

**Μια Σύντομη  
Εισαγωγή  
στη  
MAPLE**



**Χρίστος Ξενοφώντας & Γεώργιος Γεωργίου**  
Τμήμα Μαθηματικών και Στατιστικής  
Πανεπιστήμιο Κύπρου

Μάιος 2007

## 1. Εισαγωγή

Η MAPLE είναι ένα σύγχρονο, **διαδραστικό** (interactive) μαθηματικό λογισμικό πακέτο για **συμβολικούς** και αριθμητικούς υπολογισμούς, όπως και για κατασκευή γραφημάτων, που χρησιμοποιείται σε πανεπιστημιακά μαθήματα αλλά και σε ερευνητικές και άλλες εφαρμογές. Ξεκίνησε σαν ένα ερευνητικό πρόγραμμα στο πανεπιστήμιο του Waterloo στον Καναδά πριν περίπου 25 χρόνια, και σήμερα έχει εξελιχθεί στην εταιρεία MapleSoft και έχει καθιερωθεί σαν ένα από τα καλύτερα λογισμικά πακέτα για συμβολικούς υπολογισμούς.

Στόχος αυτών των σημειώσεων είναι η εξοικείωση του αναγνώστη με τα χαρακτηριστικά και τις πιο βασικές εντολές της MAPLE. Δεν θα ασχοληθούμε καθόλου με τρόπους καταχώρησης αρχείων ή την κατασκευή αρχείων εργασίας (worksheets), αν και αυτά πολλές φορές είναι χρήσιμα. Αν και η MAPLE έχει πολύ καλές δυνατότητες προγραμματισμού (δηλ. κατασκευής προγραμμάτων που έχουν δεδομένα εισόδου και εξόδου, και περιέχουν βρόχους, εντολές if, κλπ.), εμείς δεν θα μιλήσουμε για αυτά, εκτός από μια μικρή αναφορά στο τέλος των σημειώσεων.

Επισημάνουμε ότι ο καλύτερος (και ουσιαστικά ο μόνος) τρόπος εκμάθησης της MAPLE είναι η συστηματική ενασχόληση με αυτή και η διερεύνησή της από τον ίδιο τον χρήστη. Το πακέτο είναι εφοδιασμένο με ένα εκτενές σύστημα βοήθειας όπου κάθε εντολή επεξηγείται αναλυτικά και με αντιπροσωπευτικά παραδείγματα.

Για περισσότερες πληροφορίες επισκεφτείτε την ιστοσελίδα της εταιρείας MapleSoft

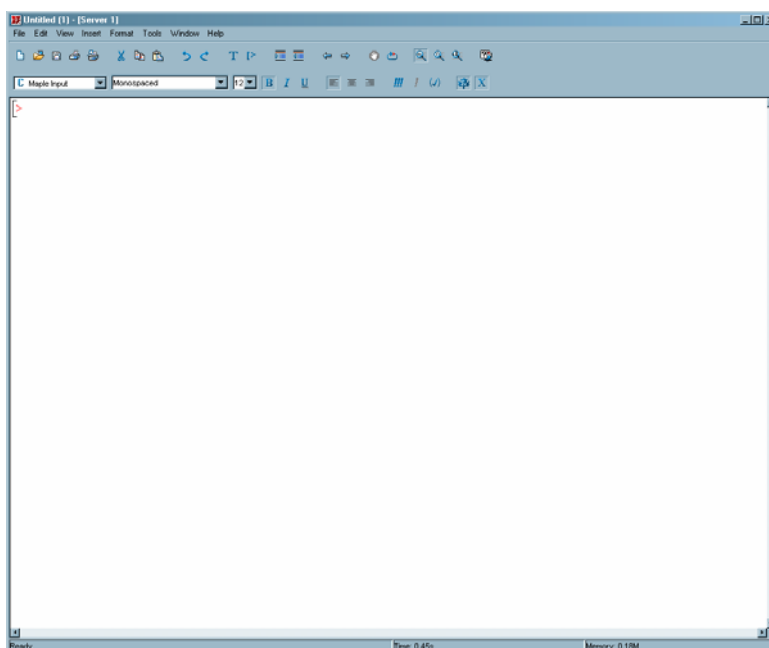
<http://www.maplesoft.com/>

## 2. Ξεκινώντας με τη MAPLE

Για να χρησιμοποιήσουμε τη MAPLE πρέπει να την εγκαταστήσουμε πρώτα στον υπολογιστή μας. Το εικονίδιο του πακέτου για την εκδοχή 9 φαίνεται πιο κάτω:



Μπορούμε να ξεκινήσουμε το πρόγραμμα με διπλό κλικ πάνω στο εικονίδιο αυτό<sup>1</sup>. Μετά από λίγο, αφού το πρόγραμμα φορτώσει, θα εμφανιστεί στην οθόνη μας το παράθυρο έναρξης της MAPLE:



Η έξοδος από το πρόγραμμα μπορεί να γίνει με τους εξής τρόπους:

- με κλικ στο τετραγωνίδιο [x] που βρίσκεται πάνω δεξιά στο παράθυρο της MAPLE, και
- με την επιλογή *File* → *Exit* στο παράθυρο εργασίας.

Τα παραδείγματα που παραθέτουμε έχουν γίνει με την εκδοχή 9 της MAPLE.

### Βασικές πράξεις

Η MAPLE μπορεί να χρησιμοποιηθεί σαν απλή αριθμομηχανή. Για τις βασικές πράξεις χρησιμοποιούνται τα σύμβολα που φαίνονται στον πιο κάτω πίνακα:

Σύμβολο	Πράξη
+	Πρόσθεση
-	Αφαίρεση
*	Πολλαπλασιασμός
/	Διαίρεση
^	Ύψωση σε δύναμη

<sup>1</sup> Αν χρησιμοποιούμε unix, γράφουμε `xmaple` στη γραμμή εντολής.

**ΣΗΜΕΙΩΣΗ:**

Μετά από κάθε εντολή, πρέπει να βάζουμε ερωτηματικό ( ; ) για να πάρουμε αποτέλεσμα. Αυτό αντιστοιχεί στο να ‘ζητάμε από την MAPLE να επεξεργαστεί’ την εντολή που γράψαμε.

**Παράδειγμα**

```
> 2+3;
```

```
5
```

```
> 4*12
```

```
Warning, inserted missing semicolon at end of statement,
4*12;
```

```
48
```

Επειδή δεν βάλουμε ερωτηματικό στο τέλος της δεύτερης εντολής, η MAPLE το βάζει από μόνη της και μας προειδοποιεί για τυχόν λάθος. Σε παλιότερες εκδοχές της MAPLE, αν γινόταν αυτό το λάθος τότε θα παίρναμε μια προειδοποίηση από το πρόγραμμα και δεν θα βλέπαμε την απάντηση μέχρι που να γράφαμε το ερωτηματικό.

**Προτεραιότητα πράξεων**

Όπως και στις γλώσσες προγραμματισμού FORTRAN και C, η MAPLE ακολουθεί τους συνήθεις αλγεβρικούς κανόνες για την σειρά εκτέλεσης πράξεων:

1. Πρώτα εκτελούνται οι πράξεις μέσα σε παρενθέσεις από τα μέσα προς τα έξω.
2. Μετά εκτελούνται οι υψώσεις σε δύναμη.
3. Μετά εκτελούνται οι πολλαπλασιασμοί και διαιρέσεις από τα αριστερά προς τα δεξιά.
4. Τέλος, εκτελούνται οι προσθέσεις και αφαιρέσεις από τα αριστερά προς τα δεξιά.

**Παράδειγμα**

```
> 6-4*5^2;
```

```
-94
```

```
> ((6-4)*5)^2;
```

```
100
```

Μπορούμε να κάνουμε πολλές άλλες μαθηματικές πράξεις χρησιμοποιώντας εντολές βιβλιοθήκης, όπως για παράδειγμα

```
> sqrt(2);
```

```
 $\sqrt{2}$ 
```

Παρατηρούμε ότι η MAPLE δεν μας έδωσε σαν απάντηση την ρίζα του 2 σε δεκαδική μορφή, αλλά την συμβολική της μορφή, μια και είναι λογισμικό που μας επιτρέπει να κάνουμε συμβολικές πράξεις. Έτσι, έχουμε επίσης

```
> 2^72; (2/5)^48; 3^(1/2);
```

```
4722366482869645213696
281474976710656
-----
3552713678800500929355621337890625
      √3
```

Από τα πιο πάνω βλέπουμε επίσης ότι έχουμε τη δυνατότητα να γράφουμε πολλές εντολές σε μια γραμμή αφού, φυσικά, μετά από κάθε εντολή βάλουμε ερωτηματικό. Η κάθε απάντηση δίνεται σε ξεχωριστή γραμμή.

