

Sistemas de Numeração.

No início deste curso dissemos que computadores só entendem informações em números binários, hexadecimais ou octais.

Agora teremos a oportunidade de conhecer mais a fundo esses sistemas de numeração, inclusive aprenderemos como é realizada a conversão entre eles.

Números decimais

Números decimais são os que estamos acostumados a lidar na Matemática convencional. Também são conhecidos como números de **base 10**. Isso porque compreendem dez símbolos numéricos: os números **0, 1, 2, 3, 4, 5, 6, 7, 8 e 9**.

Através da combinação desses dez números é possível representar todos os números decimais.

Quando um número decimal é lido da direita para a esquerda, a primeira posição do mesmo é representada pelo número **0**. A posição do próximo número da esquerda para a direita é representada pelo número **1** e assim por diante. Por exemplo, vejamos isso com o número **26802**:

número decimal	2	6	8	0	2
coluna	4	3	2	1	0

←
posição da direita para a esquerda começando em 0

O sistema decimal é baseado em potências de 10. Levando em conta a figura acima que mostra como é realizada a identificação da coluna do número, em um sistema decimal cada número é definido pela soma de cada algarismo multiplicado por 10 elevado à potência correspondente à coluna do mesmo:

número decimal	2	6	8	0	2
coluna	4	3	2	1	0
potências	10^4	10^3	10^2	10^1	10^0
base 10	2×10^4	6×10^3	8×10^2	0×10^1	2×10^0
resultado	20000	6000	800	0	2
somatória	$20000 + 6000 + 800 + 0 + 2 = 26802$				

Agora que vimos uma tabela detalhando como funcionam os números decimais, vejamos mais alguns exemplos:

$$369 = (3 \times 10^2) + (6 \times 10^1) + (9 \times 10^0)$$

$$2768 = (2 \times 10^3) + (7 \times 10^2) + (6 \times 10^1) + (8 \times 10^0)$$

E assim por diante...

Números binários.

Números binários são os mais importantes em computação. Quando falamos que computadores atuais são digitais, significa que processam os dados no formato binário.

Números binários também são conhecidos como números de **base 2**. Compreendem somente dois caracteres: o **0** e o **1**.

Quando um número binário é lido da direita para a esquerda, assim como com os números decimais, sua posição na coluna inicial é considerada **0**. O próximo dígito mais à esquerda é considerado de posição **1** e assim por diante, como na figura:

número binário	0	1	0	0	1
coluna	4	3	2	1	0

←
posição da direita para a esquerda começando em 0

Os números binários são baseados em potências de 2 e, de forma semelhante aos números decimais, podem ser definidos pela soma de cada algarismo multiplicado por 2 (que é a sua base) elevado à potência que corresponde à coluna correspondente do mesmo:

número binário	0	1	0	0	1
coluna	4	3	2	1	0
potências	2^4	2^3	2^2	2^1	2^0
base 2	0×2^4	1×2^3	0×2^2	0×2^1	1×2^0
resultado	0	8	0	0	1
somatória (decimal)	$0 + 8 + 0 + 0 + 1 = 9$				

Para ficar bem claro, seguem mais alguns exemplos:

$$1101 = (1 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0) = 13$$

$$1110 = (1 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (0 \times 2^0) = 14$$

Cada dígito de um número binário é conhecido como **bit**. Nos exemplos da figura acima temos um número binário de **4 bits**.

Para mensurar a capacidade de processamento de um computador assim como a capacidade de armazenamento é utilizado o conceito de **byte** (representado pela letra **B** maiúscula).

Um **byte** equivale a aproximadamente **8 bits** e é representado pela letra **b** minúscula. Um **bit** pode ser representado somente por duas entidades: ou um dígito **0** ou um dígito **1**.

Um caractere digitado no teclado tem, aproximadamente, **8 bits**.

