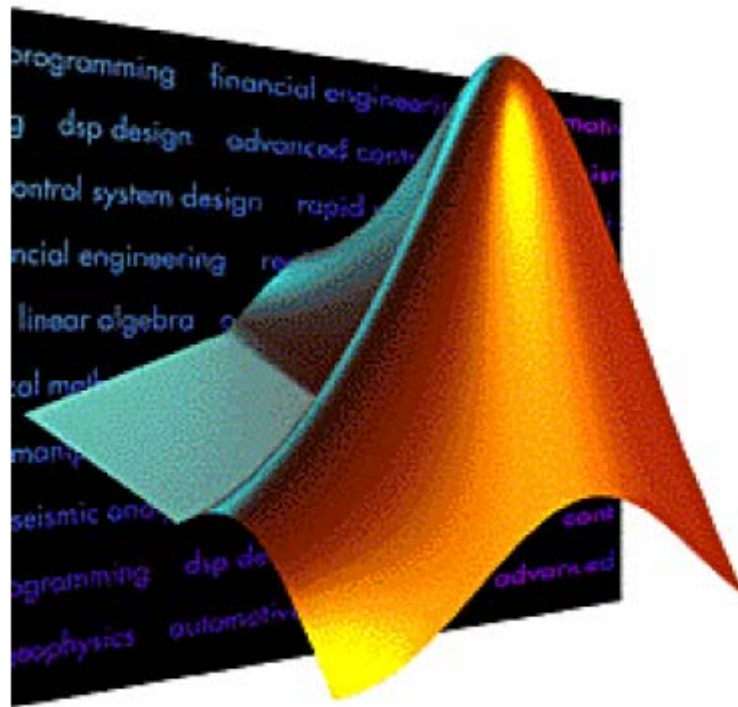




Τ.Ε.Ι ΚΡΗΤΗΣ – ΠΑΡΑΡΤΗΜΑ ΧΑΝΙΩΝ
ΤΜΗΜΑ ΦΥΣΙΚΩΝ ΠΟΡΩΝ ΚΑΙ ΠΕΡΙΒΑΛΛΟΝΤΟΣ
ΤΟΜΕΑΣ ΥΔΑΤΙΚΩΝ ΠΟΡΩΝ & ΓΕΩΠΕΡΙΒΑΛΛΟΝΤΟΣ



ΓΕΩΦΥΣΙΚΗ ΣΕ ΜΑΤΛΑΒ



ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Πετράκη Μαρία - Πετρουλάκη Κοριακή

ΑΠΡΙΛΙΟΣ 2012



Τ.Ε.Ι ΚΡΗΤΗΣ – ΠΑΡΑΡΤΗΜΑ ΧΑΝΙΩΝ
ΤΜΗΜΑ ΦΥΣΙΚΩΝ ΠΟΡΩΝ ΚΑΙ ΠΕΡΙΒΑΛΛΟΝΤΟΣ
ΤΟΜΕΑΣ ΥΔΑΤΙΚΩΝ ΠΟΡΩΝ & ΓΕΩΠΕΡΙΒΑΛΛΟΝΤΟΣ



ΓΕΩΦΥΣΙΚΗ ΣΕ MATLAB

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Πετράκη Μαρία - Πετρουλάκη Κυριακή

Επιβλέπουσα : *Δρ. Κόκκινου Ελένη*
Επίκουρη Καθηγήτρια

Επιτροπή Αξιολόγησης : *Δρ. Ελένη Κόκκινου*
: *Δρ. Σάλτας Βασίλειος*
: *Δρ. Σουπιός Παντελής*

Ημερομηνία Παρουσίασης :
Αύξων Αριθμός Πτυχιακής Εργασίας :

Ευχαριστίες

Μέσα από αυτό το σημείωμα ευχαριστούμε θερμά την καθηγήτριά μας, κυρία Ελένη Κόκκινου, για όλη την βοήθεια που μας προσέφερε στο διάστημα της επιμέλειας της πτυχιακής μας εργασίας. Αρχικά θέλουμε να την ευχαριστήσω για την επιλογή του θέματος και την υποστήριξη της σε όλη την διάρκεια εκπόνησης της εργασίας μας. Θερμές ευχαριστίες οφείλομε επίσης στην τριμελή εξεταστική επιτροπή. Τέλος θα θέλαμε να ευχαριστήσουμε τους γονείς μας για την αγάπη τους, για το ενδιαφέρον και για ότι μας έχουν χαρίσει όλα αυτά τα χρόνια και που για μας ήταν πολύτιμο.

Περίληψη

Στα πλαίσια της παρούσας εργασίας παρουσιάζεται η χρήση της Matlab για την επεξεργασία και απεικόνιση γεωφυσικών δεδομένων. Αρχικά παρουσιάζονται τα βασικά στοιχεία χρήσης της Matlab και στην συνέχεια οι αλγόριθμοι που αναπτύχθηκαν και τα αποτελέσματα τους.

Abstract

In the context of the present work, the use of Matlab is presented, focusing in the processing and graphic presentation of geophysical data. Initially the basic principles concerning the use of Matlab are discussed. Thereinafter the algorithms, developed in the context of the present work, as well their results, are presented.

ΠΕΡΙΕΧΟΜΕΝΑ

	Τ.Ε.Ι ΚΡΗΤΗΣ – ΠΑΡΑΡΤΗΜΑ ΧΑΝΙΩΝ	1
1	ΑΝΑΠΤΥΞΗ ΑΛΓΟΡΙΘΜΩΝ	5
1.1	Εισαγωγή	5
1.2	Τρόποι παρουσίασης αλγορίθμων	5
1.3	Το Λογικό Διάγραμμα.	5
1.4	Τι είναι μεταβλητές και τι είναι σταθερές;	- 7 -
1.5	Βασικές δομές αλγορίθμων	- 8 -
1.6	Λογικές Συνθήκες	- 17 -
1.7	Διακδικασίες και Συναρτήσεις	- 18 -
1.8	Πίνακες	- 20 -
2	ΕΙΣΑΓΩΓΗ ΣΤΗΝ MATLAB	- 24 -
2.1	Εισαγωγή	- 24 -
2.2	Χρήσιμες ιδιότητες του MATLAB	- 25 -
2.3	Βασικοί τύποι αρχείων του MATLAB	- 25 -
2.4	Βασικά στοιχεία χρήσης της Matlab	- 25 -
2.5	Προγραμματίζοντας στο matlab	- 27 -
2.6	Εισαγωγή δεδομένων στο Matlab	- 29 -
2.7	Εισαγωγή δεδομένων μέσω Import Wizard	- 30 -
2.8	Εξαγωγή δεδομένων από το Matlab	- 34 -
2.9	Αρχεία κειμένου (script files)	- 37 -
2.10	Αρχεία συναρτήσεων (function files)	- 38 -
2.11	Λογικοί τελεστές και σχεσιακοί τελεστές	- 39 -
2.12	Δομές ελέγχου ροής προγράμματος	- 40 -
2.13	Γραφικά στο Matlab	- 41 -
2.14	Εντολές υλοποίησης διαφόρων μορφών γραφημάτων στο χώρο	- 48 -
2.15	Αποθήκευση γραφημάτων	- 53 -
3	ΕΠΕΞΕΡΓΑΣΙΑ ΓΕΩΦΥΣΙΚΩΝ ΔΕΔΟΜΕΝΩΝ	- 55 -
3.1	Εισαγωγή	- 55 -
3.2	Σεισμολογικά δεδομένα	- 55 -
3.3	Διαδικασία παρεμβολής σε γεωφυσικά δεδομένα	- 57 -
4	BIBΛΙΟΓΡΑΦΙΑ	- 63 -

1 ΑΝΑΠΤΥΞΗ ΑΛΓΟΡΙΘΜΩΝ

1.1 Εισαγωγή

Αλγόριθμος είναι ένας πεπερασμένος αριθμός συγκεκριμένων βημάτων (εντολών) που απαιτούνται για την επίλυση ενός προβλήματος.

Οι αλγόριθμοι για να είναι αποτελεσματικοί θα πρέπει να χαρακτηρίζονται από κάποιες ιδιότητες:

1. Το σύνολο των εντολών ενός αλγορίθμου θα πρέπει να είναι πεπερασμένο. (Η εκτέλεση των εντολών του να ολοκληρώνεται σε εύλογο χρονικό διάστημα)
2. Κάθε εντολή θα πρέπει να εκτελείται σε πεπερασμένο χρόνο.
3. Οι εντολές ενός αλγορίθμου θα πρέπει να είναι διατυπωμένες με ακρίβεια και σαφήνεια.
4. Οι εντολές ενός αλγορίθμου θα πρέπει να είναι διατυπωμένες με απλά «λόγια».

1.2 Τρόποι παρουσίασης αλγορίθμων

Σκοπός λοιπόν ενός αλγορίθμου είναι να εκφράσει τη διαδικασία επίλυσης ενός προβλήματος με τρόπο κατανοητό. Έτσι λοιπόν υπάρχουν κάποιοι τρόποι έκφρασης (παρουσίασης) αλγορίθμων:

1. Ελεύθερο κείμενο
2. Φυσική γλώσσα με βήματα
3. Ψευδοκώδικας
4. Λογικό διάγραμμα ή διάγραμμα ροής

Εμείς δεν θα ασχοληθούμε με τους τρεις πρώτους αλλά μόνο με το λογικό διάγραμμα παρακάτω.

1.3 Το Λογικό Διάγραμμα.

Στο Λογικό Διάγραμμα (ΛΔ) η αναπαράσταση του αλγορίθμου γίνεται με την βοήθεια διαφόρων συγκεκριμένων σχημάτων. Τα σχήματα που συνήθως χρησιμοποιούνται φαίνονται παρακάτω μαζί με την λειτουργία που αντιπροσωπεύουν:

Αρχή, τέλος



Είσοδος δεδομένων προβλήματος

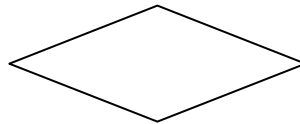


Επεξεργασία

(π.χ. υπολογισμοί αριθμητικών παραστάσεων)



Σύγκριση (π.χ. $a > b$)



Εκτύπωση



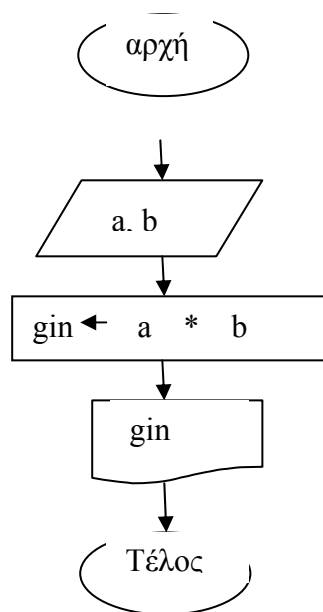
Ροή



Σύνδεση



Παράδειγμα χρήσης σχημάτων ΛΔ για τη δημιουργία αλγορίθμου: Ο παρακάτω αλγόριθμος επιλύει το πρόβλημα της εύρεσης του γινομένου δύο αριθμών (a, b).



1.4 Τι είναι μεταβλητές και τι είναι σταθερές;

Επειδή θα ασχοληθούμε παρακάτω με μεταβλητές, καλό είναι στο σημείο αυτό να πούμε αναφέρουμε μερικά λόγια. Οι μεταβλητές που χρησιμοποιούμε στους αλγορίθμους αλλά και στα προγράμματα είναι περίπου το ίδιο με τις μεταβλητές στα μαθηματικά.

Μια μεταβλητή δεν είναι τίποτε άλλο παρά μια θέση μνήμης στην κύρια μνήμη ενός Η/Υ. Κάθε αλγόριθμος(πρόγραμμα) για να δώσει τη λύση στο πρόβλημα που αντιμετωπίζει θα πρέπει να επεξεργαστεί δεδομένα και να παράξει αποτελέσματα. Τα δεδομένα που δίνονται κατά τη διάρκεια εκτέλεσης του αλγορίθμου (προγράμματος) αλλά και τα αποτελέσματα που παράγονται φυλάσσονται στην κύρια μνήμη του Η/Υ.

Μια μεταβλητή έχει ένα όνομα και μπορεί να πάρει ένα μόνο περιεχόμενο κάθε φορά, το οποίο βεβαίως μπορεί να αλλάζει κατά τη διάρκεια εκτέλεσης ενός προγράμματος, γι' αυτό άλλωστε ονομάζεται και μεταβλητή. Το όνομα για μια μεταβλητή θα πρέπει να επιλέγεται με προσοχή και να είναι σχετικό με τη χρήση της μεταβλητής. Π.χ. Εάν χρειαζόμαστε μια μεταβλητή για να της δώσουμε ως περιεχόμενο την μεγαλύτερη τιμή από ένα πλήθος αριθμών ένα κατάλληλο όνομα θα ήταν το: MAX. Από την άλλη μεριά έχουμε τις σταθερές. Όπως αναφέρει και το όνομα τους δεν μπορούν να αλλάζουν περιεχόμενο.

Με ποιόν τρόπο όμως ορίζονται οι μεταβλητές έτσι ώστε να μπορούν να χρησιμοποιηθούν σε έναν αλγόριθμο (πρόγραμμα).

Με ποιους τρόπους όμως μπορούμε να δώσουμε περιεχόμενο ή να αλλάξουμε το προηγούμενο περιεχόμενο μιας μεταβλητής; Αυτό γίνεται στους αλγόριθμους(προγράμματα) με τη χρήση της εντολής **εκχώρισης τιμής σε μεταβλητή**. Η εντολή αυτή αποτελείται από 2 μέρη, το αριστερό και το δεξί. Διαχωριστικό μεταξύ των 2 μερών είναι το βελάκι. Το αριστερό μέρος είναι πάντοτε μια μεταβλητή. Το δεξί μέρος μπορεί να είναι διάφορα πράγματα π.χ. μια σταθερά (αριθμός ή κείμενο), μια μεταβλητή, μια παράσταση που περιέχει ή όχι μεταβλητές. Σε κάθε περίπτωση υπολογίζεται η τιμή του δεξιού μέρους και δίνεται ως περιεχόμενο στη μεταβλητή που βρίσκεται στο αριστερό μέρος. Π.χ.

1. $a \leftarrow 0$ (ορίζεται η μεταβλητή a και της δίνεται το περιεχόμενο 0)

\leftarrow

2. $a \quad \beta$ (αν η μεταβλητή β έχει περιεχόμενο τον αριθμό 5, τότε το 5 γίνεται περιεχόμενο της μεταβλητής a)

3. $a = \beta + 2$ (αν η μεταβλητή β έχει περιεχόμενο τον αριθμό 5, τότε υπολογίζεται η τιμή της παράστασης $5+2$ δηλαδή 7 και το 7 γίνεται περιεχόμενο της μεταβλητής a)

1.5 Βασικές δομές αλγορίθμων

Για να γίνεται ευκολότερη η αντιμετώπιση της επίλυσης των προβλημάτων με αλγορίθμους και στη συνέχεια με προγράμματα σε κάποια γλώσσα προγραμματισμού, υπάρχουν κάποιες βασικές λογικές δομές αλγορίθμων με τον συνδυασμό των οποίων μπορούμε να επιλύσουμε οποιοδήποτε πρόβλημα (που έχει λύση φυσικά):

1. Η διαδοχή
2. Η επιλογή βάση του ελέγχου κάποιας συνθήκης
3. Η επαναληπτική δομή.

1. Η διαδοχή

Ας ξεκινήσουμε με τη διαδοχή. Κατά τη διαδοχική εκτέλεση εντολών, οι εντολές εκτελούνται η μία μετά την άλλη σειριακά. Δηλαδή:

εντολή 1;

εντολή 2;

...

εντολή n;

Ακολουθεί ένα παράδειγμα που δίνει τη σειριακή εκτέλεση των εντολών:

αλγόριθμος εμβαδό_τραπεζίου (πραγματικός B,β,υ,E)

δεδομένα B,β,υ

αποτελέσματα E

πραγματικός HA

αρχή

HA:=(B+β)/2;

E:=HA*υ;

τέλος

Παράδειγμα αλγόριθμου από την καθημερινή ζωή:

αλγόριθμος για να φτιάξουμε καφέ

δεδομένα νερό, βρύση, μπρίκι, κουζίνα, φλυτζάνι, δοχείο του καφέ

αποτελέσματα φλυτζάνι καφέ

εντολή 1: βάλε το μπρίκι κάτω από τη βρύση;

« « **2:** περίμενε να γεμίσει το μπρίκι;

« « **3:** κλείσε τη βρύση;

« « **4:** άναψε την κουζίνα;

« « **5:** περίμενε να βράσει το νερό;

« « **6:** σβήσε την κουζίνα;

« « **7:** πάρε το δοχείο του καφέ από το ράφι;

« « **8:** βγάλε το καπάκι από το δοχείο του καφέ;

« « **9:** βγάλε ένα κουταλάκι του καφέ;

« « **10:** βάλε το κουταλάκι του καφέ στο φλυτζάνι;

« « **11:** βάλε το καπάκι στο δοχείο του καφέ;

« « **12:** ξαναβάλε το δοχείο του καφέ στο ράφι;

« « **13:** βάλε νερό στο φλυτζάνι μέχρι να γεμίσει;

2. Η επιλογή βάσης του ελέγχου κάποιας συνθήκης

Στην περίπτωση αυτή έχουμε τον έλεγχο μιας συνθήκης, αν η συνθήκη ισχύει (ΝΑΙ) τότε εκτελείται η εντολή(ες)¹ αν δεν ισχύει (ΟΧΙ) εκτελείται η εντολή(ες)²

Όμως στη διαδικασία επίλυσης πολλών προβλημάτων χρειάζεται να αποφασιστεί ποιά θα είναι η επόμενη εντολή η οποία δεν είναι αναγκαστικά η επόμενη στη σειρά εντολών. Έτσι εισάγουμε την απλή επιλογή η οποία είναι σε μορφή ψευδοκώδικα ως εξής:

αν <συνθήκη> **τότε**

ομάδα εντολών 1;

αλλιώς

ομάδα εντολών 2;

τέλος αν

Η εντολή χρησιμοποιεί τέσσερις βοηθητικές λέξεις : αν, τότε, αλλιώς, τέλος αν.

Παράδειγμα βιβλίου

αλγόριθμος Απόλυτη_τιμή (**πραγματικός** x,y)

δεδομένα x

αποτελέσματα y

αρχή

αν $x > 0$ τότε

$y := x;$

αλλιώς

$y := -x;$

τέλοςαν

τέλος

Τρόπος εκτέλεσης εντολής επιλογής:

Αρχικά ελέγχεται η συνθήκη. Αν αυτή ικανοποιείται εκτελείται το σώμα εντολών 1, αλλιώς εκτελείται το σώμα εντολών 2.

Παράδειγμα από τη καθημερινή ζωή:

αν ο υπόγειος έχει απεργία τότε

 πάρε το λεωφορείο του αεροδρομίου

αλλιώς

 πάρε τον υπόγειο σιδηρόδρομο

Παρόμοια εντολή με την απλή επιλογή είναι και η περιορισμένη επιλογή. Η μορφή της σε ψευδοκώδικα είναι η εξής:

αν <συνθήκη> τότε

 ομάδα εντολών 1;

τέλοςαν

Το βιβλίο μας δίνει το εξής παράδειγμα:

αλγόριθμος Απόλυτη_τιμή (**πραγματικός** x)

δεδομένα x

αποτελέσματα x

αρχή

αν $x < 0$ τότε

$x := -x;$

τέλοςαν

τέλος

Παράδειγμα από την καθημερινή ζωή:

 πάρε το βάζο του καφέ από το ράφι

αν το βάζο είναι άδειο τότε

 πάρε το καινούργιο βάζο από το ντουλάπι

Επίσης, εκτός από την απλή επιλογή υπάρχει και η πολλαπλή επιλογή. Είναι δυνατόν σε κάποιο σημείο του προγράμματος οι εντολές να χωριστούν σε περισσότερες από μία ομάδες και να εκτελεστούν οι εντολές μόνο μιας ομάδας. Σε μορφή ψευδοκώδικα η δομή της πολλαπλής επιλογής είναι:

ανάλογα <μεταβλητή>

ίσημε <τιμή1>

ομάδα εντολών 1;

ίσημε <τιμή2>

ομάδα εντολών 2;

...

