UFPR
UNIVERSIDADE FEDERAL DO PARANÁ

# Métodos Numéricos

## Erros – Ponto Flutuante

**Professor Volmir Eugênio Wilhelm**
**Professora Mariana Kleina**

# Representação Numérica

O conjunto dos números representáveis em qualquer máquina é finito, e portanto discreto, ou seja não é possível representar em uma máquina todos os números de um dado intervalo [a, b].

O resultado de uma simples operação aritmética ou o cálculo de uma função, realizadas com esses números, podem conter erros.

# Representação Numérica

## Representação em Ponto Fixo

### Número inteiro

Seja $n \neq 0$ formado por t=(k+1) dígitos

$$n_{(\beta)} = \pm(d_k d_{k-1} \ldots d_1 d_0) = \pm(d_k \beta^k + d_{k-1}\beta^{k-1} + \ldots + d_1\beta^1 + d_0\beta^0),$$

onde $\beta$ é a base e os $d_i$, i = 0,1,...,k são inteiros tal que $0 \leq d_i < \beta$ e $d_k \neq 0$

Exemplo ($\beta$=10): $1950_{(10)} = 1\times10^3 + 9\times10^2 + 5\times10^1 + 0\times10^0$

### Número Fracionário

Seja $x_I$ a parte inteira do número real x, então sua parte fracionária $x_F = x - x_I$ pode ser representada por

$$x_{F(\beta)} = \pm(0,d_n d_{n-1} \ldots d_1) = \pm(d_n\beta^{-1} + d_{n-1}\beta^{-2} + \ldots + d_1\beta^{-n}),$$

Exemplo ($\beta$=2): $110{,}011_{(2)} = (1\times2^2 + 1\times2^1 + 0\times2^0) + (0\times2^{-1} + 1\times2^{-2} + 1\times2^{-3})$

# Representação Numérica

## Representação em Ponto Fixo

$$x = \pm \sum_{i=k}^{n} d_i \beta^{-i}$$

- $k$ e $n$ são inteiros satisfazendo $k < n$ e usualmente $k \leq 0$ e $n \geq 0$
- $d_i$ são inteiros satisfazendo $0 \leq d_i < \beta$ e $d_1 \neq 0$

Exemplo ($\beta$=10):

$$1897{,}26 = \pm \sum_{i=-3}^{2} x_i \beta^{-i} = 1\times10^3 + 8\times10^2 + 9\times10^1 + 7\times10^0 + 2\times10^{-1} + 6\times10^{-2}$$

# Representação Numérica

## Representação em Ponto Flutuante

A representação geral de um número $x$ na base $\beta$ pode ser dada por

$$x = \text{sinal} \times \text{mantissa} \times \beta^{\text{expoente}}$$

$$x = \pm d \times \beta^{e}$$

Ou ainda, por

$$\pm(0,d_1 d_2 d_3 \ldots d_t) \times \beta^{e}$$

- $t$ é o número de dígitos na mantissa (dígitos significativos);
- $0 \leq m_j \leq (\beta - 1)$, $d_1 \neq 0$;
- $e$ é o expoente no intervalo $[l, u]$

