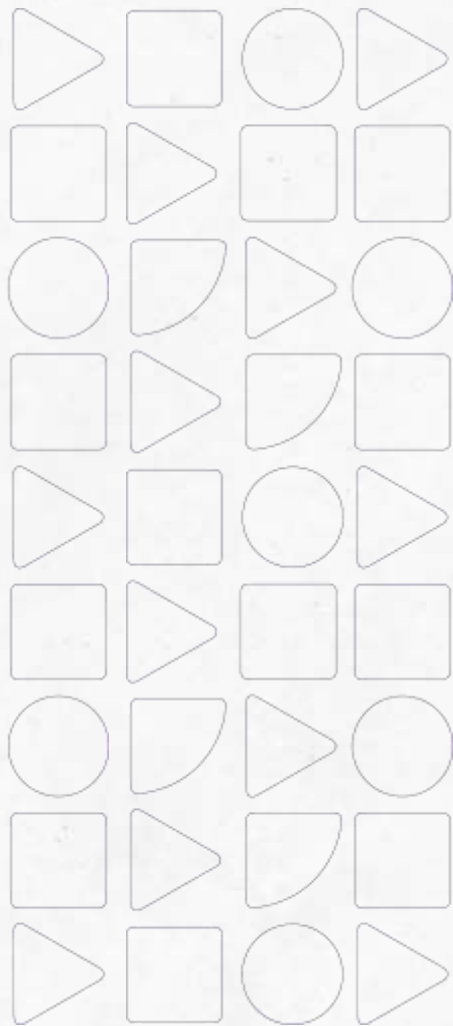




**Sistema operacional: lógica booleana**

**Disciplina:** Tecnologia e Sistemas de Informação



## Conteúdo:

Sistema operacional - Lógica Booleana.

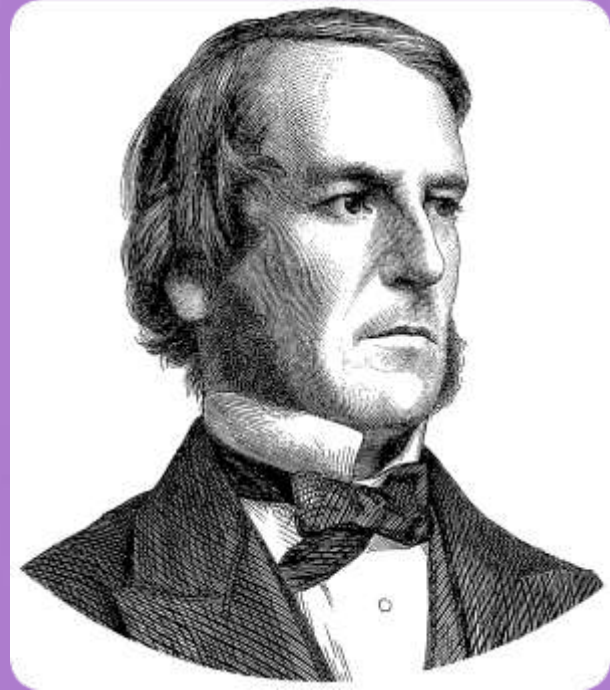
## Habilidades:

- Conhecer a importância da lógica Booleana para o funcionamento do sistema;
- Compreender as operações lógicas de alto nível e suas regras.

# Bloco 1

▶ ◻ ○ ▶ ◻ ○ ▶ ◻ ○ ▶ ◻ ○ ▶ ◻ ○ ▶ ◻ ○ ▶ ◻ ○ ▶

# Lógica Booleana, esse é um nome estranho?



## Estrutura lógica moderna

George Boole, conhecido como o pai da estrutura lógica moderna, é um matemático que fez uma grande diferença na história.

Por volta de 1847, ele aplicou um sistema lógico de números entre 0 e 1, conhecido hoje como números binários.



Mas...

