Cálculo Numérico Computacional

Aula 03

Erros Absolutos e Relativos
Arredondamento e Truncamento
Aritmética de Ponto Flutuante
Instabilidade Numérica

Prof. Lucas Zanovello lucas.tahara@unesp.br

Cálculo Numérico Computacional

Agenda:

1. Breve revisão da aula passada;
2. Erros absolutos e relativos;
3. Arredondamento e truncamento;
4. Operações aritméticas de ponto flutuante;
5. Instabilidade numérica;
6. Encerramento.

Cálculo Numérico Computacional

1. Breve revisão da aula passada

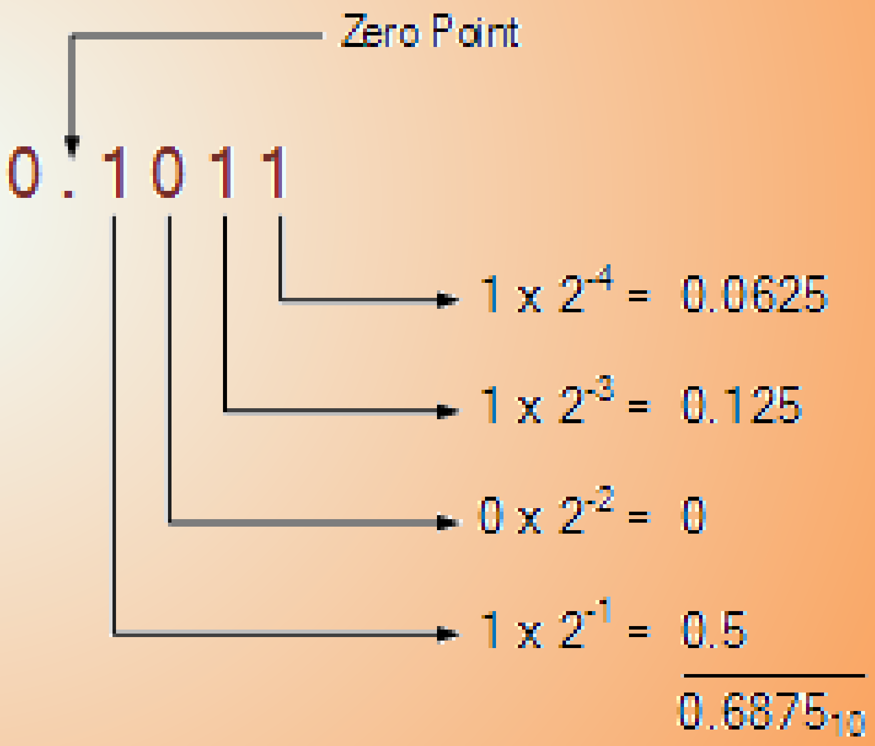
- Os resultados dependem da precisão de dados de entrada, da representação dos dados no computador e das operações numéricas efetuadas;
- A máquina usa um sistema de representação diferente do nosso, e por isso todos os cálculos possuem erros inerentes ao processo.

Cálculo Numérico Computacional

Base 2 para base 10 (inteiros)