Quando falamos que um **byte** equivale a aproximadamente **8 bits**, estamos tratando da mensuração dos bytes que temos contato no dia-a-dia. Por exemplo, vejamos uma tabela que relaciona bits e bytes:

byte (B)	Correspondência em bytes	bit (b)	Correspondência em bits	Potência
1 byte	1 byte	8 bits	8 bits	2^0
1 kB	1.024 bytes	8 kb	8.192 bits	2^{10}
1 MB	1.048.576 bytes	8.192 kb	8.388.608 bits	2^{20}
1 GB	1.073.741.824 bytes	8.388.608 kb	8.589.934.592 bits	2^{30}
1 TB	1.099.511.627.776 bytes	8.589.934.592 kb	8.796.093.022.208 bits	2^{40}

Números Octais


Também conhecido como sistema numérico de base 8, pois utiliza 8 símbolos numéricos para sua representação: **0, 1, 2, 3, 4, 5, 6, 7 e 8**.

Foi muito utilizada em computação para representar de forma mais sucinta números binários, mas os números hexadecimais são mais utilizados para esta finalidade nos dias de hoje.

Similar aos números decimais e binários utiliza a posição colunar como elemento para determinação do expoente.

Dessa forma, um número **octal** segue normas parecidas com os números decimais e binários no que se refere à exponenciação:

número octal	1	2
coluna	1	0
potências	8^1	8^0
base 8	1×8^1	2×8^0
resultado	8	2
somatória (decimal)	$8 + 2 = 10$	



Números Hexadecimais

Os números hexadecimais são conhecidos como de base 16 e são utilizados na programação de microprocessadores. Oferecem uma forma mais legível para leitura, e, por isso, são muito utilizados em programação de **baixo nível**, por proporcionar uma facilidade em converter um número binário de 4 bits.

Utilizam-se de dezesseis algarismos, ou dígitos hexadecimais: **0**, **1**, **2**, **3**, **4**, **5**, **6**, **7**, **8**, **9**, **A**, **B**, **C**, **D**, **E** e **F**.

Os números hexadecimais têm a capacidade de reduzir uma longa sequência de números binários em poucos caracteres, pois qualquer combinação de números binários de 4 dígitos pode ser representada por um único símbolo hexadecimal.


Vejamos uma tabela comparativa entre Decimais, Binários, Hexadecimais e Octais:

Decimal	Binário	Hexadecimal	Octal
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	8	10
9	1001	9	11
10	1010	A	12

11	1011	B	13
12	1100	C	14
13	1101	D	15
14	1110	E	16
15	1111	F	17

Vejamos uma figura que ilustra a utilização da base 16 em um número hexadecimal:

número hexadecimal	4	C
coluna	1	0
potências	16^1	16^0
base 16	4×16^1	12×16^0
resultado	64	12
somatória (decimal)	$64 + 12 = 76$	



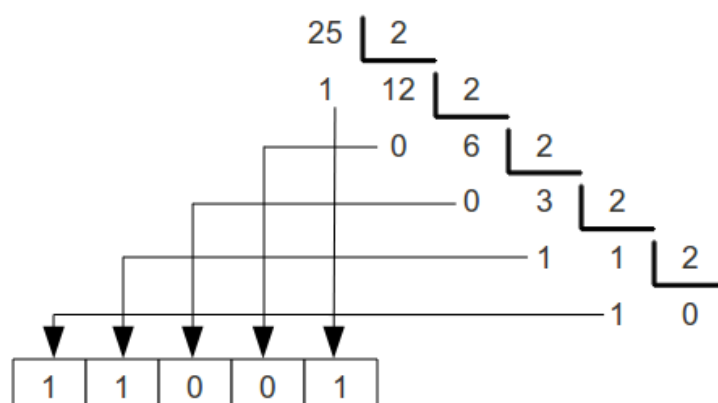
Note que o dígito "C" do número hexadecimal foi substituído pelo seu correspondente (12) na tabela acima.

Conversão Entre Bases Numéricas.

Conversão de Decimal para Binário

Para encontrar o número binário correspondente a um número decimal, são realizadas sucessivas divisões do número decimal por 2.

Em seguida, o resto da divisão de cada operação é coletado de forma invertida, da última para a primeira operação de divisão, como na figura, onde foi obtido o número binário correspondente ao número decimal **25**:



Na figura acima vemos que o número decimal foi dividido sucessivamente por 2 e os resultados foram coletados da última para a primeira divisão, formando o número binário.

Conversão de Binário para Decimal

Como vimos na lição anterior, para descobrir o número decimal correspondente a um número binário, basta calcular a soma de cada um dos dígitos do número binário multiplicado por 2 (que é a sua base) elevado à posição colunar do número, que, da direita para a esquerda começa em 0.

