

Tham Khảo Nhanh MATLAB

Mảng, ma trận, vẽ đồ thị, file I/O, luồng điều khiển

Cơ Bản

Cửa Sổ Lệnh

```
x = 5;           % assign (semicolon suppresses output)
x = 5           % assign and display result
disp('Hello')  % print to console
clc           % clear command window
clear        % clear all variables
```

Trợ Giúp & Thông Tin

```
help sin      % quick help for function
doc sin      % open documentation
who          % list variables in workspace
whos        % list with details (size, type)
```

Toán Tử

```
+ - * / ^      Số học (phép tính ma trận)
.* ./ .^      Phép tính theo từng phần tử
== ~= < > <= >= Toán tử so sánh
&& || ~      Logic AND, OR, NOT (scalars)
& | ~        Logic theo từng phần tử (arrays)
```

Biến & Kiểu Dữ Liệu

Kiểu Số

```
x = 3.14;      % double (default)
n = int32(42); % 32-bit integer
z = 2 + 3i;    % complex number
tf = true;     % logical
```

Kiểm Tra Kiểu

```
class(x)      Trả về tên kiểu dạng chuỗi
isa(x, 'double') Kiểm tra kiểu cụ thể
isnumeric(x)  True nếu là kiểu số
ischar(x)     True nếu là mảng ký tự
islogical(x)  True nếu là kiểu logic
```

Hằng Số Đặc Biệt

```
pi           3.14159...
Inf / -Inf   Vô cực
NaN         Not a Number
eps         Machine epsilon (~2.2e-16)
i / j       Đơn vị ảo
```

Mảng & Ma Trận

Tạo Mảng

```
v = [1 2 3 4 5]; % row vector
v = [1; 2; 3];  % column vector
A = [1 2; 3 4]; % 2x2 matrix
r = 1:5;        % [1 2 3 4 5]
r = 0:0.5:2;   % [0 0.5 1 1.5 2]
```

Hàm Tạo Tích Hợp

```
zeros(3)      % 3x3 of zeros
ones(2, 4)    % 2x4 of ones
eye(3)        % 3x3 identity
rand(2, 3)    % 2x3 uniform random
linspace(0,1,5) % 5 evenly spaced [0..1]
```

Indexing & Slicing

```
A(2, 3)       % row 2, col 3
A(1, :)      % entire first row
A(:, 2)      % entire second column
A(1:2, 1:2)  % submatrix
A(end, :)    % last row
```

Phép Tính Ma Trận

```
A'           Chuyển vị (liên hợp)
A.'          Chuyển vị (không liên hợp)
inv(A)       Ma trận nghịch đảo
det(A)       Định thức
eig(A)       Trị riêng và vector riêng
A \ b        Giải Ax = b
size(A)      Kích thước [hàng cột]
numel(A)     Tổng số phần tử
```

Luồng Điều Khiển

if / elseif / else

```
if x > 0
    disp('positive')
elseif x == 0
    disp('zero')
else
    disp('negative')
end
```

for & while

```
for i = 1:10
    fprintf('i = %d\n', i);
end
while x > 0
    x = x - 1;
end
```

switch

```
switch grade
case 'A'
    disp('Excellent')
case {'B', 'C'}
    disp('Good')
otherwise
    disp('Try harder')
end
```

Điều Khiển Vòng Lặp

```
break        Thoát vòng lặp trong cùng
continue     Bỏ qua lần lặp hiện tại
return       Thoát hàm ngay lập tức
```

Hàm

Function File

```
% Save as myfunc.m
function result = myfunc(x, y)
    result = x.^2 + y.^2;
end
```

Nhiều Đầu Ra

```
function [mn, mx] = minmax(v)
    mn = min(v);
    mx = max(v);
end
[lo, hi] = minmax([3 1 4 1 5]);
```

Hàm Ẩn Danh

```
f = @(x) x.^2 + 1;
f(3) % returns 10
g = @(x,y) x + y;
arrayfun(f, [1 2 3]) % apply to each element
```

