Cálculo Numérico / Métodos Numéricos

Representação de números em computadores

Mudança de base

Computadores são "binários"

- Por que 0 ou 1 ?
- 0 ou 1 - "fácil" de obter um sistema físico
  - Transistores tem duas posições estáveis: ligado ou desligado
- Expansão binária de um número

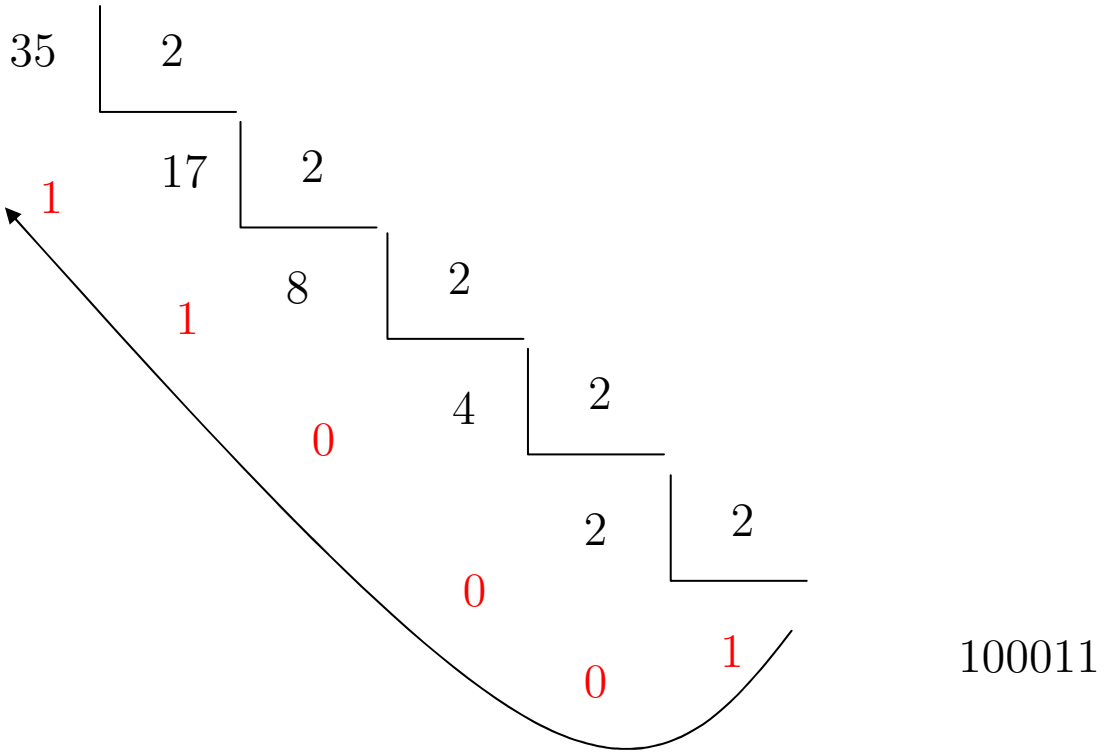
$$a_n = 1, a_i = 0 \text{ ou } 1, 0 \leq i < n$$

$$N = \pm a_n \cdot 2^n + a_{n-1} \cdot 2^{n-1} + \ldots + a_1 \cdot 2^1 + a_0 2^0$$

  - Representação binária: $(a_n, a_{n-1}, \ldots, a_1, a_0)$

Conversões entre base 10 e base 2

- Da base 2 para a base 10
  - $(100011) = 1{\cdot}2^5 + 0{\cdot}2^4 + 0{\cdot}2^3 + 0{\cdot}2^2 + 1{\cdot}2^1 + 1{\cdot}2^0$
    $= 35$
- Da base 10 para a base 2



Representação de números reais

- Representação de ponto fixo

$$23231.333448$$
$$6732.222232$$
$$0.000023$$

Representação de números reais

- Representação de ponto fixo

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

  - **k** e **n** são inteiros satisfazendo **k** $<$ **n** e usualmente **k**≤ 0 e **n**>0
  - $\mathbf{x_i}$ são inteiros satisfazendo $0 \leq \mathbf{x_i} < \boldsymbol{\beta}$

- **Exemplo:**

$$1997.16 = \sum_{i=-3}^{2} x_i \beta^{-i}$$

$$1997.16 = 1 \cdot 10^3 + 9 \cdot 10^2 + 9 \cdot 10^1 + 7 \cdot 10^0 + 1 \cdot 10^{-1} + 6 \cdot 10^{-2}$$

Representação de números reais

- Representação de ponto fixo

  - Representação à qual estamos mais habituados.
  - A diferença entre dois números representáveis é fixa.