Μπορούμε να δούμε την δεκαδική μορφή ενός αποτελέσματος με δύο τρόπους:

Γράφουμε τους αριθμούς που εμπλέκονται στις πράξεις σε δεκαδική μορφή, όπως για παράδειγμα,

```
> (2.0/5.0)^48;
```

```
7.9228162514264337594 10^-20
```

ή, χρησιμοποιούμε την εντολή **evalf** (που σημαίνει *evaluate floating point*), όπως για παράδειγμα,

```
> evalf(3^(1/2));
```

```
1.7320508075688772935
```

Με αυτή την εντολή, μπορούμε επίσης να προσδιορίσουμε των αριθμό των δεκαδικών ψηφίων που περιέχονται στην απάντηση, όπως για παράδειγμα

```
> evalf(3^(1/2),5);
```

```
1.7321
```

```
> evalf(3^(1/2),52);
```

```
1.732050807568877293527446341505872366942805253810381
```

### Ειδικές Σταθερές

Η MAPLE γνωρίζει τις ειδικές σταθερές που εμφανίζονται στα μαθηματικά, όπως για παράδειγμα  $\pi$  (= Pi),  $e$  (= exp(1)),  $i = \sqrt{-1}$  (= sqrt(-1)), κλπ.

```
> Pi; evalf(Pi,25);
```

```
π
```

```
3.141592653589793238462643
```

```
> evalf(exp(1));
```

```
2.7182818284590452354
```

```
> sqrt(-1);
```

```
i
```

### Τυπογραφικά λάθη

Τα τυπογραφικά λάθη είναι αναπόφευκτα, και η MAPLE πολλές φορές τα βρίσκει για μας:

```
> evalf( 2/3)^2);
Error, `)` unexpected
```

Εδώ, η σωστή εντολή είναι

```
> evalf( (2/3)^2 );
0.44444444444444444444444444444444
```

Επίσης, η MAPLE κάνει διάκριση μεταξύ μικρών και κεφαλαίων γραμμμάτων, έτσι η εντολή

```
> Evalf(Pi);
Evalf(π)
```

δεν μας δίνει το αναμενόμενο αποτέλεσμα, ενώ η

```
> evalf(Pi);
3.1415926535897932385
```

μας το δίνει.

Μερικές φορές θέλουμε να χρησιμοποιήσουμε το αποτέλεσμα μιας πράξης σαν ‘δεδομένο εισόδου’ σε μια άλλη πράξη. Η MAPLE μας επιτρέπει να επεξεργαστούμε μια απάντηση χρησιμοποιώντας το σύμβολο **%**. Για παράδειγμα,

```
> sqrt(5);
√5
> (%)^2;
5
```

Αυτή η δυνατότητα είναι ιδιαίτερα χρήσιμη όταν κάνουμε πολλές πράξεις στη σειρά, αφού δεν θα χρειάζεται να ξαναγράψουμε την απάντηση κάθε φορά.

### Παραστάσεις και μεταβλητές

Ξαναυπενθυμίζουμε ότι η MAPLE είναι λογισμικό για συμβολικούς υπολογισμούς, άρα μπορούμε να πούμε

```
> x^2 - 3/x + 4*x;
x^2 - 3/x + 4x
```

χωρίς να πάρουμε προειδοποίηση ή λάθος από το πρόγραμμα (σε αντίθεση με την MATLAB η οποία δεν επιτρέπει την πιο πάνω εντολή χωρίς πρώτα τον ορισμό του  $x$  σαν κάποιο συγκεκριμένο αριθμό).

Ο ορισμός μιας παράστασης ή μιας μεταβλητής γίνεται με το σύμβολο **:=**. Για παράδειγμα,

```
> expression1 := x^2 - 6*x + 9;
expression1 := x^2 - 6x + 9
```

Σε περίπτωση που δεν θέλουμε να δούμε τι ορίσαμε (ή δεν θέλουμε να εμφανίζεται στο παράθυρο εργασίας το αποτέλεσμα μιας εντολής), τότε στο τέλος βάζουμε το σύμβολο **;** αντί για το **;**.

```
> expression1 := x^2 - 6*x + 9:
```

Οποτεδήποτε θέλουμε να δούμε την τιμή μιας ενεργής παράστασης, γράφουμε απλώς το όνομά της:

```
> expression1;
```

$$x^2 - 6x + 9$$

Τώρα, αν δώσουμε συγκεκριμένη τιμή στην μεταβλητή  $x$  που εμφανίζεται στην παράσταση **expression1**, τότε έχουμε

```
> x:=4;
```

$$x := 4$$

```
> expression1;
```

$$1$$

δηλ. η παράσταση έχει πλέον πάρει την αριθμητική τιμή που αντιστοιχεί στο  $x = 4$ . Σημειώνουμε ότι το  $x$  θα συνεχίζει να έχει την τιμή 4, εκτός αν πούμε

```
> restart;
```

που θα έχει σαν αποτέλεσμα την ανανέωση όλων των μεταβλητών και παραστάσεων που έχουμε ορίσει (κάτι που ίσως να μην είναι επιθυμητό). Ο καλύτερος τρόπος να διαγράψουμε μια συγκεκριμένη τιμή από μια μεταβλητή είναι ο εξής:

```
> x:='x';
```

$$x := x$$

Έτσι έχουμε

```
> expression1;
```

$$x^2 - 6x + 9$$

ακριβώς όπως και πριν δώσουμε την τιμή 4 στο  $x$ . Έτσι, για να αποφύγουμε τις πιο πάνω 'δυσκολίες', όταν θέλουμε να αντικαταστήσουμε μια συγκεκριμένη τιμή σε μια μεταβλητή μιας παράστασης, χρησιμοποιούμε την εντολή **subs**:

```
> subs(x = 4, expression1);
```

$$1$$

### Εντολές για απλοποιήσεις

Υπάρχουν διάφορες εντολές που μας επιτρέπουν να απλοποιήσουμε (και όχι μόνο) μια παράσταση. Εδώ θα δούμε τις πιο βασικές. Έστω ότι

```
> expression1 := x^2 - 6*x + 9:
```

Τότε, η εντολή **factor** γράφει την παράσταση **expression1** σαν γινόμενο, ενώ η εντολή **expand** κάνει τις πράξεις και αναπτύσσει το γινόμενο:

```
> factor(expression1);
```

$$(x - 3)^2$$

```
> expand(%);
```

$$x^2 - 6x + 9$$

**Παράδειγμα**

> **expand( (x+1)^7 ); factor(%);**

$$x^7 + 7x^6 + 21x^5 + 35x^4 + 35x^3 + 21x^2 + 7x + 1$$

$$(x+1)^7$$

Μια άλλη σημαντική εντολή είναι η **simplify**, η οποία προσπαθεί να κάνει απλοποιήσεις:

> **simplify( 1/x + 1/x^2 );**

$$\frac{x+1}{x^2}$$

> **simplify(cos(t)^2+sin(t)^2);**

$$1$$

> **simplify((a^2-b^2)/(a+b));**

$$a-b$$

**Σημείωση:**

Η εντολή **?** (από μόνη της) ανοίγει ένα παράθυρο με πληροφορίες για το πώς να πάρουμε βοήθεια (help) για τις διάφορες εντολές. Αν γράψουμε **?** και το όνομα μιας συγκεκριμένης εντολής, τότε η MAPLE μας δίνει βοήθεια για αυτή την εντολή. Δοκιμάστε, για παράδειγμα,

> **?simplify**

**Συναρτήσεις βιβλιοθήκης**

Η MAPLE είναι εφοδιασμένη με αρκετές **συναρτήσεις βιβλιοθήκης** οι οποίες περιλαμβάνουν τους λογαρίθμους, τις εκθετικές συναρτήσεις, τις τριγωνομετρικές, τι υπερβολικές συναρτήσεις, συναρτήσεις στατιστικής ανάλυσης και άλλες. Στον πιο κάτω πίνακα φαίνονται μόνο μερικές από αυτές.

