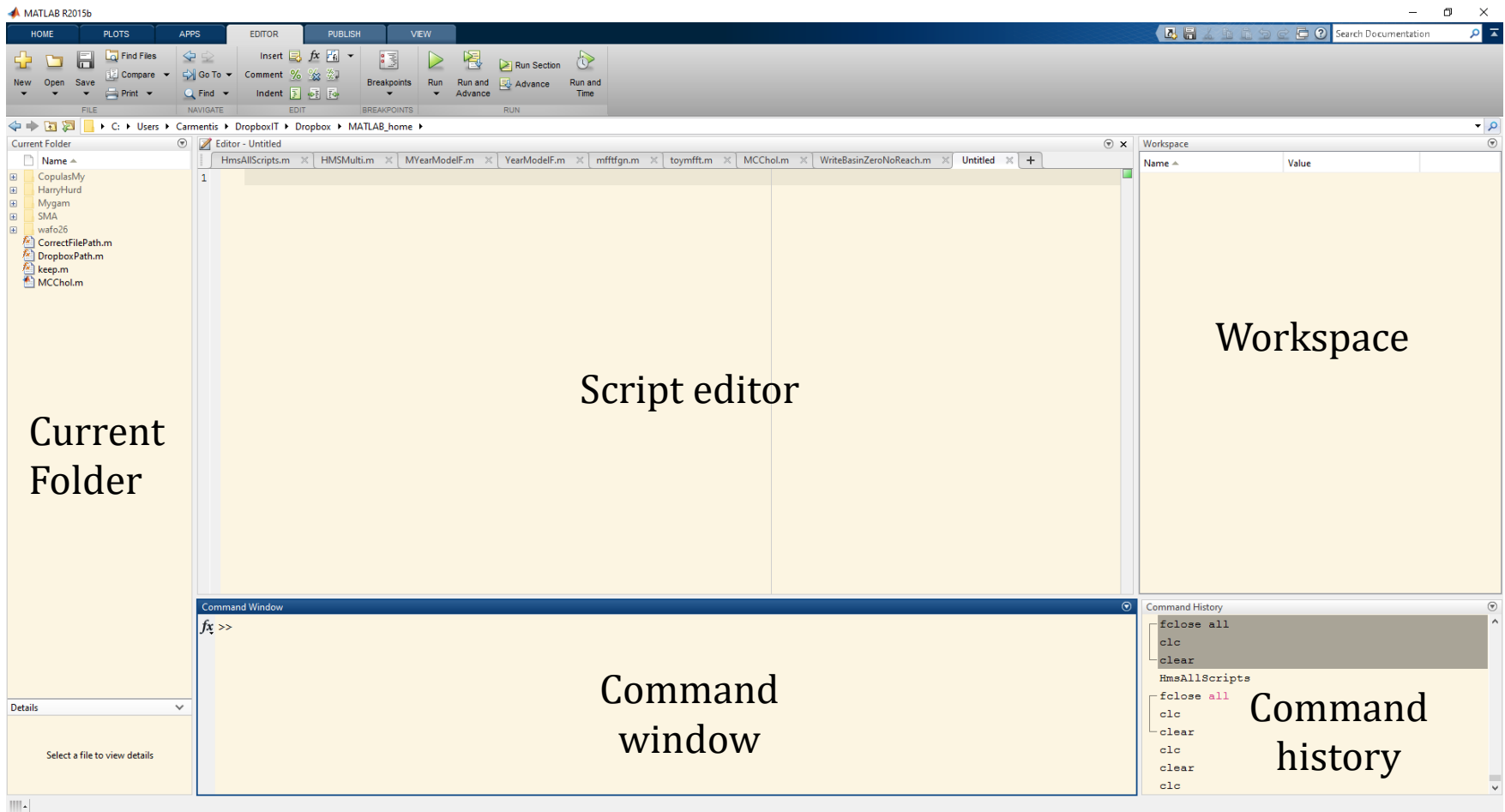


Βελτιστοποίηση Συστημάτων Υδατικών πόρων -
Υδροπληροφορική

Εισαγωγή στον προγραμματισμό με MATLAB: Εφαρμογές στην υδροπληροφορική

Τσουκαλάς Γιάννης, Ευστρατιάδης Ανδρέας & Μακρόπουλος Χρήστος
Τομέας Υδατικών Πόρων και Περιβάλλοντος
Εθνικό Μετσόβιο Πολυτεχνείο

Το προγραμματιστικό περιβάλλον MATLAB (IDE)



*IDE – Integrated Development Environment

Theme: <http://www.mathworks.com/matlabcentral/fileexchange/50446-benhager-solarized-matlab>

Back to basics #1

- Οτιδήποτε είναι μετά από το σύμβολο «%» αποτελεί σχόλιο (comment) και αγνοείται από το MATLAB

- Το MATLAB είναι case-sensitive. πχ.,

```
>> X=2
```

```
>> x=5
```

```
>> y=x+X
```

- Βασικοί τύποι μεταβλητών στο MATLAB

```
>> x=4; % class double (16 digits)
```

```
>> name='John'; % class char
```

```
>> Temp=101.2; % class double (16 digits)
```

```
>> Check=true; % class logical
```

