Operadores

martes, 31 de agosto de 2021 07:09 a. m.

Son palabras o símbolos que implican una acción sobre ciertas variables y constantes

Los operadores pueden ser unarios (trabajar con una variable), binarios (trabajar dos variables) o ternarios (tres variables)

Se clasifican de la siguiente forma:

- Aritméticos
- Relacionales
- Lógicos
- De bits
- De asignación
- De dirección

Aritméticos

Suma	+	7+2=9
Resta	-	8-1=7
Multiplicación	*	8*2=16
División	/	4/5=0.8
Módulo	%	4%2=0 5%2=1
Cast	(Tipo de Dato)	(double) 3 = 3.0

*Regularmente retornan un valor numérico

Relacionales A=5, B=6

	Operador	Ejemplo	Resultado
Igual	==	A==B	False (0)
Diferente de	!=	A!=B	True (1)
Mayor que	>	A>B	False(0)
Menor que	<	A< B	True (1)
Mayor o igual que	>=	A>=B	False(0)
Menor o igual que	<=	A<=B	True(0)

Siempre se retorna un Boolean [True (1) or False (0)]

Tabla de verdad And

A	B	Resultado
0	0	0
0	1	0
1	0	0
1	1	1

Operación AND (&)

A 0000 0010 2
B 0000 0100 4

R 0000 0000 0

Lógicos A=2 y B=4

	Operador	Ejemplo	Resultado
And	&	A & B	0
Or	|	A | B	6
Not	!	!True !False	False True
And corto circuito	&&	(A!=B) && (B>7) True (1) && Falso(0)	Falso
Or corto circuito	||	(A==B) || (A<5) False(0) || True (1)	True(1)

Tabla de verdad Or

A	B	Resultado
0	0	0
0	1	1
1	0	1
1	1	1

Operación Or (|)

A 0000 0010 2
B 0000 0100 4

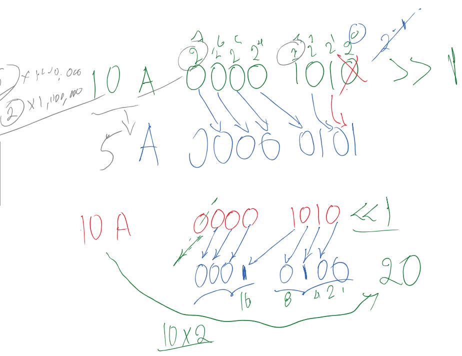
R 0000 0110 6

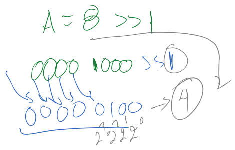
Operadores de Asignación

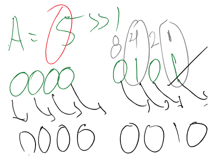
Nombre	Operador	Ejemplo con Abreviatura	Ejemplo Sin Abreviatura
Asignación	=	C=7	B=8
Autoincremento	++	A++; ++A;	A=A+1;
Autodecremento	--	A--; --A;	A=A-1;
Incremento	+=	A+=N --> A+=10	A=A+N; A=A+10;
Decremento	-=	A-=N --> A-=7	A=A-N; A=A-7;
Multiplicación	*=	A*=N ---> A*=2	A=A*N -->A=A*2;
Módulo	%=	A%=N ---> A%=1	A=A%1;

Operadores de bits A=10

Corrimiento a la derecha	>>	A >> N; N es un entero que indica cuantas posiciones recorrer (Divisiones sobre 2)	A>>1	5
Corrimiento a la izquierda	<<	A << N; N es un entero que indica cuantas posiciones recorrer (Multiplicaciones por 2)	A<<1	20
Xor, xnor, nand, nor, etc.	^			







Constantes

martes, 31 de agosto de 2021 08:33 a. m.

Para la constantes tenemos dos caminos:

Utilizando la palabra reservada "const":

```
//Sintaxis de declaración e inicialización
const <tipo dato> identificadorConstante =<<valor>>;

Ejemplo:
    const int CONSTANTE = 33;
```

Utilizando una macro(se colocan en la cabecera de los archivos .c)

```
// Sintaxis de declaración e inicialización
#define IDENTIFICADOR <<valor>>

Ejemplo:
    #define PI 3.141592
```

En cuestión de rendimiento es mejor utilizar una macro para definir constantes