Diferente de nós, humanos, as máquinas só conseguem assimilar as informações através de códigos, que chamamos de códigos binários.

**Mas... Por que o computador consegue trabalhar apenas com 0 e 1?**





## O sistema binário

O sistema binário é mais **eficiente e mais preciso** por possuir apenas **dois números em sua composição**. Por isso, é mais fácil da máquina interpretar e enviar respostas baseadas nessa codificação.

Os **números binários também são representativos, assim como os decimais**.

Por exemplo, o número 138 representa uma centena (100) uma dezena (30) e uma unidade (8).

Nós podemos converter os números binários em números decimais.

## Convertendo números binários em decimais:

Por exemplo, a conversão do número binário **11010** se dá sempre da **direita para esquerda (inversamente proporcional a nossa escrita)**.

**Precisamos multiplicar o número por dois e elevá-lo a sua posição.**

Assim:

- $(0 \times 2^0) = 0$
- $(1 \times 2^1) = 2$
- $(0 \times 2^2) = 0$
- $(1 \times 2^3) = 8$
- $(1 \times 2^4) = 16$

Somando todos os resultados:  $0 + 2 + 0 + 8 + 16 = \mathbf{26}$ .





## Agora é com vocês! Convertam os seguintes números

Usando a mesma lógica apresentada anteriormente, convertam os números a seguir:

1. 001100
2. 0110110110
3. 010
4. 01111
5. 1001000100
6. 001000101
7. 0111011
8. 0001100011
9. 110001001001
10. 01110111000

## Gabarito

1. Para o número binário 001100:

- $(0 \times 2^0) = 0$
- $(0 \times 2^1) = 0$
- $(1 \times 2^2) = 4$
- $(1 \times 2^3) = 8$
- Somando:  $0 + 0 + 4 + 8 = 12$

2. Para o número binário 0110110110:

- $(0 \times 2^0) = 0$
- $(1 \times 2^1) = 2$
- $(1 \times 2^2) = 4$
- $(0 \times 2^3) = 0$
- $(1 \times 2^4) = 16$
- $(1 \times 2^5) = 32$
- $(0 \times 2^6) = 0$
- $(1 \times 2^7) = 128$
- $(1 \times 2^8) = 256$
- $(0 \times 2^9) = 0$
- Somando:  $0 + 2 + 4 + 0 + 16 + 32 + 0 + 128 + 256 + 0 = 438$

3. Para o número binário 010:

- $(0 \times 2^0) = 0$
- $(1 \times 2^1) = 2$
- $(0 \times 2^2) = 0$
- Somando:  $0 + 2 + 0 = 2$

4. Para o número binário 01111:

- $(1 \times 2^0) = 1$
- $(1 \times 2^1) = 2$
- $(1 \times 2^2) = 4$
- $(1 \times 2^3) = 8$
- $(0 \times 2^4) = 0$
- Somando:  $1 + 2 + 4 + 8 + 0 = 15$

5. Para o número binário 1001000100:

- $(0 \times 2^0) = 0$
- $(0 \times 2^1) = 0$
- $(1 \times 2^2) = 4$
- $(0 \times 2^3) = 0$
- $(0 \times 2^4) = 0$
- $(1 \times 2^5) = 32$
- $(0 \times 2^6) = 0$
- $(0 \times 2^7) = 0$
- $(1 \times 2^8) = 256$
- $(0 \times 2^9) = 0$
- Somando:  $0 + 0 + 4 + 0 + 0 + 32 + 0 + 0 + 256 + 0 = 292$

6. Para o número binário 001000101:

- $(1 \times 2^0) = 1$
- $(0 \times 2^1) = 0$
- $(1 \times 2^2) = 4$
- $(0 \times 2^3) = 0$
- $(0 \times 2^4) = 0$
- $(0 \times 2^5) = 0$
- $(1 \times 2^6) = 64$
- $(0 \times 2^7) = 0$
- $(0 \times 2^8) = 0$
- Somando:  $1 + 0 + 4 + 0 + 0 + 0 + 64 + 0 + 0 = 69$

