

MATLAB 2024

2. gyakorlat

vektorok, mátrixok



Vektorok, alapműveletek

X sor- és oszlopvektor létrehozása:

X `sorv = [2 1 4 3];`

X `oszlopv = [6; 5; 7; 9];`

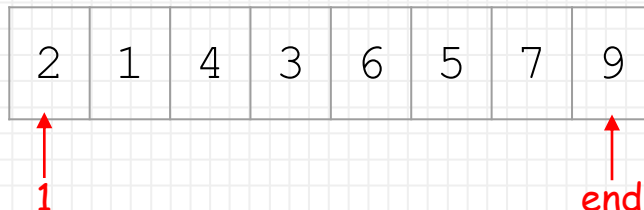
X transzponált: `sorv = sorv';`

X összefűzés: `ofv = [sorv, oszlopv'];`

X indexelés:

X **1-től kezdődik!**

X speciális index: `end` -- ez mindig az utolsó elem indexét jelenti, pl. `ofv(end)` ;



Vektorok, alpműveletek

X indexelés:

2	1	4	3	6	5	7	9
---	---	---	---	---	---	---	---

X adott elem lekérése: `elem = sorv(3);`

X adott elem felülírása: `sorv(3) = 44;`

X `length` - hossz lekérdezése

X `size` - méret lekérdezése, annyi elemű vektorral tér vissza, ahány dimenziós a mátrix, de legalább 2 - mivel a vektorra is mátrixként tekint

Vektorok, alpműveletek

X `ones`, `zeros` - csupa egyes vagy csupa nullás számjegyből álló tömb létrehozása

X ha **1 paraméterrel** használjuk, akkor négyzetes mátrixszal tér vissza: `ones(2)`

X ha **2 vagy több paraméterrel**, akkor rendre az egyes dimenziók mérete az egyes paraméterek szerint:

`zeros(1, 3)` →

1	1
1	1

0	0	0
---	---	---

X `rand` - véletlen számokból álló tömb generálása

X `eye` - egységmátrix generálása

X `diag`:

X adott diagonálissal rendelkező mátrix előállítása,

X egy mátrix diagonálisának kinyerése

Vektorok, alpműveletek

- X linspace, logspace - lineáris vagy logaritmikus beosztású vektor generálása

`linspace(2, 3, 4)` →

2	2.33	2.67	3
---	------	------	---

mettől

meddig

hány elemmel

- X beépített függvények vektorokon - `sin`, `abs`, `exp`, `round`, stb.

- X ha nem az elemek száma a fontos, hanem a felosztás pontos meghatározása: akkor kettőspont operátorral érdemes inkább: `di0 = 1:0.25:2;`

mettől

milyen
lépésközzel

meddig

1	1.25	1.5	1.75	2
---	------	-----	------	---

Vektorok, alpműveletek

X `sum, prod, mean`

```
min([2 7 1 4 6])
```

```
→ ans = 1
```

X `min, max:`

X ha **1** visszatérési értéket várunk, vagy nem is mentünk semmit → **szélsőérték**

X ha **2** visszatérési értéket várunk → **szélsőérték és annak indexe**

```
[a, b] = min([2 7 1 4 6])
```

```
→ a = 1
```

```
→ b = 3
```

X **elemenkénti műveletvégzés** - `e.*f`, `g.^2`, stb.

Vektorok, alpműveletek

1. megjegyzés: logikai kifejezések összefűzése

X `kif1 && kif2`

X skalár logikai kifejezések (`kif1` és `kif2` skalár kell, hogy legyen),

X “rövidrezáró” működés: ha `kif1` hamis, akkor nem is folytatódik a kiértékelés;

X `kif1 & kif2`

X vektorokra, **elemenként**.

A vagy-operátor (`||` illetve `|`) használata teljesen analóg módon történik.

2. megjegyzés: “kacsacsőr” elől: `<=` és nem `=<`

Vektorok, alpműveletek

- X `find` - paraméterben megadott logikai feltétel alapján keresés, azokkal a skalár indexekkel tér vissza, amely indexű elemek a feltételnek eleget tesznek
- X logikai műveletek és indexelés:
 - X logikai feltétel alapján (pl. $a < 1.2$, $b \sim 3$, stb)
 - X az **eredeti vektor hosszával megegyező logikai vektort** kapunk vissza, ami
 - X $\{0, 1\}$ számok valamilyen sorozata, ahol
 - 1 jelentése: igaz az adott elemre a feltétel
 - 0 jelentése: nem igaz az adott elemre a feltétel

Vektorok, alpműveletek

X logikai műveletek és indexelés:

X logikai feltétel alapján (pl. $a < 1.2$, $b \sim 3$, stb)

X az **eredeti vektor hosszával megegyező logikai vektort** kapunk vissza, ami

X $\{0, 1\}$ számok valamilyen sorozata, ahol

- 1 jelentése: igaz az adott elemre a feltétel
- 0 jelentése: nem igaz az adott elemre a feltétel

```
alma = [23 14 31 17 42];  
indexek1 = find(alma>24) % → [3, 5]  
alma(indexek1)           % → [31, 42]
```

23>24 14>24 31>24 17>24 42>24

```
indexek2 = alma>24 % → [0, 0, 1, 0, 1]  
alma(indexek2)    % → [31, 42]
```

Mátrixok, alapműveletek

X mátrix létrehozás, transzponálás, összefűzés, indexelés

```
alma = [11 22 33; 44 55 66];  
tmp1 = alma(1, 2); % → 22  
tmp2 = alma(2);    % → 44
```

1	11	22	33
2	44	55	66
3			

tartomány kijelölés:
kettőspont operátorral
alma(2, 2:3)

11	22	33
44	55	66

sorrend:
1. sor
2. oszlop
3. mélység
4. ...

teljes tartomány - pl összes oszlop 2. sorban:
alma(2, :)

```
korte = [77; 88];  
[alma, korte]  
[alma, korte; korte', korte']
```

77	11	22	33	77
88	44	55	66	88

```
idx = [1 3 4];  
alma(idx)  
  
idx2 = logical([1 0 1; 1 0 0]);  
alma(idx2)
```

11	22	55
----	----	----

11	22	33	77
44	55	66	88
77	88	77	88

11
44
33


1	0	1
1	0	0

Mátrixok, alapműveletek

X `size, ones, zeros, rand, diag, eye`

X `squeeze` - az 1 kiterjedésű (*singleton*) dimenziók eltávolítása

```
dio = ones(2, 1, 3); % 2 sor,  
1 oszlop, 3 mélység  
squeeze(dio)
```



1	1	1
1	1	1

X `reshape` - átméretezés

`B = reshape(A, m, n)`


B-nek legyen m sora, n oszlopa; A-ból az elemek oszlopirányban/ oszlopfolytonosan választódnak ki

X `numel` - a mátrix összes elemének száma

X `sum, prod, mean, min, max` - közös bennük: dimenzióként működnek (az első: oszlopirányban)

X logikai műveletek, logikai indexelés, elemenkénti műveletvégzés

```
alma = [1 2 3; 4 5 6]; % soronkénti  
max(alma, [], 2) % maxkereses
```



```
alma = [1 2 3; 4 5 6];  
sum(alma)
```

5	7	9
---	---	---

1	2	3
4	5	6

Feladatok

X [P-ITMAT-0014] Bevezetés a Matlab programozásba

X <https://moodle.ppke.hu/course/view.php?id=1514>

X 02 Feladatok → Vektorok és indexelés
→ Mátrix