



## Introdução à Computação

Jordana Sarmenghi Salamon

`jssalamon@inf.ufes.br`

[jordanasalamon@gmail.com](mailto:jordanasalamon@gmail.com)

<http://inf.ufes.br/~jssalamon>

Departamento de Informática

Universidade Federal do

Espírito Santo

# Agenda

---

- Sistemas de Numeração
- Conversão entre sistemas de numeração

# Sistemas de Numeração

---

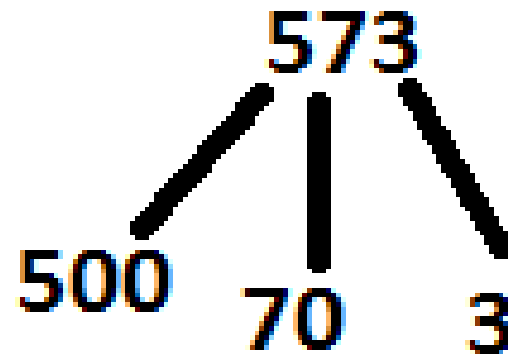
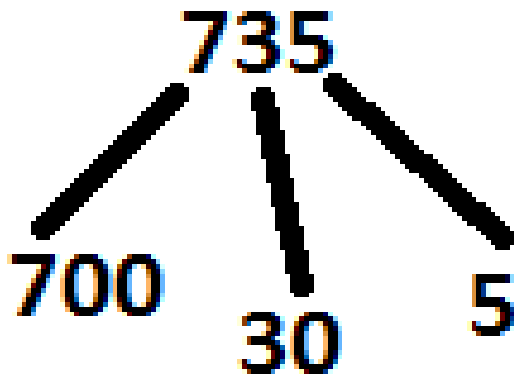
- Um sistema de numeração (ou sistema numeral), é **um sistema em que um conjunto de números são representados por numerais de uma forma consistente.**
  - Sistema decimal
  - Sistema binário
  - Sistema octal
  - Sistema hexadecimal
  - Sistema de numeração duodecimal
  - Sistema de numeração romano

- Em condições ideais, um sistema de numeração deve: **representar uma grande quantidade de números úteis** (ex.: todos os números inteiros, ou todos os números reais); **dar a cada número representado uma única descrição** (ou pelo menos uma representação padrão); e **refletir as estruturas algébricas e aritméticas dos números**.
- **Tipos de representação numérica:** *notação posicional e notação não posicional*.

- **Notação Posicional:**

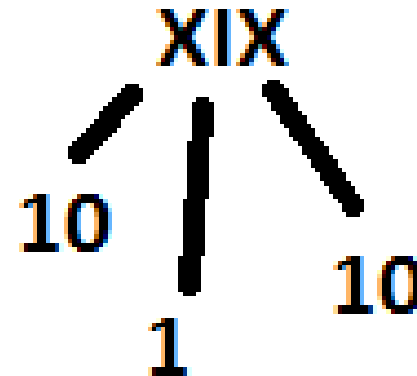
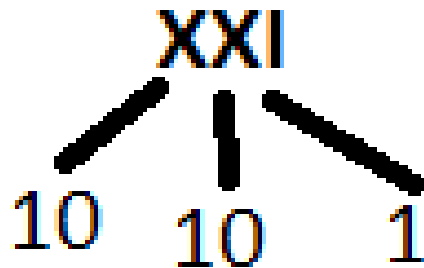
- Valor atribuído a um símbolo **dependente da posição** em que ele se encontra no conjunto de símbolos que representa uma quantidade.

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



- **Notação não Posicional:**

- Valor atribuído a um símbolo e inalterável, **independente** da posição em que se encontre no de símbolos que representam uma quantidade.