7. Para o número binário 0111011:

- $(1 \times 2^0) = 1$
- $(1 \times 2^1) = 2$
- $(1 \times 2^2) = 4$
- $(0 \times 2^3) = 0$
- $(1 \times 2^4) = 16$
- $(1 \times 2^5) = 32$
- $(0 \times 2^6) = 0$
- Somando:  $1 + 2 + 4 + 0 + 16 + 32 + 0 = 55$

8. Para o número binário 0001100011:

- $(1 \times 2^0) = 1$
- $(1 \times 2^1) = 2$
- $(0 \times 2^2) = 0$
- $(0 \times 2^3) = 0$
- $(1 \times 2^4) = 16$
- $(0 \times 2^5) = 0$
- $(0 \times 2^6) = 0$
- $(0 \times 2^7) = 0$
- $(1 \times 2^8) = 256$
- $(1 \times 2^9) = 512$
- Somando:  $1 + 2 + 0 + 0 + 16 + 0 + 0 + 0 + 256 + 512 = 787$

9. Para o número binário 110001001001:

- $(1 \times 2^0) = 1$
- $(0 \times 2^1) = 0$
- $(0 \times 2^2) = 0$
- $(1 \times 2^3) = 8$
- $(0 \times 2^4) = 0$
- $(0 \times 2^5) = 0$
- $(1 \times 2^6) = 64$
- $(0 \times 2^7) = 0$
- $(0 \times 2^8) = 0$
- $(1 \times 2^9) = 512$
- Somando:  $1 + 0 + 0 + 8 + 0 + 0 + 64 + 0 + 0 + 512 = 585$

10. Para o número binário 01110111000:

- $(0 \times 2^0) = 0$
- $(0 \times 2^1) = 0$
- $(0 \times 2^2) = 0$
- $(0 \times 2^3) = 0$
- $(0 \times 2^4) = 0$
- $(1 \times 2^5) = 32$
- $(1 \times 2^6) = 64$
- $(1 \times 2^7) = 128$
- $(1 \times 2^8) = 256$
- $(1 \times 2^9) = 512$
- Somando:  $0 + 0 + 0 + 0 + 0 + 32 + 64 + 128 + 256 + 512 = 992$

## Outro formato: Binário para Decimal

$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
32	16	8	4	2	1
1	1	0	0	1	1

$32 + 16 + 2 + 1 = 51$



## Agora é com vocês! Convertam os seguintes números

Usando a mesma lógica apresentada anteriormente, convertam os números a seguir:

1. 00110
2. 011010
3. 01011
4. 01111
5. 100100
6. 00101
7. 011011
8. 10111
9. 110001
10. 0111010



## Gabarito

1. Para o número binário 00110: 6
2. Para o número binário 011010: 26
3. Para o número binário 01011: 11
4. Para o número binário 01111: 15
5. Para o número binário 100100: 36
6. Para o número binário 00101: 5
7. Para o número binário 011011: 27
8. Para o número binário 10111: 15
9. Para o número binário 110001: 49
10. Para o número binário 0111010: 106

▶ ◻ ○ ▶ ◻ ○ ▶ ◻ ○ ▶ ◻ ○ ▶ ◻ ○ ▶ ◻ ○ ▶ ◻ ○ ▶

## E agora?

Mas, será que o contrário também pode acontecer? É possível transformar um número decimal em binário?



