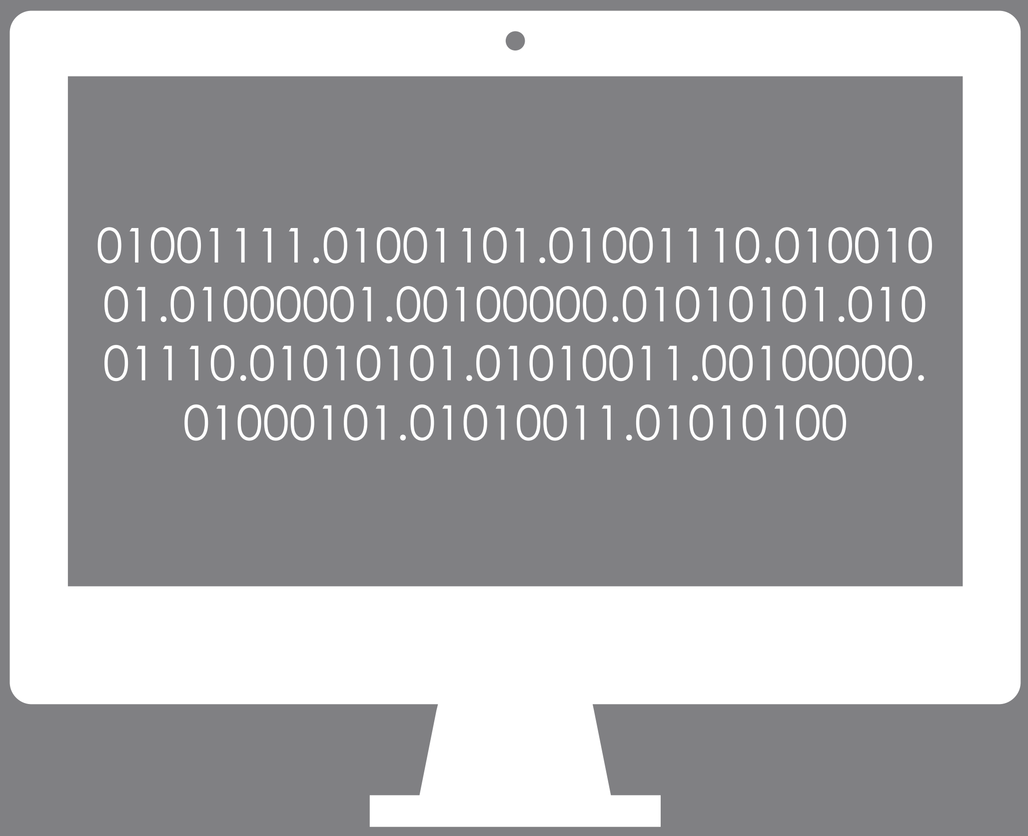
Hephaestus Academy



Arquitetura e Organização de Computadores

Sistemas Numéricos

Sistemas de numeração – *Conteúdo e Questões*

- Quantidades em computação
- Sistemas numéricos decimal, binário, octal, decimal, hexadecimal
- Valor absoluto e relativo
- Equação ponderada de número binário
- Conversões de base
- Aplicação

Sistema de Numeração

Entender os fundamentos dos sistemas de numeração é importante porque os sistemas de computação têm como base a manipulação de informações numéricas. Assim, compreenderemos como as informações são codificadas em sistemas de computação e de que modo esses dados trafegam internamente no computador e de que forma viajam através das redes de computadores.

Desde o início, o homem sentiu a necessidade de lidar com objetos contáveis e, consequentemente, precisava utilizar algum método de contagem, isto é, de um sistema de numeração.

Os egípcios possuíam um sistema pictográfico, onde eram usados símbolos para representar valores absolutos, enquanto os mesopotâmios inventaram o sistema sexagesimal (base 60). Os dois sistemas eram extremamente complexos e nunca permitiram a estes dois povos se destacarem como grandes matemáticos. Na geometria ambos tiveram grandes avanços, mas na álgebra nunca puderam ir além das operações matemáticas básicas. Outros sistemas numéricos surgiram, porém todos sempre esbarravam em dificuldades de utilização, principalmente na álgebra.

