Sistema de Numeração

Introdução a Informática

Vinícius Pádua

Sistema de Numeração

- Métodos científicos para representar os números

- Tipos
  - Notação não posicional ou Posicional
    - Difere se o algarismo tem valor fixo ou não

Notação Não Posicional

- Algarismos tem um valor fixo
- Exemplo?
  - Algarismos Romanos
    - Utilizado principalmente informar data
    - N = (I, V, X, L, C, D, M)

      (1, 5, 10, 50, 100, 500, 1000)
    - Não tem o 0
    - Regra
      - Algarismo a direita de um maior é adicionado a esse
      - Algarismo a esquerda de outro maior é subtraído do maior

Notação Não Posicional

- Algarismos Romanos
  - I V = 4
  - V I = 6
  - Exercícios
    - XIV
    - XL
    - LX
    - MCCXXXI
    - MCMXCIX

Notação Posicional

- O valor de cada algarismo modifica de acordo com a posição
  - Números Decimais: 2622

- Base de numeração
  - Indica a quantidade de algarismos disponíveis
  - Necessidade de escrever números grandes com poucos algarismos
  - Agrupamento de valores
  - Diminuir quantidade de símbolos
  - Exemplo: 100001_2 21_8 17_{10} 11_{16}

Base

- Base Decimal – 10 algarismos
- Antigas
  - Base Sexagenal – 60 algarismos
  - Base Duodecimal – 12 algarismos
- Mais comuns
  - Base Binária – 2 algarismos
  - Base Octal – 8 algarismos
  - Base Hexadecimal – 16 algarismos

Base

- Base Decimal – 10 algarismos
  - D = (0, 1, 2, 3, 4, 5, 6, 7, 8, 9)
- Base Binária – 2 algarismos
  - D = (0, 1)
- Base Octal – 8 algarismos
  - D = (0, 1, 2, 3, 4, 5, 6, 7)
- Base Hexadecimal – 16 algarismos
  - D = (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F)

Base

Base 2	Base 8	Base 10	Base 16
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

Conversão de base B para 10

- N = $d_{n-1} * b^{n-1} + d_{n-2} * b^{n-2} + \ldots + d_1 * b^1 + d_0 * b^0$

d = indica os algarismo

n-1, n-2, ...,1, 0 = posição do algarismo

b = Base

Exemplos:

101_2 em base 10

$$N = 1 * 2^2 + 0 * 2^1 + 1 * 2^0$$
$$= 4 + 0 + 1 = 5$$

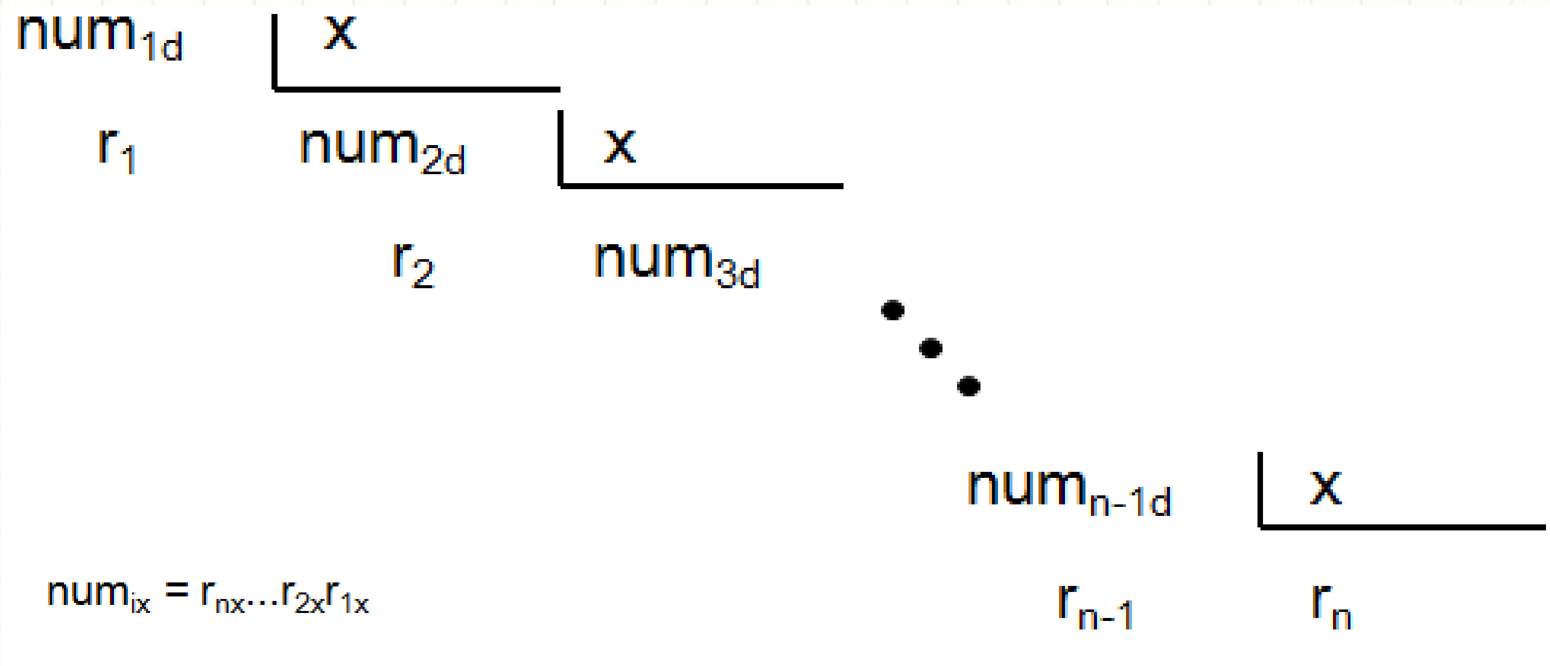
20_8 em base 10

$$N = 2 * 8^1 + 0 * 8^0$$
$$= 16 + 0 = 16$$

Conversão de base B para 10

- Exercícios, converter para a base 10:
  - 1100_2
  - 0111_2
  - 135_8
  - $ABCD_{16}$
  - $A8B2_{16}$
  - 112_3

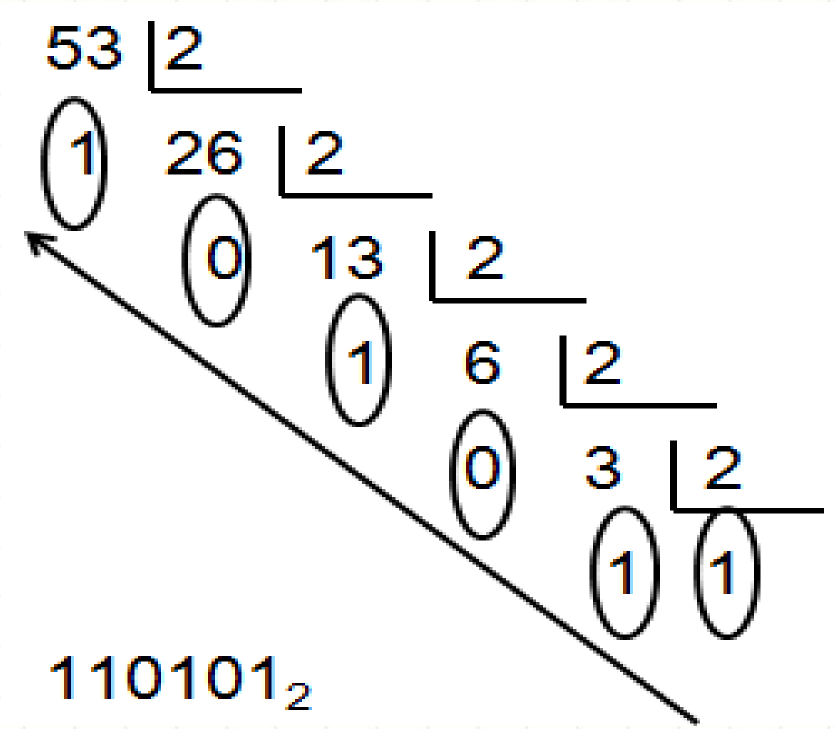
Conversão de base 10 para B



X = base

Conversão de base 10 para B

- 53_{10} para binário



Momento de Parar: quando o quociente é menor do que o valor da base
Neste caso, o valor da base é "2"

Conversão de base 10 para B

- Exercícios, converter para a base 10:
  - 12_{10} para binário
  - 7_{10} para binário
  - 93_{10} para octaldecimal
  - 43981_{10} para hexadecimal
  - 43186_{10} para hexadecimal
  - 14_{10} para base 3

Outras Conversões

- 111_2 para octal

- 111010111_2 para octal 727_8
- 001010011111_2 para octal 1237_8

Achou trabalhoso ?

