

# Sistemas de Numeração

# Por que Binário?

- Primeiros computadores projetados eram decimais
  - Mark I e ENIAC
- John von Neumann propôs processamento com dados binários (1945)
  - Simplificava o projeto de computadores
  - Usado tanto por instruções como por dados
- Relação natural entre comutadores on/off e cálculos com lógica Booleana

	
On	Off
Verdadeiro	Falso
Sim	Não
1	0

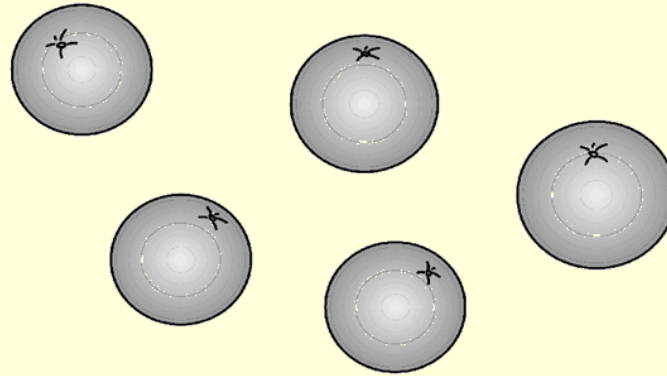
# Contagem e Aritmética

- Decimal ou sistema de base 10
  - Origem: contando nos dedos
  - “Dígito” vem do Latim *digitus*, que significa “dedo”
- *Base*: o número de dígitos diferentes no sistema numérico, incluindo zero
- *Decimal* ou *base 10*: 10 dígitos, 0 até 9
- *Binário* ou *base 2*: 2 dígitos, 0 e 1
  - *Bit* (dígito binário)
- *Octal* ou *base 8*: 8 dígitos, 0 até 7
- *Hexadecimal* ou *base 16*: 16 dígitos, 0 até F
  - Exemplos:  $10_{10} = A_{16}$ ;  $11_{10} = B_{16}$

# Considerando os Bits

- Bits são normalmente armazenados e manipulados em grupos
  - 8 bits = 1 *byte*
  - 4 bytes = 1 *palavra* (em sistemas de 32 bits)
- Número de bits usados em cálculos
  - Afetam a precisão dos resultados
  - Limitam o tamanho dos números manipulados pelo computador

# Números: Representação Física



- Diferentes numerais, mesmo número de laranjas
  - Homem das cavernas: IIIII
  - Romano: V
  - Árábico: 5
- Diferentes bases, mesmo número de laranjas
  - $5_{10}$
  - $101_2$
  - $12_3$

# Sistemas de Numeração

- Romanos: independentes da posição
- Moderno: baseado na notação posicional (valor posicional)
  - Decimal: sistema de notação **posicional** baseado em potências de 10.
  - Binário: sistema de notação **posicional** baseado potências de 2
  - Octal : sistema de notação **posicional** baseado em potências de 8
  - Hexadecimal: sistema de notação **posicional** baseado em potências de 16

# Sistemas Numéricos mais Comuns

Sistema	Base	Símbolos	Usado por humanos?	Usado por computadores?
Decimal	10	0, 1, ... 9	Sim	Não
Binário	2	0, 1	Não	Sim
Octal	8	0, 1, ... 7	Não	Não
Hexa-decimal	16	0, 1, ... 9, A, B, ... F	Não	Não

# Quantidades / Contagem (1 de 3)

Decimal	Binário	Octal	Hexa-decimal
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



# Quantidades / Contagem (2 de 3)

Decimal	Binário	Octal	Hexa-decimal
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

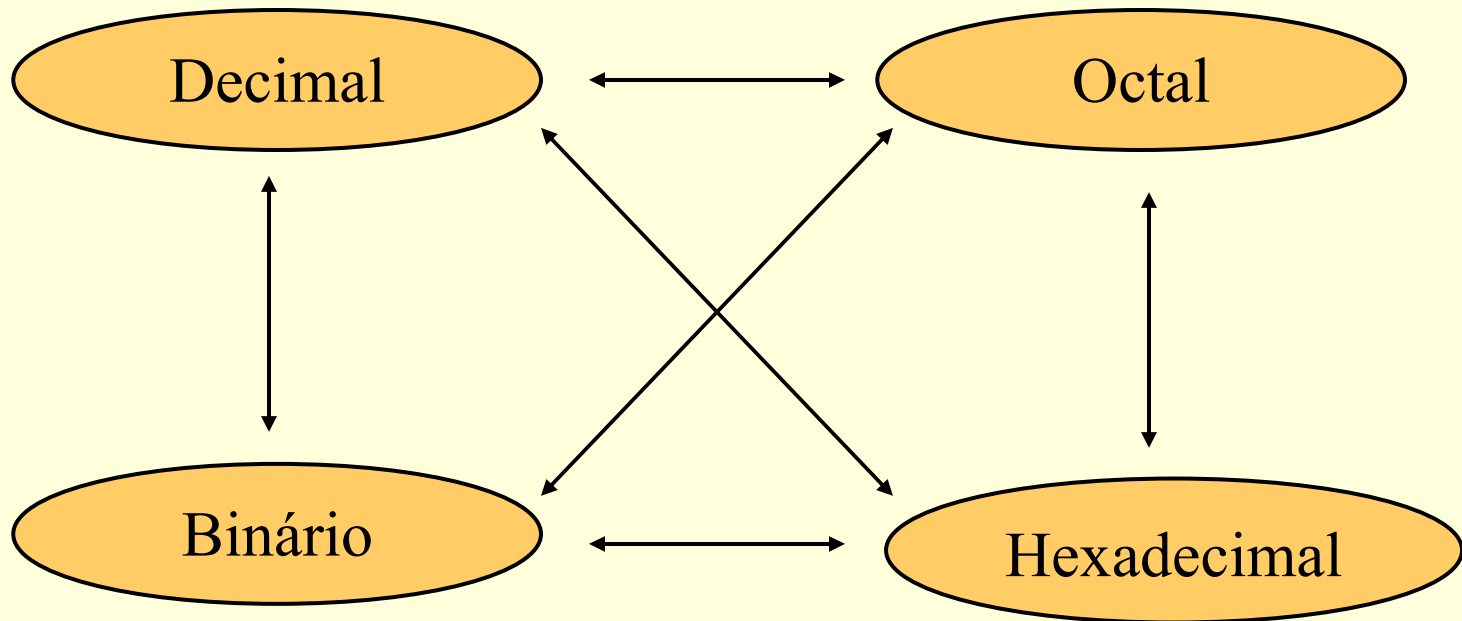
# Quantidades / Contagem (3 de 3)

Decimal	Binário	Octal	Hexa-decimal
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13
20	10100	24	14
21	10101	25	15
22	10110	26	16
23	10111	27	17

Etc.

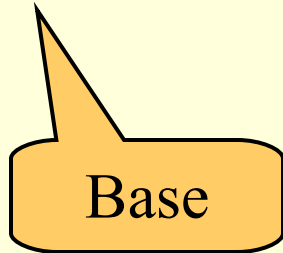
# Conversão Entre Bases

- Possibilidades:

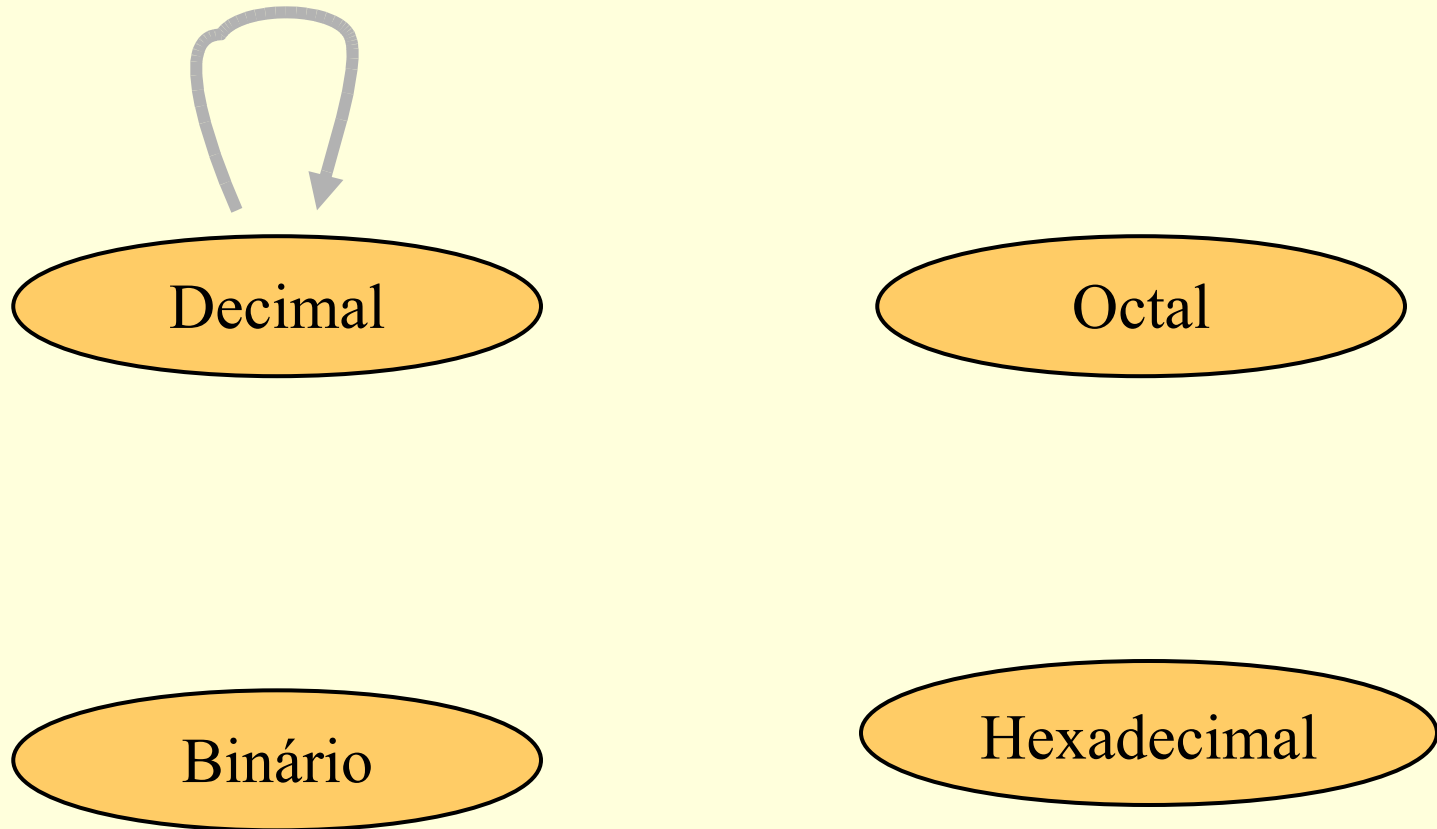


# Exemplo

$$25_{10} = 11001_2 = 31_8 = 19_{16}$$



# Decimal para Decimal (só para entender)



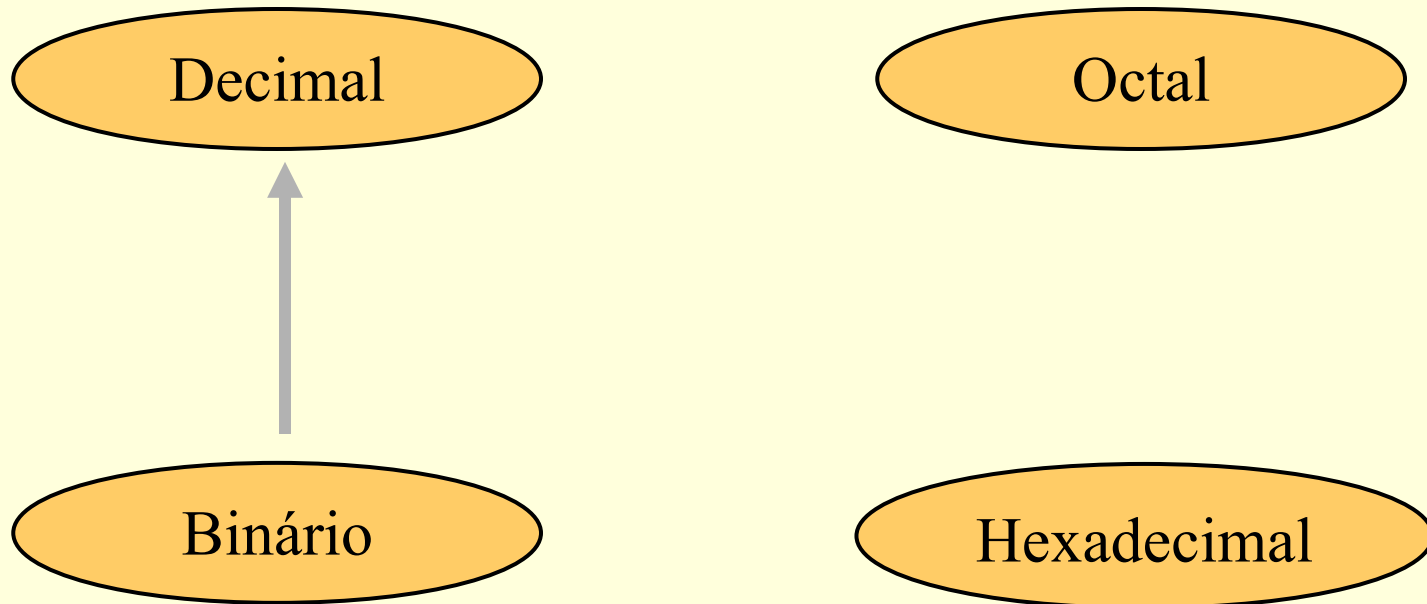
$125_{10} \Rightarrow$

$5 \times 10^0$	$=$	$5$
$2 \times 10^1$	$=$	$20$
$1 \times 10^2$	$=$	$100$
		<hr/>
		$125$

**Peso**

**Base**

# Binário para Decimal

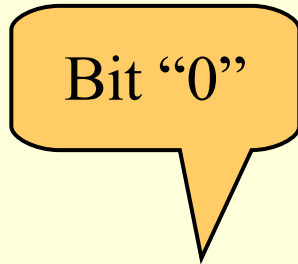


# Binário para Decimal

- Técnica
  - Multiplique cada bit por  $2^n$ , onde  $n$  é o “peso” do bit
  - O peso é a posição do bit, começando em 0 à direita
  - Adicione os resultados



