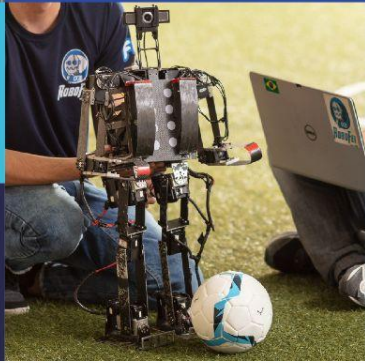




CCP010

Digital Experience





# Digital Experience

- O objetivo da disciplina Digital Experience é permitir a vivência da computação como um todo, usando como base as ODSs
  - Começamos pelo alto nível: tratando e **interpretando dados e datasets**
  - Agora a abordagem será mais de baixo nível: em **hardware e sistemas digitais**

# Sistemas Digitais

- Os **Sistemas Digitais** são a base de funcionamento dos computadores
- Um sistema digital é qualquer dispositivo que processa informações representadas em formato digital
- Operam com informações discretas, ou seja, por valores bem definidos, geralmente **binários (0 e 1)**



# Sistemas Numéricos

- **Os sistemas numéricos são fundamentais para a computação!**
- É a maneira como os computadores processam dados
- Alguns exemplos da importância dos sistemas numéricos para a computação incluem:
  - Processadores / circuitos que operam com dois estados distintos: ligado (representado por 1) e desligado (representado por 0). Comportamento modelado pelo sistema binário: 0 e 1





# Sistemas Numéricos

- O armazenamento de dados em computadores também é feito usando os estados 0 e 1
- Sistemas como o hexadecimal (base 16) e o octal (base 8), são usados como formas mais compactas de representação de grandes números binários
- Em computadores, o armazenamento é organizado em endereços de memória. Sistemas como o hexadecimal facilitam a compreensão e manipulação desses endereços



# Sistemas Numéricos

Os **sistemas numéricos** são conjuntos de símbolos e regras utilizados para representar números e realizar operações matemáticas

- Existem diferentes sistemas numéricos, cada um com sua própria **base**
- As bases numéricas mais comuns são: **Decimal**, **Binária**, **Octal** e **Hexadecimal**

# Sistemas Numéricos

- **Decimal:** 10759
- **Binário:**  $10101000000111_2$
- **Octal:**  $25007_8$
- **Hexadecimal:**  $2A07_{16}$

O número subscrito representa a base em que o número está representado. Não é utilizado para base 10

# Sistemas Numéricos

## Sistema Decimal (Base 10):

- É o sistema numérico mais usado
- Base: 10
- Dígitos: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- Cada posição em um número decimal representa uma potência de 10
  - Por exemplo, o número 254 pode ser representado assim:

$$254 = 2 \times 10^2 + 5 \times 10^1 + 4 \times 10^0$$



# Sistemas Numéricos

## Sistema Binário (Base 2):

- Usado em computadores e sistemas digitais
- Base: 2
- Dígitos: **0 e 1**
- Cada posição em um número binário representa uma potência de **2**
  - Por exemplo, o número **1011 (binário)** equivale a:

$$1011_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 11_{10}$$

# Sistemas Numéricos

## Sistema Octal (Base 8):

- Forma mais compacta de representar números binários (computação)
- Base: 8
- Dígitos: 0, 1, 2, 3, 4, 5, 6, 7
- Cada posição em um número octal representa uma potência de 8
  - Por exemplo, o número 17 (octal) equivale a:

$$17_8 = 1 \times 8^1 + 7 \times 8^0 = 15_{10}$$

# Sistemas Numéricos

## Sistema Hexadecimal (Base 16):

- Usado em programação para representar endereços de memórias
- Base: 16
- Dígitos: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
- Cada posição em um hexadecimal representa uma potência de 16
  - Por exemplo, o número 1A (hexadecimal) equivale a:

$$1A_{16} = 1 \times 16^1 + 10 \times 16^0 = 26_{10}$$

# Contagem em Diferentes Bases Numéricas

Decimal	Binário	Octal	Hexa
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Decimal	Binário	Octal	Hexa
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13
20	10100	24	14
21	10101	25	15
22	10110	26	16
23	10111	27	17
24	11000	30	18
25	11001	31	19
26	11010	32	1A
27	11011	33	1B
28	11100	34	1C
29	11101	35	1D
30	11110	36	1E
31	11111	37	1F

