її складі – числення висловлень. Але числення висловлень – досить вузька логічна система. Існують, наприклад, такі типи логічних міркувань, які не можуть бути здійснені в межах логіки висловлень:

- Кожний друг Остапа є другом Миколи. Федір не є другом Остапа. Отже, Федір не є другом Миколи;

- Просте число два – парне. Отже, існують прості парні числа.

Коректність цих висновків ґрунтується на внутрішній структурі самих речень і значенні слів «кожний» та «існують».

Логіка предикатів, як і логіка висловлень, складається з алгебри предикатів і числення предикатів.

Подібні схеми умовиводів розглядаються в логіці предикатів. Логіка предикатів дає змогу формулювати співвідношення між елементами реального світу і виводити подібні відношення або теореми.

4.1. Предикати. Основні поняття.

4.1.1. Формули алгебри предикатів

Розглянемо речення, що залежать від змінних, наприклад, «*x* – парне число», «*x* менше за *y*», «$x + y = z$», «*x* – батько *y*», «*x* та *y* – брати», "X – приватний підприємець" тощо. Якщо *x*, *y*, *z* у перших трьох реченнях замінити деякими числами, то матимемо певні висловлення, які можуть бути істинними або хибними. Наприклад, «3 – парне число», «2 менше за 5», «3 + 2 = 7». Речення "x – батько y", "x та y брати" визначають родинні відносини між членами сім'ї і перетворюються на певні висловлення, істинні або хибні, при заміні *x* й *y* іменами людей: «Іван –батько Петра», «Іван і Олег брати».

Речення "X – приватний підприємець" виражає відношення до підприємницької діяльності і перетворюється на висловлення, істинне або хибне, при заміні змінної X на П. І. Б. конкретної людини: "Пацюченко Остап Миколайович – приватний підприємець". Речення, що залежать від змінних, називаються предикатами. Кількість змінних буває різною.

Приклад 4.1. Предикат „Реєстрація фірми вимагає наявності *x*" є предикатом від однієї змінної *x*. Предикат „В умовах у необхідний налагоджений механізм керування підприємством" є предикатом від однієї змінної у. Предикат „*x* іманентно притаманний *y*" є предикатом від двох змінних *x* і *y*.

Надамо строге математичне означення предикату.

Означення 4.1. *Предикатом P (x_1, ..., x_n) називається функція $P : X_1 \times X_2 \times \ldots \times X_N \to B$, ,тобто функція P від N змінних, яка кожному елементові (x_1, x_2, ..., x_n) з декартова добутку $X_2 \times X_2 \times \ldots \times X_N$ ставить у відповідність елемент з двоелементної множини B, де множини X_1, X_2,...,X_N називаються предметними областями предикату, x_1, x_2,...,x_N – предметні змінні предикату, $B = \{0,1\}$ або $B = \{$ істина, хибність$\}$. Якщо змінні предикату* *набувають значень із деякої множини* $X = X_1 = X_2 = ... = X_N$, *то* $P(x_1, ..., x_n)$ *діє з* X^n *в* B: $X^n \to B$, *де* $B = \{1, 0\}$.

Предикат *n* аргументів називають *n*-місним, або *n*-арним, а *n*-арністю предикату P. Множина X значень змінних визначається математичним контекстом. Наприклад, предикат „х іманентно притаманний у" можна виразити предикатом двох змінних $P_1(x, y)$, основне співвідношення елементарної геометрії «будь-які дві точки *x, y* лежать на одній прямій» можна виразити предикатом двох змінних $P_2(x, y)$; а предикат «будь-які три точки лежать в одній площині» – предикатом трьох змінних $P(x, y, z)$.

Предикати позначають великими літерами латинського алфавіту, іноді з індексами: P, P_1, P_2, ..., P_m. Іноді буває зручно вказувати число змінних предикатів. У таких випадках символи предикатів доповнюють верхнім індексом, який вказує число аргументів або арність, наприклад $P^{(n)}(x_1, ..., x_n)$ – *n*-місний або *n*-арний предикат. Висловлення вважається нуль-місним або нуль-арним предикатом.

<u>Приклад 4.2</u>. Нехай $P^{(1)}(x)$ позначає предикат "х – категорія діяльності суб'єктів господарювання, що пов'язана з подоланням невизначеності, конфліктності в ситуаціях оцінювання, управління, неминучого вибору". Він набуває значення 1 тоді й тільки тоді, коли х = ризик.

Якщо $P = P(x_1, x_2, ...,x_n)$ – предикат, то він має істинне або хибне значення у залежності від значень змінних $x_1, x_2, ...,x_n$. Тому P можна розглядати як логічну змінну. Тоді, аналогічно тому, як ми робили у логіці висловлень, на предикати можна поширити операції і закони булевої алгебри. Іншими словами існує алгебра предикатів, яка є окремим випадком булевої алгебри. Таким чином над предикатами можна виконувати звичайні логічні операції, у результаті яких утворюються нові предикати.

Нагадуємо, що кількість всіх операцій (логічних функцій від двох змінних) складає 16. Розглянемо основні з них стосовно предикатів.

Означення 4.2. *Запереченням (інверсією) предикату P називається предикат* $\overline{P}$*, істинний, коли предикат P хибний, і хибний, коли P істинний.*

<u>Приклад 5.3</u>. Нехай P(x) позначає предикат „х – парне число", тоді предикат $\overline{P}(x)$ позначає предикат „х – непарне число".

Означення 4.3. *Кон'юнкцією (логічним добутком) двох предикатів* P_1 *і* P_2 *називається предикат* $P = P_1\, P_2$ *(або* $P_1 \wedge P_2$*, або* $P_1 \& P_2$*) істинний, коли обидва предикати* $P = P_1$ *і* P_2 *істинні, і хибним у всіх інших випадках.*

<u>Приклад 4.4</u>. Нехай $P^{(1)}(x)$ позначає предикат «*x* – парне число», $Q^{(1)}(x)$ – предикат «*x* ділиться на три без залишку». Тоді вираз $P=P^{(1)}(x) \& Q^{(1)}(x)$ позначає предикат «*x* ділиться на два та *x* ділиться на три», тобто позначає предикат ділення на 6 без залишку.

Треба зробити зауваження, що бінарні операції з предикатами, які ми зараз розглянемо, не обов'язково мають бути операціями з предикатами однакової арності. Якщо деякі змінні у предикатах P_1 і P_2, що задіяні в операції, співпадають, то арність предиката P, який є результатом операції, визначається як $n = n_1 + n_2 + n_3$, де

n – арність предиката P;
n_1 – кількість співпадаючих змінних у предикатах P_1 і P_2;
n_2 – кількість змінних предикатуP_1, що не співпадають зі змінними предикату P_2;
n_3 – кількість змінних предикатуP_2, що не співпадають зі змінними предикату P_1.

Приклад 4.5. Нехай $P^{(1)}(x)$ позначає предикат „*x* – стратегічна одиниця господарювання корпорації" $Q^{(2)}$ (y, z) – предикат „*Y* – лідер у перспективній галузі *Z*" економіки. Тоді предикат $R_1^{(3)}$ $(x, y, z) = P^{(1)}(x)$ $Q^{(2)}$ (y, z) є предикатом у трьох змінних x, y, z (тобто арності 3)
„*x* – стратегічна одиниця господарювання корпорації і *y* – лідер у перспективній галузі економіки *z*".
А предикат $R_2^{(2)}$ $(X, Z) = P^{(1)}(X)\&$ $Q^{(2)}$ (X, Z) є предикатом від двох змінних x, z (тобто арності 2) „x – стратегічна одиниця господарювання корпорації і x – лідер у перспективній галузі економіки Z"

Означення 4.4. *Диз'юнкцією (логічною сумою) двох предикатів P_1 і P_2 називається предикат $P = P_1 \vee P_2$, хибний в випадку, коли обидва предикати P_1 і P_2 хибні, і істинний у всіх інших випадках.*

Приклад 4.6. Нехай $P^{(1)}(x)$ позначає предикат „x - студент", $Q^{(1)}$ (y) позначає предикат „y – приватний підприємець". Тоді вираз $R_1^{(2)}$ $(x, y) = P^{(1)}(x) \vee Q^{(1)}$ (y) позначає предикат „x – студент або y – приватний підприємець", який є істинним, якщо істинний хоча б один з предикатів $P^{(1)}(x)$ і $Q^{(1)}$ (y), а ці предикати набувають значення „істина" незалежно один від одного, тому що залежать від різних змінних x і y.
Вираз $R_2^{(1)}(X) = P^{(1)}(x) \vee Q^{(1)}$ (x) позначає предикат „x – студент або приватний підприємець". Цей предикат $R_2^{(1)}(X)$ залежить від однієї змінної x і набуває значення „істина", якщо деяка людина x є або студентом, або приватним підприємцем одночасно.

Означення 4.5. *Імплікацією (логічним слідуванням) двох предикатів P_1 і P_2 називається предикат $P = P_1 \rightarrow P_2$ $(P = P_1 \supset P_2)$, хибний, коли P_1 істинний, а P_2 хибний, і істинний у всіх інших випадках.*

Приклад 4.7. Нехай $P_1^{(2)}(x, y)$ позначає предикат „прогноз показує, що можна отримати великий прибуток за рахунок випуску нових товарів x і y", $P_2^{(1)}(z)$ „при розробці нової стратегії фірми треба використати дії Z". Тоді вираз $P^{(3)}(x, y, z) = P_1^{(2)}(x, y) \rightarrow P_2^{(1)}(z)$ позначає предикат „якщо прогноз показує, що можна отримати великий прибуток за рахунок випуску нових товарів x і y, то при розробці нової стратегії фірми треба використати дії z".

Приклад 4.8. Нехай $P_1^{(1)}(x)$ позначає предикат „x – доля ринку продукції", $P_2^{(1)}(y)$ – предикат „y – темп зростання ринку продукції", $Q^{(1)}(z)$ – предикат „продукт відноситься до категорії Z". Тоді вираз $P^{(3)}(x, y, z) = P_1^{(1)}(x)$ & $P_2^{(1)}(y)$ $\rightarrow$ $Q^{(1)}(z)$ позначає предикат „Якщо x – доля ринку продукту і y – темпи зростання ринку продукту, то даний продукт відноситься до категорії z".

Приклад 4.9. Нехай $S^{(2)}$ (x, y) позначає предикат тотожності, тобто «$x = y$». Він набуває значення 1 тоді й тільки тоді, коли $x = y$. У цьому випадку вираз $P^{(2)}(x,$

$y) = \overline{S}^{(2)}(x, y) \to S^{(2)}(x, y)$ позначає предикат, що набуває істинне значення при будь-яких x та y.

Означення 4.6. *Еквівалентністю (рівнозначністю) двох предикатів P_1 і P_2 називається предикат $P = P_1 \sim P_2$ ($P = P_1 \leftrightarrow P_2$), істинний, якщо предикати P_1 і P_2 істинні або хибні одночасно, і хибний – у всіх інших випадках.*

Приклад 4.10. Нехай $P_1^{(1)}(x)$ позначає предикат „х – стратегія фірми”, $P_2^{(1)}(y)$ – предикат „у – стратегія фірми”. Тоді вираз $P^{(2)}(x, y) = P_1^{(1)}(x) \sim P_2^{(1)}(y)$ позначає предикат „стратегія фірми х еквівалентна стратегії у”. Якщо змінна х має значення „стратегія економії вкладень”, а у має значення „у – стратегія „малого пароплава” (корабля)”, то при підстановці значень змінних х і у предикат $P^{(2)}(x, y)$ стає висловленням „стратегія фірми по економії вкладень еквівалентна стратегії малого пароплава (корабля)”, яка має істинне значення.

Еквівалентність може бути виражена й іншими словесними висловлюваннями, у тому числі:

- «з того що P_1, випливає, що P_2 і навпаки»;
- «з того, що P_2, випливає, що P_1, і навпаки»;
- «умови P_1 необхідно і досить для того, щоб P_2»;
- «P_2 необхідно і досить, щоб P_1»;
- «P_1 якщо і тільки якщо P_2»;
- «P_2, якщо і тільки якщо P_1»;
- «умови P_1 і P_2 еквівалентні»;
- «P_2 тоді і тільки тоді, коли P_1».

Означення 4.7. *Нерівнозначністю (сумою за модулем 2) двох предикатів P_1 і P_2 називається предикат $P = P_1 \oplus P_2$ ($P = P_1 \sim \overline{P}_2$), істинний, якщо значення істинності предикатів P_1 і P_2 не збігається, і хибний у всіх інших випадках.*

Приклад 4.11. Побудуємо вираз $P^{(1)}(x, y) = P_1^{(1)}(x) \oplus P_2^{(1)}(y)$ з предикатів прикладу 4.10. цей вираз позначає предикат „стратегія фірми не еквівалентна стратегії y”.