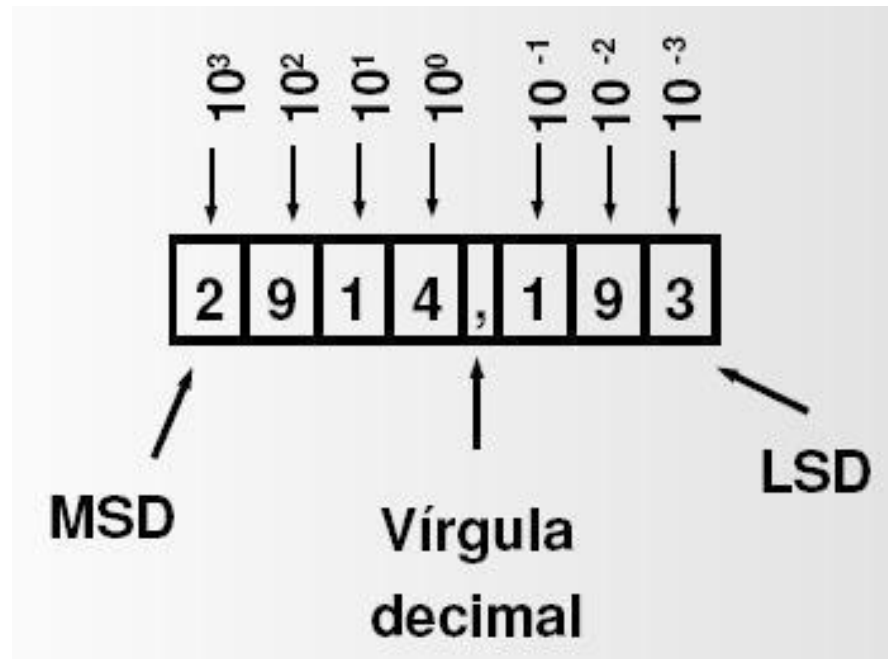


# Sistemas de Numeração

- Exemplo Notação Posicional:

- 2914,193

- $2 \cdot 10^3 + 9 \cdot 10^2 + 1 \cdot 10^1 + 4 \cdot 10^0 + 1 \cdot 10^{-1} + 9 \cdot 10^{-2} + 3 \cdot 10^{-3}$



# Sistemas de Numeração

- Base: Número mínimo que pode ser representado com um algarismo.

<b>Sistema</b>	<b>Base</b>
Binário	2
Octal	8
Decimal	10
Hexadecimal	16



- **Generalização para base qualquer**
- Seja “b” a base de representação de um número e A, B, C, D, E,... os símbolos dos algarismos deste sistema, então o número ....
- EDCBA na base “b”, escrito convencionalmente como
  - $EDCBA_b$
- representa a grandeza  $E*b^4 + D*b^3 + C*b^2 + B*b^1 + A*b^0$

- **Generalização para base qualquer**
- Em um sistema numérico posicional de base  $r$ , um número  $D$  tem seu valor dado por:

$$D = \sum_{i=-n}^{p-1} d_i r^i$$

$$d_{p-1} d_{p-2} \dots d_1 d_0, d_{-1} \dots d_{-n}$$

$r$  : base do sistema

$p$  : número de dígitos à esquerda da vírgula

$n$  : número de dígitos à direita da vírgula

## • Exemplos

- $227_{10} = 2 \times 100 + 2 \times 10 + 7$
- $227_8 = 2 \times 64 + 2 \times 8 + 7 = 128 + 16 + 7 = 151_{(10)}$
- $227_{16} = 2 \times 256 + 2 \times 16 + 7 = 512 + 32 + 7 = 551_{(10)}$
  
- E como seria em outras bases?
- $222_5$
- $111_2$
- $227_{12}$

# Sistemas de Numeração

---

- Para o computador, tudo é representado através de **números**.
- **Computador Digital**: Normalmente a informação a ser processada é de forma numérica ou texto, codificada internamente através de um código numérico.
- Em sistemas digitais o sistema de numeração **binário** é o mais importante, já fora do mundo digital o sistema **decimal** é o mais utilizado;
- Para a representação de números binários grandes utilizamos os sistemas de numeração **octal** e **hexadecimal**.

# Sistemas de Numeração

- **Necessidade de conversão:** por exemplo, em uma calculadora comum digita-se os valores em decimal e este valor é convertido em binário para o processamento;

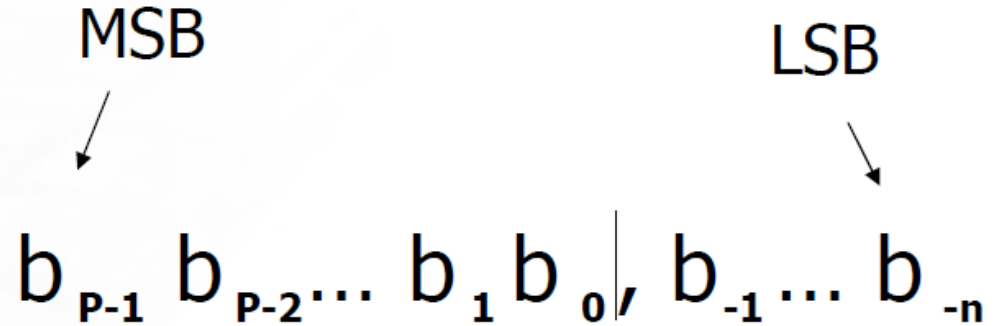


$$\begin{aligned}10 \times 5 &= \\1010 \times 101 &= \\110010 &= 50\end{aligned}$$

