

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:

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

Geração de Matrizes

A geração de matrizes pode ser feita através de:

a) Lista de elementos

Exemplo:

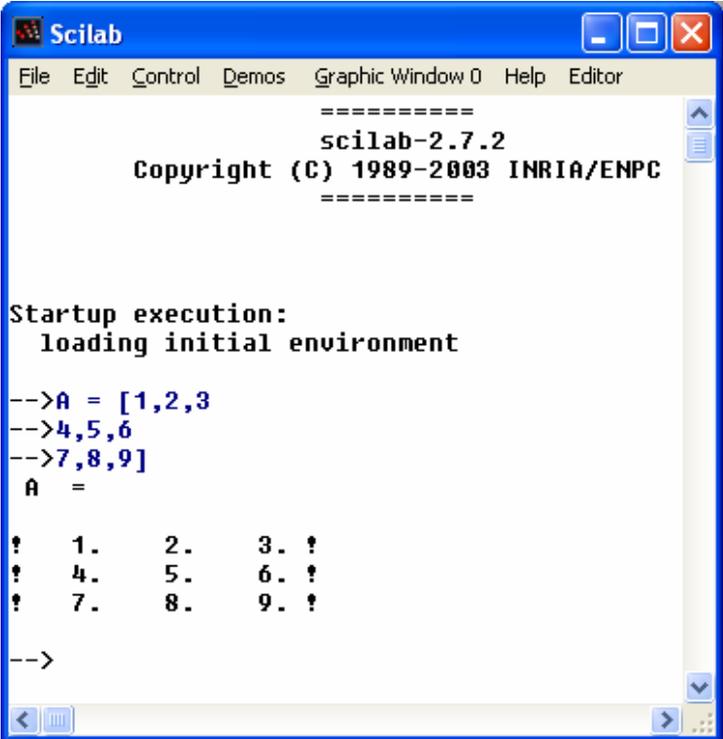
$A = [1 \ 2 \ 3; 4 \ 5 \ 6; 7 \ 8 \ 9]$

$A =$

!1 2 3!

!4 5 6!

!7 8 9!



```
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. ?

-->
```

b) Geração por comandos e funções

- Lista de Elementos
- Geração por comandos e funções
- Em programas

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!

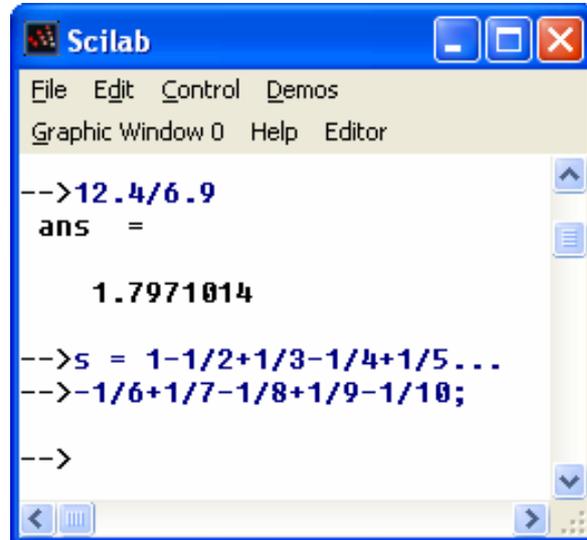
Variáveis

Variável = expressão

a) $12.4/6.9$

ans =

1.7971014



```
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;
-->
```

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

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

... → 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 .

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*.