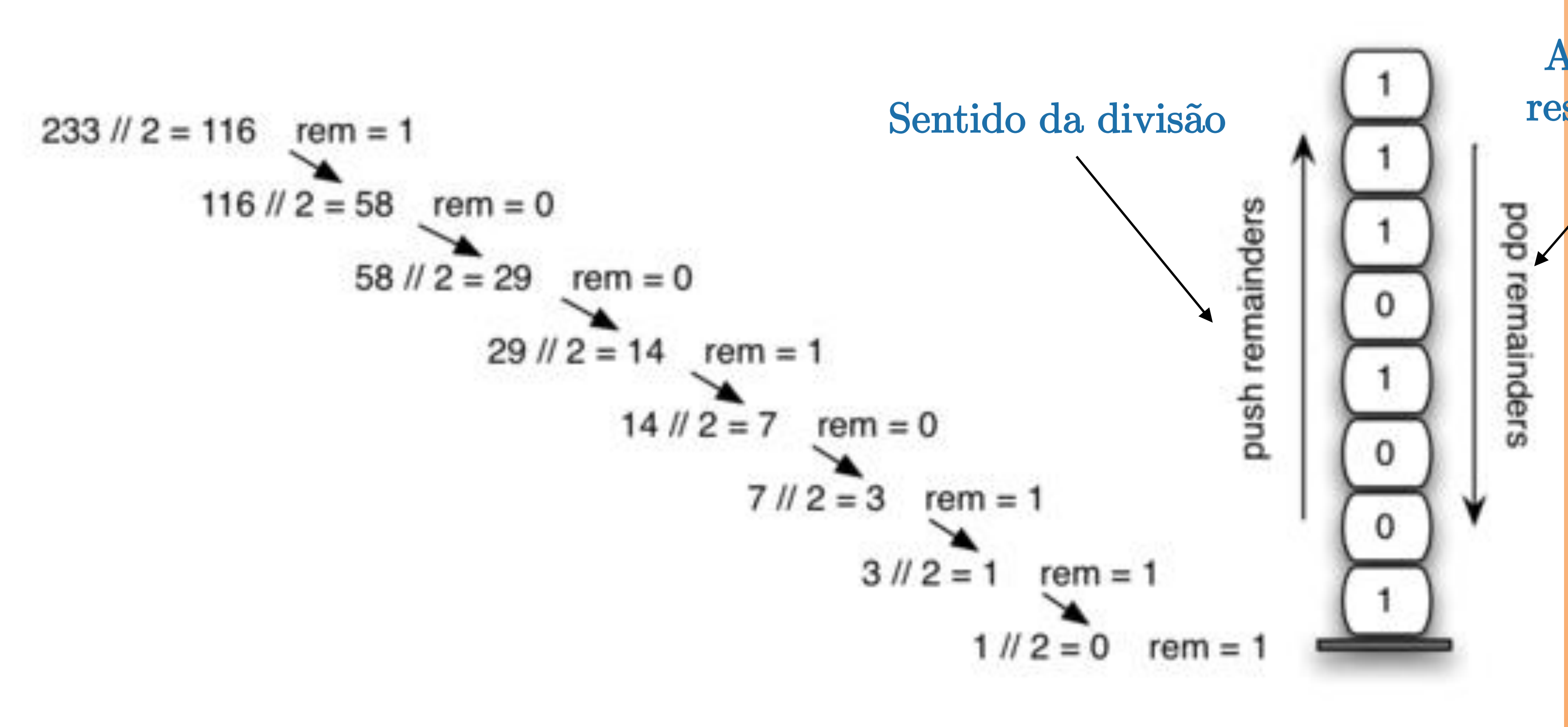
$(10011)_2 = 1\times 2^4 + 0 \times 2^3 + 0 \times 2^2 + +1 \times 2^1 + 1 \times 2^0 = 16 + 0 + 0 + 2 + 1 = 19$

Base 2 para base 10 (fracionários)



Cálculo Numérico Computacional

Base 10 para base 2 (inteiros) – divisões sucessivas



Sentido da divisão

Apresentação do resultado - binário

Cálculo Numérico Computacional

Base 10 para base 2 (fracionários) – multiplicações sucessivas

0.8125

0.8125 × 2	1.6250	1	0.625
0.625 × 2	1.250	1	0.250
0.25 × 2	0.50	0	0.5
0.5 × 2	1.00	1	0

1

Cálculo Numérico Computacional

Alguns números não possuem representação exata. Exemplo:

$$(0{,}6)_{10} = (0{,}100110011001\ldots)_2$$

Erro de conversão de base

Cálculo Numérico Computacional

Um número real, no sistema da máquina, é representado na forma:

$$\pm, (.\boldsymbol{d_1 d_2 \dots d_t}) \times \boldsymbol{\beta^e} \quad \text{ou} \quad (-\mathbf{1})^{s} \times (.\boldsymbol{d_1 d_2 \dots d_t}) \times \boldsymbol{\beta^e}$$

