SISTEMAS DE NUMERAÇÃO

Introdução à Ciência da Computação – ICC0001

Histórico

- Como surgiram os sistemas de numeração?
- Primeiro: As pessoas precisavam contar....
  - Dias, rebanho, árvores e tudo mais...
- Segundo: As pessoas precisavam anotar...
  - Utilizavam símbolos para representar as quantidades
- Exemplo:
  - Vamos anotar (identificar) os computadores desta sala usando símbolos
- Os romanos utilizavam diferentes símbolos para diferentes quantidades
  - A soma dos símbolos representava o número total
  - Não representavam o valor zero

I	V	X	L	C	D	M
1	5	10	50	100	500	1000

Histórico

- Sistema Romano utiliza algumas regras de repetição e ordem

  I – uma ocorrência
  II – duas ocorrências
  III – três ocorrências
  IV – quatro ocorrências
  V – cinco ocorrências
  VI – seis ocorrências

  - Símbolos de menor peso a esquerda representam descontos
  - Símbolos de menor peso a direita representam acréscimos

  CXLVII = ?
  MDXCIX = ?

- Os hindus (Índia) utilizavam dez símbolos (sistema decimal) [hindu-arábico]
  - 0 – Nenhuma ocorrência
  - 1 – Uma ocorrência
  - ...
  - 9 – Nove ocorrências

Introdução

- O sistema de numeração hindu-arábico difere-se do romano em um importante fator...

Introdução

- O sistema de numeração hindu-arábico difere-se do romano em um importante fator...

- No decimal a posição em que o número se encaixa da direita para a esquerda indica a grandeza deste número:
  - Decimal: 0 → zero ocorrência
  - Decimal: 1 → uma ocorrência
  - Decimal: 10 → dez ocorrências
  - Decimal: 100 → cem ocorrências

- Por isso ele é dito POSICIONAL
- O sistema romano, ao contrário, não é posicional...
  - Existe apenas a ‘regra de ordem’

Introdução

- Basicamente podemos criar sistemas de numeração posicionais com qualquer quantidade de símbolos maior que um.

- Vamos testar com o alfabeto de vogais:
  - A – zero ocorrência
  - E – uma ocorrência
  - I – duas ocorrências
  - O – três ocorrências
  - U – quatro ocorrências

- Cinco ocorrências? Dez ocorrências?

Sistemas de numeração

- Os mais comuns são:
  - Decimal (dez algarismos – 0 1 2 3 4 5 6 7 8 9)
  - Hexadecimal (16 símbolos – 0 1 2 3 4 5 6 7 8 9 A B C D E F)
  - Octal (oito algarismos – 0 1 2 3 4 5 6 7)
  - Binário (dois algarismos – 0 1)

Sistemas de numeração

- Sistema Decimal ou de base 10:
  - Composto de 10 algarismos ou símbolos;
  - Quando os algarismos expressam um número são chamados dígitos;
  - Tipo posicional: o valor do dígito depende da posição dentro do número;
  - Com N posições podemos representar 10^N números

Sistemas de numeração

- Sistema Decimal (cont.):
  - Exemplo: 1967_{10}

Sistemas de numeração

- Sistema Decimal (cont.):
  - Exemplo: 1967_{10}

1000		1 x 1000		1 x 10^3
900		9 x 100		9 x 10^2
60		6 x 10		6 x 10^1
+ 7		+7 x 1		+4 x 10^0
1967		1967		1967

Pesos expressos como potências de dez:

$1967_{10} = 1 \text{ x } 10^3 + 9 \text{ x } 10^2 + 6 \text{ x } 10^1 + 4 \text{ x } 10^0$

Sistemas de numeração

- Sistema Decimal (cont.):
  - Exemplo: $2745{,}214_{10}$

Sistemas de numeração

- Sistema Decimal (cont.):
  - Exemplo: $2745{,}214_{10}$

2000,000		2 x 1000		2×10^{3}
700,000		7 x 100		7×10^{2}
40,000		4 x 10		4×10^{1}
5,000		5 x 1		5×10^{0}
0,200		2 x 0,1		2×10^{-1}
0,010		1 x 0,01		1×10^{-2}
+ 0,004		+4 x 0,001		$+4 \times 10^{-3}$
2745,214		2745,214		2745,214