- Για να «κρύψουμε» το αποτέλεσμα μιας εντολής του MATLAB χρησιμοποιούμε το στοιχείο «;» μετά την εντολή. πχ.,

```
>> x=5; y=2;
```

```
>> f=x^2-y;
```

Back to basics #2

- Γενική μορφή εκφράσεων:

<variable name>=<expression>;

- Μη-έγκυρες εκφράσεις

>> r=a=4;

% not a valid MATLAB statement

>> a+1=press-2;

% not a valid MATLAB statement

>> 4=a;

% not a valid MATLAB statement

>> 'only the lonely'='how I feel';

% not a valid MATLAB statement

- Παραδείγματα μεταβλητών

xinit

OK

VRightInitial

OK

4you2do

Not OK

Start-up

Not OK

Vector%1

Not OK

Target1

OK

target1

OK

X_temp

OK

Back to basics #3

- Χώρος εργασίας μεταβλητών (Variable workspace):

<code>clear a v g</code>	Delete the variables a, v, g from the workspace
<code>clear</code>	Delete all variable from workspace
<code>who</code>	lists the currently defined variables
<code>whos</code>	displays a detailed list of defined variables
<code>save</code>	saves the current workspace to the file called matlab.mat
<code>save test</code>	saves the current workspace to the file called test.mat
<code>load</code>	loads variables saved in matlab.mat into the current workspace
<code>load test</code>	loads variables saved in test.mat into the current workspace
<code>cd</code>	Get the current directory
<code>cd ('dirname')</code>	Go to the directory named “dirname” e.g., dirname='c:\'

Αποθήκευση ανάκτηση δεδομένων #1

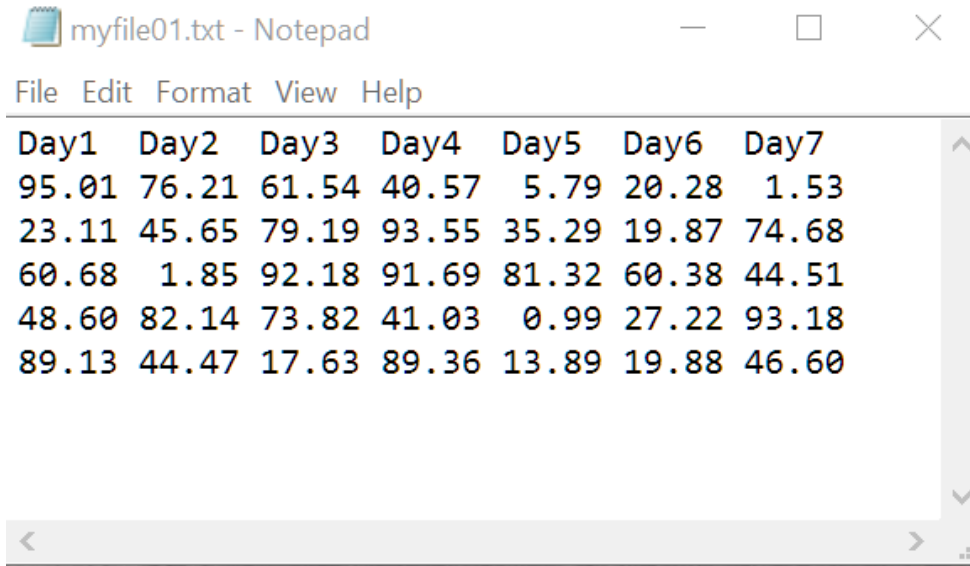
- Ανάκτηση δεδομένων από text file

```
filename = 'myfile01.txt';
```

```
delimiterIn = ' ';
```

```
headerlinesIn = 1;
```

```
A = importdata(filename,delimiterIn,headerlinesIn);
```



myfile01.txt - Notepad

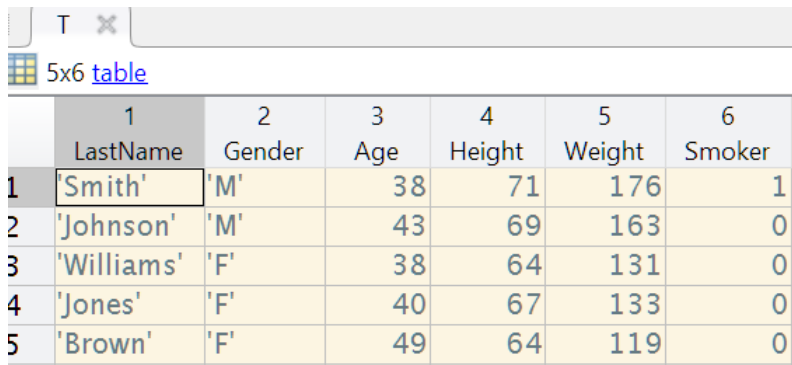
Day1	Day2	Day3	Day4	Day5	Day6	Day7
95.01	76.21	61.54	40.57	5.79	20.28	1.53
23.11	45.65	79.19	93.55	35.29	19.87	74.68
60.68	1.85	92.18	91.69	81.32	60.38	44.51
48.60	82.14	73.82	41.03	0.99	27.22	93.18
89.13	44.47	17.63	89.36	13.89	19.88	46.60

