Системы счисления и основы логики

Логические значения и операции
Логические выражения
Двоичная система записи чисел

Логические значения и операции

Как мы видели, в алгоритмах очень важную роль играют проверки условий.

Верно ли А? Если да – то делать Б, если нет – то С. Как видно, при определении истинности или ложности условия есть только две возможности: или «да» (истинно), или «нет» (ложно). Никаких других возможностей (например – «может быть») не предусматривается, потому что обрабатывать их алгоритмически не совсем ясно как.

Таким образом в алгоритмах (и вообще в логике) имеют дело с двумя логическими значениями: «истина» и «ложь». Иногда их обозначают 0 и 1 (а называют их «логический ноль» и «логическая единица»), а также буквами Л («ложь» и И («истина»).

Теперь посмотрим, какие операции бывают над логическими значениями. Сначала рассмотрим случай, когда операция производится над одной переменной. Поскольку значений переменной всего два, можно нарисовать таблицу значений для операции – она не займет много места.

значение переменной	Результат операции
И	
Л	

В правом столбце оставлено место под результат операции. Посмотрим на все возможные варианты результатов.

значение переменной	Результат операции
И	И
Л	Л

Это «тождественная» операция – она ничего не меняет.

значение переменной	Результат операции
И	И
Л	И

Эту операцию можно назвать «тождественная истина».

значение переменной	Результат операции
И	Л
Л	Л

Эту операцию можно назвать «тождественная ложь». Эти три операции применяются, но относительно редко.

значение переменной	Результат операции
И	Л
Л	И

А вот это уже знакомая нам всем операция «не»: «не истина» - это «ложь», «не ложь» - «истина».

Теперь посмотрим на логические операции над двумя переменными. Для них таблица будет побольше: для каждой комбинации из двух логических значений – своя строка.

Значение 1-й переменной	Значение 2-й переменной	Результат операции
И	И	
И	Л	
Л	И	
Л	Л	

Сколько всего будет вариантов таких табличек с заполненным последним столбцом? В каждой строке 2 варианта, поэтому всего 2x2x2x2=16 вариантов. Поэтому рассмотрим только самые важные.
Сначала о тех, с которыми мы сталкиваемся каждый день. Это операции «и» и «или».
Как мы знаем, выражение «А и Б» истинно если только истинны и А, и Б. То есть табличка будет такая:

Значение 1-й переменной	Значение 2-й переменной	Результат операции
И	И	И
И	Л	Л
Л	И	Л
Л	Л	Л

Для логического «или» наоборот, «А или Б» ложно, если только если и А, и Б ложны.
Соответственно и таблица для «А или Б» такая:

Значение 1-й переменной	Значение 2-й переменной	Результат операции
И	И	И
И	Л	И
Л	И	И
Л	Л	Л

Как кажется больше привычных нам «логических связываний» двух логических значений, кроме «и» и «или», нет. На самом деле есть еще одна такая операция: «из А следует Б». Что требуется от следования? Чтобы из истины никогда не следовала ложь. Поэтому и таблица будет такой:

Значение 1-й переменной	Значение 2-й переменной	Результат операции
И	И	И
И	Л	Л
Л	И	И
Л	Л	Л

Эта таблица означает, что из лжи может следовать все, что угодно! Т.е. если в рассуждениях опираться на ложный «факт», то из него можно вывести все, что душе угодно.

Из других возможных логических операций на 2-х значениях важными являются еще две. Одна называется «штрих Шеффера» и имеет такую таблицу:

Значение 1-й переменной	Значение 2-й переменной	Результат операции
И	И	Л
И	Л	И
Л	И	И
Л	Л	И

Другая имеет название «стрелка Пирса» и имеет следующую таблицу:

Значение 1-й переменной	Значение 2-й переменной	Результат операции
И	И	Л
И	Л	Л
Л	И	Л
Л	Л	И

Эти две логические операции в обычных рассуждениях не применяются, но оказываются очень важными для построения компьютеров. Оказывается, комбинациями одной единственной операции (или штриха, или стрелки), можно выразить все, что выразимо любой логической операцией. Что это значит в точном алгоритмическом смысле, мы рассмотрим в следующем параграфе.

Операции. Очень часто из простых утверждений строят более сложные с помощью логических операций.

Сегодня понедельник («да», истинное утверждение).

Сейчас идет дождь («да», истинное утверждение).

Сейчас понедельник и идет дождь («да», истинное утверждение).

Сейчас три часа десять минут? («да», истинное утверждение).

Логические выражения

В логике можно строить выражения способом, который в точности повторяет построение выражений в арифметике. Напомним, что в ней выражения строятся из букв, чисел, скобок и знаков действий (операций). Например,

(X+Y) x (5+X).

Вычисляется выражение подстановкой чисел (которые должны быть заданы заранее) вместо букв (называемых также переменными) и затем производятся указанные в выражении действия. Например, при X=2 и Y=3 значением указанного выражения будет

(2+3)x(5+2), то есть 35.

