


# Matematikai programok


## Mátrixalapú nyelvek – MatLab


Wettl Ferenc diái alapján


Budapesti Műszaki Egyetem  
Algebra Tanszék


2017.11.07


-  **MATLAB®** (**m**atrix **l**aboratory – The Language of Technical Computing)

-  **MATLAB®** (**m**atrix **l**aboratory – The Language of Technical Computing)
  - kereskedelmi program



-  **MATLAB®** (**m**atrix **l**aboratory – The Language of Technical Computing)
  - kereskedelmi program
  - többmillió felhasználó

-  **MATLAB®** (**ma**trix **la**boratory – The Language of Technical Computing)
  - kereskedelmi program
  - többmillió felhasználó
  - numerikus matematika, jel- és képfeldolgozás, kommunikáció, irányítástechnika, pénzügyi matematika,...




-  **MATLAB®** (**ma**trix **la**boratory – The Language of Technical Computing)
  - kereskedelmi program
  - többmillió felhasználó
  - numerikus matematika, jel- és képfeldolgozás, kommunikáció, irányítástechnika, pénzügyi matematika,...
  - <http://www.mathworks.com>

-  **MATLAB®** (**ma**trix **la**boratory – The Language of Technical Computing)
  - kereskedelmi program
  - többmillió felhasználó
  - numerikus matematika, jel- és képfeldolgozás, kommunikáció, irányítástechnika, pénzügyi matematika,...
  - <http://www.mathworks.com>
- szabad klónok:





# Mátrixalapú nyelvek

-  **MATLAB®** (**ma**trix **la**boratory – The Language of Technical Computing)
  - kereskedelmi program
  - többmillió felhasználó
  - numerikus matematika, jel- és képfeldolgozás, kommunikáció, irányítástechnika, pénzügyi matematika,...
  - <http://www.mathworks.com>
- szabad klónok:
  -  **octave**: GNU GPL, honlap [www.gnu.org/software/octave/](http://www.gnu.org/software/octave/)












-  **MATLAB®** (**ma**trix **la**boratory – The Language of Technical Computing)
  - kereskedelmi program
  - többmillió felhasználó
  - numerikus matematika, jel- és képfeldolgozás, kommunikáció, irányítástechnika, pénzügyi matematika,...
  - <http://www.mathworks.com>
- szabad klónok:
  -  **octave**: GNU GPL, honlap [www.gnu.org/software/octave/](http://www.gnu.org/software/octave/)
  -  **Scilab**: CeCILL license (GPL compatible), honlap [www.scilab.org](http://www.scilab.org)  
(fő fejlesztő: INRIA, Franciaország)







# Mátrixalapú nyelvek

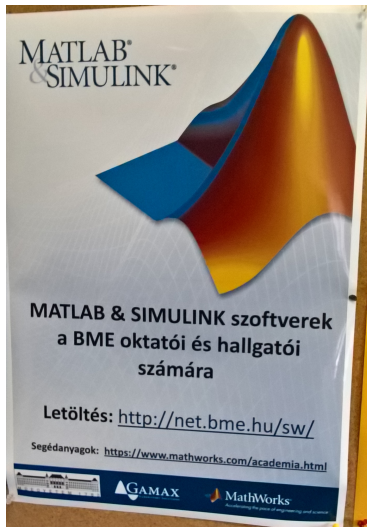
-  **MATLAB®** (**ma**trix **la**boratory – The Language of Technical Computing)
  - kereskedelmi program
  - többmillió felhasználó
  - numerikus matematika, jel- és képfeldolgozás, kommunikáció, irányítástechnika, pénzügyi matematika,...
  - <http://www.mathworks.com>
- szabad klónok:
  -  **octave**: GNU GPL, honlap [www.gnu.org/software/octave/](http://www.gnu.org/software/octave/)
  -  **Scilab**: CeCILL license (GPL compatible), honlap [www.scilab.org](http://www.scilab.org)  
(fő fejlesztő: INRIA, Franciaország)
  -  **FreeMat**: GPL, honlap [freemat.sourceforge.net/](http://freemat.sourceforge.net/)