Precisão simples

$\pm$	$\boldsymbol{e_1 e_2 \dots e_8}$	$\boldsymbol{d_1 d_2 \dots d_{23}}$

Precisão dupla

$\pm$	$\boldsymbol{e_1 e_2 \dots e_{11}}$	$\boldsymbol{d_1 d_2 \dots d_{52}}$

Cálculo Numérico Computacional

$$F(\beta, t, m, M)$$

$\boldsymbol{\beta}$: base utilizada;
$\boldsymbol{t}$: tamanho ou número de dígitos (precisão);
$\boldsymbol{m}$ e $\boldsymbol{M}$: menor e maior expoentes, respectivamente.

$$F(2{,}8,-4{,}3)$$

$\mathbf{x} =$

0	**010**	**11100110**

$$\mathbf{x} = (-1)^0 \times 2^2 \times (0.11100110) = (11.100110)_2 = (3.59375)_{10}$$

Cálculo Numérico Computacional

Exemplo 3

Dar a representação dos números a seguir num sistema de aritmética de ponto flutuante de três dígitos para $\beta = 10$, $m = -4$ e $M = 4$.

x	Representação obtida por arredondamento	Representação obtida por truncamento
1.25	0.125×10	0.125×10
10.053	0.101×10^2	0.100×10^2
–238.15	-0.238×10^3	-0.238×10^3
2.71828...	0.272×10	0.271×10
0.000007	(expoente menor que –4)	=
718235.82	(expoente maior que 4)	=

Cálculo Numérico Computacional

2. Erros absolutos e relativos

- **Erro absoluto**: diferença entre o valor exato de um número $\boldsymbol{x}$ e seu valor aproximado $\bar{\boldsymbol{x}}$:

$$\boldsymbol{EA_x = x - \bar{x}}$$

Normalmente, somente $\bar{\boldsymbol{x}}$ é conhecido, então trabalha-se com tolerâncias ou intervalos de tolerância.

Cálculo Numérico Computacional

Exemplo:

Sabe-se que $\boldsymbol{\pi} \in (\mathbf{3.14}, \mathbf{3.15})$, então tomando a aproximação

$$|\boldsymbol{EA_{\pi}}| = |\boldsymbol{\pi} - \bar{\boldsymbol{\pi}}| < \mathbf{0.01}$$

temos uma tolerância para o resultado encontrado.

Cálculo Numérico Computacional

Mas, e se...

- $\bar{\boldsymbol{x}} = \mathbf{2112.9}$ com $|\boldsymbol{EA_x}| < \mathbf{0.1}$
- $\bar{\boldsymbol{y}} = \mathbf{5.3}$ com $|\boldsymbol{EA_y}| < \mathbf{0.1}$

Os dois números estão representados com a mesma precisão?

Tudo depende da ordem de grandeza. E nesse ponto, o erro absoluto não é suficiente para descrever esta precisão.

Cálculo Numérico Computacional

- Erro relativo: erro absoluto dividido pelo valor aproximado:

$$ER_x = \frac{EA_x}{\bar{x}} = \frac{x - \bar{x}}{\bar{x}}$$

Do exemplo anterior, com $|EA_x| = |EA_y| = 0.1$, $\bar{x} = 2112.9$ e $\bar{y} = 5.3$, temos que:

$$|ER_x| = \frac{|EA_x|}{|\bar{x}|} < \frac{0.1}{2112.9} \approx 4.7 \times 10^{-5}$$

Cálculo Numérico Computacional

$$|ER_y| = \frac{|EA_y|}{|\bar{y}|} < \frac{0.1}{5.3} \approx 0.02$$

Assim, o número x é representado com maior precisão que o número y.