Pesos expressos como potências de dez:

$2745{,}214_{10} = 2 \times 10^{3} + 7 \times 10^{2} + 4 \times 10^{1} + 5 \times 10^{0} + 2 \times 10^{-1} + 1 \times 10^{-2} + 4 \times 10^{-3}$

Sistemas de numeração

- Sistema Binário ou de base 2:
  - Composto de 2 algarismos ou símbolos;
    - 0, 1;
    - Pode ser usado para caracterizar qualquer número;
    - O dígito é chamado de **bit** (**bi**nary digi**t**).

  - Também é do tipo posicional;

  - Com N posições podemos representar 2^N números

Sistemas de numeração

- Sistema Binário (cont.):
  - Contagem binária:
    - Restrita ao número de bits disponíveis – Ex.: 3 bits

Peso	$2^2=4$	$2^1=2$	$2^0=1$		Decimal
	0	0	0	000	0
	0	0	1	001	1
	0	1	0	010	2
	0	1	1	011	3
	1	0	0	100	4
	1	0	1	101	5
	1	1	0	110	6
	1	1	1	111	7

Sistemas de numeração

- Sistema Binário (cont.):
  - Representação

Exemplo: 1010_2

Pesos expressos como potências de dois:

$1010_2 = (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (0 \times 2^0)$

Decimal $= (1 \times 8) + (0 \times 4) + (1 \times 2) + (0 \times 1)$

$= 8 + 0 + 2 + 0$

$= 10_{10}$

Sistemas de numeração

- Sistema Binário (cont.):
  - Representação

Exemplo: $11011{,}11_2$

Pesos expressos como potências de dois:

$11011{,}11_2 = (1 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) + (1 \times 2^{-1}) + (1 \times 2^{-2})$

Decimal $= (1 \times 16) + (1 \times 8) + (0 \times 4) + (1 \times 2) + (1 \times 1) + (1 \times 0{,}5) + (1 \times 0{,}25)$

$= 16 + 8 + 0 + 2 + 1 + 0{,}5 + 0{,}25$

$= 27{,}75_{10}$

Sistemas de numeração

- Sistema Octal ou de base 8:
  - Composto de 8 algarismos ou símbolos;
    - 0, 1, 2, 3, 4, 5, 6, 7;
  - É do tipo posicional;
  - Com N posições podemos representar 8^N números

Sistemas de numeração

- Sistema Octal (cont.):
  - Contagem octal:

0	10		100	1000
1	11		101	1001
2	12		…	…
3	13		177	1477
4	14		200	1500
5	15	…	201	…
6	16		…	1777
7	17		777	2000

Sistemas de numeração

- Sistema Octal (cont.):
  - Representação

Exemplo: 2510_8

Pesos expressos como potências de oito:

$2510_8 = (2 \times 8^3) + (5 \times 8^2) + (1 \times 8^1) + (0 \times 8^0)$

Decimal $= 2 \times 512 + 5 \times 64 + 1 \times 8 + 0 \times 1$

$= 1024 + 320 + 8 + 0$

$= 1352_{10}$

Sistemas de numeração

- Sistema Octal (cont.):
  - Representação

Exemplo: $765{,}21_8$

Pesos expressos como potências de oito:

$765{,}21_8 = (7 \times 8^2) + (6 \times 8^1) + (5 \times 8^0) + (2 \times 8^{-1}) + (1 \times 8^{-2})$

Decimal $= 7 \times 64 + 6 \times 8 + 5 \times 1 + 2 \times 0{,}125 + 1 \times 0{,}015625$

$= 448 + 48 + 5 + 0{,}25 + 0{,}015625$

$= 501{,}265625_{10}$

Sistemas de numeração

- Sistema Hexadecimal ou de base 16:
  - Composto de 16 algarismos ou símbolos;
    - 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.
  - É do tipo posicional;
  - Com N posições podemos representar 16^N números

Sistemas de numeração

- Sistema Hexadecimal (cont.):
  - Contagem hexadecimal:

0	10		100
1	11		101
…	…		…
9	19		1FF
A	1A		200
B	1B	…	201
C	1C		…
D	1D		2FF
E	1E		300
F	1F		…

Sistemas de numeração

- Sistema Hexadecimal (cont.):
  - Representação

Exemplo: $1A5_{16}$

Pesos expressos como potências de dezesseis:

$1A5_{16} = (1 \times 16^2) + (A \times 16^1) + (5 \times 16^0)$

Decimal = $1 \times 256 + 10 \times 16 + 5 \times 1$

$= 256 + 160 + 5$

$= 421_{10}$

Sistemas de numeração

- Sistema Hexadecimal (cont.):
  - Representação

Exemplo: $C0,5F_{16}$

Pesos expressos como potências de dezesseis:

$C0,5F_{16} = (C \times 16^{1}) + (0 \times 16^{0}) + (5 \times 16^{-1}) + (F \times 16^{-2})$

Decimal = $12 \times 16 + 0 \times 1 + 5 \times 0,0625 + 15 \times 0,00390625$

$= 192 + 0 + 0,3125 + 0,05859375$

$= 192,37109375$

Sistemas de numeração

- Sistema de base B:
  - (Tipo posicional);
  - Composto de B algarismos ou símbolos;
  - Com N posições podemos representar B^N números;
  - Expresso em decimal como potências de B;

Sistemas de numeração

- Sistema de base B (cont.):
  - Representação

Exemplo: #$\$$&$_B$

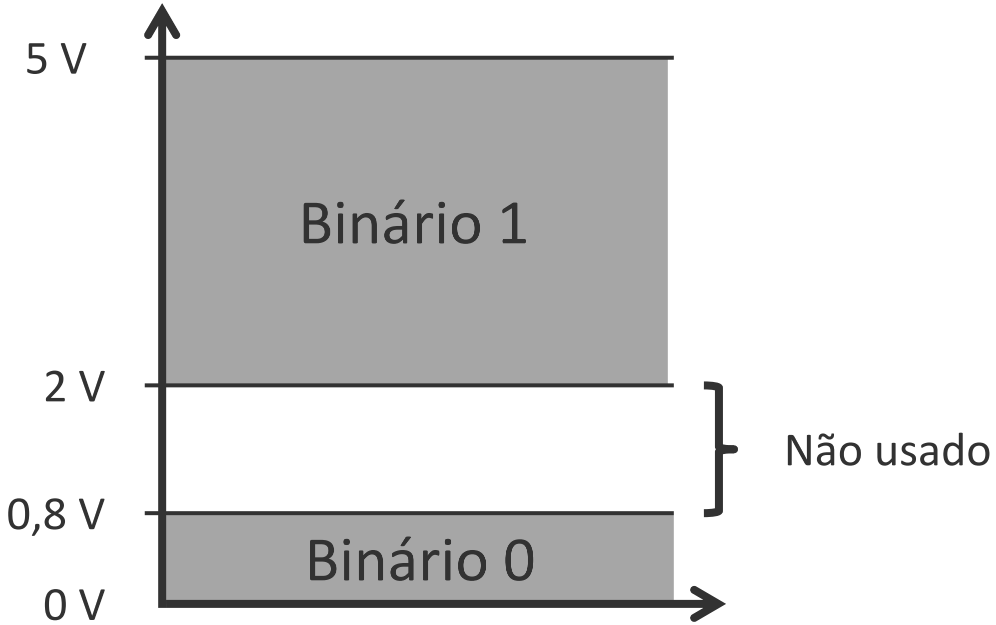
Pesos expressos como potências de “B”:

#$\$$&$_B$ = (# x B^2) + ($\$$ x B^1) + (& x B^0)

Sistemas de numeração

- Sistema decimal é de fácil compreensão humana
  - Influência da cultura
- Sistema decimal não é adequado para sistemas digitais:
  - Exigiria 10 níveis de tensão (ou corrente) distintos;
- Para sistemas digitais, o sistema binário foi considerado adequado por:
  - Simplicidade

    ligado / desligado
  - Necessita apenas dois níveis de tensão



Conversão de base

- Para cada sistema numérico apresentado vimos como obter o equivalente decimal;

- Agora vamos ver outras conversões possíveis.