- **Como os computadores representam as informações utilizando apenas dois estados possíveis - eles são totalmente adequados para números binários.**
- **Numero binário no computador: bit [de “Binary digIT”]**
  - A unidade de informação.
  - Uma quantidade computacional que pode tomar um de dois valores, tais como verdadeiro e falso ou 1 e 0, respectivamente.

- Sistema Binário

$$B = \sum_{i=-n}^{p-1} d_i 2^i$$



bit  $\rightarrow b_i = \{0, 1\}$

- MSB: most significant digit (dígito mais significativo)
- LSB: least significant digit (dígito menos significativo)

- **Sistema Octal**

- Sistema de base 8;
- Contém 8 algarismos: 0, 1, 2, 3, 4, 5, 6 e 7;
- É utilizado por ser um sistema que tem relação direta com o sistema binário.
- Os valores posicionais são:

$$8^4 - 8^3 - 8^2 - 8^1 - 8^0 - \textit{vírgula} - 8^{-1} - 8^{-2} - 8^{-3}$$



- **Sistema Hexadecimal**

- Do hexa=6 e deci=10, sistema numérico de base 16;
- Este sistema possui 16 símbolos distintos em sua contagem;
- Além dos 10 dígitos (0 a 9), utiliza as letras A, B, C, D, E e F que fazem o papel das grandezas 10,11,12,13,14,15;
- Usamos as letras maiúsculas pela necessidade de termos que representar cada uma destas grandezas com um único algarismo.

# Sistemas de Numeração

- **Resumo dos Sistemas**

<b>Sistema</b>	<b>Base</b>	<b>Algarismos</b>
Binário	2	0 e 1
Octal	8	0,1,2,3,4,5,6,7
Decimal	10	0,1,2,3,4,5,6,7,8,9
Hexadecimal	16	0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

# Sistemas de Numeração

<b>Binário</b>	<b>Octal</b>	<b>Decimal</b>	<b>Hexadecimal</b>
<b>0000</b>	<b>0</b>	<b>0</b>	<b>0</b>
<b>0001</b>	<b>1</b>	<b>1</b>	<b>1</b>
<b>0010</b>	<b>2</b>	<b>2</b>	<b>2</b>
<b>0011</b>	<b>3</b>	<b>3</b>	<b>3</b>
<b>0100</b>	<b>4</b>	<b>4</b>	<b>4</b>
<b>0101</b>	<b>5</b>	<b>5</b>	<b>5</b>
<b>0110</b>	<b>6</b>	<b>6</b>	<b>6</b>
<b>0111</b>	<b>7</b>	<b>7</b>	<b>7</b>
<b>1000</b>	<b>10</b>	<b>8</b>	<b>8</b>
<b>1001</b>	<b>11</b>	<b>9</b>	<b>9</b>
<b>1010</b>	<b>12</b>	<b>10</b>	<b>A</b>
<b>1011</b>	<b>13</b>	<b>11</b>	<b>B</b>
<b>1100</b>	<b>14</b>	<b>12</b>	<b>C</b>
<b>1101</b>	<b>15</b>	<b>13</b>	<b>D</b>
<b>1110</b>	<b>16</b>	<b>14</b>	<b>E</b>
<b>1111</b>	<b>17</b>	<b>15</b>	<b>F</b>

- O maior valor que pode ser exibido em uma base  $b$  dada uma quantidade  $n$  de dígitos é:
- $2^n - 1$
- Por exemplo, o maior número que pode ser exibido na base 2 dada uma quantidade 4 de dígitos é:
- $2^4 - 1 = 16 - 1 = 15 = 1111_{(2)}$

- **Conversão Binário -> Decimal**

- Devemos considerar os valores posicionais na base 2 e fazer a soma das potências dos bits em “1”:

$$11011_{(2)} = (1 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (1 \times 2^0)$$

$$11011_{(2)} = 27_{(10)}$$

- **Conversão Binário -> Decimal**
- Exercícios
- Converta os números abaixo de binário para decimal:
- $11010001_2$
- $11001_2$
- $11110001_2$
- $110110_2$
- $1010110_2$

