

Expresiones Aritméticas

Programación digital I
Escuela de Sistemas
Facultad de Ingeniería
Gilberto Diaz

Expresiones Aritméticas

- ➔ El computador puede realizar cálculos además de mostrar datos por pantalla.
- ➔ Son expresiones analíticas que no contiene más funciones que aquellas que pueden calcularse con las operaciones del álgebra, a saber:
 - Suma
 - Resta
 - Multiplicación
 - División

Formato de las Expresiones

- ➔ Las expresiones Aritméticas se utilizan en sentencias con el siguiente formato:

variable = expresión;

- ➔ variable: localidad de memoria referenciada por un nombre bien definido
- ➔ expresión: expresión aritmética

Operación de Asignación

- ➔ El mecanismo para asignar valores a las variables es la sentencia de asignación (=)

variable = expresión;

- ➔ Se evalúa la expresión aritmética y el resultado es almacenado en la variable
- ➔ La sentencia de asignación permite modificar el contenido de una variable

Expresiones

- ➔ Las expresiones pueden estar compuestas por:
 - Una variable
 - Una constante
 - Una combinación de operadores y operandos

Expresiones

- ➔ Operandos: constantes, variables, números u otras expresiones
- ➔ Operadores: símbolos que indican al compilador que ejecute determinadas operaciones con los elementos sobre los que opera:
 - aritméticos
 - lógicos
 - de cadena
 - relacionales

Tipos de Expresiones

- ➔ Aritméticas: Los operandos que intervienen son numéricos, los resultados son numéricos y los operadores son aritméticos
- ➔ Lógicas: Los operadores son lógicos o relacionales. Los resultados sólo pueden ser dos valores:
 - Cierto
 - Falsto

Tipos de Expresiones

- ➔ Aritméticas: Los operandos que intervienen son numéricos, los resultados son numéricos y los operadores son aritméticos
- ➔ Lógicas: Los operadores son lógicos o relacionales. Los resultados sólo pueden ser dos valores:
 - Cierto
 - Falsto

Operadores Aritméticos

Operación	Operador	Expresión	Expresión en C
Suma	+	$A + 3$	$A + 3$
Resta	-	$A - 3$	$A - 3$
Multiplic	x	$A \times 3$	$A * 3$
División	/	$A \div 3$	$A / 3$
Resto	%		$A \% 3$
Incremento	++	$A + 1$	$A++$ ó $++A$
Decrement	--	$A - 1$	$A--$ ó $--A$

Operadores de Incremento

- ➔ En el caso de tener una variable i con valor igual a 2, tenemos

$j = i++;$

- ➔ Primero se asigna el valor de i a j y luego se incrementa el valor de i
- ➔ $i = 3$ $j = 2$

Operadores de Incremento

- ➔ En el caso de tener una variable *i* con valor igual a 2, tenemos

$j = ++i;$

- ➔ Primero incrementa el valor de *i* y luego se asigna el valor de *i* a *j*

➔ **$i = 3$ $j = 3$**

Operadores de Incremento

- ➔ En el caso de tener una variable i con valor igual a 2, tenemos

$j = i--;$

- ➔ Primero se asigna el valor de i a j y luego se decrementa el valor de i
- ➔ $i = 1$ $j = 2$

Operadores de Incremento

- ➔ En el caso de tener una variable i con valor igual a 2, tenemos

$$j = -i;$$

- ➔ Primero decrementa el valor de i y luego se asigna el valor de i a j

- ➔ $i = 1$ $j = 1$

Operadores de Incremento

- ➔ Este tipo de operadores sólo se utilizan con variables enteras
- ➔ Se pueden utilizar dentro de las expresiones, ejemplo:

$x = 5 + y++;$

- ➔ Lo cual es equivalente a:

$x = 5 + y;$

$y++;$

Operadores Aritméticos

- ➔ Los operadores + - * / pueden ser utilizados con los siguientes operandos:
 - Constantes
 - Variables
 - Expresiones
- ➔ La división de enteros genera resultados enteros
$$5/2 = 2$$

Operadores Aritméticos

- ➔ El operador % da como resultado el resto de la división de los operandos involucrados. Sólo se puede utilizar con operandos enteros, ejemplo

$$5\%2 = 1$$

Operadores Aritméticos

Ejemplos:

- ➔ `area = largo * ancho;`
- ➔ `numero = 2 + 3;`
- ➔ `i = j * k;`
- ➔ `resto = 3 % 2;`

Reglas de los Operandos

- ➔ Si todos los operandos en una expresión son enteros entonces el resultado es entero
- ➔ Si alguno de los operandos es real entonces el resultado es real

Reglas de los Operandos

- ➔ Si la expresión evaluada da como resultado un número entero y se le asigna a una variable real, entonces el resultado se convertirá antes a real antes de almacenarlo en la variable

Reglas de los Operandos

- ➔ Si la expresión evaluada da como resultado un número real y se le asigna a una variable entero, entonces el resultado se convertirá antes a entero antes de almacenarlo en la variable

Reglas de los Operandos

- ➔ Cuando hay conversiones de tipos de datos más grandes a más pequeños, ocurre un truncamiento de los bits más significativos.
- ➔ Si alguno de los operandos es long double, el resto es convertido a long double

Reglas de los Operandos

- ➔ Si no, si alguno de los operandos es double, entonces el resto son convertidos a double
- ➔ Si no, si alguno de los operandos es float, el resto es convertido a float
- ➔ Si no, convierte los operandos char y short a int
- ➔ Por último, si algún operando es long el resto es convertido a long