More: <http://www.mathworks.com/help/matlab/ref/importdata.html>

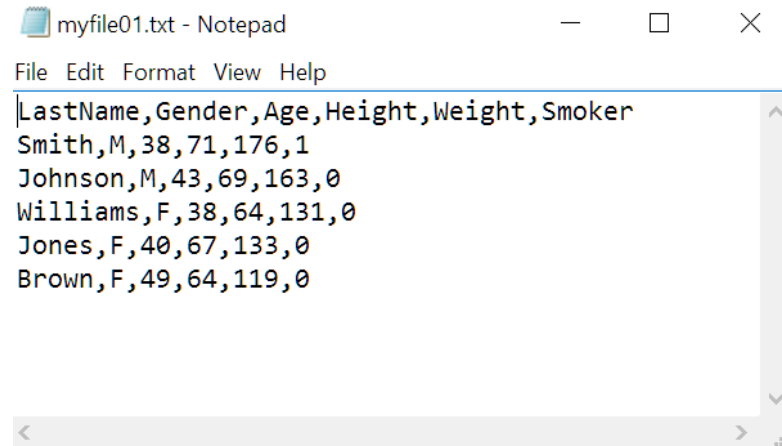
Αποθήκευση ανάκτηση δεδομένων #2

- Ανάκτηση δεδομένων από txt file

```
T = readtable('myfile01.txt')
```



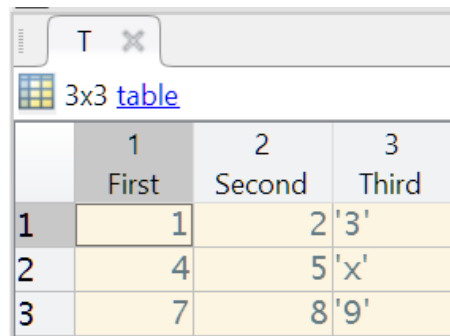
	1	2	3	4	5	6
	LastName	Gender	Age	Height	Weight	Smoker
1	'Smith'	'M'	38	71	176	1
2	'Johnson'	'M'	43	69	163	0
3	'Williams'	'F'	38	64	131	0
4	'Jones'	'F'	40	67	133	0
5	'Brown'	'F'	49	64	119	0



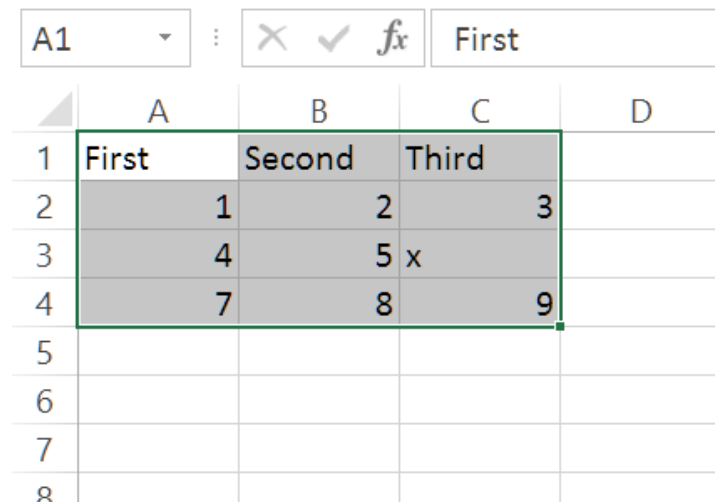
```
myfile01.txt - Notepad
File Edit Format View Help
LastName,Gender,Age,Height,Weight,Smoker
Smith,M,38,71,176,1
Johnson,M,43,69,163,0
Williams,F,38,64,131,0
Jones,F,40,67,133,0
Brown,F,49,64,119,0
```

- Ανάκτηση δεδομένων από Excel file

```
T = readtable('myExample.xlsx','Sheet',1,'Range','A1:C4')
```



	1	2	3
	First	Second	Third
1	1	2	'3'
2	4	5	'x'
3	7	8	'9'



	A	B	C	D
1	First	Second	Third	
2	1	2	3	
3	4	5	x	
4	7	8	9	
5				
6				
7				
8				

More: <http://www.mathworks.com/help/matlab/ref/readtable.html>

Αποθήκευση ανάκτηση δεδομένων #3

- Αποθήκευση δεδομένων σε Excel file

```
values = {1, 2, 3 ; 4, 5, 'x' ; 7, 8, 9};
```

```
headers = {'First','Second','Third'};
```

```
xlswrite('myExample.xlsx',[headers; values]);
```

- Ανάκτηση δεδομένων από Excel file

```
[num,txt,row] = xlsread(filename,sheet,xlRange)
```

```
[A, txt,row] = xlsread(filename,1,'A1:C4')
```

A1	:				First
	A	B	C	D	
1	First	Second	Third		
2	1	2	3		
3	4	5	x		
4	7	8	9		
5					
6					
7					
8					

Matrices (Πίνακες/ Μητρώα) #1

Δημιουργία και αναφορά σε στοιχεία πινάκων #1

Έστω ο πίνακας $D = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 4 & 6 & 8 \\ 3 & 6 & 9 & 12 \\ -5 & -3 & -1 & 0 \end{bmatrix}$

