Matemática Computacional [1] [2]

Prof. Dr. Eliandro R. Cirilo

Universidade Estadual de Londrina - UEL
Centro de Ciências Exatas - CCE
Departamento de Matemática

20 de março de 2024

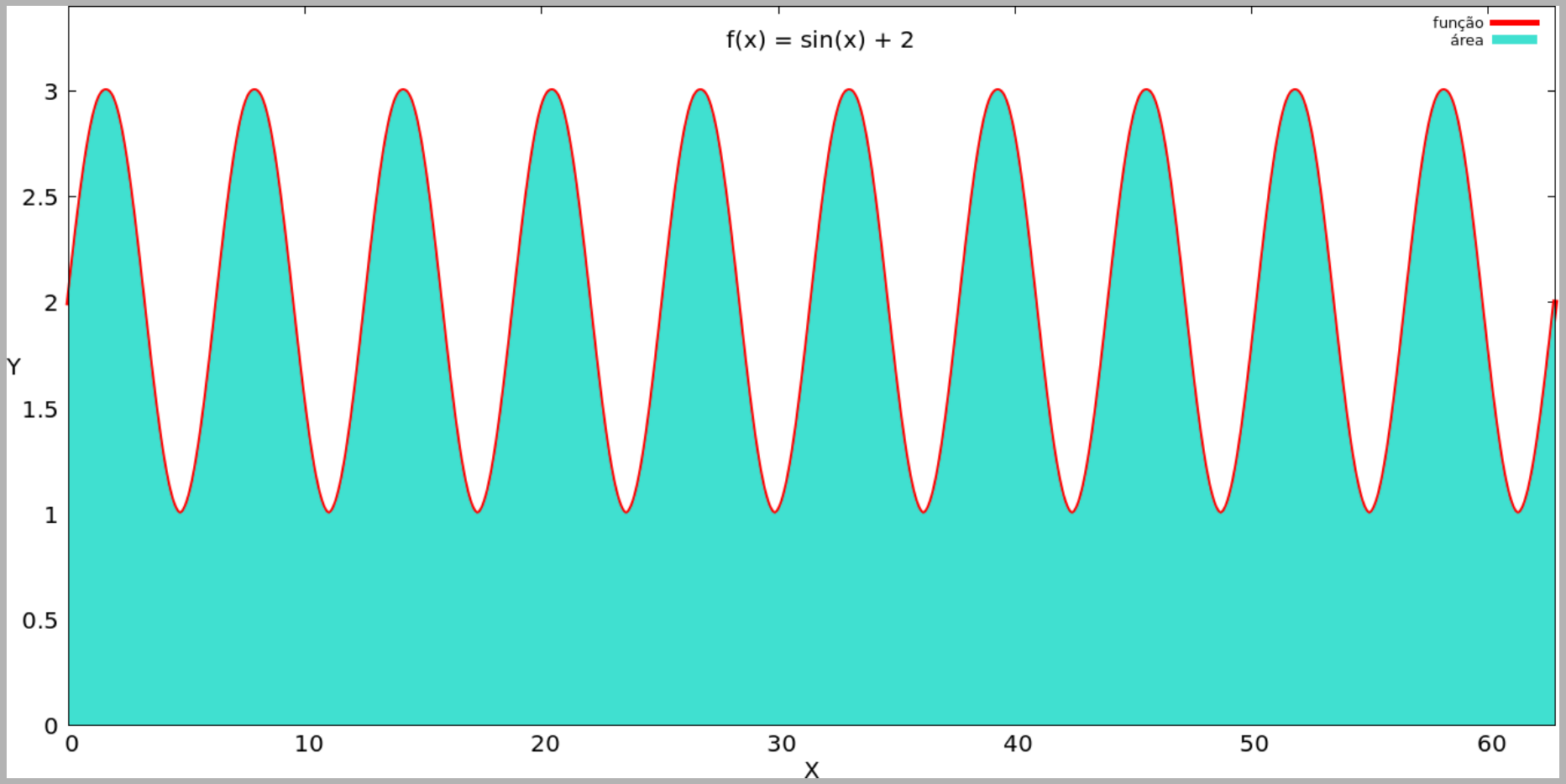
[1] GUIMARÃES, Ângelo de Moura; LAGES, Newton Alberto de Castilho. Algoritmos e estruturas de dados. Rio de Janeiro: LTC, 1994.

[2] MANZANO, Jose Augusto. Algoritmos: lógica para desenvolvimento de programação. São Paulo: Érica, 2000.