Dois conceitos são fundamentais para melhor entendermos os sistemas de numeração:
Número e Numeral.

O Número refere-se à quantidade representada, enquanto o Numeral é a representação gráfica dessa quantidade. Diferentes numerais podem representar o mesmo número (quantidade) dependendo da base do sistema de numeração.

Na computação, os sistemas de numeração mais utilizados são:

- **Decimal** ***(Base 10)***: dígitos 0, 1, 2, 3, 4, 5, 6, 7, 8 e 9 - comunicação humana.
- **Binário** ***(Base 2 → 21)***: dígitos 0 e 1 - usado em computadores digitais.
- **Octal** ***(Base 8 → 23)***: dígitos 0, 1, 2, 3, 4, 5, 6 e 7.
- **Hexadecimal** ***(Base 16 → 24)***: dígitos 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Foram tomadas emprestadas cinco letras A, B, C, D, E, F para completar os 16 símbolos. Usado para representar números grandes (Ex. endereços de memória), que se mostrados em binário, como realmente é internamente no computador, seria um número muito grande.

Tabela 1.2 - Representação de inteiros em binário, octal, decimal e hexadecimal.

Binário	Octal	Decimal	Hexa
0	0	0	0
1	1	1	1
10	2	2	2
11	3	3	3
100	4	4	4
101	5	5	5
110	6	6	6
111	7	7	7
1000	10	8	8
1001	11	9	9
1010	12	10	A
1011	13	11	B
1100	14	12	C
1101	15	13	D
1110	16	14	E
1111	17	15	F
10000	20	16	10
10001	21	17	11

Quantidades em Computação

KB - Kilobyte (~mil bytes) 210 = 1.024 bytes ou 103 Computador 1ª geração -

memória 2 KB, 3ª geração 128 KB Disquete de 5¼” (diâmetro) 360 KB.

MB - Megabyte (~milhão de bytes) 220 = 1.048.576 bytes ou 106

Disquete 3,5" - 1,44 MB, CD-ROM - 700 MB. GB - Gigabyte (~bilhão de bytes) 230 = 1.073.741.824 bytes ou 109 HD 500 GB, DVD 4.7 GB, Bluray 25GB/50GB.

TB - Terabyte (trilhão de bytes) 240 bytes ou 1012 Robô de DLT com 6

fitas de 200 GB total de 1.2 TB

PB - Petabyte (quatrilhão de bytes) 250 bytes ou 1015

- Dados armazenados em uma fitoteca (Ex. CPTEC INPE)

EB Exabyte (~Quintilhão) 260 bytes ou 1018 **ZB Zettabyte**

(~Sextilhão) 270 bytes ou 1021 **YB Yottabyte** (~Septilhão) 280 bytes

ou 1024

1YB = 1.208.925.819.614.629.174.706.176 Bytes

1YB é tanta informação que não caberia dentro de todos os HDs existentes no mundo hoje!

Curiosidade: O diretor da Escola de Astronomia e Astrofísica da Austrália, Simon Driver, diz que existem pelo menos 70 septiliões (70.000.000.000.000.000.000.000 = 7 x 1022) de estrelas no Universo, cerca de dez vezes o número estimado de grãos de areia na Terra. 1 YB é 14,3 vezes maior que esse número.

Sistema Numérico Decimal

Vários povos desenvolveram empiricamente algum tipo de sistema de numeração. Aquele que revelou ser o mais prático foi o sistema de numeração decimal, também chamado de base 10. Tal sistema surgiu pelo fato de o homem possuir 10 dedos em suas mãos, sendo mais fácil associar um objeto a cada dedo, por isso os símbolos usados no sistema decimal, e por extensão nos demais sistemas, são chamados de dígitos (de dedos).

Valor Absoluto e Valor Relativo

O sistema decimal possui dez símbolos ou dígitos (0 / 1 / 2 / 3 / 4 / 5 / 6 / 7 / 8 / 9) que são associados a quantidades absolutas. Além deste valor absoluto, cada dígito possui também um valor relativo (também chamado de valor posicional). Em outras palavras, para representar quantidades de 0 a 9 usamos os símbolos disponíveis de 0 a 9, onde os valores absolutos dos mesmos são idênticos a seus valores relativos e representam unidades.