ίσημε <τιμήn>

ομάδα εντολών n;

τέλοςανάλογα

Ο τρόπος εκτέλεσης της εντολής της πολλαπλής επιλογής είναι ο εξής:

Αρχικά επιλέγεται η τιμή της μεταβλητής. Στη συνέχεια ανάλογα με την τιμή της μεταβλητής εκτελείται η ομάδα εντολών n για n=1,2,3...

Παράδειγμα βιβλίου:

αλγόριθμος Ημέρα_εβδομάδας (ακέραιος i, Λεκτικό Ημέρα)

δεδομένα i

αποτελέσματα Ημέρα

αρχή

ανάλογα i

ίσημε 0

Ημέρα:=“Κυριακή”;

ίσημε 1

Ημέρα:=“Δευτέρα”;

ίσημε 2

Ημέρα:=“Τρίτη”;

ίσημε 3

Ημέρα:=“Τετάρτη”;

ίσημε 4

Ημέρα:=“Πέμπτη”;

ίσημε 5

Ημέρα:=“Παρασκευή”;

ίσημε 0

Ημέρα:=“Κυριακή”;

τέλοςανάλογα

τέλος

Παράδειγμα από την καθημερινή ζωή:

Έστω ότι οι διαχειριστές μιας πολυκατοικίας αλλάζουν κάθε μήνα

ανάλογα τι μήνα έχουμε, αν είναι

ίσος με Ιανουάριο,

διαχειριστής αναλαμβάνει ο κ. Πάρλας

ίσος με Φεβρουάριο,

διαχειριστής αναλαμβάνει ο κ. Μπίσκας

ίσος με Μάρτιο,

διαχειριστής αναλαμβάνει ο κ. Καστίδης

...

ίσος με Δεκέμβριο,

διαχειριστής αναλαμβάνει ο κ. Ιωσηφάκης

Τώρα θα εξηγήσουμε την έννοια της φωλιασμένης επιλογής. Όταν μια εντολή επιλογής εμπεριέχεται σε μία άλλη εντολή επιλογής, τότε καλείται φωλιασμένη επιλογή.

Παράδειγμα βιβλίου:

αλγόριθμος Πρωτοβάθμια_εξίσωση (**πραγματικός** $a, \beta, x,$

Λεκτικό Y)

δεδομένα x, Y

αρχή

αν $a < > 0$ **τότε**

$Y :=$ “υπάρχει μοναδική λύση”;

$x := -\beta/a$;

αλλιώς

αν $\beta < > 0$ **τότε**

$Y :=$ “αδύνατη”;

αλλιώς

$Y :=$ “αόριστη”;

τέλοςαν

τέλοςαν

τέλος

Παράδειγμα από την καθημερινή ζωή:

αν δε λειτουργούν οι σηματοδότες **τότε**
προχώρησε με μεγάλη προσοχή

αλλιώς

αν ο σηματοδότης είναι κόκκινος ή πορτοκαλί **τότε**
σταμάτα

αλλιώς

προχώρησε

3. Επαναληπτική δομή.

Στο σημείο αυτό θα προχωρήσουμε στην επεξήγηση των επαναληπτικών δομών. Κατά την επίλυση διαφόρων προβλημάτων συναντούμε συχνά διαδικασίες κατά τις οποίες ελέγχεται μία συνθήκη και για όσο αυτή ισχύει, επαναλαμβάνεται η εκτέλεση μίας ομάδας εντολών.

Έχουμε τριών ειδών επαναληπτικές δομές:

- i) όσο-κάνε
- ii) επανέλαβε-μέχρι
- iii) για-μέχρι-κάνε
- iv) Φωλιασμένη επανάληψη

Η επαναληπτική δομή όσο-κάνε έχει μορφή σε ψευδοκώδικα:

όσο <συνθήκη-συνέχειας> **κάνε**
ομάδα εντολών;

τέλοςόσο

Στην αρχή της εκτέλεσης αυτής της επαναληπτικής δομής, ελέγχεται η συνθήκη και εφόσον αυτή ισχύει, εκτελείται η ομάδα εντολών. Στην συνέχεια ελέγχεται ξανά η συνθήκη μέχρι να καταστεί ψευδής και τότε λήγει η εκτέλεση της επαναληπτικής δομής.

Παράδειγμα βιβλίου:

αλγόριθμος Πέμπτη_Δύναμη (**πραγματικός** α, P)

δεδομένα α

αποτελέσματα P

ακέραιος i

αρχή

P:=1; i:=1;
όσο (i<=5) **κάνε**
 P:=P*α; i:=i+1;

τέλοςόσο

τέλος

Παράδειγμα από την καθημερινή ζωή:

 πάρε το πρώτο όνομα του καταλόγου
όσο δεν έχουν εξαντληθεί τα ονόματα **κάνε** (το εξής)
 πάρε το επόμενο όνομα
 αν είναι το όνομα που ψάχνεις **τότε**
 τέλος
αλλιώς
 συνέχισε

Η επαναληπτική δομή επανάλαβε-μέχρι σε ψευδοκώδικα:

επανάλαβε
 ομάδα εντολών;
μέχρι <συνθήκη-τέλους>

Στην δομή επανάληψης επανάλαβε-μέχρι η ομάδα εντολών του βρόγχου εκτελείται τουλάχιστον μια φορά. Δηλαδή σε αυτή την δομή ο έλεγχος γίνεται μετά την εκτέλεση της ομάδας εντολών.

Παράδειγμά βιβλίου:

αλγόριθμος Υπολογισμός_ν (**ακέραιος** ν, Π)
δεδομένα Π
αποτελέσματα ν
ακέραιος Σ
αρχή
 ν:=0; Σ:=0;
 επανάλαβε
 ν:=ν+1; Σ:=Σ+ν*ν;
 μέχρι (Π<=Σ)
τέλος

Παράδειγμα από την καθημερινή ζωή:

 πάρε το πρώτο όνομα του καταλόγου
επανάλαβε

αν το όνομα είναι αυτό που ψάχνεις **τότε**

απέσπασε τον αντίστοιχο αριθμό

αλλιώς

πάρε το επόμενο όνομα του καταλόγου

μέχρι το όνομα να είναι αυτό που ψάχνεις ή να τελειώσει ο

κατάλογος

Η επαναληπτική δομή για-μέχρι-κάνε:

Όταν ο αριθμός των επαναλήψεων είναι εκ των προτέρων γνωστός χρησιμοποιούμε αυτή τη δομή ελέγχου. Σε μορφή ψευδοκώδικα η επαναληπτική δομή για-μεχρι-κάνε:

για <μτ>:=<ατ> **μέχρι** <ττ> **μεταβολή** <μβ> **κάνε**

ομάδα εντολών;

Τέλοςγια

(όπου μτ: μετρητής, ατ: αρχική τιμή, ττ: τελική τιμή,

μβ: μεταβολή)

Σε αυτή την επαναληπτική δομή δεν είναι αναγκαίο να αρχικοποιήσουμε και να αυξήσουμε το μετρητή με ειδικές εντολές ανάθεσης.

Παράδειγμα βιβλίου:

αλγόριθμος Πέμπτη_Δύναμη (**πραγματικός** α, P)

δεδομένα α

αποτελέσματα P

ακέραιος i

αρχή

P:=1;

για i:=1 **μέχρι** 5 **μεταβολή** 1 **κάνε**

P:=P*α;

τέλοςγια

τέλος

Παράδειγμα από την καθημερινή ζωή:

για τον πρώτο **μέχρι** τον τελευταίο ένοικο με **μεταβολή** 1 (**κάνε**)

πάρε τα λεφτά του ενοικίου

Τώρα ας δώσουμε την έννοια της φωλιασμένης επανάληψης. Είναι δυνατό στην ομάδα εντολών μιας επαναληπτικής δομής να συνυπάρχουν και άλλες επαναληπτικές δομές.

Ας δώσουμε ένα παράδειγμα:

αλγόριθμος Άθροισμα (ακέραιος n , S)

δεδομένα n

αποτελέσματα S

ακέραιος i, j, P

αρχή

$j:=1;$

$S:=0;$

όσο ($j \leq 100$) **κάνε**

$P:=1;$

$i:=1;$

όσο $i \leq n$ **κάνε**

$P:=P*j;$

$i:=i+1;$

τέλοςόσο

$S:=S+P;$

$j:=j+1;$

τέλοςόσο

τέλος

Μέχρι τώρα είδαμε ότι είναι δυνατό μια δομή επιλογής να είναι φωλιασμένη μέσα σε μια άλλη δομή επιλογής, ενώ μια επαναληπτική δομή να είναι φωλιασμένη σε μια άλλη επαναληπτική δομή. Εν γένει, οποιαδήποτε από τις άλλες δομές μπορεί να φωλιαστεί σε οποιαδήποτε από τις δομές επανάληψης ή επιλογής.

Ας δώσουμε ένα παράδειγμα:

αλγόριθμος καταμέτρηση_θετικών (ακέραιος N , θετικοί)

δεδομένα N

αποτελέσματα θετικοί

ακέραιος i, A

αρχή

θετικοί:=0;

για $i:=1$ **μέχρι** N **κάνε**

διάβασε $A;$

αν ($A > 0$) **τότε**

θετικοί:=θετικοί+1;

τέλοςαν
 τέλοςγια
 τέλος

1.6 Λογικές Συνθήκες

Ας εξηγήσουμε τώρα τις Λογικές συνθήκες στις δομές επανάληψης και επιλογής. Στις δομές επανάληψης και επιλογής οι λογικές συνθήκες παίζουν σημαντικό ρόλο. Μία λογική συνθήκη μπορεί να πάρει μία από τις δύο δυνατές τιμές (αληθής ή ψευδής).

Διακρίνουμε δύο είδη τελεστών:

- A) Αριθμητικοί
- B) Λογικοί

Οι αριθμητικοί τελεστές είναι:

- i) < μικρότερο
- ii) > μεγαλύτερο
- iii) = ίσο
- iv) <= μικρότερο ή ίσο
- v) >= μεγαλύτερο ή ίσο
- vi) != διαφορετικό

Οι λογικοί τελεστές είναι:

- i) Άρνηση
- ii) Διάζευξη
- iii) Σύζευξη

Για να κατανοήσουμε τους λογικούς τελεστές χρησιμοποιούμε την άλγεβρα Boole με 2 τιμές, που ορίζεται πάνω σε ένα σύνολο 2 στοιχείων $B = \{0,1\}$ ή αντίστοιχα $B = \{\text{ψευδής}, \text{αληθής}\}$. Ο ακόλουθος πίνακας συνοψίζει τις λογικές συνθήκες, οι οποίες υλοποιούνται με την άλγεβρα Boole.

A	B	Σύζευξη	Διάζευξη	Άρνηση A
ψευδής	ψευδής	ψευδής	ψευδής	αληθής
ψευδής	αληθής	ψευδής	αληθής	αληθής
αληθής	ψευδής	ψευδής	αληθής	ψευδής
αληθής	αληθής	αληθής	αληθής	ψευδής

Ας δώσουμε ένα παράδειγμα:

Υπολογισμός της παράστασης $Y=x/(x+1)-1/(x-2)$.

αλγόριθμος Παράσταση (πραγματικός x, Y , Λεκτικός Λ)

δεδομένα x

αποτελέσματα Y, Λ

αρχή

αν $x=-1$ ή $x=2$ **τότε**

$\Lambda :=$ “η παράσταση δεν ορίζεται”;

αλλιώς

$Y := x/(x+1)-1/(x-2)$;

τέλοςαν

τέλος

1.7 Διαδικασίες και Συναρτήσεις

Συχνά, στη διαδικασία επίλυσης ενός προβλήματος διαπιστώνουμε ότι μπορούμε να χρησιμοποιήσουμε μεθόδους που μας είναι γνωστές από λύσεις άλλων προβλημάτων. Στις περιπτώσεις αυτές, υλοποιούμε τις λύσεις των υποπροβλημάτων ως υποαλγόριθμους, τους οποίους μπορούμε να χρησιμοποιήσουμε μέσα στον κύριο αλγόριθμο με τις ονομασίες που εμείς βάζουμε. Στη βιβλιογραφία συναντούμε 2 είδη υποαλγορίθμων:

Διαδικασίες

Συναρτήσεις

Η δήλωση της διαδικασίας γίνεται χρησιμοποιώντας τη βοηθητική λέξη διαδικασία, το όνομα που εμείς θέλουμε να βάλουμε στη διαδικασία και τις δηλώσεις των παραμέτρων.

διαδικασία όνομα διαδικασίας (τύποι και ονόματα μεταβλητών)

Για παράδειγμα, η διαδικασία για την εύρεση του μεγίστου μεταξύ δύο αριθμών είναι:

διαδικασία Μέγιστος2 (πραγματικός x, y, M)

δεδομένα x, y

αποτελέσματα M

πραγματικός TEMP

αρχή

αν $x > y$ **τότε**

TEMP := x ;

αλλιώς

TEMP:=y;

τέλοςαν

M:=TEMP;

τέλος

Αυτή η διαδικασία μπορεί να χρησιμοποιηθεί από τον ολοκληρωμένο αλγόριθμο που υπολογίζει το μέγιστο από τρεις αριθμούς:

αλγόριθμος Μέγιστος3

πραγματικός α, β, γ, M, X

διαδικασία Μέγιστος2(πραγματικός, πραγματικός, πραγματικός)

αρχή

διάβασε α,β,γ;

Μέγιστος2(α,β,X);

Μέγιστος2(X,γ,M);

εμφάνισε M;

τέλος

Όμοια, η γενική μορφή της δήλωσης μίας συνάρτησης είναι:

συνάρτηση όνομα συνάρτησης (τύποι και ονόματα μεταβλητών)

Σε αυτήν την περίπτωση ο υποαλγόριθμος επιστρέφει ένα αποτέλεσμα με το ίδιο το όνομα του υποαλγόριθμου.

Ας δούμε ένα παράδειγμα:

συνάρτηση Μέγιστος2 (πραγματικός x, y) : πραγματικός

δεδομένα x, y

αρχή

αν x>y **τότε**

Μέγιστος2:=x;

αλλιώς

Μέγιστος2:=y;

τέλοςαν

τέλος

Ας δούμε και έναν πλήρες αλγόριθμο ο οποίος υπολογίζει το μέγιστο μεταξύ τριών αριθμών, χρησιμοποιώντας τη συνάρτηση Μέγιστος2:

αλγόριθμος Μέγιστος3

πραγματικός α,β,γ,M,X

συνάρτηση

Μέγιστος2(πραγματικός, πραγματικός,πραγματικός)

αρχή

διάβασε α,β,γ;

X:=Μέγιστος2(α,β);

M:=Μέγιστος2(γ,X);

εμφάνισε M;

τέλος

1.8 Πίνακες

Σε πολλά προβλήματα το πλήθος των υπό επεξεργασία δεδομένων είναι μεγάλο. Κατά συνέπεια, η παράστασή τους και η περιγραφή της επεξεργασίας τους είναι πολύπλοκη. Ας θεωρήσουμε ότι καταγράφουμε τις πωλήσεις κάθε μήνα μιας συγκεκριμένης εταιρίας. Για ευκολία χρησιμοποιούμε τη δομή του πίνακα που έχει (για το συγκεκριμένο παράδειγμα) ως εξής:

Μήνας	1	2	3	4	5	6	7	8	9	10	11	12
Πωλήσεις	100	100	110	90	178	95	130	110	200	70	50	155

Ας δώσουμε έναν ορισμό του πίνακα:

Πίνακας είναι ένα σύνολο στοιχείων ίδιου τύπου, καθένα από τα οποία διαθέτει ένα μοναδικό συνδιασμό τιμών διατακτικών αριθμών (δεικτών) που προσδιορίζουν τη θέση τους μέσα στον πίνακα.

Χρησιμοποιούμε τους πίνακες όταν θέλουμε να επεξεργαστούμε ένα πλήθος ομοειδών στοιχείων.

Δήλωση πίνακα σε αλγόριθμο:

πραγματικός πίνακας A[1:10]

πραγματικός πίνακας B[1:5, 1:7]

Ας δούμε τώρα ένα παράδειγμα:

Θεωρούμε ότι θέλουμε να φυλάξουμε τους βαθμούς πέντε μαθητών στη Φυσική, τη Χημεία, τα Μαθηματικά και την Έκθεση, όπως φαίνεται στον πίνακα:

Μαθητής	Φυσική	Μαθ/τικά	Χημεία	Έκθεση
1	17	15	15	17
2	15	12	18	13
3	14	17	15	15
4	17	14	14	14
5	12	16	17	16

Θα χρησιμοποιήσουμε έναν πίνακα με 5 γραμμές για το πλήθος των μαθητών και 4 στήλες για το πλήθος των μαθημάτων:

ακέραιος βαθμός[1:5, 1:4];

ακέραιος i, j;

αρχή

για i:=1 μέχρι 5 κάνε

για j:=1 μέχρι 4 κάνε

διάβασε βαθμός[i, j];

τέλοςγια

τέλοςγια

τέλος

Έτσι ο πίνακας-βαθμός θα περιέχει τις ακόλουθες τιμές:

βαθμός [μαθητής, μάθημα]

τιμή

[1,1]	[1,2]	[1,3]	[1,4]	17	15	15	17
[2,1]	[2,2]	[2,3]	[2,4]	15	12	18	13
[3,1]	[3,2]	[3,3]	[3,4]	14	17	15	15
[4,1]	[4,2]	[4,3]	[4,4]	17	14	14	14
[5,1]	[5,2]	[5,3]	[5,4]	12	16	17	16

Τεχνικές αναζήτησης στοιχείων μέσα σε πίνακες

Στο σημείο αυτο θα δούμε δύο τεχνικές αναζήτησης στοιχείων μέσα σε πίνακες:

την σειριακή αναζήτηση

την δυαδική αναζήτηση

Ξεκινάμε με την σειριακή αναζήτηση. Έστω ένας γραμμικός πίνακας με N στοιχεία. Δίνουμε τον αλγόριθμο της σειριακής αναζήτησης στο παρακάτω παράδειγμα:

αλγόριθμος Σειριακή_Αναζήτηση_1 (ακέραιος N,K , **πραγματικός** L ,
πραγματικός πίνακας $A[1:N]$, **λογικός** f)

δεδομένα N,A,L

αποτελέσματα K

ακέραιος i

αρχή

$K:=0;$

$f:=FALSE;$

για $i:=1$ **μέχρι** N **κάνε**

αν $A[i]:=L$ **τότε**

$K:=i;$

$f:=TRUE;$

τέλοςαν

τέλοςγια

τέλος

Πρέπει να παρατηρήσουμε ότι η επαναληπτική δομή συνεχίζει ακόμα και αν βρεθεί το στοιχείο που ψάχνουμε.

Ας περάσουμε τώρα στη δεύτερη τεχνική, αυτήν της δυαδικής αναζήτησης. Σε αυτήν την περίπτωση, θεωρούμε ότι ο πίνακας είναι ταξινομημένος ώστε να χρησιμοποιήσουμε την αντίστοιχη τεχνική. Η βασική ιδέα είναι η εξής:

Έστω ότι ψάχνουμε σε έναν τηλεφωνικό κατάλογο για να ελέγξουμε εάν υπάρχει ένα τυχαίο όνομα και ότι έχουμε όλα τα ονόματα σε έναν πίνακα. Πηγαίνουμε στο μέσο του πίνακα και αν το όνομα είναι αυτό που ψάχνουμε, η αναζήτηση σταματά. Εάν δεν είναι αυτό, ψάχνουμε στη συνέχεια στο μισό του πίνακα όπου το όνομα θα βρίσκεται αλφαβητικά. Έτσι διαμερίζεται ο πίνακας σε κομμάτια έως ότου να βρούμε το όνομα ή να έχουμε εξαντλήσει την αναζήτηση και να μην υπάρχει το όνομα.