Sumário

1. Problemáticas
2. Conceitos Introdutórios
3. Lógica de Programação
4. Vetores, Matrizes, Tipos Derivados
5. Programação em Octave
6. Gráficos
7. Avaliação

Problemáticas: Um simples cálculo de área



- $X = [0 : 20\pi]$
- $\Delta X = 0.1; 0.001; 0.00001; 0.00000000001 \Rightarrow \uparrow$ T.C.[3]

[3] Tempo Computacional

Problemáticas: Multiplicação Matricial

- $$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \\ a_{41} & a_{42} & a_{43} \end{bmatrix} \cdot \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} & b_{15} \\ b_{21} & b_{22} & b_{23} & b_{24} & b_{25} \\ b_{31} & b_{32} & b_{33} & b_{34} & b_{35} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} & c_{15} \\ c_{21} & c_{22} & c_{23} & c_{24} & c_{25} \\ c_{31} & c_{32} & c_{33} & c_{34} & c_{35} \\ c_{41} & c_{42} & c_{43} & c_{44} & c_{45} \end{bmatrix}$$

  $\Leftrightarrow$ 20 elementos calculados

- $A_{40\times30} \cdot B_{30\times50} = C_{40\times50}$

  $\Leftrightarrow$ 2.000 elementos calculados

- $A_{40000\times30000} \cdot B_{30000\times50000} = C_{40000\times50000}$

  $\Leftrightarrow$ 2.000.000.000 elementos calculados $\uparrow T.C.$[4]

[4] Tempo Computacional

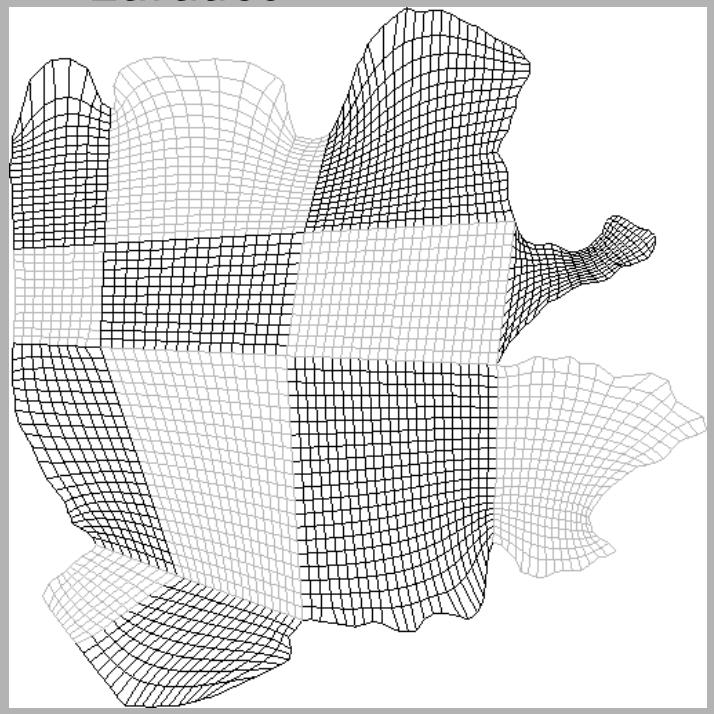
Problemáticas: Equações de Navier-Stokes "$\approx 8 horas$"