# Exemplo



$101011_2 \Rightarrow$

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

$$1 \times 2^1 = 2$$

$$0 \times 2^2 = 0$$

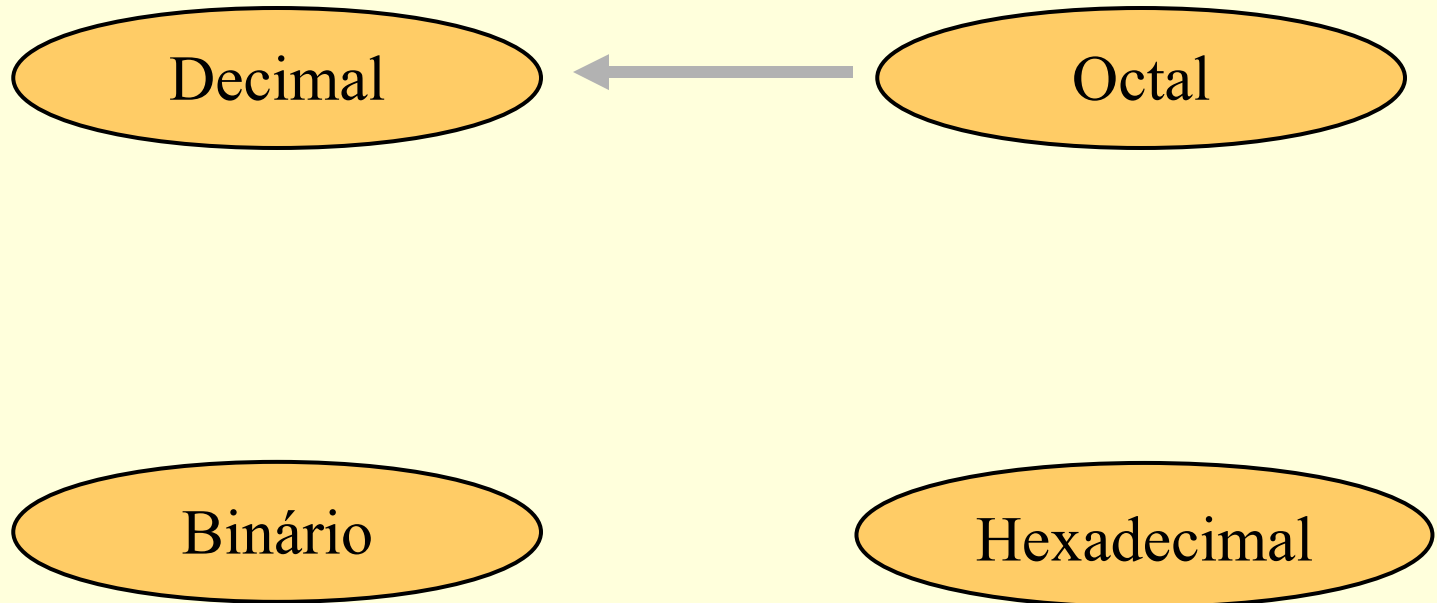
$$1 \times 2^3 = 8$$

$$0 \times 2^4 = 0$$

$$1 \times 2^5 = \underline{32}$$

$43_{10}$

# Octal para Decimal



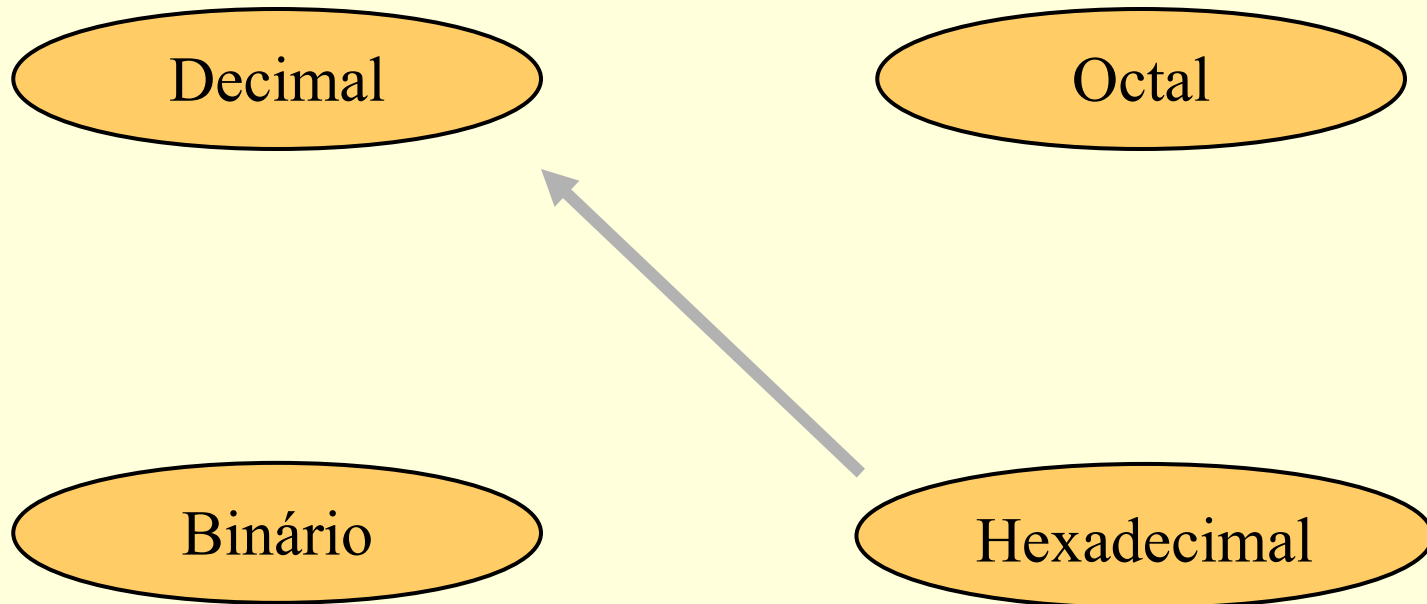
# Octal para Decimal

- Técnica
  - Multiplique cada bit por  $8^n$ , onde  $n$  é o “peso” do bit
  - O peso é a posição do bit, começando em 0 à direita
  - Adicione os resultados

# Exemplo

$$\begin{array}{r} 724_8 \Rightarrow \\ 4 \times 8^0 = 4 \\ 2 \times 8^1 = 16 \\ 7 \times 8^2 = 448 \\ \hline 468_{10} \end{array}$$

# Hexadecimal para Decimal



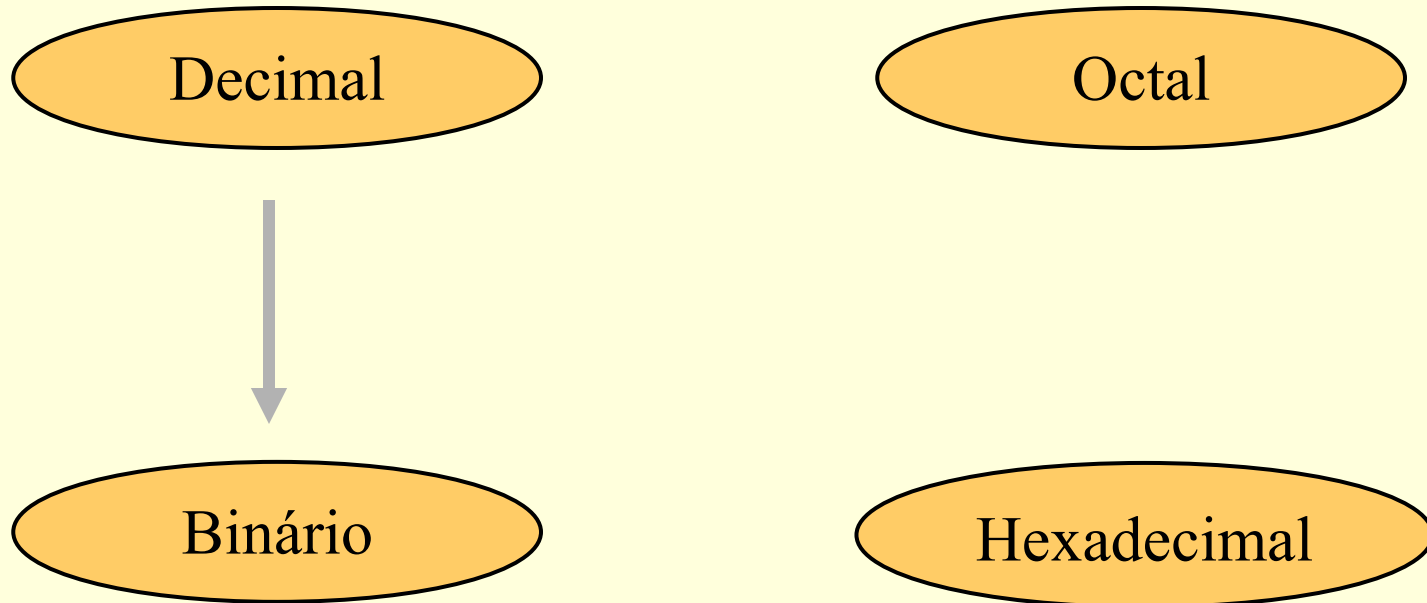
# Hexadecimal para Decimal

- Técnica
  - Multiplique cada bit por  $16^n$ , onde  $n$  é o “peso” do bit
  - O peso é a posição do bit, começando de 0 à direita
  - Adicione os resultados

# Exemplo

$$\begin{array}{r} \text{ABC}_{16} \Rightarrow \\ \text{C} \times 16^0 = 12 \times 1 = 12 \\ \text{B} \times 16^1 = 11 \times 16 = 176 \\ \text{A} \times 16^2 = 10 \times 256 = 2560 \\ \hline 2748_{10} \end{array}$$

# Decimal para Binário





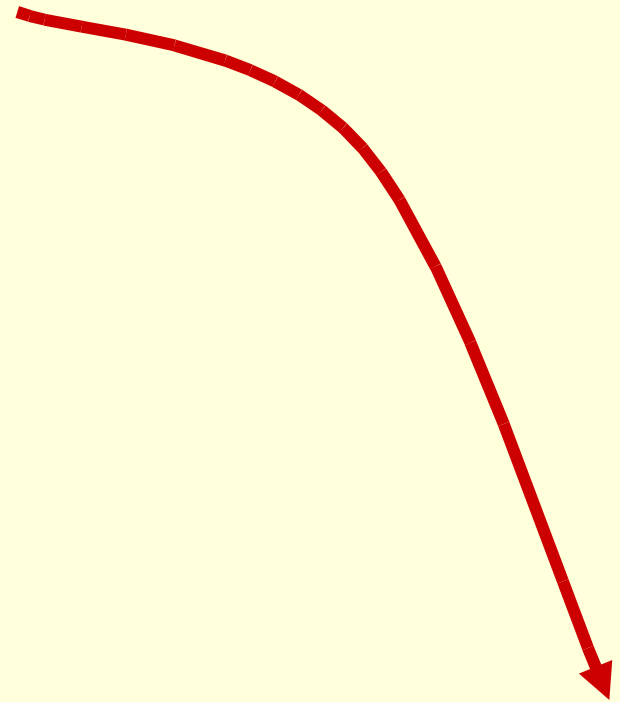
# Decimal para Binário

- Técnica
  - Divida por dois, guardando os restos
  - Primeiro resto é o bit 0 (bit menos significativo)
  - Segundo resto é o bit 1
  - Etc.