Тоді вираз $R^{(2)}(x, y) = P_1^{(1)}(x) \oplus P_2^{(1)}(y)$ позначає предикат „стратегія x фірми не еквівалентна стратегії y”. При підстановці значень змінних x і y з прикладу 4.10 предикат $R^{(2)}(x, y)$ стає висловленням „стратегія економії вкладень фірми не еквівалентна стратегії малого корабля”, яке має хибне значення.

Якщо змінна x має значення „стратегія загарбнення ринків”, а змінна y має значення „стратегія малого пароплава (корабля)”, то при підстановці значень x і y предикат $P^{(1)}(x, y)$ стає висловленням „стратегія загарбнення ринків не еквівалентна стратегії малого пароплава (корабля)”, яке має значення „істина”.

Ми навели основні приклади операції з предикатами, тобто основні приклади логічних операцій від двох предикатів. Можна розглянути всі можливі 16, але якщо ми згадаємо про повні системи функцій, розглянуті в §1.3, то очевидно, що це є зайвим. Очевидно, також, що логічні функції від трьох і більше предикатів зводяться до функцій від двох.

Весь апарат алгебри логіки є пристосованим до алгебри предикатів (також, як і до логіки висловлень). Можна розглядати таблиці істинності і т.і., але є дуже велика відмінність у порівнянні з алгеброю висловлень. Про висловлення можна сказати, що його значення істинності предикатів легко визначається. Значення істинності предикатів визначати не так легко, тому що для цього треба здійснити інтерпретацію всіх змінних, від яких залежить даний предикат, а це дуже велика робота. Предметні області (множини, на яких визначаються змінні предикату) теоретично можуть бути нескінченими. В такому випадку стандартна процедура перевірки істинності предикатів теж може бути нескінченою. Тому що потрібна підстановка усіх можливих значень змінних, а це неможливо за скінчений час.

Але згадаємо, що алгебра логіки оперує змінними, значення яких вже відомо. Тому в межах алгебри предикатів вважатимемо, що предикати мають певні значення істинності.

4.1.2. Зв'язок між предикатами і відношеннями.

Якщо уважно придивитися, то можна зрозуміти, що існує взаємно однозначна відповідність між n-арними відношеннями $R \subseteq X^n$ і n-арними предикатами $P(x_1, x_2, ..., x_n)$, $P: X^n \rightarrow B$; а саме:

1) кожному n-арному відношенню R відповідає предикат $P(x_1, x_2, ..., x_n)$ такий, що $P(a_1, a_2, ..., a_n)=1$, якщо і тільки якщо $(a_1, a_2, ..., a_n) \in R$ (для скорочення запису істинне значення предикату позначатимемо 1, а хибне - 0);

2) усякий предикат $P(x_1, x_2, ..., x_n)$ визначає відношення R таке, що $(a_1, a_2, ..., a_n) \in R$, якщо і тільки якщо $P(a_1, a_2, ..., a_n)=1$.

При цьому R *задає область істинності предиката* P.

<u>Приклад 4.12</u>. Розглянемо предикат порядку Q визначений на множині натуральних чисел: $N^2 \rightarrow B$; тобто $Q(x, y)=1$ тоді і тільки тоді, коли $x \le y$. Зрозуміло, що двомісному предикату порядку $Q(x, y)$ взаємно однозначно відповідає двомісне відношення A „бути не більше", або „x не більше за y". Пара $(x, y) \in A$ тоді і тільки тоді, коли $Q(x, y)=1$.

<u>Приклад 4.13</u>. Розглянемо предикат кратності D: $N^2 \rightarrow B$; тобто $D(x, y)=1$ тоді і тільки тоді, коли x ділиться на y без залишку.

Двомісному предикатові „x ділиться на y без залишку" взаємно однозначно відповідає двомісне відношення A_1 – „ділитися", $(x, y) \in A_1$ тоді і тільки тоді, коли $D(x, y)=1$.

Для позначення двомісних предикатів крім префіксного запису $P^{(2)}(x_1, x_2)$ використовується інфіксний запис $x_1 P x_2$.

Предикат кратності $D(x, y)$ – це 2-арний предикат, з областю визначення (предметною областю) якого можуть служити будь-які множини дійсних чисел, наприклад множина N. $D(10,5)$ - висловлення, значення якого є істина, тобто істинне висловлення. $D(3,5)$ - хибне висловлення. $D(9,x)$, $D(x,3)$ – одномісні предикати, істинність яких залежить від того, яким числом буде замінений символ x, але $D(a, 1)$ – щире висловлення, тому що для будь-якого елемента $a \in N$ має місце: $D(a, 1)=1$ (будь-яке натуральне число поділяється на одиницю).

<u>Приклад 4.14</u>. Записати формулою логіки предикатів властивість кратності цілих чисел.

Складене висловлення, що є формулюванням властивості транзитивності відношення кратності цілих чисел: „якщо *a* ділиться на *b* і *b* ділиться на с, то *a* ділиться на *c*”, складається з трьох простих висловлень $D(a, b)$, $D(b, c)$ і $D(a, c)$. Отже, властивість транзитивності відношення кратності можна записати у вигляді складного висловлення (логічної формули): $(D(a, b)\&D(b, c))\rightarrow D(a, c)$.

Приклад 4.15. Дамо словесні формулювання наступних складних висловлень:

1. «Число *a* не ділиться на число *b*, і сума a і b дорівнює *c*»:
$\neg D(a,b)\&S(a, b,c)$.
2. «Від перестановки місць доданків *a* і *b* сума с не змінюється» – властивість комутативності арифметичної операції додавання:
$S(a, b, c)\sim S(b, a, c)$;
3. «Число 3*n* є парним тоді і тільки тоді, коли *n* є парним»: $P_1\sim P_2$, де $P_1=P(3n)$, $P_2=P(n)$, де $P(x)$ – предикат число x є парний.

Вправа 4.1

Нехай $P^{(1)}(x)$ позначає предикат „x – парне число”, $Q^{(2)}(y, z)$ – предикат „*y* ділиться на *z* без залишку”.

Які предикати будуються за виразами:

1) $\overline{P}^{(1)}(x)$.
2) $\overline{Q}^{(2)}(y,z)$.
3) $P^{(1)}(x)\,\&\,Q^{(2)}(x,y)$.
4) $P^{(1)}(x)\,\&\,Q^{(2)}(y,x)$.
5) $P^{(1)}(x)\,\&\,\overline{Q}^{(2)}(x,y)$.
6) $P^{(1)}(x)\vee Q^{(2)}(x,y)$.
7) $P^{(1)}(x)\vee \overline{Q}^{(2)}(x,y)$.
8) $P^{(1)}(x)\rightarrow Q^{(2)}(x,y)$.
9) $P^{(1)}(x)\rightarrow \overline{Q}^{(2)}(x,y)$.
10) $P^{(1)}(x)\leftrightarrow Q^{(2)}(x,y)$.
11) $P^{(1)}(x)\leftrightarrow \overline{Q}^{(2)}(x,y)$.
12) $\overline{P}^{(1)}(x)\leftrightarrow Q^{(2)}(x,y)$.

Нехай $P_1^{(2)}(x,y)$ позначає предикат „Фірма x орієнтована на випуск продукції y”. $P_2^{(1)}(y)$ позначає предикат „y відповідності до матриці портфельного аналізу продукт y є продуктом категорії „зірка””. $P_3^{(3)}(y,z,t)$ - позначає предикат „продукт y дає прибуток z, але потребує вкладень t”.

Які предикати будуються за виразами:

1) $P_1^{(1)}(x,y)\,\&\,P_2^{(1)}(y)$.
2) $P_1^{(1)}(x,y)\vee P_2^{(1)}(y)$.
3) $P_1^{(1)}(x,y)\,\&\,P_2^{(1)}(y)\,\&\,P_3^{(3)}(y,z,t)$.

4) $P_2^{(1)}(y) \& P_3^{(3)}(y,z,t) \to P_1^{(2)}(x,y)$.

5) $\overline{P}_2^{(1)}(y) \to P_3^{(3)}(y,z,t)$.

6) $\overline{P}_2^{(1)}(y) \to \overline{P}_3^{(3)}(y,z,t)$.

Предикатові *B(x,y)* відповідає відношення «більше», що визначене на множині *A*, де A=$\{4,5,6,7\}$. Наведіть всі інтерпретації і визначить, чи істинні вони.

Скільки різних предикатів можна задати на одноелементній, двохелементній і трьохелементній множині?

Записати за допомогою предикатної формули:

а) комутативність множення;

б) асоціативність додавання;

в) асоціативність множення;

г) дистрибутивність зліва множення щодо додавання;

д) дистрибутивність справа множення щодо додавання;

е) транзитивність рівності.

4.2. Квантори

Задля побудови складних предикатів крім операцій алгебри логіки можна застосовувати ще операції зв'язування квантором.

Квантор загальності. Нехай *P* (*x*) – деякий предикат, який набуває значення 1 або 0 для кожного елемента *x* множини *X*. Тоді під виразом $\forall x P(x)$ матимемо на увазі істинне висловлення, коли *P* (*x*) – істинне для кожного елемента *x* із множини *X*, і хибне – в іншому випадку. Читається цей вираз так: «для всіх *x P* (*x*)» або «для будь-якого *x P* (*x*)». Це висловлення вже не залежить від х. Символ $\forall$ називається квантором загальності.

Квантор існування. Нехай *P* (*x*) – деякий предикат. Під виразом $\exists x P(x)$ будемо розуміти істинне висловлення, коли існує елемент множини *X*, для якого *P* (*x*) – істинне, та хибне – в іншому випадку. Читається цей вираз так: «існує *x* таке, що *P* (*x*)» або «існує *x*, для якого *P* (*x*)». Символ $\exists$ називається квантором існування.

Означення 4.8. *Перехід від P(x) до $\forall x P(x)$ або $\exists x P(x)$ називається зв'язуванням змінної x, або „навішуванням" квантора на змінну x (або предикат P(x)), або квантифікацією змінної x. Змінна x при цьому називається зв'язаною. Вирази $\forall x P(x)$ і $\exists x P(x)$ вже не залежать від x, при фіксованих P і X мають певні значення. Вони представляють конкретні висловлення відносно всіх елементів x області визначення X.*

Квантори $\forall x$ і $\exists x$ називаються двоїстими один до одного.

Операцію зв'язування квантором можна застосовувати також до предикатів більшого числа змінних.

Означення 4.9. *Вираз, перед яким стоїть квантор $\forall x$ або $\exists x$, називається областю дії квантора загальності або існування.*

Всі входження змінної *x* в цей вираз є зв'язаними.

Приклад 4.16. Нехай $P_1(x)$ позначає предикат „x – успішний підприємець”, який визначений на множині людей X. Тоді зрозуміло, що $\forall x P(x)$ має хибне значення, а $\exists x P(x)$ має істинне значення.

4.3. Формули алгебри предикатів

На мові предикатів можна скласти набагато складніші речення, ніж на мові логіки висловлень. Уведемо поняття формули логіки предикатів. Алфавіт цієї логіки містить такі символи:

- предметних змінних $x_1, x_2, ..., x_n ...$;
- предикатів $P_1, P_2, ..., P_k$ (можливо з позначкою арності $P_i^{(n)}$);
- логічних операцій $\bar{\ }, \&, \vee, \to$ ($\supset$), $\leftrightarrow$ ($\sim$);
- кванторів $\exists, \forall$;
- дужки і кому.

Щоб зменшити кількість індексів, символи предметних змінних можна позначати x, y, z, а символи предикатів – P, S, Q, R тощо.

Означення 4.10.

1 . Якщо P – символ предиката, $x_1,...,x_n$ *– символи предметних змінних, то* $P(x_1,...,x_n)$ *– формула, яка називається атомарною.*

Всі предметні змінні атомарних формул є вільними, зв’язаних змінних немає.

Слово в алфавіті алгебри предикатів називається формулою, якщо воно задовольняє наступному індуктивному визначенню:

Означення 4.12.

2. Нехай P – формула. Тоді $\overline{P}$ *– також формула (вільні й зв’язані змінні формули* $\overline{P}$ *– це відповідно вільні та зв’язані змінні формули P).*

3. Нехай P і S – формули, причому немає таких предметних змінних, які були б зв’язаними в одній формулі, але вільними в іншій. Тоді

$(P \vee S), (P \& S), (P \sim S)$

є формулами, в яких вільні змінні формул P та S залишаються вільними, а зв’язані змінні формул P і S – зв’язаними.

4. Нехай P – формула, яка містить вільну змінну x. Тоді

$\forall xP, \exists xP$

також є формулами. Змінна x у них – зв’язана. Інші ж змінні, які у формулі P є вільними, залишаються вільними й у формулах $\forall xP, \exists xP$*. Змінні, які у формулі P є зв’язаними, залишаються зв’язаними й у формулах* $\forall xP, \exists xP$*. У першій із формул* $\forall xP$ *формула P називається областю дії квантора* $\forall$*, а в другій* $\exists xP$ *– областю дії квантора* $\exists$.