Outras Conversões

- Toda base potência de 2 a conversão é facilitada

- Base 2 para 8 (2^3)
- 111010111_2 para octal

$(111)_2\ (010)_2\ (111)_2$

$7 \quad 2 \quad 7 \quad = (727)_8$

- 111010111_2 para hexadecimal (2^4)

$(1)_2\ (1101)_2\ (0111)_2$

$1 \quad D \quad 7 \quad = (1D7)_8$

Outras Conversões

- Exercício
  - 001010011111_2 para octal
  - 001010011111_2 para hexa
  - 327_8 para binário
  - 673_8 para binário
  - $F50_{16}$ para binário

  - 3174_8 para hexadecimal
  - 254_8 para hexadecimal
  - $2E7A_{16}$ para octal

Operações com Binários

- Soma
- Subtração
- Multiplicação
- Divisão

Operações com Binários

- Soma de Binários
  - Semelhante a decimal
  - 0 + 0 = 0   0 + 1 = 1   1 + 0 = 1   1 + 1 = 0 (“vai 1”)

101101	100101	1100011
+101111	+1010111	+1011011

Operações com Binários

- Subtração de Binários
  - Semelhante a decimal
  - 0 - 0 = 0   1 - 0 = 1   1 - 1 = 0   0 – 1 = 1 (“empréstimo de 2”)

```
  100    101101     101100101
-   1   -100111    -100111011
```

Operações com Binários

- Multiplicação de Binários
  - Semelhante a decimal
  - 0 x 0 = 0 0 x 1 = 0 1 x 0 = 0 1 x 1 = 1

110	10101	10010
x 101	x 1101	x 100

Operações com Binários

- Divisão de Binários
  - Semelhante a decimal

101010 |<u>110</u>

100100<u>|100</u>

11000 |<u>11</u>

100011<u>|101</u>

100101<u>|100</u>

Números Fracionados

- Quanto será $0{,}1011101_2$ em decimal?

- N = $d_{-1} * b^{-1} + d_{-2} * b^{-2} + \ldots + d_{-m} * b^{-m}$

- $0{,}1011101_2$

$$1*2^{-1} + 0*2^{-2} + 1*2^{-3} + 1*2^{-4} + 1*2^{-5} + 0*2^{-6} + 1*2^{-7}$$

$$0{,}5+0{,}125+0{,}0625+0{,}03125+0{,}0078125=0{,}72655625_{10}$$

Números Fracionados

- Quanto será $0,37_{10}$ em binário com 4 bits?

- $0,37 \times 2 = 0,74$ 0
- $0,74 \times 2 = 1,48$ 1
- $0,48 \times 2 = 0,96$ 0
- $0,96 \times 2 = 1,92$ 1

- $0,37_{10} = 0,0101_2$

Representação dos Números

- Como computador reconhece .... ?
  - Números Negativos
  - Virgula das casas decimais

- Quantidade de algarismos que o processador reconhece
  - 2 Bits – 2 Algarismos – 4 números
  - 4 Bits – 4 Algarismos – 16 números
  - 16 Bits
  - 32 Bits
  - 64 Bits

Se forçar um 5 número?

Overflow
underflow

Números Negativos

- Como representar números negativos
  - Sinal-magnitude
  - Complemento de 1
  - Complemento de 2

Sinal-Magnitude

- Utiliza o bit mais significativo para indicar o sinal
  - 0 = Positivo e 1 = negativo

SINAL	MAGNITUDE

Binário	com Sinal	sem Sinal
00000000	+0	0
00000001	1	1
...	...	...
01111111	127	127
10000000	–0	128
10000001	–1	129
...	...	...
11111111	–127	255

Sinal-Magnitude

- Qual limite de números posso armazenar em um 6 bits?