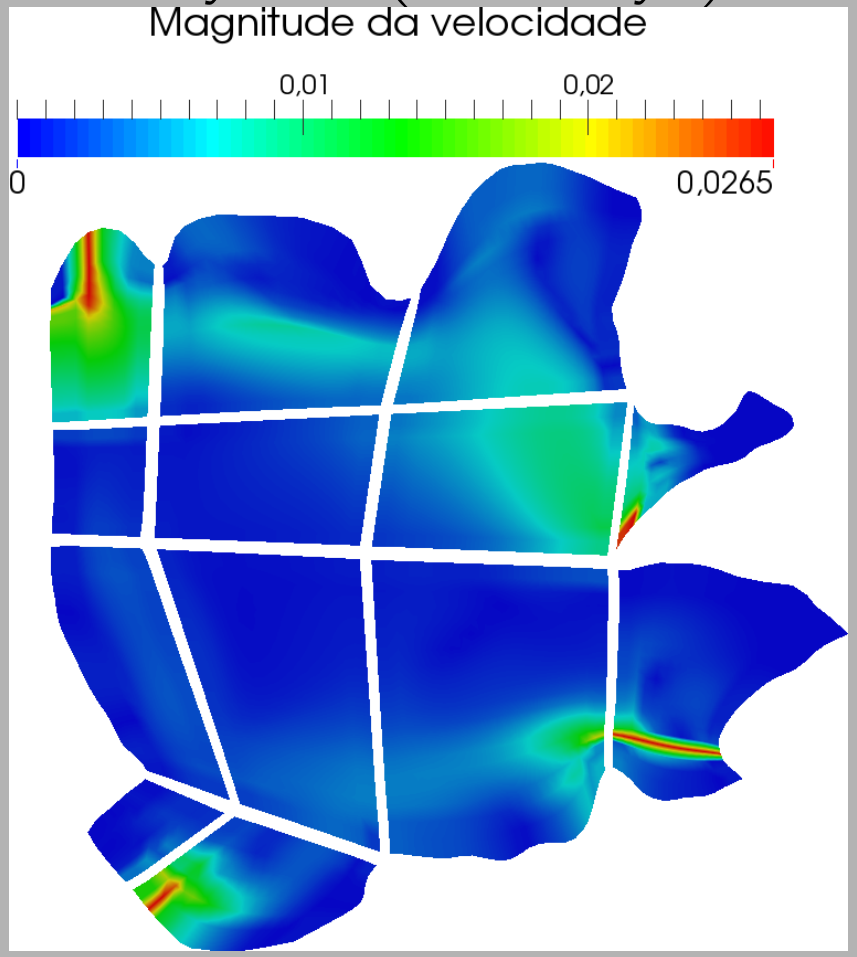
$$\frac{\partial}{\partial t}(\rho u) + \frac{\partial}{\partial x}(\rho u u) + \frac{\partial}{\partial y}(\rho u v) = -\frac{\partial p}{\partial x} + \mu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)$$

$$\frac{\partial}{\partial t}(\rho v) + \frac{\partial}{\partial x}(\rho u v) + \frac{\partial}{\partial y}(\rho v v) = -\frac{\partial p}{\partial y} + \mu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right)$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$$

