Universidade Federal da Fronteira Sul
Curso de Ciência da Computação
Campus Chapecó

Sistemas de Numeração

Prof. Luciano L. Caimi
lcaimi@uffs.edu.br

Sistemas de Numeração

O número é um conceito abstrato que representa a idéia de quantidade.

Um **Sistema de Numeração (SN)** é o conjunto de símbolos utilizados para a representação de quantidades e as regras que definem a forma de representação. Os SN podem ser:

- Não posicional
- Posicional

Sistemas de Numeração

Sistema de Numeração Não Posicional

Cada símbolo representa um valor fixo, independente de sua posição relativa ao número.

Exemplo: sistema de algarismos romanos.

Símbolos: I, V, X, L, C, D, M.

Regras:

- Cada símbolo colocado à direita de um maior é adicionado a este.
- Cada símbolo colocado à esquerda de um maior tem o seu valor subtraído do maior.

Sistemas de Numeração

Sistema de Numeração Posicional

O valor de cada símbolo é determinado de acordo com a sua posição no número.

Um sistema de numeração é determinado fundamentalmente pela BASE, que indica a quantidade de símbolos e o valor de cada símbolo.

Todos os sistemas posicionais, independen-te da BASE, possuem as mesmas regras de formação, contagem e operações aritméticas básicas.

Sistemas de Numeração

Teorema Fundamental da Numeração

O teorema fundamental da numeração expressa a característica principal dos sistemas posicionais:

$$N^o = \sum_{i=-d}^{n} (digito)_i * (base)^i$$

expandindo

$$\ldots + a_3 * B^3 + a_2 * B^2 + a_1 * B^1 + a_0 * B^0 + a_{-1} * B^{-1} + \ldots$$

Onde:
i = posição em relação à vírgula,
d = nº de dígitos à direita da vírgula,
n = nº de dígitos à esquerda da vírgula –1,
dígito = cada um símbolos dos que compõem o número.

Sistemas de Numeração

Teorema Fundamental da Numeração

O valor total do número é a soma dos valores relativos de cada algarismo (decimal).

$\boxed{735=700+30+5}$

O algarismo 5 representa 5 unidades, o algarismo 3 representa 3 dezenas, e por último que o algarismo 7 representa 7 centenas …

$\boxed{573=500+70+3}$

Já no 2º exemplo é diferente

Sistemas de Numeração

- **Base Decimal: 10 símbolos**

Elementos: 0,1,2,3,4,5,6,7,8,9

Esta “regra” é válida independente da base de numeração utilizada

Contagem:

..00
..01 “encheu a casa menos significativa: reinicia a contagem nesta casa e usa o próximo símbolo na casa a sua esquerda”
..02
..03
..04
..05 assim: ..10 ..20 ..100
..06 ..11 ..21
..07 ....
..08 ..19 ..99
..09

Sistemas de Numeração

Base Decimal

cada casa decimal possui um peso 10 vezes maior do que a casa a sua direita

...	100	10	1

Considerando um número com N dígitos (ou casas) teremos capacidade de representar

B^N valores diferentes (onde B é a base de numeração).

Para 3 dígitos decimais teremos:

$10^3 = 1000$ valores

Base Decimal

O maior valor a ser representado com N dígitos será:

$B^N - 1$ (onde B é a base de numeração).

Para 3 dígitos decimais teremos:

$10^3 - 1 = 999$

Considerando o maior símbolo possível

9	9	9

Sistemas de Numeração

Base Decimal:

Adição: quando a soma em uma determinada casa excede o maior símbolo da base, devemos deixar o excedente e levar o peso da base para a casa mais a esquerda valendo 1.

Exemplo:

	1	1	1	
	2	4	5	3
+		9	7	9
	3	3	4	2

3+9=12= 10+2

5+8+1=14=10+4

1+4+8=13=10+2

Sistemas de Numeração

Base Decimal:

Subtração: quando uma determinada casa necessita “pedir emprestado” a casa a sua esquerda fica com um a menos e a casa solicitante recebe o peso da base (10).

