

Л. Ф 02. ОПЕРАЦИИ И ВЫРАЖЕНИЯ

1. ПОНЯТИЕ ВЫРАЖЕНИЯ И ОПЕРАЦИИ [по 1 и 2]

В любой Фортран-программе должен присутствовать, по крайней мере, один выполняемый оператор. Простейшим выполняемым оператором является *оператор присваивания*, который имеет следующий вид:

$$V=A.$$

Здесь V — имя переменной, A — выражение.

Правило выполнения: переменной с именем V присваивается значение *выражения* A .

Выражение - это формула, по которой вычисляется значение. Выражение состоит из *операндов* и нуля или более *операций*.

Операндами выражения являются объекты данных и/или их подобъекты, например элементы массива.

Пример выражения

$$2.0*\cos(x/4.5).$$

Используемые в выражениях **операции** разделяются на *двуместные* (бинарные) и *одноместные* (унарные + и -). В двуместной операции участвуют два операнда, в одноместной – один.

Пример.

$b + c$! Простое выражение с двуместной операцией

$-b$! Простое выражение с одноместной операцией

Общий вид выражения, в котором присутствуют двуместные операции:

операнд *операция* операнд *операция* операнд...

Замечание. В Фортране, в отличие от СИ, унарным операциям не может предшествовать знак другой операции, т.е. два знака не могут стоять рядом.

Пример:

$k = 12/-a$! Ошибка

$k = 12/(-a)$! Правильно

С каждым встроенным типом данных связан набор встроенных операций. Операции делятся на арифметические операции; операции отношения и логические операции.

2. АРИФМЕТИЧЕСКИЕ ВЫРАЖЕНИЯ [Бартенев, 1999]

2.1. Понятие арифметической операции и выражения

С числовыми типами данных применяются **арифметические операции**:

** — возведение в степень (операция с наивысшим приоритетом)

*, / — умножение, деление

+, - — сложение, вычитание (операции с низшим приоритетом).

Операции перечислены в порядке убывания приоритета.

Арифметическое выражение состоит из числовых операндов и арифметических операций. Если в выражении присутствует несколько операций, то они, кроме возведения в степень, выполняются в соответствии с приоритетом *слева направо*. Операции возведения в степень выполняются *справа налево*. Так, выражение $-a+b+c$ будет выполнено в следующем порядке $(((-a)+b)+c)$. А выражение $a**b**c$ вычисляется так: $(a**(b**c))$. Заключенные в *круглые скобки* подвыражения вычисляются в первую очередь.

Пример.

$k = 2*2**2/2/2$! 2

! Проиллюстрируем последовательность вычислений, расставив скобки:

$k = ((2 * (2**2))/2)/2$! 2

! Проиллюстрируем влияние скобок на результат выражения

$K = 2 ** 8/2+2$! 130

$K = 2** (8/2+2)$! 64

$k = 2 ** (8/(2 + 2))$! 4

Запрещается в арифметических выражениях:

- делить на нуль;
- возводить равный нулю операнд в отрицательную или нулевую степень;
- возводить отрицательный операнд в нецелочисленную степень.

Пример.

$a = (-2)**2.2$! Ошибка - нарушение последнего ограничения

Не следует в арифметических выражениях комбинировать переменные, различающиеся на несколько порядков, например:

`real(4) :: x = 1.0e+30, y = -1.0e +30, z = 5.0`

`print *, (x + y) + z` ! 5.000000 (правильно)

`print *, x + (y + z)` ! 0.000000E+00 (ошибка)

2.2. Целочисленное деление

Рассмотрим простую программу:

`real(4) dp, dn`

`dp = 3/2`

`dn = -3/2`

`print *, dp, dn` ! 1.0 -1.0

`end`

Результат 1.0 и -1.0 вместо ожидаемых значений 1.5 и -1.5 имеет простое объяснение: 3, -3 и 2 – целые числа, и результатом деления будут также целые числа – целая часть числа 1.5 и целая часть числа -1.5, то есть 1 и -1. Затем, по-

сколько переменные dp и dn имеют тип REAL(4), целые числа 1 и -1 будут преобразованы в стандартный вещественный тип.

Чтобы получить ожидаемый с позиции обычной арифметики результат в программе можно записать:

$dp = 3./2.$ или $dp = 3/2.$ или $dp = 3./2$

2.3. Ранг и типы арифметических операндов [Самохин, 1994]

Когда в арифметическом выражении операнды (переменные и константы) имеют разный тип, то тип результата определяется типом операнда максимального ранга. **Ранг операнда** зависит от его типа в соответствии со списком:

1. INTEGER (низший ранг)
2. REAL
3. REAL*8
4. COMPLEX
5. COMPLEX*16 (высший ранг)

Например, результатом арифметического выражения с операндами INTEGER и REAL будет величина, относящаяся к типу REAL.

Особый случай: операции над операндами REAL*8 и COMPLEX породят величину COMPLEX*16, а не COMPLEX.

