



Universidade Federal do Rio Grande do Norte
Centro de Tecnologia
Dept°. de Engenharia de Computação e Automação



COMPUTAÇÃO NUMÉRICA

Notas de Aula

Índice

COMPUTAÇÃO NUMÉRICA.....	1
NOTAS DE AULAÍNDICE	1
ÍNDICE.....	2
1 INTRODUÇÃO AO SCILAB.....	4
1.1 GERAÇÃO DE MATRIZES	4
1.2 VARIÁVEIS.....	5
1.3 COMANDOS ELEMENTARES	5
1.4 NÚMEROS E EXPRESSÕES ARITMÉTICAS	5
1.5 FORMATO DE SAÍDA	6
1.6 OUTROS COMANDOS	6
1.7 OPERAÇÕES COM MATRIZES.....	6
1.8 OPERAÇÕES ELEMENTO A ELEMENTO	7
1.9 OPERADORES RELACIONAIS	8
1.10 OPERADORES LÓGICOS.....	8
1.11 FUNÇÕES ELEMENTARES	8
1.12 VETORES E SUBSCRITOS	8
1.13 GERAÇÃO DE TABELAS DE VALORES.....	8
1.14 MANIPULAÇÃO DE LINHAS E COLUNAS DE MATRIZES.....	9
1.15 POLINÔMIOS	9
1.16 RECURSOS GRÁFICOS	10
1.17 COMANDOS PARA CONTROLE DE FLUXO	11
1.18 PROGRAMANDO NO SCILAB	11
1.19 FUNÇÕES NO SCILAB	12
2 REPRESENTAÇÃO NUMÉRICA E ERROS	14
2.1 INTRODUÇÃO E DEFINIÇÕES	14
2.2 MUDANÇA DE BASE.....	15
2.3 REPRESENTAÇÃO NUMÉRICA.....	15
2.4 ARREDONDAMENTO	16
2.5 PRECISÃO	16
3 SISTEMA DE EQUAÇÕES ALGÉBRICAS LINEARES	18
3.1 INTRODUÇÃO	18
3.2 FORMULAÇÃO.....	18
3.3 CLASSIFICAÇÃO DOS SISTEMAS	18
3.4 MÉTODOS DE SOLUÇÃO	18
3.5 MÉTODO DE GAUSS	19
3.5.1 <i>Princípio do método:</i>	19
3.5.2 <i>Transformações Elementares</i>	19
3.5.3 <i>Substituição Reversa</i>	19
3.5.4 <i>Triangulação</i>	20
3.5.5 <i>Técnicas para Aprimorar a Solução</i>	21
3.6 MÉTODO DE JORDAN	21
3.6.1 <i>Cálculo de Determinantes</i>	22
3.7 MÉTODO DA FATORAÇÃO LU.....	22
3.7.1 <i>Aplicação da fatoração LU na solução de sistemas de equações lineares:</i>	24
3.8 MÉTODOS ITERATIVOS	25
3.8.1 <i>Introdução</i>	25
3.8.2 <i>Forma Geral</i>	26
3.9 MÉTODO DE JACOBI.....	26
3.10 MÉTODO DE GAUSS-SEIDEL	26
3.11 CONVERGÊNCIA DOS MÉTODOS ITERATIVOS	27

1 INTRODUÇÃO AO SCILAB

O programa **SCILAB** é um ambiente apropriado ao desenvolvimento de software para computação numérica. Esse programa foi concebido e é mantido pelo *Institut de Recherche en Informatique et en Automatique* (INRIA). O objetivo desta etapa do curso é apresentar os comandos básicos necessários ao desenvolvimento de programas simples, relativos aos algoritmos dos métodos numéricos estudados nas demais etapas desta disciplina.

As principais características que fazem o Scilab uma ferramenta de grande utilidade no aprendizado dos métodos numéricos, são:

- Interatividade com o usuário;
- Grande habilidade em operações com matrizes e vetores;
- Simplicidade de programação;
- Existência de *toolboxes*, com diversos métodos já programados, permitindo uma validação dos resultados obtidos com os programas desenvolvidos pelos estudantes;
- Distribuição gratuita.

1.1 Geração de Matrizes

A geração de matrizes pode ser feita através de:

Lista de elementos

Exemplo:

$A = [1 \ 2 \ 3; 4 \ 5 \ 6; 7 \ 8 \ 9]$

A =

!1 2 3!

!4 5 6!

!7 8 9!

Geração por comandos e funções

Exemplos:

$x = [-1.3 \ \text{sqrt}(3) \ (1+2+3)*4/5]$ ←

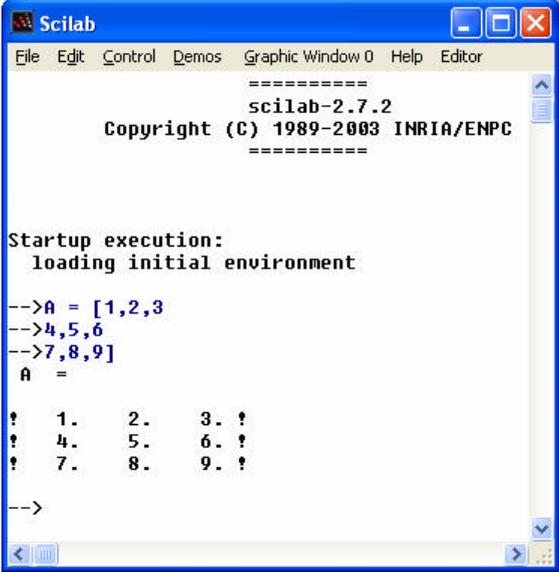
x =

!-1.3 1.7320508 4.8!

$x(5) = \text{abs}(x(1))$, obtendo-se:

x =

!-1.3 1.7320508 4.8 0. 1.3!



```
Scilab
File Edit Control Demos Graphic Window 0 Help Editor
=====
scilab-2.7.2
Copyright (C) 1989-2003 INRIA/ENPC
=====

Startup execution:
loading initial environment

-->A = [1,2,3
-->4,5,6
-->7,8,9]
A =
? 1. 2. 3. ?
? 4. 5. 6. ?
? 7. 8. 9. ?

-->
```

1.2 Variáveis

Variável = expressão

12.4/6.9

ans =

1.7971014

s = 1-1/2+1/3-1/4+1/5-1/6+1/7...

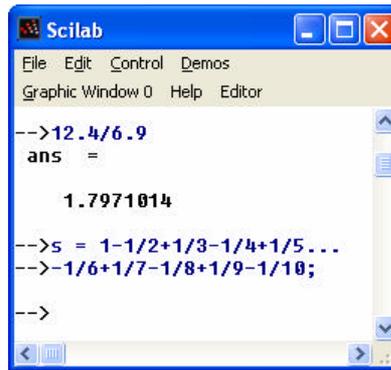
-1/8+1/9-1/10;

... ? Continua uma expressão em outra linha

; ? Ao final de uma expressão o cálculo é feito mas o resultado não é apresentado

O nome de uma variável pode ter no máximo 24 caracteres e o primeiro caracter tem que ser uma letra.

O SCILAB na forma normal é “case-sensitive” – variável *A* é diferente de *a*.



```
Scilab
File Edit Control Demos
Graphic Window 0 Help Editor
-->12.4/6.9
ans =
    1.7971014
-->s = 1-1/2+1/3-1/4+1/5...
-->-1/6+1/7-1/8+1/9-1/10;
-->
```

1.3 Comandos Elementares

- *whos* () → Lista e dimensiona as variáveis
- *clear* → Remove todas as variáveis do espaço de trabalho
- *who* → Lista as variáveis
- *predef* → Predefine e protege variáveis, evitando de ser excluídas com *clear*.

1.4 Números e Expressões Aritméticas

- Representação: 3, -3, 0.0001, 5.0e-20, 2.1e20
- Números complexos: `%i = sqrt(-1)`: $z = 3 + 4 * \%i$;
- Os cálculos são feitos internamente com 16 dígitos significativos (dupla precisão).

