2 Representação de números no computador:

(Quarteroni, pdf)

* bit (0 ou 1), com ou sem corrente elétrica

→ menor unidade de informação

→ números binários

* representação de números:

$$(1\,2\,3\,8)_{10} = 1\times 10^3 + 2\times 10^2 + 3\times 10^1 + 8\times 10^0$$

$$= 1\times 1000 + 2\times 100 + 3\times 10 + 8\times 1 = 1238$$

⇒ base 2: (somente 0 e 1)

$$(10010)_2 = 1\times 2^4 + 0\times 2^3 + 0\times 2^2 + 1\times 2^1 + 0\times 2^0$$

$$= 1\times 16 + 0\times 8 + 0\times 4 + 1\times 2 + 0\times 1$$

$$= 16 + 2 = (18)_{10}$$

$$(0)_2 = 0\times 2^0 = (0)_{10}$$

$$(1)_2 = 1\times 2^0 = (1)_{10}$$

$$(10)_2 = 1\times 2^1 + 0\times 2^0 = (2)_{10}$$

$$(11)_2 = 1\times 2^1 + 1\times 2^0 = (3)_{10}$$

$$(100)_2 = 1\times 2^2 + 0\times 2^1 + 0\times 2^0 = (4)_{10}$$

⋮

ex) $(28)_{10}$

28 | 2
0 — 14 | 2
0 — 7 | 2
1 — 3 | 2
1 — 1 | 2
1 — 0

$= (11100)_2$

precisamos de 5 bits para representar $(28)_{10}$ no computador.

* Números reais: $\longrightarrow \mathbb{R}$ (infinitos)

$x = 0{,}25$ $\qquad x = \frac{1}{3} = 0{,}333\ldots$ $\qquad x = \sqrt{2}$ (irracional, $\neq \frac{A}{B}$)

$\Rightarrow$ Ponto flutuante: número finito de casas decimais

$$x = (-1)^{s}\, 0.\underbrace{a_1 a_2 \ldots a_t}_{\text{mantissa } m} \times \beta^{e}$$

$\begin{cases} s \rightarrow \text{sinal} \\ m \rightarrow \text{mantissa} \\ e \rightarrow \text{expoente} \end{cases}$ $\qquad \beta \rightarrow$ base

ex)
- $250{,}32 \Rightarrow (-1)^{0}\, 0{,}\underbrace{25032} \times 10^{3}$
- $-0{,}33 \Rightarrow (-1)^{1}\, 0{,}\underbrace{33} \times 10^{0}$
- $0{,}001 \Rightarrow (-1)^{0}\, 0{,}\underbrace{1} \times 10^{-2}$

obs: se $a_1 \neq 0$, representação é única

* Padrão IEEE 754: base 2, armazena:
- s (0 ou 1)
- m
- e

$\left.\begin{matrix} m \\ e \end{matrix}\right\}$ em binário

- byte = 8 bits
- half = 16 bits (2 bytes)
- single = 32 bits (4 bytes)
- double = 64 bits (8 bytes)

s	e					m									
	1	2	3	4	5	1	2	3	4	5	6	7	8	9	10

* half: • 5 bits: e, 10 bits: m (16 total)

  • $|x_{max}| = 0{,}655 \times 10^{5}$
  • $|x_{min}| = 0{,}610 \times 10^{-4}$
  • precisão decimal: 3 casas

* single: • 8 bits e, 23 bits m (32 total)

  • $|x_{max}| = 0{,}3402824 \times 10^{39}$
  • $|x_{min}| = 0{,}1175494 \times 10^{-37}$
  • precisão decimal: 7 casas

* double: • 11 bits e, 52 bits m (64 total)
  • $|x_{max}| = 0{,}1797693\ldots \times 10^{309}$
  • $|x_{min}| = 0{,}22250\ldots \times 10^{-307}$
  • precisão decimal: 15 casas

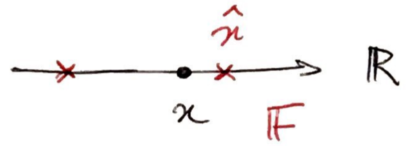
obs: se $|x| < |x_{min}| \rightarrow x = 0$ (underflow)

se $|x| > |x_{max}| \rightarrow \pm\infty$ (overflow)

* exemplo de memória: 1 milhão de dados:

  • half: $10^6 \times 2$ bytes = 2 Mb
  • single: " $\times 4$ " = 4 Mb
  • double: " $\times 8$ " = 8 Mb.

* erro em ponto flutuante:



- arredondamento: minimiza o erro absoluto $|x-\hat{x}|$.
- erro relativo: $\dfrac{|x-\hat{x}|}{|x|}$ é aprox. constante para qualquer x.

ex) precisão de 4 casas decimais:

$\Rightarrow$ ponto fixo:

a) $x = 3507{,}6$; $\hat{x} = 3507$

$|x-\hat{x}| = 0{,}4$ (erro absoluto)

$\dfrac{|x-\hat{x}|}{|x|} \approx 1{,}1 \times 10^{-4}$ (erro relativo)

armazena zeros

b) $x = 0{,}0035076$; $\hat{x} = 0{,}004$

$|x-\hat{x}| = 0{,}0004924$ (erro absoluto)

$\dfrac{|x-\hat{x}|}{|x|} \approx 0{,}14$ (erro relativo)

$\Rightarrow$ ponto flutuante:

a) $x = 3507{,}6 = 0{,}35076 \times 10^{4}$; $\hat{x} = 0{,}3508 \times 10^{4}$

$|x-\hat{x}| = 0{,}4$

$\dfrac{|x-\hat{x}|}{|x|} \approx 1{,}1 \times 10^{-4}$

b) $x = 0{,}0035076 = 0{,}35076 \times 10^{-2}$; $\hat{x} = 0{,}3508 \times 10^{-2}$

$|x-\hat{x}| = 0{,}0004$

$\dfrac{|x-\hat{x}|}{|x|} \approx 1{,}1 \times 10^{-4}$

erro relativo da ordem da precisão

* operações aritméticas com ponto flutuante :

a) associativa $(x+y)+z \neq (x+y)+z$ X em $\mathbb{F}$

b) comutativa: $x+y = y+x$ ✓

$xy = yx$ ✓

c) distributiva: $x(y+z) \neq xy + xz$ X em $\mathbb{F}$

obs: trabalhe com números na ordem de 1 !

ex) $\dfrac{(1+x)-1}{x}$

Revisão : • números em computador: ponto flutuante

$$x = (-1)^s \, 0.m \times 2^e$$

| s | e | m |

bits (binário)

• single: 32 bits (4 bytes), $(-1)^s \, 0, \underbrace{a_1 a_2 a_3 a_4 a_5 a_6 a_7}_{} \times \underline{10}^e$

$-37 \leqslant e \leqslant 39$ 7 casas em decimal

• double: 64 bits (8 bytes), $(-1)^s \, 0, \underbrace{a_1 a_2 \cdots a_{15}}_{} \times \underline{10}^e$

$-307 \leqslant e \leqslant 309$ 15 casas em decimal

$\Rightarrow$ ordem das operações importa $\Rightarrow$ erro de arredondamento

$\Rightarrow$ trabalhar com números na ordem de 1