Números e Expressões Aritméticas

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

Operadores Aritméticos

+ Adição

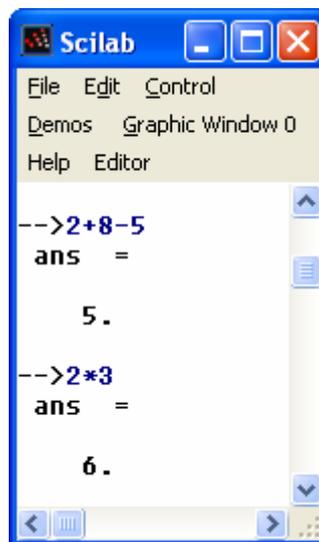
- Subtração

* Multiplicação

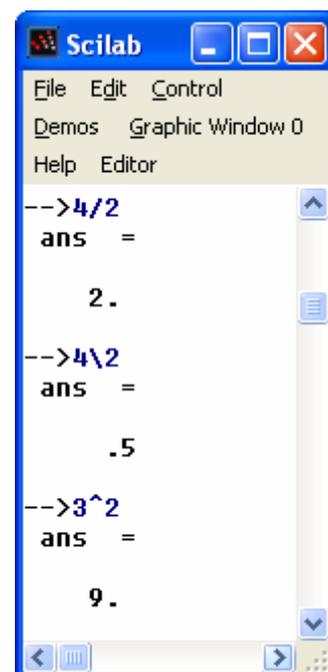
/ Divisão à direita

\ 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.
```

Formato de Saída

`%pi` ←

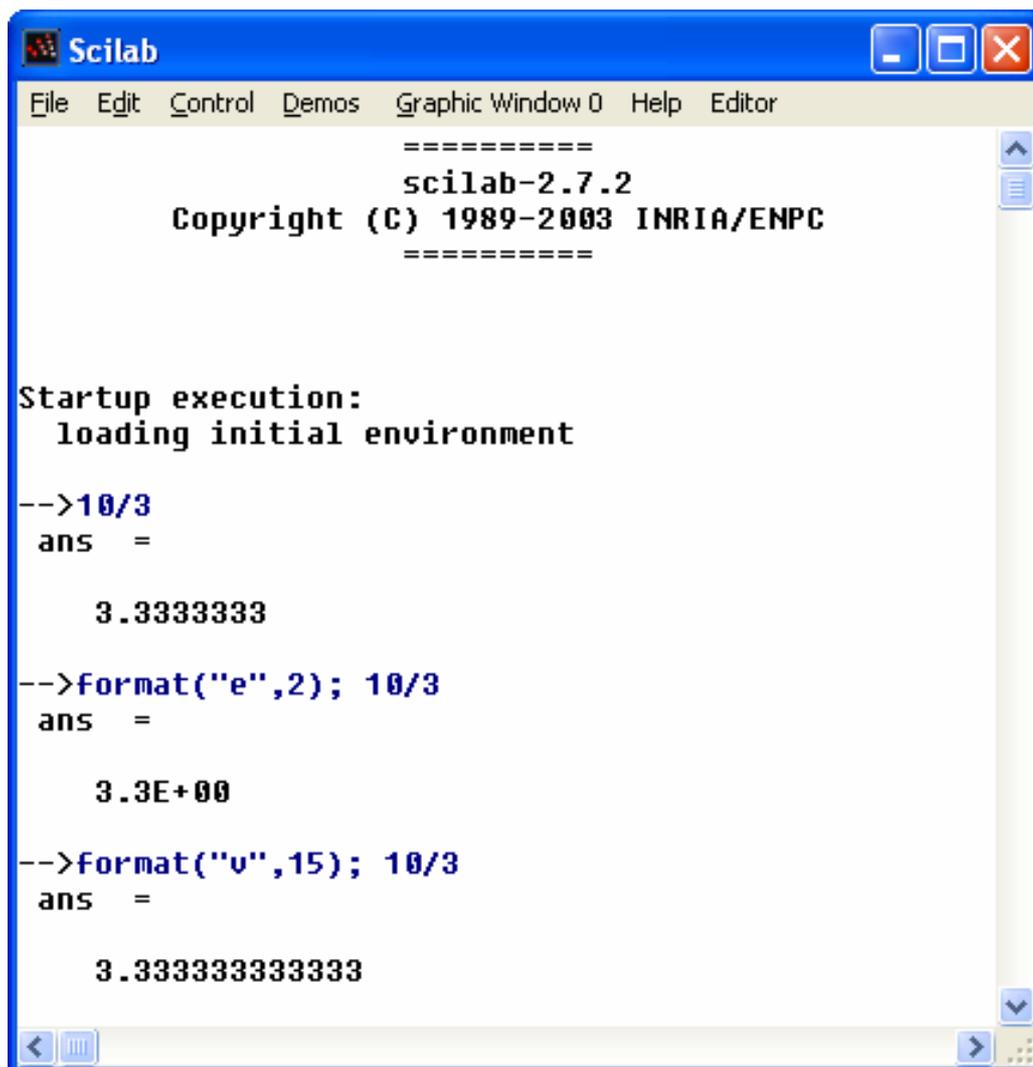
`%pi =`

`3.1415927`

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

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

long: nº 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 =

3.333333333333333
```

Outros Comandos

help: Informa sobre os comandos e funções do SCILAB.