Συνάρτηση	Ερμηνεία
<b>sin</b>	ημίτονο
<b>cos</b>	συνημίτονο
<b>tan</b>	εφαπτομένη
<b>arcsin</b>	τόξο ημιτόνου
<b>arccos</b>	τόξο συνημιτόνου
<b>arctan</b>	τόξο εφαπτομένης
<b>exp</b>	εκθετική συνάρτηση
<b>ln</b>	φυσικός λογάριθμος
<b>log</b>	λογάριθμος με βάση το 10
<b>abs</b>	απόλυτη τιμή
<b>sqrt</b>	τετραγωνική ρίζα

**Παράδειγμα**

> **sin(Pi/2);**

$$1$$



```
> arctan(1);
```

$$\frac{1}{4}\pi$$

```
> sin(a+b);
```

$$\sin(a+b)$$

```
> expand(%);
```

$$\sin(a)\cos(b) + \cos(a)\sin(b)$$

```
> ln(exp(x));
```

$$\ln\left(e^x\right)$$

```
> simplify(%);
```

$$\ln\left(e^x\right)$$

Εδώ βλέπουμε ότι η εντολή **simplify** δεν κατάφερε να απλοποιήσει την πιο πάνω παράσταση, αν και η απλοποίηση είναι προφανής. Ο λόγος είναι ότι η MAPLE δεν υποθέτει *τίποτα* για το  $x$ , άρα δεν θέλει να δώσει βεβαιασμένη απάντηση. Αν της πούμε όμως ότι το  $x$  είναι θετικό τότε

```
> simplify(%,assume = positive);
```

$$x$$

παίρνουμε την απάντηση που θέλουμε.

### 3. Συναρτήσεις, γραφικές παραστάσεις και εξισώσεις

Στο εδάφιο αυτό θα συζητήσουμε τρόπους με τους οποίους μπορούμε να ορίσουμε μια (μαθηματική) συνάρτηση, όπως επίσης και εντολές που μας δίνουν γραφικές παραστάσεις.

Για να ορίσουμε την συνάρτηση  $f(x) = x^2$ , γράφουμε

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

$$f := x \rightarrow x^2$$

Έτσι, αν γράψουμε

```
> f(x);
```

$$x^2$$

η MAPLE μας λέει ότι η  $f(x)$  ισούται με  $x^2$ . Παρατηρούμε ότι αν πούμε

```
> f;
```

$$f$$

τότε δεν παίρνουμε  $x^2$ , μια και η  $f$  είναι συνάρτηση του  $x$  και αυτό πρέπει να διευκρινιστεί.

Με αυτό τον ορισμό, μπορούμε να πούμε

```
> f(1); f(-19); f(33.7);
```

$$1$$

$$361$$

$$1135.69$$

ή ακόμα και

```
> f(t+2);
```

$$(t+2)^2$$

```
> f(sqrt(b));
```

$$b$$

Φυσικά, μπορούμε να ορίσουμε μια παράσταση που εξαρτάται από τη μεταβλητή  $x$ , χωρίς να χρησιμοποιήσουμε την πιο πάνω δομή, ως εξής:

```
> expr := 3*x^2 - 4*x - 5;
```

$$expr := 3x^2 - 4x - 5$$

Αν τώρα θέλουμε να δούμε την τιμή της **expr** όταν  $x = 3.5$  τότε θα πρέπει να χρησιμοποιήσουμε την εντολή **subs** που είδαμε πιο πριν:

```
> subs(x=3.5, expr);
```

$$17.75$$

Αν, όμως, είχαμε ορίσει την παράσταση σαν μια συνάρτηση του  $x$ ,

```
> g:=x->3*x^2 - 4*x - 5;
```

$$g := x \rightarrow 3x^2 - 4x - 5$$

τότε για να πάρουμε την τιμή που αντιστοιχεί στο  $x = 3.5$ , απλώς λέμε

```
> g(3.5);
```

17.75

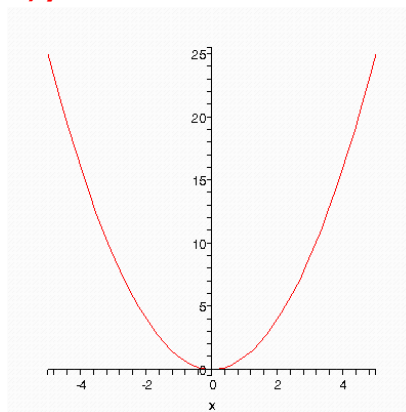
### Η εντολή plot

Για να πάρουμε την γραφική παράσταση μιας συνάρτησης  $f(x)$  στο διάστημα  $[a, b]$ , γράφουμε

```
plot(f(x), x = a..b)
```

Για παράδειγμα, με  $f(x)$  την συνάρτηση που ορίσαμε πιο πάνω, έχουμε

```
> plot(f(x), x=-5..5);
```



### Παράδειγμα

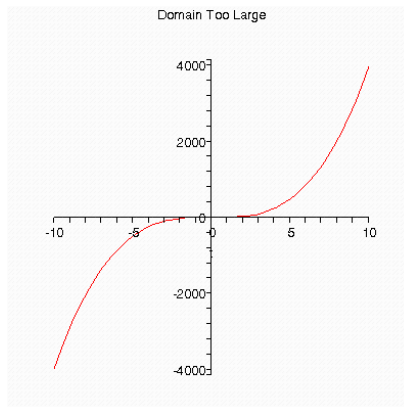
Έστω

```
> h:=t-> 4*t^3-3*t;
```

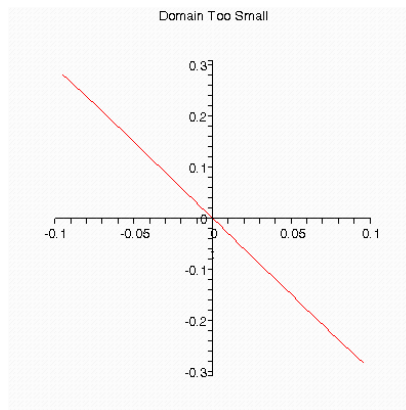
$$h := t \rightarrow 4t^3 - 3t$$

Θα κατασκευάσουμε διάφορες γραφικές παραστάσεις της συνάρτησης σε διαφορετικά διαστήματα, και κάθε φορά θα βάζουμε και τίτλο χρησιμοποιώντας επιπλέον προαιρετικά δεδομένα εισόδου στην εντολή **plot**.

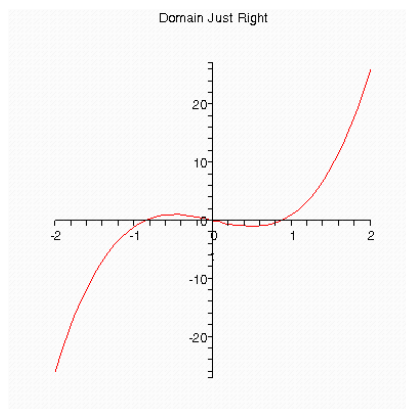
```
> plot(h(t), t=-10..10, title="Domain Too Large");
```



```
> plot(h(t), t=-0.1..0.1, title="Domain Too Small");
```

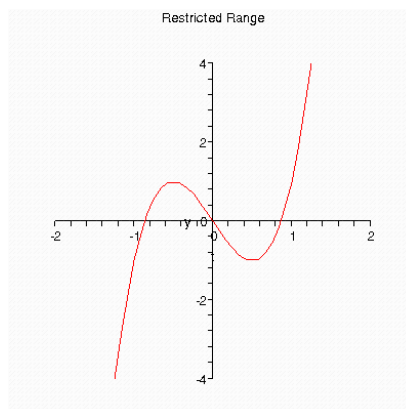


```
> plot(h(t),t=-2..2, title="Domain Just Right");
```