Representação de números reais

- Representação de ponto flutuante

$$.24234235 \times 10^{4}$$
$$.52423423 \times 10^{-3}$$
$$.73836224 \times 10^{0}$$

Representação de números reais

- Representação de ponto flutuante

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

  - **β** é a base do sistema de numeração
  - **e** é o expoente
  - **d** é a mantissa. **d** é um número em ponto fixo:

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

  - frequentemente: k=1

    $0 \leq d_i < \beta$ i=1,...,t (número de dig. sign.)

    $\beta^{-1} \leq d < 1$

    $-m \leq e \leq M$

Representação de números reais

- $d_1 \neq 0$ representa o sistema de **números em ponto flutuante normalizado**.

- Como representar o zero ?
  - mantissa = 0
  - e = -m

Exemplos

- $0.35 =$
  - mantissa: $(3 \times 10^{-1} + 5\times 10^{-2})$
  - e $= 0$
  - $= 0.35 \times 10^{0}$

- $-5.127 =$
  - mantissa: $-(5 \times 10^{-1} + 1\times 10^{-2} + 2 \times 10^{-3} + 7 \times 10^{-4})$
  - e $= 1$
  - $= -0.5127 \times 10^{1}$

- $0.0003 =$
  - mantissa: (3×10^{-1})
  - e $= -3$
  - 0.3×10^{-3}

Notação

- Representação de um sistema de notação com base β, número de dígitos significativos t e expoentes mínimo e máximo m e M:
  - F(β, t, m, M)

$$\pm\ 0.d_1d_2d_3...d_t \times \beta^e$$

  - $d_1 \neq 0$;
  - $m \leq e \leq M$

Exemplos

- Represente os números 0.35, 5391 e 0.0003 no sistema F(10,3,-2,2)

  - O.35:

    $(3\times10^{-1} + 5\times10^{-2} + 0\times10^{-3})\times 10^{0}$

    0.350×10^{0}

  - 5391

    $(5\times10^{-1} + 3\times10^{-2} + 9\times10^{-3}) \times 10^{4}$ overflow

  - 0.0003

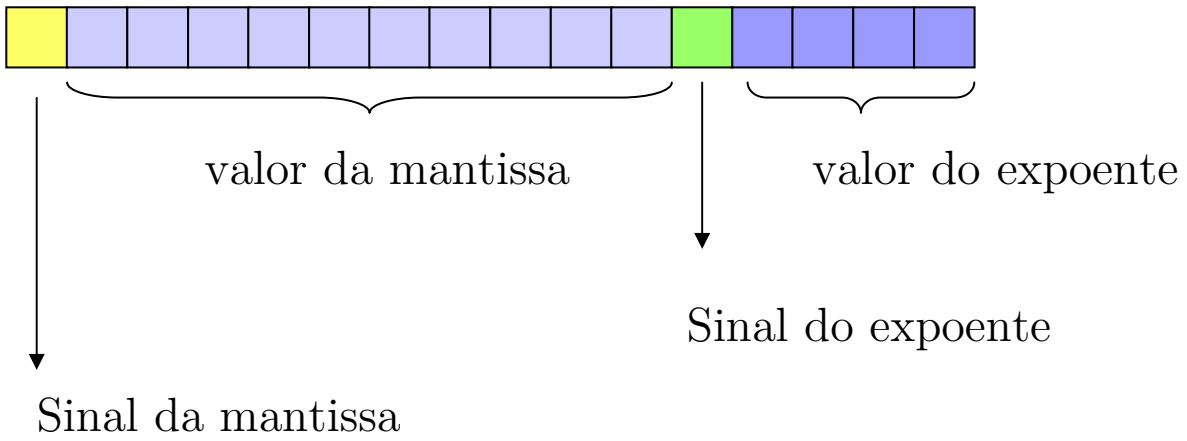
    $(3\times10^{-1} + 0\times10^{-2} + 0\times10^{-3}) \times 10^{-3}$ underflow

Exemplo (Cálculo Numérico. Sperandio, Mendes e Silva)

- Tome o sistema de representação dado por

F(2,10,-15,15)

a) Represente de alguma maneira como esse sistema pode ser armazenado em um computador binário.

