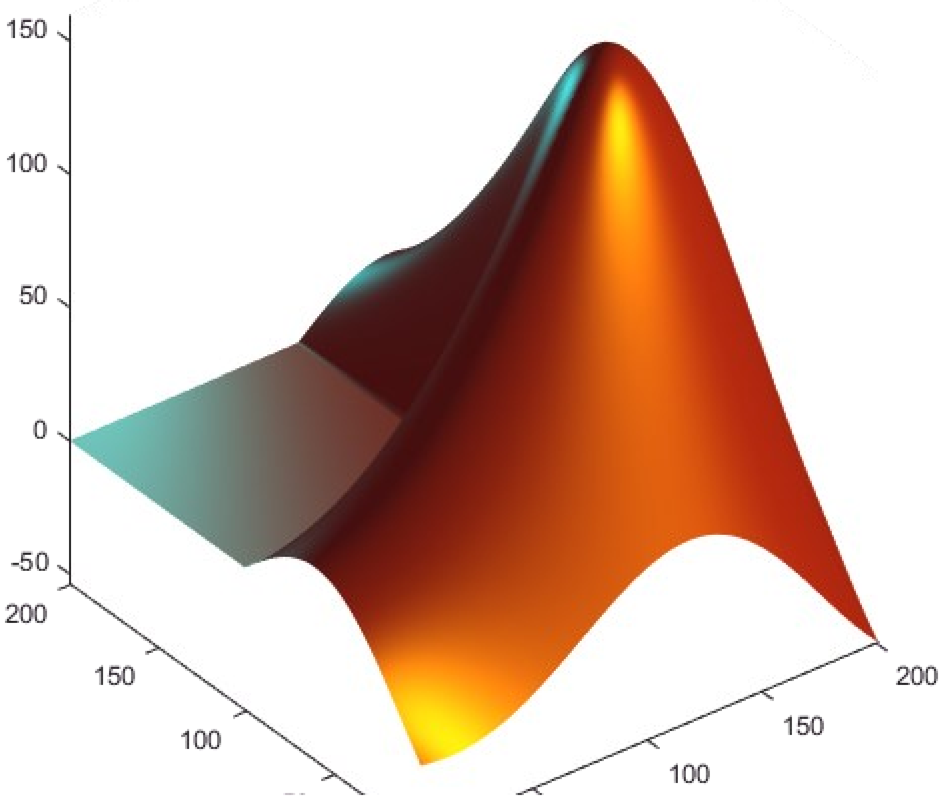
UNIVERSIDADE FEDERAL DE ALAGOAS
CAMPUS SERTÃO
EIXO TECNOLOGIA

UNIVERSIDADE FEDERAL DE ALAGOAS



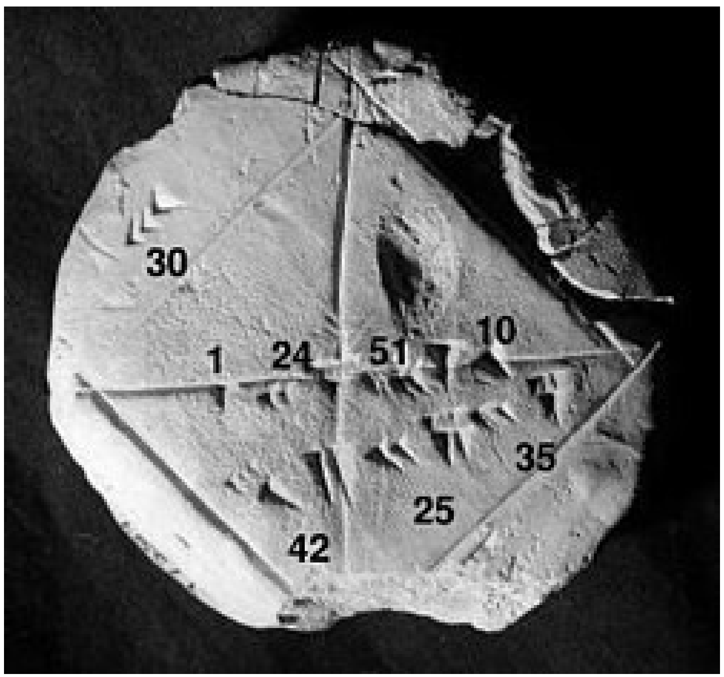
Cálculo Numérico

Prof. Dr. Alverlando Silva Ricardo

Aula 2: PARTE I: Introdução aos Métodos Numéricos

Exemplo: Matemática Babilônica

Tabuleta de argila babilônica com inscrições. A diagonal mostra uma aproximação da raiz quadrada de 2, com seis casas decimais.



O cálculo foi feito na base 60:

$H = 1 + 24/60 + 51/(60^2) + 10/(60^3)$

$H = 1 + 0{,}4 + 0{,}014166 + 0{,}000046$

$H = 1{,}414212$

Exemplo: Matemática Babilônica

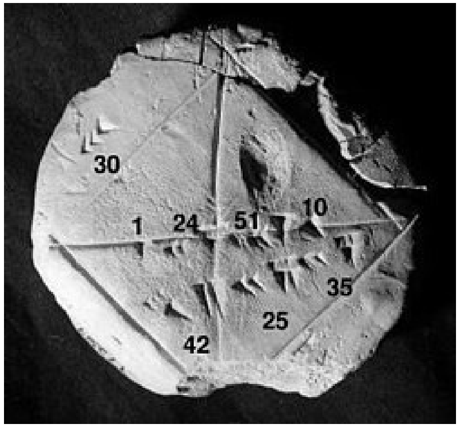
A maioria das tábuas de argila datam de 1800 até 1600 a.C, e cobre tópicos a quais incluem frações, álgebra, equações quadráticas e equações cúbicas além do teorema de Pitágoras.

O impressionante é que a solução desses problemas listados acima foram resolvidas com técnicas aproximadas, e em muitos casos, com precisão impressionante.

Exemplo: Matemática Babilônica

A maioria das tábuas de argila datam de 1800 até 1600 a.C, e cobre tópicos a quais incluem frações, álgebra, equações quadráticas e equações cúbicas além do teorema de Pitágoras.

Se compararmos a solução obtida pela tabuleta com a de um calculadora moderna:



1,414212

1,41421356237373095

Exemplo: Matemática Babilônica

- O Cálculo numérico é muito anterior as máquinas de calcular e aos computadores.

- Diferentes povos da antiguidade desenvolveram e utilizaram algoritmos ou soluções numéricas;

- Por serem soluções de aplicação prática, não atraíram os matemáticos da época, e por muito tempo foram consideradas soluções de qualidade duvidosa.

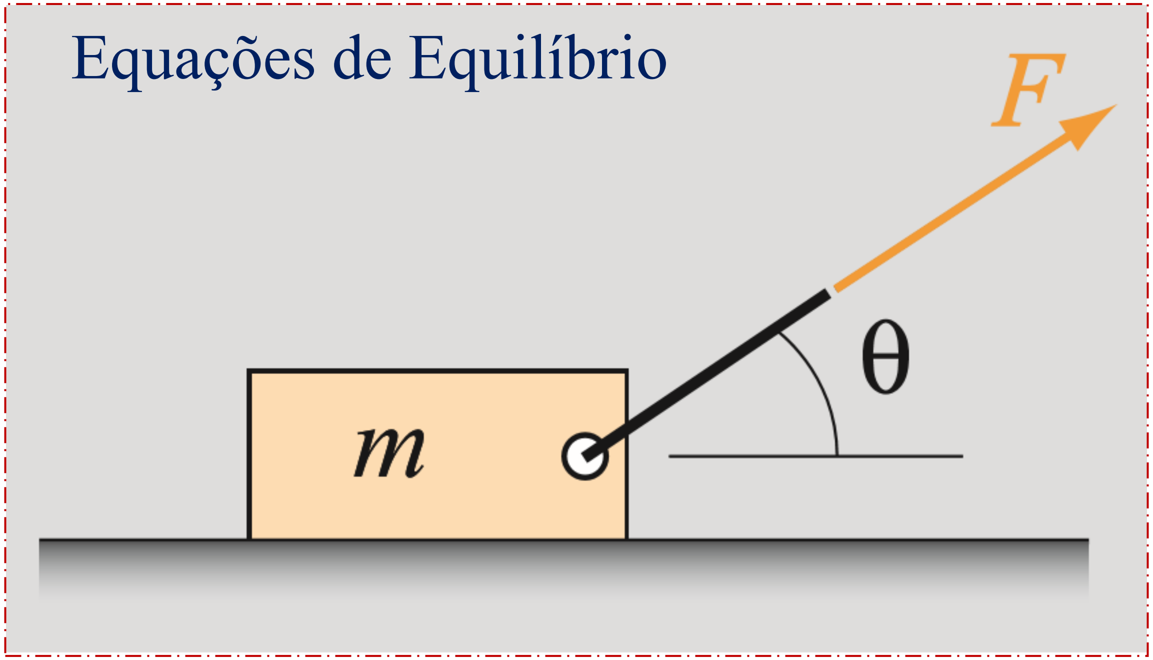
Exemplo: Matemática Babilônica

- A astronomia, física e engenharia foram as principais fontes de inspiradoras para a proposição de problemas que exigiam soluções aproximadas.

- Newton, Lagrange, Euler, Gauss, Jacobi, Fourier e etc construíram e utilizaram vários métodos numéricos.

