FUNDAMENTOS DA COMPUTAÇÃO

Prof. Dr. Ruy Ferreira (ruy@ufmt.br / prof_ruy@hotmail.com / @profruy)

Texto produzido para apoiar o ensino da disciplina-título.

SISTEMAS DE NUMERAÇÃO

Informação

O trabalho realizado no computador depende da transformação da linguagem humana para a linguagem entendida pelo hardware. Os conceitos dessa linguagem intermediária estão situados no campo da Informática. Por sua vez, a palavra Informática é origem francesa, e é formada pela contração de duas outras palavras: informação e automática.

Os seguintes conceitos são importantes no estudo de sistemas de informação:

Dado – É a menor unidade da informação.

Dados - São fatos que descrevem eventos e entidades. Os dados referem a mais de um fato. Um único fato é referido como item.

Evento - Algo que acontece em certo tempo; ocorrência significativa para um sistema de informação.

Entidade - Pessoa, lugar ou coisa; objeto de interesse para um sistema de informação.

Os dados são representados por diversos tipos de símbolos tais como letras do alfabeto, números, pontos e traços, sinais, figuras, etc. Estes símbolos podem ser arrumados e re-arrumados em diversas combinações representando fatos. Quando são arrumados de forma utilizável, denominam-se **informação**.

Informação - É um conjunto de dados significativos e relevantes que descrevem eventos ou entidades. No sentido mais comum "informação" significa fatos. No mundo da computação a informação está presente sempre que um sinal é transmitido de um lugar para outro. A informação pode ser armazenada em: livros, discos, fitas, diagramas, etc. Quando nos referimos ao armazenamento, transmissão, combinação, comparação de mensagens, dizemos que há: **Processamento de Informações ou de dados**.

DICA IMPORTANTE

Um **SISTEMA DE INFORMAÇÃO** pode ser definido como um ***conjunto de componentes inter-relacionados*** trabalhando juntos para coletar, recuperar, processar, armazenar e distribuir informações com a finalidade de facilitar o planejamento, o controle, a coordenação, a análise e o processo decisório onde for necessário.

TECNOLOGIA DA INFORMAÇÃO (TI) É a ***área de conhecimento*** responsável por criar, administrar e manter a gestão da informação através de dispositivos e equipamentos para acesso, operação e armazenamento dos dados, de forma a gerar informações para tomada de decisão.

Representação

Dois termos que aparecem com frequência na terminologia da informática são **bit** e o **byte**. Cada sinal elétrico que o computador processa é chamado de **BIT** – **Bi**nary Digi**t** e é representado por **"0"** ou **"1".** Onde: "1" → 5 volts (ligados, i.e., passando corrente elétrica); "0" → 0 volts (desligado, i.e., não passando corrente elétrica).

BIT - É a menor partícula de informação em um computador, mas um único bit não consegue representar todas as letras, números e caracteres especiais com os quais o computador trabalha. É necessário agrupá-los e cada grupo é chamado de **Byte**.

BYTE - É usualmente um grupo (conjunto) de 8 bits e equivale a um **caractere**.

Caractere - É a unidade básica de armazenamento de informação na maioria dos sistemas, ou seja, é a representação gráfica de uma letra, número ou símbolo especial do alfabeto. Um dos padrões hoje utilizados é o ASCII (American Standard Code for Information Interchange).

ASCII - É o conjunto de caracteres que contém os dígitos de 0 a 9, todas as letras minúsculas e maiúsculas, os sinais de pontuação, 32 caracteres de controle e 128 caracteres especiais que incluem frações, letras de alfabeto estrangeiro e símbolos de gráficos de linha para desenhar quadros e formas.

Os microprocessadores (chip) para fazer cálculos, comparações, entre outras operações, usam palavras de tamanho diferente, em função da tecnologia. Assim um processador pode usar palavras conforme descrito no quadro a seguir:

1 byte - 8 bits	**2 bytes - 16 bits**	**4 bytes - 32 bits**

PALAVRA - É a quantidade de bits que a **CPU** processa por vez. Naqueles de 8 bits os termos byte, caractere e palavra se confundem, pois todos têm 8 bits. Nos microprocessadores modernos já temos palavra de 16 a 64 bits, i.e., 2 a 8 bytes.

Exemplo:

Transferindo da memória para o microprocessador o vocábulo **ARTE**, em função do tamanho da palavra.

