



Expresiones Aritméticas

Prof. Judith Barrios Albornoz

Departamento de Computación

Escuela de Ingeniería de Sistemas

Facultad de Ingeniería

Universidad de Los Andes

Semestre A_2013

La Operación **Asignación**

variable = expresión;

- La sentencia de asignación es un modo de darle valores a una variable
- La expresión se evalúa y el resultado es asignado a la variable
- La sentencia de asignación modifica el contenido de una variable

variable = valor constante;

x = 3;

variable = variable;

x = y;

variable = expresión;

x = (y + 1)/2;

La Operación **Asignación**

Ejemplos

✓ $a = b;$

✓ $\text{suma} = 60;$

¿Cuál será el **valor** que tomará la **variable D** después de la ejecución de las siguientes instrucciones?

$A = 12;$

$B = A + 5;$

$C = B - 2;$

$D = C + (A + B);$

Expresiones

Puede ser:

- ✓ Una **variable**
- ✓ Una **constante**
- ✓ Una *combinación* **de operadores y operandos**

Expresiones

Operandos: constantes, variables u otras expresiones

Operadores: Símbolo que dice al compilador que ejecute determinadas operaciones con los elementos sobre los que opera

(aritméticos, lógicos, de cadena y de comparación)

Tipos de expresiones

- ✓ **Aritméticas**: Los operandos que intervienen en ella son numéricos, el resultado es numérico y los operadores son aritméticos
- ✓ **Lógicas**: Su resultado es **CIERTO** o **FALSO**.
 - ✓ Se construyen mediante los operadores de comparación y lógicos

Operadores Aritméticos en C/C++

Operación	Operador	Operador en C/C++	Ejemplo
Suma	+	+	24 + 56
Resta	-	-	5 - 4
Multiplicación	x	*	4 * 5
División	/ ó —	/	34 / 6
Resto o módulo		%	68 % 2
Incremento unitario		++	7++
Decremento unitario		--	6--

Operadores Aritméticos en C/C++

Incremento (++)

- Para **n = 2**, la ejecución de la sentencia

`i = n++;` */* post-incremento*/*

asigna el valor 2 a i y después incrementa (en 1) el valor de n, asignándole el valor 3

`i = 2`

`n = 3`

- La ejecución de la sentencia

`i = ++n;` */*pre-incremento*/*

incrementa (en 1) el valor de n, es decir le asigna el valor 4 y después asigna este valor a i

`n = 4`

`i = 4`

Operadores Aritméticos en C/C++

Decremento (--)

- Para **n = 5**, la ejecución de la sentencia
`i = n--;` /* **post-decremento** */

asigna el valor **5** a **i** y después decrementa (en 1) el valor de **n**
a 4

i = 5
n = 4

- La ejecución de la sentencia
`i = --n;` /* **pre-decremento** */

decrementa (en 1) el valor de **n a 3** y después asigna este valor
a i

n = 3
i = 3

Operadores Aritméticos en C/C++

Incremento/ decremento

👉 Siempre que pueda, no utilice **++a;** o **--a;** aunque son operaciones válidas - *son confusas* -

✓ Puede utilizar **incrementos/ decrementos** en una expresión, por ejemplo

$x = 5 + y ++;$

son equivalentes a

$x = 5 + y;$

$y ++;$

Operadores Aritméticos en C/C++

- Los operadores **+**, **-**, *****, **/** se pueden usar con operandos (*constante, variable o expresión*) tanto enteros como reales
- La división de enteros da como resultado un entero
$$17/5 = 3$$
- El operador **módulo** (**%**) sólo puede ser utilizado con operandos enteros
- La expresión **a % b** da como resultado el **resto** de la **división entera** de dos números **a** y **b**
$$17\%5 = 2$$

Operadores Aritméticos en C/C++

Ejemplos

- $a + b * c$
- $(\text{votos/electores}) * 100$
- $a * x * x + b * x + c$
- $(\text{alto} - \text{bajo}) / 2$
- $\text{horas} * 3600 + \text{minutos} * 60 + \text{segundos}$
- $\text{numero} \% 2$