Para valores acima de nove unidades, acrescentamos outro dígito à esquerda, cujo valor absoluto não corresponde ao seu valor relativo, isto é, o dígito da esquerda representa quantidades (grupos) de dez unidades, por isto tal dígito é chamado de casa das dezenas. Por exemplo:

- **10** - O dígito 1 da esquerda representa um grupo de 10 unidades e o 0 da direita representa um grupo de zero unidade, assim 10 + 0 = 10.
- **11** - O dígito 1 da esquerda representa um grupo de 10 unidades e o 1 da direita representa
- um grupo de uma unidade, assim 10 + 1 = 11.
- **12** - O dígito 1 da esquerda representa um grupo de 10 unidades e o 2 da direita representa um grupo de duas unidades, assim 10 + 2 = 12.
- **19** - O dígito 1 da esquerda representa um grupo de 10 unidades e o 9 da direita representa um grupo de nove unidades, assim 10 + 9 = 19.
- **23** - O dígito 2 da esquerda representa dois grupos de 10 unidades e o 3 da direita representa um grupo de três unidades, assim 20 + 3 = 2 x 10 + 3 = 23.
- **35** - O dígito 3 da esquerda representa três grupos de 10 unidades e o 5 da direita representa um grupo de cinco unidades, assim 30 + 5 = 3 x 10 + 5 = 35.
- **99** - O dígito 9 da esquerda representa nove grupos de 10 unidades e o 9 da direita representa um grupo de nove unidades, assim 90 + 9 = 9 x 10 + 9 = 99.
- Para valores acima de 99 unidades, acrescentamos outro dígito à esquerda, cujo valor representa quantidades (grupos) de cem unidades, chamado de casa das centenas:
- **100** - O dígito 1 da esquerda representa um grupo de 100 unidades; o 0 do meio representa um grupo de zero dezena e o 0 da direita representa um grupo de zero unidade, assim 100 + 0 + 0 = 100.
- **217** - O dígito 2 da esquerda representa dois grupos de 100 unidades; o 1 do meio representa um grupo de uma dezena e o 7 da direita representa um grupo de sete unidades, assim 200 + 10 + 7 = 2 x 100 + 1 x 10 + 7 = 217.
- **999** - O dígito 9 da esquerda representa nove grupos de 100 unidades; o 9 do meio representa nove grupos de dezenas e o 9 da direita representa um grupo nove unidades, assim 900 + 90 + 9 = 9 x 100 + 9 x 10 + 9 = 999.

Para valores acima de 999 unidades, acrescentamos outro dígito à esquerda, cujo valor representa quantidades (grupos) de mil unidades, chamado de casa do milhar. A cada dígito que vamos acrescentando à esquerda, vamos aumentando o valor relativo do mesmo, sendo que este valor relativo será sempre uma potência de 10, por isso o valor relativo é também chamado de valor ponderado.

Equação Ponderada de um Número

Como mostrado, podemos representar qualquer número do sistema numérico decimal da seguinte maneira:

N = an x bn-1 + an-1 x bn-2 + an-2 x bn-3 + . . . + a3 x b2 + a2 x b1 + a1 x b0 Onde:

- N → Número representado.
- a → Algarismo ou dígito.
- b → Base do sistema numérico, no caso do sistema decimal a base é 10.
- n subscrito → Número de dígitos do número representado.
- Expoente (n-1) sobrescrito → Representa a potência da base.

Sistema Numérico Binário

Além do sistema numérico decimal vários outros sistemas numéricos eram conhecidos, inclusive o binário, porém, a utilização prática dos mesmos só se tornou possível com a invenção do computador digital a partir da década de 50.

Todas as informações armazenadas ou trafegando em computadores digitais estão codificadas de forma binária, só podem assumir os valores 0 ou 1.
É fundamental entender como esses dados são codificados, manipulados, armazenados e como trafegam internamente no computador, e externamente nas redes de computadores.

O sistema binário é utilizado nos computadores eletrônicos digitais, por representar adequadamente os possíveis estados de componentes eletrônicos de maneira simples, como:

- Ligado ou Desligado;
- Aceso ou Apagado;
- +5V ou 0V etc.

