ΕΦΑΡΜΟΣΜΕΝΑ ΜΑΘΗΜΑΤΙΚΑ ΙΙ

για ΠΟΛΙΤΙΚΟΥΣ ΜΗΧΑΝΙΚΟΥΣ

Τεύχος 3

ΣΥΝΤΟΜΟ ΒΙΟΓΡΑΦΙΚΟ ΤΟΥ ΣΥΓΓΡΑΦΕΑ

Ο Νικόλαος Ι. Ιωακειμίδης γεννήθηκε στην Καλλιθέα Αττικής το 1950. Τελείωσε το Γυμνάσιο (1965) και το Λύκειο (1968) στον Πειραιά. Είναι διπλωματούχος Μηχανολόγος–Ηλεκτρολόγος Μηχανικός (1973) του Εθνικού Μετσοβίου Πολυτεχνείου (Ε.Μ.Π.) και Διδάκτωρ Μηχανικός (1976) πάλι του Ε.Μ.Π. Η διπλωματική εργασία του (1973) και η διδακτορική διατριβή του (1976) αναφέρονται στην Ελαστικότητα και στη Θραυστομηχανική.

Από το 1970 σαν σπουδαστής ήταν ανεπίσημος συνεργάτης του Εργαστηρίου Αντοχής Υλικών του Ε.Μ.Π. Το 1976 υπήρξε ερευνητής στο ίδιο εργαστήριο. Από το 1976 μέχρι το 1980 ήταν Επιμελητής στις Έδρες Μηχανικής Α και Β του Ε.Μ.Π. Από το 1980 μέχρι το 1982 ήταν Έκτακτος Καθηγητής και από το 1982 μέχρι σήμερα είναι Καθηγητής της Πολυτεχνικής Σχολής του Πανεπιστημίου Πατρών. Από το 1982 σαν Καθηγητής εργάζεται στο Γενικό Τμήμα στον Τομέα Εφαρμοσμένων Μαθηματικών και Μηχανικής.

Στο Ε.Μ.Π. συμμετέσχε στα εργαστήρια Πειραματικής Αντοχής Υλικών και έκανε φροντιστήρια μαθημάτων Μηχανικής. Επίσης δίδαξε ένα εξάμηνο το μάθημα Αντοχή Υλικών. Στην Πολυτεχνική Σχολή του Πανεπιστημίου Πατρών δίδαξε μαθήματα Μαθηματικών, Εφαρμοσμένων Μαθηματικών και Πιθανοθεωρίας–Στατιστικής σε διάφορα τμήματα. Κατά τα τελευταία έτη διδάσκει αποκλειστικά στο Τμήμα Πολιτικών Μηχανικών τα μαθήματα Εφαρμοσμένα Μαθηματικά ΙΙ και ΙΙΙ και εκτελεί τα σχετικά εργαστήρια στο Υπολογιστικό Κέντρο του Τμήματος. Διδάσκει επίσης το μεταπτυχιακό μάθημα Εφαρμοσμένα Μαθηματικά πάλι στο Τμήμα Πολιτικών Μηχανικών.

Το ερευνητικό έργο του αναφέρεται στη Μηχανική, τη Θραυστομηχανική, τα Εφαρμοσμένα Μαθηματικά, τη Μιγαδική Ανάλυση, τις Ολοκληρωτικές Εξισώσεις, την Αριθμητική Ανάλυση (Αριθμητική Ολοκλήρωση και επίλυση Ολοκληρωτικών Εξισώσεων) και τους Συμβολικούς Υπολογισμούς στη Μηχανική. Είναι συγγραφέας και συσυγγραφέας πολλών επιστημονικών εργασιών που έχουν δημοσιευθεί στην Αγγλική γλώσσα σε πολλά διεθνή περιοδικά των πιο πάνω επιστημονικών περιοχών (και μία εργασία στη Θεωρία Προσεγγίσεως) σε διάφορες χώρες. Σαν μεγαλύτερη ερευνητική συμβολή του θεωρεί την αναγωγή προβλημάτων ρωγμών σε ιδιόμορφες και υπεριδιόμορφες ολοκληρωτικές εξισώσεις και την επίλυσή τους με χρήση της μεθόδου της αριθμητικής ολοκληρώσεως.

Πέρα από τις επιστημονικές εργασίες του έχει κρίνει πολλές εργασίες άλλων συγγραφέων για διεθνή επιστημονικά περιοδικά της Μηχανικής, της Θραυστομηχανικής, των Εφαρμοσμένων Μαθηματικών και της Αριθμητικής Αναλύσεως. Υπήρξε κριτής για τα περιοδικά κριτικών *Applied Mechanics Reviews* και *Mathematical Reviews* και μέλος της Εκδοτικής Επιτροπής του περιοδικού *International Journal of Solids and Structures*.

Κατά τα τελευταία έτη καταβάλλει συστηματική προσπάθεια για την αναβάθμιση των μαθημάτων Εφαρμοσμένα Μαθηματικά ΙΙ και ΙΙΙ που διδάσκει στο Τμήμα Πολιτικών Μηχανικών της Πολυτεχνικής Σχολής του Πανεπιστημίου Πατρών: στην αίθουσα διδασκαλίας, στο εργαστήριο, μέσω συχνών εξετάσεων προόδου και εργαστηρίου και μέσω των διδακτικών αυτών βιβλίων Εφαρμοσμένων Μαθηματικών ΙΙ και ΙΙΙ που απευθύνονται αποκλειστικά σε Πολιτικούς Μηχανικούς και όχι γενικά σε Μηχανικούς. Στην προσπάθειά του αυτή έχει τύχει πολύτιμης βοήθειας και συμπαραστάσεως κάθε μορφής από πολλούς συναδέλφους του καθώς και της ενεργής συμμετοχής των φοιτητών και φοιτητριών του Πολιτικών Μηχανικών και είναι ευγνώμων σε όλους και όλες γι’ αυτά.

Τέλος στα Πανεπιστημιακά θέματα η θέση του ήταν και είναι υπέρ του Πανεπιστημίου στην κλασική του μορφή, όπως την έχει ζήσει και τη ζει και ο ίδιος επί πολλά χρόνια. Επομένως είναι εναντίον κάθε επιχειρηματικής ή οικονομικής διεισδύσεως τρίτων στον Πανεπιστημιακό χώρο. Είναι επίσης θερμός υποστηρικτής της ελεύθερης έρευνας στα Πανεπιστήμια με την έννοια ότι τα ερευνητικά αποτελέσματα πρέπει να είναι απόλυτα προσιτά σε κάθε μέρος του κόσμου χωρίς περιορισμούς μέσω της δημοσιεύσεώς τους είτε σε βιβλία είτε σε περιοδικά είτε στο διαδίκτυο. Τούτο έχει πράξει και ο ίδιος χωρίς καμία εξαίρεση.

ΕΦΑΡΜΟΣΜΕΝΑ ΜΑΘΗΜΑΤΙΚΑ ΙΙ
για ΠΟΛΙΤΙΚΟΥΣ ΜΗΧΑΝΙΚΟΥΣ

Τεύχος 3

ΧΡΗΣΙΜΕΣ ΕΝΤΟΛΕΣ ΤΗΣ *MATHEMATICA*
για ΠΟΛΙΤΙΚΟΥΣ ΜΗΧΑΝΙΚΟΥΣ

Νικόλαος Ι. Ιωακειμίδης

Τομέας Εφαρμοσμένων Μαθηματικών και Μηχανικής,
Γενικό Τμήμα Πολυτεχνικής Σχολής Πανεπιστημίου Πατρών

GOTSIS
ΕΚΔΟΣΕΙΣ

1η Έκδοση
Πάτρα, 2008

Νικόλαος Ι. Ιωακειμίδης *(συγγραφέας, Γενικό Τμήμα, Πολυτεχνική Σχολή, Πανεπιστήμιο Πατρών)*
ΕΦΑΡΜΟΣΜΕΝΑ ΜΑΘΗΜΑΤΙΚΑ ΙΙ *για ΠΟΛΙΤΙΚΟΥΣ ΜΗΧΑΝΙΚΟΥΣ, Τεύχος 3:*
ΧΡΗΣΙΜΕΣ ΕΝΤΟΛΕΣ ΤΗΣ MATHEMATICA *για ΠΟΛΙΤΙΚΟΥΣ ΜΗΧΑΝΙΚΟΥΣ*
1η Έκδοση: Φεβρουάριος 2008

ISBN 978-960-98187-4-2 (Αυτό το τεύχος. Παράκληση για χρήση του ISBN για παραγγελίες.)
ISBN SET 978-960-98187-1-1 (Εφαρμοσμένα Μαθηματικά ΙΙ *για Πολιτικούς Μηχανικούς* SET)

ΑΠΟΚΛΕΙΣΤΙΚΗ ΔΙΑΘΕΣΗ, ΕΠΙΚΟΙΝΩΝΙΑ ΜΕ ΤΟΝ ΕΚΔΟΤΗ

Βιβλιοπωλείο «Γνώση», Οδός Αράτου 41, 262.21 ΠΑΤΡΑ
Τηλέφωνο: (+30) 2610 226453, Fax: (+30) 2610 226690,
E-mail: gnossis@otenet.gr

ΕΚΤΥΠΩΣΗ–ΒΙΒΛΙΟΔΕΣΙΑ

«Ταχύτυπο», Ταχυεκτυπώσεις – Γραβάνης Ε.Π.Ε., Πάροδος Διοδώρου 160, Βελβίτσι, 264.43 ΠΑΤΡΑ
Τηλέφωνα: (+30) 2610 461780 έως (+30) 2610 461790, E-mail: info@tachytypo.gr

ΦΩΤΟΓΡΑΦΙΑ ΕΞΩΦΥΛΛΟΥ

Η κεντρική είσοδος και ένα πολύ μικρό τμήμα του κτιρίου του Τμήματος Πολιτικών Μηχανικών της Πολυτεχνικής Σχολής του Πανεπιστημίου Πατρών στην Πανεπιστημιούπολη στο Ρίο Πατρών

ΕΠΙΚΟΙΝΩΝΙΑ ΜΕ ΤΟ ΣΥΓΓΡΑΦΕΑ

Νικόλαος Ι. Ιωακειμίδης
Τομέας Εφαρμοσμένων Μαθηματικών και Μηχανικής
Γενικό Τμήμα, Πολυτεχνική Σχολή, Πανεπιστήμιο Πατρών
Πανεπιστημιούπολη Πατρών, 265.04 ΡΙΟ, ΠΑΤΡΑ
Τηλέφωνα: (+30) 2610 432257, (+30) 2610 997378,
E-mail: n.ioakimidis@upatras.gr, http://www.des.upatras.gr/amm/ioakimidis/ioakimidis.htm

ΠΕΡΙΕΧΟΜΕΝΑ

ΤΑ ΜΕΡΗ ΑΥΤΩΝ ΤΩΝ ΔΙΔΑΚΤΙΚΩΝ ΒΙΒΛΙΩΝ ΕΦΑΡΜΟΣΜΕΝΩΝ ΜΑΘΗΜΑΤΙΚΩΝ ΙΙ ΚΑΙ ΙΙΙ *για ΠΟΛΙΤΙΚΟΥΣ ΜΗΧΑΝΙΚΟΥΣ*

1. **ΕΦΑΡΜΟΣΜΕΝΑ ΜΑΘΗΜΑΤΙΚΑ ΙΙ *για ΠΟΛΙΤΙΚΟΥΣ ΜΗΧΑΝΙΚΟΥΣ***

    - ***ΜΕΡΟΣ Α:*** **ΕΦΑΡΜΟΣΜΕΝΕΣ ΣΥΝΗΘΕΙΣ ΔΙΑΦΟΡΙΚΕΣ ΕΞΙΣΩΣΕΙΣ**
      για ΠΟΛΙΤΙΚΟΥΣ ΜΗΧΑΝΙΚΟΥΣ (στο Τεύχος 1)
    - **ΕΦΑΡΜΟΣΜΕΝΕΣ ΑΣΚΗΣΕΙΣ ΚΑΙ NOTEBOOKS ΙΙ**
      για ΠΟΛΙΤΙΚΟΥΣ ΜΗΧΑΝΙΚΟΥΣ (στο Τεύχος 2)
    - **ΧΡΗΣΙΜΕΣ ΕΝΤΟΛΕΣ ΤΗΣ *MATHEMATICA***
      για ΠΟΛΙΤΙΚΟΥΣ ΜΗΧΑΝΙΚΟΥΣ (στο Τεύχος 3)

2. **ΕΦΑΡΜΟΣΜΕΝΑ ΜΑΘΗΜΑΤΙΚΑ ΙΙΙ *για ΠΟΛΙΤΙΚΟΥΣ ΜΗΧΑΝΙΚΟΥΣ***

    - ***ΜΕΡΟΣ Β:*** **ΕΦΑΡΜΟΣΜΕΝΕΣ ΔΙΑΦΟΡΙΚΕΣ ΕΞΙΣΩΣΕΙΣ ΜΕ ΜΕΡΙΚΕΣ ΠΑΡΑΓΩΓΟΥΣ**
      για ΠΟΛΙΤΙΚΟΥΣ ΜΗΧΑΝΙΚΟΥΣ (στο Τεύχος 1)
    - ***ΜΕΡΟΣ Γ:*** **ΕΦΑΡΜΟΣΜΕΝΕΣ ΟΛΟΚΛΗΡΩΤΙΚΕΣ ΕΞΙΣΩΣΕΙΣ**
      για ΠΟΛΙΤΙΚΟΥΣ ΜΗΧΑΝΙΚΟΥΣ (επίσης στο Τεύχος 1)
    - ***ΜΕΡΟΣ Δ:*** **ΕΦΑΡΜΟΣΜΕΝΕΣ ΜΙΓΑΔΙΚΕΣ ΣΥΝΑΡΤΗΣΕΙΣ**
      για ΠΟΛΙΤΙΚΟΥΣ ΜΗΧΑΝΙΚΟΥΣ (επίσης στο Τεύχος 1)
    - **ΕΦΑΡΜΟΣΜΕΝΕΣ ΑΣΚΗΣΕΙΣ ΚΑΙ NOTEBOOKS ΙΙΙ**
      για ΠΟΛΙΤΙΚΟΥΣ ΜΗΧΑΝΙΚΟΥΣ (στο Τεύχος 2)

ΠΡΟΛΟΓΟΣ

Στον παρόν Τεύχος 3 του συγγράμματος ***Εφαρμοσμένα Μαθηματικά ΙΙ για Πολιτικούς Μηχανικούς*** *με τον τίτλο* ***Χρήσιμες Εντολές της Mathematica για Πολιτικούς Μηχανικούς*** *έχει γίνει μια μάλλον σύντομη καταγραφή των κυριότερων εντολών της Mathematica που είναι χρήσιμες για τον Πολιτικό Μηχανικό κυρίως στα Εφαρμοσμένα Μαθηματικά ΙΙ, αλλ' επίσης και σε άλλα μαθήματα, όπως στα Εφαρμοσμένα Μαθηματικά ΙΙΙ, και στην επιστήμη του γενικότερα.*

Παρουσιάζονται 224 εντολές της Mathematica ταξινομημένες σε 21 notebooks εντολών. Η παρουσίαση της κάθε εντολής είναι γενικά συνοπτική: σκοπός της, τρόπος συντάξεώς της, περιγραφή της και παραδείγματα σε μερικές περιπτώσεις από την Επιστήμη του Πολιτικού Μηχανικού. Στη συνέχεια παρατίθεται και ένα εισαγωγικό notebook που αφορά στις animations (στα κινούμενα σχήματα, στην κίνηση σε σχήματα) μια πολύ ενδιαφέρουσα δυνατότητα της Mathematica.

Ο αναγνώστης/η αναγνώστρια Πολιτικός Μηχανικός θεωρείται πως είναι ήδη γενικά εξοικειωμένος/εξοικειωμένη με τη Mathematica κυρίως μετά τη μελέτη και τη χρήση στα Εφαρμοσμένα Μαθηματικά Ι του τόσο αξιόλογου, καλογραμμένου και ενδιαφέροντος βιβλίου του συναδέλφου κ. Κωνσταντίνου Παπαδάκη για την ***Εισαγωγή στο Mathematica****. Εδώ απλά γίνεται μια σύντομη παρουσίαση των χρήσιμων εντολών της Mathematica για τον Πολιτικό Μηχανικό, χωρίς όμως εκτενή περιγραφή τους και χωρίς πάρα πολλά παραδείγματα και επεξηγήσεις σε αντίθεση με το βιβλίο του κ. Παπαδάκη. Δηλαδή εδώ πρόκειται για ένα είδος κάπως εκτενούς «ευρετηρίου» των χρήσιμων εντολών της Mathematica για Πολιτικούς Μηχανικούς με έμφαση στο μάθημα Εφαρμοσμένα Μαθηματικά ΙΙ, αλλά και στο μάθημα Εφαρμοσμένα Μαθηματικά ΙΙΙ.*

Στο τεύχος αυτό πέρα από τις κλασικές εντολές της Mathematica αναφέρονται και αρκετές εξειδικευμένες εντολές της ακόμη και πολύ εξειδικευμένες εντολές που δεν περιλαμβάνονται στο βιβλίο του κ. Παπαδάκη και μερικές φορές σχεδόν σε κανένα βιβλίο για τη Mathematica. Μερικές από αυτές μπορούν μάλιστα να θεωρηθούν πάρα πολύ εξειδικευμένες. Παραδείγματος χάρη, τέτοια είναι η εντολή Biharmonic *για το διαρμονικό τελεστή, ο οποίος παρουσιάζεται στην τασική συνάρτηση του Airy $A(x,y)$ στην Επίπεδη Ελαστικότητα και επίσης στο βέλος κάμψεως $w(x,y)$ στις Πλάκες. Ανάλογα πάρα πολύ εξειδικευμένη είναι και η εντολή* PlotHamiltonianField *για τη σχεδίαση πεδίου Χαμιλτονιανής. Αυτή όμως η εντολή είναι ιδανική για τη σχεδίαση του πεδίου ταχύτητας σε συνηθισμένη ροή ιδεατού ρευστού με βάση τη ροϊκή συνάρτηση (ή συνάρτηση ροής) $\Psi(x,y)$. Από την άλλη πλευρά η επίσης εξειδικευμένη εντολή* MiniMaxApproximation *επιτρέπει στον Πολιτικό Μηχανικό να δημιουργεί πάρα πολύ καλές προσεγγίσεις συναρτήσεων πολύ καλύτερες από εκείνες της απλής πολυωνυμικής παρεμβολής.*

Όπως και στα κύρια μέρη αυτών των διδακτικών συγγραμμάτων ***Εφαρμοσμένα Μαθηματικά ΙΙ και ΙΙΙ για Πολιτικούς Μηχανικούς****, έτσι κι εδώ έχουν συμπεριληφθεί ορισμένες (αν και όχι πάρα πολλές) εφαρμογές από την Επιστήμη του Πολιτικού Μηχανικού: Ταλαντώσεις, Δυναμική των Κατασκευών, Μηχανική των Υλικών (Ελαστικότητα, Δοκοί), Ρευστομηχανική, Πλάκες και Περιβαλλοντική Μηχανική. Εύλογα ιδιαίτερη έμφαση δόθηκε στο notebook για την επίλυση διαφορικών εξισώσεων.*

Φυσικά το βιβλίο αυτό δε θα υπήρχε χωρίς την πρωτοβουλία του Τμήματος Πολιτικών Μηχανικών να εισαγάγει εργαστήριο στο Υπολογιστικό Κέντρο στα Εφαρμοσμένα Μαθηματικά, μια ιδιαίτερα αξιέπαινη πρωτοβουλία. Στην πρωτοβουλία αυτή ήρθε αρωγός και το Γενικό Τμήμα, Τομέας Εφαρμοσμένων Μαθηματικών και Μηχανικής, όπου ανατίθενται τα μαθήματα Εφαρμοσμένων Μαθηματικών του Τμήματος Πολιτικών Μηχανικών, με την αγορά όλου του αναγκαίου λογισμικού της Mathematica: πάνω από πενήντα άδειες διαρκούς χρήσεως. Και στα δύο αυτά Τμήματα της

Πολυτεχνικής Σχολής του Πανεπιστημίου Πατρών είμαι ειλικρινά ευγνώμων. Είμαι επίσης ευγνώμων στο Διευθυντή του Υπολογιστικού Κέντρου κ. Γεώργιο Τσόκο για τη συνεχή βοήθειά του κατά τη διάρκεια των εργαστηρίων Εφαρμοσμένων Μαθηματικών και για την άριστη λειτουργία του Υπολογιστικού Κέντρου γενικότερα.

Ίσως όμως όλα αυτά δεν είναι τόσο σημαντικά μπροστά στο «αγκάλιασμα» από πολλούς φοιτητές και φοιτήτριες Πολιτικούς Μηχανικούς του εργαστηρίου στα Εφαρμοσμένα Μαθηματικά και στο έντονο ενδιαφέρον τους γι' αυτό. Τούτο μάλιστα έχει αυξηθεί κατά τα τελευταία Ακαδημαϊκά Έτη. Αυτό το ενδιαφέρον υπήρξε και η κύρια ώθηση σ' εμένα, ώστε να ξεκινήσω σιγά–σιγά την προετοιμασία του παρόντος βιβλίου με τη σύνοψη των χρήσιμων εντολών της Mathematica για Πολιτικούς Μηχανικούς.

Είναι σίγουρα κοινοτοπία, αλλ' ας λεχθεί κι εδώ για την πληρότητα αυτού του σύντομου προλόγου, ότι ο ηλεκτρονικός υπολογιστής αποτελεί αναγκαίο «εργαλείο» του Πολιτικού Μηχανικού για την εκτέλεση κάθε είδους υπολογισμών στην εργασία του, εδώ κυρίως συμβολικών υπολογισμών, που συχνά όμως καταλήγουν σε αριθμητικά αποτελέσματα. Φυσικά είναι αυτονόητο ότι καμία εντολή δε μπορεί να συνταχθεί για πρόβλημα του Πολιτικού Μηχανικού (και όχι μόνο ...), εάν ο συντάκτης της, έστω και ο ειδήμων στον υπολογιστή, εδώ στους συμβολικούς υπολογισμούς και στη Mathematica, δεν κατέχει πλήρως τις σχετικές θεωρητικές γνώσεις. Αυτές είναι που θα του επιτρέψουν να προγραμματίσει σωστά, ώστε να βρει τα ζητούμενα αποτελέσματα. Δεν είναι αρκετό να γνωρίζει τις εντολές τις Mathematica. Πρέπει πρώτα απ' όλα να ξέρει Μαθηματικά και να κατέχει την Επιστήμη του Πολιτικού Μηχανικού!

Στο σημείο αυτό θα ήθελα να ευχαριστήσω ξανά τις Gotsis Εκδόσεις στην Πάτρα και ιδιαίτερα τον υπεύθυνό τους κ. Άγγελο Γκότση για το ενδιαφέρον τους στο παρόν βιβλίο και την τόσο επιμελημένη προετοιμασία και εκτύπωσή του. Πρόκειται, επαναλαμβάνεται, για μια ιδιαίτερα αξιέπαινη πρωτοβουλία των Gotsis Εκδόσεων και μάλιστα ουσιαστικά χωρίς ιδιαίτερο οικονομικό όφελος. Το τελευταίο δυστυχώς ισχύει εξαιτίας του τόσου εξειδικευμένου χαρακτήρα των διδακτικών αυτών βιβλίων που δεν επιτρέπει την ευρεία διάθεσή τους σε πλατύ αναγνωστικό κοινό και τα περιορίζει σε Πολιτικούς Μηχανικούς.

*Εντούτοις θεωρώ πως είναι προτιμότερο να έχουν οι φοιτητές και οι φοιτήτριες Πολιτικοί Μηχανικοί τα «δικά τους» βιβλία Εφαρμοσμένων Μαθηματικών παρά να προσφεύγουν σε γενικά βιβλία Μαθηματικών ή ακόμη και σε γενικά βιβλία Εφαρμοσμένων Μαθηματικών. Αυτά μερικές φορές δε μπορούν να τους δώσουν/να τις δώσουν το βέλτιστο για τις δικές τους ανάγκες στην Επιστήμη του Πολιτικού Μηχανικού. Μάλιστα σε ορισμένες περιπτώσεις δε δείχνουν καθαρά τη συσχέτιση ανάμεσα στα Εφαρμοσμένα Μαθηματικά και στα άλλα μαθήματα της Επιστήμης του Πολιτικού Μηχανικού. Με τα παρόντα δύο συγγράμματα **Εφαρμοσμένα Μαθηματικά ΙΙ και ΙΙΙ** γίνεται εδώ η προσπάθεια να καταστεί εμφανές πως τα μαθήματα Εφαρμοσμένα Μαθηματικά ΙΙ και ΙΙΙ δεν είναι αποκομμένα από τα άλλα μαθήματα που διδάσκεται ο Πολιτικός Μηχανικός, αλλά είναι φυσιολογικά συνδεδεμένα με αυτά σε μια αλυσίδα γνώσεων.*

Δεν πρέπει επίσης να μη σημειωθεί ότι ο όρος «η Mathematica» που χρησιμοποιείται εδώ αντί για τον όρο «το Mathematica» για το ίδιο πρόγραμμα φαίνεται να οφείλεται στο Στέφανο Τραχανά, που τον εισήγαγε στο δικό του βιβλίο για τη Mathematica το 2001 (αναφέρεται στη βιβλιογραφία).

Τελειώνοντας, θα ήθελα να σημειώσω πως με μεγάλη χαρά μου θα δεχθώ κάθε υπόδειξη για τη βελτίωση και αυτού του βιβλίου είτε με τη διόρθωση λαθών που έχουν παρεισφρύσει είτε και με ουσιαστικότερες υποδείξεις ως προς την επιλογή των εντολών, το περιεχόμενο, τα παραδείγματα και τις εφαρμογές, κλπ. Θα είμαι πραγματικά ευγνώμων για κάθε τέτοια υπόδειξη!

Πάτρα, Φεβρουάριος 2008
Νικόλαος Ι. Ιωακειμίδης
e-mail: n.ioakimidis@upatras.gr

ΠΑΡΑΤΗΡΗΣΕΙΣ

- ***ΠΑΡΑΤΗΡΗΣΗ ΓΙΑ ΤΟ ΠΡΟΓΡΑΜΜΑ ΥΠΟΛΟΓΙΣΜΩΝ***

Το περιεχόμενο αυτού του Τεύχους 3 των **Εφαρμοσμένων Μαθηματικών ΙΙ για Πολιτικούς Μηχανικούς** αφορά σε **Χρήσιμες Εντολές της *Mathematica* για Πολιτικούς Μηχανικούς**. Χρησιμοποιήθηκε η έκδοση (version) 4.1 της *Mathematica* που είναι διαθέσιμη στο γράφοντα. Το κύριο μέρος του τεύχους αυτού αποτελείται από είκοσι δύο notebooks της *Mathematica*: εικοσιένα για χρήσιμες εντολές της και ένα για κίνηση σε σχήματα (animations). Έτσι από εκπαιδευτικής απόψεως ο χρήστης κι η χρήστρια της *Mathematica* Πολιτικός Μηχανικός πρέπει να παίρνουν τα ίδια ακριβώς αποτελέσματα για τις ίδιες εντολές, εφόσον βέβαια χρησιμοποιούν την έκδοση (version) 4.1 της *Mathematica*. Για άλλες εκδόσεις της *Mathematica* προφανώς θα υπάρχουν κάποιες μικροδιαφορές, όχι όμως ιδιαίτερα σημαντικές για τον Πολιτικό Μηχανικό.

- ***ΠΑΡΑΤΗΡΗΣΗ ΓΙΑ ΤΗΝ ΟΡΘΟΤΗΤΑ ΤΩΝ ΥΠΟΛΟΓΙΣΜΩΝ***

Η *Mathematica* είναι ένα εξαιρετικά αξιόπιστο μαθηματικό πρόγραμμα συμβολικών και αριθμητικών υπολογισμών επιπλέον και με εκπληκτικές δυνατότητες στις γραφικές παραστάσεις. Εντούτοις κάθε σύνθετο πρόγραμμα, όπως είναι και η *Mathematica*, είναι πιθανό να περιέχει έναν πολύ μικρό αριθμό λαθών. Κατά συνέπεια, ενώ τα αποτελέσματα των υπολογισμών με τη *Mathematica* στα notebooks που ακολουθούν είναι σωστά στη συντριπτική πλειονότητά τους, εντούτοις σε εξαιρετικά σπάνιες περιπτώσεις μπορεί να είναι εσφαλμένα. Επομένως είναι καλό ο Πολιτικός Μηχανικός να επαληθεύει τα αποτελέσματα της *Mathematica*. Αυτό μπορεί να γίνεται είτε (α) με άμεση επαλήθευση, για παράδειγμα αντικαθιστώντας τη λύση μιας διαφορικής εξισώσεως στη διαφορική εξίσωση που καταλήγει έτσι σε ταυτότητα είτε (β) κάνοντας τους υπολογισμούς και με δεύτερο, διαφορετικό τρόπο και βρίσκοντας το ίδιο (ή απόλυτα ισοδύναμο) αποτέλεσμα, για παράδειγμα λύνοντας μια γραμμική διαφορική εξίσωση με σταθερούς συντελεστές (i) με τον κλασικό τρόπο και (ii) με τη μέθοδο του μετασχηματισμού Laplace και συγκρίνοντας τα δύο αποτελέσματα.

- ***ΠΑΡΑΤΗΡΗΣΗ ΓΙΑ ΤΟ ΕΜΠΟΡΙΚΟ ΣΗΜΑ MATHEMATICA***

Η λέξη *Mathematica* αποτελεί κατατεθειμένο εμπορικό σήμα (registered trademark, σήμα κατατεθέν) της εταιρείας Wolfram Research, Inc., 100 Trade Center Drive, Champaign, IL 61820–7237, Η.Π.Α. Δικτυακός τόπος: http://www.wolfram.com. Η χρήση της λέξεως *Mathematica* εδώ γίνεται με πλήρη αναγνώριση του εμπορικού αυτού σήματος της εταιρείας Wolfram Research, Inc.

■ *Notebook E0*

ΠΑΡΑΤΗΡΗΣΕΙΣ ΚΑΙ ΕΝΤΟΛΕΣ ΓΕΝΙΚΗΣ ΧΡΗΣΕΩΣ

5 ΠΑΡΑΤΗΡΗΣΕΙΣ ΚΑΙ 21 ΕΝΤΟΛΕΣ: **Γ1. Timing, Γ2. Needs, Γ3. Off, Γ4. Clear, Γ5. Remove, Γ6. Sign, Γ7. Options, Γ8. SetOptions, Γ9. Attributes, Γ10. SetAttributes, Γ11. ClearAttributes, Γ12. Short, Γ13. ?, Γ14. ??, Γ15. OutPutForm, Γ16. TraditionalForm, Γ17. FullForm, Γ18. Head, Γ19. Function, Γ20. Evaluate, Γ21. Map**

■ ΠΑΡΑΤΗΡΗΣΗ R1: ΕΜΦΑΝΙΣΗ ΕΙΔΙΚΩΝ ΣΥΜΒΟΛΩΝ

Εκτός από τα γράμματα (μικρά και κεφαλαία) η *Mathematica* εμφανίζει αρκετά εύκολα και πάρα πολλά μαθηματικά σύμβολα τα οποία δεν είναι διαθέσιμα στο πληκτρολόγιο με ακολουθίες **Esc**, δηλαδή **Esc** κατάλληλοι χαρακτήρες και ξανά **Esc**. Παραδείγματα τέτοιων ακολουθιών **Esc** είναι τα εξής:

Esc int Esq → $\int$, **Esc cint Esc** → $\oint$, **Esc sum Esc** → $\sum$, **Esc prod Esc** → $\prod$, **Esc inf Esc** → ∞,
Esc +- Esc → $\pm$, **Esc -+ Esc** → $\mp$, **Esc <= Esc** → $\leq$, **Esc >= Esc** → $\geq$, **Esc dd Esc** → d,
Esc ee Esc → e, **Esc ii Esc** → i, **Esc cross Esc** → $\times$, **Esc pd Esc** → ∂, **Esc == Esc** → $==$,
Esc and Esc → $\wedge$, **Esc or Esc** → $\vee$, **Esc not Esc** → $\neg$, **Esc -> Esc** → $\rightarrow$, **Esc === Esc** → $\equiv$,
Esc deg Esc → $^\circ$, **Esc del Esc** → ∇, **Esc => Esc** → $\Rightarrow$, **Esc <=> Esc** → $\Leftrightarrow$, **Esc elem Esc** → $\in$,
Esc ex Esc → $\exists$, **Esc fa Esc** → $\forall$ και πάρα πολλά ακόμη μαθηματικά σύμβολα: όλα τα σύμβολα.

■ ΠΑΡΑΤΗΡΗΣΗ R2: ΕΜΦΑΝΙΣΗ ΕΛΛΗΝΙΚΩΝ ΓΡΑΜΜΑΤΩΝ

Η *Mathematica* εμφανίζει εύκολα και τα Ελληνικά γράμματα (τόσο τα κεφαλαία όσο και και μικρά) σαν σύμβολα στις εντολές της με **Esc** ένα αντίστοιχο Λατινικό γράμμα και ξανά **Esc**. Για ορισμένα όμως γράμματα η αντιστοιχία δεν είναι προφανής. Τα γράμματα αυτά είναι τα εξής:

Esc q Esq → θ, **Esc Q Esc** → Θ, **Esc x Esc** → ξ, **Esc X Esc** → Ξ, **Esc u Esc** → υ, **Esc Y Esc** → Υ,
Esc c Esc → χ, **Esc y Esc** → ψ, **Esc Y Esc** → Ψ, **Esc o Esc** → ω, **Esc O Esc** → Ω, **Esc fs Esc** → ς

Μια εναλλακτική δυνατότητα (που όμως δε συνιστάται στις εντολές της *Mathematica*, ενώ συνιστάται στη γραφή κειμένου όπως συμβαίνει εδώ) αποτελεί το γύρισμα του πληκτρολογίου στα Ελληνικά, πριν να γραφεί κάποιο Ελληνικό γράμμα, και μετά επιστροφή στο Λατινικό πληκτρολόγιο. Παραδείγματα:

```
In[1]:= {"αβγδεζηθικλμνξπρστυφχψως", "ΓΔΘΛΞΠΣΦΨΩ"};
```

■ ΠΑΡΑΤΗΡΗΣΗ R3: ΠΡΟΤΕΡΑΙΟΤΗΤΑ ΤΩΝ ΠΡΑΞΕΩΝ

Η *Mathematica* χρησιμοποιεί τους συνηθισμένους κανόνες για την προτεραιότητα των πράξεων: προηγούνται οι υψώσεις σε δυνάμεις. Ακολουθούν με ίσες προτεραιότητες οι πολλαπλασιασμοί και οι διαιρέσεις. Τέλος έρχονται οι προσθέσεις και οι αφαιρέσεις (κι αυτές με ίσες, αλλά τώρα τόσο μικρές προτεραιότητες). Οι προτεραιότητες αυτές μπορούν να αλλάζουν με παρενθέσεις. Παραδείγματα:

```
In[2]:= {q = 2 3^3 + 4 + 5 / 6, q == 2 (3^3) + 4 + (5 / 6), q == 2 (3^3) + (4 + 5) / 6 }
```

Out[2]= $\left\{\frac{353}{6}, \text{True}, \text{False}\right\}$

■ ΠΑΡΑΤΗΡΗΣΗ R4: ΠΑΡΕΝΘΕΣΕΙΣ, ΑΓΚΥΛΕΣ ΚΑΙ ΑΓΚΙΣΤΡΑ

Η *Mathematica* χρησιμοποιεί τις παρενθέσεις ( ) στις παραστάσεις της, για να καθορίσει την προτεραιότητα στις αριθμητικές και αλγεβρικές πράξεις, όπου αυτή χρειάζεται να καθορισθεί είτε ουσιαστικά είτε απλά για να γίνει σαφής η παράσταση. Τις αγκύλες [ ] τις χρησιμοποιεί στις εντολές της (και στις συναρτήσεις της), για να περικλείουν τα ορίσματα. Δεν επιτρέπεται να χρησιμοποιούνται αγκύλες σε αλγεβρικές παραστάσεις. Τα άγκιστρα { } τα χρησιμοποιεί σε λίστες, σύνολα και μητρώα. Παραδείγματα:

```
In[3]:= {1 + 2 / 3, (1 + 2) / 3, (a + b)^2, Expand[(a + b)^2],
        Cos[ω t], v = {a, b, c}, Union[{d, e}, {e, f}] }
```

Out[3]= $\left\{\frac{5}{3}, 1, (a+b)^2, a^2 + 2\,a\,b + b^2, \text{Cos}[t\,\omega], \{a, b, c\}, \{d, e, f\}\right\}$

■ ΠΑΡΑΤΗΡΗΣΗ R5: ΟΡΙΣΜΟΣ ΣΥΝΑΡΤΗΣΕΩΝ

Στη *Mathematica* μπορούν εύκολα να ορισθούν συναρτήσεις, όπως φαίνεται στα παραδείγματα που ακολουθούν. Σημειώνεται με έμφαση η *υποχρεωτική* χρήση της κάτω παύλας μετά από καθένα όρισμα (μεταβλητή) στους ορισμούς συναρτήσεων. Αυτή είναι απόλυτα αναγκαία. Στον ορισμό συναρτήσεως χρησιμοποιείται το σύμβολο ίσον, δηλαδή το =. Όμως σε ορισμένες (όχι και πολλές ...) περιπτώσεις που είναι επιθυμητός ο καθυστερημένος (όχι ο άμεσος) υπολογισμός της συναρτήσεως που ορίζεται χρησιμοποιείται το σύμβολο := αντί για το σύμβολο =. Παραδείγματα:

```
In[4]:= {p[t_] = p0 Cos[ω t], p[0], p[1], p[t0]}
```

Out[4]= $\{\text{Cos}[t\,\omega]\,p_0, p_0, \text{Cos}[\omega]\,p_0, \text{Cos}[\omega\,t_0]\,p_0\}$

```
In[5]:= {u[x_, y_, z_] = x y^2 z^3, u[a, b, c]}
```

Out[5]= $\{x\,y^2\,z^3, a\,b^2\,c^3\}$

```
In[6]:= {f[k_, n_] := Table[m^k, {m, 1, n}], f[2, 3], f[5, 5]}
```

```
Out[6]= {Null, {1, 4, 9}, {1, 32, 243, 1024, 3125}}
```

■ ΕΝΤΟΛΗ Γ1: ΧΡΟΝΟΣ ΜΗΧΑΝΗΣ ΓΙΑ ΥΠΟΛΟΓΙΣΜΟ

Timing[*Παράσταση*]

Η εντολή αυτή υπολογίζει την παράσταση στο όρισμά της και δίνει σαν αποτέλεσμα μια λίστα με δύο στοιχεία. Το πρώτο στοιχείο της λίστας είναι ο χρόνος μηχανής (CPU) σε δευτερόλεπτα ο οποίος απαιτήθηκε για τον υπολογισμό και το δεύτερο στοιχείο της λίστας είναι το ίδιο το αποτέλεσμα του υπολογισμού, το αποτέλεσμα που θα παίρναμε και χωρίς τη χρήση της εντολής **Timing**. Παράδειγμα:

```
In[7]:= Timing[Expand[(a + b + c)^2]]
Out[7]= {0. Second, a^2 + 2 a b + b^2 + 2 a c + 2 b c + c^2}
```

■ ΕΝΤΟΛΗ Γ2: ΚΛΗΣΗ (ΦΟΡΤΩΜΑ) ΠΑΚΕΤΟΥ

Needs["*Πακέτο*"] ή ισοδύναμα **<< *Πακέτο***

Με την εντολή αυτή καλούμε (φορτώνουμε) ένα πακέτο της *Mathematica* που περιέχει χρήσιμες για μας εντολές στους παραπέρα υπολογισμούς μας. Σημειώνουμε ότι το σύμβολο της βαρείας ‘ (που υπεισέρχεται στα ονόματα των πακέτων της *Mathematica*) βρίσκεται στο πάνω αριστερό άκρο του πληκτρολογίου στο ίδιο πλήκτρο με την Ελληνική περισπωμένη: ακριβώς από κάτω. *Παρατήρηση:* Εάν κατά λάθος χρησιμοποιηθεί κάποια εντολή της *Mathematica* πριν από την κλήση (το φόρτωμα) του πακέτου που την ορίζει, τότε αυτή θα πρέπει να απομακρυνθεί με την εντολή **Remove** (και όχι με την εντολή **Clear**, που δε φέρνει κανένα αποτέλεσμα σ’ αυτήν την περίπτωση) πριν από την κλήση του πακέτου ή λίγο μετά. Η κλήση του πακέτου θα έπρεπε να είχε γίνει από την αρχή! Παραδείγματα:

```
In[8]:= {Re[2 + 3 ⅈ], Re[Cosh[2 + 3 ⅈ]]}
Out[8]= {2, Re[Cosh[2 + 3 ⅈ]]}

In[9]:= Needs["Algebra`ReIm`"]

In[10]:= Re[Cosh[2 + 3 ⅈ]]
Out[10]= Cos[3] Cosh[2]

In[11]:= LaplacePDE = Laplacian[F[x, y, z], Cartesian[x, y, z]] == 0
Out[11]= Laplacian[F[x, y, z], Cartesian[x, y, z]] == 0

In[12]:= Remove[Laplacian, Cartesian]

In[13]:= Needs["Calculus`VectorAnalysis`"]

In[14]:= LaplacePDE = Laplacian[G[x, y, z], Cartesian[x, y, z]] == 0
Out[14]= G^(0,0,2)[x, y, z] + G^(0,2,0)[x, y, z] + G^(2,0,0)[x, y, z] == 0
```

■ ΕΝΤΟΛΗ Γ3: ΜΗ ΕΚΤΥΠΩΣΗ ΜΗΝΥΜΑΤΩΝ

{Off[General :: spell], Off[General :: spell1]} (χωρίς κενό στο σύμβολο : :)

Στις δύο πιο πάνω μορφές της η εντολή **Off** αποτρέπει την εκτύπωση μηνυμάτων σχετικών με την ορθογραφία συμβόλων: **spell** και **spell1** αντίστοιχα. Αυτό μας διευκολύνει μερικές φορές και πράγματι συνιστάται, όταν είμαστε σίγουροι ότι δε χρειαζόμαστε τα σχετικά προειδοποιητικά μηνύματα. Άλλες όμως φορές μας είναι ιδιαίτερα βλαβερό, επειδή η μη εκτύπωση του προειδοποιητικού μηνύματος σε περίπτωση κάποιου ορθογραφικού λάθους δε μας επιτρέπει να το επισημάνουμε και να το βρούμε εύκολα. Άρα καλό είναι η εντολή αυτή να χρησιμοποιείται με φειδώ και κυρίως στην τελική φάση της προετοιμασίας ενός notebook με τα περισσότερα ορθογραφικά λάθη ήδη διορθωμένα. Παραδείγματα:

```
In[15]:= {list1a = {a}, list1b = {a, b}}

  General::spell1 :
   Possible spelling error: new symbol name "list1b" is similar to existing symbol "list1a".

Out[15]= {{a}, {a, b}}

In[16]:= {list1c = {a, b, c}, list1d = {a, b, c, d}}

  General::spell :
   Possible spelling error: new symbol name "list1c" is similar to existing symbols {list1a, list1b}.

  General::spell :
   Possible spelling error: new symbol name "list1d" is similar to existing symbols {list1a, list1b, list1c}.

Out[16]= {{a, b, c}, {a, b, c, d}}

In[17]:= {Off[General::spell], Off[General::spell1]};

In[18]:= {list1e = {e, f, g}, list1f = {e, f, g, h, i}}

Out[18]= {{e, f, g}, {e, f, g, h, i}}
```

■ ΕΝΤΟΛΗ Γ4: ΣΒΗΣΙΜΟ (ΚΑΘΑΡΙΣΜΑ) ΜΕΤΑΒΛΗΤΗΣ

Clear[*Μεταβλητή*]

Clear[*Μεταβλητή-1, Μεταβλητή-2, Μεταβλητή-3, ...*]

Clear["@"]

***Μεταβλητή* =.** ή ***ΛίσταΜεταβλητών* =.**

Σβήνει (καθαρίζει) μια μεταβλητή που έχει συγκεκριμένη τιμή, δηλαδή η μεταβλητή αυτή γίνεται ελεύθερη: χωρίς καμία τιμή. Έτσι θα μπορεί να χρησιμοποιείται σαν ελεύθερη μεταβλητή χωρίς καθορισμένη τιμή. Ανάλογα σβήνει και μια συνάρτηση. Επίσης ανάλογα και για πολλές μεταβλητές ή/και συναρτήσεις. Στην τρίτη μορφή της σβήνει όλες τις μεταβλητές ή συναρτήσεις που αρχίζουν από μικρό γράμμα. Η τελευταία μορφή της εντολής με το ίσον και μετά μία τελεία μπορεί επίσης να χρησιμοποιείται. Όταν υπάρχει δείκτης στη μεταβλητή (ή στις μεταβλητές), η τελευταία μορφή είναι και η μόνη που είναι αποτελεσματική, λειτουργεί για το σβήσιμο της μεταβλητής. Παραδείγματα:

```
In[19]:= {a = 1, b = 2, c = 3, {a, b, c}, Clear[a], {a, b, c}, Clear["@"], {a, b, c}}

Out[19]= {1, 2, 3, {1, 2, 3}, Null, {a, 2, 3}, Null, {a, b, c}}

In[20]:= {u[t_] = u0 Cos[ω t], u[0], u[T], Clear[u], u[0], u[T]}

Out[20]= {u0 Cos[t ω], u0, u0 Cos[T ω], Null, u[0], u[T]}

In[21]:= {α = π / 2, β = π / 3, γ = π, α + β + γ, Clear[α, β, γ], α + β + γ}
```

Out[21]= $\left\{\frac{\pi}{2}, \frac{\pi}{3}, \pi, \frac{11\pi}{6}, \text{Null}, \alpha+\beta+\gamma\right\}$

```
In[22]:= {p1[t_] = p₁, p2[t] = p₂ Cos[ω t], p3[t_] = p₃ Sin[ω t]};

In[23]:= {p[t_] = p1[t] + p2[t] + p3[t], p[t₀], Clear["@"], p[t₀]}

Out[23]= {p₁ + Cos[t ω] p₂ + Sin[t ω] p₃, p₁ + Cos[ω t₀] p₂ + Sin[ω t₀] p₃, Null, p[t₀]}
```

■ ΕΝΤΟΛΗ Γ5: ΚΑΘΑΡΙΣΜΑ ΕΝΤΟΛΗΣ ΜΗ ΦΟΡΤΩΜΕΝΟΥ ΠΑΚΕΤΟΥ

Remove[*ΕντολήΠακέτου*]

Καθαρίζει μια εντολή πακέτου της *Mathematica* που από απροσεξία *(κακώς!)* χρησιμοποιήθηκε πριν από την κλήση (το φόρτωμα) του ίδιου του πακέτου. (Ενεργεί επίσης ανάλογα με την εντολή **Clear** και σε συνηθισμένες περιπτώσεις μεταβλητών.) Ένα παράδειγμα δόθηκε προηγουμένως στην εντολή **Needs** (εντολή Γ2).

■ ΕΝΤΟΛΗ Γ6: ΠΡΟΣΗΜΟ ΑΡΙΘΜΟΥ

Sign[*Αριθμός*]

Δίνει το πρόσημο (sign) ενός αριθμού, δηλαδή +1, εάν ο αριθμός είναι θετικός, 0, εάν αυτός είναι μηδέν, και –1, εάν είναι αρνητικός. Παραδείγματα:

```
In[24]:= {Sign[-3], Sign[0], Sign[2], Sign[N[e]], Sign[2 - 10 + 3^2], Sign[Sin[100]]}

Out[24]= {-1, 0, 1, 1, 1, -1}
```

■ ΕΝΤΟΛΗ Γ7: ΕΠΙΛΟΓΕΣ ΕΝΤΟΛΗΣ

Options[*Εντολή*]

Δίνει τις επιλογές (options) της εντολής που αναφέρεται σαν όρισμα και τις αρχικές τιμές τους που έχει καθορίσει η *Mathematica*. Ακολουθούν παραδείγματα για τις επιλογές των εντολών **Integrate** (για ολοκλήρωση), **NDSolve** (για αριθμητική επίλυση διαφορικών εξισώσεων), **LaplaceTransform** (για μετασχηματισμό Laplace) και **FourierTransform** (για μετασχηματισμό Fourier):

```
In[25]:= Options[Integrate]

Out[25]= {Assumptions → {}, GenerateConditions → Automatic, PrincipalValue → False}

In[26]:= Options[NDSolve]

Out[26]= {AccuracyGoal → Automatic, Compiled → True, DifferenceOrder → Automatic,
          InterpolationPrecision → Automatic, MaxRelativeStepSize → 1/10, MaxSteps → Automatic,
          MaxStepSize → ∞, Method → Automatic, PrecisionGoal → Automatic, SolveDelayed → False,
          StartingStepSize → Automatic, StoppingTest → None, WorkingPrecision → 16}

In[27]:= Options[LaplaceTransform]

Out[27]= {Assumptions → {}, GenerateConditions → False,
          PrincipalValue → False, Analytic → True}
```

```
In[28]:= Options[FourierTransform]

Out[28]= {Assumptions → {}, GenerateConditions → False, FourierParameters → {0, 1}}
```

■ ΕΝΤΟΛΗ Γ8: ΚΑΘΟΡΙΣΜΟΣ ΕΠΙΛΟΓΩΝ ΕΝΤΟΛΗΣ

SetOptions[*Εντολή, Επιλογή-1 → Τιμή1, Επιλογή-2 → Τιμή2, . . .*]

Για την εντολή στο πρώτο όρισμα η εντολή αυτή **SetOptions** καθορίζει τιμές για μία ή περισσότερες από τις επιλογές της. (Τούτο έχει νόημα μόνο εάν οι τιμές που καθορίζονται διαφέρουν από τις τιμές που θέτει μόνη της από την αρχή η *Mathematica* για τις επιλογές αυτές.) Οι τιμές των επιλογών που καθορίζονται ισχύουν συνεχώς, μέχρι να ξαναααλλαχθούν με τη χρήση της ίδιας εντολής **SetOptions**.

Για παράδειγμα, η εντολή **FourierTransform** (για τον υπολογισμό του μετασχηματισμού Fourier μιας συναρτήσεως) έχει, όπως ήδη είδαμε λίγο πιο πάνω, τις εξής τρεις επιλογές:

```
In[29]:= Options[FourierTransform]

Out[29]= {Assumptions → {}, GenerateConditions → False, FourierParameters → {0, 1}}
```

Από αυτές η τρίτη **FourierParameters** είναι κρίσιμη για τον υπολογισμό του μετασχηματισμού Fourier επηρεάζοντας τον ορισμό του. Οι αρχικές τιμές των παραμέτρων αυτών (0 και 1) δεν είναι εκείνες που χρησιμοποιεί ο Πολιτικός Μηχανικός. Επομένως πρέπει να αλλάξουν. Ο Πολιτικός Μηχανικός θέλει οι παράμετροι αυτές να έχουν τις τιμές -1 και 1. Αντί όμως να το δηλώνει αυτό ρητά κάθε φορά που χρησιμοποιεί την εντολή **FourierTransform** μπορεί να το δηλώσει μια και καλή με την παρούσα εντολή **SetOptions**, συγκεκριμένα

```
In[30]:= SetOptions[FourierTransform, FourierParameters → {1, -1}]

Out[30]= {Assumptions → {}, GenerateConditions → False, FourierParameters → {1, -1}}
```

Τώρα πια η *Mathematica* έχει αλλάξει τις τιμές των παραμέτρων Fourier: **FourierParameters**

```
In[31]:= Options[FourierTransform]

Out[31]= {Assumptions → {}, GenerateConditions → False, FourierParameters → {1, -1}}
```

και επομένως θα χρησιμοποιεί συνεχώς το νέο ορισμό του μετασχηματισμού Fourier: αυτόν που επιθυμεί στ' αλήθεια ο Πολιτικός Μηχανικός.

