

Introdução à Ciência da Computação  
Introdução à Engenharia de Computação



# Conversões de Base

---

Profa. Ana Marilza Pernas  
Profa. Giovani Farias  
Profa. Lisane Brisolara de Brisolara  
**Prof. Rafael Iankowski Soares**

Universidade Federal de Pelotas



# Sumário

---

## Conversões:

- Base 10 para outras bases
- Entre as Bases 2, 8 e 16
- Entre duas Bases quaisquer
- Partes Fracionárias entre Bases quaisquer



# Base de um Sistema de Numeração

---

- A **base** de um sistema é a quantidade de algarismos diferentes disponível na representação
  - Base 10 (decimal)
  - Base 2 (binário)
  - Base 16 (hexadecimal)
  - Base 8 (octal)



# Revisando

---

- Conversão de uma base qualquer para base 10
  - Método do Somatório de Produtos (Polinomial)
  - Método dos pesos

# Sistema Binário

Diagram illustrating the conversion of the binary number **11010** to decimal using the sum of products method.

The binary digits are **1 1 0 1 0**, with positions **4 3 2 1 0** indicated above them. A blue arrow points to the position labels with the word "posição".

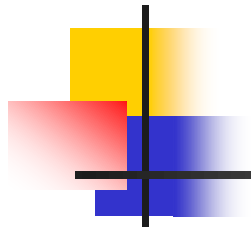
The base is **2**.

The conversion is shown as a sum of products:

$$\mathbf{1} \times 2^{\mathbf{4}} + \mathbf{1} \times 2^{\mathbf{3}} + \mathbf{0} \times 2^{\mathbf{2}} + \mathbf{1} \times 2^{\mathbf{1}} + \mathbf{0} \times 2^{\mathbf{0}}$$

The result is **= 26<sub>10</sub>**.

Legenda: Esquema que ilustra conversão de binário para decimal pelo método de somatório de produtos (descrição dos passos no próximo slide)



# **BASE 10 PARA OUTRAS BASES**



# Base 10 para outras Bases

---

- Método das divisões sucessivas
  - Divide o número pela base do sistema destino e pega o quociente e divide de novo pelo valor da base destino até que não seja mais possível fazer a divisão
  - Para montar o número pegue os restos das divisões de baixo para cima (ou da direita para a esquerda)
  - Assim sempre geraremos algarismos de 0 até a base destino (nunca maiores)



# Decimal para Binário

■ Exemplo:  $19_{(\text{base } 10)} = ??_{(\text{base } 2)}$

$$19/2 = Q:9, \mathbf{R:1}$$

$$9/2 = Q:4, \mathbf{R:1}$$

$$4/2 = Q:2, \mathbf{R:0}$$

$$2/2 = Q:1, \mathbf{R:0}$$

$$1/2 = Q:0, \mathbf{R:1}$$

$$10011_{(\text{base } 2)}$$







# Decimal para Octal

---

- Exemplo:  $83_{(\text{base } 10)} = ??_{(\text{base } 8)}$
- $83/8 = \text{Q: } 10, \text{ **R:3**}$
- $10/8 = \text{Q:1 , **R: 2**}$
- $1/8 = \text{Q:0, **R:1**}$
- **$83_{(\text{base } 10)} = 123_{(\text{base } 8)}$**



# Decimal para Hexa

---

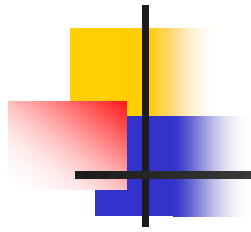
- Exemplo  $422_{(base\ 10)} = ??_{(base\ 16)}$
- $422/16 = Q:26, R:6$
- $26/16 = Q:1, R:10 \rightarrow A$
- $1/16 = Q:0, R:1$
  
- $422_{(base\ 10)} = 1A6_{(base\ 16)}$



# Decimal para Base 4

---

- Exemplo:  $19_{(\text{base } 10)} = ??_{(\text{base } 4)}$
- $19/4 = Q:4, R:3$
- $4/4 = Q:1, R:0$
- $1/4 = Q:0, R:1$
- $19_{(\text{base } 10)} = 103_{(\text{base } 4)}$