- **Conversão Binário -> Decimal**
- Exercícios
- Converta os números abaixo de binário para decimal:
- $11010001_2 = 209_{10}$
- $11001_2 = 25_{10}$
- $11110001_2 = 241_{10}$
- $110110_2 = 54_{10}$
- $1010110_2 = 86_{10}$

- **Conversão Octal -> Decimal**

- Assim como fizemos no sistema binário também utilizamos os valores posicionais:

- Ex1:  $372_{(8)} = (3 \times 8^2) + (7 \times 8^1) + (2 \times 8^0)$

$$372_{(8)} = 192 + 56 + 2$$

$$372_{(8)} = 250_{(10)}$$

- Ex2:  $24,6_{(8)} = (2 \times 8^1) + (4 \times 8^0) + (6 \times 8^{-1})$

$$24,6_{(8)} = 16 + 4 + 0,75$$

$$24,6_{(8)} = 20,75_{(10)}$$



# Conversão entre Sistemas

---

- **Conversão Octal -> Decimal**
- Exercícios
- Converta os números abaixo de octal para decimal:
- $160_8$
- $310_8$
- $624_8$
- $210_8$
- $536_8$

# Conversão entre Sistemas

---

- **Conversão Octal -> Decimal**
- Exercícios
- Converta os números abaixo de octal para decimal:
- $160_8 = 112_{10}$
- $310_8 = 200_{10}$
- $624_8 = 404_{10}$
- $210_8 = 136_{10}$
- $536_8 = 350_{10}$

# Conversão entre Sistemas

- **Conversão Hexadecimal -> Decimal**

- Assim como fizemos no sistema binário também utilizamos os valores posicionais:

- Ex1:  $356_{(16)} = (3 \times 16^2) + (5 \times 16^1) + (6 \times 16^0)$

$$356_{(16)} = 768 + 80 + 6$$

$$356_{(16)} = 854_{(10)}$$

- Ex2:  $2AF_{(16)} = (2 \times 16^2) + (10 \times 16^1) + (15 \times 16^0)$

$$2AF_{(16)} = 512 + 160 + 15$$

$$2AF_{(16)} = 687_{(10)}$$

# Conversão entre Sistemas

---

- **Conversão Hexadecimal -> Decimal**
- Exercícios
- Converta os números abaixo de hexadecimal para decimal:
  - $B4_{16}$
  - $104_{16}$
  - $D9_{16}$
  - $140_{16}$
  - $32_{16}$

# Conversão entre Sistemas

---

- **Conversão Hexadecimal -> Decimal**
- Exercícios
- Converta os números abaixo de hexadecimal para decimal:
  - $B4_{16} = 180_{10}$
  - $104_{16} = 260_{10}$
  - $D9_{16} = 217_{10}$
  - $140_{16} = 320_{10}$
  - $32_{16} = 50_{10}$

- **Conversão Base qualquer (r) -> Decimal**
- Para converter de binário, octal ou hexadecimal para decimal, use o método da soma dos pesos de cada dígito (valor posicional):

$$D = \sum_{i=-n}^{p-1} d_i r^i$$

# Conversão entre Sistemas

---

- **Conversão Decimal -> Binário**
- Há duas formas de converter o número decimal inteiro para o equivalente binário;
- A 1ª é fazer a soma das potências de 2, onde os bits “0” e “1” são colocados nos lugares apropriados:
- Ex:

$$45_{(10)} = 32 + 8 + 4 + 1 = 2^5 + 0 + 2^3 + 2^2 + 0 + 2^0$$

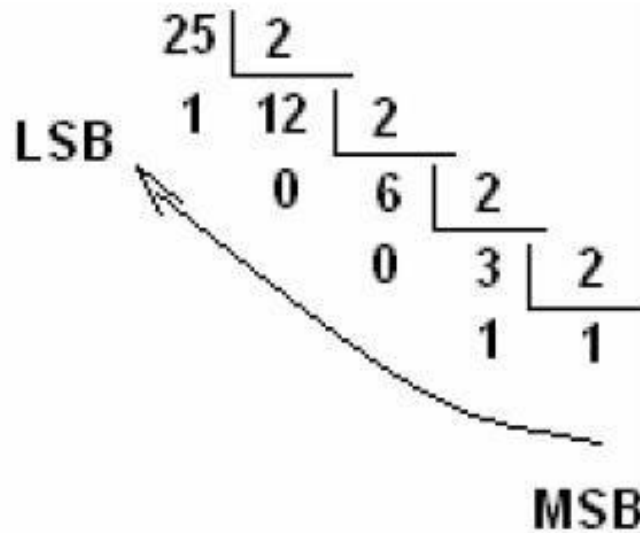
$$45_{(10)} = 101101_{(2)}$$

# Conversão entre Sistemas

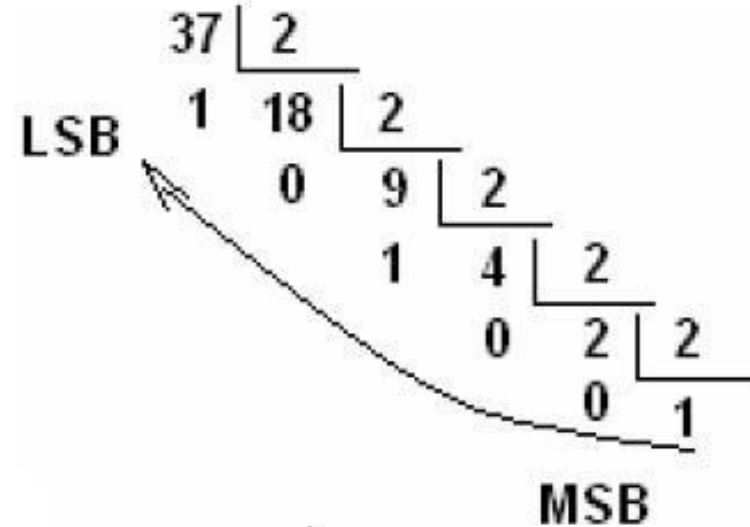
- **Conversão Decimal -> Binário**

- A 2ª forma (mais mecânica) é utilizar as divisões sucessivas por 2, e a escrita de modo inverso dos restos de cada divisão até que o quociente 0 seja obtido.

- EX:



$$25_{(10)} = 11001_{(2)}$$



$$37_{(10)} = 100101_{(2)}$$



- **Conversão Decimal -> Binário**

- Exercícios:

- Converta os números abaixo de decimal para binário:

- $47_{10}$

- $51_{10}$

- $25_{10}$

- $382_{10}$

- $426_{10}$

# Conversão entre Sistemas

---

- **Conversão Decimal -> Binário**

- Exercícios:

- Converta os números abaixo de decimal para binário:

- $47_{10} = 101111_2$

- $51_{10} = 110011_2$

- $25_{10} = 11001_2$

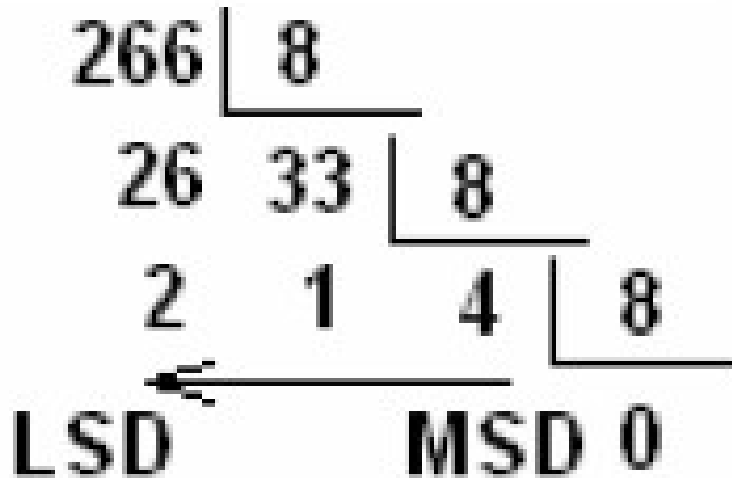
- $382_{10} = 101111110_2$

- $426_{10} = 110101010_2$

- **Conversão Decimal -> Binário**
  
- Por que não convertemos cada dígito diretamente de Decimal para Binário? R: **Porque 10 não é potência de 2.**

# Conversão entre Sistemas

- **Conversão Decimal -> Octal**
- Também utiliza-se o método das divisões sucessivas, só que agora a base é 8;
- Ex:  $266 = ?$



$$266_{(10)} = 412_{(8)}$$

# Conversão entre Sistemas

---

- **Conversão Decimal -> Octal**
- Converta os números abaixo de decimal para octal:
- $47_{10}$
- $51_{10}$
- $25_{10}$
- $382_{10}$
- $426_{10}$

