



UEM – Universidade Estadual de Maringá
DIN - Departamento de Informática
Disciplina: Fundamentos da Computação
Profª Thelma Elita Colanzi Lopes
thelma@din.uem.br

Aula 3 - Sistemas de Numeração

O ser humano, desde a antiguidade, necessitou dos números para expressar coisas quantitativas, o que não podia ser caracterizado qualitativamente necessitava ser comunicado de forma diferente. Desta forma, surgiram os números. Supõe-se que os decimais surgiram primeiramente pela facilidade gerada pela anatomia humana que possui dez dedos, cinco em cada mão, para realizar contagens. Porém, habituamo-nos tanto ao sistema decimal que o entendimento de outros sistemas de numeração é complexo. Mas inevitável a partir do momento em que o principal instrumento da modernidade, o computador, utilizou outro sistema, o binário. Assim, outros sistemas surgem, conforme a necessidade, que muitas vezes o sistema decimal não supre. Neste material, abordaremos os principais sistemas utilizados atualmente: o decimal, o binário, o octal e o hexadecimal, assim como a conversão entre alguns desses sistemas.

Um **numeral** é um símbolo ou grupo de símbolos que representa um número. Os numerais diferem dos números do mesmo modo que as palavras diferem das coisas a que se referem. Os símbolos "11", "onze" e "XI" são numerais diferentes, representando todos o mesmo número.

Um **sistema de numeração**, (ou **sistema numeral**) é um sistema em que um conjunto de números são representados por numerais de uma forma consistente. Pode ser visto como o *contexto* que permite ao numeral "11" ser interpretado como o numeral romano para *dois*, o numeral binário para *três* ou o numeral decimal para *onze*.

Em condições ideais, um sistema de numeração deve:

- Representar uma grande quantidade de números úteis (ex.: todos os números inteiros, ou todos os números reais);
- Dar a cada número representado uma única descrição (ou pelo menos uma representação padrão);
- Refletir as estruturas algébricas e aritméticas dos números.

Por exemplo, a representação comum decimal dos números inteiros fornece a cada número inteiro uma representação única como uma sequência finita de algarismos, com as operações aritméticas (adição, subtração, multiplicação e divisão) estando presentes como os algoritmos padrões da aritmética. Contudo, quando a representação decimal é usada para os números racionais ou para os números reais, a representação deixa de ser padronizada: muitos números racionais têm dois tipos de numerais, um padrão que tem fim (por exemplo 2,31), e outro que repete-se periodicamente (como 2,30999999...).

1. Sistema decimal

O **sistema decimal** é um sistema de numeração de posição que utiliza a base dez.

Baseia-se em uma numeração de posição, onde os dez algarismos indo-arábicos : 0 1 2 3 4 5 6 7 8 9 servem a contar unidades, dezenas, centenas, etc. da direita para a esquerda. Contrariamente à numeração romana, o algarismo árabe tem um valor diferente segundo sua posição no número: assim, em **111**, o primeiro algarismo significa 100, o segundo algarismo 10 e o terceiro 1, enquanto que em **VIII** (oito em numeração romana) os três **I** significam todos 1. Assim:

$$347 = 3 \cdot 100 + 4 \cdot 10 + 7 \cdot 1 = 3 \cdot 10^2 + 4 \cdot 10^1 + 7 \cdot 10^0$$

No sistema decimal o símbolo 0 (zero) posicionado à esquerda do número escrito não altera seu valor representativo. Assim: 1; 01; 001 ou 0001 representam a mesma grandeza, neste caso a unidade. O símbolo zero posto à direita implica em multiplicar a grandeza pela base, ou seja, por 10 (dez).

2. Sistema Binário

O **sistema binário** é um sistema de numeração posicional em que todas as quantidades se representam utilizando como base o número dois, dispondo-se das cifras: **zero e um** (0 e 1).

Os computadores digitais trabalham internamente com dois níveis de tensão, seu sistema de numeração natural é o sistema binário (aceso, apagado). Com efeito, num sistema simples como este é possível simplificar o cálculo, com o auxílio da lógica booleana. Em computação, chama-se um dígito binário (0 ou 1) de *bit*, que vem do inglês *Binary Digit*. Um agrupamento de 8 bits corresponde a um byte (Binary Term).

O sistema binário é base para a Álgebra booleana (de George Boole - matemático inglês), que permite fazer operações lógicas e aritméticas usando-se apenas dois dígitos ou dois estados (sim e não, falso e verdadeiro, tudo ou nada, 1 ou 0, ligado e desligado). Toda eletrônica digital e computação está baseada nesse sistema binário e na lógica de Boole, que permite representar por circuitos eletrônicos digitais (portas lógicas) os números, caracteres, realizar operações lógicas e aritméticas. Os programas de computadores são codificados sob forma binária e armazenados nas mídias (memórias, discos, etc) sob esse formato.