Ou valores booleanos:

- Sim ou Não;
- Verdadeiro ou Falso etc.

Podemos representar binários por qualquer dispositivo que tenha apenas dois estados ou condições possíveis. Exemplo: uma chave que aberta (binário 0) ou fechada (binário 1).

Apenas Dois Símbolos para Todos os Números

O sistema decimal é muito prático para uso diário em todos os setores das atividades humanas, porém na computação digital ele é impraticável, pois nestes sistemas trabalha-se apenas com dois níveis de tensão.

Exemplo: ou temos tensão (+ 5 VDC), ou não temos tensão (0 VDC). VDC - Volt Direct Current. Podemos associar estes dois níveis de tensão a dígitos, atribuindo ao 0 VDC o dígito 0, e ao + 5 VDC o dígito 1.

Neste caso passamos a trabalhar com apenas dois valores, que formam o sistema binário, ou seja, o sistema de numeração binário é constituído por apenas dois símbolos numéricos, chamados de dígitos binários, mais conhecidos por bit (palavra formada pela contração das palavras inglesas BInary digiT). Portanto, o bit pode ser definido como a unidade básica de informação do sistema numérico binário. O “0” e o “1” são chamados de valores, níveis ou estados lógicos, para diferenciá-los de níveis de tensão.

Não devemos confundir nível lógico 0 ou 1, com valores de tensão 0 VDC e 5 VDC. Já vimos que existe uma relação entre nível lógico e valor de tensão, porém esta relação depende da família lógica utilizada (TTL, ECL, MOS, ou outra qualquer) e depende ainda da lógica utilizada (lógica positiva ou lógica negativa). Por exemplo: se utilizarmos circuitos integrados da Família TTL, em Lógica Positiva temos para o nível 0 um valor de tensão próximo de 0 VDC (terra) e para o nível 1 temos um valor de tensão próximo de +5 VDC. Em Lógica Negativa o nível 0 continua próximo de 0 VDC (terra), porém o nível 1 é próximo de -5 VDC. É importante observar isso porque dependendo da lógica utilizada uma porta lógica pode mudar completamente sua característica, ou seja, uma porta NAND em lógica positiva, funciona como porta NOR em lógica negativa, e vice-versa.

No sistema binário, podemos representar qualquer quantidade utilizando apenas os dois dígitos (0 e 1), havendo, obviamente, uma relação entre os valores decimais e valores binários.

Equação Ponderada de Número Binário

Como é possível descobrir o equivalente decimal de um número binário? Em outras palavras, como podemos converter um número de base 2 para seu correspondente em base 10?

Quando definimos o sistema de numeração decimal dissemos que podemos representar qualquer número do sistema numérico decimal usando a equação ponderada do número:

N = an x bn-1 + an-1 x bn-2 + an-2 x bn-3 + . . . + a3 x b2 + a2 x b1 + a1 x b0

Nesta equação, se considerarmos a base como sendo 2 ao invés de 10, temos a representação de qualquer número dentro do sistema binário.

Por exemplo, qual o correspondente em base 10 do número 110 na base 2? 110(2) = X(10)?

Se aplicarmos a equação ponderada, teremos: N = a3 x b2 + a2 x b1 + a1 x b0

Fazendo-se as substituições: 110(2) = 1 x 2^2 + 1 x 2^1 + 0 x 2^0

110(2) = 1 x 4 + 1 x 2 + 0 x 1

110(2) = 4 + 2 + 0

110(2) = 6(10)

Qual o correspondente na base 10 do número 111110100(2)?