Δίνεται στη συνέχεια ένα αλγοριθμικό παράδειγμα δυαδικής αναζήτησης:

αλγόριθμος Δυαδική_αναζήτηση (ακέραιος N,K , **πραγματικός** S ,
πραγματικός πίνακας $A[1:N]$, **λογικός** f)

δεδομένα N,A,S

αποτελέσματα K,f

ακέραιος Left,Right,M

αρχή

Left:=1; Right:=N, K:=0; f:=FALSE;

όσο (Left<=Right) και (f=FALSE) **κάνε**

M:=(Left+Right)/2;

αν A[M]=S **τότε**

K:=M; f:=TRUE;

αλλιώς

αν A[M]<S **τότε**

Left:=M+1;

αλλιώς

Right:=M-1;

τέλοςαν

τέλοςαν

τέλοςόσο

τέλος

Πρέπει να τονίσουμε ότι ο αλγόριθμος της δυαδικής αναζήτησης είναι μεν ταχύτερος από τον αλγόριθμο της σειριακής αναζήτησης, αλλά πρέπει ο πίνακας που θα πραγματοποιήσει την αναζήτηση να είναι ταξινομημένος.

2 ΕΙΣΑΓΩΓΗ ΣΤΗΝ MATLAB

2.1 Εισαγωγή

Το MATLAB είναι ένα λογισμικό πακέτο για υψηλής απόδοσης αριθμητικούς υπολογισμούς. Παρέχει στο χρήστη ένα διαδραστικό περιβάλλον με χιλιάδες ενσωματωμένες συναρτήσεις, κατάλληλες για την υλοποίηση απαιτητικών υπολογιστικών αναλύσεων, γραφημάτων καθώς επίσης και για την παραγωγή διαφόρων *animations*. Επιπλέον, το MATLAB προσφέρει τη δυνατότητα επέκτασης σε ποικίλα πεδία εφαρμογών με τη αξιοποίηση την υψηλού επιπέδου γλώσσας προγραμματισμού, την οποία διαθέτει σε όλες τις εκδόσεις του. Για λόγους πληρότητας, να αναφερθεί ότι το όνομα MATLAB προέρχεται από τις λέξεις MATrix και LABoratory.

Το MATLAB αποτελεί ένα εξελιγμένο υπολογιστικό εργαλείο, το οποίο μπορεί να βρει εφαρμογή σε διάφορους τομείς της επιστήμης αλλά βέβαια και της πράξης, όπως για παράδειγμα τη μηχανική, την ιατρική, τις θετικές επιστήμες (Μαθηματικά – Φυσική), την οικονομία καθώς και γενικά τη βιομηχανική παραγωγή. Μάλιστα, το φάσμα των εφαρμογών του συγκεκριμένου πακέτου λογισμικού διευρύνεται συνεχώς και περισσότερο, αναδεικνύοντας με αυτό τον τρόπο τις πολλαπλές δυνατότητες του, όπως:

- Υψηλή απόδοση και ταχύτητα υπολογιστικών αναλύσεων.
- Δυνατότητα προσομοίωσης φυσικών συστημάτων.
- Δυνατότητα υλοποίησης αλγορίθμων.
- Δυνατότητα αμφίδρομης επικοινωνίας με πληθώρα άλλων προγραμμάτων και εφαρμογών.
- Υψηλής ποιότητας γραφικές απεικονίσεις και *animations*.
- Δυνατότητα σύνδεσης με διάφορες συσκευές καταγραφής.
- Φιλικότητα προς το χρήστη και διαδραστικός χαρακτήρας.

Επίσης παρέχονται από το MATLAB πολλές προαιρετικές εργαλειοθήκες, οι οποίες προορίζονται για την ανάπτυξη ειδικών εφαρμογών, όπως συμβολικοί υπολογισμοί (*symbolic computation*), επεξεργασία εικόνων (*image processing*), στατιστική (*statistics*), σχεδιασμός ελέγχου συστημάτων (*control system design*), νευρωνικά δίκτυα (*neural networks*), ασαφή λογική (*fuzzy logic*). Η λίστα με τις διαθέσιμες εργαλειοθήκες συνεχώς διευρύνεται (στην τελευταία έκδοση του λογισμικού ο αριθμός των εργαλειοθηκών έχει ξεπεράσει τις 50).

2.2 Χρήσιμες ιδιότητες του MATLAB

Το MATLAB διαθέτει έναν έντονα διαδραστικό χαρακτήρα, λαμβάνοντας τα απαραίτητα δεδομένα από το περιβάλλον εργασίας και συγκεκριμένα από το παράθυρο εντολών και εξάγοντας στο χρήστη τα αποτελέσματα των υπολογισμών είτε εκ νέου από το παράθυρο εντολών είτε, αν πρόκειται για διαγράμματα, από το παράθυρο των διαγραμμάτων. Εκτός όμως από τις προαναφερόμενες δυνατότητες, το MATLAB μπορεί να εισάγει δεδομένα και από εξωτερικά αρχεία καθώς επίσης και να εξάγει αποτελέσματα σε διαφορετικά του MATLAB αρχεία.

Με οποιονδήποτε τρόπο και αν γίνει η εισαγωγή και η εξαγωγή των αρχείων, υπάρχουν ορισμένα συγκεκριμένα χαρακτηριστικά, τα οποία διέπουν τη διαδικασία αυτή.

2.3 Βασικοί τύποι αρχείων του MATLAB

Το MATLAB μπορεί να αναγνώσει και να εγγράψει διάφορα είδη αρχείων.

M-files: Είναι τυπικά ASCII text files, συνοδευόμενα από την προέκταση **.m** στο όνομα του αρχείου. Υπάρχουν δύο βασικές κατηγορίες τέτοιων αρχείων: script files και function files (βλ. §3.1 και §3.2). Τα περισσότερα προγράμματα, τα οποία συντάσσονται μέσω του MATLAB αποθηκεύονται ως M-files. Το σύνολο των ενσωματωμένων συναρτήσεων είναι τύπου M-files. Μάλιστα μερικές ενσωματωμένες συναρτήσεις παρέχονται από το λογισμικό στο χρήστη με τον πηγαίο κώδικα σε αναγνώσιμα M-files, με αποτέλεσμα να μπορεί ο κώδικας αυτός να αντιγραφεί ή ακόμα και να τροποποιηθεί.

2.4 Βασικά στοιχεία χρήσης της Matlab

Απλοί υπολογισμοί μέσω MATLAB

- *Πράξεις μεταξύ αριθμών*
- *Καταχώρηση τιμής σε μεταβλητή*
- *Μαθηματικές και άλλες συναρτήσεις*
- *Μιγαδικοί αριθμοί*

Τέλος μια σημαντική παρατήρηση για την εκμετάλλευση των δυνατοτήτων του MATLAB είναι η χρήση του συμβόλου (%). Οτιδήποτε βρίσκεται δεξιά του συμβόλου αυτού, δε λαμβάνεται υπόψη από την υπολογιστική μηχανή του λογισμικού, με αποτέλεσμα να

χρησιμοποιείται κατά κόρον για τη εγγραφή απαραίτητων σχολίων και παρατηρήσεων, οι οποίες διευκολύνουν το χρήστη.

Βασικές μαθηματικές συναρτήσεις	
<i>sqrt</i>	υπολογίζει τη τετραγωνική ρίζα ενός αριθμού
<i>abs</i>	υπολογίζει την απόλυτη τιμή ενός αριθμού
Τριγωνομετρικές συναρτήσεις	
<i>sin</i>	υπολογίζει το ημίτονο μιας γωνίας
<i>cos</i>	υπολογίζει το συνημίτονο μιας γωνίας
<i>tan</i>	υπολογίζει την εφαπτομένη μιας γωνίας
<i>cot</i>	υπολογίζει τη συνεφαπτομένη μιας γωνίας
<i>asin</i>	υπολογίζει το τόξο ημίτονου ενός αριθμού
<i>Acos</i>	υπολογίζει το τόξο συνημίτονου ενός αριθμού
<i>Atan</i>	υπολογίζει το τόξο εφαπτομένης ενός αριθμού
<i>Acot</i>	υπολογίζει το τόξο συνεφαπτομένης ενός αριθμού
Εκθετικές και λογαριθμικές συναρτήσεις	
<i>Exp</i>	Υπολογίζει την παράσταση e^x
<i>Log</i>	Υπολογίζει το νεπέριο λογάριθμο ενός αριθμού
<i>log10</i>	Υπολογίζει το δεκαδικό

λογάριθμο ενός αριθμού

2.5 Προγραμματίζοντας στο matlab

Ένα ιδιαίτερο χαρακτηριστικό του MATLAB είναι η δυνατότητα της απλής προέκτασης του σε ποικίλες εφαρμογές, μέσω της σύνταξης από το χρήστη αυτόνομων κωδίκων και προγραμμάτων. Το λογισμικό προσφέρει στο χρήστη τη χρήση μιας ενσωματωμένης γλώσσας προγραμματισμού, η οποία διαθέτει κοινά χαρακτηριστικά με τη γλώσσα προγραμματισμού C. Μάλιστα, δεν είναι υπερβολικό να θεωρηθεί ότι η συγκεκριμένη γλώσσα προγραμματισμού του MATLAB ίσως είναι καταλληλότερη και υψηλότερου επιπέδου για εφαρμογές Μηχανικού, οι οποίες απαιτούν ισχυρά υπολογιστικά εργαλεία, κάτι το οποίο είναι δύσκολο να επιτευχθεί από τις κλασσικές γλώσσες προγραμματισμού.

Στο MATLAB ο χρήστης μπορεί να αναπτύξει κώδικα στα M-Files. Τα M-Files είναι τυπικά ASCII αρχεία κειμένου (*text files*), συνοδευόμενα από την προέκταση .m στο όνομα του αρχείου και στα οποία ο χρήστης κάνει χρήση της γλώσσας προγραμματισμού του λογισμικού. Τα περισσότερα M-Files είναι δυνατό να ενεργοποιηθούν μέσω οποιοδήποτε προγράμματος επεξεργασίας κειμένου (*editor or word processing application*). Ένα τέτοιο πρόγραμμα διαθέτει και το παρόν

λογισμικό (*M-file Editor*). Υπάρχουν δύο είδη M-files. Τα αρχεία κειμένου ή script files και τα αρχεία συναρτήσεων ή function files.

ΕΙΣΑΓΩΓΗ – ΕΞΑΓΩΓΗ ΔΕΔΟΜΕΝΩΝ ΣΤΟ MATLAB

Το MATLAB παρέχει τη δυνατότητα εισαγωγής με διάφορους τρόπους δεδομένων, τα οποία είτε είναι αποθηκευμένα σε κάποιο δίσκο-μονάδα καταγραφής είτε απλώς βρίσκονται στον σκληρό δίσκο του Η/Υ. Η διαδικασία αυτή ονομάζεται εισαγωγή δεδομένων (*importing data*). Μια άλλη δυνατότητα, η οποία προσφέρεται στο χρήστη είναι αυτή της εξαγωγής αποτελεσμάτων από το παρόν λογισμικό μέσω της αποθήκευσης των απαραίτητων δεδομένων σε κάποιο εξωτερικό αρχείο, το οποίο εν συνεχεία μπορεί να επεξεργαστεί από άλλο πρόγραμμα. Η διαδικασία αυτή ονομάζεται εξαγωγή δεδομένων (*exporting data*). Ο τρόπος με τον οποίο ο χρήστης θα επιλέξει είτε να εισάγει είτε να εξαγάγει τα δεδομένα, τα οποία επιθυμεί, εξαρτάται τόσο από το είδος της επιτελούμενης εφαρμογής όσο και από το είδος της μορφοποίησης του αρχείου εισαγωγής ή εξαγωγής. Τα αρχεία αυτά μπορεί να είναι αρχεία κειμένου (*text files*) η/και *binary* αρχεία.

Οι δυνατές μορφές (*format*) των δεδομένων, οι οποίες μπορούν να συναντηθούν σε διάφορα αρχεία, είναι οι ακόλουθες:

Δεδομένα κειμένου (*Text data*): Στα διάφορα αρχεία κειμένου, τα δεδομένα συνήθως αποτυπώνονται σύμφωνα με τον *American Standard Code for Information Interchange (ASCII)*, ο οποίος υποστηρίζει αριθμητικούς και αλφαριθμητικούς χαρακτήρες. Τα *ASCII* δεδομένα κειμένου μπορούν να προβληθούν μέσω ενός προγράμματος επεξεργασίας κειμένου (*text editor*).

***Binary data*:** Στη *binary* μορφοποίηση των δεδομένων, τα δεδομένα δεν μπορούν να αποτυπωθούν μέσω του *ASCII* και επίσης δεν είναι δυνατό να έχει ο χρήστης προσπέλαση σε αυτά μέσω προγραμμάτων επεξεργασίας κειμένων. Τα *binary* αρχεία είναι πιθανό να περιέχουν δεδομένα, τα οποία έχουν να κάνουν με εικόνες, ήχους και άλλες πολλές μορφές. Στα διάφορα αρχεία κειμένου, τα δεδομένα συνήθως αποτυπώνονται σύμφωνα με τον *American Standard Code for*

Information Interchange (ASCII), ο οποίος υποστηρίζει αριθμητικούς και αλφαριθμητικούς χαρακτήρες. Τα *ASCII* αρχεία κειμένου μπορούν να προβληθούν μέσω ενός προγράμματος επεξεργασίας κειμένου (*text editor*).

Για τις δύο προαναφερθείσες μορφές δεδομένων, το MATLAB παρέχει διάφορες συναρτήσεις, κατάλληλες για να υλοποιήσουν τόσο την εισαγωγή δεδομένων σε αυτό όσο και την εξαγωγή αποτελεσμάτων από το ίδιο το λογισμικό, έτοιμα να αξιοποιηθούν από άλλες εφαρμογές και προγράμματα. Αξίζει να σημειωθεί ότι όσο αφορά την εισαγωγή των δεδομένων, το MATLAB διαθέτει και ένα οδηγό εισαγωγής (*Import Wizard*), ο οποίος σε αρκετές περιπτώσεις μπορεί να αποδειχθεί η απλούστερη και ταχύτερη λύση σε σχέση με τη χρήση συναρτήσεων για την εισαγωγή δεδομένων.

Τέλος να επισημανθεί ότι το λογισμικό υποστηρίζει την εισαγωγή-εξαγωγή δεδομένων μέσω τυπικών συναρτήσεων, οι οποίες έχουν αναπτυχθεί στο πλαίσιο της γλώσσας προγραμματισμού C (*standard C-language file I/O function*) και βρίσκουν εφαρμογή και για τις δύο μορφές δεδομένων, οι οποίες παρουσιάστηκαν προηγούμενα. Μάλιστα, αναφορικά με αυτό το τελευταίο αυτό τρόπο εισαγωγής και εξαγωγής δεδομένων από το λογισμικό, θα αποδοθεί με τη χρήση ενός πλήρους παραδείγματος, το οποίο μάλιστα υλοποιείται από ένα *script file* και βρίσκεται στο τέλος των σημειώσεων, στο Παράρτημα Γ.

2.6 Εισαγωγή δεδομένων στο Matlab

Το MATLAB διαθέτει πληθώρα από συναρτήσεις, οι οποίες προορίζονται για την εισαγωγή δεδομένων. Η επιλογή κάθε φορά της συνάρτησης εισαγωγής εξαρτάται από τη μορφοποίηση, την οποία παρουσιάζουν τα δεδομένα, στα αρχεία, όπου είναι αποθηκευμένα. Όσο αφορά τα δεδομένα κειμένου (είτε αριθμητικά είτε αλφαριθμητικά), αυτά πρέπει να είναι μορφοποιημένα σε ένα ομοιόμορφο μοτίβο από σειρές και στήλες, χρησιμοποιώντας ένα χαρακτήρα (*delimiter*), ο οποίος να διαχωρίζει τα δεδομένα μεταξύ τους. Ο χαρακτήρας αυτός μπορεί να είναι το κενό, το σύμβολο του κόμματος (,), το αντίστοιχο του ερωτηματικού (;) ή οποιοδήποτε άλλο σύμβολο. Επιπλέον, τα αρχεία κειμένου, τα οποία περιέχουν τα δεδομένα κειμένου, είναι δυνατό να εμφανίζουν και ορισμένες σειρές από αλφαριθμητικά δεδομένα, στις οποίες καταγράφονται διάφοροι τίτλοι και κεφαλίδες των στηλών με τα αριθμητικά δεδομένα.

Για τα *binary data* δεν υπάρχουν σαφείς κανόνες και επιτρεπόμενες μορφοποιήσεις, όπως περιγράφηκε προηγούμενα για τα αρχεία κειμένου. Αυτό το γεγονός συμβαίνει διότι τα αρχεία αυτά είναι δυνατό να εμπεριέχουν πολλών ειδών δεδομένα (εικόνες, animations, ήχους).

Στη συνέχεια ακολουθούν δύο συγκεντρωτικοί πίνακες, οι οποίοι περιγράφουν τις παρεχόμενες από το λογισμικό συναρτήσεις για την εισαγωγή είτε *text data* είτε *binary data*.

Δεδομένα κειμένου		
Συναρτήσεις	Τύπος δεδομένων	delimiters
cvsread	Αριθμητικά δεδομένα	Μόνο το σύμβολο του κόμματος (;)
dlmread	Αριθμητικά δεδομένα	Οποιοσδήποτε χαρακτήρας
fscanf	Αριθμητικά δεδομένα	Οποιοσδήποτε χαρακτήρας
load	Αριθμητικά και αλφαριθμητικά δεδομένα	Μόνο το κενό (<i>space</i>)
textread	Αριθμητικά και αλφαριθμητικά δεδομένα	Οποιοσδήποτε χαρακτήρας

Binary δεδομένα		
Συναρτήσεις	Κατάληξη αρχείων	Μορφοποίηση δεδομένων
auread	.au	sound data in SUN Microsystems format
aviread	.avi	audio-visual data in AVI format
hdf	.hdf	data in Hierarchcal Data Format
imread	.jpeg, .tiff, .bmp, .png, .hdf, .pcx, .xwd, .gif, .ico, .cur	image data in many formats
load	.mat	MATLAB data in Mat-files format
wavread	.wav	sound data in Microsoft Windows format
wk1read	.wk1	data in Lotus 123 spreadsheet format
xlsread	.xls	data in Microsoft Excel spreadsheet format

Πετρούκη Μ. - Πετρούλακη Κ.

Μια από τις πλέον χρησιμοποιούμενες συναρτήσεις για την εισαγωγή δεδομένων κειμένου στο MATLAB αποτελεί η συνάρτηση **load**. Το ιδιαίτερο χαρακτηριστικό της εντολής αυτής είναι ότι μπορεί να εισάγει δεδομένα από αρχεία κειμένου, τα οποία περιέχουν είτε αριθμητικά είτε αλφαριθμητικά δεδομένα, χωριζόμενα μεταξύ τους με κενό.

filename □ όνομα εισαγόμενου αρχείου δεδομένων

format αποδίδεται με ένα συγκεκριμένο χαρακτήρα και έχει να κάνει με τον αριθμό, το είδος και των μεταβλητών, που επιστρέφει στο MATLAB. Για την όσο το δυνατό καλύτερη κατανόηση αυτού του ορίσματος, παρατίθεται στη συνέχεια από τη Βοήθεια του λογισμικού το αντίστοιχο λήμμα.

N αποδίδεται με ένα θετικό αριθμό, μεγαλύτερο του μηδενός και σχετίζεται με πλήθος των φορών, τις οποίες το σύστημα θα χρησιμοποιήσει το χαρακτήρα του **format**.

parameter – value □ προσαρμόζεται κάθε φορά η χρήση της συνάρτησης **textread**, με τη χρήση των δύο αυτών ορισμάτων. Η αναλυτική περιγραφή των ορισμάτων ακολουθεί μέσω του αντίστοιχου λήμματος της Βοήθειας του MATLAB.