5. Слово в алфавіті алгебри предикатів є формулою тільки в тому випадку, якщо це випливає з правил 1–4.

Зазначимо, що за означенням формули жодна змінна не може бути одночасно вільною та зв’язаною.

Приклад 4.17. Такі вирази є формулами логіки предикатів: $P(x_1, x_2, x_3)$– атомарна формула, в якій x_1, x_2, x_3 – вільні змінні;

$\forall x_1 \exists x_2 A_1^{(3)}(x_1, x_2, x_3) \to \forall x_1 A_1^{(2)}(x_1, x_4)$ – формула, в якій x_1, x_2 – зв’язані, а x_3, x_4 – вільні змінні.

Вираз $\exists x_2 \forall x_2 A_1^{(2)}(x_1, x_3) \,\&\, A_2^{(2)}(x_1, x_3)$ не є формулою.

Значення формули визначено лише тоді, коли задано яку-небудь інтерпретацію символів, що входять до неї.

Приклад 4.18. Нехай $P(x)$ – предикат «x – смертний» x визначений на множині людей X. Тоді вираз $\forall x P(x)$ означає «усі люди смертні». Він не залежить від змінної x, а характеризує всіх людей у цілому, тобто виражає висловлення щодо всіх x множини X.

Приклад 4.19. Нехай $P(x)$ – предикат «x – парне число», визначений на множині натуральних чисел N.

Тоді $\exists x P(x)$ – істинне. У загальному випадку висловлення $\exists x P(x)$ істинно на будь-якій множині X, що містить хоча б одне парне число, і хибне на будь-якій множині непарних чисел.

Приклад 4.20. Нехай $N(x)$ позначає предикат «x – натуральне число» визначений на множині X. Тоді вираз $\forall x\ N(x)$ – твердження «усі числа – натуральні» істинне на будь-якій підмножині X множини натуральних чисел і хибне, якщо X містить хоча б одне ненатуральне число, наприклад ціле від′ємне.

$\exists\ x(x)$ – твердження «існує натуральне x» істинно на будь-якій множині X, що містить хоча б одне натуральне число, і хибне – у протилежному випадку.

Приклад 4.21. Нехай предикатові $P(x, y)$ відповідає відношення «x любить y», що визначений на множині людей. Тоді:

$\forall\ x \exists\ y P(x, y)$ означає «для будь-якої людини x існує людина, яку вона любить» чи «усяка людина кого-небудь любить»;

$\exists\ y \forall\ x P(x, y)$ означає «існує така людина y, що його люблять усі x»;

$\forall\ x \forall\ y \mathrm{P}(x, y)$ означає «усі люди люблять усіх людей»;

$\exists\ x \exists\ y P(x, y)$ означає «існує людина, що когось любить»;

$\exists\ x \forall\ y \mathrm{P}(x, y)$ означає «існує людина, що любить усіх людей»;

$\forall\ y \exists\ x P(x, y)$ означає «для всякої людини існує людина, що його любить» чи «кожну людину хтось любить».

З вище приведеного можна зробити висновок про те, що перестановка кванторів спільності й існування змінює зміст висловлення, тобто квантори спільності й існування не володіють у загальному випадку властивістю комутативності.

4.4. Рівносильність формул логіки предикатів.

Нехай формули P_1 і P_2 мають одну і ту саму множину вільних змінних (зокрема, порожню).

Означення 4.12. *Формули P_1 та P_2 є рівносильними в заданій інтерпретації, якщо на будь-якому наборі значень вільних змінних вони набувають однакових значень. Формули P_1 і P_2 є рівносильними на множині M, якщо вони рівносильні у всіх інтерпретаціях, заданих на множині M. Формули P_1 та P_2 є*

рівносильними (в логіці предикатів), якщо вони рівносильні на всіх множинах (тоді писатимемо P_1=P_2).
Всі рівносильності алгебри логіки мають місце і в алгебрі предикатів, як і в алгебрі висловлень. В алгебрі предикатів ці рівносильності матимуть вигляд:
1. $\overline{\overline{P}} = P$;
2. $P \,\&\, P = 0$;
3. $P_1 \to P_2 = \overline{P_2} \to \overline{P_1}$;
4. $P_1 \,\&\, P_2 = P_2 \,\&\, P_1$;
5. $(P_1 \,\&\, P_2) \,\&\, P_3 = P_1 \,\&\, (P_2 \,\&\, P_3)$;
6. $P_1 \vee P_2 = P_2 \vee P_1$;
7. $(P_1 \vee P_2) \vee P_3 = P_1 \vee (P_2 \vee P_3)$;
8. $P_1 \vee P_2 = P_2 \vee P_1$;
9. $P \,\&\, P = P$;
10. $P_1 \vee (P_2 \,\&\, P_2) = P_1$;

Підкреслимо, що ці формули є рівносильностями в логіці предикатів (незалежно від області інтерпретації). Ще є рівносильності, які зв'язані з кванторами, а саме стосуються перенесення квантора через заперечення, винесення квантора за дужки, а також перенесення однойменних кванторів.
1. Перенесення квантора через заперечення. Нехай P - формула, що містить вільну змінну x. Тоді справджуються рівносильності:
$\overline{(\forall x)}P(x) = (\exists x)\overline{P}(x)$, $\overline{(\exists x)}P(x) = (\forall x)\overline{P}(x)$.
2. Винесення квантора за дужки. Нехай формула $P_1(x)$ містить вільну змінну x, формула P_2 не містить змінної x. Тоді
$(\exists x)(P_1(x) \,\&\, P_2) = (\exists x)P_1(x) \,\&\, P_2$;
$(\forall x)(P_1(x) \,\&\, P_2) = (\forall x)P_1(x) \,\&\, P_2$;
$(\exists x)(P_1(x) \vee P_2) = (\exists x)P_1(x) \vee P_2$;
$(\forall x)(P_1(x) \vee P_2) = (\forall x)P_1(x) \vee P_2$.
Зазначимо, що коли не вимагати, щоб формула P_2 не містила змінної x, то будуть справджуватися тільки дві рівносильності:
$(\forall x)(P_1(x)) \,\&\, P_2(x) = (\forall x)P_1(x) \,\&\, (\forall x)P_2(x)$;
$(\exists x)(P_2(x)) \vee P_2(x)) = (\exists x)P_1(x) \vee (\exists x)P_2(x)$.
3. Переставлення однойменних кванторів. Для кванторів $\forall$ та $\exists$ маємо
$(\forall y)(\forall x)P(x, y) = (\forall x)(\forall y)P(x, y)$;
$(\exists y)(\exists x)P(x, y) = (\exists x)(\exists y)P(x, y)$.
4. Перейменування зв'язаних змінних. Замінюючи зв'язану змінну формули A іншою змінною, що не входить у цю формулу, у кванторі й усюди в області його дії дістаємо формулу, рівносильну A.
Це твердження легко доводиться.
Розглянемо спосіб спрощення формул, що спирається на зведені рівносильності. Під довжиною формули тут і далі розумітимемо загальне число

символів предикатів, логічних символів та символів кванторів, які входять до неї. Так, формула $(\forall x_1)A_1^{(2)}(x_1, x_2) \& (\exists x_3)A_2^{(1)}(x_3)$ має довжину 5.

Формули, в яких із логічних символів застосовуються тільки символи &, $\vee$, $\overline{}$, причому символ $\overline{}$ зустрічається лише перед символами предикатів, будемо називати зведеними.

<u>Приклад 4.22</u>. Розглянемо такі формули:

$A_1^{(1)}(x_1) \vee A_2^{(2)}(x_1, x_2)$

$(\forall x_1)A_1^{(1)}(x_1) \& (\exists x_2)\overline{A}_2^{(2)}(x_2, x_3)$

$(\overline{A}_2^{(1)}(x_1) \vee A_1^{(1)}(x_2)$

$(\forall x)A_1^{(1)}(x_1) \rightarrow (\exists x_2)\overline{A}_1^{(1)}(x_2)$

$(\overline{\exists x_2})A_1^{(1)}(x_2) \rightarrow A_2^{(1)}(x_1)$

перші дві формули – зведені, а інші не є зведеними.

Теорема 6.7. *Для будь-якої формули існує рівносильна їй зведена формула, причому множина вільних і зв'язаних змінних цих формул збігаються.*

Така зведена формула називається зведеною формою заданої формули.

Користуючись рівносильностями логіки висловлення, легко вказати формулу, рівносильну заданій, що містить із логічних символів тільки символи &, $\vee$, $\overline{}$. Тому будемо вважати, що формули, які розглядаються, містять тільки ці логічні символи.

Вправа 4.2.

1. Нехай *Q(x,y)* - предикат порядку "$x < y$", визначений на множині *X*. Розглянути різні варіанти квантифікації змінних, визначити істинність одержаних виразів для різних випадків інтерпретації області визначення *X*:

1)∀ x Q(x,y).	7) ∀x ∃y Q(x,y).
2)∀ y Q(x,y).	8) ∀y ∃x Q(x,y).
3)∃ x Q(x,y).	9) ∃x ∀y Q(x,y).
4)∃ y Q(x,y).	10) ∃y ∀x Q(x,y).
5)∀x ∀y Q(x,y).	11) ∀x ∃y Q(x,y).
6)∃x ∃y Q(x,y).	12) ∃y ∀x Q(x,y).

2. Нехай *P(x,y)* – предикат "за рахунок випуску нових товарів *x* і *y* можна отримати великий прибуток". Розглянути варіанти квантифікації змінних *x* і *y* з попереднього пункту даного прикладу, визначити істинність одержаних виразів.

3. Який зміст мають предикатні формули:

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

б) $\forall x \forall y \forall z \forall i(\mathrm{P}(x, y, z) \& \mathrm{P}(x, y, i) \rightarrow \mathrm{Q}(z, i))$,

де *P*, *Q* – предикати добутку і рівності, визначені на *N*? Чи істинні ці формули? Привести приклади наборів змінних, що ілюструють висновок щодо істинності чи хибності формул.

4. Нехай предикат *P*(*x, y)* заданий на множині *A*={*a, b,* c} таблицею (табл. 4.1). Визначити істинність наступних формул:

1) $\forall x P(x, a)$; 2) $\forall y P(a, y)$;

3) $\forall xP(x, b)$; 4) $\exists xP(x, b)$;
5) $\forall yP(b, y)$; 6) $\exists yP(b, y)$;
7) $\forall x\forall yP(x, y)$; 8) $\forall x\exists yP(x, y)$;
9) $\forall y\exists xP(x, y)$; 10) $\forall y\forall xP(x, y)$;
11) $\exists x\forall yP(x, y)$; 12) $\exists x\exists yP(x, y)$.

Таблиця 4.1

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

Тема 5. Застосування логіки предикатів

5.1. Виконуваність, здійсненність і загальнозначущість

Розглянемо деяку інтерпретацію формули F в множині A. При логічній інтерпретації формул логіки предикатів можливі три основні ситуації:

Означення 5.1. *Формула $F(x_1, ..., x_n)$ називається здійсненною в заданій інтерпретації М, якщо в цій інтерпретації для формули F існує така підстановка констант $a_1, ..., a_n \in M$ замість перемінних $x_1, ..., x_n$, що $F(a_1, ..., a_n)$ стає істинним висловленням. Формула називається просто здійсненною, якщо існує така інтерпретація М, де F здійсненна.*

Означення 5.2. *Формула $F(x_1, ..., x_n)$ називається тотожно істинною в області М, якщо F здійсненна в М при будь-яких підстановках констант. Формула F називається тотожно істинною (ТІ), чи загальнозначущою, якщо вона тотожно істинна в будь-яких М.*

<u>Приклад 5.1</u>. Нехай $P^{(1)}(x)$ – це предикат „х – успішний підприємець". Тоді формула, очевидно, що $\exists x\, P^{(1)}(x)$ здійснена в множині людей. Формула $\exists x\, P^{(1)}(x) \vee \overline{P}^{(1)}(x)$ є ТІ-формулою (або загальнозначущою), тому що тотожно істинна в будь-яких областях.

Означення 5.3. *Формула F називається тотожно хибною в області М, якщо F нездійсненна в М, і тотожно хибною (ТХ), або протиріччям, якщо F нездійсненна ні в яких М.*

<u>Приклад 5.2</u>. Для предикату з прикладу 5.23 формула $\exists x\, P^{(1)}(x)$ – „існує успішний приватний підприємець" є тотожно хибною, наприклад, в множині тварин, а формула $\exists x P^{(1)}(x) \& \overline{P}^{(1)}(x)$ - „існує приватний підприємець, який є одночасно успішним і неуспішним", є ТХ-формулою (тотожно хибною) або протиріччям в будь-яких M.