Introdução

Exemplo: Um bloco sendo puxado por uma força (*F*) aplicada em um ângulo (*θ*). A relação entre a força e o ângulo é dada por:



Movimento de um bloco em uma superfície com atrito.

$$F = \frac{\mu m g}{\cos\theta + \mu \operatorname{sen}\theta}$$

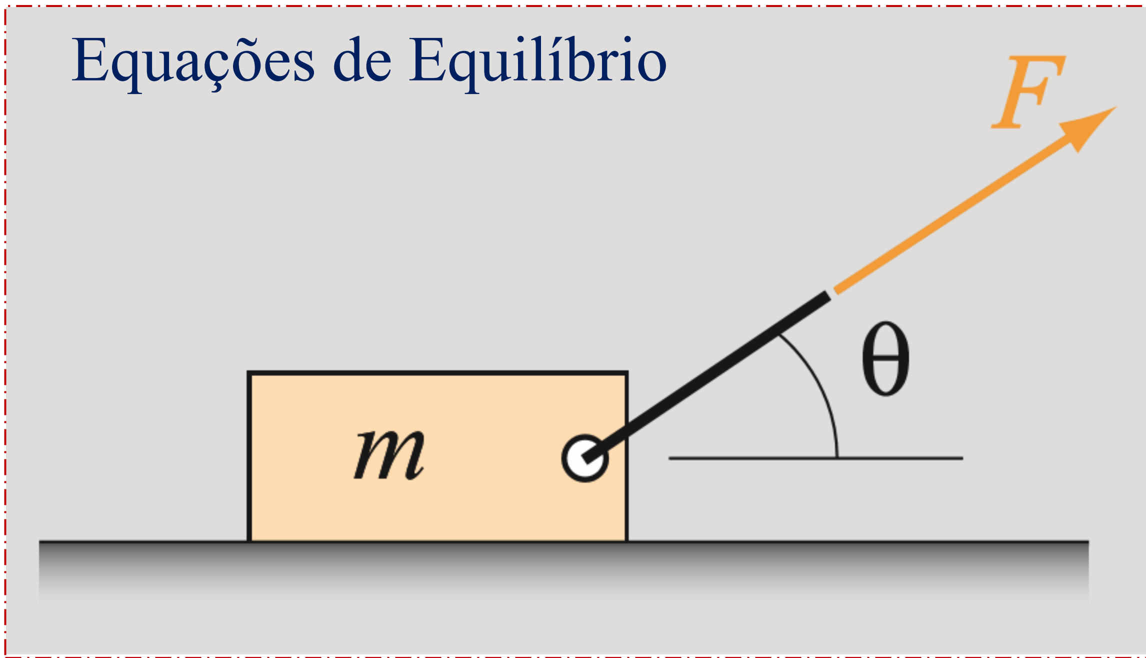
Expressão Analítica

⇩

Solução EXATA!

Introdução

Exemplo: Um bloco sendo puxado por uma força (*F*) aplicada em um ângulo (*θ*). A relação entre a força e o ângulo é dada por:



$$F = \frac{\mu m g}{\cos\theta + \mu \operatorname{sen}\theta}$$

Expressão Analítica

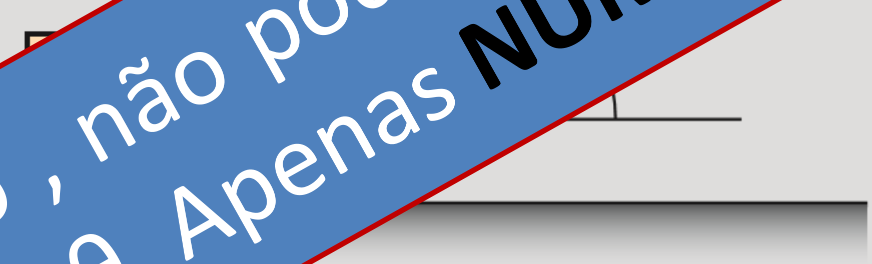
Solução EXATA!

Cada valor de *θ* produz FACILMENTE uma resposta para *F*!!! Porém, o contrário não ocorre!

Introdução

Exemplo: Um bloco sendo puxado por u... aplicada em um ângulo (*θ*). A relaçã... ângulo é dada por:

Equações de Equilíbrio



$$= \frac{\mu m g}{\cos\theta + \mu\,\mathrm{sen}\,\theta}$$

Expressão Analítica

Solução EXATA!

Logo , não pode ser resolvida **analiticamente** para θ. Apenas **NUMERICAMENTE!!!**

Ca... valor de *θ* produz FACILMENTE uma resposta para *F*!!!
Porém, o contrário não ocorre!

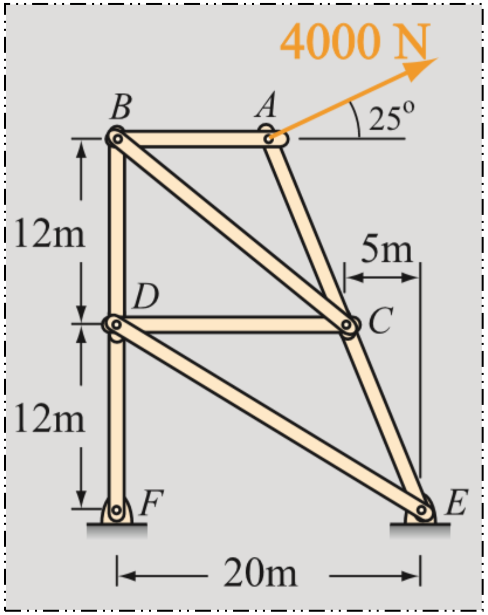
Introdução

Usando **métodos numéricos**, uma **solução aproximada** pode ser determinada com o grau de precisão especificado. Isso significa que, quando a solução numérica para θ é substituída de volta na Equação, o valor de *F* obtido não é exatamente igual ao valor **REAL** de F, embora possa ser bastante próximo.

Introdução

Outros exemplos de aplicação dos métodos numéricos:

- **Sistema com várias Equações:** Pode ser resolvido analiticamente, porém a utilização de uma solução analítica se tornar impraticável!!!



$$0{,}9231F_{AC} = 1690$$

$$-F_{AB} - 0{,}3846F_{AC} = 3625$$

$$F_{AB} - 0{,}7809F_{BC} = 0$$

$$0{,}6247F_{BC} - F_{BD} = 0$$

$$F_{CD} + 0{,}8575F_{DE} = 0$$

$$F_{BD} - 0{,}5145F_{DE} - F_{DF} = 0$$

$$0{,}3846F_{CE} - 0{,}3846F_{AC} - 0{,}7809F_{BC} - F_{CD} = 0$$

$$0{,}9231F_{AC} + 0{,}6247F_{BC} - 0{,}9231F_{CE} = 0$$

Podendo chegar a MILHARES ou MILHÕES de Equações!!!

Introdução

Outros exemplos de aplicação dos métodos numéricos:

- Resolução de EDP não lineares, que só podem se resolvidas analiticamente só em casos particulares.

$$f(x_0, y_0) + \frac{1}{1!}\left[(x - x_0)\left.\frac{\partial f}{\partial x}\right|_{x_0, y_0} + (y - y_0)\left.\frac{\partial f}{\partial y}\right|_{x_0, y_0}\right] +$$

$$\frac{1}{2!}\left[(x - x_0)^2\left.\frac{\partial^2 f}{\partial x^2}\right|_{x_0, y_0} + 2(x - x_0)(y - y_0)\left.\frac{\partial^2 f}{\partial x \partial y}\right|_{x_0, y_0} + (y - y_0)^2\left.\frac{\partial^2 f}{\partial y^2}\right|_{x_0, y_0}\right] +$$

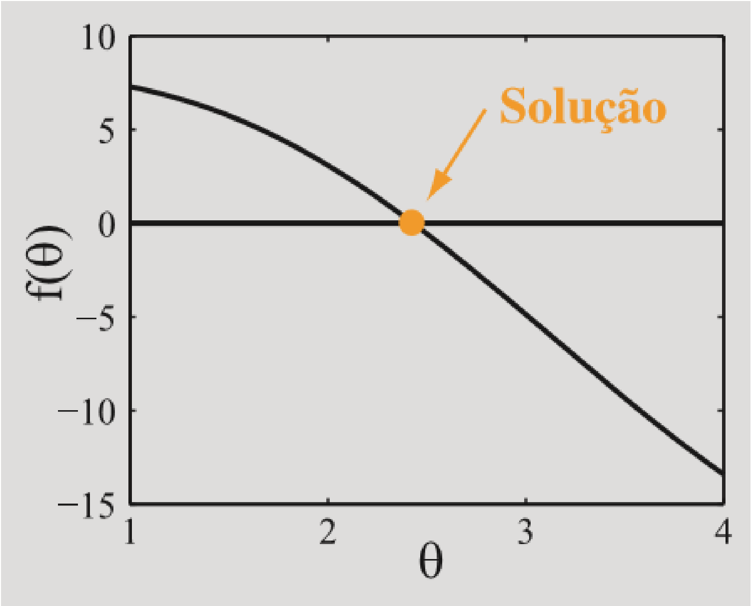
$$+ \ldots + \frac{1}{n!}\left[\sum_{k=0}^{n} \frac{n!}{k!(n-k)!}(x - x_0)^k (y - y_0)^{n-k} \left.\frac{\partial^n f}{\partial x^k \partial y^{n-k}}\right|_{x_0, y_0}\right] = 0$$

Introdução

Outros exemplos de aplicação dos métodos numéricos:

- Métodos para obtenção dos zeros/raízes de funções;