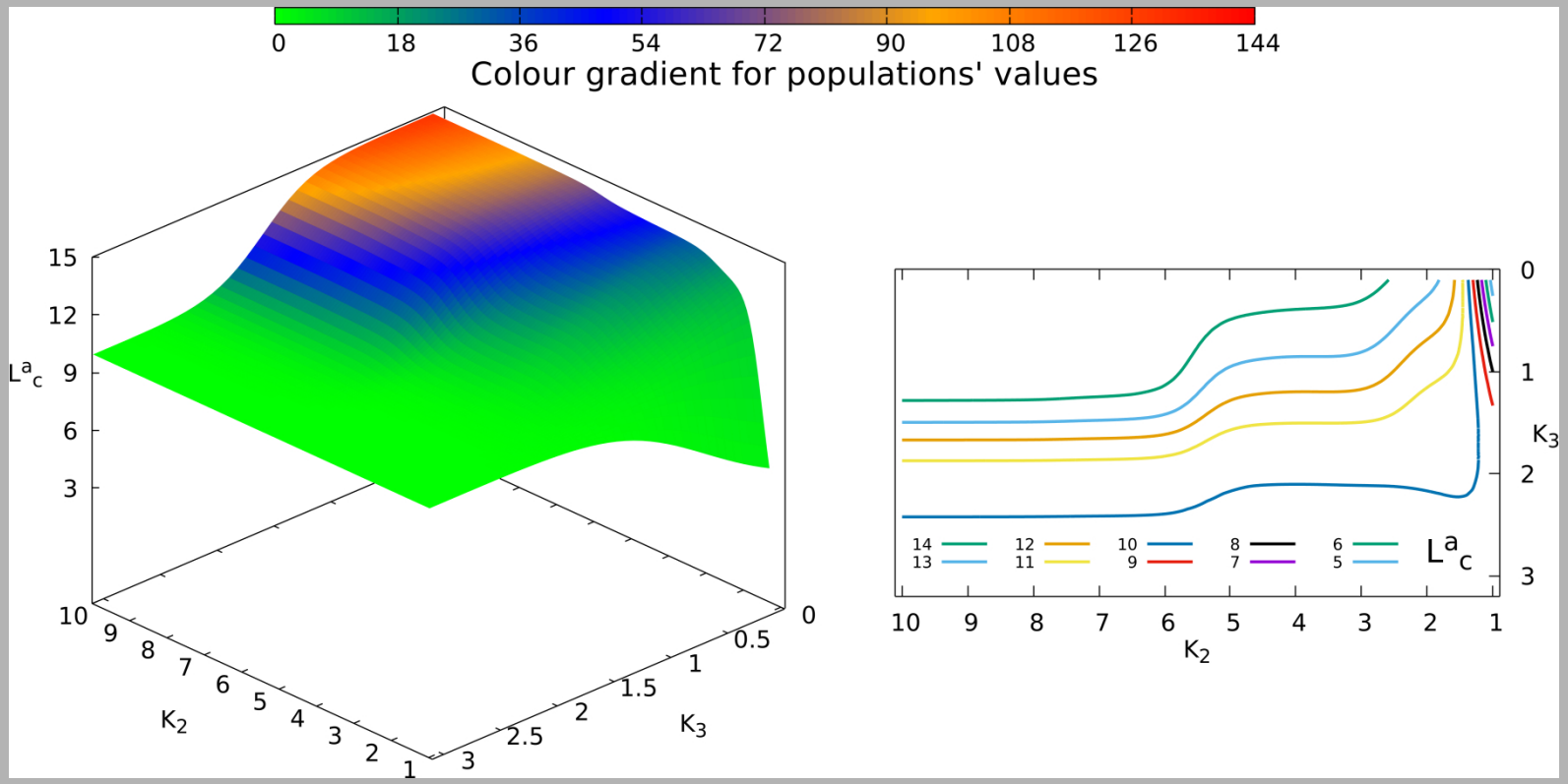
- Luruaco





Problemáticas: Equilíbrio em Mod. Biológico "$\approx 15 dias$"

$$\begin{cases} \tau \dfrac{\partial^2 S}{\partial t^2} + \left[1 - \tau \dfrac{dF(S)}{dS}\right] \dfrac{\partial S}{\partial t} = D \dfrac{\partial^2 S}{\partial x^2} + F(S) & (0, t_{end}] \times (0, L) \\ S(0,x) = S_I \quad ; \quad \dfrac{\partial S(0,x)}{\partial t} = 0 & (0, L) \\ S(t,0) = S(t,L) = 0 & [0, t_{end}] \\ \forall (t,x) \in (0, t_{end}] \times (0, L) \, ; S(t,x) < 0 \rightarrow S(t,x) = 0 & \end{cases}$$



Sumário

1. Problemáticas
2. Conceitos Introdutórios
3. Lógica de Programação
4. Vetores, Matrizes, Tipos Derivados
5. Programação em Octave
6. Gráficos
7. Avaliação

Conceitos Introdutórios

A Ciência da Computação possui várias ramificações com o objetivo de atender necessidades específicas tais como:

- Aplicações Web
- softwares comerciais
- softwares governamentais
- aplicativos para mobile
- **Computação Científica**
- entre muitos outros também .....
  - ▷ https://pt.wikipedia.org/wiki/Computação_científica