**ENTRE BASES 2, 8 E 16**



# Conversão entre as bases 2, 8 e 16

---

- Método da substituição direta
- Binário para Base 8 (Octal)
  - 3 bits são necessários para representar todos os algarismos da base 8, cujo maior algarismo é o 7 ( $111_2$ )
  - Agrupa de 3 em 3 bits, começando pelos bits menos significativos, incluindo zeros se necessário no lado direito
  - Substitui os grupos pelo algarismo octal correspondente, usando pesos do binário (1, 2, 4, 8, etc)



# Binário para Octal

---

- Ex: 10101001

- 010 101 001

└─┘ └─┘ └─┘

2      5      1

10101001 (base 2) = 251 (base 8)



# Octal para Binário

---

- Expande cada algarismo , substituindo-o por 3 bits binário
- Exemplo:  $37_{(base\ 8)}$ 
  - $7 = 111$
  - $3 = 011$
- $37_{(base\ 8)} = 011111_{(base\ 2)}$



# Conversão entre as bases 2, 8 e 16

---

- Método da substituição direta
- Binário para Base 16 (Hexa)
  - 4 bits são necessários para representar todos os algarismos da base 16, cujo maior é F (1111 em binário)
  - Agrupa de 4 em 4 bits, começando pelos bits menos significativos, incluindo zeros se necessário no lado direito
  - Substitui os grupos pelo algarismo octal correspondente, usando pesos do binário (1, 2, 4, 8, etc)





# Binário para Hexa

---

- Ex: 10110101101

- 0101 1010 1101  
    └──┘ └──┘ └──┘  
      5   10   13

Como fica em hexa?



# Binário para Hexa

---

- Ex: 10110101101

- 0101 1010 1101  
    └─┬─┘ └─┬─┘ └─┬─┘  
      5   10   13

Como fica em hexa?

10110101101 = 5AD (base 16)



# Hexa para Binário

---

- Expande cada algarismo , substituindo-o por 4 bits em binário
- Exemplo:  $3F_{(base\ 16)}$ 
  - $F = 1111$
  - $3 = 0011$
- $3F_{(base\ 16)} = 00111111_{(base\ 2)}$



# Entre as bases 8 e 16

---

- Usar a **Base 2** ou a **Base 10** como intermediária

- Base 8 -> **Base 2** -> Base 16

Ou

- Base 8 -> **Base 10** -> Base 8



# Hexa para Octal: Exemplo

- Usando a **Base 2** como intermediária

A1F<sub>(base 16)</sub> Para base 8

- 1) Substitui cada algarismo por 4 bits em binário

A    1    F

1010 0001 1111 (base 2)

- 2) Agrupa de 3 em 3 bits

101 000 011 111

- 3) Substitui cada 3 bits pelo equivalente em octal

5    0    3    7 (base 8)



# Hexa para Octal: Exemplo

A1F<sub>(base 16)</sub> Para base 8

101 000 011 111 (reagrupando os bits)

5 0 3 7 (base 8)

Passo 3)

111=7

011=3

000=0

101=5 (combinando temos 5037 na base 8)



# Octal para Hexa: Exemplo

---

- Usando a **Base 2** como intermediária
  - Substitui algarismos em octal por 3 bits binário
  - Reagrupa palavra em binário com grupos de 4 bits, cada 4 bits, substitui por algarismo correspondente em hexa



# Octal para Hexa: Exemplo

- Usando a **Base 2** como intermediária

712 (base 8)

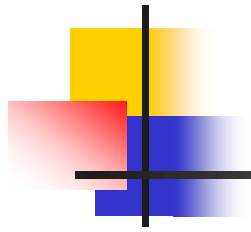
7 1 2

111 001 010 (base 2)

0001 1100 1010 (base 2)