Cálculo Numérico Computacional

3. Arredondamento e truncamento

Vamos considerar um sistema que opera com $\boldsymbol{t}$ dígitos na base 10, e $\boldsymbol{x}$ escrito na forma:

$$\boldsymbol{x = f_x \times 10^{e} + g_x \times 10^{e-t}}$$

onde

$$\boldsymbol{0.1 \leq f_x < 1 \text{ e } 0 \leq g_x < 1}$$

Cálculo Numérico Computacional

Exemplo, para $\boldsymbol{t = 4}$ e $\boldsymbol{x = 234.57}$

$$\boldsymbol{x = 0.2345 \times 10^3 + 0.7 \times 10^1}$$

Obviamente a parcela $\boldsymbol{g_x \times 10^{e-t}}$, que vale $\boldsymbol{0.7 \times 10^1}$ não pode ser incorporada na mantissa neste caso. Então, como considerar e definir o erro máximo cometido?

Cálculo Numérico Computacional

Na visão do truncamento, a parcela $\boldsymbol{g_x \times 10^{e-t}}$ é desprezada, e $\boldsymbol{\bar{x} = f_x \times 10^e}$. Avaliando os erros:

- $\boldsymbol{|EA_x| = |x - \bar{x}| = |g_x| \times 10^{e-t} < 10^{e-t}}$, pois $\boldsymbol{|g_x| < 1}$

- $\boldsymbol{|ER_x| = \dfrac{|EA_x|}{|\bar{x}|} = \dfrac{|g_x| \times 10^{e-t}}{|f_x| \times 10^e} < \dfrac{10^{e-t}}{0.1 \times 10^e} = 10^{-t+1}}$

pois $\boldsymbol{0.1}$ é o menor valor possível para $\boldsymbol{f_x}$

Cálculo Numérico Computacional

Na visão do arredondamento, modifica-se $\boldsymbol{f_x}$ e leva-se em consideração a parcela $\boldsymbol{g_x}$. Normalmente se utiliza o arredondamento simétrico:

$$\overline{x} = \begin{cases} f_x \times 10^e & \text{se } |g_x| < \frac{1}{2} \\ f_x \times 10^e + 10^{e-t} & \text{se } |g_x| \geq \frac{1}{2} \end{cases}$$

Cálculo Numérico Computacional

Para $|g_x| < \frac{1}{2}$:

- $|EA_x| = |x - \bar{x}| = |g_x| \times 10^{e-t} < \frac{1}{2} \times 10^{e-t}$

- $|ER_x| = \frac{|EA_x|}{|\bar{x}|} = \frac{|g_x| \times 10^{e-t}}{|f_x| \times 10^{e}} < \frac{0.5 \times 10^{e-t}}{0.1 \times 10^{e}} = \frac{1}{2} \times 10^{-t+1}$

Cálculo Numérico Computacional

Para $|g_x| \geq \frac{1}{2}$:

- $|EA_x| = |x - \bar{x}| = |(f_x \times 10^e + g_x \times 10^{e-t}) - (f_x \times 10^e + 10^{e-t})| = |g_x \times 10^{e-t} - 10^{e-t}| = |g_x - 1| \times 10^{e-t} \leq \frac{1}{2} \times 10^{e-t}$

- $|ER_x| = \frac{|EA_x|}{|\bar{x}|} \leq \frac{\frac{1}{2}\times 10^{e-t}}{|f_x \times 10^e + 10^{e-t}|} < \frac{\frac{1}{2}\times 10^{e-t}}{|f_x| \times 10^e} < \frac{\frac{1}{2}\times 10^{e-t}}{0.1 \times 10^e} = \frac{1}{2} \times 10^{-t+1}$

Cálculo Numérico Computacional

Em ambos os casos, tem-se:

$$|EA_x| \leq \frac{1}{2} \times 10^{e-t}$$

$$|ER_x| < \frac{1}{2} \times 10^{-t+1}$$

E apesar dos erros serem menores, o tempo para execução é maior, e dessa forma o truncamento é mais utilizado.

