Operadores de bit

Os operadores de bit tem comportamentos bem estranhos porém necessários para algumas operações. Eles trabalham bit a bit com o valor passado.

Temos 4 operadores dentre eles

* & - Similar ao comparador AND;
* | - Similar ao OU;
* ~ - Similar ao NOT;
* ^ - Similar ao XOR.

A diferença de operadores lógicos e operadores bit

Aqui estão alguns exemplos. Vamos assumir que temos as duas seguintes variáveis:

i = 15

j = 22

Se assumirmos ambém que os valores Integers foram guardados em 32 bits, o valor deles em binário seria igual a :

i: 00000000000000000000000000001111

j: 00000000000000000000000000010110

E então nós aplicamos a seguinte linha:

log = i and j

considerando que ambos são diferentes de 0, o resultado apresentado será True, então:

log:True

agora se performarmos a operação em bit

bit = i & j

O operador & vai aplicar a lógica and para cada par de bit correspondente separadamente, produzindo o seguinte resultado:

i - 00000000000000000000000000001111

j -00000000000000000000000000010110

bit = i & j - 00000000000000000000000000000110

Esses bits correspondem ao valor Integer 6.

A mesma lógica é aplicada para os outros operadores. Por exemplo o ~ ou NOT:

i = 00000000000000000000000000001111

bitneg = ~i 11111111111111111111111111110000

Também temos o >> e o <<, que apontando para esquerda divide o número por 2 e para direita multiplica o número por 2.