2.7 Εισαγωγή δεδομένων μέσω Import Wizard

Όπως έχει προαναφερθεί η εισαγωγή δεδομένων στο MATLAB είναι δυνατό να επιτευχθεί εκτός από τις συναρτήσεις εισαγωγής, μέσω του παρεχόμενου οδηγού του λογισμικού, του *Import Wizard*. Η διαδικασία είναι σχετικά απλή και έχει αρκετές ομοιότητες με την αντίστοιχη διαδικασία *Open*, η οποία λαμβάνει χώρα σε πολλές εφαρμογές στο

περιβάλλον των *Windows*. Τα απαιτούμενα βήματα για την προαναφερόμενη διαδικασία συνοψίζεται στα ακόλουθα βήματα:

Από τη βασική εργαλειοθήκη του MATLAB: *File -> Import data*

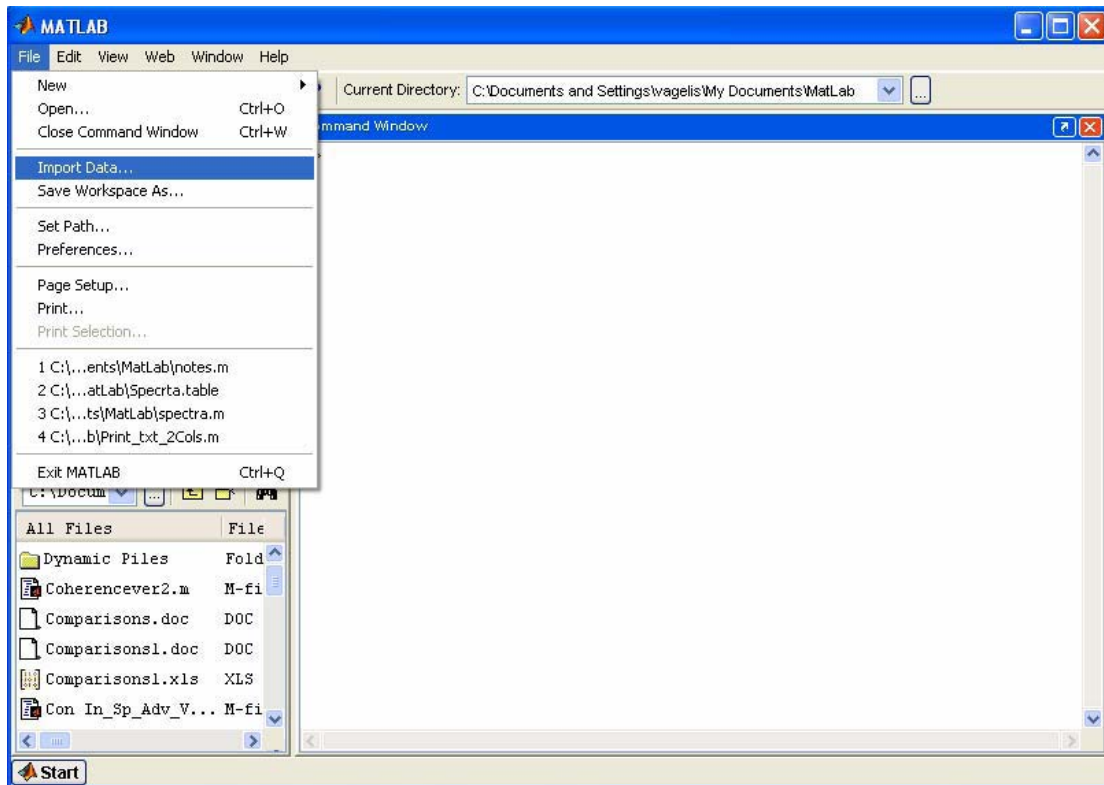
Αναζήτηση του επιθυμητού αρχείου στο κατάλληλο *directory* και εν συνεχεία επιλογή – άνοιγμα (*Open*) του εν λόγω αρχείου.

Ενεργοποίηση του *Import Wizard*. Στο παράθυρο εμφανίζεται η προεπισκόπηση του αρχείου, το οποίο επιλέχθηκε για άνοιγμα. Επίσης προσφέρεται η δυνατότητα μέσω αυτού του πλαισίου διαλόγου να προσδιοριστεί από το χρήστη αφενός ο χαρακτήρας (σύμβολο), το οποίο επιτελεί το διαχωρισμό των δεδομένων του αρχείου και αφετέρου ο αριθμός των τυχόντων σειρών στην αρχή του αρχείου, οι οποίες δεν πρέπει να συμπεριληφθούν στα εισαγόμενα δεδομένα. Μετά τους απαραίτητους χειρισμούς στο παρών παράθυρο, ο χρήστης προχωρά στην επιλογή *Next*.

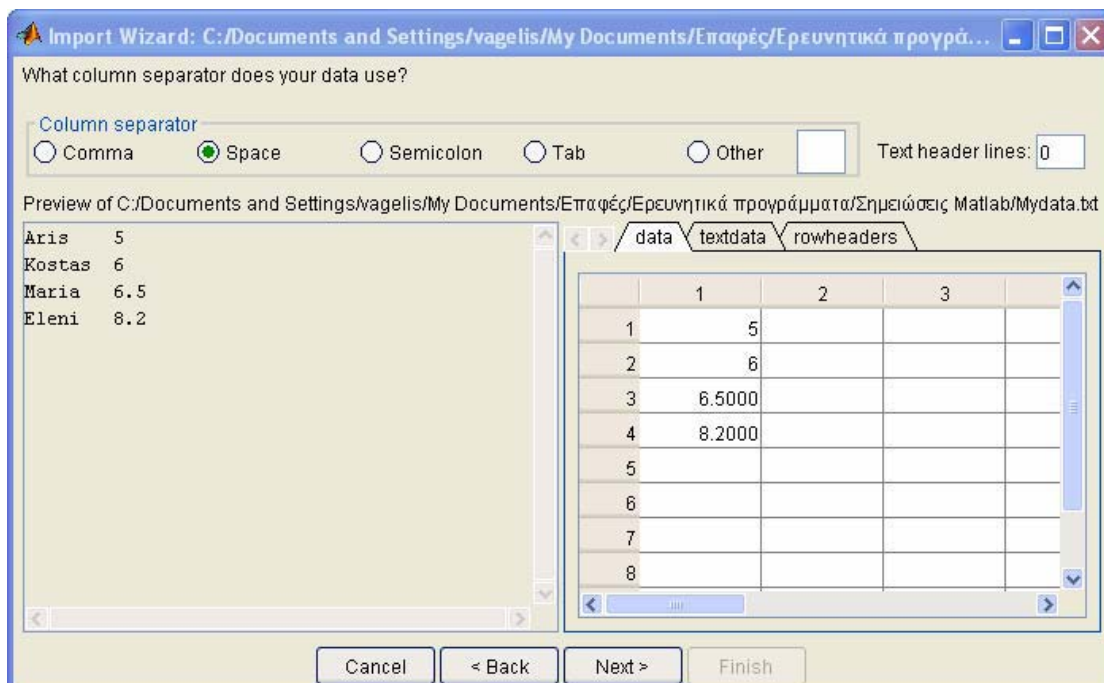
Επιλογή των δεδομένων – μεταβλητών, τις οποίες ο χρήστης επιθυμεί να εισάγει μέσα στο χώρο εργασίας (*Workspace*). Εξ' ορισμού, το λογισμικό τοποθετεί το σύνολο των αριθμητικών δεδομένων σε μια μεταβλητή και τα υπόλοιπα αλφαριθμητικά δεδομένα σε διαφορετικές μεταβλητές. Μετά τους απαραίτητους χειρισμούς στο παρών παράθυρο, ο χρήστης προχωρά στην επιλογή *Finish*.

Το σύνολο των παραπάνω βημάτων για τη διαδικασία της εισαγωγής των δεδομένων, συνοψίζεται στο ακόλουθο παράδειγμα, με τα αντίστοιχες εικόνες.

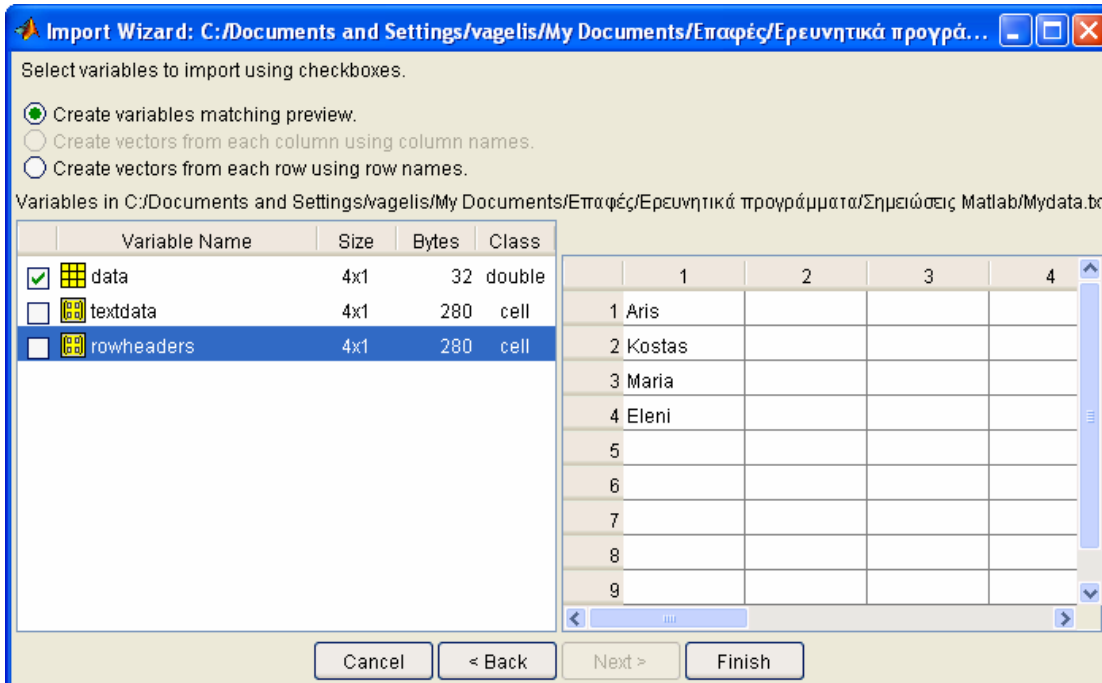
File -> Import Data και Άνοιγμα του αρχείου Mydata.txt



Ενεργοποίηση του Import Wizard. Επιλογή του κενού ως διαχωριστικού μέσου των δεδομένων στο αρχείο Mydata.txt. Στη συνέχεια, ο χρήστης επιλέγει *Next*.

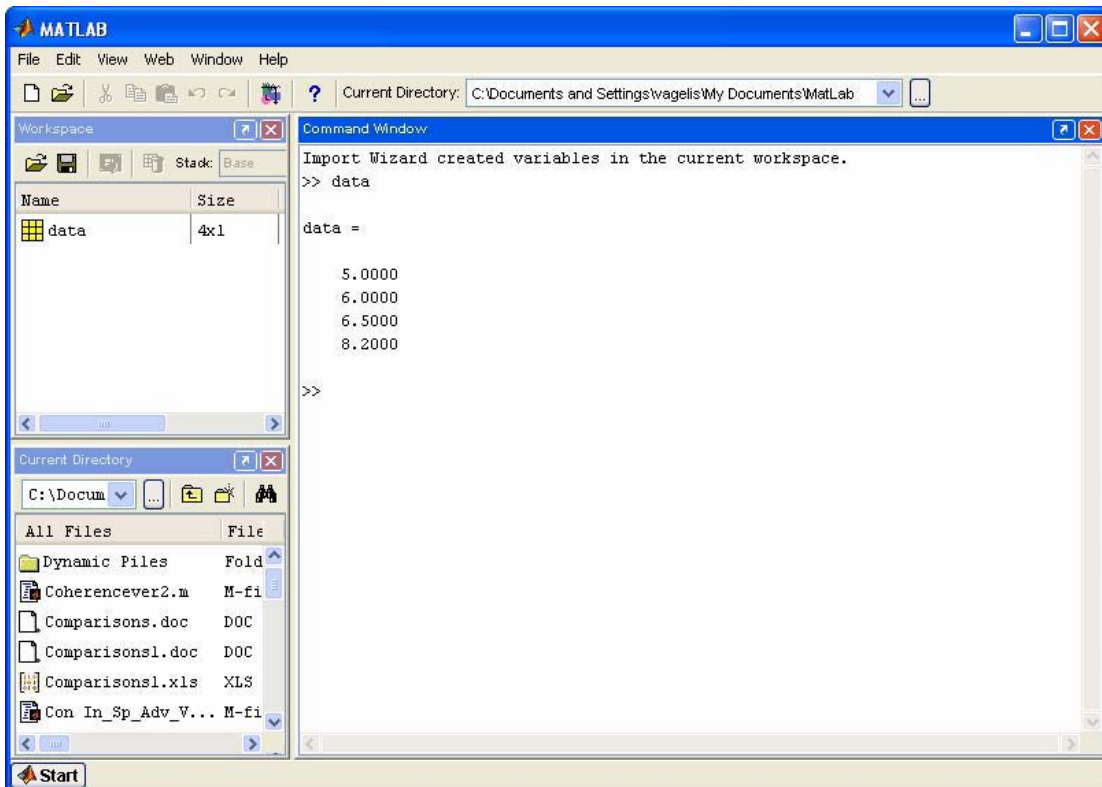


Επιλογή των αριθμητικών δεδομένων του αρχείου Mydata.txt για την εισαγωγή στο χώρο εργασίας του MATLAB. Το λογισμικό θέτει τα δεδομένα αυτά ως τιμές της αριθμητικής μεταβλητής data. Στη συνέχεια, ο χρήστης επιλέγει Finish.



Εικόνα: Import Wizard – Step 2

Εισαγωγή των αριθμητικών δεδομένων στο χώρο εργασίας του λογισμικού και εμφάνιση αυτών στο παράθυρο εντολών (*Command Window*).



Να σημειωθεί ότι η δυνατότητα με το Import Wizard μπορεί να εφαρμοσθεί είτε σε περίπτωση εισαγωγής αρχείου κειμένου είτε για αρχείο binary.

2.8 Εξαγωγή δεδομένων από το Matlab

Λεξιλόγιο κειμένων		
Συναρτήσεις	Τύπος δεδομένων	delimiters
csvwrite	Αριθμητικά δεδομένα	Μόνο το σύμβολο του κόμματος (;)
dlmwrite	Αριθμητικά δεδομένα	Οποιοσδήποτε χαρακτήρας
fprintf	Αριθμητικά και αλφαριθμητικά δεδομένα	Οποιοσδήποτε χαρακτήρας
save	Αριθμητικά δεδομένα	Μόνο το κενό (<i>space</i>) ή τα Tabs
diary	Αριθμητικά δεδομένα ή διατάξεις κελιών	Οποιοσδήποτε χαρακτήρας

Binary δεδομένα		
Συναρτήσεις	Κατάληξη αρχείων	Μορφοποίηση δεδομένων
auwrite	.au	sound data in SUN Microsystems format
avifile	.avi	audio-visual data in AVI format
hdf	.hdf	data in Hierarchcal Data Format
imwrite	.jpeg, .tiff, .bmp, .png, .hdf, .pcx, .xwd, .gif, .ico, .cur	image data in many formats
save	.mat	MATLAB data in Mat-files format
wk1write	.wk1	sound data in Microsoft Windows format
wavwrite	.wav	data in Lotus 123 spreadsheet format

Το MATLAB, στην περίπτωση της εισαγωγής των δεδομένων, έτσι και σε αυτή της εξαγωγής δεδομένων παρέχει πληθώρα από συναρτήσεις, των οποίων η επιλογή βασίζεται κυρίως στη μορφοποίηση των δεδομένων, τα οποία είναι προς εξαγωγή.

Η εξαγωγή των δεδομένων αναφέρεται σε δεδομένα-αρχεία κειμένου (*text data*) αλλά σε *binary data*. Όσο αφορά στα δεδομένα κειμένου, αυτά χαρακτηρίζονται από τα ίδια σύμβολα διαχωρισμού των δεδομένων, όπως συνέβαινε στην περίπτωση της εισαγωγής των δεδομένων.

Στη συνέχεια παρουσιάζονται δύο συγκεντρωτικοί πίνακες, οι οποίοι περιγράφουν τις παρεχόμενες από το λογισμικό συναρτήσεις για την εξαγωγή είτε *text data* είτε *binary data*.

Χρησιμοποιώντας τη συνάρτηση *save*, η εξαγωγή του διανύσματος, αποτελούμενο από τα ακόλουθα δεδομένα:

1 2 3 5 6

7 8 9 10 11

σε αρχείο κειμένου με όνομα Myexport.txt, υλοποιείται ως εξής:

```
>> A=[1 2 3 5 6;7 8 9 10 11];    □ εκτέλεση συνάρτησης εξαγωγής δεδομένων
```

```
>> save Myexport.out A -ASCII
```

Με τη διαδικασία αυτή παράγεται το αρχείο Myexport.out, το οποίο περιέχει τα δεδομένα στην ακόλουθη μορφή:

```
1.0000000e+000 2.0000000e+000 3.0000000e+000 5.0000000e+000 6.0000000e+000
7.0000000e+000 8.0000000e+000 9.0000000e+000 1.0000000e+001 1.1000000e+001
```

Αντίθετα, αν η εξαγωγή του ίδιου διανύσματος πραγματοποιηθεί μέσω της συνάρτησης **dlmwrite**, τότε προκύπτει ελαφρώς αλλαγμένο στη μορφοποίηση αρχείο αποτελεσμάτων (Myexport.out).

```
>> A=[1 2 3 5 6;7 8 9 10 11];    □ εκτέλεση συνάρτησης εισαγωγής δεδομένων
```

```
>> dlmwrite('Myexport.out',A,',';')
```

Το αρχείο εμφανίζει την ακόλουθη μορφή:

```
1;2;3;5;6 7;8;9;10;11
```

Εξ' ορισμού αν ο χρήστης δεν ορίσει το σύμβολο του διαχωρισμού των δεδομένων (*delimiter*), το σύστημα χρησιμοποιεί το σύμβολο του κόμματος (,) για το διαχωρισμό των δεδομένων. Επίσης είναι δυνατό να οριστεί το κενό σα διαχωριστικό (' ') και στην περίπτωση, κατά την οποία ανάμεσα στα δύο εισαγωγικά (*quotes*,') δεν υπάρχει ούτε το προαναφερόμενο κενό, τότε ο χρήστης δεν ορίζει κανένα διαχωριστικό.

Τέλος, η εξαγωγή συνήθως μικρών διανυσμάτων – μητρώων από το χώρο εργασίας του MATLAB, μπορεί να χρησιμοποιηθεί η συνάρτηση **diary**. Η χρήση της για την εξαγωγή του ίδιου με προηγούμενα διανύσματος, παρουσιάζεται ακολούθως:

```
>> diary Myexport.out    ενεργοποίηση της συνάρτησης diary εμφάνιση του προς εξαγωγή διανύσματος
```

```
>> A=[1 2 3 5 6;7 8 9 10 11]
```

```
A=1 2 3 5 6    τερματισμός της εκτέλεσης της συνάρτησης diary
```

```
7 8 9 10 11
```

```
>> diary off
```

Το αρχείο, το οποίο προκύπτει από την παραπάνω διαδικασία, περιέχει τα εξής:

```
A=[1 2 3 5 6;7 8 9 10 11]
```

```
A =
```

```
1 2 3 5 6
```

```
7 8 9 10 11
```

```
diary off
```

Σβήνοντας τα περιττά στοιχεία, μπορεί ο χρήστης να λάβει το επιθυμητό αποτέλεσμα.

2.9 Αρχεία κειμένου (script files)

Τα αρχεία κειμένου ή *script files* αποτελούν ένα σύνολο από εντολές του MATLAB. Τα αρχεία αυτά εκτελούνται μέσα από το παράθυρο εντολών (*Command window*), πληκτρολογώντας το όνομα του αρχείου, χωρίς την προέκταση *.m*, αριστερά από το σύμβολο της προτροπής (prompt). Όλα αυτή η διαδικασία είναι ισοδύναμη με την πληκτρολόγηση μιας προς μιας του συνόλου των εντολών, οι

οποίες περιέχονται μέσα στο εν λόγω αρχείο κειμένου, στο παράθυρο των εντολών. Οι μεταβλητές, οι οποίες χρησιμοποιούνται μέσα στους κώδικες των *script files* έχουν καθολικό χαρακτήρα (*global variables*), με αποτέλεσμα να καταγράφονται στο πλαίσιο του παραθύρου του χώρου εργασίας (*workspace*), τόσο οι μεταβλητές, οι οποίες χρησιμοποιούνται- 41 - ως δεδομένα του προβλήματος όσο και αυτές, οι οποίες φέρουν τα αποτελέσματα.