Cálculo Numérico Computacional

4. Operações aritméticas de ponto flutuante

- Em uma sequência de operações, é importante a noção da propagação do erro;

- Exemplo: consideramos um sistema de aritmética de ponto flutuante com quatro dígitos, base 10 e acumulador de precisão dupla.

$$\boldsymbol{x = 0.937 \times 10^4} \quad \text{e} \quad \boldsymbol{y = 0.1272 \times 10^2}$$

Cálculo Numérico Computacional

Vamos calcular a operação $\boldsymbol{x + y}$:

1. Escolher o número com menor expoente entre x e y e deslocar sua mantissa para a direita um número de dígitos igual à diferença absoluta entre os respectivos expoentes.

$$\boldsymbol{x = 0.937 \times 10^4} \qquad \boldsymbol{y = 0.1272 \times 10^2}$$

Assim, sendo a diferença entre expoentes igual a 2, seguimos com o passo 2.

Cálculo Numérico Computacional

2. Colocar o expoente do resultado igual ao maior expoente entre x e y.

$$y = 0.001272 \times 10^{4}$$

3. Executar a adição/subtração das mantissas e determinar o sinal do resultado.

$$x + y = (0.937 + 0.001272) \times 10^{4} = 0.938272 \times 10^{4}$$

Cálculo Numérico Computacional

4. Normalizar o valor do resultado, se necessário;
5. Arredondar o valor do resultado, se necessário;
6. Verificar se houve *overflow/underflow*.

Arredondamento: $\overline{\boldsymbol{x}+\boldsymbol{y}} = \mathbf{0.9383} \times \mathbf{10^4}$

Truncamento: $\overline{\boldsymbol{x}+\boldsymbol{y}} = \mathbf{0.9382} \times \mathbf{10^4}$

Cálculo Numérico Computacional

Vamos agora calcular a operação $\boldsymbol{x}\,.\,\boldsymbol{y}$:

1. Colocar o expoente do resultado igual à soma dos expoentes de x e y.

$$\boldsymbol{x.y = \left(0.937 \times 10^{4}\right) \times \left(0.1272 \times 10^{2}\right) = (0.937 \times 0.1272) \times 10^{6}}$$

2. Executar a multiplicação das mantissas e determinar o sinal do resultado.

$$\boldsymbol{x.y = 0.1191864 \times 10^{6}}$$

Cálculo Numérico Computacional

3. Normalizar o valor do resultado, se necessário;
4. Arredondar o valor do resultado, se necessário;
5. Verificar se houve *overflow/underflow*.

Arredondamento: $\overline{\boldsymbol{x.y}} = \mathbf{0.1192} \times \mathbf{10^6}$

Truncamento: $\overline{\boldsymbol{x.y}} = \mathbf{0.1191} \times \mathbf{10^6}$

Cálculo Numérico Computacional

Adição / subtração

- Escolher o número com menor expoente entre x e y e deslocar sua mantissa para a direita um número de dígitos igual à diferença absoluta entre os respectivos expoentes;
- Colocar o expoente do resultado igual ao maior expoente entre x e y;
- Executar a adição/subtração das mantissas e determinar o sinal do resultado;
- Normalizar o valor do resultado, se necessário;
- Arredondar o valor do resultado, se necessário;
- Verificar se houve *overflow/underflow*.

Cálculo Numérico Computacional

Multiplicação

- Colocar o expoente do resultado igual à soma dos expoentes de x e y;
- Executar a multiplicação das mantissas e determinar o sinal do resultado;
- Normalizar o valor do resultado, se necessário;
- Arredondar o valor do resultado, se necessário;
- Verificar se houve *overflow/underflow*.

Divisão

- Colocar o expoente do resultado igual à diferença dos expoentes de x (dividendo) e y (divisor);
- Executar a divisão das mantissas e determinar o sinal do resultado;
- Normalizar o valor do resultado, se necessário;
- Arredondar o valor do resultado, se necessário;
- Verificar erros.