# Mátrixalapú nyelvek

-  **MATLAB®** (**ma**trix **lab**oratory – The Language of Technical Computing)
  - kereskedelmi program
  - többmillió felhasználó
  - numerikus matematika, jel- és képfeldolgozás, kommunikáció, irányítástechnika, pénzügyi matematika,...
  - <http://www.mathworks.com>
- szabad klónok:
  -  **octave**: GNU GPL, honlap [www.gnu.org/software/octave/](http://www.gnu.org/software/octave/)
  -  **Scilab**: CeCILL license (GPL compatible), honlap [www.scilab.org](http://www.scilab.org)  
(fő fejlesztő: INRIA, Franciaország)
  -  **FreeMat**: GPL, honlap [freemat.sourceforge.net/](http://freemat.sourceforge.net/)
- Hasonló szoftverek:

-  **MATLAB®** (**ma**trix **lab**oratory – The Language of Technical Computing)
  - kereskedelmi program
  - többmillió felhasználó
  - numerikus matematika, jel- és képfeldolgozás, kommunikáció, irányítástechnika, pénzügyi matematika, ...
  - <http://www.mathworks.com>
- szabad klónok:
  -  **octave**: GNU GPL, honlap [www.gnu.org/software/octave/](http://www.gnu.org/software/octave/)
  -  **Scilab**: CeCILL license (GPL compatible), honlap [www.scilab.org](http://www.scilab.org)  
(fő fejlesztő: INRIA, Franciaország)
  -  **FreeMat**: GPL, honlap [freemat.sourceforge.net/](http://freemat.sourceforge.net/)
- Hasonló szoftverek:
  -  **julia**: MIT licensed, (high-performance dynamic programming language for technical computing), [honlap julialang.org/](http://julialang.org/)

-  **MATLAB®** (**ma**trix **la**boratory – The Language of Technical Computing)
  - kereskedelmi program
  - többmillió felhasználó
  - numerikus matematika, jel- és képfeldolgozás, kommunikáció, irányítástechnika, pénzügyi matematika,...
  - <http://www.mathworks.com>
- szabad klónok:
  -  **octave**: GNU GPL, honlap [www.gnu.org/software/octave/](http://www.gnu.org/software/octave/)
  -  **Scilab**: CeCILL license (GPL compatible), honlap [www.scilab.org](http://www.scilab.org)  
(fő fejlesztő: INRIA, Franciaország)
  -  **FreeMat**: GPL, honlap [freemat.sourceforge.net/](http://freemat.sourceforge.net/)
- Hasonló szoftverek:
  -  **julia**: MIT licensed, (high-performance dynamic programming language for technical computing), [honlap julialang.org/](http://julialang.org/)
  -  **R**: GPL, statisztikai számításokhoz



- valós és komplex számok

- valós és komplex számok

- ```
>> 1 + 1  
ans = 2
```



- valós és komplex számok

- ```
>> 1 + 1  
ans = 2
```

```
>> 2^23  
ans = 8388608
```

- valós és komplex számok

- ```
>> 1 + 1
```

```
ans = 2
```

```
>> 2^23
```

```
ans = 8388608
```

```
>> 2^-3
```

```
ans = 0.12500
```

- valós és komplex számok
- ```
>> 1 + 1  
ans = 2  
  
>> 2^23  
ans = 8388608  
  
>> 2^-3  
ans = 0.12500  
  
>> 2^123  
ans = 1.0634e+37
```

- valós és komplex számok

- ```
>> 1 + 1
```

```
ans = 2
```

```
>> 2^23
```

```
ans = 8388608
```

```
>> 2^-3
```

```
ans = 0.12500
```

```
>> 2^123
```

```
ans = 1.0634e+37
```

```
>> (1 + 2i) / (3 - 1i)
```

```
ans = 0.10000 + 0.70000i
```

- valós és komplex számok

- ```
>> 1 + 1
```

```
ans = 2
```

```
>> 2^23
```

```
ans = 8388608
```

```
>> 2^-3
```

```
ans = 0.12500
```

```
>> 2^123
```

```
ans = 1.0634e+37
```

```
>> (1 + 2i) / (3 - 1i)
```

```
ans = 0.10000 + 0.70000i
```

```
>> (1 - 1i)^8
```

```
ans = 16
```

# Számolás mátrixokkal

- valós és komplex mátrixok

# Számolás mátrixokkal

- valós és komplex mátrixok
- ```
>> [1 1; 3 3] + [1 2; 2 3]*[3 1; 2 1]^-1
```

```
ans =
```

|          |          |
|----------|----------|
| -2.00000 | 6.00000  |
| -1.00000 | 10.00000 |

