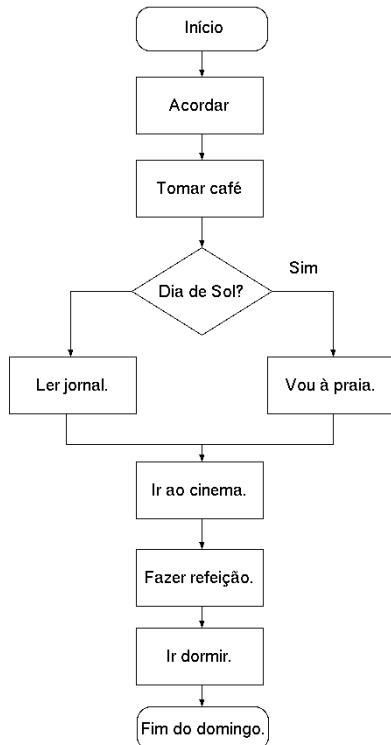


Ciência da Computação

Prof. Dr. Leandro Alves Neves

Fluxograma para um domingo



Aula 02

Sumário

- Representação de Dados
- Sistemas de Numeração
- Conversão de Dados

1. Representação de Dados

- O computador, por ser uma máquina eletrônica, só consegue processar duas informações: **presença** ou **ausência** de energia.
 - As informações Binárias (valores 0 e 1) ou Bits são representados por:
 - ~3v: Representa o Bit valor 1
 - ~0,5v: Representa o Bit valor 0
- código numérico: **BINÁRIO**

bit [de “**B**inary dig**IT**”]

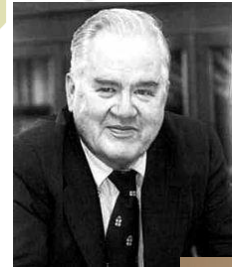
1. Representação de Dados

■ BYTE (BInary TErm)

- ❑ **Grupo ordenado de 8 bits**, para efeito de manipulação interna mais eficiente
- ❑ Tratado de forma individual, como unidade de armazenamento e transferência.



Bit (1947), termo definido por **John Wilder Tukey** (Americano), matemático



Byte (1956), termo criado por **Werner Buchholz** (Alemão), cientista da computação



<https://www.youtube.com/watch?v=LpuPe81bc2w>

1. Representação de Dados

- Letras, números e outros caracteres são codificados e decodificados pelos equipamentos via associação (Tabela).

Sistema para representação: ASCII, UNICODE e outros

Binário	Caractere
0100 0001	A
0100 0010	B
0110 0001	a
0110 0010	b
0011 1100	<
0011 1101	=
0001 1011	ESC
0111 1111	DEL

1. Representação de Dados

Bit - 2 estados: 0 e 1

Byte	B	8 bits	-
Quilobyte (ou Kilobyte)	KB	1.024 bytes	$2^{10}=1.024$
Megabyte	MB	1.024 KB	$2^{20}=1.048.576$
Gigabyte	GB	1.024 MB	$2^{30}=1.073.741.824$
Terabyte	TB	1.024 GB	$2^{40}=1.099.511.627.776$

Os valores utilizados em computação para indicar capacidade de memória e armazenamento são normalmente compostos de um número (entre 0 e 999) e uma das abreviaturas citadas (ex.: 256KB, 64MB, etc.).

2. Sistemas de numeração

- Permite **representar uma informação** usando um **conjunto de símbolos**.
- Os símbolos fazem referência a um valor numérico padronizado, chamada de “**base**”.
 - A **quantidade de diferentes símbolos** existentes num dado sistema de numeração **representa a sua base**.
- Exemplo: Sistema de numeração decimal é **composto por dez símbolos diferentes**:
0, 1, 2, 3, 4, 5, 6, 7, 8 e 9 = 10 símbolos = **base 10**