Operadores Aritméticos

+ ? Adição

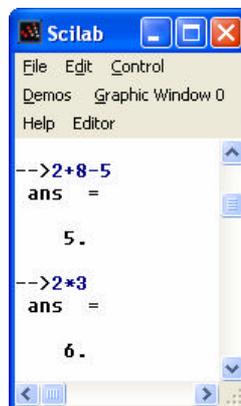
- ? Subtração

* ? Multiplicação

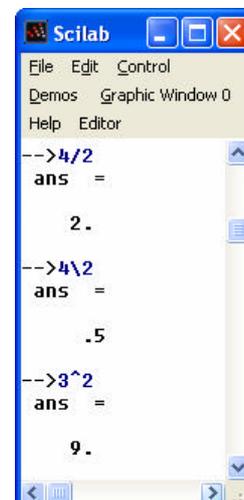
/ ? Divisão à direita

\ ? Divisão à esquerda

^ ? Potenciação



```
Scilab
File Edit Control
Demos Graphic Window 0
Help Editor
-->2+8-5
ans =
    5.
-->2*3
ans =
    6.
```



```
Scilab
File Edit Control
Demos Graphic Window 0
Help Editor
-->4/2
ans =
    2.
-->4\2
ans =
    .5
-->3^2
ans =
    9.
```

1.5 Formato de Saída

`%pi`

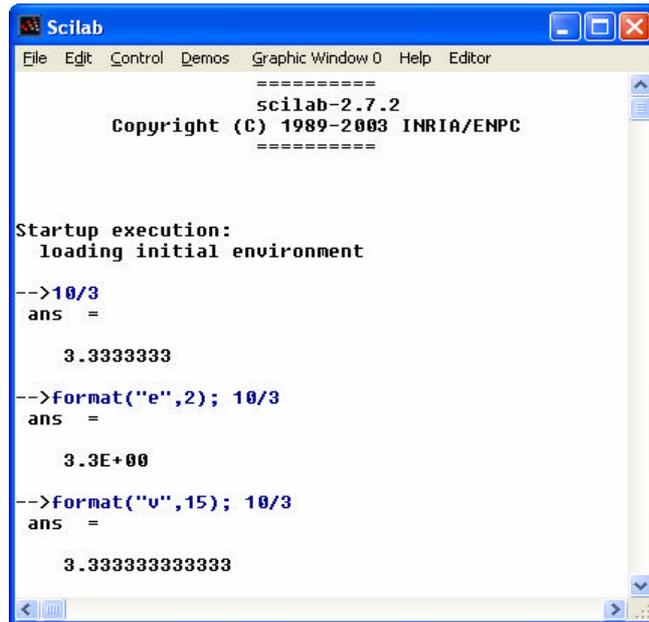
`%pi =`

3.1415927

`format ([type], [long])`

type: “e” (exponencial) ou “v”
(variável default)

long: n° máximo de dígitos
(default: 8)



```
Scilab
File Edit Control Demos Graphic Window 0 Help Editor
=====
scilab-2.7.2
Copyright (C) 1989-2003 INRIA/ENPC
=====

Startup execution:
loading initial environment

-->10/3
ans =
    3.3333333

-->format("e",2); 10/3
ans =
    3.3E+00

-->format("v",15); 10/3
ans =
    3.3333333333333
```

1.6 Outros Comandos

help: Informa sobre os comandos e funções do SCILAB.

Ex.: `help inv`, `help help`

quit: Encerra o Scilab. (`quit` ou `exit`)

save: Grava variáveis em um arquivo

Ex.: `save('varsalva', a)` ? Grava a variável a no arquivo 'varsalva'

load: Carrega as variáveis do arquivo.

Ex. `load varsalva`

1.7 Operações com Matrizes

- Transposta de uma matriz: $B = A'$
- Adição e Subtração: $C = A + B$
- Multiplicação: $C = A * B$, $C = a * A$
- Divisão:
 - Divisão à esquerda: $X = A \setminus B$ solução de $A * X = B$
 - Divisão à direita: $X = A / B$ solução de $X * A = B$

Obs.: A matriz deve ser quadrada com $\det(A) \neq 0$

1.8 Operações Elemento a Elemento

- Multiplicação: $z = x.*y$

Ex. 1: $c/x = [1\ 2\ 3]$; $y = [4\ 5\ 6]$;

$z = x.*y$

$z =$

!4 10 18!

Ex. 2: $x_1 = [1\ 2\ 3]$;

$uns = ones(size(x_1) = [1\ 1\ 1])$

$$x_1 = uns * x_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} [1\ 2\ 3] = \begin{bmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \\ 1 & 2 & 3 \end{bmatrix}$$

$$x_2 = [4\ 5\ 6], \quad x_2 = x_2 * uns = \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix} * [1\ 1\ 1] = \begin{bmatrix} 4 & 4 & 4 \\ 5 & 5 & 5 \\ 6 & 6 & 6 \end{bmatrix}$$

$$x_1 * x_2 = \begin{bmatrix} 32 & 32 & 32 \\ 32 & 32 & 32 \\ 32 & 32 & 32 \end{bmatrix} \quad x_1 \bullet * x_2 = \begin{bmatrix} 4 & 8 & 12 \\ 5 & 10 & 15 \\ 6 & 12 & 18 \end{bmatrix}$$

- Divisão: $z = x./y$ ou $z = y./x$

Ex.: $c/x = [1\ 2\ 3]$; $y = [4\ 5\ 6]$;

$z = x./y$

$z =$

!4.0000 2.5000 2.0000!

- Potência: $z = x.^y$

Ex.: $c/x = [1\ 2\ 3]$; $y = [4\ 5\ 6]$;

$z = x.^y$

$z =$

!1 32 729!

$z = x.^2$

$z =$

!1 4 9!

$z = 2.^[x\ y]$

$z =$

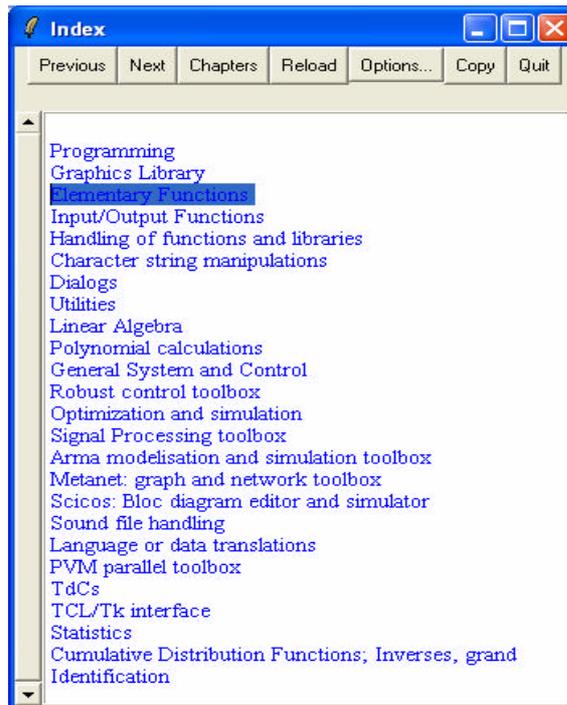
!2 4 8 16 32 64!

1.9 Operadores Relacionais

- < *Menor que*
- <= *Menor ou igual*
- > *Maior que*
- >= *Maior ou igual*
- = = *Igual*
- ~ = *Diferente*

1.10 Operadores Lógicos

- & *and*
- / *or*
- ~ *not*



1.11 Funções Elementares

Através do comando *HELP* do Scilab tem-se acesso a uma lista de funções elementares (*Elementary Functions*), cada uma acompanhada de uma breve descrição

Ex.: *sin, cos, abs, log, exp, etc.*

1.12 Vetores e Subscritos

Geração automática: $x = xi : dx : xf$

Ex.: $x = 1:5 \rightarrow x = [1 \ 2 \ 3 \ 4 \ 5]$

Ex.: $x = 3:-1:1 \rightarrow x = [3 \ 2 \ 1]$

Ex.: $x = 0:0.1:0.2 \rightarrow x = [0 \ 0.1 \ 0.2]$

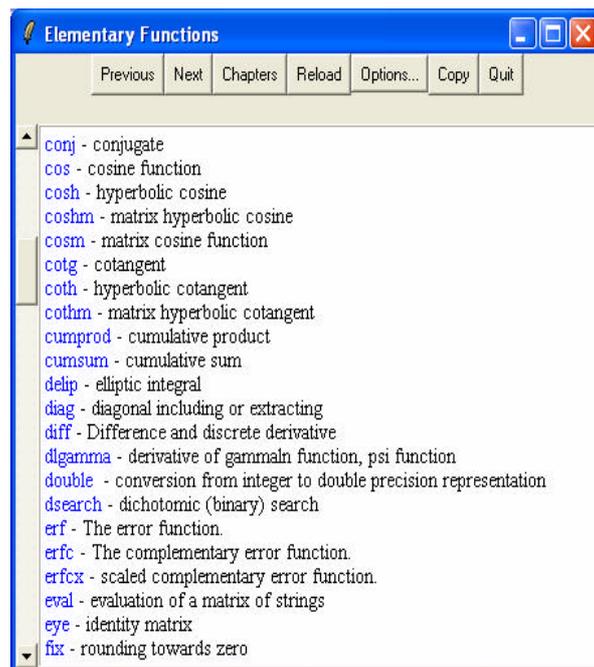
1.13 Geração de Tabelas de Valores

Dados dois vetores colunas x e y gera-se uma matriz $[x \ y]$

Ex.: $[0:0.1:0.5]'$; $y = [x.*\sin(x)]$; $[x \ y]$

ans =

```
!0.  0.      !
! .1  .0099833 !
! .2  .0397339 !
! .3  .0886561 !
! .4  .1557673 !
! .5  .2397128 !
```



1.14 Manipulação de Linhas e Colunas de Matrizes

O SCILAB apresenta grande facilidade na manipulação de vetores e matrizes, como mostra os exemplos abaixo:

Sejam **A** e **B** matrizes quadradas de ordem 10, então:

A(1:5,3) → Apresenta 5 elementos da 3ª coluna de A.