Exemplo:

		9			
	0	10	10		→ 10+5=15 ⇒ 15-8 = 7
	1	0	5	2	
-	0	4	8	1	
	0	5	7	1	

Sistemas de Numeração

- **Base Binária: 2 símbolos**

  Elementos: 0,1

Contagem:

.000 “encheu a casa menos significativa:

.001 reinicia a contagem nesta casa e

.010 usa o próximo símbolo na casa a sua

.011 esquerda”

.100

.101

.110

.111

.1000

Base Binária

- Cada casa ou dígito binário é chamado de bit (do inglês Binary Digit)

- Um agrupamento de 8 bits é chamado de Byte

- Pelo T.F.N. cada casa binária possui um peso 2 vezes maior do que a casa a sua direita

...	8	4	2	1

Sistemas de Numeração

Com n bits podemos representar: $B^n \Rightarrow 2^n$

Para 3 casas binárias (ou 3 bits) teremos:

$2^3 = 8$ valores

O maior valor a ser representado com N dígitos será:

$B^N - 1$

Para 5 bits teremos:

$2^5 - 1 = 31$

Considerando o maior símbolo possível:

1	1	1	1	1

Sistemas de Numeração

Base Binária:

Adição: quando a soma em uma determinada casa excede o maior símbolo da base, devemos deixar o excedente e levar o peso da base para a casa mais a esquerda valendo 1.

Exemplo:

	1	1	1	1		2=2+0
		1	0	1	1	3=2+1
+		0	1	1	1	
	1	0	0	1	0	

Sistemas de Numeração

Base Binária:

Subtração: quando uma determinada casa necessita “pedir emprestado” a casa a sua esquerda fica com um a menos e a casa solicitante recebe o peso da base(2).

Exemplo:

		1			
	0	2	2		2+0=2 ⇒ 2-1 = 1
	1	0	0	1	
-	0	1	1	1	
	0	0	1	0	

Sistemas de Numeração

Considerando o peso de cada casa teremos:

$2^0 = 1$	$2^{10} = 1024 = 1K$	$2^{20} = 1024K = 1M$
$2^1 = 2$	$2^{11} = 2048 = 2K$	$2^{21} = 2048K = 2M$
$2^2 = 4$	$2^{12} = 4096 = 4K$	$2^{22} = 4096K = 4M$
$2^3 = 8$	$2^{13} = 8192 = 8K$	$2^{23} = 8M$
$2^4 = 16$	$2^{14} = 16K$	$2^{24} = 16M$
$2^5 = 32$	$2^{15} = 32K$	$2^{25} = 32M$
$2^6 = 64$	$2^{16} = 64K$	$2^{26} = 64M$
$2^7 = 128$	$2^{17} = 128K$	$2^{27} = 128M$
$2^8 = 256$	$2^{18} = 256K$	$2^{28} = 256M$
$2^9 = 512$	$2^{19} = 512K$	$2^{29} = 512M$

Para valores entre 2^{30} e 2^{39}: Giga

Para valores entre 2^{40} e 2^{49}: Tera

Sistemas de Numeração

Exemplos de conversão

Considerando que 1 byte é um agrupamento de 8 bits teremos:

a) 56 bits = ? Bytes ⇒ 56/8 Bytes = 7 Bytes

b) 9 Bytes = ? bits ⇒ 9 x 8 bits = 72 bits

c) 32KBytes = ? bits ⇒ 32 * 8 Kbits

⇒ 256 * 1024 bits

⇒ 262144 bits

d) 131072Kbits = ? MBytes ⇒ 131072/8 KBytes

⇒ 16384 / 1024 KB

⇒ 16 MBytes

Sistemas de Numeração

- **Base Octal: 8 símbolos**

Elementos: 0,1,2,3,4,5,6,7

Contagem:

..00

..01 “encheu a casa menos significativa:

..02 reinicia a contagem nesta casa e

..03 usa o próximo símbolo na casa a sua

..04 esquerda”

..05 assim: ..10 ..20 ..100

..06 ..11 ..21

..07 ……… ………

..17 ..77

Sistemas de Numeração

cada casa octal possui um peso 8 vezes maior do que a casa a sua direita

...	64	8	1

Considerando um número com N dígitos (ou casas) teremos capacidade de representar

B^N valores diferentes (onde B é a base de numeração).

Para 3 dígitos octais teremos:

$8^3 = 512$ valores

Sistemas de Numeração

O maior valor a ser representado com N dígitos será:

$B^N - 1$ (onde B é a base de numeração).

Para 4 dígitos octais teremos:

$8^4 - 1 = 4095$

Considerando o maior símbolo possível

7	7	7	7	4095

Base Octal:

Adição: Quando a soma em uma determinada casa excede o maior símbolo da base, devemos deixar o excedente e levar o peso da base para a casa mais a esquerda valendo 1.