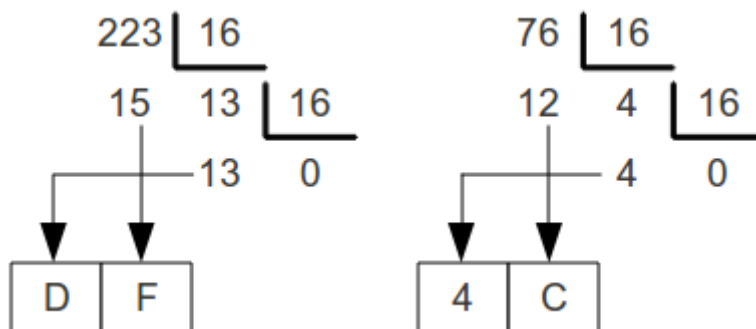
Vejamos uma conversão do número binário que obtivemos na conversão acima:

1	1	0	0	1
1×2^4	1×2^3	0×2^2	0×2^1	1×2^0
$16 + 8 + 0 + 0 + 1 = 25$				

Conversão de Decimal para Hexadecimal

A conversão de números decimais para hexadecimais é idêntica à conversão de decimal para binário, exceto que a divisão deve ser realizada por 16, que é a base dos hexadecimais.

Quando tiver dúvida sobre o valor em hexadecimal de algum resto, verifique na tabela da lição anterior.



Conversão de Hexadecimal em Decimal

A conversão de números hexadecimais em decimais é realizada através da soma dos dígitos hexadecimais multiplicados pela base 16 elevada à posição colunar contando da direita para a esquerda, começando em 0, de forma semelhante à conversão de binários em decimais:

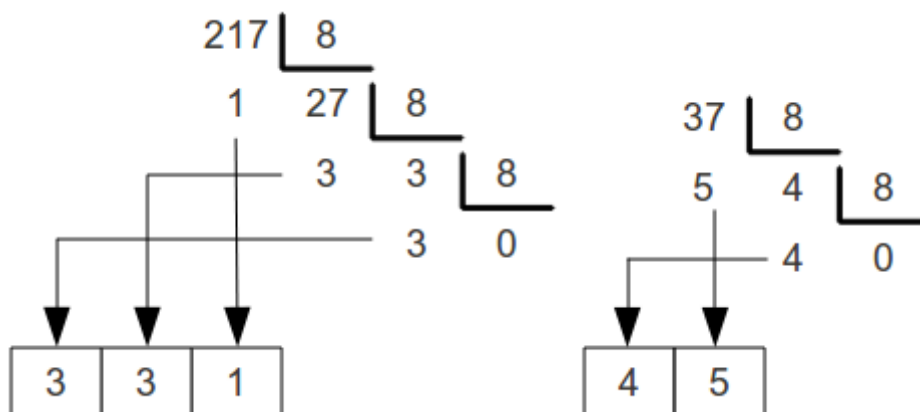
1	0	A
1×16^2	0×16^1	10×16^0
$256 + 0 + 10 = 266$		

C	B
12×16^1	11×16^0
$192 + 11 = 203$	

Note que os caracteres que definem os dígitos hexadecimais **A**, **B** e **C** foram substituídos pelos valores equivalentes em decimais **10**, **11** e **12** de acordo com a tabela da lição anterior para a realização do cálculo.

Conversão de Decimal em Octal

Assim como nas conversões anteriores, divide-se o decimal pela base para a qual se quer obter o número, no caso, 8:



Vimos que foram coletados os restos de cada divisão da última para a primeira para formar o número octal.