# Exemplo

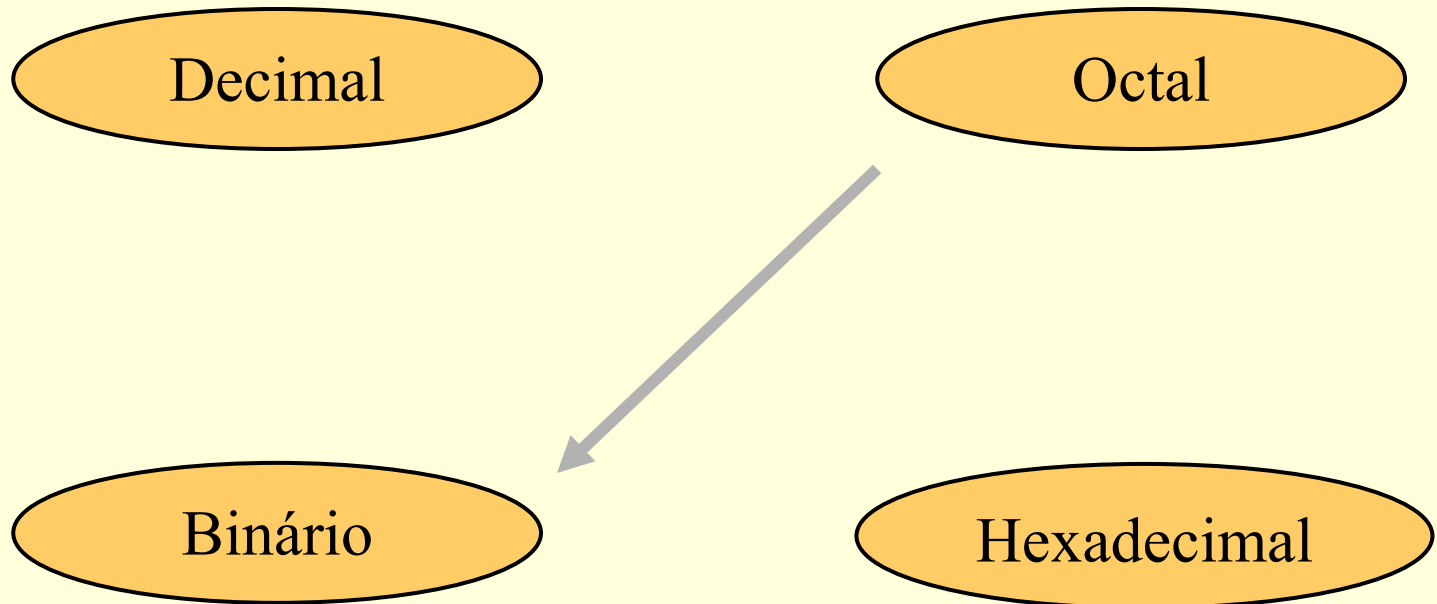
$$125_{10} = ?_2$$

2		125	
2		62	1
2		31	0
2		15	1
2		7	1
2		3	1
2		1	1
		0	1



$$125_{10} = 1111101_2$$

# Octal para Binário

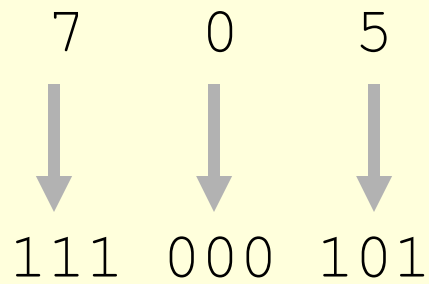


# Octal para Binário

- Técnica
  - Converta cada dígito octal para uma representação binária equivalente de 3 bits

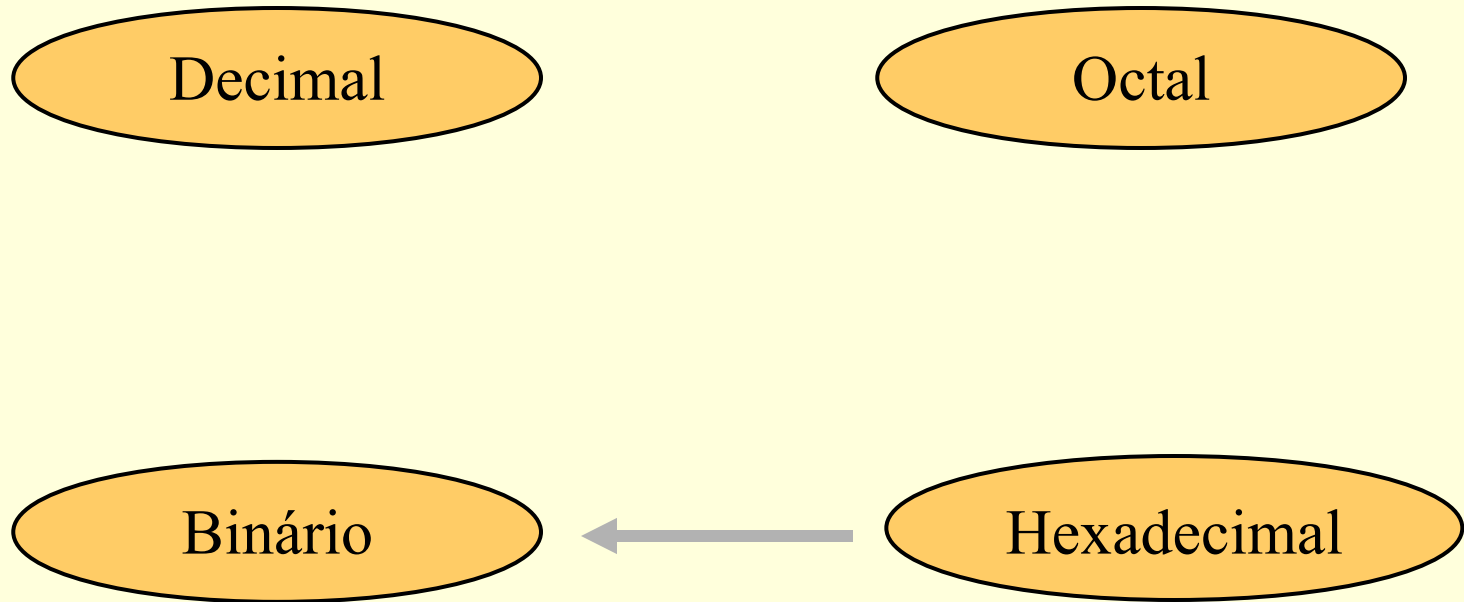
# Exemplo

$$705_8 = ?_2$$



$$705_8 = 111000101_2$$

# Hexadecimal para Binário

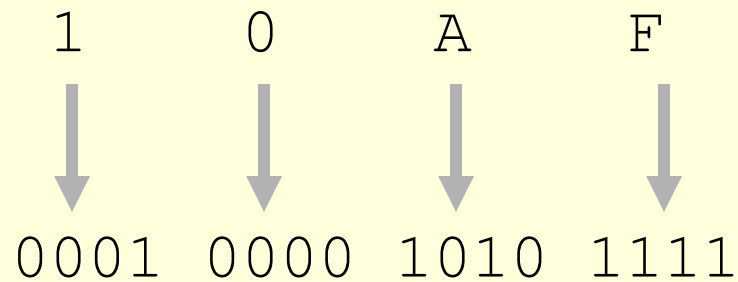


# Hexadecimal para Binário

- Técnica
  - Converta cada dígito hexadecimal para uma representação binária equivalente de 4 bits.