Cálculo Numérico Computacional

- Normalmente, o resultado exato é normalizado e então arredondado ou truncado para t dígitos, e assim armazenado na memória;

- Dependendo do tipo de operação, existe um erro inerente ao processo, e isso pode ser definido analiticamente.

Cálculo Numérico Computacional

- Adição:
- Erro absoluto: $\boldsymbol{EA_{x+y} = EA_x + EA_y}$
- Erro relativo: $\boldsymbol{ER_{x+y} = ER_x \left(\frac{\bar{x}}{\bar{x}+\bar{y}}\right) + ER_y \left(\frac{\bar{y}}{\bar{x}+\bar{y}}\right)}$

*Demonstração na lousa/livro.

Cálculo Numérico Computacional

- Subtração:
- Erro absoluto: $\boldsymbol{EA_{x-y} = EA_x - EA_y}$
- Erro relativo: $\boldsymbol{ER_{x-y} = ER_x\left(\frac{\bar{x}}{\bar{x}-\bar{y}}\right) - ER_y\left(\frac{\bar{y}}{\bar{x}-\bar{y}}\right)}$

*Demonstração na lousa/livro.

Cálculo Numérico Computacional

- Multiplicação:

- Erro absoluto: $EA_{x.y} \approx \bar{x}EA_y + \bar{y}EA_x$

- Erro relativo: $ER_{x.y} \approx ER_x + ER_y$

*Demonstração na lousa/livro.

Cálculo Numérico Computacional

- Divisão:
- Erro absoluto: $EA_{x/y} \approx \frac{\bar{y}EA_x - \bar{x}EA_y}{\bar{y}^2}$
- Erro relativo: $ER_{x/y} \approx ER_x - ER_y$

*Demonstração na lousa/livro.

EXERCÍCIOS

Cálculo Numérico Computacional

5. Instabilidade numérica

Além dos problemas dos erros causados pelas operações aritméticas, existem alguns efeitos numéricos que também contribuem para um resultado questionável:

- Cancelamento;
- Propagação do erro;
- Mal condicionamento;
- Instabilidade numérica.

*Mais detalhes podem ser encontrados em FRANCO, N. B., Cálculo numérico. Pearson Prentice Hall, São Paulo, 2007.

Cálculo Numérico Computacional

- Se um resultado intermediário de um cálculo é contaminado com um erro de arredondamento, este erro pode influenciar todos os processos subsequentes;

- Isso acaba influenciando no resultado final;

- Em alguns casos, os erros podem se cancelar com outros, ou então são desprezíveis no resultado final, e nestes casos temos algoritmos estáveis;

- No caso de os erros intermediários influenciarem demasiadamente o resultado final, temos a instabilidade numérica.

Cálculo Numérico Computacional

- Basicamente, supondo uma razão de crescimento do erro $R(\epsilon)$ para n operações:

- $R(\epsilon) = c.\epsilon \longrightarrow c$ é uma constante que não depende de n.
  Nesse caso, dizemos que $R(\epsilon)$ é uma razão de crescimento linear.

- $R(\epsilon) = k^n.\epsilon \longrightarrow k > 1$ é uma constante que está relacionada a n.
  Nesse caso, dizemos que $R(\epsilon)$ é uma razão de crescimento exponencial.

- O crescimento linear é normalmente inevitável. Já o processo que apresenta a razão de crescimento exponencial denomina-se processo instável.

Cálculo Numérico Computacional

Exemplo 2.17 - *Resolver a integral:*

$$I_n = e^{-1} \int_0^1 x^n e^x \, dx \, .$$

Solução: Vamos tentar encontrar uma fórmula de recorrência para I_n. Integrando por partes, segue que:

$$\begin{aligned} I_n &= e^{-1} \left\{ [x^n e^x]_0^1 - \int_0^1 n \, x^{n-1} e^x \, dx \right\} \\ &= 1 - n \, e^{-1} \int_0^1 x^{n-1} e^x \, dx \\ &= 1 - n \, I_{n-1} \, . \end{aligned}$$

Assim, obtemos uma fórmula de recorrência para I_n, isto é:

$$I_n = 1 - n \, I_{n-1} \, , \quad n = 1, 2, \ldots \, , \tag{2.4}$$

e desde que:

$$I_0 = e^{-1} \int_0^1 e^x \, dx = e^{-1}(e-1) = 0.6321,$$

é conhecido, podemos, teoricamente, calcular I_n, usando (2.4). Fazendo os cálculos, obtemos:

Referência: FRANCO, N. B., Cálculo numérico. Pearson Prentice Hall, São Paulo, 2007.