Означення 5.4. *Моделлю М у логіці предикатів називають множину М разом із заданою на неї сукупністю предикатів $\Sigma=\{P_1, ..., P_k\}$:*

$$M=(M; P_1, ..., P_k),$$

де М – основна множина моделі М; $\Sigma=\{P_1, ..., P_k\}$ – сигнатура моделі М′. Наприклад, сигнатура моделі М′={N; S, D, E}, що називається арифметикою натуральних чисел, включає предикати суми S, добутку D и рівності E. Аналогічно попередньому визначаються формули, здійсненні на моделі М′, тотожно щирі (ТІ-формули) і тотожно хибні (ТХ-формули) на моделі М′.

<u>Приклад 5.3</u>. Розглянемо модель $M'=\{N; S, D, E\}$ (арифметику натуральних чисел). Тоді формула $(D(x, y, z)\&D(x, y, i))\rightarrow E(z, i)$ – ТІ-формула на моделі М′ у силу єдиності значення добутку чисел з N. Дійсно, для будь-яких підстановок констант замість перемінних x, y, z, i, наприклад $a, b, c, d \in N$, формула $(D(a, b, c)\&D(a, b, d))\rightarrow E(c, d)$ має значення «істинна». Формула $\exists y D(x, x, y)$ – ТІ-формула на моделі М, що виражає існування натурального квадрата натурального числа x. Дійсно, при підстановці будь-якої константи замість вільної перемінний x формула $\exists y D(x, x, y)$ істинна. Формула $\exists x D(x, x, y)$ здійсненна на моделі N: «існує натуральне значення квадратного кореня для натурального x з N» чи «$\sqrt{y}$ – натуральне число x». Очевидно, формула істинна при підстановках замість вільної перемінної y чисел 0, 1, 4, 9, 16, ... і хибна при підстановці 2, 3, 5, 6, 7, 8, 10, ...; наприклад, $\exists xD(x, x, 4)$ – істинна, а $\exists xD(x, x, 5)$ – хибність.

Очевидно, формула A є загальнозначущою тоді й тільки тоді, коли формула $\overline{A}$ не є здійснюваною, і здійснюваною тоді й тільки тоді, коли формула не є загальнозначущою.

Очевидно, якщо P_1 і P_2 – рівносильні (в логіці предикатів) формули, то $P_1 \sim P_2$ є загальнозначущою формулою. (ТІ - формула). Приведемо приклади інших ТІ – формул.

Формула $(\forall x)A(x) \to A(y)$, де змінна не входить у формулу $A(x)$, - загальнозначуща. (ТІ - формула).

Формула $A(y) \to (\exists x)A(x)$, де змінна y не входить у формулу $A(x)$, є загальнозначущою.

Як зазначалося вище, однойменні квантори можна переставляти. Тому формули $(\exists x)(\exists y)A(x, y) \sim (\exists y)(\exists x)A(x, y)$, $(\forall x)(\forall y)A(x, y) \sim (\forall y)(\forall x)A(x, y)$ будуть загальнозначущими.

Загальнозначущою є також формула $(\exists x)(\forall y)A(x, y) \sim (\forall y)(\exists x)A(x, y)$.

Задача розпізнавання загальнозначущості формул логіки предикатів істотно складніша, ніж формул логіки висловлення. Так само, як і в логіці висловлення, вона називається проблемою розв’язуваності і формулюється так: указати ефективний спосіб (алгоритм) розпізнавання загальнозначущості формул (тобто чи є задана формула загальнозначущою).

Загалом ця проблема в логіці предикатів – нерозв’язувана. Наведемо це твердження без доведення.

Теорема Черча. *Не існує алгоритму, який для будь-якої формули логіки предикатів установлює, загальнозначуща вона чи ні.*

Однак у деяких окремих випадках проблема розв’язуваності вирішується. Наприклад, якщо розглядати формули логіки предикатів, які містять тільки одномісні предикатні символи, то такий алгоритм існує. Логіка, в якій використовуються тільки одномісні предикати, відповідає логіці, що була описана ще Аристотелем.

Алгоритм перевірки загальнозначущості формул, які містять тільки одномісні предикатні символи, ґрунтується на такому твердженні.

Твердження. Нехай F – формула, що містить n одномісних предикатних символів. Для того щоб формула F була здійснюваною, необхідно й достатньо, щоб вона була здійснюваною в усіх інтерпретаціях (M, f) із множиною M, які містять не більш як 2^n елементів.

Вправа 5.1.

1. Визначити істинність, хибність або виконуваність в області натуральних чисел наступних формул:

а) $\forall x \forall y \forall z\ ((S(x, y, z) \& S(x, y, i)) \to R(z, i))$;

б) $(S(x,y,z)\,\&\,S(x,y,i))\to R(z,i)$;

в) $\forall x\forall y\forall z\ ((D(x,y,z)\,\&\,D(y,x,i))\to R(z,i))$;

г) $\exists y(D(x,x,y)\to S(x,x,y))$;

д) $\forall x\forall y\forall z\ ((D(x,y,z)\,\&\,\overline{R}(x,y))\to S(x,y,z))$;

е) $\exists y((S(x,y,z)\vee R(x,z))\to P(x,z))$;

ж) $P(x,z)\to(\exists y(x,y,z)\vee R(x,z))$;

з) $\exists xD(x,y,z)\to(P_1(z,y)\wedge P_1(z,x))$;

и) $\exists xS(x,y,y)\to\exists yS(x,y,y)$;

к) $\exists xD(y,x,y)\to\forall y\overline{D}(y,x,y)$;

л) $\exists x\forall yS(x,x,y)$;

м) $\exists y(x,x,y)\ \exists y(x,x,y)$;

н) $\forall yD(x,x,y)$.

Якщо S,D,R,P,P_1 – відповідно предикати суми, добутку, рівності, порядку, подільності.

5.2. Числення предикатів

Множина ТІ-формул (загальнозначущих) логіки предикатів входить у будь-яку теорію, дослідження цієї множини – важлива мета логіки предикатів. При цьому виділяються дві проблеми:

1) одержання ТІ-формул (проблема побудови *процедури, що* породжує множину ТІ-формул);

2) перевірка формули на істинність.

На відміну від логіки (алгебри) висловлень, де є стандартна процедура перевірки істинності формул (обчислення формул на наборах значень змінних), за допомогою якої очевидна організація процедури побудови, що породжує, множина ТІ-формул, у логіку предикатів прямий перебір усіх значень змінних може бути неможливий, якщо предметні змінні мають нескінченні області визначення.

Тобто у логіці предикатів, на відміну від логіки висловлення, немає ефективного способу для розпізнавання загальнозначущості формул. Тому аксіоматичний метод стає істотним при вивченні формул, які містять квантори. Визначення загальнозначущих формул, так само, як і в численні висловлень, здійснюється завданням деякої сукупності формул, що називаються аксіомами, і завданням правил виведення, що дають змогу з одних загальнозначущих формул одержувати інші.

Розглянемо аксіоматичну теорію, яку також називатимемо численням предикатів.

Числення предикатів – це аксіоматична теорія, символами якої є взагалі ті самі символи, що й у алгебрі предикатів:

- символи предметних змінних $x_1, x_2, \ldots, x_n, \ldots$;
- символи предикатів $A_1^{(t)}, A_2^{(t)}, \ldots, A_n^{(t)}, \ldots$ $(t = 0, 1, 2, \ldots)$;
- логічні символи $-$, $\&$, $\vee$, $\to$;
- символи кванторів $\forall$, $\exists$;

- дужки й кома (,).

Вище сформульоване означення формули поширюється також на числення предикатів з тією лише різнецею, що тут застосовується тільки два логічних символи – і $\to$; інші зв'язки можна ввести, наприклад, так, як це зроблено в численні висловлень.

Аксіоми числення предикатів. Які б не були формули A та B, такі формули є аксіомами:

1. $A \to (B \to A)$;
2. $(A \to (B \to C)) \to ((A \to B) \to (A \to C))$;
3. $(\overline{B} \to \overline{A}) \to ((\overline{B} \to A) \to B)$;
4. $A \,\&\, B \to A$;
5. $A \,\&\, B \to B$;
6. $(A \to B) \to ((A \to C) \to (A \to B \,\&\, C))$;
7. $A \to A \vee B$;
8. $B \to A \vee B$;
9. $(A \to C) \to ((B \to C) \to (A \vee B \to C))$;
10. $(A \to B) \to (\overline{B} \to \overline{A})$;
11. $A \to \overline{\overline{A}}$;
12. $\overline{\overline{A}} \to A$;
13. $(\forall x)A(x) \to A(y)$;
14. $A(y) \to (\exists x)A(x)$.

Правила виведення в численні предикатів:

1. Правило modus pones $\dfrac{A, A \to B}{B}$.

2. Правило зв'язування квантором загальності $\dfrac{B \to A(x)}{B \to (\forall x)A(x)}$, де формула B не містить змінної x.

3. Правило зв'язування квантором існування $\dfrac{A(x) \to B}{(\exists x)A(x) \to B}$, де формула B не містить змінної x.

4. Правило перейменування зв'язаної змінної. Зв'язану змінну формули A можна замінити (в кванторі й у всіх входженнях в області дії квантора) іншою змінною, яка не є вільною в A.

Поняття виведення, теореми, виведення з системи гіпотез визначаються в численні предикатів так само, як і в будь-якій аксіоматичній теорії.

Можна показати, що клас усіх теорем числення предикатів збігається з класом загальнозначущих формул.

Аксіоми числення предикатів є загальнозначущими формулами.

Формула, що випливає із загальнозначущої формули за будь-яким із правил виведення 1 – 4, є загальнозначущою.

Тема 6. Поняття алгоритму. Нормальні алгоритми

Одним із значних досягнень науки XX століття є теорія алгоритмів – нова математична дисципліна.

Теорія ЕОМ, практика програмування не можуть обійтись без теорії алгоритмів.

Теорія алгоритмів – це самостійна наука, яка має свій предмет. Предметом теорії алгоритмів є алгоритми.

Слово алгоритм походить від імені узбецького математика Хорезмі (на арабській - аль-Хорезмі). Цей математик у IX столітті н. е. розробив правила чотирьох дій над числами в десятковій системі числення. Сукупність цих правил у Європі стала називатись "алгоритм".

В 30-х роках XX ст. поняття алгоритм стало об’єктом математичного вивчення. Розвиток ЕОМ і методів програмування прискорив розвиток теорії алгоритмів, як необхідного засобу автоматизації. Але і в 1-му і в 2-му напрямках Теорії Алгоритму треба дати поняття алгоритму. Тому перейдемо до інтуїтивного поняття алгоритму з його властивостями, прикладами й недоліками.

Інтуїтивне поняття алгоритму:

Приклади деяких алгоритмів, із якими ми зустрічаємось у житті:

- у книгах рецептів приготування різних блюд;
- в довідниках аптечних рецептів для приготування ліків.

Приклади найпростіших алгоритмів із математики:

- правила оперування з десятковими числами (складання, віднімання, множення, ділення);
- правила оперування з комплексними числами;
- алгоритми розв’язання квадратичних рівнянь, систем рівнянь та інш.

Кожна наукова й технічна дисципліна має довідники. Такі довідники в значній своїй частині – це збірник алгоритмів, накопичених даною науковою або технічною дисципліною. Особливе значення мають алгоритми, накопичені в математиці, бо надбання математики є багатством усіх наук.

Алгоритм – це сукупність правил, які визначають процес переробки допустимих початкових даних у вихідні результати.

Дане поняття алгоритму має такі властивості або загальні риси, характерні для поняття алгоритму.

1. Дискретність: алгоритм виконується по кроках у дискретному часі.
2. Детермінованість: система величин, яка отримана на якомусь кроці алгоритму, однозначно визначається системою величин, отриманих на попередньому кроці алгоритму.
3. Елементарність: закон одержання системи величин на кожному кроці алгоритму із системи величин на попередніх кроках повинен бути простим.
4. Направленість: ця властивість передбачає, що виконання алгоритмів повинне завершуватись одержанням певних результатів.

Якщо ж спосіб одержання наступної величини не дає результату, то слід указати, що вважати результатом алгоритму.

5. Масовість: Початкова система величин може вибиратись із потенційно нескінченної множини величин.

Таке поняття алгоритму не строге, не точне, бо воно включає слова: правило, спосіб, величина, точний зміст яких не встановлено, хоч інтуїтивно зрозуміло зміст цих слів.

Алгоритмічна система – система, яка дає чітке поняття, визначення алгоритму, задає загальний спосіб побудови алгоритмів.

6.1. Рекурсивні функції

6.1.1. Теоретичні поняття і визначення

Історично першою алгоритмічною системою була система рекурсивних функцій . Ця теорія дає змогу з більш простих функцій будувати складніші функції або алгоритми. Завдяки теорії рекурсивних функцій було отримане уточнення поняття алгоритму.