% Εισαγωγή πίνακα στο MATLAB:

```
>> D = [1 2 3 4 ; 2 4 6 8 ; 3 6 9 12 ; -5 -3 -1 0]
```

% Επιλογή του υπο-πίνακα **E** (γραμμές 2 ως 4 και στήλες 1 ως 3)

```
>> E = D(2:4, 1:3)
```

% Επιλογή της 3ης στήλης

```
>> F=D(:,3) ή F=(1:4,3)
```

% Επιλογή της 2ης γραμμής

```
>> G=D(2,:) ή G=D(2,1:4)
```

% Επιλογή του στοιχείου της 2ης γραμμής & 3ής στήλης

```
>> H=D(2,3)
```

Matrices (Πίνακες/ Μητρώα) #2

Δημιουργία και αναφορά σε στοιχεία πινάκων #2

% Εισαγωγή στοιχείου στη 2ης γραμμής & 3ής στήλης

>> D(2,3)=100

% Εισαγωγή στοιχείων σε όλες τις γραμμές της 3ής & 4ης στήλης

>> D(:,3:4)=-1 ή D(:,3:end)=-1

% Διαγραφή της 3ής & 4ης στήλης

>> D(:,3:4)=[] ή D(:,3:end)=[]

Χρήσιμες συναρτήσεις για τη δημιουργία πινάκων #1

eye(3) % Πίνακας 3x3 με μοναδιαία στοιχεία στη διαγώνιο

zeros(3,2) % Πίνακας 3x2 με μηδενικά στοιχεία

ones(2,3) % Πίνακας 2x3 με μοναδιαία στοιχεία

rand(3,4) % Πίνακας 3x4 με τυχαία στοιχεία (uniform distribution)

diag(1:3) % Πίνακας με στοιχεία στη διαγώνιο τα [1,2,3]

diag(D) % Επιλογή των στοιχείων της διαγώνιου του πίνακα D

Matrices (Πίνακες/ Μητρώα) #3

Ερώτηση: Πως θα επιλέγατε τα στοιχεία του πίνακα C που βρίσκονται δεξιά από τη διαγώνιο?

```
>> load hospital
```

```
>> X = [hospital.Weight hospital.BloodPressure hospital.Age];
```

```
% υπολογισμός του μητρώου συσχετίσεων (correlation matrix)
```

```
>> C=corr(X)
```

$$C = \begin{bmatrix} 1 & 0.1558 & 0.2227 & 0.0914 \\ 0.1558 & 1 & 0.5118 & 0.1341 \\ 0.2227 & 0.5118 & 1 & 0.0806 \\ 0.0914 & 0.1341 & 0.0806 & 1 \end{bmatrix}$$

Απάντηση: C1= diag(C(1:end-1,2:end))

Matrices (Πίνακες/ Μητρώα) #4

Χρήσιμοι τελεστές πινάκων #1

$x * 3$	πολλαπλασιασμός κάθε στοιχείου του x με 3
$x + 2$	Πρόσθεση 2 σε κάθε στοιχείο του x
$x + y$	Κατά-στοιχείο πρόσθεση των x και y
$A * y$	πολλαπλασιασμός του πίνακα A και του διανύσματος y
$A * B$	πολλαπλασιασμός του πίνακα A και B
$x * y$	Δεν επιτρέπεται αν τα x και y είναι διανύσματα στήλης
$x .* y$	Κατά-στοιχείο πολλαπλασιασμός των διανυσμάτων x και y
A^3	Κύβος του τετραγωνικού πίνακα A
x^3	Δεν επιτρέπεται αν το x δεν είναι τετραγωνικός πίνακας
$x.^3$	Κατά-στοιχείο ύψωση στον κύβο
$\text{abs}(A)$	Απόλυτη τιμή κατά-στοιχείο
$\text{sqrt}(A)$	Τετραγωνική ρίζα κατά-στοιχείο

Matrices (Πίνακες/ Μητρώα) #5

Έστω:

```
>> rng('default');
```

```
>> rng(1);
```

```
>> A=randn(100,12);
```

Χρήσιμοι τελεστές πινάκων #2

`[Amin,IDmin] = min(A,[],DIM)` % operates along the dimension DIM.

`[Amax,IDmax] = max(A,[],DIM)` % operates along the dimension DIM.

`Am= mean(A,DIM)` % takes the mean along the dimension DIM of X.

`Astd= std(A,FLAG,DIM)` % takes the std. dev. along the dimension

% DIM of X. Pass in FLAG==0 to use the default normalization by N-1, or

% 1 to use N.

Matrices (Πίνακες/ Μητρώα) #6

Έστω:

```
>> rng('default');
```

```
>> rng(1);
```

```
>> B=randn(10,4);
```

Χρήσιμοι τελεστές πινάκων #3

% Επανάληψη του πίνακα B 2 φορές ως προς τον κάθετο άξονα (προς τα δεξιά)

```
>> repmat(B,1,2)
```

% Επανάληψη του πίνακα B 2 φορές ως προς τον οριζόντιο άξονα (προς τα κάτω)

```
>> repmat(B,2,1)
```

% Αναδιάταξη του πίνακα B σε 40 γραμμές και 1 στήλη

```
>> reshape(B,10*4,1) % Μετακινεί τις 4 στήλες
```

```
>> reshape(B', 10*4,1) % Μετακινεί τις 100 γραμμές
```

Matrices (Πίνακες/ Μητρώα) #7

Ερώτηση: Πως θα μετατρέπατε μια μηνιαία χρονοσειρά $Y=[y_1, y_2, \dots, y_n]$ σε πίνακα $T \times 12$ (όπου T : το πλήθος των ετών)?