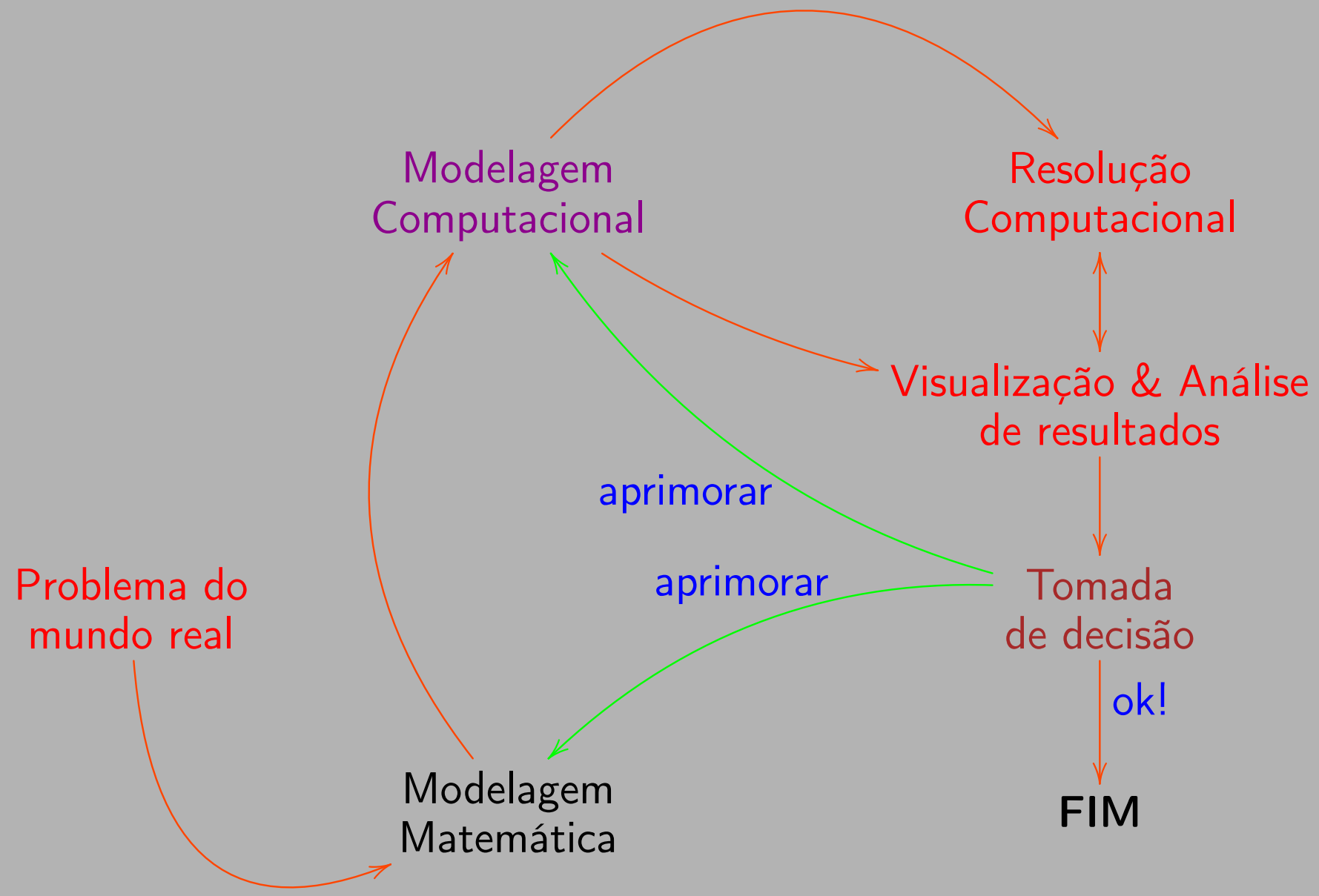
Conceitos Introdutórios

A **Computação Científica** se subdivide em outras partes, e a mais comumente explorada por Matemáticos é a **Programação de Computadores**.

```
! CÁLCULO DO NÚMERO PI

! PI = 4/1 - 4/3 + 4/5 - 4/7 + 4/9 - 4/11 + 4/13 - 4/15 + 4/17 - 4/19 + .......

 SUBROUTINE CAL_PI(N,ID)

 USE MPI

 IMPLICIT NONE

 !DECLARAÇÃO DAS VARIÁVEIS GLOBAIS
 INTEGER,INTENT(IN) ::N,ID

 !DECLARAÇÃO DAS VARIÁVEIS LOCAIS
 INTEGER        ::I
 DOUBLE PRECISION
 ::FINISH,START,TEMPO_SEQ,TEMPO_PAR,NUM_PI_PAR_REDUCE,NUM_PI_SEQ,NUM_PI_PAR

 !DECLARAÇÃO DAS VARIÁVEIS PARA O MPI
 INTEGER :: ERR_BARREIRA,ERR


!SEQUENCIAL

 IF (ID == 1) THEN

    START = MPI_WTIME()

    NUM_PI_SEQ = 0.0

    DO I = 1,N

    NUM_PI_SEQ = NUM_PI_SEQ - ((-1)**I) * (4.0 /(2*I-1))

    END DO

    FINISH = MPI_WTIME()

    TEMPO_SEQ = FINISH - START

 END IF
```

Basicamente, um **Programa de Computador** é um conjunto de instruções (da linguagem natural), codificado em linguagem computacional (obedecendo a uma semântica e Sintaxe) cujo objetivo é resolver um problema.

Conceitos Introdutórios



Conceitos Introdutórios

Modelagem Computacional (Programação Científica)

- **Octave** – programação com limitação em alocação, visualização (simples)
- **Gfortran** – programação sem prévia limitação em alocação (complexa)
- **Gnuplot** – vasta opções de visualizações (complexa)
- **Python** – usada em data science, machine learning, desenv. de web e apps (complexa)
- **R** – muito usada entre estatísticos e analistas de dados (complexa)
- **entre outras** etc ...

Na disciplina nosso foco será apenas em Octave

Em minhas pesquisas uso com mais frequência: Octave, Gfortran e Gnuplot

Sumário

1. Problemáticas
2. Conceitos Introdutórios
3. Lógica de Programação
4. Vetores, Matrizes, Tipos Derivados
5. Programação em Octave
6. Gráficos
7. Avaliação

Lógica de Programação

- A seguir iniciaremos nosso estudo sobre a Lógica de Programação.

