Unidade 3:
Linguagem de programação

3.5. Trabalhando com polinômios, vetores e matrizes

O programa Scilab trabalha com polinômios, vetores e matrizes de forma similar a vários pacotes computacionais disponíveis no mercado.

3.5.1 Polinômios

No programa Scilab existe uma sintaxe para definição de polinômios. Experimente o comando ***help poly*** na linha de comando. Veja o que a janela de ajuda traz de informações sobre a função ***poly***.

Uma forma de criar um polinômio no Scilab é a partir de suas raízes. Vejamos o exemplo:

```
--> p = poly ([1 2], ‘x’, ‘roots’)
    p = 2 - 3x + x2
```

Com este comando definimos o polinômio cujas raízes são dadas pelos valores 1 e 2. Para confirmar, utilizemos a função ***roots***. Esta função é utilizada para calcular as raízes de um polinômio inserido no argumento.

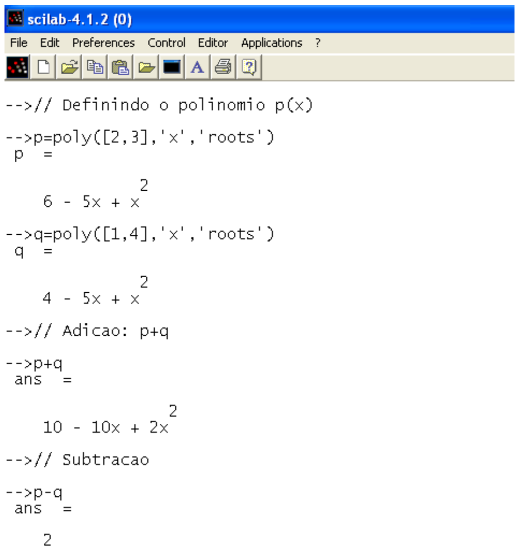
```
--> roots(p)
    ans  =
           1.
           2.
```

Outra maneira de definirmos o mesmo polinômio é através do seguinte comando:

```
--> x=poly(0,'x');
--> p=2-3*x+x^2
```

$$p = 2 - 3x + x^2$$

As operações de soma, subtração, multiplicação e divisão de polinômios são realizadas conforme ilustra a Figura 3.9.



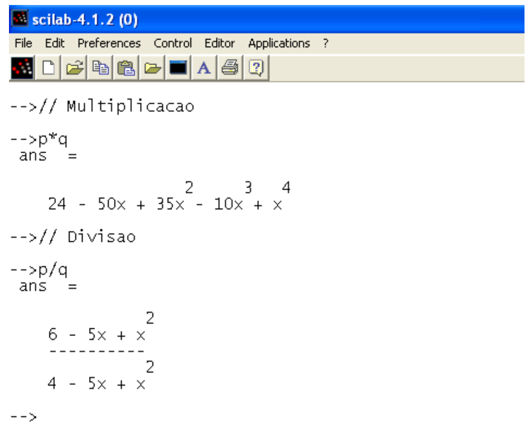


Figura 3.8: Operações de soma, subtração, multiplicação e divisão com os polinômios p(x) e q(x).

Para obter o quociente e o resto da divisão de p(x)/q(x) utilize a função ***pdiv(p,q)***.

```
--> [r,q] = pdiv(p,q)
    q  =   1
     r  =   2.
```

Se desejarmos saber o valor do polinômio p(x) em algum ponto (por exemplo, em x=3.0) utilize a função ***horner(p,3.0)***.

```
--> horner(p,3.0)
    ans  =   0.
```

3.5.2. Vetores e matrizes

O programa Scilab trabalha com vetores e matrizes de forma similar a vários pacotes computacionais disponíveis no mercado.

Os vetores são criados colocando seus componentes entre colchetes, [ ]. Contudo, podemos criar vetores linha ou vetores coluna. A diferença está no elemento separador utilizado para separar os elementos do vetor. Fica mais fácil entender com um exemplo. Suponhamos que desejamos criar um vetor coluna **x** (com três linhas e uma coluna) contendo os seguintes elementos reais:

$$x = \begin{bmatrix} 1.0 \\ 2.0 \\ 3.0 \end{bmatrix}_{3x1}$$

No Scilab utilizaremos o seguinte comando:

```
--> x=[1.0; 2.0; 3.0]
 x = 1.
     2.
     3.
```

Como observamos, o separador dos elementos na criação de um vetor coluna é o ponto e vírgula. Para criarmos um vetor linha **y** empregamos como separador o espaço (espaço em branco) ou a vírgula. Representaremos o seguinte vetor linha no Scilab:

$$y = \begin{bmatrix} 2.0 & 4.0 & 6.0 \end{bmatrix}_{1x3}$$

No Scilab utilizaremos o seguinte comando:

```
--> y=[2.0  4.0  6.0]
 y =   2.   4.   6.
```