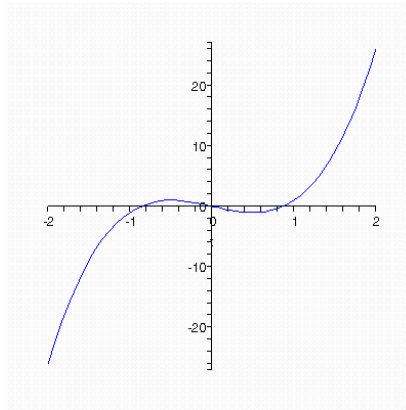
Μπορούμε ακόμη να περιορίσουμε το πεδίο τιμών της συνάρτησης (δηλ. τον άξονα των  $y$ ) ως εξής:

```
> plot(h(t),t=-2..2, y=-4..4,title="Restricted Range");
```



Για να πάρουμε την καμπύλη με, π.χ. μπλε χρώμα, λέμε

```
> plot(h(t),t=-2..2, color=blue);
```



Έστω ότι οι συναρτήσεις  $f(x)$  και  $g(x)$  είναι όπως και πριν:

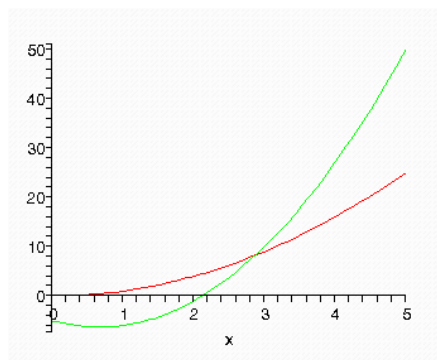
**> f(x); g(x);**

$$x^2$$

$$3x^2 - 4x - 5$$

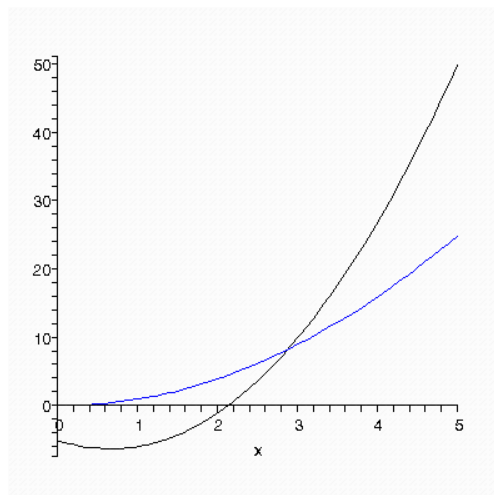
Για να πάρουμε τη γραφική παράσταση και των δύο στους ίδιους άξονες, γράφουμε

**> plot({f(x),g(x)},x=0..5);**



και για να καθορίσουμε το χρώμα της κάθε καμπύλης, λέμε

**> plot({f(x),g(x)},x=0..5,color=[black,blue]);**



Παρατηρούμε ότι η MAPLE αναθέτει τα χρώματα στην κάθε καμπύλη με ανάποδη σειρά από αυτή που έχουμε δώσει για τις συναρτήσεις.

### Συναρτήσεις πολλών μεταβλητών και η εντολή **plot3d**

Συναρτήσεις δύο, ή περισσότερων μεταβλητών μπορούν να οριστούν με τον ίδιο τρόπο:

```
> u := (x, y) -> cos(x) * cos(y);
```

$$u := (x, y) \rightarrow \cos(x) \cos(y)$$

Παίρνουμε τιμές όπως και πριν:

```
> u(Pi, y);
```

$$-\cos(y)$$

```
> u(x, 0);
```

$$\cos(x)$$

```
> u(a+b, a-b);
```

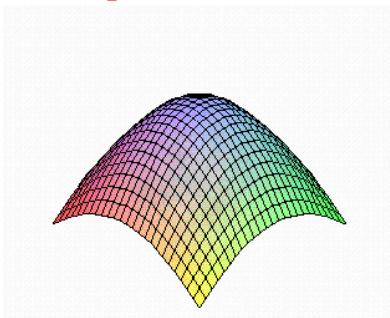
$$\cos(a+b) \cos(a-b)$$

```
> expand(%);
```

$$\cos(a)^2 \cos(b)^2 - \sin(a)^2 \sin(b)^2$$

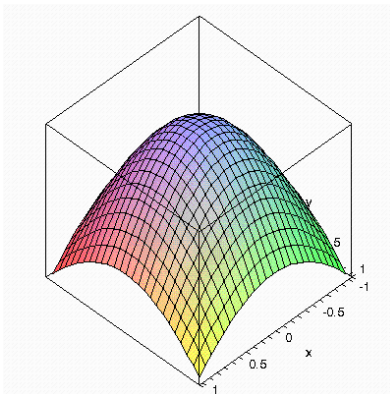
Για να κατασκευάσουμε τη γραφική παράσταση μιας συνάρτησης δύο μεταβλητών, χρησιμοποιούμε την εντολή **plot3d**, που δουλεύει με παρόμοιο τρόπο όπως και η **plot**, αλλά τώρα πρέπει να δώσουμε δύο πεδία ορισμού (ένα για την κάθε μεταβλητή):

```
> plot3d(u(x, y), x=-1..1, y=-1..1);
```



Για να βάλουμε άξονες σε μορφή κουτιού, στο γράφημα, λέμε

```
> plot3d( u(x, y), x = -1..1, y = -1..1, axes = boxed );
```



Υπάρχουν και άλλες επιλογές για τους άξονες – γράψτε **?plot3d** στη MAPLE για περαιτέρω πληροφορίες.

### Εξισώσεις

Μπορούμε να ορίσουμε μια εξίσωση στη MAPLE ως εξής:

```
> x^2-2*x+3=0;
```

$$x^2 - 2x + 3 = 0$$

Παρατηρούμε ότι εδώ δεν χρησιμοποιούμε το σύμβολο ‘:’ πριν από το ‘=’ γιατί δεν καταχωρούμε κάποια παράσταση σε μια μεταβλητή.

Αν θέλουμε να δώσουμε το όνομα **eqn1** στην εξίσωση (δηλ. να την καταχωρήσουμε στη μεταβλητή **eqn1**) τότε μπορούμε να πούμε

```
> eqn1:=%;
```

$$eqn1 := x^2 - 2x + 3 = 0$$

Για να λύσουμε την πιο πάνω εξίσωση χρησιμοποιούμε την εντολή **solve**, η οποία παίρνει σαν δεδομένα εισόδου μια εξίσωση και την μεταβλητή ως προς την οποία θέλουμε να λύσουμε την εξίσωση και δίνει σαν δεδομένα εξόδου (όταν μπορεί) τις ακριβείς λύσεις:

```
> solve(eqn1,x);
```

$$1 + I\sqrt{2}, 1 - I\sqrt{2}$$

Μπορούμε να επαληθεύσουμε τις λύσεις με αντικατάσταση:

```
> subs(x=1+sqrt(2)*I,eqn1);
```

$$(1 + I\sqrt{2})^2 + 1 - 2I\sqrt{2} = 0$$

```
> simplify(%);
```

$$0 = 0$$

```
> subs(x=1-sqrt(2)*I,eqn1);
```

$$(1 - I\sqrt{2})^2 + 1 + 2I\sqrt{2} = 0$$

```
> simplify(%);
```

$$0 = 0$$

**Παράδειγμα**

```
> eqn2:=x^3+2*x^2-x-2=0;
```

$$eqn2 := x^3 + 2x^2 - x - 2 = 0$$

Θα λύσουμε την πιο πάνω εξίσωση και θα καταχωρήσουμε τις λύσεις στη μεταβλητή  $S$ :

```
> S:= solve(eqn2, x);
```

$$S := -1, 1, -2$$

Για να δούμε μία-μία τις λύσεις, χρησιμοποιούμε τα σύμβολα '[' ]':

```
> S[1];
```

$$-1$$

```
> S[2];
```

$$1$$

```
> S[3];
```

$$-2$$

Με αυτό τον τρόπο, επαληθεύουμε, για παράδειγμα, την τρίτη λύση ως εξής:

```
> subs(x=S[3],eqn2);
```

$$0 = 0$$

Μπορούμε ακόμη, να επαληθεύσουμε όλες τις λύσεις μονομιάς:

```
> subs(x=S,eqn2);
```

$$(-1, 1, -2)^3 + 2(-1, 1, -2)^2 + (1, -1, 2) - 2 = 0$$

```
> simplify(%);
```

$$0 = 0$$

**Παράδειγμα**

Έστω η εξίσωση

```
> eqn3:=x=cos(4*x);
```

$$eqn3 := x = \cos(4x)$$

Αν προσπαθήσουμε να λύσουμε την εξίσωση αυτή με την εντολή **solve** τότε παίρνουμε το εξής:

```
> solve(eqn3,x);
```