# Exemplo

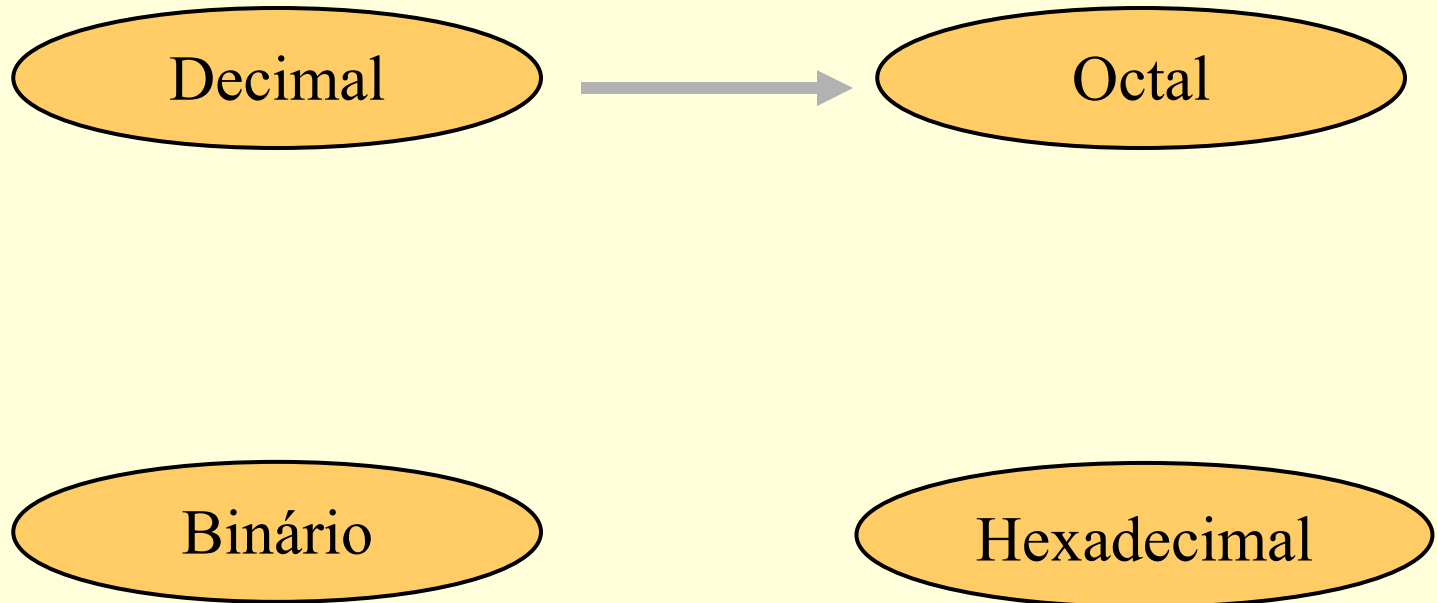
$$10AF_{16} = ?_2$$



$$10AF_{16} = 0001000010101111_2$$



# Decimal para Octal



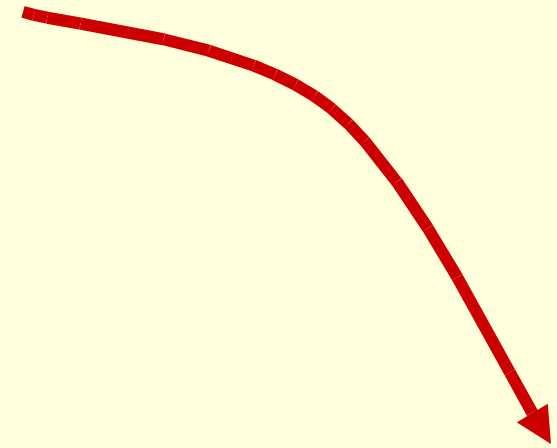
# Decimal para Octal

- Técnica
  - Divida por 8
  - Guarde os restos

# Exemplo

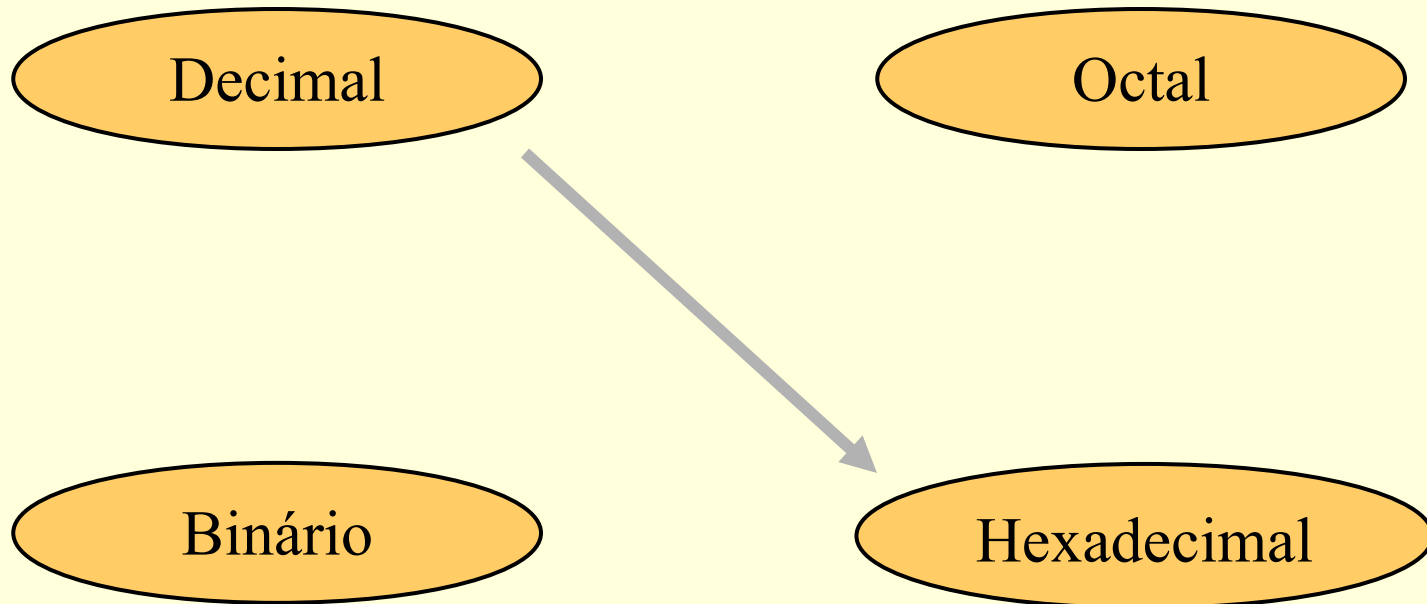
$$1234_{10} = ?_8$$

$$\begin{array}{r|l} 8 & 1234 \\ \hline 8 & 154 \quad 2 \\ \hline 8 & 19 \quad 2 \\ \hline 8 & 2 \quad 3 \\ \hline & 0 \quad 2 \end{array}$$



