Introdução à Análise Numérica 2021.2
Escola de Matemática Aplicada, Fundação Getulio Vargas
Professor Hugo A. de la Cruz Cancino
Monitor Lucas Machado Moschen — Entrega 20/08/2021

Lista 1

Exercício 1 Determine a representação em ponto flutuante (considerando precisão dupla) do número $x = 20.1$.

1. Qual o número de máquina de 64 bits usado para armazenar $fl(x)$ no computador?

   Uma sequência de passos que pode ser considerada é a seguinte:

   (i) Primeiro separamos a parte inteira da fracionária e transformamos ambas em binário: $20 = (10100)_2$ e $0.1 = (0\overline{0011})_2$. Para fazer essa conversão, na parte inteira basta dividir consecutivamente por 2 e agrupar os restos, enquanto a fracionária, basta multiplicar o número por 2 consecutivamente e tomar a parte inteira dos produtos.

   Assim, temos que $(20.1)_{10} = (10100.0\overline{0011})_2$.

   (ii) Escrevemos o número em "notação científica", isto é, deixando apenas um 1 antes da vírgula,

   $$(10100.0\overline{0011})_2 = 1.01000\overline{0011} \cdot 2^4$$

   (iii) Nesse caso, temos que o sinal é positivo e o expoente é 4. Adicionamos o viés de 1023 e obtemos que o expoente com viés será $1027 = 2^{10} + 2^2 + 2^0 = (10000000011)_2$.

   (iv) Agora basta verificar o último dígito da mantissa. Como 0.1 na base binária tem representação infinita, vamos precisar cortar. Temos 5 dígitos antes da repetição 0011. Como temos 52 espaços, faltam 47 para preencher e, então, teremos 11 vezes repetindo 0011 e, na últimas três vagas, 001. Como depois disso temos um 1, teremos que adicionar 1 a mantissa, ficando com 010 na últimas posições.

   Logo $[fl(20.1)]_{IEE754} = 0|100\ 000\ 000\ 11|01000\ 0011\ \ldots\ 0011\ 010$.

2. Determine o valor exato do erro arredondado. Ou seja, determine: $20.1 - fl(20.1)$

$$\begin{aligned}
fl(20.1) - 20.1 &= 1.010000011\ldots0011010 \cdot 2^4 - 1.010000011\ldots00110011\cdots \cdot 2^4 \\
&= 1010000011\ldots0011010 \cdot 2^{-48} - 1010000011\ldots0011001.1\overline{0011} \cdot 2^{-48} \\
&= 10 \cdot 2^{-48} - 1.1\overline{0011} \cdot 2^{-48} \\
&= 2 \cdot 2^{-48} - (1 + 0.5 + 0.1) \cdot 2^{-48} \\
&= 0.4 \cdot 2^{-48} = 0.1 \cdot 2^{-46}.
\end{aligned} \tag{1}$$

Exercício 2 Determine o equivalente decimal dos seguintes números de máquina em ponto flutuante.

Nesses exercícios, a solução é do tipo,

$$(-1)^s 2^{[c]_{10}-1023} \cdot (1+[m]_{10}),$$

em que s, c e m são os retângulos, respectivamente, com exceção dos subnormais. Claro que temos que converter os números em decimal. Um arquivo python resolve esses problemas a seguir no Github[1]. As soluções são, respectivamente,

$$-3224, 1.32421875, 2.885642660251527e^{-308}, \text{-Inf} \text{ e } \text{NaN}.$$

1. $\boxed{1 \mid 10000001010 \mid 1001001000000000000000000000000000000000000000000000}$

2. $\boxed{0 \mid 01111111111 \mid 0101001000000000000000000000000000000000000000000000}$

3. $\boxed{0 \mid 00000000000 \mid 0101001000000000000000000000000000000000000000000000}$

4. $\boxed{1 \mid 11111111111 \mid 0000000000000000000000000000000000000000000000000000}$

5. $\boxed{1 \mid 11111111111 \mid 0000000000000000000000000000000000000000000000001111}$

6. Determine os próximos números de máquina para os números fornecidos nos itens anteriores, e escreva os mesmos na forma decimal.