Operadores Aritméticos en C/C++

Los *operandos* de una expresión aritmética deben satisfacer lo siguiente:

- ✓ Si *ambos operandos* son de tipo *entero* entonces el *resultado* es de tipo *entero*
- ✓ Si uno de los *operandos* o ambos operandos son de tipo *real* entonces el *resultado* es de tipo *real*
- ✓ Si la expresión evaluada da como resultado un valor de tipo entero y la variable a la cual se le va a asignar este valor es de tipo real, entonces el resultado será convertido automáticamente al tipo real antes de realizar la asignación

Operadores Aritméticos en C/C++

Los ***operandos*** de una expresión aritmética deben satisfacer lo siguiente:

- ✓ Si la expresión evaluada da como resultado un valor de tipo *real* y la variable a la cual se le va a **asignar** este valor es de tipo *entero*, entonces el resultado será convertido automáticamente al tipo *entero* **antes** de realizar la asignación

Precedencia de los Operadores Aritméticos en C/C++

+ ()

Sub-expresiones encerradas entre paréntesis se evalúan primero

(+ **mayor** nivel de precedencia)

++ --

* / %

+ -

(- **menor** nivel de precedencia)

Regla 1:

La evaluación de los operadores con la misma precedencia se realiza de **izquierda a derecha**

Expresiones Aritméticas: Ejemplo 1

Se toman los ~~dos~~ primeros ~~operadores~~, si son de igual jerarquía, se realiza el más a la izquierda

Regla 2:

Evaluar la siguiente expresión aritmética:

$$(a + b + c + d + e) \% 5$$

$$R1 = a + b$$

$$R2 = R1 + c$$

$$R3 = R2 + d$$

$$R4 = R3 + e$$

$$R5 = R4 \% 5$$

Resultado

++ --
* / %
+ -

Expresiones Aritméticas: Ejemplo 1

Evaluar la expresión aritmética

$$(a + b + c + d + e)\%5$$

para los valores:

$$a = 3, b = 2, c = 1, d = 8, e = 4$$

$$R1 = a + b = 3 + 2 = 5$$

$$R2 = R1 + c = 5 + 1 = 6$$

$$R3 = R2 + d = 6 + 8 = 14$$

$$R4 = R3 + e = 14 + 4 = 18$$

$$R5 = R4 \% 5 = 18\%5 = 3$$

++ --
* / %
+ -

Expresiones Aritméticas: Ejemplo 2

Evaluar la siguiente expresión aritmética:

$$a \% 2 / b - c * a * y + 18 / c - 3 * h$$

$$R1 = a \% 2$$

$$R2 = R1 / b$$

$$R3 = c * a$$

$$R4 = R3 * y$$

$$R5 = R2 - R4$$

$$R6 = 18 / c$$

$$R7 = R5 + R6$$

$$R8 = 3 * h$$

$$R9 = R7 - R8$$

Se toman los dos primeros operadores, si son de diferente jerarquía, se mira el tercer operador y se realiza el de mayor jerarquía que esté más a la izquierda

Resultado

Regla 3:

Expresiones Aritméticas: Ejemplo 2

Evaluar la expresión aritmética

$$a \% 2 / b - c * a * y + 18 / c - 3 * h$$

para los valores:

$$a = 3, b = 2, c = 1, h = 8, y = 4$$

$$R1 = a \% 2 = 3 \% 2 = 1$$

$$R2 = R1 / b = 1 / 2 = 0$$

$$R3 = c * a = 1 * 3 = 3$$

$$R4 = R3 * y = 3 * 4 = 12$$

$$R5 = R2 - R4 = 0 - 12 = -12$$

$$R6 = 18 / c = 18 / 1 = 18$$

$$R7 = R5 + R6 = -12 + 18 = 6$$

$$R8 = 3 * h = 3 * 8 = 24$$

$$R9 = R7 - R8 = 6 - 24 = -18$$

++ --
* / %
+ -

Expresiones Aritméticas: Ejemplo 3

Los paréntesis se resuelven completamente antes de realizar cualquier operación fuera de ellos, siguiendo las reglas dadas