**Vamos lá?**

**Sim! É possível. Vamos aprender isso agora!**



## Transformando o número decimal em binário

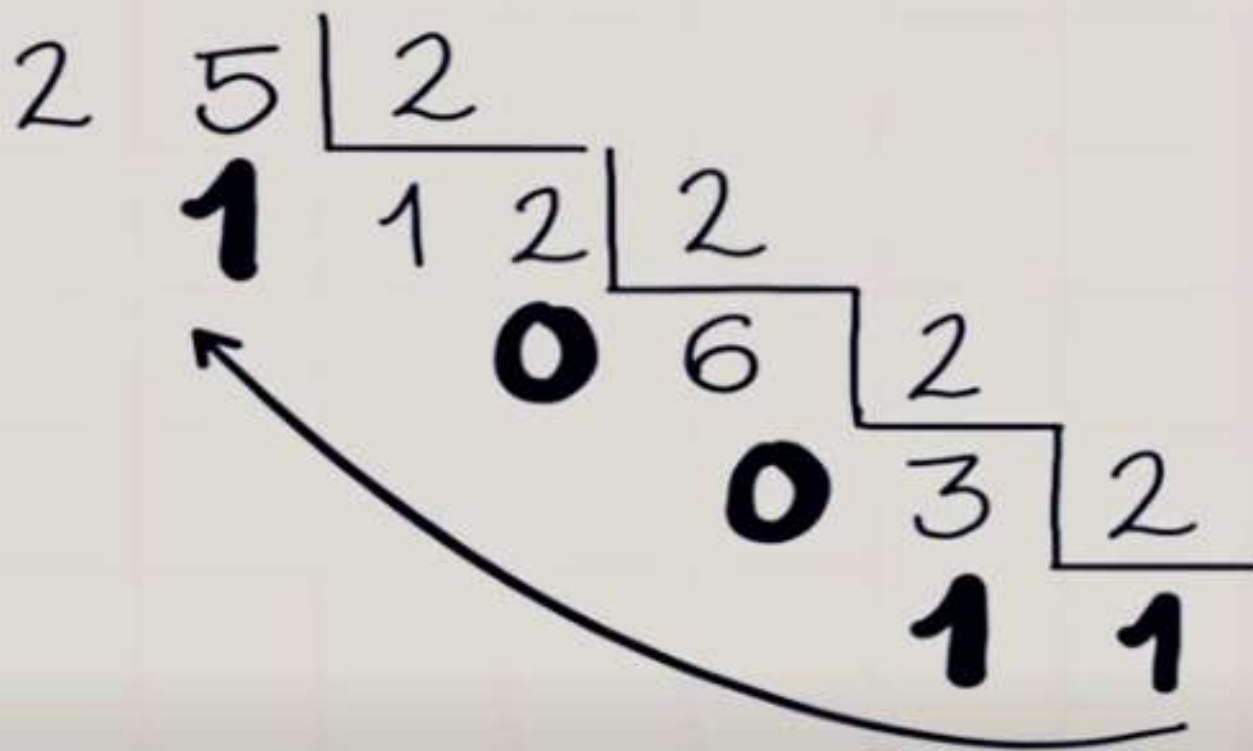
Para fazer a conversão, **nós dividimos o número decimal por 2 até não ser mais possível realizar a operação e guardamos o seu resto**. Vamos ver o exemplo do número 54:

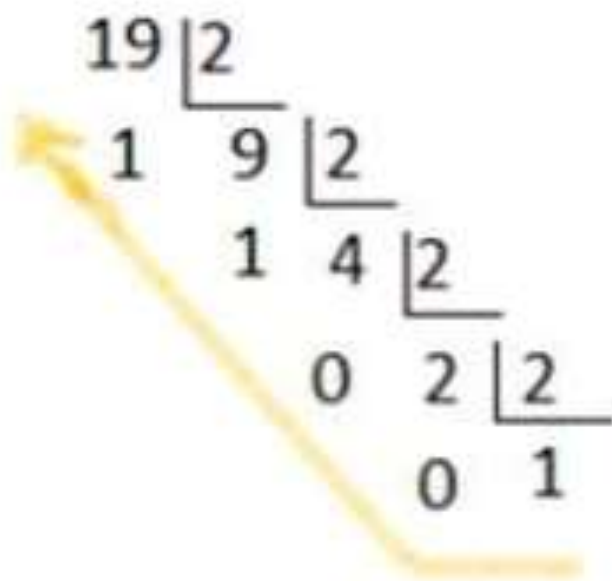
- $54/2 = 27$  e o resto é 0;
- $27/2 = 13$  e o resto é 1;
- $13/2 = 6$  e o resto é 1;
- $6/2 = 3$  e o resto é 0;
- $3/2 = 1$  e o resto é 1;
- $1/2 = 0$  e o resto é 1.