Τα αρχεία κειμένου είναι δυνατό να περιέχουν αριθμούς, μεταβλητές, εντολές και συναρτήσεις είτε του ίδιου του MATLAB (ενσωματωμένες στο λογισμικό) είτε αναπτυγμένες από το χρήστη. Από τα προηγούμενα προκύπτει το συμπέρασμα ότι τα *script files* είναι χρήσιμα στις περιπτώσεις, όπου ο χρήστης πρέπει να επαναλάβει μια συγκεκριμένη ροή εντολών και διαφορετικές κάθε φορά τιμές για τα δεδομένα.

Κανόνας 1ος: Απαγορεύεται το όνομα, το οποίο θα δοθεί στο M-file να είναι το ίδιο με το όνομα της μεταβλητής, η οποία υπολογίζεται μέσω του αρχείου κειμένου. Αν η μεταβλητή διαθέτει το ίδιο όνομα με το *script file*, τότε το MATLAB δεν θα είναι σε θέση να ενεργοποιήσει το αρχείο αυτό

Κανόνας 2ος: Το όνομα του αρχείου κειμένου πρέπει να αρχίζει με ένα γράμμα. Το υπόλοιπο τμήμα του ονόματος μπορεί να περιέχει εκτός από χαρακτήρες γραμμάτων, αριθμούς όπως επίσης και το σύμβολο της κάτω παύλας (*_*). Το όνομα του M-file είναι δυνατό να έχει όσους χαρακτήρες επιθυμεί ο χρήστης, το λογισμικό όμως λαμβάνει υπόψη του μόνο τους πρώτους 19. Τέλος δεν αποτελεί αποδεκτό σύμβολο από το Matlab η χρήση της τελείας (*.*) σε θέση άλλη εκτός από το σημείο πριν την κατάληξη του αρχείου (*.m*)

Κανόνας 3ος: Πρέπει να δίνεται προσοχή στο γεγονός της καταγραφής των μεταβλητών ενός αρχείου κειμένου στο παράθυρο του χώρου εργασίας (*Workspace*). Όταν εκτελείται ένα M-file, τότε όλες οι μεταβλητές, οι οποίες χρησιμοποιούνται, καταγράφονται στο αντίστοιχο

χώρο του Matlab, με αποτέλεσμα να είναι σκόπιμη η πραγματοποίηση ελέγχου για τυχούσα χρήση κάποιου από τα ονόματα των μεταβλητών αυτών, σε άλλες εφαρμογές.

Κανόνας 4ος: Προτείνεται ο σταδιακός έλεγχος του κώδικα, ο οποίος αναπτύσσεται μέσω ενός αρχείου κειμένου, πριν τη συνολική εφαρμογή του. Αυτό βέβαια αποσκοπεί στην αποτελεσματικότερη και χωρίς λάθη και εφαρμογή του συγκεκριμένου προγράμματος, κάτι το αποτελεί βασικό κανόνα συνολικά του προγραμματισμού και όχι μόνο του MATLAB. Η αναλυτική παρουσίαση, η οποία αφορά στην ανάπτυξη και εκτέλεση ενός αρχείου κειμένου, πραγματοποιείται για τις ανάγκες του υπολογιστή του φάσματος σχεδιασμού, το οποίο προτείνει ο Ε.Α.Κ. 2003.

2.10 Αρχεία συναρτήσεων (function files)

Ένα αρχείο συνάρτησης (*function file*) είναι επίσης ένα M-file, όπως άλλωστε και τα προαναφερθέντα αρχεία κειμένου (*script file*), με μόνη διαφορά το γεγονός ότι οι μεταβλητές στο αρχείο συνάρτησης είναι οπτικά ορισμένες (local) και όχι όπως στα αρχεία κειμένου, όπου οι μεταβλητές είναι καθολικής εμβέλειας. Τα αρχεία συνάρτησης είναι ουσιαστικά σαν τα προγράμματα ή καλύτερα τις υπό ρουτίνες, οι οποίες δημιουργούνται στο πλαίσιο των γνωστών γλωσσών προγραμματισμού όπως η Fortran Visual basic η C κ.α. Αναμφισβήτητα τα αρχεία είναι από τα πιο ευρέως χρησιμοποιούμενα κομμάτια του Matlab

Ένα αρχείο συνάρτησης πρέπει οπωσδήποτε να ξεκινά με τη γραμμή, όπου ορίζεται η συνάρτηση και συγκεκριμένα καταγράφονται αναλυτικά τόσο οι μεταβλητές εισαγωγής όσο και αυτές της εξαγωγής των αποτελεσμάτων. Χωρίς αυτή την εισαγωγική γραμμή, της οποίας η σύνταξη έχει την ακόλουθη μορφή, το αρχείο συνάρτησης μετατρέπεται σε αρχείο κειμένου

(*script* *file*).

Να σημειωθεί ότι το όνομα της συνάρτησης (*function _name*) πρέπει να ταυτίζεται με το όνομα του αρχείου M-file, χωρίς βέβαια την κατάληξη .m, το οποίο περιέχει την εν λόγω συνάρτηση. Για παράδειγμα, αν το όνομα της συνάρτησης είναι *spectra*, τότε και το όνομα του αρχείου συνάρτησης, το οποίο περιέχει τη συνάρτηση αυτή, πρέπει να είναι *spectra.m*. Ο ακριβής τρόπος σύνταξης της εισαγωγικής γραμμής εξαρτάται από αν υπάρχει εξαγόμενο αποτέλεσμα, δηλαδή αν συντρέχει λόγος για ορισμό της μεταβλητής των αποτελεσμάτων, και αν πράγματι συντρέχει ο παραπάνω λόγος, από το πλήθος των απαραίτητων μεταβλητών αποτελέσματος.

Να επισημανθεί ότι η πρώτη λέξη στη γραμμή εισαγωγής πρέπει οπωσδήποτε να είναι πληκτρολογημένη με πεζά γράμματα (*function*). Το αντίθετο αποτελεί ένα από τα συνηθέστερα λάθη στη χρήση του λογισμικού αυτού.

Ορισμένα κρίσιμα χαρακτηριστικά των αρχείων συναρτήσεων είναι τα ακόλουθα:

Οι γραμμές σχολίων, οι οποίες πάντα εκκινούν με το σύμβολο (%), μπορούν να βρίσκονται οπουδήποτε μέσα στο σώμα του αρχείου της συνάρτησης, χωρίς να λαμβάνονται υπόψη από το λογισμικό.

Το σύνολο των γραμμών των σχολίων, οι οποίες ακολουθούν αμέσως μετά από την εισαγωγική γραμμή, προβάλλονται από το MATLAB, αν η Βοήθεια του λογισμικού αναζητήσει τη συγκεκριμένη συνάρτηση. Μάλιστα, η πρώτη σειρά των σχολίων, η επονομαζόμενη *HI Line*, καταγράφεται σε ένα ειδικό αρχείο, το οποίο δημιουργείται αυτόματα στο directory του αρχείου συνάρτησης με το όνομα *contents.m*.

Αν στη γραμμή συνάρτησης ορίζεται μια μόνο μεταβλητή αποτελέσματος, τότε δεν απαιτείται η μεταβλητή αυτή να βρίσκεται εντός αγκύλων []. Στις περιπτώσεις μάλιστα, κατά τις οποίες δεν υπάρχει καμιά μεταβλητή αποτελέσματος, τότε τόσο οι αγκύλες όσο και το σύμβολο (=) είναι δυνατό να παραλειφθούν.

Οι μεταβλητές εισαγωγής είναι, όπως έχει προαναφερθεί τοπικά ορισμένες το συγκεκριμένο αρχείο συνάρτησης, δηλαδή ισχύουν μόνο για τη συγκεκριμένη συνάρτηση. Το γεγονός αυτό αποτελεί και τη μεγαλύτερη διαφορά των αρχείων συναρτήσεων και των αρχείων κειμένου.

Μετά την εκτέλεση της συνάρτησης, οι μόνες μεταβλητές, οι οποίες παραμένουν καταγεγραμμένες στο παράθυρο του χώρου εργασίας (*Workspace*), είναι οι μεταβλητές των αποτελεσμάτων. Αυτή η ισότητα καθώς και η άμεσα προηγούμενη παρέχει τη δυνατότητα έτσι ώστε να ελέγχονται με πιο απλό και αποτελεσματικό τρόπο τόσο τα δεδομένα όσο και τα αποτελέσματα μιας συνάρτησης.

Η εκτέλεση μιας συνάρτησης στο παράθυρο των εντολών του MATLAB μπορεί να πραγματοποιηθεί με διάφορους τρόπους, κάτι το οποίο παρουσιάζεται ευθύς αμέσως, κάνοντας χρήση ενός απλού παραδείγματος.

2.11 Λογικοί τελεστές και σχεσιακοί τελεστές

Το MATLAB διαθέτει τρεις λογικούς τελεστές, όπως αυτοί παρουσιάζονται στον ακόλουθο πίνακα, συνοδευόμενες από σύντομες ερμηνείες. Βέβαια στη μαθηματική λογική υπάρχει και τέταρτος λογικός τελεστής, το αποκλειστικό ή. Στο συγκεκριμένο λογισμικό ο τελευταίος αυτός λογικός τελεστής είναι η συνάρτηση *xor*.

&	Λογικό ναι
	Λογικό ή
~	Λογικό όχι
xor	Αποκλειστικό ή

2.12 Δομές ελέγχου ροής προγράμματος

Οι δομές ελέγχου ροής του προγράμματος είναι σύνολα εντολών, τα οποία απαιτούν ειδικό τρόπο σύνταξης. Στη συνέχεια της τρέχουσας ενότητας, θα παρουσιαστούν εν συντομία οι πιο συχνά εφαρμοζόμενες δομές ελέγχου στο Matlab.

Δομή for loops

Ο βρόγχος for υλοποιεί την κατ' επανάληψη εκτέλεση μιας ή και περισσότερων εντολών για ένα προκαθορισμένο αριθμό επαναλήψεων. Η γενική της μορφή είναι η εξής:

For δείκτης-μετρητής

Σύνολο εντολών

End

Να σημειωθεί ότι το σύνολο των εντολών του βρόγχου είναι δυνατό να εμπεριέχει μια οποιαδήποτε εντολή του λογισμικού, ακόμα και τον ίδιο το βρόγχο του for.

Δομή while

Η δομή **while** περιέχει μέσα στο σώμα της εντολές, οι οποίες εκτελούνται όσο η συνθήκη, την οποία ελέγχει, είναι αληθής. Η δομή αυτή συνήθως χρησιμοποιείται στην περίπτωση κατά την οποία δεν είναι προκαθορισμένος εκ των προτέρων ο αριθμός των επαναλήψεων, οι οποίες απαιτούνται για την ικανοποίηση της συνθήκης. Συγκεκριμένα, η λειτουργία της καταρχήν έγκειται στον έλεγχο για το αν η συνθήκη είναι αληθής και αν ικανοποιείται ο έλεγχος αυτός τότε εκτελούνται όλες οι εντολές, οι οποίες βρίσκονται εντός της δομής. Εν συνεχεία, πραγματοποιείται εκ νέου έλεγχος αληθείας της συνθήκης και στην περίπτωση κατά την οποία εξακολουθεί να ικανοποιείται, εξακολουθεί και αντίστοιχα η εκτέλεση των εντολών. Στην αντίθετη περίπτωση (η συνθήκη είναι πλέον ψευδής) η εκτέλεση των εντολών διακόπτεται.

Η γενική μορφή της δομής είναι η ακόλουθη:

While συνθήκη

Σύνολο εντολών

End

Δομή if

Με τη δομή if είναι δυνατό να εκτελεστούν επί μέρους εντολές, ανάλογα με τις συνθήκες, οι οποίες ικανοποιούνται σε κάθε περίπτωση. Η γενική μορφή της δομής είναι η ακόλουθη:

If συνθήκη 1

Σύνολο εντολών 1

elseif συνθήκη2

Σύνολο εντολών2

:

:

Else if συνθήκη n

Σύνολο εντολών n

Else

Σύνολο εντολών(n+1)

End

2.13 Γραφικά στο Matlab

Το MATLAB διαθέτει αξιόλογα εργαλεία για την απόδοση γραφημάτων, είτε στο επίπεδο είτε στο χώρο, ακόμα για την παραγωγή animations. Τυπικά γραφήματα στις δύο διαστάσεις, εξεζητημένα διαγράμματα και απεικονίσεις στις τρεις διαστάσεις με δυνατότητες για ιδιαίτερους χρωματισμούς και σκιάσεις, επεξεργασία και όλων των ειδών των γραφημάτων και εικόνων, μέσω των εργαλείων επεξεργασίας του λογισμικού είναι ορισμένες από τις εφαρμογές, στις οποίες μπορεί να δώσει λύσεις η χρήση του **MATLAB**

Το πιο βασικό πλεονέκτημα του λογισμικού αυτού είναι η απλότητα στο χειρισμό του συνόλου των ιδιοτήτων και χαρακτηριστικών των γραφημάτων, είτε γράφοντας κομμάτια κώδικα είτε μέσα από ένα φιλικό-διαδραστικό περιβάλλον εργασίας για τα γραφήματα.

Γραφικά στο επίπεδο

Βασικότερη και ευρέως χρησιμοποιούμενη συνάρτηση γραφικών είναι η συνάρτηση *plot*.

Με τη χρήση της συνάρτησης αυτής υλοποιούνται γραφικές παραστάσεις γραμμών (ευθειών και καμπύλων).

Η σύνταξη της εντολής αυτής για τα γραφήματα είναι η εξής :

`Plot (x values, yvalues, “style option”)`

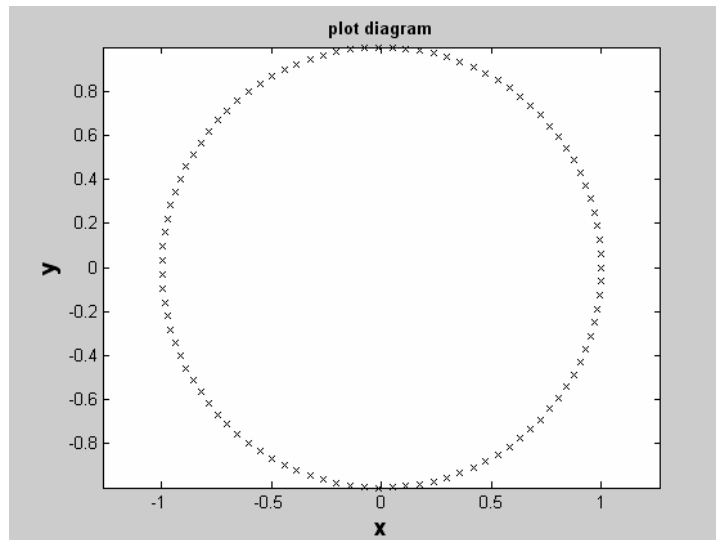
Εντολές υλοποίησης διαφόρων μορφών γραφημάτων στο επίπεδο

Στην ενότητα αναφέρονται διάφοροι τύποι – μορφές διαγραμμάτων στο επίπεδο, τα οποία παράγονται χρησιμοποιώντας τις δυνατότητες, τις οποίες προσφέρει το MATLAB.

Συνάρτηση *plot*

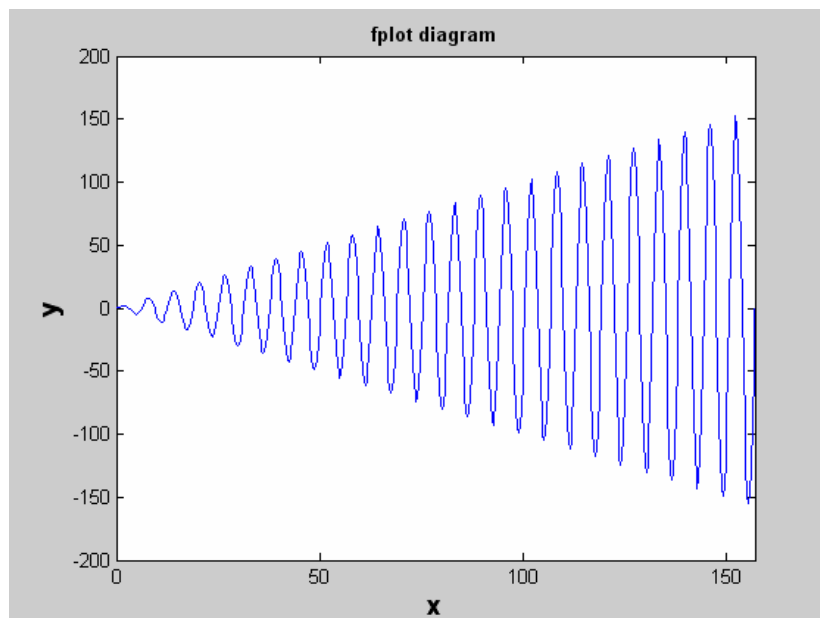
```
>> a=linspace(0,2*pi,100); □
>> x=cos(a); □
>> y=sin(a); □
>> plot(x,y,'kx') □
>>axis('equal') □
>>title('Circle of unit radius','Fontweight','bold') □
>>xlabel('x','FontSize',14,'FontWeight','demi') □
>>ylabel('y','FontSize',14,'FontWeight','demi')
```

ορισμός μεταβλητής a
ορισμός διανύσματος x
ορισμός διανύσματος y
εντολή παραγωγής γραφήματος
εντολή επεξεργασίας πλαισίου γραφήματος
εντολή επεξεργασίας τίτλου γραφήματος
εντολές επεξεργασίας τίτλων αξόνων γραφήματος



Συνάρτηση *fplot*

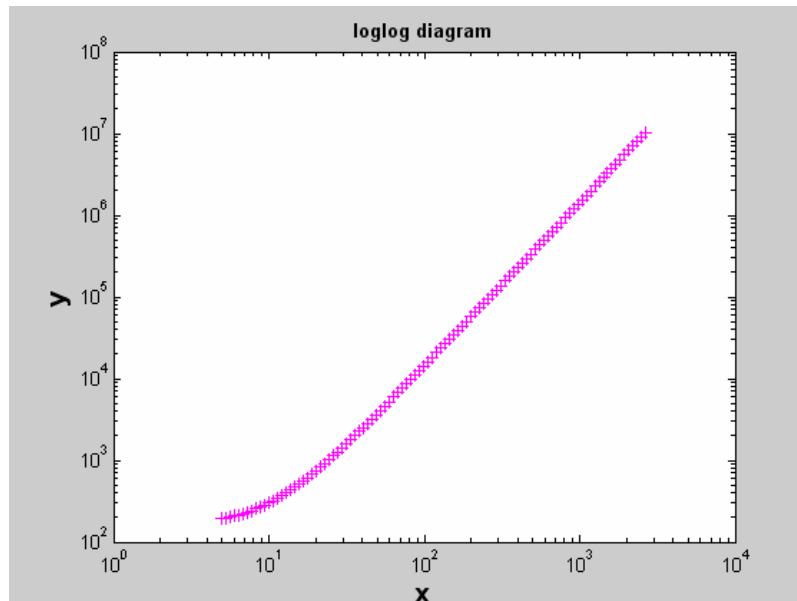
```
>>fplot('x*sin(x)',[0 50*pi]) □
>>ylabel('y','FontSize',14,'FontWeight','demi') □
>>xlabel('x','FontSize',14,'FontWeight','demi') □
>>title('fplot diagram','Fontweight','bold')
□
```