- __ __ __ __ __ __
  -
  - $-(2_{n-1} -1)$ até $+(2_{n-1} -1)$

Sinal-Magnitude

- Represente em binários os números decimais abaixo em um computador de 10 bits
  - +54
  - -54
  - +123
  - -456
- Calcular em um computador de 6 bits
  - 001101 + 001100
  - 110001 + 101001
  - 110101 + 001010
  - 001010 + 110101

Complemento de 1

- Números negativos são o inverso dos positivos
- Possui dois zeros (0 e -0)

Decimal	Sinal e magnitude	Complemento a 1
+0	00000	00000
+1	00001	00001
+2	00010	00010
+3	00011	00011
+4	00100	00100
+5	00101	00101
+6	00110	00110

Decimal	Sinal e magnitude	Complemento a 1
−16	—	—
−15	11111	10000
−14	11110	10001
−13	11101	10010
−12	11100	10011
−11	11011	10100
−10	11010	10101

Complemento de 2

- Números negativos são o inverso dos positivos, depois soma-se 1

Decimal	Sinal e magnitude	Complemento a 2
+0	00000	00000
+1	00001	00001
+2	00010	00010
+3	00011	00011
+4	00100	00100
+5	00101	00101
+6	00110	00110

Decimal	Sinal e magnitude	Complemento a 2
−16	—	10000
−15	11111	10001
−14	11110	10010
−13	11101	10011
−12	11100	10100
−11	11011	10101
−10	11010	10110

Complemento

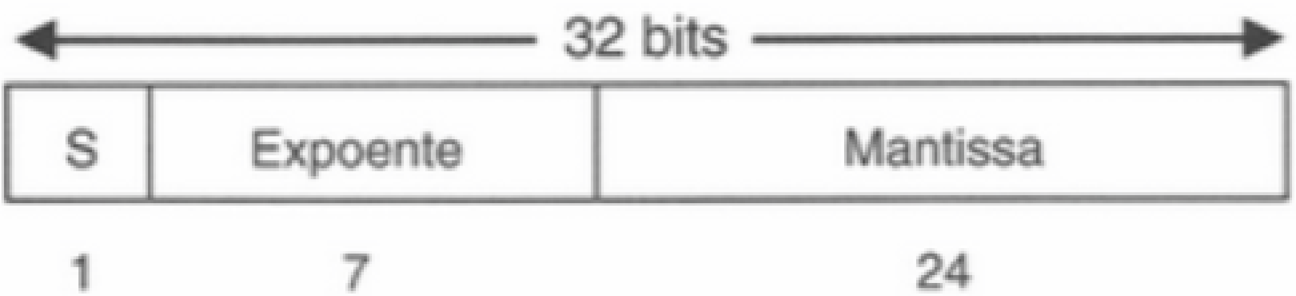
- Represente os números abaixo em C1 e C2
  - -16
  - 15
  - -14
  - -11
- Represente os números abaixo em decimal
  - 00011 C2
  - 10101 C2
  - 10100 C1

Representação de Números

- Como o computador converte e "entende”o decimal +407,375?
  - Notação científica
  - Fatores importantes
    - Total de bits
    - Tamanho da mantissa

Representação de Números

$N = \pm M \times B^{\pm E}$



S - Sinal de número

E - Expoente: 1 bit p/ sinal e 6 bits p/ magnitude

M - Mantissa, normalizada

B - Base de exponenciação = 2

Representação de Números

- $+407{,}375_{10}$
- Passos
  - Converter para binário a parte inteira e depois a fracionária
  - Representar em notação científica (expoente e mantissa)
  - Converter expoente par binário
  - Concatenar Sinal + Expoente em binário + mantissa

Representação de Números

- $+407{,}375_{10}$
  1. $407_{10} = 110010111_2$
  2. $0{,}375_{10} = 011_2$
  3. $407{,}375_{10} = 110010111{,}011_2$
  4. $110010111{,}011_2 = 0{,}110010111011 * 2^9$
  5. $9_{10} = 1001_2$
  6. Sinal + Expoente + Mantissa
  7. 0 0001001 000000000000110010111011
  8. $0\ 9\ CBB000_{16}$

DÚVIDAS