Para transformarmos um vetor coluna em um vetor linha, ou vice-versa, podemos realizar a operação de transposição de vetores. Para transpor um vetor no Scilab é empregado o símbolo **‘** (apóstrofo). Vejamos o exemplo, transpondo o vetor y:

```
--> y = y’
 y = 2.
     4.
     6.
```

A dimensão de um vetor (linha ou coluna) pode ser determinada no Scilab empregando o comando **size**. Vejamos o exemplo:

```
--> y=[2.0  4.0  6.0];
```

```
--> size (y)
 ans
      3.   1.
```

A resposta deste comando nos diz que o vetor **y** é composto por três linhas e uma coluna. Verifique no Scilab.

Os vetores podem ser multiplicados ou divididos por grandezas escalares. Vetores de mesma dimensão também podem ser somados ou subtraídos. A operação produto escalar (ou produto interno) é realizada entre dois vetores de mesma dimensão através da expressão vetorial:

$$x = \begin{bmatrix} 1.0 \\ 2.0 \\ 3.0 \end{bmatrix}_{3x1} \qquad y = \begin{bmatrix} 2.0 \\ 4.0 \\ 6.0 \end{bmatrix}_{3x1}$$

$$z = x^T \cdot y$$

O resultado será um número escalar. No Scilab fazemos:

```
--> x=[2.0;  4.0;  6.0];
--> y=[2.0;  4.0;  6.0];
--> z=x'*y
 z = 28.
```

Se dois vetores possuem dimensões diferentes, por exemplo, x_{1x3} e y_{1x4}, o produto vetorial (ou produto externo) é realizado através da expressão vetorial:

$$x = \begin{bmatrix} 1.0 \\ 2.0 \\ 3.0 \end{bmatrix}_{3x1} \qquad y = \begin{bmatrix} 1.0 \\ 2.0 \\ 3.0 \\ 4.0 \end{bmatrix}_{4x1}$$

$$z = x \cdot y^T$$

```
--> x=[1.0;  2.0;  3.0];
--> y=[1.0;  2.0;  3.0;  4.0];
--> z=x*y^T
 z  =   1.   2.   3.   4.
        2.   4.   6.   8.
```

3. 6. 9. 12.

A composição de uma matriz no ambiente Scilab é realizada de forma similar a dos vetores, separando-se cada linha por um ponto e vírgula (os elementos da mesma linha podem ser separados por espaço ou por vírgula). Suponhamos que desejamos criar uma matriz **A** (com três linhas e três colunas) contendo os seguintes elementos reais (* no texto será empregado letras minúsculas para representar os vetores e maiúsculas para representação de matrizes, ambas em negrito):

$$A = \begin{bmatrix} 1.0 & 2.0 & 3.0 \\ 4.0 & 5.0 & 6.0 \\ 7.0 & 8.0 & 9.0 \end{bmatrix}_{3x3}$$

No Scilab utilizaremos o seguinte comando:

```
--> A=[1.0  2.0  3.0 ; 4.0  5.0  6.0;  7.0  8.0  9.0]
 A =   1.    2.    3.
       4.    5.    6.
       7.    8.    9.
```

Uma matriz é transposta pelo comando ' (apóstrofo). A Tabela 3.2 apresenta uma lista de operações que podem ser realizadas com matrizes.

Tabela 3.2: Operações com matrizes (estruturais* e matriciais).

Operação	Comando Scilab	Comentário
Soma estrutural	**A+B**	A soma (estrutural) de matrizes é idêntica à matricial. Obs: as duas matrizes precisam ter o mesmo número de linhas e colunas.
Subtração estrutural	**A-B**	Subtração de matrizes estrutural é idêntica à matricial. Obs: as duas matrizes precisam ter o mesmo número de linhas e colunas..
Multiplicação estrutural	**A.*B**	Multiplicação elemento a elemento de **A** e B. Obs: As duas matrizes precisam ter a mesmo número de linhas e colunas, ou uma delas ser um escalar.
Multiplicação matricial	**A*B**	Multiplicação das matrizes **A** e **B**. O

		número de colunas da matriz **A** precisa ser igual ao número de linhas da matriz **B**.
Divisão estrutural à direita	**A./B**	Divisão elemento a elemento de **A** e **B**: a(i,j)/b(i,j). Obs: As duas matrizes precisam ter a mesmo número de linhas e colunas, ou uma delas ser um escalar.
Divisão estrutural à esquerda	**A.\B**	Divisão elemento a elemento de **A** e **B**: b(i,j)/a(i,j) Obs: As duas matrizes precisam ter a mesmo número de linhas e colunas, ou uma delas ser um escalar.
Divisão matricial à direita	**A/B**	Divisão matricial definida por **A** * inv(**B**), onde inv(**B**) é a inversa da matriz **B**.
Divisão matricial à esquerda	**A\B**	Divisão matricial definida por inv(**A**)***B**, onde inv(**A**) é a inversa da matriz **A**.
Expoente estrutural	**A.^B**	Expoente elemento a elemento de **A** e **B**: a(i,j)^b(i,j). Obs: As duas matrizes precisam ter a mesmo número de linhas e colunas, ou uma delas ser um escalar.

