

Ακαδημαϊκό Έτος : 2018-2019 (Εαρινό Εξάμηνο)

ΨΗΦΙΑΚΗ ΕΠΕΞΕΡΓΑΣΙΑ ΕΙΚΟΝΑΣ

ΕΙΣΑΓΩΓΙΚΗ ΑΣΚΗΣΗ – Εισαγωγή σε MATLAB

1. Στο “Current Directory” επιλέξτε το directory που θα σώζετε τα αρχεία σας. Με την εντολή pwd (στη command line) μπορείτε να δείτε το directory στο οποίο βρίσκεστε. Με την εντολή cd μπορείτε να αλλάξετε directory.
2. Σε command line γράψτε `>> help diary`
Χρησιμοποιήστε κατάλληλα την εντολή diary από εδώ και στο εξής (και σε κάθε εργαστήριο).
3. Σε command line γράψτε `>> lookfor size` `>> type mean`
Τι χρησιμεύουν οι παραπάνω εντολές;
4. Ανάθεση σε μεταβλητές (υπάρχει διάκριση ανάμεσα σε μικρά και κεφαλαία γράμματα)
Σε command line γράψτε: `>> a=2` `>> A=5` `>> b=3*A-5*a;` `>> c=5-2*i`
`>> whos`
5. Πράξεις αριθμών: + - * / ^ sqrt (δείτε τη με την εντολή help)
Υπολογίστε τα: `a+b` `a-c` `b*c` `A/a` `A^3` `sqrt(a)` `((a+b)*A)^2`
6. `>> help abs`
Χρησιμοποιήστε την εντολή abs για τα: `A^3` `c` `a-c`
7. `>> help clear` Χρησιμοποιήστε την εντολή clear
8. vectors – διαφορετικοί τρόποι αρχικοποίησης
`>> A=[1 2 3 4 5]` `>> B=[1; 2; 3; 4; 5]` `>> B'`
`>> C=[1:5]` `>> D=[1:1:5]` `>> E=1:5` `>> F=1:1:5`
`>> G=[11:20]` `>> H=[1:2:10]`
9. Δείτε τι κάνουν οι εντολές size και length. Να χρησιμοποιηθούν για τα A, B, B', G και H του ερωτήματος 8.
10. Πρόσβαση σε στοιχεία των vectors: `A(3)` `B(4)` `H(2)` `G(:)` `G(2:5)`
`d = [G(1:2) A(3:4) H(1) G(5:8)]`.
Χρησιμοποιήστε τη size για το d.
11. Πράξεις ανάμεσα σε vectors: + - * .* ./ .^ sqrt
Έστω τα vectors: `A=[1:3]` `B=[4:6]` `C=[7:10]`

Υπολογίστε τα:

A+B	A-B	B-A	A*B'
B*A'	A'*C	B'*C	C'*A
C'*B	A.*B	A./B	B.^A
A.^B	A.^2	B.^2	sqrt(C)

Συγκρίνετε τα A*B' και B*A' και σχολιάστε και τα υπόλοιπα αποτελέσματα.

12. Πίνακες. Ορίστε τους ακόλουθους πίνακες αφού πρώτα χρησιμοποιήσετε την εντολή clear. Στη συνέχεια χρησιμοποιείστε την εντολή size για αυτούς.

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \quad B = \begin{bmatrix} 10 & 11 & 12 \\ 13 & 14 & 15 \end{bmatrix} \quad C = \begin{bmatrix} 2 & 1 & 3 \\ 8 & 4 & 5 \\ 6 & 9 & 7 \end{bmatrix}$$

13. Πρόσβαση σε στοιχεία πινάκων:

A(3,1) A(2,2) A(2,:) A(:,3) B(1,:)
 B(:,2) C(:, :) C(:) B(:) A(1:2, 2:3)
 D = [A(1:2, 2:3); B(1:2, 1:2); C(2:3, 2:3)]

14. Πράξεις πινάκων

+	->	A+C	C+A				
-	->	A-C	C-A				
*	->	A*C	C*A	B*A	B*C	A*B'	C*B'
/	->	A/C => A*inv(C) C/A => C*inv(A)				Πότε ορίζεται αντίστροφος;	
^	->	A^2 (=A*A)		C^3 (=C*C*C)		Ορίζεται για τετραγωνικό πίνακα.	
.*	->	A.*C					
./	->	A./C		C./A			
.^	->	A.^2	A.^A	C.^A			

15. Ειδικές μορφές πινάκων

help για zeros, ones, eye, diag
 zeros(size(A)) ones(size(B)) eye(5)diag([1:5])
 Χρησιμοποιώντας άθροισμα κατάλληλων διαγωνίων πινάκων φτιάξτε τον παρακάτω πίνακα:

$$K = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 1 & 2 & 3 \\ 6 & 5 & 1 & 2 \\ 7 & 6 & 5 & 1 \end{bmatrix}$$

16. Συναρτήσεις min, max, mean, mean2.

help για κάθε μία από αυτές
 min(A) min(A(2,:)) min(A(:)) max(C(:))
 mean(C) mean(C(:)) mean2(C) max(C(:,3))

17. Αποθήκευση μεταβλητών -> .mat αρχεία

help save help load
 Αποθηκεύστε σε ένα αρχείο με όνομα pinakes τις μεταβλητές A, B και C.
 clear
 Φορτώστε το αρχείο pinakes. >> whos

18. help για: figure stem plot close hold on subplot

>> t=[1:10]

```
>> F=t.^2
>> figure;stem(t,F)
>> figure;plot(t,F)
>> figure;plot(t,F,'*')
>> figure;plot(t,F,'*r')
```

Περαιτέρω επεξεργασία: Edit -> Axes Properties, κλικ στην καμπύλη και επεξεργασία
Αποθήκευση: File -> Save, File -> Save as

19. Να σχεδιαστούν οι ακόλουθες συναρτήσεις, αφού παραχθούν διανύσματα τιμών για τη μεταβλητή t και την εκάστοτε $y(t)$. Για την παραγωγή του διανύσματος y στο b) ερώτημα, μπορείτε να χρησιμοποιήσετε τη συνάρτηση `find`.