Nivel de Precedencia

- ➔ Los delimitadores paréntesis () son utilizados para darle mayor nivel de precedencia a las expresiones.
- ➔ Es decir, las expresiones entre paréntesis son evaluadas primero

Nivel de Precedencia

- ➔ Luego, el nivel de precedencia es como sigue:
 - ++ --
 - * / %
 - + -
- ➔ Un mismo nivel de precedencia se evalúa de izquierda a derecha

Nivel de Precedencia

➔ Ejemplo:

$$(x + y + z) * (v + w)$$

$$R1 = x + y$$

$$R2 = R1 + z$$

$$R3 = v + w$$

$$R4 = R2 * R3$$

Nivel de Precedencia

➔ Ejemplo:

$$(3 + 4 + 1) * (2 + 2)$$

$$R1 = 3 + 4$$

$$R2 = 7 + 1$$

$$R3 = 2 + 2$$

$$R4 = 8 * 4$$

$$R5 = 32$$

Nivel de Precedencia

➔ Ejemplo:

$$a*b*c + d*e*f + g\%h$$

$$R1 = a*b$$

$$R2 = R1*c$$

$$R3 = d*e$$

$$R4 = R3*f$$

$$R5 = g\%h$$

$$R6 = R2 + R4 + R5$$

Nivel de Precedencia

➔ Ejemplo:

$$1*2*3 + 1*2*3 + 4\%2$$

$$R1 = 1*2$$

$$R2 = 2*3$$

$$R3 = 1*2$$

$$R4 = 2*3$$

$$R5 = 4\%2$$

$$R6 = 6 + 6 + 0$$

$$R7 = 12$$

Nivel de Precedencia

➔ Ejemplo:

$$(a * (b+c)) - 2 * d + (4*e - f)$$

$$R1 = b + c$$

$$R2 = a * R1$$

$$R3 = 4 * e$$

$$R4 = R3 - f$$

$$R5 = 2 * d$$

$$R6 = R2 - R5$$

$$R7 = R6 + R4$$

Nivel de Precedencia

➔ Ejemplo:

$$(2 * (1+1)) - 2 * 1 + (4*3 - 4)$$

$$R1 = 1 + 1$$

$$R2 = 2 * 2$$

$$R3 = 4 * 3$$

$$R4 = 12 - 4$$

$$R5 = 2 * 1$$

$$R6 = 4 - 2$$

$$R7 = 2 + 8$$

$$R8 = 10$$

Nivel de Precedencia

- ➔ Ejercicio: Evalúe la siguiente expresión para los valores $a=2$, $b=3$, $c=1$, $d=4$

$$(a * (b+c)) - 2 * a + (4*c - d) - d$$

Evaluación de Expresiones

➔ Las expresiones pueden ser evaluadas de forma secuencial

➔ Ejemplo: $1 + 2 * 3$

3

2 *

1 +

paso 1

6

1 +

paso 2

7

paso 3

Evaluación de Expresiones

- ➔ Para evaluar una expresión utilizando esta técnica deben tenerse en cuenta las siguientes reglas
 - La prioridad de los operadores de mayor a menor es $*$ / $+$ -
 - Las operaciones con igual prioridad se evalúan de izquierda a derecha
 - Al final de la operación se efectúan el resto de las operaciones pendientes hasta que el contenedor de los operadores quede vacío

Evaluación de Expresiones

- Cuando un paréntesis que cierra aparece se ejecutan todas las operaciones hasta que aparece un paréntesis que abre

Evaluación de Expresiones

➔ Ejemplo: $(1+2) * 4 / 2$

)

1 +

4

2

2 (

3 *

12 /

6

paso 1

paso 2

paso 3

paso 4

Evaluación de Expresiones

➔ Ejercicio: Evalúe la expresión

$$1*2 + 3*4 - 10/2$$

Operaciones de Asignación

➔ En C se cuenta con abreviaturas para realizar asignaciones

- `+=` `x = x + 3;` `x += 3;`
- `-=` `x = x - 3;` `x -= 3;`
- `*=` `x = x * 3;` `x *= 3;`
- `/=` `x = x / 3;` `x /= 3;`
- `%=` `x = x % 3;` `x %= 3;`

Operaciones de Asignación

➔ Ejemplos:

$x /= y + 3;$

$x = x/(y+3);$

$\text{suma} \% = 3;$

$\text{suma} = \text{suma} \% 3;$

$\text{angulo} *= 3;$

$\text{angulo} = \text{angulo} * 3;$

Asignación Múltiple

- ➔ En C podemos darle el mismo valor a diferentes variables utilizando una sola instrucción. Ejemplo:

`a = b = c = d = 1;`

- ➔ A todas las variables se les asigna el valor 1

Recapitulemos

- ➔ Hemos visto diferentes formas de almacenar valores en una variable
- ➔ ¿Podría enumerarlas?

Recapitulemos

- ➔ variable = valor constante `x = 3;`
- ➔ variable = variable `x = y;`
- ➔ variable = expresión `x = a + 3;`
- ➔ leer variable `scanf("%tipo", &x);`

Ejercicios

- ➔ En la página siguiente Ud puede encontrar ejercicios que le ayudaran a afianzar los conocimientos:

<http://webdelprofesor.ula.ve/ingenieria/amoret/pr1-flash>