

Curso Fundamental da Linguagem C

Aula 1: Dados e Informação



1 Introdução

2 Sistemas de Numeração

3 Unidades de Informação

4 Considerações Finais

Dados VS Informação

Dado: atributo ou **parte** de uma **informação**.

Informação: conjunto de **dados estruturados** com um significado específico.

Conhecimento VS Sabedoria

Conhecimento: domínio de qualquer **informação verdadeira**.

Sabedoria: capacidade de **aplicar os conhecimentos** adquiridos da melhor forma.

As **informações** podem ser codificadas em **números** de diferentes **sistemas**:

- **Decimal**: suporta **10** dígitos: $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$;
- **Binário**: suporta **2** dígitos: $\{0, 1\}$;
- **Octal**: suporta **8** dígitos: $\{0, 1, 2, 3, 4, 5, 6, 7\}$;
- **Hexadecimal**: suporta **16** dígitos: $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}, \mathbf{E}, \mathbf{F}\}$. As letras de **A** à **F** representam os números de 10 a 15.

- **Decimais:** $2021_{(10)}$, $10_{(10)}$, $20_{(10)}$, etc.
- **Binários:** $0_{(2)}$, $1_{(2)}$, $101_{(2)}$, $0101_{(2)}$, etc.
- **Octais:** $2021_{(8)}$, $10_{(8)}$, $20_{(8)}$, etc.
- **Hexadecimais:** $2d1_{(16)}$, $10_{(16)}$, $20f_{(16)}$, etc.
- **Calculadora online:**
<https://www.rapidtables.com/convert/number/hex-to-decimal.html>
- **Tabela ASCII:**
<https://web.fe.up.pt/~ee96100/projecto/Tabela>

Nota: ASCII - American Standard Code for Information Interchange.

Calcula-se através dos restos de sucessivas **divisões inteiras** pela **base (2, 8 ou 16)**, ordenados do último ao primeiro:

Para Binário:

- $20_{(10)} \rightarrow 20/2 = 10(0) \rightarrow 10/2 = 5(0) \rightarrow 5/2 = 2(1) \rightarrow 2/2 = 1(0) \rightarrow 1/2 = 0(1)$. Portanto $20_{(10)} = 10100_{(2)}$

Para Octal:

- $100_{(10)} \rightarrow 100/8 = 12(4) \rightarrow 12/8 = 1(4) \rightarrow 1/8 = 0(1)$.
Portanto $100_{(10)} = 144_{(8)}$

Para Hexadecimal:

- $100_{(10)} \rightarrow 100/16 = 6(4) \rightarrow 6/16 = 0(6)$. Portanto
 $100_{(10)} = 64_{(16)}$

Binários:

- $101_{(2)} = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 5_{(10)}$
- $1011_{(2)} = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 11_{(10)}$
- $11111_{(2)} = 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 31_{(10)}$

Octais:

- $123_{(8)} = 1 \times 8^2 + 2 \times 8^1 + 3 \times 8^0 = 83_{(10)}$
- $137_{(8)} = 1 \times 8^2 + 3 \times 8^1 + 7 \times 8^0 = 95_{(10)}$
- $216_{(8)} = 2 \times 8^2 + 1 \times 8^1 + 6 \times 8^0 = 142_{(10)}$

Hexadecimais:

- $29_{(16)} = 2 \times 16^1 + 9 \times 16^0 = 41_{(10)}$
- $2f_{(16)} = 2 \times 16^1 + f \times 16^0 = 47_{(10)}$
- $24f_{(16)} = 2 \times 16^2 + 4 \times 16^1 + f \times 16^0 = 591_{(10)}$

- Um **dígito binário** ou **binary digit** é chamado de (**bit**);
- Com um (**bit**) podemos codificar **2** informações (**0** ou **1**);
- Com dois **bits** podemos codificar **4** informações (00, 11, 01, 10);
- Com três **bits** podemos codificar **8** informações (000, 111, 001, 110, 011, 100, 010, 101);
- Com 4 (**bits**) podemos codificar **16** informações (0000, 1111, 0001, 1110, etc.);
- Em geral podemos codificar 2^n informações com n **bits**;

Um agrupamento de **8 bits** é chamado de **Byte**, portanto com um **byte** podemos ter $2^8 = 256$ **informações**. Os **bytes** podem ser agrupados em unidades maiores:

- **Kilobyte (KB)** = 1024 **Bytes**;
- **Megabyte (MB)** = 1024 **Kilobytes**;
- **Gigabyte (GB)** = 1024 **Megabytes**;
- **Terabyte (TB)** = 1024 **Gigabytes**;
- **Petabyte (PB)** = 1024 **Terabytes**;
- **Hexabyte (HB)** = 1024 **Petabytes**;
- **Zettabyte (ZB)** = 1024 **Hexabytes**;
- **Yottabyte (YB)** = 1024 **Zettabytes**;

Siga-nos nas redes sociais