Ця теорія базувалася на розгляді цілo-чисельних функцій.

Аргументи цих функцій і самі функції будуть приймати невід'ємні цілі значення.($N_0=\{0\} \cup N$)

Найпростіші функції (елементарні арифметичні функції)

1 Нуль функції тотожно рівні нулю

$$O^n(x_1, x_2, \ldots, x_n) = 0$$

2 Тотожні (селекторні) функції, які повторюють значення своїх аргументів:

$$I_i^n(x_1, x_2, ..x_i,\ldots x_n) = x_i \quad (1\le i \le n;\; n = 1, 2 \ldots)$$

3 Функція безпосереднього слідування

$$S(x) = x +1$$

Тепер сформулюємо конструктивні засоби, допустимі операції, які дозволяють із елементарних найпростіших функцій будувати більш складні арифметичні функції.

Операція (оператор) суперпозиції, або підстановки полягає в підстановці одних арифметичних функцій замість аргументів інших арифметичних функцій. Нехай

$$\left.\begin{array}{l} f_1^m(x_1, x_2, \ldots, x_m) \\ f_2^m(x_1, x_2, \ldots, x_m) \\ f_n^m(x_1, x_2, \ldots, x_m) \end{array}\right\} \text{ n функцій, кожна з яких залежить від m змінних}$$

і є функція

$f^n(x_1, x_2, \ldots, x_n)$ від n аргументів, або змінних.

Операція суперпозиції функцій f_1, f_{12}, f_{1n} з функцією f (або операція підстановки функцій f_1,f_2, f_n в функцією f) дає функцію

$$g^m(x_1, x_2, \ldots, x_m) = f^n (f_1^m(x_1, \ldots, x_m) \ldots f_n^m(x_1, \ldots, x_m))$$

Приклади

1 Нехай є функція $O(x) = 0$ і функція $S(x) = x +1$

Підставимо 1-шу функцію (нуль функцію у функцію безпосереднього слідування.

Отримаємо функцію

$g(x) = S(O(x)) = S(0) = 0+1 = 1;$

$g(x) \equiv 1=const_1(x)$ – функція – константа тотожно рівна 1.

2 Нехай $S(x) = x+1$.   Підставимо її ж замість x в себе, тобто утворимо суперпозицію $S(x)$ з самою собою

$g(x) = S(S(x)) = S(x + 1) = x+2$; $g(x) = S(S(x)) = x + 2$.

3 Нехай є функція $O^n(x_1, x_2, \dots, x_n) = 0$ і $S(x) = x + 1$

$S(S(S(O^n(x_1, x_2, \dots, x_n)))) = S(S(S(O))) = S(S(1)) = S(2) = 3$

$g(x_1, x_2, \dots, x_n) = 3$- функція константа.

4 Нехай $S(z) = z+1$, $I_3^3(x,y,z) = Z$

$h(x,y,z) = I_3^3(x,y,S(z)\ ) = I_3^3(x,y,z+1) = z+1$

x,y – фіктивні аргументи.

Операція примітивної рекурсії

Під рекурсією будемо розуміти спосіб завдання функції, при якому значення деякої функції будуть виражатися через значення цієї функції для менших значень аргументів.

Почнемо з прикладу

Розглянемо 2 функції

$I_1^1(x) = x$ – тотожну (селекторну) функцію і

$h(x,y,z) = z + 1$ - різновид функції безпосереднього слідування.

Зверніть увагу: перша функція – 1-арна , а друга – 3-арна.

Створимо 3-тю функцію із першої і другої, щоб вона залежала від двох аргументів ($f(x,y)$).

Побудуємо її так:

$f(x,0) =I(x) = x$

$f(x,1) =h(x,0,f(x,0)) = h(x,0,x) = x+1$

$f(x,2) =h(x,1,f(x,1)) = h(x,1,x+1) = x+2$

$f(x,y-1) =h(x,y-2,f(x,y-2)) = h(x,y-2,x+y-2) = x+y-1$

$f(x,y) =h(x,y-1,f(x,y-1)) = h(x,y-1,x+y-1) = x+y$

Операція примітивної рекурсії   дає змогу визначити значення функції $f(x,y)$ через значення цієї функції для менших значень аргументу.

Так за допомогою операції примітивної рекурсії і двох елементарних функцій, ми одержали функцію, що визначає суму двох чисел. $f(x,y) = x+y$.

Тепер можна дати визначення операції примітивної рекурсії в загальному вигляді.

Зауважимо, що операція примітивної рекурсії дозволяє будувати функцію від n+1 аргументу (будемо казати n+1–арну функцію) з двох заданих функцій, одна з яких n–арна, а друга n+2–арна.

Нехай дано довільні часткові функції:
n- арна функція g і n+2–арна функція h. Говорять, що n+1–арна функція f одержана операцією примітивної рекурсії з функцій g, h якщо для всіх $x_1, x_2, \ldots, x_n \in N$ маємо
$f(x_1, \ldots, x_n, 0) = g(x_1, \ldots, x_n)$,
$f(x_1, \ldots, x_n, y+1) = h(x_1, \ldots, x_n, y, f(x_1, \ldots, x_n, y))$.

Означення

Примітивно рекурсивними функціями (ПРФ) називаються такі функції, які можна побудувати з елементарних арифметичних функцій за допомогою операції суперпозиції та примітивної рекурсії застосованих скінчене число разів у довільному порядку.

Тобто функції $\left.\begin{array}{l} f(x, y) = x+y \\ f(x, y) = xy \\ f(x) = x \div 1 \end{array}\right\}$ ПРФ

$$x \div 1 = \begin{cases} x-1, & x \geq 1 \\ 0, & x = 0 \end{cases}$$

Операція мінімізації (операція найменшого кореня). Ця операція дозволяє будувати нову арифметичну n- арну функцію $f(x_1, \ldots, x_n)$ з заданої (n+1)- арної функції $g(x_1, x_2 \ldots, x_n, y)$.
Цю операцію мінімізації, або найменшого кореня розглянемо спочатку на прикладі.

Приклад

Нехай задана функція $g(x) = x^2$ - ПРФ

Допустимі значення аргументу x: 0, 1, 2,…

Введемо допоміжний аргумент y і розглянемо рівняння $g(y) = x$, тобто $y^2 = x$
Зафіксуємо допустимі значення основного аргументу x у таблиці 1.

Таблиця 1

x	Рівняння	Корінь рівняння $\mu_y(y^2 = x)$	f(x) побудована функція
0	$y^2 = 0$	0	$0 \to 0$
1	$y^2 = 1$	1	$1 \to 1$
2	$y^2 = 2$	нема	$2 \to$ не визначене
3	$y^2 = 3$	нема	$3 \to$ не виз-не

Отримане значення кореня рівняння будемо приймати за значення функції, яку ми будуємо.
Це значення функції відповідає зафіксованому значенню x

4	$y^2 = 4$	2	$4 \to 2$
5	$y^2 = 5$	нема	$5 \to$ не виз-не
6	$y^2 = 6$	…	…
7	$y^2 = 7$	нема	$7 \to$ не виз-не
8	$y^2 = 8$	нема	$8 \to$ не виз-не
9	$y^2 = 9$	3	$9 \to 3$
…	…	…	…

Аналітичний вираз побудованої функції f(x)

$$f(x) = \begin{cases} \sqrt{x}, \text{ для x, які є повним квадратом числа } x=k^2, \text{ де } k \in N_0; \\ \text{не визначена, для інших допустимих значень x.} \end{cases}$$

Формальне визначення операції мінімізації

Нехай $g(x_1, x_2 \ldots, x_{n-1}, x_n)$ n- арна часткова функція.

Зафіксуємо $a_1, a_2 \ldots, a_{n-1}, a_n$ – набір допустимих значень змінних, де $x_1= a_1$, $x_2 = a_2, \ldots, x_{n-1}= a_{n-1}$, $x_n= a_n$, $a_i \in N_0 (1 \le i \le n)$.

Розглянемо рівняння відносно змінної y

$$g(a_1, a_2 \ldots, a_{n-1}, y) = a_n \qquad (*)$$

Якщо існує розв'язок (корінь) цього рівняння, то позначимо через μ_y – найменший натуральний корінь цього рівняння.

Якщо

$g(a_1, a_2 \ldots, a_{n-1}, 0)$, $g(a_1, a_2 \ldots, a_{n-1}, 1)$ $g(a_1, a_2 \ldots, a_{n-1}, \mu_y- 1)$ (**)

визначені,тоді функція f є результатом виконання операції мінімізації і визначається так

$$f (a_1, a_2 \ldots, a_{n-1}, a_n) = \mu_y$$

Щоб підкреслити, що функція f отримана з функції g шляхом виконання операції мінімізації, використовується таке позначення:

$$f(x_1, x_2 \ldots, x_n) = \mu_y(g(x_1, x_2 \ldots, x_n, y) = x_n)$$

Якщо ж рівняння (*) не має кореня, або хоч би одне із значень, яке позначене (**) не визначене, то значення функції $f (a_1, a_2 \ldots, a_{n-1}, a_n)$ теж не визначене. При цьому вважається, що сама операція мінімізації закінчується без результату.

Функції, які будуються з елементарних арифметичних функцій за допомогою скінченого числа операцій підстановки, примітивної рекурсії та мінімізації, називаються **частково рекурсивними функціями (ЧРФ).**

Нагадаємо, що

ПРФ- це функції, які можна побудувати з елементарних арифметичних функцій за допомогою операції суперпозиції та примітивної рекурсії, застосованих скінченне число разів у довільному порядку.

$\boxed{\text{ЧРФ}}$ $\boxed{\text{ПРФ}}$

Чи рівні ці множини функцій між собою? ПРФ ≠ ЧРФ. Яка з цих множин являється підмножиною іншої?

$$\text{ПРФ} \subset \text{ЧРФ}$$

Кожна з примітивно рекурсивних функцій буде частково рекурсивною

А обернене твердження справедливе? Кожна з ЧРФ $\in$ ПРФ -? Ні
Існують ЧРФ, які не є ПРФ.

Дійсно, ПРФ - функції всюди визначені на відміну від ЧРФ.

Приклад

$g(x) = S(x) = x + 1 \rightarrow f(x) = \mu_y(S(y) = x) = \mu_y(y + 1) = x$

S(x) всюди визначена;

Частково визначена

$$\mu_y(S(y) = x) = \begin{cases} x - 1, x > 0 \\ \text{не визначена, при } x = 0. \end{cases}$$

Отже $\mu_y(S(y) = x)$ не визначена при $x = 0$ і вона не буде ПРФ ф-цією.

Приклад

$g(x) = 5^x$, $x \in N_0$

$g(y) = x$

$5^y = x$

x	Рівняння $5^y = x$	Корінь рівняння $\mu_y(5^y = x)$	Функція f(x)
0	$5^y = 0$	не має	$0 \rightarrow$ не визначена
1	$5^y = 1$	0	$1 \rightarrow 0$
2	$5^y = 2$	не має	не визначена
3	…	…	…
4	…	…	…
5	$5^y = 5$	1	$5 \rightarrow 1$
6	…	не має	не визначена
…	…	…	…
10	…	…	…
5^2	$5^y = 25$	2	$25 \rightarrow 2$
…	………	…	…..

Аналітичний вираз побудованої функції f(x)

$$f(x) = \begin{cases} \log_5 x, \ x = 5^n, \ n \in N_0 \\ \text{не визначена } x \neq 5^n \end{cases}$$

$$g(x) = x+1, f(x) = \mu_y(y+1 = x) = \begin{cases} x-1, x > 0 \\ \text{не визначена } x = 0 \end{cases}$$

$$g(x) = 7x, f(x) = \mu_y(7y = x) = \begin{cases} x/7, \ x = 7k, \ k \in N_0 \\ \text{не визначена} \end{cases}$$

Якщо функція g(x), з якої буде будуватися функція операцією мінімізації, одномісна, 1- арна, то функція $f(x) = \mu_y(g(y) = x)$ буде оберненою до $g(x)$.

Тобто, якщо необхідно побудувати ЧРФ з функції $g(x) = x^3$, то

$$g^{-1}(x) = f(x) = \mu_y(g(y) = x) = \mu_y(y^3 = x) = \begin{cases} \sqrt[3]{x}, & x = k^3,\ k \in N_0 \\ \text{не визначена} & x \neq k^3,\ k \in N_0 \end{cases}$$

Але все одно, хоч і не так просто, як для одномісних функцій, 2-х, 3-х, n-місні частково рекурсивні функції можна будувати за чіткою схемою мінімізації.

Підсумок

Ми познайомились з класом частково рекурсивних ф-цій. Поняття ЧРФ є одним із головних понять теорії алгоритмів.