1 C A (base 16)





**ENTRE DUAS BASES QUAISQUER**



# Conversão entre 2 bases **quaisquer**

- Converte da base de origem para a base 10 e da base 10 para a base de destino
- Exemplo:
  - base 4 -> **base 10** -> base 8

Ex: 330<sub>(base 4)</sub>

$$330_{\text{base } 4} = 0 \times 4^0 + 3 \times 4^1 + 3 \times 4^2 = \mathbf{60}_{\text{base } 10}$$

$$60/8 = Q: 7 (R: 4)$$

$$7/8 = Q:0 (R:7)$$

$$=\mathbf{74}_{(\text{base } 8)}$$



---

# **PARTE FRACIONÁRIA ENTRE BASES QUAISQUER**



# Números Fracionários

---

$$N_{10} = a_n \cdot b^n + a_{n-1} \cdot b^{n-1} + \dots + a_0 \cdot b^0, \underbrace{a_{-1} \cdot b^{-1} \dots a_{-m} \cdot b^{-m}}_{\substack{\text{Parte} \\ \text{fracionária}}}$$



# Números Fracionários

$$N_{10} = a_n \cdot b^n + a_{n-1} \cdot b^{n-1} + \dots + a_0 \cdot b^0, \underbrace{a_{-1} \cdot b^{-1} + \dots + a_{-m} \cdot b^{-m}}_{\text{Parte fracionária}}$$

Exemplo de conversão binária para decimal

$$10011,01_2 = ?$$

$$1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} = \\ 16 + 2 + 1 + 0,25 = 19,25$$



# Números Fracionários

---

Exemplo conversão decimal para binário:

15,65

Parte inteira usar divisões sucessivas

- $15/2 = Q:7, R:1$
- $7/2 = Q:3, R:1$
- $3/2 = Q:1, R:1$
- $1/2 = Q:0, R:1$

Parte inteira: 1111



# Números Fracionários

---

Exemplo conversão decimal para binário:

15,65

Parte fracionária

$$0,65 \times 2 = 1,3$$

$$0,3 \times 2 = 0,6$$

$$0,6 \times 2 = 1,2$$

$$0,2 \times 2 = 0,4$$

$$0,4 \times 2 = 0,8$$



Resultante: 1111,10100<sub>2</sub>



# Números fracionários

---

Octal para decimal

- $20,2_8 = 2 \cdot 8^1 + 0 \cdot 8^0 + 2 \cdot 8^{-1} = 16,25_{10}$





# Números fracionários

---

Decimal para octal

Ex:  $16,25_{10}$

**Parte inteira:**

- $16/8=2$  R:0
  - $2/8=0$  R:2
- ↑

**Parte fracionária:**

- $0,25 \times 8 = 2,0$
  - $0 \times 8 = 0,0$
- ↓

$$16,25_{10} = 20,2_8$$



# Binário para Octal Fracionário

---

- Conversão de binário para octal

- Ex: 11101111

- 011 101 111  
    └─┘ └─┘ └─┘  
      3   5   7

11101111 (base 2) = 357 (base 8)



# Binário para Octal Fracionário

- Conversão de binário para octal
- Ex: 1110,1111 (agrupar de 3 em 3 bits)
- $\underbrace{001}_{1} \underbrace{110}_{6}, \underbrace{111}_{7} \underbrace{100}_{4}$

1110,1111 (base 2) = 16,74 (base 8)



# Hexadecimal Fracionário

---

- Conversão de hexa para binário  
(cada dígito vira 4 bits em binário)

- $1A,3_{16}$

- 1      A,      3

- 0001 1010, 0011<sub>2</sub>



# Onde aprender mais ?

---

- [1] MONTEIRO, M. A. **Introdução à Organização de Computadores**. Rio de Janeiro: Livros Técnicos e Científicos, 1996.
- [2] WEBER, Raul F. **Fundamentos de Arquiteturas de Computadores**. Porto Alegre: Sagra-Luzzato, 2000.
- [3] UYEMURA. **Sistemas Digitais**. São Paulo: Pioneira Thomson Learning, 2002.