Точно так же можно поступить и с логическими операциями. Например, выражение

(X & Y) V (не X).

Вычисляется значение такого выражения при задании значений X и Y. Пусть, например, значением X будет И (то есть «истина»), а для Y значением будет Л (то есть «ложь»). Тогда, по таблице значений для логического «и», значением (X & Y) будет Л («ложь»), а для (не X) значением также будет Л. Наконец, значением (Л V Л) будет также Л.

Посмотрим это же в фильме.

Аналогично можно строить другие логические выражения.

Поскольку в логических операциях нас интересуют только их таблицы, выражения, имеющие одну и ту же таблицу, называются эквивалентными.

Вот некоторые примеры.

Если сравнить таблицы для (не (X или Y)) и ((не X) и (не Y)), то окажется, что они одинаковые. То есть эти два выражения задают одну и ту же логическую операцию (на двух данных).

То же самое оказывается и для выражений (не (X и Y)) и ((не X) или (не Y)). Эти два выражения также эквивалентны и задают одну и ту же логическую операцию.

Еще один пример – выражение (не (не X)). У него таблица совпадает с «=». То есть (не (не X)) и X – можно считать, одно и то же.

Некоторые выражения задают известные нам логические операции. Например, (не (не (X или Y))) задает операцию (X или Y). Это можно проверить, и не вычисляя всю таблицу для (не (не (X или Y))) целиком: ведь что такое (не (не Z)) нам известно – это всегда то же самое, что и просто Z.

Оказывается, с помощью только «или» и «не» можно задать вообще любую логическую операцию.

Точно так же любую логическую операцию можно задать и с помощью «и» и «не». Способ такой же, результат тоже громоздкий.

Есть и такие операции, которых одних достаточно, чтобы построить выражения, задающие любую операцию. Оказывается, именно таковы штрих Шеффера и стрелка Пирса. Посмотрим, почему.
То, что с помощью штриха Шеффера можно задать любую логическую операцию, широко

используется при разработке компьютеров: в них все собирается из абсолютно одинаковых элементов, каждый их которых работает, как штрих Шеффера. А раз можно обойтись одним-единственным стандартным элементом, это делает и проектирование и изготовление компьютеров дешевле.

Сказанное относится и к стрелке Пирса. Как легко видеть, эта логическая операция – то же самое, что и выражение не (X V Y). Поэтому из нее так же само, как и для штриха Шеффера, можно получить операцию «не», а с ее помощью – операцию «или».

И еще одно замечание. Видно, что желательно ввести специальные обозначения для логических операций (например, чтобы не путать «и» логическое с «и» просто в предложениях). Таким обозначением для «и» является ?, для «или» это ?, для «следует» это ? или ? (вообще для логических операций используются несколько разных обозначений).

Двоичная система записи чисел

То, как устроены компьютеры, диктует несколько непривычную для нас внутреннюю систему записи чисел. Эта система записи очень похожа на обычную, в которой все числа записываются по степеням десятки. Например, наше обычное число 267 – это на самом деле запись

Двоичная система записи – это просто система записи для чисел, поэтому для такой записи существуют правила сложения, умножения и т.д. – двоичная арифметика. Посмотрим, как она выглядит.

Сложение. Очевидно, можно использовать ту же схему, как и для обычных для нас десятичных чисел. Именно, складываем поразрядно (разряды в записи двоичных чисел определяются аналогично десятичным – с конца), с переносом лишних единиц в следующий разряд.

$$(1)_2 + (1)_2 = (10)_2$$

$$\frac{(100)_2}{(1000)_2} = \frac{(1)_2}{(10)_2}$$

Конечно, для двоичной арифметики случай сложения с переносом только один: $(1)_2 + (1)_2 = (10)_2$, в остальных случаях переноса нет. В десятичной арифметике почти в половине случаев сложения одноразрядных чисел перенос единички в следующий разряд есть.

Посмотрим еще пример – теперь уже сложение столбиком.

Вычитание строится опять по принципам десятичной арифметики: если в разряде не хватает единички – забираем ее у следующего, старшего разряда.

Как вы уже догадались, умножение и деление столбиком производятся опять похожим на десятичный случай способом.

Устройство дробей в двоичной арифметике опять может быть смоделировано по принципам десятичных чисел. Например, дробь a/b – снова число такое, что будучи умноженным на b, дает a. Поэтому как в десятичной арифметике $\frac{(4)_{10}}{(8)_{10}} = \frac{(1)_{10}}{(2)_{10}}$, так и в двоичной $\frac{(100)_2}{(1000)_2} = \frac{(1)_2}{(10)_2}$.

Так же само, как и десятичные, можно ввести «двоичные дроби», отделяя целую от дробной части запятой (теперь это двоичная запятая!)

Правила сложения, вычитания, умножения и деления столбиком для таких двоичных дробей с запятой снова будут аналогичны таким же для десятичных дробей.

Другие операции, которые известны для десятичных чисел, конечно, также можно ввести и для двоичных чисел. Например, возведение в степень, извлечение квадратных корней и т.д.