

Μαθηματικό Λογισμικό

Σταμάτιος-Άγγελος Ν. Αλεξανδρόπουλος
e-mail: stalexan@ee.duth.gr

https://www.researchgate.net/profile/Stamatios_Aggelos_Alexandropoulos
https://scholar.google.gr/citations?user=mht7W_YAAAAJ&hl=el
<http://cilab.math.upatras.gr>

Τμήμα Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών
Δημοκρίτειο Πανεπιστήμιο Θράκης
Κιμμέρια 67100, Ξάνθη

7 Οκτωβρίου 2020

Περιεχόμενα I

- 1 Εισαγωγή
 - Μεταβλητές
 - Τελεστές
 - Είσοδος και έξοδος δεδομένων
 - Εντολές διαχείρισης χώρου εργασίας
- 2 Διανύσματα και μητρώα
 - Πράξεις μεταξύ μητρώων και βαθμωτών
 - Πράξεις μεταξύ διανυσμάτων και μητρώων
 - Χρήσιμα μητρώα και συναρτήσεις βιβλιοθήκης
 - Μεταβολές των στοιχείων μητρώου και διανύσματος
- 3 Προγραμματισμός στο MATLAB
 - Η δομή if
 - Η δομή for
 - Η δομή while
 - M-files
- 4 Παραγωγή γραφικών
 - Η συνάρτηση plot
 - Τρισδιάστατα γραφήματα

Περιεχόμενα II

5 Βιβλιογραφία

Γιατί χρησιμοποιούμε MATLAB

- Είναι ένα ολοκληρωμένο **μαθηματικό λογισμικό** πακέτο που χρησιμοποιείται ευρύτατα στην επιστημονική κοινότητα αλλά και σε ερευνητικές και άλλες εφαρμογές με επιστημονικούς υπολογισμούς

Γιατί χρησιμοποιούμε MATLAB

- Είναι ένα ολοκληρωμένο **μαθηματικό λογισμικό** πακέτο που χρησιμοποιείται ευρύτατα στην επιστημονική κοινότητα αλλά και σε ερευνητικές και άλλες εφαρμογές με επιστημονικούς υπολογισμούς
- Έχει μεγάλες **γραφικές δυνατότητες**, **ευκολία** και **ταχύτητα υλοποίησης** αλγορίθμων και πολλές **έτοιμες ρουτίνες** (built-in).

Γιατί χρησιμοποιούμε MATLAB

- Είναι ένα ολοκληρωμένο **μαθηματικό λογισμικό** πακέτο που χρησιμοποιείται ευρύτατα στην επιστημονική κοινότητα αλλά και σε ερευνητικές και άλλες εφαρμογές με επιστημονικούς υπολογισμούς
- Έχει μεγάλες **γραφικές δυνατότητες**, **ευκολία** και **ταχύτητα υλοποίησης** αλγορίθμων και πολλές **έτοιμες ρουτίνες** (built-in).
- Δίνει την ευχέρεια στον χρήστη να το **επεκτείνει** με δικά του προγράμματα.

Γιατί χρησιμοποιούμε MATLAB

- Είναι ένα ολοκληρωμένο **μαθηματικό λογισμικό** πακέτο που χρησιμοποιείται ευρύτατα στην επιστημονική κοινότητα αλλά και σε ερευνητικές και άλλες εφαρμογές με επιστημονικούς υπολογισμούς
- Έχει μεγάλες **γραφικές δυνατότητες**, **ευκολία** και **ταχύτητα υλοποίησης** αλγορίθμων και πολλές **έτοιμες ρουτίνες** (built-in).
- Δίνει την ευχέρεια στον χρήστη να το **επεκτείνει** με δικά του προγράμματα.
- Πραγματοποιεί **αριθμητική επίλυση** προβλημάτων σε αριθμητική **πεπερασμένης** ακρίβειας.

Γιατί χρησιμοποιούμε MATLAB

- Είναι ένα ολοκληρωμένο **μαθηματικό λογισμικό** πακέτο που χρησιμοποιείται ευρύτατα στην επιστημονική κοινότητα αλλά και σε ερευνητικές και άλλες εφαρμογές με επιστημονικούς υπολογισμούς
- Έχει μεγάλες **γραφικές δυνατότητες**, **ευκολία** και **ταχύτητα υλοποίησης** αλγορίθμων και πολλές **έτοιμες ρουτίνες** (built-in).
- Δίνει την ευχέρεια στον χρήστη να το **επεκτείνει** με δικά του προγράμματα.
- Πραγματοποιεί **αριθμητική επίλυση** προβλημάτων σε αριθμητική **πεπερασμένης** ακρίβειας.
- Είναι ειδικά σχεδιασμένο για **υπολογισμούς με μητρώα**.

Όνομα μεταβλητής

- Για τα ονόματα **μεταβλητών** χρησιμοποιούνται μόνο γράμματα του αγγλικού αλφαβήτου.

Όνομα μεταβλητής

- Για τα ονόματα **μεταβλητών** χρησιμοποιούνται μόνο γράμματα του αγγλικού αλφαβήτου.
- Η MATLAB κάνει **διάκριση** μεταξύ κεφαλαίων και μικρών γραμμάτων (η μεταβλητή x είναι διαφορετική από τη μεταβλητή X).

Όνομα μεταβλητής

- Για τα ονόματα **μεταβλητών** χρησιμοποιούνται μόνο γράμματα του αγγλικού αλφαβήτου.
- Η MATLAB κάνει **διάκριση** μεταξύ κεφαλαίων και μικρών γραμμάτων (η μεταβλητή *x* είναι διαφορετική από τη μεταβλητή *X*).
- Το όνομα της μεταβλητής **αρχίζει** πάντα με **γράμμα** και μπορεί να περιέχει μόνο **γράμματα**, **αριθμούς** και **υποπαύλες** (underscores).

Όνομα μεταβλητής

- Για τα ονόματα **μεταβλητών** χρησιμοποιούνται μόνο γράμματα του αγγλικού αλφαβήτου.
- Η MATLAB κάνει **διάκριση** μεταξύ κεφαλαίων και μικρών γραμμάτων (η μεταβλητή x είναι διαφορετική από τη μεταβλητή X).
- Το όνομα της μεταβλητής **αρχίζει** πάντα με **γράμμα** και μπορεί να περιέχει μόνο **γράμματα**, **αριθμούς** και **υποπαύλες** (underscores).
- **Δε** μπορούμε να χρησιμοποιήσουμε ονόματα που έχουν **δεσμευτεί**.