Οι γραφικές παραστάσεις θα βρίσκονται σε ένα figure, η μία κάτω από την άλλη, χρησιμοποιώντας κατάλληλα τη `subplot`. Δώστε κατάλληλα ονόματα στους άξονες.

a) $y(t) = \frac{1}{t^2}$

b) $y(t) = \begin{cases} t, & 0 \leq t \leq 3 \\ 6-t, & 3 < t \leq 6 \\ 0, & \text{αλλού} \end{cases}$

20. 3D plots – απεικονίζουν μια επιφάνεια που ορίζεται από μία συνάρτηση δύο μεταβλητών, $z=f(x,y)$. Για να πάρουμε τις τιμές της z , πρέπει καταρχήν να δημιουργήσουμε ένα σύνολο από σημεία (x,y) στο πεδίο ορισμού της συνάρτησης. Αυτό επιτυγχάνεται με τη χρήση της `meshgrid`.

```
>> help meshgrid >> help mesh >> help surf >> help contour
```

π.χ.

```
>> [X,Y] = meshgrid(-2:.2:2);
>> Z = X .* exp(-X.^2 - Y.^2);
```

Στη συνέχεια σχεδιάζουμε την επιφάνεια:

```
>> mesh(X,Y,Z) >> figure; surf(X,Y,Z) >> figure; contour(X,Y,Z)
```

function -> .m files

Καλούμε συναρτήσεις στο command window μόνο με το όνομά τους (μεταβλητές εισόδου και μεταβλητές εξόδου) – χωρίς την προέκταση .m

Δομή συνάρτησης

```
function [out1, out2, out3, ...] = function_name(in1, in2, in3, ...)
```

```
%Comments
```

```
...
Code
...
```

Δεσμεύουμε μνήμη, ορίζοντας π.χ. έναν πίνακα αποτελεσμάτων στην αρχή του κώδικά μας. Διαφορετικά, κάθε φορά που αυξάνεται ο αριθμός των αποτελεσμάτων, δημιουργείται ένας νέος πίνακας, μεγαλύτερων διαστάσεων. Η δημιουργία πίνακα σε κάθε επανάληψη, δημιουργεί καθυστέρηση στην εκτέλεση του κώδικά μας.

21. Δημιουργήστε στον editor της Matlab την ακόλουθη συνάρτηση και σώστε τη.

```
function C=test1(n)

%C is the product of two vectors' elements.
%n, which is given as input, is the vectors' length.

A=randn(1,n); %A is a vector of n random numbers.
B=randn(1,n); %B is a vector of n random numbers.

for i=1:n
    C(i)=A(i)*B(i);
end
```

Θέλουμε να μετρήσουμε το χρόνο εκτέλεσης της παραπάνω συνάρτησης. help tic, help toc. Καλέστε τη συνάρτηση για $n=50000000$ και μετρήστε το χρόνο εκτέλεσής της, χρησιμοποιώντας τις εντολές tic, toc.

22. Δημιουργήστε μια παραλλαγή της παραπάνω συνάρτησης, η οποία θα έχει όνομα test2, και στην οποία πρέπει να δεσμεύετε μνήμη για το vector C. Αφού σώσετε τη συνάρτηση, καλέστε τη όπως και στο προηγούμενο ερώτημα, για $n=50000000$, και μετρήστε το χρόνο εκτέλεσής της. Τι παρατηρείτε;

Όταν μπορούμε, προσπαθούμε να αποφεύγουμε τη χρήση της εντολής for. Έτσι, αποφεύγουμε τις επαναλήψεις, οι οποίες δημιουργούν καθυστέρηση στην εκτέλεση του κώδικά μας.

23. Δημιουργήστε μια παραλλαγή της συνάρτησης test2, η οποία θα έχει όνομα test3. Στη νέα αυτή συνάρτηση, υπολογίστε το γινόμενο των στοιχείων των διανυσμάτων A και B, χωρίς να χρησιμοποιήσετε την εντολή for. Αφού σώσετε τη συνάρτηση, καλέστε τη για $n=50000000$, και μετρήστε το χρόνο εκτέλεσής της. Τι παρατηρείτε;

24. help rand, randn, find. Δημιουργήστε ένα πίνακα με τυχαίες τιμές που ακολουθούν ομοιόμορφη κατανομή, διαστάσεων 100×100 . Μετρήστε πόσα στοιχεία του πίνακα έχουν τιμή μεγαλύτερη από 0.7.

25. help double, uint8, imshow. Χρησιμοποιείτε την imshow για την απεικόνιση της lenna, αφού φορτώσετε το .mat αρχείο. Απεικονίστε την roses.jpg.

26. help imread, imwrite. Απεικονίστε την εικόνα roses.jpg, αφού τη διαβάσετε χρησιμοποιώντας την imread.

- Μηδενίστε τη Red συνιστώσα της εικόνας, απεικονίστε το αποτέλεσμα και σώστε το σε ένα νέο .jpg αρχείο, χρησιμοποιώντας τη συνάρτηση imwrite.
- Μηδενίστε τη Blue συνιστώσα της εικόνας, απεικονίστε το αποτέλεσμα και σώστε το σε ένα νέο .jpg αρχείο, χρησιμοποιώντας τη συνάρτηση imwrite.