

ДОСЛІДЖЕННЯ ЛОГІЧНИХ ВИРАЗІВ НА ІСТИННІСТЬ

Розроблено і алгоритмізовано метод дослідження логічних виразів алгебри висловлень з використанням ПК.

В алгебрі висловлень розглядають два основні питання відносно логічних формул: рівносильність і тотожню істинність. Вирішують ці питання за допомогою побудови таблиць істинності. Відомо: в цих таблицях кількість наборів значень логічних змінних для знаходження всіх значень істинності формули дорівнює 2^n , де n - кількість логічних змінних. Коли $n > 3$, таблиці громіздкі й з ними незручно працювати. При застосуванні алгебри висловлень до перевірки правильності умовиводів кількість логічних змінних часто буває більше трьох. Тому доцільно автоматизувати процес знаходження значень істинності формул алгебри висловлень.

У роботі [1] розглянуто один із способів обчислення значень аналітичних виразів, що вводяться в працюючу програму з клавіатури. Розглянемо, як можна це ж питання вирішити для логічних виразів. Спочатку проаналізуємо шлях знаходження комп'ютером істинності деякого логічного виразу. Нехай ним буде, наприклад, такий вираз:

$$(A \rightarrow \bar{B}) \vee (BC \leftrightarrow D) \wedge (B \leftrightarrow A\bar{C}) \quad (1)$$

У стандартній таблиці символів набору ПК відсутні спеціальні знаки логічних операцій (типу \vee , \wedge , \rightarrow , \leftrightarrow чи символу заперечення), тому необхідно представити логічний вираз за відповідними правилами запису логічних виразів для мов програмування високого рівня. Виберемо символи для позначення логічних операцій кон'юнкції, диз'юнкції, імплікації, заперечення та еквіваленції - відповідно "*", "+", ">", "!" (перед виразом) та "=".

Враховуючи введені позначення, формулу алгебри висловлень (1) уведемо в комп'ютер у вигляді:

$$"(A > !B) + (BC = D) * (B = A!C)" \quad (2)$$

Отримавши рядок символів (2), ПК повинен зберегти його в деякій літерній змінній Chain, провести аналіз на встановлення синтаксичних помилок ("невідомі" символи, недостача закриваючої або відкриваючої дужки тощо) та видалити символи "ПРОПУСК", якщо вони наявні у записі. Також доцільно перевести рядок у верхній регістр.

Часто логічні змінні записуються без знаку операції між ними, наприклад, "BC". Відомо, що такий запис відповідає запису "B*C", тому для визначеності потрібно все ж таки вставити символ кон'юнкції "*" між символами змінних: "B*C". З урахуванням цього для рядка (2) одержуємо

$$"(A > !B) + (B * C = D) * (B = A * !C)" \quad (3)$$

Слід встановити, скільки логічних змінних задіяно в поданому логічному виразі. Оскільки в рядку (3) зустрічаються лише латинські літери "A", "B", "C" та "D", то відповідно маємо чотири логічні змінні, яким необхідно надати конкретні значення перед обчисленням логічного виразу. Нехай, наприклад, необхідно обчислити означений вираз при наступних (вибраних довільно) значеннях змінних: A = "Істина", B = "Хиба", C = "Хиба", D = "Істина" (або A=1, B=0, C=0, D=1). Замінімо всі символи змінних у рядку (3) на вибрані значення, в результаті чого змінна Chain міститиме

$$"(1 > !0) + (0 * 0 = 1) * (0 = 1 * !0)" \quad (4)$$

Обчислення логічних значень виконується шляхом послідовного виділення та обчислення виразів у дужках, починаючи з перших внутрішніх дужок і закінчуючи зовнішніми, з урахуванням пріоритету операцій. Першочергово виконуються операції заперечення, потім кон'юнкції, диз'юнкції, імплікації та еквіваленції. Опишемо алгоритм роботи програми.

1. Знаходимо "найглибший" вираз у дужках (надалі - посилка), зберігаємо його в літерній змінній Part та, запам'ятавши його позицію в рядку, видаляємо його з Chain (нагадаємо, що Chain містить основний вираз).
2. Обчислюємо значення посилки з урахування пріоритету операцій. Отриманий результат вставляємо в те місце Chain, звідки узято посилку.
3. Якщо змінна Chain містить символи дужок, то переходимо до пункту 1.
4. Якщо змінна Chain містить більше одного символу, вважатимемо її посилкою і перейдемо до пункту 2.
5. Змінна Chain містить лише один символ, який є результатом. Процес обчислення завершився.

Для реалізації першого пункту знаходимо першу закриваючу дужку і, рухаючись вліво, знаходимо першу відкриваючу. Вираз між ними будемо вважати "найглибшим" виразом у дужках.