Συνάρτηση *log log*

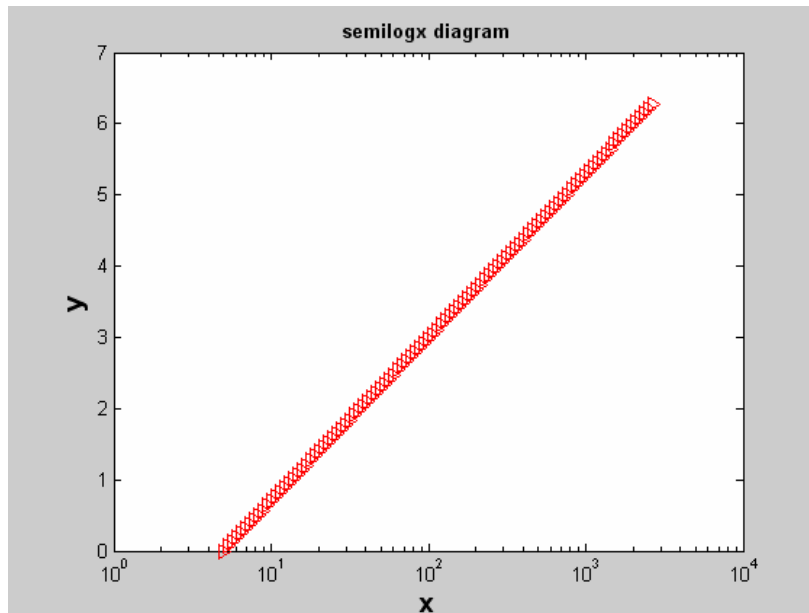
```
>> z=linspace(0,2*pi,100); □
```

```
>> x=5*exp(z); □
>> y=155+36*exp(2*z); □
>> loglog(x,y,'m+') □
>> title('loglog diagram','Fontweight','bold') □
>> ylabel('y','FontSize',14,'FontWeight','bold')
>> xlabel('x','FontSize',14,'FontWeight','bold')
```



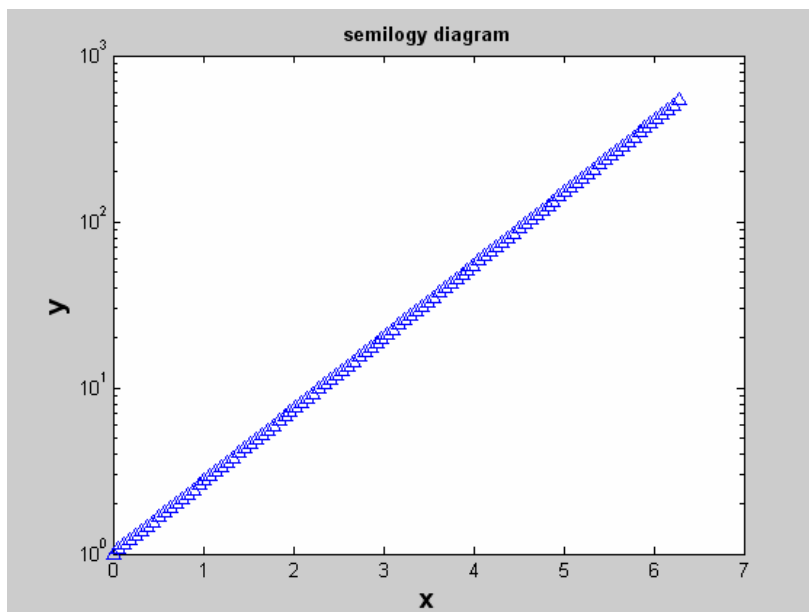
Συνάρτηση *semilogx*

```
>> z=linspace(0,2*pi,100); □
>> x=5*exp(z); □
>> y=z; □
>> semilogx(x,y,'r>') □
>> xlabel('x','FontSize',14,'FontWeight','bold') □
>> ylabel('y','FontSize',14,'FontWeight','bold') □
>> title('semilogx diagram','Fontweight','bold')
```



Συνάρτηση *semilogy*

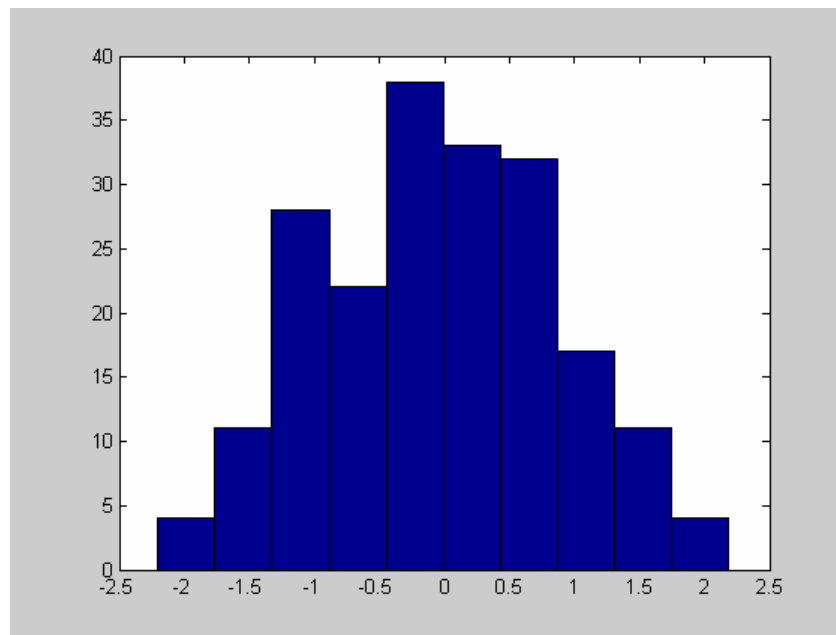
```
> z=linspace(0,2*pi,100); □
>> semilogy(z,exp(z),'b^') □
>> title('semilogy diagram','Fontweight','bold') □
>> ylabel('y','FontSize',14,'FontWeight','bold') □
>> xlabel('x','FontSize',14,'FontWeight','bold')
```



Συνάρτηση *hist*

```
>> z=randn(200,1); □
```

>> hist(z)



Συνάρτηση *barh*

```
>> magnitude=char('<=6.25','6.25-6.5','6.5-6.75',... □
```

```
'6.75-7.00','7.00-7.25','7.25-7.50','7.50-7.75'); □
```

```
>> earthquakes=[16;13;13;15;9;2;1]; □
```

```
>> barh(earthquakes) □
```

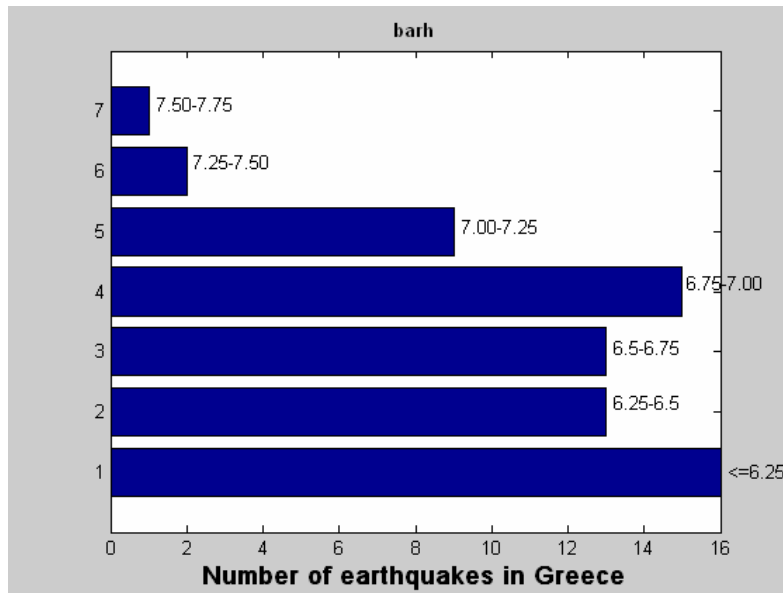
```
>> for i=1:7, □
```

```
gtext(magnitude(i,:)); □
```

```
end □
```

```
>> xlabel('Number of earthquakes in Greece','FontSize',14,'FontWeight','demi') □
```

```
>> Title('barh','Fontweight','bold')
```



Συνάρτηση *pie*

```
>> magnitude=char('<=6.25','6.25-6.5','6.5-6.75',... □
```

```
'6.75-7.00','7.00-7.25','7.25-7.50','7.50-7.75');
```

```
>> earthquakes=[16;13;13;15;9;2;1]; □
```

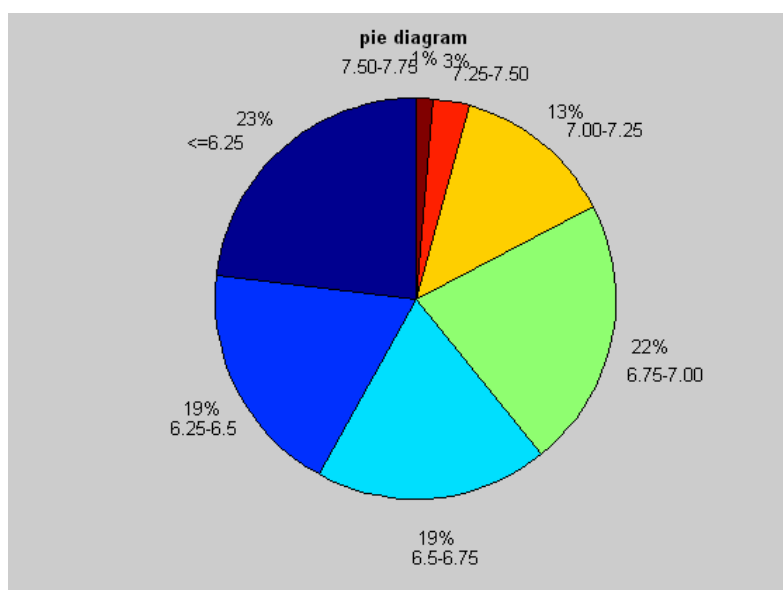
```
>> pie(earthquakes) □
```

```
>> for i=1:7, □
```

```
gtext(magnitude(i,:));
```

```
end
```

```
>> Title('pie diagram','Fontweight','bold')
```

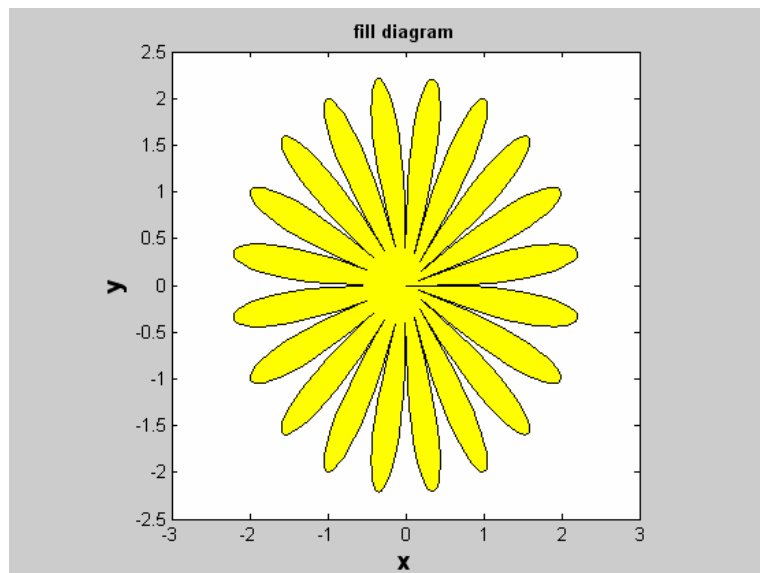


Συνάρτηση *fill*

```

>> z=linspace(0,2*pi,500); □
>> r=sqrt(abs(5*sin(10*z))); □
>> x=r.*cos(z); □
>> y=r.*sin(z); □
>> fill(x,y,'y') □
>> title('fill diagram','Fontweight','bold') □
>> xlabel('x','FontSize',14,'FontWeight','demi') □
>> ylabel('y','FontSize',14,'FontWeight','demi') □
>> axis('square')

```

**2.14 Εντολές υλοποίησης διαφόρων μορφών γραφημάτων στο χώρο**

Υπάρχουν διάφορες μορφές διαγραμμάτων στο χώρο, τα οποία παράγονται χρησιμοποιώντας τις αντίστοιχες συναρτήσεις του MATLAB.

Συνάρτηση *plot3*

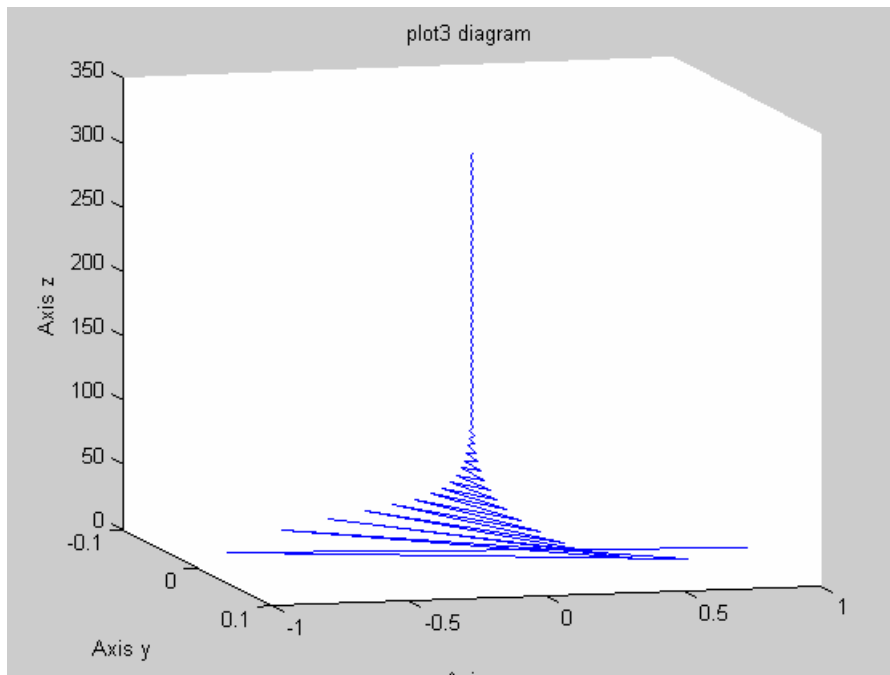
```

>> z=linspace(0,100*pi); □
>> c=exp(-2*z/32); □
>> x=c.*cos(z); □
>> y=c.*sin(z); □

```

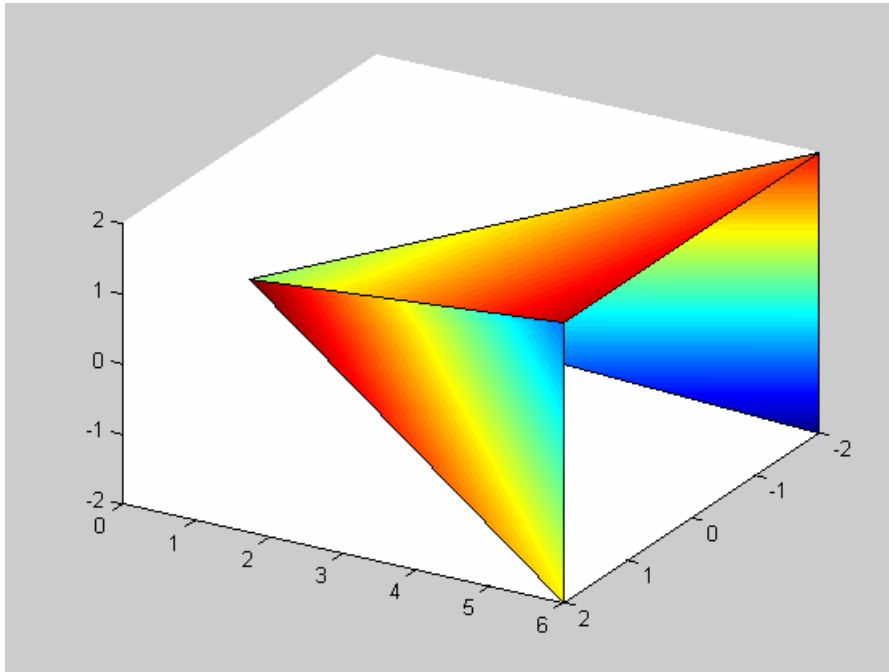


```
>> plot3(x,y,z) □
>> view(15,-10)
>> title('plot3 diagram')
>> xlabel('Axis x')
>> ylabel('Axis y') □
>> zlabel('Axis z')
```



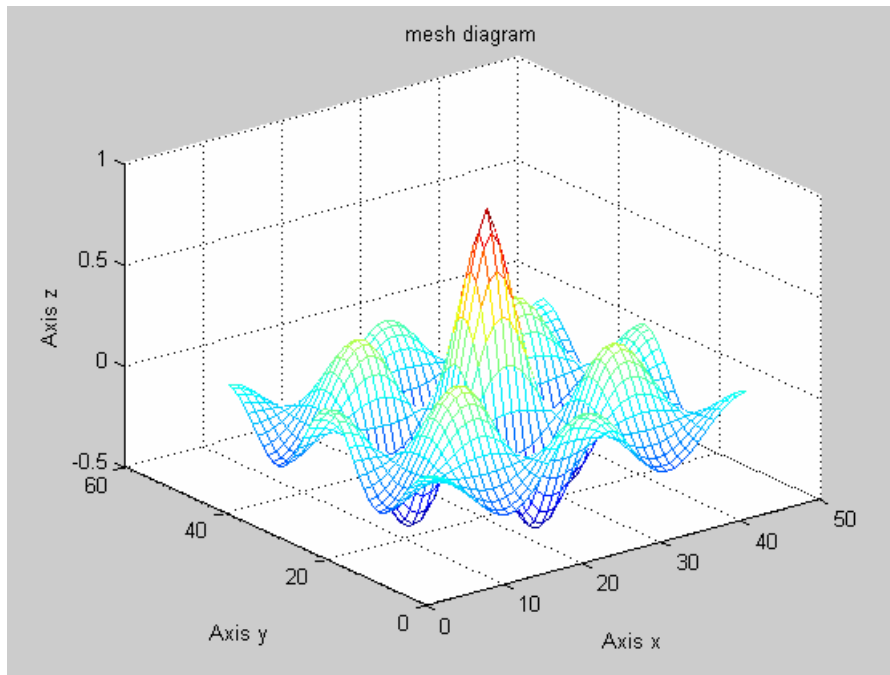
Συνάρτηση *fill3*

```
>> X=[0 0 0 0;2 2 -2 -2;...
2 -2 -2 -2;]; □
>> Y=[0 0 0 0;6 6 6 6;...
6 6 6 6]; □
>> Z=[0 0 0 0;2 2 -2 -2;...
-2 2 2 -2]; □
>> fillcolor=rand(3,4); □
>> fill3(X,Y,Z,fillcolor)
>> view(120,35)
```



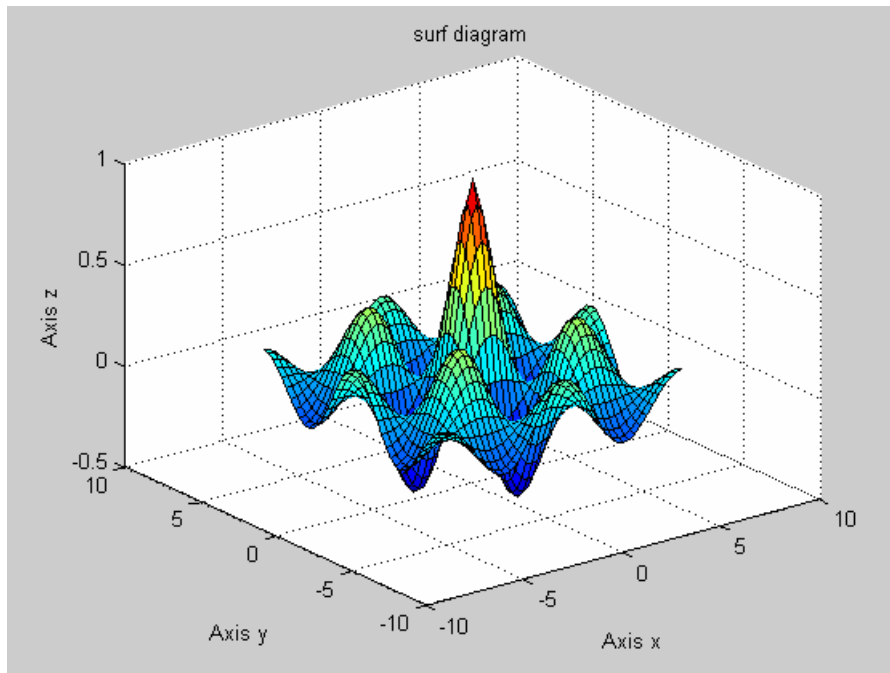
Συνάρτηση *mesh*

```
>> a=-6:0.3:6;
>> [X,Y]=meshgrid(a,a); □
>> Z=cos(X).*cos(Y).*...
exp(-sqrt(X.^2+Y.^2)/4);
>> mesh(Z) □
>> title('mesh diagram') □
>> zlabel('Axis z')
>> xlabel('Axis x') □
>> ylabel('Axis y')
```