$$f(\theta) = 8 - 4{,}5(\theta - \operatorname{sen}\theta) = 0$$

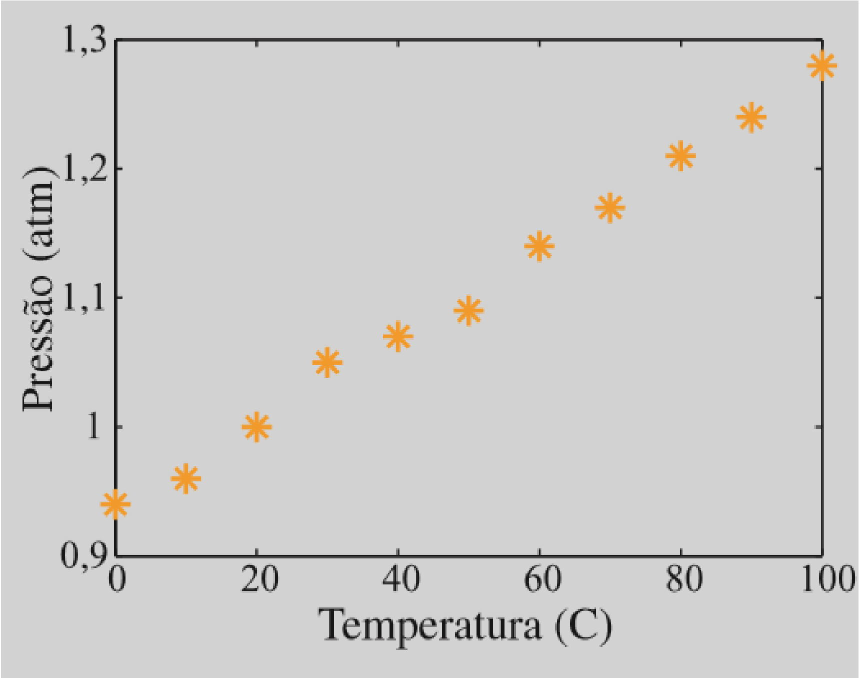


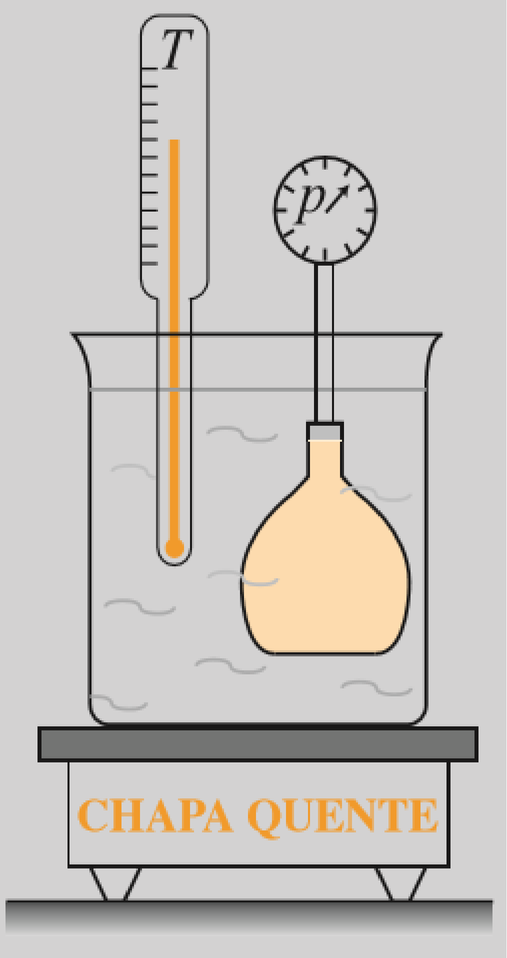
Introdução

Outros exemplos de aplicação dos métodos numéricos:

- interpolação e ajuste de valores tabelados;

T (°C)	0	10	20	30	40	50	60
p (atm.)	0,94	0,96	1,0	1,05	1,07	1,09	1,14
T (°C)	70	80	90	100			
p (atm.)	1,17	1,21	1,24	1,28			



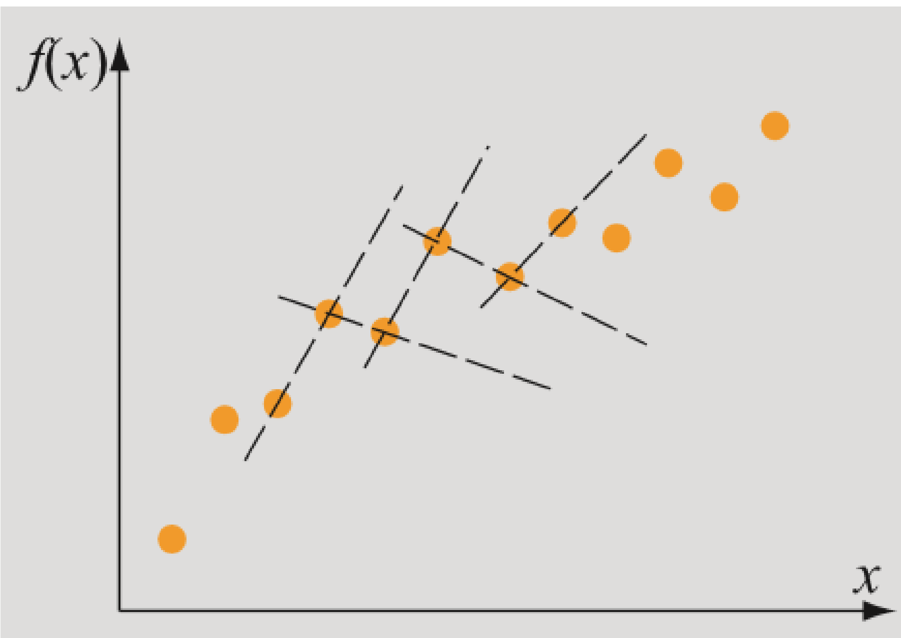


Introdução

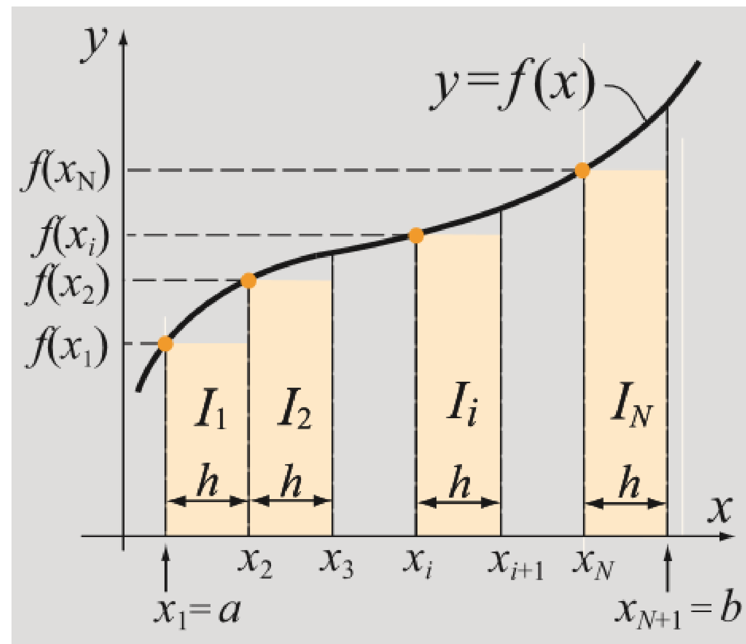
Outros exemplos de aplicação dos métodos numéricos:

- derivação e integração numérica;

Diferenciação Numérica



Integração Numérica



Aplicada:

Funções complexas;
Discretas.

Aplicada:

Funções complexas;
Discretas.

Introdução

Resumidamente:

- O *CN* corresponde a um conjunto de métodos utilizados para se obter a **solução de problemas matemáticos de forma aproximada**. Esses métodos se aplicam principalmente a **problemas** que **não apresentam uma solução exata**, portando precisam ser **resolvidos numericamente**.

- **Métodos numéricos** são técnicas matemáticas usadas na solução de problemas matemáticos que não podem ser resolvidos ou que são difíceis de se resolver analiticamente.

Introdução

- **Técnicas numéricas** para resolver problemas matemáticos foram desenvolvidas centenas e mesmo **milhares de anos atrás**;

- **Antigamente:** os cálculos tinham que ser realizados manualmente.

- **Atualmente:** métodos numéricos são utilizados em rápidos **computadores digitais**. Esses computadores permitem a execução de um grande número de cálculos tediosos e repetitivos em um **curto espaço de tempo**.

Introdução

FLOPS: *floating point operations per second, ou seja, operações de ponto flutuante por segundo.*

20 teraflops – R$ 10 milhões

(adição, subtração, multiplicação e divisão):

processamento de 10 FLOPS

COMPUTADOR DESEMPENHO	
Nome	flops
megaflop	10^{6}
gigaflop	10^{9}
teraflop	10^{12}
petaflop	10^{15}
exaflop	10^{18}
zettaflop	10^{21}
yottaflop	10^{24}

Departamento de Energia dos Estados Unidos (2023)

1,5 Exaflop: *1,5 quintilhão de cálculos matemáticos por segundo.* (US$ 600 milhões)

Introdução

Playstation 5 - 9,2 TFLOPs

Xbox Series X: 12 TFLOPs

Introdução

- Como forma de viabilizar a utilização do métodos, é comum a utilização de algumas **LINGUAGENS COMPUTACIONAIS** para elaboração de algoritmos.

C e ***C++***

- As aplicações do *CN* são **largamente utilizadas** em diversos processos da **engenharia**. Desta forma, torna-se comum a necessidade de **programar-se tais aplicações**.

Cálculo Numérico

O cálculo numérico compreende:

- A análise dos processos que resolvem problemas matemáticos por meio de **operações aritméticas**;
- O desenvolvimento de uma sequência de operações aritméticas que levem às respostas numéricas desejadas (**desenvolvimento de algoritmos**);
- O **uso de computadores** para obtenção das respostas numéricas, o que implica em escrever o método numérico como um **programa de computador**.

MATLAB: MATrix LABoratory

MATLAB

- **MATLAB: cálculos matemáticos**, **modelagem e simulações**, **análise de dados** e **processamento, visualização e traçado de gráficos**, e no desenvolvimento de **algoritmos**.

- Amplamente utilizado em universidades e faculdades: **Exatas, Ciência** e, especialmente, em **Engenharia**.

- **Na indústria**, esse programa é **usado em pesquisa, desenvolvimento e projeto**.

MATLAB

➢ Janelas do MATLAB

Janela	Finalidade
Janela de Comandos (*Command Window*)	Janela principal, permite a entrada de variáveis e roda programas
Janela de Figuras (*Figure Window*)	Contém a saída de comandos gráficos
Janela de Edição (*Editor Window*)	Cria e procura erros em arquivos contendo programas e funções
Janela de Ajuda (*Help Window*)	Fornece a informação de ajuda
Janela de Lançamento (*Lauch Pad Window*)	Permite o acesso a ferramentas, demonstrações e documentação
Janela Histórico de Comandos *(Command History Window*)	Mostra os comandos utilizados na janela de comandos
Janela da Área de Trabalho (*Workspace Window*)	Fornece informações sobre as variáveis utilizadas
Janela de Diretório Corrente (*Current Directory Window*)	Mostra os arquivos no diretório corrente

MATLAB

➢ Operações básicas:

Operação	Símbolo	Exemplo	Operação	Símbolo	Exemplo
Adição	+	5 + 3	Divisão à direita	/	5 / 3
Subtração	–	5 – 3	Divisão à esquerda	\	5 \ 3 = 3 / 5
Multiplicação	*	5 * 3	Exponenciação	^	5 ^ 3 (significa 5^3 = 125)

MATLAB

➢ Funções Matemáticas Elementares:

Comando	Descrição	Exemplo
`sqrt(x)`	Raiz quadrada	>> sqrt(81) ans = 9
`exp(x)`	Exponencial (e^x)	>> exp(5) ans = 148.4132
`abs(x)`	Valor absoluto	>> abs(-24) ans = 24
`log(x)`	Logaritmo natural, na base *e* (ln)	>> log(1000) ans = 6.9078
`log10(x)`	Logaritmo na base 10	>> log10(1000) ans = 3.0000
`sin(x)` `sind(x)`	Seno de um ângulo *x* (em radianos) Seno de um ângulo *x* (em graus)	>> sin(pi/6) ans = 0.5000 >> sind(30) ans = 0.5000
As demais funções trigonométricas são escritas da mesma maneira. As funções trigonométricas inversas são escritas simplesmente acrescentando "a" na frente dos comandos, como, por exemplo, `asin(x)`.		