# Conversão entre Sistemas

---

- **Conversão Decimal -> Octal**

- Converta os números abaixo de decimal para octal:

- $47_{10} = 57_8$

- $51_{10} = 63_8$

- $25_{10} = 31_8$

- $382_{10} = 576_8$

- $426_{10} = 652_8$

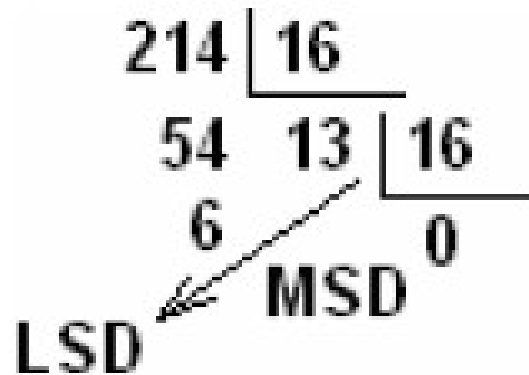
# Conversão entre Sistemas

- **Conversão Decimal -> Hexadecimal**

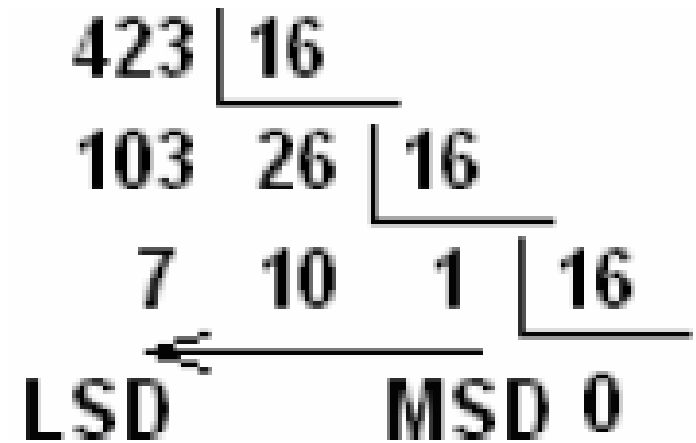
- Da mesma forma utiliza-se o processo de divisões sucessivas;

- EX1:

EX2:



$$214_{(10)} = D6_{(16)}$$



$$423_{(10)} = 1A7_{(16)}$$

# Conversão entre Sistemas

---

- **Conversão Decimal -> Hexadecimal**

- Converta os números abaixo de decimal para hexadecimal:

- $47_{10}$

- $51_{10}$

- $25_{10}$

- $382_{10}$

- $426_{10}$



# Conversão entre Sistemas

---

- **Conversão Decimal -> Hexadecimal**

- Converta os números abaixo de decimal para hexadecimal:

- $47_{10} = 2F_{16}$

- $51_{10} = 33_{16}$

- $25_{10} = 19_{16}$

- $382_{10} = 17E_{16}$

- $426_{10} = 1AA_{16}$

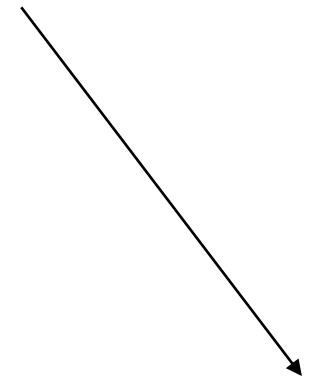
- **Conversão fracionária Decimal -> Binário**
- Tomemos o seguinte exemplo:  $91,6_{(10)} \rightarrow X_{(2)}$
- A parte inteira do número é convertida conforme o processo já demonstrado e obtemos assim o n°  **$1011011_{(2)}$** .
- A parte fracionária  $0,6_{(10)}$  é convertida da seguinte maneira:

- **Conversão fracionária Decimal -> outros**
- Multiplica-se a parte fracionária pela base “b”, neste caso o 2, e separa-se a parte inteira do produto. O resultado obtido da subtração da parte inteira do produto passa a ser o próximo multiplicando. Faz-se sucessivamente esta operação até que consiga uma precisão satisfatória. Lê-se os algarismos separados de cima para baixo.