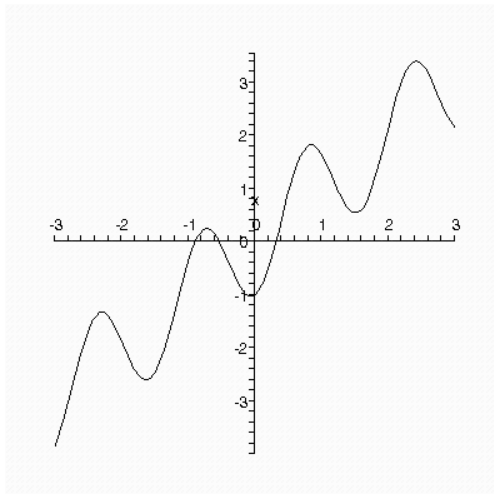
$$\cos(\text{RootOf}(Z - 4 \cos(Z)))$$

Αυτό σημαίνει ότι η MAPLE δεν κατόρθωσε να βρει την αναλυτική μορφή της λύσης της εξίσωσης. Αυτό είναι αναμενόμενο αφού η εξίσωση είναι μη αλγεβρική. Μπορούμε, όμως, να βρούμε την λύση (αν υπάρχει) προσεγγιστικά με την εντολή **fsolve** που παίρνει σαν δεδομένα εισόδου μια εξίσωση και την μεταβλητή ως προς την οποία θέλουμε να λύσουμε την εξίσωση την οποία γράφουμε ίση με μια αρχική εκτίμηση της λύσης.

Για την πιο πάνω εξίσωση βλέπουμε από την γραφική παράσταση της  $x - \cos(4x)$  ότι υπάρχουν 3 λύσεις: δύο μεταξύ του  $-1$  και  $0$  και μία μεταξύ του  $0$  και  $1$ .



```
> plot(x-cos(4*x),x=-3..3,color=black);
```



Για να τις βρούμε προσεγγιστικά, γράφουμε:

```
> fsolve(eq3, x = -1);
-0.8988262168
> fsolve(eq3, x = -0.5);
-0.5333330629
> fsolve(eq3, x = 1);
0.3130883085
```

### Παράδειγμα

Η MAPLE έχει τη δυνατότητα να λύνει και συστήματα εξισώσεων με τις πιο πάνω εντολές. Ας δούμε πως μπορούμε να λύσουμε ένα γραμμικό σύστημα με δύο εξισώσεις. Ορίζουμε πρώτα τις δύο εξισώσεις:

```
> linear1:=5*x-7*y=37; linear2:= 3*x-11*y=17;
linear1 := 5 x - 7 y = 37
linear2 := 3 x - 11 y = 17
```

Έπειτα, καταχωρούμε τη λύση του συστήματος στη μεταβλητή **soln**, χρησιμοποιώντας την εντολή **solve**, αφού στα δεδομένα εισόδου βάλουμε τις δύο εξισώσεις και τους δύο αγνώστους σε αγκύλες ( { } ):

```
> soln:=solve({linear1,linear2},{x,y});
soln := { y = 13/17, x = 144/17 }
```

Επαληθεύουμε τις λύσεις ως εξής:

```
> subs(soln,linear1); subs(soln,linear2);
37 = 37
17 = 17
```

### Παράδειγμα

Με την εντολή **fsolve** μπορούμε να λύσουμε και συστήματα μη-γραμμικών εξισώσεων, όπως δείχνουμε στο παρών παράδειγμα:

```
> neq1:=x^2+y^2=1;
                                neq1 := x2 + y2 = 1
> neq2:=x^3+y^3-2*x=0;
                                neq2 := x3 + y3 - 2 x = 0
> fsolve({neq1,neq2},{x,y});
                                {y = -.9107598701, x = -.4129363862}
```

Όταν δεν δίνουμε αρχικές εκτιμήσεις στα δεδομένα εισόδου (όπως κάναμε εδώ), τότε η MAPLE διαλέγει αρχικές εκτιμήσεις από μόνη της και βρίσκει *μια* από τις λύσεις (αν αυτή υπάρχει).

## 4. Όρια, παράγωγοι, αθροίσματα και ολοκληρώματα

Η MAPLE υπολογίζει το όριο μιας παράστασης (ή συνάρτησης) όταν μια μεταβλητή τείνει προς μια τιμή με την εντολή **limit**, η οποία δουλεύει με τον τρόπο που φαίνεται από τις εξής αντιστοιχίες:

$\lim_{x \rightarrow a} f(x)$	<code>limit(f(x), x=a);</code>
$\lim_{x \rightarrow \infty} f(x)$	<code>limit(f(x), x=infinity);</code>
$\lim_{n \rightarrow N} G(n)$	<code>limit(G(n), n=N);</code>

Για παράδειγμα,

```
> limit((x^2-1)/(x+1), x=-1);
-2
> limit((1+1/n)^n, n=infinity);
e
```

### Παράγωγοι

Για να βρούμε την παράγωγο μιας παράστασης ως προς μια μεταβλητή, χρησιμοποιούμε την εντολή **diff**, η οποία παίρνει σαν δεδομένα εισόδου την παράσταση και την μεταβλητή ως προς την οποία παραγωγίζουμε.

Για παράδειγμα

```
> y:=x^5;
y:=x^5
> diff(y,x);
5x^4
```

Μπορούσαμε να βάλουμε την παράσταση κατευθείαν στην εντολή, ως εξής:

```
> diff(x^5,x);
5x^4
```

Παρατηρήστε (και εξηγήστε) τη διαφορά μεταξύ των πιο κάτω δύο εφαρμογών της εντολής **diff**:

```
> diff(c*x^5,x);
5cx^4
> diff(c*x^5,c);
x^5
```

Για παραγώγους πιο υψηλής τάξης χρησιμοποιούμε το σύμβολο **\$** μετά από τη μεταβλητή. Για παράδειγμα, αν θέλουμε να βρούμε την 3<sup>η</sup> παράγωγο του  $cx^5$ , γράφουμε

```
> diff(c*x^5,x$3);
```

$$60 c x^2$$

Μπορούμε, επίσης, να δώσουμε μια συνάρτηση σαν δεδομένο εισόδου:

```
> f:=x->cos(x)^3;
```

$$f := x \rightarrow \cos(x)^3$$

```
> diff(f(x),x);
```

$$-3 \cos(x)^2 \sin(x)$$

```
> diff(f(x),x$5);
```

$$60 \sin(x)^3 - 183 \cos(x)^2 \sin(x)$$

```
> diff(f,x);
```

$$0$$

Η τελευταία εντολή μας έδωσε 0 γιατί δεν διευκρινίσαμε ότι η  $f$  είναι συνάρτηση του  $x$  και η MAPLE την θεώρησε σαν σταθερά.

Όταν θέλουμε να υπολογίσουμε την παράγωγο μιας συνάρτησης σαν μια καινούργια συνάρτηση, τότε αντί για την εντολή **diff**, χρησιμοποιούμε την εντολή **D**. (Το **D** για την ακρίβεια είναι ο τελεστής παραγώγισης.)