Conversão de base

Decimal para binário

- A conversão do sistema Decimal para o binário é realizado por sucessivas divisões por 2, ou seja, o número em decimal é dividido sucessivamente por 2 até que o quociente seja igual a 0

- O resto da última divisão representa o dígito mais à esquerda do número binário, o resto da próxima divisão o próximo dígito, e assim por diante

Conversão de base

Decimal para binário

$$
\begin{array}{lllll}
13 & \underline{|2} & & & \\
1 & 6 & \underline{|2} & & \\
 & 0 & 3 & \underline{|2} & \\
 & & 1 & 1 & \underline{|2} \\
 & & & 1 & 0
\end{array}
\quad \Rightarrow 13_{10} = 1101_{2}
$$

Conversão de base

Decimal para binário

1492 | 2
0 746 | 2
0 373 | 2
1 186 | 2
0 93 | 2
1 46 | 2
0 23 | 2
1 11 | 2
1 5 | 2
1 2 | 2
0 1 | 2
1 0

$1492_{10} = 10111010100_2$

Conversão de base

- Outra opção de representação das divisões sucessivas:

25_{10} para binário		
25/2 = 12	resto	1
12/2 = 6	resto	0
6/2 = 3	resto	0
3/2 = 1	resto	1
1/2 = 0	resto	1

11001_2

37_{10} para binário		
37/2 = 18,5	0,5 x 2	1
18/2 = 9,0	0 x 2	0
9/2 = 4,5	0,5 x 2	1
4/2 = 2,0	0 x 2	0
2/2 = 1,0	0 X 2	0
1/2 = 0,5	0,5 x 2	1

100101_2

Conversão de base

Decimal para octal

$135_{10} = ?_8$

135 |8

7 16 |8

0 2 |8

2 0 => $135_{10} = 207_8$

Conversão de base

Decimal para hexadecimal

$1325_{10} = ?_{16}$

1325 |16

13 [D] 82 |16

2 5 |16

5 0 => $1325_{10} = 52D_{16}$

- Lembrando que no sistema hexadecimal:
  - 0, (...), 9 [9], A [10], B[11], C [12], D [13], E [14], F [15]

Conversão de base

- Regra Geral:
- Decimal para base B

  - Divisão por B até que o quociente seja 0

  - Obtem-se o número equivalente na base B tomando os restos da última divisão até a primeira, e colocando-os lado a lado da esquerda para a direita.

$135_{10} = ?_B$

$$
\begin{array}{lllllll}
135 & \underline{|\,B} & & & & & \\
R_1 & \mathbf{Q_1} & \underline{|\,B} & & & & \\
 & R_2 & \mathbf{Q_2} & \underline{|\,B} & & & \\
 & & \mathbf{\ldots} & \mathbf{Q_{n-1}} & \underline{|\,B} & & \\
 & & & R_n & \mathbf{0} & \Rightarrow & 135_{10} = (R_n \ldots R_2R_1)_B
\end{array}
$$

Conversão de base

- Exemplo Regra Geral:
- Decimal para base 5

$$121_{10} = ?_5$$

121	5			
1	**24**	5		
	4	**4**	5	
		4	**0**	

=> $121_{10} = 441_5$

Conversão de base

- Base A para base B

Conversão indireta ou em duas etapas :

1 – Base A para decimal - polinômio

2 – Número decimal para base B – divisões sucessivas

Conversão de base

- Exemplo de conversão da Base A para a base B
  - Base 4: dígitos 0, 1, 2, 3; Base 7: dígitos 0, 1, 2, 3, 4, 5, 6;

  - **Converter 221_4 para base 7**
    - Passo 1: converter 221_4 para base decimal (polinômio)

$$2 \times 4^2 + 2 \times 4^1 + 1 \times 4^0 = (2 \times 16) + (2 \times 4) + (1 \times 1) = 32 + 8 + 1 = 41_{10}$$

    - Passo 2: converter o valor encontrado (41_{10}) para a base 7 (divisões sucessivas)