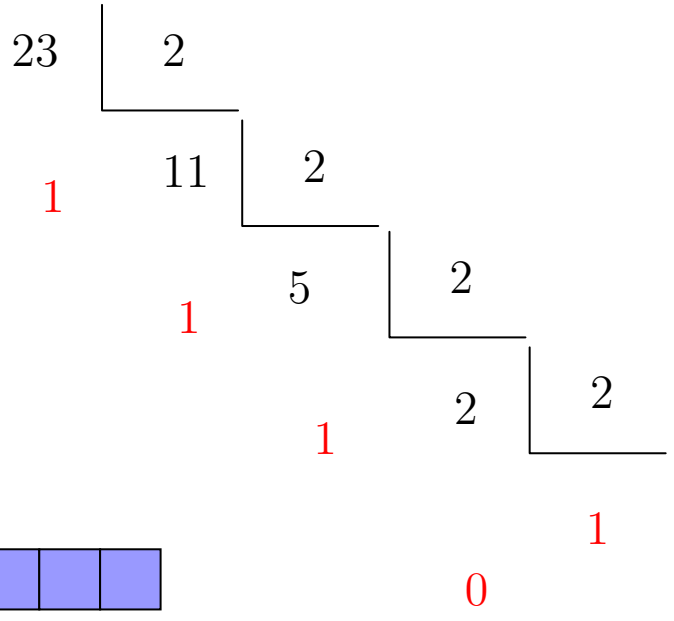


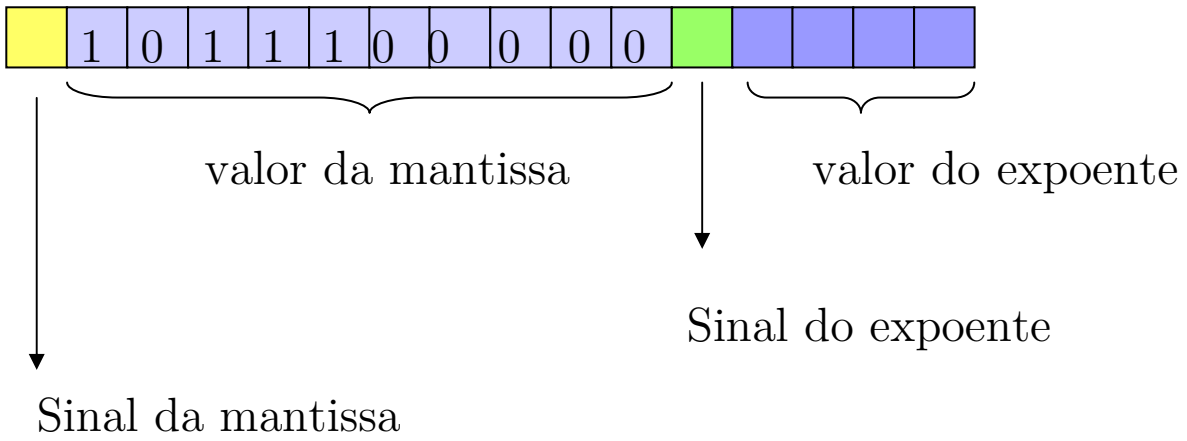
Exemplo (Cálculo Numérico. Sperandio, Mendes e Silva)

- Tome o sistema de representação dado por

F(2,10,-15,15)