Evaluar la siguiente expresión aritmética:

Regla 4:

$$a * (b + c) + c * (d + e)$$

++ --
* / %
+ -

$$R1 = b + c$$

$$R2 = a * R1$$

$$R3 = d + e$$

$$R4 = c * R3$$

$$R5 = R2 + R4$$

Resultado

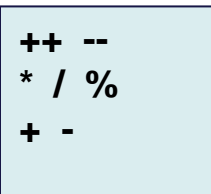
Expresiones Aritméticas: Ejemplo 3

Evaluar la siguiente expresión aritmética

$$a * (b + c) + c * (d + e)$$

para los valores:

$$a = 3, b = 2, c = 1, d = 8, e = 4$$



$$R1 = b + c = 2 + 1 = 3$$

$$R2 = a * R1 = 3 * 3 = 9$$

$$R3 = d + e = 8 + 4 = 12$$

$$R4 = c * R3 = 1 * 12 = 12$$

$$R5 = R2 + R4 = 9 + 12 = 21$$

Expresiones Aritméticas: Ejemplo 4

Evaluar la siguiente expresión aritmética:

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

Regla 5:

El paréntesis más interno se resuelve primero, siguiendo las reglas dadas

++ --
* / %
+ -

Expresiones Aritméticas: Ejemplo 4

Evaluar la siguiente expresión aritmética:

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

$$R1 = b + c$$

$$R2 = a * R1$$

$$R3 = 2 * a$$

$$R4 = R2 - R3$$

$$R5 = 4 * d$$

$$R6 = R5 - f$$

$$\mathbf{R7 = R4 + R6}$$

Regla 5:

El paréntesis más interno se resuelve primero, siguiendo las reglas dadas

++ --
* / %
+ -

Resultado

Expresiones Aritméticas: Ejemplo 4

Evaluar la expresión aritmética

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

para los valores:

$$a = 3, b = 2, c = 1, d = 8, f = 4$$

++ --
* / %
+ -

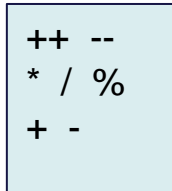
Expresiones Aritméticas: Ejemplo 4

Evaluar la expresión aritmética

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

para los valores:

$$a = 3, b = 2, c = 1, d = 8, f = 4$$



$$R1 = b + c = 2 + 1 = 3$$

$$R2 = a * R1 = 3 * 3 = 9$$

$$R3 = 2 * a = 2 * 3 = 6$$

$$R4 = R2 - R3 = 9 - 6 = 3$$

$$R5 = 4 * d = 4 * 8 = 32$$

$$R6 = R5 - f = 32 - 4 = 28$$

$$\mathbf{R7 = R4 + R6 = 3 + 28 = 31}$$

Expresiones Aritméticas: Ejemplo 5

Convertir grados Fahrenheit a grados Centígrados

Análisis E-P-S

Entradas: Temperatura en grados Fahrenheit (**fahr** $\in \mathbb{R}$)

Proceso: Calcular la temperatura en grados Centígrados

$$\text{centígrados} = (\text{fahr} - 32.0) \times 5.0/9.0$$

Salidas: Temperatura en grados centígrados (**centígrados** $\in \mathbb{R}$)

Algoritmo

0. Inicio
1. Escribir ("Introduzca la temperatura Fahrenheit: ")
2. Leer (fahr)
3. centigrados = (fahr - 32.0) \times (5.0/9.0)
4. Escribir ("Temperatura equivalente en °C es ", centigrados)
5. Fin

Algoritmo en TDSO

Identificación del programa

Algoritmo

fahrenheit2centigrados		
{pre: ninguna}		{pos: centigrados $\in \mathbb{R}$ }
1	Escribir "Introduzca la temperatura Fahrenheit "	•fahr: Real. Temperatura en grados Fahrenheit •centigrados: Real Temperatura equivalente en grados Centígrados
2	Leer fahr	
3	centigrados = (fahr - 32.0) * (5.0/9.0)	
4	Escribir "Temperatura equivalente en °C ", centigrados	
1	fahr = 72.0 -> centi = 22.0	Caso exitoso Caso exitoso Caso exitoso
2	fahr = 22.0 -> centi = -2.2	
3	fahr = 0.0 -> centi = -17.6	