A(:,3) → Apresenta a 3ª coluna de A.

A(1:5,:) → Apresenta as 5 primeiras linhas de A.

A(1:5,7:10) → Apresenta as 5 primeiras linhas e as 4 últimas colunas de A, ou seja, apresenta uma sub-matriz de A contendo os 4 últimos elementos de cada uma das 5 primeiras linhas.

A(:,[3 5 10])=B(:,1:3) → Substitui a 3ª, 5ª e 10ª colunas de A pelas 3 primeiras colunas de B.

b = A(:) → Coloca todos os elementos da matriz A em um vetor coluna.

x = [] → Representa uma matriz vazia (dimensão zero).

A(:,[2 4]) = [] → Apaga as colunas 2 e 4. (A = [] apaga toda matriz).

size(A) → Fornece a dimensão da matriz; **Ex.: [m n] = size(A)**.

A = [1 2 3; 4 5 6; 7 8 9]; B = [A(:,1) A(:,3)]

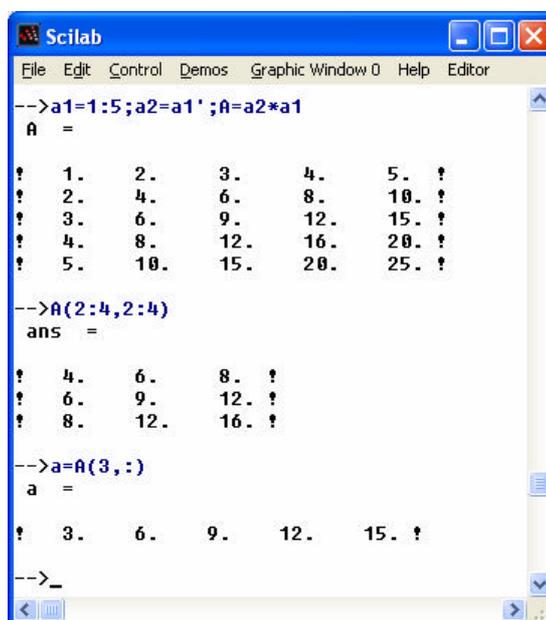
B =

```
!1. 3!  
!4. 6!  
!7. 9!
```

A = [A;[10 11 12]]

A =

```
!1 2 3!  
!4 5 6!  
!7 8 9!  
!10 11 12!
```



```
Scilab
File Edit Control Demos Graphic Window 0 Help Editor
-->a1=1:5;a2=a1';A=a2*a1
A =
? 1. 2. 3. 4. 5. ?
? 2. 4. 6. 8. 10. ?
? 3. 6. 9. 12. 15. ?
? 4. 8. 12. 16. 20. ?
? 5. 10. 15. 20. 25. ?

-->A(2:4,2:4)
ans =
? 4. 6. 8. ?
? 6. 9. 12. ?
? 8. 12. 16. ?

-->a=A(3,:)
a =
? 3. 6. 9. 12. 15. ?

-->_
```

1.15 Polinômios

x = poly(0, "x") → Define *x* como variável.

p = poly(v, "x", "flag") → Define *p* como um polinômio em *x*.

onde:

“flag” = “coeff” ou “roots”

v é um vetor contendo os coeficientes ou as raízes do polinômio

Ex.: Dado $p = y^3 - 6y^2 - 72y - 27$; para encontrar as raízes de p fazemos:

$p = \text{poly}([-27 -72 -6 1], "y", "coeff")$

$r = \text{roots}(p)$

Para encontrar o polinômio a partir das raízes faz-se:

$q = \text{poly}(r, "y", "roots")$

$q =$

$$-27 - 72y - 6y^2 + y^3$$

Para calcular o valor do polinômio para um determinado valor de y , faz-se: **horner(p, num)**, c/ **num** = valor desejado para y . Para calcular os valores do polinômio para diversos valores de y , faz-se: **horner(p, nums)** c/ **nums** = vetor com os diversos valores desejados para y .

```

Scilab
File Edit Control Demos Graphic Window 0 Help Editor
-->p = poly([-27 -72 -6 1], "y", "coeff")
p =
      2      3
- 27 - 72y - 6y + y

-->r= roots(p)
r =
! - .3883838 !
! - 5.7345099 !
!  12.122894 !

-->q=poly(r,"y","roots")
q =
      2      3
- 27 - 72y - 6y + y

-->_
  
```

1.16 Recursos Gráficos

O SCILAB dispõe de excelentes recursos gráficos permitindo a geração de gráficos 2D e 3D, além de uma série de outros recursos.

plot(y) → Plota o vetor y em função dos índices dos elementos de y .

plot(x,y) → Plota o vetor y em função do vetor x

plot(x,y,[xcap, ycap, caption])

onde: **x cap** → Título do eixo x ; **y cap** → Título do eixo y ; **caption** → Título do gráfico

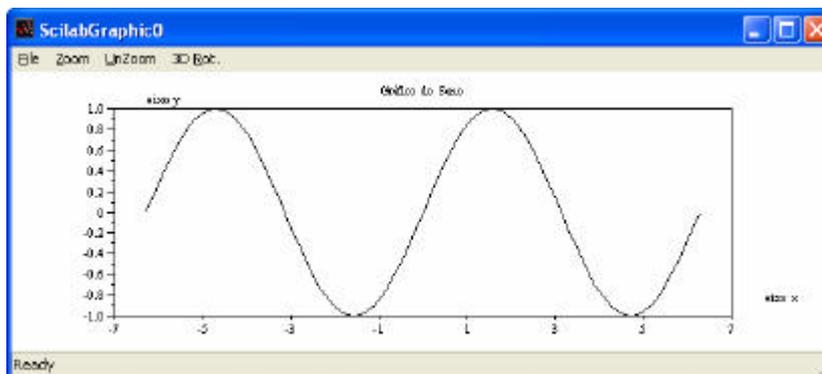
```

Scilab
File Edit Control Demos Graphic Window 0 Help Editor
=====
scilab-2.7.2
Copyright (C) 1989-2003 INRIA/ENPC
=====

Startup execution:
loading initial environment

-->x=-2*%pi:%pi/20:2*%pi;plot(x,sin(x),"eixo x","eixo y","Gráfico do Seno")

-->
  
```



1.17 Comandos para Controle de Fluxo

comando **for**

Ex.:

```
m=3;

n=3;

for i=1:m
    for j=1:n
        A(i,j)=1/(i+j-1);
    end
end
```

```
Scilab
File Edit Control Demos Graphic Window 0 Help Editor
=====
scilab-2.7.2
Copyright (C) 1989-2003 INRIA/ENPC
=====

Startup execution:
loading initial environment

-->n=3;n=3;

-->for i=1:m
-->for j=1:n
-->A(i,j)=1/(i+j-1);
-->end
-->end

-->A
A =
? 1.         .5         .3333333 ?
? .5         .3333333 .25        ?
? .3333333 .25        .2         ?
```

comando **while**

Ex.:

```
n=1;

while sqrt(n)<30
    n=n+1;
end
```

```
Scilab
File Edit Control Demos Graphic Window 0 Help Editor

-->n=1;

-->while sqrt(n)<30
-->n=n+1;
-->end

-->n
n =
900.
```

comando **if**

Ex.:

```
if n<0
    a=-n;
else
    a=n;
end
```

```
Scilab
File Edit Control Demos
Graphic Window 0 Help
Editor

-->n=-2;

-->if n<0
-->a=-n;
-->else
-->a=n;
-->end

-->a
a =
2.
```

```
Scilab
File Edit Control Demos
Graphic Window 0 Help
Editor

-->n=5;

-->if n<0
-->a=-n;
-->else
-->a=n;
-->end

-->a
a =
5.
```

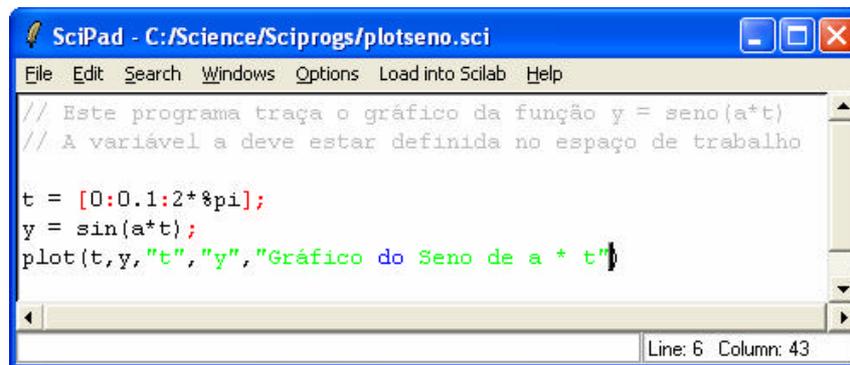
1.18 Programando no SCILAB

O SCILAB possui uma linguagem própria de programação sendo os programas armazenados em arquivos **.sci** ou **.sce**. Os programas são escritos, utilizando um editor de textos ASCII qualquer, como, por exemplo, o bloco de notas. Nas versões mais recentes, o SCILAB traz seu próprio editor ASCII, o **SciPad**, basta clicar sobre a opção **EDITOR**, da barra de menu do SCILAB.