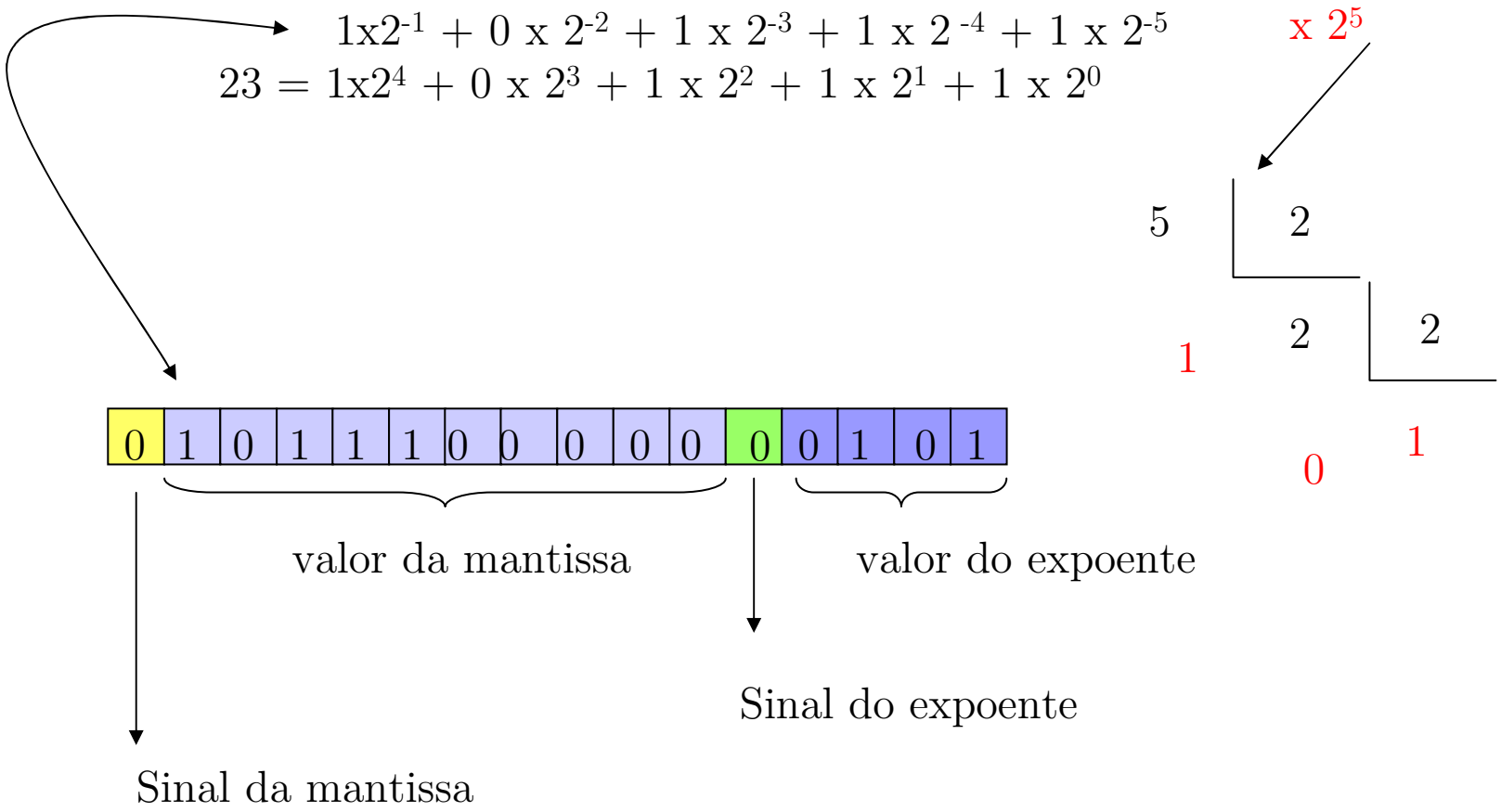
a) Represente o número $(23)_{10}$.





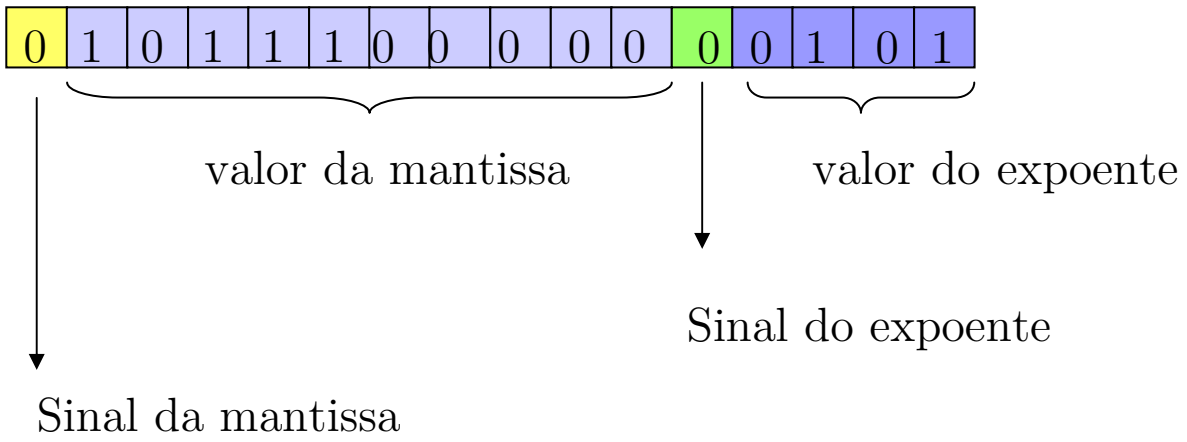
Exemplo (Cálculo Numérico. Sperandio, Mendes e Silva)

- Tome o sistema de representação dado por F(2,10,-15,15)