$$1234_{10} = 2322_8$$

# Decimal para Hexadecimal



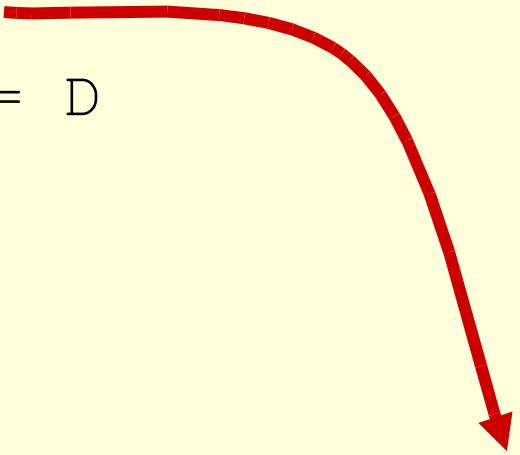
# Decimal para Hexadecimal

- Técnica
  - Divida por 16
  - Guarde os restos

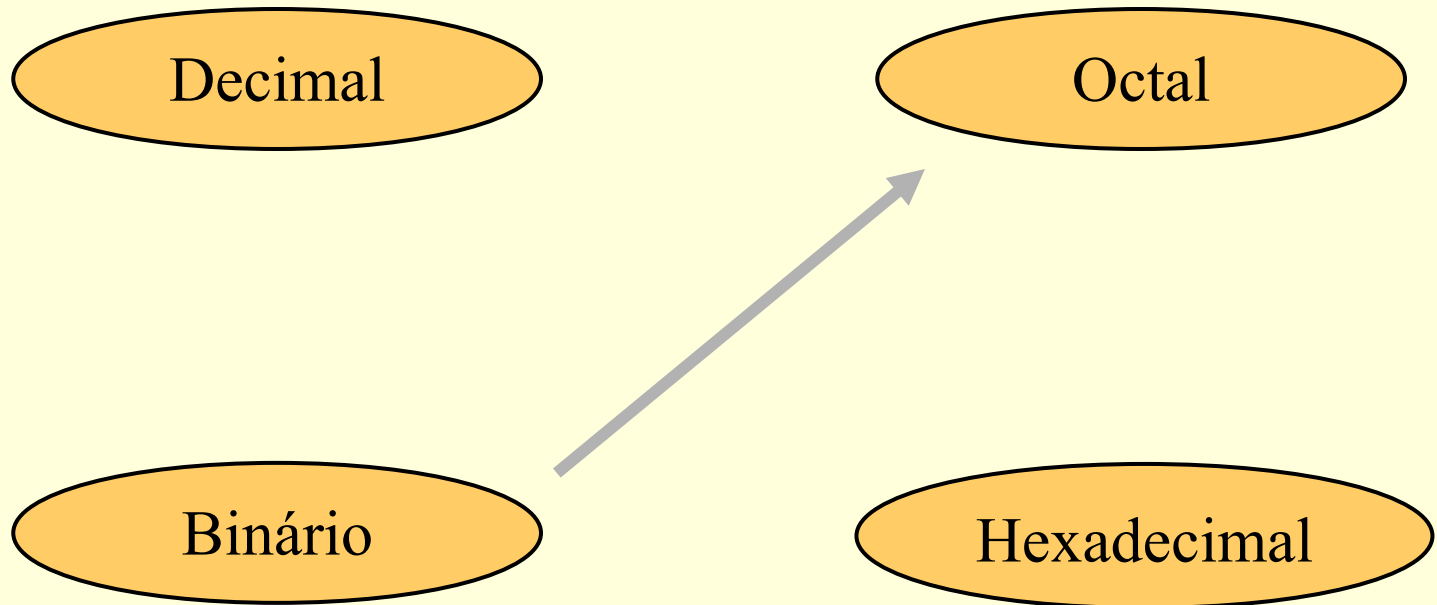
# Exemplo

$$1234_{10} = ?_{16}$$

$$\begin{array}{r|l} 16 & 1234 \\ \hline 16 & \phantom{1}77 \quad 2 \\ \hline 16 & \phantom{1}4 \quad 13 = D \\ \hline & 0 \quad 4 \end{array}$$


$$1234_{10} = 4D2_{16}$$

# Binário para Octal



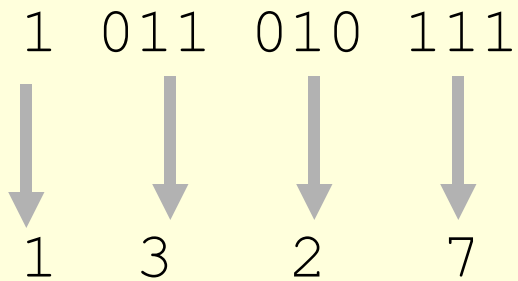
# Binário para Octal

- Técnica
  - Divida os bits em grupos de três, começando à direita
  - Converta para dígitos octais



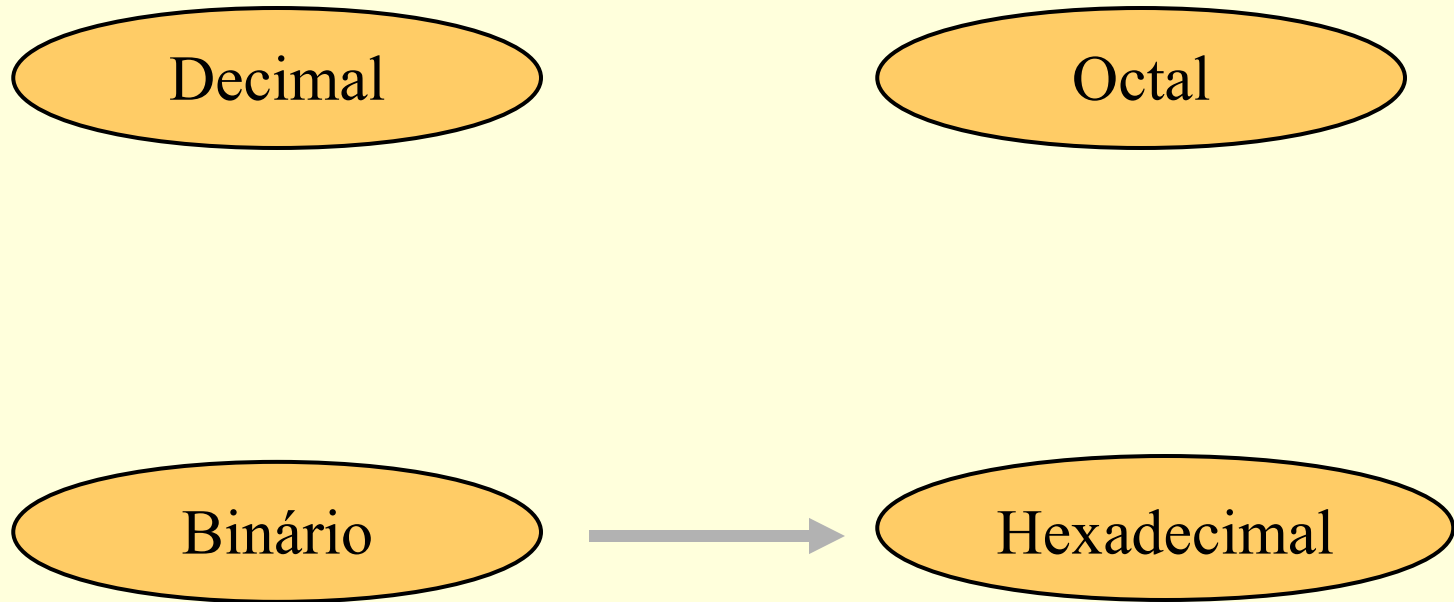
# Exemplo

$$1011010111_2 = ?_8$$



$$1011010111_2 = 1327_8$$

# Binário para Hexadecimal

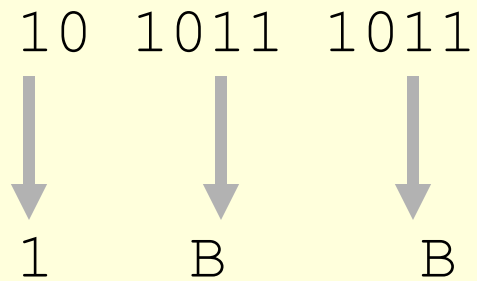


# Binário para Hexadecimal

- Técnica
  - Divida os bits em grupos de quatro, começando à direita
  - Converta para dígitos hexadecimais