➢ Criando um vetor

```
nome_da_variável=[número número... número]
```

```
nome_da_variável = linspace(xi,xf,n)
```

```
nome_da_variável= m:q:n
```

```
>> ano = [1984  1986  1988  1990  1992  1994  1996]
ano =
     1984     1986     1988     1990     1992     1994     1996
```

Vetor linha formado com a digitação dos elementos.

```
>> pnt = [2;  4;  5]
pnt =
    2
    4
    5
```

Vetor coluna formado com a digitação dos elementos.

```
>> x = [1:2:13]

x =
    1    3    5    7    9   11   13
```

Vetor linha com espaçamento constante.

```
>> va = linspace(0,8,6)
va =
       0    1.6000    3.2000    4.8000    6.4000    8.0000
```

Vetor linha com 6 elementos, primeiro elemento 0 e último elemento 8.

➢ Criando uma Matriz

```
nome_da_variável=[1ª linha de elementos;2ª linha
de elementos;...;última linha de elementos]
```

```
>> a = [5  35  43;  4  76  81;  21  32  40]
a =
     5    35    43
     4    76    81
    21    32    40
>> cd = 6; e = 3; h = 4;
>> Mat = [e, cd*h, cos(pi/3); h^2, sqrt(h*h/cd), 14]
Mat =
    3.0000   24.0000    0.5000
   16.0000    1.6330   14.0000
```

Um ponto-e-vírgula é digitado entre cada linha.

Definem-se as variáveis.

Elementos são fornecidos como expressões matemáticas.

➢ Funções usadas para manuseio de Vetores e Matrizes

Comando	Descrição	Exemplo
`length(A)`	Retorna o número de elementos no vetor A	>> A = [5 9 2 4]; >> length(A) ans = 4
`size(A)`	Retorna um vetor linha `[m,n]`, onde `m` e `n` são o tamanho `m` × `n` do arranjo `A` (`m` é o número de linhas, `n` é o número de colunas)	>> A = [6 1 4 0 12; 5 19 6 8 2] A = 6 1 4 0 12 5 19 6 8 2 >> size(A) ans = 2 5
`zeros(m,n)`	Cria uma matriz com *m* linhas e *n* colunas na qual todos os elementos são o número 0	>> zr = zeros(3,4) zr = 0 0 0 0 0 0 0 0 0 0 0 0
`ones(m,n)`	Cria uma matriz com *m* linhas e *n* colunas na qual todos os elementos são o número 1	>> ne = ones(4,3) ne = 1 1 1 1 1 1 1 1 1 1 1 1

➢ Operações com Vetores e Matrizes

```
>> VA = [8  5  4];  VB = [10  2  7];
>> VC = VA + VB
VC =
   18    7   11
>> A = [5  -3  8;  9  2  10],  B = [10  7  4; -11  15  1]
A =
    5   -3    8
    9    2   10
B =
   10    7    4
  -11   15    1
>> C = A + B

C =
   15    4   12
   -2   17   11
>> C - 8
ans =
    7   -4    4
  -10    9    3
```

Define dois vetores VA e VB.

Define um vetor VC que é igual a VA+VB.

Define duas matrizes A e B.

Define uma matriz C que é igual a A+B.

Subtrai 8 da matriz C.

8 é subtraído de cada elemento de C.

➢ Operações com Vetores e Matrizes

```
>> A = [2 -1; 8 3; 6 7], B = [4 9 1 -3; -5 2 4 6]
A =
    2   -1
    8    3
    6    7
B =
    4    9    1   -3
   -5    2    4    6
>> C = A*B
C =
   13   16   -2  -12
   17   78   20   -6
  -11   68   34   24
```

Define duas matrizes A e B.

Multiplica A*B.

C é uma matriz (3 × 4).

Símbolo	Descrição	Símbolo	Descrição
.*	Multiplicação	./	Divisão à direita
.^	Exponenciação	.\	Divisão à esquerda

➢ Funções dedicadas ao manuseio de Vetores e Matrizes

Comando	Descrição	Exemplo
`mean(A)`	Se `A` é um vetor, a função retorna o valor médio de seus elementos	>> A = [5 9 2 4]; >> mean(A) ans = 5
`sum(A)`	Se `A` é um vetor, a função retorna a soma dos elementos do vetor	>> A = [5 9 2 4]; >> sum(A) ans = 20
`sort(A)`	Se `A` é um vetor, a função arranja os elementos do vetor em ordem ascendente	>> A = [5 9 2 4]; >> sort(A) ans = 2 4 5 9
`det(A)`	A função retorna o determinante da matriz quadrada `A`	>> A = [2 4; 3 5]; >> det(A) ans = -2

➢ ARQUIVOS DE FUNÇÃO


```
function[argumentos_de_saída]= nome_da_função(argumentos_de_entrada)
```

A palavra function deve ser a primeira palavra, e deve ser digitada em letras minúsculas

Lista de argumentos de saída, digitada entre colchetes

Nome da função

Lista de argumentos de entrada digitados entre parênteses e separados por vírgulas

ARQUIVOS DE FUNÇÃO

```
function[mpag,tpag]=delmiro(montante,taxa,anos)
% calcula o pagamento mensal e total de uma hipoteca
% Argumentos de entrada:
% montante: montante total do empréstimo em R$
% taxa: taxa percentual de juros anual
% anos: número de anos Argumentos de entrada:
% mpag: parcela mensal
% tpag:  pagamento total

format bank

taxam=taxa*0.01/12;
a=1+taxam;
b=((anos*12)-1)/taxam;
mpag=montante*a^(anos*12)/(a*b);
tpag=mpag*anos*12;
end
```

PROGRAMAÇÃO NO MATLAB

PROGRAMAÇÃO NO MATLAB

- Um programa de computador é formado por uma seqüência de comandos (**ALGORITMO: Plano de programação**).

- Normalmente, mudança no fluxo de um programa requer algum tipo de processo de tomada de decisões.

- O computador deve decidir entre executar o próximo comando ou salta-lo;

- Isso é feito usando **OPERADORES LÓGICOS E RELACIONAIS**.

PROGRAMAÇÃO NO MATLAB

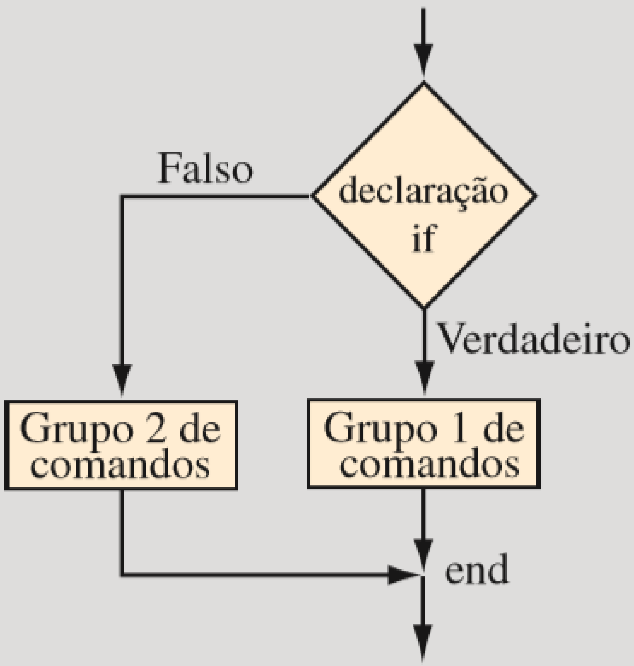
➢ Um operador relacional compara dois números determinando se uma declaração de comparação é verdadeira ou falsa (por exemplo, 5 < 8);

Operador relacional	Descrição	Operador relacional	Descrição
<	Menor que	>=	Menor ou igual a
>	Maior que	= =	Igual a
<=	Menor ou igual a	~=	Diferente de

<table>
<tr><th>Operador lógico</th><th>Nome</th><th>Descrição</th></tr>
<tr><td>&
Exemplo: A&B</td><td>AND</td><td>Atua em dois operandos (A e B). Se ambos forem verdadeiros, o resultado será verdadeiro (1); senão, o resultado será falso (0).</td></tr>
<tr><td>|
Exemplo: A|B</td><td>OR</td><td>Atua em dois operandos (A e B). Se qualquer um deles ou ambos forem verdadeiros, o resultado será verdadeiro (1); senão (ambos forem falsos), o resultado será falso (0).</td></tr>
</table>