■ ΕΝΤΟΛΗ Γ9: ΙΔΙΟΤΗΤΕΣ

Attributes[*Σύμβολο*]

Δίνει λίστα με τις ιδιότητες του συμβόλου. Παραδείγματα:

```
In[32]:= Attributes[π]

Out[32]= {Constant, Protected, ReadProtected}

In[33]:= Attributes[Plus]

Out[33]= {Flat, Listable, NumericFunction, OneIdentity, Orderless, Protected}
```

```
In[34]:= Attributes[Sin]
Out[34]= {Listable, NumericFunction, Protected}
```

```
In[35]:= Attributes[Plot]
Out[35]= {HoldAll, Protected}
```

■ ΕΝΤΟΛΗ Γ10: ΚΑΘΟΡΙΣΜΟΣ ΙΔΙΟΤΗΤΩΝ

SetAttributes[*Σύμβολο, Ιδιότητα*]

Προσθέτει την ιδιότητα στο δεύτερο όρισμα στο σύνολο των ιδιοτήτων του συμβόλου στο πρώτο όρισμα. (Αυτό το σύνολο μπορεί αρχικά να είναι και το κενό σύνολο. Επίσης αντί για μία ιδιότητα μπορούμε να έχουμε και λίστα ιδιοτήτων.) Παράδειγμα για τη δήλωση του συμβόλου ω σαν σταθεράς:

```
In[36]:= SetAttributes[ω, Constant]; Attributes[ω]
Out[36]= {Constant}
```

■ ΕΝΤΟΛΗ Γ11: ΑΠΑΛΟΙΦΗ ΙΔΙΟΤΗΤΩΝ

ClearAttributes[*Σύμβολο, Ιδιότητα*]

Απαλείφει την ιδιότητα στο δεύτερο όρισμα από το σύνολο των ιδιοτήτων του συμβόλου στο πρώτο όρισμα. (Αντί για μία ιδιότητα μπορούμε να έχουμε και λίστα ιδιοτήτων.) Παράδειγμα για την αφαίρεση της ιδιότητας του συμβόλου ω σαν σταθεράς που δηλώθηκε στην προηγούμενη εντολή:

```
In[37]:= ClearAttributes[ω, Constant]
```

■ ΕΝΤΟΛΗ Γ12: ΣΥΝΤΟΜΗ ΓΡΑΦΗ ΑΠΟΤΕΛΕΣΜΑΤΟΣ

Short[*ΠλήρηςΕντολή*]

Με την εντολή αυτή το αποτέλεσμα των υπολογισμών σε μια πλήρη εντολή της *Mathematica* γράφεται σε σύντομη μορφή μιας μόνο γραμμής με την παράλειψη πολλών ή και πάρα πολλών ενδιάμεσων όρων. Αυτό είναι χρήσιμο, μόνο εάν το αποτέλεσμα μιας εντολής είναι πολύ μακρύ και μας δυσκολεύει. Παράδειγμα για το ανάπτυγμα μιας δυνάμεως αθροίσματος με πολλούς όρους:

In[38]:= $(a + b)^{30}$ // Expand

Out[38]=
$$a^{30} + 30 a^{29} b + 435 a^{28} b^2 + 4060 a^{27} b^3 + 27405 a^{26} b^4 + 142506 a^{25} b^5 + 593775 a^{24} b^6 + 2035800 a^{23} b^7 + 5852925 a^{22} b^8 + 14307150 a^{21} b^9 + 30045015 a^{20} b^{10} + 54627300 a^{19} b^{11} + 86493225 a^{18} b^{12} + 119759850 a^{17} b^{13} + 145422675 a^{16} b^{14} + 155117520 a^{15} b^{15} + 145422675 a^{14} b^{16} + 119759850 a^{13} b^{17} + 86493225 a^{12} b^{18} + 54627300 a^{11} b^{19} + 30045015 a^{10} b^{20} + 14307150 a^9 b^{21} + 5852925 a^8 b^{22} + 2035800 a^7 b^{23} + 593775 a^6 b^{24} + 142506 a^5 b^{25} + 27405 a^4 b^{26} + 4060 a^3 b^{27} + 435 a^2 b^{28} + 30 a b^{29} + b^{30}$$

Με τη χρήση της εντολής **Short** γράφονται σαν αποτέλεσμα μόνο οι πρώτοι και οι τελευταίοι όροι του πιο πάνω αποτελέσματος. Πολλοί ενδιάμεσοι όροι (εδώ 26 όροι), ενώ υπολογίσθηκαν, έχουν παραλειφθεί:

In[39]:= **(a + b)^{30} // Expand // Short**

Out[39]//Short=

$a^{30} + 30\,a^{29}\,b + 435\,a^{28}\,b^2 + \ll 26 \gg + 30\,a\,b^{29} + b^{30}$

■ ΕΝΤΟΛΗ Γ13: ΠΛΗΡΟΦΟΡΙΕΣ ΓΙΑ ΕΝΤΟΛΗ

? *Εντολή*

Με την εντολή αυτή δίνονται πληροφορίες για μια εντολή της *Mathematica*. Παραδείγματα για τις εντολές **Simplify** (για απλοποίηση), **Integrate** (για ολοκλήρωση) και **DSolve** (για επίλυση διαφορικών εξισώσεων)

In[40]:= **? Simplify**

```
Simplify[expr] performs a sequence of algebraic
   transformations on expr, and returns the simplest form it finds.
   Simplify[expr, assum] does simplification using assumptions. More…
```

In[41]:= **? Integrate**

```
Integrate[f, x] gives the indefinite integral of f with respect to x.
   Integrate[f, {x, xmin, xmax}] gives the definite integral of f with respect
   to x from xmin to xmax. Integrate[f, {x, xmin, xmax}, {y, ymin, ymax}]
   gives a multiple definite integral of f with respect to x and y. More…
```

In[42]:= **? DSolve**

```
DSolve[eqn, y, x] solves a differential equation for the function y, with independent
   variable x. DSolve[{eqn1, eqn2, ... }, {y1, y2, ... }, x] solves a list of differential
   equations. DSolve[eqn, y, {x1, x2, ... }] solves a partial differential equation. More…
```

■ ΕΝΤΟΛΗ Γ14: ΠΛΗΡΕΙΣ ΠΛΗΡΟΦΟΡΙΕΣ ΓΙΑ ΕΝΤΟΛΗ

?? *Εντολή*

Με την εντολή αυτή δίνονται πάλι πληροφορίες για μια εντολή της *Mathematica*, τώρα όμως λίγο περισσότερες από την προηγούμενη εντολή ?. Συγκεκριμένα τώρα περιλαμβάνονται η περιγραφή της εντολής (πρώτη παράγραφος), οι ιδιότητές της (δεύτερη παράγραφος) καθώς και οι επιλογές της μαζί με τις αρχικές τιμές που τους δίνει η *Mathematica* (τρίτη παράγραφος). Παραδείγματα για τις εντολές **Simplify** (για απλοποίηση), **Integrate** (για ολοκλήρωση) και **DSolve** (για επίλυση διαφορικών εξισώσεων), όπως και προηγουμένως, και επίσης τώρα επιπλέον και **LaplaceTransform** (για τον υπολογισμό μετασχηματισμού Laplace)

In[43]:= **?? Simplify**

```
Simplify[expr] performs a sequence of algebraic
   transformations on expr, and returns the simplest form it finds.
   Simplify[expr, assum] does simplification using assumptions. More…

Attributes[Simplify] = {Protected}

Options[Simplify] = {ComplexityFunction → Automatic,
  TimeConstraint → 300, TransformationFunctions → Automatic, Trig → True}
```

```
In[44]:= ?? Integrate

        Integrate[f, x] gives the indefinite integral of f with respect to x.
           Integrate[f, {x, xmin, xmax}] gives the definite integral of f with respect
           to x from xmin to xmax. Integrate[f, {x, xmin, xmax}, {y, ymin, ymax}]
           gives a multiple definite integral of f with respect to x and y. More…

        Attributes[Integrate] = {Protected, ReadProtected}

        Options[Integrate] = {Assumptions → {}, GenerateConditions → Automatic, PrincipalValue → False}

In[45]:= ?? DSolve

        DSolve[eqn, y, x] solves a differential equation for the function y, with independent
           variable x. DSolve[{eqn1, eqn2, ... }, {y1, y2, ... }, x] solves a list of differential
           equations. DSolve[eqn, y, {x1, x2, ... }] solves a partial differential equation. More…

        Attributes[DSolve] = {Protected}

        Options[DSolve] = {DSolveConstants → C}

In[46]:= ?? LaplaceTransform

        LaplaceTransform[expr, t, s] gives the Laplace
           transform of expr. LaplaceTransform[expr, {t1, t2, ... }, {s1,
           s2, ... }] gives the multidimensional Laplace transform of expr. More…

        Attributes[LaplaceTransform] = {Protected, ReadProtected}

        Options[LaplaceTransform] =
         {Assumptions → {}, GenerateConditions → False, PrincipalValue → False, Analytic → True}
```

■ ΕΝΤΟΛΗ Γ15: ΑΠΟΤΕΛΕΣΜΑ ΣΕ ΜΟΡΦΗ ΕΞΟΔΟΥ

OutputForm[*Παράσταση*]

Με την εντολή αυτή τα αποτελέσματα των υπολογισμών (η έξοδος της *Mathematica)* εμφανίζονται στη συνηθισμένη τους μορφή εξόδου (που είναι η **StandardForm**), αλλά τώρα με τη χρήση μόνο χαρακτήρων από το πληκτρολόγιο, δηλαδή χωρίς καθόλου μα καθόλου ειδικά σύμβολα, όπως είναι π.χ. το ολόκληρωμα ή το βελάκι. Παραδείγματα:

```
In[47]:= Integrate[Sinh[Cosh[x]], x]
```

Out[47]= $\int \text{Sinh}[\text{Cosh}[x]]\, dx$

```
In[48]:= Integrate[Sinh[Cosh[x]], x] // OutputForm

Out[48]//OutputForm=
       Integrate[Sinh[Cosh[x]], x]
```

In[49]:= **DSolve[u''[t] + ω0^2 u[t] == 0, u[t], t]**

```
Out[49]= {{u[t] → C[1] Cos[t ω0] + C[2] Sin[t ω0]}}
```

In[50]:= **DSolve[u''[t] + ω0^2 u[t] == 0, u[t], t] // OutputForm**

```
Out[50]//OutputForm=
       {{u[t] -> C[1] Cos[t ω0] + C[2] Sin[t ω0]}}
```

■ ΕΝΤΟΛΗ Γ16: ΑΠΟΤΕΛΕΣΜΑ ΣΕ ΠΑΡΑΔΟΣΙΑΚΗ ΜΟΡΦΗ

TraditionalForm[*Παράσταση*]

Με την εντολή αυτή τα αποτελέσματα των υπολογισμών (η έξοδος της *Mathematica)* εμφανίζονται στην παραδοσιακή τους (traditional) μορφή, που πλησιάζει πολύ εκείνη της κλασικής στοιχειοθεσίας. Πρόκειται για μια όμορφη μορφή των αποτελεσμάτων που θυμίζει τυπωμένο κείμενο. Παραδείγματα:

```
In[51]:= Integrate[Sinh[Cosh[x]], x]

Out[51]= ∫Sinh[Cosh[x]] ⅆx

In[52]:= Integrate[Sinh[Cosh[x]], x] // TraditionalForm

Out[52]//TraditionalForm=
```

$$\int \sinh(\cosh(x))\, dx$$

```
In[53]:= DSolve[u''[t] + ω0^2 u[t] == 0, u[t], t]

Out[53]= {{u[t] → C[1] Cos[t ω0] + C[2] Sin[t ω0]}}

In[54]:= DSolve[u''[t] + ω0^2 u[t] == 0, u[t], t] // TraditionalForm

Out[54]//TraditionalForm=
```

$$\{\{u(t) \to c_1 \cos(t\,\omega 0) + c_2 \sin(t\,\omega 0)\}\}$$

■ ΕΝΤΟΛΗ Γ17: ΑΠΟΤΕΛΕΣΜΑ ΣΕ ΠΛΗΡΗ ΜΟΡΦΗ

FullForm[*Παράσταση*]

Με την εντολή αυτή τα αποτελέσματα των υπολογισμών (η έξοδος της *Mathematica)* εμφανίζονται στην πλήρη τους μορφή χωρίς καν τα σύμβολα των αριθμητικών πράξεων ή τα βελάκια. Παραδείγματα:

```
In[55]:= (a + b)^2 // FullForm

Out[55]//FullForm=
        Power[Plus[a, b], 2]

In[56]:= DSolve[u''[t] + ω0^2 u[t] == 0, u[t], t]

Out[56]= {{u[t] → C[1] Cos[t ω0] + C[2] Sin[t ω0]}}

In[57]:= DSolve[u''[t] + ω0^2 u[t] == 0, u[t], t] // FullForm

Out[57]//FullForm=
        List[List[Rule[u[t], Plus[
            Times[C[1], Cos[Times[t, \[Omega]0]]], Times[C[2], Sin[Times[t, \[Omega]0]]]]]]]
```

■ ΕΝΤΟΛΗ Γ18: ΒΑΣΙΚΟΣ ΤΕΛΕΣΤΗΣ Ή ΒΑΣΙΚΗ ΙΔΙΟΤΗΤΑ

Head[*ΠαράστασηΉΣύμβολοΉΑριθμός*]

Δίνει το βασικό τελεστή ή τη βασική ιδιότητα (το σημαντικό τμήμα, την "κεφαλή") μιας παραστάσεως, ενός συμβόλου, ενός αριθμού, κλπ. Παραδείγματα με αριθμούς (πρώτη γραμμή), απλές πράξεις (δεύτερη γραμμή), λίστες και μητρώα (τρίτη γραμμή) και συναρτήσεις (τέταρτη γραμμή):

```
In[58]:= {Head[3], Head[-5], Head[2 / 3], Head[5.2], Head[1 + 2 ⅈ], Head[a], Head[π], Head[N[π]]}
Out[58]= {Integer, Integer, Rational, Real, Complex, Symbol, Symbol, Real}
In[59]:= {Head[a + b], Head[a - b], Head[a b], Head[a / b], Head[a^2], Head[a^2 b^2]}
Out[59]= {Plus, Plus, Times, Times, Power, Times}
In[60]:= {Head[{a, b, c}], Head[{{a, b}, {c, d}}]}
Out[60]= {List, List}
In[61]:= {Head[Sqrt[a]], Head[Sin[x]], Head[Cosh[x + y]], Head[u[t]], Head[u''[t]]}
Out[61]= {Power, Sin, Cosh, u, u″}
```

■ ΕΝΤΟΛΗ Γ19: ΚΑΘΑΡΗ ΣΥΝΑΡΤΗΣΗ

Function[*Μεταβλητή, ΣυνάρτησηΤηςΜεταβλητής*]

Function[*ΛίσταΜεταβλητών, ΣυνάρτησηΤωνΜεταβλητών*]

Ορίζει μια καθαρή συνάρτηση, δηλαδή ορίζει μια συνάρτηση χωρίς να περιλαμβάνεται κατευθείαν η μεταβλητή της ή οι μεταβλητές της. Εντούτοις αυτή η καθαρή συνάρτηση μπορεί να χρησιμοποιηθεί σε οποιουσδήποτε υπολογισμούς θέτοντας τη μεταβλητή (ή τις μεταβλητές) με σύμβολο (με σύμβολα) ή με αριθμό (αριθμούς) αμέσως μετά τη συνάρτηση μέσα σε αγκύλες ως συνήθως. Παραδείγματα:

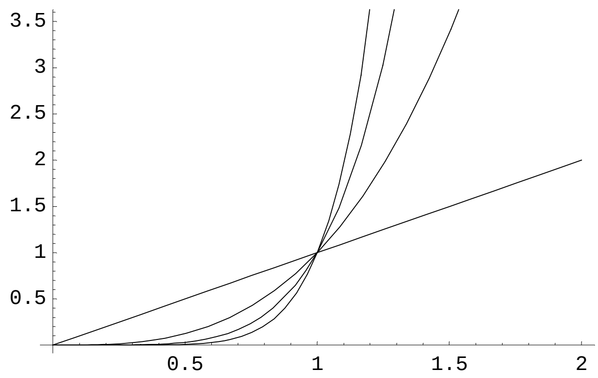
```
In[62]:= Function[z, 3 z^3 + 2 z^2 + z]
Out[62]= Function[z, 3 z^3 + 2 z^2 + z]
In[63]:= {Function[z, 3 z^3 + 2 z^2 + z][1], Function[z, 3 z^3 + 2 z^2 + z][a]}
Out[63]= {6, a + 2 a^2 + 3 a^3}
In[64]:= Function[t, C1 Cos[ω τ] + C2 Sin[ω τ]]
Out[64]= Function[t, C1 Cos[ω τ] + C2 Sin[ω τ]]
In[65]:= Function[τ, C1 Cos[ω τ] + C2 Sin[ω τ]][t]
Out[65]= C1 Cos[t ω] + C2 Sin[t ω]
```

■ ΕΝΤΟΛΗ Γ20: ΑΝΑΓΚΑΣΤΙΚΟΣ ΥΠΟΛΟΓΙΣΜΟΣ

Evaluate[*Παράσταση*]

Η εντολή αυτή υποχρεώνει τη *Mathematica* να κάνει τον υπολογισμό της παραστάσεως στο όρισμά της ακόμη και αν η παράσταση αυτή έχει τεθεί σαν όρισμα άλλης εντολής της οποίας οι ιδιότητες καθορίζουν ότι τα ορίσματά της δε θα υπολογισθούν. Η εντολή αυτή είναι χρήσιμη βασικά με την εντολή **Plot** για τη σχεδίαση λίστας συναρτήσεων στο ίδιο σχήμα με τη χρήση της εντολής **Table** σαν όρισμα της **Plot** για τον καθορισμό των συναρτήσεων. Ένα παράδειγμα με αναγκαία την εντολή **Evaluate**:

```
In[66]:= Plot[Evaluate[Table[x^n, {n, 1, 7, 2}]], {x, 0, 2}];
```



■ ΕΝΤΟΛΗ Γ21: ΕΦΑΡΜΟΓΗ ΣΥΝΑΡΤΗΣΕΩΣ ΣΕ ΛΙΣΤΑ Ή ΠΑΡΑΣΤΑΣΗ

Map[*Συνάρτηση, Λίστα΄ΗΠαράσταση*]

Εφαρμόζει τη συνάρτηση στο πρώτο όρισμά της σε κάθε ένα στοιχείο της λίστας ή της παραστάσεως στο δεύτερο όρισμά της. Παραδείγματα:

Έστω πως θέλουμε να εφαρμόσουμε την εκθετική συνάρτηση στα στοιχεία μιας λίστας

```
In[67]:= list = {a, b, c, d, e, f}

Out[67]= {a, b, c, d, e, f}
```

Προς το σκοπό αυτό ο πιο εύχρηστος τρόπος φαίνεται να είναι η χρήση των εντολών **Exp** για την εκθετική συνάρτηση και **Map** για την εφαρμογή της **Exp** στην πιο πάνω λίστα **list**:

```
In[68]:= Map[Exp, list]
```

Out[68]= $\{e^a, e^b, e^c, e^d, e^e, e^f\}$

Γενικότερα μπορούμε να εργασθούμε με οποιαδήποτε καθαρή συνάρτηση **f**

```
In[69]:= Map[f, list]

Out[69]= {f[a], f[b], f[c], f[d], f[e], f[f]}
```

In[70]:= **Map[Function[x, $c_1 x^2 + c_2$], list]**

Out[70]= $\{a^2 c_1 + c_2, b^2 c_1 + c_2, c^2 c_1 + c_2, d^2 c_1 + c_2, e^2 c_1 + c_2, f^2 c_1 + c_2\}$

Το δεύτερο όρισμα δεν είναι αναγκαστικά λίστα. Δίνουμε ένα παράδειγμα (με τετραγωνική ρίζα) όπου είναι άθροισμα και παρατηρούμε τη διαφορά του αποτελέσματος από την απλή εφαρμογή της τετραγωνικής ρίζας στο άθροισμα:

```
In[71]:= {Sqrt[a + b + c + d + e + f], Map[Sqrt, a + b + c + d + e + f]}
```

Out[71]= $\{\sqrt{a+b+c+d+e+f}, \sqrt{a}+\sqrt{b}+\sqrt{c}+\sqrt{d}+\sqrt{e}+\sqrt{f}\}$

■ *Notebook E1*

ΠΡΑΞΕΙΣ, ΣΤΑΘΕΡΕΣ, ΑΡΙΘΜΗΤΙΚΟΙ ΥΠΟΛΟΓΙΣΜΟΙ

ΠΡΑΞΕΙΣ, ΣΤΑΘΕΡΕΣ ΚΑΙ 2 ΕΝΤΟΛΕΣ: Π3. N, Π4. Chop

■ ΠΡΑΞΕΙΣ Π1: ΣΤΟΙΧΕΙΩΔΕΙΣ ΑΡΙΘΜΗΤΙΚΕΣ ΠΡΑΞΕΙΣ

Οι στοιχειώδεις αριθμητικές πράξεις εκτελούνται με τα γνωστά σύμβολα: + (συν) για την πρόσθεση, – (πλην) για την αφαίρεση, κενό ή * (αυτό είναι το επί στη *Mathematica*) για τον πολλαπλασιασμό και / για τη διαίρεση. Επίσης ˆ ή **Ctrl 6** ή **Ctrl** ˆ για την ύψωση σε δύναμη. Στις αριθμητικές αλλά και στις αλγεβρικές πράξεις απαιτείται η χρήση παρενθέσεων *(όχι αγκυλών!)* όσες φορές δε μας αρέσει η καθορισμένη προτεραιότητα των πράξεων, αλλά επιθυμούμε να την καθορίσουμε εμείς. Όσες φορές θέλουμε τα αποτελέσματα σε μορφή δεκαδικών αριθμών, μπορούμε να χρησιμοποιούμε είτε την εντολή **N** παρακάτω είτε την τελεία σε έναν τουλάχιστον από τους αριθμούς, δηλαδή ένα δεκαδικό αριθμό. Φυσικά οι πράξεις αυτές μπορούν να γίνονται και σε αλγεβρικά σύμβολα. Παραδείγματα:

```
In[1]:= {1 + 1, 2 3, 2 * 3, 2 3 == 2 * 3, 2 / (3 + 4), 5 * 6, 10 / 20, 10 / (2 * 3) == 10 / 2 3}
```

Out[1]= $\left\{2, 6, 6, \text{True}, \frac{2}{7}, 30, \frac{1}{2}, \text{False}\right\}$

```
In[2]:= {1 / 3, 1. / 3, 1 / 3., 1. / 3., 1 / 3 == 1. / 3 == 1 / 3. == 1. / 3.}
```

Out[2]= $\left\{\frac{1}{3}, 0.333333, 0.333333, 0.333333, \text{True}\right\}$

In[3]:= $\{3\verb|^|4,\ 3^4,\ 3\verb|^|4 == 3^4,\ a^2\,a^4,\ 2\ 3^4,\ (2\ 3)^4\}$

Out[3]= $\{81, 81, \text{True}, a^6, 162, 1296\}$

■ ΣΤΑΘΕΡΕΣ Π2: ΕΙΔΙΚΕΣ ΣΤΑΘΕΡΕΣ

π ή **Pi** (για τον αριθμό *π*. Από το πληκτρολόγιο με **Esc p Esc**, με ένα μόνο p.)

e ή **E** (για τη βάση *e* των φυσικών λογαρίθμων. Από το πληκτρολόγιο με **Esc ee Esc**.)

i ή **I** (για τη φανταστική μονάδα *i* με $i^2 = -1$**.** Από το πληκτρολόγιο με **Esc ii Esc**.)

∞ ή **Infinity** (για το άπειρο. Από το πληκτρολόγιο με **Esc inf Esc**.)

° ή **Degree** (για το πηλίκο π / 180: από μοίρες σε ακτίνια. Από το πληκτρολόγιο με **Esc deg Esc**.)

EulerGamma (για τη σταθερά *γ* του Euler. Η *Mathematica* δεν έχει ειδικό σύμβολο.) Παραδείγματα:

In[4]:= $\{\pi == \text{Pi},\ e == \text{E},\ i == \text{I},\ i^2,\ i^4,\ \infty,\ \infty == \text{Infinity},\ \infty + \infty,\ \infty^\infty,\ \infty^{-\infty},\ 10\,\infty,\ 10\ \text{Infinity}\}$

```
Out[4]= {True, True, True, -1, 1, ∞, True, ∞, ComplexInfinity, 0, ∞, ∞}
```

```
In[5]:= {180, 180 °, 180 Degree, 180 ° == 180 Degree, 60 ° + 30 °, γ = EulerGamma, N[γ, 30]}
```

```
Out[5]= {180, 180 °, 180 °, True, 90 °, EulerGamma, 0.577215664901532860606512090082}
```

■ ΕΝΤΟΛΗ Π3: ΑΡΙΘΜΗΤΙΚΟΣ ΥΠΟΛΟΓΙΣΜΟΣ

N[*Παράσταση*]

N[*Παράσταση, Ακρίβεια*]

Η πρώτη μορφή της εντολής αυτής **N** υπολογίζει αριθμητικά μια παράσταση με την ακρίβεια που έχει καθορισθεί στη *Mathematica*: έξι σημαντικά ψηφία, σε όσο βαθμό αυτό είναι δυνατόν στο αποτέλεσμα. Στη δεύτερη μορφή της ίδιας εντολής καθορίζεται σαν δεύτερο όρισμα και η επιθυμητή ακρίβεια του αποτελέσματος. Σχεδόν πάντοτε τα αριθμητικά αποτελέσματα της *Mathematica* είναι ακριβή σε όλα τα ψηφία που παρουσιάζονται στην οθόνη, απλά επειδή οι υπολογισμοί εσωτερικά στον υπολογιστή γίνονται με πολύ μεγαλύτερη ακρίβεια. *Παρατήρηση:* Πάρα πολύ συχνά η εντολή **N** στην πρώτη μορφή της γράφεται μετά την παράσταση που θέλουμε να υπολογισθεί. Παραδείγματα:

```
In[6]:= {π, N[π], π // N, N[π] == π // N, N[π, 50]}
Out[6]= {π, 3.14159, 3.14159, True, 3.1415926535897932384626433832795028841971693993751}

In[7]:= {ⅇ, E, N[ⅇ, 70]}
Out[7]= {ⅇ, E, 2.718281828459045235360287471352662497757247093699959574966967627724077}

In[8]:= {π, ⅇ, ⅈ, Infinity, Degree, 1 / Degree, EulerGamma} // N
Out[8]= {3.14159, 2.71828, 0. + 1. ⅈ, ∞, 0.0174533, 57.2958, 0.577216}

In[9]:= {a / 3 + b / 7 == (a / 3) + (b / 7), a / 3 + b / 7 // N,
         N[a / 3 + b / 7], (a / 3 + b / 7 // N) == N[a / 3 + b / 7]}
Out[9]= {True, 0.333333 a + 0.142857 b, 0.333333 a + 0.142857 b, True}
```

■ ΕΝΤΟΛΗ Π4: ΜΗΔΕΝΙΣΜΟΣ ΠΟΛΥ ΜΙΚΡΗΣ ΠΟΣΟΤΗΤΑΣ

Chop[*Παράσταση*]

Chop[*Παράσταση, ΟΜεγαλύτεροςΠολύΜικρόςΑριθμόςΠουΘαΜηδενισθεί*]

Μηδενίζει κάθε πολύ μικρό αριθμό ε που υπάρχει σε μια αριθμητική παράσταση, η οποία μπορεί να περιέχει και σύμβολα. Αυτοί οι πολύ μικροί αριθμοί ε συνήθως προκύπτουν από αριθμητικά σφάλματα στρογγυλεύσεως στον υπολογιστή στην περίπτωση εκτελέσεως πράξεων με δεκαδικούς αριθμούς. Μπορεί όμως να οφείλονται και σε οποιαδήποτε μη ακριβή αριθμητική μέθοδο. Στη δεύτερη μορφή της εντολής **Chop** καθορίζεται και ο πολύ μικρός αριθμός ε_0, για τον οποίο τόσο αυτός ο ίδιος όσο και όλοι οι μικρότεροί του αριθμοί (κατ' απόλυτο τιμή εννοείται) θα μηδενισθούν. Παραδείγματα:

```
In[10]:= {a + 10.^-20 b, Chop[a + 10.^-20 b], q = N[ⅇ]^(2 π ⅈ), Chop[q], q // Chop, Chop[q] == q // Chop }
Out[10]= {a + 1. × 10^-20 b, a, 1. - 2.44921 × 10^-16 ⅈ, 1., 1., True}

In[11]:= {q, Chop[q], Chop[q, 10^-10], Chop[q, 10^-15], Chop[q, 10^-20], Chop[q, 0.00000001]}
Out[11]= {1. - 2.44921 × 10^-16 ⅈ, 1., 1., 1., 1. - 2.44921 × 10^-16 ⅈ, 1.}
```

■ *Notebook E2*
ΣΥΝΑΡΤΗΣΕΙΣ *(ΠΑΡΟΥΣΙΑΖΟΝΤΑΙ 43 ΣΥΝΟΛΙΚΑ ΣΥΝΑΡΤΗΣΕΙΣ)*

■ ΣΥΝΑΡΤΗΣΗ Σ1: ΑΠΟΛΥΤΟΣ ΤΙΜΗ

Abs[*Μεταβλητή*]

Υπολογίζει την απόλυτο τιμή είτε για πραγματικούς είτε για μιγαδικούς αριθμούς. Παραδείγματα:

```
In[1]:= {Abs[-3], Abs[0], Abs[34], Abs[2 + ⅈ], Abs[2 - ⅈ], Abs[-a] == Abs[a]}
Out[1]= {3, 0, 34, √5, √5, True}
```

■ ΣΥΝΑΡΤΗΣΗ Σ2: ΤΕΤΡΑΓΩΝΙΚΗ ΡΙΖΑ

Sqrt[*Μεταβλητή*]

Υπολογίζει την τετραγωνική ρίζα είτε για πραγματικούς είτε για μιγαδικούς αριθμούς. Παραδείγματα:

```
In[2]:= {Sqrt[4], Sqrt[10.], Sqrt[a^2], Sqrt[a^2] // PowerExpand, Sqrt[a b] // PowerExpand}
Out[2]= {2, 3.16228, √(a^2), a, √a √b}
```

■ ΣΥΝΑΡΤΗΣΕΙΣ Σ3: ΕΚΘΕΤΙΚΗ ΚΑΙ ΛΟΓΑΡΙΘΜΙΚΗ ΣΥΝΑΡΤΗΣΗ

Exp[*Μεταβλητή*] , Log[*Μεταβλητή*], Log[*Βάση, Μεταβλητή*]

Οι δύο πρώτες συναρτήσεις υπολογίζουν την εκθετική συνάρτηση (exp) καθώς και τη λογαριθμική συνάρτηση (το φυσικό λογάριθμο ln) αντίστοιχα. Πρόκειται για δύο αντίστροφες συναρτήσεις. Η τρίτη συνάρτηση υπολογίζει το λογάριθμο με κάποια άλλη βάση που δηλώνεται με το πρώτο όρισμά της. Παραδείγματα:

```
In[3]:= {Exp[x] == ⅇ^x, Exp[Log[x]], Exp[Log[x]] == x, Exp[a + b] == Exp[a] Exp[b], Exp[1.]}
Out[3]= {True, x, True, True, 2.71828}

In[4]:= {Log[2, 50.], Log[ⅇ, 50.], Log[10, 50.], Log[10, 50.] == Log[50.] / Log[10.]}
Out[4]= {5.64386, 3.91202, 1.69897, True}
```

■ ΣΥΝΑΡΤΗΣΕΙΣ Σ4: ΤΡΙΓΩΝΟΜΕΤΡΙΚΕΣ / ΥΠΕΡΒΟΛΙΚΕΣ ΣΥΝΑΡΤΗΣΕΙΣ

Sin[*Μεταβλητή*] , Cos[*Μεταβλητή*] , Tan[*Μεταβλητή*] , Cot[*Μεταβλητή*] ,

Sec[*Μεταβλητή*] , Csc[*Μεταβλητή*]

Υπολογίζουν το ημίτονο (sin), το συνημίτονο (cos), την εφαπτομένη (tan), τη συνεφαπτομένη (cot), την τέμνουσα (sec) και τη συντέμνουσα (csc) αντίστοιχα.

ArcSin[*Μεταβλητή*] , ArcCos[*Μεταβλητή*] , ArcTan[*Μεταβλητή*] , ArcCot[*Μεταβλητή*] , ArcSec[*Μεταβλητή*] , ArcCsc[*Μεταβλητή*]

Υπολογίζουν τις αντίστροφες συναρτήσεις των τριγωνομετρικών συναρτήσεων, συγκεκριμένα του ημιτόνου (τόξο ημιτόνου: arcsin), του συνημιτόνου (τόξο συνημιτόνου: arccos), της εφαπτομένης (τόξο εφαπτομένης arctan), της συνεφαπτομένης (τόξο συνεφαπτομένης: arccot), της τέμνουσας (τόξο τέμνουσας: arcsec) και της συντέμνουσας (τόξο συντέμνουσας: arccsc) αντίστοιχα.

Sinh[*Μεταβλητή*] , Cosh[*Μεταβλητή*] , Tanh[*Μεταβλητή*] , Coth[*Μεταβλητή*] , Sech[*Μεταβλητή*] , Csch[*Μεταβλητή*]

Υπολογίζουν το υπερβολικό ημίτονο (sinh), το υπερβολικό συνημίτονο (cosh), την υπερβολική εφαπτομένη (tanh), την υπερβολική συνεφαπτομένη (coth), την υπερβολική τέμνουσα (sech) και την υπερβολική συντέμνουσα (csch) αντίστοιχα.

ArcSinh[*Μεταβλητή*] , ArcCosh[*Μεταβλητή*] , ArcTanh[*Μεταβλητή*] , ArcCoth[*Μεταβλητή*] , ArcSech[*Μεταβλητή*] , ArcCsch[*Μεταβλητή*]

Υπολογίζουν τις αντίστροφες συναρτήσεις των πιο πάνω υπερβολικών συναρτήσεων αντίστοιχα. Παραδείγματα:

```
In[5]:= {Sin[x], Cos[x], Tan[x], Cot[x],
          Sinh[x], Cosh[x], Tanh[x], Coth[x], Exp[x], ⅇ^x} /. x → 1.

Out[5]= {0.841471, 0.540302, 1.55741, 0.642093,
         1.1752, 1.54308, 0.761594, 1.31304, 2.71828, 2.71828}

In[6]:= {ArcSin[x], ArcCos[x], ArcTan[x], ArcCot[x],
          ArcSinh[x], ArcCosh[x], ArcTanh[x], ArcCoth[x], Log[x], Log[ⅇ, x]} /. x → 0.5

Out[6]= {0.523599, 1.0472, 0.463648, 1.10715, 0.481212,
         0. + 1.0472 ⅈ, 0.549306, 0.549306 - 1.5708 ⅈ, -0.693147, -0.693147}

In[7]:= {Sin[ArcSin[x]], Sinh[ArcSinh[x]], Cos[ArcCos[x]] == x, Tanh[ArcTanh[x]] == x}

Out[7]= {x, x, True, True}

In[8]:= {Cos[x]^2 + Sin[x]^2 == 1, Cosh[x]^2 - Sinh[x]^2 == 1, Cosh[2 x] == 2 Cosh[x]^2 - 1} // Simplify

Out[8]= {True, True, True}
```

■ ΣΥΝΑΡΤΗΣΗ Σ5: ΠΑΡΑΓΟΝΤΙΚΟ

Factorial[*ΜηΑρνητικόςΑκέραιος*] ή ***ΜηΑρνητικόςΑκέραιος*!** (με θαυμαστικό στο τέλος!)

Υπολογίζει το παραγοντικό μη αρνητικού ακέραιου αριθμού. Η δεύτερη μορφή είναι αυτή που σχεδόν πάντα χρησιμοποιείται. (Στην πράξη τη χρησιμοποιούμε μόνο για ακέραιους αριθμούς.) Παραδείγματα:

```
In[9]:= {Factorial[30], 30 !, Factorial[10] == 10 !}

Out[9]= {265252859812191058636308480000000, 265252859812191058636308480000000, True}

In[10]:= equation = n ! == Product[k, {k, 1, n}]

Out[10]= True
```

■ ΣΥΝΑΡΤΗΣΗ Σ6: ΣΥΝΑΡΤΗΣΗ ΓΑΜΜΑ

Gamma[*Μεταβλητή*]

Υπολογίζει την αρκετά γνωστή στα μαθηματικά συνάρτηση γάμμα $\Gamma(x)$, η οποία σχετίζεται και με το παραγοντικό $n!$ για θετικές ακέραιες τιμές n της μεταβλητής x. Παραδείγματα:

```
In[11]:= {Gamma[0.5], Gamma[1], Gamma[1.5], Gamma[2], N[Gamma[-1 / 2], 50]}

Out[11]= {1.77245, 1, 0.886227, 1, -3.5449077018110320545963349666822903655950989122448}

In[12]:= {id = Gamma[n + 1] == n Gamma[n], FullSimplify[id], Gamma[n + 1] == n! // FullSimplify}

Out[12]= {Gamma[1 + n] == n Gamma[n], True, True}
```

■ ΣΥΝΑΡΤΗΣΗ Σ7: ΒΗΜΑΤΙΚΗ ΣΥΝΑΡΤΗΣΗ ΤΟΥ HEAVISIDE

UnitStep[*Μεταβλητή*]

Υπολογίζει τη μοναδιαία βηματική συνάρτηση του Heaviside $H(x)$. Παραδείγματα:

```
In[13]:= {UnitStep[-1], UnitStep[0], UnitStep[1]}

Out[13]= {0, 1, 1}
```

■ ΣΥΝΑΡΤΗΣΗ Σ8: ΩΣΤΙΚΗ ΣΥΝΑΡΤΗΣΗ ΔΕΛΤΑ ΤΟΥ DIRAC

DiracDelta[*Μεταβλητή*]

Υπολογίζει την ωστική (ή κρουστική) συνάρτηση δέλτα του Dirac $\delta(x)$. Παραδείγματα:

```
In[14]:= {DiracDelta[-1], DiracDelta[0], DiracDelta[1], D[UnitStep[x], x] }

Out[14]= {0, DiracDelta[0], 0, DiracDelta[x]}
```

■ ΣΥΝΑΡΤΗΣΕΙΣ Σ9: ΣΥΝΑΡΤΗΣΗ ΣΦΑΛΜΑΤΟΣ ΚΑΙ ΣΥΜΠΛΗΡΩΜΑΤΙΚΗ ΣΥΝΑΡΤΗΣΗ ΣΦΑΛΜΑΤΟΣ

Erf[*Μεταβλητή*] , Erfc[*Μεταβλητή*]

Υπολογίζουν τη συνάρτηση σφάλματος, error function: $\mathrm{erf}(x)$ καθώς και τη συμπληρωματική συνάρτηση σφάλματος, complementary error function: $\mathrm{erfc}(x)$ αντίστοιχα. Πρόκειται για δύο συναρτήσεις με άθροισμα πάντοτε τη μονάδα: $\mathrm{erf}(x)+\mathrm{erfc}(x)=1$. Παραδείγματα:

```
In[15]:= {Erf[x] + Erfc[x], Erfc[x] == 1 - Erf[x]} // FullSimplify

Out[15]= {1, True}

In[16]:= {Erf[0], Erfc[0], Erf[3.], Erfc[3.], N[Erfc[10], 40]}

Out[16]= {0, 1, 0.999978, 0.0000220905, 2.088487583762544757000786294957788611561× 10^-45}
```

■ ΣΥΝΑΡΤΗΣΕΙΣ Σ10: ΟΡΘΟΓΩΝΙΑ ΠΟΛΥΩΝΥΜΑ (5 πολυώνυμα)

LegendreP[*Βαθμός, Μεταβλητή*] (πολυώνυμο Legendre $P_n(x)$)

ChebyshevT[*Βαθμός, Μεταβλητή*] (πολυώνυμο Chebyshev πρώτου είδους $T_n(x)$)

ChebyshevU[*Βαθμός, Μεταβλητή*] (πολυώνυμο Chebyshev δευτέρου είδους $U_n(x)$)

LaguerreL[*Βαθμός, Μεταβλητή*] (πολυώνυμο Laguerre $L_n(x)$)

HermiteH[*Βαθμός, Μεταβλητή*] (πολυώνυμο Hermite $H_n(x)$)

Υπολογίζουν τα πιο πάνω πέντε κλασικά ορθογώνια πολυώνυμα, το καθένα από τα οποία δημιουργεί και ένα ολόκληρο σύνολο (ή σύστημα) άπειρων ορθογωνίων πολυωνύμων. Παραδείγματα:

```
In[17]:= Table[LegendreP[n, x], {n, 0, 5}] // Simplify
```

Out[17]= $\left\{1, x, \frac{1}{2}(-1+3x^2), \frac{1}{2}x(-3+5x^2), \frac{1}{8}(3-30x^2+35x^4), \frac{1}{8}x(15-70x^2+63x^4)\right\}$

```
In[18]:= Table[Integrate[LegendreP[m, x] LegendreP[n, x], {x, -1, 1}], {m, 0, 4}, {n, 0, 4}]
```

Out[18]= $\left\{\{2, 0, 0, 0, 0\}, \left\{0, \frac{2}{3}, 0, 0, 0\right\}, \left\{0, 0, \frac{2}{5}, 0, 0\right\}, \left\{0, 0, 0, \frac{2}{7}, 0\right\}, \left\{0, 0, 0, 0, \frac{2}{9}\right\}\right\}$

```
In[19]:= {n = 4, LegendreP[n, x], ChebyshevT[n, x],
          ChebyshevU[n, x], LaguerreL[n, x], HermiteH[n, x]} // Simplify
```

Out[19]= $\left\{4, \frac{1}{8}(3-30x^2+35x^4), 1-8x^2+8x^4, 1-12x^2+16x^4, 1-4x+3x^2-\frac{2x^3}{3}+\frac{x^4}{24}, 4(3-12x^2+4x^4)\right\}$

```
In[20]:= Table[ChebyshevT[n, x] == Cos[n ArcCos[x]], {n, 0, 10}] // Simplify
```

```
Out[20]= {True, True, True, True, True, True, True, True, True, True, True}
```

■ ΣΥΝΑΡΤΗΣΕΙΣ Σ11: ΣΥΝΑΡΤΗΣΕΙΣ BESSEL (4 συναρτήσεις)

BesselJ[*Τάξη, Μεταβλητή*] (συνάρτηση Bessel $J_\nu(x)$ πρώτου είδους)

BesselY[*Τάξη, Μεταβλητή*] (συνάρτηση Bessel $Y_\nu(x)$ δευτέρου είδους)

BesselI[*Τάξη, Μεταβλητή*] (τροποποιημένη συνάρτηση Bessel $I_\nu(x)$ πρώτου είδους)

BesselK[*Τάξη, Μεταβλητή*] (τροποποιημένη συνάρτηση Bessel $K_\nu(x)$ δευτέρου είδους)

Υπολογίζουν τις τέσσερις πιο πάνω συναρτήσεις Bessel. Παραδείγματα:

```
In[21]:= {BesselJ[1, 3.], BesselY[1, 3.], BesselI[1, 3.], BesselK[1, 3.]}
```

```
Out[21]= {0.339059, 0.324674, 3.95337, 0.0401564}
```

```
In[22]:= {D[BesselJ[1, x], x], D[BesselY[1, x], x]}
```

Out[22]= $\left\{\frac{1}{2}(\text{BesselJ}[0, x]-\text{BesselJ}[2, x]), \frac{1}{2}(\text{BesselY}[0, x]-\text{BesselY}[2, x])\right\}$

```
In[23]:= Series[BesselJ[0, x], {x, 0, 15}]
```

Out[23]= $1-\frac{x^2}{4}+\frac{x^4}{64}-\frac{x^6}{2304}+\frac{x^8}{147456}-\frac{x^{10}}{14745600}+\frac{x^{12}}{2123366400}-\frac{x^{14}}{416179814400}+O[x]^{16}$

■ *Notebook E3*

ΕΝΤΟΛΕΣ ΚΥΡΙΩΣ ΓΙΑ ΑΛΓΕΒΡΑ

16 ΕΝΤΟΛΕΣ: **A1. Numerator, A2. Denominator, A3. Factor, A4. Expand, A5. ExpandAll, A6. PowerExpand, A7. Eliminate, A8. Apart, A9. Together, A10. Exponent, A11. Variables, A12. Collect, A13. Coefficient, A14. CoefficientList, A15. Simplify, A16. FullSimplify**

■ ΕΝΤΟΛΗ A1: ΑΡΙΘΜΗΤΗΣ ΚΛΑΣΜΑΤΟΣ

Numerator[*Κλάσμα*]

Η εντολή αυτή υπολογίζει τον αριθμητή ενός κλάσματος. Παράδειγμα στην αμέσως επόμενη εντολή **Denominator**.

■ ΕΝΤΟΛΗ A2: ΠΑΡΟΝΟΜΑΣΤΗΣ ΚΛΑΣΜΑΤΟΣ

Denominator[*Κλάσμα*]

Η εντολή αυτή υπολογίζει τον παρονομαστή ενός κλάσματος. Παράδειγμα και για τις δύο ταυτόχρονα εντολές **Numerator** και **Denominator**:

```
In[1]:= {f = (a x^2 + b ) / (c x^10 + e Sin[x]), n = Numerator[f], d = Denominator[f], f == n / d, n == f d}
```

Out[1]= $\left\{\frac{b + a x^2}{c x^{10} + e \,\mathrm{Sin}[x]},\ b + a x^2,\ c x^{10} + e \,\mathrm{Sin}[x],\ \mathrm{True},\ \mathrm{True}\right\}$

■ ΕΝΤΟΛΗ A3: ΠΑΡΑΓΟΝΤΟΠΟΙΗΣΗ

Factor[*Παράσταση*]

Factor[*Παράσταση,* GaussianIntegers → True]

Factor[*Παράσταση,* Extension → *Ρίζα*]

Παραγοντοποιεί μια παράσταση, δηλαδή την αναλύει σε γινόμενα παραγόντων. Επιπλέον με την επιλογή **GaussianIntegers → True** η παραγοντοποίηση χρησιμοποιεί και τη φανταστική μονάδα *i*. Επίσης με την επιλογή **Extension → *Ρίζα*** η παραγοντοποίηση χρησιμοποιεί και τη ρίζα που δηλώνεται. Παραδείγματα:

```
In[2]:= {Factor[x^2 - a^2], Factor[x^4 - a^4], Factor[x^6 - a^6]}
```

Out[2]= $\{-(a - x)\ (a + x),\ -(a - x)\ (a + x)\ (a^2 + x^2),\ -(a - x)\ (a + x)\ (a^2 - a x + x^2)\ (a^2 + a x + x^2)\}$

```
In[3]:= {Factor[x^4 + 25], Factor[x^4 + 25, GaussianIntegers → True]}
```

Out[3]= $\{25 + x^4,\ (-5\, i + x^2)\ (5\, i + x^2)\}$

```
In[4]:= {Factor[x^6 - 10], Factor[x^6 - 10, Extension → Sqrt[10]],
        Factor[x^6 - 10, Extension → 10^(1/6)] }
```

Out[4]= $\left\{-10+x^6,\ -\left(\sqrt{10}-x^3\right)\left(\sqrt{10}+x^3\right),\ -\left(10^{1/6}-x\right)\left(10^{1/6}+x\right)\left(10^{1/3}-10^{1/6}\,x+x^2\right)\left(10^{1/3}+10^{1/6}\,x+x^2\right)\right\}$

```
In[5]:= {Factor[Cosh[x]^2 - Sinh[x]^2], Factor[Cos[x]^2 + Sin[x]^2],
        Factor[Cos[x]^2 + Sin[x]^2, GaussianIntegers → True]}
```

Out[5]= $\{(\mathrm{Cosh}[x]-\mathrm{Sinh}[x])\,(\mathrm{Cosh}[x]+\mathrm{Sinh}[x]),\ \mathrm{Cos}[x]^2+\mathrm{Sin}[x]^2,\ (\mathrm{Cos}[x]-i\,\mathrm{Sin}[x])\,(\mathrm{Cos}[x]+i\,\mathrm{Sin}[x])\}$

■ ΕΝΤΟΛΗ Α4: ΑΝΑΠΤΥΓΜΑ ΠΑΡΑΣΤΑΣΕΩΣ

Expand[*Παράσταση*]

Αναπτύσσει μια παράσταση εκτελώντας τους πολλαπλασιασμούς σε δυνάμεις ή γινόμενα. Παραδείγματα στην αμέσως επόμενη εντολή **ExpandAll**.

■ ΕΝΤΟΛΗ Α5: ΠΛΗΡΕΣ ΑΝΑΠΤΥΓΜΑ ΠΑΡΑΣΤΑΣΕΩΣ

ExpandAll[*Παράσταση*]

Αναπτύσσει μια παράσταση εκτελώντας τους πολλαπλασιασμούς σε δυνάμεις ή γινόμενα, όπως και η προηγούμενη εντολή **Expand**, και επιπλέον περιλαμβάνει στο ανάπτυγμα και τους παρονομαστές, όταν υπάρχει κλάσμα ή υπάρχουν κλάσματα. Παραδείγματα:

```
In[6]:= {Expand[(a + b)^5], e1 = Expand[(a - b) (a^2 + b^2 + a b)], Factor[e1]}
```

Out[6]= $\{a^5+5\,a^4\,b+10\,a^3\,b^2+10\,a^2\,b^3+5\,a\,b^4+b^5,\ a^3-b^3,\ (a-b)\,(a^2+a\,b+b^2)\}$

```
In[7]:= {Expand[(α + β)^2 / (γ + δ)^2], ExpandAll[(α + β)^2 / (γ + δ)^2]}
```

Out[7]= $\left\{\frac{\alpha^2}{(\gamma+\delta)^2}+\frac{2\,\alpha\,\beta}{(\gamma+\delta)^2}+\frac{\beta^2}{(\gamma+\delta)^2},\ \frac{\alpha^2}{\gamma^2+2\,\gamma\,\delta+\delta^2}+\frac{2\,\alpha\,\beta}{\gamma^2+2\,\gamma\,\delta+\delta^2}+\frac{\beta^2}{\gamma^2+2\,\gamma\,\delta+\delta^2}\right\}$

■ ΕΝΤΟΛΗ Α6: ΑΝΑΠΤΥΓΜΑ ΔΥΝΑΜΕΩΣ ΚΑΙ ΡΙΖΑΣ ΓΙΝΟΜΕΝΟΥ

PowerExpand[*Παράσταση*]

Τροποποιεί μια παράσταση αναπτύσσοντας τη δύναμη ή τη ρίζα ενός γινομένου. Επίσης αναπτύσσει το λογάριθμο ενός γινομένου σε άθροισμα λογαρίθμων. Υποθέτει πάντοτε τους όρους του γινομένου θετικούς αριθμούς (καλύτερα μη αρνητικούς). Γι' αυτό κυρίως και είναι χρήσιμη. Παραδείγματα:

```
In[8]:= {PowerExpand[(a b c^2)^10], Sqrt[a^2 b c] // PowerExpand, Log[a b c^d] // PowerExpand}
```

Out[8]= $\{a^{10}\,b^{10}\,c^{20},\ a\,\sqrt{b}\,\sqrt{c},\ \mathrm{Log}[a]+\mathrm{Log}[b]+\mathrm{Log}[c]\,(c\,x^{10}+e\,\mathrm{Sin}[x])\}$

Το παρακάτω δεύτερο παράδειγμα είναι χρήσιμο στις Ταλαντώσεις με απόσβεση, όπου φυσικά η ιδιοσυχνότητα (η φυσική κυκλική συχνότητα) του μηχανικού συστήματος ω είναι πάντοτε θετικός αριθμός:

In[9]:= `{Sqrt[ω^2 (1 - ξ^2)], Sqrt[ω^2 (1 - ξ^2)] // PowerExpand}`

Out[9]= $\left\{\sqrt{(1-\xi^2)\,\omega^2},\ \sqrt{1-\xi^2}\,\omega\right\}$

■ ΕΝΤΟΛΗ Α7: ΑΠΑΛΟΙΦΗ ΜΕΤΑΒΛΗΤΩΝ ΣΕ ΕΞΙΣΩΣΕΙΣ

Eliminate[*ΛίσταΕξισώσεων, Μεταβλητή*]

Eliminate[*ΛίσταΕξισώσεων, ΛίσταΜεταβλητών*]

Απαλείφει τη μεταβλητή ή τις μεταβλητές του δεύτερου ορίσματός της από τη λίστα εξισώσεων στο πρώτο όρισμά της. Παραδείγματα:

In[10]:= `Eliminate[{x + 3 y^2 == 5, x^6 + y^4 + x y == 6}, y]`

Out[10]= $580\,x - 93\,x^2 + 7\,x^3 + x^4 - 522\,x^6 - 180\,x^7 + 18\,x^8 + 81\,x^{12} == -841$

In[11]:= `{eqs = {x^2 + y^2 + z^2 == z, x^2 - y^2 - z^2 == 2 z^3, x^2 + 3 y^2 == 3 z}, Eliminate[eqs, {x, y}]}`

Out[11]= $\{\{x^2 + y^2 + z^2 == z,\ x^2 - y^2 - z^2 == 2\,z^3,\ x^2 + 3\,y^2 == 3\,z\},\ z + 3\,z^2 + 2\,z^3 == 0\}$

■ ΕΝΤΟΛΗ Α8: ΑΝΑΛΥΣΗ ΚΛΑΣΜΑΤΩΝ

Apart[*ΚλασματικήΠαράσταση*]

Αναλύει ένα κλάσμα ή μια κλασματική παράσταση γενικότερα σε απλά κλάσματα. Παραδείγματα:

In[12]:= `{e1 = Apart[1 / (x^2 - a^2)], e2 = Apart[(a x + b) / (x^3 - a^3)]}`

Out[12]= $\left\{\frac{1}{2\,a\,(-a+x)} - \frac{1}{2\,a\,(a+x)},\ \frac{a^2+b}{3\,a^2\,(-a+x)} + \frac{a^3 - 2\,a\,b - a^2\,x - b\,x}{3\,a^2\,(a^2+a\,x+x^2)}\right\}$

■ ΕΝΤΟΛΗ Α9: ΣΥΜΠΤΥΞΗ ΚΛΑΣΜΑΤΩΝ

Together[*ΚλασματικήΠαράσταση*]

Συμπτύσσει όλα τα κλάσματα μιας κλασματικής παραστάσεως σε ένα ενιαίο κλάσμα. Παραδείγματα:

In[13]:= `{Together[1 / (x - a) + 2 / (x + b)^2], {e2, Together[e2]}}`

Out[13]= $\left\{\frac{2\,a - b^2 - 2\,x - 2\,b\,x - x^2}{(a-x)\,(b+x)^2},\ \left\{\frac{a^2+b}{3\,a^2\,(-a+x)} + \frac{a^3 - 2\,a\,b - a^2\,x - b\,x}{3\,a^2\,(a^2+a\,x+x^2)},\ \frac{-b - a\,x}{(a-x)\,(a^2+a\,x+x^2)}\right\}\right\}$

■ ΕΝΤΟΛΗ Α10: ΒΑΘΜΟΣ ΠΟΛΥΩΝΥΜΟΥ

Exponent[*Πολυώνυμο, Μεταβλητή*]

Υπολογίζει το βαθμό πολυωνύμου ως προς την καθοριζόμενη μεταβλητή (ή παράσταση γενικότερα). Παραδείγματα:

In[14]:= `{Exponent[a x^2 + b x + c, x], Exponent[a x^20 y^3 + b x y^5 + c x^2 y, y], Exponent[e^(10 x), e^x]}`

Out[14]= $\{2, 5, 10\}$

■ ΕΝΤΟΛΗ Α11: ΜΕΤΑΒΛΗΤΕΣ ΠΟΛΥΩΝΥΜΟΥ

Variables[*Πολυώνυμο*]

Δίνει λίστα με όλες τις μεταβλητές ενός πολυωνύμου. Παραδείγματα:

```
In[15]:= {Variables[ x^2 + 3 x y + 5 y^2], pol = a x^2 + b x y + c y^2; Variables[pol]}

Out[15]= {{x, y}, {a, b, c, x, y}}
```

■ ΕΝΤΟΛΗ Α12: ΠΟΛΥΩΝΥΜΙΚΗ ΜΟΡΦΗ

Collect[*Πολυώνυμο, Μεταβλητή*]

Collect[*Πολυώνυμο, Μεταβλητή*, Simplify]

Μετασχηματίζει ένα ακατάστατα γραμμένο πολυώνυμο στη συνηθισμένη μορφή του ως προς τη μεταβλητή που δίνεται. Με τη χρήση και της επιλογής **Simplify** (δεύτερη μορφή) κάνει επιπλέον και απλοποίηση των συντελεστών του πολυωνύμου έναν–έναν χωριστά όσο μπορεί. Παραδείγματα:

```
In[16]:= {Collect[2 x + x^5 - 3 x^3 + 1 + 3 x - 8, x],
          {p1 = x^2 - 2 y + x^2 y, Collect[p1, x], Collect[p1, y]}}

Out[16]= {-7 + 5 x - 3 x^3 + x^5, {x^2 - 2 y + x^2 y, -2 y + x^2 (1 + y), x^2 + (-2 + x^2) y}}

In[17]:= {p2 = (x^3 + x^2 y - y + y^2)^2, Collect[p2, x],
          Collect[p2, x, Simplify], Simplify[Collect[p2, x]]}

Out[17]= {(x^3 - y + x^2 y + y^2)^2, x^6 + 2 x^5 y + y^2 + x^4 y^2 - 2 y^3 + y^4 + x^3 (-2 y + 2 y^2) + x^2 (-2 y^2 + 2 y^3),
          x^6 + 2 x^5 y + 2 x^3 (-1 + y) y + x^4 y^2 + 2 x^2 (-1 + y) y^2 + (-1 + y)^2 y^2, (x^3 + x^2 y + (-1 + y) y)^2}
```

■ ΕΝΤΟΛΗ Α13: ΣΥΝΤΕΛΕΣΤΗΣ ΠΟΛΥΩΝΥΜΟΥ

Coefficient[*Πολυώνυμο, Μεταβλητή^Δύναμη*] ή **Coefficient[*Πολυώνυμο, Μεταβλητή, Δύναμη*]**

Coefficient[*Πολυώνυμο, ΛίσταΜεταβλητώνΥψωμένωνΣεΔυνάμεις*]

Υπολογίζει το συντελεστή ή τους συντελεστές ενός πολυωνύμου που αντιστοιχούν στη μεταβλητή που δίνεται. Αυτή είναι υψωμένη σε δύναμη $k > 0$. (Για $k = 0$ ισχύει μόνο η δεξιά σύνταξη μετά το "ή".) Παραδείγματα:

```
In[18]:= pol = (a x + b y^2)^8;

In[19]:= {c1 = Coefficient[pol, x^5], c2 = Coefficient[pol, x, 5], c1 == c2}

Out[19]= {56 a^5 b^3 y^6, 56 a^5 b^3 y^6, True}
```

Σημειώνουμε ξανά ότι όταν $k = 0$, τότε μπορεί να χρησιμοποιηθεί μόνο η δεξιά μορφή της εντολής:

```
In[20]:= {Coefficient[pol, x, 0], Coefficient[pol, y, 0], Coefficient[pol, y^10]}

Out[20]= {b^8 y^16, a^8 x^8, 56 a^3 b^5 x^3}
```

Στο τελευταίο αυτό παράδειγμα σημειώνεται ότι η δύναμη *k* στο x^k είναι θετική, ώστε να μπορέσει να λειτουργήσει σωστά η εντολή αυτή έτσι όπως είναι γραμμένη:

```
In[21]:= {tb = Table[x^k, {k, 1, 10}], Coefficient[pol, tb]}

Out[21]= {{x, x^2, x^3, x^4, x^5, x^6, x^7, x^8, x^9, x^10},
  {8 a b^7 y^14, 28 a^2 b^6 y^12, 56 a^3 b^5 y^10, 70 a^4 b^4 y^8, 56 a^5 b^3 y^6, 28 a^6 b^2 y^4, 8 a^7 b y^2, a^8, 0, 0}}
```

■ ΕΝΤΟΛΗ Α14: ΣΥΝΤΕΛΕΣΤΕΣ ΠΟΛΥΩΝΥΜΟΥ

CoefficientList[*Πολυώνυμο, Μεταβλητή*]

Υπολογίζει όλους μαζί τους συντελεστές ενός πολυωνύμου, όχι έναν–έναν, αντίθετα με την προηγούμενη εντολή **Coefficient**. Συνέχεια του προηγούμενου παραδείγματος (για το ίδιο πολυώνυμο **pol**):

```
In[22]:= CoefficientList[pol, x]

Out[22]= {b^8 y^16, 8 a b^7 y^14, 28 a^2 b^6 y^12, 56 a^3 b^5 y^10, 70 a^4 b^4 y^8, 56 a^5 b^3 y^6, 28 a^6 b^2 y^4, 8 a^7 b y^2, a^8}
```

■ ΕΝΤΟΛΗ Α15: ΑΠΛΟΠΟΙΗΣΗ ΠΑΡΑΣΤΑΣΕΩΣ

Simplify[*Παράσταση*]

Simplify[*ΛίσταΠαραστάσεων*]

Simplify[*Παράσταση, Υπόθεση*]

Simplify[*Παράσταση, ΛίσταΥποθέσεων*]

Simplify[Λίστα*Παραστάσεων, ΛίσταΥποθέσεων*]

Απλοποιεί αλγεβρικά, τριγωνομετρικά, εκθετικά, κλπ. την παράσταση ή τη λίστα παραστάσεων που δίνεται. Υπάρχει επίσης η δυνατότητα να χρησιμοποιεί και υπόθεση ή λίστα υποθέσεων. Πολύ συχνά μπαίνει μετά την προς απλοποίηση παράσταση π.χ. με //**Simplify**. Παραδείγματα στην αμέσως επόμενη εντολή **FullSimplify**.