Casos de prueba

Documentación de las variables y casos de prueba

Numeración de pasos y de casos de prueba

Expresiones Aritméticas: Ejemplo 5

Codificación

```
/*Convertir grados Fahrenheit a grados Centígrados*/
```

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
    const float MULT = 5.0/9.0, SUB = 32.0;
```

```
    float fahr, centigrados;
```

```
    printf ("Introduzca la temperatura Fahrenheit: %f\n");
```

```
    scanf ("%f", &fahr);
```

```
    centigrados = (fahr - SUB) * MULT;
```

```
    printf ("Temperatura equivalente en °C es %f\n" , centigrados);
```

```
}
```

Expresiones Aritméticas: Ejemplo 5

Corrida en frío (prueba de ejecución a mano)

temperaturas en grados Fahrenheit: 72.0,
28.0, 80.0, 110.0

mult	sub	fahr	centi
0.55	32.0	72.0	22.0
0.55	32.0	28.0	-2.2
0.55	32.0	80.0	26.4
0.55	32.0	110.0	42.9

Expresiones Aritméticas: Ejemplo 6

Convertir bolívares a dólares

Análisis E-P-S

Entradas: Cantidad en bolívares ($\text{bolivares} \in \mathbb{R}$),
valor del dólar ($\text{dolar} \in \mathbb{R}$)

Proceso: Calcular la cantidad equivalente de
bolívares en dólares

$$\text{dolares} = \text{bolivares} \div \text{dolar}$$

Salida: Cantidad en dólares ($\text{dolares} \in \mathbb{R}$)

Diseño

bolivaresAdolares		
{pre: ninguna}		{pos: dolares $\in \mathbb{R}$ }
1	Escribir “Introduzca valor de la tasa de cambio ”	• dolar : Real. Tasa de cambio actual
2	Leer dolar	
3	Escribir “Introduzca la cantidad en bolivares “	• bolivares : Real Cantidad en bolívares que se desea cambiar
4	Leer bolivares	
5	dolares = bolivares / dolar	• dolares : Real. Cantidad en dolares equivalente a la cantidad en bolívares
6	Escribir bolivares, “Bs. equivalen a \$ ”, dolares	

Diseño

bolivaresAdolares		
{pre: ninguna}		{pos: dolar $\in \mathbb{R}$ }
1	Escribir “Introduzca valor de la tasa de cambio ”	• dolar : Real. Tasa de cambio actual
2	Leer dolar	• bolivares : Real
3	Escribir “Introduzca la cantidad en bolivares “	Cantidad en bolívares que se desea cambiar
4	Leer bolivares	• dolares : Real. Cantidad en dolares equivalente a la cantidad en bolívares
5	dolares = bolivares / dolar	
6	Escribir bolivares, “Bs. equivalen a \$ ”, dolares	
1	dolar = 4.3 bolivares = 1000.0 -> dolares = ?	Resultado ?
2	dolar = 1.0, bolivares = 1000.0-> dolares = ?	Resultado ?
3	dolar = -1.0, bolivares = 50.0 -> dolares = ?	Resultado ?
4	dolar = 9.0, bolivares = 60000.0 -> dolares = ?	Resultado ?
5	dolar = 0.0, bolivares = 10000.0 -> dolares = ?	Resultado ?
		División por cero!!!???

Expresiones Aritméticas: Ejemplo 6

Codificación ...a completar

/* Convertir bolivares a dolares*/

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
    float dolar, bolivares, dolares;
```

```
    Escribir.... "Introduzca valor de la tasa de cambio" ; /*dejar una linea*/
```

```
    leer....dolar;
```

```
    Escribir .... "Introduzca la cantidad en bolivares: "; /*dejar una linea*/
```

```
    leer ... bolivares;
```

```
    dolares = bolivares / dolar;
```

```
    escribir... bolivares ..... "Bs. equivalen a $" ... dolares; /*dejar una linea*/
```

```
}
```