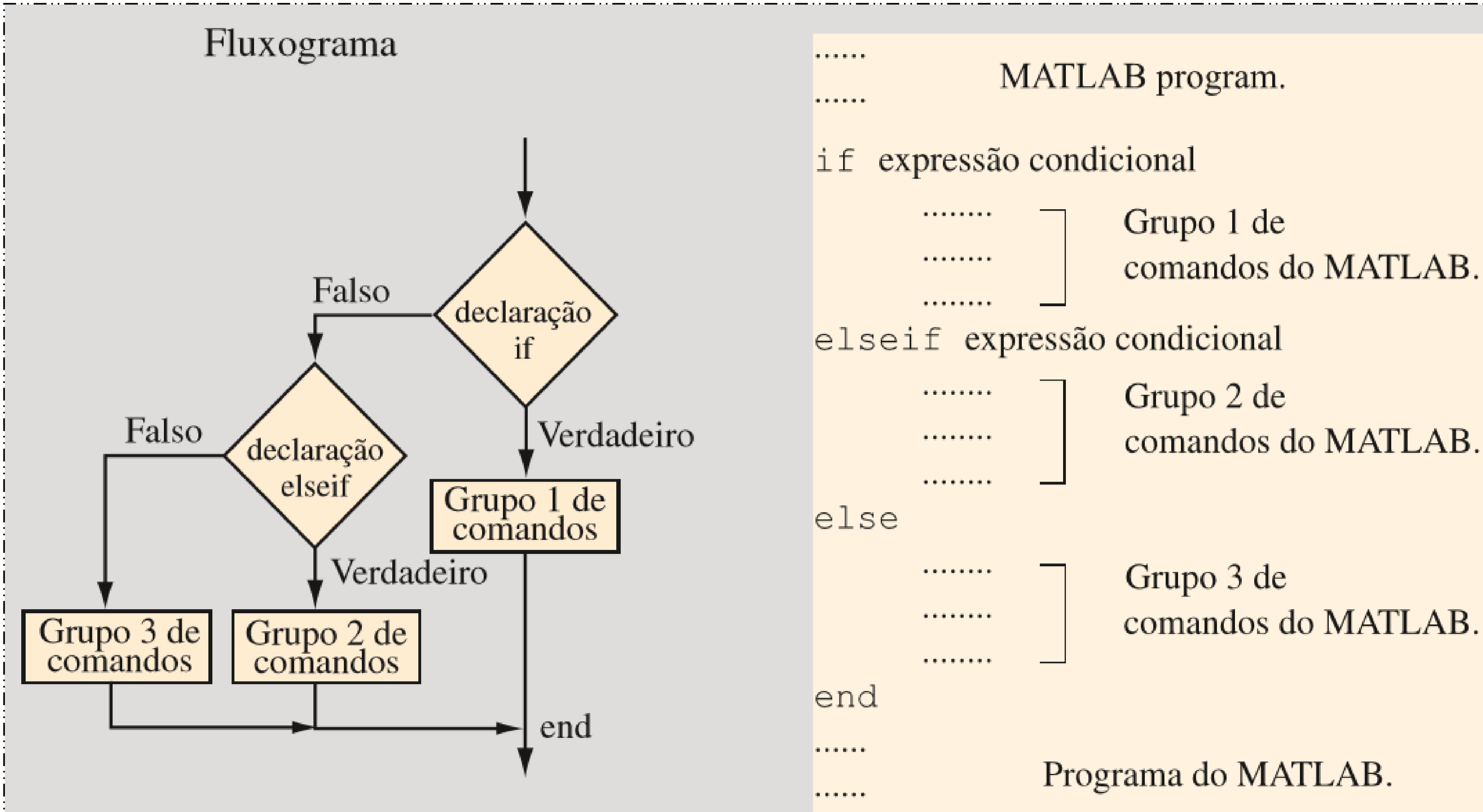
➢ Declarações condicionais, estruturas if-else-end

Fluxograma




```
......          Programa do MATLAB
......
if  expressão condicional
        ........         Grupo 1 de
        ........         comandos do MATLAB
        ........
else
        ........         Grupo 2 de
        ........         comandos do MATLAB
        ........
end
......          Programa do MATLAB
......
```

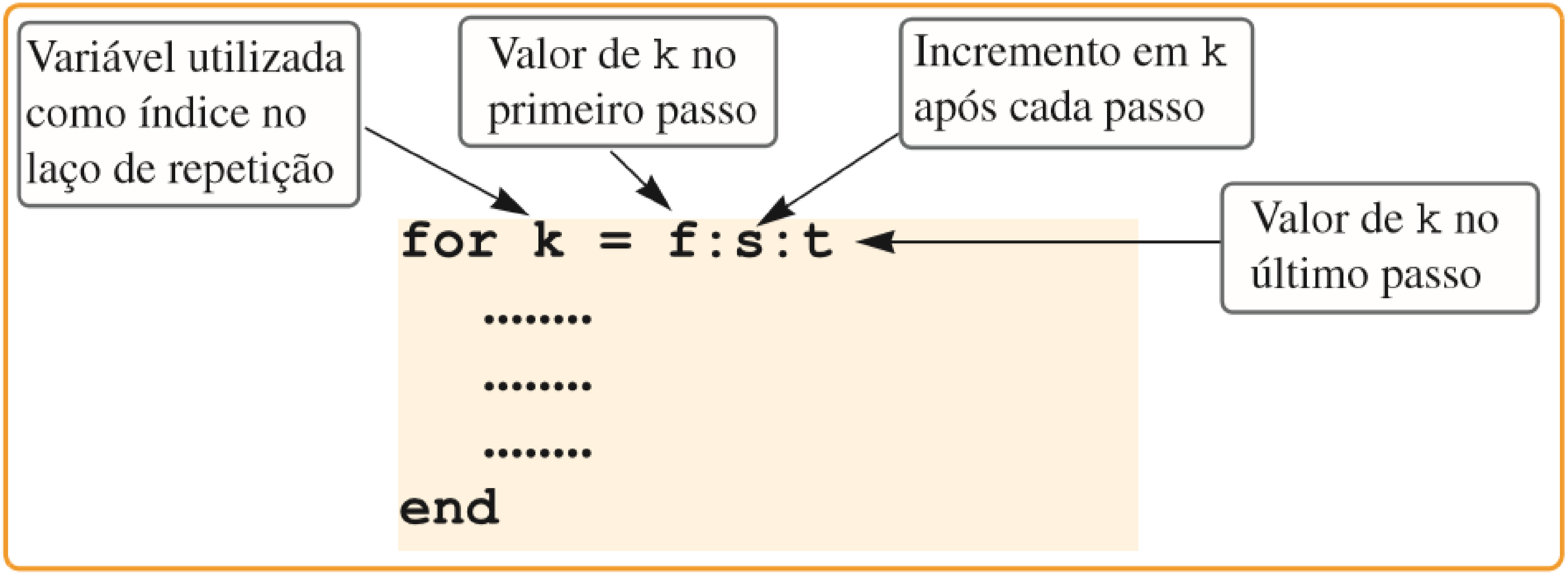
➢ Declarações condicionais, estruturas if-elseif-else-end



➢ Laços de repetição (loops)

➢ Em um laço de repetição, a execução de um comando é repetida várias vezes de forma consecutiva;

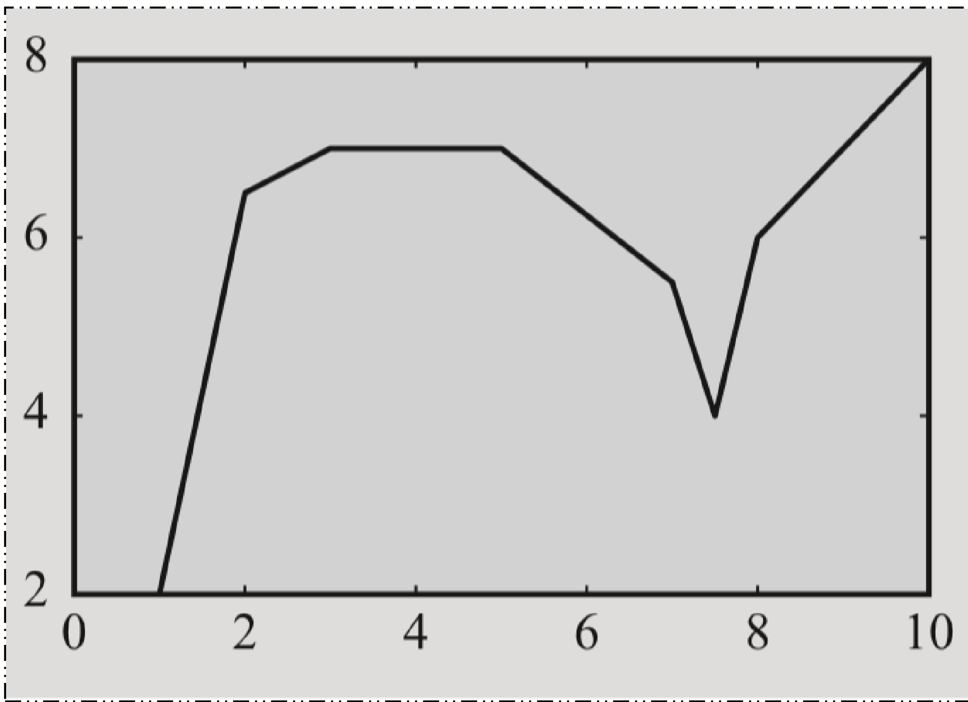
➢ Cada rodada de execução é chamada de passo.



➢ TRAÇADO DE GRÁFICOS

`plot(x,y)`

```
>> x = [1  2  3  5  7  7.5  8  10];
>> y = [2  6.5  7  7  5.5  4  6  8];
>> plot(x,y)
```



➢ TRAÇADO DE GRÁFICOS

```
plot(x,y,'especificadores de linha',)
```

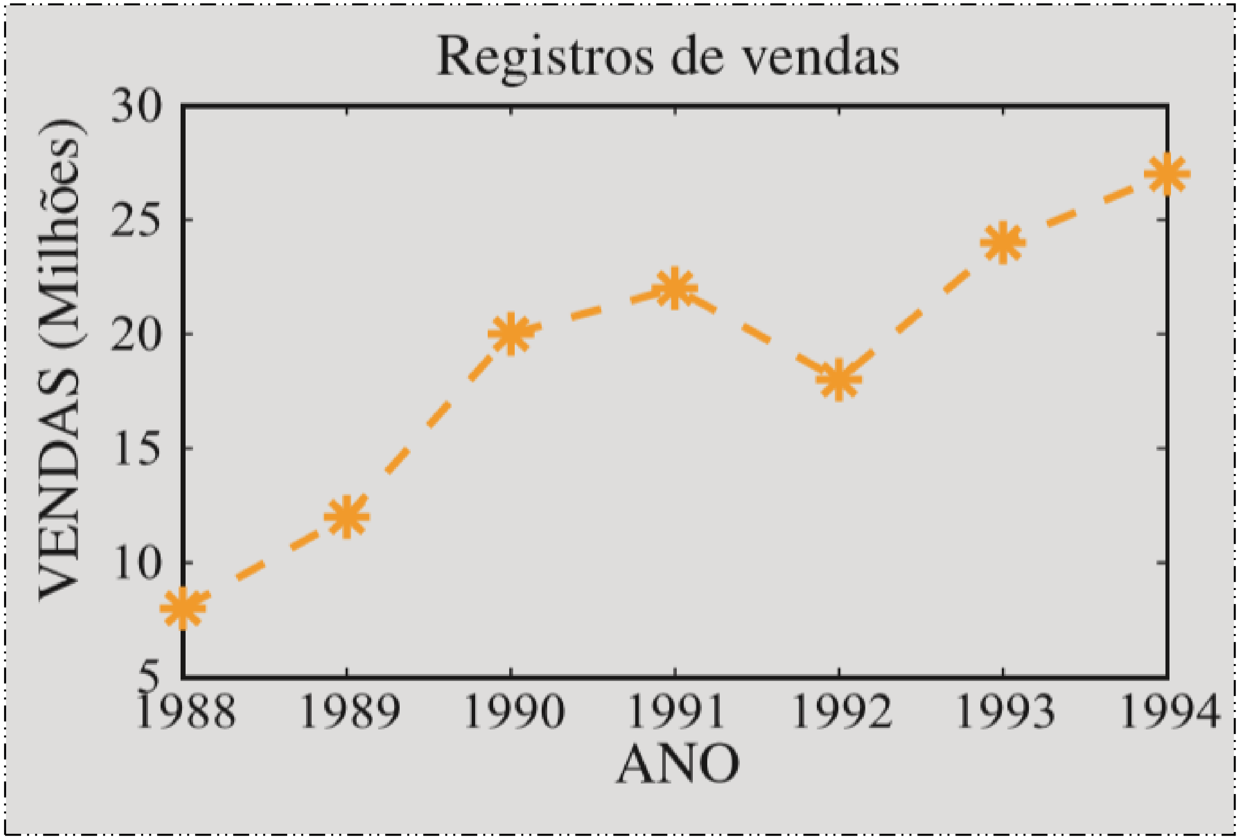
Estilo de linha	Especificador	Estilo de linha	Especificador
sólido (padrão)	-	pontuado	:
traço	--	traço-ponto	-.