Considerando que o número binário é o **resto** das divisões, sempre **de baixo para cima**, o resultado é **110110**.

$$\begin{array}{r}
 54 \overline{) 27} \\
 \underline{0} \phantom{00} \\
 27 \overline{) 13} \\
 \underline{1} \phantom{00} \\
 13 \overline{) 2} \\
 \underline{0} \phantom{00} \\
 2 \overline{) 3} \\
 \underline{1} \phantom{00} \\
 3 \overline{) 1} \\
 \underline{1} \phantom{00}
 \end{array}$$

1 1 0 1 1 0





$$19 = 10011$$



## Agora é com vocês! convertam os seguintes números:

Usando a mesma lógica apresentada anteriormente, convertam os números a seguir:

- 12
- 23
- 30
- 72
- 64
- 92
- 41
- 86
- 27
- 98





## Gabarito

- 1) 12 em binário é: 1100
- 2) 23 em binário é: 10111
- 3) 30 em binário é: 11110
- 4) 72 em binário é: 1001000
- 5) 64 em binário é: 1000000
- 6) 92 em binário é: 1011100
- 7) 41 em binário é: 101001
- 8) 86 em binário é: 1010110
- 9) 27 em binário é: 11011
- 10) 98 em binário é: 1100010

# Bloco 2



## Responda:

Vocês já ouviram falar em *bit* ou *byte*?



## Vamos lá!

O *bit* é a menor unidade de informação em um sistema digital e representa um valor binário, podendo assumir apenas dois estados: 0 ou 1. Ele é a base do sistema binário, fundamental para a computação.

Já o *byte* é uma unidade de armazenamento de dados composta por 8 *bits*. Cada *bit* dentro de um *byte* pode ser 0 ou 1, resultando em 256 combinações possíveis ( $2^8$ ).

Ele é usado para representar caracteres, números e informações em geral.



## Importante!

**Bytes** são usados para armazenar caracteres, números, instruções de programa e outros dados nos sistemas computacionais. A quantidade de *bits* e *bytes* em um computador é usada para medir a capacidade de armazenamento e processamento de dados.



## Representação binária de cada letra

Agora, vocês irão escrever uma palavra que represente você.

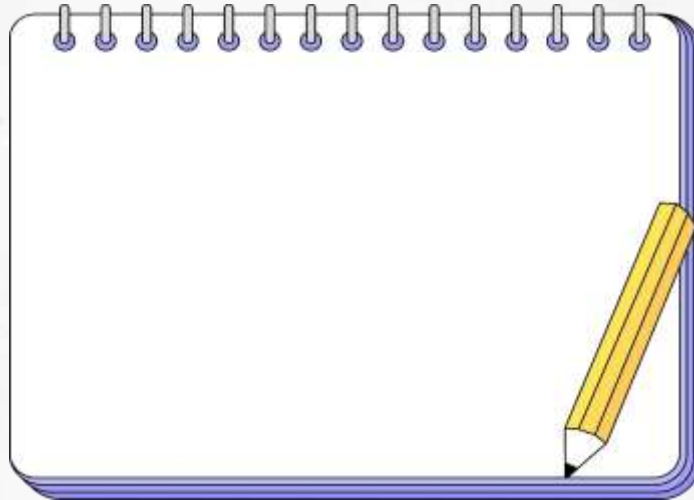
- **Exemplo:** Rafael acha que a palavra que o representa é “**Ternura**”. Transformando isso em números binários seria: “01010100 01100101 01110010 01101110 01110101 01110010 01100001”.