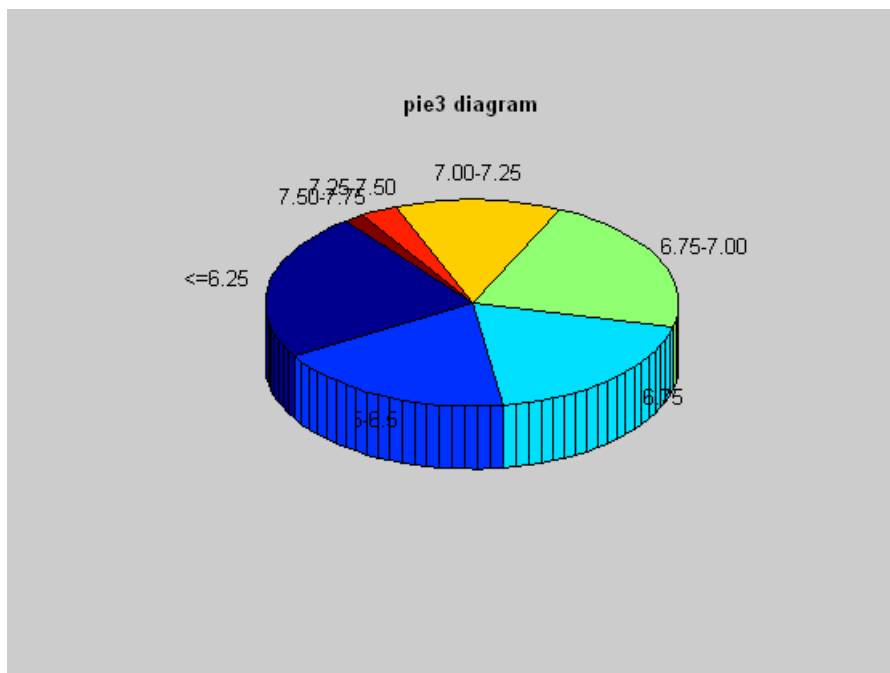
Συνάρτηση *surf*

```
>> a=-6:0.3:6; □
>> [X,Y]=meshgrid(a,a); □
>> Z=cos(X).*cos(Y).*...
exp(-sqrt(X.^2+Y.^2)/4); □
>> surf(X,Y,Z) □
>> zlabel('Axis z') □
>> ylabel('Axis y') □
>> xlabel('Axis x') □
>> title('surf diagram')
```



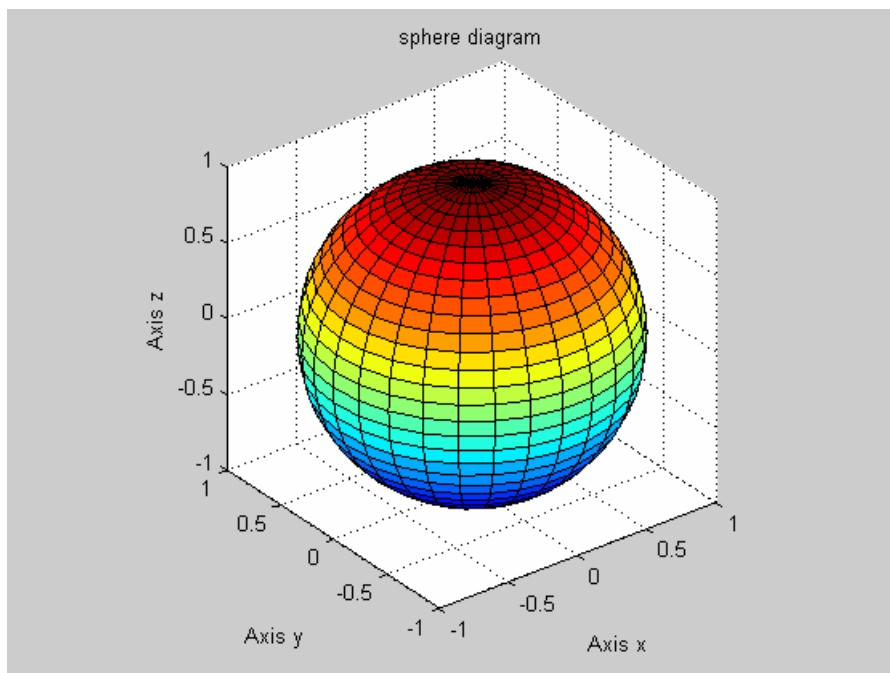
Συνάρτηση *pie3*

```
>> earthquakes=[16;13;13;15;9;2;1];
>> magnitude={'<=6.25','6.25-6.5','6.5-6.75',...
'6.75-7.00','7.00-7.25','7.25-7.50','7.50-7.75'};
>> pie3(earthquakes,magnitude)
>> Title('pie3 diagram','Fontweight','bold')
```



Συνάρτηση *sphere*

```
>> [a,b,c]=sphere(30);
>> surf(a,b,c) □
>> axis('square') □
>> ylabel('Axis y')
>> xlabel('Axis x')
>> zlabel('Axis z')
>> title('sphere diagram')
```

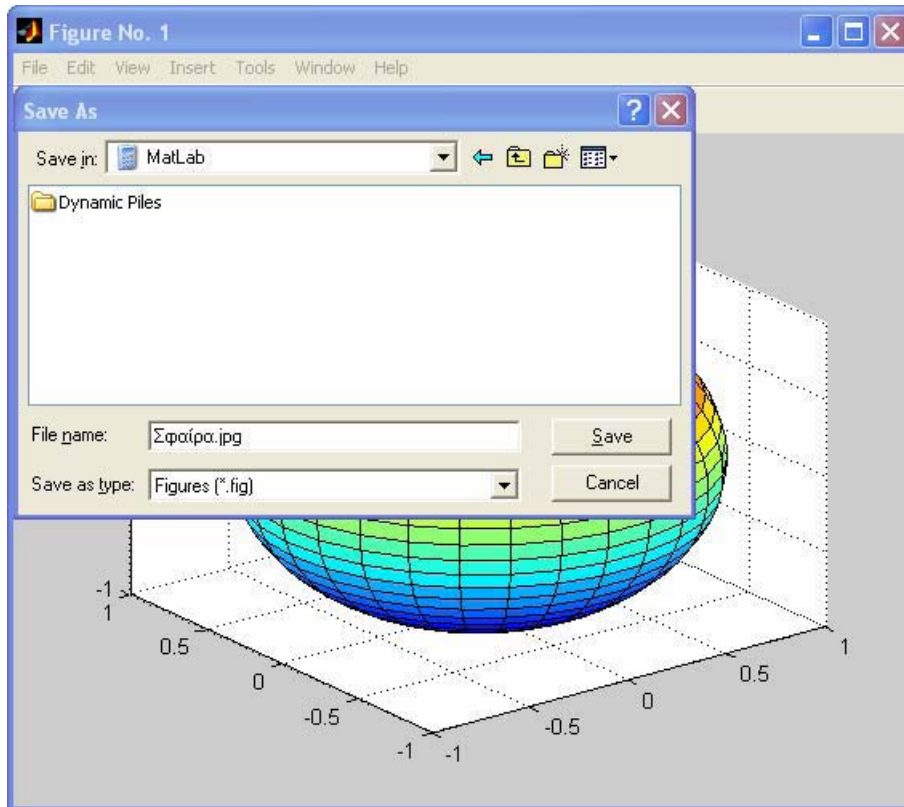
**2.15 Αποθήκευση γραφημάτων**

Από τις προηγούμενες ενότητες καθίσταται αντιληπτό ότι οι δυνατότητες, τις οποίες παρέχει το MATLAB για σχεδίαση τόσο δυσδιάστατων όσο και τρισδιάστατων γραφικών απεικονίσεων είναι μεγάλες. Τα γραφήματα αυτά είναι πιθανό ο χρήστης να επιθυμεί να τα έχει στη διάθεση του ως αρχεία εικόνων, με τις διάφορες γνωστές καταλήξεις αρχείων εικόνων, για να μπορεί να τα επεξεργαστεί κατά βούληση.

Η αποθήκευση των διαγραμμάτων, τα οποία παράγονται από το λογισμικό είναι μια απλή διαδικασία, αφού είναι δυνατή η πραγματοποίησή της μέσω της εργαλειοθήκης του παράθυρου των γραφημάτων (*Figure window*). Στο παράθυρο αυτό, ακολουθώντας τη γνωστή διαδικασία, η οποία συναντάται σε εφαρμογές Windows, είναι δυνατό να αποθηκεύσει (*Save*) τα γραφήματα ως αρχεία με κατάληξη *.fig*, την προτείνει το ίδιο το

λογισμικό. Τα αρχεία αυτά μπορούν σε κάποια άλλη εφαρμογή του MATLAB να επαναχρησιμοποιηθούν.

Ο χρήστης όμως είναι σε θέση να αποθηκεύσει τα γραφήματα ως αρχεία εικόνων με οποιαδήποτε επιθυμητή κατάληξη (.bmp, .tiff, .jpeg, .jpg), αρκεί στο πλαίσιο διαλόγου του *Save As* να πληκτρολογήσει το όνομα του αρχείου, συνοδευόμενο από την κατάληξη.



Εικόνα : Αποθήκευση διαγραμμάτων ως αρχεία εικόνων

3 ΕΠΕΞΕΡΓΑΣΙΑ ΓΕΩΦΥΣΙΚΩΝ ΔΕΔΟΜΕΝΩΝ

3.1 Εισαγωγή

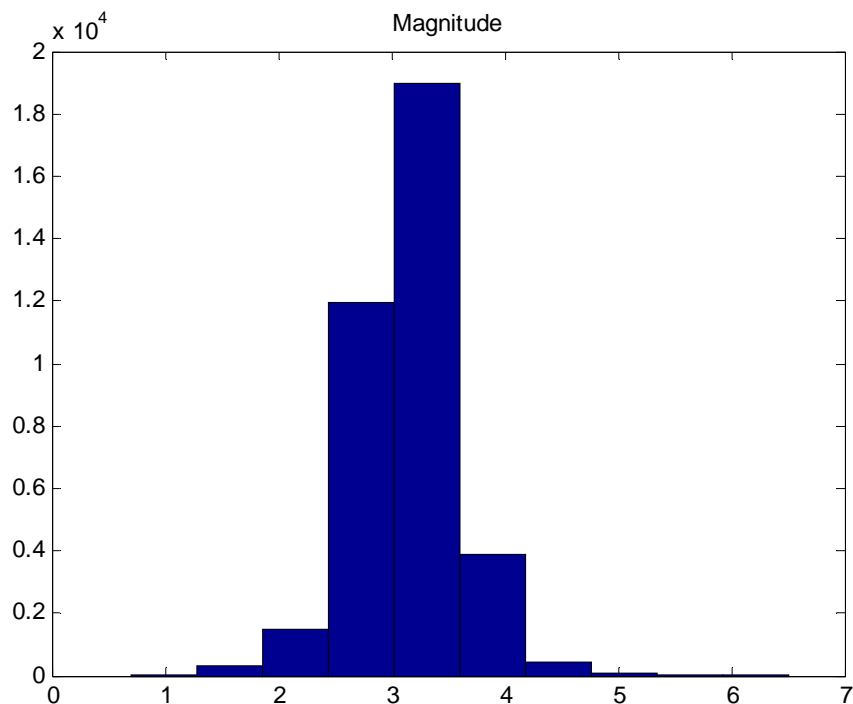
Στα πλαίσια του συγκεκριμένου κεφαλαίου παρουσιάζεται μεθοδολογία προσέγγισης συγκεκριμένων γεωφυσικών προβλημάτων.

3.2 Σεισμολογικά δεδομένα

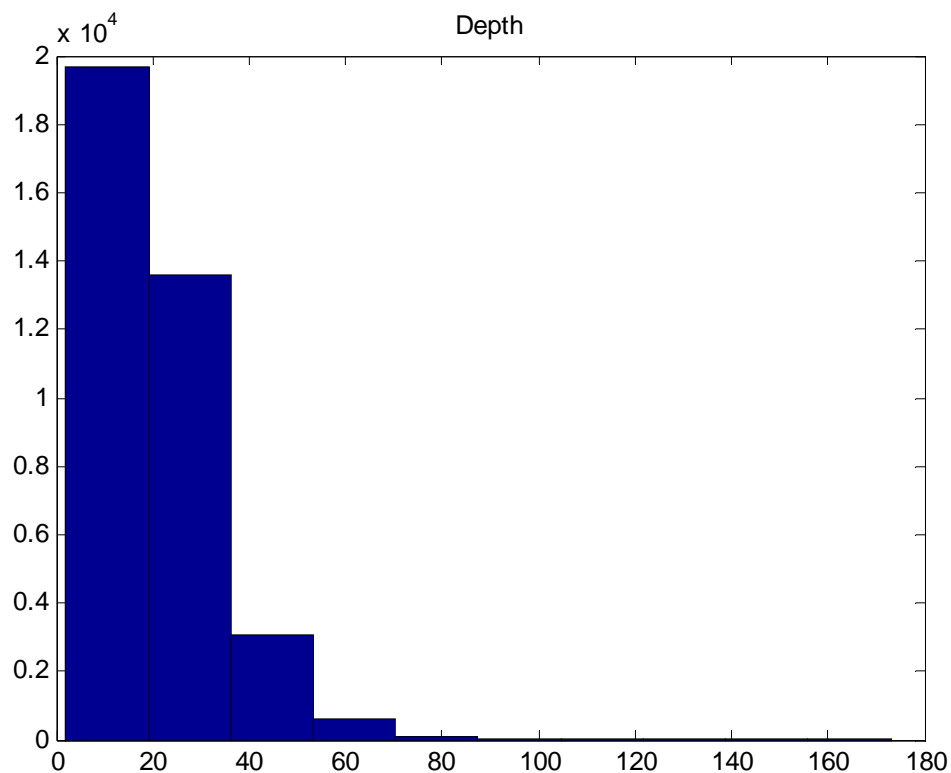
Η συγκεκριμένη εφαρμογή κάνει στατιστική επεξεργασία σεισμολογικών δεδομένων από αρχείο .txt. Τα δεδομένα προέρχονται από την ελεύθερη σε πρόσβαση βάση του αστεροσκοπείου Αθηνών.

Η διαδικασία επεξεργασίας έχει ως εξής:

1. Άνοιγμα παράθυρου για την επιλογή του αρχείου.
2. Ανάγνωση αρχείου δεδομένων και αποθήκευση σε πίνακα με συγκεκριμένο όνομα.
3. Υπολογίζονται τα ιστογράμματα για κάθε παράμετρο (μέγεθος και βάθος), για την χρονική περίοδο που αντιστοιχούν (σχ. 3.1α, β).
4. Υπολογισμός απλών στατιστικών παραμέτρων για μέγεθος και βάθος (σχ. 3.2).



(α)



(β)

Σχήμα 3.1 Ιστογράμματα που παρουσιάζουν την κατανομή κάθε μίας από τις παραμέτρους στο αρχείο εισόδου που χρησιμοποιήθηκε

The screenshot shows the MATLAB workspace with a table of statistics for various parameters. The parameters listed include DEPTH, MAGNITUDE, and their respective statistical measures (Min, Max, Mean, Standard Deviation, Range).

Name	Value	Min	Max
DEPTH	<37180x1 double>	2	173
MAGNITUDE	<37170x1 double>	0.7000	6.5000
Max_Depth	173	173	173
Max_Magnitude	6.5000	6.5000	6.5000
Mean_Depth	19.9533	19.9...	19.9...
Mean_Magnitude	3.1439	3.1439	3.1439
Min_Depth	2	2	2
Min_Magnitude	0.7000	0.7000	0.7000
Range_Depth	171	171	171
Range_Magnitude	5.8000	5.8000	5.8000
Standard_Deviat...	13.7238	13.7...	13.7...
Standard_Deviat...	0.4188	0.4188	0.4188

Σχήμα 3.2 Η μέση τιμή κάθε παραμέτρου ανά μήνα

3.3 Διαδικασία παρεμβολής σε γεωφυσικά δεδομένα

Πριν την παρουσίαση της εφαρμογής, θεωρείται σημαντικό να αναφέρουμε κάποια στοιχεία για την διαδικασία της παρεμβολής.

Η παρεμβολή (Νισταζάκης, 2005) είναι μια διαδικασία κατά την οποία αν έχουμε έναν αριθμό γνωστών πειραματικών δεδομένων και θέλουμε να κατασκευάσουμε μία συνάρτηση $y(x)$ η οποία να διέρχεται ακριβώς από όλα τα ζεύγη σημείων (x_i, y_i) . Πρέπει δηλαδή να ικανοποιείται η ακόλουθη συνθήκη:

$$\Delta y_i = y_i - y(x_i) = 0$$

χωρίς να μας ενδιαφέρει η συμπεριφορά της συνάρτησης $y(x)$ σε κάθε ενδιάμεσο σημείο μεταξύ δύο διαδοχικών σημείων (x_i, y_i) και (x_{i+1}, y_{i+1}) .

Υπάρχουν αρκετές μέθοδοι παρεμβολής και ένας από αυτούς είναι η πολυωνυμική. Ο λόγος για τον οποίο ονομάζεται έτσι η μέθοδος είναι ότι μας ενδιαφέρουν πολυωνυμικές συναρτήσεις $y(x)$. Η απλούστερη πολυωνυμική παρεμβολή είναι η γραμμική παρεμβολή και βασίζεται στη γνωστή πρόταση της γεωμετρίας ότι από δύο διαφορετικά σημεία του επιπέδου διέρχεται μία και μόνο μία ευθεία.

Ετσι, αν έχουμε δοσμένα δύο σημεία του επιπέδου (x_1, y_1) και (x_2, y_2) , με $x_1 \neq x_2$ τότε η ευθεία που διέρχεται από αυτά έχει κλίση:

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1}$$

και εξίσωση

$$y(x) = y_1 + \lambda(x - x_1) = y_1 + \frac{y_2 - y_1}{x_2 - x_1}(x - x_1)$$

Η αμέσως επόμενη γενίκευση αφορά ένα πολυώνυμο δεύτερου βαθμού (παραβολή), το οποίο διέρχεται από 3 σημεία για το οποίο φυσικά ισχύει ότι:

Από τρία δεδομένα σημεία του επιπέδου (x_i, y_i) , $i=1,2,3$, με $x_1 \neq x_2 \neq x_3$ διέρχεται η γραφική παράσταση ενός και μόνο ενός πολυωνύμου $y(x)$ 2ου βαθμού.

Η παραπάνω πρόταση μπορεί να γενικευτεί. Ετσι, για περισσότερα από 3 σημεία θα ισχύει:

Από N δεδομένα σημεία του επιπέδου (x_i, y_i) , $i=1,2,3$, με $x_1 \neq x_2 \neq x_3$ διέρχεται η γραφική παράσταση ενός και μόνο ενός πολυωνύμου $y(x)$ $(N-1)$ -οστού βαθμού.

Αυτό λοιπόν που μας ενδιαφέρει στην περίπτωση πολυωνυμικής παρεμβολής είναι να κατασκευάσουμε ένα πολυώνυμο το οποίο θα έχει βαθμό κατά ένα μικρότερο από τον αριθμό των σημείων από τα οποία πρέπει να περάσει η γραφική του παράσταση.