   A ideia nesse exercício é adicionar $+1$ no último termo. Em particular, os últimos dos itens são NaN. No primeiro e segundo item, adicionamos $2^{[c]_{10}-1023-52}$. No terceiro item, adicionamos $2^{x+1-1023-52}$ em que x é o número de *leading zeros*.

Exercício 3 Converte em binário ou converte em decimal, segundo seja o caso, e determine $fl(x)$:

O arquivo python mencionado também faz essas contas.

1. $x = 1/4$

   O binário é 0.01.

   $fl\left(\frac{1}{4}\right) = 0\ 01111111101\ 0000000000000000000000000000000000000000000000000000.$

2. $x = 1/3$

   O binário é $0.\overline{01}$.

   $fl\left(\frac{1}{3}\right) = 0\ 01111111101\ 0101010101010101010101010101010101010101010101010101.$

3. $x = 2/3$

   O binário é $1.\overline{01}$.

   $fl\left(\frac{2}{3}\right) = 0\ 01111111110\ 0101010101010101010101010101010101010101010101010101.$

[1] `https://github.com/lucasmoschen/ta-sessions/tree/master/Numerical_Analysis/lists/list1`

4. $x = 0.9$

   O binário é $0.1\overline{1100}$.

   $fl(0.9) = 0\ 01111111110\ 1100110011001100110011001100110011001100110011001101$

5. $x = 0.\overline{1000111}$

   O decimal é $71/127$.

   $fl\left(\frac{71}{127}\right) = 0\ 01111111110\ 0011110001111000111100011110001111000111100011111001$

6. $x = 0.101\overline{100011}$

   O decimal é $125/252$.

   $f_l\left(\frac{125}{252}\right) = 0\ 01111111101\ 1111101111110111111011111101111110111111011111110000$

Exercício 4 Para quais $k \in \mathbb{N}$ o número $5 + 2^k$ é representado de forma exata no computador.

Temos que $5 + 2^k = 2^k + 2^2 + 2^0 = 10\ldots0101$, que o $k+1$-th digito é 1. Assim,

$$[5 + 2^k]_{10} = [1.00\ldots0101]_2 \cdot 2^k.$$

O sinal desse número é 0 e o expoente será o valor binário de $k - 1023$. A mantissa terá tamanho k mais uma sequência de 0 para completar as 52 casas. Logo, queremos que $0 \leq k \leq 52$.

Exercício 5 Considere a equação recursiva

$$x_{n+1} = \frac{22}{7}x_n - \frac{3}{7}x_{n-1}; \qquad x_0 = 1, \quad x_1 = \frac{1}{7} \tag{2}$$

1. Demonstre que a equação acima tem solução $x_n = \left(\frac{1}{7}\right)^n$.

   Defina $S_n = [x_{n+1}, x_n]$. Teremos que $S_0 = [1/7, 1]$. Também defina

   $$A = \begin{bmatrix} 22/7 & -3/7 \\ 1 & 0 \end{bmatrix},$$

   em que $S_{n+1} = AS_n$. Portanto, teremos que $S_n = A^n S_0$. Para isso, vamos diagonalizar a matriz A. Os autovalores de A são soluções da equação característica:

   $$7\lambda^2 - 22\lambda + 3 = 0,$$

   cujas soluções são $\lambda_1 = 3$ e $\lambda_2 = 1/7$. Se P é a matriz de autovetores, teremos que

   $$S_n = P \begin{bmatrix} 3^n & 0 \\ 0 & 7^{-n} \end{bmatrix} P^{-1} S_0,$$

   isto é, $x_n = A3^n + B7^{-n}$, de forma que $x_0 = A + B$ e $x_1 = 3A + B/7$. Resolvendo o sistema de equações, obtemos que $A = 0$ e $B = 1$, isto é,