3. ВЫРАЖЕНИЯ ОТНОШЕНИЯ И ЛОГИЧЕСКИЕ ВЫРАЖЕНИЯ [по Бартеневу, 1999]

3.1. Выражения отношения

Выражение отношения имеет вид

$$A \# B,$$

где # — операция отношения, **сравнивает** значения двух арифметических или символьных выражений. **Результатом выражения отношения** является истина — .TRUE. или ложь — .FALSE.

Операндами выражения отношения могут быть как скаляры, так и массивы.

Операции отношения в Фортране 90 могут быть записаны в двух формах

.LT.	или	<	меньше;
.LE.	или	<=	меньше равно;
.GT.	или	>	больше;
.GE.	или	>=	больше равно;
.EQ.	или	==	равно;
.NE.	или	/=	не равно.

Пробелы в записи обозначения операции являются ошибкой:

a .le. b ! Ошибка. Правильно: a.le.b
a <= b ! Ошибка. Правильно: a <= b

Все операции отношения являются **двуместными (бинарными)** и должны появляться между операндами. Выполняются операции отношения слева направо.

Если в выражении отношения один операнд имеет вещественный, а другой целый тип, то перед выполнением операции целочисленный операнд преобразовывается в вещественный тип.

Операнды выражения отношения могут иметь и комплексный тип. В этом случае можно применять только операции .NE. (/=) и .EQ. (==).

Пример.

real:: a = 1.2, b = - 5.5

logical fl

fl = a > b ! .TRUE.

fl = a > abs(b) ! .FALSE.

! Встроенная функция ABS(x) возвращает |x|.

3.2. Логические выражения

Логические выражения состоят из отношений, логических переменных, логических констант и логических операций.

Логические выражения имеют **результатом логическое значение** истина — .TRUE. или ложь — .FALSE..

Логические операции:

.NOT.	логическое НЕ (отрицание);
.AND.	логическое И;
.OR.	логическое ИЛИ;
.XOR.	логическое исключающее ИЛИ;
.EQV.	эквивалентность;
.NEQV.	неэквивалентность.

Все логические операции, кроме отрицания, являются бинарными. Логическая операция .NOT. является унарной и располагается перед операндом. Выполняются логические операции слева направо.

В табл. 1 приведены результаты логических операций над логическими переменными x и y , принимающими значения "истина" (И) и "ложь" (Л).

Таблица 1. Таблица истинности

x	y	x.AND.y	x.OR.y	.NOT.x	x.XOR.y	x.EQV.y	x.NEQV.y
И	И	И	И	Л	Л	И	Л
И	Л	Л	И	Л	И	Л	И
Л	И	Л	И	И	И	Л	И
Л	Л	Л	Л	И	Л	И	Л

Операции отношения и логические операции выполняются слева направо, то есть, если две последовательные операции имеют равный приоритет, первоначально выполняется левая операция.

4. ПРИОРИТЕТ ВЫПОЛНЕНИЯ ОПЕРАЦИЙ [по Бартеньеву, 1999]

Когда арифметические операции, операции отношения и логические операции присутствуют в одном выражении, приоритет выполнения в порядке убывания операций таков:

- арифметические операции;
- операции отношения;
- логические операции.

В табл. 2 встроенные операции Фортрана расположены в порядке убывания приоритета. Каждая ячейка таблицы содержит операции с равным приоритетом.

Таблица 2. Приоритет выполнения операций

Арифметические			Отношения		Логические			
**	*	+	.LT., <	.EQ., ==	.NOT.	.AND.	.OR.	.XOR. .EQV. .NEQV.
	/	-	.LE., <=	.NE., /=				
			.GT., >					
			.GE., >=					

Пример. Вычислить результат логического выражения:

$x/a == 1$.OR. $b/(a + b) < 1$.AND. .NOT. $b == a$.OR. $x \neq 6$
 при $x = 6.0$, $a = 2.0$ и $b = 3.0$.

Вычислив результаты арифметических операций, получим:
 $3.0 == 1$.OR. $0.6 < 1$.AND. .NOT. $b == a$.OR. $x \neq 6$

Затем выполняются операции отношения:
 .FALSE. .OR. .TRUE. .AND. .NOT. .FALSE. .OR. .FALSE.

Далее выполняются пошагово логические операции с учетом их приоритета. После выполнения .NOT. .FALSE.:

.FALSE. .OR. .TRUE. .AND. .TRUE. .OR. .FALSE.

После выполнения .TRUE. .AND. .TRUE.:
 .FALSE. .OR. .TRUE. .OR. .FALSE.

Окончательный результат: .TRUE.

ЛИТЕРАТУРА

1. Бартенев О.В. Фортран для студентов. — М.: Диалог-МИФИ, 1999. —400 с.
2. Самохин А.Б., Самохина А.С. Фортран и вычислительные методы. — М.: Русина, 1994. — 120 с.