Cor de linha	Especificador	Cor de linha	Especificador	Cor de linha	Especificador	Cor de linha	Especificador
vermelha	r	azul	b	magenta	m	preto	k
verde	g	ciano	c	amarelo	y	branco	w

Marcador	Especificador	Marcador	Especificador	Marcador	Especificador
sinal de mais	+	asterisco	*	quadrado	s
círculo	o	ponto	.	diamante	d

➢ TRAÇADO DE GRÁFICOS

```
xlabel('texto no formato string')
ylabel('texto no formato string')
title('texto no formato string')
```



➢ TRAÇADO DE GRÁFICOS

```
ano=[1988:1:1994];
vendas=[8 12 20 22 18 24 27];
plot(ano,vendas,'--r*','linewidth',2,'markersize',16)
xlabel('ANO')
ylabel('VENDAS(Milhões)')
title('Registros de Vendas')
```

➢ Exemplo:

Criar um algoritmo para a solução da equação quadrática:

$$ax^2 + bx + c = 0$$

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \qquad x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

Algoritmo para se obter as raízes reais de uma equação quadrática

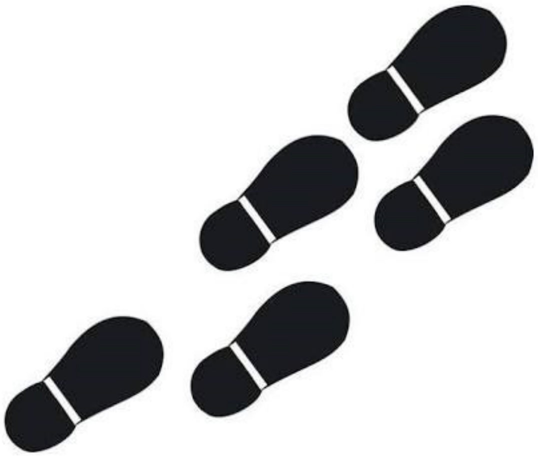
São dadas as três constantes da equação quadrática, *a*, *b* e *c*.

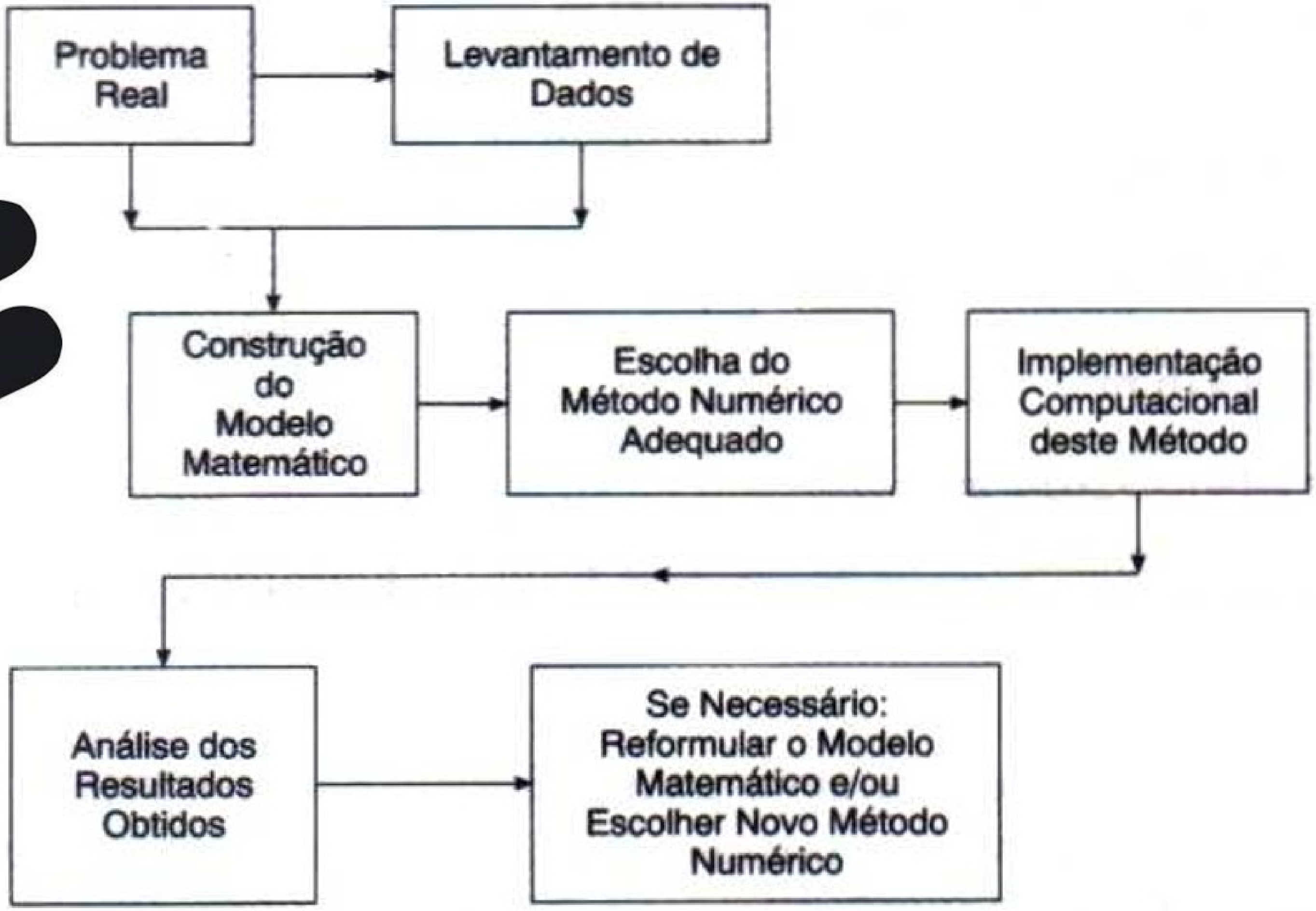
1. Calcule o valor do discriminante $D = b^2 - 4ac$.
2. Se $D \geq 0$, calcule as duas raízes usando a Eq. (1.19).
3. Se $D = 0$, calcule a raiz: $x = \frac{-b}{2a}$ e mostre a mensagem: "A equação possui raiz única".
4. Se $D < 0$, mostre a mensagem: "A equação não possui raízes reais".

ERROS EM SOLUÇÕES NUMÉRICAS

Modelagem e Resolução

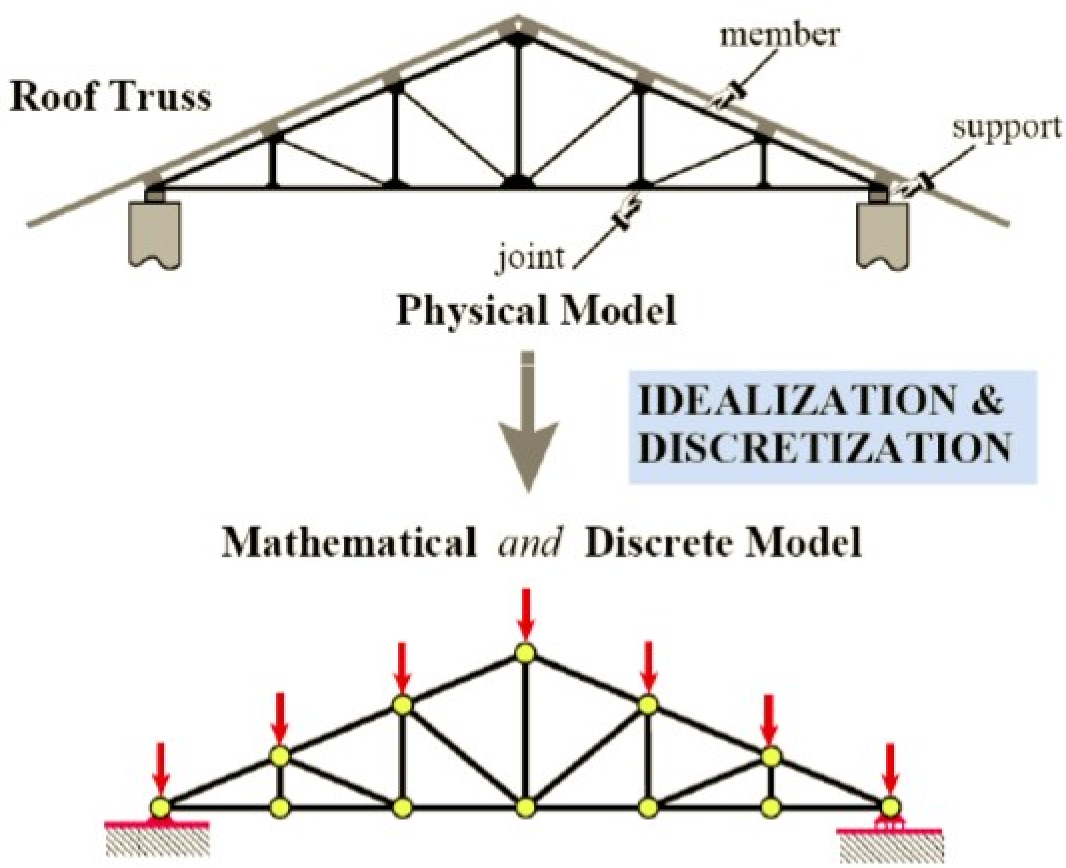
➢ A utilização de **simuladores numéricos** para determinação da solução de um problema requer a execução da seguinte sequência de etapas:



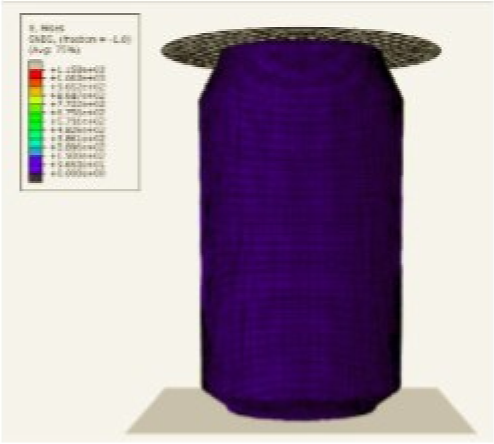


Modelagem e Resolução

- Modelo estrutural x problema real:

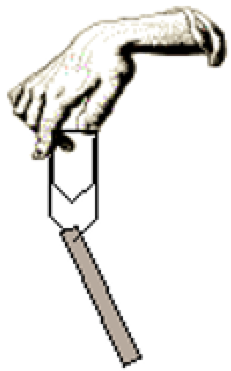


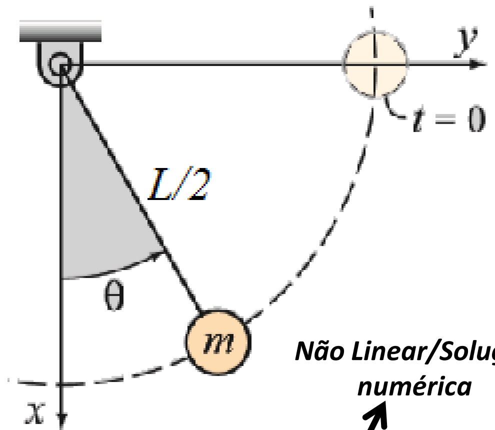
ABAQUS

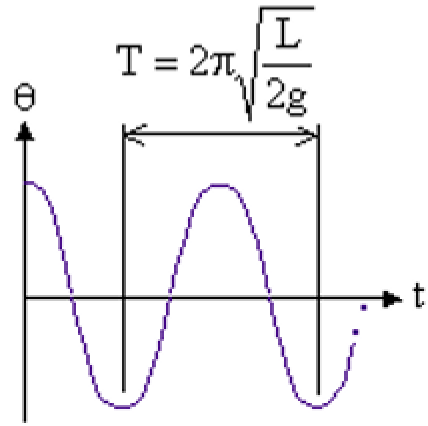


Modelagem e Resolução

ENTENDIMENTO DE UM PROBLEMA







PROBLEMA REAL

Modelagem:

•Observação do fenômeno.
•Levantamento dos efeitos dominantes (idealizações).
•Referência aos conhecimentos prévios físicos e matemáticos.

Não Linear/Solução numérica

$$\frac{d^2\theta}{dt^2} + \frac{2g}{L}\,\mathrm{sen}\,\theta = 0$$

Linear/Solução analítica

$$\theta \ll 1 \Rightarrow \frac{d^2\theta}{dt^2} + \frac{2g}{L}\theta = 0$$

MODELO MATEMÁTICO

•Permite analisar situações reais ou fazer previsões de comportamentos.

SOLUÇÃO

Resolução:

•Procedimentos analíticos ou numéricos.
•Exige uma confrontação com situações experimentais para validação do modelo matemático.

Modelagem e Resolução

Tensão de engenharia:

$$\sigma = \frac{P}{A_0}$$

✖

Tensão verdadeira:

$$\sigma_v = \frac{P}{A}$$

Deformação específica de engenharia:

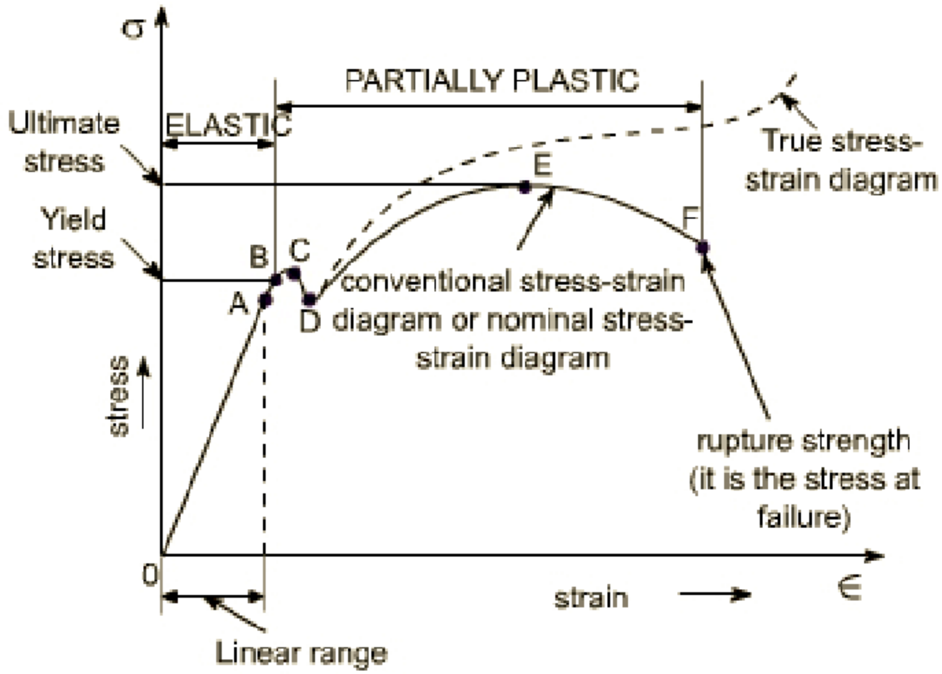
$$\varepsilon = \frac{\delta}{L_0}$$

✖

Deformação específica verdadeira:

$$\varepsilon_v = \Sigma\Delta\varepsilon = \Sigma(\Delta L / L)$$

$$\varepsilon_v = \int_{L_0}^{L} \frac{dL}{L} = \ln\frac{L}{L_0}$$

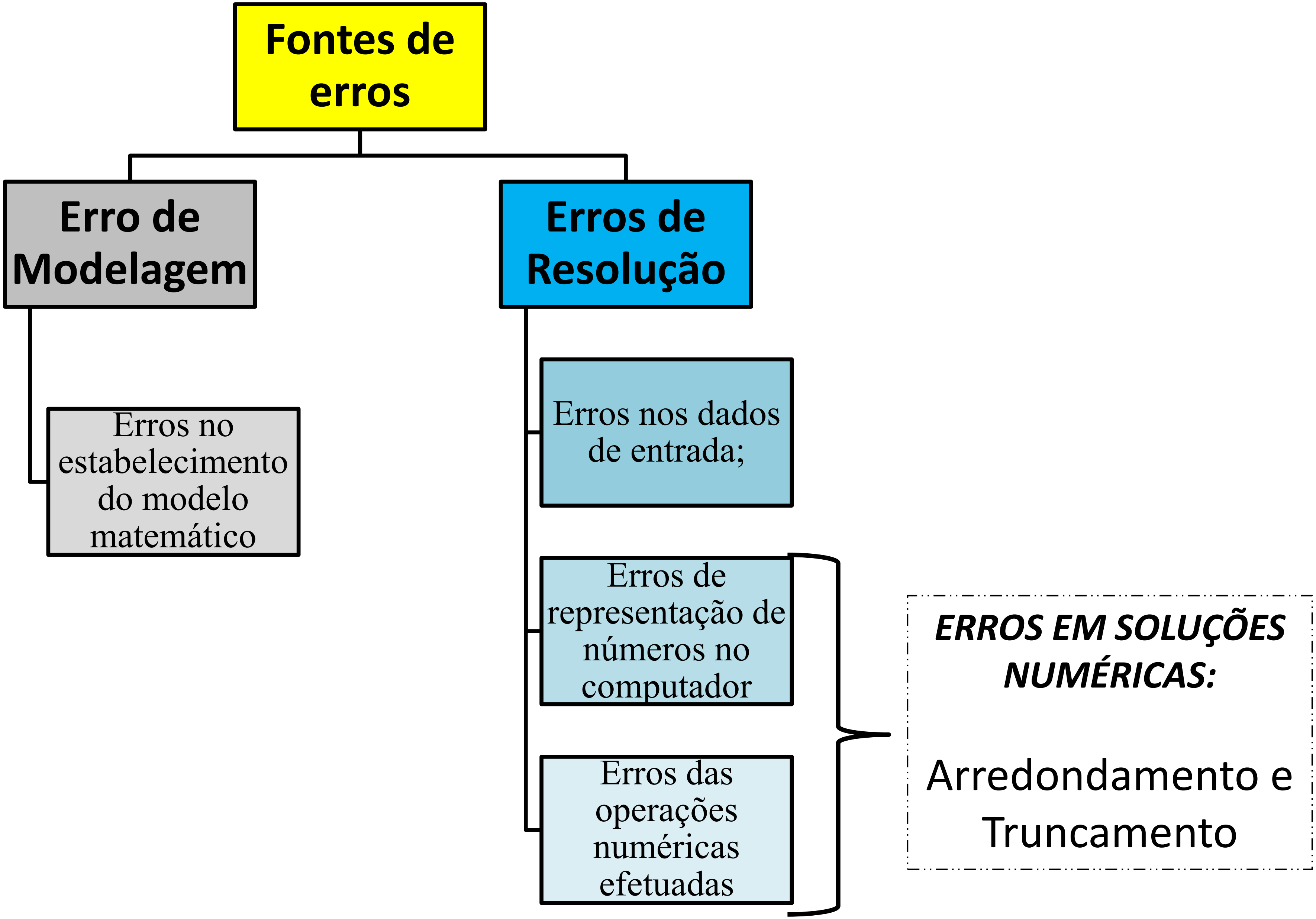


Cálculo Numérico

- Espera-se, com isso, obter **respostas confiáveis** para problemas matemáticos.
- No entanto, não é raro acontecer que os **resultados** obtidos estejam **distantes** do que se **esperava** obter.