Για παράδειγμα,

```
> f:=x->x^5;
```

$$f := x \rightarrow x^5$$

```
> D(f);
```

$$x \rightarrow 5 x^4$$

Αν, τώρα, καταχωρήσουμε την απάντηση με το όνομα **fprime**, μπορούμε να βρούμε τιμές της παραγώγου εύκολα:

```
> fprime:=D(f);
```

$$fprime := x \rightarrow 5 x^4$$

```
> fprime(2); fprime(3.5); fprime(-7.14);
```

$$80$$

$$750.3125$$

$$12994.59808080$$

Αυτό μπορεί και να γίνει κατευθείαν με την εντολή **D**:

```
> D(f)(2);
```

$$80$$

```
> D(f)(3.5);
```

$$750.3125$$

```
> D(f)(x);
```

$$5 x^4$$

**Παράδειγμα**

Παρατηρήστε τι μας δίνει η MAPLE όταν ζητήσουμε το πιο κάτω όριο για μια συνάρτηση  $p(x)$  που δεν έχει οριστεί:

```
> limit((p(x+h)-p(x))/h,h=0);
```

$$D(p)(x)$$
**Παράδειγμα**

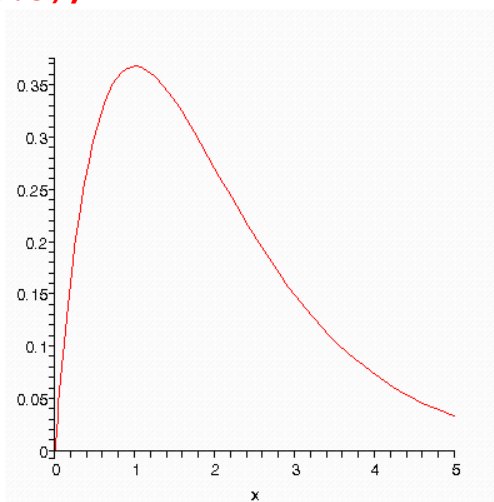
Έστω η συνάρτηση

```
> f:=x->x*exp(-x);
```

$$f:=x \rightarrow x e^{(-x)}$$

της οποίας η γραφική παράσταση έχει ως εξής:

```
> plot(f(x),x=0..5);
```



Θα βρούμε τις συντεταγμένες του μέγιστου σημείου ( $x_{max}$ ,  $y_{max}$ ) και του σημείου καμπής ( $x_{infl}$ ,  $y_{infl}$ ):

```
> xmax:=solve(diff(f(x),x)=0,x);
```

$$xmax := 1$$

```
> ymax:=f(xmax);
```

$$ymax := e^{(-1)}$$

```
> xinfl:=solve(diff(f(x),x$2)=0,x);
```

$$xinfl := 2$$

```
> yinlf:=f(xinfl);
```

$$yinlf := 2 e^{(-2)}$$

Άρα, η συνάρτηση έχει μέγιστο σημείο το  $(1, 1/e)$  σημείο καμπής το  $(2, 2/e^2)$ .

## Αθροίσματα

Τώρα, ας δούμε πως μπορούμε να υπολογίσουμε αθροίσματα με τη MAPLE. Η εντολή

```
sum(S(i), i = 0..N)
```

αντιστοιχεί στο  $\sum_{i=0}^N S(i)$ . (Το  $N$  μπορεί να είναι άπειρο.) Για παράδειγμα,

```
> sum(i^a, i=0..10);
```

$$1 + 2^a + 3^a + 4^a + 5^a + 6^a + 7^a + 8^a + 9^a + 10^a$$

```
> sum(k^2, k=0..5);
```

55

```
> sum(1/k, k=1..infinity);
```

$\infty$

```
> sum(1/n!, n=0..infinity);
```

e

```
> sum(1/i^2, i=1..infinity);
```

$\frac{1}{6}\pi^2$

## Παράδειγμα

Ορίζουμε στη MAPLE το εξής άθροισμα  $S_N = \sum_{i=1}^N F\left(\frac{i}{N}\right) \frac{1}{N}$  σαν συνάρτηση του  $N$ , όπου η  $F(x)$  είναι κάποια συνάρτηση:

```
> S:=N->sum(F(i/N)*(1/N), i=1..N);
```

$$S := N \rightarrow \sum_{i=1}^N \left( \frac{F\left(\frac{i}{N}\right)}{N} \right)$$

Έχουμε

```
> S(1);
```

$F(1)$

```
> S(2);
```

$$\frac{1}{2} F\left(\frac{1}{2}\right) + \frac{1}{2} F(1)$$

```
> S(3);
```

$$\frac{1}{3} F\left(\frac{1}{3}\right) + \frac{1}{3} F\left(\frac{2}{3}\right) + \frac{1}{3} F(1)$$

Έστω τώρα ότι

```
> F:=x->x^3;
```

$F := x \rightarrow x^3$

Τότε

```

> s(1);
1
> s(2);
9
16
> s(3);
4
9
> limit(s(N),N=infinity);
1
4

```

Τι νομίζετε ότι αντιπροσωπεύει η τιμή που πήραμε;

### Ολοκληρώματα

Για να υπολογίσουμε το αόριστο ολοκλήρωμα  $\int f(x)dx$  γράφουμε

```
int(f(x),x);
```

και για το ορισμένο ολοκλήρωμα  $\int_a^b f(x)dx$  λέμε

```
int(f(x),x=a..b);
```

Για παράδειγμα,

```

> f:=t->t^2;
f:=t→t2
> int(f(t),t);
1
3 t3
> diff(%,t);
t2
> int(f(t),t=0..2);
8
3
> int(t^2,t=0..2);
8
3

```

Η MAPLE μπορεί να υπολογίσει και ολοκληρώματα των οποίων οι απαντήσεις περιέχουν γνωστές (σε ορισμένους) μαθηματικές συναρτήσεις, όπως για παράδειγμα

```

> int(sin(t^2),t=0..2);
1
2 FresnelS(2√2/√π)√2√π

```

Για να δούμε την πιο πάνω τιμή σε δεκαδική μορφή λέμε

```
> evalf(%);
```

```
0.80477648934375611025
```

Μπορούσαμε να λέγαμε απευθείας

```
> evalf(int(sin(t^2),t=0..2));
```

```
0.80477648934375611025
```

αλλά πρέπει να έχουμε υπόψη μας ότι αυτός ο συνδυασμός εντολών θα δώσει μια προσέγγιση στο ολοκλήρωμα όταν αυτό δεν μπορεί να υπολογιστεί αναλυτικά. Φυσικά, ορισμένες φορές αυτό είναι επιθυμητό, αλλά όταν δεν είναι πρέπει να είμαστε προσεχτικοί. Δείτε, για παράδειγμα, την (μικρή αλλά υπαρκτή) διαφορά στα πιο κάτω:

```
> evalf(int(sin(t^2),t=0..2));
```

```
0.80477648934375611025
```

```
> int(sin(t^2),t=0.0..2.0);
```

```
0.80477648934375611030
```

Όταν η MAPLE δεν μπορεί να υπολογίσει κάποιο ολοκλήρωμα, τότε μας επιστρέφει σαν απάντηση αυτό που ζητήσαμε. Για παράδειγμα,

```
> int( sin(t)^(1/3), t=0..2);
```

$$\int_0^2 \sin(t)^{(1/3)} dt$$

Φυσικά, σε αυτή την περίπτωση, μια και έχουμε ορισμένο ολοκλήρωμα, θα μπορούσαμε να χρησιμοποιήσουμε την εντολή **evalf** που θα μας δώσει μια προσέγγιση για την τιμή του.

### Σημείωση:

Μπορούμε να υπολογίσουμε αόριστα και ορισμένα πολλαπλά ολοκληρώματα με την εντολή **int**, ως εξής:

$\iint f(x,y)dx dy$	<code>int(int(f(x,y),x),y);</code>
$\int_c^d \int_a^b f(x,y)dx dy$	<code>int(int(f(x,y),x=a..b),y=c..d);</code>

Για παράδειγμα,

```
> int(int(x*y,x),y);
```

$$\frac{1}{4}x^2 y^2$$

```
> int(int(x*exp(y),x=0..2),y=0..3);
```

$$2e^3 - 2$$



## 5. Πίνακες και διανύσματα

Αν και η MAPLE δεν είναι το πιο ‘κατάλληλο’ λογισμικό για επεξεργασία πινάκων και διανυσμάτων, οι συμβολικές της ικανότητες της δίνουν ορισμένα πλεονεκτήματα που δεν μπορούν να περιφρονηθούν.