Έστω:

```
>> rng('default');
```

```
>> rng(1);
```

```
>> Ym=rand(1,12)*50
```

```
>> Ystd=rand(1,12)*5
```

```
>> Y= repmat(Ym,100,1)+randn(100,12).* repmat(Ystd,100,1)
```

```
>> Y(Y<0)=0
```

Απάντηση: $Y_c = \text{reshape}(Y', 100 \times 12, 1)$

Ερώτηση: Πως θα μετατρέπατε μια μηνιαία χρονοσειρά $Y=[y_1, y_2, \dots, y_n]$ σε πίνακα $T \times 12$ (όπου T : το πλήθος των ετών)?

Απάντηση: $Y_{\text{mat}} = \text{reshape}(Y_c, 12, 100)'$

Matrices (Πίνακες/ Μητρώα) #8

Αναζήτηση στοιχείων υπό συνθήκη

```
>> logicMat=Y<10 & Y>4  
>> YlogicMat=Y(logicMat)  
>> YlogicMat=Y(Y<20 & Y>19)  
>> [m,n]=find(Y<10 & Y>8)
```

Ερώτηση: Πως θα εντοπίζατε τιμές μεγαλύτερες από ένα κατώφλι (πχ $Y_n > 48$)

Απάντηση: `Ylarger48=Y(Y>48)`

Ερώτηση: Πως θα εντοπίζατε τη «θέση» των τιμών?

Απάντηση: `[Y48t,Y48c]=find(Y>48)`

Matrices (Πίνακες/ Μητρώα) #9

Επίλυση γραμμικού συστήματος

% Create a 3x3 matrix

```
>> A= [3 5 -1 ; 0 4 2 ; -2 1 5]
```

% Create a 3x1 matrix (column vector)

```
>> b = [2 ; 1 ; -4]
```

% Solve the system #1

% notice the backslash operator “\” (which performs Gaussian elimination to
% solve the system)

```
>> X=A\b
```

% Solve the system #2

```
>> x = inv(A)*b
```

% Solve the system #3

```
>> X=linsolve(A,b)
```

% What linsolve does?

```
>> help linsolve
```

Έστω το σύστημα:
$$\begin{bmatrix} 3 & 5 & 1 \\ 0 & 4 & 2 \\ -2 & 1 & 5 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = \begin{Bmatrix} 2 \\ 1 \\ -4 \end{Bmatrix}$$

Γραφήματα #1

Απλά γραφήματα

```
x = linspace(-2*pi,2*pi,100);
```

```
y1 = sin(x);
```

```
y2 = cos(x);
```

```
figure(1)
```

```
plot(x,y1, 'r',x,y2, 'b--o')
```

```
title('2-D Line Plot')
```

```
xlabel('x')
```

```
ylabel('f(x)')
```

```
figure(2)
```

```
plot(x,y1, 'r', 'LineWidth',3)
```

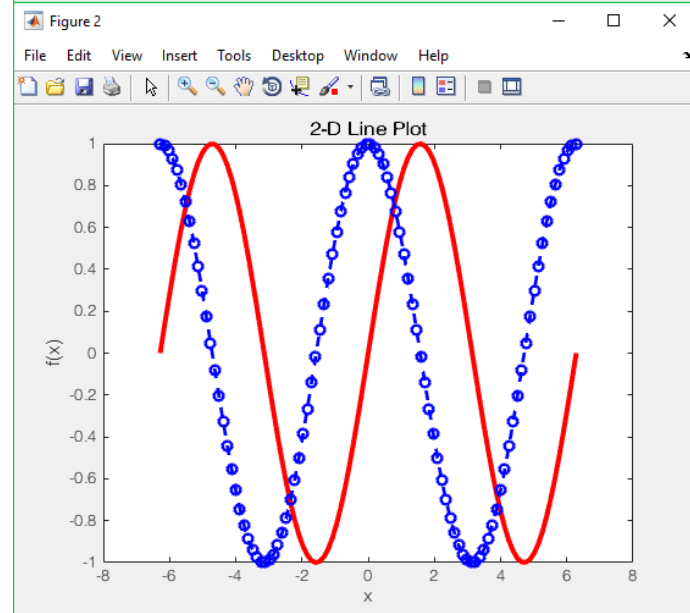
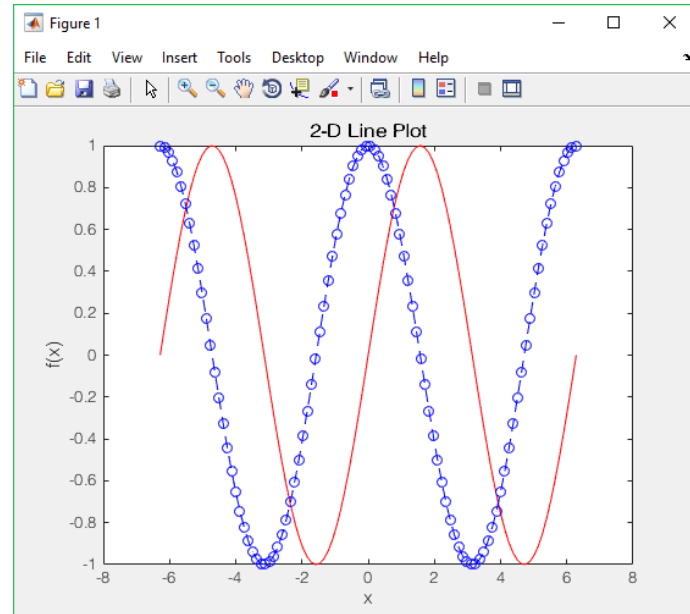
```
hold on;
```

```
plot(x,y2, 'b--o', 'LineWidth',2)
```

```
title('2-D Line Plot')
```

```
xlabel('x')
```

```
ylabel('f(x)')
```