- 8 bits = 1 byte = 1 caractere por vez. Logo, necessita 4 operações, uma para cada letra.
- 16 bits = 2 bytes = 2 caracteres por vez. Logo, necessita 2 operações, uma para cada letra.
- 32 bits = 4 bytes = 4 caracteres por vez. Logo, necessita 1 operação, uma para cada letra.
- 64 bits = 8 bytes = 8 caracteres por vez. Logo, necessita 1 operação, uma para cada letra e poderia ainda transferir mais 4 caracteres.

Unidades de Medida

Tanto para quantificar a memória principal do equipamento como para indicar a capacidade de armazenamento, são usados múltiplos de bytes, como:

K - Kilo (mil) **M** - Mega (milhão)

G - Giga (bilhão) **T** - Tera (trilhão)

bit = 0 ou 1

byte = 8 bits = 2^8 = 256 combinações (números)

1 Kb = 2^{10} = 1024 bytes (1 Kilobyte)

1 Mb = 2^{20} = 1024 Kbytes = 1.048.576 bytes (1 Megabyte)

1 Gb = 2^{30} = 1024 Mbytes = 1.073.741.824 bytes (1 Gigabyte)

Sistemas de Numeração

Sistemas Posicionais são aqueles em que o valor atribuído a um símbolo depende da posição em que se encontra no conjunto no conjunto de símbolos que está representando uma quantidade.

Base: Determina a quantidade de símbolos diferentes que podem ocupar uma posição em um valor quantitativo de determinado sistema posicional.

Ex.: no Sistema decimal - Base 10 (0-9)

Sistema binário - Base 2 (0-1)

VALOR POSICIONAL OU POTÊNCIA: É o fator que altera o valor de um determinado símbolo quando ele muda de posição.

- Notação posicional

$$\ldots + a\alpha^3 + b\alpha^2 + c\alpha^1 + d\alpha^0 + e\alpha^{-1} + f\alpha^{-2} + g\alpha^{-3} + \ldots$$

Onde: a,b,d,d,e,f,g,...são os valores

α é a base

Ex.: 235,41

$2.10^2+3.10^1+5.10^0+4.10^{-1}+1.10^{-2}$

200 +30 +5 +0,4 +0,01

235,41

De uma forma geral pode-se dizer que os sistemas de numeração surgiram da necessidade humana em contar coisas. As primeiras contagens faziam a correspondência um a um, associando a cada objeto de uma coleção um objeto de outra coleção.

Quando precisamos contar uma grande quantidade de coisas, vamos separando os objetos em montes ou em grupos, pois isto facilita a contagem. Contar por dúzia, por exemplo, é uma forma de agrupar de 12 em 12. A contagem de grandes quantidades é facilitada quando se conta de forma agrupada.

Diversas civilizações da Antiguidade, além da egípcia, desenvolveram seus próprios sistemas de numeração. Alguns deles deixaram vestígios, apesar de terem sido abandonados. Assim, por exemplo, na contagem do tempo, agrupamos de 60 em 60; sessenta segundos compõem um minuto e sessenta minutos compõem uma hora. Isto é conseqüência da numeração desenvolvida na Mesopotâmia, há mais de 4000 anos. Lá era usada a base sessenta.

Essa idéia de agrupar marcas foi utilizada nos sistemas mais antigos de numeração. Os egípcios da antiguidade criaram um sistema muito interessante para escrever números, baseado em agrupamentos:

1 era representado por uma marca que se parecia com um bastão: |
2 por duas marcas ||
E assim por diante:

3	|||	7	|||||||
4	||||	8	||||||||
5	|||||	9	|||||||||
6	||||||		

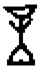
Quando chegavam a 10, eles trocavam as dez marcas: |||||||||| por ∩ , que indicava o agrupamento. Feito isso, continuavam até o 19:

10	∩	15	∩ |||||
11	∩ |	16	∩ ||||||
12	∩ ||	17	∩ |||||||
13	∩ |||	18	∩ ||||||||
14	∩ ||||	19	∩ |||||||||

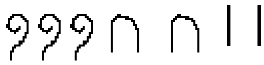
O 20 era representado por ∩ ∩ . E continuavam:

30 ∩ ∩ ∩
40 ∩ ∩ ∩ ∩
......
90 ∩ ∩ ∩ ∩ ∩ ∩ ∩ ∩ ∩

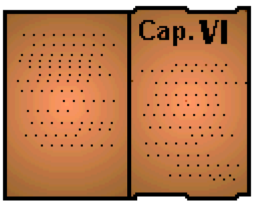
Para registrar 100, ao invés de ∩ ∩ ∩ ∩ ∩ ∩ ∩ ∩ ∩ ∩ , trocavam esse agrupamento por um símbolo novo, que parecia um pedaço de corda enrolada: 𓍢. Juntando vários símbolos de 100, escreviam 200, 300 até o 900. Dez marcas de 100 eram trocadas por um novo símbolo, a figura da flor de lótus:

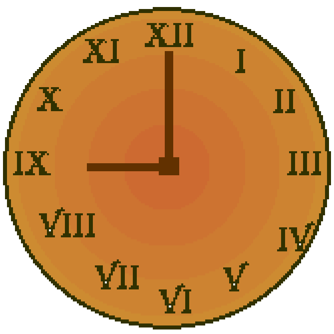
. Desta forma, trocando cada dez marcas iguais por uma nova, eles escreviam todos os números de que necessitavam. Veja os símbolos usados pelos egípcios e o que significava cada marca.

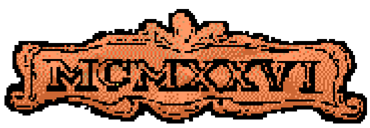
Símbolo egípcio	descrição	nosso número
	bastão	1
	calcanhar	10
	rolo de corda	100
	flor de lótus	1000
	dedo apontando	10000
	peixe	100000
	homem	1000000

Observe como eles escreviam o número 322: . Ou seja, 100 + 100 + 100 + 10 + 10 + 1 + 1. Entretanto, usando o sistema egípcio, fica trabalhoso registrar certas quantidades. Experimente, por exemplo, escrever 999 no sistema egípcio e compare com a nossa maneira de escrevê-lo.

Os romanos não eram tão bons com os números como os egípcios. Mas, criaram seu próprio sistema de numeração. Ainda hoje usamos vestígios de uma numeração antiga, são os símbolos de numeração romana, que podem ser observados nos mostradores de relógios, indicação de datas e de capítulos de livros.







Estes são os símbolos usados no sistema de numeração romano:

I	V	X	L	C	D	M
1	5	10	50	100	500	1000

Para não repetir 4 vezes um mesmo símbolo, os romanos utilizavam subtração. Observe alguns números que seriam escritos com 4 símbolos e como os romanos passaram a escrevê-los:

quatro	nove	quarenta	quarenta e quatro	novecentos
IV	IX	XL	XLIV	CM
5-1	10-1	50-10	(50-10)+(5-1)	1000-100

Assim como no sistema egípcio, também na numeração romana é trabalhoso escrever certos números. Veja:

Três mil oitocentos e oitenta e oito
MMMDCCCLXXXVIII
1000+1000+1000+500+100+100+100+50+10+10+10+5+1+1+1

O Sistema de Numeração Decimal

Baseado em 10 dígitos, o sistema decimal tem o seguinte valor posicional:

Algarismo	Posição	Valor	Obs.
5 →	2 →	$5x10^2 = 500$	A soma vale 546
4 →	1 →	$4x10^1 = 40$	
6 →	0 →	$6x10^0 = 6$	

O sistema decimal possui dez símbolos:

0 1 2 3 4 5 6 7 8 9

Operar com decimais é intuitivo:

2 + 4 = 613 - 2 = 1122 * 3 = 6645 / 5 = 9

O Sistema de Numeração Binário

É o sistema de numeração dos computadores atuais utilizado internamente pelo hardware. No sistema binário são utilizados os dígitos 1 ou 0 para a representação de quantidades, isto é, todas as informações armazenadas ou processadas no computador usam apenas DUAS grandezas. Portanto a sua base é 2 (número de dígitos do sistema).

Com esses dois dígitos representamos qualquer caractere, seja ele um número, um sinal ou letra do alfabeto. Assim com1 digito (0 ou 1) poderíamos representar:

Digito	Correspondência
0	0
1	1

Com dois dígitos poderíamos representar:

Digitos	Correspondência
00	0
01	1
10	2
11	3

Com quatro dígitos poderíamos representar:

Decimal	Binário
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

E assim por diante. Com 8 dígitos podemos representar 2^8 que permite representar até 256 caracteres.