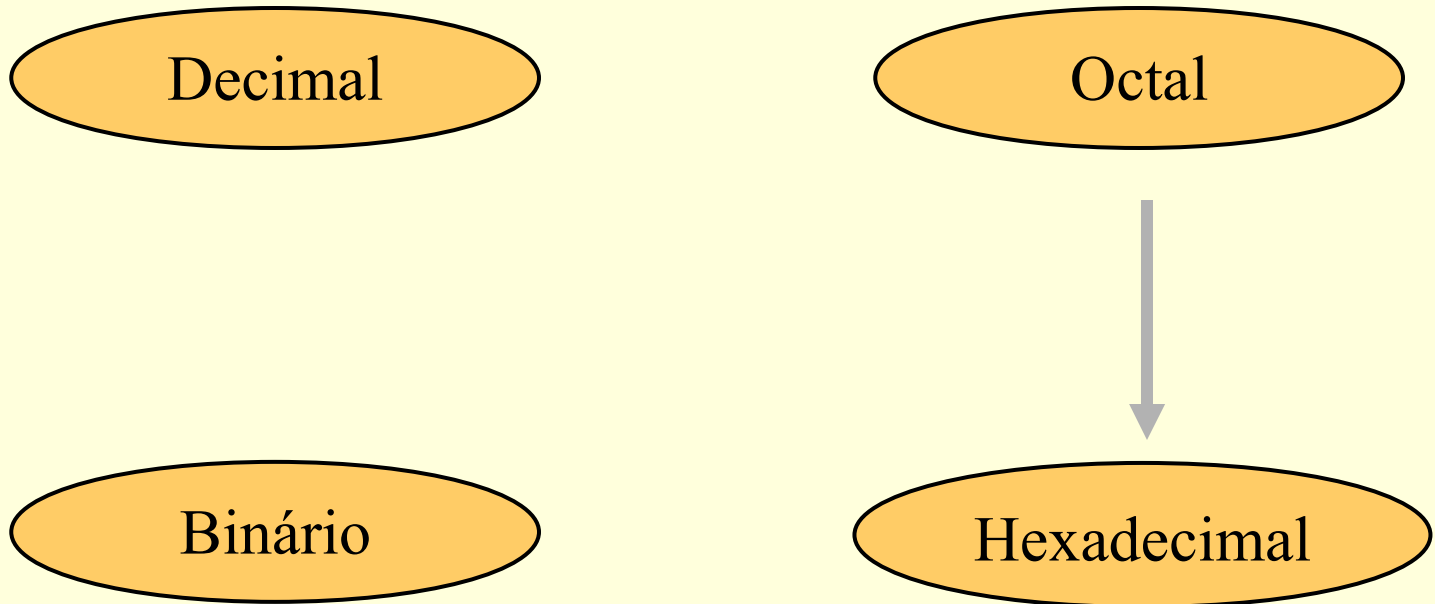
# Exemplo

$$1010111011_2 = ?_{16}$$



$$1010111011_2 = 2BB_{16}$$

# Octal para Hexadecimal

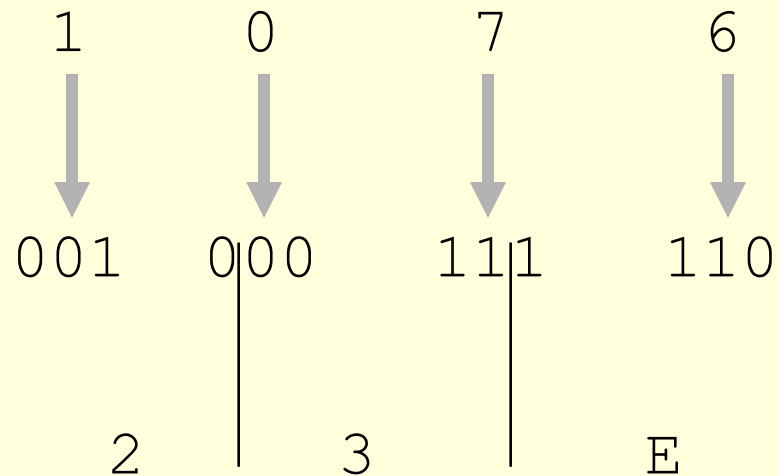


# Octal para Hexadecimal

- Técnica
  - Use Binário como uma representação intermediária

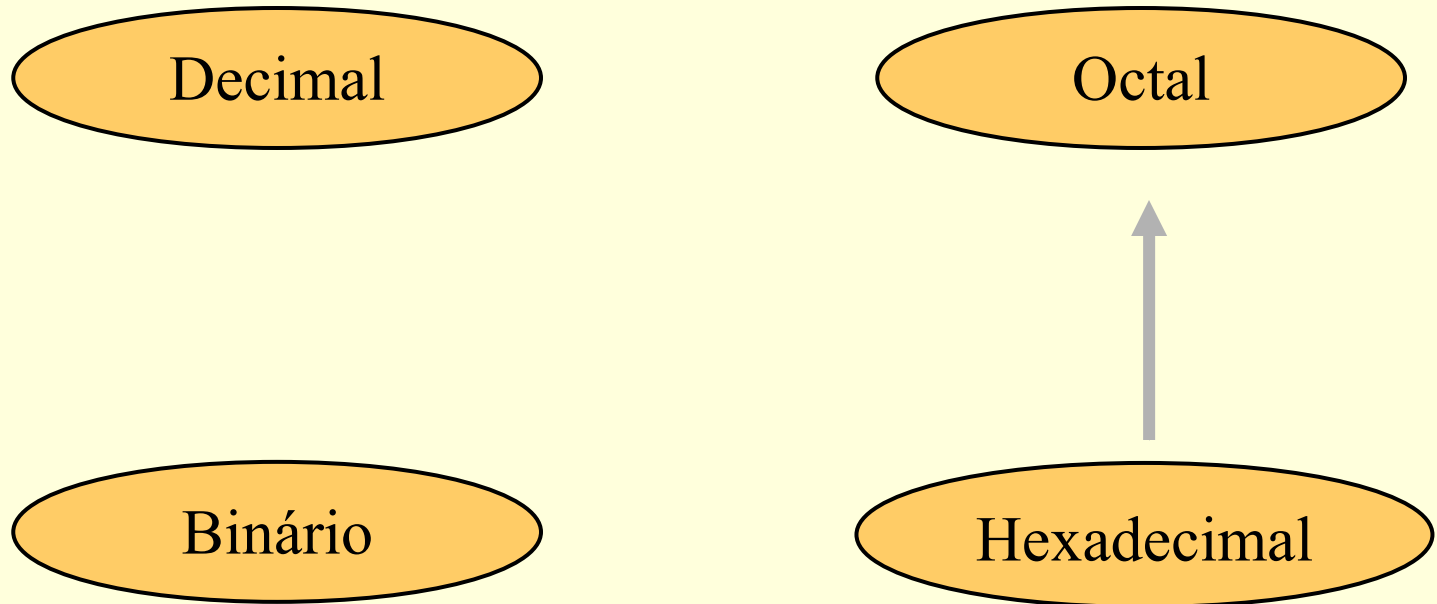
# Exemplo

$$1076_8 = ?_{16}$$



$$1076_8 = 23E_{16}$$

# Hexadecimal para Octal



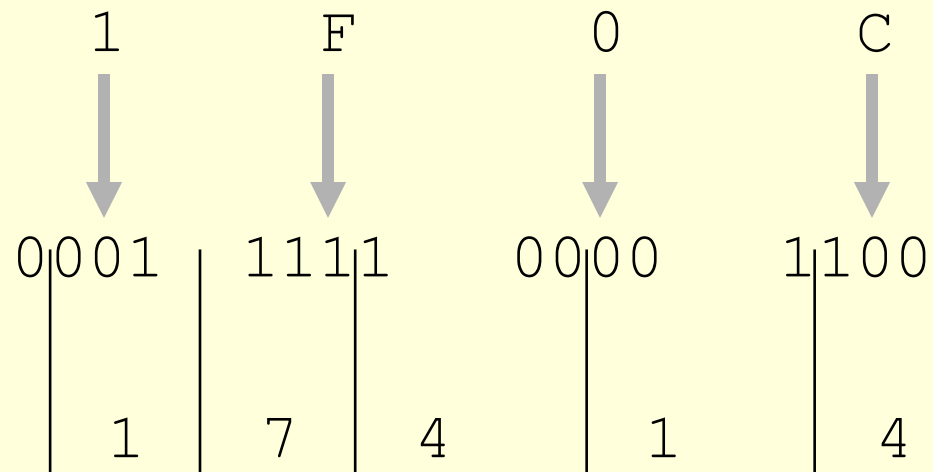


# Hexadecimal para Octal

- Técnica
  - Use Binário como uma representação intermediária

# Exemplo

$$1F0C_{16} = ?_8$$



$$1F0C_{16} = 17030_8$$

# Exercício – Converta ...

Decimal	Binário	Octal	Hexa-decimal
33			
	1110101		
		703	
			1AF

Não use calculadora!

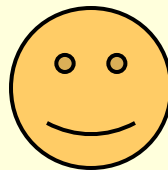
Pule a resposta

Resposta

# Exercício – Converta ...

Resposta

Decimal	Binário	Octal	Hexa-decimal
33	100001	41	21
117	1110101	165	75
451	111000011	703	1C3
431	110101111	657	1AF



# Potências mais Comuns (1 de 2)

- Base 10

Potência	Prefixo	Símbolo	Valor
$10^{-12}$	pico	p	.000000000001
$10^{-9}$	nano	n	.000000001
$10^{-6}$	micro	$\mu$	.000001
$10^{-3}$	mili	m	.001
$10^3$	kilo	k	1000
$10^6$	mega	M	1000000
$10^9$	giga	G	1000000000
$10^{12}$	tera	T	1000000000000