   $$x_n = \left(\frac{1}{7}\right)^n.$$

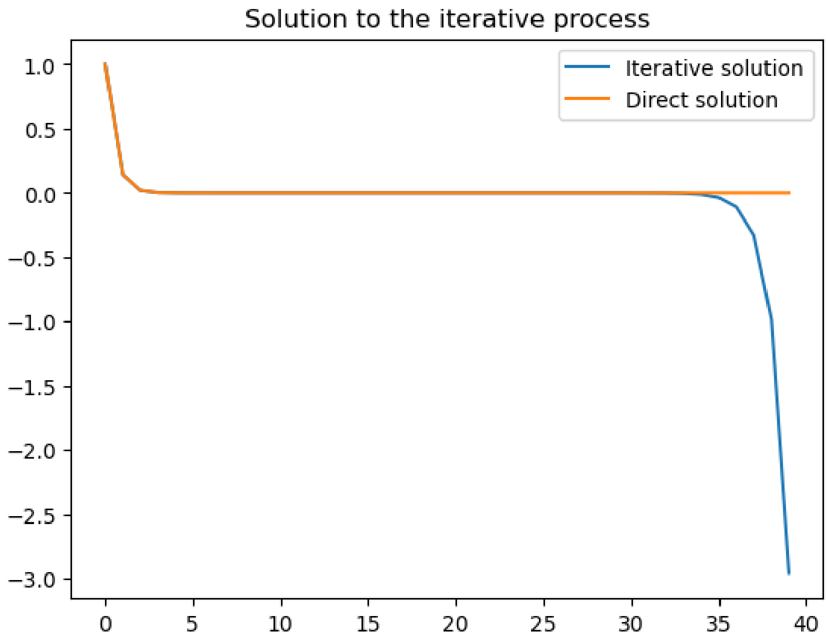

Figura 1: Comparação dos valores de x para a implementação computacional e usando a solução direta.

2. Implemente o processo iterativo (2) para calcular x_n.

   A implementação em Python se encontra no mesmo link no Github.

3. Compare para diferentes valores de n os valores de x_n obtidos usando a solução em a) e usando a implementação computacional feita em b). Por que a partir de um certo valor de n os valores são completamente diferentes? Reflita sobre isso!

   Considere a figure 1 para comparar os valores de x. Observe que usando a versão iterativa anda com a solução direta, mas a partir de um dado instante, expressão diverge para $-\infty$. Observamos que existe um dado momento em que a solução troca de sinal. Essa troca de sinal, faz com que as soluções subsequentes sejam sempre negativas e em diante elas continuem reduzindo, infinitamente.

4. Faça uma análise de estabilidade do algoritmo implementado em b) para calcular x_n.

   Suponha que $x_1 = 1/7 + \delta$, em que δ é, em geral, um pequeno número. Nesse caso, temos que a solução será

$$x_n = \frac{7}{20}\delta \cdot 3^n + \left(1 + \frac{7}{20}\delta\right)\left(\frac{1}{7}\right)^n = \left(\frac{1}{7}\right)^n + \frac{7}{20}\delta(3^n + 7^{-n}).$$

   Nesse caso, se $\delta < 0$, a solução converge para $-\infty$ (que é o caso do exercício anterior) e, se $\delta > 0$, a solução converge para $+\infty$.

Exercício 6 Faça as seguintes operações (que dão zero como resultado na aritmética exata) em "IEEE double precision computer arithmetic", usando a "Rounding to Nearest Rule"e verifique sua resposta no computador.

(a) $(8.3 - 7.3) - 1$ Temos que $8.3 = 2^{1024+2-1023}(1 + (0.00001\overline{0011})_2)$ que no computador é armazenado como

$$2^{1024+2-1023}(1 + (0.000010011 \cdots 010)_2).$$

Já $7.3 = 2^{1024+1-1023}(1 + (0.1101\overline{0011})_2)$ que no computador se armazena como

$$2^{1024+1-1023}(1 + (0.11010011 \cdots 0011)_2).$$

Subtraindo esses valores, temos que, no computador

$$8.3 - 7.3 = 2^3(1 + (0.000010011 \cdots 0011010)_2) - 2^2(1 + (0.11010011 \cdots 00110011)_2)$$

e, então,

$$8.3 - 7.3 = 4 + (0.010011 \cdots 0011010)_2 - (11.010011 \cdots 00110011)_2 = 1 + 2^{-50}$$

Logo $(8.3 - 7.3) - 1 = 2^{-50}$.

(b) $(8.4 - 7.4) - 1$

(c) $(8.8 - 7.8) - 1$