Tema Laborator 1

NICOLETA RADU

# Exercitiul 1

a. Accesarea unei valori prin specificarea indexului.

b. A fost incercata accesarea tuturor valorilor vectorului de la index 1 pana la index 7. Indexul introdus a fost gresit deoarece marimea vectorului este de 6, nu de 7. Rezultatul a fost un mesaj de eroare.

c. Accesarea tuturor valorilor vectorului de la primul index pana la ultimul.

d. Accesarea tuturor valorilor vectorului de la primul index pana la penultimul.

e. Accesarea valorilor de la index 6 pana la index 1, cu pas -2. Ultima valoare nu a fost printata deoarece pasul a trecut de marimea vectorului.

f. Accesarea unor anumite valori prin specificarea indexilor.

g. O alta metoda de accesare a unor valori prin specificarea indexilor.

h. Accesarea unei valori prin specificarea unui index de start, a unui pas si a unui sfarsit.

i. Transformarea din vector linie in vector coloana. Valorile au fost printate prin specificarea indexilor.

j. Introducerea unor valori noi in mijlocul vectorului.

k. Accesarea valorilor din vector prin specificarea pozitiilor acestora.

l. Construirea unei matrice de 3x7 formata din valorile vectorului x?

# Exercitiul 2

1. >> x = [2,5,1,6,7,9,12]

**>> u = x([2:2:end])**

u =

5 6 9

1. **>> length(x)**

ans =

7

1. **>> xN = x([end:-1:1])**

xN =

12 9 7 6 1 5 2

1. Varianta 1:

**>> y = x([1 2 6 7])**

Varianta 2:

**>> y = [x(1) x(2) x(6) x(7)]**

# Exercitiul 3

**>> m = [32:2:75]**

m =

Columns 1 through 17

32 34 36 38 40 42 44 46 48 50 52 54 56 58 60 62 64

Columns 18 through 22

66 68 70 72 74

# Exercitiul 4

**c1 =**

1.0000 + 2.0000i

>> c2 = -5i

**c2 =**

0.0000 - 5.0000i

>> c3 = 2.5j + 6

**c3 =**

6.0000 + 2.5000i

>> c4 = -3i + 1

**c4 =**

1.0000 - 3.0000i

# Exercitiul 5

1. **>> M = [3 4 ; 2 1 ; 6 3 ; 8 5]**

M =

3 4

2 1

6 3

8 5

1. **>> size(M)**

ans =

4 2

1. **>> c = [x(1) + y(1) \* i; x(2) + y(2) \* j; x(3) + y(3) \* i; x(4) + y(4) \* j]**

c =

3.0000 + 4.0000i

2.0000 + 1.0000i

6.0000 + 3.0000i

8.0000 + 5.0000i

# Exercitiul 6

1. **a = [2:2:20**]

a =

2 4 6 8 10 12 14 16 18 20

1. **>> b= [10:-2:-4]**

b =

10 8 6 4 2 0 -2 -4

1. **>> c = [-2:1:35**]

c =

Columns 1 through 17

-2 -1 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14

Columns 18 through 34

15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

Columns 35 through 38

32 33 34 35

1. **>> d = [4.50:-0.50:-2.50]**

d =

Columns 1 through 10

4.5000 4.0000 3.5000 3.0000 2.5000 2.0000 1.5000 1.0000 0.5000 0

Columns 11 through 15

-0.5000 -1.0000 -1.5000 -2.0000 -2.5000

# Exercitiul 7

1. **>> v = [v(1:end), 5,6]**

v =

2 3 4 5 6

1. **>> v = [21, v(1:end)]**

v =

21 2 3 4 5 6

1. **>> v = [v(1:3), 3.5 , v(4:end)]**

v =

21.0000 2.0000 3.0000 3.5000 4.0000 5.0000 6.0000

1. **>> v = [v(8), v(1), v(2:1:end-1)]**

v =

6.0000 21.0000 2.0000 3.0000 3.5000 4.0000 5.0000

1. >> **v = [v(end:-1:1)**]

v =

5.0000 4.0000 3.0000 3.5000 2.0000 21.0000 6.0000

1. **>> v = [v(2:1:end-1)]**

v =

4.0000 3.0000 3.5000 2.0000 21.0000

1. **>> v = [v(1:1:2), v(4:1:end)]**

v =

4 3 2 21

# Exercitiul 8

1. Coloanele matricei A s-au transformat in linii (**transpura matricei A** a fost printata)
2. **Toate valorile** de pe coloanele 1 si 4 au fost printate.
3. Elementele de pe **liniile [2 3]** si **coloanele [3 1**] au fost printate.
4. Matricea a fost printata sub forma unui **vector coloana**.
5. **Intreaga matrice** a fost printata.
6. A fost printata **intreaga matrice** si apoi au fost printate **randurile 1 si 2**.
7. Au fost printate valorile de pe **linia 2** si **coloanele 1 si 3**.
8. A fost printata **transpusa vectorului coloana**.
9. **Intreaga matrice** a fost printata.
10. Matricea a fost printata de doua ori **una sub cealalta**.
11. Matricea a fost printata de doua ori **una langa cealalta**.

# Exercitiul 9

1. **>> B = A(:, 2:2:end)**

B =

7 7

1 6

1. 5
2. **>> C = A(1:2:end,** **:)**

C =

2 7 9 7

8 1 2 5

1. **>> D = A(1:2:end, 1:2)**

D =

2 7

8 1

>>

# Exercitiul 10

1. **>> N = [N(:,1:2) m]**

N =

2 3 5

4 5 6

1. **>> N = [n ; N(:, 1:3)**]

N =

1 1 1

2 3 5

4 5 6

1. **>> N = [N([3], :) ; N([2], :) ; N([1], :)]**

N =

4 5 6

2 3 5

1 1 1

1. **>> [N([3], [2])]**

ans =

1

1. **>> [N([3], :)]**

ans =

1 1 1

1. **>> rez = N(:)**

rez =

4

2

1

5

3

1

6

5

1

1. **>> [lin,col]=size(N**)

lin =

3

col =

3

1. **>> N = [N(2:3, :)]**

N =

2 3 5

1 1 1

1. **>> max(size(N))** si **>> max(length(N))**ans =

3

1. **>> N = [N([2],:)]**

N =

1 1 1