■ ΕΝΤΟΛΗ Α16: ΠΛΗΡΗΣ ΑΠΛΟΠΟΙΗΣΗ ΠΑΡΑΣΤΑΣΕΩΣ

FullSimplify[*Παράσταση*] και ανάλογα για όλες τις δυνατότητες σύνταξης της εντολής Simplify

Η εντολή **FullSimplify** εκτελεί απλοποιήσεις ανάλογα με την προηγούμενη εντολή **Simplify**, διαθέτει όμως πολύ μεγαλύτερη συλλογή κανόνων απλοποιήσεων και επομένως απαιτεί συνήθως πολύ περισσότερο χρόνο εργασίας. Και σ' αυτή μπορούν να χρησιμοποιούνται λίστα παραστάσεων, υπόθεση ή λίστα υποθέσεων ακριβώς όπως και στην εντολή **Simplify**. Πολύ συχνά και η εντολή **FullSimplify** μπαίνει κι αυτή μετά την προς πλήρη απλοποίηση παράσταση π.χ. με //**FullSimplify**. Παραδείγματα και των δύο εντολών **Simplify** και **FullSimplify**:

Στο πρώτο αυτό παράδειγμα αρκεί η χρήση της εντολής **Simplify**. Παρατηρούμε ότι γίνονται απλοποιήσεις όχι μόνο σε αλγεβρικές, αλλά και σε τριγωνομετρικές και σε υπερβολικές παραστάσεις. Είναι λοιπόν γενικά εφαρμόσιμη η εντολή **Simplify**.

```
In[23]:= {Simplify[a^2 + b^2 - 2 a b], Simplify[Cos[x]^2 + Sin[x]^2], Cosh[x]^2 - Sinh[x]^2 // Simplify}
```

```
Out[23]= {(a - b)^2, 1, 1}
```

Στο δεύτερο αυτό παράδειγμα, για να γίνει η απλοποίηση, χρειάζεται, εύλογα, να έχει υποτεθεί από πριν ότι ο αριθμός *x* είναι θετικός:

```
In[24]:= {Simplify[Sqrt[x^2]], Simplify[Sqrt[x^2], x > 0]}
```

Out[24]= $\{\sqrt{x^2}, x\}$

Στο τρίτο αυτό παράδειγμα παρατηρούμε ότι η εντολή **Simplify** δεν είναι επαρκής στην παρούσα απλοποίηση παραγοντικών. Απαιτείται η χρήση της εντολής **FullSimplify**:

```
In[25]:= {(k + 1)! == (k + 1) k!, (k + 1)! == (k + 1) k! // Simplify,
          (k + 1)! == (k + 1) k! // FullSimplify}
```

```
Out[25]= {(1 + k)! == (1 + k) k!, (1 + k)! == (1 + k) k!, True}
```

Τα ίδια ακριβώς συμβαίνουν και στο επόμενο τέταρτο παράδειγμα με τη συνάρτηση γάμμα:

```
In[26]:= {expr = Gamma[k + 1] == k!, expr // Simplify, expr // FullSimplify}
```

```
Out[26]= {Gamma[1 + k] == k!, Gamma[1 + k] == k!, True}
```

Επίσης τα ίδια και στο επόμενο πέμπτο παράδειγμα με τις δύο γνωστές συναρτήσεις τόξο ημιτόνου και τόξο συνημιτόνου:

```
In[27]:= {ArcSin[x] + ArcCos[x] == π / 2 // Simplify, ArcSin[x] + ArcCos[x] == π / 2 // FullSimplify}
```

Out[27]= $\{\mathrm{ArcCos}[x] + \mathrm{ArcSin}[x] == \frac{\pi}{2}, \mathrm{True}\}$

Όμως στο πιο κάτω έκτο παράδειγμα με υπερβολικές συναρτήσεις αρκεί η χρήση της εντολής **Simplify**:

```
In[28]:= {2 Cosh[x] Sinh[x], 2 Cosh[x] Sinh[x] // Simplify}
```

```
Out[28]= {2 Cosh[x] Sinh[x], Sinh[2 x]}
```

Εδώ στο έβδομο παράδειγμα απαιτούνται δύο ταυτόχρονα υποθέσεις για να γίνει η απλοποίηση. Δίνεται σαν δεύτερο όρισμα της εντολής **Simplify** λίστα με δύο υποθέσεις:

```
In[29]:= Simplify[Sqrt[-a^2] Sqrt[ω^2 - ξ^2 ω^2], {a > 0, ω > 0}]
```

Out[29]= $i\, a \sqrt{1 - \xi^2}\, \omega$

Στο επόμενο, στο όγδοο παράδειγμα η εντολή **Simplify** εφαρμόζεται σε λίστα παραστάσεων:

```
In[30]:= Simplify[{ArcCosh'[x] , ArcSinh'[x]}, -1 < x < 1]
```

Out[30]= $\left\{\frac{1}{\sqrt{-1 + x^2}}, \frac{1}{\sqrt{1 + x^2}}\right\}$

Μπορούμε να έχουμε και λίστα παραστάσεων και λίστα υποθέσεων όπως στο τελευταίο μας παράδειγμα:

```
In[31]:= Simplify[{Sqrt[x^2 y^2], Sqrt[x^2 y^4 z^6], Sqrt[(x y z)^2]}, {x > 0, y > 0, z > 0}]
```

Out[31]= $\{x\, y, x\, y^2 z^3, x\, y\, z\}$

■ *Notebook E4*
ΕΝΤΟΛΕΣ ΓΙΑ ΤΡΙΓΩΝΟΜΕΤΡΙΚΕΣ ΚΑΙ ΥΠΕΡΒΟΛΙΚΕΣ ΣΥΝΑΡΤΗΣΕΙΣ

4 ΕΝΤΟΛΕΣ: T1. ExpToTrig, T2. TrigToExp, T3. TrigExpand, T4. TrigReduce

ΠΑΡΑΤΗΡΗΣΕΙΣ: (α) Συχνά οι εντολές αυτές μπαίνουν μετά την παράσταση στην οποία αναφέρονται με το σύμβολο // μπροστά τους. Βέβαια αυτό δεν είναι με κανέναν τρόπο υποχρεωτικό. Είναι όμως εξυπηρετικό! (β) Για την απλοποίηση τριγωνομετρικών ή/και υπερβολικών παραστάσεων συνήθως αρκεί η χρήση της εντολής **Simplify** ή της εντολής **FullSimplify**, η οποία είναι και ισχυρότερη. Παράδειγμα:

```
In[1]:= {tr = (Tan[α] + Tan[β]) / (1 - Tan[α] Tan[β]); tr // Simplify, tr // FullSimplify}
Out[1]= {Tan[α + β], Tan[α + β]}
```

■ ΕΝΤΟΛΗ T1: ΕΚΘΕΤΙΚΗ ΜΟΡΦΗ ΣΕ ΤΡΙΓΩΝΟΜΕΤΡΙΚΗ / ΥΠΕΡΒΟΛΙΚΗ

ExpToTrig[*ΕκθετικήΠαράσταση*]

Η εντολή αυτή μετατρέπει μια παράσταση που περιέχει εκθετικές συναρτήσεις (ή πραγματικές ή μιγαδικές) σε ισοδύναμη παράσταση με υπερβολικές ή/και τριγωνομετρικές συναρτήσεις. Παραδείγματα:

In[2]:= {ExpToTrig[$e^{a x}$], ExpToTrig[{$e^{i a x}$, $e^{-i a x}$}], {$e^{i \omega t}$, $e^{-i \omega t}$} // ExpToTrig}

Out[2]= {Cosh[a x] + Sinh[a x], {Cos[a x] + i Sin[a x], Cos[a x] - i Sin[a x]}, {Cos[t ω] + i Sin[t ω], Cos[t ω] - i Sin[t ω]}}

In[3]:= expr = A $e^{\beta x}$ + B $e^{-\beta x}$ + C $e^{i \gamma x}$ + D $e^{-i \gamma x}$ // ExpToTrig // Simplify

Out[3]= C Cos[x γ] + D Cos[x γ] + (A + B) Cosh[x β] + i C Sin[x γ] - i D Sin[x γ] + (A - B) Sinh[x β]

■ ΕΝΤΟΛΗ T2: ΤΡΙΓΩΝΟΜΕΤΡΙΚΗ / ΥΠΕΡΒΟΛΙΚΗ ΜΟΡΦΗ ΣΕ ΕΚΘΕΤΙΚΗ

TrigToExp[*Τριγωνομετρική / ΥπερβολικήΠαράσταση*]

Η εντολή αυτή μετατρέπει μια παράσταση με τριγωνομετρικές ή/και υπερβολικές συναρτήσεις ή αντίστροφές τους σε ισοδύναμη παράσταση με εκθετικές ή λογαριθμικές συναρτήσεις. Παραδείγματα:

In[4]:= {TrigToExp[Cosh[a x]], TrigToExp[Tan[a x]], {Cos[ω t], Sin[ω t]} // TrigToExp}

Out[4]= $\left\{\frac{e^{-a x}}{2} + \frac{e^{a x}}{2}, \frac{i\,(e^{-i a x} - e^{i a x})}{e^{-i a x} + e^{i a x}}, \left\{\frac{1}{2} e^{-i t \omega} + \frac{1}{2} e^{i t \omega}, \frac{1}{2} i\, e^{-i t \omega} - \frac{1}{2} i\, e^{i t \omega}\right\}\right\}$

In[5]:= {ArcTanh[x], ArcTan[x], ArcSin[x]} // TrigToExp

Out[5]= $\left\{-\frac{1}{2} \mathrm{Log}[1 - x] + \frac{1}{2} \mathrm{Log}[1 + x], \frac{1}{2} i\, \mathrm{Log}[1 - i x] - \frac{1}{2} i\, \mathrm{Log}[1 + i x], -i\, \mathrm{Log}\left[i x + \sqrt{1 - x^2}\right]\right\}$

In[6]:= A Cosh[β x] + B Sinh[β x] + C Cos[β x] + D Sin[β x] // TrigToExp

Out[6]= $\frac{1}{2} A\, e^{-x \beta} - \frac{1}{2} B\, e^{-x \beta} + \frac{1}{2} C\, e^{-i x \beta} + \frac{1}{2} i\, D\, e^{-i x \beta} + \frac{1}{2} C\, e^{i x \beta} - \frac{1}{2} i\, D\, e^{i x \beta} + \frac{1}{2} A\, e^{x \beta} + \frac{1}{2} B\, e^{x \beta}$

■ ΕΝΤΟΛΗ Τ3: ΤΡΙΓΩΝΟΜΕΤΡΙΚΟ / ΥΠΕΡΒΟΛΙΚΟ ΑΝΑΠΤΥΓΜΑ

TrigExpand[*Τριγωνομετρική / ΥπερβολικήΠαράσταση*]

Η εντολή αυτή αναπτύσσει μια παράσταση που περιέχει τριγωνομετρικές ή υπερβολικές συναρτήσεις σε ισοδύναμη παράσταση με ανάλογες συναρτήσεις για τις βασικές μεταβλητές μόνο, π.χ. α, β, γ, x ή y, δηλαδή χωρίς αθροίσματα ούτε πολλαπλάσιά τους σαν ορίσματα, π.χ. χωρίς 2α ή $\alpha+\beta$. Παραδείγματα:

```
In[7]:= {TrigExpand[Cosh[α + β + γ]], Sin[α - β] // TrigExpand}

Out[7]= {Cosh[α] Cosh[β] Cosh[γ] + Cosh[γ] Sinh[α] Sinh[β] +
          Cosh[β] Sinh[α] Sinh[γ] + Cosh[α] Sinh[β] Sinh[γ], Cos[β] Sin[α] - Cos[α] Sin[β]}

In[8]:= TrigExpand[Cosh[5 x]]

Out[8]= Cosh[x]^5 + 10 Cosh[x]^3 Sinh[x]^2 + 5 Cosh[x] Sinh[x]^4

In[9]:= Cosh[x] + Cosh[2 x] + Cosh[3 x] // TrigExpand

Out[9]= Cosh[x] + Cosh[x]^2 + Cosh[x]^3 + Sinh[x]^2 + 3 Cosh[x] Sinh[x]^2

In[10]:= {Sinh[α + β] // TrigExpand, Sin[α + β] // TrigExpand}

Out[10]= {Cosh[β] Sinh[α] + Cosh[α] Sinh[β], Cos[β] Sin[α] + Cos[α] Sin[β]}
```

■ ΕΝΤΟΛΗ Τ4: ΤΡΙΓΩΝΟΜΕΤΡΙΚΗ / ΥΠΕΡΒΟΛΙΚΗ ΣΥΜΠΤΥΞΗ

TrigReduce[*Τριγωνομετρική / ΥπερβολικήΠαράσταση*]

Η εντολή αυτή μετατρέπει μια παράσταση που περιέχει τριγωνομετρικές ή υπερβολικές συναρτήσεις με γινόμενα ή/και δυνάμεις σε ισοδύναμη παράσταση με άθροισμα τριγωνομετρικών ή υπερβολικών όρων (χωρίς γινόμενα ή δυνάμεις), αλλά με πιο σύνθετα ορίσματα, π.χ. 2α ή 5β ή $\alpha+\beta$. Παραδείγματα:

```
In[11]:= {TrigReduce[Cosh[a] Cosh[b]],
          Cos[α] Cos[β] // TrigReduce, Sin[α] Sin[β] // TrigReduce}

Out[11]= {1/2 (Cosh[a - b] + Cosh[a + b]), 1/2 (Cos[α - β] + Cos[α + β]), 1/2 (Cos[α - β] - Cos[α + β])}

In[12]:= {TrigReduce[Cosh[x]^3 Sinh[x]^2], Cos[a x + b]^4 // TrigReduce}

Out[12]= {1/16 (-2 Cosh[x] + Cosh[3 x] + Cosh[5 x]), 1/8 (3 + 4 Cos[2 b + 2 a x] + Cos[4 b + 4 a x])}

In[13]:= {Cosh[a x]^2 - Sinh[a x]^2, {Cosh[a x]^2 - Sinh[a x]^2, Cos[ω t]^2 + Sin[ω t]^2} // TrigReduce}

Out[13]= {Cosh[a x]^2 - Sinh[a x]^2, {1, 1}}

In[14]:= {Sinh[α] Cosh[β] + Cosh[α] Sinh[β], Cos[α] Cos[β] - Sin[α] Sin[β]} // TrigReduce

Out[14]= {Sinh[α + β], Cos[α + β]}

In[15]:= {tr = Cos[α + β + γ] // TrigExpand, tr // TrigReduce}

Out[15]= {Cos[α] Cos[β] Cos[γ] - Cos[γ] Sin[α] Sin[β] -
           Cos[β] Sin[α] Sin[γ] - Cos[α] Sin[β] Sin[γ], Cos[α + β + γ]}
```

■ *Notebook E5*
ΕΝΤΟΛΕΣ ΓΙΑ ΑΘΡΟΙΣΜΑΤΑ, ΣΕΙΡΕΣ ΚΑΙ ΓΙΝΟΜΕΝΑ

5 ΕΝΤΟΛΕΣ: **S1. Zeta, S2. Sum, S3. NSum, S4. Product, S5. NProduct**

■ ΕΝΤΟΛΗ S1: ΣΥΝΑΡΤΗΣΗ ΖΗΤΑ

Zeta[*Μεταβλητή*]

Υπολογίζει τη συνάρτηση ζήτα, που παρουσιάζεται σε ορισμένες αριθμητικές σειρές. Παραδείγματα:

```
In[1]:= {Zeta[0], ZetaTable = Table[Zeta[k], {k, 2, 5}], N[ZetaTable]}
```

Out[1]= $\left\{-\frac{1}{2}, \left\{\frac{\pi^2}{6}, \text{Zeta}[3], \frac{\pi^4}{90}, \text{Zeta}[5]\right\}, \{1.64493, 1.20206, 1.08232, 1.03693\}\right\}$

■ ΕΝΤΟΛΗ S2: ΑΘΡΟΙΣΜΑ / ΣΕΙΡΑ

Sum[*Παράσταση*, {*Δείκτης*, *ΑρχικήΤιμήΤουΔείκτη*, *ΤελικήΤιμήΤουΔείκτη*}]

Sum[*Παράσταση*, {*Δείκτης*, *ΑρχικήΤιμήΤουΔείκτη*, *ΤελικήΤιμήΤουΔείκτη*, *ΒήμαΜεταβολής*}]

Υπολογίζει σε κλειστή μορφή το άθροισμα με τιμές του δείκτη *k*, *n*, κλπ. από την αρχική του τιμή μέχρι την τελική του τιμή. Η τελική αυτή τιμή μπορεί να είναι το άπειρο (**Infinity** ή καλύτερα ∞ με **Esc inf Esc**), οπότε δεν έχουμε απλό άθροισμα, αλλά σειρά. Η παράσταση στο άθροισμα μπορεί να περιέχει εκτός από το δείκτη και σύμβολα. Όταν το βήμα που μεταβάλλεται ο δείκτης δεν είναι το ένα, τότε πρέπει υποχρεωτικά να αναφέρεται κι αυτό σαν τέταρτο στοιχείο στο δεύτερο όρισμα. Παραδείγματα:

```
In[2]:= {s = Sum[Sin[k π / 5], {k, 2, 5}], N[s, 40]}
```

Out[2]= $\left\{\frac{1}{2}\sqrt{\frac{1}{2}\left(5-\sqrt{5}\right)}+\sqrt{\frac{1}{2}\left(5+\sqrt{5}\right)}, 2.489898284882780273401584621397837055409\right\}$

```
In[3]:= {s1[n_] = Sum[k^12, {k, 1, n}] // Simplify, s1[1], s1[2], s1[3], s1[4]}
```

Out[3]= $\left\{-\frac{691\,n}{2730}+\frac{5\,n^3}{3}-\frac{33\,n^5}{10}+\frac{22\,n^7}{7}-\frac{11\,n^9}{6}+n^{11}+\frac{n^{12}}{2}+\frac{n^{13}}{13}, 1, 4097, 535538, 17312754\right\}$

```
In[4]:= {s2[n_] = Sum[1 / k^n, {k, 1, ∞}], s2[n] == Zeta[n], s8 = s2[8], N[s8, 50]}
```

Out[4]= $\left\{\text{Zeta}[n], \text{True}, \frac{\pi^8}{9450}, 1.0040773561979443393786852385086524652589607906499\right\}$

```
In[5]:= {s3 = Sum[(-1)^k / k!, {k, 0, ∞}], N[s3, 25], s4 = Sum[(-1)^k / k!, {k, 1, ∞, 2}], N[s4, 25]}
```

Out[5]= $\left\{\frac{1}{e}, 0.3678794411714423215955238, -\sqrt{\frac{\pi}{2}}\,\text{BesselI}\left[\frac{1}{2}, 1\right], -1.175201193643801456882382\right\}$

```
In[6]:= {Sum[x^k, {k, 0, n}], Sum[x^k, {k, 0, ∞}], Sum[x^k / k!, {k, 0, n}], Sum[x^k / k!, {k, 0, ∞}]}
```

Out[6]= $\left\{\frac{-1+x^{1+n}}{-1+x}, \frac{1}{1-x}, \frac{e^x\,(1+n)\,\text{Gamma}[1+n, x]}{\text{Gamma}[2+n]}, e^x\right\}$

■ ΕΝΤΟΛΗ S3: ΑΡΙΘΜΗΤΙΚΟΣ ΥΠΟΛΟΓΙΣΜΟΣ ΑΘΡΟΙΣΜΑΤΟΣ / ΣΕΙΡΑΣ

NSum[*Παράσταση, {Δείκτης, ΑρχικήΤιμήΤουΔείκτη, ΤελικήΤιμήΤουΔείκτη*}]

NSum[*Παράσταση, {Δείκτης, ΑρχικήΤιμήΤουΔείκτη, ΤελικήΤιμήΤουΔείκτη, ΒήμαΜεταβολής*}]

Εντολή ανάλογη με την προηγούμενη εντολή **Sum**, αλλά τώρα το άθροισμα ή η σειρά υπολογίζεται αριθμητικά πάλι με τιμές του δείκτη *k*, *n*, κλπ. από την αρχική του τιμή μέχρι και την τελική του τιμή. Η τελική αυτή τιμή μπορεί ξανά να είναι το άπειρο (**Infinity** ή καλύτερα ∞ με **Esc inf Esc**), οπότε δεν έχουμε απλό άθροισμα, αλλά σειρά. Η παράσταση στο άθροισμα μπορεί να περιέχει και πάλι εκτός από το δείκτη και σύμβολα. Όταν το βήμα που μεταβάλλεται ο δείκτης δεν είναι το ένα, τότε πρέπει υποχρεωτικά να αναφέρεται κι αυτό, ακριβώς όπως και στην προηγούμενη εντολή **Sum**. Παραδείγματα:

In[7]:= `{Sum[1 / (1 - Exp[k^2]), {k, 2, 5}], NSum[1 / (1 - Exp[k^2]), {k, 2, 5}]}`

Out[7]= $\left\{\frac{1}{1-e^4}+\frac{1}{1-e^9}+\frac{1}{1-e^{16}}+\frac{1}{1-e^{25}},\ -0.0187809\right\}$

In[8]:= `{s = Sum[Cos[k] / Cosh[k], {k, 1, ∞}], N[s], NSum[Cos[k] / Cosh[k], {k, 1, ∞}]}`

Out[8]= $\left\{\sum_{k=1}^{\infty}\frac{\mathrm{Cos}[k]}{\mathrm{Cosh}[k]},\ 0.126836,\ 0.126836\right\}$

■ ΕΝΤΟΛΗ S4: ΓΙΝΟΜΕΝΟ

Product[*Παράσταση, {Δείκτης, ΑρχικήΤιμήΤουΔείκτη, ΤελικήΤιμήΤουΔείκτη*}]

Product[*Παράσταση, {Δείκτης, ΑρχικήΤιμήΤουΔείκτη, ΤελικήΤιμήΤουΔείκτη, ΒήμαΜεταβολής*}]

Εντελώς ανάλογη εντολή με την εντολή **Sum** (γι' αθροίσματα), αλλ' εδώ για γινόμενα. Παραδείγματα:

In[9]:= `{Product[n, {n, 1, 10}], pr[x_] = Product[n (x^2 + 1), {n, 1, 10}], pr[50]}`

Out[9]= $\left\{3628800,\ 3628800\,(1+x^2)^{10},\ 347456107640142612072743050206907236288 00\right\}$

In[10]:= `{{Product[(1 + x^2 / (k π)^2), {k, 1, ∞}], Product[(1 - x^2 / (k π)^2), {k, 1, ∞}]} // PowerExpand, Product[1 - 1 / k^2, {k, 2, ∞}], Product[1 + x^(2^k), {k, 0, ∞}]}`

Out[10]= $\left\{\left\{\frac{\mathrm{Sinh}[x]}{x},\ \frac{\mathrm{Sin}[x]}{x}\right\},\ \frac{1}{2},\ \frac{1}{1-x}\right\}$

■ ΕΝΤΟΛΗ S5: ΑΡΙΘΜΗΤΙΚΟΣ ΥΠΟΛΟΓΙΣΜΟΣ ΓΙΝΟΜΕΝΟΥ

NProduct[*Παράσταση, {Δείκτης, ΑρχικήΤιμήΤουΔείκτη, ΤελικήΤιμήΤουΔείκτη*}]

NProduct[*Παράσταση, {Δείκτης, ΑρχικήΤιμήΤουΔείκτη, ΤελικήΤιμήΤουΔείκτη, ΒήμαΜεταβολής*}]

Πλήρως ανάλογη εντολή με την εντολή **NSum** (γι' αθροίσματα), αλλ' εδώ για γινόμενα. Παραδείγματα:

In[11]:= `{p = Product[(1 + 1 / Cosh[n]), {n, 1, ∞}], N[p], NProduct[(1 + 1 / Cosh[n]), {n, 1, ∞}]}`

Out[11]= $\left\{\prod_{n=1}^{\infty}(1+\mathrm{Sech}[n]),\ 2.42827,\ 2.42827\right\}$

■ *Notebook E6*
ΕΝΤΟΛΕΣ ΓΙΑ ΤΟΝ ΑΠΕΙΡΟΣΤΙΚΟ ΛΟΓΙΣΜΟ
8 ΕΝΤΟΛΕΣ: Λ1. Limit, Λ2. D, Λ3. Dt, Λ4. FindMinimum,
Λ5. Integrate, Λ6. NIntegrate, Λ7. Series, Λ8. Normal

ΠΑΡΑΤΗΡΗΣΗ: Για να αποφευχθούν τα δύο μηνύματα προειδοποιήσεως λαθών **spell** και **spell1** (που δεν παρουσιάζουν καμία χρησιμότητα εδώ), χρησιμοποιήθηκε η διπλή εντολή (δύο εντολές σε μία λίστα)

```
In[1]:= {Off[General::spell], Off[General::spell1]};
```

■ ΕΝΤΟΛΗ Λ1: ΟΡΙΟ

Limit[*Συνάρτηση, Μεταβλητή* → *Σημείο*]

Limit[*Συνάρτηση, Μεταβλητή* → *Σημείο,* Direction → 1]

Limit[*Συνάρτηση, Μεταβλητή* → *Σημείο,* Direction → –1]

Υπολογίζει το όριο μιας συναρτήσεως, το μονόπλευρο όριο από αριστερά: με **Direction → 1** ή το μονόπλευρο όριο από δεξιά: με **Direction → –1**. Παραδείγματα:

Πρώτα ένα πολύ απλό όριο, ένα συνηθισμένο δίπλευρο όριο (το ίδιο και από αριστερά και από δεξιά):

```
In[2]:= {lm1 = Limit[x^2, x → 2], lm2 = Limit[x^2, x → 2, Direction → 1],
          lm3 = Limit[x^2, x → 2, Direction → -1], lm1 == lm2 == lm3}

Out[2]= {4, 4, 4, True}
```

Είναι όμως δυνατόν να υπάρχουν μόνο τα μονόπλευρα όρια και να είναι μάλιστα διαφορετικά. (Σ' αυτήν την περίπτωση η παράλειψη της επιλογής **Direction** δίνει το μονόπλευρο όριο από δεξιά. Τούτο όμως προκαλεί κάποια σύγχυση, επειδή μπορεί εύλογα να νομισθεί ότι υπάρχει το κοινό όριο, ενώ δεν υπάρχει.)

```
In[3]:= {lma = Limit[Cot[x], x → 0, Direction → 1],
          lmb = Limit[Cot[x], x → 0, Direction → -1], lma == lmb, lmc = Limit[Cot[x], x → 0]}

Out[3]= {-∞, ∞, False, ∞}
```

Και τώρα δύο κλασικά όρια

```
In[4]:= {Limit[Sin[x] / x, x → 0], Limit[Sinh[c x] / x , x → 0]}

Out[4]= {1, c}
```

Όρια για την εκθετική συνάρτηση στο συν/πλην άπειρο

```
In[5]:= {Limit[Exp[x], x → ∞], Limit[ⅇ^x, x → ∞], Limit[ⅇ^x, x → -∞], Limit[ⅇ^-x, x → ∞]}

Out[5]= {∞, ∞, 0, 0}
```

Η παράγωγος μιας συναρτήσεως είναι φυσικά κι αυτή ένα όριο (και πολύ γνωστό μάλιστα!). Παράδειγμα:

```
In[6]:= {g[x_] = Sinh[x], g'[x], lmg = Limit[(g[x + h] - g[x]) / h, h → 0], lmg == g'[x]}

Out[6]= {Sinh[x], Cosh[x], Cosh[x], True}
```

Ανάλογα όχι μόνο η πρώτη παράγωγος, αλλά και η δεύτερη παράγωγος: κι αυτή είναι ένα όριο!

```
In[7]:= {v[x_] = Cos[a x + b],
          {v'[x], lmv1 = Limit[(v[x + h] - v[x - h]) / (2 h), h → 0], lmv1 == v'[x]},
          {v''[x], lmv2 = Limit[(v[x + h] + v[x - h] - 2 v[x]) / h^2, h → 0], lmv2 == v''[x]}}

Out[7]= {Cos[b + a x], {-a Sin[b + a x], -a Sin[b + a x], True},
          {-a^2 Cos[b + a x], -a^2 Cos[b + a x], True}}
```

Τέλος ένα απλό όριο γινομένου πέντε συναρτήσεων και τέσσερα όρια (για $x \to 0$) συναρτήσεων Bessel:

```
In[8]:= Limit[{Sin[x] Sinh[x] Cot[x] Coth[x] Exp[x],
           BesselJ[0, x], BesselY[0, x], BesselI[0, x], BesselK[0, x]}, x → 0]

Out[8]= {1, 1, -∞, 1, ∞}
```

■ ΕΝΤΟΛΗ Λ2: ΠΑΡΑΓΩΓΙΣΗ

D[*Συνάρτηση, Μεταβλητή*]

D[*Συνάρτηση, {Μεταβλητή, ΦορέςΠαραγωγίσεως}*]

D[*Συνάρτηση, Μεταβλητή1, Μεταβλητή2, Μεταβλητή3, ...*]

D[*Συνάρτηση, {Μεταβλητή1, ΦορέςΠαραγωγίσεως1}, {Μεταβλητή2, ΦορέςΠαραγωγίσεως2}, ...*]

Υπολογίζει τη συνήθη παράγωγο ή τη μερική παράγωγο *(ή ίδια εντολή!)* κάθε τάξεως. Για συνήθεις παραγώγους χρησιμοποιείται και ο τόνος σαν απλούστερο σύμβολο. Παραδείγματα:

Οι γνωστές παραγωγίσεις για τα πέντε βασικά μεγέθη: *v, θ, M, Q* και *p* σε μια συνήθη δοκό σε κάμψη:

```
In[9]:= Clear[v]; {v[x], v'[x], v''[x], v'''[x], v''''[x], v''''[x] == D[v[x], {x, 4}],
          θ[x_] = v'[x], M[x_] = EI θ'[x], Q[x_] = M'[x], p[x_] = Q'[x], EI v''''[x] == p[x]}

Out[9]= {v[x], v′[x], v″[x], v^(3)[x], v^(4)[x], True, v′[x], EI v″[x], EI v^(3)[x], EI v^(4)[x], True}
```

Διάφορες συνήθεις παράγωγοι και μερικές παράγωγοι:

```
In[10]:= {D[f[x], x], D[f[x], {x, n}], v[t_] = D[u[t], t], a[t_] = D[u[t], {t, 2}],
           D[g[x, y], x, y], D[g[x, y], x, {y, 5}], D[g[x, y, z], {x, 3}, {y, 4}, z]}

Out[10]= {f′[x], f^(n)[x], u′[t], u″[t], g^(1,1)[x, y], g^(1,5)[x, y], g^(3,4,1)[x, y, z]}
```

Η πολύ γνωστή ιδιότητα στην παραγώγιση ότι γενικά (για την ακρίβεια σχεδόν πάντα) η εναλλαγή της σειράς των παραγωγίσεων σε μερικές παραγώγους δεν επηρεάζει το αποτέλεσμα είναι γνωστή εκ των προτέρων (έχει ενσωματωθεί) και στη *Mathematica* από τους προγραμματιστές της:

```
In[11]:= {D[h[x, y], x, y] == D[h[x, y], y, x], D[h[x, y, z], x, y, z] == D[h[x, y, z], z, y, x],
           ver = D[h[x, y, z], x, {y, 2}, {z, 3}] == D[h[x, y, z], {z, 3}, x, {y, 2}]}

Out[11]= {True, True, True}
```

Να και μερικές αληθινές παραγωγίσεις για τριγωνομετρικές συναρτήσεις (η τελευταία αρκετά δύσκολη!):

```
In[12]:= {D[Sin[x], x], D[Tan[x], x], D[Tanh[x], x], D[Tanh[x], {x, 9}] // Simplify}

Out[12]= {Cos[x], Sec[x]^2, Sech[x]^2,
          2 (78095 - 88234 Cosh[2 x] + 14608 Cosh[4 x] - 502 Cosh[6 x] + Cosh[8 x]) Sech[x]^10}
```

Η τριγωνομετρική συνάρτηση $\cos \omega_0 t$ επαληθεύει, όπως γνωρίζουμε, τη γραμμική διαφορική εξίσωση δευτέρας τάξεως του αρμονικού ταλαντωτή στο κλασικό μηχανικό σύστημα υλικού σημείου–ελατηρίου:

```
In[13]:= {{u[t_] = Cos[ω0 t], u'[t], u''[t]}, u''[t] + ω0^2 u[t] == 0}

Out[13]= {{Cos[t ω0], -Sin[t ω0] ω0, -Cos[t ω0] ω0^2}, True}
```

Να και μια κάπως δυσκολότερη παραγώγιση (εδώ μερική παραγώγιση) που θα απαιτούσε αρκετό χρόνο, αν γινόταν χωρίς τον υπολογιστή: με το χέρι. (Κι αν δεν γινόταν κάποιο λάθος στις πράξεις με το χέρι!)

```
In[14]:= D[Cos[x^2 y^4 Sin[2 z]], x, {y, 4}, {z, 2}] // FullSimplify

Out[14]= -16 x^3 y^4 Cos[x^2 y^4 Sin[2 z]] (6 + 6 Cos[4 z] (279 + 208 x^8 y^16 Sin[2 z]^4) +
              x^4 y^8 (8 (811 + 140 x^4 y^8) Sin[2 z]^4 - 6675 Sin[4 z]^2)) -
          64 x Sin[2 z] (-3 + x^4 y^8 (-4035 Cos[2 z]^2 + (1809 + 1459 x^4 y^8 - 4 x^8 y^16 +
                    1891 x^4 y^8 Cos[4 z] + 4 x^8 y^16 Cos[8 z]) Sin[2 z]^2)) Sin[x^2 y^4 Sin[2 z]]
```

Μια αρμονική συνάρτηση (στις δύο διαστάσεις), δηλαδή μια συνάρτηση που επαληθεύει τη διδιάστατη εξίσωση του Laplace:

```
In[15]:= {u[x_, y_] = c Cosh[a x] Sin[a y],
          ux2 = D[u[x, y], {x, 2}], uy2 = D[u[x, y], {y, 2}], s = ux2 + uy2, s == 0}

Out[15]= {c Cosh[a x] Sin[a y], a^2 c Cosh[a x] Sin[a y], -a^2 c Cosh[a x] Sin[a y], 0, True}
```

Μια άλλη συνάρτηση (επίσης στις δύο διαστάσεις) που δεν είναι αρμονική, είναι όμως διαρμονική, όπως συμβαίνει με την τασική συνάρτηση του Airy στην Επίπεδη Ελαστικότητα στη Μηχανική των Υλικών:

```
In[16]:= {A[x_, y_] = a (x^4 - y^4) + b x y + c y^2 + d x + e y + f,
          check1 = D[A[x, y], {x, 2}] + D[A[x, y], {y, 2}] == 0,
          check2 = D[A[x, y], {x, 4}] + 2 D[A[x, y], {x, 2}, {y, 2}] + D[A[x, y], {y, 4}] == 0}

Out[16]= {f + d x + e y + b x y + c y^2 + a (x^4 - y^4), 2 c + 12 a x^2 - 12 a y^2 == 0, True}
```

Και τέλος μια μη τετριμμένη (αρκετά πολύπλοκη!) εφαρμογή του γνωστού κανόνα παραγωγίσεως γινομένου συναρτήσεων: εδώ για την τρίτη παράγωγο του γινομένου των συναρτήσεων $f(x)$, $g(x)$ και $h(x)$:

```
In[17]:= D[f[x] g[x] h[x], {x, 3}]

Out[17]= Cosh[x] f[x] h[x] + 3 Sinh[x] (h[x] f′[x] + f[x] h′[x]) +
          3 Cosh[x] (2 f′[x] h′[x] + h[x] f″[x] + f[x] h″[x]) +
          Sinh[x] (3 h′[x] f″[x] + 3 f′[x] h″[x] + h[x] f^(3)[x] + f[x] h^(3)[x])
```

■ ΕΝΤΟΛΗ Λ3: ΟΛΙΚΟ ΔΙΑΦΟΡΙΚΟ ΚΑΙ ΟΛΙΚΗ ΠΑΡΑΓΩΓΟΣ

Dt[*Συνάρτηση*]

Dt[*Συνάρτηση, Μεταβλητή*]

Dt[*Συνάρτηση*, {*Μεταβλητή, ΦορέςΠαραγωγίσεως*}]

Στην πρώτη μορφή της υπολογίζει το (ολικό) διαφορικό μιας συναρτήσεως ως προς τις μεταβλητές της. Στη δεύτερη μορφή της υπολογίζει την ολική παράγωγο μιας συναρτήσεως ως προς τη μεταβλητή που δίνεται και στην τρίτη μορφή της την αντίστοιχη ολική παράγωγο *n*-τάξεως. Παραδείγματα:

Το διαφορικό μιας συναρτήσεως μιας μεταβλητής (επίσης και με απλούστερη τελική γραφή):

```
In[18]:= {df = Dt[f[x]], df1 = df /. Dt[x] → dx} // Simplify

Out[18]= {Dt[x] f′[x], dx f′[x]}
```

Το διαφορικό μιας συναρτήσεως τριών μεταβλητών (και πάλι και με απλούστερη τελική γραφή):

```
In[19]:= {dg = Dt[g[x, y, z]], dg1 = dg /. {Dt[x] → dx, Dt[y] → dy, Dt[z] → dz}}
```

Out[19]= $\{\mathrm{Dt}[z]\, g^{(0,0,1)}[x, y, z] + \mathrm{Dt}[y]\, g^{(0,1,0)}[x, y, z] + \mathrm{Dt}[x]\, g^{(1,0,0)}[x, y, z],$
$dz\, g^{(0,0,1)}[x, y, z] + dy\, g^{(0,1,0)}[x, y, z] + dx\, g^{(1,0,0)}[x, y, z]\}$

Ένα οποιοδήποτε σύμβολο *c*, *z*, κλπ. μπορεί εύκολα να θεωρηθεί σταθερά με τη χρήση της εντολής:

```
In[20]:= SetAttributes[{c, z}, Constant]
```

Το διαφορικό ως προς ένα τέτοιο σύμβολο (που δηλώνει σταθερά) φυσικά μηδενίζεται:

```
In[21]:= {Dt[f[z]], Dt[f[c, z]], Dt[f[x, z]], dg2 = Dt[g[x, y, z, c]]}
```

Out[21]= $\{0, 0, \mathrm{Dt}[x]\, f^{(1,0)}[x, z], \mathrm{Dt}[y]\, g^{(0,1,0,0)}[x, y, z, c] + \mathrm{Dt}[x]\, g^{(1,0,0,0)}[x, y, z, c]\}$

In[22]:= **Df = Dt[$c^2 + x^2 + y^2$ + Sin[c x y]]**

Out[22]= 2 x Dt[x] + 2 y Dt[y] + Cos[c x y] (c y Dt[x] + c x Dt[y])

Με τη χρήση της εντολής **Dt** μπορούμε να σχηματίσουμε και την πλήρη (ή ακριβή) διαφορική εξίσωση που έχει σαν λύση μια πεπλεγμένη συνάρτηση της μορφής $u(x, y) = c$. Παράδειγμα για μια περιφέρεια

In[23]:= **{Circumference = $x^2 + y^2$ == c, ODE = Dt[Circumference], ODE /. {Dt[x] → dx, Dt[y] → dy}}**

Out[23]= {$x^2 + y^2$ == c, 2 x Dt[x] + 2 y Dt[y] == 0, 2 dx x + 2 dy y == 0}

Και δεύτερο παράδειγμα για τις γραμμές ροής σε διδιάστατη μόνιμη ροή ιδεατού ρευστού ταχύτητας *U* στο άπειρο (ομοιόμορφη ροή) γύρω από κύλινδρο ακτίνας *a* (πιο καλά που παρεμποδίζεται από κύλινδρο). Πρώτα δίνεται η ροϊκή συνάρτηση (ή συνάρτηση ροής) $\Psi(x, y)$, αφού βέβαια η ταχύτητα *U* του ρευστού στο άπειρο και η ακτίνα του κυλίνδρου *a* υποτεθούν εδώ (σωστά!) και οι δυο τους σταθερές:

In[24]:= **SetAttributes[{U, a}, Constant]; Ψ[x_, y_] = U (y − y a^2 / ($x^2 + y^2$))**

Out[24]= $U\left(y - \frac{a^2 y}{x^2 + y^2}\right)$

Στη συνέχεια, παίρνοντας το διαφορικό της ροϊκής συναρτήσεως σχηματίζουμε τη σχετική μη γραμμική συνήθη διαφορική εξίσωση πρώτης τάξεως για τις γραμμές ροής. (Αυτή είναι προφανώς μια πλήρης ή ακριβής διαφορική εξίσωση.) Εδώ βέβαια η διαφορική εξίσωση είναι γραμμένη με τη χρήση διαφορικών:

```
In[25]:= FlowLinesDifferentialEquation1 = Dt[Ψ[x, y]] == 0 // FullSimplify
```

Out[25]= $U\left(\mathrm{Dt}[y] + \frac{a^2\,(2\,x\,y\,\mathrm{Dt}[x] + (-x^2 + y^2)\,\mathrm{Dt}[y])}{(x^2 + y^2)^2}\right) == 0$

Η ίδια εξίσωση μπορεί να γραφεί και σε απλούστερη μορφή με ορισμένες παραπέρα παρεμβάσεις μας:

```
In[26]:= FlowLinesDifferentialEquation2 =
          Numerator[Together[FlowLinesDifferentialEquation1[[1]] / U]] == 0 /.
            {Dt[x] → dx, Dt[y] → dy} // Simplify
```

$$\text{Out[26]= } dy\,(x^2+y^2)^2 + a^2\,(2\,dx\,x\,y + dy\,(-x^2+y^2)) == 0$$

Στη δεύτερη μορφή της η εντολή **Dt** βρίσκει την ολική παράγωγο (total derivative) της συναρτήσεως στο πρώτο όρισμα ως προς τη μεταβλητή στο δεύτερο όρισμα. (Υπενθυμίζεται ότι το *c* δηλώνει σταθερά.)

$$\text{In[27]:= } \{Dt[x^2 + y^2 + c\,x\,y,\ x],\ Dt[x^2 + y^2 + c\,x\,y,\ y]\}\ //\ Simplify$$

$$\text{Out[27]= } \{2\,x + c\,y + (c\,x + 2\,y)\,Dt[y,\ x],\ c\,x + 2\,y + (2\,x + c\,y)\,Dt[x,\ y]\}$$

Με την τρίτη μορφή της εντολής **Dt** υπολογίζονται ανάλογα και ολικές παράγωγοι ανωτέρας τάξεως

$$\text{In[28]:= } \{Dt[u^3 + v^3,\ \{u,\ 2\}],\ Dt[u^3 + v^3,\ \{u,\ 3\}]\}\ //\ Simplify$$

$$\text{Out[28]= } \{6\,u + 6\,v\,Dt[v,\ u]^2 + 3\,v^2\,Dt[v,\ \{u,\ 2\}],\ 3\,(2 + 2\,Dt[v,\ u]^3 + 6\,v\,Dt[v,\ u]\,Dt[v,\ \{u,\ 2\}] + v^2\,Dt[v,\ \{u,\ 3\}])\}$$

Τέλος ο γνωστός κανόνας της αλυσίδας (ή της αλληλουχίας των παραγώγων) σε μια απλή μορφή του:

```
In[29]:= Dt[f[u, v, w], t]
```

$$\text{Out[29]= } Dt[w,\ t]\,f^{(0,0,1)}[u,\ v,\ w] + Dt[v,\ t]\,f^{(0,1,0)}[u,\ v,\ w] + Dt[u,\ t]\,f^{(1,0,0)}[u,\ v,\ w]$$

■ ΜΙΑ ΕΦΑΡΜΟΓΗ ΣΤΗ ΡΕΥΣΤΟΜΗΧΑΝΙΚΗ

Η ολική παράγωγος είναι πολύ χρήσιμη στη Ρευστομηχανική, όπου καλείται και υλική παράγωγος (material derivative). Εδώ θεωρούμε μη μόνιμη ροή στις τρεις διαστάσεις (x, y, z). Επαναφέρουμε το σύμβολο z να δηλώνει μεταβλητή και υπενθυμίζουμε την ολική (ή υλική) παράγωγο μιας συναρτήσεως στη ροή αυτή:

```
In[30]:= ClearAttributes[z, Constant]; Dft = Dt[f[x, y, z, t], t]
```

$$\text{Out[30]= } f^{(0,0,0,1)}[x,\ y,\ z,\ t] + Dt[z,\ t]\,f^{(0,0,1,0)}[x,\ y,\ z,\ t] + Dt[y,\ t]\,f^{(0,1,0,0)}[x,\ y,\ z,\ t] + Dt[x,\ t]\,f^{(1,0,0,0)}[x,\ y,\ z,\ t]$$

Με τη χρήση των συνιστωσών της ταχύτητας **V**, που είναι φυσικά οι χρονικές παράγωγοι των (x, y, z):

```
In[31]:= VelocityComponents =
           {Dt[x, t] → u[x, y, z, t], Dt[y, t] → v[x, y, z, t], Dt[z, t] → w[x, y, z, t]};
```

η ίδια ολική (ή υλική) παράγωγος γράφεται και στην πολύ πιο κατανοητή μορφή της

```
In[32]:= Dft1 = Dft /. VelocityComponents
```

$$\text{Out[32]= } f^{(0,0,0,1)}[x,\ y,\ z,\ t] + w[x,\ y,\ z,\ t]\,f^{(0,0,1,0)}[x,\ y,\ z,\ t] + v[x,\ y,\ z,\ t]\,f^{(0,1,0,0)}[x,\ y,\ z,\ t] + u[x,\ y,\ z,\ t]\,f^{(1,0,0,0)}[x,\ y,\ z,\ t]$$

Προφανώς το διάνυσμα **V** της ταχύτητας του ρευστού (εδώ στις τρεις διαστάσεις) είναι το διάνυσμα

```
In[33]:= V[x_, y_, z_, t_] = {u[x, y, z, t], v[x, y, z, t], w[x, y, z, t]};
```

Καλούμε τώρα το πακέτο Διανυσματικής Αναλύσεως και ορίζουμε Καρτεσιανές συντεταγμένες (x, y, z):

```
In[34]:= Needs["Calculus`VectorAnalysis`"]
```

```
In[35]:= SetCoordinates[Cartesian[x, y, z]];
```

Παρατηρούμε ότι η παραπάνω ολική (ή υλική) παράγωγος μπορεί να γραφεί και στην ισοδύναμη μορφή

```
In[36]:= Dft1 == D[f[x, y, z, t], t] + V[x, y, z, t].Grad[f[x, y, z, t]]

Out[36]= True
```

Με χρήση της ολικής (ή υλικής) παραγώγου γράφουμε την εξίσωση της συνεχείας (με ρ την πυκνότητα):

```
In[37]:= ContinuityEquation =
          Dt[ρ[x, y, z, t], t] + ρ[x, y, z, t] Div[V[x, y, z, t], Cartesian[x, y, z]] == 0 /.
           VelocityComponents
```

Out[37]= $\rho^{(0,0,0,1)}[x, y, z, t] + w[x, y, z, t]\,\rho^{(0,0,1,0)}[x, y, z, t] + v[x, y, z, t]\,\rho^{(0,1,0,0)}[x, y, z, t] + \rho[x, y, z, t]\,(w^{(0,0,1,0)}[x, y, z, t] + v^{(0,1,0,0)}[x, y, z, t] + u^{(1,0,0,0)}[x, y, z, t]) + u[x, y, z, t]\,\rho^{(1,0,0,0)}[x, y, z, t] == 0$

■ ΕΝΤΟΛΗ Λ4: ΤΟΠΙΚΟ ΕΛΑΧΙΣΤΟ ΣΥΝΑΡΤΗΣΕΩΣ

FindMinimum[*Συνάρτηση*, {*Μεταβλητή*, *ΣημείοΕνάρξεως*}]

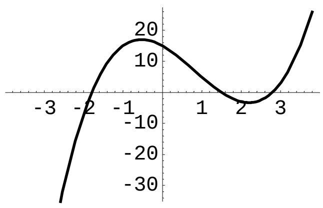
**FindMinimum[*Συνάρτηση*, {*Μεταβλητή*, *ΣημείοΕνάρξεως*,
ΑριστερόΆκροΔιαστήματος, *ΔεξιόΆκροΔιαστήματος*}]**

FindMinimum[*Συνάρτηση*, {*Μεταβλητή*, *ΑριστερόΣημείοΕνάρξεως*, *ΔεξιόΣημείοΕνάρξεως*}]

FindMinimum[*Συνάρτηση*, {*Μεταβλητή-1*, *ΣημείοΕνάρξεως-1*}, {*Μεταβλητή-2*, *ΣημείοΕνάρξεως-2*}]

Στην πρώτη μορφή της η εντολή αυτή υπολογίζει αριθμητικά ένα τοπικό ελάχιστο της συναρτήσεως η οποία καθορίζεται στο πρώτο όρισμα. Η μεταβλητή και το σημείο ενάρξεως της ευρέσεως του ελαχίστου καθορίζονται στη λίστα (με δύο στοιχεία) του δευτέρου ορίσματος. Πέρα από το τοπικό ελάχιστο η εντολή δίνει επίσης και το σημείο όπου παρουσιάζεται το ελάχιστο αυτό. Στη δεύτερη μορφή της ίδιας εντολής στη λίστα του δεύτερου ορίσματος καθορίζεται επίσης και το διάστημα $[a, b]$ μέσα στο οποίο θα γίνει ο έλεγχος του ελαχίστου. Στη συνέχεια στην τρίτη μορφή της ίδιας εντολής στο δεύτερο όρισμα (εκτός από τη μεταβλητή) δεν καθορίζεται ένα σημείο ενάρξεως, αλλά καθορίζονται δύο σημεία c και d ενάρξεως της διαδικασίας ευρέσεως του ελαχίστου. Η τρίτη αυτή μορφή είναι αυτή που απαιτείται να χρησιμοποιηθεί, εφόσον η συνάρτηση δε διαθέτει αναλυτική παράγωγο. Τέλος η ίδια ακριβώς εντολή μπορεί ασφαλώς να χρησιμοποιηθεί και για συναρτήσεις πολλών μεταβλητών, όπως φαίνεται στην τέταρτη μορφή της (εκεί όμως για δύο μόνο μεταβλητές). Παραδείγματα: Πρώτα μια απλή συνάρτηση μιας μεταβλητής (ένα τριτοβάθμιο πολυώνυμο) και η γραφική της παράσταση:

```
In[38]:= φ[x_] = 2 x^3 - 5 x^2 - 7 x + 15;
         Plot[φ[x], {x, -3.8, 3.8}, PlotStyle → Thickness[0.01], ImageSize → 156];
```



Εύρεση τώρα του τοπικού ελαχίστου με την παρούσα εντολή **FindMinimum** ξεκινώντας την αναζήτησή του από το σημείο $x = 0$, μια όχι και τόσο καλή αρχική προσέγγιση σύμφωνα με το παραπάνω σχήμα:

```
In[39]:= φm1 = FindMinimum[φ[x], {x, 0}]

Out[39]= {-3.30405, {x → 2.19756}}
```

Τελικά το τοπικό ελάχιστο βρέθηκε σωστά. Έχει την τιμή -3.30405 και είναι στο σημείο $x = 2.19756$:

```
In[40]:= {LocalMinimum = φm1[[1]], PointOfLocalMinimum = φm1[[2, 1, 2]]}

Out[40]= {-3.30405, 2.19756}
```

Φυσικά για το τοπικό μέγιστο της ίδιας συναρτήσεως $\varphi(x)$, αρκεί να υπολογισθεί το τοπικό ελάχιστο της αντίθετής της συναρτήσεως, δηλαδή της συναρτήσεως $-\varphi(x)$. Για να το κάνουμε κι αυτό τώρα:

```
In[41]:= φm2 = FindMinimum[-φ[x], {x, 0}]

Out[41]= {-17.0078, {x → -0.530892}}

In[42]:= {LocalMaximum = -φm2[[1]], PointOfLocalMaximum = φm2[[2, 1, 2]]}

Out[42]= {17.0078, -0.530892}
```

Τα αποτελέσματα αυτά συμφωνούν με το πιο πάνω σχήμα. Είναι προφανές ότι οι δύο πιο πάνω τιμές (για το τοπικό ελάχιστο και το τοπικό μέγιστο) μηδενίζουν την πρώτη παράγωγο $\varphi'(x)$ της συναρτήσεώς μας $\varphi(x)$. Αυτό είναι σωστό για μια παραγωγίσιμη συνάρτηση, όπως είναι το πολυώνυμο που εξετάζουμε:

```
In[43]:= Solve[φ'[x] == 0, x] // N

Out[43]= {{x → -0.530892}, {x → 2.19756}}
```

Αποπειρόμαστε τώρα να δηλώσουμε από την αρχή και το διάστημα μέσα στο οποίο υποψιαζόμαστε ότι κείται το τοπικό ελάχιστο της συναρτήσεως. Επιλέγουμε αρχικά το διάστημα $[-2, 2]$:

```
In[44]:= FindMinimum[φ[x], {x, 0, -2, 2}]

        FindMinimum::regex : Reached the point {2.55042} which is outside the region {{-2., 2.}}.

Out[44]= FindMinimum[φ[x], {x, 0, -2, 2}]
```

Το πιο πάνω διάστημα $[-2, 2]$ απέτυχε. Δεν υπάρχει τοπικό ελάχιστο εκεί. Αντίθετα το διάστημα $[-3, 3]$ είναι επιτυχές. Εκεί μέσα κείται το σημείο τοπικού ελαχίστου $x = 2.19756$:

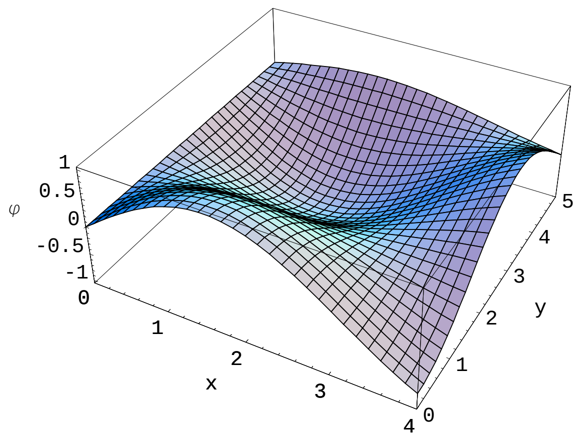
```
In[45]:= FindMinimum[φ[x], {x, 0, -3, 3}]

Out[45]= {-3.30405, {x → 2.19756}}
```

Με την τρίτη μορφή της εντολής δηλώνουμε στο δεύτερο όρισμα δύο σημεία (και όχι μόνο ένα) για την έναρξη της διαδικασίας προσδιορισμού του τοπικού ελαχίστου. Φυσικά προκύπτει ξανά το ίδιο ελάχιστο:

```
In[46]:= FindMinimum[φ[x], {x, 1, 2}]

Out[46]= {-3.30405, {x → 2.19756}}
```

Και ένα τελευταίο παράδειγμα: αυτό για μια συνάρτηση φ με δύο μεταβλητές: τη συνάρτηση $\sin x \cos y$ με γραφική παράσταση

```
In[47]:= Plot3D[Sin[x] Cos[y], {x, 0, 4}, {y, 0, 5},
          PlotPoints → 30, AxesLabel → {"x", "y", "φ  "}, ImageSize → 280];
```



η οποία έχει πραγματικά ένα τοπικό ελάχιστο στην παραπάνω ορθογωνική περιοχή $[0, 4]\times[0, 5]$. Αυτό το τοπικό ελάχιστο το προσδιορίζουμε και τώρα με την ίδια εντολή **FindMinimum**:

```
In[48]:= FindMinimum[Sin[x] Cos[y], {x, 1}, {y, 3}]

Out[48]= {-1., {x → 1.5708, y → 3.14159}}
```

■ ΕΝΤΟΛΗ Λ5: ΟΛΟΚΛΗΡΩΣΗ

Integrate[*Συνάρτηση, Μεταβλητή*]

Integrate[*Συνάρτηση*, {*Μεταβλητή, ΚάτωΌριοΟλοκληρώσεως, ΆνωΌριοΟλοκληρώσεως*}]

Integrate[*Συνάρτηση*, {*Μεταβλητή-1, ΚάτωΌριοΟλοκληρώσεως-1, ΆνωΌριοΟλοκληρώσεως-1*}, {*Μεταβλητή-2, ΚάτωΌριοΟλοκληρώσεως-2, ΆνωΌριοΟλοκληρώσεως-2*}, ...]