# Sistemas Binários - *Bit*

- Na base binária, utilizamos o termo **Bit**:
  - simplificação de Dígito Binário (*Binary Digit*)
- **É a menor unidade de informação que pode ser armazenada ou transmitida**
- Um bit pode assumir somente 2 valores: **0** e **1**
- Um conjunto de 8 bits é chamado de **byte**
- Cada byte tem 256 valores possíveis ( $2^8$ ): de **0 a 255** de decimal

# Sistemas Binários - *Bit*

- Exemplo: com 3 bits podemos representar 8 números de 0 a 7:

Binário	Decimal
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

$2^3 = 8$  números



# Conversões de Bases

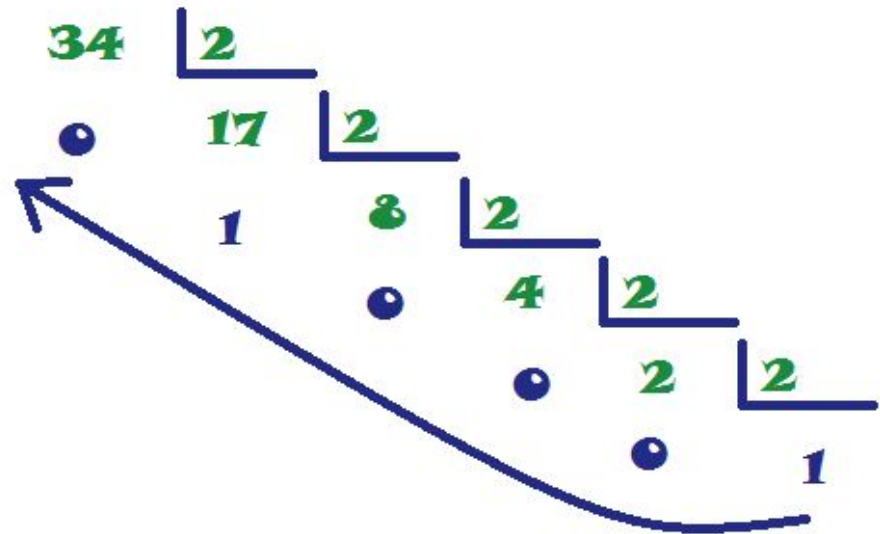
- Conversões numéricas entre bases são utilizadas em muitos casos
- Algumas razões:
  - Somos acostumados com a base numérica **decimal**
  - Sistemas Digitais, como computadores, trabalham com base **binária**
  - As bases **octal** e **hexadecimal** são muito utilizadas pela possibilidade de compactar a representação binária

# Conversões entre as Bases de Numeração

## Decimal para Binário

Conversão: Decimal para Binário

Converter o número 34 para binário



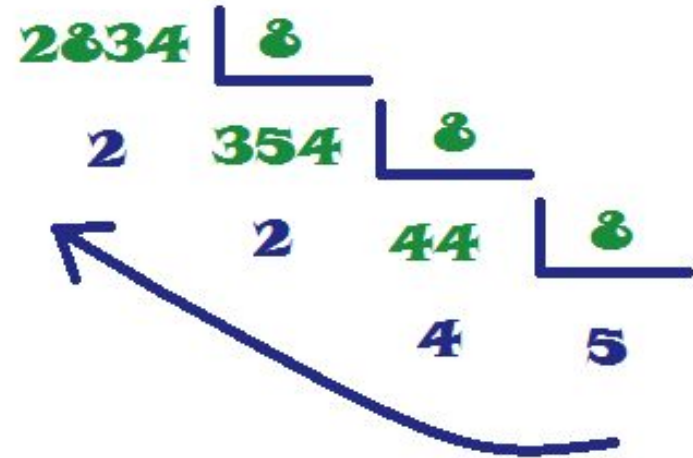
**Resultado:  $100010_2$**

# Conversões entre as Bases de Numeração

## Decimal para Octal

Conversão: Decimal para Octal

Converter o número 2834 para octal



**Resultado: 5422<sub>8</sub>**

# Conversões entre as Bases de Numeração

## Decimal para Hexadecimal

Conversão: Decimal para Hexadecimal

Converter o número 2834 para hexadecimal



# Exercício

Converter **160** para **binário**, **octal** e **hexadecimal**

# Conversões entre as Bases de Numeração

## Binário para Decimal

Conversão: Binário para Decimal

Converter o número  $100010_2$  para decimal

Somamos cada número multiplicado por **2** elevado a um número sequencial / posicional que começa de 0

$$1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$$

Resultado da somatória:

**34**



# Conversões entre as Bases de Numeração

## Octal para Decimal

Conversão: Octal para Decimal

Converter o número  $5422_8$  para decimal

Somamos cada número multiplicado por 8 elevado a um número sequencial / posicional que começa de 0

$$5 \times 8^3 + 4 \times 8^2 + 2 \times 8^1 + 2 \times 8^0$$

Resultado da somatória:

**2834**

# Conversões entre as Bases de Numeração

## Hexadecimal para Decimal

Conversão: Hexadecimal para  
Decimal

Converter o número  $B12_{16}$  para  
decimal

Somamos cada número multiplicado por **16**  
elevado a um número sequencial / posicional  
que começa de 0

$$11 \times 16^2 + 1 \times 16^1 + 2 \times 16^0$$

Resultado da somatória:  
**2834**

# Conversões entre as Bases de Numeração

## Binário para Hexadecimal

Conversão: Binário para  
Hexadecimal

Converter o número **1001101101**<sub>2</sub>  
para hexadecimal

1. separamos os dígitos em grupos de 4:

**0100 1101 1101**

2. fazemos a conversão binário-decimal para  
cada grupo separadamente:

**4 13 13**

3. trocamos os números maiores que 9 por  
letras:

**4DD**<sub>16</sub>

# Conversões entre as Bases de Numeração

## Hexadecimal para Binário

Conversão: Hexadecimal para Binário

Converter o número **4DD**<sub>16</sub> para binário

1. separamos os dígitos do número:

**4 D D**

2. fazemos a conversão das letras para números:

**4 13 13**

3. fazemos a conversão de cada número separadamente para binário:

**0100 1101 1101**<sub>2</sub>

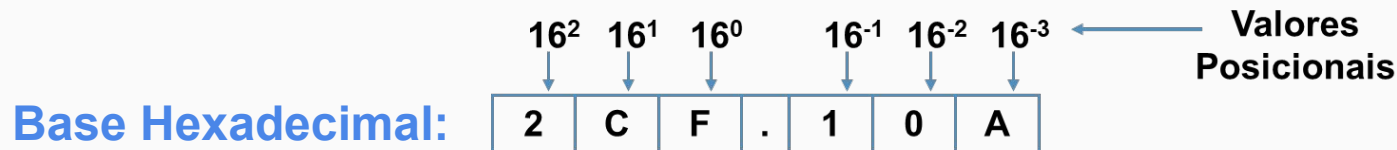
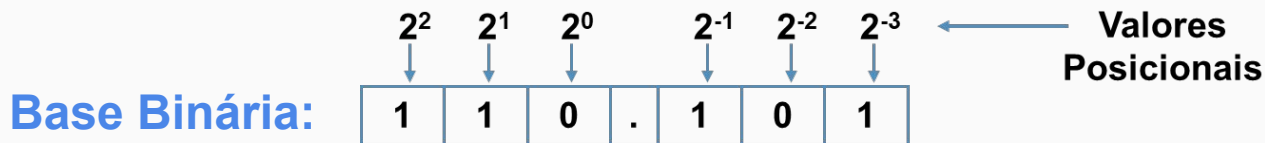
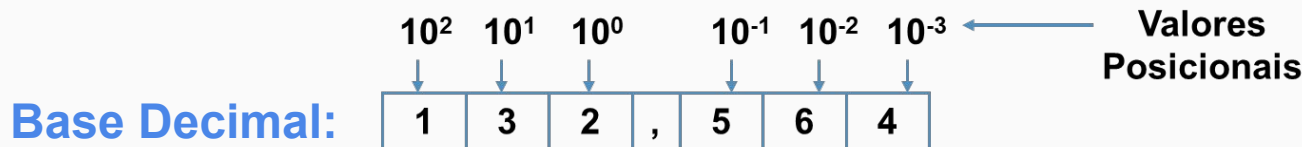
# Exercícios

Converter  $FA1_{16}$  para **decimal**

Converter  $111010_2$  para **decimal** e **hexadecimal**

# Conversões entre as Bases de Numeração

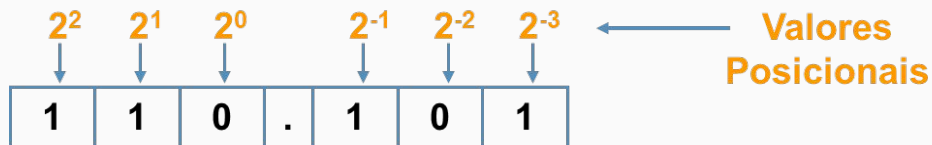
**Números fracionários:** valores posicionais com expoentes negativos





