Toán tử trong Go

1. Toán tử số học

Với A := 10 và B := 20

#	Operator	Description	Example
1	+	Phép cộng	A + B = 30
2	-	Phép trừ	A - B = -10
3	*	Phép nhân	A * B = 200
4	/	Phép chia	B / A = 2
5	%	Lấy phần dư	B % A = 0
6	++	Tăng giá trị thêm 1	A++ = 11
7	--	Giảm giá trị đi 1	A-- = 9

2. Toán tử so sánh

Với A := 10 và B := 20

#	Operator	Description	Example
1	==	So sánh bằng nhau	(A == B) is not true.
2	!=	So sánh khác nhau	(A != B) is true.
3	>	So sánh lớn hơn	(A > B) is not true.
4	<	So sánh nhỏ hơn	(A < B) is true.
5	>=	So sánh lớn hơn hoặc bằng	(A >= B) is not true.
6	<=	So sánh nhỏ hơn hoặc bằng	(A <= B) is true.

3. Toán tử logical

Với A := true và B := false

#	Operator	Description	Example
1	&&	AND	(A && B) is false.
2	||	OR	(A || B) is true.
3	!	NOT	!(A && B) is true.

4. Toán tử bitwise

#	p	q	p & q	p | q	p ^ q
1	0	0	0	0	0
2	0	1	0	1	1
3	1	1	1	1	0
4	1	0	0	1	1

Ví dụ: Với A := 60 và B:= 13

```
A = 0011 1100
B = 0000 1101
-----------------
A&B = 0000 1100
A| B = 0011 1101
A^B = 0011 0001
~A  = 1100 0011
```

#	Operator	Description	Example
1	&	Binary AND Operator copies a bit to the result if it exists in both operands.	(A & B) will give 12, which is 0000 1100
2	|	Binary OR Operator copies a bit if it exists in either operand.	(A | B) will give 61, which is 0011 1101
3	^	Binary XOR Operator copies the bit if it is set in one operand but not both.	(A ^ B) will give 49, which is 0011 0001
4	<<	Binary Left Shift Operator. The left operands value is moved left by the number of bits specified by the right operand.	A << 2 will give 240 which is 1111 0000
5	>>	Binary Right Shift Operator. The left operands value is moved right by the number of bits specified by the right operand.	A >> 2 will give 15 which is 0000 1111

5. Toán tử gán

#	Operator	Description	Example
1	=	Simple assignment operator, Assigns values from right side operands to left side operand	C = A + B will assign value of A + B into C
2	+=	Add AND assignment operator, It adds right operand to the left operand and assign the result to left operand	C += A is equivalent to C = C + A

3	-=	Subtract AND assignment operator, It subtracts right operand from the left operand and assign the result to left operand	C -= A is equivalent to C = C - A
4	*=	Multiply AND assignment operator, It multiplies right operand with the left operand and assign the result to left operand	C *= A is equivalent to C = C * A
5	/=	Divide AND assignment operator, It divides left operand with the right operand and assign the result to left operand	C /= A is equivalent to C = C / A
6	%=	Modulus AND assignment operator, It takes modulus using two operands and assign the result to left operand	C %= A is equivalent to C = C % A
7	<<=	Left shift AND assignment operator	C <<= 2 is same as C = C << 2
8	>>=	Right shift AND assignment operator	C >>= 2 is same as C = C >> 2
9	&=	Bitwise AND assignment operator	C &= 2 is same as C = C & 2
10	^=	bitwise exclusive OR and assignment operator	C ^= 2 is same as C = C ^ 2
11	|=	bitwise inclusive OR and assignment operator	C |= 2 is same as C = C | 2

6. Toán tử khác

#	Operator	Description	Example
1	&	Returns the address of a variable.	&a; provides actual address of the variable.
2	*	Pointer to a variable.	*a; provides pointer to a variable.

7. Các ưu tiên của toán tử trong Go

#	Category	Operator	Associativity
1	Postfix	() [] -> . ++ - -	Left to right
2	Unary	+ - ! ~ ++ - - (type)* & sizeof	Right to left
3	Multiplicative	* / %	Left to right
4	Additive	+ -	Left to right
5	Shift	<< >>	Left to right
6	Relational	< <= > >=	Left to right
7	Equality	== !=	Left to right
8	Bitwise AND	&	Left to right

9	Bitwise XOR	^	Left to right
10	Bitwise OR	|	Left to right
11	Logical AND	&&	Left to right
12	Logical OR	||	Left to right
13	Conditional	?:	Right to left
14	Assignment	= += -= *= /= %=>>= <<= &= ^= |=	Right to left
15	Comma	,	Left to right

Revision #6
Created 23 September 2019 16:22:57 by Admin
Updated 17 March 2021 13:28:02 by Laptrinh.vn