Υπολογίζει αναλυτικά (εφόσον τούτο είναι δυνατόν) το αόριστο ή το ορισμένο ολοκλήρωμα μιας συναρτήσεως, π.χ. το ολοκλήρωμα $\int_a^b f(x)\,dx$ στη μία διάσταση (απλό ολοκλήρωμα), στις δύο διαστάσεις (διπλό ολοκλήρωμα), στις τρεις διαστάσεις (τριπλό ολοκλήρωμα), κλπ. Παραδείγματα: Πρώτα απλά η δήλωση μερικών ολοκληρωμάτων (χωρίς υπολογισμούς τους):

```
In[49]:= {Integrate[f[x], x], Integrate[h[t], {t, a, b}], Integrate[ϕ[τ], {τ, 0, t}]}
```

Out[49]= $\left\{\int f[x]\,dx,\ \int_a^b h[t]\,dt,\ \int_0^t \phi[\tau]\,d\tau\right\}$

```
In[50]:= Integrate[f[x, y, z], {x, xmin, xmax}, {y, ymin, ymax}, {z, zmin, zmax}]
```

Out[50]= $\int_{x_{min}}^{x_{max}}\int_{y_{min}}^{y_{max}}\int_{z_{min}}^{z_{max}} f[x, y, z]\,dz\,dy\,dx$

Δύο μέτριας δυσκολίας αόριστα ολοκληρώματα με τις συναρτήσεις sinh x και erf x αντίστοιχα:

```
In[51]:= {Integrate[x^5 Sinh[x], x], Integrate[x Erf[x], x]} // Simplify
```

Out[51]= $\left\{x\,(120+20\,x^2+x^4)\,\mathrm{Cosh}[x]-5\,(24+12\,x^2+x^4)\,\mathrm{Sinh}[x],\ \frac{1}{4}\left(\frac{2\,e^{-x^2}\,x}{\sqrt{\pi}}+(-1+2\,x^2)\,\mathrm{Erf}[x]\right)\right\}$

Ένα ορισμένο ολοκλήρωμα που περιλαμβάνει τη συνάρτηση σφάλματος erf *x* και η αριθμητική τιμή του:

```
In[52]:= {i1 = Integrate[x^20 Erf[x], {x, 0, 1}] // Simplify, N[i1], N[i1, 30]}
```

Out[52]= $\left\{\frac{9864101 + e\left(-3628800 + \sqrt{\pi}\,\text{Erf}[1]\right)}{21\,e\,\sqrt{\pi}},\ 0.039149,\ 0.0391490327501211619198661036284\right\}$

Σε ορθογωνική πλάκα στην περιοχή $P = [0, a] \times [0, b]$ που καταπονείται από συγκεκριμένη κατανεμημένη κάθετη φόρτιση $p(x, y)$:

```
In[53]:= p[x_, y_] = Cos[(x / a + y / b)] (C x^2 + D x y + F y^2);
```

το συνολικό φορτίο **TotalPlateLoad** στην πλάκα θα είναι το ολοκλήρωμα της παραπάνω συναρτήσεως φορτίσεώς της $p(x, y)$ σε ολόκληρη την ορθογωνική πλάκα:

```
In[54]:= TotalPlateLoad = Integrate[p[x, y], {x, 0, a}, {y, 0, b}] // Simplify
```

Out[54]= $a\,b\,(-1 + 2\,\text{Cos}[1])\,(a\,b\,D\,(-1 + 2\,\text{Sin}[1]) + a^2\,C\,(-1 + \text{Cos}[1] + 2\,\text{Sin}[1]) + b^2\,F\,(-1 + \text{Cos}[1] + 2\,\text{Sin}[1]))$

Ήταν ένα πολύ δύσκολο στον υπολογισμό διπλό ολοκλήρωμα. Σε πιο αριθμητική μορφή η τιμή του είναι:

```
In[55]:= N[TotalPlateLoad]
```

Out[55]= $0.0806046\,a\,b\,(1.22324\,a^2\,C + 0.682942\,a\,b\,D + 1.22324\,b^2\,F)$

■ ΕΝΤΟΛΗ Λ6: ΑΡΙΘΜΗΤΙΚΗ ΟΛΟΚΛΗΡΩΣΗ

NIntegrate[*Συνάρτηση*, {*Μεταβλητή*, *ΚάτωΌριοΟλοκληρώσεως*, *ΆνωΌριοΟλοκληρώσεως*}]

NIntegrate[*Συνάρτηση*, {*Μεταβλητή-1*, *ΚάτωΌριοΟλοκληρώσεως-1*, *ΆνωΌριοΟλοκληρώσεως-1*}, *Μεταβλητή-2*, *ΚάτωΌριοΟλοκληρώσεως-2*, *ΆνωΌριοΟλοκληρώσεως-2*}, . . .]

Υπολογίζει αριθμητικά το ορισμένο ολοκλήρωμα μιας συναρτήσεως, π.χ. το ολοκλήρωμα $\int_a^b f(x)\,dx$, και αντίστοιχα ολοκληρώματα στις περισσότερες διαστάσεις (διπλά, τριπλά, κλπ.). Δεν επιχειρεί να υπολογίσει το ολοκλήρωμα αναλυτικά και στη συνέχεια να προσδιορίσει την αριθμητική τιμή του. Ξεκινάει από την αρχή αριθμητικά με τη χρήση γνωστών και αποτελεσματικών μεθόδων αριθμητικής ολοκληρώσεως στην Αριθμητική Ανάλυση με έμφαση στην αρκετά γνωστή μέθοδο Gauss–Kronrod για τα μονοδιάστατα ορισμένα ολοκληρώματα. Εντούτοις σε ειδικές περιπτώσεις χρησιμοποιούνται και άλλες κατάλληλες μέθοδοι αριθμητικής ολοκληρώσεως. Παραδείγματα:

Σαν πρώτο παράδειγμα θεωρούμε το ορισμένο ολοκλήρωμα

```
In[56]:= i1a = Integrate[Cosh[Cosh[x]], {x, 0, 1}]
```

Out[56]= $\int_0^1 \text{Cosh}[\text{Cosh}[x]]\,dx$

που δε μπορεί να υπολογισθεί αναλυτικά με κανέναν τρόπο. Η αριθμητική τιμή του είναι

```
In[57]:= i1b = N[i1a, 80]

Out[57]= 1.7973014338206229139158534583827084172862255596647603158786891448089498129301071
```

Τούτη η τιμή υπολογίσθηκε με την παρούσα εντολή αριθμητικής ολοκληρώσεως **NIntegrate**, την οποία κάλεσε μόνη της η εντολή αριθμητικού υπολογισμού **N**, γιατί δε μπόρεσε να βρεθεί αναλυτική έκφραση του ολοκληρώματος αυτού. Αυτό μπορεί βέβαια να γίνει κι απευθείας με την παρούσα εντολή **NIntegrate**:

```
In[58]:= i1c = NIntegrate[Cosh[Cosh[x]], {x, 0, 1}]

Out[58]= 1.7973
```

Μπορούμε φυσικά να αυξήσουμε την ακρίβεια στους υπολογισμούς που γίνονται στον υπολογιστή με τη χρήση της επιλογής **WorkingPrecision** με αρχική τιμή της το 16, δηλαδή τη συνήθη ακρίβεια εκτελέσεως αριθμητικών πράξεων στον υπολογιστή. Αυτό φαίνεται από τις επιλογές της εντολής **NIntegrate**:

```
In[59]:= Options[NIntegrate]

Out[59]= {AccuracyGoal → ∞, Compiled → True, GaussPoints → Automatic,
          MaxPoints → Automatic, MaxRecursion → 6, Method → Automatic, MinRecursion → 0,
          PrecisionGoal → Automatic, SingularityDepth → 4, WorkingPrecision → 16}
```

Για παράδειγμα, εδώ θέτουμε **WorkingPrecision → 80** (αντί μόνο 16) για το ίδιο ολοκλήρωμα:

```
In[60]:= i1d = NIntegrate[Cosh[Cosh[x]], {x, 0, 1}, WorkingPrecision → 80]

Out[60]= 1.7973014338206229139158534583827084172862255596647603158786891448089498129301071
```

Και φυσικά έχουμε

```
In[61]:= i1b - i1d
```

Out[61]= $0. \times 10^{-80}$

Αντίθετα, εάν ένα ολοκλήρωμα μπορεί να υπολογισθεί αναλυτικά, όπως είναι το ολοκλήρωμα

```
In[62]:= i2a = Integrate[Exp[x] Erf[x] Cosh[x], {x, 0, 1}] // Simplify
```

Out[62]= $$\frac{1}{4}\left(\frac{-2+\frac{2}{e}}{\sqrt{\pi}}+\left(2-e+e^{2}\right)\mathrm{Erf}[1]\right)$$

τότε η εντολή

```
In[63]:= i2b = N[Integrate[Exp[x] Erf[x] Cosh[x], {x, 0, 1}], 53]

Out[63]= 1.2270487744100148460724866531312701615428241177135480
```

δεν καλεί την παρούσα αριθμητική εντολή **NIntegrate**. Απλά χρησιμοποιεί την αναλυτική έκφραση του ολοκληρώματος. Η εντολή **NIntegrate** μπορεί κι αυτή να χρησιμοποιηθεί από την αρχή, αν το θέλουμε:

```
In[64]:= i2c = NIntegrate[Exp[x] Erf[x] Cosh[x], {x, 0, 1}, WorkingPrecision → 60]

Out[64]= 1.2270487744100148460724866531312701615428241177135480
```

Φυσικά με την εντολή **NIntegrate** μπορούν να υπολογισθούν και διπλά και τριπλά ολοκληρώματα, όπως το πιο κάτω διπλό ολοκλήρωμα στην ορθογωνική περιοχή $R = [0, 1]\times[0, 2]$ (χωρίς αναλυτική έκφραση):

In[65]:= {**i3a** = **Integrate**[**Exp**[(**x** + **y**) (x^2 + y^2)] **Cosh**[**x y**], {**x, 0, 1**}, {**y, 0, 2**}], **i3b** = **N**[**i3a**]}

Out[65]= $$\left\{\int_0^1\int_0^2 e^{(x+y)\,(x^2+y^2)}\,\mathrm{Cosh}[x\,y]\,dy\,dx,\ 62186.1\right\}$$

■ ΕΝΤΟΛΗ Λ7: ΣΕΙΡΑ TAYLOR

Series[*Συνάρτηση*, {*Μεταβλητή*, *ΣημείοΜεΒάσηΤοΟποίοΒρίσκεταιΗΔυναμοσειρά*, *ΜέγιστηΔύναμη*}]

Υπολογίζει τη σειρά Taylor (ειδική περίπτωσή της είναι η σειρά Maclaurin γύρω από το σημείο 0) μιας επαρκώς παραγωγίσιμης συναρτήσεως ως προς μία μεταβλητή γύρω από ένα συγκεκριμένο σημείο και με όρους μέχρι μια συγκεκριμένη μέγιστη δύναμη (και όρο υπολοίπου στο τέλος). Παραδείγματα:

Η σειρά Maclaurin (σειρά Taylor για $x = 0$) μιας αυθαίρετης παραγωγίσιμης συναρτήσεως $f(x)$ με όρους μέχρι και x^5. (Στο τέλος της σειράς εμφανίζεται βέβαια και ο όρος υπολοίπου της.)

```
In[66]:= Series[f[x], {x, 0, 5}]
```

Out[66]= $f[0] + f'[0]\,x + \frac{1}{2} f''[0]\,x^2 + \frac{1}{6} f^{(3)}[0]\,x^3 + \frac{1}{24} f^{(4)}[0]\,x^4 + \frac{1}{120} f^{(5)}[0]\,x^5 + O[x]^6$

Ανάλογα η σειρά Taylor στο σημείο $x = x_0$ μιας αυθαίρετης παραγωγίσιμης συναρτήσεως $f(x)$ με όρους μέχρι και $(x - x_0)^3$. (Στο τέλος της σειράς εμφανίζεται και πάλι ο όρος υπολοίπου της.)

```
In[67]:= Series[f[x], {x, x0, 3}]
```

Out[67]= $f[x0] + f'[x0]\,(x - x0) + \frac{1}{2} f''[x0]\,(x - x0)^2 + \frac{1}{6} f^{(3)}[x0]\,(x - x0)^3 + O[x - x0]^4$

Οι σειρές Maclaurin (Taylor για $x = 0$) για τις συναρτήσεις $\cosh ax$ και $\cos ax$ με όρους μέχρι και x^{14}:

```
In[68]:= {Series[Cosh[a x], {x, 0, 14}], Series[Cos[a x], {x, 0, 14}]}
```

Out[68]= $\Big\{1 + \frac{a^2 x^2}{2} + \frac{a^4 x^4}{24} + \frac{a^6 x^6}{720} + \frac{a^8 x^8}{40320} + \frac{a^{10} x^{10}}{3628800} + \frac{a^{12} x^{12}}{479001600} + \frac{a^{14} x^{14}}{87178291200} + O[x]^{15},$
$1 - \frac{a^2 x^2}{2} + \frac{a^4 x^4}{24} - \frac{a^6 x^6}{720} + \frac{a^8 x^8}{40320} - \frac{a^{10} x^{10}}{3628800} + \frac{a^{12} x^{12}}{479001600} - \frac{a^{14} x^{14}}{87178291200} + O[x]^{15}\Big\}$

Και τώρα η πολύ δυσκολότερη στον υπολογισμό της σειρά Maclaurin (Taylor για $x = 0$) για τη συνάρτηση $\tan bx$, εδώ με όρους μέχρι και x^{24}:

```
In[69]:= Series[Tan[b x], {x, 0, 24}]
```

Out[69]= $b\,x + \frac{b^3 x^3}{3} + \frac{2 b^5 x^5}{15} + \frac{17 b^7 x^7}{315} + \frac{62 b^9 x^9}{2835} + \frac{1382 b^{11} x^{11}}{155925} + \frac{21844 b^{13} x^{13}}{6081075} + \frac{929569 b^{15} x^{15}}{638512875} +$
$\frac{6404582 b^{17} x^{17}}{10854718875} + \frac{443861162 b^{19} x^{19}}{1856156927625} + \frac{18888466084 b^{21} x^{21}}{194896477400625} + \frac{113927491862 b^{23} x^{23}}{2900518163668125} + O[x]^{25}$

■ ΕΝΤΟΛΗ Λ8: ΜΕΤΑΤΡΟΠΗ ΣΕΙΡΑΣ TAYLOR ΣΕ ΠΟΛΥΩΝΥΜΟ

Normal[*ΣειράTaylor*]

Μετατρέπει μια σειρά Taylor σε πολυώνυμο αποκόπτοντας τον όρο που δεν αναγράφεται σαφώς, δηλαδή παραλείποντας τον *O*-όρο υπολοίπου στο τέλος της σειράς Taylor. Έτσι η κανονικοποιημένη αυτή σειρά Taylor *(μονάχα αυτή!)* μπορεί να χρησιμοποιηθεί σε γραφικές παραστάσεις. Παραδείγματα:

Υπολογίζουμε καταρχήν τη σειρά Maclaurin (σειρά Taylor για $x = 0$) **s1** της συναρτήσεως $\sin ax$ με όρους μέχρι και x^7. Στη συνέχεια με εφαρμογή της παρούσας εντολής **Normal** τροποποιούμε τη σειρά αυτή, ώστε να φύγει εντελώς ο *O*-όρος υπολοίπου της. Παίρνουμε έτσι την αντίστοιχη πιο εύχρηστη σειρά **s2**:

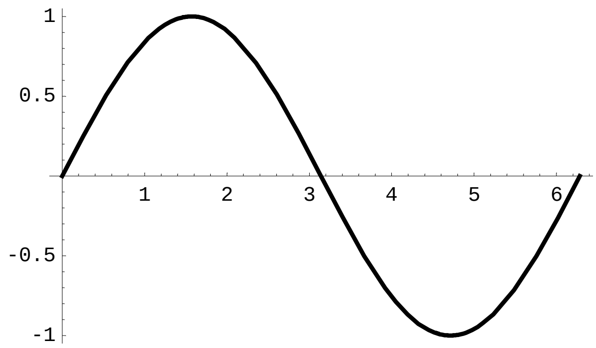
```
In[70]:= {s1 = Series[Sin[a x], {x, 0, 7}], s2 = Normal[s1], Normal[s1] == s1 // Normal}
```

Out[70]= $\left\{a\,x - \frac{a^3 x^3}{6} + \frac{a^5 x^5}{120} - \frac{a^7 x^7}{5040} + O[x]^8,\ a\,x - \frac{a^3 x^3}{6} + \frac{a^5 x^5}{120} - \frac{a^7 x^7}{5040},\ \mathrm{True}\right\}$

Επιχειρούμε να συγκρίνουμε τη συνάρτησή μας με τη σειρά Maclaurin **s1** που υπολογίσαμε σχεδιάζοντάς τις και τις δύο στο ίδιο σχήμα (για $a = 1$). Δυστυχώς η σχεδίαση της σειράς Maclaurin **s1** αποτυγχάνει απλά εξαιτίας του *O*-όρου υπολοίπου στο τέλος της. (Εμφανίζεται και σωρεία λαθών στην οθόνη ...)

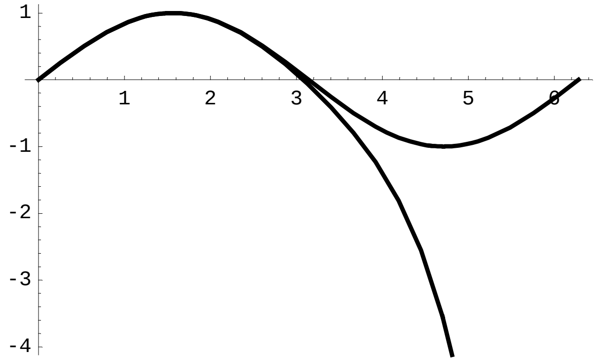
```
In[71]:= Plot[{Sin[x], s1 /. a → 1}, {x, 0, 2 π}, PlotStyle → Thickness[0.008]];
```

```
SeriesData::ssdn :
 Attempt to evaluate a series at the number 2.617993877991494`*^-7; returning Indeterminate.

SeriesData::ssdn :
 Attempt to evaluate a series at the number 2.617993877991494`*^-7; returning Indeterminate.

Plot::plnr : s1 /. a → 1 is not a machine-size real number at x = 2.617993877991494`*^-7.

SeriesData::ssdn :
 Attempt to evaluate a series at the number 0.25488992540742256`; returning Indeterminate.

General::stop : Further output of SeriesData::ssdn will be suppressed during this calculation.

Plot::plnr : s1 /. a → 1 is not a machine-size real number at x = 0.25488992540742256`.

Plot::plnr : s1 /. a → 1 is not a machine-size real number at x = 0.5328694051959509`.

General::stop : Further output of Plot::plnr will be suppressed during this calculation.
```



Αν όμως χρησιμοποιήσουμε την ουσιαστικά ισοδύναμη σειρά **s2** (χωρίς τον όρο υπολοίπου στο τέλος), τότε πραγματικά παίρνουμε το σχήμα που θέλουμε. Ελέγχουμε έτσι και την ακρίβεια του αναπτύγματος σε σειρά Maclaurin (ή γενικότερα σειρά Taylor) στο παράδειγμά μας συγκρίνοντάς το με τη συνάρτηση. Από το παρακάτω σχήμα παρατηρούμε αρκετά καλή σύμπτωση μέχρι περίπου το σημείο $x = 3$. (Μετά όχι!)

```
In[72]:= Plot[{Sin[x], s2 /. a → 1}, {x, 0, 2 π}, PlotStyle → Thickness[0.008]];
```



■ *Notebook E7*

ΕΝΤΟΛΕΣ ΓΙΑ ΛΙΣΤΕΣ, ΔΙΑΝΥΣΜΑΤΑ ΚΑΙ ΣΥΝΟΛΑ

28 ΕΝΤΟΛΕΣ: **L1. List, L2. Table, L3. TableForm, L4. Max, L5. Min, L6. Mean, L7. StandardDeviation, L8. Variance, L9. ListQ, L10. VectorQ, L11. MemberQ, L12. Length, L13. First, L14. Last, L15. Rest, L16. Part, L17. Reverse, L18. Prepend, L19. Append, L20. Partition, L21. Dot, L22. Flatten, L23. Sort, L24. Count, L25. Join, L26. Union, L27. Intersection, L28. Complement,**

■ ΕΝΤΟΛΗ L1: ΑΜΕΣΗ ΔΗΜΙΟΥΡΓΙΑ ΛΙΣΤΑΣ

List[*Στοιχείο-1, Στοιχείο-2, ..., Στοιχείο-n*] ή **{*Στοιχείο-1, Στοιχείο-2, ..., Στοιχείο-n*}**

Δημιουργεί τη λίστα των στοιχείων που δίνονται. Ακριβώς το ίδιο και για ένα διάνυσμα. *(Πάλι η ίδια εντολή!)* Σχεδόν πάντα προτιμάται η δεύτερη μορφή της, που είναι και απλούστερη. Παραδείγματα:

```
In[1]:= {L1 = {a, b, c, d}, FullForm[L1], L2 = List[a, b, c, d], L1 == L2}

Out[1]= {{a, b, c, d}, List[a, b, c, d], {a, b, c, d}, True}
```

■ ΕΝΤΟΛΗ L2: ΔΗΜΙΟΥΡΓΙΑ ΛΙΣΤΑΣ ΜΕ ΤΗ ΧΡΗΣΗ ΔΕΙΚΤΗ / ΔΕΙΚΤΩΝ

Table[*Παράσταση*, {*ΑριθμόςΑντιγράφωνΤηςΠαραστάσεωςΣτηΛίσταΠουΘαΔημιουργηθεί*}]

Table[*Παράσταση*, {*Δείκτης, ΑρχικήΤιμήΤουΔείκτη, ΤελικήΤιμήΤουΔείκτη*}]

Table[*Παράσταση*, {*Δείκτης, ΑρχικήΤιμήΤουΔείκτη, ΤελικήΤιμήΤουΔείκτη, ΒήμαΜεταβολής*}]

Table[*Παράσταση*, {*ΠρώτοςΔείκτης, ΑρχικήΤιμήΤουΠρώτουΔείκτη, ΤελικήΤιμήΤουΠρώτουΔείκτη*}, {*ΔεύτεροςΔείκτης, ΑρχικήΤιμήΤουΔεύτερουΔείκτη, ΤελικήΤιμήΤουΔεύτερουΔείκτη*}]

Με την εντολή αυτή δημιουργούνται εύκολα λίστες με βάση την παράσταση στο πρώτο όρισμα της εντολής. Η πρώτη μορφή της εντολής απλά δημιουργεί λίστα με *n* αντίγραφα της παραστάσεως με τον αριθμό *n* των αντιγράφων να δηλώνεται σαν το μοναδικό στοιχείο της λίστας του δεύτερου ορίσματος της εντολής. Παραδείγματα:

```
In[2]:= list1 = Table[a, {20}]

Out[2]= {a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a, a}

In[3]:= list2 = Table[{a, b}, {10}]

Out[3]= {{a, b}, {a, b}, {a, b}, {a, b}, {a, b}, {a, b}, {a, b}, {a, b}, {a, b}, {a, b}}
```

Στη δεύτερη μορφή της εντολής **Table** υπάρχει συγκεκριμένος δείκτης, π.χ. *k* ή *n*, που παίρνει όλες τις ακέραιες τιμές από την αρχική τιμή του μέχρι την τελική τιμή του ανά ένα και συνήθως η παράσταση του πρώτου ορίσματος της εντολής εξαρτάται από το δείκτη αυτό. (Τούτη η εργασία μας θυμίζει έντονα την εντολή **Do**, που υπάρχει στις κλασικές γλώσσες προγραμματισμού, αλλά και στη *Mathematica*.) Έτσι κατασκευάζουμε πολύ εύκολα λίστες χωρίς να ταλαιπωρούμαστε με τον άμεσο τρόπο δηλώσεώς τους με την πρώτη εντολή **List** (αυτή συνήθως απλά με άγκιστρα). Σημειώνουμε επίσης στο σημείο αυτό ότι αν η αρχική τιμή του δείκτη είναι το 1, τότε η αρχική αυτή τιμή (1) μπορεί να παραλειφθεί. Παραδείγματα:

In[4]:= **tb1 = Table[x^k, {k, 1, 19}]**

Out[4]= $\{x, x^2, x^3, x^4, x^5, x^6, x^7, x^8, x^9, x^{10}, x^{11}, x^{12}, x^{13}, x^{14}, x^{15}, x^{16}, x^{17}, x^{18}, x^{19}\}$

In[5]:= **{tb2 = Table[x^k, {k, 19}], tb1 == tb2}**

Out[5]= $\{\{x, x^2, x^3, x^4, x^5, x^6, x^7, x^8, x^9, x^{10}, x^{11}, x^{12}, x^{13}, x^{14}, x^{15}, x^{16}, x^{17}, x^{18}, x^{19}\}, \text{True}\}$

In[6]:= **Table[Integrate[x Cosh[k x], {x, 0, 1}] // N, {k, 1, 9}]**

Out[6]= {0.632121, 1.12288, 2.33177, 5.17821, 11.9122, 28.0434, 67.1611, 163.037, 400.165}

In[7]:= **Table[ChebyshevT[n, x], {n, 0, 6}]**

Out[7]= $\{1, x, -1+2x^2, -3x+4x^3, 1-8x^2+8x^4, 5x-20x^3+16x^5, -1+18x^2-48x^4+32x^6\}$

In[8]:= **Table[ChebyshevU[n, x], {n, 0, 6}]**

Out[8]= $\{1, 2x, -1+4x^2, -4x+8x^3, 1-12x^2+16x^4, 6x-32x^3+32x^5, -1+24x^2-80x^4+64x^6\}$

Η τρίτη μορφή της ίδιας εντολής **Table** διαφέρει από τη δεύτερη μορφή της απλά στο ότι το βήμα μεταβολής του δείκτη μπορεί να μην είναι το ένα. Τώρα το βήμα αυτό δηλώνεται ρητά σαν τέταρτο στοιχείο της λίστας του δεύτερου ορίσματος της εντολής. Παραδείγματα:

In[9]:= **Table[x^k / k!, {k, 0, 14, 2}]**

Out[9]= $\left\{1, \frac{x^2}{2}, \frac{x^4}{24}, \frac{x^6}{720}, \frac{x^8}{40320}, \frac{x^{10}}{3628800}, \frac{x^{12}}{479001600}, \frac{x^{14}}{87178291200}\right\}$

In[10]:= **Table[Cos[k π / 10], {k, 0, 24, 3}] // N**

Out[10]= {1., 0.587785, −0.309017, −0.951057, −0.809017, 0., 0.809017, 0.951057, 0.309017}

Τέλος η τέταρτη μορφή της ίδιας εντολής **Table** επιτρέπει την κατασκευή πίνακα, δηλαδή λίστας με στοιχεία πάλι λίστες, γιατί διαθέτει δύο δείκτες, π.χ. τους δείκτες *k* και *m*, στο δεύτερο και στο τρίτο όρισμά της. (Μπορεί να υπάρχει και τέταρτο όρισμα, αν χρειάζεται με την προσθήκη ενός ακόμη δείκτη.) Παράδειγμα:

In[11]:= **tb3 = Table[1 / (k + m − 1), {k, 1, 3}, {m, 1, 3}]**

Out[11]= $\left\{\left\{1, \frac{1}{2}, \frac{1}{3}\right\}, \left\{\frac{1}{2}, \frac{1}{3}, \frac{1}{4}\right\}, \left\{\frac{1}{3}, \frac{1}{4}, \frac{1}{5}\right\}\right\}$

Η εμφάνιση αυτή του τελευταίου αποτελέσματος μπορεί να βελτιωθεί σημαντικά με τη χρήση της εντολής **MatrixForm**, που θα την αναφέρουμε στο αμέσως επόμενο Notebook E8 (εντολή M4) ως εξής:

In[12]:= **tb3 // MatrixForm**

Out[12]//MatrixForm=

$$\begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \end{pmatrix}$$

Επ' ευκαιρία σημειώνουμε και την εξής σχετική εντολή (χωρίς όμως την εμφάνιση των παρενθέσεων):

■ ΕΝΤΟΛΗ L3: ΕΜΦΑΝΙΣΗ ΣΕ ΜΟΡΦΗ ΠΙΝΑΚΑ

TableForm[*Πίνακας*] ή ***Πίνακας*//TableForm**

Η εντολή αυτή μετατρέπει έναν πίνακα από μονοδιάστατη (σε μια γραμμή) σε διδιάστατη μορφή, που είναι πολύ πιο καθαρή και εύκολη στην κατανόησή της. Αυτή είναι και η σωστή μορφή παρουσιάσεως ενός πίνακα (γενικού πίνακα, όχι μητρώου της Γραμμικής Άλγεβρας!). Παραδείγματα:

```
In[13]:= tb3 // TableForm
```

Out[13]//TableForm=

1	$\frac{1}{2}$	$\frac{1}{3}$
$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{4}$
$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{5}$

```
In[14]:= tb4 = {{a, 10, 105, 0.82}, {b, 20, 124, 0.65}, {c, 12, 132, 0.71}} // TableForm
```

Out[14]//TableForm=

a	10	105	0.82
b	20	124	0.65
c	12	132	0.71

■ ΕΝΤΟΛΗ L4: ΜΕΓΙΣΤΟΣ ΑΡΙΘΜΟΣ

Max[*ΛίσταΑριθμών*]

Max[*Αριθμός-1, Αριθμός-2, ...*]

Υπολογίζει το μεγαλύτερο αριθμό σε μια λίστα αριθμών (πρώτη μορφή) ή απλά το μεγαλύτερο από τα ορίσματά της (δεύτερη μορφή). Τα ορίσματα αυτά πρέπει να είναι αριθμοί και όχι αλγεβρικά σύμβολα. Παραδείγματα στην επόμενη εντολή **Min**.

■ ΕΝΤΟΛΗ L5: ΕΛΑΧΙΣΤΟΣ ΑΡΙΘΜΟΣ

Min[*ΛίσταΑριθμών*]

Min[*Αριθμός-1, Αριθμός-2, ...*]

Υπολογίζει το μικρότερο αριθμό σε μια λίστα αριθμών (πρώτη μορφή) ή απλά το μικρότερο από τα ορίσματά της (δεύτερη μορφή). Τα ορίσματα αυτά πρέπει να είναι αριθμοί και όχι αλγεβρικά σύμβολα. Παραδείγματα των εντολών **Max** και **Min**:

```
In[15]:= ListOfNumbers = {5, -10, 20, -7, 15, -100, 10*10, -5*6*7 (2+3)^2, Cosh[1]}

Out[15]= {5, -10, 20, -7, 15, -100, 100, -5250, Cosh[1]}

In[16]:= {MaximumNumber = Max[ListOfNumbers], MinimumNumber = Min[ListOfNumbers]}

Out[16]= {100, -5250}
```

ή πιο απλά

```
In[17]:= Max[{5, -10, 20, -7, 15, -100, 10*10, -5*6*7 (2+3)^2, Cosh[1]}]

Out[17]= 100

In[18]:= Min[{5, -10, 20, -7, 15, -100, 10*10, -5*6*7 (2+3)^2, Cosh[1]}]

Out[18]= -5250
```

ακόμη και χωρίς καθόλου λίστα: απλοί αριθμοί σαν ορίσματα:

```
In[19]:= Max[5, -10, 20, -7, 15, -100, 10*10, -5*6*7 (2+3)^2, Cosh[1]]

Out[19]= 100

In[20]:= Min[5, -10, 20, -7, 15, -100, 10*10, -5*6*7 (2+3)^2, Cosh[1]]

Out[20]= -5250
```

Φυσικά η λίστα των αριθμών μπορεί να δημιουργηθεί με την εντολή **Table**

```
In[21]:= tb = Table[Sin[k], {k, 1, 10}]

Out[21]= {Sin[1], Sin[2], Sin[3], Sin[4], Sin[5], Sin[6], Sin[7], Sin[8], Sin[9], Sin[10]}

In[22]:= {{Max[tb], Min[tb]}, {Max[tb], Min[tb]} // N}

Out[22]= {{Sin[8], Sin[5]}, {0.989358, -0.958924}}
```

Αν τολμήσουμε να χρησιμοποιήσουμε αλγεβρικά σύμβολα χωρίς καθορισμένες τιμές, τότε βέβαια δε θα μπορέσουμε να πάρουμε κάποιο συγκεκριμένο μέγιστο παρά τις φιλότιμες προσπάθειες της *Mathematica* μέχρι το βαθμό που μπορεί:

```
In[23]:= {Max[a, b], Max[1, a], Max[-3, -2, -1, 0, 1, 2, 3, a]}

Out[23]= {Max[a, b], Max[1, a], Max[3, a]}

In[24]:= {minimum = Min[ Table[Sin[k], {k, 1, 1000}], {a, b, c}], N[minimum, 30]}

Out[24]= {Min[a, b, c, Sin[344]], Min[-0.999990339506170900963215766381, a, b, c]}
```

■ ΕΝΤΟΛΗ L6: ΜΕΣΗ ΤΙΜΗ

Mean[*ΛίσταΑριθμητικώνΔεδομένων*]

Υπολογίζει τη μέση τιμή των αριθμητικών δεδομένων στη λίστα ***ΛίσταΑριθμητικώνΔεδομένων***. Για τη χρήση της εντολής αυτής καθώς και των δύο επόμενων εντολών **StandardDeviation** και **Variance** θα πρέπει να έχει προηγουμένως φορτωθεί το πακέτο Στατιστικής **Statistics‘DescriptiveStatistics‘**. Παράδειγμα: Καταρχήν φορτώνεται το πακέτο **Statistics‘DescriptiveStatistics‘**:

```
In[25]:= Needs["Statistics`DescriptiveStatistics`"]
```

Στη συνέχεια εισάγονται τα δεδομένα **data** και, εάν θέλουμε, υπολογίζεται και ο αριθμός τους *n*

```
In[26]:= data = {10, 12, 7, 15, 11, 8, 19, 12, 11, 14, 13, 15, 9, 19, 20, 10, 9, 16, 9, 12};
```

```
In[27]:= n = Length[data]

Out[27]= 20
```

Τώρα υπολογίζεται η μέση τιμή μ των δεδομένων αυτών με τη χρήση της παρούσας εντολής:

```
In[28]:= {μ = Mean[data], N[μ], N[μ, 50]}
```

Out[28]= $\left\{\frac{251}{20}, 12.55, 12.550000000000000000000000000000000000000000000000\right\}$

Εναλλακτικά αυτή θα μπορούσε βέβαια να είχε υπολογισθεί και απευθείας με τη χρήση αθροίσματος:

```
In[29]:= {μ1 = (1 / n) Sum[data[[k]], {k, 1, n}], μ1 == μ}
```

Out[29]= $\left\{\frac{251}{20}, \text{True}\right\}$

■ ΕΝΤΟΛΗ L7: ΤΥΠΙΚΗ ΑΠΟΚΛΙΣΗ

StandardDeviation[*ΛίσταΑριθμητικώνΔεδομένων*]

Υπολογίζει την αμερόληπτη εκτιμήτρια της τυπικής αποκλίσεως (διαίρεση με $n-1$ κι όχι με n) των αριθμητικών δεδομένων στη λίστα ***ΛίσταΑριθμητικώνΔεδομένων***. Για τη χρήση και της εντολής αυτής πρέπει να έχει φορτωθεί το πακέτο **Statistics‘DescriptiveStatistics‘**. Παράδειγμα: Υπολογισμός της τυπικής αποκλίσεως σ των δεδομένων **data** με τη χρήση της παρούσας εντολής:

```
In[30]:= {σ = StandardDeviation[data], N[σ], N[σ, 50]}
```

Out[30]= $\left\{\frac{\sqrt{\frac{5459}{95}}}{2}, 3.79022, 3.7902228791568722900017063295123747094398224904636\right\}$

Εναλλακτικά αυτή θα μπορούσε βέβαια να είχε υπολογισθεί και κατευθείαν με τη χρήση αθροίσματος:

```
In[31]:= {σ1 = Sqrt[(1 / (n - 1)) Sum[(data[[k]] - μ)^2, {k, 1, n}]], σ1 == σ}
```

Out[31]= $\left\{\frac{\sqrt{\frac{5459}{95}}}{2}, \text{True}\right\}$

■ ΕΝΤΟΛΗ L8: ΔΙΑΣΠΟΡΑ

Variance[*ΛίσταΑριθμητικώνΔεδομένων*]

Υπολογίζει την αμερόληπτη εκτιμήτρια της διασποράς (διαίρεση με $n-1$ κι όχι με n) των αριθμητικών δεδομένων στη λίστα ***ΛίσταΑριθμητικώνΔεδομένων***. Για τη χρήση και της εντολής αυτής (όπως και των δύο προηγούμενων εντολών **Mean** και **StandardDeviation**) πρέπει να έχει προηγουμένως φορτωθεί το πακέτο **Statistics‘DescriptiveStatistics‘**. Παράδειγμα: Υπολογισμός της διασποράς των δεδομένων **data** με τη χρήση της εντολής:

```
In[32]:= {var = Variance[data], N[var], N[var, 50]}
```

Out[32]= $\left\{\frac{5459}{380}, 14.3658, 14.365789473684210526315789473684210526315789473684\right\}$

Εναλλακτικά αυτή θα μπορούσε βέβαια να είχε υπολογισθεί και απευθείας με τη χρήση αθροίσματος:

```
In[33]:= {var1 = (1 / (n - 1)) Sum[(data[[k]] - μ)^2, {k, 1, n}], var1 == var}
```

Out[33]= $\left\{\frac{5459}{380}, \text{True}\right\}$

Προφανώς ισχύουν οι εξής σχέσεις που συνδέουν την τυπική απόκλιση με τη διασπορά:

```
In[34]:= {var == σ^2, σ == Sqrt[var]}
```

```
Out[34]= {True, True}
```

Επίσης ισχύει και ο ακόλουθος τρόπος άμεσου υπολογισμού της διασποράς, όμως με ήδη γνωστή τη μέση τιμή. (Υπενθυμίζεται και πάλι ότι εδώ τόσο η τυπική απόκλιση όσο και η διασπορά υπολογίζονται με διαίρεση του αθροίσματος με $n-1$ και όχι με n. Τούτο οδηγεί σε σχετικές αμερόληπτες εκτιμήτριες.)

```
In[35]:= {var2 = (1 / (n - 1)) Sum[data[[k]]^2, {k, 1, n}] - (n / (n - 1)) μ^2, var1 == var2 == var}
```

Out[35]= $\left\{\frac{5459}{380}, \text{True}\right\}$

■ ΕΝΤΟΛΗ L9: ΕΛΕΓΧΟΣ ΠΑΡΑΣΤΑΣΕΩΣ ΑΝ ΕΙΝΑΙ ΛΙΣΤΑ

ListQ[*Παράσταση*]

Ελέγχει αν μια παράσταση είναι λίστα ή όχι. Παραδείγματα στην αμέσως επόμενη εντολή.

■ ΕΝΤΟΛΗ L10: ΕΛΕΓΧΟΣ ΠΑΡΑΣΤΑΣΕΩΣ ΑΝ ΕΙΝΑΙ ΔΙΑΝΥΣΜΑ

VectorQ[*Παράσταση*]

Ελέγχει αν μια παράσταση είναι διάνυσμα *(όχι γενικά λίστα: μόνο διάνυσμα!)* ή όχι. Παραδείγματα:

```
In[36]:= {VectorQ[V], ListQ[a], VectorQ[{a}], F = {a, {c, d}}, ListQ[F], VectorQ[F]}

Out[36]= {False, False, True, {a, {c, d}}, True, False}
```

■ ΕΝΤΟΛΗ L11: ΕΛΕΓΧΟΣ ΣΤΟΙΧΕΙΟΥ ΑΝ ΑΝΗΚΕΙ ΣΕ ΛΙΣΤΑ

MemberQ[*Λίστα, ΣτοιχείοΓιαΈλεγχο*]

Ελέγχει εάν το στοιχείο του δεύτερου ορίσματος ανήκει στη λίστα του πρώτου. Παραδείγματα:

```
In[37]:= MemberQ[{a, b, c, d}, e]

Out[37]= False

In[38]:= MemberQ[{a, b, c, d}, c]

Out[38]= True

In[39]:= MemberQ[Table[x^k, {k, 1, 100}], x^50]

Out[39]= True
```

■ ΕΝΤΟΛΗ L12: ΑΡΙΘΜΟΣ ΣΤΟΙΧΕΙΩΝ ΛΙΣΤΑΣ

Length[*Λίστα*]

Μετράει των αριθμό των στοιχείων μιας λίστας ή διανύσματος (ειδική περίπτωση). Παραδείγματα:

```
In[40]:= Clear[μ]; {V = {s, t, u, v, w}, Length[V], Length[{a}], Length[{a, a}],
          Length[{{a, b}, {c, d, e, f}}], Length[Table[x^k, {k, 1, 100}]], {Equation = μ^4 - β^4 == 0,
           Solutions = Solve[Equation, μ], NumberOfSolutionsOfEquation = Length[Solutions]}}

Out[40]= {{s, t, u, v, w}, 5, 1, 2, 2, 100,
          {-β^4 + μ^4 == 0, {{μ → -β}, {μ → -ⅈ β}, {μ → ⅈ β}, {μ → β}}, 4}}
```

■ ΕΝΤΟΛΕΣ L13 και L14: ΠΡΩΤΟ ΚΑΙ ΤΕΛΕΥΤΑΙΟ ΣΤΟΙΧΕΙΟ ΛΙΣΤΑΣ

First[*Λίστα*] και **Last[*Λίστα*]**

Δίνουν το πρώτο και το τελευταίο στοιχείο μιας λίστας (ή ενός διανύσματος) αντίστοιχα. (Ένα διάνυσμα εκφράζεται κι αυτό στη *Mathematica* μέσω μιας λίστας. Αλλά η λίστα ενός διανύσματος δε μπορεί να έχει στοιχεία που να είναι κι αυτά λίστες.) Παραδείγματα δίνονται στην επόμενη εντολή **Rest**.

■ ΕΝΤΟΛΗ L15: ΑΦΑΙΡΕΣΗ ΤΟΥ ΠΡΩΤΟΥ ΣΤΟΙΧΕΙΟΥ ΛΙΣΤΑΣ

Rest[*Λίστα*]

Επιστρέφει τη λίστα που δίνεται σαν όρισμα, αφού όμως πρώτα αφαιρέσει το πρώτο στοιχείο της. Παραδείγματα γι' αυτήν την εντολή και τις δύο προηγούμενες εντολές **First** και **Last**:

```
In[41]:= {V = {s, t, u, v, w}, First[V], Last[V], Rest[V],
          tb = Table[x^k, {k, 0, 6}], First[tb], Last[tb], Rest[tb]}

Out[41]= {{s, t, u, v, w}, s, w, {t, u, v, w},
          {1, x, x^2, x^3, x^4, x^5, x^6}, 1, x^6, {x, x^2, x^3, x^4, x^5, x^6}}
```

■ ΕΝΤΟΛΗ L16: ΣΤΟΙΧΕΙΟ ΛΙΣΤΑΣ

Part[*Λίστα, ΑριθμόςΣτοιχείου*] ή σχεδόν πάντα ***Λίστα*⟦*ΑριθμόςΣτοιχείου*⟧**

(Συνήθως οι διπλές αγκύλες δημιουργούνται από το πληκτρολόγιο με **Esc [[Esc** και **Esc]] Esc**. Μπορούν βέβαια να χρησιμοποιηθούν και οι απλούστεροι συμβολισμοί **[[** και **]]** αντίστοιχα χωρίς τα ειδικά σύμβολα της *Mathematica* ⟦ και ⟧ αντίστοιχα για τις διπλές αγκύλες τις οποίες χρησιμοποιεί αυτή.) Η πιο πάνω εντολή **Part** δίνει το στοιχείο της λίστας που αντιστοιχεί στον αριθμό στοιχείου του δεύτερου ορίσματος της εντολής αυτής. Παραδείγματα:

```
In[42]:= {V, Part[V, 1], Part[V, 1] == First[V],
          V⟦5⟧, Part[V, Length[V]] == V⟦Length[V]⟧ == Last[V]}

Out[42]= {{s, t, u, v, w}, s, True, w, True}
```

■ ΕΝΤΟΛΗ L17: ΑΝΤΙΣΤΡΟΦΗ ΤΗΣ ΣΕΙΡΑΣ ΤΩΝ ΣΤΟΙΧΕΙΩΝ ΛΙΣΤΑΣ

Reverse[*Λίστα*]

Αντιστρέφει πλήρως τη σειρά εμφανίσεως των στοιχείων μιας λίστας. Παραδείγματα:

```
In[43]:= {V, Reverse[V], GreekVowels = {α, ϵ, η, ι, o, υ, ω}, Reverse[GreekVowels]}
Out[43]= {{s, t, u, v, w}, {w, v, u, t, s}, {α, ϵ, η, ι, o, υ, ω}, {ω, υ, o, ι, η, ϵ, α}}
```

■ ΕΝΤΟΛΗ L18: ΠΡΟΣΘΗΚΗ ΣΤΟΙΧΕΙΟΥ ΣΤΗΝ ΑΡΧΗ ΛΙΣΤΑΣ

Prepend[*Λίστα, Στοιχείο*]

Προσαρτά, προσθέτει στην αρχή της λίστας ***Λίστα*** το στοιχείο ***Στοιχείο*** σαν πρώτο στοιχείο της. Παράδειγμα στην επόμενη εντολή **Append**.