# Ponto Flutuante

## Representação em Ponto Fixo *versus* Ponto Flutuante

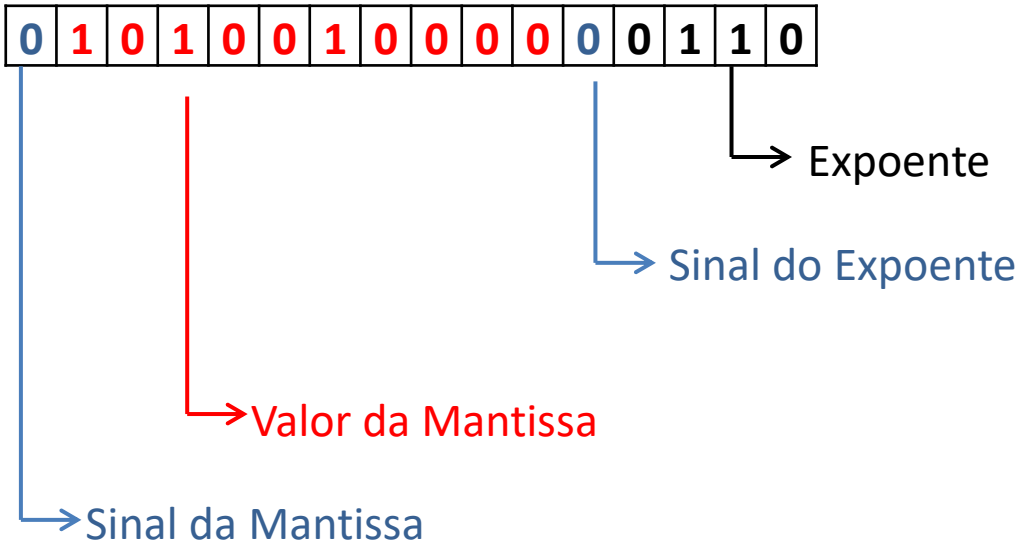
Exemplos

| Ponto Fixo | Ponto Flutuante |
|---|---|
| 3,1415 | $0{,}31415 \times 10^{1}$ |
| 0,00000001 | $0{,}1 \times 10^{-7}$ |
| $3{,}15576 \times 10^{9}$ | $0{,}315576 \times 10^{10}$ |
| 15368 | $0{,}15368 \times 10^{5}$ |
| -5272,052 | $-0{,}5272052 \times 10^{4}$ |
| 0,365 | $0{,}365 \times 10^{0}$ |
| 0,025 | $0{,}25 \times 10^{-1}$ |

# Ponto Flutuante

## Representação de número binário em ponto flutuante numa máquina

$+41_{(10)} = +101001_{(2)} = +0{,}101001 \times 2^{6} = +0{,}101001 \times 2^{+110}$



# Ponto Flutuante

## Representação de número binário em ponto flutuante numa máquina

$-1{,}75_{(10)} = -1{,}11_{(2)} = -0{,}111 \times 2^{1}$

| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

$+13{,}625_{(10)} = +1101{,}101_{(2)} = +0{,}1101101 \times 2^{4}$

| 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

$-0{,}15625_{(10)} = -0{,}00101_{(2)} = -0{,}101 \times 2^{-2}$

| 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

$+39_{(10)} = +100111_{(2)} = +0{,}100111 \times 2^{6}$

| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

$+0{,}0130615234375_{(10)} = +0{,}0000001101011_{(2)} = +0{,}1101011 \times 2^{-6}$

| 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

# Ponto Flutuante

**Ponto flutuante (Padrão IEEE 754)**
(IEEE-Institute Of Electrical And Electronic Engineers)

- O formato de precisão simples (**float**) ocupa **32 bits**.


- O formato de precisão dupla (**double**) ocupa **64 bits**.


(PS: O valor do expoente, pelo padrão IEEE 754 é armazenado pela notação com peso, também chamada de notação por excesso de valor.)

# Ponto Flutuante

## Aritmética do Ponto Flutuante

Os computadores representam números na forma de ponto flutuante. Na aritmética de ponto flutuante o número é representado na forma:

$$\pm(0,d_1d_2d_3...d_t) \times \beta^e$$

onde

$\beta$ é a base;

t é o número de dígitos na mantissa;

$0 \leq d_j \leq (\beta-1)$, $j=1,...,t$, $d_1 \neq 0$;

e é o expoente no intervalo [l, u]

# Ponto Flutuante

## Aritmética do Ponto Flutuante

Exemplo A: Seja uma máquina que opera no sistema $\beta=10$, $t=3$, $e \in [-5, 5]$

Os números são representados na forma

$$\pm(0,d_1d_2d_3) \times 10^e,\ 0 \le d_j \le 9,\ j=1,2,3,\ d_1 \neq 0;$$

Nesta máquina, em módulo, o

menor número: $m = (0,100) \times 10^{-5} = 0,000001$

maior número: $M = (0,999) \times 10^{+5} = 99900$

O número $x = 235,89 = 0,23589 \times 10^3$ nesta máquina (que opera com três dígitos) será representado por **$x = 0,235 \times 10^3$** se for **usado o truncamento**, e **$x = 0,236 \times 10^3$** se for **usado o arredondamento**.