Ex.: Programa para traçar a curva do seno de $a*t$.

plotseno.sci

```
// Este programa traça o gráfico da função y = seno(a*t)
// A variável a deve estar definida no espaço de trabalho
t = [0:0.1:2*%pi];
y = sin(a*t);
plot(t,y, "t", "y", "Gráfico do Seno de a * t")
```



The screenshot shows the SciPad editor window titled "SciPad - C:/Science/Sciprags/plotseno.sci". The menu bar includes File, Edit, Search, Windows, Options, Load into Scilab, and Help. The code is as follows:

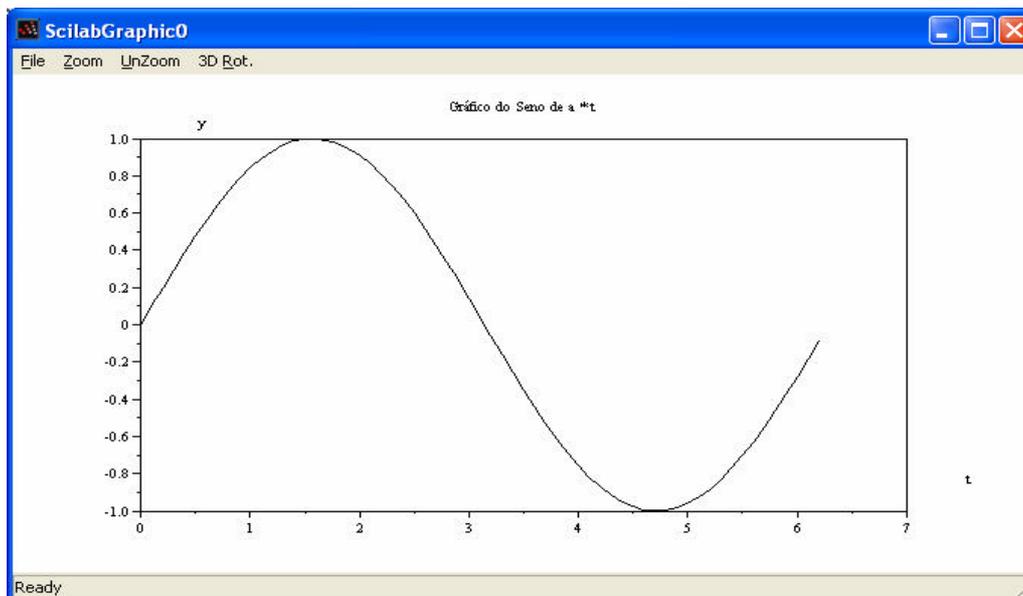
```
// Este programa traça o gráfico da função y = seno(a*t)
// A variável a deve estar definida no espaço de trabalho
t = [0:0.1:2*%pi];
y = sin(a*t);
plot(t,y, "t", "y", "Gráfico do Seno de a * t")
```

The status bar at the bottom right indicates "Line: 6 Column: 43".

No ambiente SCILAB digitam-se os comandos:

--> a = 1; exec("plotseno.sci")

O SCILAB abre uma janela gráfica na qual é mostrada a variação de y com t.



1.19 Funções no SCILAB

O SCILAB dispõe de um grande número de funções, que nada mais são do que programas com entrada de dados via argumentos. Ex.: $\sin(x)$.

É possível criar novas funções, basta editar um arquivo e escrever um programa como mostra o exemplo abaixo.

Ex.: Função para calculo da média

media.sce

```
function y = media(x)  
// Esta função calcula a média dos elementos de um vetor  
// ou o valor médio dos elementos de cada coluna  
// de uma matriz  
[m,n] = size(x);  
if m == 1  
    m = n;  
end  
y = sum(x)/m;  
endfunction
```

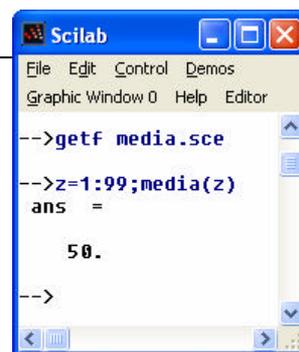
Pode-se, em seguida, utilizar essa função:

```
--> getf media.sce
```

```
--> z=1:99; media(z)
```

```
ans =
```

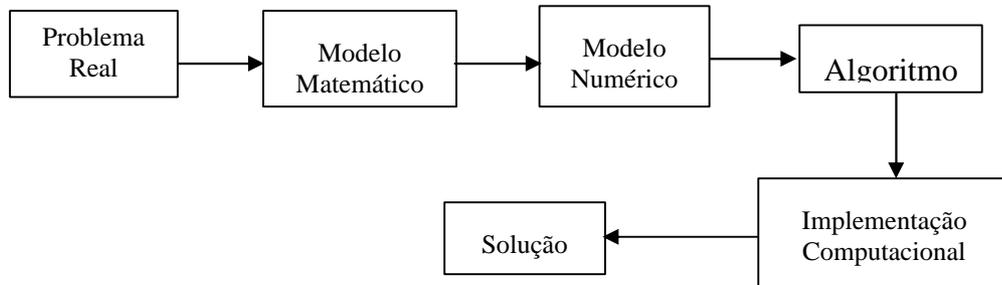
```
50.
```



2 REPRESENTAÇÃO NUMÉRICA E ERROS

2.1 Introdução e Definições

Fases do processo de solução de um problema real:



Na passagem de cada fase para a fase seguinte é inevitável que erros sejam incorporados.

Por exemplo, a transformação do problema real em um modelo matemático introduz erros devido à desconsideração de fenômenos com grau de incerteza elevado (resistência do ar, velocidade do vento).

Já nas transformações entre as etapas designadas por (b), (c), (d) e (e) existe um outro tipo de erro associado:

O erro numérico. Esse erro depende fundamentalmente do tipo de representação numérica, bem como do volume de cálculos efetuado.

O erro numérico pode ser formalmente definido como: $Ev = \text{Valor verdadeiro} - \text{Valor aproximado}$. Podendo ser classificado como:

- **Erro de truncamento:** é decorrente da representação de um processo infinito através de um processo finito.

Ex.: A avaliação de funções “implícitas” em computadores, tais como: exponencial, funções trigonométricas, etc., é realizada através do seu desenvolvimento em série de Taylor:

$$f(x+h) = f(x) + f'(x)h + f''(x)\frac{h^2}{2!} + f'''(x)\frac{h^3}{3!} + \dots + f^{(n)}(x)\frac{h^n}{n!}$$

- **Erro de arredondamento:** é proveniente da representação finita de um número em um computador.

O arredondamento pode ser efetuado de duas formas: **descarte**, ou, **assumindo o número significativo mais próximo**.