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

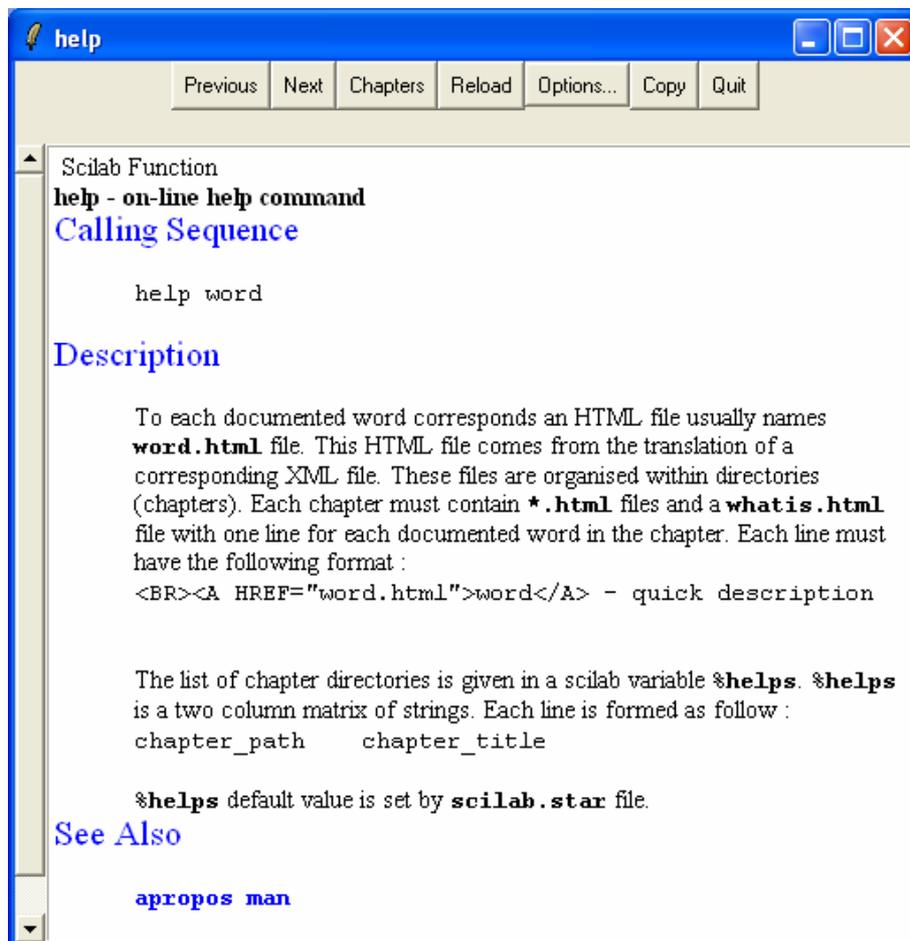
quit: Encerra o Scilab. (*quit* ou *exit*)

save: Grava variáveis em um arquivo

Ex.: `save('varsalva', a)`: grava *a* no arquivo '*varsalva*'

load: Carrega as variáveis do arquivo.

Ex. `load varsalva`



Operações com Matrizes

a) Transposta de uma matriz: $B = A'$

b) Adição e Subtração: $C = A + B$

c) Multiplicação: $C = A * B$, $C = \alpha * A$

d) Divisão:

d.1) Divisão à esquerda: $X = A \setminus B$ solução de $A * X = B$

d.2) Divisão à direita: $X = A / B$ solução de $X * A = B$

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

Operações Elemento a Elemento

a) Multiplicação: $z = x.*y$

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

$z = x.*y$ ←

$z =$

$[4 \ 10 \ 18]$

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

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

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

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

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

b) Divisão: $z = x \setminus y$ ou $z = y ./ x$

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

$z = x \setminus y$ ←

$z =$

!4.0000 2.5000 2.0000!

c) Potência: $z = x.^y$

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

$z = x.^y$ ←

$z =$

!1 32 729!