Exemplo

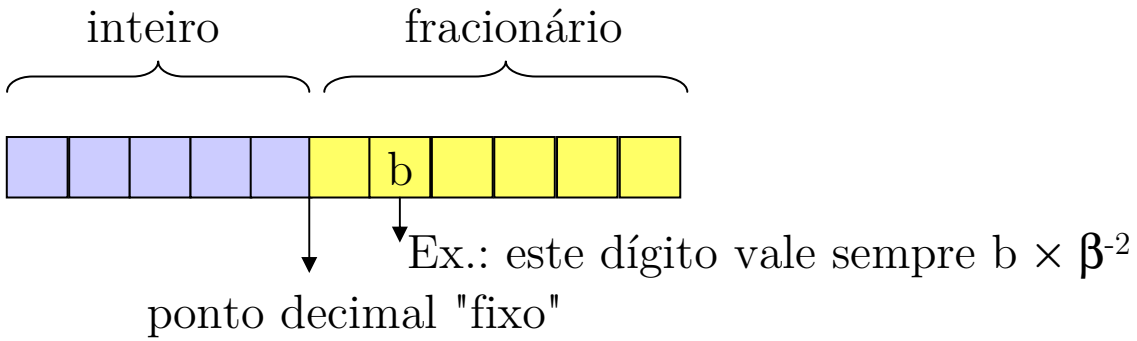
Por que no caso da mantissa, completamos com zeros ao final do espaço reservado e no caso do expoente, ao início ?



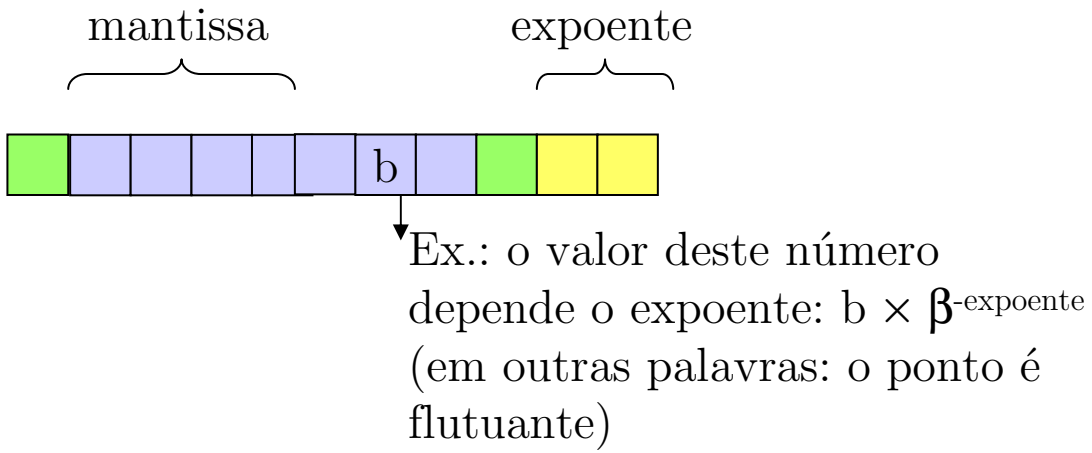
Diferenças: ponto fixo × flutuante

Suponha que temos 10 dígitos disponíveis:

Ponto fixo:

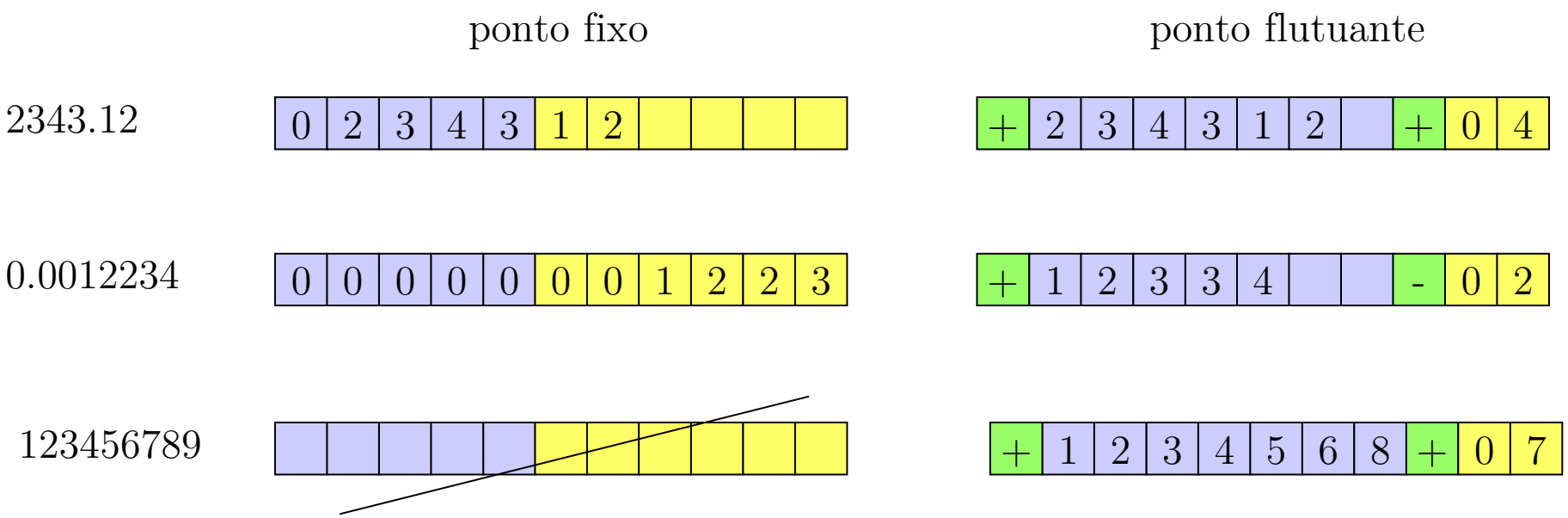


Ponto flutuante:



Exemplo: ponto fixo × flutuante

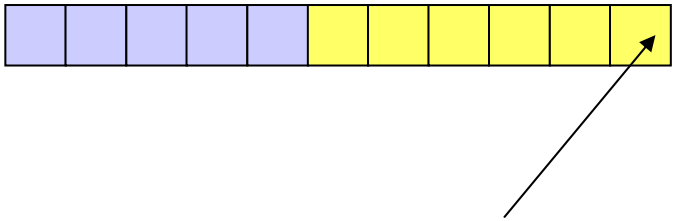
Base 10.



(Questão em aberto: arredondamento!)

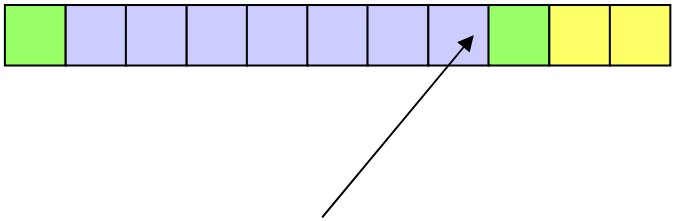
Espaçamento entre dois números representáveis