# Conversões entre as Bases de Numeração

**Números fracionários:** valores posicionais com expoentes negativos



$$110.101_2$$

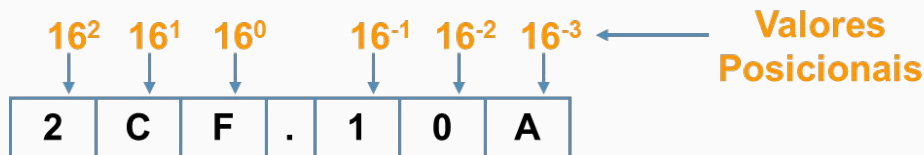
$$= 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$$

$$= 4 + 2 + 0 + 0,5 + 0 + 0,125$$

$$= 6.625$$

# Conversões entre as Bases de Numeração

**Números fracionários:** valores posicionais com expoentes negativos



$$2CF.10A_{16}$$

$$= 2 \times 16^2 + 12 \times 16^1 + 15 \times 16^0 + 1 \times 16^{-1} + 0 \times 16^{-2} + 10 \times 16^{-3}$$

$$= 512 + 192 + 15 + 0,0625 + 0 + 0,00244$$

$$= 719.0649$$

# Conversões entre as Bases de Numeração

## Números fracionários: Decimal para Binário

Conversão: Decimal Fracionário  
para Binário

Converter o número **132.564** para  
binário

$$0.564 \times 2 = 1.128$$

$$0.128 \times 2 = 0.256$$

$$0.256 \times 2 = 0.512$$

$$0.512 \times 2 = 1.024$$

$$0.024 \times 2 = 0.048$$

- a. Para a parte inteira, aplicar o método das **divisões** sucessivas por 2
- b. Para a parte fracionária, aplicar o método das **multiplicações** sucessivas por 2, até obter zero
  - a. Definir quantos bits deseja obter na parte fracionária
  - b. Multiplicar o número fracionário por 2, a parte inteira é o binário equivalente



# Conversões entre as Bases de Numeração

## Números negativos

- A conversão de números negativos para binário, por exemplo, é um processo que requer o uso de representações específicas, já que o sistema binário padrão, por si só, não lida diretamente com números negativos
- O **complemento de dois** é a técnica mais utilizada em sistemas digitais para representar números negativos

# Conversões entre as Bases de Numeração

## Números negativos

- No complemento de dois, o bit mais significativo (o primeiro bit à esquerda) representa o **bit de sinal**, onde:
  - 0 indica um número positivo
  - 1 indica um número negativo

# Conversões entre as Bases de Numeração

## Números negativos

- Exemplo de conversão: **-5** para binário de 8 bits
  1. Converter o valor absoluto do número para binário
  2. Inverter todos os bits (complemento de um)
  3. Adicionar 1 ao resultado (complemento de dois)

# Conversões entre as Bases de Numeração

## Números negativos

1. Converter o valor absoluto do número para binário (8 bits):

5:  $00000101_2$

2. Inverter todos os bits (complemento de um):

$11111010_2$

3. Adicionar 1 ao resultado (complemento de dois):

$11111011_2 = -5$

# Exercícios para entrega no Moodle

1 . Os computadores utilizam o sistema binário, ou de base 2, que é um sistema de numeração em que todas as quantidades se representam com base nos números 0 e 1.

- a) Como será a representação binária do número 2024 em um computador?
- b) Como será a representação desse mesmo número nas bases octal e hexadecimal?
- c) Se os computadores trabalham representando informações com números binários, por que estudar as bases octal e hexadecimal?



# Exercícios para entrega no Moodle

2 . Realize as seguintes conversões:

- a) 325 para binário
- b)  $10100_2$  para decimal
- c) 4554 para hexadecimal
- d)  $ABAE_{16}$  para decimal
- e)  $10111000_2$  para hexadecimal
- f) 23,1875 para binário
- g) 0,1 para binário
- h)  $11101,01_2$  para decimal
- i) 678,25 para binário
- j)  $11100,011_2$  para decimal
- k)  $A64_{16}$  para binário
- l)  $D52_{16}$  para decimal

# Exercícios para entrega no Moodle

3 . A maioria das pessoas pode contar até 10 nos dedos das mãos. Porém, cientistas da computação podem fazer melhor:

- a) Se você considerar cada dedo como um **bit binário**, com o dedo estendido indicando 1 e o dedo recolhido indicando 0, até quanto você pode contar usando as mãos?
- b) Se você considerar o dedão da mão esquerda como sendo um **bit de sinal** para números de **complemento de dois**, qual é faixa de números que é possível ser expressa dessa forma?