Operando com binários:

Adição decimal:

$$\begin{array}{r} 2 \\ +\,3 \\ \hline 5 \end{array} \qquad \begin{array}{r} 4 \\ +\,5 \\ \hline 9 \end{array} \qquad \begin{array}{r} 8 \\ +\,9 \\ \hline 17 \end{array} \qquad \begin{array}{r} 8 \\ +\,10 \\ \hline 18 \end{array}$$

Adição binária:

$$\begin{array}{r} 0 \\ +\,0 \\ \hline 0 \end{array} \qquad \begin{array}{r} 1 \\ +\,0 \\ \hline 1 \end{array} \qquad \begin{array}{r} 0 \\ +\,1 \\ \hline 1 \end{array} \qquad \begin{array}{r} 1 \\ +\,1 \\ \hline 10 \end{array}$$

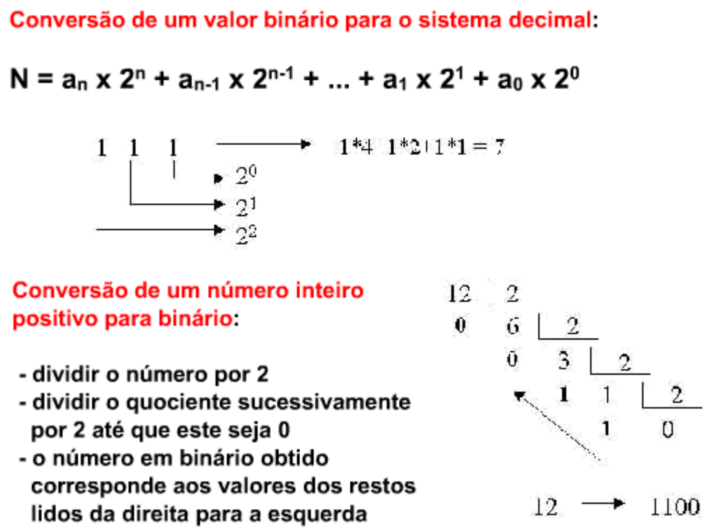
$$\begin{array}{r} 111 \\ 111 \\ +\,101 \\ \hline 1100 \end{array} \qquad \begin{array}{r} 1111 \\ 11010011 \\ +\,11100101 \\ \hline 110111000 \end{array}$$

Conversão Decimal-Binário

$(134)_{10} = (1000110)_2$

Divide-se o número decimal por 2 e tomam-se os restos:

$134 \div 2 = 67$	Resto = 0
$67 \div 2 = 33$	Resto = 1
$33 \div 2 = 16$	Resto = 1
$16 \div 2 = 8$	Resto = 0
$8 \div 2 = 4$	Resto = 0
$4 \div 2 = 2$	Resto = 0
$2 \div 2 = 1$	Resto = 0
$1 \div 2 = 0$	Resto = 1



Conversão Binário – Decimal

$10110)_2 = (22)_{10}$

$0 \rightarrow 0 \quad \rightarrow 0 \times 2^0 = 0 \times 1 \;= 0$

$1 \rightarrow 1 \quad \rightarrow 1 \times 2^1 = 1 \times 2 \;= 2$

$1 \rightarrow 2 \quad \rightarrow 1 \times 2^2 = 1 \times 4 \;= 4$

$0 \rightarrow 3 \quad \rightarrow 0 \times 2^3 = 0 \times 8 \;= 0$

$1 \rightarrow 4 \quad \rightarrow 1 \times 2^4 = 0 \times 16 = 16$

22

Sistema Hexadecimal

Simplificar a notação binária. Possui 16 símbolos:

0 1 2 3 4 5 6 7 8 9 A B C D E F

Sistema Hexadecimal – Montagem

00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F

10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 0F

20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 0F

30 31 32 33 34 35 36 37 38 39 3A 3B 3C 3D 3E 0F

e assim por diante...

Conversão Decimal – Hexa

$\mathbf{(45)_{10} = (2D)_{16}}$

Basta dividir por 16 e tomar os restos:

$45 \div 16 = 2$ Resto = 13

$2 \div 16 = 0$ Resto = 2

Resto = 13 → Resto = D

Resto = 2 → Resto = 2

Conversão Hexa – Binário

Tabela de conversão

Hexa	Binário	Hexa	Binário	Hexa	Binário
0	0000	6	0110	C	1100
1	0001	7	0111	D	1101
2	0010	8	1000	E	1110
3	0011	9	1001	F	1111
4	0100	A	1010		
5	0101	B	1011		

Conversão Hexa-Binário

$\mathbf{(E0A2)_{16}=(1110000010100010)_2}$

Utilizar a tabela de conversão para cada algarismo hexadecimal:

E = 0111 **0 = 0000**

A = 1010 **2 = 0010**

E	**0**	**A**	**2**
0111	**0000**	**1010**	**0010**

Conversão Binário-Hexa

$(11101100011)_2 = (7E3)_{16}$

Separar o número binário em grupos de 4 bits da direita para a esquerda:

111 1110 0011

Completar 4 bits no 1º grupo:

0111 1110 0011

Consultar a tabela:

0111=7 1110=E 0011=3

Regras para conversão de bases

REGRA DE CONVERSÃO PARA DECIMAL

Método da soma dos produtos de cada dígito pela base, elevada ao expoente dado pela posição do dígito.