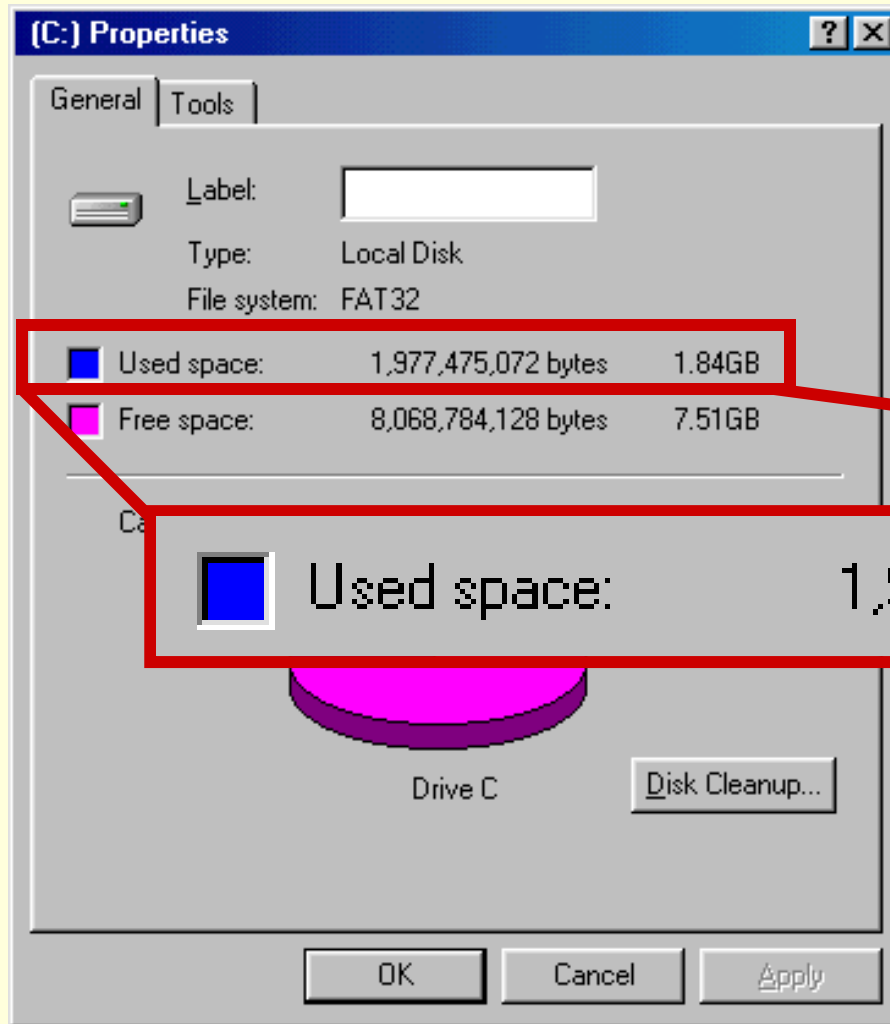
# Potências mais Comuns (2 de 2)

- Base 2

Potência	Prefixo	Símbolo	Valor
$2^{10}$	kilo	k	1024
$2^{20}$	mega	M	1048576
$2^{30}$	Giga	G	1073741824

- O que são os valores de “k”, “M”, e “G”?
- Em computação, em particular com memórias, a interpretação de base-2 geralmente se aplica

# Exemplo



No laboratório...

1. Clique duplo em Meu Computador
2. Clique com o botão direito em C:
3. Clique em Propriedades

$$/ 2^{30} =$$

# Exercício – Espaço Livre

- Determine o “espaço livre” de todos os *drives* de um computador do laboratório

Drive	Espaço Livre	
	Bytes	GB
A:		
C:		
D:		
E:		
etc.		



# Revisão – multiplicando potências

- Para bases comuns, adicione os expoentes

$$a^b \times a^c = a^{b+c}$$

$$2^6 \times 2^{10} = 2^{16} = 65,536$$

ou ...

$$2^6 \times 2^{10} = 64 \times 2^{10} = 64k$$

# Adição Binária (1 de 2)

- Dois valores de 1-bit

A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	10

“dois”

# Adição Binária (2 de 2)

- Dois valores de  $n$ -bits
  - Adicione os bits individualmente
  - Propague as sobras
  - E.g.,

$$\begin{array}{r} \phantom{+} \overset{1}{1}01\overset{1}{0}1 \\ + 11001 \\ \hline 101110 \end{array} \qquad \begin{array}{r} \phantom{+} 21 \\ + 25 \\ \hline 46 \end{array}$$

# Multiplicação (1 de 3)

- Decimal (só para entender)

$$\begin{array}{r} 35 \\ \times 105 \\ \hline 175 \\ 000 \\ 35 \\ \hline 3675 \end{array}$$

# Multiplicação (2 de 3)

- Binário, dois valores de 1-bit

A	B	$A \times B$
0	0	0
0	1	0
1	0	0
1	1	1

# Multiplicação (3 de 3)

- Binário, dois valores de  $n$ -bits
  - Como no caso de valores decimais
  - E.g.,

$$\begin{array}{r} 1110 \\ \times 1011 \\ \hline 1110 \\ 1110 \\ 0000 \\ 1110 \\ \hline 10011010 \end{array}$$

# Frações

- Decimal para decimal (só para entender)

$$\begin{array}{r} 3.14 \Rightarrow \\ 4 \times 10^{-2} = 0.04 \\ 1 \times 10^{-1} = 0.1 \\ 3 \times 10^0 = 3 \\ \hline 3.14 \end{array}$$

# Frações

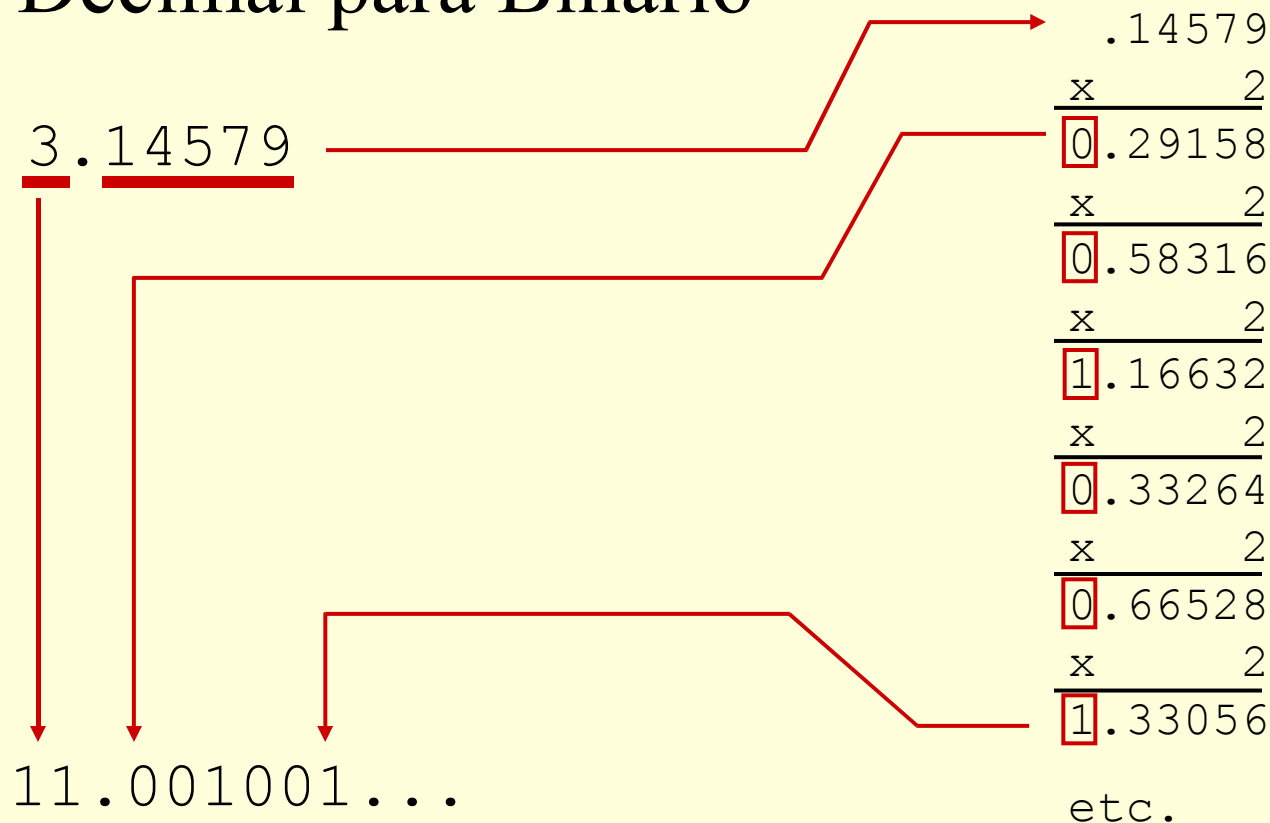
- Binário para decimal

$$\begin{array}{r} 10.1011 \Rightarrow \\ 1 \times 2^{-4} = 0.0625 \\ 1 \times 2^{-3} = 0.125 \\ 0 \times 2^{-2} = 0.0 \\ 1 \times 2^{-1} = 0.5 \\ 0 \times 2^0 = 0.0 \\ 1 \times 2^1 = 2.0 \\ \hline 2.6875 \end{array}$$



# Frações

- Decimal para Binário



# Exercício – Converta ...

Decimal	Binário	Octal	Hexa-decimal
29.8			
	101.1101		
		3.07	
			C.82

Não use calculadora!

Pule a resposta

Resposta

# Exercício – Converta ...

Resposta

Decimal	Binário	Octal	Hexa-decimal
29.8	11101.110011...	35.63...	1D.CC...
5.8125	101.1101	5.64	5.D
3.109375	11.000111	3.07	3.1C
12.5078125	1100.10000010	14.404	C.82

