

Операции и выражения. (OK)

План ответа:

1. Понятия

- 1.1 «выражение»;
- 1.2 «операция»,
- 1.3 «операнд»,
- 1.4 «побочный эффект»,
- 1.5 «приоритет»,
- 1.6 «ассоциативность»

2. арифметические операции;

3. составное присваивание;

4. операции инкремента и декремента;

5. операции сравнения;

6. логические операции;

7. порядок вычисления выражений (включая особенности вычисления логических выражений).

1. Любой язык программирования включает в себя средства описания данных и средства описания действий.

Средства описания действий указывают действия, исполняемые программой.

1.1 Простейшим средством описания действий является выражение.

Выражение задает действия, которые вычисляют единственное значение.

Оно состоит из констант, переменных, а также знаков операций и скобок.

1.2 Операция - это любое действие, которое изменяет элементы данных (отсечка)

1.3 Элементы данных, к которым применяется операция, называют операндами.

Каждая операция имеет операнды определенных типов и задает способ получения по значениям этих операндов нового значения определенного типа.

1.4 Некоторые операции имеют побочный эффект (side effect). При выполнении этих операций, кроме основного эффекта

- вычисления значения - происходят изменения объектов или файлов.

1.5 В языке Си операции по способу записи подразделяются на

Правила приоритетов операций при вычислении выражений определяют порядок, в котором выполняются операции, имеющие разные приоритеты.

$a + b * c \Rightarrow a + (b * c)$

1.6 На вопрос о том, какой из операторов, имеющих равный приоритет, выполняется первым, отвечают правила ассоциативности.

Операции одинакового приоритета либо левоассоциативны, либо правоассоциативны.

$a - b + c \Rightarrow ((a - b) + c)$

3. После вычисления значения выражения нужно сохранить результат в переменной. Для этих целей используется операция присваивания. $v = e$

Формально алгоритм работы операции присваивания в языке Си можно описать следующим образом:

Вычислить l-значение первого операнда (v) операции.

Вычислить r-значение второго операнда (e) операции.

Присвоить вычисленное r-значение вычисленному l-значению объекта данных;

Возвратить вычисленное r-значение как результат выполнения операции.

Во многих языках программирования присваивание - это оператор.

В языке Си присваивание - это операция такая же как, например, сложение.

Значением выражения $a = b$ будет значение переменной b, приведенное к типу переменной a, и в качестве побочного эффекта

это значение будет присвоено переменной *a*.

Благодаря тому, что присваивание - это операция, несколько присваиваний могут быть объединены в «цепочку»: *i = j = k = 0*;

Операция присваивания правоассоциативна: *i = (j = (k = 0))*;

В операции присваивания «старое» значение переменной часто используется для вычисления «нового» значения этой же переменной:

i = i + 2;

Операции составного присваивания позволяют нам короче записать этот оператор: *i += 2*;

Операция составного присваивания обладает теми же свойствами, что и обычная операция присваивания.

Например, она обладает правой ассоциативностью: *i += j += k; => i += (j += k)*;

4. В языке Си существуют операции, которые позволяют еще более сократить запись. Это операции инкремента «++» и декремента «--».

Операции инкремента и декремента могут записываться как в префиксной (*++i*), так и в постфиксной формах (*i++*).

Операции инкремента и декремента, как и операция присваивания, обладают побочным эффектом: они изменяют значения своих операндов.

В случае префиксного инкремента *++i* вычисление выражения возвращает значение *i + 1* и увеличивает на 1 значение переменной *i*:

В случае постфиксного инкремента *i++* вычисление выражения возвращает значение *i* и увеличивает на 1 значение переменной *i*:

5. Операции сравнение эквивалентны соответствующим математическим операциям за одним исключением.

В выражении они возвращают целое значение 0, когда операция ложна, или целое значение 1, в противном случае.

Выражение *i < j < k* допустимо в языке Си, но имеет значение отличное от аналогичного математического выражения.

i < j < k => (i < j) < k

В выражении сначала проверяется меньше ли значение переменной *i* значения переменной *j*,

а затем 0 или 1 (как результат этого сравнения) будут сравниваться со значением *k*.

6. При вычислении логических операций используется так называемая сокращенная схема.

Сначала вычисляется значение левого операнда, затем правого.

Если результат операции может быть определен по значению только левого операнда, то правый операнд не вычисляется.

&& || !

Для логических операций «И» и «ИЛИ» порядок вычисления операндов фиксирован: сначала вычисляется левый операнд, затем правый.

7. Язык Си не определяет порядок (в общем), в котором вычисляются подвыражения.

(a+b)(c-d)* - не определено

Большинство выражений имеют одно и то же значение независимо от того, в каком порядке вычисляются подвыражения. Это может быть не так

– если подвыражение изменяет один из своих операндов

a = 5; c = (b = a + 2) – (a = 1); // ОШИБКА

– к подобным ситуациям может также привести использование операций инкремента и декремента

*i = 2; j = i * i++; // ОШИБКА*