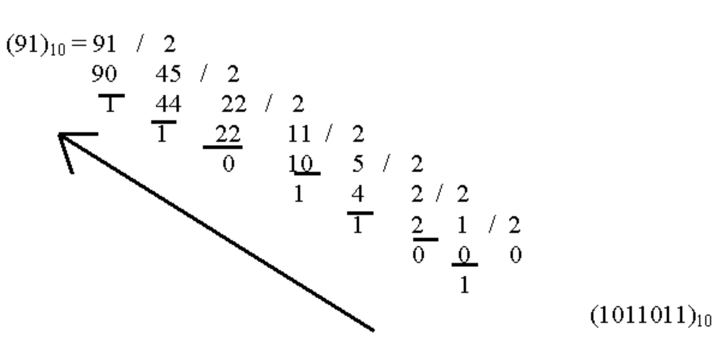
REGRA DE CONVERSÃO DE DECIMAL

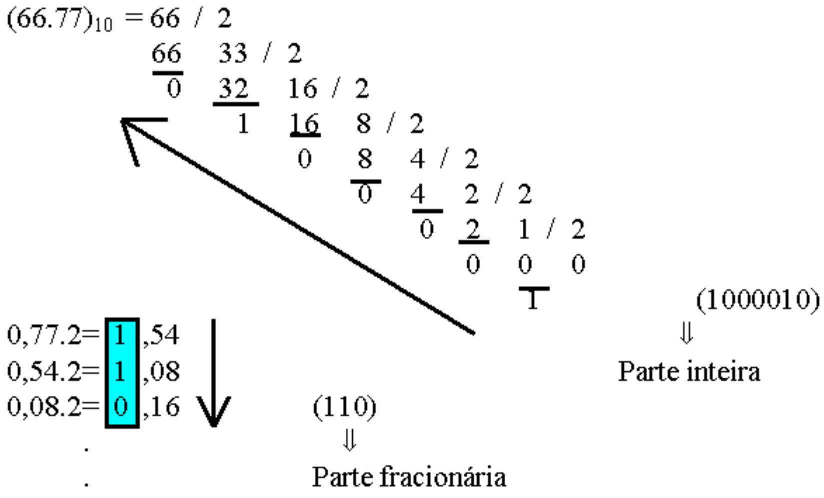
* <u>Parte inteira:</u> Dividimos o número a ser convertido pela base desejada. Tomamos o quociente resultante, dividindo novamente pela base desejada e repetindo a operação até que o quociente seja 0. Os restos da divisão formam a parte inteira do número convertido. O primeiro resto representa o último dígito da parte inteira do número.

* <u>Parte Fracionária:</u> Multiplicamos a parte fracionária do número a ser convertido pela base desejada. Tomamos a parte fracionária do número resultante e repetimos a operação. A parte inteira dos produtos obtidos representam a parte fracionária do número procurado.

A) Conversão de DECIMAL para BINÁRIO

Parte inteira: Pela regra (Divide o número pela base 2). Ex.:





$(66.77)_{10} = (1000010.110)_2$

B) Conversão de BINÁRIO para DECIMAL

Soma dos produtos de cada dígito pela base, elevada ao expoente correspondente a posição do dígito.

C) Conversão de HEXADECIMAL para DECIMAL

Soma dos produtos de cada dígito pela base 16, elevada ao expoente correspondente a posição do dígito.

D) Conversão de HEXADECIMAL para BINÁRIO

- Converter cada dígito isoladamente (Pela tabela dada anteriormente);
- Concatenar o resultado. Ex.:

$(A3)_{16} =$ A=1010
3=0011
$= (10100011)_2$

$(3A.BF)_{16} = (00111010.10111111)$

E) Conversão de BINÁRIO para HEXADECIMAL

- Converter cada dígito isoladamente (Pela tabela dada anteriormente);
- Concatenar o resultado.

OBS: Parte Inteira: Inicia da direita para esquerda;
Parte Fracionária: Inicia da esquerda para direita.

Ex.:

$(00011101\ .\ |\ 10001100)_2 = (1D.7C)_{16}$

1 D | 7 C