ponto fixo

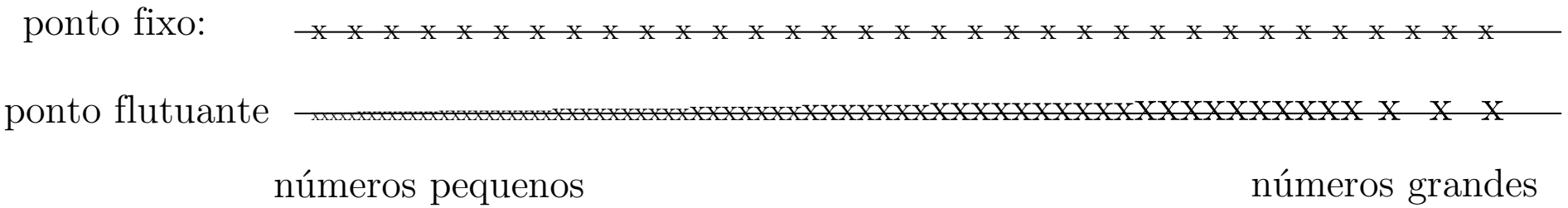


Sempre igual ao valor unitário desta casa.
No caso: 10^{-6}

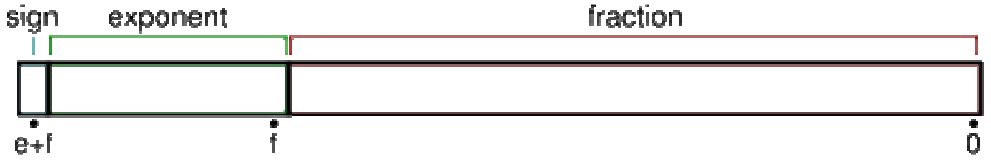
ponto flutuante



Depende do intervalo considerado:
No caso: $10^{-7} \times 10^{\text{expoente}}$



Curiosidade: IEEE 754



Type	Sign	Exponent	Significand	Value
Zero	0	0000 0000	000 0000 0000 0000 0000 0000	0.0
One	0	0111 1111	000 0000 0000 0000 0000 0000	1.0
Minus One	1	0111 1111	000 0000 0000 0000 0000 0000	−1.0
Smallest denormalized number	*	0000 0000	000 0000 0000 0000 0000 0001	$\pm 2^{-23} \times 2^{-126} = \pm 2^{-149} \approx \pm 1.4 \times 10^{-45}$
"Middle" denormalized number	*	0000 0000	100 0000 0000 0000 0000 0000	$\pm 2^{-1} \times 2^{-126} = \pm 2^{-127} \approx \pm 5.88 \times 10^{-39}$
Largest denormalized number	*	0000 0000	111 1111 1111 1111 1111 1111	$\pm(1-2^{-23}) \times 2^{-126} \approx \pm 1.18 \times 10^{-38}$
Smallest normalized number	*	0000 0001	000 0000 0000 0000 0000 0000	$\pm 2^{-126} \approx 1.18 \times 10^{-38}$
Largest normalized number	*	1111 1110	111 1111 1111 1111 1111 1111	$\pm(1-2^{-24}) \times 2^{128} \approx \pm 3.4 \times 10^{38}$
Positive infinity	0	1111 1111	000 0000 0000 0000 0000 0000	$+\infty$
Negative infinity	1	1111 1111	000 0000 0000 0000 0000 0000	$-\infty$
Not a number	*	1111 1111	non zero	NaN
* Sign bit can be either 0 or 1 .				

mais informação: http://grouper.ieee.org/groups/754/
http://en.wikipedia.org/wiki/IEEE_floating-point_standard

Mudança de base

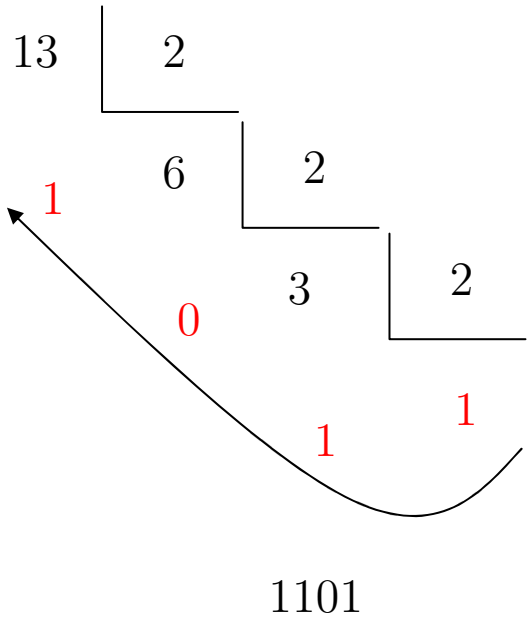
Mudança de base

- Da base 2 para a base 10
  - $N_2 = 1010.1110$

$$N_{10} = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3} + 0 \times 2^{-4} = 10.875$$

Mudança de base

- Da base 10 para a base 2
  - $N_{10} = 13.75$



0.75

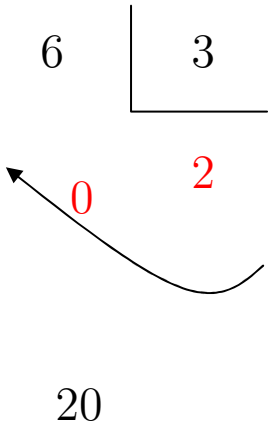
$0.75 \times 2 = 1.50$
$0.50 \times 2 = 1.00$
$0.00 \times 2 = 0.00$

$$(13.75)_{10} = (1101.110)_2$$

E para outras bases ?

- 12.20 da base 4 para a base 3

$$(12.20)_4 = (1\times 4^1 + 2 \times 4^0 + 2 \times 4^{-1} + 0 \times 4^{-2})_{10} = (6.5)_{10}$$



0.50

$0.50 \times 3 = 1.50$
$0.50 \times 3 = 1.50$
$0.50 \times 3 = 1.50$
...

$$(12.20)_4 = (6.5)_{10} = (20.111...)_3$$

Arredondamento

Arredondamento

- F(β,t,m,M)

$$s + \tfrac{1}{2}\beta^{-t} = 0.d_1 d_2 d_3 \ldots d_{t-1} d_t d_{t+1} \ldots$$

base 10:	$\frac{1}{2}\beta^{-t} =$
t=1:	0.05
t=2:	0.005
t=3:	0.0005
...	

Exemplo:

- F(10,5,-4,4)

$$10.232242 = 0.10232242 \times 10^{2}$$

$$\text{mantissa} + \ 0.5 \times 10^{-5} = 0.10232\cancel{742}$$

Se for em outra base, 0.5 tem que estar expresso na base considerada.