■ ΕΝΤΟΛΗ L19: ΠΡΟΣΘΗΚΗ ΣΤΟΙΧΕΙΟΥ ΣΤΟ ΤΕΛΟΣ ΛΙΣΤΑΣ

Append[*Λίστα, Στοιχείο*]

Προσαρτά, προσθέτει στο τέλος της λίστας ***Λίστα*** το στοιχείο ***Στοιχείο*** σαν τελευταίο στοιχείο της. Παράδειγμα:

```
In[44]:= {Prepend[{a, b, c, d, e}, f], Append[{a, b, c, d, e}, f]}
Out[44]= {{f, a, b, c, d, e}, {a, b, c, d, e, f}}
```

■ ΕΝΤΟΛΗ L20: ΧΩΡΙΣΜΟΣ ΛΙΣΤΑΣ ΣΕ ΥΠΟΛΙΣΤΕΣ

Partition[*Λίστα, ΑκέραιοςΜεγαλύτεροςΤουΈνα*]

Χωρίζει, διαμερίζει τη λίστα ***Λίστα*** σε υπολίστες, την καθεμιά τους με αριθμό στοιχείων ίσων με το θετικό ακέραιο (και μεγαλύτερο του 1) που αναφέρεται στο δεύτερο όρισμα της εντολής. Παράδειγμα:

```
In[45]:= Clear[n];
         Partition[{a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z}, 3]
Out[45]= {{a, b, c}, {d, e, f}, {g, h, i}, {j, k, l}, {m, n, o}, {p, q, r}, {s, t, u}, {v, w, x}}
```

■ ΕΝΤΟΛΗ L21: ΕΣΩΤΕΡΙΚΟ ΓΙΝΟΜΕΝΟ ΔΙΑΝΥΣΜΑΤΩΝ

Dot[*Διάνυσμα-1, Διάνυσμα-2*] ή ***Διάνυσμα-1 . Διάνυσμα-2***

Υπολογίζει το εσωτερικό γινόμενο **AB** δύο διανυσμάτων **A** και **B** που δίνονται. Παραδείγματα:

In[46]:= $\{A = \{a_1, a_2, a_3\},\ B = \{b_1, b_2, b_3\},\ \{A.B,\ B.A,\ A.B == B.A\}\}$

Out[46]= $\{\{a_1, a_2, a_3\}, \{b_1, b_2, b_3\}, \{a_1 b_1 + a_2 b_2 + a_3 b_3, a_1 b_1 + a_2 b_2 + a_3 b_3, \text{True}\}\}$

■ ΕΝΤΟΛΗ L22: ΚΑΤΑΡΓΗΣΗ ΕΣΩΤΕΡΙΚΩΝ ΑΓΚΙΣΤΡΩΝ ΣΕ ΛΙΣΤΑ

Flatten[*Λίστα*]

Αφαιρεί, καταργεί όλα τα εσωτερικά άγκιστρα σε μια λίστα. Την κάνει "επίπεδη". Παραδείγματα:

```
In[47]:= {Flatten[{{{a}, {b, {{c}}}, {d}}, e}],
          mat = {{a, b, c}, {d, e, f}, {g, h, i}}, Flatten[mat]}

Out[47]= {{a, b, c, d, e}, {{a, b, c}, {d, e, f}, {g, h, i}}, {a, b, c, d, e, f, g, h, i}}
```

■ ΕΝΤΟΛΗ L23: ΤΑΞΙΝΟΜΗΣΗ ΤΩΝ ΣΤΟΙΧΕΙΩΝ ΛΙΣΤΑΣ

Sort[*Λίστα*]

Ταξινομεί τα στοιχεία της λίστας η οποία δίνεται, εάν αυτά είναι είτε γράμματα είτε αριθμοί. (Οι αριθμοί προηγούνται από τα γράμματα.) Παραδείγματα:

```
In[48]:= {Sort[{5, 10, -6, 8, 3}], Sort[{a, f, b, z, p, -3, 4, 2, -1}],
          Sort[{D, c5, aa, 50, BC, DD, -3, deq, b10}]}

Out[48]= {{-6, 3, 5, 8, 10}, {-3, -1, 2, 4, a, b, f, p, z}, {-3, 50, aa, b10, BC, c5, D, DD, deq}}
```

■ ΕΝΤΟΛΗ L24: ΑΡΙΘΜΟΣ ΦΟΡΩΝ ΕΜΦΑΝΙΣΕΩΣ ΣΤΟΙΧΕΙΟΥ ΛΙΣΤΑΣ

Count[*Λίστα, ΠιθανόΌχιΣίγουροΣτοιχείοΤηςΛίστας*]

Μετράει των αριθμό των φορών που ένα πιθανό (όχι σίγουρο) στοιχείο μιας λίστας εμφανίζεται στη λίστα. Παραδείγματα:

```
In[49]:= {G = {a, b, 3, c, 4, d, a, 2, e, 3, c, 5, 3, b, 3, 1, d, c, 3, c}; Count[G, d], Count[G, p],
          setG = Union[G], Table[Count[G, setG⟦k⟧], {k, 1, Length[setG]}], Length[setG]}

Out[49]= {2, 0, {1, 2, 3, 4, 5, a, b, c, d, e}, {1, 1, 5, 1, 1, 2, 2, 4, 2, 1}, 10}
```

■ ΕΝΤΟΛΗ L25: ΔΗΜΙΟΥΡΓΙΑ ΛΙΣΤΑΣ ΑΠΟ ΕΠΙΜΕΡΟΥΣ ΛΙΣΤΕΣ

Join[*Λίστα-1, Λίστα-2, ..., Λίστα-n*]

Ενώνει στη σειρά τις λίστες που δίνονται. Τα διπλά και τα πολλαπλά στοιχεία διατηρούνται και δεν αλλάζει καθόλου η σειρά εμφανίσεως των στοιχείων στην τελική λίστα. (Αντίθετα όμως τα διπλά και τα πολλαπλά στοιχεία δε διατηρούνται στην επόμενη εντολή **Union**. Αυτή χρησιμοποιείται, όταν οι λίστες είναι σύνολα.) Παράδειγμα:

```
In[50]:= {w1 = {b, e, a, m}, w2 = {p, l, a, t, e}, w3 = {s, h , e , l, l}}

Out[50]= {{b, e, a, m}, {p, l, a, t, e}, {s, h, e, l, l}}

In[51]:= Join[w1, w2, w3]

Out[51]= {b, e, a, m, p, l, a, t, e, s, h, e, l, l}
```

■ ΕΝΤΟΛΗ L26: ΕΝΩΣΗ ΣΥΝΟΛΩΝ

Union[*Λίστα*]

Union[*Λίστα-1, Λίστα-2, . . . , Λίστα-n*]

Η πρώτη μορφή της εντολής αυτής (με ένα όρισμα) μετατρέπει μια λίστα σε σύνολο εξαλείφοντας διπλές ή και πολλαπλές εμφανίσεις του ίδιου στοιχείου στη λίστα, όσες φορές παρουσιάζονται. Η δεύτερη μορφή κάνει την ένωση σε λίστες σαν να ήσαν σύνολα, δηλαδή χωρίς την πολλαπλή εμφάνιση του ίδιου στοιχείου (αντίθετα με την προηγούμενη εντολή **Join**, στο αποτέλεσμα της οποίας είναι δυνατή η πολλαπλή εμφάνιση του ίδιου στοιχείου). Τα στοιχεία στο αποτέλεσμα είναι ταξινομημένα. Παραδείγματα:

```
In[52]:= {L1 = {a, b, c, c, b, a}, L2 = {b, f, b, e, f, f}, L3 = {a, b, d, e, e, f, a, b},
          Union[L1, L2, L3], Union[L1], Union[L1, L1], Union[L1, {}], Union[{}, {}]}

Out[52]= {{a, b, c, c, b, a}, {b, f, b, e, f, f}, {a, b, d, e, e, f, a, b},
          {a, b, c, d, e, f}, {a, b, c}, {a, b, c}, {a, b, c}, {}}
```

■ ΕΝΤΟΛΗ L27: ΤΟΜΗ ΣΥΝΟΛΩΝ

Intersection[*Λίστα-1, Λίστα-2, . . . , Λίστα-n*]

Ανάλογα με την εντολή **Union** εκτελεί τομή των συνόλων που παριστάνονται με τις σχετικές λίστες (ή σύνολα) εξαλείφοντας επ' ευκαιρία τα διπλά ή πολλαπλά στοιχεία, αν παρουσιάζονται. Παράδειγμα:

```
In[53]:= {S1 = {3, 5, 5, 10, 7}, S2 = {3, 7, 4, 7, 7, 4, 6, 4, 7, 9}}

Out[53]= {{3, 5, 5, 10, 7}, {3, 7, 4, 7, 7, 4, 6, 4, 7, 9}}

In[54]:= {Union[S1, S2], Intersection[S1, S2]}

Out[54]= {{3, 4, 5, 6, 7, 9, 10}, {3, 7}}

In[55]:= {Union[S1, S2] == Intersection[S1, S2], Union[S1, S1] == Intersection[S1, S1]}

Out[55]= {False, True}
```

■ ΕΝΤΟΛΗ L28: ΣΥΜΠΛΗΡΩΜΑ ΣΥΝΟΛΟΥ

Complement[*Λίστα-1, Λίστα-2*]

Υπολογίζει το συμπλήρωμα (ή τη διαφορά) $C = A - B$ ενός συνόλου B (δεύτερο όρισμα) ως προς ένα σύνολο A (πρώτο όρισμα). Εδώ μπορούμε να έχουμε γενικότερα και λίστες A και B αντί αποκλειστικά σύνολα. Παραδείγματα:

```
In[56]:= {A = {a, b, c, c, a, d, c, a, a, b, d}, Complement[A, {b, c, d}], Complement[A, {c, d}],
          Complement[{a, b, c, d}, {e, f, g}], Complement[{a, b}, {}], Complement[{}, {a, b}]}

Out[56]= {{a, b, c, c, a, d, c, a, a, b, d}, {a}, {a, b}, {a, b, c, d}, {a, b}, {}}
```

■ *Notebook E8*

ΕΝΤΟΛΕΣ ΓΙΑ ΤΗ ΓΡΑΜΜΙΚΗ ΑΛΓΕΒΡΑ

17 ΕΝΤΟΛΕΣ: **M1. Dot, M2. MatrixPower, M3. MatrixQ, M4. MatrixForm, M5. IdentityMatrix, M6. DiagonalMatrix, M7. Transpose, M8. Inverse, M9. Dimensions, M10. Tr, M11. Det, M12. Normalize, M13. CharacteristicPolynomial, M14. Eigenvalues, M15. Eigenvectors, M16. Eigensystem, M17. RowReduce**

■ ΕΝΤΟΛΗ M1: ΠΟΛΛΑΠΛΑΣΙΑΣΜΟΣ ΜΗΤΡΩΩΝ

Dot[*Μητρώο-1, Μητρώο-2*] ή ***Μητρώο-1 . Μητρώο-2*** *(με τελεία το σύμβολο του πολλαπλασιασμού)*

Εκτελεί τον πολλαπλασιασμό **AB** δύο μητρώων **A** και **B**. (Ανάλογα και περισσότερων μητρώων.) Τονίζεται ξανά ότι ο πολλαπλασιασμός μητρώων γίνεται με τελεία (αν δε θέλουμε να χρησιμοποιήσουμε την ίδια την εντολή **Dot**) και όχι με κενό ή με αστεράκι * αντίθετα από ό,τι συμβαίνει στις κοινές αριθμητικές και αλγεβρικές παραστάσεις. Επίσης η πρόσθεση μητρώων **A** + **B** γίνεται με το συνηθισμένο σύμβολο + και η αφαίρεση μητρώων **A** − **B** με το συνηθισμένο σύμβολο − , ακριβώς όπως συμβαίνει και στις κοινές αριθμητικές και αλγεβρικές παραστάσεις. Παραδείγματα:

```
In[1]:= {A = {{1, 2}, {3, 4}}, B = {{5, 6}, {7, 8}}, A + B, A - B, A.B, Dot[C, D, E, F] == C.D.E.F}

Out[1]= {{{1, 2}, {3, 4}}, {{5, 6}, {7, 8}}, {{6, 8}, {10, 12}},
         {{-4, -4}, {-4, -4}}, {{19, 22}, {43, 50}}, True}
```

■ ΕΝΤΟΛΗ M2: ΥΨΩΣΗ ΜΗΤΡΩΟΥ ΣΕ ΔΥΝΑΜΗ

MatrixPower[*Μητρώο, Δύναμη*]

Υπολογίζει τη *n*-στή δύναμη ενός μητρώου **A**. Δεν επιτρέπεται να χρησιμοποιείται ο συνηθισμένος συμβολισμός $\mathbf{A}^n$ για τη *n*-στή δύναμη του μητρώου **A**. Παραδείγματα:

```
In[2]:= {A = {{1, 2}, {3, 4}}, A.A.A, MatrixPower[A, 3],
         A.A.A == MatrixPower[A, 3], B.B.B.B.B.B.B.B.B.B == MatrixPower[B, 10]}

Out[2]= {{{1, 2}, {3, 4}}, {{37, 54}, {81, 118}}, {{37, 54}, {81, 118}}, True, True}
```

■ ΕΝΤΟΛΗ M3: ΕΛΕΓΧΟΣ ΠΑΡΑΣΤΑΣΕΩΣ ΑΝ ΕΙΝΑΙ ΜΗΤΡΩΟ

MatrixQ[*Παράσταση*]

Ελέγχει αν μια παράσταση είναι μητρώο ή όχι. Απλές λίστες δε θεωρούνται μητρώα. Παραδείγματα:

```
In[3]:= {MatrixQ[1], VectorQ[1], MatrixQ[a^2], VectorQ[a^2], MatrixQ[{a^2}], MatrixQ[{1, 2}],
         VectorQ[{1, 2}], MatrixQ[{{1}, {2}}], MatrixQ[{{1, 2}, {3, 4}, {5, 6}, {7, 8}}]}

Out[3]= {False, False, False, False, False, False, True, True, True}
```

■ ΕΝΤΟΛΗ Μ4: ΕΜΦΑΝΙΣΗ ΣΕ ΜΟΡΦΗ ΜΗΤΡΩΟΥ

MatrixForm[*Μητρώο*] ή ***Μητρώο*** **// MatrixForm**

Μετατρέπει ένα μητρώο **A** από τη μορφή που οι σειρές του εμφανίζονται στην ίδια γραμμή η μία μετά την άλλη σε μορφή μητρώου, δηλαδή σε αληθινή μορφή γραμμών και στηλών, και το παρουσιάζει στην οθόνη στη μορφή αυτή. ***Δυστυχώς ένα μητρώο που είναι και παρουσιάζεται σ' αυτήν τη μορφή δε μπορεί να χρησιμοποιηθεί σε κανέναν υπολογισμό με μητρώα.*** Η εντολή αυτή είναι βέβαια πάρα πολύ χρήσιμη, αλλά δυστυχώς μόνο για την παρουσίαση ενός μητρώου στην οθόνη. Παραδείγματα:

```
In[4]:= {A, A1 = MatrixForm[A], B, B1 = B // MatrixForm}
```

Out[4]= $\left\{\{\{1, 2\}, \{3, 4\}\}, \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \{\{5, 6\}, \{7, 8\}\}, \begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix}\right\}$

```
In[5]:= {AB = A.B, AB // MatrixForm, A1.B1}
```

Out[5]= $\left\{\{\{19, 22\}, \{43, 50\}\}, \begin{pmatrix} 19 & 22 \\ 43 & 50 \end{pmatrix}, \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} . \begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix}\right\}$

■ ΕΝΤΟΛΗ Μ5: ΜΟΝΑΔΙΑΙΟ ΜΗΤΡΩΟ

IdentityMatrix[*ΤάξηΤουΜητρώου*]

Δημιουργεί το μοναδιαίο μητρώο (ή ταυτοτικό μητρώο) $\mathbf{I}_n$ τάξεως *n*. Παραδείγματα:

In[6]:= $\{i_2 = \text{IdentityMatrix}[2],\ i_2\ //\ \text{MatrixForm},\ i_3 = \text{IdentityMatrix}[3],\ i_3\ //\ \text{MatrixForm}\}$

Out[6]= $\left\{\{\{1, 0\}, \{0, 1\}\}, \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \{\{1, 0, 0\}, \{0, 1, 0\}, \{0, 0, 1\}\}, \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}\right\}$

■ ΕΝΤΟΛΗ Μ6: ΔΙΑΓΩΝΙΟ ΜΗΤΡΩΟ

DiagonalMatrix[*Λίστα*]

Δημιουργεί το διαγώνιο μητρώο με στοιχεία της κύριας διαγωνίου του τα στοιχεία της λίστας η οποία δίνεται στο όρισμα της εντολής και φυσικά όλα τα υπόλοιπα στοιχεία του μηδενικά. Παραδείγματα:

In[7]:= $\{A = \text{DiagonalMatrix}[\{1, 2\}],\ M = \text{DiagonalMatrix}[\{m_1, m_2, m_3, m_4\}]\}$

Out[7]= $\{\{\{1, 0\}, \{0, 2\}\}, \{\{m_1, 0, 0, 0\}, \{0, m_2, 0, 0\}, \{0, 0, m_3, 0\}, \{0, 0, 0, m_4\}\}\}$

```
In[8]:= {A // MatrixForm, M // MatrixForm}
```

Out[8]= $\left\{\begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}, \begin{pmatrix} m_1 & 0 & 0 & 0 \\ 0 & m_2 & 0 & 0 \\ 0 & 0 & m_3 & 0 \\ 0 & 0 & 0 & m_4 \end{pmatrix}\right\}$

```
In[9]:= DiagonalMatrix[{1, 1, 1, 1, 1}] == IdentityMatrix[5]

Out[9]= True
```

```
In[10]:= ZeroMatrix3 = DiagonalMatrix[{0, 0, 0}]
Out[10]= {{0, 0, 0}, {0, 0, 0}, {0, 0, 0}}
In[11]:= ZeroMatrix[n_] := DiagonalMatrix[Table[0, {k, 1, n}]]
In[12]:= {ZeroMatrix[1] // MatrixForm, ZeroMatrix[2] // MatrixForm}
```

Out[12]= $\left\{\begin{pmatrix}0\end{pmatrix}, \begin{pmatrix}0 & 0\\ 0 & 0\end{pmatrix}\right\}$

■ ΕΝΤΟΛΗ Μ7: ΑΝΑΣΤΡΟΦΟ ΜΗΤΡΩΟ

Transpose[*Μητρώο*]

Δίνει το ανάστροφο μητρώο $\mathbf{A}^T$ ενός μητρώου **A**, δηλαδή με τις γραμμές του να έχουν γίνει τώρα στήλες και τις στήλες γραμμές. Το μητρώο μπορεί να είναι ένα γενικό μητρώο διαστάσεων $m \times n$ και όχι κατ' ανάγκη ένα τετραγωνικό μητρώο διαστάσεων $n\times n$. Παραδείγματα:

```
In[13]:= {A = {{a, b}, {c, d}}; A // MatrixForm, AT = Transpose[A]; AT // MatrixForm}
```

Out[13]= $\left\{\begin{pmatrix}a & b\\ c & d\end{pmatrix}, \begin{pmatrix}a & c\\ b & d\end{pmatrix}\right\}$

```
In[14]:= {B = {{1, 2}, {3, 4}, {5, 6}}; B // MatrixForm, BT = Transpose[B]; BT // MatrixForm}
```

Out[14]= $\left\{\begin{pmatrix}1 & 2\\ 3 & 4\\ 5 & 6\end{pmatrix}, \begin{pmatrix}1 & 3 & 5\\ 2 & 4 & 6\end{pmatrix}\right\}$

In[15]:= $\{K = \{\{k_{11}, k_{12}\}, \{k_{21}, k_{22}\}\}, K_m = K$ // MatrixForm}

Out[15]= $\left\{\{\{k_{11}, k_{12}\}, \{k_{21}, k_{22}\}\}, \begin{pmatrix}k_{11} & k_{12}\\ k_{21} & k_{22}\end{pmatrix}\right\}$

In[16]:= $\{K_{T1}$ = Transpose[K], K_{T1} // MatrixForm, K_{T2} = Transpose[K_m]}

Out[16]= $\left\{\{\{k_{11}, k_{21}\}, \{k_{12}, k_{22}\}\}, \begin{pmatrix}k_{11} & k_{21}\\ k_{12} & k_{22}\end{pmatrix}, \text{Transpose}\left[\begin{pmatrix}k_{11} & k_{12}\\ k_{21} & k_{22}\end{pmatrix}\right]\right\}$

■ ΕΝΤΟΛΗ Μ8: ΑΝΤΙΣΤΡΟΦΟ ΜΗΤΡΩΟ

Inverse[*ΑντιστρέψιμοΤετραγωνικόΜητρώο*]

Δίνει το αντίστροφο μητρώο $\mathbf{A}^{-1}$ αντιστρέψιμου τετραγωνικού μητρώου **A** ($\det \mathbf{A} \neq 0$). Παραδείγματα:

```
In[17]:= {A = {{a, b}, {c, d}}; A // MatrixForm, AI = Inverse[A]; AI // MatrixForm,
          AI.A // MatrixForm // Simplify, B = {{e, f}, {0, 0}}; B // MatrixForm, BI = Inverse[B]}

        Inverse::sing : Matrix {{e, f}, {0, 0}} is singular.
```

Out[17]= $\left\{\begin{pmatrix}a & b\\ c & d\end{pmatrix}, \begin{pmatrix}\frac{d}{-b\,c+a\,d} & -\frac{b}{-b\,c+a\,d}\\ -\frac{c}{-b\,c+a\,d} & \frac{a}{-b\,c+a\,d}\end{pmatrix}, \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}, \begin{pmatrix}e & f\\ 0 & 0\end{pmatrix}, \text{Inverse}[\{\{e, f\}, \{0, 0\}\}]\right\}$

```
In[18]:= {A.AI == AI.A == IdentityMatrix[2], A.AI == Dot[A, AI] == AI.A == Dot[AI, A]} // Simplify
Out[18]= {True, True}
```

■ ΕΝΤΟΛΗ Μ9: ΔΙΑΣΤΑΣΕΙΣ ΜΗΤΡΩΟΥ

Dimensions[*Μητρώο*]

Προσδιορίζει τις διαστάσεις *m* και *n* ενός μητρώου **A**. Παραδείγματα:

```
In[19]:= {A = {{a, b, c, d}, {e, f, g, h}}, Dimensions[A], Dimensions[Transpose[A]]}
Out[19]= {{{a, b, c, d}, {e, f, g, h}}, {2, 4}, {4, 2}}
```

■ ΕΝΤΟΛΗ Μ10: ΙΧΝΟΣ ΜΗΤΡΩΟΥ

Tr[*ΤετραγωνικόΜητρώο*]

Υπολογίζει το ίχνος Tr **A** ενός τετραγωνικού μητρώου **A**, που είναι βέβαια ένας αριθμός: το άθροισμα των στοιχείων της κύριας διαγωνίου του μητρώου **A** και όχι μητρώο. Παραδείγματα:

```
In[20]:= {Tr[{{a, b}, {c, d}}], Tr[IdentityMatrix[10]], Tr[DiagonalMatrix[{m1, m2, m3, m4}]]}
```

$$\text{Out[20]= } \{a + d,\ 10,\ m_1 + m_2 + m_3 + m_4\}$$

■ ΕΝΤΟΛΗ Μ11: ΟΡΙΖΟΥΣΑ ΜΗΤΡΩΟΥ

Det[*ΤετραγωνικόΜητρώο*]

Υπολογίζει την ορίζουσα det **A** ή |**A**| ενός τετραγωνικού μητρώου **A**. Παραδείγματα:

```
In[21]:= {Det[{{a, b}, {c, d}}],
          Det[{{1, -4, 5}, {5, -6, 10}, {9, 4, -6}}], Det[{{e, f, g}, {h, i, j}}]}

        Det::matsq : Argument {{e, f, g}, {h, i, j}} at position 1 is not a square matrix.

Out[21]= {-b c + a d, -114, Det[{{e, f, g}, {h, i, j}}]}
```

$$\text{In[22]:= } \{K = \{\{k_{11}, k_{12}\}, \{k_{12}, k_{22}\}\},\ M = \{\{m_1, 0\}, \{0, m_2\}\},\ \text{Det}[K - \omega^2 M] == 0 \text{ // Simplify}\}$$

$$\text{Out[22]= } \{\{\{k_{11}, k_{12}\}, \{k_{12}, k_{22}\}\},\ \{\{m_1, 0\}, \{0, m_2\}\},\ (k_{11} - \omega^2 m_1)(k_{22} - \omega^2 m_2) == k_{12}^2\}$$

■ ΕΝΤΟΛΗ Μ12: ΚΑΝΟΝΙΚΟΠΟΙΗΣΗ ΔΙΑΝΥΣΜΑΤΟΣ

Normalize[*Διάνυσμα*]

Υπολογίζει την κανονικοποιημένη μορφή $\mathbf{A}_n$ ενός διανύσματος **A**, δηλαδή το αντίστοιχο διάνυσμα με μέτρο 1. Πρόκειται για μια από τις εντολές του πακέτου **LinearAlgebra'Orthogonalization'**, το οποίο θα πρέπει φυσικά να έχει φορτωθεί πριν από την κλήση αυτής της εντολής **Normalize**. Παραδείγματα:

```
In[23]:= << LinearAlgebra`Orthogonalization`
```

$$\text{In[24]:= } \{\text{Normalize}[\{1, -2, 10, 4\}],\ A = \{a, b, c\},\ A_n = \text{Normalize}[\{a, b, c\}],\ A_n . A_n \text{ // Simplify}\}$$

$$\text{Out[24]= } \left\{\left\{\frac{1}{11}, -\frac{2}{11}, \frac{10}{11}, \frac{4}{11}\right\}, \{a, b, c\}, \left\{\frac{a}{\sqrt{a^2+b^2+c^2}}, \frac{b}{\sqrt{a^2+b^2+c^2}}, \frac{c}{\sqrt{a^2+b^2+c^2}}\right\}, 1\right\}$$

■ ΕΝΤΟΛΗ M13: ΧΑΡΑΚΤΗΡΙΣΤΙΚΟ ΠΟΛΥΩΝΥΜΟ ΜΗΤΡΩΟΥ

CharacteristicPolynomial[*ΤετραγωνικόΜητρώο, ΜεταβλητήΧαρακτηριστικούΠολυωνύμου*]

Υπολογίζει το χαρακτηριστικό πολυώνυμο $p(\lambda)$ του τετραγωνικού μητρώου **A**, το οποίο καθορίζεται στο πρώτο όρισμα ως προς τη μεταβλητή λ που καθορίζεται στο δεύτερο όρισμα. Παραδείγματα:

```
In[25]:= {CharacteristicPolynomial[K, λ], CharacteristicPolynomial[M, λ]}
```

Out[25]= $\{\lambda^2 - \lambda\, k_{11} - k_{12}^2 - \lambda\, k_{22} + k_{11}\, k_{22},\ \lambda^2 - \lambda\, m_1 - \lambda\, m_2 + m_1\, m_2\}$

```
In[26]:= {chp = CharacteristicPolynomial[-λ M + K, λ]; Collect[chp, λ]}
```

Out[26]= $\{-k_{12}^2 + k_{11}\, k_{22} + \lambda\, (-k_{11} - k_{22} - k_{22}\, m_1 - k_{11}\, m_2) + \lambda^2\, (1 + m_1 + m_2 + m_1\, m_2)\}$

■ ΕΝΤΟΛΗ M14: ΙΔΙΟΤΙΜΕΣ ΜΗΤΡΩΟΥ

Eigenvalues[*ΤετραγωνικόΜητρώο*]

Υπολογίζει τις ιδιοτιμές λ_k ενός τετραγωνικού μητρώου **A**. Παραδείγματα:

```
In[27]:= {A = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}}, λs = Eigenvalues[A], N[{λs[[2]], λs[[3]]}]}
```

Out[27]= $\left\{\{\{1, 2, 3\}, \{4, 5, 6\}, \{7, 8, 9\}\}, \left\{0, \frac{3}{2}\left(5 - \sqrt{33}\right), \frac{3}{2}\left(5 + \sqrt{33}\right)\right\}, \{-1.11684, 16.1168\}\right\}$

```
In[28]:= {λs = Eigenvalues[{{a, b}, {c, d}}], Length[λs]}
```

Out[28]= $\left\{\left\{\frac{1}{2}\left(a + d - \sqrt{a^2 + 4\, b\, c - 2\, a\, d + d^2}\right), \frac{1}{2}\left(a + d + \sqrt{a^2 + 4\, b\, c - 2\, a\, d + d^2}\right)\right\}, 2\right\}$

■ ΕΝΤΟΛΗ M15: ΙΔΙΟΔΙΑΝΥΣΜΑΤΑ ΜΗΤΡΩΟΥ

Eigenvectors[*ΤετραγωνικόΜητρώο*]

Υπολογίζει τα ιδιοδιανύσματα $\boldsymbol{\delta}_k$ ενός τετραγωνικού μητρώου **A** (φυσικά κατά προσέγγιση μιας πολλαπλασιαστικής σταθεράς). Παραδείγματα:

```
In[29]:= {δs = Eigenvectors[A], δs // N, Length[δs]}
```

Out[29]= $\left\{\left\{\{1, -2, 1\}, \left\{-\frac{15 - \sqrt{33}}{-33 + 7\sqrt{33}}, \frac{4\left(-6 + \sqrt{33}\right)}{-33 + 7\sqrt{33}}, 1\right\}, \left\{-\frac{-15 - \sqrt{33}}{33 + 7\sqrt{33}}, \frac{4\left(6 + \sqrt{33}\right)}{33 + 7\sqrt{33}}, 1\right\}\right\},\right.$
$\left.\{\{1., -2., 1.\}, \{-1.28335, -0.141675, 1.\}, \{0.283349, 0.641675, 1.\}\}, 3\right\}$

```
In[30]:= {B = {{a, b}, {b, c}}; B // MatrixForm, T = Eigenvectors[B] // FullSimplify}
```

Out[30]= $\left\{\begin{pmatrix} a & b \\ b & c \end{pmatrix}, \left\{\left\{-\frac{-a + \sqrt{4\, b^2 + (a - c)^2} + c}{2\, b}, 1\right\}, \left\{\frac{a + \sqrt{4\, b^2 + (a - c)^2} - c}{2\, b}, 1\right\}\right\}\right\}$

```
In[31]:= TT = Transpose[T]; TT // MatrixForm
```

Out[31]//MatrixForm=

$$\begin{pmatrix} -\frac{-a+\sqrt{4\, b^2+(a-c)^2}+c}{2\, b} & \frac{a+\sqrt{4\, b^2+(a-c)^2}-c}{2\, b} \\ 1 & 1 \end{pmatrix}$$

```
In[32]:= BT = T.B.TT // FullSimplify; BT // MatrixForm
```

Out[32]//MatrixForm=

$$\begin{pmatrix} \frac{a\,(4\,b^2+(a-c)^2)-\sqrt{4\,b^2+(a-c)^2}\,(a^2+2\,b^2-a\,c)}{2\,b^2} & 0 \\ 0 & \frac{a\,(4\,b^2+(a-c)^2)+\sqrt{4\,b^2+(a-c)^2}\,(a^2+2\,b^2-a\,c)}{2\,b^2} \end{pmatrix}$$

■ ΕΝΤΟΛΗ Μ16: ΙΔΙΟΤΙΜΕΣ ΚΑΙ ΙΔΙΟΔΙΑΝΥΣΜΑΤΑ ΜΗΤΡΩΟΥ

Eigensystem[*ΤετραγωνικόΜητρώο*]

Υπολογίζει τόσο τις ιδιοτιμές λ_k όσο και τα ιδιοδιανύσματα $\boldsymbol{\delta}_k$ ενός τετραγωνικού μητρώου **A**. Στο α-ποτέλεσμα οι ιδιοτιμές λ_k αναφέρονται πρώτες: είναι το πρώτο στοιχείο της λίστας αποτελεσμάτων, ενώ τα ιδιοδιανύσματα $\boldsymbol{\delta}_k$ αναφέρονται έπειτα: είναι το δεύτερο στοιχείο της λίστας. Παραδείγματα:

```
In[33]:= Eigensystem[A]
```

Out[33]= $\left\{\left\{0, -\frac{3}{2}\left(-5+\sqrt{33}\right), \frac{3}{2}\left(5+\sqrt{33}\right)\right\}, \left\{\{1, -2, 1\}, \left\{-\frac{15-\sqrt{33}}{-33+7\sqrt{33}}, \frac{4\left(-6+\sqrt{33}\right)}{-33+7\sqrt{33}}, 1\right\}, \left\{-\frac{-15-\sqrt{33}}{33+7\sqrt{33}}, \frac{4\left(6+\sqrt{33}\right)}{33+7\sqrt{33}}, 1\right\}\right\}\right\}$

```
In[34]:= {{Eigenvalues[A], Eigensystem[A][[1]]} // N, Eigenvectors[A] == Eigensystem[A][[2]]}

Out[34]= {{{0., -1.11684, 16.1168}, {0., -1.11684, 16.1168}}, True}

In[35]:= {B // MatrixForm, SB = Eigensystem[B] // FullSimplify; SB // MatrixForm}
```

Out[35]= $\left\{\begin{pmatrix} a & b \\ b & c \end{pmatrix}, \begin{pmatrix} \frac{1}{2}\left(a-\sqrt{4\,b^2+(a-c)^2}+c\right) & \frac{1}{2}\left(a+\sqrt{4\,b^2+(a-c)^2}+c\right) \\ \left\{-\frac{-a+\sqrt{4\,b^2+(a-c)^2}+c}{2\,b}, 1\right\} & \left\{\frac{a+\sqrt{4\,b^2+(a-c)^2}-c}{2\,b}, 1\right\} \end{pmatrix}\right\}$

■ ΕΝΤΟΛΗ Μ17: ΑΝΗΓΜΕΝΗ ΚΑΤΑ ΓΡΑΜΜΕΣ ΜΟΡΦΗ ΜΗΤΡΩΟΥ

RowReduce[*Μητρώο*]

Υπολογίζει την ανηγμένη κατά γραμμές μορφή ενός μητρώου **A** μέσω των κατάλληλων προσθέσεων πολλαπλασίων των γραμμών του μητρώου **A**. Ειδικά εάν το μητρώο **A** είναι τετραγωνικό (διαστάσεων $n\times n$) και μη ιδιάζον, τότε προκύπτει τελικά το μοναδιαίο μητρώο $\mathbf{I}_n$. Η εντολή αυτή είναι πολύ χρήσιμη για την επίλυση συστημάτων γραμμικών αλγεβρικών εξισώσεων με απαλοιφή Gauss. Παραδείγματα:

```
In[36]:= {A = {{1, 3, 7}, {2, 10, -8}, {-5, 3, -6}}; A // MatrixForm,
          AR = RowReduce[A]; AR // MatrixForm, AR == IdentityMatrix[3]}
```

Out[36]= $\left\{\begin{pmatrix} 1 & 3 & 7 \\ 2 & 10 & -8 \\ -5 & 3 & -6 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \text{True}\right\}$

```
In[37]:= {B = {{1, 3, 7, -a}, {2, 10, -8, -b}, {-5, 3, -6, -c}};
          B // MatrixForm, BR = RowReduce[B]; BR // MatrixForm}
```

Out[37]= $\left\{\begin{pmatrix} 1 & 3 & 7 & -a \\ 2 & 10 & -8 & -b \\ -5 & 3 & -6 & -c \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 & \frac{1}{512}(36\,a-39\,b+94\,c) \\ 0 & 1 & 0 & \frac{1}{512}(-52\,a-29\,b-22\,c) \\ 0 & 0 & 1 & \frac{1}{256}(-28\,a+9\,b-2\,c) \end{pmatrix}\right\}$

■ *Notebook E9*
ΕΝΤΟΛΕΣ ΓΙΑ ΔΙΔΙΑΣΤΑΤΕΣ ΓΡΑΦΙΚΕΣ ΠΑΡΑΣΤΑΣΕΙΣ

9 ΕΝΤΟΛΕΣ: **V1. Plot, V2. FilledPlot, V3. ImplicitPlot, V4. ParametricPlot, V5. Show, V6. GraphicsArray, V7. ListPlot, V8. ContourPlot, V9. DensityPlot**

■ ΕΝΤΟΛΗ V1: ΔΙΔΙΑΣΤΑΤΗ ΓΡΑΦΙΚΗ ΠΑΡΑΣΤΑΣΗ

Plot[*ΣυνάρτησηΜιαςΜεταβλητής, {Μεταβλητή, ΑρχικήΤιμή, ΤελικήΤιμή}, Επιλογή-1, Επιλογή-2, . . .*]

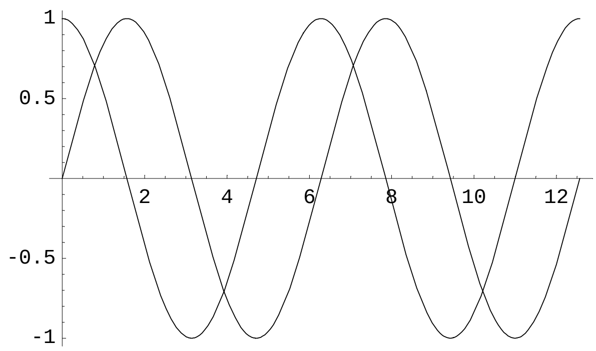
Plot[*ΛίσταΣυναρτήσεωνΜιαςΜεταβλητής, {Μεταβλητή, ΑρχικήΤιμή, ΤελικήΤιμή}, Επιλογή-1, . . .*]

Κάνει τη γραφική παράσταση της συναρτήσεως που δίνεται ή των συναρτήσεων που δίδονται (σε λίστα) στο πρώτο όρισμα. Η γραφική αυτή παράσταση γίνεται στο διάστημα που καθορίζεται στο δεύτερο όρισμα μετά τη μεταβλητή. Προαιρετικά ακολουθεί μία επιλογή ή συνήθως περισσότερες από μία επιλογές για τον τρόπο εμφανίσεως της γραφικής παραστάσεως. Οι επιλογές (options) αυτές είναι οι ακόλουθες τριάντα με καθεμιά τους να ακολουθείται από την αρχική τιμή που της δίνει η *Mathematica*:

```
In[1]:= Options[Plot]

Out[1]= {AspectRatio → 1/GoldenRatio, Axes → Automatic, AxesLabel → None,
  AxesOrigin → Automatic, AxesStyle → Automatic, Background → Automatic,
  ColorOutput → Automatic, Compiled → True, DefaultColor → Automatic,
  Epilog → {}, Frame → False, FrameLabel → None, FrameStyle → Automatic,
  FrameTicks → Automatic, GridLines → None, ImageSize → Automatic,
  MaxBend → 10., PlotDivision → 30., PlotLabel → None, PlotPoints → 25,
  PlotRange → Automatic, PlotRegion → Automatic, PlotStyle → Automatic,
  Prolog → {}, RotateLabel → True, Ticks → Automatic, DefaultFont :→ $DefaultFont,
  DisplayFunction :→ $DisplayFunction, FormatType :→ $FormatType, TextStyle :→ $TextStyle}
```

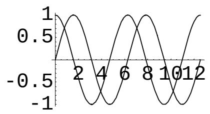
Μερικές (όχι και πάρα πολλές) από τις επιλογές αυτές θα τις επιδείξουμε σε παραδείγματα παρακάτω. Ακολουθούν παραδείγματα της εντολής **Plot**. Και πρώτα–πρώτα οι γραφικές παραστάσεις των δύο βασικών τριγωνομετρικών συναρτήσεων συνημίτονο ($\cos x$) και ημίτονο ($\sin x$) στο διάστημα $[0, 4\pi]$ και οι δυο τους σε ένα ενιαίο σχήμα και χωρίς τη χρήση επιλογών, δηλαδή με τη χρήση των αρχικών τιμών των επιλογών. (Βέβαια τις αρχικές τιμές μπορούμε να τις αλλάζουμε. Θα δώσουμε πολλά παραδείγματα.)

```
In[2]:= Plot[{Cos[x], Sin[x]}, {x, 0, 4 π}];
```

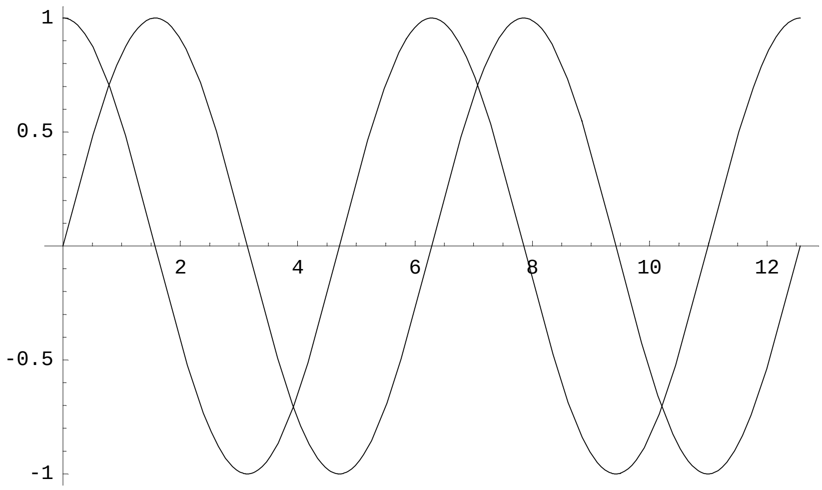


Σημειώνουμε ότι θέτοντας την Αγγλική άνω τελεία (το Ελληνικό ερωτηματικό) ; στο τέλος της εντολής **Plot** απλά δεν εμφανίζεται η ένδειξη **- Graphics -** αμέσως μετά τη γραφική παράσταση, ενώ εμφανίζεται η ίδια η γραφική παράσταση. Συχνά θα χρησιμοποιούμε αυτήν τη δυνατότητα. Με την επιλογή **ImageSize** μπορούμε να μικρύνουμε ή να μεγαλώσουμε το πραγματικό μέγεθος της γραφικής παραστάσεως. Τροποποιήσεις του προηγούμενου παραδείγματος με τη χρήση της επιλογής **ImageSize**:

```
In[3]:= Plot[{Cos[x], Sin[x]}, {x, 0, 4 π}, ImageSize → 100];
```



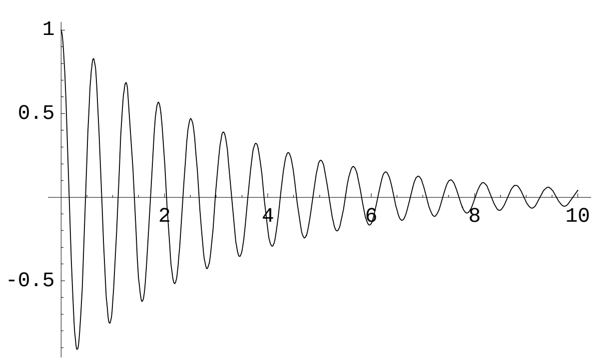

```
In[4]:= Plot[{Cos[x], Sin[x]}, {x, 0, 4 π}, ImageSize → 400];
```



Και τώρα το κλασικό για τον Πολιτικό Μηχανικό πρόβλημα των ταλαντώσεων με απόσβεση στο μονοβάθμιο μηχανικό σύστημα μάζας–ελατηρίου–αποσβεστήρα σε μια ειδική περίπτωση για τη μετατόπιση $u(t)$ του υλικού σημείου. Εδώ θα κάνουμε τη γραφική παράσταση της μετατοπίσεως αυτής $u(t)$ για συγκεκριμένες τιμές της (κυκλικής) ιδιοσυχνότητας ω_0 ($\omega_0 = 10$) και του λόγου αποσβέσεως ξ ($\xi = 0.03$, μια συνηθισμένη τιμή σε κτίρια), πρώτα χωρίς καμία επιλογή στην εντολή **Plot**:

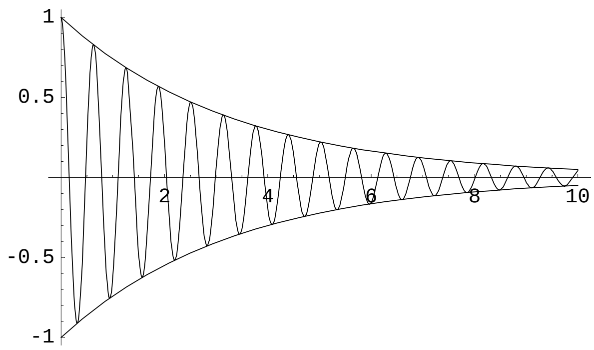
```
In[5]:= {ω0 = 10, ξ = 0.03, ωD = ω0 Sqrt[1 - ξ^2], u[t_] = ⅇ^(-ξ ω0 t) Cos[ωD t]};
```

```
In[6]:= p1 = Plot[u[t], {t, 0, 10}];
```



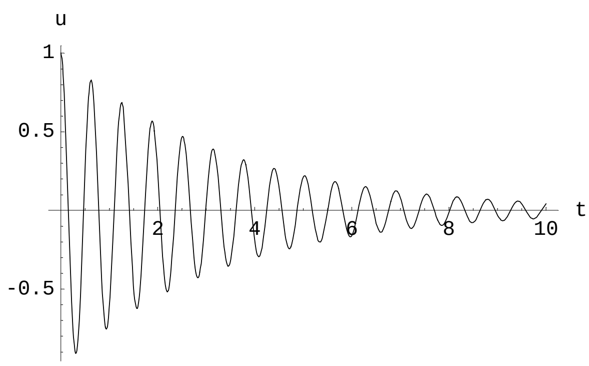
Με την εντολή **Plot** μπορούμε να σχεδιάζουμε και λίστα συναρτήσεων στην ίδια γραφική παράσταση. Για παράδειγμα, έχουμε τη δυνατότητα να σχεδιάσουμε όχι μόνο τη μετατόπιση $u(t)$ του υλικού σημείου στις παρούσες ταλαντώσεις με απόσβεση, αλλά ταυτόχρονα και τα προσημασμένα εύρη των ταλαντώσεων, όπου βέβαια (στα εύρη αυτά) δεν έχουμε εδώ τη συνημιτονική συνάρτηση. Να λοιπόν τρεις γραφικές παραστάσεις στο ίδιο σχήμα και μάλιστα με μία μόνο χρήση της εντολής **Plot**:

In[7]:= **p2** = **Plot**[{**u**[**t**], $e^{-\xi\,\omega_0\,t}$, $-e^{-\xi\,\omega_0\,t}$}, {**t**, **0**, **10**}];



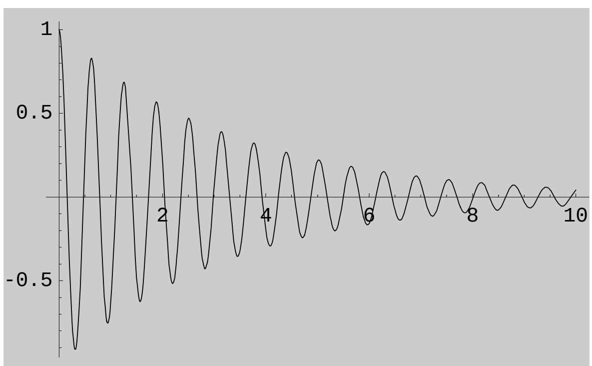
Με την επιλογή **AxesLabel** μπορούμε να βάλουμε ενδείξεις στους δύο άξονες της γραφικής παραστάσεως. Οι ενδείξεις αυτές είναι σε λίστα με δύο στοιχεία και το καθένα στοιχείο είναι συμβολοσειρά (string) που περικλείεται σε εισαγωγικά. (Για απλά σύμβολα τα εισαγωγικά μπορούν να παραλείπονται.)

In[8]:= **p3** = **Plot**[**u**[**t**], {**t**, **0**, **10**}, **AxesLabel** -> {**"t"**, **"u"**}];



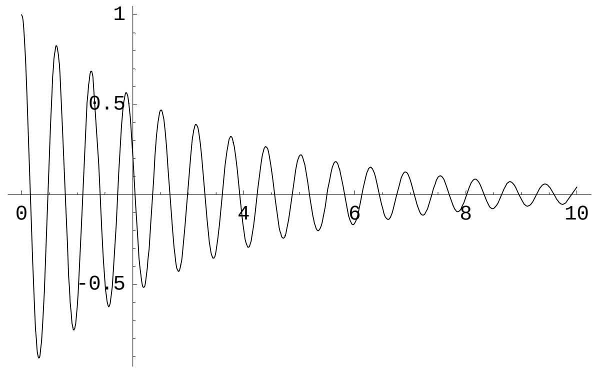
Με την επιλογή **Background** μπορούμε να έχουμε φόντο στη γραφική παράσταση είτε έγχρωμο είτε απλά γκρίζο (όπως εδώ). Σημειώνεται ότι όσο μεγαλύτερη είναι η τιμή **GrayLevel** (που παίρνει τιμές μεταξύ 0 και 1) στην επιλογή αυτή **Background** (όπως στο αμέσως πιο κάτω παράδειγμα), τόσο πιο ανοικτόχρωμο, πιο φωτεινό, λιγότερο μαύρο είναι το φόντο στη γραφική παράσταση.

In[9]:= **p4** = **Plot**[**u**[**t**], {**t**, **0**, **10**}, **Background** → **GrayLevel**[**0.8**]];



Με την επιλογή **AxesOrigin** μπορούμε εμείς να επιλέξουμε την αρχή των αξόνων όπου κατά τη γνώμη μας αυτή μας διευκολύνει, για παράδειγμα στο σημείο (2, 0):

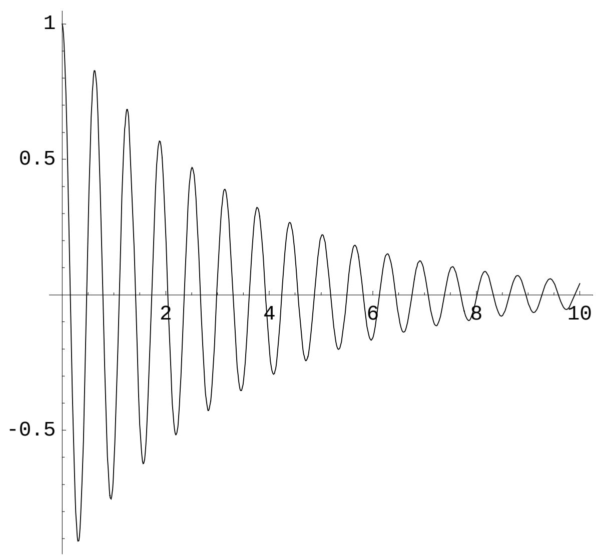
```
In[10]:= p5 = Plot[u[t], {t, 0, 10}, AxesOrigin → {2, 0}]
```




```
Out[10]= - Graphics -
```

Με την επιλογή **AspectRatio** μπορούμε να καθορίσουμε την πραγματική αναλογία ύψος προς πλάτος της γραφικής παραστάσεως. Στο παράδειγμα που δίνουμε παρακάτω η αναλογία αυτή καθορίσθηκε ίση με 1, δηλαδή τετραγωνική γραφική παράσταση. Η αρχική τιμή της *Mathematica* είναι **1/GoldenRatio** με

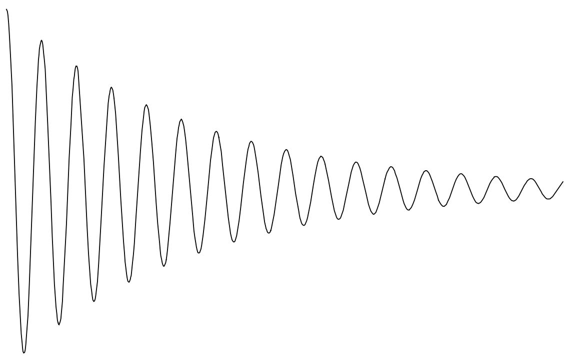
```
In[11]:= N[{GoldenRatio, r = (Sqrt[5] + 1) / 2}, 35]

Out[11]= {1.6180339887498948482045868343656381, 1.6180339887498948482045868343656381}

In[12]:= N[{1 / GoldenRatio, 1 / r}, 35]

Out[12]= {0.61803398874989484820458683436563812, 0.61803398874989484820458683436563812}

In[13]:= p6 = Plot[u[t], {t, 0, 10}, AspectRatio → 1]
```




```
Out[13]= - Graphics -
```

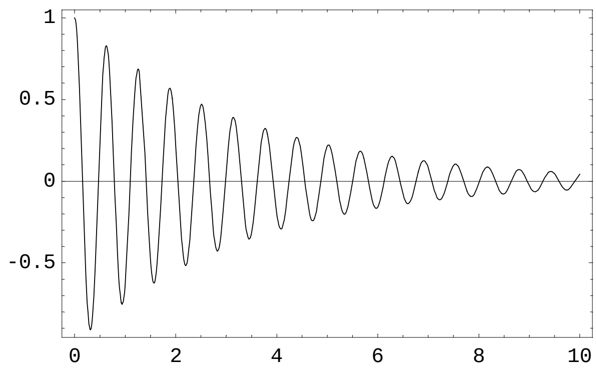
Με την επιλογή **Axes** μπορούμε να ζητήσουμε να μην εμφανίζονται καθόλου οι δύο άξονες στη γραφική παράσταση: **Axes → False**:

```
In[14]:= p7 = Plot[u[t], {t, 0, 10}, Axes → False];
```



Με την επιλογή **Frame** μπορούμε να έχουμε πλαίσιο, περίγραμμα στη γραφική παράσταση:

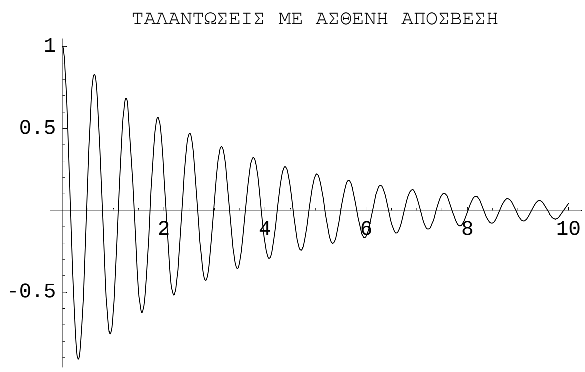
```
In[15]:= p8 = Plot[u[t], {t, 0, 10}, Frame → True]
```




```
Out[15]= - Graphics -
```

Με την επιλογή **PlotLabel** μπορούμε να βάλουμε τίτλο στη γραφική παράσταση:

```
In[16]:= p9 = Plot[u[t], {t, 0, 10}, PlotLabel → "ΤΑΛΑΝΤΩΣΕΙΣ ΜΕ ΑΣΘΕΝΗ ΑΠΟΣΒΕΣΗ"];
```



Με την επιλογή **DefaultFont** με αρχική τιμή **Courier** στις 10 στιγμές (points)

```
In[17]:= $DefaultFont

Out[17]= {Courier, 10.}
```

μπορούμε να καθορίσουμε κάποια άλλη γραμματοσειρά αντί για την **Courier** σε ολόκληρη τη γραφική παράσταση, π.χ. τη γνωστή γραμματοσειρά **Arial** και το μέγεθός της σε στιγμές (points), π.χ. 12 pt:

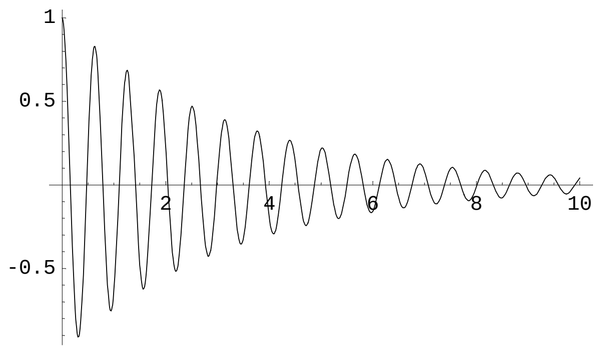
```
In[18]:= p10 = Plot[u[t], {t, 0, 10},
         PlotLabel → "ΤΑΛΑΝΤΩΣΕΙΣ ΜΕ ΑΣΘΕΝΗ ΑΠΟΣΒΕΣΗ", DefaultFont → {"Arial", 12}];
```

Με την επιλογή **DisplayFunction** να παίρνει την τιμή **Identity**, δηλαδή **DisplayFunction → Identity**, η γραφική παράσταση γίνεται εσωτερικά στη *Mathematica*, αλλά δεν εμφανίζεται στην οθόνη. Είναι όμως διαθέσιμη στη *Mathematica* και μπορεί να χρησιμοποιηθεί αργότερα, π.χ. μέσω της εντολής **Show**.

