**MATLAB Tutorial**

1. **The environment**
2. **Arrays**
3. **Create an array**

***Create an array from numbers***

>> a = [1 2 3]

a =

1 2 3

Or

>> a = 1:3

***Create an array from other arrays***

>> b = [a a]

b =

1 2 3 1 2 3

>> b = [a; a]

b =

1 2 3

1 2 3

>> b = [a; 1]

Error using vertcat

Dimensions of matrices being concatenated are not consistent.

***Create special arrays***

>> z = zeros(2,3)

z =

0 0 0

0 0 0

>> z = ones(2,3)

z =

1 1 1

1 1 1

>> z = eye(2)

z =

1 0

0 1

>> z = rand(2,3)

z =

0.2760 0.6551 0.1190

0.6797 0.1626 0.4984

>> whos z

Name Size Bytes Class Attributes

z 2x3 48 double

1. **All operators act on arrays**

>> a = [1 2; 3 4]

a =

1 2

3 4

>> a + a

ans =

2 4

6 8

>> a \* a

ans =

7 10

15 22

>> a .\* a

ans =

1 4

9 16

>> b = a(:)

b =

1

3

2

4

>> reshape(b,[2 2])

ans =

1 2

3 4

1. **Index**

>> a = magic(3)

a =

8 1 6

3 5 7

4 9 2

>> a(:,1)

ans =

8

3

4

>> a(1,:)

ans =

8 1 6

>> find(a>5)

ans =

1

6

7

8

>> a(a>5) = 0

a =

0 1 0

3 5 0

4 0 2

1. **Cell array**

***Cell***

>> a = 1:3

a =

1 2 3

>> b = 'dogs'

b =

dogs

>> c = 'cats'

c =

cats

>> d\_cell = {a,b,c}

d\_cell =

[1x3 double] 'dogs' 'cats'

>> d3 = d\_cell{3}

d3 =

cats

***struct***

***…***

1. **for, while, if/else**

**Meanwhile, we will learn how to write a script/function (.m file). We will also learn how to debug.**

This function is to check whether a matrix is an identity matrix.

function is\_eye(a\_mat)

% Default a\_mat, if no input

if nargin < 1

a\_mat = magic(3);

end

% Check size first

% if/else

if size(a\_mat,1) ~= size(a\_mat,2)

disp('not an identity matrix!');

return;

else

disp('size is OK!');

N = size(a\_mat,1);

end

% Check element

% for loop

for i = 1:N

for j = 1:N

flag = check\_elem(i,j,a\_mat(i,j));

if ~flag

disp('NOT an identity matrix!');

return

end

end

end

disp('IS an identity matrix!');

disp(a\_mat);

% Just for illustration of

% while loop

i = 1;

while i <= min(100,N)

disp(a\_mat(i,i));

i = i+1;

% if i > 100 % will only display the first 100 diag elements

% break

% end

end

end

% check the (i,j) element

function flag = check\_elem(i,j,val)

if i == j

flag = val == 1;

else

flag = val == 0;

end

end

1. **Plot**

***1-D data***

>> x = linspace(1,50,10);

>> y = 3 \* x + 1;

>> plot(x,y,'ko-')

***2-D data***

>> a = magic(3)

a =

8 1 6

3 5 7

4 9 2

>> imagesc(a)

>> contourf(a)

1. **Input/Output**

**Output**

>> dlmwrite('a.txt',a,'delimiter',' ')

fprintf: formatted output, mixed data (string + numbers)

**Input**

>> b = load('a.txt')

importdata: data with headers

textscan: mixed data…

1. **Example**

generate\_data.m