2.1 Operações de conversão com binários

2.1.1 Binários em decimais

Dado um número N, binário, para expressá-lo em decimal, deve-se escrever cada número que o compõe (bit), multiplicado pela base do sistema (base = 2), elevado à posição que ocupa. Uma posição à esquerda da vírgula representa uma potência positiva e à direita uma potência negativa. A soma de cada multiplicação de cada dígito binário pelo valor das potências resulta no número real representado.

Exemplo 1: 1011(binário)

$$1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 11$$

Portanto, 1011 é 11 em decimal

Exemplo 2: 1100 (binário)

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

$$1 \times 2^2 \Rightarrow 1 \times 4 = 4$$

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

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

Realizada esta operação, soma-se os valores encontrados: 8 + 4 + 0 + 0 e temos o resultado: 12, logo, o número 1100 em binário corresponde ao número 12 em decimal.

2.1.2 Decimais em binários

- **Decimais inteiros em binários**

Dado um número decimal inteiro, para convertê-lo em binário, basta dividi-lo sucessivamente por 2, anotando o resto da divisão inteira:

12(dec) -> bin

$$12 / 2 = 6 + 0$$

$$06 / 2 = 3 + 0$$

$$03 / 2 = 1 + 1$$

$$01 / 2 = 0 + 1$$

$$12(\text{dec}) = 1100(\text{bin})$$

Observe que os números devem ser lidos **de baixo para cima**: **1100** é 12 em binário.

Existe um método muito simples para converter binário em decimal, e vice-versa.

128 64 32 16 8 4 2 1	
0 0 0 0 1 0 1 0	= 10 (2+8=10)
0 0 0 1 1 0 0 0	= 24 (8+16=24)
1 1 0 0 0 0 0 0	= 192 (64+128=192)
1 0 1 1 1 0 1 0	= 186 (2+8+16+32+128=186)

- **Decimais fracionários em binários**

Exemplo **1:** **0.5625₁₀**
Parte inteira = 0₁₀ = 0₂ Parte fracionária = 0.5625₁₀
Multiplica-se a parte fracionária por 2 sucessivamente, até que ela seja igual a zero ou cheguemos na precisão desejada.

fração x 2 = vai-um + fração seguinte

$$0.5625 \times 2 = 1 + 0.1250$$

$$0.1250 \times 2 = 0 + 0.2500$$

$$0.2500 \times 2 = 0 + 0.5000$$

$$0.5000 \times 2 = 1 + 0.0000 \text{ <--- nesta linha a fração zerou, finalizamos a conversão}$$

Anotando a sequência de “vai-um” na ordem de cima para baixo, temos: **1001**

Portanto, 0.5625₁₀ = 0.1001₂

No entanto, é mais comum nunca zerarmos a fração seguinte da multiplicação. Neste caso, devemos parar as multiplicações quando atingirmos a precisão desejada.

Exemplo **2:** **67.575₁₀**
Parte inteira = 67₁₀ = 1000011₂ Parte fracionária = 0.575₂

fração x 2 = vai-um + fração seguinte

$$0.5750 \times 2 = 1 + 0.1500$$

$$0.1500 \times 2 = 0 + 0.3000$$

$$0.3000 \times 2 = 0 + 0.6000 \text{ <- esta fração e suas subseqüentes serão repetidas em breve.}$$

$$0.6000 \times 2 = 1 + 0.2000$$

$$0.2000 \times 2 = 0 + 0.4000$$

$$0.4000 \times 2 = 0 + 0.8000$$

$$0.8000 \times 2 = 1 + 0.6000 \text{ <- inicia aqui a repetição da fração 0.6000 e suas subseqüentes}$$

$$0.6000 \times 2 = 1 + 0.2000$$

Ou seja, entramos em um ciclo sem fim. Escolhemos uma precisão e finalizamos o processo quando esta precisão for atingida, então na ordem de cima para baixo, temos: **10010011₂**

2.2 Operações Aritméticas com Binários

2.2.1 Soma de Binários

Para somar dois números binários, o procedimento é o seguinte:

$$\begin{array}{r} 1100 \\ + 111 \\ \hline =10011 \end{array}$$

Explicando: Os números binários são base 2, ou seja, há apenas dois algarismos: 0 (zero) ou 1 (um). Na soma de 0 com 1 o total é 1. Quando se soma 1 com 1, o resultado é 2, mas como 2 em binário é 10, o resultado é 0 (zero) e passa-se o outro 1 para a "frente", ou seja, para ser somado com o próximo elemento, conforme destacado em negrito.

