

# Portugol Studio - Dominando a Lógica de Programação

Rodrigo R. Silva

{Portugol  Studio}

# Bitwise AND

## Bitwise AND

Muito semelhante ao operador lógico 'e', o operador binário AND, ou conjunção binária devolve um bit 1 sempre que ambos operandos forem '1', conforme podemos confirmar pela tabela verdade, onde A e B são bits de entrada e S é o bit-resposta, ou bit de saída:

## Bitwise AND

Muito semelhante ao operador lógico 'e', o operador binário AND, ou conjunção binária devolve um bit 1 sempre que ambos operandos forem '1', conforme podemos confirmar pela tabela verdade, onde A e B são bits de entrada e S é o bit-resposta, ou bit de saída:

A	B	S
0	0	0
0	1	0
1	0	0
1	1	1

Sua sintaxe é o operador '&' entre os dois inteiros.

Sua sintaxe é o operador '&' entre os dois inteiros.

Para se fazer a operação Bitwise 5 AND 3

0101 (decimal 5)

AND

0011 (decimal 3)

-----

0001 (decimal 1)

# Bitwise OR

## Bitwise OR

O operador binário OR, ou disjunção binária devolve um bit 1 sempre que pelo menos um dos operandos seja '1', conforme podemos confirmar pela tabela de verdade, onde A e B são os bits de entrada e S é o bit-resposta, ou bit de saída:



## Bitwise OR

O operador binário OR, ou disjunção binária devolve um bit 1 sempre que pelo menos um dos operandos seja '1', conforme podemos confirmar pela tabela de verdade, onde A e B são os bits de entrada e S é o bit-resposta, ou bit de saída:

A	B	S
0	0	0
0	1	1
1	0	1
1	1	1

Sua sintaxe é o operador '|' (Digito de Canalização (em inglês: pipe)) entre os dois inteiros.

Sua sintaxe é o operador '|' (Digito de Canalização (em inglês: pipe)) entre os dois inteiros.

Para se fazer a operação Bitwise 5 OR 3

0101 (decimal 5)

OR

0011 (decimal 3)

---

0111 (decimal 7)

# Bitwise NOT

## Bitwise NOT

Muito semelhante ao operador lógico 'não', o operador unário NOT, ou negação binária devolve um bit 1 sempre que ambos operandos forem '1', conforme podemos confirmar pela tabela de verdade, onde A é o bit de entrada e S é o bit-resposta, ou bit de saída:

## Bitwise NOT

Muito semelhante ao operador lógico 'não', o operador unário NOT, ou negação binária devolve um bit 1 sempre que ambos operandos forem '1', conforme podemos confirmar pela tabela de verdade, onde A é o bit de entrada e S é o bit-resposta, ou bit de saída:

A	S
0	1
1	0

Sua sintaxe é o operador  $\sim$  entre os dois inteiros.

Sua sintaxe é o operador  $\sim$  entre os dois inteiros.

Para se fazer a operação Bitwise NOT 7

0111 (decimal 7)

NOT

---

1000 (decimal 8)



# OBRIGADO!



[github.com/Prof-Rodrigo-Silva](https://github.com/Prof-Rodrigo-Silva)



[@rodrigo-rosa-da-silva-472928138](#)

{Portugol  Studio}