Expresiones Aritméticas: Ejemplo 7

Dada la ecuación $y = 3x - 1$, calcular el valor de y_1 y y_2 para dos valores enteros dados x_1 y x_2

Análisis E-P-S

Entradas: $x_1 \in \mathbb{Z}, x_2 \in \mathbb{Z}$

Proceso: Calcular $y_1 = 3x_1 - 1$ y $y_2 = 3x_2 - 1$

Salidas: $y_1 \in \mathbb{Z}, y_2 \in \mathbb{Z}$

Algoritmo

0. Inicio
1. Escribir (“Introduzca los valores de x_1 y x_2 ”)
2. Leer (x_1, x_2)
3. $y_1 = 3x_1 - 1$
4. $y_2 = 3x_2 - 1$
5. Escribir (“ $x_1=$ “, x_1 , “;”, “ $y_1=$ “, y_1)
6. Escribir (“ $x_2=$ “, x_2 , “;”, “ $y_2=$ “, y_2)
7. Fin

Diseño

Ejemplo7

{pre: ninguna}

{pos: $x1, x2, y1, y2 \in \mathbb{Z}$ }

1	Escribir "Introduzca los valores de x1 y x2 "
2	Leer x1, x2
3	$y1 = 3 * x1 - 1$
4	$y2 = 3 * x2 - 1$
5	Escribir "X1= ", x1, " , Y1= ", y1
6	Escribir "X2= ", x2, " , Y2= ", y2

- x1, x2**: Entero. Valores de entrada
- y1, y2**: Entero. Valores calculados correspondientes a **x1** y **x2**, respectivamente

Diseño

Ejemplo7

{pre: ninguna}

{pos: $x1, x2, y1, y2 \in \mathbb{Z}$ }

1	Escribir "Introduzca los valores de x1 y x2 "
2	Leer x1, x2
3	$y1 = 3 * x1 - 1$
4	$y2 = 3 * x2 - 1$
5	Escribir "X1= ", x1, " , Y1= ", y1
6	Escribir "X2= ", x2, " , Y2= ", y2

- x1, x2**: Entero. Valores de entrada
- y1, y2**: Entero. Valores calculados correspondientes a x1 y x2, respectivamente

1	$x1 = 0, x2 = 0 \rightarrow y1 = -1, y2 = -1$
2	$x1 = 4, x2 = 3 \rightarrow y1 = 11, y2 = 8$
3	$x1 = -5, x2 = -4 \rightarrow y1 = -16, y2 = -13$
4	$x1 = -8, x2 = 10 \rightarrow y1 = -25, y2 = 29$

Caso exitoso
Caso exitoso
Caso exitoso
Caso exitoso

Expresiones Aritméticas: Ejemplo 7

Codificación – revisar si hay errores y completar

```
/* Calcular  $y1 = 3x1 - 1$  y  $y2 = 3x2 - 1$  */
```

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
    int x1, y1, x2, y2;
```