Conversão de Octal em Decimal

A conversão de números octais em decimais é obtida através da soma dos dígitos do número octal multiplicados pela base 8 elevada à posição colunar do dígito, começando em 0 da direita para a esquerda:

3	3	1
3×8^2	3×8^1	1×8^0
$192 + 24 + 1 = 217$		

4	5
4×8^1	5×8^0
$32 + 5 = 37$	

Conversão de Binário em Hexadecimal

Para converter um número binário em hexadecimal, separa-se o número binário em grupos de 4 bits, da direita para a esquerda. Em seguida, transforma-se cada grupo de 4 bits em hexadecimal. Ao final, simplesmente une-se os resultados em um só:

0	1	0	1	1	0	1	1
0	1	0	1	1	0	1	1
0 x2 ³	1 x2 ²	0 x2 ¹	1 x2 ⁰	1 x2 ³	0 x2 ²	1 x2 ¹	1 x2 ⁰
0 + 4 + 0 + 1 = 5 ₁₀				8 + 0 + 2 + 1 = 11 ₁₀			
5 ₁₀ = 5 ₁₆				11 ₁₀ = B ₁₆			
5B							

Caso o número de dígitos do número binário não seja múltiplo de 4, completa-se os dígitos à esquerda com zeros (0):

		1	1	1	1	0	1
0	0	1	1	1	1	0	1
0×2^3	0×2^2	1×2^1	1×2^0	1×2^3	1×2^2	0×2^1	1×2^0
$0 + 0 + 2 + 1 = 3_{10}$				$8 + 4 + 0 + 1 = 13_{10}$			
$3_{10} = 3_{16}$				$13_{10} = D_{16}$			
3D							

Conversão de Binário em Octal

Para converter números binários em octais, separa-se os dígitos do número binário em grupos de 3 bits da direita para a esquerda. Em seguida transforma-se cada grupo individual de 3 bits em octal. Ao final, unem-se os resultados:

1	1	0	1	0	1
1	1	0	1	0	1
1×2^2	1×2^1	0×2^0	1×2^2	0×2^1	1×2^0
4 + 2 + 0			4 + 0 + 1		
$6_{10} = 6_8$			$5_{10} = 5_8$		
65					

Caso o número de dígitos do número binário não seja múltiplo de 3, completa-se os dígitos à esquerda com zeros (0):

	1	1	1	1	1
0	1	1	1	1	1
0×2^2	1×2^1	1×2^0	1×2^2	1×2^1	1×2^0
$0 + 2 + 1$			$4 + 2 + 1$		
$3_{10} = 3_8$			$7_{10} = 7_8$		
37					

Conversão de Hexadecimal em Binário

Para converter números hexadecimais em binários, decompõem-se o número hexadecimal diretamente em binários de 4 dígitos. Os zeros mais à esquerda do resultado binário podem ser omitidos:

1				2				F			
$0 + 0 + 0 + 1 = 1$				$0 + 0 + 2 + 0 = 2$				$8 + 4 + 2 + 1 = 15$			
0×2^3	0×2^2	0×2^1	1×2^0	0×2^3	0×2^2	1×2^1	0×2^0	1×2^3	1×2^2	1×2^1	1×2^0
0	0	0	1	0	0	1	0	1	1	1	1
			1	0	0	1	0	1	1	1	1

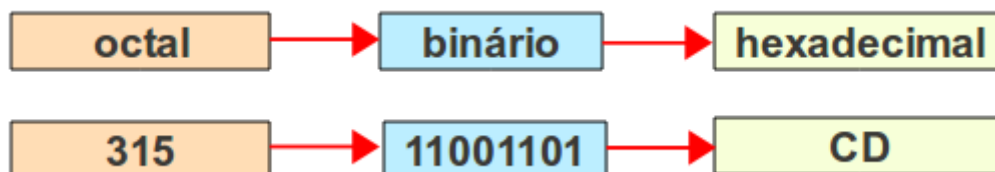
Conversão de Octal em Binário

Para converter números octais em binários, decompõem-se o número octal diretamente em binários de 3 dígitos. Os zeros mais à esquerda do resultado binário podem ser omitidos:

1			2			3		
$0 + 0 + 1 = 1$			$0 + 2 + 0 = 2$			$0 + 2 + 1 = 3$		
0×2^2	0×2^1	1×2^0	0×2^2	1×2^1	0×2^0	1×2^2	1×2^1	1×2^0
0	0	1	0	1	0	0	1	1
		1	0	1	0	0	1	1

Conversão de Octal em Hexadecimal

Para converter um número octal em hexadecimal, transforma-se primeiro o octal em binário e em seguida o binário em hexadecimal:



Conversão de Hexadecimal em Octal

Para converter um número hexadecimal em octal, transforma-se primeiro o hexadecimal em binário e em seguida o binário em octal:

