

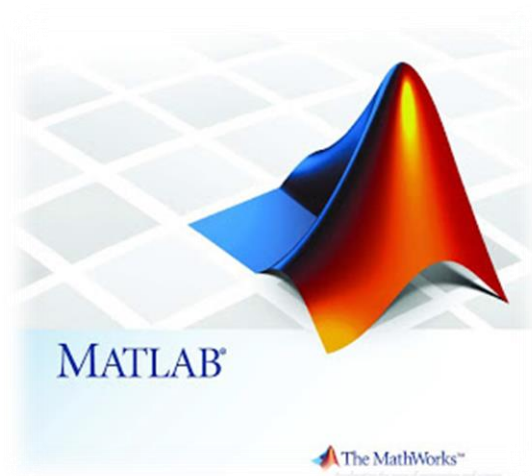


Τμήμα Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών

ΣΗΜΑΤΑ ΚΑΙ ΣΥΣΤΗΜΑΤΑ

ΕΡΓΑΣΤΗΡΙΑΚΟΣ ΟΔΗΓΟΣ

Εργαστηριακή Άσκηση 0 «Εισαγωγή στο Matlab»



Μιχάλης Παρασκευάς
Καθηγητής

Οκτώβριος 2023

Έκδοση: 4.0

Πίνακας Περιεχομένων

1. Στόχος Εργαστηριακής Άσκησης	3
2. Εισαγωγή	3
3. Το περιβάλλον εργασίας του Matlab	3
4. Εισαγωγή στη γλώσσα Matlab	4
4.1 Απλές πράξεις με Matlab	5
4.2 Εκχώρηση τιμών σε μεταβλητές	5
4.3 Εντολές who και whos	6
4.4 Ενσωματωμένες συναρτήσεις και σταθερές Matlab	6
4.5 Μορφή μεταβλητών	10
4.6 Διανύσματα	10
4.7 Πράξεις με διανύσματα	13
4.8 Διανύσματα Στήλης και Πίνακες	14
4.9 Πολλαπλασιασμός Πινάκων	17
4.10 Χρήσιμες συναρτήσεις για τη δημιουργία πινάκων	18
5. Η εργαλειοθήκη Symbolic	19
5.1 Συμβολικές μεταβλητές	19
5.2 Παραγωγή συμβολικών συναρτήσεων και επίλυση εξισώσεων	19
5.3 Ολοκλήρωση Συμβολικών Συναρτήσεων	20
5.4 Συμβολικό άθροισμα σειράς	21
6. Σχεδιασμός Γραφικών Παραστάσεων	22
6.1 Χρήσιμες συναρτήσεις για την μορφοποίηση των γραφημάτων	25
7. Άλυτες Ασκήσεις	26

Άσκηση 1: Εισαγωγή στο MATLAB

1. Στόχος Εργαστηριακής Άσκησης

Ο στόχος της πρώτης εργαστηριακής άσκησης είναι η εξοικείωση των φοιτητών με το περιβάλλον εργασίας του MATLAB.

2. Εισαγωγή

Το Matlab αποτελεί μια υψηλού επιπέδου γλώσσα και ένα διαδραστικό περιβάλλον προγραμματισμού για την υλοποίηση αριθμητικών υπολογισμών και γραφικών απεικονίσεων. Χρησιμοποιώντας το Matlab μπορούμε εύκολα να αναλύσουμε δεδομένα, να αναπτύξουμε αλγορίθμους, να δημιουργήσουμε μοντέλα και εφαρμογές.

Το απλό συντακτικό της γλώσσας του Matlab, τα εργαλεία της διεπαφής χρήστη και οι πολυάριθμες ενσωματωμένες μαθηματικές συναρτήσεις μας παρέχουν τη δυνατότητα να προσεγγίσουμε πολλαπλές πτυχές ενός προβλήματος και να φτάσουμε σε μία λύση πολύ πιο εύκολα και γρηγορότερα από ότι με συμβατικά εργαλεία (π.χ. τα υπολογιστικά φύλλα Microsoft Excel) ή άλλες γλώσσες προγραμματισμού (π.χ. C/C++, JAVA, κλπ).

Το Matlab μπορεί να χρησιμοποιηθεί για μια πληθώρα προβλημάτων στα γνωστικά αντικείμενα: ανάλυση και επεξεργασία σήματος, αναλογικών και ψηφιακών επικοινωνιών, επεξεργασία εικόνας και ήχου, συστημάτων ελέγχου, στατιστικής ανάλυσης, οικονομικών εφαρμογών, υπολογιστικής βιολογίας, μαθηματικών, μηχανικής, κλπ.

Διαθέτει πλούσια και ισχυρά λειτουργικά χαρακτηριστικά, όπως:

- Υψηλού επιπέδου γλώσσα για αριθμητικούς υπολογισμούς, γραφική αναπαράσταση συναρτήσεων και ανάπτυξη εφαρμογών.
- Διαδραστικό περιβάλλον για ανάλυση, σχεδιασμό και επίλυση προβλημάτων.
- Ενσωματωμένες μαθηματικές συναρτήσεις για: γραμμική άλγεβρα, στατιστική, ανάλυση Fourier, φίλτρα, βελτιστοποίηση, αριθμητική ανάλυση, επίλυση συνήθων διαφορικών εξισώσεων και πολλά άλλα.
- Ενσωματωμένες δυνατότητες γραφικών για απεικόνιση δεδομένων και δημιουργία μεγάλης ποικιλίας γραφημάτων.
- Εργαλεία δημιουργίας διαδραστικών εφαρμογών.

Στον συγκεκριμένο εργαστηριακό οδηγό θα χρησιμοποιήσουμε το Matlab για την ανάλυση και επεξεργασία Σημάτων Συνεχούς και Διακριτού Χρόνου.

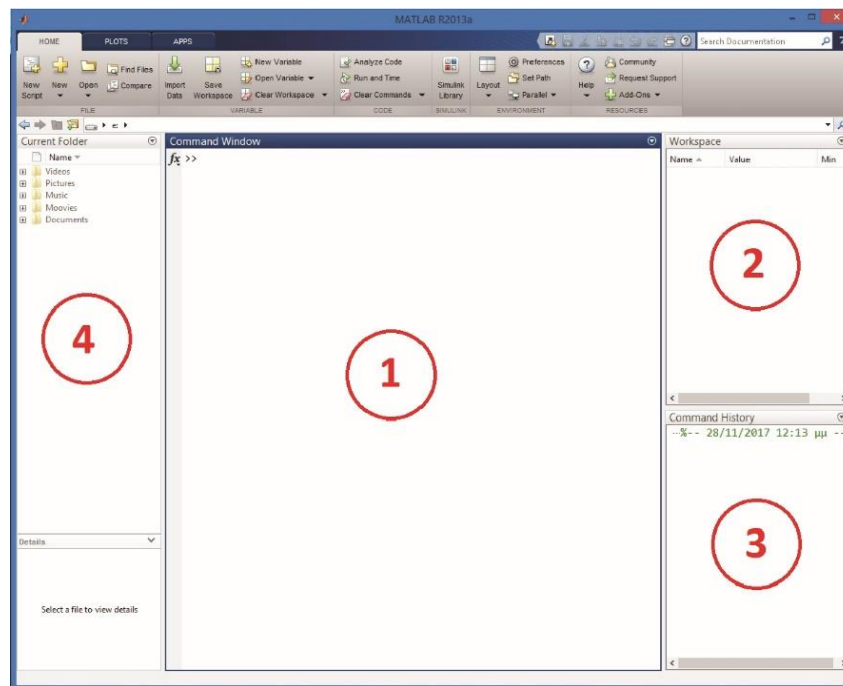
3. Το περιβάλλον εργασίας του Matlab

Ο παρών εργαστηριακός οδηγός αναφέρεται στην έκδοση Matlab R2013. Εκτός από διάφορες εικαστικές βελτιώσεις και νέες εργαλειοθήκες που έχουν προστεθεί στην τελευταία έκδοση, η βασική δομή του περιβάλλοντος εργασίας παραμένει η ίδια με τις προηγούμενες εκδόσεις.

Η βασική δομή του περιβάλλοντος εργασίας (διεπαφή) αποτελείται από 4 επιμέρους

τμήματα (Εικόνα 1):

1. **Τη γραμμή εντολών (Command Window):** Η γραμμή εντολών είναι ο χώρος στον οποίο ο χρήστης αλληλεπιδρά με το σύστημα, δίνοντας εντολές και λαμβάνοντας το αποτέλεσμα.
2. **Τον χώρο εργασίας (Workspace):** Εδώ εμφανίζονται όλες οι μεταβλητές που έχουμε δημιουργήσει και τα δεδομένα που περιέχουν κατά την τρέχουσα στιγμή.



Σχήμα 1 – Το γραφικό περιβάλλον εργασίας του Matlab

3. **Το ιστορικό εντολών (Command History):** Όπως δηλώνει το όνομα του, αυτό το τμήμα διατηρεί ιστορικό των εντολών που έχουμε δώσει.
4. **Τον τρέχοντα φάκελο (Current Folder):** Ένα σημαντικό τμήμα του περιβάλλοντος είναι ο τρέχοντας φάκελος. Και είναι σημαντικό διότι το Matlab έχει πρόσβαση στα αρχεία που βρίσκονται σε αυτόν τον φάκελο. Για να γίνει αυτό κατανοητό ας θεωρήσουμε την περίπτωση που θέλουμε στο Matlab να αναπαράγουμε ένα αρχείο ήχου ή να καλέσουμε μια συνάρτηση (σύνολο από εντολές), την οποία έχουμε αποθηκεύσει σε ένα εξωτερικό αρχείο. Για να μπορέσει το Matlab να προσπελάσει το αρχείο ή την συνάρτηση, τότε θα πρέπει αυτά να βρίσκονται στον τρέχοντα φάκελο.

4. Εισαγωγή στη γλώσσα Matlab

Ο γλώσσα προγραμματισμού του Matlab είναι κατά βάση μια γλώσσα χειρισμού βαθμωτών μεγεθών (scalars), διανυσμάτων (vectors) και πινάκων (Matrices). Άλλωστε από τον συνδυασμό των λέξεων **MAT**rix **LAB**oratory προκύπτει και το όνομά του. Το Matlab διαθέτει πολύ απλό συντακτικό για τη δημιουργία τέτοιων μεγεθών, χωρίς να χρειάζεται να ορίσουμε

μεταβλητές, να συμπεριλάβουμε βιβλιοθήκες ή να καλέσουμε κάποια `main()` συνάρτηση.

4.1 Απλές πράξεις με Matlab

Στο Matlab μπορούμε να εκτελέσουμε απλές αριθμητικές πράξεις με τους παρακάτω τελεστές:

Τελεστές	Αλγεβρική Πράξη
+	Πρόσθεση
-	Αφαίρεση
*	Πολλαπλασιασμός
/	Διαίρεση
^	Ύψωση σε δύναμη
'	Αντιστροφή
()	Παρενθέσεις
=	Εκχώρηση τιμής σε μεταβλητή

Παράδειγμα 1 – Στοιχειώδεις πράξεις στο Matlab

Εκτελέστε στο Matlab τις ακόλουθες πράξεις:

Εντολές Matlab	Αποτέλεσμα
1+2	ans = 3
2*3	ans = 6
1+2*3	ans = 7
$((1+2)*3)-2)/2$	ans = 3.5000
10^2	ans = 100

Σημείωση: Η μεταβλητή **ans** (=απάντηση) δέχεται πάντα και εμφανίζει στην οθόνη το αποτέλεσμα της τελευταίας δοθείσας εντολής, στην περίπτωση που αυτή δεν εκχωρεί το αποτέλεσμα σε κάποια άλλη μεταβλητή.

4.2 Εκχώρηση τιμών σε μεταβλητές

Στο προηγούμενο παράδειγμα δίναμε αριθμητικές παραστάσεις τις οποίες υπολόγιζε το Matlab και εκχωρούσε την τιμή τους στην προκαθορισμένη μεταβλητή **"ans"**. Στο Matlab μπορούμε πολύ εύκολα να δημιουργήσουμε μια μεταβλητή και να της **εκχωρήσουμε** μια αριθμητική τιμή ή την τιμή που προκύπτει από τον υπολογισμό μίας αλγεβρικής παράστασης. Αυτό γίνεται χρησιμοποιώντας τον τελεστή **"="**.

Παράδειγμα 2

Να δημιουργήσετε τις μεταβλητές x , y , A , και να εκχωρήσετε σε κάθε μία τις παρακάτω τιμές, αντιστοίχως:

$$\begin{aligned}x &\leftarrow 2 \\y &\leftarrow 5 \\A &\leftarrow x*y\end{aligned}$$

Απάντηση: Για να θέσουμε την τιμή 2 στη μεταβλητή x , δίνουμε την εντολή:

$$x = 2 \qquad x = 2$$

Για να θέσουμε την τιμή 5 στη μεταβλητή y , δίνουμε την εντολή:

$$y = 5 \qquad y = 5$$

Για να εκχωρήσουμε το γινόμενο των μεταβλητών x και y στη μεταβλητή A , δίνουμε την εντολή:

$$A = x * y \qquad A = 10$$

Σημείωση: Στο Matlab οι μεταβλητές είναι “case sensitive”, δηλαδή οι κεφαλαίοι και οι πεζοί χαρακτήρες προσδιορίζουν **διαφορετικές** μεταβλητές. Έτσι, το A είναι διαφορετική μεταβλητή από το a .

4.3 Εντολές who και whos

Ανά πάσα στιγμή μπορούμε να δούμε τον τύπο και το χώρο (σε bytes) των μεταβλητών που έχουμε δεσμεύσει στο Matlab, χρησιμοποιώντας τις εντολές “**who**” και “**whos**”.

Εντολές Matlab	Αποτέλεσμα																				
who	Your variables are: A x y																				
whos	<table> <thead> <tr> <th>Name</th> <th>Size</th> <th>Bytes</th> <th>Class</th> <th>Attributes</th> </tr> </thead> <tbody> <tr> <td>A</td> <td>1x1</td> <td>8</td> <td>double</td> <td></td> </tr> <tr> <td>x</td> <td>1x1</td> <td>8</td> <td>double</td> <td></td> </tr> <tr> <td>y</td> <td>1x1</td> <td>8</td> <td>double</td> <td></td> </tr> </tbody> </table>	Name	Size	Bytes	Class	Attributes	A	1x1	8	double		x	1x1	8	double		y	1x1	8	double	
Name	Size	Bytes	Class	Attributes																	
A	1x1	8	double																		
x	1x1	8	double																		
y	1x1	8	double																		
whos A	<table> <thead> <tr> <th>Name</th> <th>Size</th> <th>Bytes</th> <th>Class</th> <th>Attributes</th> </tr> </thead> <tbody> <tr> <td>A</td> <td>1x1</td> <td>8</td> <td>double</td> <td></td> </tr> </tbody> </table>	Name	Size	Bytes	Class	Attributes	A	1x1	8	double											
Name	Size	Bytes	Class	Attributes																	
A	1x1	8	double																		

4.4 Ενσωματωμένες συναρτήσεις και σταθερές Matlab

Το Matlab διαθέτει μια πληθώρα ενσωματωμένων συναρτήσεων για τον υπολογισμό μαθηματικών και τριγωνομετρικών συναρτήσεων, στατιστική ανάλυση, επεξεργασία σημάτων, δημιουργία φίλτρων, βελτιστοποίηση, επίλυση διαφορικών εξισώσεων και για

πολλές άλλες λειτουργίες.

Μπορούμε να κάνουμε αναζήτηση στις συναρτήσεις του Matlab δίνοντας την εντολή:

lookfor [λέξη-κλειδί]

Αυτή η εντολή θα επιστρέψει όλες εκείνες τις συναρτήσεις που έχουν στην περιγραφή τους την λέξη-κλειδί.

Για παράδειγμα, εάν θέλουμε να αναζητήσουμε συναρτήσεις για την παρα-γωγή τυχαίων αριθμών μια λέξη κλειδί πιθανώς να είναι η "random", οπότε δίνοντας την εντολή:

lookfor random

το Matlab απαντά επιστρέφοντας το όνομα της συνάρτησης συνοδευόμενο από μία σύντομη περιγραφή.

Στη συνέχεια μπορούμε να δούμε αναλυτικά το υλικό βοήθειας για κάποια συνάρτηση δίνοντας την εντολή:

help [όνομα_συνάρτησης]

Για παράδειγμα η συνάρτηση **rand** παράγει ομοιόμορφα κατανεμημένους ψευδοτυχαίους αριθμούς. Για να δούμε τη βοήθεια σχετικά με τη συνάρτηση **rand** δίνουμε την εντολή:

help rand

και λαμβάνουμε ως απάντηση βοηθητικό υλικό για τη συνάρτηση rand.

Η εντολή **doc όνομα_συνάρτησης** ανοίγει το αρχείο τεκμηρίωσης του Matlab για τη συγκεκριμένη συνάρτηση που έχουμε προσδιορίσει.

Για παράδειγμα:

```
doc rand
doc lookfor
doc sin
doc sqrt
```


Παράδειγμα 3

Χρησιμοποιώντας τις εντολές **lookfor** και **help** να βρείτε τη συνάρτηση που υπολογίζει τον φυσικό λογάριθμο¹. Κατόπιν να γράψετε την εντολή Matlab που υπολογίζει τον φυσικό λογάριθμο του **2π**. Το αποτέλεσμα να εκχωρηθεί στη μεταβλητή **A**.

Απάντηση:

```
lookfor logarithm
A = log(2*pi)
```

Στη συνέχεια αναζητήστε την εντολή που υπολογίζει το εκθετικό ενός αριθμού με βάση το e (≈ 2.718) (δηλαδή το e^x) και εκχωρήστε στη μεταβλητή **B** την τιμή του εκθετικού της μεταβλητής **A** (δηλαδή $B = e^A$).

Απάντηση:

```
lookfor exponential
B = exp(A)
```

Υπόδειξη: Λέξεις κλειδιά: **logarithm** και **exponential**

Η σταθερά π στο Matlab συμβολίζεται **πάντα** ως **pi**

Παράδειγμα 4

Υπολογίστε την τιμή της παράστασης και απεικονίστε το αποτέλεσμα στην οθόνη:

$$X = \frac{1}{2\pi} e^{\log_e(2\pi)}$$

Απάντηση: $X = (1/(2*pi)) * \exp(\log(2*pi))$

Μπορούμε να χρησιμοποιήσουμε τις παρακάτω **ενσωματωμένες συναρτήσεις** του Matlab για να εκτελέσουμε με εύκολο τρόπο διάφορους μαθηματικούς υπολογισμούς.

¹ Ο **φυσικός λογάριθμος** έχει ως βάση την σταθερά e (≈ 2.718) και η χρήση του είναι διαδεδομένη στον μαθηματικό λογισμό. Ο **δυαδικός αλγόριθμος** έχει ως βάση τον αριθμό $b = 2$ και χρησιμοποιείται πολύ στην επιστήμη των υπολογιστών.

Παράδειγμα 5

Μαθηματική έκφραση	Εντολές Matlab	Αποτέλεσμα
$\sqrt{100}$	<code>sqrt(100)</code>	ans = 10
$\sqrt{2}^2$	<code>sqrt(2)^2</code>	ans = 2.0000
$\eta\mu(\pi/2)$	<code>sin(pi/2)</code>	ans = 1
$\sigma\upsilon\nu(\pi)$	<code>cos(pi)</code>	ans = -1
$\epsilon\phi(\pi/4)$	<code>tan(pi/4)</code>	ans = 1.0000
Τόξο $\epsilon\phi(\pi/4)$	<code>atan(pi/4)</code>	ans = 0.6658
$\log_{10}(100)$	<code>log10(100)</code>	ans = 2
$\log_2(32)$	<code>log2(32)</code>	ans = 5
e	<code>exp(1)</code>	ans = 2.7183
e^2	<code>exp(2)</code>	ans = 7.3891
e^2	<code>exp(1)^2</code>	ans = 7.3891
$\ln_e e$	<code>reallog(exp(1))</code>	ans = 1

Παράδειγμα 6

Να υπολογίσετε στο Matlab τις ακόλουθες αριθμητικές παραστάσεις. Καταγράψτε την εντολή καθώς και την τιμή των μεταβλητών που προέκυψε.

Αριθμητικές παραστάσεις	Εντολές Matlab
$A = 2 \eta\mu\left(3\frac{\pi}{2}\right)$	A = 2*sin(3*pi/4) A = 1.4142
$B = 3A + 2\sigma\upsilon\nu\left(\frac{\pi}{4}\right)$	B = 3*A+2*cos(pi/4) B = 5.6569
$C = (A + B)^2$	C = (A+B)^2 C = 50.000
$D = \sqrt{C}$	D = sqrt(C) D = 7.0711
$E = e^{-D}$	E = exp(-D) E = 8.4933e-04

Σημείωση: Πριν από την εκτέλεση της δραστηριότητας δώστε την εντολή `clear`, η οποία θα

διαγράψει τις προϋπάρχουσες μεταβλητές. Έτσι θα είστε βέβαιοι ότι οι μεταβλητές που χρησιμοποιείτε δεν έχουν κάποια τιμή από προηγούμενο υπολογισμό.

4.5 Μορφή μεταβλητών

Στις προηγούμενες δραστηριότητες χρησιμοποιήσαμε τη σταθερά π . Οι γνωστές σταθερές είναι ήδη δηλωμένες στο Matlab και μπορούμε να τις χρησιμοποιήσουμε χωρίς να απαιτείται να προβούμε σε κάποια δήλωσή τους.

```
pi
ans = 3.1416
```

Το Matlab επιστρέφει την τιμή του π με τέσσερα δεκαδικά ψηφία. Με την εντολή **format** μπορούμε να αλλάξουμε τον τρόπο με τον οποίο το Matlab απεικονίζει μία μεταβλητή, χωρίς φυσικά να επηρεάζεται η τιμή της.

Παράδειγμα 7

Μορφή	Εντολή Matlab	Αποτέλεσμα
Μορφή long	format long pi	ans = 3.14159265358979
Επιστημονική μορφή	format long e pi ή format short e pi	ans = 3.141592653589793e+00 ή ans = 3.141592653589793e+00
Μορφή bank	format bank pi	ans = 3.14
Κλασματική μορφή	format rat pi	ans = 355/113
Επιστροφή στην προκαθορισμένη μορφή	format pi	ans = 3.1416

4.6 Διανύσματα

Στο Matlab μπορούμε πολύ εύκολα να δημιουργήσουμε μεταβλητές τύπου διανύσματος (πίνακα). Μία μεταβλητή διανύσματος περιέχει περισσότερες από μία θέσεις, άρα μπορεί να αποθηκεύσει ταυτόχρονα πολλές τιμές. Τα διανύσματα είναι εξαιρετικά χρήσιμα στην επεξεργασία σημάτων. Για να ορίσουμε ένα διάνυσμα περικλείουμε με αγκύλες τα στοιχεία (τιμές) που το αποτελούν.

Παράδειγμα 8

Καταχωρούμε πολλές τιμές στην μεταβλητή ans.

```
[ 1 2 3 4 ]          ans = 1    2    3    4
```

Μπορούμε να εκχωρήσουμε πολλές τιμές σε μία μεταβλητή τύπου διανύσματος.

```
A = [ 1 2 3 4 ]          A = 1            2            3            4
```

Τα στοιχεία ενός διανύσματος προσδιορίζονται με δείκτες ξεκινώντας από τον αριθμό 1. Για να προσπελάσουμε μία συγκεκριμένη θέση ενός διανύσματος, καλούμε τη μεταβλητή με τη χρήση δείκτη (εντός παρενθέσεων) που προς-διορίζει τη θέση:

```
A(1)          ans = 1
A(3)          ans = 3
```

Προσοχή ! Η πρώτη θέση του διανύσματος αριθμείται με την τιμή 1 και όχι με το 0, όπως συμβαίνει σε άλλες γλώσσες προγραμματισμού.

Ένας άλλος τρόπος για να δημιουργήσουμε ένα διάνυσμα είναι με τη χρήση του συμβόλου της άνω-κάτω τελείας «:» στην μορφή:

Μεταβλητή = πρώτο_στοιχείο : βήμα : τελευταίο_στοιχείο

Εάν το βήμα παραλειφθεί από την εντολή, τότε ως βήμα θεωρείται η τιμή 1. Δηλαδή η εντολή:

Μεταβλητή = πρώτο_στοιχείο : τελευταίο_στοιχείο

είναι ισοδύναμη με:

Μεταβλητή = πρώτο_στοιχείο : 1 : τελευταίο_στοιχείο

Παράδειγμα 9

Στο παράδειγμα αυτό θα δούμε πως μπορούμε να καταχωρήσουμε τιμές σε μεταβλητή τύπου διανύσματος.

a = 1 : 1 : 5	a = 1 2 3 4 5
b = 0 : 0.1 : 1	b = 0 0.1000 0.2000 0.3000 0.4000 0.5000 0.6000 0.7000 0.8000 0.9000 1.0000
c = 0 : pi/8 : pi	c = 0 0.3927 0.7854 1.1781 1.5708 1.9635 2.3562 2.7489 3.1416

Προσπέλαση συγκεκριμένων θέσεων του διανύσματος:

b(9)	ans = 0.8000
c(8)	ans = 2.7489

Σημείωση: Στο Matlab όλες οι μεταβλητές αντιμετωπίζονται ως πίνακες. Ακόμα και ένας απλός αριθμός θεωρείται πίνακας διαστάσεων 1x1. Ένα στοιχείο ενός διανύσματος μπορεί να είναι κάθε τύπου, ακόμα και άλλα διανύσματα:

format short d = [a b c]	d = 1.0000 2.0000 3.0000 4.0000 5.0000 0 0.1000 0.2000 0.3000 0.4000 0.5000 0.6000 0.7000 0.8000 0.9000 1.0000 0 0.3927 0.7854 1.1781 1.5708 1.9635 2.3562 2.7489 3.1416
e = [d -3:1:1]	e = 1.0000 2.0000 3.0000 4.0000 5.0000 0 0.1000 0.2000 0.3000 0.4000 0.5000 0.6000 0.7000 0.8000 0.9000 1.0000 0.3927 0.7854 1.1781 1.5708 1.9635 2.3562 2.7489 3.1416 -3.0000 -2.0000 -1.0000 0 1.0000

Επίσης μια συνάρτηση μπορεί να ενεργεί πάνω σε ένα διάνυσμα:

cos(a)	ans = 0.5403 -0.4161 -0.9900 -0.6536 0.2837
sin(0:pi/8:pi)	ans = 0 0.3827 0.7071 0.9239 1.0000 0.9239 0.7071 0.3827 0.0000

Παράδειγμα 10

Δημιουργήστε ένα διάνυσμα με όνομα E που να περιέχει τις τιμές e^n , όπου $0 \leq n \leq 10$ και $n \in \mathbb{Z}$.

```
n = 0 : 1 : 10
E = exp(n)
```

4.7 Πράξεις με διανύσματα

Για να εκτελέσουμε πράξεις μεταξύ δύο διανυσμάτων απαιτείται τα διανύσματα αυτά να έχουν το ίδιο μήκος, δηλαδή τις ίδιες διαστάσεις.

Παράδειγμα 11

<code>clear</code>	Σημείωση: Δίνουμε την εντολή <code>clear</code> για να διαγράψουμε τις προηγούμενες μεταβλητές.
<code>A = 0:5</code>	A = 0 1.00 2.00 3.00 4.00 5.00
<code>B = 5:10</code>	B = 5.00 6.00 7.00 8.00 9.00 10.00
<code>C = A + B</code>	C = 5.00 7.00 9.00 11.00 13.00 15.00
<code>D = A - B</code>	D = -5.00 -5.00 -5.00 -5.00 -5.00 -5.00

Η πρόσθεση ή η αφαίρεση ενός διανύσματος με ένα βαθμωτό μέγεθος (δηλαδή μεταβλητή μίας θέσης), προσθέτει ή αφαιρεί από κάθε στοιχείο του διανύσματος το βαθμωτό μέγεθος:

$$E = D - 1 \qquad E = -6.00 \quad -6.00 \quad -6.00 \quad -6.00 \quad -6.00 \quad -6.00$$

Η πράξη του πολλαπλασιασμού και της διαίρεσης μεταξύ πινάκων διαφέρει από την πρόσθεση και την αφαίρεση, επειδή **για να ορίζεται το γινόμενο δύο πινάκων, απαιτείται ο αριθμός των στηλών του πρώτου πίνακα (A) να ισούται με τον αριθμό των γραμμών του δεύτερου πίνακα (B)**. Επομένως, πριν πολλαπλασιάσουμε δύο πίνακες, θα πρέπει να έχουμε εξασφαλίσει αυτή τη συνθήκη. Θα δούμε αναλυτικά τον πολλαπλασιασμό των πινάκων στην ενότητα 4.9.

Για να πολλαπλασιάσουμε έναν πίνακα **στοιχείο προς στοιχείο** με έναν άλλο, πρέπει να χρησιμοποιήσουμε το σύμβολο της τελείας «.» **πριν** τον τελεστή της πράξης («*» για πολλαπλασιασμό και «/» για διαίρεση). Προσοχή! Αυτή η πράξη **δεν ισοδυναμεί** με τον πολλαπλασιασμό των πινάκων.

$$\begin{aligned} E &= A .* B & E &= 0 \quad 6.00 \quad 14.00 \quad 24.00 \quad 36.00 \quad 50.00 \\ F &= A ./ B & F &= 0 \quad 0.17 \quad 0.29 \quad 0.38 \quad 0.44 \quad 0.50 \end{aligned}$$

Παράδειγμα 12

Η ύψωση σε δύναμη κάθε στοιχείου ενός πίνακα πραγματοποιείται επίσης με τη χρήση του συμβόλου της τελείας «.» πριν από το σύμβολο «^» της ύψωσης σε δύναμη.

<code>G = F.^2</code>	G = 0 0.03 0.08 0.14 0.20 0.25
<code>H = [0:pi/8:pi/2].^2</code>	H = 0 0.1542 0.6169 1.3879 2.4674
<code>I = cos([0:pi/8:pi/2].^2)</code>	I = 1.0000 0.9881 0.8157 0.1819 -0.7812

$J = \cos([0:\pi/8:\pi/2]).^2$	$J =$	1.0000	0.8536	0.5000	0.1464
		0.0000			

Παράδειγμα 13

Δώστε την εντολή **clear** ώστε να διαγράψετε όλες τις προηγούμενες μεταβλητές από το Workspace και κατόπιν:

- (α) Δημιουργήστε το διάνυσμα A με τιμές από 0 έως $\pi/2$, με βήμα $\pi/8$.
- (β) Δημιουργήστε το διάνυσμα B με τιμές από 1 έως 5, με βήμα 1.

Δώστε την κατάλληλη εντολή στο Matlab ώστε να υψωθεί κάθε στοιχείο του διανύσματος A στο αντίστοιχο στοιχείο του διανύσματος B. Το αποτέλεσμα να αποθηκευθεί στο διάνυσμα C.

Απάντηση:

```
clear
A = 0 : pi/8 : pi/2;
B = 1 : 1 : 5;
C = A.^B
```

Αποτελέσματα:

```
C = 0      0.1542      0.4845      1.9263      9.5631
```

4.8 Διανύσματα Στήλης και Πίνακες

Μέχρι στιγμής είδαμε διανύσματα γραμμής. Για να δημιουργήσουμε διανύσματα στήλης εργαζόμαστε ακριβώς όπως στα διανύσματα γραμμής, με μόνη διαφορά ότι μεταξύ των στοιχείων του διανύσματος βάζουμε το σύμβολο του ερωτηματικού ";".

Ένας άλλος τρόπος για να δημιουργήσουμε διάνυσμα στήλης είναι να δημιουργήσουμε αρχικά ένα διάνυσμα γραμμής και κατόπιν να χρησιμοποιήσουμε το σύμβολο «'» της αναστροφής πίνακα, όπως φαίνεται παρακάτω. Αυτή η μέθοδος λειτουργεί και αντίστροφα.

Παράδειγμα 14

```
A = [1; 2; 3; 4]      A =
      1
      2
      3
      4
B = [0:1:3]'         B =
```

```

                                0
                                1
                                2
                                3
C = A'                          C =
                                1    2    3    4
D = C'                          D =
                                1
                                2
                                3
                                4

```

Ομοίως, χρησιμοποιώντας το σύμβολο του ερωτηματικού «;» μπορούμε να δημιουργήσουμε πίνακες. Όπως αναφέρθηκε παραπάνω, τα περιεχόμενα ενός διανύσματος μπορεί να είναι κάθε τύπου, ακόμα και άλλα διανύσματα. Ένα διάνυσμα στήλης του οποίου τα στοιχεία είναι διανύσματα γραμμής, αποτελεί έναν πίνακα.

Παράδειγμα 15

clear	D =
A = [1 2 3 4];	1 2 3 4
B = [5 6 7 8];	5 6 7 8
C = [9 10 11 12];	9 10 11 12
D = [A ; B; C ;]	

Παρατηρούμε ότι τοποθετώντας το σύμβολο του ερωτηματικού «;» **μετά** από μια εντολή, το Matlab δεν εκτυπώνει το αποτέλεσμα στην οθόνη.

D(2,3)	ans =
	7
E = D'	E =
	1 5 9
	2 6 10
	3 7 11
	4 8 12

Μπορούμε να εκχωρήσουμε τιμές σε συγκεκριμένα στοιχεία του πίνακα. Ακολουθούν διάφοροι τρόποι:

1) Σε συγκεκριμένη θέση του πίνακα:

$E(2,3)$ ans =
10

2) Σε συγκεκριμένη στήλη του πίνακα:

$E(1,2) = 0;$ E =
 $E(2,2) = 0;$ 1 0 9
 $E(3,2) = 0;$ 2 0 10
 $E(4,2) = 0;$ 3 0 11
E 4 0 12

3) Σε συγκεκριμένη περιοχή γραμμών και στηλών του πίνακα:

$E(1:4, 1:3) = 1$ E =
1 1 1
1 1 1
1 1 1
1 1 1

4) Σε συγκεκριμένες θέσεις μίας συγκεκριμένης στήλης του πίνακα:

$E(1:4,2) = 0$ E =
1 0 1
1 0 1
1 0 1
1 0 1

5) Σε όλες τις θέσεις μίας στήλης του πίνακα:

$E(:,1) = 0;$ E =
 $E(:,2) = 1;$ 0 1 2
 $E(:,3) = 2$ 0 1 2
0 1 2
0 1 2

Παρατηρήσεις:

- Χρησιμοποιώντας το σύμβολο «:» μπορούμε να αναφερθούμε σε ολόκληρες γραμμές ή στήλες του πίνακα.
- Αν θέλουμε να αναφερθούμε σε όλες τις θέσεις της k γραμμής του πίνακα A αρκεί να γράψουμε A(k,:).
- Εάν θέλουμε να αναφερθούμε σε όλες τις θέσεις της m στήλης του πίνακα A αρκεί να γράψουμε A(:,m).
- Για να αναφερθούμε σε ένα τμήμα του πίνακα A, π.χ. από την γραμμή k1 έως την γραμμή k2 και από τη στήλη m1 έως τη στήλη m2, γράφουμε A(k1:k2, m1:m2).

Παράδειγμα 16

Δώστε τις εντολές Matlab ώστε να δημιουργήσετε τον πίνακα A με διαστάσεις 11x11. Τα στοιχεία του A από την 4 έως την 8 γραμμή και από την 4 έως την 8 στήλη είναι μηδενικά. Όλα τα υπόλοιπα είναι 1.

```
A = ones(11,11);
A(4:8, 4:8) = 0 ;
```

Στη συνέχεια να δημιουργήσετε τον ίδιο πίνακα αντικαθιστώντας αμοιβαία τα μηδενικά (0) με τον αριθμό ένα (1).

```
B = zeros(11,11);
B(4:8, 4:8) = 1;
```

4.9 Πολλαπλασιασμός Πινάκων

Ο πολλαπλασιασμός δύο πινάκων $C = A \times B$ είναι το αλγεβρικό γινόμενο των πινάκων A και B και δίνεται από τη σχέση:

$$C(i, j) = \sum_{k=1}^n A(i, k) B(k, j)$$

Όταν πολλαπλασιάζουμε δύο πίνακες, πρέπει το πλήθος των στηλών του A να ισούται με το πλήθος των γραμμών του B. Ένα βαθμωτό μέγεθος μπορεί να πολλαπλασιάσει έναν πίνακα οποιωνδήποτε διαστάσεων.

Παράδειγμα 17

```
clear
A = ones(3,3)
```

A =			
	1	1	1
	1	1	1
	1	1	1

```
B = ones(3,1)+[0:2]'
```

B =
1
2
3

```
C = A*B
```

C =
6
6
6

```
D = 2*C'
```

D = 12 12 12

4.10 Χρήσιμες συναρτήσεις για τη δημιουργία πινάκων

Παρατίθενται μερικές χρήσιμες συναρτήσεις για τη δημιουργία πινάκων:

zeros(m,n)	Παράγει έναν πίνακα m x n διαστάσεων γεμάτο με μηδενικά.
ones(m,n)	Παράγει έναν πίνακα m x n διαστάσεων γεμάτο με τον αριθμό ένα.
rand(m,n)	Παράγει έναν πίνακα m x n διαστάσεων γεμάτο με ομοιόμορφα κατανομημένους τυχαίους αριθμούς.
randn(m,n)	Παράγει έναν πίνακα m x n διαστάσεων γεμάτο με κανονικά κατανομημένους αριθμούς.
eye(m,n)	Παράγει έναν πίνακα m x n διαστάσεων με άσσους στην κύρια διαγώνιο και μηδενικά σε όλα τα άλλα στοιχεία.

Παράδειγμα 18

Τροποποιώντας κατάλληλα την παρακάτω εντολή Matlab να δημιουργήσετε τον πίνακα που εμφανίζεται στο αποτέλεσμα:

```
A = 9*ones(5, 5) - 9*eye(5, 5)
```

```
A = 0 9 9 9 9
9 0 9 9 9
9 9 0 9 9
9 9 9 0 9
9 9 9 9 0
```

5. Η εργαλειοθήκη Symbolic

Χρησιμοποιώντας την εργαλειοθήκη Symbolic του Matlab μπορούμε να εκτελέσουμε αλγεβρικούς ή συμβολικούς υπολογισμούς, όπως την παραγοντοποίηση πολυωνύμων, την επίλυση αλγεβρικών εξισώσεων, κλπ, χωρίς να απαιτείται ο προκαθορισμός των μεταβλητών.

5.1 Συμβολικές μεταβλητές

Για να ορίσουμε μια συμβολική μεταβλητή χρησιμοποιούμε τη δήλωση `syms` ως εξής:

```
syms x y
```

Παράδειγμα 19

Εντολές Matlab	Αποτέλεσμα
<code>syms x y</code> <code>(x+y)^2</code>	<code>ans = (x + y)^2</code>
<code>expand(ans)</code>	<code>ans = x^2 + 2*x*y + y^2</code>
<code>factor(ans)</code>	<code>ans = (x + y)^2</code>

Η συνάρτηση `expand()` ορίζει στο Matlab να εκτελέσει αναλυτικά τις πράξεις μέσα στην παράσταση, ενώ η συνάρτηση `factor()` να επιστρέψει την παραγοντοποιημένη μορφή της.

5.2 Παραγωγήιση συμβολικών συναρτήσεων και επίλυση εξισώσεων

Η συνάρτηση `diff()` υπολογίζει την παράγωγο μίας συμβολικής παράστασης σε σχέση με την ανεξάρτητη μεταβλητή που εμφανίζεται στην παράσταση. Αν η παράσταση έχει περισσότερες από μία ανεξάρτητες μεταβλητές τότε η συνάρτηση `diff()` συντάσσεται ως `diff(F, v)` όπου F είναι η συμβολική παράσταση και v η μεταβλητή ως προς την οποία παραγωγίζεται η παράσταση.

Η συνάρτηση `solve()` επιλύει συμβολικές αλγεβρικές εξισώσεις και συντάσσεται `solve(eq1,var1)`. Το eq1 είναι η συμβολική έκφραση. Αν η έκφραση δεν είναι εξίσωση, τότε το Matlab αναζητά τα σημεία που μηδενίζεται η εξίσωση, διαφορετικά όπως ορίζεται από την εξίσωση. Το var1 είναι η άγνωστη μεταβλητή. Σε περίπτωση που αυτή είναι μόνο μία, μπορούμε να μην την δηλώσουμε.

Παράδειγμα 20

(α) Να υπολογίσετε την παράγωγο της συνάρτησης $y = x^3 + 4x^2 - 23$.

Για τον υπολογισμό της παραγώγου χρησιμοποιήστε τη συνάρτηση `diff()`.

Εντολές Matlab	Αποτέλεσμα
syms x y = x^3 + 4*x^2 - 23	y = x^3 + 4*x^2 - 23
Yd = diff(y)	Yd = 3*x^2 + 8*x

(β) Στη συνέχεια χρησιμοποιήστε τη συνάρτηση `solve()` για να βρείτε τις ρίζες της συνάρτησης και της παραγώγου.

```
Yd_roots = solve(Yd)      Yd_roots =      0      -8/3
dA = subs(Yd, x, -4)     dA = 16
```

Χρησιμοποιήστε τη συνάρτηση `subs()`² για να ελέγξετε το πρόσημο της παραγώγου για τιμές στα διαστήματα που ορίζουν οι ρίζες της παραγώγου.

```
dB = subs(Yd, x, -1)      dB = -5
dC = subs(Yd, x, 1)      dC = 11
```

(γ) Αποφανθείτε για την μονοτονία της συνάρτησης.

Διαστήματα του άξονα X'X που ορίζουν οι ρίζες	Μονοτονία Συνάρτησης
$(-\infty, -8/3)$	Yd(-4) = 16 → Αύξουσα
$(-8/3, 0)$	Yd(-1) = -5 → Φθίνουσα
$(0, +\infty)$	Yd(1) = 11 → Αύξουσα

5.3 Ολοκλήρωση Συμβολικών Συναρτήσεων

Για τον υπολογισμό του ολοκληρώματος μιας συμβολικής έκφρασης χρησιμοποιούμε τη συνάρτηση `int()`. Σε περίπτωση που η παράσταση έχει περισσότερες από μία ανεξάρτητες μεταβλητές, η συνάρτηση `int()` συντάσσεται `int(F, v)` όπου F είναι η συμβολική παράσταση και v η μεταβλητή ως προς την οποία ολοκληρώνεται η παράσταση.

Μπορούμε επίσης να ορίσουμε τα όρια ολοκλήρωσης ως `int(F, v, a, b)` όπου a είναι το κάτω όριο ολοκλήρωσης και b το άνω όριο ολοκλήρωσης.

² Η εντολή `A = subs(y,x,2)` θα αντικαταστήσει στη συνάρτηση y την τιμή του x με 2 και θα εκχωρήσει το αποτέλεσμα στη μεταβλητή A.