Se $|x| > M \Rightarrow$ *overflow* (por exemplo $x = 0,267 \times 10^6$)

Se $|x| < m \Rightarrow$ *underflow* (por exemplo $x = 0,789 \times 10^{-7}$)

# Ponto Flutuante

## Aritmética do Ponto Flutuante

Exemplo B: Seja uma máquina com $\beta=10$, l = −2 e u = 2 e t = 3.

Sendo assim temos:

$0{,}35 = 0{,}0350 \times 10^{0}$

$-5{,}172 \approx -0{,}517 \times 10^{1}$

$0{,}0123 = 0{,}123 \times 10^{-1}$

$5391{,}3 = 0{,}53913 \times 10^{4} \Rightarrow$ *overflow*

$0{,}0003 = 0{,}300 \times 10^{-3} \Rightarrow$ *underflow*

# Ponto Flutuante

## Aritmética do Ponto Flutuante

Exemplo C: Seja uma máquina com $\beta = 10$, l = −4, u = 4 e t = 3.

| x | Arredondamento | Truncamento |
|---|---|---|
| 0,002557 | $0{,}256 \times 10^{-2}$ | $0{,}255 \times 10^{-2}$ |
| 1,250000 | $0{,}125 \times 10^{1}$ | $0{,}125 \times 10^{1}$ |
| 10,053000 | $0{,}101 \times 10^{2}$ | $0{,}100 \times 10^{2}$ |
| -253,150000 | $-0{,}253 \times 10^{3}$ | $-0{,}253 \times 10^{3}$ |
| 2,718280 | $0{,}272 \times 10^{1}$ | $0{,}271 \times 10^{1}$ |
| 0,000002 | *underflow* | expoente < -4 |
| 817235,890000 | *overflow* | expoente > +4 |

# Ponto Flutuante

## Formato simples (32 bits ou 4 bytes) – algumas calculadoras

Base decimal: Máquina com $\beta = 10$, $l = -38$, $u = 38$ e $t = 7$.
(Base binária - o computador: Máquina com $\beta = 2$, $l = -126$, $u = 127$ e $t = 23$ bits)

máximo: $0{,}111111...111 \times 2^{127} = 0{,}340 \times 10^{38}$

mínimo: $0{,}100000...000 \times 2^{-126} = 0{,}218 \times 10^{-38}$

Veja mais em http://grouper.ieee.org/groups/754/

# Ponto Flutuante

## Formato duplo (64 bits ou 8bytes) – IEEE 754 no Excel, Matlab, Octave

Base decimal: Máquina com $\beta = 10$, $l = -308$, $u = 308$ e $t = 15$.
(Base Binária - No computador: Máquina com $\beta = 2$, $l = -1023$, $u = 1024$ e $t = 52$ bits)

máximo: $0{,}111111...111 \times 2^{1023} = 0{,}179769313486232 \times 10^{308}$

mínimo: $0{,}100000...000 \times 2^{-1023} = 0{,}22250738585072 \times 10^{-308}$

Exemplo A (no Excel):

| Célula | Digite | Resultado |
|---|---|---|
| A1 | 1,2E200 | 1,20E+200 |
| B1 | 1E100 | 1E+100 |
| C1 | =(A1 + B1) | 1,20E+200 |
| D1 | =SE(C1=A1;"verdadeiro") | verdadeiro |

Isso é causado pela especificação IEEE ao armazenar somente 15 dígitos significativos de precisão. Para poder armazenar o cálculo acima (exatamente), o Excel solicitará pelo menos 100 dígitos de precisão. (http://support.microsoft.com/kb/78113/pt-br)

Exemplo B(no Excel B):

$(0{,}5 - 0{,}4 - 0{,}1) = -0{,}277555756156289 \times 10^{-017}$