# Számolás mátrixokkal

- valós és komplex mátrixok

- `>> [1 1; 3 3] + [1 2; 2 3]*[3 1; 2 1]^-1`

`ans =`

`-2.00000 6.00000`

`-1.00000 10.00000`

`>> [1 1; 3 3] + [1 2; 2 3] * [2 1; 1 1]^-1`

`ans =`

`3.3307e-16 4.0000e+00`

`2.0000e+00 7.0000e+00`



# Számolás mátrixokkal

- valós és komplex mátrixok

```
>> [1 1; 3 3] + [1 2; 2 3]*[3 1; 2 1]^-1
```

```
ans =
```

```
-2.00000    6.00000  
-1.00000   10.00000
```

```
>> [1 1; 3 3] + [1 2; 2 3] * [2 1; 1 1]^-1
```

```
ans =
```

```
3.3307e-16    4.0000e+00  
2.0000e+00    7.0000e+00
```

```
>> [1; 2]
```

```
ans =
```

```
1  
2
```

```
>> ans'
```

```
ans =
```

```
1    2
```

• >> zeros (2, 3)

ans =

0 0 0

0 0 0

>> ones (1, 4)

ans =

1 1 1 1

• >> zeros (2, 3)

ans =

0 0 0

0 0 0

>> ones (1, 4)

ans =

1 1 1 1

>> eye (3)

ans =

Diagonal Matrix

1 0 0

0 1 0

0 0 1

>> diag ([2 3 1])

ans =

Diagonal Matrix

2 0 0

0 3 0

0 0 1

- $A/B = C$  jelentése:  $A = CB$

# Mátrixok osztása és inverze

- $A/B = C$  jelentése:  $A = CB$
- $A \setminus B = C$  jelentése:  $B = AC$

# Mátrixok osztása és inverze

- $\mathbf{A}/\mathbf{B} = \mathbf{C}$  jelentése:  $\mathbf{A} = \mathbf{CB}$
- $\mathbf{A} \backslash \mathbf{B} = \mathbf{C}$  jelentése:  $\mathbf{B} = \mathbf{AC}$
- Az  $\mathbf{Ax} = \mathbf{b}$  egyenletrendszer megoldására két lehetőség:

# Mátrixok osztása és inverze

- $\mathbf{A}/\mathbf{B} = \mathbf{C}$  jelentése:  $\mathbf{A} = \mathbf{CB}$
- $\mathbf{A} \backslash \mathbf{B} = \mathbf{C}$  jelentése:  $\mathbf{B} = \mathbf{AC}$
- Az  $\mathbf{Ax} = \mathbf{b}$  egyenletrendszer megoldására két lehetőség:  
(1)  $\mathbf{x} = \mathbf{A} \backslash \mathbf{b}$ , (2) invertálható együtthatómátrix esetén  $\mathbf{A}^{-1}\mathbf{b}$

# Mátrixok osztása és inverze

- $\mathbf{A}/\mathbf{B} = \mathbf{C}$  jelentése:  $\mathbf{A} = \mathbf{CB}$
- $\mathbf{A} \backslash \mathbf{B} = \mathbf{C}$  jelentése:  $\mathbf{B} = \mathbf{AC}$
- Az  $\mathbf{Ax} = \mathbf{b}$  egyenletrendszer megoldására két lehetőség:  
(1)  $\mathbf{x} = \mathbf{A} \backslash \mathbf{b}$ , (2) invertálható együtthatómátrix esetén  $\mathbf{A}^{-1}\mathbf{b}$
- Oldjuk meg a  $\begin{cases} x - 3y = 15 \\ 4x + 2y = 18 \end{cases}$  egyenletrendszert kétféleképp!



# Mátrixok osztása és inverze

- $\mathbf{A}/\mathbf{B} = \mathbf{C}$  jelentése:  $\mathbf{A} = \mathbf{CB}$
- $\mathbf{A} \backslash \mathbf{B} = \mathbf{C}$  jelentése:  $\mathbf{B} = \mathbf{AC}$
- Az  $\mathbf{Ax} = \mathbf{b}$  egyenletrendszer megoldására két lehetőség:  
(1)  $\mathbf{x} = \mathbf{A} \backslash \mathbf{b}$ , (2) invertálható együtthatómátrix esetén  $\mathbf{A}^{-1}\mathbf{b}$
- Oldjuk meg a  $\begin{cases} x - 3y = 15 \\ 4x + 2y = 18 \end{cases}$  egyenletrendszert kétféleképp!
- ```
>> [1 -3; 4 2] \ [15; 18]
ans =
     6
    -3
```