```
    printf("Introduzca los valores de x1 y x2\n");
```

```
    scanf("%d %d", &x1, &x2);
```

```
    y1 = 3*x1 - 1;
```

```
    y2 = 3*x2 - 1;
```

```
    printf("x1=%d ,y1=%d", x1, y1);
```

```
    scanf("x2=%d ,y2=%d", x2, y2);
```

```
}
```

Operadores de Asignación en C/C++

Operador	Expresión	Expresión abreviada
=	$a = a + 2$	
+=	$x = x + 5$	$x += 5$
-=	$y = y - z$	$y -= z$
*=	$a = a * b$	$a *= b$
/=	$y = y / 2$	$y /= 2$
%=	$mes = mes \% 2$	$mes \% = 2$

La asignación **$A += (B * C) + (A * D)$** es válida!!!

Operadores de Asignación en C/C++

Ejemplos de los operadores de asignación

<code>a += b;</code>	◦	<code>a = a + b;</code>
<code>x -= y;</code>	◦	<code>x = x - y;</code>
<code>pot *= 2;</code>	◦	<code>pot = pot * 2;</code>
<code>x /= y + 1;</code>	◦	<code>x = x / (y + 1);</code>
<code>suma %= 4;</code>	◦	<code>suma = suma % 4;</code>

Sentencia de Asignación Múltiple

$A = B = C = D = 1$

asigna a las variables **A, B, C y D** el valor de **1**

Use asignación múltiple solamente
para asignar el mismo valor a
diferentes variables

Recordar que....

Hemos visto **cuatro maneras de asignar valores a las variables:**

variable = valor constante **$x = 3$**

variable = variable **$x = y$**

variable = expresión **$x = (y + 1)/2$**

Leer (variable) **Leer (x) ° scanf ("%f", &x);**

Expresiones Aritméticas

Ejercicio 1

- Evaluar cada una de las siguientes expresiones aritméticas:

a) $A \% B + C / D - 6$

b) $A + 2 * (3 + B)$

c) $3 * (A \% (B / C)) + 5$

d) $6 * 5 / 10 * 2 + 10$

e) $(6 * 5) / (10 * 2) + 10$

f) $(6 * 5) / (10 * 2 + 10)$

g) $(6 * 5) / (10 * (2 + 10))$

h) $A * B / C * D$

Expresiones Aritméticas

Ejercicios 2

- Evaluar la expresión

$$4 / 2 * 3 / 6 + 6 / 2 / 1 / 5 \% 2 / 4 * 2$$

- Escribir las siguientes expresiones algebraicas como expresiones en \mathbb{C}

a) $4x - 2y + 7$

b) $\frac{a + b}{c - d}$

c) $\frac{3x + 2y}{2z}$

d) $\frac{x + y}{x} - \frac{3x}{5}$

e) $\frac{y_2 - y_1}{x_2 - x_1}$

Expresiones Aritméticas

Ejercicios 3

- Si $A = 4$, $B = 5$ y $C = 1$, evaluar las siguientes expresiones

a) $B * A - B * B / 4 * C$

b) $(A * B) / 3 * 3$

c) $((((B + C) / 2 * A + 10) * 3 * B) - 6$

- Evaluar la siguiente expresión para $A = 2$ y $B = 5$
 $3 * A - 4 * B / A \% 2$

Expresiones Aritméticas

Ejercicio 4

- Dadas las declaraciones:

`float w;`

`int i, j = 2, k = 4;`

Determinar el valor de las variables `i`, `w` y `k` después de

la ejecución de las siguientes sentencias de asignación:

`i = j / k;`

`w = i / j;`

`k = i % j;`

`w = 8.0 / j;`

Expresiones Aritméticas

Ejercicios 5

1- Realice el análisis E-P-S, diseño y codificación en C para resolver este problema:

- La fuerza de atracción entre dos cuerpos es igual al producto de una constante k por el cociente que resulta de dividir el producto de las masas de los cuerpos por el cuadrado de su distancia.

Realice la corrida en frío para los valores de $k = 0,1$, masa del cuerpo 1 = 3 kg., masa del cuerpo 2 = 5 kg., y la distancia entre los dos cuerpos = 2,5 m

2- Realice el análisis E-P-S, diseño y codificación en C para resolver este problema:

- Calcular el número total de hojas que tiene un árbol, si por ejemplo, tiene doscientos ochenta y cuatro ramas y sabiendo que cada rama tiene como promedio trescientas cuarenta y siete hojas

Expresiones Aritméticas

Ejercicios 6

Realice el análisis E-P-S, diseño y codificación en C para resolver los siguientes problemas:

1. Multiplicar tres números enteros x, y, z
1. Calcular el promedio de seis notas enteras
2. Si un cuerpo pesa a kp (kilopondios) en un lugar en el que la gravedad es $g = 9.8 \text{ m/s}^2$, calcular su masa

Expresiones Aritméticas

Ejercicio 7

Realice el **análisis E-P-S, diseño** y codificación en **C** para resolver este problema:

- Dado el peso de una persona en libras, calcular su peso en kilogramos y gramos
- Leer cuatro números reales. Calcular y escribir su producto, su suma y su media aritmética
- Leer el radio de un círculo y calcular e imprimir su superficie y la longitud de la circunferencia