Γραφήματα #2

Ιστογράμματα

```
>> hist(randn(1000,1),50)
```

Γραφήματα X-Y

```
>> scatter(cos(x),sin(x))
```

Συνοπτική παράσταση δείγματος (box plot)

```
>> boxplot(randn(1000,1))
```

Γρήγορα γραφήματα

```
>> ezplot('x^2-5')
```

3D γραφήματα

% Επιστρέφει δύο μητρώα. Το πρώτο περιέχει τις τετμημένες και το δεύτερο τις
% τεταγμένες των κυττάρων ενός καννάβου.

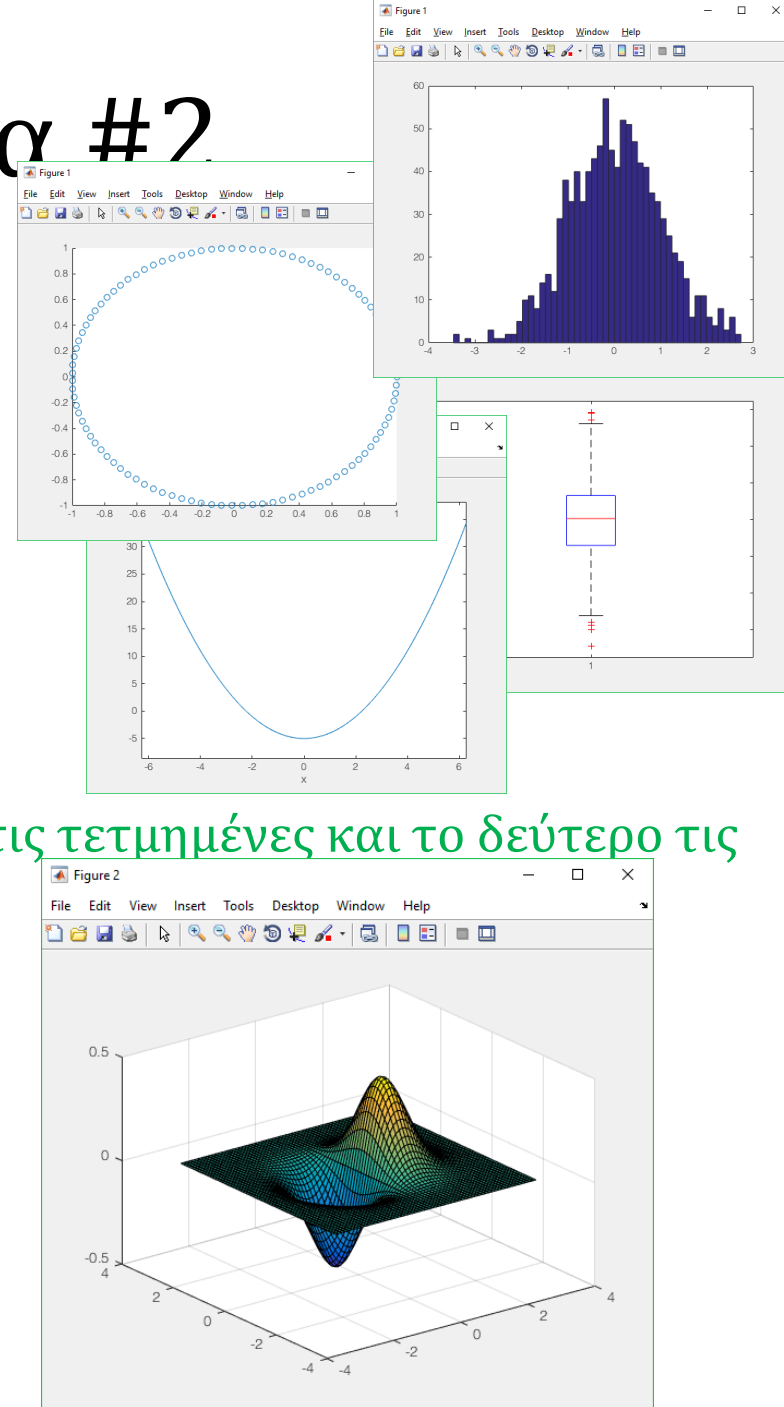
```
>> [X Y]=meshgrid(-3:3)
```

```
>> [X Y]=meshgrid(-3:0.1:3);
```

```
>> Z=X.*exp(-X.^2-Y.^2);
```

```
>> figure(1)
```

```
>> mesh(X,Y,Z)
```



Προγραμματισμός #1

Scripts

Χρησιμοποιούν την **ίδια** περιοχή μνήμης (Variable workspace or Base workspace) με τη γραμμή εντολών (command line).
Περιορίζονται σε συγκεκριμένες εφαρμογές.