# Mátrixok osztása és inverze

- $\mathbf{A}/\mathbf{B} = \mathbf{C}$  jelentése:  $\mathbf{A} = \mathbf{CB}$
- $\mathbf{A} \backslash \mathbf{B} = \mathbf{C}$  jelentése:  $\mathbf{B} = \mathbf{AC}$
- Az  $\mathbf{Ax} = \mathbf{b}$  egyenletrendszer megoldására két lehetőség:  
(1)  $\mathbf{x} = \mathbf{A} \backslash \mathbf{b}$ , (2) invertálható együtthatómátrix esetén  $\mathbf{A}^{-1}\mathbf{b}$
- Oldjuk meg a  $\begin{cases} x - 3y = 15 \\ 4x + 2y = 18 \end{cases}$  egyenletrendszert kétféleképp!
- ```
>> [1 -3; 4 2] \ [15; 18]
ans =
     6
    -3

>> [1 -3; 4 2]^-1 * [15; 18]
ans =
     6
    -3
```

- Mátrixok osztása akkor is elvégezhető, ha a „nevezőben” lévő mátrix nem invertálható (ekkor a később tanulandó általánosított inverzzel számol).

- Mátrixok osztása akkor is elvégezhető, ha a „nevezőben” lévő mátrix nem invertálható (ekkor a később tanulandó általánosított inverzzel számol).
- Ezzel mindig megkapható az egyenletrendszer sortérbe eső megoldása, a redukált lépcsős alakkal az összes. Az egyenletrendszer legyen

$$x + y + 2z = 2$$

$$2x + 2y + 3z = 4$$

- Mátrixok osztása akkor is elvégezhető, ha a „nevezőben” lévő mátrix nem invertálható (ekkor a később tanulandó általánosított inverzzel számol).
- Ezzel mindig megkapható az egyenletrendszer sortérbe eső megoldása, a redukált lépcsős alakkal az összes. Az egyenletrendszer legyen

$$x + y + 2z = 2$$

$$2x + 2y + 3z = 4$$

```
>> [1 1 2; 2 2 3] \ [4; 8]
```

```
ans =
```

```
2.0000e+00
```

```
2.0000e+00
```

```
7.1054e-15
```

- Mátrixok osztása akkor is elvégezhető, ha a „nevezőben” lévő mátrix nem invertálható (ekkor a később tanulandó általánosított inverzzel számol).
- Ezzel mindig megkapható az egyenletrendszer sortérbe eső megoldása, a redukált lépcsős alakkal az összes. Az egyenletrendszer legyen

$$x + y + 2z = 2$$

$$2x + 2y + 3z = 4$$

```
>> [1 1 2; 2 2 3] \ [4; 8]
```

```
ans =
```

```
2.0000e+00
```

```
2.0000e+00
```

```
7.1054e-15
```

```
>> rref ([1 1 2; 2 2 3])
```

```
ans =
```

```
1    1    0
```

```
0    0    1
```

- A legtöbb számolás **duplapontosságú lebegőpontos** számokkal történik. Az egész típus inkább csak adatok tárolására vagy megjelenítésére való.

- A legtöbb számolás **duplapontosságú lebegőpontos** számokkal történik. Az egész típus inkább csak adatok tárolására vagy megjelenítésére való.
- **integer**  $b$  biten, vagy **unsigned integer**: int8, uint8, int16, uint16, int32, uint32, int64, uint64.



- A legtöbb számolás **duplapontosságú lebegőpontos** számokkal történik. Az egész típus inkább csak adatok tárolására vagy megjelenítésére való.
- **integer**  $b$  biten, vagy **unsigned integer**: int8, uint8, int16, uint16, int32, uint32, int64, uint64.

```
>> 10 * rand (2, 3)  
ans =
```

```
8.390913    0.500552    6.421794  
9.608188    1.873848    0.028212
```

```
>> int8 (ans)  
ans =
```

```
8      1      6  
10     2      0
```

