



SciLab

- a short introduction -

What is Scilab?

- a free open source pseudo clone of MATLAB
- a comfortable system for numerical calculation
 - solutions for systems of equations
 - eigenvalues, eigenvectors
 - solution of differential equations
 - simple programming language

available at www.scilab.org

Scalars, vectors and matrices

- scalars: handling like a calculator
- without assignment to a variable:
 - Variable is „ans“
- no specification of the type of a variable is necessary.
- column vector
 - > $b = [x_1 ; x_2 ; x_3]$
- row vector
 - > $b = [x_1 \ x_2 \ x_3];$ //using a semicolon gives no output
- matrix
 - > $A = [x_{11} \ x_{12} \ x_{13}; \ x_{21},x_{22},x_{23}; \ \tan(x), \ x_{32},1+2*\%i]$

Special matrices

- identity matrix of dimension (4,4)
--> $I = \text{eye}(4,4)$
- diagonal matrix of vektor b
--> $B = \text{diag}(b)$
- matrices consisting of ones or zeros
--> $C = \text{ones}(n,m)$, $Z = \text{zeros}(C)$
- access the upper (lower) triangular matrix
--> $T = \text{triu}(C)$ ($T = \text{tril}(C)$)

Operations

- all known „normal“ operations
 - $[+, -, *, /]$ (mind the dimension depending on the operator!)
- element-wise operators: prepend a dot
 - $\rightarrow B^2 = B^{\wedge 2}$ squares the quadratic Matrix B
 - $\rightarrow B.^2 = B.^{\wedge 2}$ squares the entries of the Matrix B
- transposed matrices/vectors
 - $\rightarrow A^t = A'$, $a^t = a'$
- $\exp(A)$, \sqrt{A} , $\sin(A)$, etc
 - for element-wise operation of these functions:
 - $\expm(A)$, $\sqrt{m}(A)$, $\sinm(A)$

Referencing, extracting and merging vectors and matrices

- referencing a coefficient of matrix A
--> $A_{mn} = A(m,n)$
- extracting a row / column
--> $A(:,n)$ / $A(m,:)$
- extracting entry 1 and 2 from row 1 and 3
--> $A([1\ 3],[1\ 2])$
- merging matrices A, B, C, D with compatible dimensions
--> $M=[A\ B\ C\ D]$

Programming in Scilab

- „for“-loop
 - „for“ iterates over the components of an row vector
 - --> `v = [1 2 3 4]`
 - --> `y=0; for k=v, y=y+k, end`
 - Or: --> `y=0; for i=1:1:4, y=y+v(i), end // i= i_beg: i_incr: i_end`
- „while“-loop
 - Repeats a set of instructions while a condition is true
 - --> `x=1; while x<14 , x=2*x, end`
- „if then else“-construction
 - Boolean request of a condition
 - --> `if x>0 then, y=-x, else, y=x, end`
 - more conditions: `elseif`
 - --> `if x>0 then, y=-x, elseif x=0 then, y=1, else, y=x, end`

Lists

- „list“ – a simple listing of Scilab objects

- --> L=list(rand(2,2), [„just a“ „text sample“], 3)

- --> M=L(1) //extracts entry 1

- --> L(2)(3) = L(2)(2); L(2)(2)=„boring“ //inserts a part of an entry

- typed list – gives parameters and the list a name

- --> TL=tlist([*name of the list*, *name p1*, *name p2*], p1,p2)

- example:

- > c=20; o=35; s=10;

- > T=tlist([„Busses“, ‘overland_busses‘, ‘omnibusses‘,
‘schoolbusses‘],c,o,s)

- > T.omnibusses

Functions

- functions define operations for „formal“ parameters
 - function `y=name_of_function(x,z)`
 `y=operation(x,z)`
 `endfunction`
 - example:
 - > `function y=faculty(n)`
 `y=prod(1:n)`
 `endfunction`
 - > `m=faculty(6)`

An application of a tlist and a function: operator overloading

- 4-vector: $\hat{p}_1 = (E_1, \bar{p}_1)$, $\hat{p}_2 = (E_2, \bar{p}_2)$,
 $\Rightarrow \hat{p}_1 \cdot \hat{p}_2 = E_1 \cdot E_2 - \bar{p}_1 \cdot \bar{p}_2$
- try in the editor (press editor in the menu):

```
function [y]=vec4(e,p)
    y=tlist(['element'; 'E'; 'p'],e,p(1:3));
endfunction
```

```
function [z]=%element_m_element(a,b)
    z=a.e*b.e;
    for i=1:3
        z=z-a.p(i)*b.p(i);
    end
endfunction
```

```
a=vec4(25,[2,3,4]);
```

```
b=vec4(15,[2,4,7]);
```

```
a*b
```

- press execute in the editor

Plotting

```
--> plot2d(x,y) //x,y must be column vectors
```

- **example:**

```
--> x=linspace(-1,1,61)
```

```
--> y=x.^2
```

```
--> plot2d(x,y)
```

- **you can also plot several curves simultaneous**

```
--> y2=1-x^2
```

```
--> y3=2*y
```

```
--> xbas() //clears graph
```

```
--> plot2d([x x x],[y y2 y3]) //[y y2 y3] = Matrix!
```

```
--> xtitle("Title","x","y") //Titel and labels
```

Help

- Includet help
 - > help()
 - > apropos topic
- online help:
 - www.scilab.org
- tutorial: show your desktop and the shared folder
- of course: your tutor 😊

.....good luck