



# Representação de Dados

José Gustavo de Souza Paiva



# Introdução

- **Sistemas de Numeração**

- Objetivo → prover símbolos e convenções para representar quantidades, de forma a registrar a informação quantitativa e poder processá-la
- Tradicionalmente feita com números

# Sistemas de Numeração

- Método mais tradicional → numeração posicional
  - O valor representado pelo algarismo no número depende da posição em que ele aparece na representação
  - Inventado inicialmente pelos chineses
- A posição ocupada por cada algarismo em um número altera seu valor de uma potência de 10 (na base 10) para cada casa à esquerda
- Exemplo
  - No sistema decimal (base 10), no número 125 o algarismo 1 representa 100 (uma centena ou  $10^2$ ), o 2 representa 20 (duas dezenas ou  $1 \times 10^1$ ) e o 5 representa 5 mesmo (5 unidades ou  $5 \times 10^0$ ). Assim, em nossa notação,  $125 = 1 \times 10^2 + 2 \times 10^1 + 5 \times 10^0$



# Sistemas de Numeração

- **Base** de um sistema → quantidade de algarismos disponível na representação
- A base 10 é hoje a mais usualmente empregada, embora não seja a única utilizada
  - No comércio pedimos uma dúzia de rosas, e também marcamos o tempo em minutos e segundos (base 60)
- Computadores utilizam a base 2 (sistema binário) e os programadores, por facilidade, usam em geral bases que sejam potências de 2

# Sistemas de Numeração

- Algarismos
  - Base 10 → 0, 1, 2, 3, 4, 5, 6, 7, 8 e 9
  - Base 2 → 0 e 1
  - Base 16 → 0 a 9, A, B, C, D, E e F
  - Base 8 → 0 a 7
  - Generalizando, temos que uma base **b** qualquer disporá de **b** algarismos, variando entre 0 e (b - 1)
- A representação **125,38** (base **10**) significa  $1 \times 10^2 + 2 \times 10^1 + 5 \times 10^0 + 3 \times 10^{-1} + 8 \times 10^{-2}$
- Em qual base está o número 9B3?
- E o número 11?
- A resposta para o primeiro número torna-se óbvia, pois o algarismo B só existe na base 16 (hexadecimal)
- Entretanto, o segundo número pode estar em qualquer base conhecida
- Para diferenciar as bases, utiliza-se um número situado à direita inferior do número indicando a qual base ele pertence

# Sistemas de Numeração

- Generalizando, representamos uma quantidade **N** qualquer, numa dada base **b**, com um número tal como segue

- $$N_b = a_n \cdot b^n + \dots + a_2 \cdot b^2 + a_1 \cdot b^1 + a_0 \cdot b^0 + a_{-1} \cdot b^{-1} + a_{-2} \cdot b^{-2} + \dots + a_{-n} \cdot b^{-n}$$

- Sendo que

- $a_n \cdot b^n + \dots + a_2 \cdot b^2 + a_1 \cdot b^1 + a_0 \cdot b^0$  é a parte inteira
- $a_{-1} \cdot b^{-1} + a_{-2} \cdot b^{-2} + \dots + a_{-n} \cdot b^{-n}$  é a parte fracionária

# Sistemas de Numeração

- Intuitivamente, sabemos que o maior número que podemos representar, com  $n$  algarismos, na base  $b$ , será o número composto  $n$  vezes pelo maior algarismo disponível naquela base (ou seja,  $b-1$ )
  - Por exemplo, o maior número que pode ser representado na base 10 usando 3 algarismos será 999 (ou seja,  $10^3 - 1 = 999$ )
- Generalizando, podemos ver que o maior número inteiro  $N$  que pode ser representado, em uma dada base  $b$ , com  $n$  algarismos ( $n$  "casas"), será  $N = b^n - 1$ 
  - Assim, o maior número de 2 algarismos na base 16 será  $FF_{16}$  que, na base 10, equivale a  $255_{10} = 16^2 - 1$