A representação científica de um número é feita na forma:

$$x \rightarrow m \times b^e$$

onde:

m ? mantissa

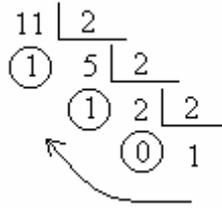
b ? base

e ? expoente

2.2 Mudança de Base

a) $11_{10} = ?_2$

$$\Rightarrow 11_{10} = 1011_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$$



b) $0,65_{10} = ?_2$

$$\Rightarrow 0,65_{10} = 0,101001\dots_2$$

$\frac{0,65}{2}$	$\frac{0,3}{2}$	$\frac{0,6}{2}$	$\frac{0,2}{2}$	$\frac{0,4}{2}$	$\frac{0,8}{2}$	$\frac{0,6}{2}$
$0,30$	$0,6$	$0,2$	$0,4$	$0,8$	$0,6$	$0,2$

c) $11,25_{10} = 11 + 0,25$

$$\Rightarrow 0,01_2$$

$\frac{0,25}{2}$	$\frac{0,50}{2}$
$0,50$	$0,00$

$$11,25_{10} = 1011_2 + 0,01_2 = 1011,01_2$$

$$\begin{aligned} 1011,01_2 &= 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} = \\ &= 8 + 0 + 2 + 1 + 0 + 0,25 = 11,25_{10} \end{aligned}$$

2.3 Representação Numérica

Representação Geral: $x = \pm \left[\frac{d_1}{b} + \frac{d_2}{b^2} + \dots + \frac{d_t}{b^t} \right] \cdot b^e$

onde:

d_i ? inteiros / $0 \leq d_i \leq b - 1$; $i = 1, 2, \dots, n$

t ? nº de dígitos significativos

e ? expoente ; $e_i \leq e \leq e_s$

Ex1: Para $b = 10$

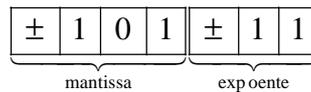
$$0,415_{10} = + \left[\frac{4}{10} + \frac{1}{10^2} + \frac{5}{10^3} \right] \cdot 10^0 \Rightarrow 41,5_{10} = 0,415_{10} \times 10^2 = \left[\frac{4}{10} + \frac{1}{10^2} + \frac{5}{10^3} \right] \cdot 10^2$$

Ex2: A representação binária do decimal 5 é: $101_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 5_{10}$.

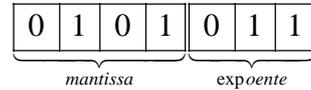
De acordo com a representação geral, tem-se: $101_2 = 0,101_2 \times 2^3 = \left[\frac{1}{2} + \frac{0}{2^2} + \frac{1}{2^3} \right] \cdot 2^3$, onde

o expoente 3 pode ainda ser escrito na forma binária: $3_{10} = 11_2$.

Assim, o número 5 pode ser caracterizado, na forma binária, pelo vetor:



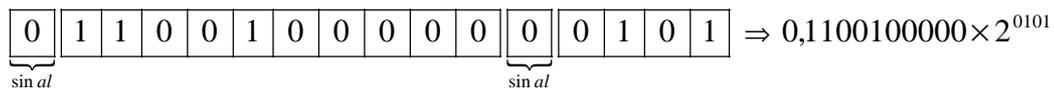
O sinal + ou - pode ser caracterizado por mais um dígito (bit); adota-se 0 para + e 1 para -. Assim:



Ex3: Uma calculadora possui um sistema de representação numérica de base $\beta = 2$, com $t = 10$ algarismos significativos da mantissa e expoentes inferior e superior $e_i = -15$ e $e_s = 15$, respectivamente. Verificar o nº de bits necessários à representação de um número, p. ex., 25.

$$25_{10} = 11001_2 = 0,11001_2 \times 2^5 = 0,\overbrace{11001}_{\text{mant}}_2 \times 2^{\overbrace{101}_{\text{exp}}}$$

O máximo valor absoluto de expoente a ser representado é $15_{10} = 1111_2$. Portanto, para o expoente, devem ser reservados 04 bits e mais 01 para o seu sinal. Contando com o bit do sinal da mantissa e com os 10 algarismos significativos, tem-se um total de 16 bits. A representação do número 25 ficaria:



O maior número com representação possível nessa máquina é: $0,1111111111 \cdot 2^{1111} = 32736_{10}$.

2.4 Arredondamento

É importante observar que, nem todos os números reais podem ser representados através do sistema acima. Por exemplo, o número 17 pode ser representado na seguinte forma:

$$17_{10} = 10001 = 0,10001 \times 2^{101} \equiv \boxed{0} \boxed{1} \boxed{0} \boxed{0} \boxed{0} \boxed{1} \boxed{} \boxed{} \boxed{} \boxed{} \boxed{} \boxed{} \boxed{0} \boxed{0} \boxed{1} \boxed{0} \boxed{1}$$

O número, imediatamente superior a 17, que pode ser representado, nesse sistema, é:

$$0,1000100001 \times 2^{101} \equiv \boxed{0} \boxed{1} \boxed{0} \boxed{0} \boxed{0} \boxed{1} \boxed{0} \boxed{0} \boxed{0} \boxed{0} \boxed{1} \boxed{0} \boxed{0} \boxed{1} \boxed{0} \boxed{1}$$

que corresponde a $17 + \frac{2^5}{2^{10}} = 17,0313$.

Portanto, o número 17,03 não pode ser representado de forma exata neste sistema. Nesse caso a representação é feita assumindo-se o número significativo mais próximo: $17 + 1/2^5$.

2.5 Precisão

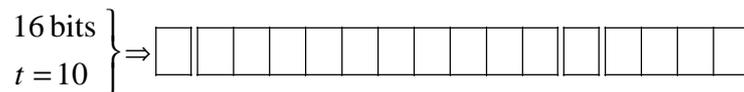
Define o número de casas decimais exatas da mantissa. É determinada pelo último bit da mantissa. No caso acima, tem-se:

$$\text{precisão} \leq \frac{1}{b^t} = \frac{1}{2^{10}} \approx 10^{-3}$$

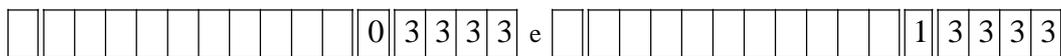
Exercício: Uma máquina de 16 bits possui $b = 4$ e $t = 10$. Determinar:

- O maior número que pode ser representado na máquina;
- A sua precisão.
- Representar o número 15,25 nessa máquina.

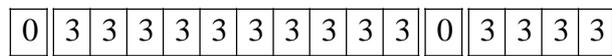
Solução:



Logo, como $b = 4$ ($b - 1 = 3$), os valores máximo e mínimo para o expoente serão, respectivamente:



e, o maior número que pode ser representado nesta máquina é:



que equivale:

$$x_{\text{máx}} = \left[\frac{3}{4} + \frac{3}{4^2} + \frac{3}{4^3} + \frac{3}{4^4} + \frac{3}{4^5} + \frac{3}{4^6} + \frac{3}{4^7} + \frac{3}{4^8} + \frac{3}{4^9} + \frac{3}{4^{10}} \right] \cdot 4^e \Rightarrow$$

$$x_{\text{máx}} = \left[3 \cdot 4^9 + 3 \cdot 4^8 + 3 \cdot 4^7 + 3 \cdot 4^6 + 3 \cdot 4^5 + 3 \cdot 4^4 + 3 \cdot 4^3 + 3 \cdot 4^2 + 3 \cdot 4^1 + 3 \cdot 4^0 \right] \cdot 4^{-10} \cdot 4^e \Rightarrow$$

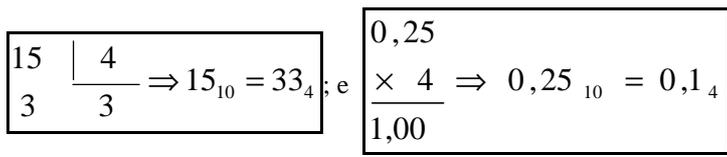
$$x_{\text{máx}} = [1.048.575] \cdot 4^{-10} \cdot 4^e \approx 1 \cdot 4^e \Rightarrow \boxed{x_{\text{máx}} \approx 4^e}$$

onde: $e = \left[\frac{3}{4} + \frac{3}{4^2} + \frac{3}{4^3} + \frac{3}{4^4} \right] \cdot 4^4 = [3 \cdot 4^3 + 3 \cdot 4^2 + 3 \cdot 4^1 + 3 \cdot 4^0] \cdot 4^{-4} \cdot 4^4 \Rightarrow \boxed{e = 255 \cdot 4^{-4} \cdot 4^4 = 255}$

logo: $x_{\text{máx}} \approx 4^{255} = 3,3520 \times 10^{153}$

A precisão da máquina em questão é dada por: $\text{precisão} \leq \frac{1}{b^t} = \frac{1}{4^{10}} = 9,5367 \times 10^{-7} \approx 10^{-6}$

O número 15,25 é representado nessa máquina, por:



logo: $15,25_{10} = 33,1_4 = (0,331 \times 4^2)_4 \Rightarrow \boxed{0 \ 3 \ 3 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 2}$

3 SISTEMA DE EQUAÇÕES ALGÉBRICAS LINEARES

3.1 Introdução

São inúmeros os problemas de engenharia que recaem na solução de um sistema de equações lineares.

