

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ CENTRO DE
CIÊNCIAS, TECNOLOGIA E PRODUÇÃO
CURSO DE CIÊNCIAS DA COMPUTAÇÃO**

RICARDO MAGNO YOMURA

OPERAÇÕES DE BITWISE E SHIFT DE BITS

CURITIBA

2025

SUMÁRIO

INTRODUÇÃO..... 3

CAPÍTULO 1..... 3

CAPÍTULO 2..... 3

CONCLUSÃO..... 5

REFERÊNCIAS..... 6

INTRODUÇÃO

Durante o processo de programação de softwares, foi se desenvolvendo formas de otimizar os códigos, para que os programas utilizem cada vez menos da capacidade do computador. O *Bitwise* o *Shift de Bit* têm um papel crucial na melhora de performance dos códigos, uma vez que permitem a interação direta entre programa e o próprio bit na forma mais simples.

CAPÍTULO 1

BITWISE

O Bitwise trabalha bit a bit, que ao invés da sua forma ser decimal, é representada pela sequência binária. Os operadores são:

Tabela 1: Operadores Bitwise

OPERADOR	DESCRIÇÃO	EXEMPLO
AND Sinal "&"	Compara bit a bit dos números, quando na sequência, os dois números são 1, ele retorna 1.	12 = 1100 5 = 0101 12 & 5 = 4 (pois 4 = 0100)
OR Sinal " "	Compara bit a bit, quando na sequência um dos números é 1, ele retorna 1.	12 = 1100 5 = 0101 12 5 = 13 (pois 13 = 1101)
XOR Sinal "^"	Caso os bits comparados sequencialmente forem diferentes, retorna 1.	12 = 1100 5 = 0101 12 ^ 5 = 9 (pois, 9 = 1001)
NOT Sinal "~"	O Not inverte todos os bits	12 = 1100 ~12 = 0011

CAPÍTULO 2

SHIFT DE BIT

As transições feitas pelos Shift de Bits são definidas por operadores que podem ser classificados entre si, com sinal ou sem sinal. Aqueles com sinal, podem representar tanto números positivos quanto negativos, aqueles sem sinal, representarão apenas números positivos. Para definir a quanto deve ser deslocado, deve-se buscar o número desejado (representado por "x" na tabela 2) e escrever o operador junto à quantidade de bits a serem movidos.

É de suma importância ressaltar que os operadores de deslocamento funcionam apenas para os tipos: "int", "ulong", "uint" e "long".

Tabela 2: Operadores de Shift de Bit

OPERADOR	DESCRIÇÃO	EXEMPLO
<<	Faz com que o último bit seja deslocado para a esquerda, e é influenciado pelo sinal.	x = 0010 x << 2; x = 1000 (podendo ser negativo)
>>	Faz com que o último bit seja deslocado para a direita, e é influenciado pelo sinal.	x = 0010 x >> 1; x = 0001 (podendo ser negativo)
<<< ou >>>	Atua da mesma forma que os operadores ">>" e "<<", entretanto não é influenciado pelo sinal.	x = 10001100 x <<< 3 x = 01100000 (não podendo ser negativo)

Além disso, existem propriedades para os Shift, como as apresentadas na citação do livro *"computer organization and design"* (livro em inglês), de David A. Patterson e John L. Hennessy: *"Shift left logical fornece um benefício bônus. Deslocar para a esquerda por i bits dá o mesmo resultado que multiplicar por 2ⁱ, assim como deslocar um número decimal por i dígitos é equivalente a multiplicar por 10ⁱ."* (Pg. 88).

A utilização desses operadores (Bitwise e Shift) é amplamente aplicada em tarefas que necessitam de precisão e velocidade, como sistemas embarcados, criptografias, entre outros. Entretanto, nas linguagens de programação mais modernas, em que há recursos mais eficientes que substituem os operadores, isso ocorre, pois na atualidade é priorizada a facilidade e legibilidade dos códigos.

Exemplo:

Na linguagem de python, é comum usarem as "Listas" ou "Arrays", que substituem o "AND" e o "OR", como no exemplo da Tabela 3.

Tabela 3: Exemplo de diferença entre AND e Lista.

AND	if Y >= 20 <u>and</u> X >= 4000:
Lista	<u>Variável</u> = [Y >= 20, X >= 4000]

CONCLUSÃO

Pode-se concluir então, que com o desenvolvimento de softwares mais avançados, será cada vez menor a necessidade do uso de Bitwise e de Shift de Bits, sendo substituídos por formas mais simples e incorporadas de cada linguagem. Entretanto, ainda é uma forma poderosa para simplificar e modificar o código utilizando diretamente o bit. Sendo os operadores AND, OR e NOT úteis para tipos de linguagens antigas. Não apenas isso, mas o << e >> podem ajudar no desenvolvimento de operações matemáticas, encurtando a quantidade de linhas.

REFERÊNCIAS:

MICROSOFT, Learn.Microsoft, “Operadores bit a bit e de deslocamento (referência do C#)”. Disponível em: [Operadores bit a bit e de deslocamento – Executam operações booleanas \(AND, NOT, OR, XOR\) e de deslocamento em bits individuais em tipos integrais - C# reference | Microsoft Learn](#)

Curso de Programação, Programae, “O que é operador bitwise em programação?”. Disponível em: [O que é operador bitwise em programação?](#)

DIOGO, Matheus, “Conheça os operadores Bitwise (Bit-a-Bit)”. Disponível em: [Conheça os operadores Bitwise \(Bit-a-Bit\) | iMasters](#)

PATTERSON, David A. e HENNESSY, John L., *Computer Organization and Design*, 5ª Edição. Publicado pela editora: Morgan Kaufmann Publishers.