Por quê?

Fontes de erros



Fontes de erros

- ***Suponha o seguinte problema:*** você está em cima de um edifício que não sabe a altura, mas precisa determiná-la. Tudo que tem em mãos é uma bola de metal e um cronômetro. ***O que fazer***?

Fontes de erros

- ***Suponha o seguinte problema:*** você está em cima de um edifício que não sabe a altura, mas precisa determiná-la. Tudo que tem em mãos é uma bola de metal e um cronômetro. ***O que fazer***?

Equação que governa o problema:

$$S = S_0 + v_0 \cdot t + \frac{1}{2} g \cdot t^2$$

Onde:

- s é a posição final;
- s_0 é a posição inicial;
- v_0 é a velocidade inicial;
- t é o tempo percorrido;
- g é a aceleração gravitacional.

Fontes de erros

- ***Suponha o seguinte problema:*** você está em cima de um edifício que não sabe a altura, mas precisa determiná-la. Tudo que tem em mãos é uma bola de metal e um cronômetro. ***O que fazer***?

A bola é solta do topo do edifício e, com o cronômetro, marcou-se que ela levou 3 segundos para atingir o solo. Com isso podemos concluir, a partir da equação acima, que a altura do edifício é de 44,1 metros.

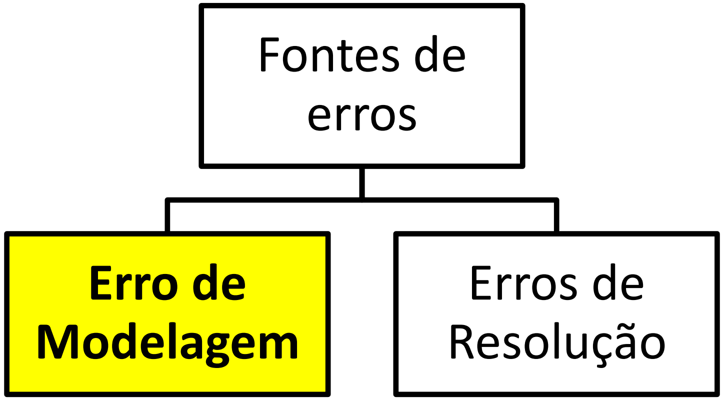
$$S = S_0 + v_0 \cdot t + \frac{1}{2} g \cdot t^2 = 0 + 0 \cdot 3 + \frac{1}{2} \cdot 9{,}8 \cdot 3^2 = 44{,}1\text{m}$$

Essa resposta é confiável?
Onde estão os erros?

Fontes de erros

➢ ***Erros de modelagem do problema:***

- Resistência do ar;
- Velocidade do vento;
- Forma do objeto, etc.

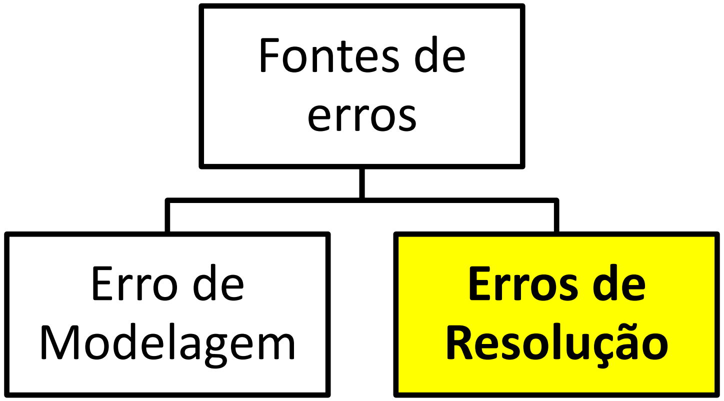


Estes erros estão associados, em geral, à simplificação do modelo matemático adotado.

Fontes de erros

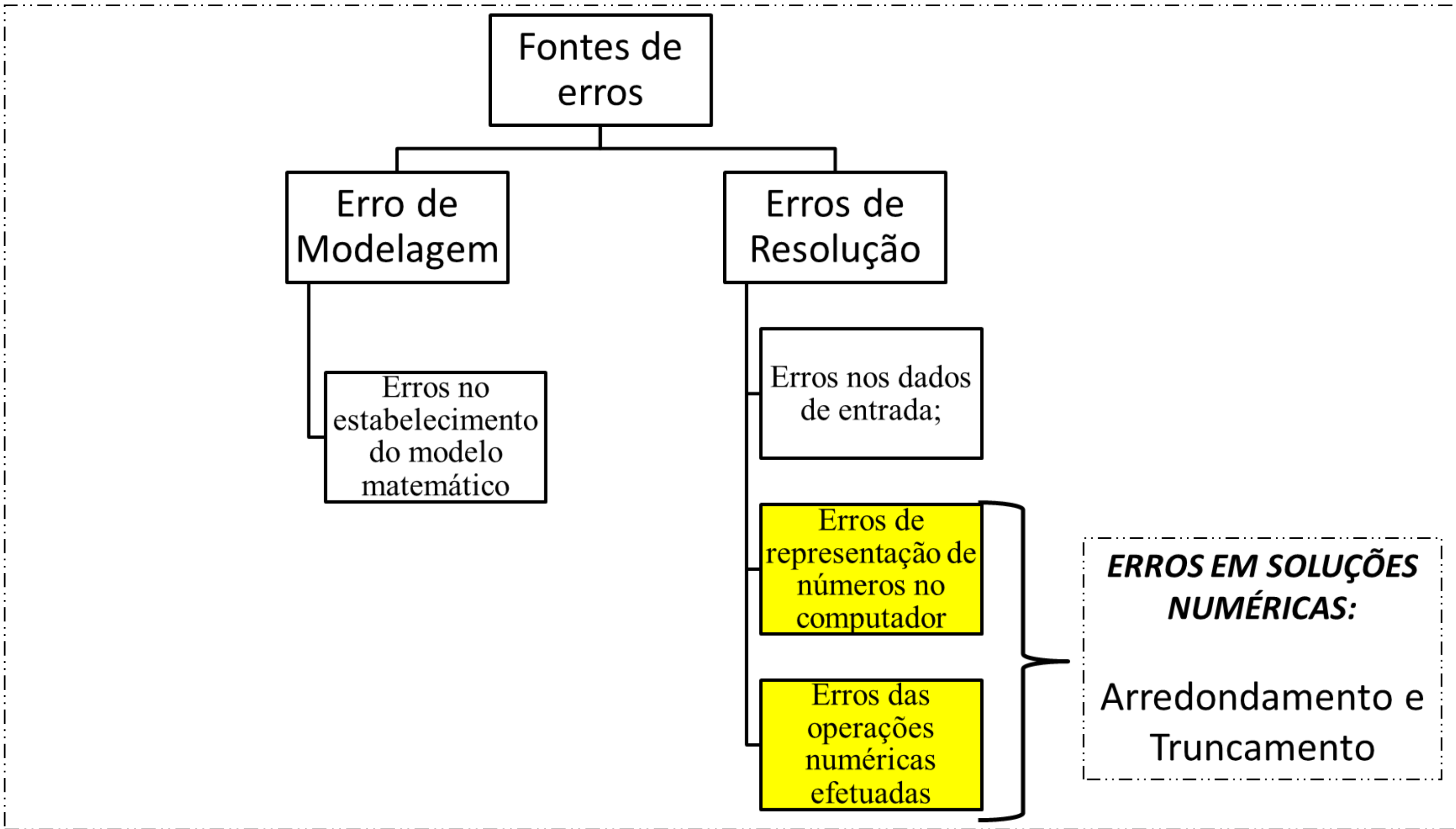
➢ ***Erros de resolução do problema:***

- Precisão dos dados de entrada (leitura do cronômetro $t = 2{,}8\ seg$, $h = 38{,}42\ m$, gravidade);
- Operações numéricas efetuadas;
- Erro de truncamento (troca de uma série infinita por uma série finita).



Fontes de erros

Os dados de entrada contêm uma imprecisão inerente. Ou seja, não há como evitar que ocorram. Por isso, estudaremos os ERROS EM SOLUÇÕES NUMÉRICAS.



REPRESENTAÇÃO NUMÉRICA

REPRESENTAÇÃO NUMÉRICA

- **Definição:** conjunto de regras que definem uma maneira de representar quantidades matemáticas.

- **Motivação 1:** Calcular a área de uma circunferência de raio 100 metros.

- **Resultados:**

- 31400 m^2
- 31416 m^2
- 31415,9265 m^2

Por que os resultados diferem?

É possível obter o valor exato da área?

REPRESENTAÇÃO NUMÉRICA

- **Definição:** conjunto de regras que definem uma maneira de representar quantidades matemáticas.

- **Motivação 1:** Calcular a área de uma circunferência de raio 100 metros.

- **Resultados:**
  - 31400 m^2
  - 31416 m^2
  - 31415,9265 m^2

- **π :**
  - $\pi = 3,14$
  - $\pi = 3,1416$
  - $\pi = 3,141592654$

REPRESENTAÇÃO NUMÉRICA

- **Motivação 2:** Deseja-se calcular a somatória $s = \sum_{i=1}^{30000}(x_i)$, para $x_i = 0.5$ e $x_i = 0.11$.

- **Resultados:**

	x_i =0.5	x_i= 0.11
Calculadora	15000	3300
Computador	15000	3299,99691

Por que os resultados diferem?

REPRESENTAÇÃO NUMÉRICA

- Os erros ocorridos nos dois problemas dependem da representação dos números na maquina utilizada – ***base de máquina.***
  - Cada computador possui uma precisão numérica diferente (***número máximo de dígitos***).
  - Essa precisão depende do hardware, do sistema operacional, etc...
  - Um computador normalmente opera no sistema ou base binária (***zeros e uns***);
  - **Qual a base utilizada no nosso dia-a-dia?**

REPRESENTAÇÃO NUMÉRICA

➢ **Qual a base utilizada no nosso dia-a-dia?**

- A base decimal, onde se utiliza os algarismos: 0, 1, 2, 3, 4, 5, 6, 7, 8 e 9.
- Um computador normalmente opera no sistema ou base binária (***zeros e uns***);
- Interação entre usuário e computador:

REPRESENTAÇÃO NUMÉRICA

- **Exemplos:**

- $(100110)_2 = (38)_{10}$
- $(11001)_2 = (25)_{10}$

Como realizar essa conversão?

...CONTINUA