# Tartományok (ranges)

- kezdet:lépésköz:vég

# Tartományok (ranges)

- kezdet:lépésköz:vég

- `>> 1:4`

`ans =`

1      2      3      4

# Tartományok (ranges)

- kezdet:lépésköz:vég

- `>> 1:4`

`ans =`

1      2      3      4

`>> 4:1`

`ans = [] (1x0)`

# Tartományok (ranges)

- kezdet:lépésköz:vég

- `>> 1:4`

`ans =`

1      2      3      4

`>> 4:1`

`ans = [] (1x0)`

`>> 9:-3:1`

`ans =`

9      6      3

# Tartományok (ranges)

- kezdet:lépésköz:vég

• `>> 1:4`

`ans =`

`1 2 3 4`

`>> 4:1`

`ans = [] (1x0)`

`>> 9:-3:1`

`ans =`

`9 6 3`

`>> 1.1:.237:2.1`

`ans =`

`1.1000 1.3370 1.5740 1.8110 2.0480`

- ```
>> a = 3
```

```
a = 3
```

- `>> a = 3`

`a = 3`

`>> m = [`

`1 2 a`

`2 a 4]`

`m =`

`1 2 3`

`2 3 4`



- `>> a = 3`

`a = 3`

`>> m = [`

`1 2 a`

`2 a 4]`

`m =`

`1 2 3`

`2 3 4`

`>> m' * m`

`ans =`

`5 8 11`

`8 13 18`

`11 18 25`

# Indexek

• >> M

M =

8	1	6
10	2	0

• >> M

M =

8	1	6
10	2	0

>> M (1, 3)

ans = 6

• >> M

M =

8	1	6
10	2	0

>> M (1, 3)

ans = 6

>> M (1, [2 3 1])

ans =

1	6	8
---	---	---

• >> M

M =

8	1	6
10	2	0

>> M (1, 3)

ans = 6

>> M (1, [2 3 1])

ans =

1	6	8
---	---	---

>> M (:, [2 3 1])

ans =

1	6	8
2	0	10

## 1 Programozás

## 2 Vektorosítás

# Függvények

- Külön .m fájlba tesszük, azzal a névvel, mint maga a függvény.

# Függvények

- Külön .m fájlba tesszük, azzal a névvel, mint maga a függvény.
- ```
function fv1  
    1 + 1  
end
```



# Függvények

- Külön .m fájlba tegyük, azzal a névvel, mint maga a függvény.

- ```
function fv1  
    1 + 1  
end
```

- ekkor így hívható:

```
>> fv1  
ans = 2
```

# Függvények

- Külön .m fájlba tegyük, azzal a névvel, mint maga a függvény.

- ```
function fv1  
    1 + 1  
end
```

- ekkor így hívható:

```
>> fv1  
ans = 2
```

```
function fv2 (a, b)  
    a^2 + b^2  
end
```

# Függvények

- Külön .m fájlba tegyük, azzal a névvel, mint maga a függvény.

- ```
function fv1
    1 + 1
end
```

- ekkor így hívható:

```
>> fv1
ans = 2
```

```
function fv2 (a, b)
    a^2 + b^2
end
```

- ```
>> fv2 (3, 4)
ans = 25
```

- *% muvelet.m*  
function [amegb, aszorb] = muvelet (a, b)  
 amegb = a + b;  
 aszorb = a \* b;  
end

- *% muvelet.m*  
function [amegb, aszorb] = muvelet (a, b)  
 amegb = a + b;  
 aszorb = a \* b;  
end  
  
>> [c d] = muvelet (2, 3)  
c = 5  
d = 6

- `% muvelet.m`  
`function [amegb, aszorb] = muvelet (a, b)`  
    `amegb = a + b;`  
    `aszorb = a * b;`  
`end`  
  
`>> [c d] = muvelet (2, 3)`  
`c = 5`  
`d = 6`

- A függvénydefiníció általános alakja (a színes rész opcionális):

- `% muvelet.m`  
`function [amegb, aszorb] = muvelet (a, b)`  
    `amegb = a + b;`  
    `aszorb = a * b;`  
`end`  
  
`>> [c d] = muvelet (2, 3)`  
`c = 5`  
`d = 6`
- A függvénydefiníció általános alakja (a színes rész opcionális):  
`%függvénynév.m`

- `% muvelet.m`  
`function [amegb, aszorb] = muvelet (a, b)`  
    `amegb = a + b;`  
    `aszorb = a * b;`  
`end`  
  
`>> [c d] = muvelet (2, 3)`  
`c = 5`  
`d = 6`

- A függvénydefiníció általános alakja (a színes rész opcionális):  
`%függvénynév.m`  
`function visszaadott érték = függvénynév (argumentumlista)`



- *% muvelet.m*

```
function [amegb, aszorb] = muvelet (a, b)
    amegb = a + b;
    aszorb = a * b;
end
```

```
>> [c d] = muvelet (2, 3)
c =    5
d =    6
```

- A függvénydefiníció általános alakja (a színes rész opcionális):

```
%függvénynév.m
```

```
function visszaadott érték = függvénynév (argumentumlista)
    matlab utasítások
```