ЧРФ представляють собою найбільш загальний клас конструктивно створюваних арифметичних ф-цій.

Всі ЧРФ можуть бути обчислені шляхом деякого алгоритму.

Так алгоритм, який супроводить операцію примітивної рекурсії, ми записували цілком чітко.

Алгоритм, який супроводить операцію мінімізації можна коротко сформулювати так.

Ввести додатковий аргумент і розглянути рівняння

$$g(x_1, \ldots x_{n-1}, y) = x_n$$

Зафіксувати допустимі значення аргументів $x_1=a_1, \ldots, x_n=a_n$.

Надавати додатковому аргументу послідовні значення починаючи з 0, до тих пір, поки не виконається рівність

$$g(x_1, \ldots x_{n-1}, \mu) = x_n$$

Отримане значення додаткового аргументу μ прийняти за значення функції f, яке відповідає значенням зафіксованих аргументів. Тобто

$$f(a_1, \ldots a_n,) = \mu$$

У противному випадку вважати, що операція мінімізації закінчується без результату, а функція f не визначена для даних значень аргументів. (Тобто якщо процедура обчислення значень функції працює нескінченно, не видаючи ніякого результату, то функція f - не визначена).

На прикладах алгоритмів, що супроводять операції примітивної рекурсії й мінімізації, ми бачимо, що частково рекурсивні функції обчислюються за деякими алгоритмами, тобто є обчислюваними функціями.

Поняття обчислюваної функції у теорії алгоритмів існує.

Обчислюваними функціями називаються числові функції, значення яких можна обчислити шляхом деякого алгоритму.

Поняття алгоритму у даному визначенні інтуїтивне, значить і поняття обчислюваної функції виявляється теж інтуїтивним.

На відміну від цих інтуїтивних понять, поняття частково рекурсивної функції є точним.

А як же пов'язані поняття алгоритму, обчислюваної функції і частково рекурсивної функції ?

Ми пам’ятаємо, що теорія рекурсивних функцій з’явилась у зв’язку з необхідністю формалізації поняття алгоритму.
Дійсно ЧРФ відповідає нашому інтуїтивному уявленню про алгоритм.
А навпаки ? З практики алгоритмізації виявляється, які б класи чітко окреслених алгоритмів до сих пір не будувались, виявлялось, що числові функції обчислювані шляхом цих алгоритмів, були частково рекурсивними.
Тому у 1936 р. американський математик А.Черч висунув таку тезу:

Теза Черча

Клас алгоритмічно (або машино) обчислюваних часткових числових функцій співпадає з класом усіх частково рекурсивних функцій.

Теза Черча являється гіпотезою. Вона зв’язує неточне інтуїтивне поняття алгоритму, обчислюваної функції з точним математичним поняттям частково рекурсивної функції.
Розмитість тези не дозволяє довести її, але на користь її справедливості говорять багато чітко доведених тверджень.
Тобто є достатньо вагомих підстав прийняти тезу Черча як основу теорії алгоритмів.

6.1.2. Контрольні питання

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.
10. Взяття обмеженої суми, обмеженого добутку.
11. Що таке примітивно рекурсивні функції
12. Що таке частково рекурсивні функції .
13. Теза Черча.

6.2. Нормальні алгоритми Маркова

6.2.1. Теоретичні поняття і визначення

Інший підхід до уточнення інтуїтивного поняття алгоритму зробив російський математик Марков (1903 – 1979 р.). Він висловив думку про можливість запису будь-яких алгоритмів в термінах слів і словникових функцій, які одним словам ставлять у відповідність інші слова деякого алфавіту.

Нагадуємо, що алгоритм – це сукупність правил, які визначають процес переробки допустимих початкових даних у вихідні результати.

Марков розглядав початкові дані і вихідні результати у вигляді слів, які складаються із символів деякого скінченого алфавіту.

Ідею нормального алгоритму Маркова (НАМ) розглянемо на прикладі таблиці слів, що перетворюються за допомогою Марковських підстановок.

	Перетворюване слово	Марковська підстановка	Результат
1	245	$4 \rightarrow 00$	2005
2	функція	функція $\rightarrow$ tg x	tg x
3	телефон	теле $\rightarrow$	фон
4	екран	теле $\rightarrow$ (цю підстановку не можна застосовувати до слова екран)	результату немає

Поняття алфавіту

Абстрактним алфавітом називається будь-яка скінчена сукупність символів, а буквами алфавіту - символи цієї сукупності.

Наприклад:

- український алфавіт A = {а, б, …..я};
- алфавіт A = {0, 1, …..9} - алфавіт, що складається з цифр, де цифри є буквами цього алфавіту;
- A = {I, *} - алфавіт, що складається з двох символів "риска" і "зірочка", які є буквами цього алфавіту.

Кількість букв в алфавіті називають його обсягом.

Слово в алфавіті – це кожна впорядкована сукупність букв даного алфавіту (ab, ba – різні слова)

Довжина слова – це число букв у слові.

Слово P = xy має довжину 2.

Слово, що не містить жодної букви називається пустим, його довжина 0.

Нехай є такий алфавіт A = {x, y, z}, слово Q = xz має два входження у слово S, S = xy(xz)yyxx(xz)yx.

Особливий інтерес для нас має перше входження (рахуючи зліва-направо) слова Q у слово S.

Ми можемо його замінити на інше слово T, T = zzz, тобто $Q \rightarrow T$, тоді отримаємо слово S_1 = xyzzzyyxx(xz)yx.

Визначення операції заміни одного слова іншим

Нехай A - деякий алфавіт і P,Q – слова в цьому алфавіті. Припустимо, що серед букв алфавіту A немає букви $\rightarrow$.

Вираз виду $P \rightarrow Q$ називатимемо підстановкою або формулою підстановки, а слова P,Q – лівою й правою частиною підстановки, відповідно.

За змістом <u>підстановку</u> треба розуміти як <u>правило перетворення</u> слів в алфавіті А, сформульоване так: "Якщо в слово S входить слово Р, то слово Р замінити словом Q. Якщо ж у слово S не входить слово Р, то залишити слово S незмінним".

Якщо ліва частина підстановки $P \to Q$ входить у слово S, то її називають <u>підстановкою застосовною до даного слова S.</u>
У противному випадку підстановка $P \to Q$ вважається <u>незастосовною до даного слова S</u>.

<u>Види підстановок</u>

Прості підстановки:

- $P \to Q$ — якщо в лівій і правій частині підстановки не пусте слово
- $\to Q$ — якщо в лівій частині підстановки пусте слово
- $P \to$ — якщо в правій частині підстановки пусте слово

Заключна підстановка: $P \to \bullet\, Q$

Нехай вираз $P \to [\bullet]\, Q$ означає будь-яку з формул підстановки:

$$P \to Q$$
$$\to Q$$
$$P \to$$
$$P \to \bullet\, Q$$

Тепер введемо поняття схеми підстановок

<u>Нормальна схема в алфавіті А</u> – це скінчена непуста впорядкована сукупність підстановок

$$\left\{\begin{array}{l} P_1 \to [\bullet]\, Q_1 \\ P_2 \to [\bullet]\, Q_2 \\ \ldots\ldots \\ P_k \to [\bullet]\, Q_k \end{array}\right.$$

<u>Нормальний алгоритм Маркова</u> задається нормальною схемою в алфавіті А. Підстановки схеми застосовуються до слова відповідно до таких правил:

а) перевірку застосовності підстановок до перетворюваного слова S на будь-якому етапі переробки слова треба починати з першої підстановки;

б) якщо вона застосовна до слова S, то застосовувати її треба до першого входження її лівої частини у слово S.

Якщо жодна з перших і підстановок $1 \le i < k$ незастосовна до S, а $(i + 1)$ підстановка застосовна, то застосовується ця підстановка знов-таки до першого входження її лівої частини в перетворюване слово;

в) процес перетворення слова продовжується доти, поки не дістанемо слово, до якого жодна з підстановок сукупності не застосовна або поки до слова не буде застосована заключна підстановка.

Приклад
Нормальний алгоритм Маркова задано схемою:

$$\begin{cases} bab \to aa \\ aa \to b \\ bb \to a \end{cases}$$

Знайти результат дії цього алгоритму на слово S = abaaabb.

Крок алгоритму	Перетворюване слово	Застосовна підстановка	Результат
1	abaaabb	aa → b	abbabb
2	abbabb	bab → aa	abaab
3	abaab	aa → b	abbb
4	abbb	bb → a	aab
5	aab	aa → b	bb
6	bb	bb → a	a

В результаті застосування алгоритму перетворення до слова S отримали S_1= a.

Якщо процес перетворення слова S закінчується після скінченого числа застосувань підстановок схеми алгоритму, то алгоритм називається застосовним до слова S.

Коли ж процес перетворення слова S не може закінчитися, то алгоритм називається незастосовним до слова S.

Якщо даний алгоритм позначити буквою f , то f (abaaabb) = a.

Приклад нормального алгоритму Маркова , заданого алфавітом
A= {I, +} і схемою алгоритму:

$$ff \begin{cases} I + I \to + II \\ + I \to I \end{cases}$$

Даний НА виконує додавання 2-х натуральних чисел в унарній системі числення.
Застосуємо алгоритм до слова P = III + II

Крок алгоритму	Перетворюване слово	Підстановка	Результат
1	III + II	I + I → + II	II + III
2	II + III	I + I → + II	I + IIII
3	I + IIII	I + I → + II	+ IIIII
4	+ IIIII	+ I → • I	IIIII

f (III + II) = IIIII.

Ми на цих прикладах бачимо, що НАМ можна розглядати як функцію, яка одним словам ставить у відповідність інші слова деякого алфавіту.

Нехай А – деякий скінчений алфавіт,
S(A) – множина слів в алфавіті А,

D_f - множина слів, які належать S(A) та до яких НАМ f є застосовним.

Тоді НАМ f визначає словникову функцію f : $S(A) \to S(A)$, причому область визначення цієї функції є $D_f \le S(A)$,

тобто, якщо слово P належать D_f, то нормальний алгоритм f визначає слово Q, як Q = f(P).

Процес знаходження слова Q називають обчисленням. Тобто НАМ обчислює словникову функцію, а функція f називається обчислюваною за Марковим, або НАМ-обчислюваною.

Нехай А і В два алфавіти. Якщо кожна буква, що входить до алфавіту А є буквою алфавіту В, то говорять, що алфавіт В є розширенням алфавіту А, а А є частиною В.

Приклад Нехай А = {0, 1, 2, 3, …, 8, 9} і В – розширення А
В = {0, 1, 2, 3, …, 8, 9, x, y }

Розглянемо НАМ над А , тобто НА у розширенні алфавіту А (тобто з використанням ще букв x, y).

НАМ для обчислення функції S(x) = x + 1, коли x записано в десятковій системі числення. НА запишемо у вигляді 3-х стовпчиків формул підстановок

$x0 \to 0x$	$0x \to \bullet 1$	$0y \to \bullet 1$
$x1 \to 1x$	$1x \to \bullet 2$	$1y \to \bullet 2$
$x2 \to 2x$	$2x \to \bullet 3$	$2y \to \bullet 3$
$x3 \to 3x$	$3x \to \bullet 4$	$3y \to \bullet 4$
$x4 \to 4x$	$4x \to \bullet 5$	$4y \to \bullet 5$
$x5 \to 5x$	$5x \to \bullet 6$	$5y \to \bullet 6$
$x6 \to 6x$	$6x \to \bullet 7$	$6y \to \bullet 7$
$x7 \to 7x$	$7x \to \bullet 8$	$7y \to \bullet 8$
$x8 \to 8x$	$8x \to \bullet 9$	$8y \to \bullet 9$
$x9 \to 9x$	$9x \to y0$	$9y \to y0$
		$y \to \bullet 1$
		$\to x$

Розглянемо поняття еквівалентності 2-х алгоритмів.
Два алгоритми над алфавітом А називаються еквівалентними відносно А, якщо вони мають одну й ту саму область визначення і результати їх застосування до одного й того самого слова збігаються.

Для НАМ висувається гіпотеза, яку не можна довести, але можна прийняти, спираючись на попередній досвід математики й інформатики. Ця гіпотеза називається принципом нормалізації.

Принцип нормалізації

Будь-який алгоритм над скінченим алфавітом А являється еквівалентним відносно А деякому нормальному алгоритму Маркова над А.

Довести принцип нормалізації Маркова, як і гіпотезу Черча не можна, бо у формулювання цієї гіпотези входить інтуїтивне поняття алгоритму. Нормальні алгоритми Маркова - це чітко визначені алгоритми. НАМ приймається як ще одна стандартна форма будь-якого алгоритму.

6.3. Контрольні питання

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.
9. Принцип нормалізації.