$z = x.^2$ ←

$z = !1 \ 4 \ 9!$

$z = 2.^[x \ y]$ ←

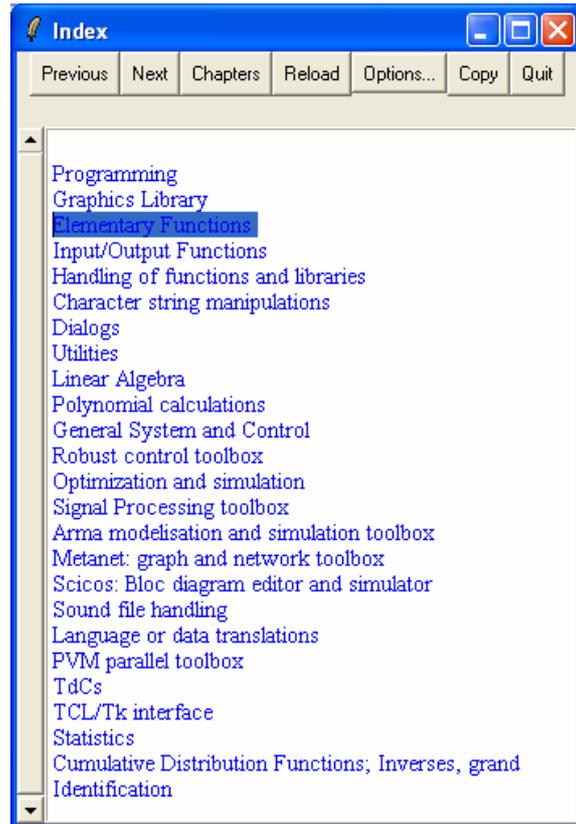
$z = !2 \ 4 \ 8 \ 16 \ 32 \ 64!$

Operadores Relacionais

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

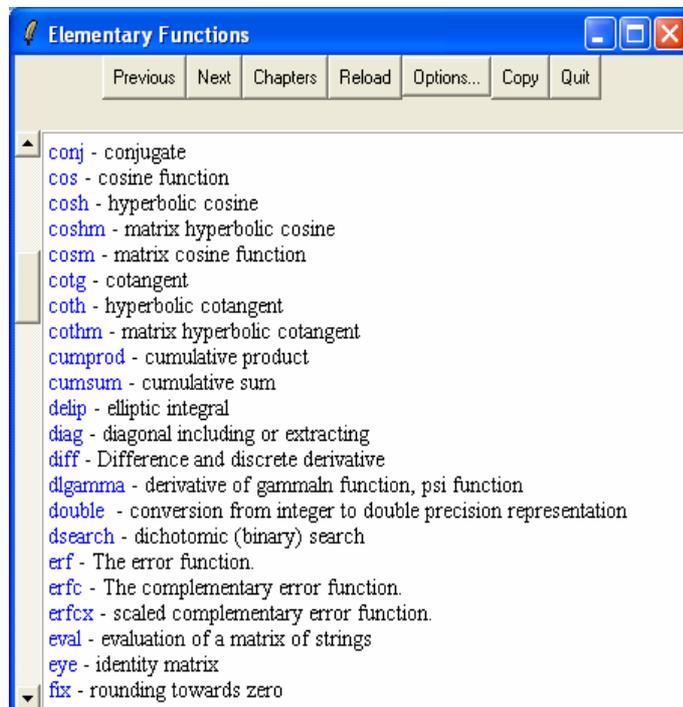
Operadores Lógicos

- & and
- | or
- ~ not



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.