2.2.2 Subtração de Binários

Para subtrair dois números binários, o procedimento é o seguinte:

$$\begin{array}{r} 1101110 \\ - 10111 \\ \hline = 1010111 \end{array}$$

Explicando: Quando temos 0 menos 1, precisamos "emprestar" do elemento vizinho. Esse empréstimo vem valendo 2 (dois), pelo fato de ser um número binário. Então, no caso da coluna 0 - 1 = 1, porque na verdade a operação feita foi 2 - 1 = 1. Esse processo se repete e o elemento que cedeu o "empréstimo" e valia 1 passa a valer 0. Os elementos que "emprestaram" para seus vizinhos estão em negrito. Perceba, que, logicamente, quando o valor for zero, ele não pode "emprestar" para ninguém, passando-se o "pedido" para o próximo elemento.

2.2.3 Multiplicação de Binários

A multiplicação entre binários é similar a realizada normalmente. A única diferença está no momento de somar os termos resultantes da operação:

$$\begin{array}{r} 1011 \\ \times 1010 \\ \hline 0000 \\ + 1011 \\ + 0000 \\ + 1011 \\ \hline = 11\mathbf{0}1110 \end{array}$$

Perceba que na soma de 0 e 1 o resultado será 1, mas na soma de 1 com 1, ao invés do resultado ser 2, ele será 0 (zero) e passa-se o 1 para a próxima coluna, veja destaque em negrito.

2.2.4 Divisão de Binários

Essa operação também é similar a realizada entre números decimais:

$$\begin{array}{r} 110 \overline{) 10} \\ - 10 \quad 11 \\ \hline 010 \\ - 10 \\ \hline 00 \end{array}$$

Deve-se observar somente a regra para subtração entre binários. Nesse exemplo a divisão de 110 por 10 teve como resultado 11.

3. Sistema octal

Sistema Octal é um sistema de numeração cuja base é 8, ou seja, utiliza 8 símbolos para a representação de quantidade. No ocidente, estes símbolos são os algarismos arábicos: 0 1 2 3 4 5 6 7

O octal foi muito utilizado em informática como uma alternativa mais compacta ao binário na programação em linguagem de máquina. Hoje, o sistema hexadecimal é mais utilizado como alternativa ao binário.

Este sistema também é um sistema posicional e a posição de seus algarismos determinada em relação à vírgula decimal. Caso isso não ocorra, supõe-se implicitamente colocada à direita do número. A aritmética desse sistema é semelhante a dos sistemas decimal e binário, o motivo pelo qual não será apresentada.

Exemplo:

Qual o número decimal representado pelo número octal 4701?

Resposta: $4 \times 8^3 + 7 \times 8^2 + 0 \times 8^1 + 1 \times 8^0 = 2048 + 448 + 0 + 1 = 2497$

4. Sistema hexadecimal

O **sistema hexadecimal** é um sistema de numeração vinculado à informática, já que os computadores interpretam as linguagens de programação em *bytes*, que são compostos de oito dígitos. À medida que os computadores e os programas aumentam a sua capacidade de processamento, funcionam com múltiplos de oito, como 16 ou 32. Por este motivo, o sistema hexadecimal, de 16 dígitos, é um *standard* na informática.

Como o nosso sistema de numeração só dispõe de dez dígitos, devemos incluir seis letras para completar o sistema. Estas letras e o seu valor em decimal são: A = 10, B = 11, C = 12, D = 13, E = 14 e F = 15.

O sistema hexadecimal é posicional e por ele o valor numérico associado a cada signo depende da sua posição no número, e é proporcional as diferentes potências da base do sistema que neste caso é 16.

Vejamos um exemplo numérico: $3E0,A (16) = 3 \times 16^2 + E \times 16^1 + 0 \times 16^0 + A \times 16^{-1} = 3 \times 256 + 14 \times 16 + 0 \times 1 + 10 \times 0,0625 = 992,625 (10)$

A utilização do sistema hexadecimal nos computadores, deve-se a que um dígito hexadecimal representa quatro dígitos binários; portanto, dois dígitos hexadecimais representam oito dígitos binários (8 bits = 1 byte), que, como é sabido, é a unidade básica de armazenamento de informação.

4.1 Tabela de conversão entre decimal, binário e hexadecimal

Decimal	Binário	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7

Decimal	Binário	Hexadecimal
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Exercício - Faça as seguintes conversões:

1. 0, 12510 para binário
2. 0, 10112 para decimal
3. 10001 para decimal
4. 10001 para binário
5. 124 para binário

Bibliografia:

www.pt.wikipedia.org