Όνομα μεταβλητής

- Για τα ονόματα **μεταβλητών** χρησιμοποιούνται μόνο γράμματα του αγγλικού αλφαβήτου.
- Η MATLAB κάνει **διάκριση** μεταξύ κεφαλαίων και μικρών γραμμάτων (η μεταβλητή *x* είναι διαφορετική από τη μεταβλητή *X*).
- Το όνομα της μεταβλητής **αρχίζει** πάντα με **γράμμα** και μπορεί να περιέχει μόνο **γράμματα**, **αριθμούς** και **υποπαύλες** (underscores).
- **Δε** μπορούμε να χρησιμοποιήσουμε ονόματα που έχουν **δεσμευτεί**.

Τύπος μεταβλητής

Όνομα μεταβλητής

- Για τα ονόματα **μεταβλητών** χρησιμοποιούνται μόνο γράμματα του αγγλικού αλφαβήτου.
- Η MATLAB κάνει **διάκριση** μεταξύ κεφαλαίων και μικρών γραμμάτων (η μεταβλητή x είναι διαφορετική από τη μεταβλητή X).
- Το όνομα της μεταβλητής **αρχίζει** πάντα με **γράμμα** και μπορεί να περιέχει μόνο **γράμματα**, **αριθμούς** και **υποπαύλες** (underscores).
- **Δε** μπορούμε να χρησιμοποιήσουμε ονόματα που έχουν **δεσμευτεί**.

Τύπος μεταβλητής

- πραγματικές

Όνομα μεταβλητής

- Για τα ονόματα **μεταβλητών** χρησιμοποιούνται μόνο γράμματα του αγγλικού αλφαβήτου.
- Η MATLAB κάνει **διάκριση** μεταξύ κεφαλαίων και μικρών γραμμάτων (η μεταβλητή *x* είναι διαφορετική από τη μεταβλητή *X*).
- Το όνομα της μεταβλητής **αρχίζει** πάντα με **γράμμα** και μπορεί να περιέχει μόνο **γράμματα**, **αριθμούς** και **υποπαύλες** (underscores).
- **Δε** μπορούμε να χρησιμοποιήσουμε ονόματα που έχουν **δεσμευτεί**.

Τύπος μεταβλητής

- πραγματικές
- αλφαριθμητικές (strings)

Όνομα μεταβλητής

- Για τα ονόματα **μεταβλητών** χρησιμοποιούνται μόνο γράμματα του αγγλικού αλφαβήτου.
- Η MATLAB κάνει **διάκριση** μεταξύ κεφαλαίων και μικρών γραμμάτων (η μεταβλητή x είναι διαφορετική από τη μεταβλητή X).
- Το όνομα της μεταβλητής **αρχίζει** πάντα με **γράμμα** και μπορεί να περιέχει μόνο **γράμματα**, **αριθμούς** και **υποπαύλες** (underscores).
- **Δε** μπορούμε να χρησιμοποιήσουμε ονόματα που έχουν **δεσμευτεί**.

Τύπος μεταβλητής

- πραγματικές
- αλφαριθμητικές (strings)
- λογικές (logical)

Όνομα μεταβλητής

- Για τα ονόματα **μεταβλητών** χρησιμοποιούνται μόνο γράμματα του αγγλικού αλφαβήτου.
- Η MATLAB κάνει **διάκριση** μεταξύ κεφαλαίων και μικρών γραμμάτων (η μεταβλητή x είναι διαφορετική από τη μεταβλητή X).
- Το όνομα της μεταβλητής **αρχίζει** πάντα με **γράμμα** και μπορεί να περιέχει μόνο **γράμματα**, **αριθμούς** και **υποπαύλες** (underscores).
- **Δε** μπορούμε να χρησιμοποιήσουμε ονόματα που έχουν **δεσμευτεί**.

Τύπος μεταβλητής

- πραγματικές
- αλφαριθμητικές (strings)
- λογικές (logical)
- μιγαδικές

Όνομα μεταβλητής

- Για τα ονόματα **μεταβλητών** χρησιμοποιούνται μόνο γράμματα του αγγλικού αλφαβήτου.
- Η MATLAB κάνει **διάκριση** μεταξύ κεφαλαίων και μικρών γραμμάτων (η μεταβλητή x είναι διαφορετική από τη μεταβλητή X).
- Το όνομα της μεταβλητής **αρχίζει** πάντα με **γράμμα** και μπορεί να περιέχει μόνο **γράμματα**, **αριθμούς** και **υποπαύλες** (underscores).
- **Δε** μπορούμε να χρησιμοποιήσουμε ονόματα που έχουν **δεσμευτεί**.

Τύπος μεταβλητής

- πραγματικές
- αλφαριθμητικές (strings)
- λογικές (logical)
- μιγαδικές

Ο τύπος των μεταβλητών **δε** χρειάζεται να έχει δηλωθεί εκ των προτέρων όπως σε άλλες γλώσσες προγραμματισμού.

- Η εκχώρηση τιμής σε μια μεταβλητή γίνεται με το σύμβολο '='

- Η **εκχώρηση** τιμής σε μια μεταβλητή γίνεται με το σύμβολο '='
- Το **αποτέλεσμα** της κάθε εντολής εμφανίζεται στο παράθυρο εργασίας. Για να **μην εμφανίζεται**, προσθέτουμε το ερωτηματικό ';' στο τέλος της εντολής.

- Η **εκχώρηση** τιμής σε μια μεταβλητή γίνεται με το σύμβολο '='
- Το **αποτέλεσμα** της κάθε εντολής εμφανίζεται στο παράθυρο εργασίας. Για να **μην εμφανίζεται**, προσθέτουμε το ερωτηματικό ';' στο τέλος της εντολής.
- Το περιεχόμενο μιας μεταβλητής εμφανίζεται πληκτρολογώντας το **όνομά** της στο **Command Window**.

- Η **εκχώρηση** τιμής σε μια μεταβλητή γίνεται με το σύμβολο '='
- Το **αποτέλεσμα** της κάθε εντολής εμφανίζεται στο παράθυρο εργασίας. Για να **μην εμφανίζεται**, προσθέτουμε το ερωτηματικό ';' στο τέλος της εντολής.
- Το περιεχόμενο μιας μεταβλητής εμφανίζεται πληκτρολογώντας το **όνομά** της στο **Command Window**.

Παράδειγμα

```
1 >> a=10;  
2 >> b=20  
3 b =  
4  
5      20  
6 >> c=a+b  
7 c =  
8  
9      30  
10 >> a*b  
11 ans =  
12  
13      200
```

Η εκχώρηση μητρώων και διανυσμάτων σε μεταβλητές είναι απαραίτητη και διέπεται από τους εξής κανόνες:

- Τα στοιχεία του μητρώου/διανύσματος γράφονται ανάμεσα σε αγκύλες [. . .].

Η εκχώρηση μητρώων και διανυσμάτων σε μεταβλητές είναι απαραίτητη και διέπεται από τους εξής κανόνες:

- Τα στοιχεία του μητρώου/διανύσματος γράφονται ανάμεσα σε αγκύλες [. . .].
- Τα στοιχεία μιας γραμμής του μητρώου/διανύσματος χωρίζονται με κόμματα ή κενό.

Η εκχώρηση μητρώων και διανυσμάτων σε μεταβλητές είναι απαραίτητη και διέπεται από τους εξής κανόνες:

- Τα στοιχεία του μητρώου/διανύσματος γράφονται ανάμεσα σε αγκύλες [. . .].
- Τα στοιχεία μιας γραμμής του μητρώου/διανύσματος χωρίζονται με κόμματα ή κενό.
- Η αλλαγή γραμμής στο μητρώο δηλώνεται είτε με ερωτηματικό (;) είτε με αλλαγή γραμμής στο παράθυρο εντολών της MATLAB.

Η εκχώρηση μητρώων και διανυσμάτων σε μεταβλητές είναι απαραίτητη και διέπεται από τους εξής κανόνες:

- Τα στοιχεία του μητρώου/διανύσματος γράφονται ανάμεσα σε **αγκύλες** `[. . .]`.
- Τα στοιχεία μιας γραμμής του μητρώου/διανύσματος χωρίζονται με **κόμματα** ή **κενό**.
- Η **αλλαγή γραμμής** στο μητρώο δηλώνεται είτε με **ερωτηματικό** `(;)` είτε με **αλλαγή γραμμής** στο παράθυρο εντολών της MATLAB.

Παράδειγμα

```
1 >> x=[8 5 -3 0.5]
2
3 x =
4
5      8      5      -3      0.5
6
7 >> A=[0 0; 1 0]
8
9 A =
10
11      0      0
12      1      0
```

Οι αριθμητικοί τελεστές

- 1 Η σειρά με την οποία εκτελούνται οι πράξεις ακολουθεί τους συνήθεις αλγεβρικούς κανόνες

Πράξη	Τελεστής
Πρόσθεση	+
Αφαίρεση	-
Πολλαπλασιασμός	*
Διαίρεση	/
Υψωση σε δύναμη	^

Οι αριθμητικοί τελεστές

- 1 Η σειρά με την οποία εκτελούνται οι πράξεις ακολουθεί τους συνήθεις αλγεβρικούς κανόνες
- 2 Στο MATLAB υπάρχουν επίσης οι παρακάτω τελεστές συσχέτισης, οι οποίοι σε απλές μεταβλητές επιστρέφουν την τιμή **1**, αν η συσχέτιση είναι **αληθής** και **0**, αν είναι **ψευδής**.

Πράξη	Τελεστής
Πρόσθεση	+
Αφαίρεση	-
Πολλαπλασιασμός	*
Διαίρεση	/
Υψωση σε δύναμη	^

Ερμηνεία	Τελεστής
Ίσον	==
Διάφορο	~=
Μεγαλύτερο	>
Μεγαλύτερο ή ίσον	>=
Μικρότερο	<
Μικρότερο ή ίσον	<=

Οι λογικοί τελεστές

Τελεστής	Σημασία
&	Λογικός τελεστής και
	Λογικός τελεστής είτε
~	Λογικός τελεστής άρνησης

Παράδειγμα

```

1 >> x=10;
2 >> x>10
3 ans =
4
5      0
6 >> x==10
7 ans =
8      1

```

```

1 >> A=[1 2;3 4];
2 >> B=ones(2);
3 >> A==B
4 ans =
5      1      0
6      0      0
7 >> A>3
8 ans =
9      0      0
10     0      1

```

Οι λογικοί τελεστές

Τελεστής	Σημασία
&	Λογικός τελεστής και
	Λογικός τελεστής είτε
~	Λογικός τελεστής άρνησης

Αν και υπάρχει καθορισμένη προτεραιότητα, είναι καλό την προτεραιότητα των πράξεων να τη δηλώνουμε με παρενθέσεις.

Παράδειγμα

```

1 >> x=10;
2 >> x>10
3 ans =
4
5      0
6 >> x==10
7 ans =
8      1

```

```

1 >> A=[1 2;3 4];
2 >> B=ones(2);
3 >> A==B
4 ans =
5      1      0
6      0      0
7 >> A>3
8 ans =
9      0      0
10     0      1

```

Είσοδος δεδομένων

- Η εντολή `input` έχει τη γενική μορφή `RESULT = input(PROMPT)`

Είσοδος δεδομένων

- Η εντολή `input` έχει τη γενική μορφή `RESULT = input(PROMPT)`
- Το σύστημα αναμένει από τον χρήστη να εισάγει την τιμή της μεταβλητής `RESULT`.

Είσοδος δεδομένων

- Η εντολή `input` έχει τη γενική μορφή `RESULT = input(PROMPT)`
- Το σύστημα αναμένει από τον χρήστη να εισάγει την τιμή της μεταβλητής `RESULT`.
- Το `PROMPT` μπορεί να είναι αριθμός ή αλφαριθμητικό ή διάνυσμα ή μητρώο ή ακόμα το αποτέλεσμα μιας ολόκληρης παράστασης.

Είσοδος δεδομένων

- Η εντολή `input` έχει τη γενική μορφή `RESULT = input(PROMPT)`
- Το σύστημα αναμένει από τον χρήστη να εισάγει την τιμή της μεταβλητής `RESULT`.
- Το `PROMPT` μπορεί να είναι αριθμός ή αλφαριθμητικό ή διάνυσμα ή μητρώο ή ακόμα το αποτέλεσμα μιας ολόκληρης παράστασης.