- Existem muitos materiais sobre o tema, porém utilizaremos o material intitulado:

Apostila de Lógica de Programação

- A apostila foi desenvolvida por *Isadora Lopes Barbosa Vasconcellos* e suas orientadoras, *Dra. Annabell Del Real Tamariz* e *Dra. Silvia Cristina Freitas Batista*, do Instituto Federal de Educação, Ciência e Tecnologia Fluminense (IFFluminense).

- Ressalta-se que esta apostila está licenciada com uma Licença Creative Commons Atribuição-Não Comercial 4.0 Internacional.

Lógica de Programação

Em lógica de programação abordamos:

Módulo 1 - Algoritmos

Módulo 2 - Variáveis, Constantes e Operadores

Módulo 3 - Estruturas de Seleção

Módulo 4 - Estruturas de Repetição

$\mathcal{A}_1$:

Considerando que:

i são dadas duas notas N_1, N_2 onde a média aritmética semestral é dada por $MS = \dfrac{N_1 + N_2}{2}$

ii se $MS \geq 7$, MS é convertida em média final aprovado MFA

iii se $MS < 3$, MS é convertida em média final reprovado MFR

iv se $3 \leq MS < 7$, necessita-se da terceira nota N_3 (exame)

v a média aritmética com exame é $ME = \dfrac{MS + N_3}{2}$

vi se $ME \geq 7$, ME é convertida em média final aprovado MFA (caso contrário é convertida em média final reprovado MFR)

Construa o Fluxograma e o Pseudocódigo do problema.

$\mathcal{A}_2$:

Análogo ao realizado na $\mathcal{A}_1$, mas considerando uma média aritmética ponderada com os pesos P_1 e P_2, de modo que os pesos sejam definidos no tempo de execução.

$\mathcal{A}_3$:

Construa o fluxograma dos Pseudocódigos das páginas:

- 43 - Algoritmo verifica_numero
- 44 - Algoritmo verifica_numero
- 45/46 - Algoritmo novo_salario
- 49/50 - Algoritmo informa_sexo

$\mathcal{A}_4$:

Construa o fluxograma dos Pseudocódigos das páginas:

- 58 - Algoritmo Imprimir_numeros_de_1_a_100
- 61 - Algoritmo Imprimir_numeros_de_1_a_10
- 64 - Algoritmo Imprimir_numeros_de_1_a_10

Sumário

1. Problemáticas
2. Conceitos Introdutórios
3. Lógica de Programação
4. Vetores, Matrizes, Tipos Derivados
5. Programação em Octave
6. Gráficos
7. Avaliação

Vetores, Matrizes

- Em computação científica é muito comum utilizar a estrutura matricial como recurso para resolução de problemas.

- Uma matriz é um agrupamento de posições na memória do computador cujas mesmas podem ser acessadas por intermédio de um único nome.

$$V = (v_i)_3 = \begin{pmatrix} v_1 & v_2 & v_3 \end{pmatrix} \quad \text{ou} \quad V = (v_{ij})_{1\times 3} = \begin{pmatrix} v_{11} & v_{12} & v_{13} \end{pmatrix}$$

$$B = (b_{ij})_{3\times 3} = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix}$$

$$C = (c_{ijk})_{3\times 4\times 2}$$

$$D = (d_{ijkmnpq})_{2\times 4\times 15\times 9\times 7\times 7\times 7}$$

Algoritmo do produto interno entre vetores

Algoritmo 1: Produto interno entre vetores

```
1 início
    Entrada: A = (a_i)_2, B = (b_i)_2
2   PI ← 0
3   para i = 1...2 faça
4       PI ← PI + a_i * b_i
5   fim
6   retorna PI
7 fim
```

- Teste de mesa com:

$$A = (2 \;\; 3), B = (4 \;\; -1)$$

Algoritmo da soma de duas matrizes

Algoritmo 2: Soma de duas matrizes

```
1 início
    Entrada: M = (m_ij)_{3×3}, N = (n_ij)_{3×3}
2   para i = 1...3 faça
3     para j = 1...3 faça
4       s_ij ← m_ij + n_ij
5     fim
6   fim
7   retorna S = (s_ij)_{3×3}
8 fim
```

- Teste de mesa com:

$$M = \begin{pmatrix} 1 & 3 & -2 \\ 0 & 4 & 5 \\ -4 & -2 & 3 \end{pmatrix} \qquad N = \begin{pmatrix} 1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & 1 \end{pmatrix}$$

Algoritmo da multiplicação por um escalar κ

Algoritmo 3: Multiplicação por um escalar κ

```
1  início
   | Entrada: C = (c_ijk)_{3×4×2}, κ
2  | para i = 1...3 faça
3  | | para j = 1...4 faça
4  | | | para k = 1...2 faça
5  | | | | r_ijk ← κ * c_ijk
6  | | | fim
7  | | fim
8  | fim
9  | retorna R = (r_ijk)_{3×4×2}
10 fim
```

