Vektorok létrehozása, műveletek vektorokkal:

Sorvektor létrehozása:

```
>> a=[1,4,6,-2]

a =
     1     4     6    -2
```

vessző helyett szóközzel is lehet szeparálni az egyes elemeket:

```
>> b=[1 4, -2 1]

b =
     1     4    -2     1
```

algebrai műveletek:

```
>> a+b

ans =
     2     8     4    -1

>> a-b

ans =
     0     0     8    -3
>> 2*a

ans =
     2     8    12    -4
```

transzponálás:

```
>> c=a'

c =
     1
     4
     6
    -2
```

oszlopvektor definiálása (pontosvessző választja el az elemeket):

```
>> d=[2;1;-4;0]

d =
     2
     1
    -4
     0

>> c-d

ans =
    -1
     3
    10
    -2
```

```
>> a

a =
     1     4     6    -2
```

számtani sorozat elemeit tartalmazó vektor létrehozása:

```
>> e=1:6

e =
     1     2     3     4     5     6

>> 1:3:20

ans =
     1     4     7    10    13    16    19


>> 6:-1:-2

ans =
     6     5     4     3     2     1     0    -1    -2


>> 0:0.1:0.5

ans =
         0    0.1000    0.2000    0.3000    0.4000    0.5000
```

vektor koordinátái elérése, módosítása:

```
>> a(2)

ans =
     4

>> a(2)=7

a =
     1     7     6    -2
```

a vektorok dimenzióját nem kell előre deklarálni, sőt automatikusan bővíthetők:

```
>> a(7)=25

a =
     1     7     6    -2     0     0    25
```

vektor bővítése:

```
>> f=[a,0,0,e]

f =
     1     7     6    -2     0     0    25     1     2     3
4     5     6
```

részvektor képzés:

```
>> f(2:6)

ans =
     7     6    -2     0     0

>> a

a =
     1     7     6    -2     0     0    25

>> ind =[2 1 4 3]

ind =
     2     1     4     3
```

az *a* vektor elemeiből kiszedi az *ind* vektorban megadott indexű tagokat:

```
>> a(ind)

ans =
     7     1    -2     6

>> e

e =
     1     2     3     4     5     6

>> length(e)

ans =
     6

>> e(length(e):-1:1)

ans =
     6     5     4     3     2     1
```

Mátrixok definiálása, műveletek:

```
>> A=[1,2,4;-2,0,1]

A =
     1     2     4
    -2     0     1


>> B=[-2 4 3; 0 -1 1]

B =
    -2     4     3
     0    -1     1


>> A-B, A+B, -A, 3*B

ans =
     3    -2     1
    -2     1     0
ans =
    -1     6     7
    -2    -1     2
ans =
    -1    -2    -4
     2     0    -1
ans =
    -6    12     9
     0    -3     3


>> A'

ans =
     1    -2
     2     0
     4     1
```

mátrix sor- ill. oszlopdimenziója

```
>> size(B)

ans =
     2     3

>> A(2,3)

ans =
     1
```

```
>> A(1,2)=5

A =
     1     5     4
    -2     0     1
```

ha a mátrix 3. sorában adunk egy elemnek értéket, automatikusan bővíti a mátrixot a hiányzó sorral:

```
>> A(3,2)=2

A =
     1     2     4
    -2     0     1
     0     2     0
```

négyzetes mátrix determinánsának számítása:

```
>> det(A)

ans =
   -18
```

négyzetes mátrix inverzének számítása:

```
>> inv(A)

ans =
    0.1111   -0.4444   -0.2778
         0         0    0.5000
    0.2222    0.1111   -0.5556
```

mátrix szorzás akkor végezhető el, ha az első mátrix oszlopdimenziója azonos a második mátrix sordimenziójával:

```
>> A*B

??? Error using ==> *
Inner matrix dimensions must agree.

>> A*B'

ans =
    30    -1
     7     1
     8    -2
```

kisebb mátrixokból nagyobbakat tudunk összerakni: vesszővel elválasztva a mátrixokat, egymás mellé kerülnek (ha a sordimenziójuk azonos)

```
>> [A,B']

ans =
     1     5     4    -2     0
    -2     0     1     4    -1
     0     2     0     3     1
```

pontosvessző új sor kezdetet jelent:

```
>> [A;[2 3 4]]

ans =
     1     5     4
    -2     0     1
     0     2     0
     2     3     4


>> [A;A;A]

ans =
     1     5     4
    -2     0     1
     0     2     0
     1     5     4
    -2     0     1
     0     2     0
     1     5     4
    -2     0     1
     0     2     0
```

részmátrix képzése

```
>> ans(4:8,2:2)

ans =
     5
     0
     2
     5
     0

>> A

A =
     1     2     4
    -2     0     1
     0     2     0
```

mátrix 1. sorának kinyerése

```
>> A(1,:)

ans =
     1     2     4
```

a 2. oszlopvektor:

```
>> A(:,2)

ans =
     2
     0
     2
```

vektorként hivatkozva a mátrix elemeire oszlopfolytonosan is érhetjük el az egyes elemeket

```
>> A(1), A(2), A(3), A(4)

ans =
     1
ans =
    -2
ans =
     0
ans =
     2
```

Speciális vektor- ill. mátrixfüggvények, -műveletek

```
>> a=[1,4,6,-2];
>> min(a)

ans =
    -2


>> max(a)

ans =
     6


>> a

a =
     1     4     6    -2


>> sort(a)

ans =
    -2     1     4     6
```