- Como exemplos, podemos citar:
- O cálculo de esforços em problemas de estática;
- O cálculo de tensões e correntes em um circuito elétrico composto por elementos lineares;
- O balanço de massa em sistemas físicos lineares;
- A solução de equações diferenciais lineares por métodos numéricos como elementos finitos e diferenças finitas, etc.

3.2 Formulação

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2$$

Forma geral: $a_{31}x_1 + a_{32}x_2 + \dots + a_{3n}x_n = b_3$:

$$\vdots \quad \quad \quad \vdots \quad \quad \quad \vdots \quad \quad \quad \vdots$$

$$a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n$$

onde a e b são constantes e n é o número de equações.

Representação Matricial:
$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & \cdot \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \cdot \\ \cdot \\ b_n \end{pmatrix}; \text{ ou: } \mathbf{Ax} = \mathbf{b}$$

3.3 Classificação dos Sistemas

Incompatível. Ex.:
$$\begin{cases} 2x_1 + x_2 = 6 \\ 4x_1 + 2x_2 = 18 \end{cases}$$

Compatível:
$$\begin{cases} \text{Determinado - solução única} \\ \text{Indeterminado - diversas soluções.} \\ \text{Homogêneo (b = 0)} \end{cases}$$

3.4 Métodos de Solução

Classificam-se, numericamente, em Diretos e Iterativos.

Os métodos mais comuns são:

$$\begin{array}{l}
 - \text{Diretos} \left\{ \begin{array}{l} \text{Método de Gauss} \\ \text{Método da Eliminação de Jordan} \\ \text{Fatoração LU} \end{array} \right. \\
 \\
 - \text{Iterativos} \left\{ \begin{array}{l} \text{Método de Jacobi} \\ \text{Método de Gauss – Seidel} \end{array} \right.
 \end{array}$$

3.5 Método de Gauss

3.5.1 Princípio do método:

Dado um sistema linear $\mathbf{Ax} = \mathbf{b}$, obter, através de transformações elementares um sistema equivalente, $\mathbf{Tx} = \mathbf{c}$, onde \mathbf{T} é uma matriz triangular superior. Em seguida calculam-se os elementos do vetor \mathbf{x} através de um processo de substituição reversa.

3.5.2 Transformações Elementares

- Troca da ordem das equações;
- Multiplicação de uma equação por um real não nulo;
- Substituição de uma equação por uma combinação linear dela mesma com uma outra;

3.5.3 Substituição Reversa

Dado um sistema na forma triangular superior:

$$\begin{bmatrix} 1 & & & & \\ 0 & 1 & & & \\ \vdots & 0 & \ddots & & \\ \vdots & \vdots & & \ddots & \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ \vdots \\ \vdots \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1^{(n-1)} \\ \vdots \\ \vdots \\ \vdots \\ b_n^{(n-1)} \end{bmatrix}$$

A última equação permite determinar diretamente:

$$x_n = b_n^{(n-1)}$$

A penúltima equação fica:

$$x_{n-1} + a_{n-1,n}^{(n-1)} \cdot x_n = b_{n-1}^{(n-1)} \quad \therefore \quad x_{n-1} = b_{n-1}^{(n-1)} - a_{n-1,n}^{(n-1)} \cdot x_n$$

Generalizando, tem-se:

$$\boxed{x_j = b_j^{(n-1)} - \sum_{k=j+1}^n a_{jk}^{(n-1)} x_k}$$

3.5.4 Triangulação

$$\text{Ex.: } \begin{cases} 2x_1 + 3x_2 - x_3 = 5 \\ 4x_1 + 4x_2 - 3x_3 = 3 \\ 2x_1 - 3x_2 + x_3 = -1 \end{cases} \Leftrightarrow \begin{bmatrix} 2 & 3 & -1 \\ 4 & 4 & -3 \\ 2 & -3 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 5 \\ 3 \\ -1 \end{bmatrix} \Leftrightarrow A \cdot x = b$$

$$\text{Inicialmente, escreve-se a matriz aumentada do sistema (B = [A | b]): } B^{(0)} = \left[\begin{array}{ccc|c} 2 & 3 & -1 & 5 \\ 4 & 4 & -3 & 3 \\ 2 & -3 & 1 & -1 \end{array} \right]$$

Tomando-se o elemento da diagonal principal, $a_{11}^{(0)}$, como pivô, encontra-se os multiplicadores para as linhas seguintes:

$$m_{21}^{(0)} = -\frac{a_{21}^{(0)}}{a_{11}^{(0)}} = -\frac{4}{2} = -2$$

$$m_{31}^{(0)} = -\frac{a_{31}^{(0)}}{a_{11}^{(0)}} = -\frac{2}{2} = -1$$

Em seguida, substituem-se as linhas originais pela seguinte combinação linear:

$$\begin{cases} L_1^{(1)} = L_1^{(0)} & [2 \ 3 \ -1 \ | \ 5] \\ L_2^{(1)} = m_{21}^{(0)} L_1^{(0)} + L_2^{(0)} \Rightarrow [-4 \ -6 \ 2 \ | \ -10] + [4 \ 4 \ -3 \ | \ 3] \Rightarrow B^{(1)} = \left[\begin{array}{ccc|c} 2 & 3 & -1 & 5 \\ 0 & -2 & -1 & -7 \\ 0 & -6 & 2 & -6 \end{array} \right] \\ L_3^{(1)} = m_{31}^{(0)} L_1^{(0)} + L_3^{(0)} & [-2 \ -3 \ 1 \ | \ -5] + [2 \ -3 \ 1 \ | \ -1] \end{cases}$$

Toma-se agora o elemento da diagonal principal, na linha seguinte, $a_{22}^{(1)}$, como pivô e encontra-se o multiplicador para as linhas que se sucedem a esta.

$$m_{32}^{(1)} = -\frac{a_{32}^{(1)}}{a_{22}^{(1)}} = -\frac{-6}{-2} = -3$$

Em seguida, substituem-se as linhas originais pela seguinte combinação linear:

$$\begin{cases} L_1^{(2)} = L_1^{(1)} & [2 \ 3 \ -1 \ | \ 5] \\ L_2^{(2)} = L_2^{(1)} & [0 \ -2 \ -1 \ | \ -7] \\ L_3^{(2)} = m_{32}^{(1)} L_2^{(1)} + L_3^{(1)} & [0 \ 6 \ 3 \ | \ 21] + [0 \ -6 \ 2 \ | \ -6] \end{cases} \Rightarrow B^{(2)} = \left[\begin{array}{ccc|c} 2 & 3 & -1 & 5 \\ 0 & -2 & -1 & -7 \\ 0 & 0 & 5 & 15 \end{array} \right]$$

Segue-se este procedimento até que se chegue ao pivô da penúltima linha da matriz, obtendo-se assim uma matriz com apenas elementos nulos abaixo da diagonal principal. Uma vez obtida uma matriz nesta forma, procedendo-se uma substituição reversa obtêm-se a solução do sistema triangular superior, representado, neste exemplo, pela matriz aumentada $B^{(2)}$:

$$B^{(2)} = \left[\begin{array}{ccc|c} 2 & 3 & -1 & 5 \\ 0 & -2 & -1 & -7 \\ 0 & 0 & 5 & 15 \end{array} \right] \Leftrightarrow \begin{cases} 2x_1 + 3x_2 - x_3 = 5 \\ -2x_2 - x_3 = -7 \\ 5x_3 = 15 \end{cases} \Rightarrow \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

É importante relembrar que, esta é também a solução do sistema original.

Uma forma prática e compacta de realizar a triangulação é através da construção de tabelas.

Linha	Multiplicador	Matriz B		Transformação
		Matriz A	Vetor b	
1		$\langle 2 \rangle$ 3 -1	5	
2	$-4/2 = -2$	4 4 -3	3	
3	$-2/2 = -1$	2 -3 1	-1	
4		0 $\langle -2 \rangle$ -1	-7	$\rightarrow 2L_1 + L_2$
5	$-(-6/-2) = -3$	0 -6 2	-6	$\rightarrow -1L_1 + L_3$
6		0 0 5	15	$\rightarrow -3L_4 + L_5$

3.5.5 Técnicas para Aprimorar a Solução

Essas técnicas são úteis não apenas para o método de Gauss, mas também para métodos similares.

1. Uso de mais dígitos significativos (dupla precisão);
2. Pivotamento: Identificar se existem pivôs de pequeno valor e trocar linhas e/ou colunas de forma a ter os elementos da diagonal diferentes de zero e de maior valor absoluto. Isto minimiza o erro de arredondamento.
3. Escalonamento: Minimiza o erro de arredondamento. Uma matriz triangular cujos elementos da diagonal principal são diferentes da unidade, pode ter cada uma de suas linhas divididas pelo seu respectivo elemento na diagonal principal.