- Teste de mesa com:

$$(c_{ijk})_{3\times4\times2} = i + j - k^2, \kappa = -2$$

Tipos Derivados

- Em computação científica é comum também utilizarmos estruturas mais complexas que as matrizes, denomindas como Tipos Derivados.
- Um Tipo Derivado é uma variável "inventada" pelo desenvolvedor.
- Exemplos:

$$Ponto \quad := \quad Ponto\,(x, y, z)$$

$$\forall\,(x, y, z) \in \mathbb{R}^3$$

$$Aluno \quad := \quad Aluno\left(\text{nome}, \text{código}, \begin{bmatrix} n_1 & n_2 \\ n_3 & M_f \end{bmatrix}\right)$$

$$\text{nome} \in \mathbb{S}, \text{código} \in \mathbb{I}^+, \begin{bmatrix} n_1 & n_2 \\ n_3 & M_f \end{bmatrix} \in \mathbb{M}\,(\mathbb{R})_{2\times 2}$$

Tipos Derivados

- ... continuação de Exemplos:

$$Inf \quad := \quad Inf\left(Aluno, Ponto, f\left(x,y,z\right), \begin{bmatrix} \dfrac{\partial f}{\partial x} & \dfrac{\partial^2 f}{\partial x^2} \\ \\ \dfrac{\partial f}{\partial y} & \dfrac{\partial^2 f}{\partial y^2} \\ \\ \dfrac{\partial f}{\partial z} & \dfrac{\partial^2 f}{\partial z^2} \end{bmatrix}, [a,b], \int_a^b f(t)\,dt \right)$$

$$\text{Aluno} \in \mathbb{A}, \text{Ponto} \in \mathbb{P}, f\left(x,y,z\right) \in \mathbb{R},$$

$$\begin{bmatrix} \dfrac{\partial f}{\partial x} & \dfrac{\partial f}{\partial y} & \dfrac{\partial f}{\partial z} \\ \\ \dfrac{\partial^2 f}{\partial x^2} & \dfrac{\partial^2 f}{\partial y^2} & \dfrac{\partial^2 f}{\partial z^2} \end{bmatrix}^T \in \mathbb{M}\left(\mathbb{F}\right)_{2\times 3}, [a,b] \in \mathbb{IN}, \int_a^b f(t)\,dt \in \mathbb{R}$$

Algoritmo da definição do tipo derivado Ponto

Algoritmo 4: Definição do tipo derivado Ponto

1 **início**
2 | **tipo** *Ponto*
3 | | x, y, z : *real*
4 | **fim**
5 | **var** $po1, po2$: *Ponto*
6 | **var** d : *real*
7 | Definição pelo uso do caractere seletor $(\cdot)$
8 | $po1 \cdot x \leftarrow 1.0$
9 | $po1 \cdot y \leftarrow 2.0$
10 | $po1 \cdot z \leftarrow 2.0$
11 | Definição pelo uso do construtor de estrutura
12 | $po2 \leftarrow (3.0, 1.0, 4.0)$
13 | $d \leftarrow \sqrt{(po2 \cdot x - po1 \cdot x)^2 + (po2 \cdot y - po1 \cdot y)^2 + (po2 \cdot z - po1 \cdot z)^2}$
14 **fim**

$\mathcal{A}_5$:

Considere as matrizes $\boldsymbol{A} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}_{2\times 1}$ e $\boldsymbol{B} = \begin{bmatrix} 4 \\ -1 \end{bmatrix}_{2\times 1}$, que são na verdade vetores coluna. Escreva o fluxograma e pseudocódigo que calcula o produto interno $\langle \boldsymbol{A}, \boldsymbol{B} \rangle$.

$\mathcal{A}_6$:

Considere as matrizes

$$M = \begin{pmatrix} 1 & 3 & -2 \\ 0 & 4 & 5 \\ -4 & -2 & 3 \end{pmatrix}_{3\times 3} \qquad N = \begin{pmatrix} 1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & 1 \end{pmatrix}_{3\times 3}.$$

Escreva o fluxograma e pseudocódigo que calcula a multiplicação entre as matrizes $M_{3\times 3}$ e $N_{3\times 3}$, ou seja $M \,\mathbf{.}\, N$.