Az alábbi szintaxissal két outputot ad vissza a *sort* parancs:

```
[c,ind]=sort(a)

c =
    -2     1     4     6
ind =
     4     1     2     3
```

Csökkenő sorrendet az alábbi képlettel kaphatunk:

```
>> -sort(-a)

ans =
     6     4     1    -2
```

```
>> a

a =
     1     4     6    -2


>> b

b =
     1     4    -2     1
```

Koordinátánkénti szorzás vektorra ill. mátrixra. Itt két azonos dimenziójú vektort szorzunk össze, az eredmény egy ugyanolyan dimenziójú vektor, amely koordinátái a megfelelő koordináták szorzatait tartalmazza.

```
>> a .* b

ans =
     1    16   -12    -2
```

koordinátánkénti osztás

```
>> a ./ b

ans =
     1     1    -3    -2
```

koordinátánkénti hatványozás

```
>> a .^ 2

ans =
     1    16    36     4

>> A

A =
     1     2     4
    -2     0     1
     0     2     0
```

mátrix hatványozás

```
>> A^2

ans =
    -9    13     9
    -2    -8    -8
    -4     0     2
```

koordinátánkénti hatványozás

```
>> A.^2

ans =
     1    25    16
     4     0     1
     0     4     0
```

Speciális Matlab szintaxis: mátrix + skaláris szám kifekejést koordinátánként értelmezi:

```
>> A+1

ans =
     2     3     5
    -1     1     2
     1     3     1
```

Elemi függvényeket meghívhatjuk vektor ill. mátrix argumentummal, ekkor koordinátánként alkalmazza az adott függvényt, és az output egy ugyanolyan dimenziós vektor ill mátrix lesz:

```
>> a

a =
     1     4     6    -2


>> sin(a)

ans =
    0.8415   -0.7568   -0.2794   -0.9093


>> exp(a)

ans =
    2.7183   54.5982  403.4288    0.1353
```

speciális mátrix létrehozó függvények:

```
>> eye(4)

ans =
     1     0     0     0
     0     1     0     0
     0     0     1     0
     0     0     0     1

>> zeros(2,5)

ans =
     0     0     0     0     0
     0     0     0     0     0
```

```
>> ones(1,3)

ans =
     1     1     1

>> rand(2,3)

ans =
    0.9501    0.6068    0.8913
    0.2311    0.4860    0.7621

>> diag([1 4 7])

ans =
     1     0     0
     0     4     0
     0     0     7

>> diag([1 4 7],1)

ans =
     0     1     0     0
     0     0     4     0
     0     0     0     7
     0     0     0     0

>> diag([1 4 7],-1)

ans =
     0     0     0     0
     1     0     0     0
     0     4     0     0
     0     0     7     0

>> A

A =
     1     2     4
    -2     0     1
     0     2     0
```

Ha a `diag` parancs inputjaként egy mátrixot adunk meg, az output a mátrix fődiagonálisa lesz:

```
>> diag(A)

ans =

     1
     0
     0
```

mátrix sorának ill. oszlopának törlése:

```
>> A(:,2)=[]

A =
     1     4
    -2     1
     0     0
```

Az *Ax=b* **lineáris egyenletrendszer megoldása**:

```
>> A=[1,3;2,5]

A =
     1     3
     2     5

>> b=[-2;-3]

b =
    -2
    -3
```

Az egyenletrendszer megoldása beépített művelettel (ez a javasolt módszer, a mögötte levő numerikus módszer optimális):

```
>> x=A\b

x =
     1
    -1
```

ellenőrzés:

```
>> A*x

ans =
    -2
    -3
```

megoldás a megoldóképletet használva:

```
>> inv(A)*b

ans =
     1
    -1
```

Beépített konstansok

```
>> pi

ans =
    3.1416
```

Az ú.n. gépi epszilon értéke:

```
>> eps

ans =
  2.2204e-016
```

legkisebb tárolható pozitív valós szám

```
>> realmin

ans =
  2.2251e-308
```

legnagyobb tárolható pozitív valós szám

```
>> realmax

ans =
  1.7977e+308
```

A végtelen beépített konstans:

```
>> Inf

ans =
   Inf

>> 2*Inf, 4-Inf, Inf-Inf

ans =
   Inf
ans =
  -Inf
ans =
   NaN
```

Komplex számok kezelése. Az i és a j változók kezdetben a képzetes egységet tárolják, de bármelyiket átdefiniálhatjuk:

```
>>  i^2

ans =

    -1

>> i

ans =
        0 + 1.0000i
```

```
>> j

ans =
         0 + 1.0000i


>> z=sqrt(-5)

z =
         0 + 2.2361i

>> z+3-5*i

ans =
   3.0000 - 2.7639i
```

Az elemi függvények komplex függvényként vannak definiálva. Például:

```
>> log(-6)

ans =
   1.7918 + 3.1416i
```

elemi függvények a szokásos szintaxissal használhatók (pl.):

```
>> sqrt(4), abs(-3), log(8),exp(2), tan(1), fix(2.4)

ans =
     2
ans =
     3
ans =
    2.0794
ans =
    7.3891
ans =
    1.5574
ans =
     2
```

valós számok tudományos írásmódja:

```
>> 2e-3

ans =
    0.0020
```