Зупинимося більш детально на реалізації другого пункту алгоритму. Видаливши символи дужок з посилки, здійснюємо над нею наступні перетворення (заміни):

1. всі послідовності символів "!0" замінімо символом "1", а послідовності "!1" символом "0";
2. всі послідовності символів "1*1" замінімо символом "1", а послідовності "1*0", "0*1" та "0*0" символом "0";
3. всі послідовності символів "0+0" замінімо символом "0", а послідовності "1+0", "0+1" та "1+1" символом "1";
4. всі послідовності символів "1>0" замінімо символом "0", а послідовності "0>0", "0>1" та "1>1" символом "1";
5. всі послідовності символів "1=1" та "0=0" замінімо символом "1", а послідовності "0=1" та "1=0" символом "0".

Після виконання означених дій у посилці залишиться лише один символ - "0" або "1", що є результатом її обчислення. Наявність більш як одного символу в посилці свідчить про те, що рядок записано неправильно (наприклад, "0+!+01" тощо).

Розглянемо як працює наведений алгоритм з виразом (4).

Крок 1. У рядку Chain "найглибшим" виразом у дужках є вираз "(1>!0)". Зберігаємо його в змінній Part та знищуємо в Chain. Маємо:

Основний вираз (Chain)	"+(0*0=1)*(0=1*!0)"
Посилка (Part)	"(1>!0)"

Крок 2. Виконуємо обчислення посилки, здійснюючи заміни. Після усіх заміन посилка міститиме лише символ "1". Те, що залишилось у посилці, вставляємо в те місце Chain, звідки узято посилку.

Основний вираз (Chain)	"1+(0*0=1)*(0=1*!0)"
Посилка (Part)	"1"

Крок 3. Переходимо до пункту 1 (оскільки символи дужок все ще наявні в основному виразі).

Крок 4. У рядку Chain "найглибшим" виразом у дужках є "(0*0=1)". Зберігаємо його в змінній Part та знищуємо в Chain. Маємо:

Основний вираз (Chain)	"1+(0=1*!0)"
Посилка (Part)	"(0*0=1)"

Крок 5. Виконуємо обчислення посилки, здійснюючи заміни. Після усіх замін посилка міститиме лише символ "0". Те, що залишилось у посилці, вставляємо в те місце Chain, звідки узято посилку.

Основний вираз (Chain)	"1+0*(0=1*!0)"
Посилка (Part)	"0"

Крок 6. Переходимо до пункту 1 (оскільки символи дужок все ще наявні в основному виразі).

Крок 7. У рядку Chain "найглибшим" виразом у дужках є "(0*0=1)". Зберігаємо його в змінній Part та знищуємо в Chain. Маємо:

Основний вираз (Chain)	"1+0*"
Посилка (Part)	"(0=1*!0)"

Крок 8. Виконуємо обчислення посилки, здійснюючи заміни. Після усіх замін посилка міститиме лише символ "0". Те, що залишилось у посилці, вставляємо в те місце Chain, звідки узято посилку.

Основний вираз (Chain)	"1+0*0"
Посилка (Part)	"0"

Крок 9. Оскільки символи дужок відсутні в основному виразі, вважатимемо його посилкою. Зберігаємо його в змінній Part та знищуємо в Chain. Маємо:

Основний вираз (Chain)	""
Посилка (Part)	"1+0*0"

Крок 10. Виконуємо обчислення посилки, здійснюючи заміни. Після усіх замін посилка міститиме лише "1". Те, що залишилось у посилці, вставляємо в те місце Chain, звідки узято посилку.

Основний вираз (Chain)	"1"
Посилка (Part)	"1"

Крок 11. Оскільки основний вираз містить лише один символ - "1", подальше обчислення припиняємо. Отриманий результат - "Істина".

Розглянуто процес обчислення логічного значення виразу для одного набору значень логічних змінних. Програма має забезпечувати підстановку всіх можливих наборів значень логічних змінних для отримання всіх можливих значень логічного виразу. В результаті одержується таблиця значень істинності для даної формули алгебри висловлень. Крім того, виводиться повідомлення про те, чи є формула виконуваною, невиконуваною або тотожно істинною.

ЛІТЕРАТУРА

1. О.В.Вітюк, І.Г.Ленчук. Реалізація обчислень аналітичних виразів у програмуванні //Вісник Житомирського педагогічного інституту. - 1988. - №1. - С. 23-26.

Вітюк Олександр Володимирович – асистент кафедри математики та інформатики Житомирського державного педагогічного інституту ім. І.Франка.

Наукові інтереси:

- інформатика;
- методика викладання інформатики і математики.

Коломійцев Олег Петрович - старший викладач кафедри математики Житомирського державного педагогічного інституту ім. І.Франка.

Наукові інтереси:

- інформатика;
- методика викладання інформатики і математики;
- математична логіка.