Vetores e Subscritos

Geração automática: $x = x_i : dx : x_f$

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

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

Ex.: $x = 0:0.1:0.2 \rightarrow x = 0\ 0.1000\ 0.2000$

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 !  
!.1    .1557673 !  
!.2    .2397128 !
```

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!

$A = [A; [10\ 11\ 12]]$ ←

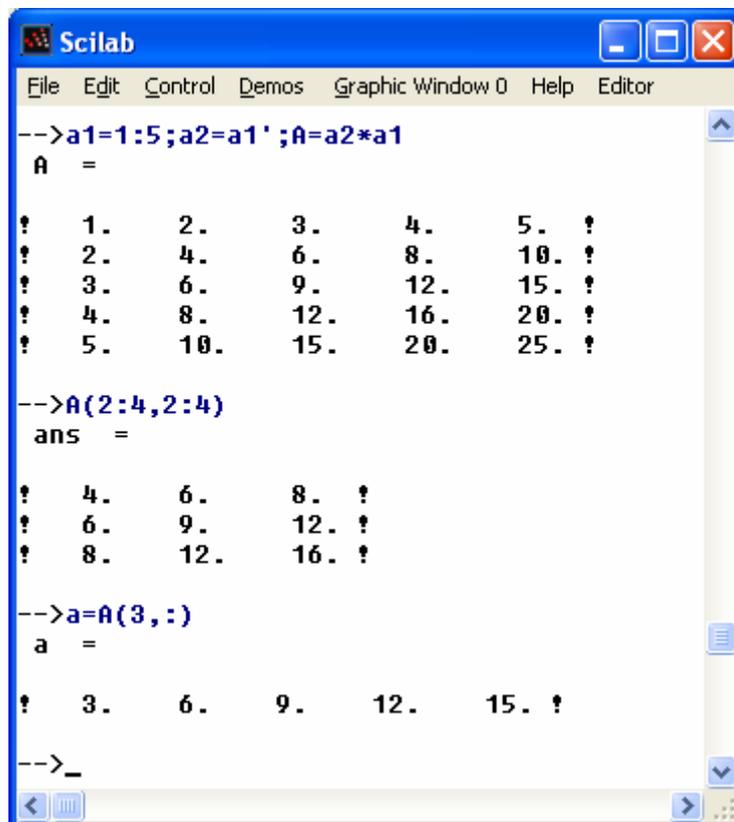
A =

!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. ?

-->_
```

Polinômios

$x = \text{poly}(0, "x")$ → Define x como variável.

$p = \text{poly}(v, "x", "flag")$ → 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:

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

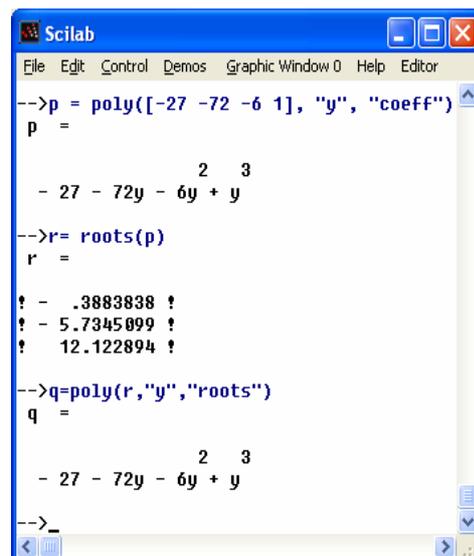
$r = \text{roots}(p)$

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

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

$q =$

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



```
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

-->_
```

Para calcular o valor do polinômio para um determinado valor de y , faz-se: $\text{horner}(p, \text{num})$, c/ num = valor desejado para y .

Para calcular os valores do polinômio para diversos valores de y , faz-se: $\text{horner}(p, \text{nums})$ c/ nums = vetor com os diversos valores desejados para y .

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 .

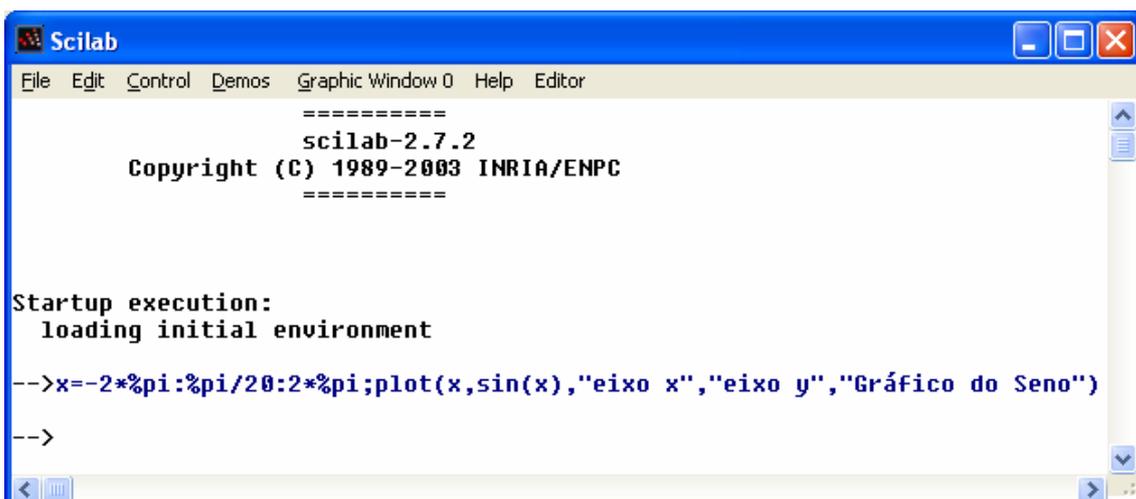
`plot(x,y)` → Plota o vetor y em função do vetor x