Παράδειγμα

Η χρήση της εντολής `input`.

```
1 >> epwnumo = input("Grapste to epwnumo sas: ", "e")
2 Grapste to epwnumo sas: Maurotsoukalos
3 epwnumo =
4
5 Maurotsoukalos
6
7 >> mo = input("Dwse ton MO tou ptuxiou sou:")
8 Dwse ton MO tou ptuxiou sou: 7.48
9 mo =
10
11      7.4800
```

Έξοδος δεδομένων

- Η εντολή `fprintf` έχει τη μορφή `fprintf(A, ...)`

Έξοδος δεδομένων

- Η εντολή `fprintf` έχει τη μορφή `fprintf(A, ...)`
- Εμφανίζει στην οθόνη τα στοιχεία των μεταβλητών A_i και πιθανά μηνύματα γραμμένα μέσα σε ' '

Έξοδος δεδομένων

- Η εντολή `fprintf` έχει τη μορφή `fprintf(A, ...)`
- Εμφανίζει στην οθόνη τα στοιχεία των μεταβλητών A_i και πιθανά μηνύματα γραμμένα μέσα σε ' '

Παράδειγμα

Η χρήση της εντολής `fprintf`.

```
1 >> x=1.117
2 x =
3     1.1170
4 >> fprintf("I timi tou diesel einai %6.3f euro", x)
5 I timi tou diesel einai 1.1170 euro
6 >>
```

- Με τις εντολές `quit` και `exit` τερματίζεται η τρέχουσα εργασία.

- Με τις εντολές `quit` και `exit` τερματίζεται η τρέχουσα εργασία.
- `clear` και `clc`
 - Η εντολή `clear` **όνομα** μεταβλητής διαγράφει τη μεταβλητή **όνομα** του χώρου εργασίας.
 - Η εντολή `clear` διαγράφει όλες τις ενεργές μεταβλητές.
 - Η εντολή `clc` «καθαρίζει» το παράθυρο εργασίας χωρίς να διαγράφει τις μεταβλητές.

- Με τις εντολές `quit` και `exit` τερματίζεται η τρέχουσα εργασία.
- `clear` και `clc`
 - Η εντολή `clear` **όνομα** μεταβλητής διαγράφει τη μεταβλητή **όνομα** του χώρου εργασίας.
 - Η εντολή `clear` διαγράφει όλες τις ενεργές μεταβλητές.
 - Η εντολή `clc` «καθαρίζει» το παράθυρο εργασίας χωρίς να διαγράφει τις μεταβλητές.
- Αν πληκτρολογήσουμε `help` **όνομα εντολής**, τότε εμφανίζεται ένα νέο παράθυρο που έχει αναλυτική περιγραφή της εντολής **όνομα**.

Περιεχόμενα I

- 1 Εισαγωγή
 - Μεταβλητές
 - Τελεστές
 - Είσοδος και έξοδος δεδομένων
 - Εντολές διαχείρισης χώρου εργασίας
- 2 Διανύσματα και μητρώα
 - Πράξεις μεταξύ μητρώων και βαθμωτών
 - Πράξεις μεταξύ διανυσμάτων και μητρώων
 - Χρήσιμα μητρώα και συναρτήσεις βιβλιοθήκης
 - Μεταβολές των στοιχείων μητρώου και διανύσματος
- 3 Προγραμματισμός στο MATLAB
 - Η δομή if
 - Η δομή for
 - Η δομή while
 - M-files
- 4 Παραγωγή γραφικών
 - Η συνάρτηση plot
 - Τρισδιάστατα γραφήματα

Περιεχόμενα II

5 Βιβλιογραφία

Έστω το μητρώο A με στοιχεία a_{ij} , αν πληκτρολογήσουμε την εντολή:

$A(i, j)$: εμφανίζεται στην οθόνη το a_{ij} στοιχείο, δηλαδή αυτό που βρίσκεται στη γραμμή i και τη στήλη j .

$A(:, j)$: εμφανίζεται στην οθόνη ένα διάνυσμα στήλη που περιλαμβάνει τα στοιχεία της j στήλης του μητρώου A

$A(i, :)$: εμφανίζεται στην οθόνη ένα διάνυσμα γραμμή που περιλαμβάνει τα στοιχεία της i γραμμής του μητρώου A

$A(k_1 : k_2, l_1 : l_2)$: εμφανίζεται στην οθόνη το υπομητρώο του A που ορίζεται από τις γραμμές k_1 έως k_2 και τις στήλες l_1 έως l_2 .

- Πρόσθεση και αφαίρεση

Μπορούμε να προσθέσουμε ή να αφαιρέσουμε έναν αριθμό x από όλα τα στοιχεία ενός μητρώου A και ενός διανύσματος u .

$$A + x \quad \text{και} \quad u + x \quad (\text{Πρόσθεση})$$

$$A - x \quad \text{και} \quad u - x \quad (\text{Αφαίρεση})$$

- Πολλαπλασιασμός και διαίρεση

Μπορούμε να πολλαπλασιάσουμε και να διαιρέσουμε όλα τα στοιχεία ενός πίνακα A και ενός διανύσματος u με ένα αριθμό x .

$$x * A \quad \text{και} \quad x * u \quad (\text{Πολλαπλασιασμός})$$

$$x/A \quad \text{και} \quad x/u \quad (\text{Δεξιά διαίρεση})$$

Οι πράξεις μεταξύ μητρώων και διανυσμάτων γίνονται με τα σύμβολα που φαίνονται στον πίνακα.

Πράξη	Σύμβολο
Πρόσθεση	+
Αφαίρεση	-
Πολλαπλασιασμός	*
Δεξιά διαίρεση	/
Αριστερή διαίρεση	\
Ύψωση σε δύναμη	^

Οι πράξεις μεταξύ μητρώων και διανυσμάτων γίνονται με τα σύμβολα που φαίνονται στον πίνακα.

Πράξη	Σύμβολο
Πρόσθεση	+
Αφαίρεση	-
Πολλαπλασιασμός	*
Δεξιά διαίρεση	/
Αριστερή διαίρεση	\
Ύψωση σε δύναμη	^

- Η αριστερή διαίρεση $A \setminus B$ είναι ισοδύναμη με τη διαίρεση $B : A$

Οι πράξεις μεταξύ μητρώων και διανυσμάτων γίνονται με τα σύμβολα που φαίνονται στον πίνακα.

Πράξη	Σύμβολο
Πρόσθεση	+
Αφαίρεση	-
Πολλαπλασιασμός	*
Δεξιά διαίρεση	/
Αριστερή διαίρεση	\
Ύψωση σε δύναμη	^

- Η αριστερή διαίρεση $A \setminus B$ είναι ισοδύναμη με τη διαίρεση $B : A$
- Η δεξιά διαίρεση A / B είναι ισοδύναμη με τη διαίρεση $A : B$

Οι πράξεις μεταξύ μητρώων και διανυσμάτων γίνονται με τα σύμβολα που φαίνονται στον πίνακα.

Πράξη	Σύμβολο
Πρόσθεση	+
Αφαίρεση	-
Πολλαπλασιασμός	*
Δεξιά διαίρεση	/
Αριστερή διαίρεση	\
Ύψωση σε δύναμη	^

- Η αριστερή διαίρεση $A \setminus B$ είναι ισοδύναμη με τη διαίρεση $B : A$
- Η δεξιά διαίρεση A / B είναι ισοδύναμη με τη διαίρεση $A : B$
- Η ύψωση μητρώου σε δύναμη υπάγεται στους περιορισμούς του γινομένου μητρώων.

- `eye` δημιουργεί ένα $m \times n$ μητρώο με 1 στην κεντρική διαγώνιο
- `zeros` δημιουργεί ένα $m \times n$ μητρώο που όλα του τα στοιχεία είναι 0
- `ones` δημιουργεί ένα $m \times n$ μητρώο που όλα του τα στοιχεία είναι 1
- `rand` δημιουργεί ένα $m \times n$ μητρώο με τυχαία στοιχεία επιλεγμένα από μια ομοιόμορφη κατανομή στο διάστημα $[0, 1]$.
- `randn` δημιουργεί ένα $m \times n$ μητρώο με τυχαία στοιχεία επιλεγμένα από τη συνήθη κανονική κατανομή με μέση τιμή το 0, διασπορά 1 και τυπική απόκλιση 1.

Η μορφή που πρέπει να έχουν οι παραπάνω εντολές είναι:

όνομα εντολής(m, n)

όπου m το πλήθος των γραμμών και n το πλήθος των στηλών του μητρώου που θέλουμε να κατασκευάσουμε.

Συνάρτηση	Διεργασία
max	μέγιστο στοιχείο διανύσματος
min	ελάχιστο στοιχείο διανύσματος
length	μήκος διανύσματος
sort	ταξινόμηση σε αύξουσα σειρά
sum	άθροισμα στοιχείων διανύσματος
prod	γινόμενο στοιχείων
norm	νόρμα διανύσματος ή μητρώου
median	διάμεσος
mean	μέση τιμή
std	τυπική απόκλιση
diag	διαγώνιο μητρώο ή η διαγώνιος μητρώου
size	μέγεθος μητρώου
det	ορίζουσα τετραγωνικού μητρώου
trace	ίχνος μητρώου
rank	βαθμός μητρώου
inv	αντίστροφος αντιστρέψιμου μητρώου
eig	ιδιοτιμές και ιδιοδιανύσματα μητρώου

Εκχώρηση τιμής σε μητρώο ή διάνυσμα

$x(i) = 0$: εκχωρεί την τιμή 0 στην i θέση του διανύσματος x .

$A(i,j) = 0$: εκχωρεί την τιμή 0 στην (i,j) θέση του μητρώου A .

$A(i,:) = 0$: εκχωρεί την τιμή 0 στην i γραμμή του μητρώου A .

$A(:,j) = 0$: εκχωρεί την τιμή 0 στην j στήλη του μητρώου A .

$A(k_1 : k_2, l_1 : l_2) = 0$: εκχωρεί την τιμή 0 στο υπομητρώο του A που ορίζεται από τις γραμμές k_1 έως k_2 και τις στήλες l_1 έως l_2 .

Πράξεις κατά τα στοιχεία διανύσματος ή μητρώου

Ο **πολλαπλασιασμός**, η **διαίρεση** και η **ύψωση σε δύναμη** μπορεί να γίνεται στα **στοιχεία** ενός πίνακα **ένα προς ένα**, αρκεί πριν από το σύμβολο της πράξης να εμφανίζεται μια τελεία. Δηλαδή, η εντολή

μεταβλητή **.τελεστής** αριθμός (1)

δημιουργεί το μητρώο ή διάνυσμα του οποίου τα στοιχεία είτε θα πολλαπλασιαστούν με τον ίδιο αριθμό, είτε θα διαιρεθούν με τον ίδιο αριθμό, είτε θα υψωθούν εις τον εαυτό τους.

Παράδειγμα

Πράξεις κατά τα στοιχεία διανύσματος ή μητρώου.

```

1  A =
2      1      2      5      6
3      3      4      7      8
4      9      0      3      4
5  >> B=2.*A
6  B =
7      2      4     10     12
8      6      8     14     16
9     18      0      6      8
10 >> C=A./3
11 C =
12     0.3333     0.6667     1.6667     2.0000
13     1.0000     1.3333     2.3333     2.6667
14     3.0000           0     1.0000     1.3333
15 >> D=A.^4
16 D =
17           1           16           625           1296
18          81          256          2401          4096
19         6561           0           81           256

```

Περιεχόμενα I

- 1 Εισαγωγή
 - Μεταβλητές
 - Τελεστές
 - Είσοδος και έξοδος δεδομένων
 - Εντολές διαχείρισης χώρου εργασίας
- 2 Διανύσματα και μητρώα
 - Πράξεις μεταξύ μητρώων και βαθμωτών
 - Πράξεις μεταξύ διανυσμάτων και μητρώων
 - Χρήσιμα μητρώα και συναρτήσεις βιβλιοθήκης
 - Μεταβολές των στοιχείων μητρώου και διανύσματος
- 3 Προγραμματισμός στο MATLAB
 - Η δομή if
 - Η δομή for
 - Η δομή while
 - M-files
- 4 Παραγωγή γραφικών
 - Η συνάρτηση plot
 - Τρισδιάστατα γραφήματα

Περιεχόμενα II

5 Βιβλιογραφία

Η πιο συνήθης εντολή ελέγχου ροής είναι η **if** με σύνταξη:

```
if      συνθήκη 1  
        Εντολές  
elseif  συνθήκη 2  
        Εντολές  
else  
        Εντολές  
end
```


Η πιο συνήθης εντολή ελέγχου ροής είναι η `if` με σύνταξη:

```
if      συνθήκη 1
        Εντολές
elseif  συνθήκη 2
        Εντολές
else
        Εντολές
end
```

Παράδειγμα

Τι εκτελεί το παρακάτω παράδειγμα;

```
1 >> x=1:3;
2 >> y=[0 -1 1];
3 >> if (x>0) | (y>0)
4     z=x+y
5     end
6 z =
7     1     1     4
```

Η πιο συνηθής εντολή ελέγχου ροής είναι η *if* με σύνταξη:

```
if      συνθήκη 1
        Εντολές
elseif  συνθήκη 2
        Εντολές
else
        Εντολές
end
```

Συγκρίνει τις τιμές δύο διανυσμάτων με το 0 και όταν μια από αυτές είναι μεγαλύτερες από το 0, τις αθροίζει.

