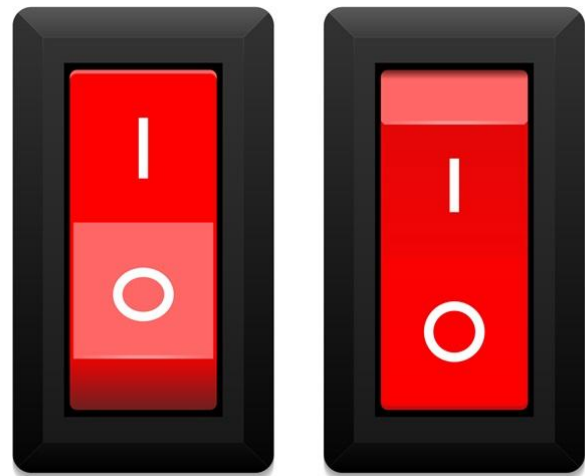


Operações lógicas com números binários

Algoritmos, Lógica e Linguagens de Programação



Nossa tabela "mágica"

128	64	32	16	8	4	2	1
-----	----	----	----	---	---	---	---



Utilize a tabela para:

Converter os seguintes números para decimal:

- a) 110001
- b) 11110000
- c) 10101010
- d) 00110

Utilize a tabela para:

Converter os seguintes números para **binário:**

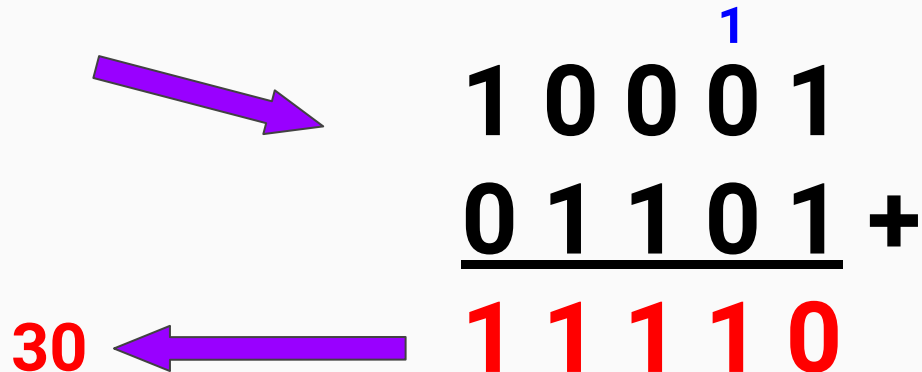
- a) 95
- b) 198
- c) 251
- d) 127

Cálculos binários

Com números binários realizamos qualquer tipo de cálculo que possa ser realizado com números decimais.

Ex: $10001 = 17$

$01101 = 13$


$$\begin{array}{r} 10001 \\ 01101 \\ \hline 11110 \end{array} +$$

30

Realize os seguintes cálculos:

a) $110001 + 010010$

b) $11110000 + 00011111$

c) $1010 + 1111$

d) $00110 + 10$

Operações lógicas

- A lógica binária, ou bitwise operation é a base de todo o cálculo computacional.
- Na verdade, são estas operações mais básicas que constituem todo o poderio dos computadores.
- Qualquer operação, por mais complexa que pareça, é traduzida internamente pelo processador para estas operações.

Operações lógicas

- Para iniciar, veremos 3 operadores lógicos:
 - **NOT (Negação)**
 - **AND (E)**
 - **OR (OU)**

Operador NOT

- O operador **NOT**, ou negação binária é o operador que recebe como entrada apenas um valor, e sua função é simplesmente inverter os valores.
- Ou seja, se o valor de entrada for VERDADEIRO, o resultado será FALSO e se o valor de entrada for FALSO, o resultado será VERDADEIRO.
- Símbolo de representação: \sim

Operador NOT

- **Ex:**

- $\sim 1 = 0$ (Lê-se: negação de um igual a zero)
- $\sim 0 = 1$
- $\sim 1001 = 0110$
- $\sim 101010 = 010101$

Operador AND

- O Operador **AND** (**E**) resulta em um valor VERDADEIRO se os dois valores de entrada da operação forem VERDADEIROS, caso contrário o resultado é FALSO.
- Símbolo de representação: **\wedge**

Operador AND


Tabela de resultados:

- $0 \wedge 0 = 0$
- $0 \wedge 1 = 0$
- $1 \wedge 0 = 0$
- $1 \wedge 1 = 1$

Ex:

$$\begin{array}{r} 10010 \\ \underline{11011} \wedge \\ 10010 \end{array}$$

Operador OR

- O Operador **OR** (**OU**) resulta em um valor VERDADEIRO se ao menos UM dos dois valores de entrada da operação for VERDADEIRO, caso contrário o resultado é FALSO.
- Símbolo de representação: 

Operador OR

Tabela de resultados:

- $0 \vee 0 = 0$
- $0 \vee 1 = 1$
- $1 \vee 0 = 1$
- $1 \vee 1 = 1$

Ex:

1 0 0 1 0

1 1 0 1 1 \vee

1 1 0 1 1

Exercícios

a) $10011 \wedge 11101$

b) $11101 \vee 00010$

c) ~ 0001111

d) $00101 \wedge 10101$

e) $10001 \vee 10010$

f) ~ 10101111

g) $\sim (0101 \wedge 1001)$

h) $1010 \vee (\sim 1000)$

i) ~ 23

j) $33 \wedge 41$

k) $\sim (110 \vee 121)$