- *% muvelet.m*  

```
function [amegb, aszorb] = muvelet (a, b)
    amegb = a + b;
    aszorb = a * b;
end

>> [c d] = muvelet (2, 3)
c =    5
d =    6
```

- A függvénydefiníció általános alakja (a színes rész opcionális):  

```
%függvénynév.m
function visszaadott érték = függvénynév (argumentumlista)
    matlab utasítások
end
```

- `% muvelet.m`

```
function [amegb, aszorb] = muvelet (a, b)
    amegb = a + b;
    aszorb = a * b;
end

>> [c d] = muvelet (2, 3)
c =    5
d =    6
```

- A függvénydefiníció általános alakja (a színes rész opcionális):

```
%függvénynév.m
```

```
function visszaadott érték = függvénynév (argumentumlista)
```

```
    matlab utasítások
```

```
end
```

- Ha a visszaadott érték nincsen vagy nem áll „=” jobb oldalán, akkor a visszaadott érték az utoljára kiszámolt érték.

- `% muvelet.m`  
`function [amegb, aszorb] = muvelet (a, b)`  
 `amegb = a + b;`  
 `aszorb = a * b;`  
`end`  
  
`>> [c d] = muvelet (2, 3)`  
`c = 5`  
`d = 6`

- A függvénydefiníció általános alakja (a színes rész opcionális):

```
%függvéynév.m
function visszaadott érték = függvéynév (argumentumlista)
    matlab utasítások
end
```

- Ha a visszaadott érték nincsen vagy nem áll „=” jobb oldalán, akkor a visszaadott érték az utoljára kiszámolt érték.

- csendes kiértékelés

# Logikai értékek (1 = igaz, 0 = hamis, típusa bool)

- ```
>> 4 > 1  
ans = 1
```

# Logikai értékek (1 = igaz, 0 = hamis, típusa bool)

- `>> 4 > 1`

`ans = 1`

`>> 4 < 1`

`ans = 0`

# Logikai értékek (1 = igaz, 0 = hamis, típusa bool)

- ```
>> 4 > 1
ans = 1

>> 4 < 1
ans = 0

>> 4 == 1
ans = 0
```

# Logikai értékek (1 = igaz, 0 = hamis, típusa bool)

- ```
>> 4 > 1
ans = 1

>> 4 < 1
ans = 0

>> 4 == 1
ans = 0

>> 4 >= 1
ans = 1
```



# Logikai értékek (1 = igaz, 0 = hamis, típusa bool)

```
• >> 4 > 1
  ans = 1

>> 4 < 1
  ans = 0

>> 4 == 1
  ans = 0

>> 4 >= 1
  ans = 1

>> (4 >= 1) == 1
  ans = 1

>> whos
      ans          1x1    logical
```

# Logikai értékek (1 = igaz, 0 = hamis, típusa bool)

```
• >> 4 > 1
  ans = 1

>> 4 < 1
  ans = 0

>> 4 == 1
  ans = 0

>> 4 >= 1
  ans = 1

>> (4 >= 1) == 1
  ans = 1

>> whos
      ans          1x1    logical

>> a = true
  a = 1
```

# Feltételes utasítás

- `if` (*feltétel*)

# Feltételes utasítás

- `if` (*feltétel*)  
*matlab* utasítások

# Feltételes utasítás

- `if` (*feltétel*)

*matlab* utasítások

`else`

*% ez a rész*

# Feltételes utasítás

- if (*feltétel*)

*matlab utasítások*

else

*% ez a rész*

*matlab utasítások*

*% kimaradhat*

# Feltételes utasítás

- if (*feltétel*)

*matlab utasítások*

else

*% ez a rész*

*matlab utasítások*

*% kimaradhat*

end

# Feltételes utasítás (példa)

```
• function valosvagykomplex (a, b, c)
    d = b^2 - 4*a*c;  % ; hogy ne írja ki d értéket
    if (d >= 0)
        "valos"
    else
        "komplex"
    end
end
```



# Feltételes utasítás (példa)

```
• function valosvagykomplex (a, b, c)
    d = b^2 - 4*a*c;  % ; hogy ne írja ki d erteket
    if (d >= 0)
        "valos"
    else
        "komplex"
    end
end

>> valosvagykomplex (1, 1, 1)
ans = komplex
```

# Feltételes utasítás (példa)

```
• function valosvagykomplex (a, b, c)
    d = b^2 - 4*a*c;  % ; hogy ne írja ki d erteket
    if (d >= 0)
        "valos"
    else
        "komplex"
    end
end
```

```
>> valosvagykomplex (1, 1, 1)
ans = komplex
```

```
>> valosvagykomplex (1,2,1)
ans = valos
```

1 Programozás

2 Vektorosítás

# Vektorosítás

- A cél, hogy a kódismétlés és a ciklusutasítások helyett vektor-/mátrixműveleteket használjunk.