Cálculo Numérico / Métodos Numéricos

Operações numéricas em ponto flutuante

Instabilidade numérica

Mal condicionamento

Arredondamento

- F(β,t,m,M)

$$s + \tfrac{1}{2}\beta^{-t} = 0.d_1 d_2 d_3 \ldots d_{t-1} d_t d_{t+1} \ldots$$

base 10: $\tfrac{1}{2}\beta^{-t} =$

t=1: 0.05
t=2: 0.005
t=3: 0.0005
...

Atenção: somar à mantissa (e não ao número em si)

Arredondamento (exemplo)

- Represente o número 1234.56 no sistema F(10,3,5,5), com arredondamento:

0.123456×10^4

mantissa: $0.123456 \quad + \quad 0.5 \times 10^{-3}$

$0.123456 + \ 0.0005 = 0.123956$

Resposta: 0.123×10^4

Arredondamento (exemplo 2)

- Represente o número $(1001.1)_2$ no sistema F(2,3,5,5), com arredondamento:

0.10011×2^4

mantissa: $0.10011 + (0.5 \times 2^{-3})_{10}$

$0.10011 + (0.1)_2 \times 2^{-3}$

$0.10011 + (0.0001)$

0.10101

Resposta: 0.101×2^4

Efeito do arredondamento

- Após cada operação, perdemos informação.
  - Exemplo (t = 3):

$$(3.18/5.05) \times 11.4 =$$
$$0.630 \times 11.4 =$$
$$7.182$$
$$7.18$$

$$(3.18 \times 11.4)\ /5.05 =$$
$$36.3/5.05 =$$
$$7.188$$
$$7.19$$

Efeitos numéricos

- Cancelamento
  - efeito numérico que causa perda de dígitos significativos quando subtraímos dois números muito próximos.
  - Ex.:

$$\sqrt{9876} - \sqrt{9875}$$

$$= 0.9937806599 \times 10^2 \text{ - } 0.9937303457 \times 10^2$$
$$= 0.0000503142 \times 10^2$$
$$= 0.5031420000 \times 10^{-2}$$

$$\sqrt{x} - \sqrt{y} = \frac{x-y}{\sqrt{x}+\sqrt{y}}$$

Efeitos numéricos

- Propagação do erro
  - efeito numérico que causa perda de dígitos significativos quando na soma de vários números, uma soma intermediária é muito maior que a soma final.

  - Ex.: F(10,3,5,5)

$$
\begin{aligned}
& 100 + 0.000100 - 100 \\
&= \underbrace{0.100 \times 10^{3} + 0.100 \times 10^{-3}} - 0.100 \times 10^{3} \\
&= \qquad\qquad 0.100 \times 10^{3} \qquad\qquad - 0.100 \times 10^{3} \\
&= 0
\end{aligned}
$$

Instabilidade numérica

- Os erros podem ir se acumulando durante o processo
- Erros intermediários podem anular-se
  - Estabilidade

- Instabilidade: os erros intermediários têm uma influência muito grande no resultado final.

Instabilidade numérica (exemplo)

$$x_n = (\tfrac{1}{3})^n$$

Podemos provar que x_n é dado pela seguinte sequência:

$$x_0 = 1, x_1 = \tfrac{1}{3}$$

$$x_{n+1} = \tfrac{13}{3}x_n - \tfrac{4}{3}x_{n-1}$$

Atenção: a fórmula está correta! O erro é numérico!

$x_0 = 1.0000000$
$x_1 = 0.3333333$ (7 correctly rounded significant digits)
$x_2 =$
$x_3 =$
$x_4 =$
$x_5 =$
$x_6 =$
$x_7 =$
$x_8 =$
$x_9 =$
$x_{10} = 0.0035887$
$x_{11} = 0.0142927$
$x_{12} = 0.0571502$
$x_{13} = 0.2285939$
$x_{14} = 0.9143735$
$x_{15} = 3.657493$

De (Numerical analysis - Kincaid, Cheney), p. 48

Mal condicionamento

- Problema mal condicionado: problema que não depende continuamente dos dados.
- Em um problema mal condicionado, uma leve variação nos dados de entrada pode levar a soluções completamente diferentes.
- Por que isso é importante ?
  - (Dados são provenientes de medidas, observações, etc. e estão sujeitos a erros)

Mal condicionamento (exemplo)

$$\begin{cases} x + y = 2 \\ x + 1.01y = 2.01 \end{cases}$$