Obs.: Em alguns problemas é necessário um estudo de condicionamento da matriz e um processo de otimização da solução se faz necessário.

3.6 Método de Jordan

Dado um sistema, o Método de Jordan consiste em obter, através de transformações elementares, um sistema equivalente cuja matriz de coeficientes seja diagonal.

$$\text{Ex.: } \begin{cases} x_1 + x_2 + 2x_3 = 4 \\ 2x_1 - x_2 - x_3 = 0 \\ x_1 - x_2 - x_3 = -1 \end{cases} \Leftrightarrow \begin{bmatrix} 1 & 1 & 2 \\ 2 & -1 & -1 \\ 1 & -1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ 0 \\ -1 \end{bmatrix} \Leftrightarrow A \cdot x = b$$

Lin	Multiplicador	Matriz B		Transformação
		Matriz A	Vet. b	
1	$-\frac{2}{1} = -2$	$\langle 1 \rangle$ 1 2	4	
2	$\frac{1}{1}$	2 -1 -1	0	
3	$-\frac{1}{1} = -1$	1 -1 -1	-1	
4	$-\frac{1}{-3} = \frac{1}{3}$	1 1 2	4	$\rightarrow L_1$
5	$-\frac{-2}{-3} = -\frac{2}{3}$	0 $\langle -3 \rangle$ -5	-8	$\rightarrow -2L_1 + L_2$
6	$-\frac{-2}{-3} = -\frac{2}{3}$	0 -2 -3	-5	$\rightarrow -1L_1 + L_3$
7	$-\frac{1/3}{1/3} = -1$	1 0 1/3	4/3	$\rightarrow (1/3)L_5 + L_4$
8	$\frac{1}{3}$	0 -3 -5	-8	$\rightarrow L_5$
9	$-\frac{-5}{1/3} = 15$	0 0 $\langle 1/3 \rangle$	1/3	$\rightarrow (-2/3)L_5 + L_6$
10		1 0 0	1	$\rightarrow -1L_9 + L_7$
11		0 -3 0	-3	$\rightarrow 15L_9 + L_8$
12		0 0 1/3	1/3	$\rightarrow L_9$

3.6.1 Cálculo de Determinantes

Uma maneira simples de calcular o determinante de uma matriz **A** consiste em encontrar uma matriz **B**, triangular ou diagonal, que seja obtida a partir de **A**, através de transformações elementares.

Demonstra-se que, se **A** e **B** são equivalentes, então: $\det(A) = \det(B)$

$$\text{Ex.: } A = \begin{bmatrix} 2 & 3 & -1 \\ 4 & 4 & -3 \\ 2 & -3 & 1 \end{bmatrix} \begin{matrix} \rightarrow L_1 \\ \rightarrow L_2 - 2L_1 \\ \rightarrow L_3 - 3L_2 \end{matrix} \Rightarrow \bar{A} = \begin{bmatrix} 2 & 3 & -1 \\ 0 & -2 & -1 \\ 0 & -6 & 2 \end{bmatrix} \begin{matrix} \rightarrow L_1 \\ \rightarrow L_2 \\ \rightarrow L_3 - 3L_2 \end{matrix} \Rightarrow$$

$$B = \begin{bmatrix} 2 & 3 & -1 \\ 0 & -2 & -1 \\ 0 & 0 & 5 \end{bmatrix} \Rightarrow \det(A) = \det(B) = 2 \cdot (-2) \cdot 5 = -20$$

3.7 Método da Fatoração LU

Seja a matriz **A** dada abaixo, que deve ser fatorada em duas matrizes triangulares, uma superior e a outra, inferior.

$$A = L \cdot U = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix};$$

$$\text{sendo: } \mathbf{L} = \begin{bmatrix} l_{11} & 0 & \cdots & 0 \\ l_{21} & l_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{bmatrix} \text{ e } \mathbf{U} = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{nn} \end{bmatrix}$$

Efetuando o produto $\mathbf{L} \cdot \mathbf{U}$ e igualando os elementos da matriz produto aos elementos correspondentes em \mathbf{A} , obtém-se n^2 equações, envolvendo os elementos de \mathbf{L} e de \mathbf{U} . Observa-se, entretanto, que cada uma das matrizes triangulares possui $\frac{n^2 - n}{2} + n$ elementos desconhecidos, definindo-se um total de $(n^2 + n)$ incógnitas, número esse maior que o número de equações que se precisa resolver. Isso significa que infinitas matrizes \mathbf{L} e \mathbf{U} são soluções do problema de fatoração. Para contornar o problema de que o número de incógnitas é maior que o número de equações, costuma-se atribuir valores aos elementos da diagonal de uma das matrizes triangulares.

Apresentamos a seguir a solução proposta por Banachiwitz. A fim de simplificar os cálculos, atribui-se o valor 1 aos elementos da diagonal principal de \mathbf{L} : $l_{ii} = 1, \quad i = 1, \dots, n$. Com isso, fica definida a 1ª linha da matriz \mathbf{U} , de acordo com as equações abaixo:

$$\begin{cases} a_{11} = 1 \cdot u_{11} \\ a_{12} = 1 \cdot u_{12} \\ \vdots \\ a_{1n} = 1 \cdot u_{1n} \end{cases}$$

Uma vez definido o valor de u_{1j} , pode-se então calcular toda a 1ª coluna da matriz \mathbf{L} , através das equações definidas pelo produto das linhas de \mathbf{L} pela primeira coluna de \mathbf{U} :

$$\begin{cases} l_{11} = 1 \\ l_{21} = a_{21}/u_{11} \\ \vdots \\ l_{n1} = a_{n1}/u_{11} \end{cases}$$

Agora, multiplicando-se a 2ª linha de \mathbf{L} pela 2ª coluna de \mathbf{U} , de acordo com:

$$\begin{cases} u_{22} = a_{22} - l_{21}u_{12} \\ u_{23} = a_{23} - l_{21}u_{13} \\ \vdots \\ u_{2n} = a_{2n} - l_{21}u_{1n} \end{cases}$$

Analogamente, multiplicando as linhas restantes de \mathbf{L} pela 2ª coluna de \mathbf{U} , determinam-se os elementos restantes da 2ª coluna de \mathbf{L} , i. e., os elementos situados abaixo de $l_{22} = 1$, de acordo com:

$$l_{i2} = \frac{1}{u_{22}}(a_{i2} - l_{i1}u_{12}), \quad i = 3, \dots, n$$

Notar que os elementos da 2ª coluna de \mathbf{L} não dependem dos elementos da 2ª linha de \mathbf{U} , exceto do elemento da diagonal, u_{22} .

3.7.1 Aplicação da fatoração LU na solução de sistemas de equações lineares:

Seja o sistema: $\mathbf{Ax} = \mathbf{b}$. Fatorando-se a matriz $\mathbf{A} = \mathbf{L} \cdot \mathbf{U}$, tem-se: $\mathbf{L} \cdot \mathbf{U} \cdot \mathbf{x} = \mathbf{b} \hat{=} \mathbf{L} \cdot (\mathbf{U} \cdot \mathbf{x}) = \mathbf{b}$.

Definindo-se: $\mathbf{U} \cdot \mathbf{x} = \mathbf{y}$

tem-se: $\mathbf{L} \cdot \mathbf{y} = \mathbf{b}$

Ambos os sistemas são triangular. Resolvendo-se $\mathbf{L} \cdot \mathbf{y} = \mathbf{b}$, por substituição direta, determina-se \mathbf{y} . Em seguida, resolve-se o outro sistema triangular, $\mathbf{U} \cdot \mathbf{x} = \mathbf{y}$, através de substituição reversa, para finalmente encontrar-se o vetor solução, \mathbf{x} .