41 | 7
6 5 | 7
5 0

$$\Rightarrow 221_4 = 41_{10} = 56_7$$

Conversão de base com simplificações

- Binário para octal/octal para binário:
  - Por serem bases múltiplas, existe uma associação fácil e rápida;
  - Digitos octais são convertidos nos equivalentes binários de três em três bits e vice-versa:

Dígito Octal	0	1	2	3	4	5	6	7
Equivalente binário	000	001	010	011	100	101	110	111

$472_8 = ?_2$

4	7	2_8
⇓	⇓	⇓
100	111	010_2

$101100011001_2 = ?_8$

101	100	011	001_2
⇓	⇓	⇓	⇓
5	4	3	1_8

Conversão de base com simplificações

- Binário para hexadecimal / hexadecimal para binário:
  - Por serem bases múltiplas, existe uma associação fácil e rápida;
  - Digitos hexadecimais são convertidos nos equivalentes binários de quatro em quatro bits e vice-versa:

Dígito hexa	0	1	2	3	4	5	6	7
Equivalente binário	0000	0001	0010	0011	0100	0101	0110	0111
Dígito hexa	**8**	**9**	**A (10)**	**B (11)**	**C (12)**	**D (13)**	**E (14)**	**F (15)**
Equivalente binário	1000	1001	1010	1011	1100	1101	1110	1111

$9F2_{16} = ?_2$

9	F	2_{16}
⇩	⇩	⇩
1001	1111	0010_2

$(0011\ 1010\ 0110)_2 = ?_{16}$

0011	1010	0110_2
⇩	⇩	⇩
3	A	6_{16}

Conversão de base com simplificações

- Octal para hexadecimal / hexadecimal para octal:
  - Conversão indireta: converte-se primeiro para binário
    - X_8 => Y_2 => Z_{16}
    - X_{16} => Y_2 => Z_8

$9F2_{16} = ?_8$

9	F	2_{16}
1001	1111	0010_2

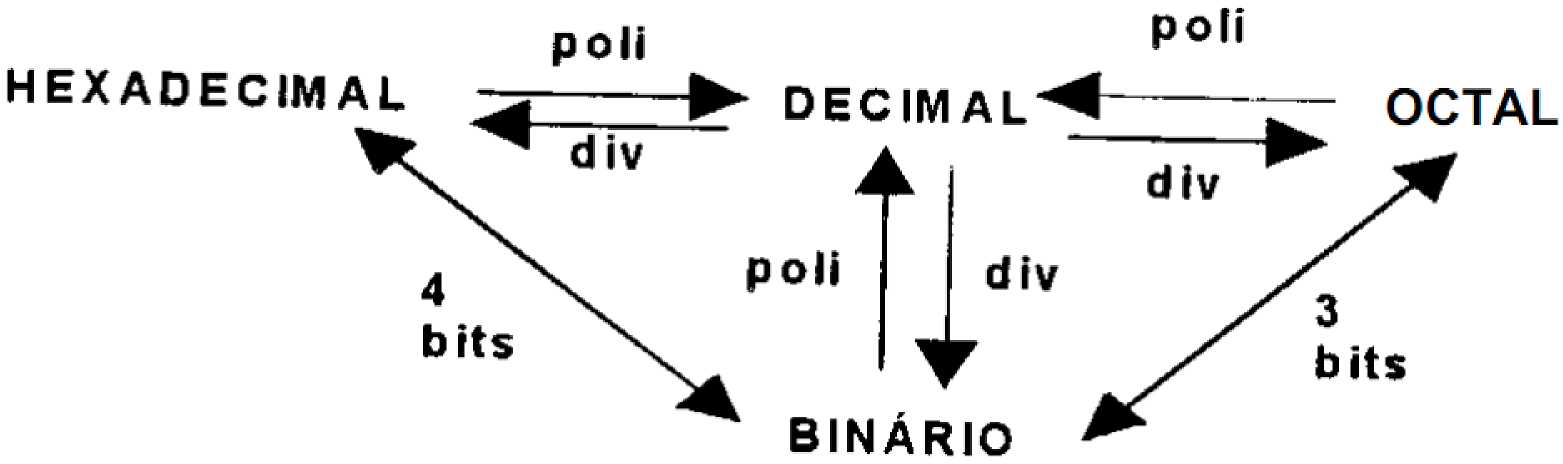
4	7	6	2_8

$1646_8 = ?_{16}$

1	6	4	6_8
001	110	100	110_2

3	A	6_{16}

Resumo



$$\text{Número} = a_n b^n + a_{n-1} b^{n-1} + a_{n-2} b^{n-2} + \ldots + a_0 b^0$$

