

Sistemas de Numeração

Por
José Luís Carneiro



- Métodos para representar quantidades
 - Romano (I, V, X, L, C, D, M)
 - Indo-arábico (1, 2, 3, 4, 5, 6, 7, 8, 9, 0)
- Número \neq Numeral \neq Algarismo
- Sistemas de numeração posicionais
 - A posição de cada algarismo determina o seu valor
- Bases de um sistema de numeração
 - Quantidade de algarismos disponíveis
 - Há diversas bases (de acordo com a necessidade)

- Base 10 – Sistema Decimal:
 - 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- Base 2 – Sistema Binário:
 - 0, 1
- Base 8 – Sistema Octal
 - 0, 1, 2, 3, 4, 5, 6, 7
- Base 16 – Sistema Hexadecimal
 - 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

Ex. 1: sistema decimal – base 10

- Um número é composto por numerais multiplicados por potências de 10:

Cento e setenta e um **(171)**₁₀:

$$1 \times 10^0 = 1$$

$$7 \times 10^1 = 70$$

$$1 \times 10^2 = \underline{100}$$

$$171$$

Ex. 2: sistema binário – base 2

- Um número é composto por numerais multiplicados por potências de 2:

Cento e setenta e um (**10101011**)₂:

$$1 \times 2^0 = 1$$

$$1 \times 2^1 = 2$$

$$1 \times 2^3 = 8$$

$$1 \times 2^5 = 32$$

$$1 \times 2^7 = \underline{128}$$

171

Ex. 3: de decimal para binário

- Efetuamos sucessivas divisões por dois e consideramos o resto (de baixo para cima):

Seja o número 35:

$$35 = 17 \times 2 + 1$$

$$17 = 8 \times 2 + 1$$

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

$$4 = 2 \times 2 + 0$$

$$2 = 1 \times 2 + 0$$

$$1 = 0 \times 2 + 1$$

(100011)₂

Ex. 3: tabela de representação

Número	Base 10	Base 2	Base 8	Base 16
Zero	0	0	0	0
Um	1	1	1	1
Dois	2	10	2	2
Três	3	11	3	3
Quatro	4	100	4	4
Cinco	5	101	5	5
Seis	6	110	6	6
Sete	7	111	7	7
Oito	8	1000	10	8
Nove	9	1001	11	9
Dez	10	1010	12	A
Onze	11	1011	13	B
Doze	12	1100	14	C
Treze	13	1101	15	D
Quatorze	14	1110	16	E
Quinze	15	1111	17	F

- Desenvolvido em 1679 por Leibniz (co-criador do Cálculo)
- Possui apenas dois algarismos:
 - Componentes eletrônicos possuem dois estados elétricos: “ligado” ou “desligado”
 - Armazenamento e manipulação mais simples
 - Representação dos números mais longa
- Um circuito contendo um sinal elétrico é chamado de “bit” (*Binary Digit*)

Unidades de grandeza

- 8 bits = 1 byte
- Múltiplos em potências de 2 ($2^{10} = 1024$):
 - 1 Kilobyte KB = 1024 bytes (2^{10} bytes)
 - 1 Megabyte MB = 1024 KB (2^{20} bytes)
 - 1 Gigabyte GB = 1024 MB (2^{30} bytes)
 - 1 Terabyte TB = 1024 GB (2^{40} bytes)
 - 1 Petabyte PB = 1024 TB (2^{50} bytes)
 - 1 Exabyte EB = 1024 PB (2^{60} bytes)
 - 1 Zettabyte ZB = 1024 EB (2^{70} bytes)
 - 1 Yottabyte YB = 1024 ZB (2^{80} bytes)

- *Extended Binary-Coded Decimal Interchange Code*
 - Criada pela IBM em 1964
 - Utiliza 8 bits para representar um caractere, permite 256 caracteres
 - Ainda utilizada em computadores de grande porte (*mainframes*)

- *American Standard Code for Information Interchange*
 - Começou a tornar-se padrão em 1967
 - Utilizava 7 bits para representar um caractere, permitia 128 caracteres
 - Gerou diversas extensões e variações:
 - *ASCII Extended* – 256 caracteres;
 - ISO 8859 – 256 caracteres;
 - UTF-8 – mais de 1 milhão de caracteres;

- Páginas de código das páginas *Web*:
 - Pode dificultar a exibição de caracteres acentuados e símbolos especiais.
 - A codificação pode ser configurada nos navegadores (*browsers*).
- Anexos de *e-mail*, às vezes, são convertidos em seqüências de caracteres ASCII para permitir a transmissão sem perda:
 - UUencode, Base64, BinHex ou MIME.

Trecho das Instruções de Importação do programa DCTF da Secretaria da Receita Federal:

2. ESPECIFICAÇÕES TÉCNICAS

Organização: Seqüencial ASCII de Hex 20 a Hex 7E.

Delimitadores de registro (EOL): Hex 0D + Hex 0A.

Finalizador de arquivo (EOF): Hex 1A, opcional.

Tamanho de registro: variável.

Características dos registros: conforme leiaute.

Nome do arquivo: nome válido de arquivo conforme definição MS-DOS;
extensão: obrigatoriamente SRF;

Caracteres
expressos em
hexadecimal