2. Sistemas de numeração

- Por exemplo: 5.031.902 na **base 10**:

$$\underline{5} \times 10^6 + \underline{0} \times 10^5 + \underline{3} \times 10^4 + \underline{1} \times 10^3 + \underline{9} \times 10^2 + \underline{0} \times 10^1 + \underline{2} \times 10^0 = 5.031.902$$

- Alguns tipos de representações possíveis:

- Decimal: base 10
- **Binário: base 2 → 2 símbolos → 0 e 1**
- Hexadecimal: base 16
- Octal: base 8

2. Sistemas de numeração

■ Sistemas de Numeração básicos:

Sistema	Base	Algarismos
Binário	2	0,1
Ternário	3	0,1,2
Octal	8	0,1,2,3,4,5,6,7
Decimal	10	0,1,2,3,4,5,6,7,8,9
Duodecimal	12	0,1,2,3,4,5,6,7,8,9,A,B
Hexadecimal	16	0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

2. Sistemas de numeração

■ Sistemas de Numeração básicos:

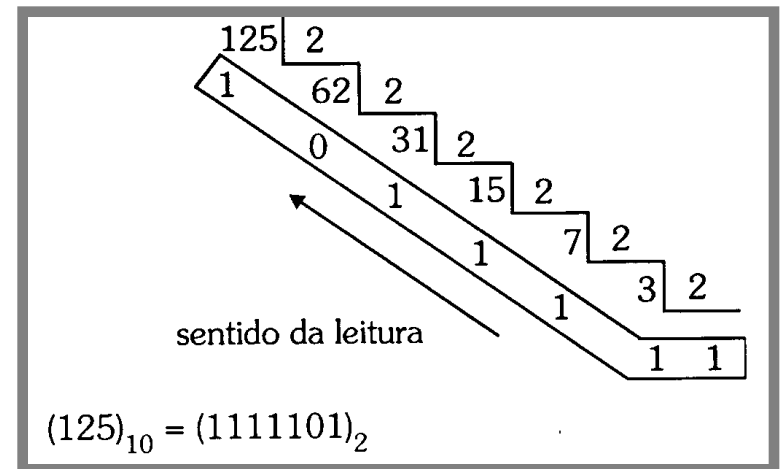
Decimal	Binário	Octal	Hexadecimal
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
.	.	.	.
.	.	.	.
.	.	.	.

3. Conversão de Dados

Decimal para Binário

- Divisão (Decimal (base 10 \Rightarrow base 2))
- Dividir o número por **b (base do sistema)** e também os resultados obtidos consecutivas vezes.

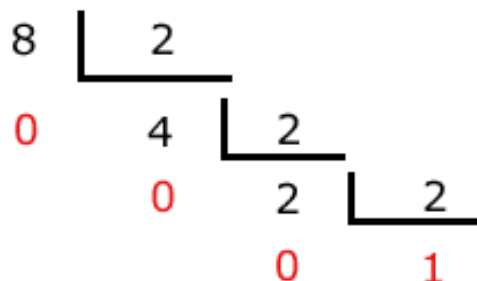
125 \Rightarrow em binário (Base 2)?



3. Conversão de Dados

Decimal para Binário: 8,375  **em binário (Base 2)?**

parte inteira:



0,500

x 2

1,000

o processo pára aqui, pois a parte do número

depois da virgula é nula.

parte fracionária:

0,375 parte fracionária

x 2 base

0,750

x 2

1,500

quando atingimos o número 1, e a parte do número após a virgula não for nula, separamos esta última e reiniciamos o processo:

Sendo assim, podemos escrever: $0,011_2 = 0,375_{10}$. Para completarmos a conversão efetuamos a composição da parte inteira com a fracionária:

$$1000,011_2 = 8,375_{10}$$

3. Conversão de Dados

Binário para Decimal

- (Binário (base 2 \Rightarrow base 10)

$$(01111101)_b \Rightarrow (?)_{dec}$$

$$0 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$

$$(00001001)_b \Rightarrow (?)_{dec}$$

$$(00001011)_b \Rightarrow (?)_{dec}$$

3. Conversão de Dados

Conversão de Números Fracionários

- Lei de Formação ampliada

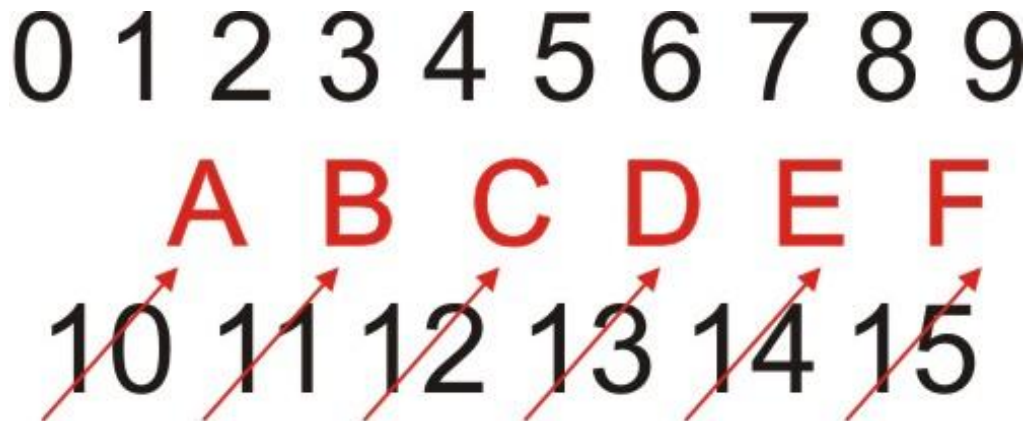
$$\text{Número} = \underbrace{a \cdot b^n + a \cdot b^{n-1} + a \cdot b^{n-2} + \dots + a \cdot b^0}_{\text{parte inteira}} + \underbrace{a \cdot b^{-1} + a \cdot b^{-2} + \dots + a \cdot b^{-m}}_{\text{parte fracionária}}$$

Exemplo: $(101,110)_b = (?)_{\text{dec}}$

$$1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3} \Rightarrow (5,75)_{\text{dec}}$$

3. Conversão de Dados

- **Representação de Números Hexadecimais:**
 - Sistema que adota 16 algarismos



3. Conversão de Dados

■ **Representação de Números Hexadecimais:**

□ Seja (1 A 7 B)

$$1 = 1$$

$$A = 10$$

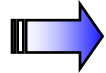

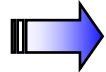

$$7 = 7$$

$$B = 11$$

3. Conversão de Dados

■ Decimal para Hexadecimal:

□ Seja 666 em decimal:

$666 / 16 = 41$		resto 10		A
$41 / 16 = 2$		resto 9		9
$2 / 16 = 0$		resto 2		2

$$666 = 29A$$

3. Conversão de Dados

■ Hexadecimal para Decimal:

□ Seja 120 em decimal:

$$\begin{array}{rclcl}
 120 / 16 = 7 & \Rightarrow & \text{resto} & 8 & \uparrow & 8 \\
 \cancel{7} / 16 = 7 & \Rightarrow & & 7 & & 7
 \end{array}
 \quad \text{Resultado: } 78$$

$$78 \text{ (hex)} \Rightarrow ?_b$$

$$7 \times 16^1 + 8 \times 16^0 = 120$$

- Até aqui vimos o seguinte:
 - A representação da informação
 - Lógica binária e informação digital
 - Codificação de informações
 - Conversões entre bases
 - **Próximo Conteúdo:**
 - Linguagem de máquina, Linguagem de alto nível.
 - Tipos de linguagens e suas características.
 - Compilação

Bibliografia

- TANENBAUM, Andrew S. (1992). Organização Estruturada de Computadores, 5ª Edição, São Paulo: Prentice/Hall.
 - Apêndices A e B

- MONTEIRO, M. A. Introdução à Organização de Computadores, Rio de Janeiro, LTC, 4 ed., 2002.
 - Capítulos 2 e 3