# Conversão entre Sistemas

- Conversão fracionária Decimal -> outros
- Veja o exemplo:
- $0,6_{(10)} \rightarrow X_{(2)}$
- $0,6 \times 2 = 1,2$ 
  - Menos a parte inteira (1) = 0,2
  - Vezes 2 = 0,4
    - » Menos a parte inteira(0) = 0,4
    - » Vezes 2 = 0,8
      - » Menos a parte inteira(0) = 0,8
      - » Vezes 2 = 1,6
        - » Menos a parte inteira(1) = 0,6
        - » E assim sucessivamente

**MSB**



**LSB**

# Conversão entre Sistemas

---

- **Conversão fracionária Decimal -> outros**
- Veja o exemplo:
- $91,6_{(10)} \rightarrow X_{(2)}$
- **$1011011,1001_{(2)}$**
- Fazendo a verificação:
- $1 \times 2^{-1} + 0 \times 2^{-2} + 0 \times 2^{-3} + 1 \times 2^{-4}$
- $= \frac{1}{2} + \frac{1}{16} = \frac{9}{16} = 0,5625$
- Note que houve uma diferença de precisão na representação da grandeza nas diferentes bases.

# Conversão entre Sistemas

---

- **Conversão fracionária Decimal -> outros**
  
- Revisão:
- Para a parte inteira: divisões sucessivas por (b);
- Para a parte fracionária: multiplicações sucessivas por (b).

# Conversão entre Sistemas

---

- **Conversão fracionária Decimal -> Binário**
- Exercício:
- Converta os números fracionários decimal em binário:
  - $10,5_{(10)}$
  - $6,30_{(10)}$
  - $50,25_{(10)}$
  - $9,90_{(10)}$

# Conversão entre Sistemas

---

- **Conversão fracionária Decimal -> Binário**
- Exercício:
- Converta os números fracionários decimal em binário:
  - $10,5_{(10)} = 1010,10_{(2)}$
  - $6,30_{(10)} = 110,01001_{(2)}$
  - $50,25_{(10)} = 110010,010_{(2)}$
  - $9,90_{(10)} = 1001,1110_{(2)}$



- **Conversão Octal -> Binário**

- A principal vantagem do sistema octal é a transcrição de cada dígito octal para binário, sem a necessidade de cálculos:

- EX1:  $472_{(8)} = [100][111][010]$

$$472_{(8)} = 100111010_{(2)}$$

- EX2:  $5431_{(8)} = [101][100][011][001]$

$$5431_{(8)} = 101100011001_{(2)}$$

OCTAL	BINÁRIO
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

# Conversão entre Sistemas

---

- **Conversão Octal -> Binário**

- Exercício:

- Converta os números abaixo de octal para binário:

- $153_{(8)}$

- $361_{(8)}$

- $274_{(8)}$

- $523_{(8)}$

- $462_{(8)}$

# Conversão entre Sistemas

---

- **Conversão Octal -> Binário**

- Exercício:

- Converta os números abaixo de octal para binário:

- $153_{(8)} = 001101011_{(2)}$

- $361_{(8)} = 011110001_{(2)}$

- $274_{(8)} = 010111100_{(2)}$

- $523_{(8)} = 101010011_{(2)}$

- $462_{(8)} = 100110010_{(2)}$

# Conversão entre Sistemas

---

- **Conversão Hexadecimal -> Binário**

- Assim como na conversão octal para binário, utilizamos a substituição de cada dígito hexadecimal para seu correspondente binário;

- EX1:  $9F2_{(16)} = [1001][1111][0010]$

$$9F2_{(16)} = 100111110010_{(2)}$$

- EX2:  $1784_{(16)} = [0001][0111][1000][0100]$

$$1784_{(16)} = 0001011110000100_{(2)}$$

# Conversão entre Sistemas

---

- **Conversão Hexadecimal -> Binário**

- Exercício:

- Converta os números abaixo de hexadecimal para binário:

- $D20_{(16)}$

- $8463_{(16)}$

- $F59_{(16)}$

- $B47_{(16)}$

- $B1A4_{(16)}$

# Conversão entre Sistemas

---

- **Conversão Hexadecimal -> Binário**

- Exercício:

- Converta os números abaixo de hexadecimal para binário:

- $D20_{(16)} = 110100100000_{(2)}$

- $8463_{(16)} = 1000010001100011_{(2)}$

- $F59_{(16)} = 111101011001_{(2)}$

- $B47_{(16)} = 101101000111_{(2)}$

- $B1A4_{(16)} = 1011000110100100_{(2)}$

# Conversão entre Sistemas

---

- **Conversão Binário -> Octal**

- A conversão de números binários inteiros para octais inteiros se dá substituindo o conjunto de cada 3 binários pelo octal equivalente;
- Esta divisão deverá ser feita da direita (LSB) para esquerda (MSB); se faltar bits à esquerda preencher com zeros.