Για να μπορέσουμε να χρησιμοποιήσουμε τις εντολές για πίνακες και διανύσματα, πρέπει πρώτα να φορτώσουμε το πακέτο **linalg**, ως εξής:

```
> with(linalg);
```

```
[BlockDiagonal, GramSchmidt, JordanBlock, LUdecomp, QRdecomp, Wronskian, addcol, addrow, adj, adjoint, angle, augment, backsub, band, basis, bezout,
blockmatrix, charmat, charpoly, cholesky, col, coldim, colspace, colspan, companion, concat, cond, copyinto, crossprod, curl, definite, delcols, delrows,
det, diag, diverge, dotprod, eigenvals, eigenvalues, eigenvectors, eigenvects, entermatrix, equal, exponential, extend, ffgausselim, fibonacci, forwardsub,
frobenius, gausselim, gaussjord, geneqns, genmatrix, grad, hadamard, hermite, hessian, hilbert, htranspose, ihermite, indexfunc, innerprod, intbasis,
inverse, ismith, issimilar, iszero, jacobian, jordan, kernel, laplacian, leastsqrs, linsolve, matadd, matrix, minor, minpoly, mulcol, mulrow, multiply, norm,
normalize, nullspace, orthog, permanent, pivot, potential, randmatrix, randvector, rank, ratform, row, rowdim, rowspace, rowspan, rref, scalarmul,
singularvals, smith, stackmatrix, submatrix, subvector, subbasis, swapcol, swaprow, sylvester, toeplitz, trace, transpose, vandermonde, vecpotent, vectdim,
vector, wronskian]
```

Η λίστα που παίρνουμε περιέχει όλες τις εντολές που μας είναι διαθέσιμες με αυτό το πακέτο. Αν δεν θέλουμε να τις δούμε, βάζουμε ‘:’ αντί για ‘;’ στο τέλος. Αφού φορτώσουμε το πακέτο, μπορούμε, για παράδειγμα, να ορίσουμε τον πίνακα

$$A = \begin{bmatrix} 2 & 1 \\ -1 & 5 \end{bmatrix}$$

με δύο ισοδύναμους τρόπους:

```
> A:=matrix([[2,1],[-1,5]]);
```

$$A := \begin{bmatrix} 2 & 1 \\ -1 & 5 \end{bmatrix}$$

```
> A:=matrix(2,2,[2,1,-1,5]);
```

$$A := \begin{bmatrix} 2 & 1 \\ -1 & 5 \end{bmatrix}$$

Αν θέλουμε να δούμε τον πίνακα  $A$ , λέμε

```
> evalm(A);
```

$$\begin{bmatrix} 2 & 1 \\ -1 & 5 \end{bmatrix}$$

(Το προέρχεται από το *evaluate matrix*.) Αν πούμε απλώς  $A$  θα πάρουμε

```
> A;
```

$A$

Για να δούμε το στοιχείο του πίνακα  $A$  που βρίσκεται, π.χ., στη θέση  $(2, 1)$  λέμε

```
> A[2,1];
```

```
-1
```

Αν πούμε

```
> A[3,1];
```

```
Error, 1st index, 3, larger than upper array bound 2
```

παίρνουμε λάθος μια και ο πίνακας είναι 2×2.

Ορίζουμε επίσης

```
> B:=matrix([[a,-b],[c,1/d]]);
```

$$B := \begin{bmatrix} a & -b \\ c & \frac{1}{d} \end{bmatrix}$$

```
> C:=matrix([[7,-2,1],[9,3,8]]);
```

$$C := \begin{bmatrix} 7 & -2 & 1 \\ 9 & 3 & 8 \end{bmatrix}$$

Για να προσθέσουμε πίνακες πρέπει να πούμε

```
> evalm(A+B);
```

$$\begin{bmatrix} 2+a & 1-b \\ -1+c & 5+\frac{1}{d} \end{bmatrix}$$

Αν πούμε απλώς  $A + B$  θα πάρουμε

```
> A+B;
```

```
A + B
```

Το ίδιο ισχύει και για τις υπόλοιπες πράξεις.

```
> evalm(2*A);
```

$$\begin{bmatrix} 4 & 2 \\ -2 & 10 \end{bmatrix}$$

```
> evalm(C+B);
```

```
Error, (in ) matrix dimensions incompatible
```

Για να πολλαπλασιάσουμε πίνακες λέμε

```
> multiply(A,C);
```

$$\begin{bmatrix} 23 & -1 & 10 \\ 38 & 17 & 39 \end{bmatrix}$$

```
> multiply(C,A);
```

```
Error, (in multiply) non matching dimensions for vector/matrix product
```

Μπορούμε, ισοδύναμα, να πούμε

**> evalm( A &\* B );**

$$\begin{bmatrix} 2a+c & -2b+\frac{1}{d} \\ -a+5c & b+\frac{5}{d} \end{bmatrix}$$

όπου τα σύμβολα **&\*** υποδηλώνουν τον πολλαπλασιασμό.

Ο ανάστροφος ενός πίνακα δίνεται από την εντολή **transpose**:

**> transpose(B);**

$$\begin{bmatrix} a & c \\ -b & \frac{1}{d} \end{bmatrix}$$

Η ορίζουσα ενός πίνακα δίνεται από την εντολή **det**:

**> det(A);**

11

**> det(B);**

$$\frac{a+bcd}{d}$$

Ο αντίστροφος ενός πίνακα υπολογίζεται με διάφορους τρόπους:

**> inverse(A);**

$$\begin{bmatrix} \frac{5}{11} & \frac{-1}{11} \\ \frac{1}{11} & \frac{2}{11} \end{bmatrix}$$

**> evalm(1/A);**

$$\begin{bmatrix} \frac{5}{11} & \frac{-1}{11} \\ \frac{1}{11} & \frac{2}{11} \end{bmatrix}$$

**> evalm(A^(-1));**

$$\begin{bmatrix} \frac{5}{11} & \frac{-1}{11} \\ \frac{1}{11} & \frac{2}{11} \end{bmatrix}$$

Επιβεβαιώνουμε ότι  $A^{-1}A = AA^{-1} = I$ :

**> evalm(inverse(A) &\* A);**

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

```
> evalm(A &* inverse(A));
```

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

ή μέσω της εντολής

```
> multiply(A,1/A);
```

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

### Διανύσματα

Τα διανύσματα ορίζονται με παρόμοιο τρόπο:

```
> v:=vector(3,[1,cos(t),sin(t)]);
```

$$v := [1, \cos(t), \sin(t)]$$

```
> evalm(v);
```

$$[1, \cos(t), \sin(t)]$$

```
> u:=vector(3,[-1,cos(t),sin(t)]);
```

$$u := [-1, \cos(t), \sin(t)]$$

Το εσωτερικό γινόμενο δύο διανυσμάτων δίνεται από την εντολή **innerprod**:

```
> innerprod(u,v);
```

$$-1 + \cos(t)^2 + \sin(t)^2$$

```
> simplify(%);
```

$$0$$

Η γωνία μεταξύ δύο διανυσμάτων δίνεται από την εντολή **angle**:

```
> angle(u,v);
```

$$\arccos\left(\frac{-1 + \cos(t)^2 + \sin(t)^2}{1 + \cos(t)^2 + \sin(t)^2}\right)$$

```
> simplify(%);
```

$$\frac{1}{2}\pi$$

Η 2-νόρμα ενός διανύσματος δίνεται από την εντολή **norm**:

```
> norm(u,2);
```

$$\sqrt{1 + |\cos(t)|^2 + |\sin(t)|^2}$$

```
> simplify(% , assume=positive);
```

$$\sqrt{2}$$

Το εξωτερικό δύο διανυσμάτων του  $\mathbb{R}^3$  δίνεται από την εντολή **crossprod**:

```
> crossprod(u,v);
```

$$[0, 2 \sin(t), -2 \cos(t)]$$

**Γραμμικά συστήματα**

Έστω

```
> A:=matrix(3,3,[2,2,-1,3,4,2,1,4,3]);
```

$$A := \begin{bmatrix} 2 & 2 & -1 \\ 3 & 4 & 2 \\ 1 & 4 & 3 \end{bmatrix}$$

```
> b:=vector(3,[1,-2,3]);
```

$$b := [1, -2, 3]$$

Η λύση του συστήματος  $Ax = b$ , μπορεί να βρεθεί με διάφορους τρόπους. Ο πιο απλός είναι μέσω της εντολής **linsolve**:

```
> x:=linsolve(A,b);
```

$$x := \left[ \frac{-24}{7}, 3, \frac{-13}{7} \right]$$

Επαληθεύουμε:

```
> multiply(A,x);
```

$$[1, -2, 3]$$

Μπορούμε να τυπώσουμε το  $x$  ή κάποιο από τα στοιχεία του στην οθόνη, με τον ίδιο τρόπο όπως και πριν:

```
> evalm(x);
```

$$\left[ \frac{-24}{7}, 3, \frac{-13}{7} \right]$$

```
> x[3];
```

$$\frac{-13}{7}$$

Ένας άλλος τρόπος εξεύρεσης της λύσης του συστήματος  $Ax = b$  είναι ο εξής:

```
> x:=multiply(inverse(A),b);
```

$$x := \left[ \frac{-24}{7}, 3, \frac{-13}{7} \right]$$

αλλά αυτό δεν συστήνεται στην πράξη.