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

`x cap` → Título do eixo x

`y cap` → Título do eixo y

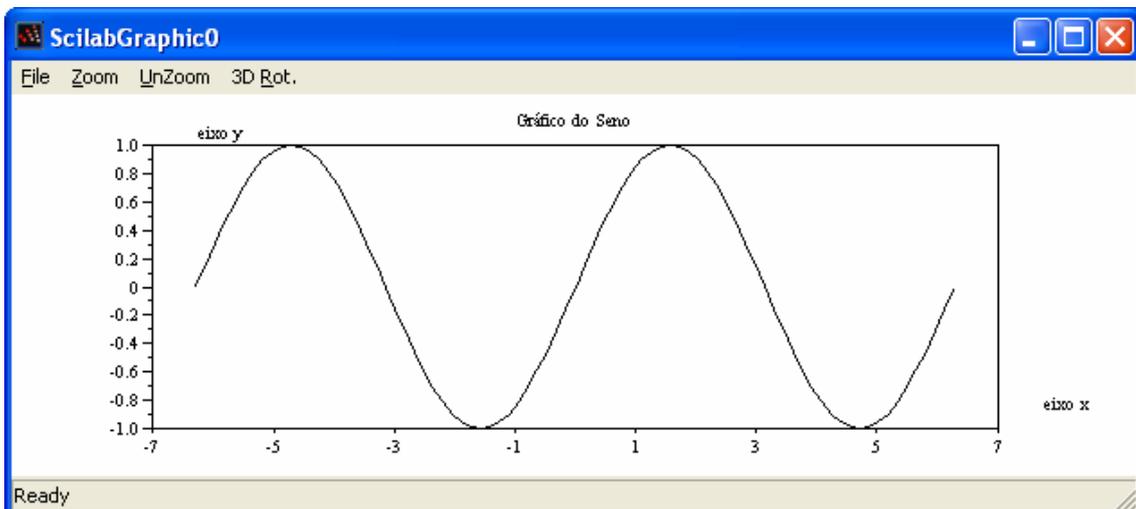
`caption` → Título do gráfico



```
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")
-->
```

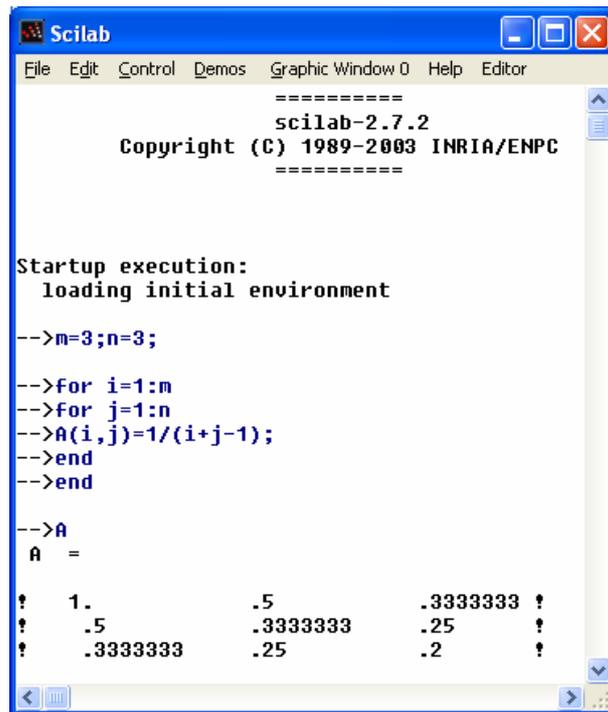


Comandos para Controle de Fluxo

a) 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
=====

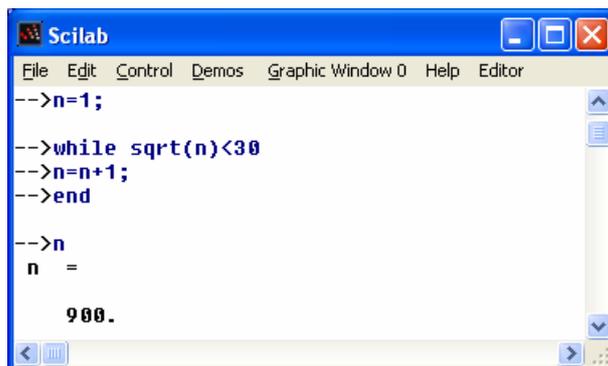
Startup execution:
loading initial environment