- EX1:

$$100111010_{(2)} = [100][111][010]$$

$$100111010_{(2)} = 472_{(8)}$$

# Conversão entre Sistemas

---

- **Conversão Binário -> Octal**

- Exercício:

- Converta os números abaixo de binário para octal:

- $10101110101010_{(2)}$

- $10101111_{(2)}$

- $1111000101010_{(2)}$

- $1010100001011_{(2)}$

- $101110100111_{(2)}$



# Conversão entre Sistemas

---

- **Conversão Binário -> Octal**

- Exercício:

- Converta os números abaixo de binário para octal:

- $10101110101010_{(2)} = 25652_{(8)}$

- $10101111_{(2)} = 257_{(8)}$

- $1111000101010_{(2)} = 17052_{(8)}$

- $1010100001011_{(2)} = 12413_{(8)}$

- $101110100111_{(2)} = 5647_{(8)}$

# Conversão entre Sistemas

---

- **Conversão Binário -> Hexadecimal**
- Análogo à conversão Binário -> Octal, só que agrupando 4 dígitos ao invés de 3.
- EX:

$$1110100110_{(2)} = [0011][1010][0110]$$

$$1110100110_{(2)} = 3A6_{(16)}$$

# Conversão entre Sistemas

---

- **Conversão Binário -> Hexadecimal**

- Exercício:

- Converta os números abaixo de binário para hexadecimal:

- $10101110101010_{(2)}$

- $10101110_{(2)}$

- $1111000101010_{(2)}$

- $1010100001011_{(2)}$

- $101110100111_{(2)}$

# Conversão entre Sistemas

---

- **Conversão Binário -> Hexadecimal**

- Exercício:

- Converta os números abaixo de binário para hexadecimal:

- $10101110101010_{(2)} = 2BAA_{(16)}$

- $10101110_{(2)} = AE_{(16)}$

- $1111000101010_{(2)} = 1E2A_{(16)}$

- $1010100001011_{(2)} = 150B_{(16)}$

- $101110100111_{(2)} = BA7_{(16)}$

# Conversão entre Sistemas

- Conversão Octal <-> Hexadecimal

- Converter para Binário e depois para Octal ou Hexadecimal.

- EX:

$$B2F_{(16)} = [1011][0010][1111]$$

$$B2F_{(16)} = 101100101111_{(2)}$$

$$B2F_{(16)} = [101][100][101][111]$$

$$B2F_{(16)} = 5457_{(8)}$$

# Conversão entre Sistemas

---

- **Conversão Octal <-> Hexadecimal**

- Exercício:

- Converta os números abaixo:

- $36C_{(16)}$

- $E29_{(16)}$

- $1B4_{(16)}$

- $2650_{(8)}$

- $3457_{(8)}$

# Conversão entre Sistemas

---

- **Conversão Octal <-> Hexadecimal**

- Exercício:

- Converta os números abaixo:

- $36C_{(16)} = 1554_{(8)}$

- $E29_{(16)} = 7051_{(8)}$

- $1B4_{(16)} = 664_{(8)}$

- $2650_{(8)} = 5A8_{(16)}$

- $3457_{(8)} = 72F_{(16)}$

- **Resumo das conversões**
- De binário, octal ou hexadecimal para decimal, use o método da soma dos pesos de cada dígito (valor posicional);
- De decimal para binário, octal ou hexadecimal, utilize o método das divisões/multiplicações sucessivas;
- De binário para octal ou hexadecimal, agrupe os bits da direita para esquerda e converta cada grupo;



- **Resumo das conversões**
- De octal ou hexadecimal para binário converta cada dígito em 3 (octal) ou 4 (hexadecimal) bits equivalentes;
- De octal para hexadecimal ou (vice-versa) utilize a conversão para binário, daí então faça a conversão desejada.



## Introdução à Computação

Jordana Sarmenghi Salamon

`jssalamon@inf.ufes.br`

[jordanasalamon@gmail.com](mailto:jordanasalamon@gmail.com)

<http://inf.ufes.br/~jssalamon>

Departamento de Informática

Universidade Federal do

Espírito Santo