Cálculo Numérico Computacional

$$I_0 = 0.6321\ ,\ I_1 = 0.3679\ ,\ I_2 = 0.2642\ ,\ I_3 = 0.2074\ ,$$

$$I_4 = 0.1704\ ,\ I_5 = 0.1480\ ,\ I_6 = 0.1120\ ,\ I_7 = 0.216\ .$$

O resultado obtido para I_7 está claramente errado, desde que:

$$I_7 < e^{-1} \max_{0 \leq x \leq 1} (e^x) \int_0^1 x^n \, dx < \frac{1}{n+1}\ ,$$

isto é, $I_7 < \frac{1}{8} = 0.1250$. Além disso a sequência I_n é uma sequência decrescente. Para ver que a instabilidade existe, vamos supor que o valor de I_0 esteja afetado de um erro ϵ_0. Vamos supor ainda que todos as operações aritméticas subsequentes são calculadas exatamente. Denotando por I_n o valor exato da integral e por $\tilde{I}_n$ o valor calculado assumindo que só existe erro no valor inicial, obtemos que:

$$\tilde{I}_0 = I_0 + \epsilon_0\ ,$$

e assim:

$$\tilde{I}_n = 1 - n\,\tilde{I}_{n-1}\ ,\quad n = 1, 2, \ldots\ . \tag{2.5}$$

Seja r_n o erro, isto é:

$$r_n = \tilde{I}_n - I_n\ .$$

Referência: FRANCO, N. B., Cálculo numérico. Pearson Prentice Hall, São Paulo, 2007.

Cálculo Numérico Computacional

Subtraindo (2.4) de (2.5), segue que:

$$r_n = -n\, r_{n-1}\ , \quad n = 1, 2, \ldots .$$

Aplicando essa fórmula repetidamente, obtemos:

$$r_n = -n r_{n-1} = (-n)^2 r_{n-2} = \ldots = (-n)^n r_0\ ,$$

e portanto

$$r_n = (-n)^n\ \epsilon_0\ ,$$

desde que $r_0 = \epsilon_0$. Assim, a cada passo do cálculo, o erro cresce do fator n. Surge então a pergunta: Como encontrar o valor exato de I_n? Para este caso em particular, observe que: *uma relação de recorrência ser instável na direção crescente de n não impede de ser estável na direção decrescente de n.* Assim, resolvendo (2.4), para I_{n-1}, obtemos:

$$I_{n-1} = \frac{(1 - I_n)}{n}\ . \qquad (2.6)$$

Se usada nessa forma, a relação também precisa de um valor inicial. Entretanto, não é fácil encontrar esse valor pois todo I_n onde $n > 0$ é desconhecido. Mas sabemos que $I_n \to 0$ quando $n \to \infty$. Assim, tomando $I_{20} = 0$ e usando (2.6) para $n = 20, 19, 18, \ldots$, obtemos: $I_7 = 0.1123835$ onde agora todos os dígitos estão corretos. É interessante notar que começando com $I_7 = 0$, obtemos $I_0 = 0.6320$. Isto ocorre porque neste caso o erro está sendo reduzido substancialmente a cada passo, isto é, a cada passo o erro decresce do fator $\frac{1}{n}$.

Referência: FRANCO, N. B., Cálculo numérico. Pearson Prentice Hall, São Paulo, 2007.

Cálculo Numérico Computacional

Próxima aula:

Aula 04

- Métodos diretos: eliminação de Gauss;
- Exercícios.