# Vektorosítás

- A cél, hogy a kódismétlés és a ciklusutasítások helyett vektor-/mátrixműveleteket használjunk.
- Így hatékonyabb, tömörebb kód jön létre.

# Vektorosítás

- A cél, hogy a kódismétlés és a ciklusutasítások helyett vektor-/mátrixműveleteket használjunk.
- Így hatékonyabb, tömörebb kód jön létre.
- Írjunk kódot, mely egy adott  $(v_1, v_2, \dots, v_n)$  vektorból a különbségek  $(v_2 - v_1, v_3 - v_2, \dots, v_n - v_{n-1})$  vektorát képzí!

# Vektorosítás

- A cél, hogy a kódismétlés és a ciklusutasítások helyett vektor-/mátrixműveleteket használjunk.
- Így hatékonyabb, tömörebb kód jön létre.
- Írjunk kódot, mely egy adott  $(v_1, v_2, \dots, v_n)$  vektorból a különbségek  $(v_2 - v_1, v_3 - v_2, \dots, v_n - v_{n-1})$  vektorát képzzi!

```
>> l = [3 4 6 2 5 1]
l =
     3     4     6     2     5     1
>> l(2:6) - l(1:5)
ans =
     1     2    -4     3    -4
```

- Alkalmazzuk az  $f(x) = x^2 + 3x + 1$  képletet egy mátrixra!

```
function keplet(a)
    a^2 + 3*a + 1
end
>> keplet( [1 1;0 2])
ans =
     5     7
     1    11
```



- Alkalmazzuk az  $f(x) = x^2 + 3x + 1$  képletet egy **mátrixra**!

```
function keplet(a)
    a^2 + 3*a + 1
end
>> keplet( [1 1;0 2])
ans =
     5     7
     1    11
```

- Alkalmazzuk az  $f(x) = x^2 + 3x + 1$  képletet egy **mátrix minden elemére**!

```
function keplet(a)
    a .^ 2 + 3 .* a + 1
end
>> keplet ([1:3 5])
ans =
     5    11    19    41
```

# Kérdések

- Mire és hogyan használhatók az `eye`, `ones`, `zeros` és `diag` parancsok?

# Kérdések

- Mire és hogyan használhatók az `eye`, `ones`, `zeros` és `diag` parancsok?
- Hozzunk létre egy `m` nevű véletlen  $4 \times 5$ -ös mátrixot, melynek elemei 0-tól 9-ig terjedő egészek!

# Kérdések

- Mire és hogyan használhatók az `eye`, `ones`, `zeros` és `diag` parancsok?
- Hozzunk létre egy `m` nevű véletlen  $4 \times 5$ -ös mátrixot, melynek elemei 0-tól 9-ig terjedő egészek!
- Soroljuk fel a tanult adattípusokat! Melyik paranccsal jutunk ehhez az információhoz egy adatról?

# Kérdések

- Mire és hogyan használhatók az `eye`, `ones`, `zeros` és `diag` parancsok?
- Hozzunk létre egy `m` nevű véletlen  $4 \times 5$ -ös mátrixot, melynek elemei 0-tól 9-ig terjedő egészek!
- Soroljuk fel a tanult adattípusokat! Melyik paranccsal jutunk ehhez az információhoz egy adatról?
- Írjunk függvényt, melynek  $x$  a bemenete és az  $x + \frac{4}{x^2} + \frac{1}{x^3}$  értéket adja vissza!

# Kérdések

- Mire és hogyan használhatók az `eye`, `ones`, `zeros` és `diag` parancsok?
- Hozzunk létre egy `m` nevű véletlen  $4 \times 5$ -ös mátrixot, melynek elemei 0-tól 9-ig terjedő egészek!
- Soroljuk fel a tanult adattípusokat! Melyik paranccsal jutunk ehhez az információhoz egy adatról?
- Írjunk függvényt, melynek  $x$  a bemenete és az  $x + \frac{4}{x^2} + \frac{1}{x^3}$  értéket adja vissza!
- Írjunk meg ugyanezt a függvényt, de úgy, hogy egy mátrixra alkalmazva, annak minden elemére kiszámítsa az értékét!