Тема 7. Машина Тюрінга

7.1. Теоретичні поняття і визначення

Алгоритмічну систему, в якій правила, що визначають дію алгоритму, побудовані за командно-адресним принципом для деякої машини з необмеженою пам’яттю, називають машиною Тюрінга (МТ).

Розглянемо найпростіший приклад.

Нехай є нескінченна з 2-х боків паперова стрічка, на якій можна писати букви в клітинках. Є головка, яка: може знаходитись в одному із 2-х різних станів,читає і записує букви в клітинки стрічки; може рухатися вліво й вправо(п) вздовж стрічки та стояти(с).

Складемо таблицю, яка описує машину Тюрінга для обчислення функції S(x) = x + 1 в алфавіті A={I}. # - спеціальний пустий символ.

Символ {!} означає зупинку МТ.

	q_0	q_1
#	Iq_1c	
I	$Iq_0п$	!

#	I	I	#

q_0

#	I	I	#

q_0

#	I	I	#	

q_0

#	I	I	I	#

q_1

x = II   початкове число

$Iq_0 \rightarrow Iq_0п$

$IIq_0 \rightarrow Iq_0п$

□$q_0 \rightarrow Iq_1c$

$Iq_1 \rightarrow I$

S = III   результат

<u>Висновок</u>

Яке б число не було записане на стрічці МТ, головка буде рухатися вправо до пустої клітинки, запише в пусту клітинку I і зупиниться. На стрічці буде записано число x + 1, замість x. Тобто буде обчислена функція S(x) = x +1.

Тепер перейдемо до формального визначення Машини Тюрінга (МТ).

МТ запропонована англійським математиком А. Тюрінгом в 1937 р. Це алгоритмічна система, яка визначає загальний способів завдання алгоритмів.

1. У цій системі виконується запис інформації на нескінченній в обидва боки стрічці, розбитій на клітини. У клітинку можна записати тільки одну букву скінченого алфавіту $A = \{a_1, \ldots a_n\}$.

2. Цю букву можна оглядати (читати, записувати) спеціальним чутливим елементом – так званою головкою МТ.

Головка МТ може знаходитись в одному із скінченої кількості різних станів:

$$Q = \{q_0, q_1, \ldots q_m\}$$

Головка може записувати в клітинці, яку оглядає, будь-яку букву з алфавіту А і залишатись на місці або переміщуватись праворуч чи ліворуч уздовж інформаційної стрічки на одну клітинку.

Задати МТ означає задати таку п'ятірку об'єктів:

$A, Q, q_0, \#, P$, де

A – алфавіт машини (зовнішній алфавіт);

Q – скінчена непуста множина станів машини, $(A \cap Q = \varnothing)$;

q_0 - початковий стан МТ;

\# - спеціальний пустий символ $(\# \notin A \cup Q)$;

P - програма роботи МТ;

МТ виконує команди.

Команда має вигляд $aq \to bq'd$ $(aqbq'd)$

$a, b, \in A$

$q', q' \in Q$

$d = \{S, L, R\}$.

Програма МТ – це скінчена множина команд, причому не існує таких 2-х команд, які б мали однакові перші 2 символи.

Як виконується команда $aq \to bq'd$?

Головка МТ:

- знаходиться в стані q;
- читає записану в клітинці букву a;
- записує в цю клітинку нову букву b (яка може збігатися з буквою a);
- переходить у стан q′(який може збігатися зі станом q) і переміщується вздовж стрічки на величину d.

Якщо d = S, то головка залишається на місці;

……. d = R, то відбувається зсув головки вправо на 1 клітинку;

……. d = L, то відбувається зсув головки вліво на 1 клітинку.

МТ зупиняється тоді і тільки тоді, коли жодна з команд програми не може застосовуватись або вказується символ {!}, що означає зупинку МТ.

Результатом роботи МТ після її зупинки є слово, записане на стрічці. Це слово називається <u>заключним</u>.

В цьому випадку МТ вважається <u>застосовною</u> до початкового слова.

Якщо МТ обробляє початкове слово нескінченно довго, то МТ. вважається <u>не застосовною</u> до початкового слова.

Функціонування МТ можна описати за допомогою протоколу роботи над заданим початковим словом.

Нехай:

1) $\#a_1, \ldots a_k a_{k+1}, \ldots a_k\#$ - слово, що виникає на стрічці в процесі роботи МТ.
2) МТ знаходиться в стані q.
3) Головка стоїть проти k-го символу слова.

Слово $\#a_1 \ldots a_k q a_{k+1} \ldots a_m\#$ ($\#a_1 \ldots a_k \underset{q}{a_{k+1}} \ldots a_m\#$) називається <u>конфігурацією</u>.

Послідовність конфігурацій при роботі МТ називається <u>протоколом</u> МТ.

Розглянемо приклад.

Створити МТ, яка всяке натуральне число n, написане в двійковій системі перетворює на число n + 1 теж записане в 2-ковій системі. Тобто реалізує S(x) = x + 1 обчислення функції безпосереднього слідування.

Програму можна записувати: - у вигляді послідовності команд;

- у вигляді спеціальної таблиці.

	Стан МТ		
Символ	q_0	q_1	q_2
#	$\#q_1L$	$1q_2L$	
0	$0q_0R$	$1q_2L$	
1	$1q_0R$	$0q_1L$	!

Протокол МТ

1) $\# \underset{q_0}{1}\ 0\ \#$ ← початкова конфігурація

2) $\# 1\ \underset{q_0}{0}\ \#$

3) $\# 1\ 0\ \underset{q_0}{\#}$

4) $\# 1\ \underset{q_1}{0}\ \#$

5) $\underset{q_2}{\#}\ 1\ 0\ \#$ ← заключна конфігурація

Початкове слово w = 10

Результат $w_1 = 11$.

З кожною МТ пов'язана часткова функція (словесна функція)
$f_{мт}: S(A) \to S(A)$, де $S(A)$ – множина слів в алфавіті А.

1) Якщо МТ зупиняється при роботі над початковим словом w, то значення функції визначене і $f_{мт}(w) = w'$, де w' - заключне слово на стрічці МТ.
2) Якщо МТ не зупиняється, значення функції не визначене на слові w.

Нехай задана часткова функція f на множині S(A).
Ця функція називається обчислюваною за Тюрінгом, якщо існує МТ така, що :

1) Зовнішній алфавіт МТ співпадає з А.
2) $f(w) \equiv f_{мт}(w), \forall(w) \in S(A)$

Приклад 1
Наведемо приклад МТ, яка незастосовна до будь-якого не пустого слова в алфавіті А = {I}

#	I	#

	Стан МТ	
Символ	q_0	
#	Iq_0L	
I	Iq_0L	

1) # I #
q_0
2) # I #
q_0
3) # I I #
q_0
4) # I I I #
q_0 Протокол МТ нескінченний.

Дія цієї МТ полягає в тому, що, почавши роботу з будь-якого непустого слова w, до цього слова приписується зліва символ I. Машина не зупиняється. Результат роботи не визначений, оскільки машина не застосовна до будь-якого непустого слова $w \in S(A)$.
Приклад 2
МТ, яка реалізує алгоритм додавання 2-х натуральних чисел в алфавіті А = {I}, В = {I, +}
Ідея алгоритму

#	I	+	I	#

1. Кожну I 1-го доданку стерти й записати її в кінці 2-го доданку.
2 Стерти знак +.

	Стан МТ			
Символ	q_0	q_1	q_2	q_3
#	$\#q_0 R$	$Iq_2 S$	$\#q_0 S$	!
I	$\#q_1 R$	$Iq_1 R$	$Iq_2 L$	
+	$\#q_3 S$	$+q_1 R$	$+q_2 L$	

P = I + I

1) # I + I # q_0	2) # # + I # q_1	3) # + I # q_1
4) # + I # q_1	5) # + I I # q_2	6) # + I I # q_2
7) # + I I # q_2	8) # + I I # q_2	9) # + I I # q_0
10) # + I I # q_0	11) # I I # q_3	12) # I I # q_3
	f_{MT}(I+I) = II	P_1 = I I

7.2. Суперпозиція машин Тюрінга

З математичної точки зору МТ – це просто визначений алгоритм для переробки слів.

Кожна МТ забезпечує знаходження значення деякої словникової функції в алфавіті A

f_{MT}: S(A), де S(A) – множина слів A.

Тому природно ввести операцію суперпозиції МТ. Ця операція суперпозиції МТ буде еквівалентна суперпозиції функцій, що представляють ці МТ.

Нехай МТ M_1 обчислює функцію $f_1(w)$ в алфавіті A. A = $\{a_1, \ldots, a_n\}$
M_2 обчислює функцію $f_2(w)$ в алфавіті A.
Значення суперпозиції $f_2(f_1(w))$ визначене тоді і тільки тоді, коли визначена ф-ція $f_1(w)$ і визначено значення функції f_2 на слові $f_1(w)$.
Побудуємо МТ M, яка буде суперпозицією машин M_1 і M_2 і обчислює суперпозицію $f_2(f_1(p))$.
Програма для обчислення суперпозиції $f_2(f_1(p))$ являється об'єднанням програм для обчислення функцій f_1 і f_2.
Початковий стан МТ M_2 повинен збігатися із заключним станом МТ M_1. Усі стани M_2, крім початкового, повинні відрізнятися від станів M_1.

Після перетворення програми M_2, програми M_1 та M_2 об'єднуються в одну таблицю-програму МТ M.

Початковим станом M стає початковий стан M_1 Заключним станом M стає заключний стан M_2

Суперпозицію M машин M_1 і M_2 будемо позначати $M_2 \circ M_1$.
Як змінити стан МТ M_2 ? Треба всі номери станів крім початкового збільшити на число, що збігається з номером заключного стану M_1

Можливість будувати МТ за допомогою суперпозиції, розгалуження, циклів приводять до думки, що мова тюрінгівського програмування дозволяє реалізувати будь-який алгоритм (в інтуїтивному розумінні).

<u>Теза Тюрінга</u>

Класи обчислюваних і обчислюваних за Тюрінгом функцій збігаються.

Довести тезу Тюрінга, як і тезу Черча, неможливо, оскільки саме поняття алгоритму (в інтуїтивному розумінні) є неточним. Це не теорема і не постулат математичної теорії. Це твердження, яке пов'язує теорію з тими об'єктами, для опису яких її створено.

Підтвердженням тези Тюрінга є,

- по перше математична практика;
- по друге, той факт, що опис алгоритму у термінах будь-якої іншої відомої алгоритмічної моделі може бути зведений до його опису у вигляді МТ.

7.3. Еквівалентність алгоритмічних систем

Алгоритмічні системи (машини Тюрінга, нормальні алгоритми Маркова та частково рекурсивні функції)приводять до точних формальних визначень обчислюваної функції (за Марковим, за Тюрінгом, ЧРФ), а значить уточнюють поняття алгоритму.

Зупинимось на спільних рисах даних алгоритмічних систем.

Спільні риси даних алгоритмічних систем

1. Спосіб реалізації обчислюваної функції задається скінченним числом елементарних інструкцій, тобто програмою

(для ЧРФ - це рекурсивна схема,
для НАМ – це нормальна схема,
для МТ – це програма МТ).

2. Обчислення значень функції за заданою програмою виконується за чіткими правилами

(для ЧРФ – це чіткі правила обчислення з використанням операцій суперпозиції, примітивної рекурсії, мінімізації,
для НАМ – це правила виконання підстановок
для МТ – це виконання кроків МТ за програмою.

3. В кожній алгоритмічній системі вказується єдиний спосіб представлення вхідної і вихідної інформації (тобто значень аргументів і значень функцій).

Ці спільні риси алгоритмічних систем наштовхують на думку, що дані алгоритмічні системи можуть служити точним формальним визначенням інтуїтивного поняття алгоритму.

Справедливе наступне твердження.

Класи ЧРФ, функцій обчислюваних за Тюрінгом, функцій обчислюваних за Марковим співпадають між собою.

Тобто алгоритмічні системи ЧРФ, НАМ, МТ – еквівалентні. Це значить, що наукові результати отримані за допомогою алгоритмів однієї з алгоритмічних систем можуть бути отримані і в рамках іншої алгоритмічної системи. Деякі задачі краще, легше розв'язувати в рамках однієї алгоритмічної системи, а деякі в рамках іншої. Не можна відмовлятися від жодної з них.

7.4. Повнота і адекватність формалізованого числення предикатів

Спираючись на теорему Геделя про існування моделі, можна довести теорему, обернену до теореми виправданості, тобто твердження про те, що *із семантичної вивідності випливає її синтаксична вивідність.*

Дійсно, нехай $\Phi \models F$, Покажемо, що тоді множина формул $\Phi \cup \{\neg F\}$ суперечлива. Припустимо, що це не так. Тоді ця множина має модель M, тобто на M виконується формула $\neg F$ і всі формули із Φ. Із останнього, виходячи з умови $\Phi \vdash F$, випливає, що на M виконується й F. Одержано суперечність. Отже, множина $\Phi \cup \{\neg F\}$ суперечлива, а тому із неї вивідна довільна формула, зокрема $\Phi \cup \{\neg F\} \vdash F$. Тоді за теоремою дедукції маємо $\Phi \vdash \{\neg F\} \to F$. Враховуючи, що, крім того, формула $(\neg F \to \neg F) \to \ \to F$ є теоремою формалізованого числення висловлень, за правилом MP робимо висновок, що $\Phi \vdash F$. Тобто, ми показали, що коли $\Phi \models F$, то $\Phi \vdash F$.