111110100(2) = a9 x b8 + a8 x b7 + a7 x b6 + a6 x b5 + a5 x b4 + a4 x b3 + a3 x b2 + a2 x b1 + a1 x b0

111110100(2) = 1 x 2^8 + 1 x 2^7 + 1 x 2^6 + 1 x 2^5 + 1 x 2^4 + 0 x 2^3 + 1 x 2^2 + 0 x 2^1 + 0 x 2^0

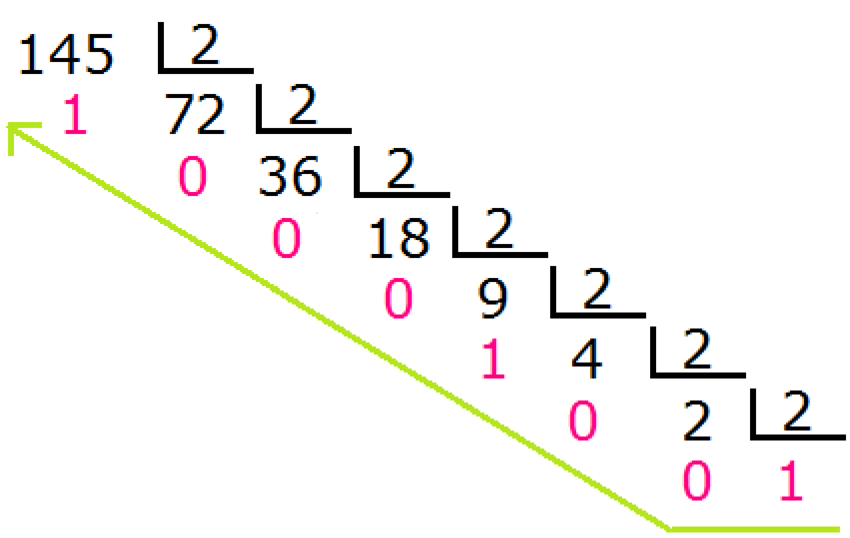
111110100(2) = 1 x 256 + 1 x 128 + 1 x 64 + 1 x 32 + 1 x 16 + 0 x 8 + 1 x 4 + 0 x 2 + 0 x 1

111110100(2) = 256 + 128 + 64 + 32 + 16 + 0 + 4 + 0 + 0

111110100(2) = 500(10)

Conversão de Base Decimal para Base Binária

Como converter um número decimal em binário?



Como o sistema binário é formado por potências de 2 basta dividir o número decimal por 2, em seguida seu quociente por 2, depois o novo quociente por 2 e assim por diante até que o quociente seja zero. Daí, toma-se os restos na ordem inversa, desde o último resto até o primeiro.

Sistemas Octal e Hexadecimal

Nos circuitos de comunicação de dados são comuns códigos em barramentos de 8 e 16 bits.

Para barramentos com 8 bits temos o sistema numérico octal, que é constituído somente pelos algarismos 0 - 1 - 2 - 3 - 4 - 5 - 6 e 7.

Para barramentos com 16 bits temos o sistema numérico hexadecimal, o qual é constituído pelos algarismos 0 - 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - A - B - C - D – E - F, sendo que os algarismo de A a F correspondem aos números de 10 a 15 respectivamente

Conversão de Bases Octal e Hexadecimal para Base Decimal

A conversão de um número octal ou hexadecimal para decimal também pode ser feita pela equação ponderada:

N = an x bn-1 + an-1 x bn-2 + an-2 x bn-3 + . . . + a3 x b2 + a2 x b1 + a1 x b0

Convertendo os números 276(8) para base 10.

N = a3 x b2 + a2 x b1 + a1 x b0

276(8) = 2 x 8^2 + 7 x 8^1 + 6 x 8^0

276(8) = 2 x 64 + 7 x 8 + 6 x 1
276(8) = 128 + 56 + 6
276(8) = 190(10)

Convertendo os números 276(16) para base 10.

N = a3 x b2 + a2 x b1 + a1 x b0
276(16) = 2 x 16^2 + 7 x 16^1 + 6 x 16^0
276(16) = 2 x 256 + 7 x 16 + 6 x 1
276(16) = 512 + 112 + 6
276(16) = 630(10)

Conversão da Base 2 para qualquer Base potência de 2
Considerando-se que 8 = 23 e 16 = 24, podemos fazer uma conversão de qualquer número decimal para binário e a partir do binário convertemos para octal ou hexadecimal de modo bem fácil.

