

OSNOVE MATLABA OZ. OCTAVE, 2. DEL

Navedene strani so iz knjige *Octave z uvodom v numerične metode*.

Matrike: (str. 30-32)

Matriko zapišemo v oglatih oklepajih []. Elemente v vrstici ločimo s presledkom ali vejico. Vrstice ločimo s podpičji. Do elementov matrike dostopamo z okroglimi oklepaji ().

Primer: generiramo matriko $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$

```
>> A=[1 2; 3 4]
```

```
A =
```

```
    1    2
    3    4
```

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

```
ans = 3
```

Posebne matrike lahko generiramo tudi s posebnimi ukazi: `ones`, `zeros`, `eye`, `rand`, `diag` (str. 32-33, 55-56).

Primer: generirajmo tridiagonalno matriko A dimenzije 5×5 , ki ima na diagonalni vektor $(2, 2, 2, 2, 2)$ na prvi naddiagonalni vektor $(1, 1, 1, 1)$ in na prvi poddiagonalni vektor $(-1, -1, -1, -1)$

```
>> d=[2 2 2 2 2];
```

```
>> u=[1 1 1 1];
```

```
>> l=[-1 -1 -1 -1];
```

```
>> A=diag(d)+diag(u,1)+diag(l,-1)
```

```
A =
```

```
    2    1    0    0    0
   -1    2    1    0    0
    0   -1    2    1    0
    0    0   -1    2    1
    0    0    0   -1    2
```

Računske operacije z matrikami: (str. 41)

`+`, `-`, `*`, `/`, `\`, `^`, `'`

Nekatere matrične funkcije: (str. 30-31, 59-60, 64-65)

`size` - dimenzije matrike, `det` - determinanta, `inv` - inverz, `rank` - rang, `norm` - norma

Matrične norme:

- prva - maksimalna stolpična vsota: $\|A\|_1 = \max_k \sum_{j=1}^n |a_{jk}|$
- neskončna - maksimalna vrstična vsota: $\|A\|_\infty = \max_j \sum_{k=1}^n |a_{jk}|$
- druga ali evklidska

Ukaz `norm` kličemo z dvema argumentoma - dano matriko in oznako $(1, 2, \text{inf})$ norme. Privzeta vrednost drugega argumenta je 2.

Primer: izračunajmo dimenzije, determinanto, inverz, rang in evklidsko normo matrike

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

```
>> A=[1 2; 3 4];
>> size(A)
ans = 2     2
>> det(A)
ans = -2
>> inv(A)
ans =
    -2.0000    1.0000
     1.5000   -0.5000
>> rank(A)
ans = 2
>> norm(A)
ans = 5.4650
```

Sistem linearnih enačb: (str. 134, 165, 169)

Sistem linearnih enačb $Ax = b$ rešimo z levim deljenjem $x = A \setminus b$.

Vektor b mora biti podan kot stolpec.

Primer: rešimo sistem $Ax = b$, kjer je $A = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 1 & 2 \\ 1 & 2 & 1 \end{bmatrix}$ in $b = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$

```
>> A=[1 2 1; 2 1 2; 1 1 2];
>> b=[4; 5; 4];
>> x=A\b
x =
     1
     1
     1
```

Programiranje:

Osnovni elementi programiranja so zanke in pogojni stavki.

Zanka for (str. 90, 93-94):

```
for pogoj
    stavki
end
```

Primer: v vektor zapišemo komponente vektorja

```
for i=1:5
    x(i)=i^2;
end
```

Zanka while (str. 90, 92, 94):

```
while pogoj
    stavki
end
```

Primer: razpolavljamo a , dokler ni manjši ali enak 10

```
a=100;
while a>10
    a=a/2;
end
```

Pogojni stavek if (str. 90):

```
if izraz
    stavki
else if
    stavki
else
    stavki
end
```

Primer: če je $n > 5$, dobi a vrednost 1, sicer pa 0

```
if n>5
    a=1;
else
    a=0;
end
```

m-datoteke: (str. 98-102)

Poznamo dva tipa m-datotek: skripte in funkcije.

Skripte so zgolj daljše kode, ki jih ne želimo pisati v ukazno vrstico.

Funkcije pa se vedno začnejo z ukazom `function`. Funkcijam podamo vhodne in izhodne argumente. Tako napisano funkcijo kličemo na enak način kot vgrajene funkcije.

Ime m-datoteke naj bo enako imenu funkcije.

Primer: napišimo funkcijo, ki izračuna obseg in ploščino kroga z danim polmerom r

```
function [o,p]=krog(r)
o=2*pi*r;
p=pi*r^2;
```

Uporaba funkcije:

```
>> [o,p]=krog(3)
o = 18.8496
p = 28.2743
```