Враховуючи дане твердження й попередні результати, можна сформулювати наступну теорему.

Теорема (*теорема адекватності*). Формула F синтаксично вивідна із множини формул Ф тоді й тільки тоді, коли вона семантично вивідна із Φ: $\Phi \vdash F \Leftrightarrow \Phi \models F$.

Теорема (*теорема Геделя про повноту формалізованого числення предикатів*). Клас довідних замкнених формул збігається з класом загальнозначущих (тотожно істинних) формул: $\vdash F \Leftrightarrow \models F$.

Ця теорема безпосередньо випливає із попередньої теореми, якщо покласти $\Phi = \varnothing$,

Вона справедлива й для відкритих формул. Дійсно, якщо $\models F(x_1, \ldots, x_n)$, де $x_1, \ldots, x_n$ – вільні предметні змінні в формулі F, то за означенням квантора загальності, маємо $\models \forall x_1 \ldots \forall x_n F(x_1, \ldots, x_n)$, а це, в свою чергу, рівносильна $\vdash \forall x_1 \ldots \forall x_n F(x_1, \ldots, x_n)$. За властивостями вивідності останнє рівносильна тому, що $\vdash F(x_1, \ldots, x_n)$.

7.5. Неповнота формалізованого числення предикатів в абсолютному й вузькому розумінні

Теорему Геделя про повноту формалізованого числення предикатів можна трактувати як деяку зовнішню повноту формалізованого числення предикатів, його повноту стосовно логіки предикатів: у цій теорії можуть бути формально доведені всі загальнозначущі формули логіки предикатів.

Розглянемо тепер питання внутрішньої повноти формалізованого числення предикатів, тобто з’ясуємо чи буде ця теорія абсолютно повною й повною у вузькому розумінні. Оскільки множина теорем формалізованого числення предикатів збігається із множиною тавтологій (загальнозначущих формул) логіки предикатів, а в логіці предикатів існують виконувані, але не загальнозначущі формули, то формалізоване числення предикатів не є абсолютно повною теорією. Що ж стосується повноти формалізованого числення предикатів у вузькому розумінні, то числення предикатів (на відміну від числення висловлень) такої властивість не має. Для доведення наведемо приклад формули, яка не є теоремою формалізованого числення предикатів, додавання якої до аксіом числення предикатів (із збереженням правил виведення) приводить до несуперечливої формалізованої аксіоматичної теорії.

Розглянемо формулу $\exists x F(x) \rightarrow \forall x F(x)$. Ця формула не є загальнозначущою. Тому, за теоремою Геделя про повноту вона не може бути доведена в формалізованому численні предикатів. З іншого боку, додавши до аксіом формалізованого числення предикатів цю формулу, одержимо несуперечливу формалізовану теорію T. Її несуперечливість можна довести наступним чином.

Розглянемо модель цієї теорії на одноелементній множині $M=\{a\}$. Дана формула тотожно істинна на М. Враховуючи, що на М можна визначити для кожного натурального n лише два n- місні (n- арні) предикати P_1^n і P_2^n причому $\lambda\left(P_1^n(a, \ldots, a)\right)=0$ і $\lambda\left(P_2^n(a, \ldots, a)\right)=1$, неважко показати, що всі аксіоми нової теорії T тотожно істинні на даній моделі й правила виведення від тотожно істинних на М формул приводять до тотожно істинних на М формул. Таким чином, доведено твердження: будь-яка теорія T тотожно істина на одноелементній множині М. Отже, якби для деякої формули F обидві формули F і $\neg F$ були теоремами теорії T, то вони були б тотожно істинними на одноелементній множині М, що неможливо. Тому розширена теорія T несуперечлива, що й доводить неповноту у вузькому розумінні формалізованого числення предикатів.

7.6. Формальна арифметика

Формальна арифметика – це логіко-математичне числення, котре формалізує елементарну теорію чисел. Мова цього числення крім логічних зв’язок і рівностей містить нелогічну константу 0Ю двомісні функціональні символи + і, одномісний функціональний символ $'$. Терми будуються із константи 0 і змінних за допомогою функціональних символів. Зокрема, натуральні числа зображуються термами виду $0'''^{\cdots}{}'$. Атомарні формули – це рівність термів, інші формули будуються із атомарних за допомогою логічних зв’язок. У значенні аксіом вибираються логічні аксіоми (аксіоми формалізованого числення предикатів) і наступні нелогічні (арифметичні) формули:

$x = y \rightarrow (x = z \rightarrow y = z), \quad \neg(x'=0);$

$x = y \rightarrow x' = y', \quad x' = y' \rightarrow x = y;$

$x + 0 = x, \quad x + y' = (x + y)';$

$x \cdot 0 = 0,\ x \cdot y' = (x \cdot y) + x;$

$(F(0) \wedge \forall x\, (F(x) \rightarrow F(x'))) \rightarrow \forall x\, F(x),$

де $F(x)$ – довільна формула теорії з однією вільною предметною змінною x. Остання формула це схема аксіом індукції.

Стандартною моделлю арифметики є множина натуральних чисел, в якій $0' = 1$, $x' = x + 1$ і $+$ та $\cdot$ – операції додавання та множення натуральних чисел.

Засоби формальної арифметики є достатніми для виведення теорем, які установлюються в стандартних курсах елементарної теорії чисел.

Формальна арифметика відіграє виключно важливу роль в основах математики. Це пояснюється тим, що якраз арифметика лежить в основах класичної математики, проблема несуперечності якої зводиться до несуперечність арифметики. Ця змістовна сторона знайшла своє найвище вираження й на формальному рівні.

Одним із шляхів виходу з кризи в основах математики на початку XX ст., обумовленого виявом парадоксів в теорії множин (наприклад, парадокс Рассела), повинен був стати гільбертовський шлях формалізації математики й логіки. Кожна конкретна математична теорія повинна бути переведена на мову відповідної формальної системи таким чином, щоб кожне осмислене (хибне чи істинне) речення змістовної теорії виражалось би деякою формулою формальної системи. Тоді природно було б сподіватися, що цей метод формалізації дозволить будувати весь позитивний зміст математичних теорій на такій точній і, здавалось би, надійній основі як поняття вивідної формули (теореми формальної системи). Оскільки формальні системи самі виявляються точними (або, як говорили в школі Гільберта, фінітними) математичними об’єктами, можна було очікувати, що вдасться одержати фінітні доведення тверджень про несуперечність, тобто доведення, які в певному розумінні були б ефективними, незалежними від тих потужних засобів, які в класичних математичних теоріях якраз і є причиною складності в їх обґрунтуванні. Проте результати, одержані Геделем на початку 1930-х рр., розвіяли сподівання, пов’язані з програмою Гільберта. Гедель довів наступні дві теореми, які одержали спільну назву “теореми про неповноту формальної арифметики”.

Теорема (перша теорема Геделя про неповноту формальної арифметики). Будь-яка природна несуперечлива формалізація S арифметики або довільної іншої математичної теорії, котра містить арифметику неповна і недоповнювальна. Неповнота означає, що в S існує змістовно істинна, але нерозв’язна формула, тобто така формула A, що ні A, ні її заперечення $\neg A$ не вивідні в S. Недоповнювальність в S означає, що якою б скінченною множиною додаткових аксіом (наприклад, нерозв’язними в S формулами) не розширити систему S, то в новій формальній системі обов’язково появляться свої нерозв’язні формули.

Теорема (друга теорема Геделя про неповноту формальної арифметики). Якщо формалізована арифметика дійсно несуперечлива, то не зважаючи на те, що твердження про її несуперечність виразимо на її власній мові, доведення цього твердження, засобами, що формалізуються в ній самій, неможливо.

Ця теорема, як і перша, поширюється на будь-яку несуперечливу формальну систему, яка містить формальну арифметику.

Із першої теореми Геделя про неповноту арифметики видно, що семантичне поняття істинності в арифметиці, а отже, і в усій математиці неможливо вичерпно формалізувати через синтаксичне поняття довідності в якій би то не було одній формально логічній системі. Друга теорема Геделя про неповноту арифметики показує, що й основна мета першорядної програми Гільберта про формалізацію математики виявляється недосяжною. Ця мета полягала в тому, щоб довести формальну несуперечність арифметики, користуючись при цьому так званими "фінітними" методами, тобто лише такими методами доведення, які застосовуються в самій арифметиці.

7.7. Контрольні питання

1. Алгоритмічна система Машина Тюрінга (МТ).
2. Поняття функції обчислюваної за Тюрінгом.
3. Суперпозиція МТ.
4. Розгалуження МТ.
5. Теза Тюрінга .
6. Еквівалентність теорій (МТ, нормальних алгоритмів Маркова та ЧРФ).
7. Поняття про алгоритмічно розв'язувані проблеми та алгоритмічно не розв'язувані проблеми.

СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

Основна

1) Клини С. Математическая логика. – Москва: Наука, 1973.
2) Нікольський Ю.В. Дискретна математика: підручник. – Львів: Магнолія-2006, 2010.– 431 с.
3) Міхайленко В.М. Дискретна математика: підручник. – Київ: ЄУ, 2003.– 318 с.
4) Пильщиков В.Н. Машина Тьюринга и алгоритмы Маркова. Решение задач. (Учебно-методическое пособие) - Москва: МГУ, 2006. – 47 с.
5) Стрелковська І.В. Дискретна математика: навч. посіб. − Одеса: ОНАЗ ім. О. С. Попова, 2010. − 196 с.
6) Лісовик Л.П., Шкільняк С.С. Теорія алгоритмів. – ВПЦ Київський університет. – Київ, 2003.
7) Мендельсон Э. Введение в математическую логику. – Москва: Наука, 1976.
8) Нікітченко М.С., Шкільняк С.С. Математична логіка та теорія алгоритмів. – Київ, 2008.
9) Шкільняк С.С. Математична логіка: приклади і задачі. – ВПЦ Київський університет. – Київ, 2007.

Додаткова

1. Капітонова Ю.В., Кривий С.Л., Летичевський О.А. та ін. Основи дискретної математики. – Київ, 2002.
2. Катленд Н. Вычислимость. Введение в теорию рекурсивных функций. – Москва, 1983.
3. Лисовик Л.П., Редько В.Н. Алгоритмы и формальные системы. – Київ, 1981.
4. Мальцев А.И. Алгебраические системы. – Москва: Наука, 1970.
5. Клини С. Введение в метаматематику. – Москва: ИЛ, 1957.
6. Нікітченко М.С., Шкільняк С.С. Основи математичної логіки. – Київ, 2006.
7. Роджерс Х. Теория рекурсивных функций и эффективная вычислимость. – Москва: Мир, 1972.
8. Семантика модальных и интенсиональных логик. – Москва: Прогресс, 1981. – 494 с.
9. Справочная книга по математической логике / Под ред. Дж. Барвайса: В 4 т. – Москва, 1982–1983.
10. Такеути Г. Теория доказательств. – Москва: Мир, 1978.
11. Фейс Р. Модальная логика. – Москва: Мир,, 1974.
12. Чень Ч., Ли Р. Математическая логика и автоматическое доказательство теорем. – Москва, 1983.
13. Шенфилд Дж. Математическая логика. – Москва: Наука, 1975.

14. Смирнова Е.А. Логика и философия. – Москва: РОССПЕН, 1996.
15. Шкільняк С.С. Відношення логічного наслідку в композиційно-номінативних логіках // Пробл. програмування. – 2010. – № 1.
16. Belnap N., Steel T. The logic of questions and answers. – New Haven and London: Yale Univ. Press, 1976.
17. Ішмуратов А.Т. Вступ до філософської логіки. – Київ, 1997.

Навчальне видання

МАТЕМАТИЧНА ЛОГІКА ТА ТЕОРІЯ АЛГОРИТМІВ

Курс лекцій

Укладачі:

Шебаніна Олена В'ячеславівна
Клочан Віра Павлівна
Клочан Ірина Володимирівна та ін.

Формат 60х84 1/16. Ум. друк. арк. 3,334.
Тираж 50 прим. Зам. № __

Надруковано у видавничому відділі
Миколаївського національного аграрного університету
54020, м. Миколаїв, вул. Георгія Гонгадзе, 9

Свідоцтво суб'єкта видавничої справи ДК № 4490 від 20.02.2013 р.