Παράδειγμα 21

(α) Να υπολογίσετε το αόριστο ολοκλήρωμα $F = \int 3x^2 + 8x$ με τη συνάρτηση `int()`.

Εντολές Matlab	Αποτέλεσμα
<pre>syms x a b f = 3*x^2 + 8*x; F = int(f)</pre>	$F = x^2*(x + 4)$

(β) Να υπολογίσετε το ολοκλήρωμα $Fab = \int_a^b 3x^2 + 8x$

$$Fab = \text{int}(f, a, b)$$

$$Fab = b^2*(b + 4) - a^2*(a + 4)$$

(γ) Στη συνέχεια να υπολογίσετε το ολοκλήρωμα $Fd = \int_{-2}^2 3x^2 + 8x$

$$Fd = \text{int}(f, -2, 2)$$

$$Fd = 16$$

(δ) Χρησιμοποιώντας κατάλληλα την εντολή `subs()` αντικαταστήστε στο ερώτημα II τα a και b με τις τιμές $a = -2$ και $b = 2$ ώστε να επιβεβαιώσετε ότι προκύπτει το ίδιο αποτέλεσμα με το ερώτημα III.

$$\text{subs}(Fab, \{a,b\}, \{-2,2\})$$

$$\text{ans} = 16$$

5.4 Συμβολικό άθροισμα σειράς

Για τον υπολογισμό του αθροίσματος μιας συμβολικής σειράς χρησιμοποιούμε τη συνάρτηση `symsum()`. Η σύνταξη της εντολής είναι `symsum(F, n, a, b)` όπου F είναι η συμβολική έκφραση, n ο δείκτης της σειράς, a το κάτω όριο και b το άνω όριο.

Παράδειγμα 22

Να υπολογίσετε το άθροισμα

$$\sum_{n=0}^{10} \frac{1}{(1+n)^2}$$

Απάντηση:

Εντολές Matlab	Αποτέλεσμα
<pre>syms n S = 1/((1+n)^2); symsum(S,0,10)</pre>	<pre>ans = 239437889/153679680</pre>

6. Σχεδιασμός Γραφικών Παραστάσεων

Η συνάρτηση `ezplot()` μπορεί να απεικονίσει γραφικά μια συμβολική παράσταση. Το Matlab θα υπολογίσει από μόνο του τα όρια και την κλίμακα των αξόνων.

Παράδειγμα 23

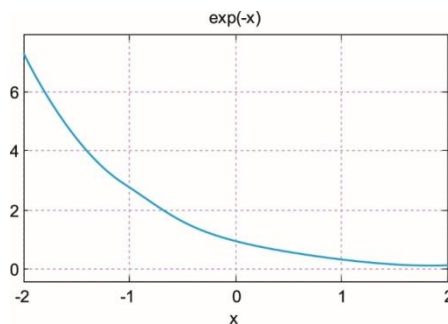
Να σχεδιαστεί η συνάρτηση $y(x) = e^{-x}$.

Απάντηση:

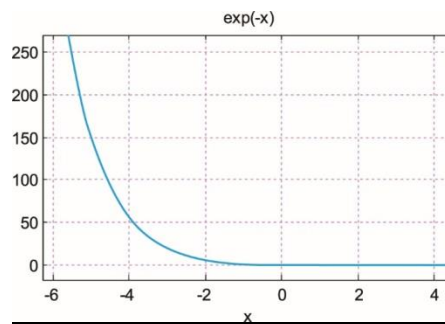
```
syms x;
y = exp(-x);
ezplot(y); grid on
```

Η εντολή `ezplot(y, min, max)` θα αναπαραστήσει τη συνάρτηση y από \min ως \max στον άξονα του x .

```
ezplot(y, -2, 2);
grid on
```



(α)



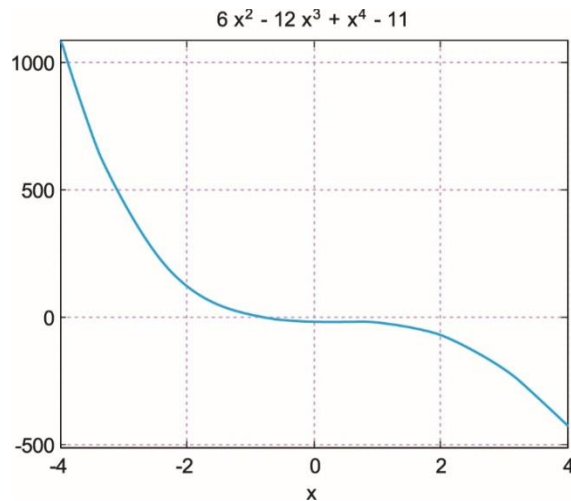
(β)

Σχήμα 2 Συνάρτηση $y(x) = e^{-x}$ σχεδιασμένη: (α) στην αυτόματη κλίμακα τιμών άξονα x , (β) σε καθορισμένη κλίμακα τιμών άξονα x .

Παράδειγμα 24

Χρησιμοποιήστε τη συνάρτηση `ezplot()` για να αναπαραστήσετε γραφικά τη συνάρτηση $y(x) = x^4 - 12x^3 + 6x^2 - 11$, στο διάστημα τιμών $x = [-4, 4]$.

```
syms x
y = x^4 - 12*x^3 + 6*x^2 - 11;
ezplot(y, -4, 4); grid on
```



Σχήμα 3 Γραφική παράσταση συνάρτησης $y(x) = x^4 - 12x^3 + 6x^2 - 11$

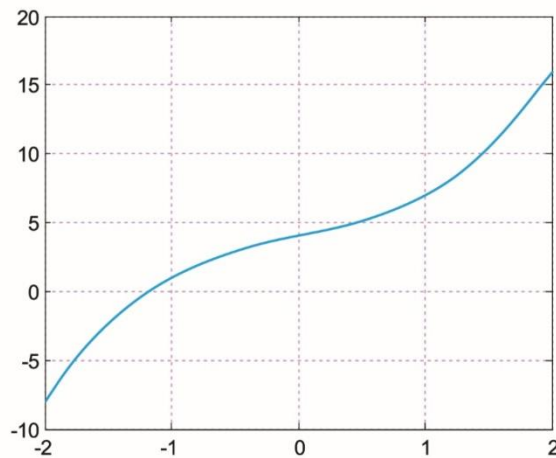
Η εντολή `plot(x, y)` λειτουργεί με διανύσματα αριθμητικών δεδομένων, όπου τα x και y είναι διανύσματα ίσου μήκους. Η εντολή `plot(x, y)` θεωρεί τα διανύσματα x και y σαν λίστες συντεταγμένων μιας αλληλουχίας σημείων στο καρτεσιανό επίπεδο, τα οποία σημεία συνδέει ένα ευθύγραμμο τμήμα.

Παράδειγμα 25

Χρησιμοποιήστε τη συνάρτηση `plot()` για να αναπαραστήσετε γραφικά τη συνάρτηση $y(x) = x^3 + 2x + 4$ στο διάστημα τιμών $x = [-4, 4]$.