Η διαδικασία ευρεσης του πολυωνύμου είναι αρκετά απλή:

(α) έστω ότι μας ενδιαφέρει το πολυώνυμο να περνάει από 3 σημεία τα οποία είναι τα (x_1, y_1) , (x_2, y_2) και (x_3, y_3) .

(β) προφανώς το πολυώνυμο που μας ενδιαφέρει θα είναι $2^{\text{ου}}$ βαθμού. Άρα θα έχει τη μορφή: $f(x) = ax^2 + bx + c$. Το κάθε σημείο (x_i, y_i) θα πρέπει να ικανοποιεί την εξίσωση αυτή.

(γ) άρα δημιουργείται το ακόλουθο σύστημα εξισώσεων:

$$y_1 = ax_1^2 + bx_1 + c$$

$$y_2 = ax_2^2 + bx_2 + c$$

$$y_3 = ax_3^2 + bx_3 + c$$

το σύστημα αυτό είναι ένα σύστημα 3X3 και μέσω των γνωστών μεθόδων επίλυσης γραμμικών συστημάτων υπολογίζουμε τους αγνωστούς a, b, c .

Το πρόγραμμα λοιπόν που πρέπει να κατασκευαστεί θα πρέπει αρχικά να ρωτάει τον χρήστη για τον αριθμό των πειραματικών δεδομένων που θα εισαχθούν. Στη συνέχεια, να διαβάζει από το πληκτρολόγιο τον κατάλληλο αριθμό σημείων και να κατασκευάζει τους αντίστοιχους επαυξημένους πίνακες. Από τη στιγμή που έχουν κατασκευαστεί οι επαυξημένοι πίνακες το πρόβλημα έγγυται στην επίλυση του γραμμικού συστήματος, το οποίο είναι μία διαδικασία η οποία έχει ήδη συζητηθεί και διδαχθεί.

Έτσι λοιπόν το πρόγραμμα που θα φτιάχνει τον επαυξημένο πίνακα για τον υπολογισμό των συντελεστών του πολυωνύμου θα το ονομάσουμε `parembolh1.m` και θα έχει την ακόλουθη μορφή:

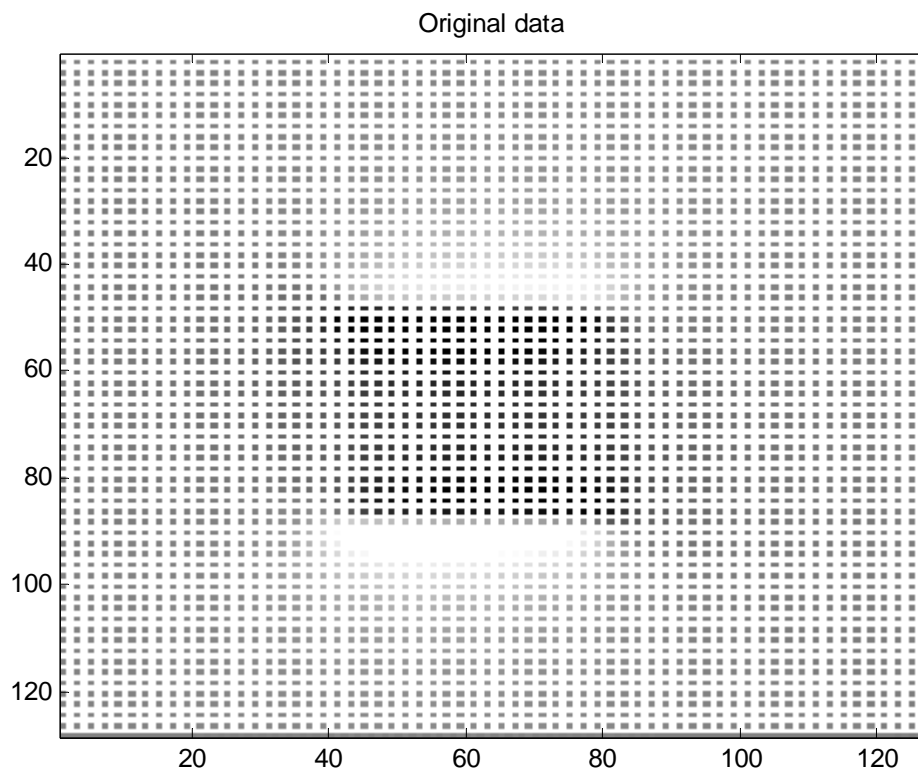
Πρόγραμμα:

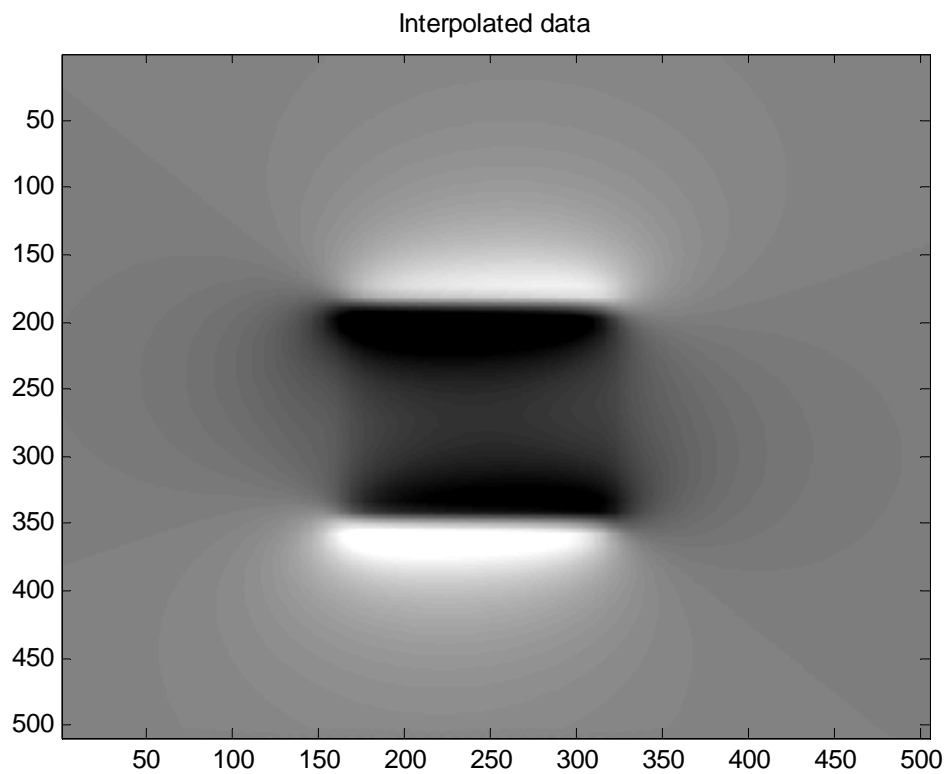
```
clear;
clc;
close all;
fprintf('arithmos shmeiwn=');
numb1 = input('');
for(n1=1:1:numb1)
    fprintf('x%d=',n1);
    x(n1) = input('');
    fprintf('y%d=',n1);
    y(n1) = input('');
end
for(m=1:1:numb1+1)
    for(k=1:1:numb1)
        if(m<=numb1)
            matr(k,m) = x(k)^(numb1-m);
```

```
else  
matr(k,m) = y(k);  
end  
end  
end
```

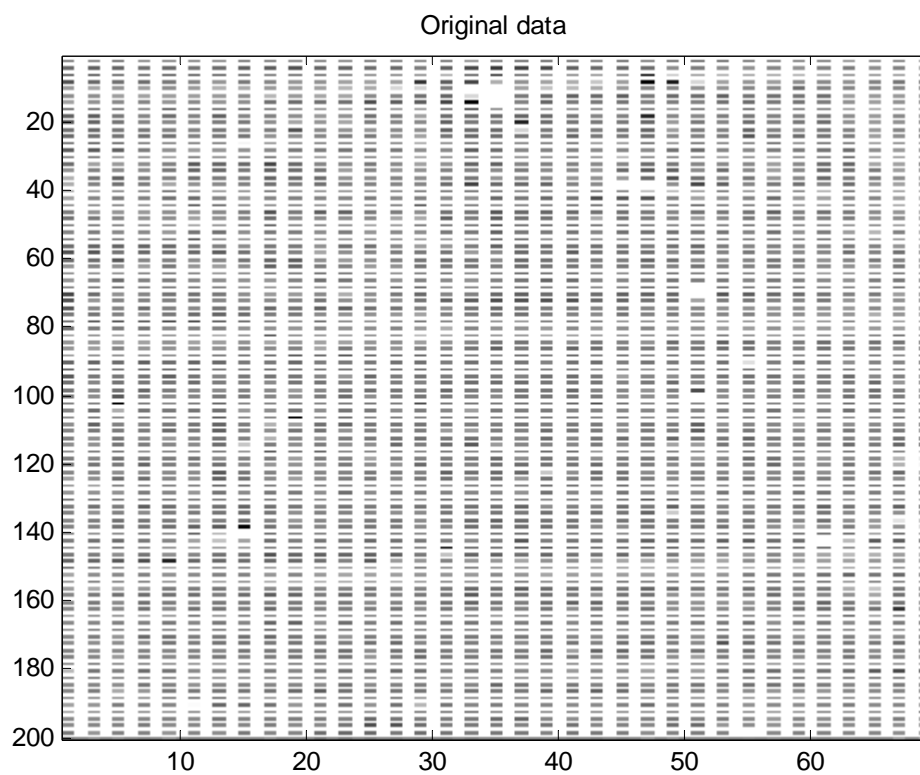
Παρατηρούμε ότι, το πρόγραμμα αυτό χωρίζεται σε δύο τμήματα. Στο πρώτο ο χρήστης εισάγει μέσω του πληκτρολογίου τις τιμές των ζευγών που τον ενδιαφέρουν και στο δεύτερο ο υπολογιστής φτιάχνει τον επαυξημένο πίνακα μέσω του οποίου θα υπολογιστούν οι συντελεστές του πολωνύμου παρεμβολής.

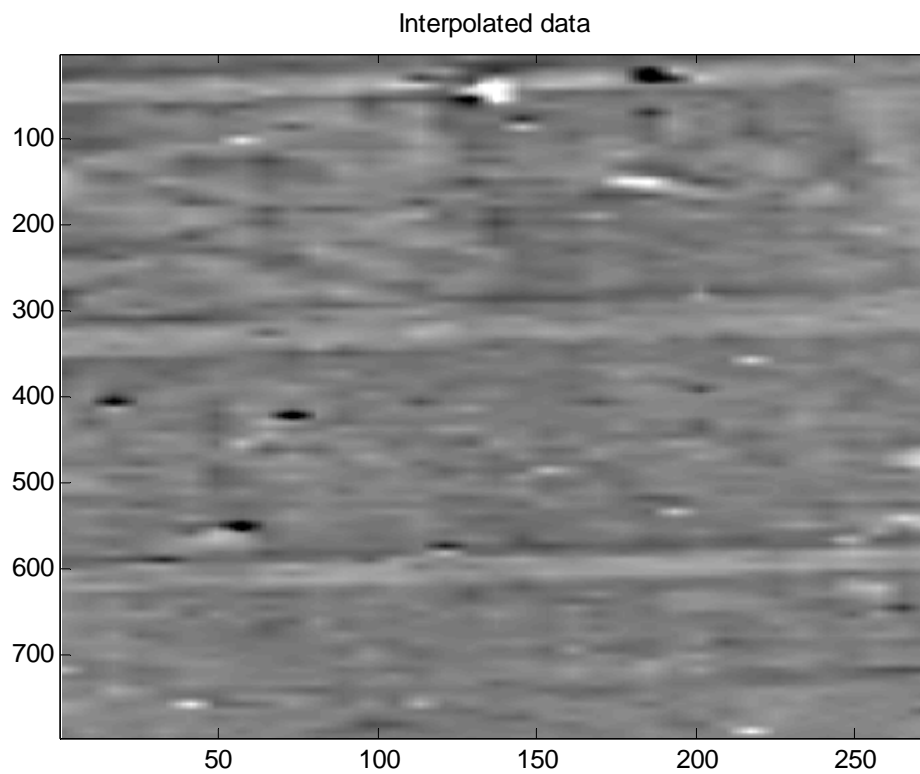
Στο σχήμα 3.3 παρουσιάζονται παραδείγματα γεωφυσικών μαγνητικών δεδομένων χαρτογράφησης από την εργασία των *Panagiotakis C., Kokinou E., Sarris A., 2011, Curvilinear Structure Enhancement and Detection in Geophysical images Based on a Multiple Filtering Scheme, IEEE Transactions on Geoscience and Remote Sensing v.49, no.6, 2040-2048, (DOI: 10.1109/TGRS.2010.2102042)* με και χωρίς την εφαρμογή της διαδικασίας παρεμβολής.



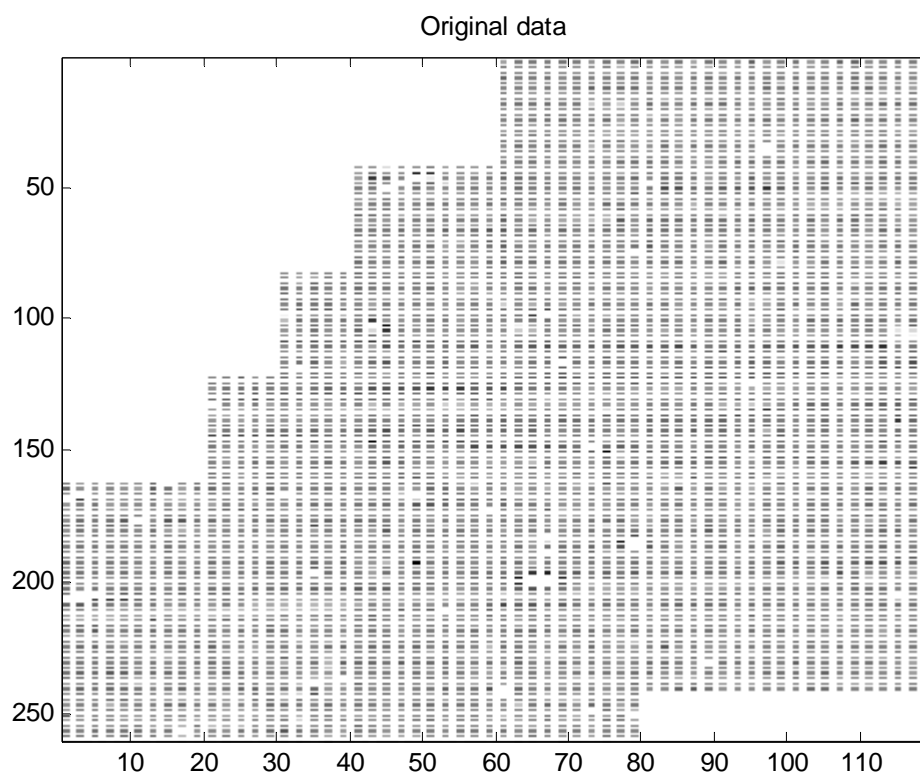


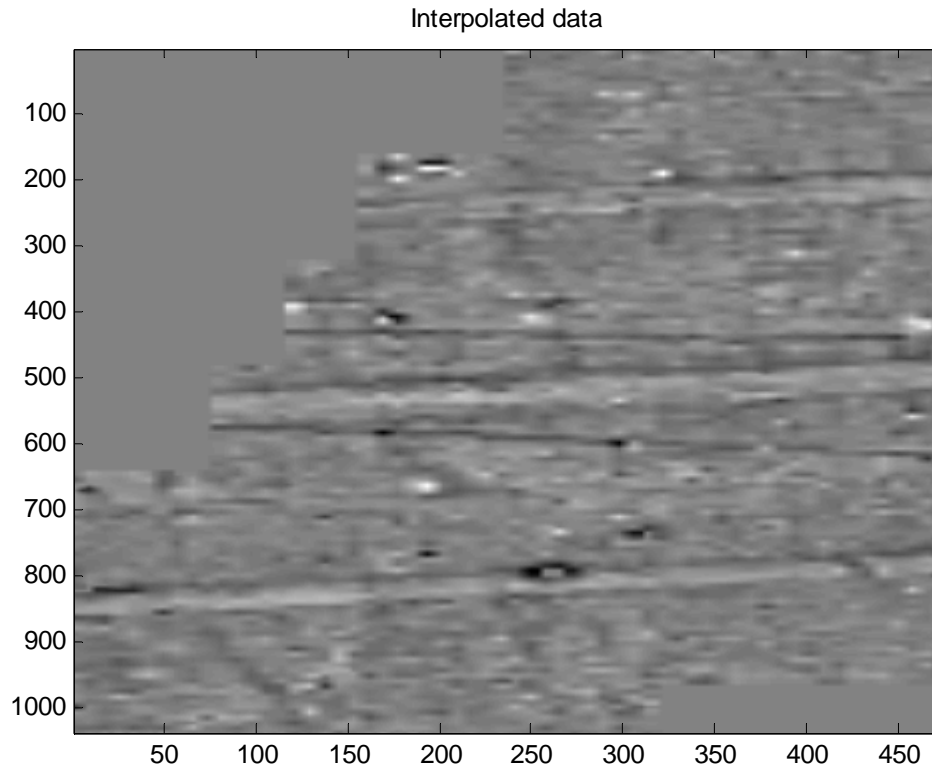
(α) - συνθετικά δεδομένα





(β) πραγματικά δεδομένα





(γ) πραγματικά δεδομένα

Σχήμα 3.3 Παραδείγματα με και χωρίς την διαδικασία παρεμβολής σε γεωφυσικά δεδομένα από μαγνητική χαρτογράφηση

4 ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Biran A. and Breiner M. (2003), “MATLAB 6 για Μηχανικούς”, Εκδόσεις Τζιόλα, Θεσσαλονίκη
2. Elter D.M. (1997), “Engineering problems solving with MATLAB”, Prentice Hall
3. Etter D., Kuncicky D. and Moore H. (2004), “Introduction to MATLAB 7”, Pearson Education Limited
4. Hanselman D.C and Littlefield B.L. (2004), “Mastering MATLAB 7 – A comprehensive tutorial and Reference”, Prentice Hall
5. The Mathworks, Inc. (2001), “Using MATLAB 6”
6. The Mathworks, Inc., “MATLAB Documentation, v.6.5”
7. Pratar R. (2006), “Getting Started with MATLAB 7”, Oxford University Press, New York
8. Wilson H., Turcotte L. and Halpern D. (2003), “Advanced Mathematics and Mechanics Applications using MATLAB”, Chapman & Hall/CRC
9. Λουκογεωργάκη Ε και Αγγελίδης Δ. (2006), “Εισαγωγικές Διδακτικές σημειώσεις για το λογισμικό MATLAB”, Πανεπιστημιακές σημειώσεις, Α.Π.Θ., Θεσσαλονίκη
10. Παπαρρίζος Κ. (2004), “MATLAB 6.5”, Εκδόσεις Ζήτη, Θεσσαλονίκη
11. Χατζίκος Ε. (2007), “MATLAB 7 για Μηχανικούς”, Εκδόσεις Τζιόλα, Θεσσαλονίκη
12. <http://www.clab.edc.uoc.gr>
13. Νισταζάκης Ε.Ε., 2005, ΕΦΑΡΜΟΣΜΕΝΑ ΜΑΘΗΜΑΤΙΚΑ ΚΑΙ ΑΡΙΘΜΗΤΙΚΕΣ ΜΕΘΟΔΟΙ ΜΕ ΧΡΗΣΗ ΤΟΥ MATLAB, ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ, ΤΜΗΜΑ ΣΤΑΤΙΣΤΙΚΗΣ ΚΑΙ ΑΝΑΛΟΓΙΣΤΙΚΗΣ ΕΠΙΣΤΗΜΗΣ
14. Panagiotakis C., Kokinou E., Sarris A., 2011, Curvilinear Structure Enhancement and Detection in Geophysical images Based on a Multiple Filtering Scheme, IEEE Transactions on Geoscience and Remote Sensing v.49, no.6, 2040-2048, (DOI: 10.1109/TGRS.2010.2102042)