Exemplo:

$$
\begin{array}{cccccc}
 & 1 & 1 & 1 & & \\
 & 5 & 5 & 7 & 4 & \\
+ & 0 & 6 & 3 & 5 & \\
\hline
 & 6 & 4 & 3 & 1 &
\end{array}
$$

9=8+1

11=8+3

12=8+4

Base Binária:

Subtração: quando uma determinada casa necessita “pedir emprestado” a casa a sua esquerda fica com um a menos e a casa solicitante recebe o peso da base(8).

Exemplo:

		7			
	5	8	8		8+0=8 ⇒ 8-3 = 5
	6	0	0	5	
-	0	4	3	2	
	0	3	5	3	

Sistemas de Numeração

Base Hexadecimal: 16 símbolos

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

Contagem:

..00
..01
..02
..03
..04
..05
..06
..07
..08
..09
..0A
..0B
..0C

..0D
..0E
..0F

“**encheu a casa menos significativa:** reinicia a contagem nesta casa e usa o próximo símbolo na casa a sua esquerda”

..10
..11
..12
....

..19
..1A
..1B
.....

..1F
..20
....

Sistemas de Numeração

cada casa hexadecimal possui um peso 16 vezes maior do que a casa a sua direita

...	256	16	1

Considerando um número com N dígitos (ou casas) teremos capacidade de representar

B^N valores diferentes (onde B é a base de numeração).

Para 3 dígitos hexadecimais teremos:

16^3 = 4096 valores

Sistemas de Numeração

O maior valor a ser representado com N dígitos será:

$B^N - 1$ (onde B é a base de numeração).

Para 4 dígitos hexa, teremos:

$16^4 - 1 = 65535$

Considerando o maior símbolo possível

F	F	F	F	65535

Sistemas de Numeração

Base Hexadecimal:

Adição: Quando a soma em uma determinada casa excede o maior símbolo da base, devemos deixar o excedente e levar o peso da base para a casa mais a esquerda valendo 1.

Exemplo:

$$
\begin{array}{ccccc}
 & 1 & 1 & 1 & \\
 & E & B & C & A \\
+ & 0 & A & D & 9 \\
\hline
 & F & 6 & A & 3
\end{array}
$$

A+9=10+9 = 19 = 16+3

C+D+1=12+13+1=26

= 16+10 = 16+A

B+A+1=11+10+1=22

=16+6

Sistemas de Numeração

Base Hexadecimal:

Subtração: quando uma determinada casa necessita "pedir emprestado" a casa a sua esquerda fica com um a menos e a casa solicitante recebe o peso da base (16).

Exemplo:

		15			
	5	16	16		→ 16+5=21 ⇒ 21-B =21-11
	6	0	5	F	=10 = A
-	0	4	B	B	
	5	B	A	4	

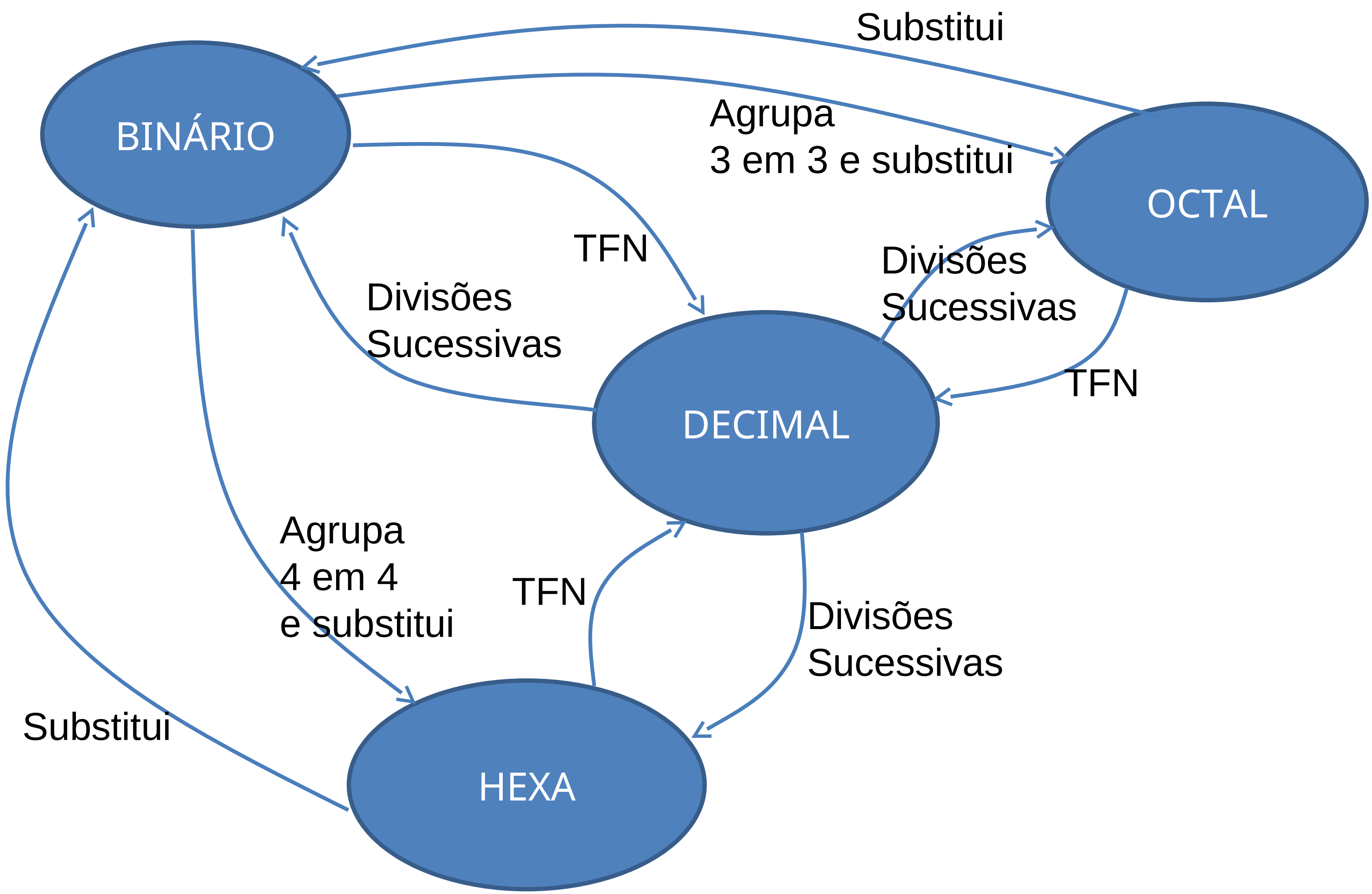
Conversão entre Bases

- **São 4 técnicas básicas**

Base Origem	Base Destino	Técnica
Base Qualquer →	Decimal	Teorema Fundamental da Numeração
Decimal →	Base Qualquer	Divisões Sucessivas
Octal/Hexa →	Binário	Substituir de 3 em 3 / 4 em 4
Binário →	Octal/Hexa	Agrupar de 3 em 3 / 4 em 4 e substituir

Conversão entre Bases

São 4 técnicas básicas



Códigos Numéricos

Codificação de Números Decimais

- **Código BCD (Binary Coding Decimal)**

O código BCD é um sistema de representação dos dígitos decimais desde 0 até 9 com um código binário de 4 bits. Esse código BCD usa o sistema de pesos posicionais 8421 do código binário puro.

Apesar de usar 4 bits existem apenas dez códigos válidos. Os números binários de 4 bits representando os números decimais desde 10 até 15 são inválidos no sistema BCD.

Ex: 238 = 001000111000

Decimal	BCD
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Códigos Numéricos

- **Código excesso 3**

Características
• valor binário + 3
• mantém faixa central de valores

Ex:
691 = 100111000100

Decimal	Excesso 3
0	0011
1	0100
2	0101
3	0110
4	0111
5	1000
6	1001
7	1010
8	1011
9	1100

Códigos Numéricos

- **Código Johnson**

Decimal	Johnson
0	00000
1	00001
2	00011
3	00111
4	01111
5	11111
6	11110
7	11100
8	11000
9	10000

Características

- variação de 1 bit de um código para o seguinte

Ex:
417 = 011110000111100

Códigos Numéricos

- **Códigos 2 em 5**

Características

- Grupo de códigos onde 2 entre 5 dígitos recebem o valor 1

- Cada posição tem um peso associado

- O zero tem codificação especial

Ex:
804 = 001010110001010

= 100101100001001

Código	2 em 5	
Decimal	**01236**	**74210**
0	01100	11000
1	11000	00011
2	10100	00101
3	10010	00110
4	01010	01001
5	00110	01010
6	10001	01100
7	01001	10001
8	00101	10010
9	00011	10100

Códigos Numéricos

- **Código GRAY**

Características
- Palavras adjacentes variam apenas 1 bit
- Cíclico
- Refletido
- Bit mais significativo é igual ao código binário natural

Conversão Gray → Binário??

Conversão Binário → Gray???

Decimal	GRAY
0	0000
1	0001
2	0011
3	0010
4	0110
5	0111
6	0101
7	0100
8	1101
9	1001
10	1111
11	1110
...	