Απάντηση:

```
x = -2 : 0.1 : 2;
y = x.^3 + 2*x + 4;
plot(x, y); grid on
```



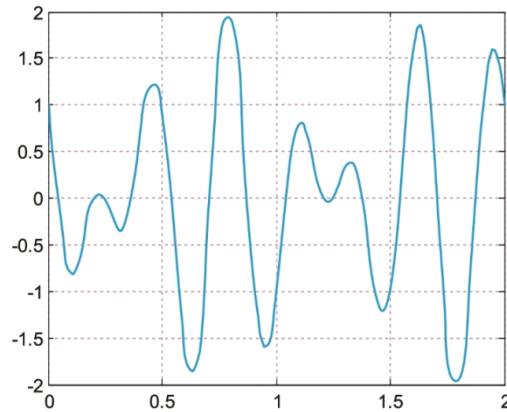
Σχήμα 4 Γραφική παράσταση συνάρτησης $y(x) = x^3 + 2x + 4$

Παράδειγμα 26

Να απεικονίσετε τη συνάρτηση $x(t) = \cos(5\pi t) - \sin(7\pi t)$ στο διάστημα χρόνου $t = [0,2]$ με βήμα δειγματοληψίας $ts = 0.01$.

Απάντηση:

```
ts = 0.01; t = 0 : ts : 2;
x = cos(5*pi*t) - sin(7*pi*t); plot(t, x); grid on
```



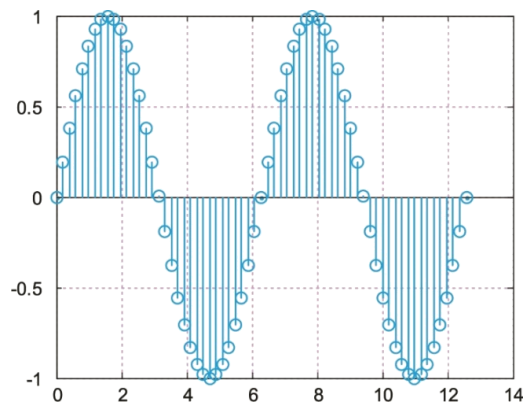
Σχήμα 5 Γραφική παράσταση συνάρτησης $x(t) = \cos(5\pi t) - \sin(7\pi t)$

Παράδειγμα 27

Να σχεδιάσετε με την εντολή **stem(x, n)** τη συνάρτηση $x(n) = \cos(n)$, $0 \leq n \leq 4\pi$ με βήμα $\pi/16$.

Απάντηση:

```
n = 0 : pi/16 : 4*pi; x = sin(n); stem(n, x); grid on
```



Σχήμα 6 Γραφική παράσταση συνάρτησης $x(n) = \cos(n)$

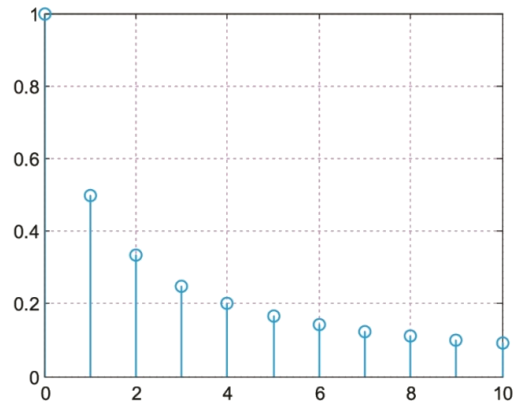
Παράδειγμα 28

Να σχεδιάσετε τους όρους της σειράς:

$$x(n) = \sum_{n=0}^{10} \frac{1}{1+n}$$

Απάντηση:

```
n = 0:10; x = 1./(1+n); stem(n, x); grid on
```



Σχήμα 7 Γραφική παράσταση συνάρτησης $x(n) = \sum_{n=0}^{10} 1/(1+n)$

6.1 Χρήσιμες συναρτήσεις για την μορφοποίηση των γραφημάτων

Εντολή/Συνάρτηση	Αποτέλεσμα
<code>hold on/off</code>	Διατηρεί το τρέχον γράφημα και τις ιδιότητες των αξόνων ώστε ένα επόμενο γράφημα να εμφανιστεί στο ίδιο παράθυρο.
<code>grid on/off</code>	Προσθέτει ένα πλέγμα στο γράφημα.
<code>xlabel('string')</code>	Προσθέτει την λεζάντα string στον x άξονα.
<code>ylabel('string')</code>	Προσθέτει την λεζάντα string στον y άξονα.
<code>axis([x1,x2,y1,y2])</code>	Διαμορφώνει τους άξονες από x1 έως x2 και y1 έως y2.
<code>title('string')</code>	Προσθέτει τίτλο με λεζάντα string στο γράφημα.
<code>xlim([x1, x2])</code>	Διαμορφώνει τον άξονα x από x1 έως x2.
<code>ylim([y1, y2])</code>	Διαμορφώνει τον άξονα y από y1 έως y2.

7. Άλυτες Ασκήσεις

1. Δώστε την κατάλληλη εντολή Matlab ώστε να δημιουργηθεί ο πίνακας του αποτελέσματος. Χρησιμοποιήστε τις συναρτήσεις **ones()**, **zeros()** και **eye()**.

```
A =
  1   1   1   0   0   0   1   0   0
  1   1   1   0   0   0   0   1   0
  1   1   1   0   0   0   0   0   1
```

2. Δώστε την εντολή: **A = magic(4)**. Να υπολογίσετε το άθροισμα των στοιχείων για κάθε στήλη και γραμμή του πίνακα A. Επίσης, να υπολογίσετε το άθροισμα των στοιχείων της διαγωνίου. Τι συμπέρασμα βγάξετε για τον πίνακα που δημιουργεί η συνάρτηση **magic()**;

Βοήθεια: Για να υπολογίσετε το άθροισμα των στοιχείων χρησιμοποιήστε τη συνάρτηση **sum()** και για να εξάγετε την διαγώνιο χρησιμοποιήστε τη συνάρτηση **diag()**.

3. Δημιουργήστε τον πίνακα A διαστάσεων 4 x 4 με τη συνάρτηση **randi(N,N)**. Χρησιμοποιήστε τη συνάρτηση **inv()** για να υπολογίσετε τον αντίστροφο του πίνακα A. Υπολογίστε το γινόμενο του αντίστροφου πίνακα του A με τον πίνακα A. Τι συμπέρασμα βγάξετε από το αποτέλεσμα που προκύπτει;
4. Υπολογίστε το παρακάτω ολοκλήρωμα:

$$I = \int_0^1 \sqrt{x} dx$$

Στη συνέχεια να εκτελέσετε στο Matlab τις εξής εντολές:

```
clear
a = 0 : 0.001 : 1;
b = sqrt(a);
i = trapz(a, b)
```

Τι συμπεραίνετε ότι εκτελεί η εντολή **trapz()**;

5. Δημιουργήστε ένα διάνυσμα A με την εντολή **A = rand(1000,1)** και χρησιμοποιήστε την εντολή **hist(A)** για να αναπαραστήσετε το ιστόγραμμα του.
6. Δημιουργήστε ένα διάνυσμα **B = randn(1000,1)** και αναπαραστήστε το ιστόγραμμα του όπως στην προηγούμενη άσκηση. Τι συμπεραίνετε;
7. Δημιουργήστε τη συνάρτηση δύο μεταβλητών $f(x,y) = y/(1+x^2+y^2)$. Χρησιμοποιήστε τη συνάρτηση **ezsurf()** για να αναπαραστήσετε τη γραφική παράσταση της $f(x,y)$.