Παράδειγμα

Τι εκτελεί το παρακάτω παράδειγμα;

```
1 >> x=1:3;
2 >> y=[0 -1 1];
3 >> if (x>0) | (y>0)
4     z=x+y
5     end
6 z =
7     1     1     4
```

Η εντολή **for** εκτελεί διαδοχικά τις εντολές του βρόγχου **για** συγκεκριμένο αριθμό επαναλήψεων. Η σύνταξή της είναι η ακόλουθη:

```
for μετρητής = αρχή : βήμα : τέλος
    Εντολές
end
```

Η εντολή `for` εκτελεί διαδοχικά τις εντολές του βρόγχου `για` συγκεκριμένο αριθμό επαναλήψεων. Η σύνταξή της είναι η ακόλουθη:

```
for μετρητής = αρχή : βήμα : τέλος
    Εντολές
end
```

Παράδειγμα

Τι εκτελεί το παρακάτω παράδειγμα;

```
1 >> for i=1:5
2     for j=1:5
3         A(i,j)=(i+j)^3-2*i*j;
4     end
5 end
6 >> A
7 A =
8     6     23     58    117    206
9     23     56    113    200    323
10    58    113    198    319    482
11   117    200    319    480    689
12   206    323    482    689    950
```

Η εντολή `for` εκτελεί διαδοχικά τις εντολές του βρόγχου για συγκεκριμένο αριθμό επαναλήψεων. Η σύνταξή της είναι η ακόλουθη:

`for` μετρητής = αρχή : βήμα : τέλος
 Εντολές
`end`

Υπολογίζει τις τιμές των στοιχείων ενός μητρώου A διάστασης 5×5 συναρτήσει των συντεταγμένων του μέσω της σχέσης

$$A(i,j) = (i+j)^3 - 2 \cdot i \cdot j.$$

Παράδειγμα

Τι εκτελεί το παρακάτω παράδειγμα;

```

1 >> for i=1:5
2     for j=1:5
3         A(i,j)=(i+j)^3-2*i*j;
4     end
5 end
6 >> A
7 A =
8     6     23     58    117    206
9     23     56    113    200    323
10    58    113    198    319    482
11   117    200    319    480    689
12   206    323    482    689    950

```

Η εντολή **while** εκτελεί διαδοχικά τις εντολές του βρόχου **όσο** ισχύει η συνθήκη που έχει καθορίσει ο χρήστης. Η σύνταξή της είναι η ακόλουθη:

```
while  συνθήκη
      Εντολές
end
```

Η εντολή **while** εκτελεί διαδοχικά τις εντολές του βρόχου **όσο** ισχύει η συνθήκη που έχει καθορίσει ο χρήστης. Η σύνταξή της είναι η ακόλουθη:

```
while  συνθήκη
      Εντολές
end
```

Παράδειγμα

Τι ενέργεια επιτελεί το παρακάτω παράδειγμα;

```
1 >> s=0;
2 >> x=5;
3 >> while s<50
4     x=x+2;
5     s=s+1;
6 end
7 >> x
8
9 x =
10
11     105
```

Η εντολή **while** εκτελεί διαδοχικά τις εντολές του βρόχου **όσο** ισχύει η συνθήκη που έχει καθορίσει ο χρήστης. Η σύνταξή της είναι η ακόλουθη:

```
while  συνθήκη
      Εντολές
end
```

Αυξάνει την τιμή της μεταβλητής x κατά 2 και της μεταβλητής s κατά 1 όσο η μεταβλητή s έχει τιμή μικρότερη του 50.

Παράδειγμα

Τι ενέργεια επιτελεί το παρακάτω παράδειγμα;

```
1 >> s=0;
2 >> x=5;
3 >> while s<50
4     x=x+2;
5     s=s+1;
6 end
7 >> x
8
9 x =
10
11     105
```


Τα **m-files** διακρίνονται σε:

- Αρχεία εντολών (**script**), δεν έχουν ορίσματα εισόδου και εξόδου αλλά εκτελούν μια ακολουθία εντολών σε μεταβλητές του χώρου εργασίας. **Αν και τα script δεν επιστρέφουν τιμές, όποια μεταβλητή δημιούργησαν παραμένει στο workspace.**
- Αρχεία συναρτήσεων (**function m-files**), περιλαμβάνουν μια γραμμή ορισμού συνάρτησης, δέχονται **ορίσματα εισόδου** και επιστρέφουν **μεταβλητές εξόδου**. Το όνομα μιας νέας function **δεν** πρέπει να συμπίπτει με όνομα συνάρτησης βιβλιοθήκης. Η μορφή που έχουν τα αρχεία συναρτήσεων είναι η εξής:

```
function [o1, o2, ...] = name(i1, i2, ...)  
    Εντολές  
end
```

όπου:

o_i: οι μεταβλητές εξόδου

name: το όνομα που δίνουμε στη συνάρτηση

i_j: οι μεταβλητές εισόδου

Τα **m-files** διακρίνονται σε:

- Αρχεία εντολών (**script**), δεν έχουν ορίσματα εισόδου και εξόδου αλλά εκτελούν μια ακολουθία εντολών σε μεταβλητές του χώρου εργασίας. **Αν και τα script δεν επιστρέφουν τιμές, όποια μεταβλητή δημιούργησαν παραμένει στο workspace.**
- Αρχεία συναρτήσεων (**function m-files**), περιλαμβάνουν μια γραμμή ορισμού συνάρτησης, δέχονται **ορίσματα εισόδου** και επιστρέφουν **μεταβλητές εξόδου**. Το όνομα μιας νέας function **δεν** πρέπει να συμπίπτει με όνομα συνάρτησης βιβλιοθήκης. Η μορφή που έχουν τα αρχεία συναρτήσεων είναι η εξής:

```
function [o1, o2, ...] = name(i1, i2, ...)  
    Εντολές  
end
```

όπου:

o_i : οι μεταβλητές εξόδου

name: το όνομα που δίνουμε στη συνάρτηση

i_j : οι μεταβλητές εισόδου

Τόσο τα αρχεία script όσο και τα αρχεία συναρτήσεων:

- 1 Δημιουργούνται ξεχωριστά με κάποιο συντάκτη (editor), όπως ο notepad ή ο wordpad ή ακόμα με το συντάκτη του MATLAB.

Τόσο τα αρχεία script όσο και τα αρχεία συναρτήσεων:

- 1 Δημιουργούνται ξεχωριστά με κάποιο συντάκτη (editor), όπως ο notepad ή ο wordpad ή ακόμα με το συντάκτη του MATLAB.
- 2 **Πρέπει** να βρίσκονται στο φάκελο εργασίας ([working directory](#)) ή στον φάκελο (directory) του MATLAB.

Τόσο τα αρχεία script όσο και τα αρχεία συναρτήσεων:

- 1 Δημιουργούνται ξεχωριστά με κάποιο συντάκτη (editor), όπως ο notepad ή ο wordpad ή ακόμα με το συντάκτη του MATLAB.
- 2 **Πρέπει** να βρίσκονται στο φάκελο εργασίας (**working directory**) ή στον φάκελο (directory) του MATLAB.
- 3 Μπορούν να καλούν **άλλα m-files** ή ακόμα τον ίδιο τον **εαυτό τους**.

Περιεχόμενα I

- 1 Εισαγωγή
 - Μεταβλητές
 - Τελεστές
 - Είσοδος και έξοδος δεδομένων
 - Εντολές διαχείρισης χώρου εργασίας
- 2 Διανύσματα και μητρώα
 - Πράξεις μεταξύ μητρώων και βαθμωτών
 - Πράξεις μεταξύ διανυσμάτων και μητρώων
 - Χρήσιμα μητρώα και συναρτήσεις βιβλιοθήκης
 - Μεταβολές των στοιχείων μητρώου και διανύσματος
- 3 Προγραμματισμός στο MATLAB
 - Η δομή if
 - Η δομή for
 - Η δομή while
 - M-files
- 4 Παραγωγή γραφικών
 - Η συνάρτηση plot
 - Τρισδιάστατα γραφήματα

Περιεχόμενα II

5 Βιβλιογραφία

- Χρησιμοποιείται για τη δημιουργία της γραφικής αναπαράστασης μιας μονοδιάστατης συνάρτησης (`plot()`).

- Χρησιμοποιείται για τη δημιουργία της γραφικής αναπαράστασης μιας μονοδιάστατης συνάρτησης (`plot()`).
- Απαραίτητη προϋπόθεση είναι η ύπαρξη **δύο διανυσμάτων** (x, y) **ίσου μήκους**, που να περιέχουν τα σημεία των οποίων θέλουμε να δημιουργήσουμε τη γραφική αναπαράσταση.

- Χρησιμοποιείται για τη δημιουργία της γραφικής αναπαράστασης μιας μονοδιάστατης συνάρτησης (`plot()`).
- Απαραίτητη προϋπόθεση είναι η ύπαρξη **δύο διανυσμάτων** (x, y) **ίσου μήκους**, που να περιέχουν τα σημεία των οποίων θέλουμε να δημιουργήσουμε τη γραφική αναπαράσταση.
- Χρήσιμες εντολές για τις γραφικές αναπαραστάσεις παρουσιάζονται παρακάτω:

`xlabel("onoma1")`: ετικέτα για τον οριζόντιο άξονα

`ylabel("onoma2")`: ετικέτα για τον κατακόρυφο άξονα

`title("onoma3")`: ο τίτλος της γραφικής αναπαράστασης

`legend("onoma4")`: η λεζάντα της γραφικής αναπαράστασης

`text(x_i, y_i , "string")`: προσθήκη κειμένου στη θέση (x_i, y_i)

`grid`: δημιουργία πλέγματος

`figure`: άνοιγμα (άλλου) παραθύρου γραφικών

`hold`: πάγωμα του τρέχοντος παραθύρου γραφικών για το σχεδιασμό και άλλων καμπυλών

- Χρησιμοποιείται για τη δημιουργία της γραφικής αναπαράστασης μιας μονοδιάστατης συνάρτησης (`plot()`).
- Απαραίτητη προϋπόθεση είναι η ύπαρξη **δύο διανυσμάτων** (x, y) **ίσου μήκους**, που να περιέχουν τα σημεία των οποίων θέλουμε να δημιουργήσουμε τη γραφική αναπαράσταση.
- Χρήσιμες εντολές για τις γραφικές αναπαραστάσεις παρουσιάζονται παρακάτω:

`xlabel("onoma1")`: ετικέτα για τον οριζόντιο άξονα

`ylabel("onoma2")`: ετικέτα για τον κατακόρυφο άξονα

`title("onoma3")`: ο τίτλος της γραφικής αναπαράστασης

`legend("onoma4")`: η λεζάντα της γραφικής αναπαράστασης

`text(x_i, y_i , "string")`: προσθήκη κειμένου στη θέση (x_i, y_i)

`grid`: δημιουργία πλέγματος

`figure`: άνοιγμα (άλλου) παραθύρου γραφικών

`hold`: πάγωμα του τρέχοντος παραθύρου γραφικών για το σχεδιασμό και άλλων καμπυλών

Να δημιουργήσετε τη γραφική αναπαράσταση της $y = x^2$ στο διάστημα $[-10,10]$

Παράδειγμα

```
1 x=[-10:1:10];
2 y=x.^2;
3 plot(x,y)
4 xlabel("x")
5 ylabel("y")
6 title("grafikh anaparastash")
7 legend("y=x^2")
```

Κοινά γραφήματα

- Με την εντολή `plot` μπορούμε να έχουμε διαφορετικές καμπύλες στο ίδιο γράφημα.