# Sistemas de Numeração

- Resumindo, os sistemas de numeração obedecem às seguintes regras
  - A base  $B$  de um sistema é igual à quantidade de algarismos distintos utilizados
  - Quando uma posição é ocupada pelo maior algarismo e ela deve ser aumentada de uma unidade, esta posição recebe o símbolo nulo e a posição seguinte deve ser aumentada de uma unidade
  - O algarismo mais a direita tem peso um. O algarismo imediatamente a sua esquerda tem peso  $B$ , o seguinte  $B^2$ , o seguinte  $B^3$ , etc.
  - O valor de um número é determinado pela soma dos valores de cada algarismo multiplicado com seu peso



# Representação Binária

- Os computadores modernos utilizam apenas o sistema binário, isto é, todas as informações armazenadas ou processadas no computador usam apenas DUAS grandezas, representadas pelos algarismos 0 e 1
- Havendo apenas dois algarismos, portanto dígitos binários, o elemento mínimo de informação nos computadores foi apelidado de **bit** (uma contração do inglês *binary digit*).
- Na base 2, o número "10" vale dois
- Mas se  $10_2 = 2_{10}$ , então dez é igual a dois?

# Representação Binária

- Na realidade, "10" não significa necessariamente "dez" na base 2
- O número  $10_2$  seria lido "um-zero" na base 2 e vale  $2_{10}$  (convertido para "dois" na base dez)
- $10_5$  seria lido "um-zero" na base 5 e vale  $5_{10}$  (convertido para "cinco" na base dez)
- $10_{10}$  pode ser lido como "um-zero" na base 10 ou então como "dez" na base dez
- $10_{16}$  seria lido "um-zero" na base 16 e vale  $16_{10}$  (convertido para "dezesesseis" na base dez)

# Representação Binária

- Observação → o número " $10_b$ " vale sempre igual à base
  - Isso ocorre porque em uma dada base **b** os algarismos possíveis vão sempre de 0 a  $(b - 1)$ !
  - Como o maior algarismo possível em uma dada base **b** é igual a  $(b-1)$ , o próximo número será  $(b - 1 + 1 = b)$
  - Mas devemos lembrar que, quando uma posição é ocupada pelo maior algarismo e ela deve ser aumentada de uma unidade, esta posição recebe o símbolo nulo e a posição seguinte deve ser aumentada de uma unidade
  - Portanto será sempre 10 e assim, **numa dada base qualquer, o valor da base será sempre representado por "10"**

# Representação Binária

Repr.Binária	Potência	Repr.Decimal
1	$2^0$	1
10	$2^1$	2
100	$2^2$	4
1000	$2^3$	8
10000	$2^4$	16
100000	$2^5$	32
1000000	$2^6$	64
10000000	$2^7$	128
100000000	$2^8$	256
1000000000	$2^9$	512
10000000000	$2^{10}$	1.024



# Representação Binária

- A representação binária é perfeitamente adequada para utilização pelos computadores
- No entanto, um número representado em binário pode apresentar muitos bits, ficando longo e passível de erros quando manipulado por seres humanos
- Para facilitar a visualização e manipulação de grandezas processadas em computadores, são usualmente adotadas as representações octal (base 8) e principalmente hexadecimal (base 16)
- **IMPORTANTE** → O computador opera apenas na base 2 e as representações octal e hexadecimal não são usadas no computador, elas se destinam apenas à manipulação de grandezas pelos programadores



# Representação Octal/Hexadecimal

- Em projetos de informática é usual representar quantidades usando sistemas em potências do binário (octal e principalmente hexadecimal)
  - Reduz o número de algarismos da representação
  - Facilita a compreensão da grandeza e evita erros
- No sistema octal (base 8), cada três bits são representados por apenas um algarismo octal (de 0 a 7)
- No sistema hexadecimal (base 16), cada quatro bits são representados por apenas um algarismo hexadecimal (de 0 a F)

# Representação Octal/Hexadecimal

Base 10	Base 2	Base 8	Base 16
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	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

# Representação Octal/Hexadecimal

- A base 16 ou sistema hexadecimal pode ser indicada também por um "H" ou "h" após o número
  - Exemplo: FFH significa que o número FF (ou 255 em decimal) está em hexadecimal
  - Não confundir o "H" ou "h" com mais um dígito, mesmo porque em hexadecimal só temos algarismos até "F" e portanto não existe um algarismo "H"





# Referências

- Rui Mano, Organização de Computadores
  - <http://wwwusers.rdc.puc-rio.br/rmano/index.html>