Hàm Tích Hợp Hữu Dụng

```
sum(v)       Tổng các phần tử
mean(v)      Giá trị trung bình
max(v) / min(v) Giá trị lớn nhất / nhỏ nhất
sort(v)      Sắp xếp tăng dần
find(v > 3)  Chỉ số thỏa điều kiện
length(v)    Độ dài vector
```

Vẽ Đồ Thị

Đồ Thị 2D

```
x = 0:0.1:2*pi;
plot(x, sin(x), 'r-', 'LineWidth', 2)
xlabel('x'); ylabel('sin(x)')
title('Sine Wave'); grid on
legend('sin(x)')
```

Nhiều Đồ Thị

```
hold on
plot(x, sin(x), 'b-')
plot(x, cos(x), 'r--')
hold off
subplot(1,2,1); plot(x, sin(x))
subplot(1,2,2); plot(x, cos(x))
```

Loại Đồ Thị Khác

```
bar(x, y)    Biểu đồ thanh
histogram(data) Biểu đồ tần số
scatter(x, y) Biểu đồ phân tán
pie(data)    Biểu đồ tròn
surf(X, Y, Z) Đồ thị mặt 3D
imagesc(A)   Hiển thị ma trận như ảnh
```

Lưu Hình

```
saveas(gcf, 'plot.png')
exportgraphics(gcf, 'plot.pdf')
```

File I/O

File Văn Bản

```
data = readmatrix('data.csv');
writematrix(A, 'output.csv')
T = readtable('data.csv');
writetable(T, 'output.csv')
```

MAT Files

```
save('workspace.mat') % save all variables
save('data.mat', 'x', 'y') % save specific vars
load('data.mat') % load into workspace
S = load('data.mat'); % load into struct
```

File I/O Cấp Thấp

```
fid = fopen('log.txt', 'w');
fprintf(fid, 'Value: %f\n', 3.14);
fclose(fid);
lines = readlines('log.txt');
```

Thao Tác Chuỗi

String vs Char Array

```
s = "Hello"; % string (double quotes)
c = 'Hello'; % char array (single quotes)
s + " World" % "Hello World" (string)
[c, ' World'] % "Hello World" (char concat)
```

Tham Khảo Nhanh MATLAB

Hàm String

strlength(s)	Độ dài chuỗi
upper(s) / lower(s)	Chuyển đổi hoa/thường
contains(s, pat)	True nếu tìm thấy pattern
replace(s, old, new)	Thay thế chuỗi con
split(s, delim)	Tách thành mảng
join(arr, delim)	Nối mảng chuỗi
strip(s)	Xóa khoảng trắng đầu/cuối

Định Dạng

```
sprintf('x = %.2f', 3.14159) % "x = 3.14"
fprintf('i = %d\n', 42) % print to console
num2str(3.14) % number to string
str2double("3.14") % string to number
```

Cell & Struct

Cell Arrays

```
C = {1, 'hello', [1 2 3]}; % mixed types
C{2} % access: 'hello'
C{end+1} = true; % append element
cellfun(@length, C) % apply func to each
```

Structs

```
s.name = 'Alice';
s.age = 30;
s.scores = [90 85 92];
fieldnames(s) % {'name', 'age', 'scores'}
rmfield(s, 'age') % remove field
```

Mảng Struct

```
people(1).name = 'Alice'; people(1).age = 30;
people(2).name = 'Bob'; people(2).age = 25;
{people.name} % {'Alice', 'Bob'}
[people.age] % [30, 25]
```

Mẫu Phổ Biến

Phép Tính Vectorized

```
% Avoid loops – use vectorization
v = 1:1000;
result = sum(v.^2); % fast
idx = v(v > 500 & v < 600); % logical indexing
```

Thao Tác Table

```
T = table([25;30], ["A";"B"], 'VariableNames', ...
         {'Age', 'Grade'});
T.Age % access column
T(T.Age > 25, :) % filter rows
```

Xử Lý Lỗi

```
try
    result = riskyFunction(x);
catch ME
    fprintf('Error: %s\n', ME.message);
end
```

Đo Thời Gian

```
tic
heavyComputation();
toc % prints elapsed time
```