Κοινά γραφήματα

- Με την εντολή `plot` μπορούμε να έχουμε διαφορετικές καμπύλες στο ίδιο γράφημα.
- Η εντολή `plot` παρέχει μια ευρεία **επιλογή χρωμάτων, συμβόλων και τύπων γραμμών** μέσω της εντολής

$$\text{plot}(x, y, \text{' [χρώμα][σύμβολο][γραμμή]'})$$

Χρώμα		Σύμβολο		Γραμμή
b	μπλε	.	point	- συνεχής
g	πράσινο	o	circle	: λεπτή διακεκομμένη
r	κόκκινο	x	x-mark	- αδρή διακεκομμένη
c	κυανό	+	plus	-. διακεκομμένη-τελείες
m	μοβ	*	star	
y	κίτρινο	\$	square	
k	μαύρο	d	diamond	
w	άσπρο	v	triangle (down)	
		^	triangle (up)	
		<	triangle (left)	
		>	triangle (right)	
		p	pentagram	
		h	hexagram	

- Να δημιουργήσετε την **κοινή γραφική** αναπαράσταση των $y = \sin(x)$, $y = \cos(x)$ και $y = \tan(x)$ στο διάστημα $[-10,10]$

- Να δημιουργήσετε την κοινή γραφική αναπαράσταση των $y = \sin(x)$, $y = \cos(x)$ και $y = \tan(x)$ στο διάστημα $[-10,10]$

Παράδειγμα

```
1 x=[-10:0.5:10];
2 y1=sin(x);
3 y2=cos(x);
4 y3=tan(x);
5 plot(x,y1,"b.-",x,y2,"go--",x,y3,"rx-.")
6 xlabel("x")
7 ylabel("y")
8 title("koinh grafikh anaparastash sin cos tan")
9 legend("sin(x)", "cos(x)", "tan(x)")
```



Η συνάρτηση surf

Για να δημιουργήσουμε «**τρισδιάστατο**» γράφημα θα χρειαστούμε τις παρακάτω εντολές:

```
[x, y] = meshgrid(διάστημα για το x, διάστημα για το y);  
z = τύπος συνάρτησης;  
surf(x, y, z)
```

Η συνάρτηση surf

Για να δημιουργήσουμε «**τρισδιάστατο**» γράφημα θα χρειαστούμε τις παρακάτω εντολές:

```
[x, y] = meshgrid(διάστημα για το x, διάστημα για το y);  
z = τύπος συνάρτησης;  
surf(x, y, z)
```

Να δημιουργήσετε τη γραφική αναπαράσταση για τη συνάρτηση $f(x, y) = 2 \cdot x^2 \cdot y + x + y^3$ στο διάστημα $[-4, 4] \times [-4, 4]$.

Η συνάρτηση surf

Για να δημιουργήσουμε «**τρισδιάστατο**» γράφημα θα χρειαστούμε τις παρακάτω εντολές:

```
[x, y] = meshgrid(διάστημα για το x, διάστημα για το y);  
z = τύπος συνάρτησης;  
surf(x, y, z)
```

Να δημιουργήσετε τη γραφική αναπαράσταση για τη συνάρτηση $f(x, y) = 2 \cdot x^2 \cdot y + x + y^3$ στο διάστημα $[-4, 4] \times [-4, 4]$.

Παράδειγμα

```
1 [x, y] = meshgrid(-5:0.1:5, -5:0.1:5);  
2 z = 2.*x.^2.*y+x+y.^3;  
3 surf(x, y, z)
```

Ισοϋψείς καμπύλες

$[x, y] = \text{meshgrid}(\text{διάστημα για το } x, \text{διάστημα για το } y);$
 $z = \text{τύπος συνάρτησης};$
 $\text{contour}(x, y, z)$

$[x, y] = \text{meshgrid}(\text{διάστημα για το } x, \text{διάστημα για το } y);$
 $z = \text{τύπος συνάρτησης};$
 $\text{contourf}(x, y, z)$

Ισοϋψείς καμπύλες

$[x, y] = \text{meshgrid}(\text{διάστημα για το } x, \text{διάστημα για το } y);$
 $z = \text{τύπος συνάρτησης};$
 $\text{contour}(x, y, z)$

$[x, y] = \text{meshgrid}(\text{διάστημα για το } x, \text{διάστημα για το } y);$
 $z = \text{τύπος συνάρτησης};$
 $\text{contourf}(x, y, z)$

Να δημιουργήσετε τα γραφήματα ισοϋψών για τη συνάρτηση $f(x, y) = 2 \cdot x^2 \cdot y + x + y^3$ στο διάστημα $[-10, 10] \times [-10, 10]$.

Ισοϋψείς καμπύλες

$[x, y] = \text{meshgrid}(\text{διάστημα για το } x, \text{ διάστημα για το } y);$
 $z = \text{τύπος συνάρτησης};$
 $\text{contour}(x, y, z)$

$[x, y] = \text{meshgrid}(\text{διάστημα για το } x, \text{ διάστημα για το } y);$
 $z = \text{τύπος συνάρτησης};$
 $\text{contourf}(x, y, z)$

Να δημιουργήσετε τα γραφήματα ισοϋψών για τη συνάρτηση $f(x, y) = 2 \cdot x^2 \cdot y + x + y^3$ στο διάστημα $[-10, 10] \times [-10, 10]$.

Παράδειγμα

```
1 [x,y] = meshgrid(-10:0.1:10, -10:0.1:10);
2 z = 2.*x.^2.*y+x+y.^3;
3 subplot(1,2,1), contour(x,y,z), title("contour")
4 subplot(1,2,2), contourf(x,y,z), title("contourf")
```

Περιεχόμενα I

- 1 Εισαγωγή
 - Μεταβλητές
 - Τελεστές
 - Είσοδος και έξοδος δεδομένων
 - Εντολές διαχείρισης χώρου εργασίας
- 2 Διανύσματα και μητρώα
 - Πράξεις μεταξύ μητρώων και βαθμωτών
 - Πράξεις μεταξύ διανυσμάτων και μητρώων
 - Χρήσιμα μητρώα και συναρτήσεις βιβλιοθήκης
 - Μεταβολές των στοιχείων μητρώου και διανύσματος
- 3 Προγραμματισμός στο MATLAB
 - Η δομή if
 - Η δομή for
 - Η δομή while
 - M-files
- 4 Παραγωγή γραφικών
 - Η συνάρτηση plot
 - Τρισδιάστατα γραφήματα

Περιεχόμενα II

5 Βιβλιογραφία

Βιβλιογραφία - Αναφορές

- Gilat A., MATLAB: An Introduction with Applications, Second Edition, John Wiley & Sons, 2004.
- Chapra S.C., Applied Numerical Methods with MATLAB for Engineering and Science with Engineering Subscription Card, McGraw-Hill, 2004.
- Hunt B.R., Lipsman R.L., Rosenberg J.M., Coombes K.R., Osborn J.E. and Stuck G.J., A guide to MATLAB for beginners and experienced users, Second Edition, Cambridge University Press, 2006.
- Quarteroni A. and Saleri F, Scientific Computing with MATLAB and OCTAVE, Springer, 2006.
- Malek-Madani R., Advanced Engineering Mathematics with Mathematica and MATLAB, Pearson Higher Education, 1998.

Ευχαριστώ πολύ για την προσοχή σας!