Ex.: Resolver o sistema abaixo, através do método da fatoração LU:

$$\begin{bmatrix} -3 & 5 & 8 & -1 \\ 2 & -1 & 5 & 0 \\ 10 & -5 & 0 & 5 \\ 4 & 2 & -1 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 27 \\ 15 \\ 20 \\ 9 \end{bmatrix}$$

Após o 1º passo, tem-se:

$$U^{(1)} = \begin{bmatrix} -3 & 5 & 8 & -1 \\ 0 & u_{22} & u_{23} & u_{24} \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & 0 & u_{44} \end{bmatrix} \text{ e } L^{(1)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -2/3 & 1 & 0 & 0 \\ -10/3 & l_{32} & 1 & 0 \\ -4/3 & l_{42} & l_{43} & 1 \end{bmatrix}$$

Em seguida, calculam-se:

$$(I) \begin{cases} u_{22} = a_{22} - l_{21}u_{12} = 7/3 \\ u_{23} = a_{23} - l_{21}u_{13} = 31/3 \\ u_{24} = a_{24} - l_{21}u_{14} = -2/3 \end{cases} \text{ e } (II) \begin{cases} l_{32} = \frac{1}{u_{22}}[a_{32} - l_{31}u_{12}] = 5 \\ l_{42} = \frac{1}{u_{22}}[a_{42} - l_{41}u_{12}] = 26/7 \end{cases}$$

Com isso, tem-se:

$$U^{(2)} = \begin{bmatrix} -3 & 5 & 8 & -1 \\ 0 & 7/3 & 31/3 & -2/3 \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & 0 & u_{44} \end{bmatrix} \text{ e } L^{(2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -2/3 & 1 & 0 & 0 \\ -10/3 & 5 & 1 & 0 \\ -4/3 & 26/7 & l_{43} & 1 \end{bmatrix}$$

Notar que, nesse estágio dos cálculos (e, desde o princípio), existe um elemento nulo na diagonal. Isso não se constitui um obstáculo para a continuidade do processo de fatoração. A determinação da 3ª linha de \mathbf{U} e da 3ª coluna de \mathbf{L} é feita através das equações:

$$\begin{aligned} u_{33} &= a_{33} - l_{31}u_{13} - l_{32}u_{23} = -25 \\ u_{34} &= a_{34} - l_{31}u_{14} - l_{32}u_{24} = 5 \end{aligned} \text{ e } l_{43} = \frac{1}{u_{33}}[a_{43} - l_{41}u_{13} - l_{42}u_{23}] = 201/175$$

Dessa forma, após o 3º passo da fatoração, tem-se:

$$U^{(3)} = \begin{bmatrix} -3 & 5 & 8 & -1 \\ 0 & 7/3 & 31/3 & -2/3 \\ 0 & 0 & -25 & 5 \\ 0 & 0 & 0 & u_{44} \end{bmatrix} \text{ e } L^{(3)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -2/3 & 1 & 0 & 0 \\ -10/3 & 5 & 1 & 0 \\ -4/3 & 26/7 & 201/175 & 1 \end{bmatrix}$$

Finalmente, o elemento u_{44} é determinado: $u_{44} = a_{44} - l_{41}u_{14} - l_{42}u_{24} = -126/35$, obtendo-se:

$$U^{(4)} = \begin{bmatrix} -3 & 5 & 8 & -1 \\ 0 & 7/3 & 31/3 & -2/3 \\ 0 & 0 & -25 & 5 \\ 0 & 0 & 0 & -126/35 \end{bmatrix} \text{ e } L^{(4)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -2/3 & 1 & 0 & 0 \\ -10/3 & 5 & 1 & 0 \\ -4/3 & 26/7 & 201/175 & 1 \end{bmatrix}$$

É importante notar que, com exceção dos elementos das diagonais principais, os demais elementos de cada uma das matrizes ocupam o espaço dos zeros na outra matriz. Além disso, como a diagonal principal de \mathbf{L} é fixada como tendo todos os seus elementos iguais a um, podemos armazenar todos os valores calculados, para ambas as matrizes, em uma única matriz com dimensão igual a da matriz \mathbf{A} . Caso seja necessário economizar espaço de memória, as matrizes \mathbf{L} e \mathbf{U} poderão, inclusive, ser armazenadas no mesmo espaço de memória reservado para a matriz \mathbf{A} .

$$A^{(4)} = \begin{bmatrix} & -3 & 5 & 8 & -1 \\ & & 7/3 & 31/3 & -2/3 \\ -2/3 & & & -25 & 5 \\ -10/3 & 5 & & & -126/35 \\ -4/3 & 26/7 & 201/175 & & \end{bmatrix}$$

Resolvendo $\mathbf{L}\mathbf{y} = \mathbf{b}$, através de substituição direta, obtém-se: $\mathbf{y} = \begin{bmatrix} 27 \\ 33 \\ -55 \\ -504/35 \end{bmatrix}$

Resolvendo agora $\mathbf{U}\mathbf{x} = \mathbf{y}$, através de substituição reversa, determina-se: $\mathbf{x} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$

3.8 Métodos Iterativos

3.8.1 Introdução

Considere o sistema de equações lineares $\mathbf{Ax} = \mathbf{b}$, cuja solução é o vetor \mathbf{x}^* . Esta solução deve ser obtida como o limite para o qual converge a sequência $\{x^{(1)}, \dots, x^{(t)}, \dots\}$. Tal sequência é gerada a partir de uma aproximação inicial $\mathbf{x}^{(0)}$ e obedece a uma regra preestabelecida. Esse procedimento caracteriza os métodos iterativos.

3.8.2 Forma Geral

Os métodos iterativos para solução de sistemas lineares apresentam a seguinte forma geral: $\mathbf{x} = \mathbf{C}\mathbf{x} + \mathbf{d}$, onde \mathbf{C} é uma matriz ($n \times n$) e \mathbf{d} um vetor ($n \times 1$). Assim, partindo de uma aproximação de \mathbf{x} , digamos $\mathbf{x}^{(t)}$, pode-se obter uma melhor aproximação $\mathbf{x}^{(t+1)}$, através de:

$$\boxed{x^{(t+1)} = Cx^{(t)} + d}$$

3.9 Método de Jacobi

Utilizando a equação de iteração e as matrizes \mathbf{C} e \mathbf{d} definidas anteriormente, define-se o método de Jacobi através dos seguintes passos:

1. Estima-se uma aproximação inicial $\mathbf{x}^{(0)}$ para x ;
2. Geram-se aproximações sucessivas $\mathbf{x}^{(t)}$, a partir da equação de iteração, até que: $\max_i |\mathbf{x}_i^{(t+1)} - \mathbf{x}_i^{(t)}| < \mathbf{e}$. Onde \mathbf{e} é a tolerância predefinida.

OBS.: Um critério adicional de parada pode ser ainda introduzido, baseado no estabelecimento de um número máximo admissível de iterações.

$$\left. \begin{array}{l} \text{Note que: } Ax = b \\ x = Cx + d \end{array} \right\} \Rightarrow \left\{ \begin{array}{l} d = \begin{bmatrix} b_1 & b_2 & \dots & b_n \\ a_{11} & a_{22} & \dots & a_{nn} \end{bmatrix} \\ C = \begin{bmatrix} 0 & -\frac{a_{12}}{a_{11}} & \dots & -\frac{a_{1n}}{a_{11}} \\ -\frac{a_{21}}{a_{22}} & 0 & \dots & -\frac{a_{2n}}{a_{22}} \\ \vdots & \vdots & \ddots & \vdots \\ -\frac{a_{n1}}{a_{nn}} & -\frac{a_{n2}}{a_{nn}} & \dots & 0 \end{bmatrix} \end{array} \right.$$

3.10 Método de Gauss-Seidel

Esse método utiliza sempre as últimas atualizações de cada variável, na equação de iteração.

$$x_i^{(t+1)} = d_i + \left\{ \sum_{k=1}^{i-1} c_{ik} x_k^{(t+1)} + \sum_{k=i+1}^n c_{ik} x_k^{(t)} \right\}$$

Em geral, esse procedimento acelera a convergência do sistema. O método é adequado para sistemas com um número elevado de equações desde que ele atenda a condição da matriz \mathbf{A} ser diagonalmente dominante. Isto se aplica, por exemplo, a sistemas de equações lineares gerados quando da aplicação do método das diferenças finitas ou no cálculo de *splines*.

Uma das variações desse método é o chamado método da sobre-relaxação, com equação iterativa mostrada abaixo:

$$x^{(k+1)} = \lambda x_{G.S}^{(k+1)} + (1 - \lambda) x^{(k)}$$

onde λ ($0 < \lambda < 1$) é um fator peso com finalidade de acelerar a convergência.

3.11 Convergência dos métodos iterativos

Seja o sistema: $x = Cx + d$

Subtraindo dessa equação a equação de iteração, tem-se: $x^{(t+1)} - x = C(x^{(t)} - x)$

Definindo-se o erro da k -ésima iteração com: $e^{(t)} = x^{(t)} - x$, tem-se: $e^{(t+1)} = Ce^{(t)}$

Teorema: Partindo da equação anterior, demonstra-se que:

A condição $\sum_{i=1}^n |C_{ij}| \leq L < 1$, $j = 1, \dots, n$ é suficiente para que $x^{(t+1)} = Cx^t + d$ convirja. Como consequência, o critério definido por $|a_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}|$, $i = 1, \dots, n$ garante a convergência.

A desigualdade apresentada no teorema, define o “critério das linhas”. E, a matriz que satisfaz a esse critério é conhecida como “*matriz diagonalmente dominante*”.