# Kérdések

- Mire és hogyan használhatók az `eye`, `ones`, `zeros` és `diag` parancsok?
- Hozzunk létre egy `m` nevű véletlen  $4 \times 5$ -ös mátrixot, melynek elemei 0-tól 9-ig terjedő egészek!
- Soroljuk fel a tanult adattípusokat! Melyik paranccsal jutunk ehhez az információhoz egy adatról?
- Írjunk függvényt, melynek  $x$  a bemenete és az  $x + \frac{4}{x^2} + \frac{1}{x^3}$  értéket adja vissza!
- Írjunk meg ugyanezt a függvényt, de úgy, hogy egy mátrixra alkalmazva, annak minden elemére kiszámítsa az értékét!
- Mi az eredménye a  $1 == (2 > 4)$  és a  $0 == (2 > 4)$  műveletnek? Ezeknek mi az adattípusa?

# Kérdések

- Mire és hogyan használhatók az `eye`, `ones`, `zeros` és `diag` parancsok?
- Hozzunk létre egy `m` nevű véletlen  $4 \times 5$ -ös mátrixot, melynek elemei 0-tól 9-ig terjedő egészek!
- Soroljuk fel a tanult adattípusokat! Melyik paranccsal jutunk ehhez az információhoz egy adatról?
- Írjunk függvényt, melynek  $x$  a bemenete és az  $x + \frac{4}{x^2} + \frac{1}{x^3}$  értéket adja vissza!
- Írjunk meg ugyanezt a függvényt, de úgy, hogy egy mátrixra alkalmazva, annak minden elemére kiszámítsa az értékét!
- Mi az eredménye a `1 == (2 > 4)` és a `0 == (2 > 4)` műveletnek? Ezeknek mi az adattípusa?
- Egy  $4 \times 5$ -ös véletlen mátrixon végezzünk elemi sorműveleteket!



# Kérdések

- Mire és hogyan használhatók az `eye`, `ones`, `zeros` és `diag` parancsok?
- Hozzunk létre egy `m` nevű véletlen  $4 \times 5$ -ös mátrixot, melynek elemei 0-tól 9-ig terjedő egészek!
- Soroljuk fel a tanult adattípusokat! Melyik paranccsal jutunk ehhez az információhoz egy adatról?
- Írjunk függvényt, melynek  $x$  a bemenete és az  $x + \frac{4}{x^2} + \frac{1}{x^3}$  értéket adja vissza!
- Írjunk meg ugyanezt a függvényt, de úgy, hogy egy mátrixra alkalmazva, annak minden elemére kiszámítsa az értékét!
- Mi az eredménye a  $1 == (2 > 4)$  és a  $0 == (2 > 4)$  műveletnek? Ezeknek mi az adattípusa?
- Egy  $4 \times 5$ -ös véletlen mátrixon végezzünk elemi sorműveleteket!
- Írjunk függvényt, mely az argumentumába írt számról eldönti az előjelét és azt karakterláncként kiírja!

# Kérdések

- Képezzünk egy  $10 \times 10$ -es mátrixot, melynek blokkmátrix alakja:

$$\begin{bmatrix} \mathbf{O}_5 & \mathbf{I}_5 \\ \mathbf{I}_5 & \mathbf{O}_5 \end{bmatrix}$$

# Kérdések

- Képezzünk egy  $10 \times 10$ -es mátrixot, melynek blokkmátrix alakja:

$$\begin{bmatrix} \mathbf{O}_5 & \mathbf{I}_5 \\ \mathbf{I}_5 & \mathbf{O}_5 \end{bmatrix}$$

- Oldjuk meg az

$$x - 2y = 1$$

$$2x + y = 7$$

egyenletrendszert mátrixosztással, és az inverzzel való szorzással.

## Kérdések

- Képezzünk egy  $10 \times 10$ -es mátrixot, melynek blokkmátrix alakja:

$$\begin{bmatrix} \mathbf{O}_5 & \mathbf{I}_5 \\ \mathbf{I}_5 & \mathbf{O}_5 \end{bmatrix}$$

- Oldjuk meg az

$$x - 2y = 1$$

$$2x + y = 7$$

egyenletrendszert mátrixosztással, és az inverzzel való szorzással.

- Számítsuk ki az

$$x - 2y + 3z = -1$$

$$2x + y + z = 3$$

egyenletrendszer egyetlen sortérbe eső megoldását, majd az együtthatómátrix redukált lépcsős alakját!