Não escreva a palavra na folha, só os números, pois, ao terminar, você deve entregar a folha ao seu colega para que ele adivinhe a palavra que está escrita em números binários.

Tabela ASCII			
A	1000001	N	1001110
B	1000010	O	1001111
C	1000011	P	1010000
D	1000100	Q	1010001
E	1000101	R	1010010
F	1000110	S	1010011
G	1000111	T	1010100
H	1001000	U	1010110
I	1001001	V	1010110
J	1001010	W	1010111
K	1001011	X	1011000
L	1001100	Y	1011001
M	1001101	Z	1011010

## Para finalizar, pesquisem e respondam:

1. O que significa dizer que um computador tem 200 *gigabytes*?
2. O que significa dizer que um computador tem 200 *terabytes*?
- 3.



# TABELA DE GRANDEZAS

Espaço

Medida	Base 2	Unidade
1 Bit (b) - binary digit	0 / 1	0 ou 1 - menor unidade de dado
1 Byte (B)	(2 <sup>0</sup> )	Conjunto de 8 bits
1 Kilobyte (KB)	(2 <sup>10</sup> )	1024 Bytes
1 Megabyte (MB)	(2 <sup>20</sup> )	1024 Kilobytes
1 Gigabyte (GB)	(2 <sup>30</sup> )	1024 Megabytes
1 Terabyte (TB)	(2 <sup>40</sup> )	1024 Gigabytes
1 Petabyte (PB)	(2 <sup>50</sup> )	1024 Terabytes
1 Exabyte (EB)	(2 <sup>60</sup> )	1024 Petabytes
1 Zettabyte (ZB)	(2 <sup>70</sup> )	1024 Exabytes
1 Yottabyte (YB)	(2 <sup>80</sup> )	1024 Zettabytes

Base "2" devido ao número binário - 2<sup>10</sup> (2 elevado a potência 10) = 1024.





## EXERCÍCIOS

- 1) Quantos DVD's (4,7 GB) são necessários para fazer backup de 1 TB de dados?
- 1) Quantos Pendrives de 64 GB preciso para armazenar 2,7 Terabytes de arquivos? Apresente o cálculo.
- 1) 7 Terabytes corresponde a quantos Megabytes?
- 1) Transforme 6.443.4555 bytes em Megabytes.

## Gabarito

1) Número de DVDs necessários  $\approx$  **218,3**

Como você não pode ter um pedaço de DVD, arredondamos para o próximo número inteiro.

Portanto, você precisaria de aproximadamente **219 DVDs** para fazer backup de 1 TB de dados.

2) Número de pendrives necessários = 2,700 GB / 64 GB por pendrive.

Número de pendrives necessários  $\approx$  **42,19**.

Arredondando para o próximo número inteiro, precisaríamos de aproximadamente **43 pendrives** de 64 GB para armazenar 2,7 TB de arquivos.

3) 7 terabytes correspondem a aproximadamente **7.343,872 megabytes**.

4) 6.443.4555 bytes correspondem a aproximadamente **6.14 megabytes**.

## TRABALHO PRÁTICO

Vamos criar programas que convertem automaticamente números decimais em binários e números binários em decimal?

## CONVERSOR DE DECIMAL PARA BINÁRIO

Digite um número decimal para converter para binário:

O número binário é: 10010110000

## CONVERSOR DE BINÁRIO PARA DECIMAL

Digite um número binário para converter para decimal:

A decorative grid of geometric shapes (triangles, squares, circles, and quarter-circles) arranged in a 10x4 pattern on the left side of the slide.

# Referências Bibliográficas

PROZ EDUCAÇÃO. *Apostila de Tecnologia e Sistema de Informação*. 2023