-->m=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       ?
```

b) comando *while*

Ex.:

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

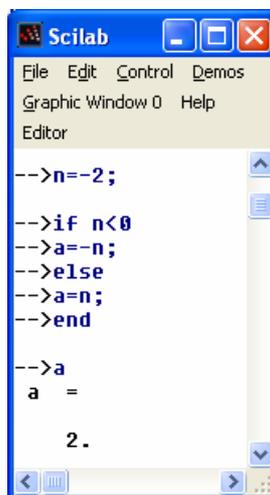


```
Scilab
File Edit Control Demos Graphic Window 0 Help Editor
-->n=1;
-->while sqrt(n)<30
-->n=n+1;
-->end
-->n
n =
900.
```

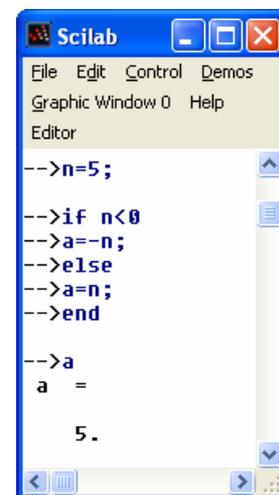
c) comando *if*

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.
```

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  $y = \text{seno}(a*t)$   
// 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  $a * t$ ")
```

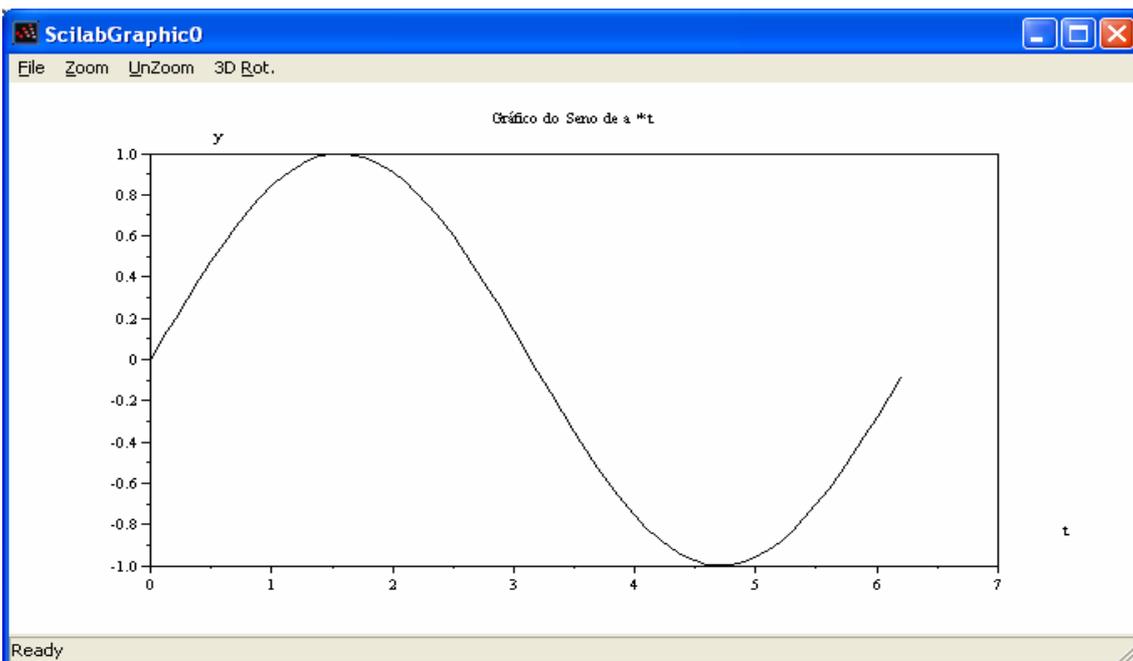
```
SciPad - C:/Science/Sciprog/plotseno.sci
File Edit Search Windows Options Load into Scilab Help
// Este programa traça o gráfico da função y = seno(a*t)
// A variável a deve estar definida no espaço de trabalho

t = [0:0.1:2*pi];
y = sin(a*t);
plot(t,y,"t","y","Gráfico do Seno de a * t")
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 .



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 a essa função:

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

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

```
ans =
```

```
50.
```