* Operações estruturais: são operações entre matrizes executadas elemento a elemento; a operação é executada sobre os elementos correspondentes nas matrizes.

No Scilab é possível criar um vetor ou matriz nula, um vetor ou matriz unitária ou uma matriz identidade de forma simples empregando os seguintes comandos: **zeros**(nl, nc), **ones**(nl, nc) e **eye**(nl, nc), onde nl é o número de linhas e nc o número de colunas (se nl ou nc for igual a 1 teremos um vetor linha ou coluna). A Figura 3.9 ilustra esses comandos.

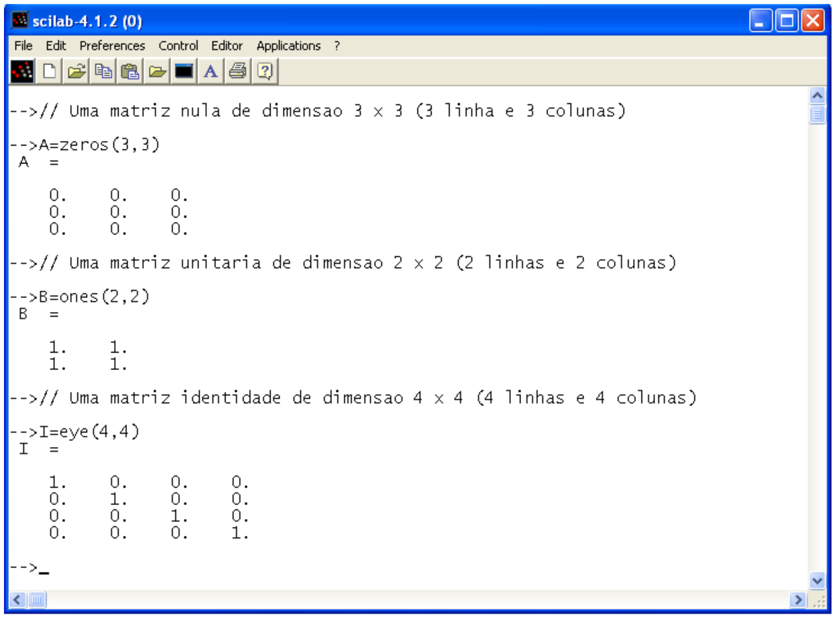


Figura 3.9: Criando matriz de zeros, uns e identidade no Scilab.

No programa Scilab o valor do determinante de uma matriz **A** (**A** deve ser uma matriz quadrada) é facilmente calculado empregando a função **det**( ), onde o argumento desta função é a matriz que se deseja obter o determinante, no caso det(**A**).

Também a inversa de uma matriz **A** (**A** deve ser uma matriz quadrada) é calculado empregando a função **inv**( ), onde o argumento desta função é a matriz que se deseja obter o determinante, no caso **inv**(**A**). Atenção para as propriedades do cálculo matricial. Vejamos o cálculo do determinante e da inversa de uma matriz $\mathbf{A}_{3x3}$, Figura 3.10.

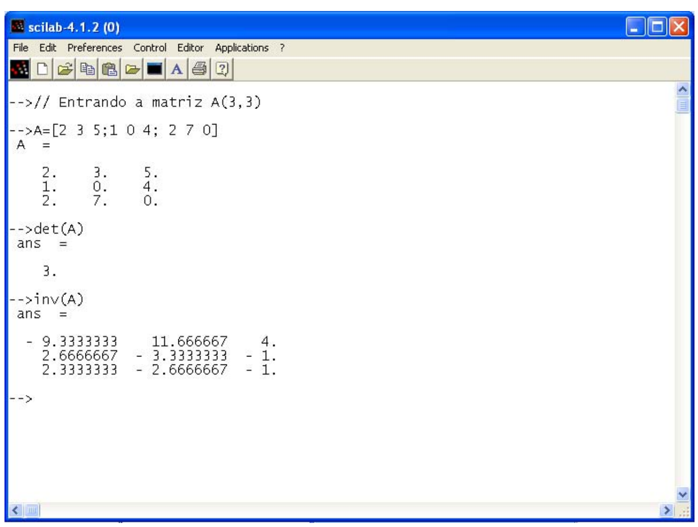


Figura 3.10: Avaliando o valor do determinante e a inversa de uma matriz $\mathbf{A}_{3x3}$.