

# Лекція 12

## Робота з масивами та векторами в Матлаб.



## Лекція 4

**Тема:** Робота з масивами та векторами в Матлаб.

1. Поняття масиву, його числа вимірів та розміру.
2. Вектор-рядок, вектор стовпчик, їх введення та зберігання в Матлаб.
3. Дії з векторами в Матлаб.
4. Транспонування та 3 спряження векторів.

# 1. Поняття масиву, його числа вимірів та розміру.

У системі MATLAB визначено 15 базових типів даних, кожен із яких є тим чи іншим видом масиву, інакше кажучи, всі об'єкти системи MATLAB є масивами. Двовимірний масив мінімального розміру – це порожній масив розміру  $0 \times 0$ , максимальний розмір та розмірності багатовимірного масиву не обмежені. Для введення масивів у стовпчик при розділенні елементів масиву використовується крапка з комою – ;, а при введенні масивів у рядки між елементами використовується кома – ,.

Розмірність (dimensions) — кількість вимірів (рядки, стовпці, сторінки тощо).

Розмір (size) — кількість елементів у кожному вимірі.

```
1 A = [1 2 3; 4 5 6];  
2 size_A = size(A)  
3 ndims_A = ndims(A)  
4
```

```
size_A =  
      2      3  
  
ndims_A =  
      2
```

Приклад в Matlab

Результат програми

## 2. Вектор-рядок, вектор стовпчик, їх введення та зберігання в Матлаб.

У MATLAB вектор — це одномірний масив, який може бути у вигляді:

- Вектор-рядок ( $1 \times n$ ) — горизонтальне розташування елементів.

Використовуються пробіли або коми для розділення елементів:

```
row_vec = [10 20 30 40]; % або [10, 20, 30, 40]
```

- Вектор-стовпчик ( $n \times 1$ ) — вертикальне розташування елементів.

Використовується крапка з комою для розділення рядків:

```
col_vec = [10; 20; 30; 40];
```

Після створення вектора в MATLAB, його можна зберегти у змінну для подальшого використання:

```
1 a = [1 2 3]; % вектор-рядок
2 b = [1; 2; 3]; % вектор - стовпчик
3 size(a) % перевірка розміру
4 size(b) % перевірка розміру
```

# 3. Дії з векторами в Матлаб

Для покомпонентних операцій вектори мають бути однакової довжини!

- Арифметичні операції ( додавання / віднімання)

```
a = [1, 2, 3];
```

```
b = [4, 5, 6];
```

```
a + b      % → [5 7 9]
```

```
a - b      % → [-3 -3 -3]
```

- Множення та ділення

Покомпонентне множення `.*`

`a .* b`      `% → [4 10 18]`

Покомпонентне ділення `./`

`b ./ a`      `% → [4.00 2.50 2.00]`

- Множення на скаляр

`2 * a`      `% → [2 4 6]`

- Логічні операції

```
x = [5, 10, 15];
```

```
x > 8
```

```
x == 10
```

Повертає логічний вектор (1 – істина, 0 – хибність).

```
1x3 logical array
```

```
0    1    1
```

```
ans =
```

```
1x3 logical array
```

```
|
```

```
0    1    0
```

- Інші корисні функції

| Функція                | Опис                   | Приклад                            |
|------------------------|------------------------|------------------------------------|
| <code>sum(v)</code>    | Сума всіх елементів    | <code>sum([1 2 3]) → 6</code>      |
| <code>prod(v)</code>   | Добуток всіх елементів | <code>prod([2 3 4]) → 24</code>    |
| <code>max(v)</code>    | Максимальний елемент   | <code>max([5 9 7]) → 9</code>      |
| <code>min(v)</code>    | Мінімальний елемент    | <code>min([5 9 7]) → 5</code>      |
| <code>mean(v)</code>   | Середнє значення       | <code>mean([1 2 3]) → 2</code>     |
| <code>length(v)</code> | Кількість елементів    | <code>length([1 2 3 4]) → 4</code> |

## 4. Транспонування та 3 спряження векторів.

- Транспонування — це перетворення вектор-рядка у вектор-стовпчик і навпаки. ( `'` )

```
x = [1, 2, 3];      % вектор-рядок
y = x.';           % транспонований вектор
disp (y)           % виведення y

v = [1; 2; 3]; % вектор-стовпчик
p = v.'; % транспонований вектор
disp (p)
```

>> transponuvannya

1  
2  
3

1      2      3

- Комплексне спряження та транспонування ( `'` )

Оператор ( `'` ) виконує одночасно транспонування та комплексне спряження.

```
z = [1 + 2i, 3 - 4i]; % вектор-рядок з комплексними числами
z_conj_transpose = z'; % Транспонування з комплексним спряженням
disp(z_conj_transpose); % Вивести результат
```

>> transponuvannya

1.0000 - 2.0000i  
3.0000 + 4.0000i

- Комплексне спряження без транспонування: `conj()`

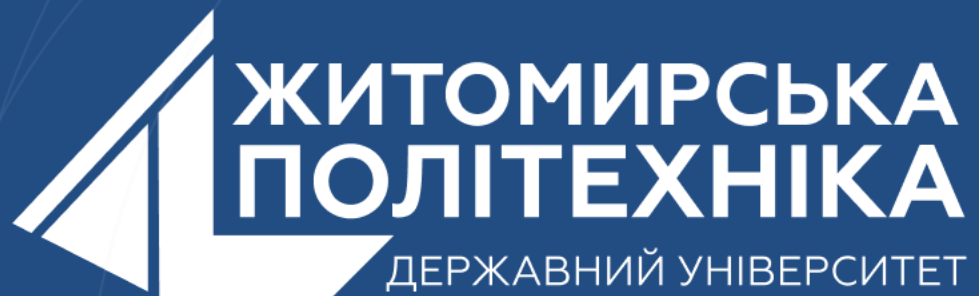
---

```
z = [1 + 2i, 3 - 4i]; % Вектор з комплексними числами  
conj_z = conj(z); % Комплексне спряження вектора без транспонування  
disp(conj_z); % Вивести результат
```

```
>> transponuvannya  
1.0000 - 2.0000i    3.0000 + 4.0000i
```

`conj(z)` — повертає вектор, де кожен елемент є комплексним спряженням відповідного елемента з вектора `z`. Ви зберігаєте результат в змінній `conj_z` і потім виводите її за допомогою `disp`.

Telegram Instagram Facebook @ZTUEDUUA



- Розвиваємо лідерів
- Створюємо інновації
- Змінюємо світ на краще