$\mathcal{A}_7$:

Escreva o fluxograma e o pseudocódigo que implementa o tipo derivado:

$$Aluno \quad := \quad Aluno\left(\text{nome}, \text{código}, \begin{bmatrix} n_1 & n_2 \\ n_3 & M_f \end{bmatrix}\right)$$

$$\text{nome} \in \mathbb{S}, \text{código} \in \mathbb{I}^+, \begin{bmatrix} n_1 & n_2 \\ n_3 & M_f \end{bmatrix} \in \mathbb{M}(\mathbb{R})_{2\times 2}$$

de modo que tenhamos três entradas ($Aluno_1$, $Aluno_2$, e $Aluno_3$), e que os alunos sejam ordenados em ordem decrescente do valor $M_f = \dfrac{n_1 + n_2 + n_3}{3}$. Escolha convenientemente as respectivas notas n_1, n_2 e n_3 dos alunos, tal que as médias sejam as especificadas no exemplo a seguir.

Exemplo:

Entradas	Saídas
$Aluno_1$ $M_f = 5.7$	$Aluno_3$ $M_f = 8.2$
$Aluno_2$ $M_f = 3.2$	$Aluno_1$ $M_f = 5.7$
$Aluno_3$ $M_f = 8.2$	$Aluno_2$ $M_f = 3.2$

Sumário

1. Problemáticas
2. Conceitos Introdutórios
3. Lógica de Programação
4. Vetores, Matrizes, Tipos Derivados
5. Programação em Octave
6. Gráficos
7. Avaliação

Programação em Octave

- A seguir iniciaremos nosso estudo sobre Programação Científica com o *Octave*.

- Existem muitos materiais sobre o tema, porém utilizaremos o material intitulado:

Apostila de Introdução ao Octave/Matlab [5]

`http://www.uel.br/laboratorios/labsan/arquivosEliandro/OCTAVE_MATLAB.pdf`

GNU Octave [6]

`https://octave.org/octave.pdf`

[5] Material desenvolvido pelo Grupo PET-TELE - UFF.

[6] Material desenvolvido por: John W. Eaton; David Bateman; Søren Hauberg; Rik Wehbring.

Programação em Octave

Na programação em Octave abordamos:

I - os capítulos de 1 a 7

que nos dá uma visão introdutória sobre declarações, execuções diretas, e definições primárias sobre a programação;

II - e o capítulo 10

onde abordamos métodos mais sofisticados e elaborados sobre a programação por meio da construção de funções e *Scrits*;

Sumário

1. Problemáticas
2. Conceitos Introdutórios
3. Lógica de Programação
4. Vetores, Matrizes, Tipos Derivados
5. Programação em Octave
6. Gráficos
7. Avaliação

Gráficos em Octave

- Basicamente, um gráfico pode ser definido como uma representação geométrica que descreve as variações de dados. Tal representação, muitas vezes, busca descrever um fenômeno por meio de seus dados.
- No Octave existem alguns tipos de gráficos e muitas possibilidades de customizações dos mesmos.

Sumário

1. Problemáticas
2. Conceitos Introdutórios
3. Lógica de Programação
4. Vetores, Matrizes, Tipos Derivados
5. Programação em Octave
6. Gráficos
7. Avaliação

1º Avaliação/Dissertativa

Prova escrita agendada para o dia 21/03/2024.

O conteúdo para esta prova é dado pelos temas abordados nos itens 3.1 e 3.2 do conteúdo programático do curso.

2º Avaliação/Orientações

- Utilize o **Overleaf** [7] para escrever sobre a sua avaliação (Projeto).
- Compartilhe a sua 2º avaliação com o professor.
- É necessário o envio do PDF final da sua 2º avaliação na data agendada.
- Use o "Template-Projeto-MATCOMP" para produzir o Projeto.

http://www.uel.br/laboratorios/labsan/perfilEliandro.html

[7] Acesse a plataforma no link – `https://pt.overleaf.com/login`

2º Avaliação/Projeto

Modele, resolva/implemente, e produza um texto técnico sobre ao menos um problema aplicado que esteja aderente a um dos seguintes temas: **Cálculo Diferencial e Integral** ou **Álgebra Linear**.

Para o problema aplicado devem ser exibidas as seguintes seções [8]:

1. Descrição do problema aplicado.
2. Modelagem matemática do problema.
3. Modelagem computacional (fluxograma e/ou pseudocódigo) do problema.
4. Implementação computacional, em Octave, do modelo computacional.
5. Resultados com discussões.
6. Referências bibliográficas.

[8] Use o "Template-Projeto-MATCOMP" como base para o presente projeto.

2º Avaliação/Apresentação

Faça uma apresentação, de até 15 minutos, no dia 21/05/2024.

Nesta apresentação exiba os resultados de seu projeto, conforme a sequência das seções escritas no documento PDF enviado.