Exemplo:
Para converter 1996(10) para base 8. Inicialmente, transforma-se da base 10 para base 2 →
1996(10) = 11111001100(2)

Em seguida, converte-se da base 2 para a base 8 (octal), simplesmente separando o número binário de três em três (8 = 2^3) algarismos a partir da direita e fazemos a conversão de cada um desses grupos de três algarismos para a base 8.
1996(10) = 11111001100(2)
1996(10) = 011 111 001 100(2)

1996(10) = 3 7 1 4(8)

Para converter para a base hexadecimal, separamos o número binário de quatro em quatro (16 = 2^4) algarismos a partir da direita e fazemos a conversão de cada um desses grupos de quatro algarismos para a base 16.
1996(10) = 11111001100(2)
1996(10) = 0111 1100 1100(2)
1996(10) = 7 C C(16)
Portanto, para converter da base 10 para qualquer outra base potência de 2, basta converter o número base 10 para a base 2 e dela converter para a base desejada.

Conversão Octal para Hexadecimal e Hexa para Octal
Converte-se primeiro o octal ou hexa para binário, em seguida converte-se de binário para o sistema de numeração desejado octal ou hexa, agrupando de 3 em 3 dígitos, se octal ou de 4 em 4 dígitos, se hexadecimal.
AFF(16) = (x)8
1010 1111 1111 = 101 011 111 111 = 5 3 7 7(8)

Aplicação: Determinação do Endereço de Rede
Quem trabalha com redes ou já precisou configurar um endereço IPv4, já se perguntou por que os números vão até 255?

Na realidade o endereço IPv4 é um número de 32 bits dividido em 4 blocos (octetos) de 8 bits cada, separados por um ponto. Para melhor visualização pelos humanos é apresentado em decimal. Ex: 192.168.10.1 Não IP → 320.234.10.1

Esses endereços IPv4 são compostos de duas partes. Uma parte que identifica o endereço da rede e outra que identifica o host (PC) dentro da rede.

Ao contrário do endereço IP, que é formado por valores entre 0 e 255, a máscara de sub-rede é normalmente formada por apenas dois valores: 0 ou 255. Exemplo 255.255.0.0 ou 255.0.0.0, onde o valor 255 indica a parte endereço IP referente à rede, e o valor 0 indica a parte endereço IP referente ao host.

Dado o endereço IPv4 de um host e a máscara, é possível calcular o endereço de rede. Para isso, aplica-se o operador “AND” lógico entre o endereço IPv4 e a máscara, tendo como resultado o endereço de rede.

Relembrando os operadores lógicos “AND” e “OR”

Exemplo:

Qual o endereço da rede a que pertence o host cujo IPv4 é 10.34.23.134 e cuja máscara é 255.0.0.0?
Segundo a regra, para se obter o endereço da rede de um host dado seu IP e máscara, faz-se uma operação “AND” entre o endereço de host e a máscara. Assim, a primeira ação é transformar os endereços de host e máscara que estão em decimal pontuado para binário para que seja possível a operação “AND”.

Assim:
Endereço IP do host *10.34.23.134* em binário é: 00001010.00100010.00010111.10000110
Máscara *255.0.0.0* em binário é: 11111111.00000000.00000000.00000000

Realizando a operação booleana AND entre o endereço IP e a máscara de sub-rede, produz-se o endereço de rede deste host.

Assim:
00001010.00100010.00010111.10000110 AND
11111111.00000000.00000000.00000000
00001010.00000000.00000000.00000000

Logo o endereço de rede é: 00001010.00000000.00000000.00000000

Que convertendo para decimal pontuado temos: 10.0.0.0 que é o endereço da rede.

Questões:

1) Converta os números binários em decimal:
a- 1100(2)
b- 101010(2)
c- 110001101(2)

2) Converta os números decimais em binário:
a- 34(10)
b- 298(10)
c- 1986(10)

3) Converta os números de octal para decimal:
a- 76(8)
b- 327(8)
c- 30456(8)

4) Converta os números de hexadecimal para decimal:
a- 6AC(16)
b- ABC(16)
c- 3CDF(16)

5) Converta os números binário em octal:
a- 110001(2)
b- 111101010(2)
c- 1011010011(2)

6) Converta os números binário em hexadecimal:
a- 110001(2)
b- 10110110(2)
c- 110001011(2)

7) Converta os endereços IP de decimal pontuado para binário pontuado:
a- 196.227.42.34
b- 207.148.55.14
c- 217.189.64.32