Μπορούμε, ακόμη, να κατασκευάσουμε τον επαυξημένο πίνακα  $[A | b]$  με την εντολή **augment**:

```
> A_b:=augment(A,b);
```

$$A\_b := \begin{bmatrix} 2 & 2 & -1 & 1 \\ 3 & 4 & 2 & -2 \\ 1 & 4 & 3 & 3 \end{bmatrix}$$

και να κάνουμε πρώτα απαλοιφή Gauss με την εντολή **gausselim**:

```
> A_b_elim:=gausselim(A_b);
```

$$A\_b\_elim := \begin{bmatrix} 2 & 2 & -1 & 1 \\ 0 & 1 & \frac{7}{2} & \frac{-7}{2} \\ 0 & 0 & -7 & 13 \end{bmatrix}$$

και μετά ανάδρομη αντικατάσταση με την εντολή **backsub**:

> **x:=backsub(A\_b\_elim);**

$$x := \left[ \frac{-24}{7}, 3, \frac{-13}{7} \right]$$

Ένας άλλος τρόπος είναι μέσω της ανηγμένης κλιμακωτής μορφής του επαυξημένου πίνακα (δηλ. απαλοιφή Gauss-Jordan) και η εντολή για αυτή τη μέθοδο είναι **rref**:

> **rref(A\_b);**

$$\begin{bmatrix} 1 & 0 & 0 & \frac{-24}{7} \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & \frac{-13}{7} \end{bmatrix}$$

### Ιδιοτιμές και ιδιοδιανύσματα

Έστω

> **A:=matrix(3,3,[2,-4,5,0,2,-7/5,1,1,-1]);**

$$A := \begin{bmatrix} 2 & -4 & 5 \\ 0 & 2 & \frac{-7}{5} \\ 1 & 1 & -1 \end{bmatrix}$$

Για να βρούμε τις ιδιοτιμές και τα αντίστοιχα ιδιοδιανύσματα του πίνακα  $A$  (δηλ. τους αριθμούς  $\lambda$  και τα μη-μηδενικά διανύσματα  $x$  έτσι ώστε  $Ax = \lambda x$ ), προχωρούμε ως εξής: Βρίσκουμε τις ιδιοτιμές με την εντολή **eigenvals**:

> **lambda:=eigenvals(A);**

$$\lambda := 1, 1 + \frac{1}{5}\sqrt{165}, 1 - \frac{1}{5}\sqrt{165}$$

Για να τυπώσουμε στην οθόνη την, π.χ., δεύτερη ιδιοτιμή λέμε

> **lambda[2];**

$$1 + \frac{1}{5}\sqrt{165}$$

Τα ιδιοδιανύσματα δίδονται από την εντολή **eigenvects**:

> **x:=eigenvects(A);**

$$x := \left[ 1, 1, \left\{ \left[ 1, \frac{7}{3}, \frac{5}{3} \right] \right\} \right], \left[ 1 + \frac{1}{5}\sqrt{165}, 1, \left\{ \left[ -\frac{30}{7} - \frac{1}{7}\sqrt{165}, 1, \frac{5}{7} - \frac{1}{7}\sqrt{165} \right] \right\} \right], \left[ 1 - \frac{1}{5}\sqrt{165}, 1, \left\{ \left[ -\frac{30}{7} + \frac{1}{7}\sqrt{165}, 1, \frac{5}{7} + \frac{1}{7}\sqrt{165} \right] \right\} \right]$$

Η πιο πάνω λίστα περιέχει διάφορες πληροφορίες. Για παράδειγμα,

**> x[1][1];**

1

είναι η πρώτη ιδιοτιμή,

**> x[1][2];**

1

είναι η αλγεβρική της πολλαπλότητα, και

**> x[1][3];**

$$\left\{ \left[ 1, \frac{7}{3}, \frac{5}{3} \right] \right\}$$

το αντίστοιχο ιδιοδιάνυσμα.

Παρομοίως,

**> x[2][1];**

$$1 + \frac{1}{5}\sqrt{165}$$

είναι η δεύτερη ιδιοτιμή,

**> x[2][2];**

1

είναι η αλγεβρική της πολλαπλότητα, και

**> x[2][3];**

$$\left\{ \left[ -\frac{30}{7} - \frac{1}{7}\sqrt{165}, 1, \frac{5}{7} - \frac{1}{7}\sqrt{165} \right] \right\}$$

το αντίστοιχο ιδιοδιάνυσμα. Τι πρέπει να γράψουμε για να πάρουμε τις πληροφορίες για την τρίτη ιδιοτιμή και το αντίστοιχο ιδιοδιάνυσμα;

Τέλος, ας ελέγξουμε ότι πράγματι  $Ax_1 = \lambda_1 x_1$  :

**> x1:=vector(3,[1,7/3,5/3]);**

$$x1 := \left[ 1, \frac{7}{3}, \frac{5}{3} \right]$$

**> lambda[1];**

1

**> evalm(A &\* x1 = lambda[1]\*x1);**

$$\left[ 1, \frac{7}{3}, \frac{5}{3} \right] = \left[ 1, \frac{7}{3}, \frac{5}{3} \right]$$

## Επίλογος

Υπάρχουν πάμπολλες εντολές και πληροφορίες τις οποίες δεν καλύψαμε σε αυτή την εισαγωγή. Ελπίζουμε ότι δώσαμε αρκετά βασικά στοιχεία που να επιτρέψουν στον αναγνώστη να πειραματιστεί και να μάθει πως να χρησιμοποιεί πιο πολύπλοκες (και ίσως πιο χρήσιμες) εντολές.

Κλείνοντας θα θέλαμε να δείξουμε μέσω ενός απλού παραδείγματος, ορισμένα στοιχεία προγραμματισμού. Συγκεκριμένα, χρησιμοποιώντας ένα βρόχο **for** θα λύσουμε την εξίσωση  $x - \cos(4x) = 0$  με τη μέθοδο του Newton, αρχίζοντας από την αρχική τιμή  $x_0 = 0$ .

Πρώτα ορίζουμε την συνάρτηση  $f(x) = x - \cos(4x)$  και ένα (αρχικά άδειο) διάνυσμα  $x$ , του οποίου η πρώτη συνιστώσα είναι 0:

```
> f:=x->x-cos(4*x):
> x:=vector(10):
> x[1]:=0;
```

$$x_1 := 0$$

Στη συνέχεια κάνουμε ένα βρόχο από το 2 μέχρι το 10 και υπολογίζουμε τις υπόλοιπες συνιστώσες του  $x$  (δηλ. τις καλύτερες προσεγγίσεις της λύσης), μέσω του τύπου

$$x_n = x_{n-1} - \frac{f(x_{n-1})}{f'(x_{n-1})},$$

χρησιμοποιώντας καθοδόν εντολές που είδαμε στα προηγούμενα εδάφια:

```
> for n from 2 to 10 do
>   x[n]:=evalf( x[n-1] - f(x[n-1])/D(f)(x[n-1]), 20);
> end;
```

$$x_2 := 1.$$

$$x_3 := 1.8157238945019766452$$

$$x_4 := 1.5245041235005818442$$

$$x_5 := -.5305087371488888418$$

$$x_6 := -.53331909544837266594$$

$$x_7 := -.53333306256566166202$$

$$x_8 := -.53333306291483343518$$

$$x_9 := -.53333306291483343541$$

$$x_{10} := -.53333306291483343542$$

Το διάνυσμα με την ακολουθία προσεγγίσεων είναι:

```
> evalm(x);
```

```
[0, 1., 1.815723895, 1.524504124, -.5305087371, -.5333190954, -.5333330626, -.5333330629, -.5333330629, -.5333330629]
```