Vs

Functions

Χρησιμοποιεί την **δική τους αποκλειστική** περιοχή μνήμης (Variable workspace).
Ανταλλάσσουν δεδομένα με άλλες συναρτήσεις ή τον χρήστη μέσω των ορισμάτων τους.
Ωστόσο μπορεί να αλληλοεπιδράσουν με το base workspace με τη βοήθεια **global** μεταβλητών.

More:

http://www.mathworks.com/help/matlab/matlab_prog/base-and-function-workspaces.html

http://www.mathworks.com/help/matlab/matlab_prog/share-data-between-workspaces.html

Προγραμματισμός #2

Scripts

Πληκτρολογήστε:

```
>> edit test1.m
```

Ονομάστε το “ex1.m” εισάγετε:

```
cost = 50;
```

```
profit = 10;
```

```
sale_price = cost + profit
```

Σώστε με το εικονίδιο ή πατώντας **ctr+s**

Τώρα μπορείτε να το καλέσετε από την
command line

```
>> test1
```

Functions

Ομοίως δημιουργείτε τις συναρτήσεις:

```
function [A]=area(r)
```

```
% This is a function
```

```
A= pi*r^2;
```

```
end
```

```
function [P]=perimeter(a,b)
```

```
% This is another example of a function
```

```
P= 2*(a+b);
```

```
end
```

Πληκτρολογήστε :

```
>> A10=area(10)
```

```
>> A5=area(5)
```

```
>> P23=perimeter(2,3)
```

```
>> P37=perimeter(3,7)
```

Προγραμματισμός #3

Έλεγχος ροής με χρήση if statement

if expression

...

elseif expression2

...

else

...

end

Παράδειγμα

function cost2(boxes)

% This is an example of an If Elseif construct

if boxes < 7

cost = boxes*2.5

elseif boxes < 9

cost = boxes*2

elseif boxes > 9

cost = boxes*1.5

end

Προγραμματισμός #4

Έλεγχος ροής με χρήση while statement

While *expression*

...

...

else

...

end

Παράδειγμα

% This is an example of a While loop

tol = 0.0;

n = 0;

while tol < 10

n = n + 1;

tol = tol + 2;

end

Προγραμματισμός #5

Έλεγχος ροής με χρήση switch-case statement

switch *expression*

case expression1

 ...

 otherwise

 ...

end

Παράδειγμα

function cost3(boxes)

% This is an example of the Switch Case construct

switch boxes

case 6

 cost = boxes* 2.5

case 10

 cost = boxes*1.5

otherwise

 cost = 0

end

Προγραμματισμός #6

Έλεγχος ροής με χρήση FOR loop

```
for iterator = vector
```

```
...
```

```
End
```

Παράδειγμα

```
%This is an example of another FOR loop
```

```
for n = 1:3
```

```
    for m = 1:3
```

```
        y(m,n) = m^2 + m*n + n^2;
```

```
    end
```

```
end
```

Δομές δεδομένων

Ορισμός

Οι δομές (structures) είναι ένας τύπος δεδομένων που επιτρέπει την αποθήκευση οργανωμένης πληροφορίας που συντίθεται από διαφορετικούς απλούς τύπους δεδομένων (αριθμοί, αλφαριθμητικά, μητρώα,...).

Μητρώο δομών

```
>> patient(2).name = 'Ann Lane';  
>> patient(2).billing = 28.50;  
>> patient(2).test = [68 70 68; 118 118 119; 172 170 169];
```

Όλα σε μία γραμμή

```
>> simplestrct(1)=struct('FirstName', 'Nikos', 'LastName',  
'Foufotos', 'Phone', 2107722841, 'Vector', [1 2 3])  
simplestrct(2)=struct('FirstName', 'John', 'LastName',  
'Doe', 'Phone', 2107722828, 'Vector', [3 2 1])  
  
>> simplestrct(1).FirstName  
>> simplestrct(2).Phone
```

Μέθοδοι βελτιστοποίησης

Βελτιστοποίηση πολυμεταβλητής συνάρτησης με περιορισμούς

$$\min_x f(x) \text{ such that } \begin{cases} c(x) \leq 0 \\ ceq(x) = 0 \\ A \cdot x \leq b \\ Aeq \cdot x = beq \\ lb \leq x \leq ub, \end{cases}$$