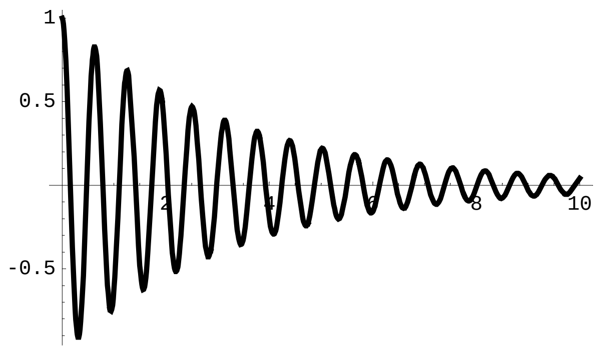
```
In[19]:= p11 = Plot[u[t], {t, 0, 10}, DisplayFunction → Identity]

Out[19]= - Graphics -
```

Με την επιλογή **PlotStyle** μπορούμε να καθορίσουμε εμείς (με τη σχετική τιμή **Thickness**) το πάχος της γραμμής της γραφικής παραστάσεως:

```
In[20]:= p12 = Plot[u[t], {t, 0, 10}];
```

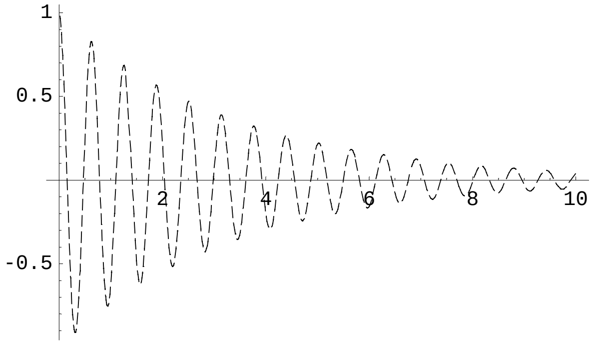



```
In[21]:= p13 = Plot[u[t], {t, 0, 10}, PlotStyle → Thickness[0.01]];
```



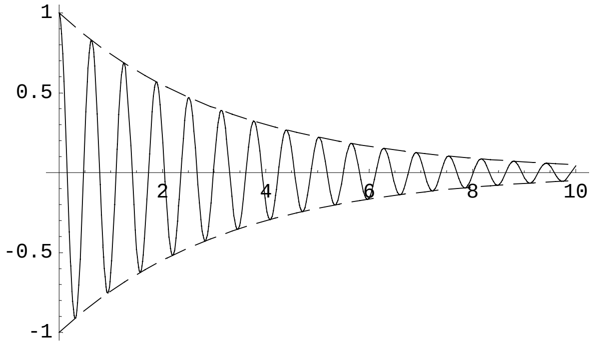
ή το να παρουσιάζεται η γραμμή αυτή διακεκομμένη μέσω της τιμής **Dashing** (σε λίστα με δύο στοιχεία):

```
In[22]:= p14 = Plot[u[t], {t, 0, 10}, PlotStyle → Dashing[{0.02, 0.01}]];
```



Στο παρακάτω σχήμα η κύρια γραφική παράσταση της μετατοπίσεως δεν είναι διακεκομμένη, ενώ οι δύο δευτερεύουσες γραφικές παραστάσεις για τα προσημασμένα εύρη της ταλαντώσεως είναι:

```
In[23]:= p15 = Plot[{u[t], ⅇ^(-ξ ω0 t), -ⅇ^(-ξ ω0 t)}, {t, 0, 10},
           PlotStyle → {Dashing[{1, 0}], Dashing[{0.04, 0.02}], Dashing[{0.04, 0.02}]}];
```

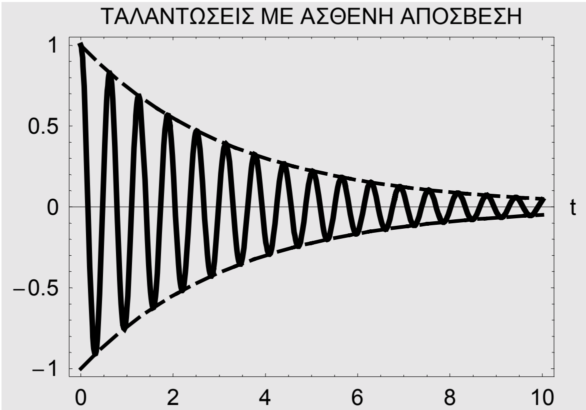


Εδώ δείχνουμε την ίδια γραφική παράσταση με έναν ολόκληρο συνδυασμό επιλογών ελπίζοντας ότι έτσι παίρνουμε μια αρκετά εμφανίσιμη και αξιοπρεπή γραφική παράσταση:

```
In[24]:= p16 = Plot[{u[t], ⅇ^(-ξ ω0 t), -ⅇ^(-ξ ω0 t)},
           {t, 0, 10}, AspectRatio → 0.7, AxesLabel -> {"t", "u(t)"},
           PlotLabel → "ΤΑΛΑΝΤΩΣΕΙΣ ΜΕ ΑΣΘΕΝΗ ΑΠΟΣΒΕΣΗ", DefaultFont → {"Arial", 11},
           PlotStyle → {Thickness[0.012], {Thickness[0.008], Dashing[{0.04, 0.02}]},
             {Thickness[0.008], Dashing[{0.04, 0.02}]}},
           Background → GrayLevel[0.85], Frame → False];
```

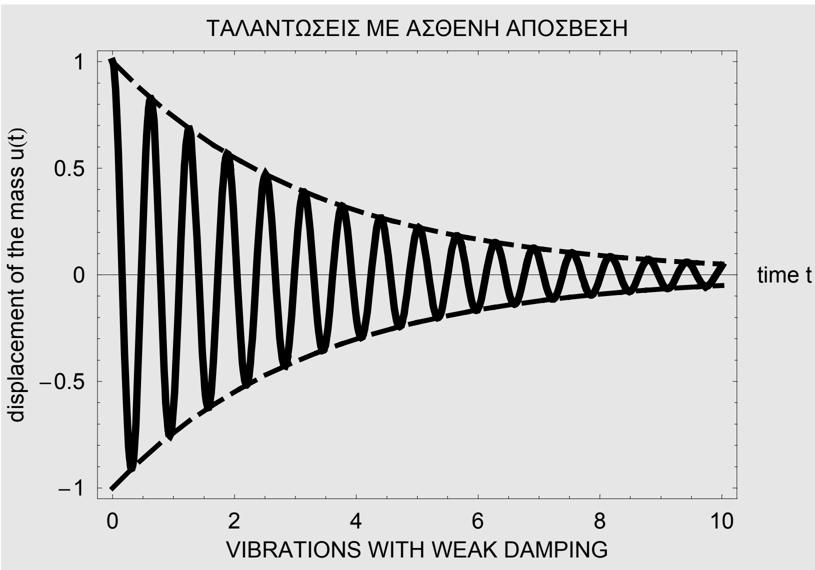
τώρα με τη χρήση και πλαισίου στη γραφική παράσταση

```
In[25]:= p17 =
   Plot[{u[t], e^(-ξ ω0 t), -e^(-ξ ω0 t)}, {t, 0, 10}, AspectRatio → 0.7, AxesLabel -> {"t", ""},
    PlotLabel → "ΤΑΛΑΝΤΩΣΕΙΣ ΜΕ ΑΣΘΕΝΗ ΑΠΟΣΒΕΣΗ", DefaultFont → {"Arial", 11},
    PlotStyle → {Thickness[0.012], {Thickness[0.008], Dashing[{0.04, 0.02}]},
      {Thickness[0.008], Dashing[{0.04, 0.02}]}},
    Background → GrayLevel[0.9], Frame → True];
```



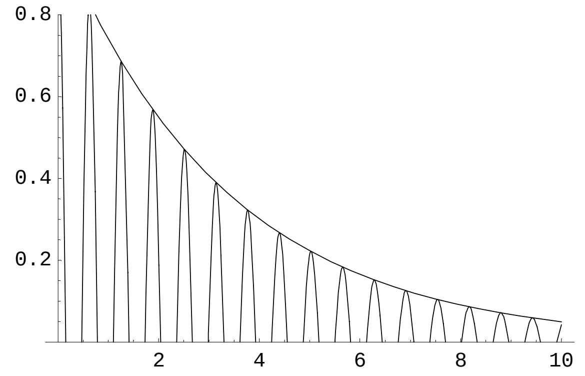
και σε μεγαλύτερο μέγεθος με περισσότερες ενδείξεις με την επιλογή **FrameLabel**. Αυτές οι ενδείξεις είναι σε λίστα τεσσάρων ενδείξεων: κάτω, αριστερά, πάνω και δεξιά, όπως φαίνεται στο παράδειγμα (με την τέταρτη ένδειξη κενή, για να μπει σ' αυτήν η ένδειξη του άξονα **time t** από την επιλογή **AxesLabel**). Αντίθετα στον αριστερό άξονα η ένδειξη που μπήκε προέρχεται από την επιλογή **FrameLabel** και όχι από την επιλογή **AxesLabel**.

```
In[26]:= p18 = Plot[{u[t], e^(-ξ ω0 t), -e^(-ξ ω0 t)}, {t, 0, 10}, AspectRatio → 0.7,
    AxesLabel -> {"time t", ""}, DefaultFont → {"Arial", 11},
    PlotStyle → {Thickness[0.012], {Thickness[0.008], Dashing[{0.04, 0.02}]},
      {Thickness[0.008], Dashing[{0.04, 0.02}]}},
    Background → GrayLevel[0.9], Frame → True, ImageSize → 400,
    FrameLabel → {"VIBRATIONS WITH WEAK DAMPING",
      "displacement of the mass u(t)", "ΤΑΛΑΝΤΩΣΕΙΣ ΜΕ ΑΣΘΕΝΗ ΑΠΟΣΒΕΣΗ", ""}];
```



Με την επιλογή **PlotRange** καθορίζουμε εμείς το διάστημα της γραφικής παραστάσεως στον κατακόρυφο άξονα:

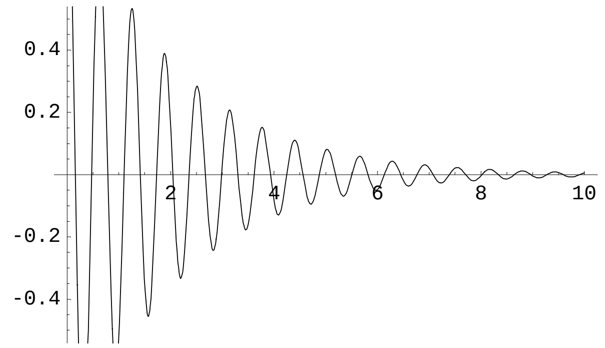
In[27]:= **p19 = Plot[{u[t], $e^{-\xi\,\omega_0\,t}$}, {t, 0, 10}, PlotRange → {0, 0.8}];**



Πολλές φορές η εντολή **Plot** της *Mathematica* επιλέγει να αποκόπτει μόνη της τμήματα από τη γραφική παράσταση, επειδή θεωρεί ότι οι τιμές της συναρτήσεως που παριστάνεται γραφικά είναι είτε πολύ μεγάλες είτε πολύ μικρές και επομένως το σχετικό τμήμα της γραφικής παραστάσεως δεν πρέπει να εμφανισθεί. Παράδειγμα με λόγο αποσβέσεως των ταλαντώσεων τώρα $\xi = 0.05$ αντί αρχικά $\xi = 0.03$:

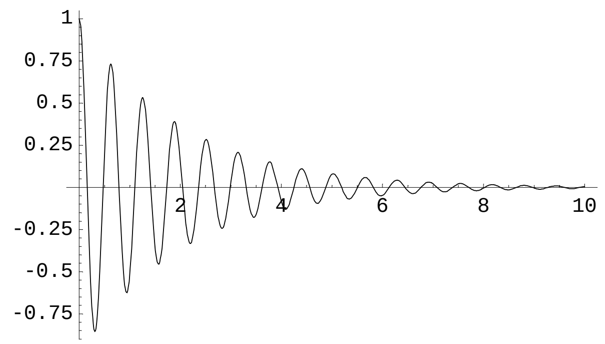
In[28]:= **{ξ = 0.05, ω_0 = 10, ω_D = ω_0 Sqrt[1 − ξ^2], u[t_] = $e^{-\xi\,\omega_0\,t}$ Cos[ω_D t]};**

In[29]:= **p20 = Plot[u[t], {t, 0, 10}];**



Σε τέτοιες περιπτώσεις η επιλογή **PlotRange → All** (η ίδια επιλογή **PlotRange**, που είχε προηγουμένως χρησιμοποιηθεί για τον καθορισμό του διαστήματος στον κατακόρυφο άξονα, αλλά τώρα με την τιμή **All**) μας επιτρέπει να έχουμε την πλήρη γραφική παράσταση χωρίς κοψίματά της:

In[30]:= **p21 = Plot[u[t], {t, 0, 10}, PlotRange → All];**



■ ΕΝΤΟΛΗ V2: ΔΙΔΙΑΣΤΑΤΗ ΓΡΑΦΙΚΗ ΠΑΡΑΣΤΑΣΗ ΜΕ ΣΚΙΑΣΗ

FilledPlot[*ΣυνάρτησηΜιαςΜεταβλητής*, {*Μεταβλητή*, *ΑρχικήΤιμή*, *ΤελικήΤιμή*}, *Επιλογή-1*, ...]

FilledPlot[*ΛίσταΣυναρτήσεωνΜιαςΜεταβλητής*, {*Μεταβλητή*,*ΑρχικήΤιμή*,*ΤελικήΤιμή*}, *Επιλογή-1*,...]

Πρόκειται για εντολή του πακέτου **Graphics‘FilledPlot‘**, που πρέπει να φορτωθεί πρώτα με την εντολή **Needs** ή με την ισοδύναμη εντολή **<<**. Η εντολή **FilledPlot** αυτού του πακέτου κάνει τη γραφική παράσταση της συναρτήσεως που δίνεται ή των δύο συναρτήσεων που δίνονται (σε λίστα) στο πρώτο όρισμά της. Η γραφική αυτή παράσταση γίνεται στο διάστημα της μεταβλητής το οποίο καθορίζεται στο δεύτερο όρισμά της. Το χαρακτηριστικό της εντολής αυτής που κάνει την ουσιαστική διαφορά της από την προηγούμενη εντολή **Plot** είναι η σκίαση, η κάλυψη, το "γέμισμα" της περιοχής ανάμεσα στην καμπύλη και στον άξονα της μεταβλητής (π.χ. τον άξονα x) με γκρίζο χρώμα ή με άλλο χρώμα. Αυτό ισχύει για την πρώτη μορφή της εντολής **FilledPlot**. Για τη δεύτερη μορφή της ίδιας εντολής η περιοχή που καλύπτεται με γκρίζο ή άλλο χρώμα είναι η περιοχή ανάμεσα στις καμπύλες που δίνονται στη λίστα του πρώτου ορίσματος και σχεδιάζονται. Προαιρετικά ακολουθεί μία επιλογή ή συνήθως περισσότερες επιλογές για τον τρόπο του υπολογισμού και κυρίως της εμφανίσεως της γραφικής παραστάσεως. Οι επιλογές αυτές αναφέρονται λίγο παρακάτω. Ουσιαστικά είναι οι ίδιες τριάντα επιλογές με εκείνες της εντολής **Plot** με την προσθήκη τριών νέων επιλογών: των τριών πρώτων επιλογών από τις παρακάτω επιλογές, από τις οποίες σημαντικότερη είναι η επιλογή **Fills**, η οποία καθορίζει τον τρόπο σκιάσεως, καλύψεως, "γεμίσματος" της περιοχής που θα καλυφθεί. Ακολουθούν το φόρτωμα του πακέτου **Graphics‘FilledPlot‘**, λίστα των επιλογών της εντολής του **FilledPlot** και παραδείγματα.

Φόρτωμα του πακέτου **Graphics‘FilledPlot‘** είτε με την πρώτη είτε με τη δεύτερη (όχι και με τις δύο μαζί) από τις δύο παρακάτω απόλυτα ισοδύναμες εντολές:

```
In[31]:= Needs["Graphics`FilledPlot`"]
```

```
In[32]:= << Graphics`FilledPlot`
```

Όλες οι επιλογές (33 συνολικά επιλογές) της εντολής **FilledPlot**. Οι καινούργιες επιλογές σχετικά με τη βασική εντολή **Plot** (αυτή με 30 επιλογές) είναι οι τρεις πρώτες με πιο σημαντική τους την επιλογή **Fills**.

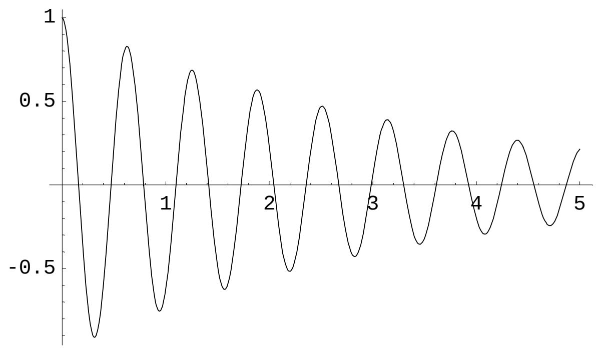
```
In[33]:= Options[FilledPlot]

Out[33]= {Fills → Automatic, Curves → Back, AxesFront → True, AspectRatio → 1/GoldenRatio,
  Axes → Automatic, AxesLabel → None, AxesOrigin → Automatic, AxesStyle → Automatic,
  Background → Automatic, ColorOutput → Automatic, Compiled → True,
  DefaultColor → Automatic, Epilog → {}, Frame → False, FrameLabel → None,
  FrameStyle → Automatic, FrameTicks → Automatic, GridLines → None,
  ImageSize → Automatic, MaxBend → 10., PlotDivision → 30., PlotLabel → None,
  PlotPoints → 25, PlotRange → Automatic, PlotRegion → Automatic,
  PlotStyle → Automatic, Prolog → {}, RotateLabel → True, Ticks → Automatic,
  DefaultFont :→ $DefaultFont, DisplayFunction :→ $DisplayFunction,
  FormatType :→ $FormatType, TextStyle :→ $TextStyle}
```

Θεωρούμε και πάλι το κλασικό για τον Πολιτικό Μηχανικό παράδειγμα των ταλαντώσεων με απόσβεση. (Πρόκειται για το ίδιο ουσιαστικά παράδειγμα με εκείνο στην προηγούμενη εντολή **Plot**.) Η συνάρτηση η οποία θα σχεδιασθεί είναι η μετατόπιση της μάζας του σχετικού μονοβάθμιου μηχανικού συστήματος, π.χ. η οριζόντια μετατόπιση της πλάκας μονώροφου ιδεατού κτιρίου διατμήσεως που προσεγγίζεται από αντίστοιχο πλαίσιο με τη μάζα του συγκεντρωμένη στο ύψος της πλάκας του μονώροφου κτιρίου:

In[34]:= $\{\omega_0 = 10,\ \xi = 0.03,\ \omega_D = \omega_0\ \mathrm{Sqrt}[1 - \xi^2],\ u[t\_] = e^{-\xi\,\omega_0\,t}\ \mathrm{Cos}[\omega_D\ t]\};$

Πρώτα–πρώτα παρουσιάζουμε τη συνηθισμένη γραφική παράσταση της μετατοπίσεως αυτής $u(t)$ με τη βασική εντολή **Plot** στο χρονικό διάστημα [0, 5]:

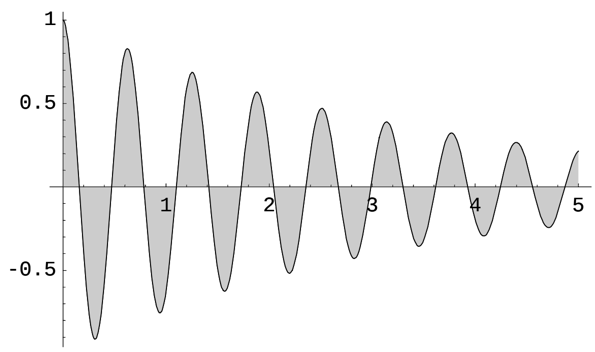
```
In[35]:= fp0 = Plot[u[t], {t, 0, 5}]
```




```
Out[35]= - Graphics -
```

Τώρα η ίδια γραφική παράσταση με την εντολή **FilledPlot** και με την επιλογή της **Fills** για τη σκίαση, την κάλυψη, το "γέμισμα", της περιοχής ανάμεσα στην καμπύλη η οποία σχεδιάζεται και στον άξονα του χρόνου t. Η κάλυψη αυτή γίνεται εδώ με ασθενές γκρίζο χρώμα μέσω της επιλογής **Fills → GrayLevel[0.8]**:

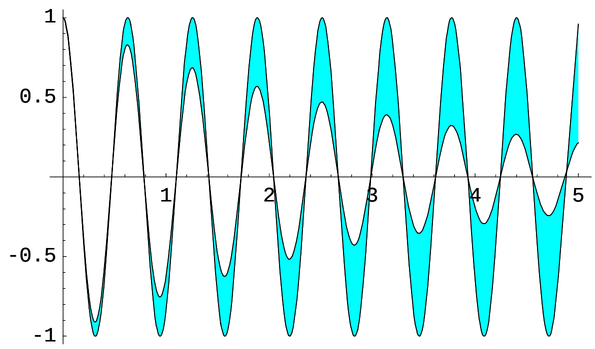
```
In[36]:= fp1 = FilledPlot[u[t], {t, 0, 5}, Fills -> GrayLevel[0.8]]
```




```
Out[36]= - Graphics -
```

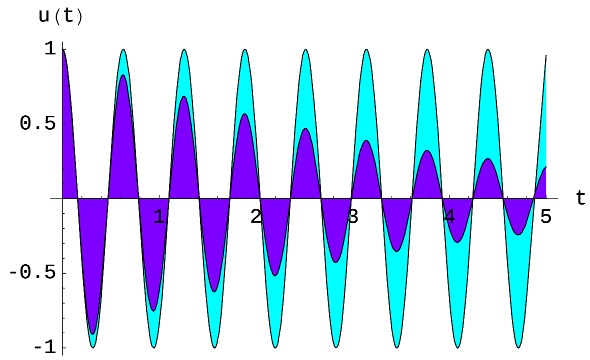
Στο επόμενο, το δεύτερο παράδειγμα της εντολής **FilledPlot** σχεδιάζεται επίσης και η μετατόπιση, όταν δεν υπάρχει απόσβεση: με $\xi = 0$. Η περιοχή που καλύπτεται είναι εκείνη ανάμεσα στην καμπύλη χωρίς απόσβεση (με σταθερό εύρος) και στην καμπύλη με απόσβεση (με εύρος που μειώνεται εκθετικά με το χρόνο t). Σημειώνεται ότι και οι τρεις επόμενες εντολές έχουν ; στο τέλος τους, ώστε να μην εμφανισθεί η ένδειξη **- Graphics -** μετά τη γραφική παράσταση. Αυτό μας είναι χρήσιμο εδώ για οικονομία χώρου στη σελίδα, δηλαδή για να χωρέσουν και οι τρεις επόμενες γραφικές παραστάσεις σε μία μόνο σελίδα. Σημειώνεται επίσης ότι οι δύο πρώτες γραφικές παραστάσεις της επόμενης σελίδας, οι **fp2** και **fp3**, έχουν έγχρωμη σκίαση (πράσινη η πρώτη και πράσινη/μωβ η δεύτερη), που δε φαίνεται όμως στην παρούσα ασπρόμαυρη εκτύπωση. (Η *Mathematica* διαθέτει και ειδικές δυνατότητες καθορισμού χρώματος.)

```
In[37]:= fp2 = FilledPlot[{u[t], Cos[ωD t]}, {t, 0, 5}];
```



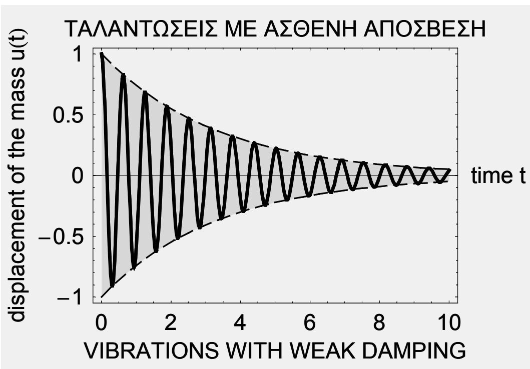
Και μια λίγο πιο ενδιαφέρουσα παραλλαγή της προηγούμενης γραφικής παραστάσεως με καλύψεις εδώ δύο περιοχών: (α) ανάμεσα στην καμπύλη χωρίς απόσβεση και στην καμπύλη με απόσβεση και παραπέρα (β) ανάμεσα στην καμπύλη με απόσβεση και στον άξονα του χρόνου *t* (δύο διαφορετικές καλύψεις):

```
In[38]:= fp3 = FilledPlot[{u[t], Cos[ωD t], u[t], 0}, {t, 0, 5}, AxesLabel → {"t", "u(t)"}];
```



Και τώρα η γραφική παράσταση **fp1** με πολλές επιλογές: όχι μόνο την πρώτη επιλογή **Fills** της εντολής **FilledPlot**, αλλά και πολλές άλλες επιλογές δανεισμένες στην εντολή **FilledPlot** από την εντολή **Plot**:

```
In[39]:= fp4 = FilledPlot[{ⅇ^(-ξ ω0 t), u[t], -ⅇ^(-ξ ω0 t)}, {t, 0, 10}, Frame → True,
          ImageSize → 260, AspectRatio → 0.7, AxesLabel -> {"time t", ""},
          DefaultFont → {"Arial", 11}, Background → GrayLevel[0.94],
          Fills → {GrayLevel[0.84], GrayLevel[0.84]},
          PlotStyle → {{Thickness[0.005], Dashing[{0.035, 0.02}]}, Thickness[0.010],
            {Thickness[0.005], Dashing[{0.035, 0.02}]}}, PlotRange → All,
          FrameLabel → {"VIBRATIONS WITH WEAK DAMPING", "displacement of the mass u(t)",
            "ΤΑΛΑΝΤΩΣΕΙΣ ΜΕ ΑΣΘΕΝΗ ΑΠΟΣΒΕΣΗ", ""}];
```



■ ΕΝΤΟΛΗ V3: ΓΡΑΦΙΚΗ ΠΑΡΑΣΤΑΣΗ ΠΕΠΛΕΓΜΕΝΗΣ ΣΥΝΑΡΤΗΣΕΩΣ

ImplicitPlot[*ΕξίσωσηΔύοΜεταβλητών*, {*Μεταβλητή-1, ΑρχικήΤιμή-1, ΤελικήΤιμή-1*}, *Επιλογές*]

ImplicitPlot[*ΛίσταΕξισώσεωνΔύοΜεταβλητών*,{*Μεταβλητή-1,ΑρχικήΤιμή-1,ΤελικήΤιμή-1*},*Επιλογές*]

**ImplicitPlot[*ΕξίσωσηΔύοΜεταβλητών*, {*Μεταβλητή-1, ΑρχικήΤιμή-1, ΤελικήΤιμή-1*},
{*Μεταβλητή-2, ΑρχικήΤιμή-2, ΤελικήΤιμή-2*}, *Επιλογές*]**

**ImplicitPlot[*ΛίσταΕξισώσεωνΔύοΜεταβλητών*, {*Μεταβλητή-1, ΑρχικήΤιμή-1, ΤελικήΤιμή-1*},
{*Μεταβλητή-2, ΑρχικήΤιμή-2, ΤελικήΤιμή-2*}, *Επιλογές*]**

Πρόκειται για εντολή του πακέτου **Graphics‘ImplicitPlot‘**, που πρέπει να φορτωθεί πρώτα με την εντολή **Needs** ή με την ισοδύναμη εντολή **<<** . Στην πρώτη μορφή της η εντολή **ImplicitPlot** αυτού του πακέτου σχεδιάζει τη γραφική παράσταση μιας συναρτήσεως $y = y(x)$ που ορίζεται σαν πεπλεγμένη συνάρτηση μέσω μιας εξισώσεως της μορφής $f(x, y) = 0$. Αυτή η εξίσωση δίνεται στο πρώτο όρισμα της εντολής. Η μεταβλητή, εδώ η x, και το διάστημα τιμών της (στον οριζόντιο άξονα), π.χ. από a έως b, καθορίζονται στο δεύτερο όρισμα. Τέλος προαιρετικά ακολουθούν και επιλογές (options) ανάλογες με εκείνες της εντολής **Plot**. Αυτές αφορούν στον τρόπο παρουσιάσεως της γραφικής παραστάσεως. Η δεύτερη μορφή της εντολής **ImplicitPlot** αποτελεί απλά γενίκευση της πρώτης μορφής της για τη σχεδίαση περισσότερων της μιας συναρτήσεων που ορίζονται σε πεπλεγμένη μορφή στο ίδιο σχήμα. Αντίστοιχες είναι και οι δύο τελευταίες μορφές της εντολής **ImplicitPlot**, όπου όμως καθορίζεται και το διάστημα της δεύτερης μεταβλητής (της συναρτήσεως στον κατακόρυφο άξονα) και επίσης χρησιμοποιείται εντελώς διαφορετικός αλγόριθμος δημιουργίας της γραφικής παραστάσεως. Πρώτα καλούμε (φορτώνουμε) το πακέτο **Graphics‘ImplicitPlot‘** με την εντολή

```
In[40]:= Needs["Graphics`ImplicitPlot`"]
```

Σαν παράδειγμα θεωρούμε τώρα το πρόβλημα μιας ισόπλευρης τριγωνικής πλάκας ύψους a και δυσκαμψίας D υπό ομοιόμορφη (σταθερή) κάθετη κατανεμημένη φόρτιση p_0. Στο ενδιαφέρον αυτό πρόβλημα του Πολιτικού Μηχανικού το βέλος κάμψεως (ή η βύθιση) της πλάκας $w = w(x, y)$ δίνεται από τον τύπο

In[41]:= $w[x\_, y\_] = (p_0 / (64\, D\, a))\ (x^3 - a\ (x^2 + y^2) - 3\, x\, y^2 + (4/27)\ a^3)\ ((4/9)\ a^2 - x^2 - y^2)\ /.\ \{p_0 \to 64,\ D \to 1,\ a \to 1\};$

In[42]:= $\{wmax = w[0, 0],\ N[wmax],\ N[wmax, 60]\}$

Out[42]= $\left\{\frac{16}{243},\ 0.0658436,\ 0.0658436213991769547325102880658436213991769547325102880658436\right\}$

Εδώ, για να κάνουμε τη γραφική παράσταση, υποθέσαμε ήδη συγκεκριμένες τιμές των σταθερών p_0, D και a. Η μέγιστη τιμή του βέλους κάμψεως παρουσιάζεται προφανώς στο κέντρο της πλάκας (λόγω της συμμετρίας στη γεωμετρία και στη φόρτιση), είναι η $w(0,0)$ και υπολογίσθηκε και αυτή. Η γραφική αυτή παράσταση θα αφορά σε ισοϋψείς καμπύλες της τριγωνικής πλάκας, δηλαδή σε καμπύλες με το ίδιο βέλος κάμψεως w_0. Οι σχετικές εξισώσεις, οι εξισώσεις των ισοϋψών καμπύλων, είναι προφανώς οι εξής:

```
In[43]:= TriangularPlateContour[x_, y_, w0_] = w[x, y] == w0
```

Out[43]= $\left(-x^2 - y^2 + \frac{4}{9}\right)\left(x^3 - x^2 - 3\, y^2\, x - y^2 + \frac{4}{27}\right) == w0$

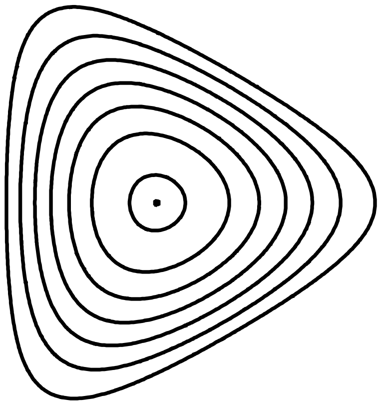
Με χρήση της εντολής **Table** παίρνουμε κατάλληλα επτά από τις ισοϋψείς αυτές καμπύλες, στις οποίες προσθέσαμε προσεκτικά και το κέντρο της τριγωνικής πλάκας, όπου έχουμε το μέγιστο βέλος κάμψεως

```
In[44]:= ListOfEquations =
          Append[Table[TriangularPlateContour[x, y, w0], {w0, 0.010, 0.064, 0.009}],
           TriangularPlateContour[x, y, wmax - 0.00001]];
```

και κάνουμε τη γραφική παράστασή τους (σε ένα ενιαίο σχήμα) με χρήση της εντολής **ImplicitPlot** στην τελευταία μορφή της. Έτσι μπορέσαμε να καθορίσουμε και το διάστημα τιμών στον κατακόρυφο άξονα: στον άξονα *y*. Έτσι καταφέραμε επίσης να περιορίσουμε τη γραφική παράσταστη στην αληθινή πλάκα:

```
In[45]:= ImplicitPlot[ListOfEquations, {x, -0.40, 0.50}, {y, -0.45, 0.45}, PlotPoints → 200,
          PlotStyle → Thickness[0.010], PlotLabel → "ΙΣΟΫΨΕΙΣ ΣΕ ΙΣΟΠΛΕΥΡΗ ΤΡΙΓΩΝΙΚΗ ΠΛΑΚΑ",
          DefaultFont → {"Arial-Bold", 9.2}, Axes → False, ImageSize → 205];
```

ΙΣΟΫΨΕΙΣ ΣΕ ΙΣΟΠΛΕΥΡΗ ΤΡΙΓΩΝΙΚΗ ΠΛΑΚΑ



■ ΕΝΤΟΛΗ V4: ΠΑΡΑΜΕΤΡΙΚΗ ΔΙΔΙΑΣΤΑΤΗ ΓΡΑΦΙΚΗ ΠΑΡΑΣΤΑΣΗ

**ParametricPlot[{*ΠαραμετρικήΕξίσωσηΜεΜίαΠαράμετρο-1,*
ΠαραμετρικήΕξίσωσηΜεΜίαΠαράμετρο-2 },**
{*Παράμετρος, ΑρχικήΤιμή, ΤελικήΤιμή*}, *Επιλογή-1, Επιλογή-2, . . .*]

**ParametricPlot[*ΛίσταΖευγώνΠαραμετρικώνΕξισώσεωνΜεΜίαΠαράμετρο,*
{*Παράμετρος, ΑρχικήΤιμή, ΤελικήΤιμή*}, *Επιλογή-1, Επιλογή-2, . . .*]**

Στην πρώτη μορφή της σχεδιάζει τη γραφική παράσταση συναρτήσεως που ορίζεται παραμετρικά στο πρώτο όρισμα μέσω λίστας δύο εξισώσεων $x = x(t)$ και $y = y(t)$, όπου *t* είναι η παράμετρος. Στο δεύτερο όρισμα ορίζονται η παράμετρος και το διάστημα μεταβολής της. Σε αντίθεση με τις δύο προηγούμενες εντολές **FilledPlot** και **ImplicitPlot** η εντολή αυτή **ParametricPlot** είναι μέρος του πυρήνα (των βασικών εντολών) της *Mathematica* και όχι κάποιου πακέτου της. Στη δεύτερη μορφή της απλά σχεδιάζει περισσότερες της μιας συναρτήσεις που ορίζονται παραμετρικά με την ίδια όμως παράμετρο *t* και στο ίδιο διάστημα της παραμέτρου. Στο τέλος μπορούν να μπουν και επιλογές (options) που συμπίπτουν μάλιστα με εκείνες της βασικής εντολής **Plot**, όπως φαίνεται με την εντολή

```
In[46]:= Options[ParametricPlot] == Options[Plot]

Out[46]= True
```

Σαν παράδειγμα στην παρούσα εντολή **ParametricPlot** δανειζόμαστε ένα παράδειγμα του Πολιτικού Μηχανικού από τη Θραυστομηχανική (ή Μηχανική της Θραύσεως). Συγκεκριμένα θεωρούμε μια ευθύγραμμη ρωγμή κατά μήκος του άξονα *Ox* σε ένα ελαστικό μέσον με εφελκυστική φόρτιση στο άπειρο κάθετα στη ρωγμή. Τότε με κατάλληλη πειραματική διάταξη που βασίζεται στη στοιχειώδη Οπτική δημιουργείται πάνω σε ένα πέτασμα μια χαρακτηριστική καμπύλη που καλείται ***καυστική***. Με μέτρηση ενός μήκους πάνω στην καυστική αυτή, συνήθως της μέγιστης διαστάσεώς της παράλληλα με τον κατακόρυφο άξονα *y*, προσδιορίζεται άμεσα ο συντελεστής εντάσεως τάσεων *K* στο άκρο της ρωγμής που θεωρούμε, εδώ στο δεξιό άκρο της. Υψηλές τιμές του συντελεστή εντάσεως τάσεων *K* οδηγούν σε διάδοση της ρωγμής και θραύση. Σύμφωνα με την ενδιαφέρουσα σχετική θεωρία των Manogg και Θεοχάρη (1964 έως 1980) η καυστική από ανάκλαση του φωτός πάνω στο δοκίμιο (ή από διάθλασή του) έχει τις εξής παραμετρικές εξισώσεις $x = x(\theta)$ και $y = y(\theta)$ με $-\pi < \theta < \pi$, που εδώ δίνονται σε αδιάστατη μορφή:

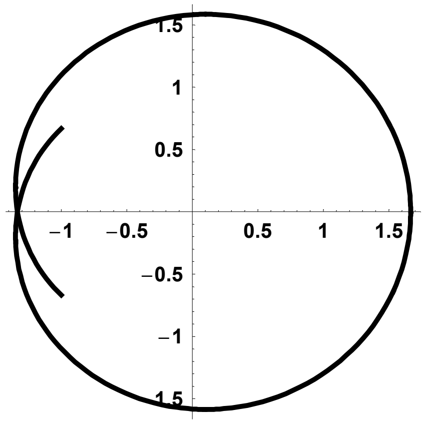
```
In[47]:= {x[θ_] = Cos[θ] + (2 / 3) Cos[3 θ / 2], y[θ_] = Sin[θ] + (2 / 3) Sin[3 θ / 2]};
```

Στ' αλήθεια δεν είναι εύκολο να απαλείψουμε την παράμετρο, που εδώ τη δηλώσαμε με θ και όχι με t, απλά επειδή σχετίζεται με την πολική γωνία θ στο επίπεδο του δοκιμίου. Δίνουμε και μερικές επιλογές:

```
In[48]:= CausticOptions = {PlotStyle → Thickness[0.012],
           PlotPoints → 200, PlotLabel → "ΚΑΥΣΤΙΚΗ ΣΤΟ ΑΚΡΟ ΡΩΓΜΗΣ",
           DefaultFont → {"Arial-Bold", 10}, AspectRatio → 1, ImageSize → 220};
```

και προχωράμε τώρα στη γραφική παράσταση της καυστικής μας "υπολογίζοντας" με **Evaluate** τις επιλογές μας. Σημειώνουμε ότι το διάστημα της παραμέτρου θ είναι $[-\pi, \pi]$, επειδή πρόκειται για την πολική γωνία πάνω στο δοκίμιο από την κάτω πλευρά της ρωγμής με $\theta = -\pi$ μέχρι και την πάνω πλευρά με $\theta = \pi$

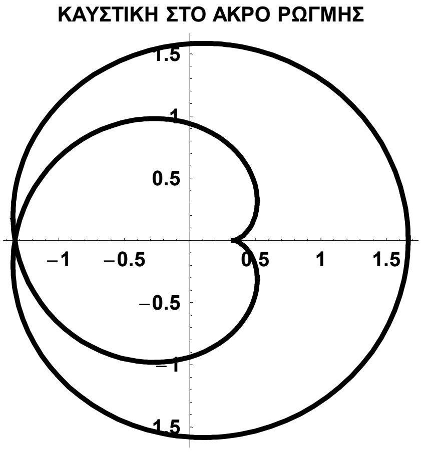
```
In[49]:= Caustic1 = ParametricPlot[{x[θ], y[θ]}, {θ, -π, π}, Evaluate[CausticOptions]];
```



Μπορούμε να πούμε ότι το διάστημα $[-\pi, \pi]$ της παραμέτρου θ αντιστοιχεί στην καυστική από ανάκλαση με θετικό συνολικό συντελεστή, ενώ τα διαστήματα $[-2\pi, -\pi]$ και επίσης $[\pi, 2\pi]$ (και τα δύο μαζί) στην ίδια καυστική αντιστοιχούν σε αρνητικό συνολικό συντελεστή. Όπως θα δούμε και στο επόμενο σχήμα, είναι προφανές ότι πρακτικά η αρχική καυστική (αυτή η οποία έχει θετικό συνολικό συντελεστή, η εξωτερική καυστική) είναι πιο ευδιάκριτη. Επομένως είναι και προτιμότερη στις πειραματικές μετρήσεις για τον προσδιορισμό του συντελεστή εντάσεως τάσεων *K*. Βέβαια όλα αυτά γενικεύονται και σε πολλά άλλα προβλήματα ρωγμών, μέσων με γωνιακά σημεία, κλπ. στη Θραυστομηχανική (ή Μηχανική της Θραύσεως).

Η πλήρης καυστική, σαν να είχαμε ανάκλαση του φωτός πάνω στο ρηγματωμένο δοκίμιο ταυτόχρονα και με θετικό συντελεστή μεγενθύνσεως και με αρνητικό συντελεστή μεγενθύνσεως, ενώ βέβαια αυτό δεν είναι από φυσικής απόψεως εφικτό, είναι η εξής, τώρα όμως με διάστημα της παραμέτρου θ το $[-2\pi, 2\pi]$:

```
In[50]:= Caustic2 = ParametricPlot[{x[θ], y[θ]}, {θ, -2 π, 2 π}, Evaluate[CausticOptions]];
```



■ ΕΝΤΟΛΗ V5: ΕΜΦΑΝΙΣΗ ΕΤΟΙΜΩΝ ΓΡΑΦΙΚΩΝ ΠΑΡΑΣΤΑΣΕΩΝ

Show[*ΓραφικήΠαράσταση-1, ΓραφικήΠαράσταση-2, . . . , Επιλογή-1, Επιλογή-2, . . .*]

Πρόκειται για εντολή που χρησιμοποιεί μία ή περισσότερες ήδη έτοιμες γραφικές παραστάσεις και τις παρουσιάζει. Μα αφού είναι έτοιμες; Πρώτα–πρώτα τις παρουσιάζει όλες μαζί, έστω και αν δημιουργήθηκαν από εντελώς διαφορετικές εντολές γραφικών παραστάσεων. Έπειτα μπορούμε έτσι να αλλάξουμε επιλογές και αυτό είναι χρήσιμο ακόμη και σε μία έτοιμη γραφική παράσταση. Άρα πρόκειται για μια ενδιαφέρουσα και χρήσιμη εντολή. Θα τη δούμε ξανά και στην επόμενη εντολή: τη **GraphicsArray**.

Σαν πρώτο παράδειγμα χρησιμοποιούμε την έτοιμη γραφική παράσταση της καυστικής σε άκρο ρωγμής που δημιουργήσαμε στην προηγούμενη εντολή, τώρα με νέο τίτλο, μέσα σε πλαίσιο και σε κάπως μικρότερο μέγεθος. Έχουμε έτσι νέες επιλογές χωρίς δεύτερη σχεδίαση του σχήματος: της καυστικής:

```
In[51]:= Show[Caustic1, PlotLabel → "ΚΑΥΣΤΙΚΗ ΑΠΟ ΑΝΑΚΛΑΣΗ", Frame → True, ImageSize → 180];
```

Μπορούμε επίσης να δημιουργήσουμε και ένα πρόχειρο σχήμα **Crack** για τη ρωγμή (crack) και να το παρουσιάσουμε μετά μαζί με την καυστική (caustic) **Caustic1**: η καυστική μαζί με τη ρωγμή στο ίδιο σχήμα!

```
In[52]:= Crack = Plot[0, {x, -1.35, -0.025},
           PlotStyle → Thickness[0.04], AspectRatio → 0.1, Ticks → {{}, {}}];
```

```
In[53]:= Show[Caustic1, Crack, ImageSize → 155];
```

■ ΕΝΤΟΛΗ V6: ΔΙΑΤΑΞΗ ΕΤΟΙΜΩΝ ΓΡΑΦΙΚΩΝ ΠΑΡΑΣΤΑΣΕΩΝ

GraphicsArray[{*ΓραφικήΠαράσταση-1, ΓραφικήΠαράσταση-2, ...*}]

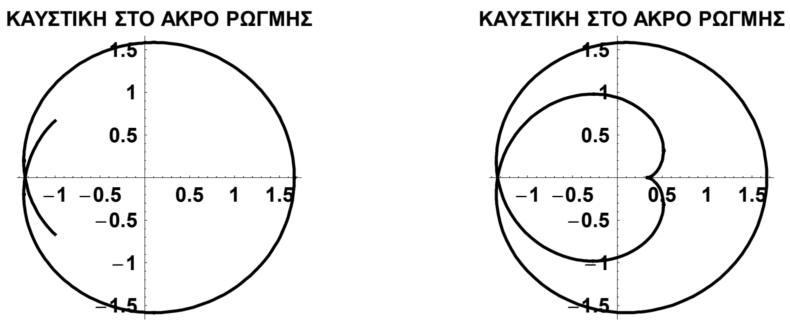
GraphicsArray[{{*ΓραφικήΠαράσταση-1*}, {*ΓραφικήΠαράσταση-2*}, ...}}]

GraphicsArray[{*ΛίσταΓραφικώνΠαραστάσεων-1, ΛίσταΓραφικώνΠαραστάσεων-2, ...*}]

Η εντολή αυτή δε δημιουργεί γραφικές παραστάσεις ούτε εμφανίζει γραφικές παραστάσεις που δημιουργήθηκαν με άλλες εντολές και ήδη υπάρχουν. Απλά τοποθετεί έτοιμες γραφικές παραστάσεις σε μία γραμμή (πρώτη μορφή της εντολής) ή σε μία στήλη (δεύτερη μορφή της εντολής) ή σε μία διάταξη πίνακα (τρίτη μορφή της εντολής με την πρώτη λίστα γραφικών παραστάσεων στην πρώτη γραμμή του πίνακα, τη δεύτερη λίστα στη δεύτερη γραμμή, κλπ.). Το αποτέλεσμα αυτής της εντολής **GraphicsArray** το χρησιμοποιεί έπειτα η προηγούμενη εντολή **Show** για την αληθινή εμφάνιση της γραμμής ή της στήλης ή του πίνακα (ορθογωνικής διατάξεως) γραφικών παραστάσεων στην οθόνη.

Σαν παράδειγμα θεωρούμε τις δύο γραφικές παραστάσεις καυστικών **Caustic1** και **Caustic2** που έχουμε ήδη δημιουργήσει με την προπροηγούμενη εντολή **ParametricPlot**. Εδώ με χρήση της πρώτης μορφής αυτής της εντολής **GraphicsArray** τις βάζουμε σε μία γραμμή και τις εμφανίζουμε με την εντολή **Show**

```
In[54]:= Show[GraphicsArray[{Caustic1, Caustic2}], GraphicsSpacing → 0.5, ImageSize → 408];
```



(Εδώ καθορίσαμε και την απόσταση των γραφικών παραστάσεων.) Ανάλογα με τη χρήση της δεύτερης μορφής της εντολής **GraphicsArray** μπορούμε να εμφανίσουμε τις ίδιες καυστικές σε μορφή στήλης:

```
In[55]:= Show[GraphicsArray[{{Caustic1}, {Caustic2}}], DisplayFunction → Identity];
```

Για να εμφανισθεί το αποτέλεσμα, απλά πρέπει να θέσουμε **DisplayFunction → $DisplayFunction**. Τέλος ένα παράδειγμα χρήσεως της τρίτης μορφής της ίδιας εντολής παρουσιάζεται στο τέλος του notebook.

■ ΕΝΤΟΛΗ V7: ΓΡΑΦΙΚΗ ΠΑΡΑΣΤΑΣΗ ΛΙΣΤΑΣ

ListPlot[*ΛίσταΑριθμών, Επιλογή-1, Επιλογή-2, . . .*]

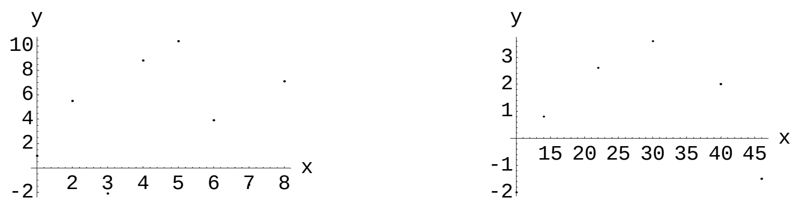
ListPlot[*ΛίσταΣημείων, Επιλογή-1, Επιλογή-2, . . .*]

Στην πρώτη, την πάνω μορφή της η εντολή **ListPlot** κάνει τη γραφική παράσταση της λίστας των n αριθμών y_k (που δίνεται) σαν να ήταν η λίστα σημείων $(k,\ y_k)$. Δηλαδή χρησιμοποιεί τους αριθμούς y_k σαν τεταγμένες αντιστοιχίζοντας αυθαίρετα σ' αυτούς τους θετικούς ακέραιους αριθμούς k σαν τετμημένες. Έτσι δημιουργούνται τα σημεία $(k,\ y_k)$, τα οποία και παριστάνονται τελικά στη γραφική παράσταση. Πρόκειται λοιπόν για γραφική παράσταση των n σημείων $(k,\ y_k)$. Αντίθετα στη δεύτερη μορφή της το όρισμα της εντολής **ListPlot** είναι κατευθείαν λίστα σημείων $(x_k,\ y_k)$. Δύο παραδείγματα: πρώτα με λίστα αριθμών **NumbersList** και αμέσως μετά με λίστα σημείων **PointsList**:

```
In[56]:= {NumbersList = {1.0, 5.5, -2.1, 8.8, 10.4, 3.9, -1.4, 7.1},
          PointsList = {{10, -2}, {14, 0.8}, {22, 2.6}, {30, 3.6}, {40, 2.0}, {46, -1.5}}};
```

```
In[57]:= {lp1 = ListPlot[NumbersList, AxesLabel → {"x", "y"}, DisplayFunction → Identity],
          lp2 = ListPlot[PointsList, AxesLabel → {"x", "y"}, DisplayFunction → Identity]};
```

```
In[58]:= Show[GraphicsArray[{lp1, lp2}], GraphicsSpacing → 0.5, ImageSize → 415];
```



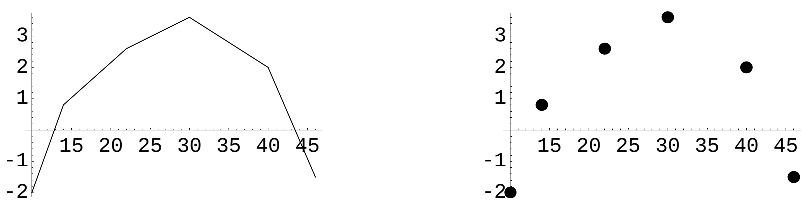
Η εντολή **ListPlot** δέχεται πολλές επιλογές (options), ακριβώς όπως και η εντολή **Plot**, που δε θεωρείται σκόπιμο να επαναληφθούν εδώ. Μια πολύ χρήσιμη από τις επιλογές αυτές είναι η επιλογή **PlotJoined** (αρχικά **PlotJoined → False**). Όταν όμως η επιλογή αυτή είναι **True**, δηλαδή **PlotJoined → True**, τότε τα σημεία της γραφικής παραστάσεως ενώνονται με ευθύγραμμα τμήματα. Παίρνουμε έτσι μια εποπτικά πολύ σαφέστερη εικόνα της γραφικής παραστάσεως των σημείων. Ξανά το προηγούμενο παράδειγμα:

```
In[59]:= lp3 = ListPlot[PointsList, PlotJoined → True, DisplayFunction → Identity];
```

Μια δεύτερη χρήσιμη δυνατότητα στην εντολή **ListPlot** αφορά στον καθορισμό του μεγέθους των κύκλων που παριστάνουν τα σημεία. Αυτό πετυχαίνεται εύκολα με τον καθορισμό του **PointSize** μέσω της επιλογής **PlotStyle**. (Δυστυχώς οι δύο πιο πάνω ενδιαφέρουσες δυνατότητες δε μπορούν να χρησιμοποιηθούν ταυτόχρονα.) Παράδειγμα τελικά με την εμφάνιση και των δύο γραφικών παραστάσεων:

```
In[60]:= lp4 = ListPlot[PointsList, PlotStyle → PointSize[0.04], DisplayFunction → Identity];
```

```
In[61]:= Show[GraphicsArray[{lp3, lp4}], GraphicsSpacing → 0.5, ImageSize → 415];
```



■ ΕΝΤΟΛΗ V8: ΓΡΑΦΙΚΗ ΠΑΡΑΣΤΑΣΗ ΙΣΟΣΤΑΘΜΙΚΩΝ ΚΑΜΠΥΛΩΝ

ContourPlot[*ΣυνάρτησηΔύοΜεταβλητών*, {*Μεταβλητή-1, ΑρχικήΤιμή-1, ΤελικήΤιμή-1*},
{*Μεταβλητή-2, ΑρχικήΤιμή-2, ΤελικήΤιμή-2*}, *Επιλογή-1, Επιλογή-2,* ...]

Σχεδιάζει τις ισοσταθμικές καμπύλες (ή καμπύλες στάθμης, που πολύ συχνά καλούνται και ισοϋψείς) της συναρτήσεως δύο μεταβλητών που δίνεται στο πρώτο όρισμα της εντολής στην περιοχή που δίνεται στο δεύτερο και στο τρίτο όρισμα της εντολής. Η εντολή αυτή δέχεται και επιλογές ανάλογες με την εντολή **Plot**, αλλά και μερικές ειδικές επιλογές. Έτσι π.χ. με την επιλογή **ContourLines** (αρχικά **True**) καθορίζεται εάν θα φαίνονται στη γραφική παράσταση οι ισοσταθμικές καμπύλες. Με την επιλογή **Contours** (αρχικά **10**) καθορίζεται ο αριθμός των ισοσταθμικών καμπύλων που θα σχεδιασθούν. Επίσης με την επιλογή **ContourShading** (αρχικά **True**) καθορίζεται εάν θα υπάρχει σκίαση μεταξύ των ισοσταθμικών καμπύλων. Αρκετά συχνά η σκίαση δεν είναι επιθυμητή, οπότε πρέπει να μπαίνει ρητά και η επιλογή **ContourShading → False**. Είναι βέβαια διαθέσιμες και πολλές ακόμη επιλογές.

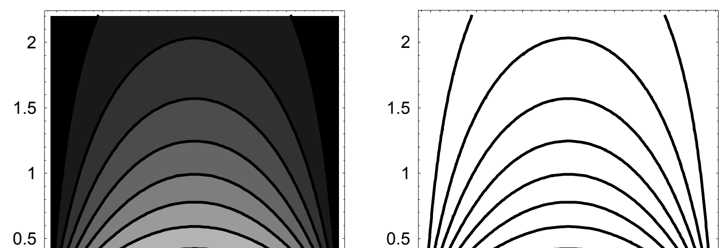
Σαν ένα σχετικό παράδειγμα αναφέρουμε το πολύ γνωστό θερμοκρασιακό πεδίο μέσα στην ημιλωρίδα $[-\pi/2, \pi/2] \times [0, \infty)$, όταν η θερμοκρασία $T(x, y)$ είναι 1 στη βάση της ημιλωρίδας και 0 στις δύο πλευρές της. Αυτό το πρόβλημα συνοριακών τιμών έχει την εξής λύση για τη θερμοκρασία $T(x, y)$ στην ημιλωρίδα

```
In[62]:= T[x_, y_] = (2 / π) ArcTan[Cos[x] / Sinh[y]];
```

Σχεδιάζουμε τις ισοσταθμικές καμπύλες (εδώ καλύτερα τις ισόθερμες ή ισοθερμοκρασιακές καμπύλες) στο παρόν πρόβλημα με την εντολή **ContourPlot** και με κατάλληλες επιλογές: (α) πρώτα με σκίαση μεταξύ των ισοσταθμικών καμπύλων και (β) μετά χωρίς σκίαση. Αφού σχηματίσουμε τις καμπύλες αυτές, τις παρουσιάζουμε σε ένα ενιαίο σχήμα με χρήση των ήδη γνωστών μας εντολών **Show** και **GraphicsArray**:

```
In[63]:= plot1 = ContourPlot[T[x, y], {x, -π / 2, π / 2},
           {y, 0.001, 2.2}, PlotPoints → 100, ContourStyle → Thickness[0.010],
           DefaultFont → {"Arial", 8.5}, DisplayFunction → Identity];

In[64]:= plot2 = ContourPlot[T[x, y], {x, -π / 2, π / 2}, {y, 0.001, 2.2},
           PlotPoints → 100, ContourStyle → Thickness[0.010], ContourShading → False,
           DefaultFont → {"Arial", 8.5}, DisplayFunction → Identity];

In[65]:= TemperatureContours = Show[GraphicsArray[{plot1, plot2}],
           PlotLabel → "ΙΣΟΘΕΡΜΕΣ ΚΑΜΠΥΛΕΣ ΣΕ ΗΜΙΛΩΡΙΔΑ", DefaultFont → {"Arial-Bold", 12},
           ImageSize → 412, DisplayFunction → $DisplayFunction];
```

ΙΣΟΘΕΡΜΕΣ ΚΑΜΠΥΛΕΣ ΣΕ ΗΜΙΛΩΡΙΔΑ



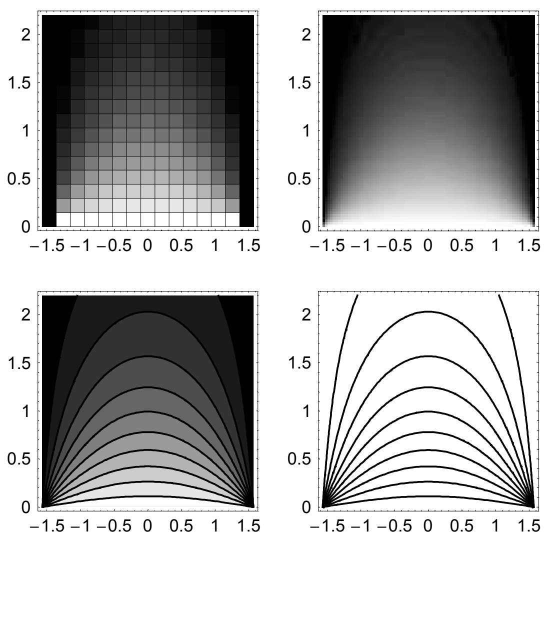
■ ΕΝΤΟΛΗ V9: ΣΧΕΔΙΑΣΗ ΔΙΑΓΡΑΜΜΑΤΟΣ ΠΥΚΝΟΤΗΤΑΣ

**DensityPlot[*ΣυνάρτησηΔύοΜεταβλητών*, {*Μεταβλητή-1, ΑρχικήΤιμή-1, ΤελικήΤιμή-1*},
{*Μεταβλητή-2, ΑρχικήΤιμή-2, ΤελικήΤιμή-2*}, *Επιλογή-1, Επιλογή-2,* . . .]**

Η εντολή αυτή σχεδιάζει το διάγραμμα πυκνότητας της συναρτήσεως δύο μεταβλητών που δίνεται στο πρώτο όρισμα της εντολής στην περιοχή που δίνεται στο δεύτερο και στο τρίτο όρισμα της εντολής. Λέγοντας διάγραμμα πυκνότητας εννούμε απλά τη σκίαση στο επίπεδο με σκούρα χρώματα για μικρές τιμές της συναρτήσεως που βαθμιαία γίνονται πιο ανοικτά όσο προχωράμε σε μεγάλες τιμές της συναρτήσεως. Έτσι παίρνουμε μια πρόχειρη διδιάστατη παράσταση της συναρτήσεως κάπως ανάλογη με εκείνη της εντολής **ContourPlot**, αλλ' εδώ υπολογιστικά πολύ πιο εύκολη. Η εντολή αυτή δέχεται και επιλογές ανάλογες με την εντολή **Plot**, αλλά και μερικές ειδικές επιλογές. Έτσι π.χ. με την επιλογή **Mesh** (αρχικά **True**) καθορίζεται εάν θα φαίνεται στο σχήμα το σχετικό ορθογώνιο πλέγμα. Συχνά δε θέλουμε να φαίνεται θέτοντας **Mesh → False**. Επίσης με την επιλογή **PlotPoints** καθορίζουμε την ανάλυση του διαγράμματος πυκνότητας που θέλουμε να δημιουργήσουμε.

Σαν παράδειγμα, συνεχίζουμε εδώ το παράδειγμα της προηγούμενης εντολής **ContourPlot** για το θερμοκρασιακό πεδίο σε ημιλωρίδα, τώρα βέβαια για το διάγραμμα πυκνότητας της θερμοκρασίας $T(x,\ y)$. Πρώτα με μικρή ανάλυση και πλέγμα και έπειτα με μεγάλη ανάλυση και χωρίς πλέγμα. Με χρήση των εντολών **Show** και **GraphicsArray** παρουσιάζουμε και τα δύο αυτά διαγράμματα πυκνότητας στο πάνω μέρος ενιαίου σχήματος. Στο κάτω μέρος του ίδιου ενιαίου σχήματος επαναλαμβάνουμε (για σύγκριση) τα διαγράμματα της προηγούμενης σελίδας, που είχαμε φτειάξει εκεί με τη συγγενή εντολή **ContourPlot**:

```
In[66]:= plot3 = DensityPlot[T[x, y], {x, -π/2, π/2}, {y, 0.001, 2.2},
           DefaultFont → {"Arial", 8.5}, DisplayFunction → Identity];

In[67]:= plot4 = DensityPlot[T[x, y], {x, -π/2, π/2}, {y, 0.001, 2.2}, PlotPoints → 200,
           DefaultFont → {"Arial", 8.5}, Mesh → False, DisplayFunction → Identity];

In[68]:= Show[GraphicsArray[{{plot3, plot4}, {plot1, plot2}}], ImageSize → 280,
          GraphicsSpacing → 0.07, DisplayFunction → $DisplayFunction];
```



■ *Notebook E10*
ΕΝΤΟΛΕΣ ΓΙΑ ΤΡΙΔΙΑΣΤΑΤΕΣ ΓΡΑΦΙΚΕΣ ΠΑΡΑΣΤΑΣΕΙΣ
2 ΕΝΤΟΛΕΣ: **W1. Plot3D, W2. ParametricPlot3D**

■ ΕΝΤΟΛΗ W1: ΤΡΙΔΙΑΣΤΑΤΗ ΓΡΑΦΙΚΗ ΠΑΡΑΣΤΑΣΗ

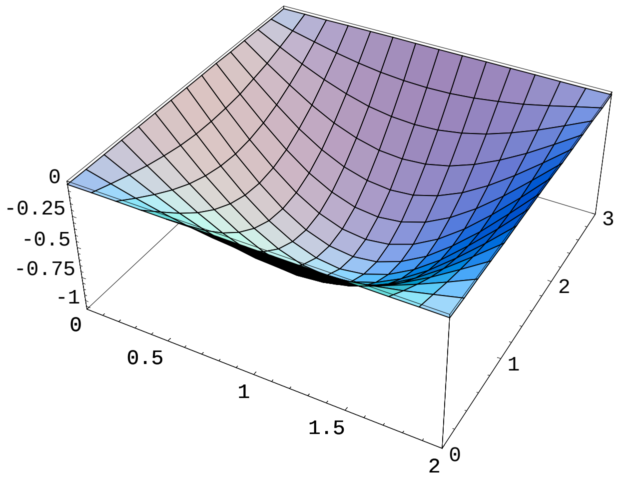
Plot3D[*ΣυνάρτησηΔύοΜεταβλητών, {Μεταβλητή-1, ΑρχικήΤιμή-1, ΤελικήΤιμή-1},*
**   *{Μεταβλητή-2, ΑρχικήΤιμή-2, ΤελικήΤιμή-2}, Επιλογή-1, Επιλογή-2, . . .*]**

Κάνει την τριδιάστατη γραφική παράσταση της συναρτήσεως των δύο μεταβλητών που δίνεται στο πρώτο όρισμα. Η γραφική αυτή παράσταση γίνεται στο διάστημα της πρώτης μεταβλητής το οποίο καθορίζεται στη λίστα του δευτέρου ορίσματος μετά την ίδια την πρώτη μεταβλητή και στο διάστημα της δεύτερης μεταβλητής που καθορίζεται στη λίστα του τρίτου ορίσματος μετά την ίδια τη δεύτερη μεταβλητή. Προαιρετικά ακολουθεί μία επιλογή ή συνήθως περισσότερες από μία επιλογές για τον τρόπο υπολογισμού και κυρίως εμφανίσεως της γραφικής παραστάσεως. Οι επιλογές (options) αυτές είναι εδώ 40, αρκετά περισσότερες από τις 30 επιλογές που έχει η εντολή **Plot**. Αμέσως πιο κάτω υπενθυμίζουμε τις επιλογές της εντολής **Plot** και έπειτα παραθέτουμε και τις 40 επιλογές της εντολής **Plot3D**. Και στις δύο περιπτώσεις η καθεμία επιλογή ακολουθείται από την αρχική τιμή της.

```
In[1]:= Options[Plot]

Out[1]= {AspectRatio → 1/GoldenRatio, Axes → Automatic, AxesLabel → None,
        AxesOrigin → Automatic, AxesStyle → Automatic, Background → Automatic,
        ColorOutput → Automatic, Compiled → True, DefaultColor → Automatic,
        Epilog → {}, Frame → False, FrameLabel → None, FrameStyle → Automatic,
        FrameTicks → Automatic, GridLines → None, ImageSize → Automatic,
        MaxBend → 10., PlotDivision → 30., PlotLabel → None, PlotPoints → 25,
        PlotRange → Automatic, PlotRegion → Automatic, PlotStyle → Automatic,
        Prolog → {}, RotateLabel → True, Ticks → Automatic, DefaultFont :→ $DefaultFont,
        DisplayFunction :→ $DisplayFunction, FormatType :→ $FormatType, TextStyle :→ $TextStyle}

In[2]:= Options[Plot3D]

Out[2]= {AmbientLight → GrayLevel[0], AspectRatio → Automatic, Axes → True,
        AxesEdge → Automatic, AxesLabel → None, AxesStyle → Automatic,
        Background → Automatic, Boxed → True, BoxRatios → {1, 1, 0.4}, BoxStyle → Automatic,
        ClipFill → Automatic, ColorFunction → Automatic, ColorFunctionScaling → True,
        ColorOutput → Automatic, Compiled → True, DefaultColor → Automatic,
        Epilog → {}, FaceGrids → None, HiddenSurface → True, ImageSize → Automatic,
        Lighting → True, LightSources → {{{1., 0., 1.}, RGBColor[1, 0, 0]},
          {{1., 1., 1.}, RGBColor[0, 1, 0]}, {{0., 1., 1.}, RGBColor[0, 0, 1]}},
        Mesh → True, MeshStyle → Automatic, Plot3Matrix → Automatic, PlotLabel → None,
        PlotPoints → 15, PlotRange → Automatic, PlotRegion → Automatic, Prolog → {},
        Shading → True, SphericalRegion → False, Ticks → Automatic, ViewCenter → Automatic,
        ViewPoint → {1.3, -2.4, 2.}, ViewVertical → {0., 0., 1.}, DefaultFont :→ $DefaultFont,
        DisplayFunction :→ $DisplayFunction, FormatType :→ $FormatType, TextStyle :→ $TextStyle}
```

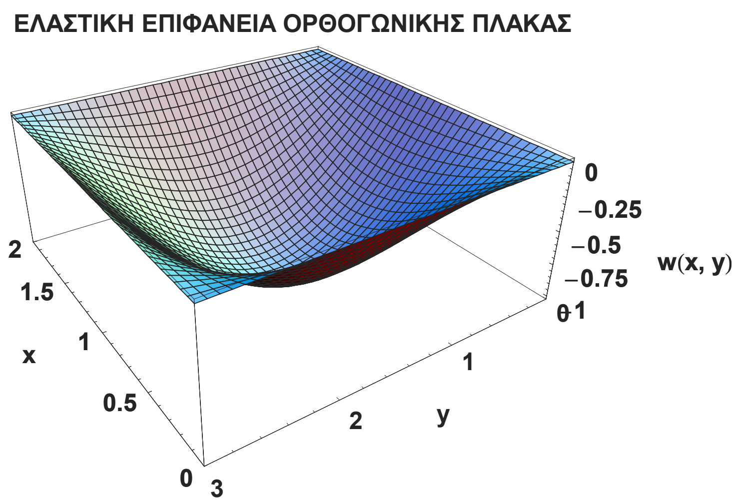
Αρκετές από τις επιλογές αυτές συμπίπτουν με τις αντίστοιχες επιλογές της εντολής **Plot**, ενώ μερικές άλλες είναι καινούργιες. Ορισμένες από αυτές θα τις επιδείξουμε σε ένα παράδειγμα παρακάτω. Ακολουθεί το παράδειγμα της εντολής **Plot3D**. Αυτό αφορά στη γραφική παράσταση της τριγωνομετρικής συναρτήσεως $-\sin(\pi x/2)\sin(\pi y/3)$. Η συνάρτηση αυτή δηλώνει το βέλος κάμψεως σε ορθογωνική πλάκα $P = [0, 2] \times [0, 3]$ με απλή στήριξη στο σύνορό της και υπό αντίστοιχη (απόλυτα ανάλογης μορφής) κατανεμημένη κάθετη εξωτερική φόρτιση. Το βέλος κάμψεως φαίνεται στην πιο κάτω γραφική παράσταση:

```
In[3]:= Plot3D[-Sin[π x / 2] Sin[π y / 3], {x, 0, 2}, {y, 0, 3}, ImageSize → 300];
```



Σημειώνουμε εδώ ότι ανάλογα με την εντολή **Plot**, εάν θέσουμε την Αγγλική άνω τελεία (το Ελληνικό ερωτηματικό) ; στο τέλος της εντολής **Plot3D**, απλά δεν εμφανίζεται η ένδειξη **- SurfaceGraphics -** αμέσως μετά τη γραφική παράσταση, ενώ εμφανίζεται η ίδια η γραφική παράσταση. Συχνά θέλουμε να χρησιμοποιούμε αυτήν τη δυνατότητα. Αυτό το κάνουμε και στα δύο παραδείγματα εδώ για την πλάκα. Παρακάτω χρησιμοποιούμε επιπλέον και ορισμένες επιλογές της εντολής **Plot3D**. Για τη συστηματική χρήση όλων αυτών των επιλογών γίνεται παραπομπή στο εγχειρίδιο και στη βοήθεια της *Mathematica*.

```
In[4]:= Plot3D[-Sin[π x / 2] Sin[π y / 3], {x, 0, 2}, {y, 0, 3},
          AxesLabel → {"x    ", "y", "          w(x, y)"}, PlotPoints → 40,
          PlotLabel → "ΕΛΑΣΤΙΚΗ ΕΠΙΦΑΝΕΙΑ ΟΡΘΟΓΩΝΙΚΗΣ ΠΛΑΚΑΣ", ViewPoint → {-1.5, 1, 1},
          DefaultFont → {"Arial-Bold", 12}, DefaultColor → GrayLevel[0.15], ImageSize → 360];
```



■ ΕΝΤΟΛΗ W2: ΠΑΡΑΜΕΤΡΙΚΗ ΤΡΙΔΙΑΣΤΑΤΗ ΓΡΑΦΙΚΗ ΠΑΡΑΣΤΑΣΗ

ParametricPlot3D[{*ΠαραμετρικήΕξίσωσηΜεΜίαΠαράμετρο-1,*
ΠαραμετρικήΕξίσωσηΜεΜίαΠαράμετρο-2,
***ΠαραμετρικήΕξίσωσηΜεΜίαΠαράμετρο-3*},**
{*Παράμετρος, ΑρχικήΤιμή, ΤελικήΤιμή*}, *Επιλογή-1, Επιλογή-2, ...*]

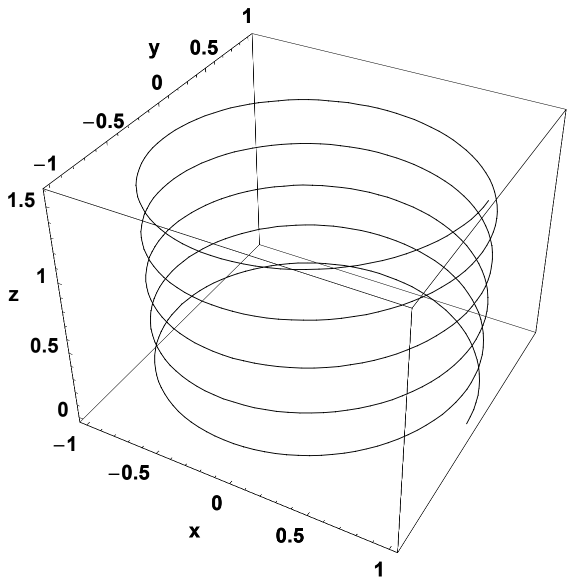
ParametricPlot3D[{*ΠαραμετρικήΕξίσωσηΜεΔύοΠαραμέτρους-1,*
ΠαραμετρικήΕξίσωσηΜεΔύοΠαραμέτρους-2,
***ΠαραμετρικήΕξίσωσηΜεΔύοΠαραμέτρους-3*},**
{*Παράμετρος-1, ΑρχικήΤιμή-1, ΤελικήΤιμή-1*},
{*Παράμετρος-2, ΑρχικήΤιμή-2, ΤελικήΤιμή-2*}, *Επιλογή-1, Επιλογή-2, ...*]

Στην πρώτη μορφή της (τη μονοπαραμετρική μορφή) η εντολή **ParametricPlot3D** σχεδιάζει στις τρεις διαστάσεις την καμπύλη $x = x(t)$, $y = y(t)$, $z = z(t)$ που καθορίζεται από τρεις παραμετρικές εξισώσεις με την ίδια παράμετρο t. Η γραφική αυτή παράσταση γίνεται στο διάστημα της παραμέτρου το οποίο καθορίζεται στη λίστα του δεύτερου ορίσματος της εντολής αμέσως μετά την ίδια την παράμετρο. Σαν σχετικό παράδειγμα παρουσιάζεται στο πάνω μέρος της επόμενης σελίδας ένα απλό ελατήριο. Στη δεύτερη μορφή της (τη διπαραμετρική μορφή) η ίδια εντολή **ParametricPlot3D** σχεδιάζει στις τρεις διαστάσεις την καμπύλη $x = x(s, t)$, $y = y(s, t)$, $z = z(s, t)$, η οποία καθορίζεται και πάλι από τρεις παραμετρικές εξισώσεις, αλλά τώρα με δύο παραμέτρους: τις s και t. Η γραφική αυτή παράσταση γίνεται στο διάστημα της πρώτης παραμέτρου που καθορίζεται στη λίστα του δεύτερου ορίσματος της εντολής αμέσως μετά την ίδια την πρώτη παράμετρο και στο διάστημα της δεύτερης παραμέτρου το οποίο καθορίζεται στη λίστα του τρίτου ορίσματος της εντολής αμέσως μετά την ίδια τη δεύτερη παράμετρο. Σαν σχετικό παράδειγμα παρουσιάζεται στο κάτω μέρος της επόμενης σελίδας ένα ημισφαίριο. Προαιρετικά και στις δύο μορφές της παρούσας εντολής **ParametricPlot3D** ακολουθεί μία επιλογή ή συνήθως περισσότερες από μία επιλογές. Αυτές αφορούν στον τρόπο υπολογισμού και κυρίως εμφανίσεως της γραφικής παραστάσεως. Οι επιλογές (options) αυτές είναι εδώ 36. Φυσικά και εδώ η καθεμία επιλογή ακολουθείται από την αρχική τιμή της που της έχει δώσει η *Mathematica.* (Εμείς βέβαια μπορούμε να τροποποιήσουμε αυτήν την αρχική τιμή της κάθε επιλογής.)

```
In[5]:= Options[ParametricPlot3D]

Out[5]= {AmbientLight → GrayLevel[0.], AspectRatio → Automatic, Axes → True,
  AxesEdge → Automatic, AxesLabel → None, AxesStyle → Automatic,
  Background → Automatic, Boxed → True, BoxRatios → Automatic, BoxStyle → Automatic,
  ColorOutput → Automatic, Compiled → True, DefaultColor → Automatic,
  Epilog → {}, FaceGrids → None, ImageSize → Automatic, Lighting → True,
  LightSources → {{{1., 0., 1.}, RGBColor[1, 0, 0]}, {{1., 1., 1.}, RGBColor[0, 1, 0]},
    {{0., 1., 1.}, RGBColor[0, 0, 1]}}, Plot3Matrix → Automatic, PlotLabel → None,
  PlotPoints → Automatic, PlotRange → Automatic, PlotRegion → Automatic,
  PolygonIntersections → True, Prolog → {}, RenderAll → True, Shading → True,
  SphericalRegion → False, Ticks → Automatic, ViewCenter → Automatic,
  ViewPoint → {1.3, -2.4, 2.}, ViewVertical → {0., 0., 1.}, DefaultFont :→ $DefaultFont,
  DisplayFunction :→ $DisplayFunction, FormatType :→ $FormatType, TextStyle :→ $TextStyle}
```

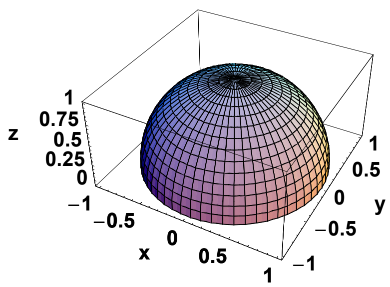
Σχεδίαση ενός ελατηρίου με πέντε σπείρες, αφού $10\pi = 5(2\pi)$:

```
In[6]:= ParametricPlot3D[{Cos[t], Sin[t], t / 20}, {t, 0, 10 π},
          PlotPoints → 500, AxesLabel → {"x", "y", "z"}, DefaultFont -> "Arial-Bold"];
```

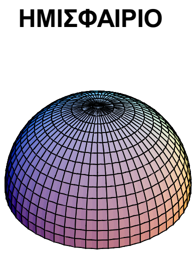


Με τις επιλογές **Boxed → False** και επίσης **Axes → False** μπορούμε να αποφύγουμε την εμφάνιση του "κουτιού" που περικλείει τη γραφική παράσταση καθώς και των αξόνων. Σχεδίαση ενός ημισφαιρίου με "κουτί" και άξονες και αμέσως πιο κάτω χωρίς "κουτί" και χωρίς άξονες, αλλά με τον τίτλο **ΗΜΙΣΦΑΙΡΙΟ**:

```
In[7]:= ParametricPlot3D[{Cos[θ] Cos[ϕ], Sin[θ] Cos[ϕ], Sin[ϕ]},
          {θ, 0, 2 π}, {ϕ, 0, π}, ImageSize → 190, PlotPoints → 30,
          AxesLabel → {"x    ", "     y", "z          "}, DefaultFont -> "Arial-Bold"];
```




```
In[8]:= ParametricPlot3D[{Cos[θ] Cos[ϕ], Sin[θ] Cos[ϕ], Sin[ϕ]}, {θ, 0, 2 π}, {ϕ, 0, π},
          ImageSize → 162, PlotPoints → 30, Boxed → False, Axes → False,
          PlotLabel → "ΗΜΙΣΦΑΙΡΙΟ", DefaultFont → {"Arial-Bold", 12}];
```



■ *Notebook E11*
ΕΝΤΟΛΕΣ ΓΙΑ ΛΟΓΙΚΟΥΣ ΥΠΟΛΟΓΙΣΜΟΥΣ

5 ΕΝΤΟΛΕΣ: **G1. And, G2. Or, G3. Not, G4. Implies, G5. LogicalExpand**

■ ΕΝΤΟΛΗ G1: ΛΟΓΙΚΟ ΚΑΙ

And[*Παράσταση-1, Παράσταση-2, Παράσταση-3, ...*]

ή ***Παράσταση-1* && *Παράσταση-2* && *Παράσταση-3* && ...**

ή ***Παράσταση-1* ∧ *Παράσταση-2* ∧ *Παράσταση-3* ∧ ...**

Δίνει αποτέλεσμα **True**, εάν όλες ανεξαιρέτως οι παραστάσεις είναι αληθείς (true), και **False**, εάν έστω και μία παράσταση είναι ψευδής (false). Ισοδύναμα σύμβολα είναι τα **&&** και ∧ . (Από μαθηματικής–λογικής απόψεως προτιμάται το δεύτερο σύμβολο, το ∧ . Ανάλογα ισχύουν και στις επόμενες δύο εντολές, όπου προτιμώνται τα σύμβολα ∨ και ¬ αντίστοιχα.) Παραδείγματα:

```
In[1]:= {And[A == A, B == B, C == C, D == D], x^2 == x x ⋀ 3 == 2 + 1 ⋀ 3 < 5, 5 > 3 ∧ 5 < 1, 2 < 3 ∧ 3 < 4}
Out[1]= {True, True, False, True}

In[2]:= {Beam[B] = True, Elastic[B] = True, Beam[B] ∧ Elastic[B]}
Out[2]= {True, True, True}
```

■ ΕΝΤΟΛΗ G2: ΛΟΓΙΚΟ Ή

Or[*Παράσταση-1, Παράσταση-2, Παράσταση-3, ...*]

ή ***Παράσταση-1* ¦¦ *Παράσταση-2* ¦¦ *Παράσταση-3* ¦¦ ...**

ή ***Παράσταση-1* ∨ *Παράσταση-2* ∨ *Παράσταση-3* ∨ ...**

Δίνει αποτέλεσμα **True**, εάν έστω και μία παράσταση είναι αληθής (true), και **False**, εάν όλες ανεξαιρέτως οι παραστάσεις είναι ψευδείς (false). Ισοδύναμα σύμβολα είναι τα ¦¦ και ∨. Παραδείγματα:

```
In[3]:= {Or[2 > 5, 2 < 5], x^2 == x x ⋁ 3 == 2 + 2 ⋁ 3 > 5, 5 < 1 ∨ 5 < 2 ∨ 5 < 3 , (A == A) ∨ ( A ≠ A)}
Out[3]= {True, True, False, True}
```

■ ΕΝΤΟΛΗ G3: ΛΟΓΙΚΟ ΟΧΙ

Not[*Παράσταση*] ή **! *Παράσταση*** ή ***¬Παράσταση***

Δίνει αποτέλεσμα **True**, εάν η παράσταση είναι ψευδής (false), και **False**, εάν η παράσταση είναι αληθής (true). Ισοδύναμα σύμβολα είναι τα **!** και ¬ . Παραδείγματα:

```
In[4]:= {2 < 5, Not[ 2 < 5], 3 == 2 + 2, Not[3 == 2 + 2], A == A, ! (A == A), A ≠ A, ! (A != A), ¬ (A ≠ A)}
Out[4]= {True, False, False, True, True, False, False, True, True}
```

■ ΕΝΤΟΛΗ G4: ΛΟΓΙΚΗ ΣΥΝΕΠΑΓΩΓΗ

Implies[*Παράσταση-1, Παράσταση-2*] ή ***Παράσταση-1 ⇒ Παράσταση-2***

Αφορά στη λογική συνεπαγωγή. Δίνει αποτέλεσμα **True** εκτός κι αν η πρώτη παράσταση είναι αληθής (true), ενώ η δεύτερη παράσταση είναι ψευδής (false). Παραδείγματα:

```
In[5]:= {Implies[(1 < 2) ∧ (2 < 3), 1 < 3], Implies[2 == 1 + 1, 1 + 1 == 2], 2 < 3 ∧ 3 < 4 ⇒ 2 > 4}
Out[5]= {True, True, False}
```

■ ΕΝΤΟΛΗ G5: ΛΟΓΙΚΟ ΑΝΑΠΤΥΓΜΑ

LogicalExpand[*Παράσταση*]

Αναπτύσσει λογικά παραστάσεις που περιέχουν λογικούς συνδέσμους, όπως το **And**, το **Or**, το **Not** και το **Implies**. Αυτό έχει συχνά σαν συνέπεια την απλοποίηση των παραστάσεων αυτών. Παραδείγματα:

```
In[6]:= LogicalExpand[PartialDifferentialEquation[eqn] ⇒ DifferentialEquation[eqn]]
Out[6]= DifferentialEquation[eqn] || ! PartialDifferentialEquation[eqn]

In[7]:= {implication = Building[bld] ⇒ Structure[bld], LogicalExpand[implication]}
Out[7]= {Implies[Building[bld], Structure[bld]], ! Building[bld] || Structure[bld]}

In[8]:= {e1 = Implies[And[A, B], Or[B, C]], TraditionalForm[e1], LogicalExpand[e1]}
Out[8]= {Implies[A && B, B || C], (A ∧ B) ⇒ (B ∨ C), True}

In[9]:= {e2 = (A ∧ B) ∨ A ∨ B, e3 = LogicalExpand[e2], TraditionalForm[{e2, e3}]}
Out[9]= {A && B || A || B, A || B, {A ∧ B ∨ A ∨ B, A ∨ B}}

In[10]:= {e4 = ((A ⇒ B) ∧ (B ⇒ C)) ⇒ (A ⇒ C), e4 // TraditionalForm, e4 // LogicalExpand}
Out[10]= {Implies[Implies[A, B] && Implies[B, C], Implies[A, C]], (A ⇒ B ∧ B ⇒ C) ⇒ (A ⇒ C), True}

In[11]:= {i1 = Implies[A ∧ A ∧ A ∧ A, A], LogicalExpand[i1], A ∧ A ∧ A ∧ A ⇒ A // LogicalExpand}
Out[11]= {Implies[A && A && A && A, A], True, True}
```

Ένα απλό παράδειγμα με γέφυρες μεγάλου μήκους:

```
In[12]:= LongBridge[A_] = long[A] ∧ Bridge[A]; LongBridge[B] ⇒ Bridge[B] // LogicalExpand
Out[12]= True
```

Ένα απλό παράδειγμα με γραμμικές και μη γραμμικές διαφορικές εξισώσεις με εφαρμογή του στην εξίσωση του κύματος (wave equation):

```
In[13]:= NonlinearDE[A_] = ¬ LinearDE[A];

In[14]:= {i2 = Implies[¬ NonlinearDE[WaveDE], LinearDE[WaveDE]], i2 // LogicalExpand}
Out[14]= {Implies[LinearDE[WaveDE], LinearDE[WaveDE]], True}
```

■ *Notebook E12*
ΕΝΤΟΛΕΣ ΓΙΑ ΤΗ ΔΙΑΝΥΣΜΑΤΙΚΗ ΑΝΑΛΥΣΗ

11 ΕΝΤΟΛΕΣ: **Δ1. Cross, Δ2. SetCoordinates, Δ3. CoordinateSystem**
Δ4. Grad, Δ5. Div, Δ6. Curl, Δ7. Laplacian, Δ8. Biharmonic
Δ9. PlotVectorField, Δ10. PlotGradientField, Δ11. PlotHamiltonianField

■ ΕΝΤΟΛΗ Δ1: ΕΞΩΤΕΡΙΚΟ ΓΙΝΟΜΕΝΟ ΔΙΑΝΥΣΜΑΤΩΝ

Cross[*Διάνυσμα1, Διάνυσμα2*] ή ***Διάνυσμα1*** × ***Διάνυσμα2*** (το σύμβολο × με **Esc cross Esc**)

Υπολογίζει το εξωτερικό γινόμενο **A** × **B** δύο διανυσμάτων **A** (το πρώτο διάνυσμα) και **B** (το δεύτερο διάνυσμα). Συχνά είναι καλύτερη η χρήση της δεύτερης μορφής της εντολής αυτής με το σύμβολο × . Για το εσωτερικό γινόμενο **AB** χρησιμοποιείται απλά η τελεία . αντί για το σύμβολο × . (Σημειώνεται ότι η εντολή αυτή **Cross** ανήκει στον πυρήνα της *Mathematica*. Έτσι δε χρειάζεται να φορτώσουμε κάποιο πακέτο της πριν τη χρησιμοποιήσουμε.) Παραδείγματα:

```
In[1]:= {A = {1, 2, 3}, B = {4, 5, 6}, Cross[A, B], A×B, Cross[{1, 2, 3}, {4, 5, 6}]}
Out[1]= {{1, 2, 3}, {4, 5, 6}, {-3, 6, -3}, {-3, 6, -3}, {-3, 6, -3}}
In[2]:= {A.B, A×B, B×A, A×B == B×A, A×B == -B×A, A.(A×B), A×A, B×B}
Out[2]= {32, {-3, 6, -3}, {3, -6, 3}, False, True, 0, {0, 0, 0}, {0, 0, 0}}
In[3]:= {v1 = Cross[{a1, a2, a3}, {b1, b2, b3}], v2 = {a1, a2, a3}×{b1, b2, b3}, v1 == v2}
Out[3]= {{-a3 b2 + a2 b3, a3 b1 - a1 b3, -a2 b1 + a1 b2}, {-a3 b2 + a2 b3, a3 b1 - a1 b3, -a2 b1 + a1 b2}, True}
In[4]:= {Av = {a1, a2, a3}, Bv = {b1, b2, b3}, Cv = {c1, c2, c3}};
In[5]:= q1 = (Av×Bv).Cv; q2 = Av.(Bv×Cv); q3 = Det[{Av, Bv, Cv}]; {q1, q1 == q2 == q3} // Simplify
Out[5]= {a3 (-b2 c1 + b1 c2) + a2 (b3 c1 - b1 c3) + a1 (-b3 c2 + b2 c3), True}
```

■ ΕΝΤΟΛΗ Δ2: ΚΑΘΟΡΙΣΜΟΣ ΣΥΣΤΗΜΑΤΟΣ ΣΥΝΤΕΤΑΓΜΕΝΩΝ

SetCoordinates[*ΣύστημαΣυντεταγμένων*[*Συντεταγμένες*]]

Καθορίζει το σύστημα συντεταγμένων και τις επιθυμητές συντεταγμένες που θα χρησιμοποιηθούν μέχρι να αλλάξουν. Τα κυριότερα συστήματα συντεταγμένων είναι οι Καρτεσιανές συντεταγμένες, (Cartesian coordinates), οι κυλινδρικές συντεταγμένες (cylindrical coordinates) και τέλος οι σφαιρικές συντεταγμένες (spherical coordinates). Όταν θέλουμε να χρησιμοποιήσουμε πολικές συντεταγμένες στη *Mathematica*, καθορίζουμε αντί γι' αυτές κυλινδρικές συντεταγμένες υποθέτοντας σιωπηλά πως η μεταβλητή *z* θα είναι απούσα. *Παρατήρηση:* Η εντολή αυτή και οι επόμενες έξι εντολές στο παρόν notebook (μέχρι και την εντολή Δ8) απαιτούν τη χρήση του πακέτου **VectorAnalysis** της *Mathematica*.

Δηλαδή δεν ανήκουν στον πυρήνα της *Mathematica,* ενώ αντίθετα η προηγούμενη εντολή **Cross** για το εξωτερικό γινόμενο ανήκει. Έτσι πρέπει πρώτα να φορτώσουμε το πακέτο αυτό **VectorAnalysis**:

```
In[6]:= << Calculus`VectorAnalysis`

In[7]:= SetCoordinates[Cartesian[x, y, z]]

Out[7]= Cartesian[x, y, z]

In[8]:= SetCoordinates[Cylindrical[r, θ, z]]

Out[8]= Cylindrical[r, θ, z]

In[9]:= SetCoordinates[Spherical[ρ, θ, ϕ]]

Out[9]= Spherical[ρ, θ, ϕ]
```

■ ΕΝΤΟΛΗ Δ3: ΑΝΑΦΟΡΑ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ ΣΥΝΤΕΤΑΓΜΕΝΩΝ

CoordinateSystem

Η εντολή αυτή (χωρίς κανένα όρισμα σε αγκύλες) απλά αναφέρει σαν αποτέλεσμά της (έξοδό της) το σύστημα συντεταγμένων το οποίο ισχύει κατά τη στιγμή της εκτελέσεώς της. Παραδείγματα:

```
In[10]:= CoordinateSystem

Out[10]= Spherical

In[11]:= {SetCoordinates[Cylindrical[r, θ, z]], CoordinateSystem}

Out[11]= {Cylindrical[r, θ, z], Cylindrical}

In[12]:= {SetCoordinates[Cartesian[x, y, z]], CoordinateSystem}

Out[12]= {Cartesian[x, y, z], Cartesian}
```

■ ΕΝΤΟΛΗ Δ4: ΚΛΙΣΗ (Ή ΒΑΘΜΙΔΑ)

Grad[*Συνάρτηση*]

Grad[*Συνάρτηση, ΣύστημαΣυντεταγμένων*[*Συντεταγμένες*]]

Υπολογίζει την κλίση (ή βαθμίδα) ενός βαθμωτού (όχι διανυσματικού) πεδίου στο σύστημα συντεταγμένων που καθορίζεται στην εντολή, αλλιώς στο προκαθορισμένο σύστημα. Παραδείγματα:

In[13]:= **Grad[x^2 y Cos[x] Sin[a z]]**

Out[13]= {2 x y Cos[x] Sin[a z] – x^2 y Sin[x] Sin[a z], x^2 Cos[x] Sin[a z], a x^2 y Cos[x] Cos[a z]}

In[14]:= **Grad[x^2 y Cos[x] Sin[a z], Cartesian[x, y, z]]**

Out[14]= {2 x y Cos[x] Sin[a z] – x^2 y Sin[x] Sin[a z], x^2 Cos[x] Sin[a z], a x^2 y Cos[x] Cos[a z]}

In[15]:= **{Grad[f[x, y, z], Cartesian[x, y, z]], Grad[f[x, y], Cartesian[x, y, z]]}**

Out[15]= {{$f^{(1,0,0)}$[x, y, z], $f^{(0,1,0)}$[x, y, z], $f^{(0,0,1)}$[x, y, z]}, {$f^{(1,0)}$[x, y], $f^{(0,1)}$[x, y], 0}}

```
In[16]:= {Grad[f[r, θ, z], Cylindrical[r, θ, z]], Grad[f[r, θ], Cylindrical[r, θ, z]]}
```

Out[16]= $\left\{\left\{f^{(1,0,0)}[r, \theta, z], \frac{f^{(0,1,0)}[r, \theta, z]}{r}, f^{(0,0,1)}[r, \theta, z]\right\}, \left\{f^{(1,0)}[r, \theta], \frac{f^{(0,1)}[r, \theta]}{r}, 0\right\}\right\}$

```
In[17]:= {Grad[f[ρ, θ, ϕ], Spherical[ρ, θ, ϕ]], Grad[f[ρ], Spherical[ρ, θ, ϕ]]}
```

Out[17]= $\left\{\left\{f^{(1,0,0)}[\rho, \theta, \phi], \frac{f^{(0,1,0)}[\rho, \theta, \phi]}{\rho}, \frac{\mathrm{Csc}[\theta]\, f^{(0,0,1)}[\rho, \theta, \phi]}{\rho}\right\}, \{f'[\rho], 0, 0\}\right\}$

■ ΕΝΤΟΛΗ Δ5: ΑΠΟΚΛΙΣΗ

Div[*ΔιανυσματικήΣυνάρτηση*]

Div[*ΔιανυσματικήΣυνάρτηση, ΣύστημαΣυντεταγμένων*[*Συντεταγμένες*]]

Υπολογίζει την απόκλιση (divergence) ενός διανυσματικού (όχι βαθμωτού) πεδίου στο σύστημα συντεταγμένων που καθορίζεται στην εντολή, αλλιώς στο προκαθορισμένο σύστημα. Παραδείγματα:

```
In[18]:= {Div[{x, y, z}, Cartesian[x, y, z]],
          Div[{x^2 y, Cosh[a z], Sin[y z]}, Cartesian[x, y, z]]}
```

Out[18]= $\{3, 2\,x\,y + y\,\mathrm{Cos}[y\,z]\}$

```
In[19]:= {Div[{u[x, y, z], v[x, y, z], w[x, y, z]}, Cartesian[x, y, z]]}
```

Out[19]= $\{w^{(0,0,1)}[x, y, z] + v^{(0,1,0)}[x, y, z] + u^{(1,0,0)}[x, y, z]\}$

```
In[20]:= {Div[{u[r, θ], v[r, θ], 0}, Cylindrical[r, θ, z]]}
```

Out[20]= $\left\{\frac{u[r, \theta] + v^{(0,1)}[r, \theta] + r\, u^{(1,0)}[r, \theta]}{r}\right\}$

■ ΕΝΤΟΛΗ Δ6: ΠΕΡΙΣΤΡΟΦΗ (Ή ΣΤΡΟΒΙΛΙΣΜΟΣ)

Curl[*ΔιανυσματικήΣυνάρτηση*]

Curl[*ΔιανυσματικήΣυνάρτηση, ΣύστημαΣυντεταγμένων*[*Συντεταγμένες*]]

Υπολογίζει την περιστροφή (ή στροβιλισμό) ενός διανυσματικού (όχι βαθμωτού) πεδίου στο σύστημα συντεταγμένων που καθορίζεται στην εντολή, αλλιώς στο προκαθορισμένο σύστημα. (Σημειώνεται ότι και οι τρεις τελευταίες εντολές **Grad**, **Div** και **Curl** παρουσιάζουν ιδιαίτερο ενδιαφέρον στη Ρευστομηχανική (ή Μηχανική των Ρευστών) και στη Μηχανική του Συνεχούς Μέσου γενικότερα. Παραδείγματα:

```
In[21]:= {Curl[{x, y, z}, Cartesian[x, y, z]],
          Curl[{x^2 y, Cosh[a z], Sin[y z]}, Cartesian[x, y, z]]}
```

Out[21]= $\{\{0, 0, 0\}, \{z\,\mathrm{Cos}[y\,z] - a\,\mathrm{Sinh}[a\,z], 0, -x^2\}\}$

```
In[22]:= {Curl[{u[x, y], v[x, y], 0}, Cartesian[x, y, z]]}
```

Out[22]= $\{\{0, 0, -u^{(0,1)}[x, y] + v^{(1,0)}[x, y]\}\}$

```
In[23]:= {Curl[{u[r, θ], v[r, θ], 0}, Cylindrical[r, θ, z]], Curl[Grad[F[x, y, z]]]}
```

Out[23]= $\left\{\left\{0, 0, \frac{v[r, \theta] - u^{(0,1)}[r, \theta] + r\, v^{(1,0)}[r, \theta]}{r}\right\}, \{0, 0, 0\}\right\}$

■ ΕΝΤΟΛΗ Δ7: ΛΑΠΛΑΣΙΑΝΗ (LAPLACIAN)

Laplacian[*Συνάρτηση*]

Laplacian[*Συνάρτηση, ΣύστημαΣυντεταγμένων*[*Συντεταγμένες*]]

Υπολογίζει τη Λαπλασιανή (τη Laplacian) $\nabla^2 \equiv \Delta$ μιας βαθμωτής συναρτήσεως. Παραδείγματα:

```
In[24]:= CoordinateSystem
Out[24]= Cartesian
```

```
In[25]:= {u[x_, y_] = x^2 y^3, Laplacian[u[x, y]], Laplacian[u[x, y], Cartesian[x, y, z]] }
```

Out[25]= $\{x^2 y^3,\ 6 x^2 y + 2 y^3,\ 6 x^2 y + 2 y^3\}$

```
In[26]:= Laplacian[F[x, y, z], Cartesian[x, y, z]]
```

Out[26]= $F^{(0,0,2)}[x, y, z] + F^{(0,2,0)}[x, y, z] + F^{(2,0,0)}[x, y, z]$

```
In[27]:= Laplacian[F[x, y, z]] == Div[Grad[F[x, y, z]]]
Out[27]= True
```

```
In[28]:= Laplacian[F[r, θ, z], Cylindrical[r, θ, z]] // Simplify
```

Out[28]= $F^{(0,0,2)}[r, \theta, z] + \frac{F^{(0,2,0)}[r, \theta, z]}{r^2} + \frac{F^{(1,0,0)}[r, \theta, z]}{r} + F^{(2,0,0)}[r, \theta, z]$

```
In[29]:= Laplacian[F[r, θ], Cylindrical[r, θ, z]] // Simplify
```

Out[29]= $\frac{F^{(0,2)}[r, \theta]}{r^2} + \frac{F^{(1,0)}[r, \theta]}{r} + F^{(2,0)}[r, \theta]$

```
In[30]:= Laplacian[F[ρ, θ, ϕ], Spherical[ρ, θ, ϕ]]
```

Out[30]= $\frac{1}{\rho^2}(\mathrm{Csc}[\theta]\ (\mathrm{Csc}[\theta]\ F^{(0,0,2)}[\rho, \theta, \phi] + \mathrm{Cos}[\theta]\ F^{(0,1,0)}[\rho, \theta, \phi] +$
$\mathrm{Sin}[\theta]\ F^{(0,2,0)}[\rho, \theta, \phi] + 2\rho\ \mathrm{Sin}[\theta]\ F^{(1,0,0)}[\rho, \theta, \phi] + \rho^2\ \mathrm{Sin}[\theta]\ F^{(2,0,0)}[\rho, \theta, \phi]))$

```
In[31]:= Laplacian[F[ρ], Spherical[ρ, θ, ϕ]] // Simplify
```

Out[31]= $\frac{2 F'[\rho]}{\rho} + F''[\rho]$

■ ΕΝΤΟΛΗ Δ8: ΔΙΑΡΜΟΝΙΚΟΣ ΤΕΛΕΣΤΗΣ

Biharmonic[*Συνάρτηση*]

Biharmonic[*Συνάρτηση, ΣύστημαΣυντεταγμένων*[*Συντεταγμένες*]]

Εφαρμόζει το διαρμονικό τελεστή $\nabla^4 \equiv \Delta^2$ (που ισοδυναμεί με διπλή εφαρμογή της Λαπλασιανής) σε μια βαθμωτή συνάρτηση. (Σημειώνεται ότι ο διαρμονικός τελεστής παρουσιάζεται σε προβλήματα Επίπεδης Ελαστικότητας: τασική συνάρτηση του Airy, και επίσης σε προβλήματα πλακών.) Παραδείγματα:

```
In[32]:= {CoordinateSystem, Laplacian[F[x, y]], Biharmonic[F[x, y]]}
```

Out[32]= $\{\mathrm{Cartesian},\ F^{(0,2)}[x, y] + F^{(2,0)}[x, y],\ F^{(0,4)}[x, y] + 2 F^{(2,2)}[x, y] + F^{(4,0)}[x, y]\}$

```
In[33]:= Biharmonic[F[x, y]] == Laplacian[Laplacian[F[x, y]]]

Out[33]= True

In[34]:= Biharmonic[F[r], Cylindrical[r, θ, z]] // Simplify
```

Out[34]= $\frac{F'[r] - r F''[r] + 2 r^2 F^{(3)}[r] + r^3 F^{(4)}[r]}{r^3}$

```
In[35]:= Biharmonic[F[ρ], Spherical[ρ, θ, ϕ]] // Simplify
```

Out[35]= $\frac{4 F^{(3)}[\rho]}{\rho} + F^{(4)}[\rho]$

■ ΕΝΤΟΛΗ Δ9: ΣΧΕΔΙΑΣΗ ΔΙΑΝΥΣΜΑΤΙΚΟΥ ΠΕΔΙΟΥ

PlotVectorField[{*x*-ΣυνιστώσαΤουΔιανυσματικούΠεδίου, *y*-ΣυνιστώσαΤουΔιανυσματικούΠεδίου},
{*Μεταβλητή-x, ΑρχικήΤιμή-x, ΤελικήΤιμή-x*},
{*Μεταβλητή-y, ΑρχικήΤιμή-y, ΤελικήΤιμή-y*}, *Επιλογή-1, Επιλογή-2*, ...]