a_n = algarismo, b = base do número

n = quantidade de algarismo – 1

Importantíssimo

- Lembrar de colocar a base do número:

$$1001_2 \neq 1001_{10} \neq 1001_8 \neq 1001_7$$

Assume-se que um número sem base está implicitamente na base 10. Mas, coloquem a base em todos os casos.

Números não inteiros

- Como representamos números não inteiros?
- Uso da vírgula
- Para cada casa à direita depois da vírgula nós diminuímos a grandeza do número.

Números não inteiros

- No sistema decimal:

2000,000		2 x 1000		2 x 10^{3}
700,000		7 x 100		7 x 10^{2}
40,000		4 x 10		4 x 10^{1}
5,000		5 x 1		5 x 10^{0}
0,200		2 x 0,1		2 x 10^{-1}
0,010		1 x 0,01		1 x 10^{-2}
+ 0,004		+4 x 0,001		+4 x 10^{-3}
2745,214		2745,214		2745,214

- Cada casa a direita diminui a ordem de grandeza em 10 vezes

Números não inteiros

- Nos demais sistemas esta redução também é verdadeira!

- Quanto vale:

$$111{,}010_2 = 1x2^2 + 1x2^1 + 1x2^0 + 0x2^{-1} + 1x2^{-2} + 0x2^{-3} = 7{,}25_{10}$$

- Então é **assim** que fazemos a conversão de forma direta?
  - Sim

Números não inteiros

Conversão Inteira:

- Binário para decimal:
  - Potência de base 10
- Decimal para binário:
  - Divisões sucessivas por 2

- Para converter $7{,}25_{10}$ para binário basta divisões sucessivas por 2?
  - Não!

Números não inteiros

Conversão de números não inteiros

- Binário para decimal:
  - Potência de base 10
- Decimal para binário:
  - **Parte inteira →** Divisões sucessivas por 2 até quociente ser zero
  - **Parte fracionária →** Multiplicar por dois até parte fracionária ser zero

    (cada resultado da multiplicação é um digito binário)
- Converter: $7{,}25_{10} = X_2$
  - 7,25

7 / 2 = 3 (resto 1)
3 / 2 = 1 (resto 1)
1 / 2 = 0 (resto 1)
Parte inteira = 111_2

0,25 * 2 = **0**,5
0,5 * 2 = **1**,0
Parte fracionária = ,01
$7{,}25_{10} = 111{,}01_2$

Números não inteiros

Converter: $4{,}3_{10} = X_2$

$4 / 2 = 2$ (resto 0)

$2 / 2 = 1$ (resto 0)

$1 / 2 = 0$ (resto 1)

Parte inteira = 100_2

0,3 * 2 = **0**,6

0,6 * 2 = **1**,2

0,2 * 2 = **0**,4

0,4 * 2 = **0**,8

0,8 * 2 = **1**,6

0,6 * 2 = **1**,2

(....)

Se multiplicação se repetir → é uma dízima periódica

Parte fracionária ≈ ,0100110011...

$4{,}3_{10} = 100{,}0100110011..._2$

Números não inteiros

A regra da multiplicação fracionária se aplica para a conversão de decimal para outras bases também:

Converter: $14{,}808_{10} = X_5$

$14 / 5 = 2$ (resto 4)

$2 / 5 = 0$ (resto 2)

Parte inteira = 24_5

$0{,}808 * 5 = \mathbf{4}{,}04$

$0{,}04 * 5 = \mathbf{0}{,}2$

$0{,}2 * 5 = \mathbf{1}{,}0$

Parte fracionária = $,401_5$

$14{,}808_{10} = 24{,}401_5$

SISTEMAS DE NUMERAÇÃO

Introdução à Ciência da Computação – ICC0001