[x,fval] = fmincon(fun,x0,A,b,Aeq,beq,lb,ub,nonlcon,options)

```
fun = @(x)100*(x(2)-x(1)^2)^2 + (1-x(1))^2;
```

```
x0 = [0.5,0];
```

```
A = [1,2]; b = 1;
```

```
Aeq = [2,1]; beq = 1;
```

```
lb=[0 0]; ub=[1 1];
```

```
[x,fval] = fmincon(fun,x0,A,b,Aeq,beq,lb,ub)
```

```
nvars=2;
```

```
[x,fval] = ga(fun,nvars,A,b,Aeq,beq,lb,ub)
```

Απλοποιημένη προσομοίωση ταμιευτήρα #1

```
function [f,F,TabF]=simRes(k,d,s0,Inflow)
[n,~]=size(Inflow);

st1_tmp=NaN(n,1); Rt=NaN(n,1);
st2_tmp=NaN(n,1); Yt=NaN(n,1);
st3_tmp=NaN(n,1); Def=NaN(n,1);

st_1=s0;
for i=1:n
    st1_tmp(i,1)=Inflow(i,1)+st_1;
    Rt(i,1)=min(st1_tmp(i,1),d);
    st2_tmp(i,1)=st1_tmp(i,1)-Rt(i,1);
    Yt(i,1)=max(0,st2_tmp(i,1)-k);
    st3_tmp(i,1)=st2_tmp(i,1)-Yt(i,1);
    Def(i,1)=d-Rt(i,1);
    st_1=st3_tmp(i,1);
end

f=sum(Def);

F.S1t=st1_tmp; F.Rt=Rt;
F.S2t=st2_tmp; F.Yt=Yt;
F.S3t=st3_tmp; F.Def=Def;

TabF=table(st1_tmp,Rt,st2_tmp,Yt,st3_tmp,Def,'VariableNames',...
    {'S1t' 'Rt' 'S2t' 'Yt' 'S3t' 'Def'});
end
```

Απλοποιημένη προσομοίωση ταμιευτήρα #2

Γράψτε:

Inflow= [6.7;12.4;25.2;21.7;23.1;26;22.1;13.1;4.4;2.9;2.1;1.5];

Και εκτελέστε:

```
[f,F,tabF]=simRes(30,10,15,Inflow);
```

tabF =

S1t	Rt	S2t	Yt	S3t	Def
21.7	10	11.7	0	11.7	0
24.1	10	14.1	0	14.1	0
39.3	10	29.3	0	29.3	0
51	10	41	11	30	0
53.1	10	43.1	13.1	30	0
56	10	46	16	30	0
52.1	10	42.1	12.1	30	0
43.1	10	33.1	3.1	30	0
34.4	10	24.4	0	24.4	0
27.3	10	17.3	0	17.3	0
19.4	10	9.4	0	9.4	0
10.9	10	0.9	0	0.9	0

Workspace	
Name ^	Value
f	0
F	1x1 struct
Inflow	12x1 double
tabF	12x6 table

Βιβλιογραφία

Βιβλία

- Lent, C. S., (2013). Learning to Program with MATLAB: Building GUI Tools. Publisher: Wiley. ISBN 10: 0470936444, ISBN 13: 9780470936443.
- Kattan, P. I., (2008). MATLAB For Beginners: A Gentle Approach. Publisher: CreateSpace Independent Publishing Platform. ISBN-10: 1438203098, ISBN-13: 978-1438203096.

Σημειώσεις

- Ρόζος Ε., Μακρόπουλος Χ. (2011). Προγραμματισμός σε Matlab για προβλήματα βελτιστοποίησης. Σημειώσεις στα πλαίσια του μαθήματος: Βελτιστοποίηση συστημάτων υδατικών πόρων –Υδροπληροφορική. ΕΜΠ.

Cheat-Sheets

- A Matlab Cheat-sheet (MIT 18.06, Fall 2007), <http://web.mit.edu/18.06/www/Spring09/matlab-cheatsheet.pdf>
- Giordano Fusco, MATLAB Reference Sheet, CS 412 UW-Madison, http://userpage.chemie.fu-berlin.de/~naundorf/PCF/MATLAB_Reference_Sheet-1.pdf
- Thor Nielsen, Matlab Cheat Sheet, <http://www.econ.ku.dk/pajhede/Cheatsheet.pdf>
- Jialong He, MATLAB Quick Reference, http://www.cs.cmu.edu/~tom/10601_fall2012/recitations/matlab_quickref.pdf

Προγραμματισμός #7

Πειράματα με ταχύτητα

```
>> edit slow.m
```

Εισάγετε

```
function [ret] = slow(x)
```

```
for ii=1:length(x)
```

```
    s(ii)=sqrt(x(ii));
```

```
end;
```

```
ret=sum(s);
```

Σώστε και κλείστε τον επεξεργαστή

```
>> x=-10000:0.5:10000;
```

```
>> tic; slow(x); toc;
```

```
>> edit slow.m
```

Εισάγετε στη δεύτερη γραμμή του κειμένου το ακόλουθο

```
s=zeros(1,length(x));
```

Σώστε και κλείστε τον επεξεργαστή

```
>> tic; slow(x); toc;
```

Συμβουλές για ταχύτητα

1. Εκχώρηση μνήμης εκ των προτέρων
2. Χρήση διανυσματικών πράξεων (αποφυγή for-loop)
3. Χρήση κατάλληλου τελεστή (tic-toc)