Σχεδιάζει στο επίπεδο το διδιάστατο διανυσματικό πεδίο το οποίο καθορίζεται στη λίστα του πρώτου ορίσματος με δύο βέβαια συνιστώσες εδώ στο επίπεδο. Η σχεδίαση αυτή γίνεται στην ορθογωνική περιοχή του επιπέδου που καθορίζεται στις δύο επόμενες λίστες για την πρώτη (συνήθως *x*) και τη δεύτερη (συνήθως *y*) μεταβλητή αντίστοιχα. Ακολουθούν οι επιλογές, εάν χρειάζονται. Σημειώνεται επίσης ότι η παρούσα εντολή **PlotVectorField** θα χρησιμοποιηθεί και στις διαφορικές εξισώσεις πρώτης τάξεως της μορφής $y' = f(x, y)$ στο Notebook E15 για τις διαφορικές εξισώσεις: Εντολή D3. Θα πρέπει επίσης να σημειωθεί στο σημείο αυτό ότι η παρούσα εντολή **PlotVectorField** είναι μια εντολή του πακέτου **Graphics'PlotField'**. Αυτό θα πρέπει να έχει ήδη κληθεί (φορτωθεί) πριν από οποιαδήποτε χρήση της παρούσας εντολής **PlotVectorField** με τον πιο κάτω τρόπο με την εντολή **Needs** ή με τον ισοδύναμο τρόπο με τη χρήση του συμβόλου << . (Παραδείγματα της εντολής αυτής **PlotVectorField** για την εφαρμογή της στη διδιάστατη μόνιμη αστρόβιλη ροή ιδεατού ρευστού στη Ρευστομηχανική (ή Μηχανική των Ρευστών) θα δοθούν στη μεθεπόμενη εντολή **PlotHamiltonianField**.)

```
In[36]:= Needs["Graphics`PlotField`"];
```

Υπάρχουν πάρα πολλές (37) επιλογές διαθέσιμες στην εντολή αυτή **PlotVectorField**. Αυτές είναι οι εξής:

```
In[37]:= {opt = Options[PlotVectorField], NumberOfPlotVectorFieldOptions = Length[opt]}

Out[37]= {{ScaleFactor → Automatic, ScaleFunction → None, MaxArrowLength → None,
  ColorFunction → None, AspectRatio → Automatic, HeadScaling → Automatic,
  HeadLength → 0.02, HeadCenter → 1, HeadWidth → 0.5, HeadShape → Automatic,
  ZeroShape → Automatic, AspectRatio → 1/GoldenRatio, Axes → False, AxesLabel → None,
  AxesOrigin → Automatic, AxesStyle → Automatic, Background → Automatic,
  ColorOutput → Automatic, DefaultColor → Automatic, Epilog → {}, Frame → False,
  FrameLabel → None, FrameStyle → Automatic, FrameTicks → Automatic,
  GridLines → None, ImageSize → Automatic, PlotLabel → None, PlotRange → All,
  PlotRegion → Automatic, Prolog → {}, RotateLabel → True, Ticks → Automatic,
  DefaultFont :→ $DefaultFont, DisplayFunction :→ $DisplayFunction,
  FormatType :→ $FormatType, TextStyle :→ $TextStyle, PlotPoints → Automatic}, 37}
```

■ ΕΝΤΟΛΗ Δ10: ΣΧΕΔΙΑΣΗ ΠΕΔΙΟΥ ΚΛΙΣΕΩΣ (Ή ΒΑΘΜΙΔΑΣ)

PlotGradientField[*ΒαθμωτόΔυναμικόΤουΔιανυσματικούΠεδίου,*

{Μεταβλητή-x, ΑρχικήΤιμή-x, ΤελικήΤιμή-x},

***{Μεταβλητή-y, ΑρχικήΤιμή-y, ΤελικήΤιμή-y}, Επιλογή-1, Επιλογή-2, ...*]**

Η εντολή αυτή **PlotGradientField** είναι παραπλήσια, αλλά λίγο λιγότερο γενική από την προηγούμενη εντολή **PlotVectorField**. Συγκεκριμένα η διαφορά εδώ είναι ότι το διανυσματικό πεδίο που σχεδιάζεται είναι η κλίση (ή βαθμίδα, το grad: εντολή Δ4 πιο πάνω) του βαθμωτού δυναμικού φ στο πρώτο όρισμά της. Δηλαδή σχεδιάζεται το διανυσματικό πεδίο $\mathbf{V}_1 = \text{grad}\,\varphi \equiv \nabla\varphi := (\partial\varphi/\partial x, \partial\varphi/\partial y)$. (Η σχεδίαση γίνεται ακριβώς όπως και στην προηγούμενη εντολή.) Πρέπει να σημειωθεί στο σημείο αυτό ότι η παρούσα εντολή **PlotGradientField** είναι κι αυτή μια εντολή του πακέτου **Graphics‘PlotField‘** που θα πρέπει να έχει ήδη κληθεί ακριβώς όπως και πριν. Παραδείγματα και αυτής της εντολής θα δοθούν στην επόμενη εντολή **PlotHamiltonianField**. Οι επιλογές της (options) συμπίπτουν με εκείνες της προηγούμενης εντολής **PlotVectorField**, αλλά και με εκείνες της επόμενης εντολής **PlotHamiltonianField**:

```
In[38]:= Options[PlotVectorField] ==
          Options[PlotGradientField] == Options[PlotHamiltonianField]

Out[38]= True
```

■ ΕΝΤΟΛΗ Δ11: ΣΧΕΔΙΑΣΗ ΠΕΔΙΟΥ ΧΑΜΙΛΤΟΝΙΑΝΗΣ

PlotHamiltonianField[*ΒαθμωτήΧαμιλτονιανήΤουΔιανυσματικούΠεδίου,*

{Μεταβλητή-x, ΑρχικήΤιμή-x, ΤελικήΤιμή-x},

***{Μεταβλητή-y, ΑρχικήΤιμή-y, ΤελικήΤιμή-y}, Επιλογή-1, Επιλογή-2, ...*]**

Η εντολή αυτή **PlotHamiltonianField** είναι παραπλήσια, αλλά λίγο λιγότερο γενική από την προπροηγούμενη εντολή **PlotVectorField** κι εντελώς ανάλογη με την προηγούμενη εντολή **PlotGradientField**. Η μόνη διαφορά είναι ότι εδώ σχεδιάζεται (στο επίπεδο πάλι) το διανυσματικό πεδίο της Hamiltonian (Χαμιλτονιανής) της βαθμωτής συναρτήσεως στο πρώτο όρισμα της εντολής αυτής. Δηλαδή σχεδιάζεται το διανυσματικό πεδίο $\mathbf{V}_2 = (\partial\varphi/\partial y, -\partial\varphi/\partial x)$. (Η σχεδίαση γίνεται ακριβώς όπως και στις δύο προηγούμενες εντολές και με τις ίδιες επιλογές.) Πρέπει επίσης να σημειωθεί ξανά στο σημείο αυτό ότι η παρούσα εντολή **PlotGradientField** είναι κι αυτή μια εντολή του πακέτου **Graphics‘PlotField‘** που θα πρέπει να έχει ήδη κληθεί (φορτωθεί) ακριβώς όπως και πριν. Ακολουθούν η κλήση του πακέτου **Graphics‘PlotField‘** (με τον ένα ή τον άλλο από τους δύο παρακάτω τρόπους, όχι και με τους δύο μαζί!) και παραδείγματα και των τριών εντολών **PlotVectorField**, **PlotGradientField** και **PlotHamiltonianField**:

```
In[39]:= Needs["Graphics`PlotField`"]

In[40]:= << Graphics`PlotField`
```

Για τα παραδείγματα των πιο πάνω τριών εντολών θα χρησιμοποιήσουμε εδώ τη διδιάστατη (επίπεδη) μόνιμη (σταθερή) αστρόβιλη ροή ιδεατού (ασυμπίεστου, δηλαδή με σταθερή πυκνότητα ρ, και χωρίς συνεκτικότητα, ιξώδες, δηλαδή με $\mu = 0$) ρευστού στη Ρευστομηχανική (ή Μηχανική των Ρευστών). Η ροή αυτή παρουσιάζει βέβαια ενδιαφέρον για τον Πολιτικό Μηχανικό. Σαν πρώτο παράδειγμα θεωρούμε την ίδια ροή, όταν οι δύο συνιστώσες *u* και *v* της ταχύτητας **V** του ρευστού δίνονται από τους απλούς τύπους

```
In[41]:= {u[x_, y_] = Sinh[x] Cos[y], v[x_, y_] = -Cosh[x] Sin[y]}

Out[41]= {Cos[y] Sinh[x], -Cosh[x] Sin[y]}
```

Επομένως η διανυσματική ταχύτητα **V** = **V**(*x*, *y*) του ρευστού στη ροή δίνεται από τον τύπο

```
In[42]:= V[x_, y_] = {u[x, y], v[x, y], 0}

Out[42]= {Cos[y] Sinh[x], -Cosh[x] Sin[y], 0}
```

Είναι απόλυτα αποδεκτή η ροή αυτή, επειδή για την ταχύτητα **V** του ρευστού στη ροή αυτή ισχύουν οι εξισώσεις (α) της συνεχείας (γραμμένη εδώ τόσο σε βαθμωτή όσο και σε διανυσματική μορφή):

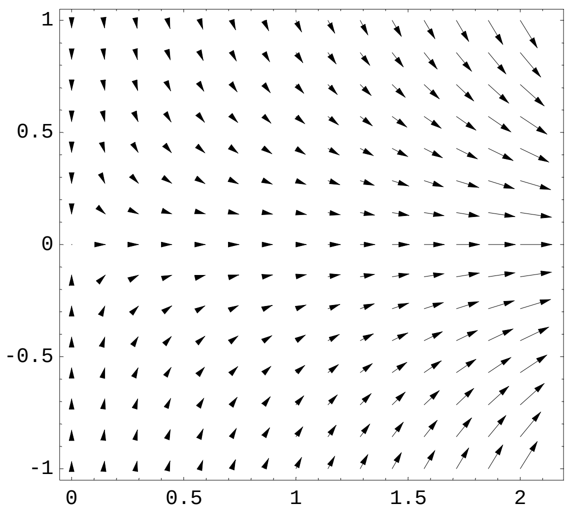
```
In[43]:= {D[u[x, y], x] + D[v[x, y], y] == 0, Div[V[x, y]] == 0}

Out[43]= {True, True}
```

και (β) του αστρόβιλου της ροής (επίσης γραμμένη εδώ και σε βαθμωτή και σε διανυσματική μορφή):

```
In[44]:= {D[v[x, y], x] - D[u[x, y], y] == 0, Curl[V[x, y]] == {0, 0, 0}}

Out[44]= {True, True}
```

Είναι λοιπόν απόλυτα αποδεκτές οι δύο συνιστώσες *u*(*x*, *y*) και *v*(*x*, *y*) της ταχύτητας **V**, που ορίσθηκαν παραπάνω σαν συναρτήσεις. Σχεδιάζουμε τώρα με την εντολή **PlotVectorField** το πεδίο ταχύτητας (velocity field) στην ορθογωνική περιοχή $0 \le x \le 2$ και $-1 \le y \le 1$ και μάλιστα μέσα σε πλαίσιο (frame):

```
In[45]:= PlotVectorField[{u[x, y], v[x, y]},
           {x, 0, 2}, {y, -1, 1}, Frame → True, ImageSize → 275];
```



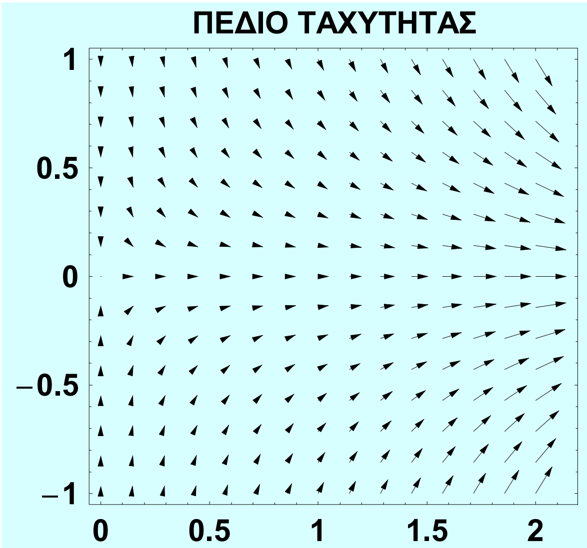
Τί μας δείχνει το παραπάνω σχήμα; Μα μας δείχνει με βελάκια τη διανυσματική ταχύτητα **V** του ρευστού σε πολλά σημεία της ορθογωνικής περιοχής που επιλέξαμε. Η διεύθυνση του κάθε βέλους εκφράζει τη διεύθυνση της ταχύτητας και το μήκος του το μέτρο της ταχύτητας. Έχουμε λοιπόν αποκτήσει μια σαφή εικόνα για την κίνηση του ρευστού μη ξεχνώντας βέβαια ότι την παρούσα ροή την έχουμε υποθέσει μόνιμη (σταθερή, χωρίς μεταβολή με το χρόνο). Ωραία λοιπόν ως εδώ με το παρόν παράδειγμα!

Παρενθετικά ας σημειώσουμε ότι ορίζοντας τη διδιάστατη ταχύτητα $\mathbf{V}_2$ του ρευστού σαν

```
In[46]:= V2[x_, y_] = {u[x, y], v[x, y]}
Out[46]= {Cos[y] Sinh[x], -Cosh[x] Sin[y]}
```

μπορούμε ασφαλώς να πάρουμε ακριβώς το ίδιο πιο πάνω σχήμα με την ουσιαστικά ισοδύναμη εντολή

```
In[47]:= PlotVectorField[V2[x, y], {x, 0, 2}, {y, -1, 1}, Background → RGBColor[0.85, 1, 1],
          PlotLabel -> "ΠΕΔΙΟ ΤΑΧΥΤΗΤΑΣ", DefaultFont → {"Arial-Bold", 15}, Frame → True];
```



Ας προχωρήσουμε τώρα λιγάκι ακόμη. Στη διδιάστατη μόνιμη αστρόβιλη ροή ιδεατού ρευστού, την οποία υποθέσαμε (και μια ροή με αυτές τις ιδιότητες είναι μια συνηθισμένη, κλασική ροή στη Ρευστομηχανική), οι δύο συνιστώσες της ταχύτητας **V** του ρευστού μπορούν να προκύψουν σαν η κλίση (η βαθμίδα, το grad) ενός κατάλληλου βαθμωτού δυναμικού ροής: του δυναμικού ταχύτητας $\Phi(x, y)$. Αυτό το γνωρίζουμε από τις Παραγράφους B3.4.2 και B3.4.3 του Μέρους B των διδακτικών βιβλίων. Συγκεκριμένα στη ροή αυτή μπορεί να διαπιστωθεί ότι το δυναμικό ταχύτητας $\Phi(x, y)$ έχει τη μορφή

```
In[48]:= Φ[x_, y_] = Cosh[x] Cos[y]
Out[48]= Cos[y] Cosh[x]
```

οπότε οι συνιστώσες u και v της ταχύτητας **V** του ρευστού θα είναι πραγματικά οι

```
In[49]:= {u[x_, y_] = D[Φ[x, y], x], v[x_, y_] = D[Φ[x, y], y]}
Out[49]= {Cos[y] Sinh[x], -Cosh[x] Sin[y]}
```

(Όπως παρατηρούμε είναι οι ίδιες οι πιο πάνω συνιστώσες της ταχύτητας.) Και διανυσματικά βέβαια προκύπτουν οι ίδιες ακριβώς συνιστώσες με εφαρμογή της εντολής **Grad** στο δυναμικό ταχύτητας $\Phi(x, y)$:

```
In[50]:= V[x_, y_] = Grad[Φ[x, y]]
Out[50]= {Cos[y] Sinh[x], -Cosh[x] Sin[y], 0}
```

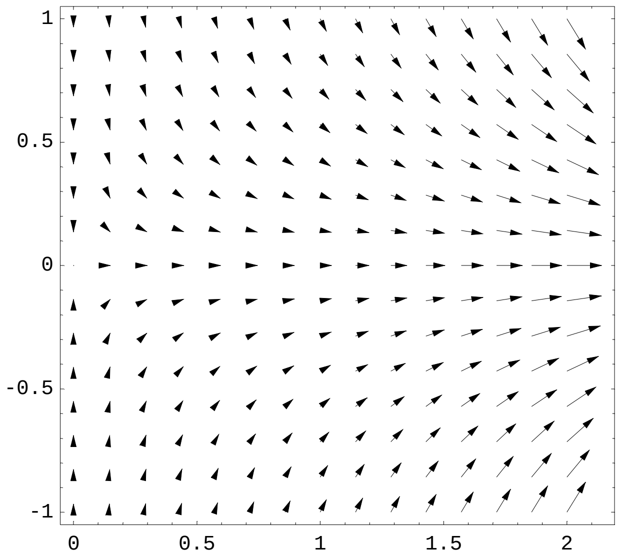
```
In[51]:= V[x, y] == {u[x, y], v[x, y], 0}
Out[51]= True
```

Γνωρίζουμε βέβαια πως το δυναμικό ταχύτητας, η συνάρτηση $\Phi(x, y)$, οφείλει να επαληθεύει τη διδιάστατη εξίσωση του Laplace. Δηλαδή πρέπει να είναι αρμονική συνάρτηση. Αυτό πράγματι συμβαίνει εδώ:

```
In[52]:= Laplacian[Φ[x, y]] == 0

Out[52]= True
```

Κατά συνέπεια, γνωρίζοντας σε μια διδιάστατη μόνιμη αστρόβιλη ροή το δυναμικό ταχύτητας $\Phi(x, y)$, μπορούμε θαυμάσια να χρησιμοποιήσουμε την προηγούμενη εντολή **PlotGradientField** για τη σχεδίαση του ίδιου ακριβώς διανυσματικού πεδίου: του πεδίου ταχύτητας **V** του ιδεατού ρευστού. Συγκεκριμένα

```
In[53]:= PlotGradientField[Φ[x, y], {x, 0, 2}, {y, -1, 1}, Frame → True, ImageSize → 300];
```



Υπενθυμίζεται ότι στην παρούσα ροή η ίδια διανυσματική ταχύτητα $\mathbf{V} = \mathbf{V}(x, y)$ μπορεί να προκύψει και από τη ροϊκή συνάρτηση (ή συνάρτηση ροής) $\Psi(x, y)$. Αυτό το γνωρίζουμε από την Παράγραφο B3.4.4 του Μέρους B των διδακτικών βιβλίων. Συγκεκριμένα εδώ η ροϊκή συνάρτηση $\Psi(x, y)$ έχει τη μορφή

```
In[54]:= Ψ[x_, y_] = Sinh[x] Sin[y]

Out[54]= Sin[y] Sinh[x]
```

Αυτή δίνει ξανά τις ίδιες ακριβώς συνιστώσες της ταχύτητας $\mathbf{V} = \mathbf{V}(x, y)$ του ιδεατού ρευστού

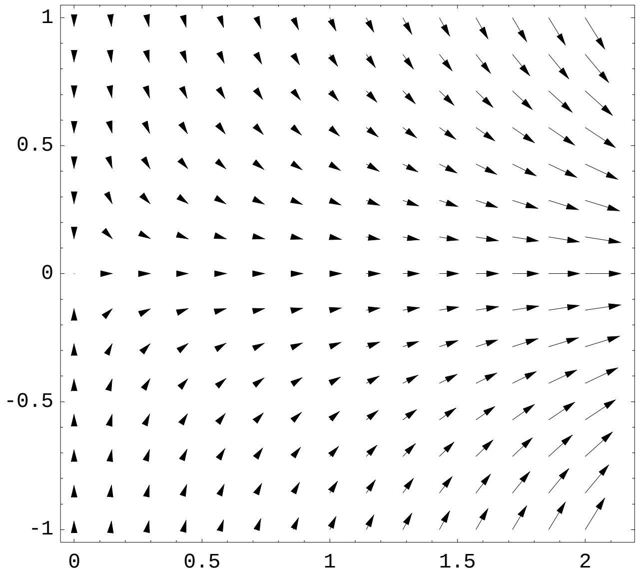
```
In[55]:= {u[x_, y_] = D[Ψ[x, y], y], v[x_, y_] = -D[Ψ[x, y], x]}

Out[55]= {Cos[y] Sinh[x], -Cosh[x] Sin[y]}
```

Φυσικά, όπως το δυναμικό ταχύτητας $\Phi(x, y)$, έτσι και η ροϊκή συνάρτηση $\Psi(x, y)$ πρέπει να επαληθεύει τη διδιάστατη εξίσωση του Laplace, να είναι αρμονική συνάρτηση. Αυτό συμβαίνει στ' αλήθεια κι εδώ:

```
In[56]:= Laplacian[Ψ[x, y]] == 0

Out[56]= True
```

Κατά συνέπεια, γνωρίζοντας σε μια διδιάστατη μόνιμη αστρόβιλη ροή τη ροϊκή συνάρτηση (ή συνάρτηση ροής) $\Psi(x, y)$, μπορούμε να χρησιμοποιήσουμε την παρούσα εντολή **PlotHamiltonianField** για τη σχεδίαση του ίδιου ακριβώς διανυσματικού πεδίου: για την ταχύτητα του ιδεατού ρευστού. Συγκεκριμένα εδώ

```
In[57]:= PlotHamiltonianField[Ψ[x, y], {x, 0, 2}, {y, -1, 1}, Frame → True, ImageSize → 310];
```



Πήραμε λοιπόν και πάλι το ίδιο ακριβώς διανυσματικό πεδίο (κι ήταν αναμενόμενο αυτό!): το πεδίο για τη διανυσματική ταχύτητα $\mathbf{V} = \mathbf{V}(x, y)$ του ρευστού και στην παρούσα διδιάστατη, μόνιμη και αστρόβιλη ροή.

Συμπεραίνουμε ότι στο είδος της ροής που μόλις αναφέρθηκε μπορούμε να εργαζόμαστε είτε (α) με τις ίδιες τις δύο συνιστώσες της ταχύτητας **V** του ρευστού (σε λίστα με δύο στοιχεία) ή με το αντίστοιχο διδιάστατο διάνυσμα $\mathbf{V}_2(x, y)$ και την προπροηγούμενη εντολή **PlotVectorField** είτε (β) με το δυναμικό ταχύτητας $\Phi(x, y)$ και την προηγούμενη εντολή **PlotGradientField** είτε τέλος (γ) με τη ροϊκή συνάρτηση (ή συνάρτηση ροής) $\Psi(x, y)$ και την παρούσα εντολή **PlotHamiltonianField** ανάλογα βέβαια με το τί ξέρουμε: τις συνιστώσες της ταχύτητας (ή τη διανυσματική ταχύτητα **V**), (β) το δυναμικό ταχύτητας $\Phi(x, y)$ ή (γ) τη ροϊκή συνάρτηση (ή συνάρτηση ροής) $\Psi(x, y)$ και τί μας διευκολύνει ασφαλώς κάθε φορά.

■ ΡΟΗ ΓΥΡΩ ΑΠΟ (ΣΤΑΘΕΡΟ ΣΤΕΡΕΟ ΚΥΚΛΙΚΟ) ΚΥΛΙΝΔΡΟ

Ας δώσουμε και ένα δεύτερο και σίγουρα πολύ πιο γνωστό παράδειγμα διδιάστατης (επίπεδης) μόνιμης (σταθερής) αστρόβιλης ροής ιδεατού ρευστού. Πρόκειται για τη ροή γύρω από έναν κυκλικό κύλινδρο με κέντρο την αρχή των αξόνων (0, 0) και ακτίνα a (όμως με μηδενική τη συνολική κυκλοφορία Γ γύρω από τον κύλινδρο). Πιο συγκεκριμένα εδώ εξετάζουμε μια ομοιόμορφη ροή με ταχύτητα U παράλληλη προς τον άξονα Ox που παρεμποδίζεται από τον κύλινδρο που ήδη αναφέραμε και επομένως τροποποιείται, αλλάζει πολύ στην περιοχή του κυλίνδρου. Στη ροή αυτή το δυναμικό ταχύτητας $\Phi(x, y)$ έχει τη μορφή

In[58]:= $\Phi[x\_, y\_] = \mathrm{If}[x^2 + y^2 \geq a^2, U\, x\, (1 + a^2 / (x^2 + y^2)), 0]$

Out[58]= $\mathrm{If}\left[x^2 + y^2 \geq a^2,\ U\, x \left(1 + \frac{a^2}{x^2 + y^2}\right),\ 0\right]$

(Δηλαδή μέσα στον κύκλο, τον κύλινδρο δεν υπάρχει δυναμικό ταχύτητας $\Phi(x, y)$, είναι προφανές αυτό!) Επίσης η ροϊκή συνάρτηση (ή συνάρτηση ροής) $\Psi(x, y)$ έχει την αντίστοιχη και απόλυτα ανάλογη μορφή

In[59]:= $\Psi[x\_, y\_] = \mathrm{If}[x^2 + y^2 \geq a^2, U\, y\, (1 - a^2 / (x^2 + y^2)), 0]$

Out[59]= $\mathrm{If}\left[x^2 + y^2 \geq a^2,\ U\, y \left(1 - \frac{a^2}{x^2 + y^2}\right),\ 0\right]$

Φυσικά και οι δυο τους είναι αρμονικές συναρτήσεις. (Καλή θα είναι κι αυτή η επαλήθευση. Χρειάζεται!)

```
In[60]:= {{Φ[x, y][[2]], Laplacian[Φ[x, y][[2]]]}, {Ψ[x, y][[2]], Laplacian[Ψ[x, y][[2]]]}} // Simplify
```

Out[60]= $\left\{\left\{U\,x\left(1+\frac{a^2}{x^2+y^2}\right),\ 0\right\},\ \left\{U\,y\left(1-\frac{a^2}{x^2+y^2}\right),\ 0\right\}\right\}$

Είτε η μία είτε η άλλη από τις δύο αυτές συναρτήσεις μας είναι απόλυτα επαρκής για τη σχεδίαση του παρόντος πεδίου ροής, αφού βέβαια καθορισθούν αριθμητικά οι δύο σταθερές *a* (ακτίνα του κυλίνδρου) και *U* (ταχύτητα του ιδεατού ρευστού στο άπειρο). Εδώ τις παίρνουμε για διευκόλυνση ίσες με τη μονάδα:

```
In[61]:= {a = 1, U = 1};
```

Θα θεωρήσουμε πρώτα το δυναμικό ταχύτητας *Φ*(*x*, *y*). Προηγουμένως όμως ας σχεδιάσουμε τον ίδιο τον κύλινδρο **Cylinder** (εδώ σε τομή βέβαια). Αυτό μπορεί να γίνει αρκετά εύκολα με την εντολή

```
In[62]:= Cylinder = Show[Graphics[Disk[{0, 0}, a]],
    PlotRange → {{-5, 5}, {-1.35, 1.35}}, AspectRatio → 1.35 / 5,
    PlotLabel → "ΚΥΛΙΝΔΡΟΣ ΣΕ ΤΟΜΗ", DefaultFont → {"Courier-Bold", 14},
    Background → RGBColor[1., 1, 0.6], ImageSize → 280];
```

Σχεδιάζουμε τώρα και το διανυσματικό πεδίο της ταχύτητας **V** του ρευστού με τη χρήση της εντολής **PlotGradientField**, αφού αποφασίσαμε εδώ να χρησιμοποιήσουμε πρώτα το δυναμικό ταχύτητας *Φ*(*x*, *y*):

```
In[63]:= VelocityField1 = PlotGradientField[Φ[x, y], {x, -4, 4},
    {y, -3, 3}, ScaleFactor → 0.55, Background → RGBColor[0.85, 1, 1],
    PlotLabel → "ΠΕΔΙΟ ΤΑΧΥΤΗΤΑΣ ΓΥΡΩ ΑΠΟ ΚΥΛΙΝΔΡΟ",
    DefaultFont → {"Arial-Bold", 14}, ImageSize → 371];
```

Καλύτερα είναι όμως να βελτιώσουμε την εμφάνιση του ίδιου πεδίου ταχύτητας εισάγοντας στο σχήμα και τον ίδιο τον κύλινδρο **Cylinder** που τον έχουμε κιόλας έτοιμο. Αυτός θα κάνει το σχήμα πολύ πιο σαφές στην παρούσα ροή γύρω από κύλινδρο. Ας χρησιμοποιήσουμε επίσης την επιλογή **Frame → True**:

```
In[64]:= VelocityField2 = Show[Cylinder, VelocityField1,
    Background → RGBColor[0.85, 1, 1], PlotRange → {{-4.1, 4.4}, {-3.2, 3.2}},
    AspectRatio → 3.2 / 4.25, PlotLabel -> "ΠΕΔΙΟ ΤΑΧΥΤΗΤΑΣ ΓΥΡΩ ΑΠΟ ΚΥΛΙΝΔΡΟ",
    DefaultFont → {"Arial-Bold", 14}, Frame → True, ImageSize → 371];
```

Πιο ουσιαστικό είναι το να κάνουμε ορισμένες παρατηρήσεις. Ναι, το διάνυσμα της ταχύτητας **V** του ρευστού προσπερνάει τον κύλινδρο (που αποτελεί το εμπόδιο στη ροή). Το ρευστό ρέει από πάνω και από κάτω από τον κύλινδρο. Όμως για να τον παρακάμψει αναγκάζεται να αυξήσει σημαντικά την ταχύτητά του **V**. Η ταχύτητα στα σημεία (0, 1) και (0, −1) (το υψηλότερο και το χαμηλότερο σημείο του κυλίνδρου αντίστοιχα) είναι κατά μέτρο διπλάσια από την ταχύτητα *U* του ρευστού στο άπειρο: μακριά από τον κύλινδρο. Στο σχήμα φαίνεται περίπου διπλάσια. Το ότι είναι ακριβώς διπλάσια προκύπτει με υπολογισμό:

```
In[65]:= Φ[x, y]⟦2⟧
```

Out[65]= $x\left(1+\frac{1}{x^2+y^2}\right)$

```
In[66]:= {V[x_, y_] = Grad[Φ[x, y]⟦2⟧] // FullSimplify, Limit[V[x, y], y → ∞]}
```

Out[66]= $\left\{\left\{1+\frac{-x^2+y^2}{(x^2+y^2)^2},\ -\frac{2\,x\,y}{(x^2+y^2)^2},\ 0\right\},\ \{1,0,0\}\right\}$

Επομένως πραγματικά προκύπτει διπλάσια ταχύτητα από το άπειρο, εκεί με **V** = (*U*, 0, 0) = (1, 0, 0):

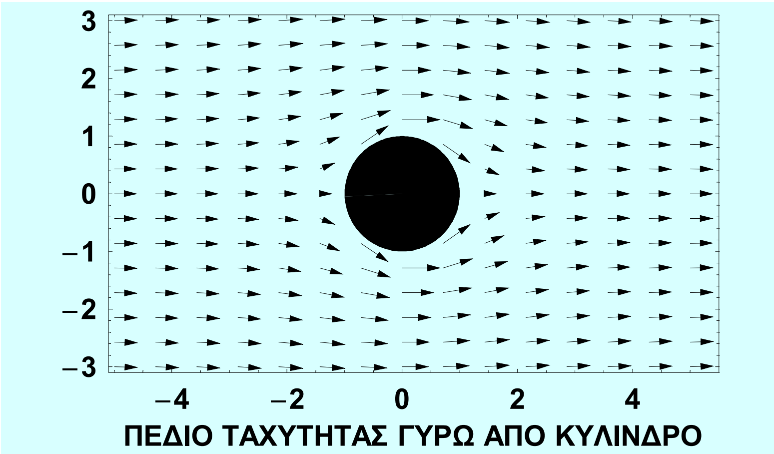
```
In[67]:= {V[0, 1], V[0, -1]}

Out[67]= {{2, 0, 0}, {2, 0, 0}}
```

Παρατηρούμε επίσης από το πιο πάνω σχήμα ότι η ταχύτητα **V** του ρευστού στην περιφέρεια που αποτελεί το σύνορο του κυκλικού μας κυλίνδρου εφάπτεται στην περιφέρεια αυτή, δεν έχει κάθετη συνιστώσα σ' αυτήν. Είναι εύλογο αυτό, αφού ο κύλινδρος είναι στερεός. Δε μπορεί το ρευστό να μπει στον κύλινδρο. Πρέπει να τον παρακάμψει και αυτό πράγματι κάνει, όπως φαίνεται στο προηγούμενο σχήμα. Ειδικά στα σημεία (−1, 0) και (1, 0) η ταχύτητα **V** μηδενίζεται: **V** = **0**: σημεία (γραμμές) ανακοπής της ροής.

Μια που έτυχε να γνωρίζουμε εδώ και τη ροϊκή συνάρτηση (ή συνάρτηση ροής) *Ψ*(*x*, *y*), μπορούμε τώρα να σχεδιάσουμε το ίδιο ακριβώς πεδίο ταχύτητας **V** και με τη χρήση της εντολής **PlotHamiltonianField**. Ας κάνουμε κι αυτήν τη σχεδίαση αλλάζοντας μάλιστα και λίγο τις επιλογές μας:

```
In[68]:= VelocityField3 = PlotHamiltonianField[Ψ[x, y], {x, -5, 5},
           {y, -3, 3}, ScaleFactor → 0.65, DisplayFunction → Identity];
```

```
In[69]:= VelocityField4 = Show[Cylinder, VelocityField3, AspectRatio → 3.1 / 5.3,
           PlotRange → {{-5.1, 5.5}, {-3.1, 3.1}}, Frame → True, PlotLabel → None,
           FrameLabel → {"ΠΕΔΙΟ ΤΑΧΥΤΗΤΑΣ ΓΥΡΩ ΑΠΟ ΚΥΛΙΝΔΡΟ", "", "", ""}, DefaultFont →
            {"Arial-Bold", 14}, Background → RGBColor[0.85, 1, 1], ImageSize → 380];
```



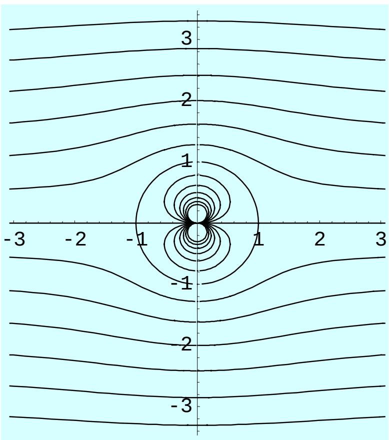
Φυσικά έχουμε πάντα και τη δυνατότητα για τη σχεδίαση αυτού του πεδίου ταχύτητας **V** να χρησιμοποιήσουμε την εντολή **PlotVectorField** σε συνδυασμό ασφαλώς με την ταχύτητα **V** του ρευστού σε κάθε σημείο της ροής, όχι όμως και μέσα στον κύλινδρο, γιατί εκεί δεν υπάρχει ροή. Και επιπλέον στο κέντρο (0, 0) του κυλίνδρου η ανύπαρκτη αυτή ταχύτητα θα μας δώσει λάθος λόγω απειρισμού της. Άρα χρειάζεται η εντολή **If** και στον ορισμό της ταχύτητας **V**. Δε θα προχωρήσουμε όμως στη σχετική σχεδίαση.

Πριν το τέλος κι αυτής της εφαρμογής στη Ρευστομηχανική, ας κάνουμε κάτι πιο ενδιαφέρον. Ας φορτώσουμε το πακέτο **Graphics'ImplicitPlot'** για τη σχεδίαση συναρτήσεων που είναι σε πεπλεγμένη μορφή:

```
In[70]:= Needs["Graphics`ImplicitPlot`"]
```

Τώρα μπορούμε να σχεδιάσουμε εύκολα μερικές γραμμές ροής (streamlines) για την ομοιόμορφη, παράλληλη ροή που παρεμποδίζεται από τον κύλινδρο. Στις γραμμές αυτές η ροϊκή συνάρτηση $\Psi(x, y)$ είναι βέβαια σταθερή. Μερικές τέτοιες γραμμές ροής **StreamLines** δημιουργούμε πολύ εύκολα με την εντολή:

```
In[71]:= StreamLines = ImplicitPlot[Table[Ψ[x, y]⟦2⟧ == c, {c, -3, 3, 0.5}], {x, -5, 5},
           PlotStyle → Thickness[0.004], Background → RGBColor[0.85, 1, 1], ImageSize → 190];
```

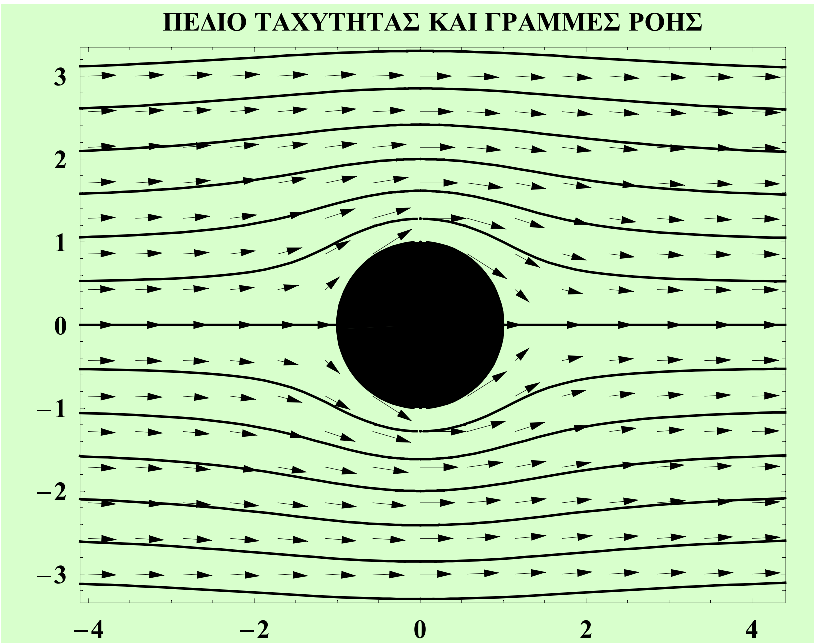


Δεν ισχύουν όμως στο εσωτερικό του κυλίνδρου. (Εκεί δεν υπάρχει καθόλου ροή. Άρα ούτε και γραμμές ροής!) Επομένως τελικά είναι πολύ προτιμότερο να δείξουμε μαύρο τον κύλινδρο με τη σύνθετη εντολή

```
In[72]:= Show[Cylinder, StreamLines, PlotRange → {{-5, 5}, {-3, 3}}, AspectRatio → 3 / 5,
           PlotLabel -> "ΡΟΗ ΠΟΥ ΠΑΡΕΜΠΟΔΙΖΕΤΑΙ ΑΠΟ ΣΤΑΘΕΡΟ ΚΥΛΙΝΔΡΟ",
           Background → RGBColor[0.85, 1, 0.8], ImageSize → 400];
```

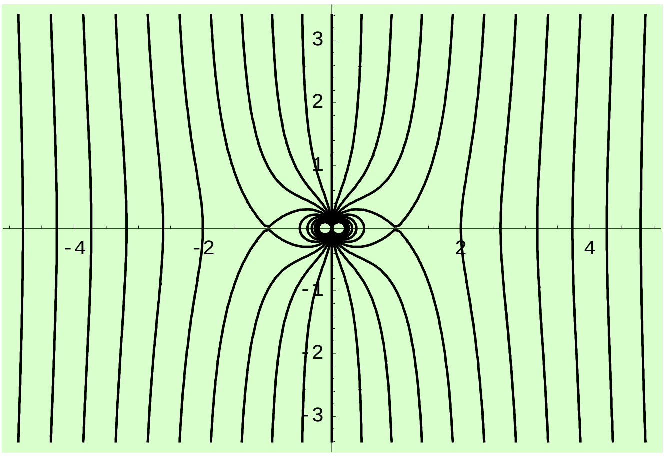
Να και ο συνδυασμός του πεδίου ταχύτητας **V** της ροής και των γραμμών ροής στο ίδιο πιο κάτω σχήμα. (Σημειώνεται και παρατηρείται ότι η ταχύτητα **V** του ρευστού εφάπτεται συνεχώς στις γραμμές ροής.)

```
In[73]:= Show[Cylinder, VelocityField2, StreamLines,
           PlotLabel → "ΠΕΔΙΟ ΤΑΧΥΤΗΤΑΣ ΚΑΙ ΓΡΑΜΜΕΣ ΡΟΗΣ", DefaultFont → {"Times-Bold", 13},
           PlotRange → {{-4.1, 4.4}, {-3.35, 3.35}}, AspectRatio → 3.35 / 4.25,
           Background → RGBColor[0.85, 1, 0.8], Frame → True, ImageSize → 400];
```



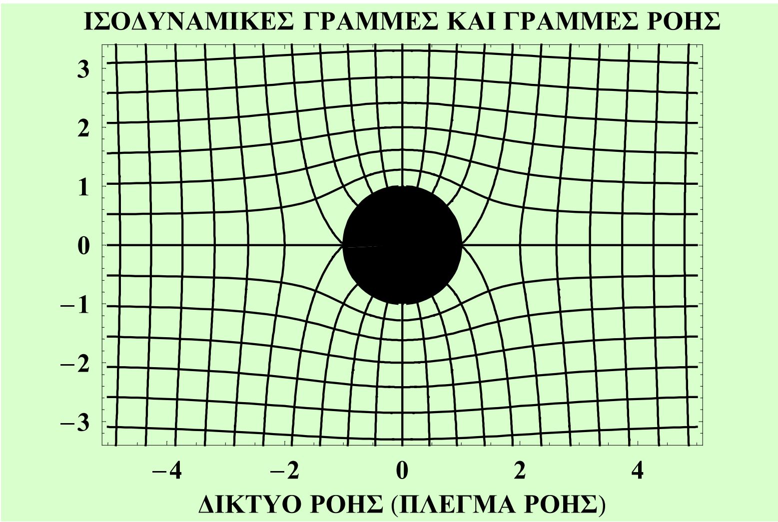
Ακριβώς όπως σχεδιάσαμε πριν με την εντολή **ImplicitPlot** τις γραμμές ροής, streamlines (με σταθερή τη ροϊκή συνάρτηση: $\Psi(x, y) = c$), το ίδιο ακριβώς μπορούμε να σχεδιάσουμε και τις ισοδυναμικές γραμμές, equipotential lines (με σταθερό το δυναμικό ταχύτητας: $\Phi(x, y) = c$). Για να δούμε λιγάκι κι αυτό το σχήμα:

```
In[74]:= EquipotentialLines =
          ImplicitPlot[Table[Φ[x, y][[2]] == c, {c, -5, 5, 0.5}], {x, -5, 5}, {y, -3.4, 3.4},
           AspectRatio → 3.4 / 5, PlotPoints → 150, PlotStyle → Thickness[0.004],
           Background → RGBColor[0.85, 1, 0.8], ImageSize → 324];
```



(Σημειώνεται πως αντί για την εντολή **ImplicitPlot** μπορεί να χρησιμοποιηθεί και η εντολή **ContourPlot**.) Μπορούμε βέβαια να παρουσιάσουμε στο ίδιο σχήμα και τις ισοδυναμικές γραμμές και τις γραμμές ροής:

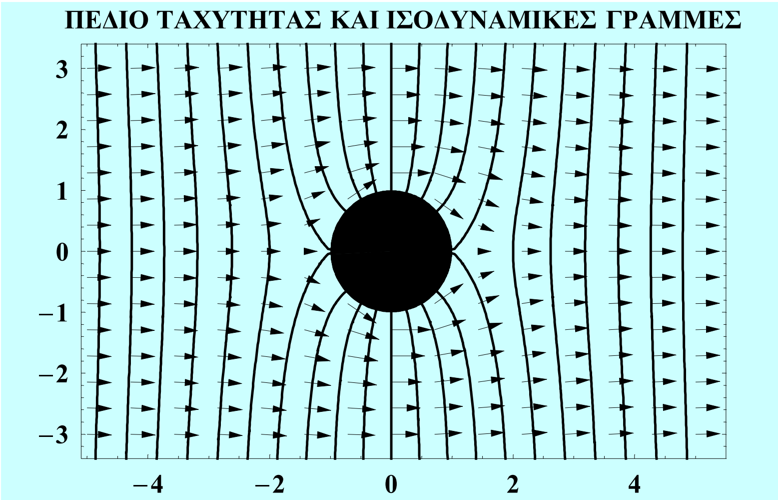
```
In[75]:= Show[Cylinder, EquipotentialLines, StreamLines,
          PlotLabel -> "ΙΣΟΔΥΝΑΜΙΚΕΣ ΓΡΑΜΜΕΣ ΚΑΙ ΓΡΑΜΜΕΣ ΡΟΗΣ",
          FrameLabel → {"ΔΙΚΤΥΟ ΡΟΗΣ (ΠΛΕΓΜΑ ΡΟΗΣ)", "", "", ""},
          DefaultFont → {"Times-Bold", 13},
          PlotRange → {{-5.1, 5.1}, {-3.4, 3.4}}, AspectRatio → 3.4 / 5.1,
          Background → RGBColor[0.85, 1, 0.8], Frame → True, ImageSize → 382];
```



Παρατηρούμε φυσικά ότι, όπως ήδη γνωρίζουμε από την Παράγραφο B3.4.5 του Μέρους B των διδακτικών βιβλίων, οι ισοδυναμικές γραμμές και οι γραμμές ροής αποτελούν σύστημα ορθογωνίων τροχιών. (Η κάθε ισοδυναμική γραμμή τέμνει κάθε γραμμή ροής κάθετα, ορθογώνια και αντίστροφα βέβαια.) Όλες μαζί αυτές οι γραμμές (ή καμπύλες): τόσο οι γραμμές ροής (streamlines) όσο και οι ισοδυναμικές γραμμές (equipotential lines) σχηματίζουν ένα δίκτυο ροής (ή πλέγμα ροής, flow net), όπως στο σχήμα.

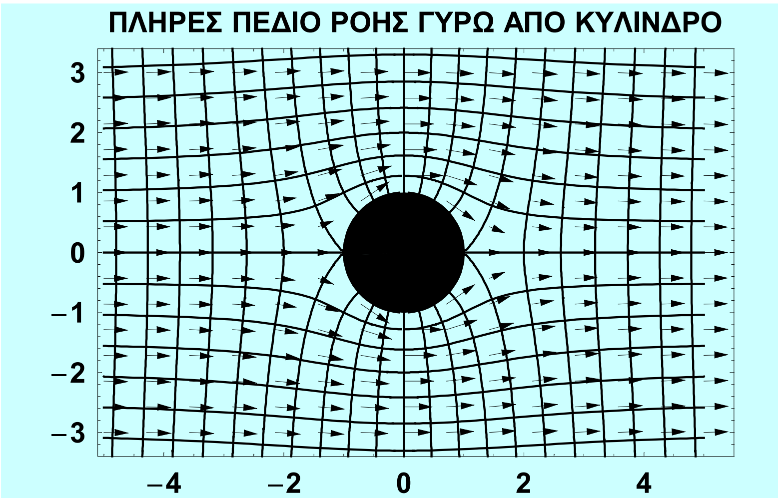
Ασφαλώς μπορούμε να παρουσιάσουμε στο ίδιο σχήμα τόσο τις ισοδυναμικές γραμμές (equipotential lines) όσο και το πεδίο ταχύτητας (velocity field) στην παρούσα ροή (flow) γύρω από τον κύλινδρο:

```
In[76]:= Show[Cylinder, VelocityField4, EquipotentialLines,
           PlotLabel → "ΠΕΔΙΟ ΤΑΧΥΤΗΤΑΣ ΚΑΙ ΙΣΟΔΥΝΑΜΙΚΕΣ ΓΡΑΜΜΕΣ",
           DefaultFont → {"Times-Bold", 13},
           PlotRange → {{-5.1, 5.5}, {-3.4, 3.4}}, AspectRatio → 3.4 / 5.3,
           Background → RGBColor[0.8, 1, 1], Frame → True, ImageSize → 382];
```



Έχουμε τέλος τη δυνατότητα να παρουσιάσουμε στο ίδιο σχήμα (α) το πεδίο ταχύτητας της ροής μας, (β) τις ισοδυναμικές γραμμές και (γ) τις γραμμές ροής ταυτόχρονα (ξανά με τη χρήση της εντολής **Show**):

```
In[77]:= Show[Cylinder, VelocityField4, EquipotentialLines,
           StreamLines, PlotLabel → "ΠΛΗΡΕΣ ΠΕΔΙΟ ΡΟΗΣ ΓΥΡΩ ΑΠΟ ΚΥΛΙΝΔΡΟ",
           DefaultFont → {"Arial-Bold", 14},
           PlotRange → {{-5.1, 5.5}, {-3.4, 3.4}}, AspectRatio → 3.4 / 5.3,
           Background → RGBColor[0.8, 1, 1], Frame → True, ImageSize → 382];
```



Αρκετά ως εδώ! Σταματάμε! Πήραμε ήδη μια καλή εικόνα των δυνατοτήτων της *Mathematica* για τη γραφική παράσταση διδιάστατων πεδίων μόνιμης αστρόβιλης ροής ιδεατού ρευστού στη Ρευστομηχανική: (α) πεδίο ταχύτητας, (β) ισοδυναμικές γραμμές και (γ) γραμμές ροής. (Με τις γνώσεις που αποκτήσαμε μπορούμε να σχεδιάσουμε πάρα πολλά πεδία ροής, όπως τα πεδία ροής της πηγής και της δίνης.) Εδώ οφείλουμε πάντως να διερωτηθούμε πόση ώρα στ' αλήθεια θα χρειαζόταν ο Πολιτικός Μηχανικός, για να ετοιμάσει χωρίς τον υπολογιστή του το πιο πάνω σχήμα και σωστά μάλιστα και επίσης με κάπως αξιοπρεπή εμφάνιση (όπως πιο πάνω); Άραγε αξίζει να αφιερώσει όλο αυτόν το χρόνο σε υπολογισμούς;

■ *Notebook E13*
ΕΝΤΟΛΕΣ ΓΙΑ ΤΗΝ ΕΠΙΛΥΣΗ ΕΞΙΣΩΣΕΩΝ

10 ΕΝΤΟΛΕΣ: **E1. Solve, E2. Roots, E3. LinearSolve, E4. Reduce, E5. NSolve, E6. FindRoot, E7. BesselJZeros, E8. BesselYZeros, E9. BesselJPrimeZeros, E10. BesselYPrimeZeros**

■ ΕΝΤΟΛΗ Ε1: ΕΠΙΛΥΣΗ ΕΞΙΣΩΣΕΩΝ

Solve[*Εξίσωση, Άγνωστος*]

Solve[*ΛίσταΕξισώσεων, ΛίσταΑγνώστων*]

Πρόκειται για τη συνηθισμένη, τη γενική εντολή επιλύσεως εξισώσεων, συνήθως αλγεβρικών, μερικές φορές και υπερβατικών (που περιέχουν, π.χ., τριγωνομετρικές ή υπερβολικές συναρτήσεις). Η εξίσωση μπορεί να περιέχει και παραμέτρους. Με την εντολή αυτή επιλύονται επίσης και συστήματα εξισώσεων. Σε κάθε περίπτωση πρέπει να χρησιμοποιείται το σύμβολο της εξισώσεως (το διπλό ίσον ==) και όχι το σύμβολο καθορισμού τιμής (το απλό ίσον =). Στις πολυωνυμικές εξισώσεις κλειστές λύσεις υπάρχουν συνήθως μέχρι και την τεταρτοβάθμια εξίσωση. Η λύση δίνεται πάντοτε σε μορφή λίστας κανόνων αντικαταστάσεως για τον άγνωστο ή τους αγνώστους που έχουν προσδιορισθεί. Παραδείγματα τώρα: μια τριτοβάθμια εξίσωση με αριθμητικούς συντελεστές εδώ με τρεις πραγματικές λύσεις:

In[1]:= `sol1 = Solve[x^3 + 2 x^2 - 1 == 0, x]`

Out[1]= $\left\{\{x \to -1\}, \left\{x \to \frac{1}{2}\left(-1-\sqrt{5}\right)\right\}, \left\{x \to \frac{1}{2}\left(-1+\sqrt{5}\right)\right\}\right\}$

In[2]:= `{sol1[[1]], sol1[[2]], sol1[[3]], NumberOfSolutions = Length[sol1]}`

Out[2]= $\left\{\{x \to -1\}, \left\{x \to \frac{1}{2}\left(-1-\sqrt{5}\right)\right\}, \left\{x \to \frac{1}{2}\left(-1+\sqrt{5}\right)\right\}, 3\right\}$

In[3]:= `NumericalApproximationsOfSolutions = N[sol1, 18]`

Out[3]= $\{\{x \to -1.00000000000000000\}, \{x \to -1.61803398874989485\}, \{x \to 0.618033988749894848\}\}$

Μια δευτεροβάθμια εξίσωση ως προς *x* με συντελεστές σύμβολα με δύο πραγματικές λύσεις ως προς *x*

In[4]:= `{eq2 = a x^2 + b x + c == 0, sol2 = Solve[eq2, x], Solve[eq2, a]}`

Out[4]= $\left\{c + b\,x + a\,x^2 == 0, \left\{\left\{x \to \frac{-b-\sqrt{b^2-4\,a\,c}}{2\,a}\right\}, \left\{x \to \frac{-b+\sqrt{b^2-4\,a\,c}}{2\,a}\right\}\right\}, \left\{\left\{a \to \frac{-c-b\,x}{x^2}\right\}\right\}\right\}$

In[5]:= `{Length[sol2], Table[sol2[[k]], {k, 1, Length[sol2]}]}`

Out[5]= $\left\{2, \left\{\left\{x \to \frac{-b-\sqrt{b^2-4\,a\,c}}{2\,a}\right\}, \left\{x \to \frac{-b+\sqrt{b^2-4\,a\,c}}{2\,a}\right\}\right\}\right\}$

Μια πολύ απλή εξίσωση δεκάτου βαθμού με δέκα ρίζες ειδικά εδώ σε κλειστή μορφή, κάτι που είναι βέβαια σπάνιο. Από αυτές οι οκτώ είναι μιγαδικές και μόνο οι δύο είναι πραγματικές:

```
In[6]:= {eq3 = t^10 == 1, sol3 = Solve[eq3, t], Length[sol3], NumericalValues = N[sol3]}

Out[6]= {t^10 == 1, {{t → -1}, {t → 1}, {t → -(-1)^(1/5)}, {t → (-1)^(1/5)}, {t → -(-1)^(2/5)},
  {t → (-1)^(2/5)}, {t → -(-1)^(3/5)}, {t → (-1)^(3/5)}, {t → -(-1)^(4/5)}, {t → (-1)^(4/5)}}, 10,
 {{t → -1.}, {t → 1.}, {t → -0.809017 - 0.587785 i}, {t → 0.809017 + 0.587785 i},
  {t → -0.309017 - 0.951057 i}, {t → 0.309017 + 0.951057 i}, {t → 0.309017 - 0.951057 i},
  {t → -0.309017 + 0.951057 i}, {t → 0.809017 - 0.587785 i}, {t → -0.809017 + 0.587785 i}}}
```

Μια απλή εξίσωση με τετραγωνικές ρίζες. Η *Mathematica* προσδιορίζει μόνο τις αληθινές ρίζες και απαλείφει τις παρασιτικές ρίζες που μπορούν να παρουσιασθούν μετά από τις αναγκαίες υψώσεις σε δυνάμεις. Η *Mathematica* επαληθεύει μόνη της (αυτόματα) τις λύσεις που βρίσκει με την εντολή **Solve**

```
In[7]:= {EquationWithRoots = Sqrt[x] + Sqrt[2 x] + 5 x == 10,
         solution = Solve[EquationWithRoots, x], Length[solution], N[solution]}
```

Out[7]= $\left\{\sqrt{x}+\sqrt{2}\,\sqrt{x}+5\,x==10,\ \left\{\left\{x\to\frac{1}{50}\left(103+2\sqrt{2}-\sqrt{617+412\sqrt{2}}\right)\right\}\right\},\ 1,\ \{\{x\to 1.42385\}\}\right\}$

Μια απλή τριγωνομετρική εξίσωση που έχει άπειρες λύσεις (ρίζες). Η *Mathematica* προσδιορίζει μόνο μία από τις λύσεις αυτές (γενικά τη βασική) με προειδοποίηση όμως για την πιθανή ύπαρξη κι άλλων λύσεων

```
In[8]:= TrigonometricEquation = Sin[a x] == b; sol4 = Solve[TrigonometricEquation, x]

        Solve::ifun : Inverse functions are being used by Solve, so some solutions may not be found.
```

Out[8]= $\left\{\left\{x\to\frac{\text{ArcSin}[b]}{a}\right\}\right\}$

Μια απλή εξίσωση που περιέχει την εκθετική συνάρτηση

```
In[9]:= {ExponentialEquation = Exp[a x] Exp[b x] == 5, Solve[ExponentialEquation, x]}

        Solve::ifun : Inverse functions are being used by Solve, so some solutions may not be found.
```

Out[9]= $\left\{e^{a\,x+b\,x}==5,\ \left\{\left\{x\to\frac{\text{Log}[5]}{a+b}\right\}\right\}\right\}$

Η ίδια εντολή **Solve** είναι και η γενική εντολή επιλύσεως συστημάτων εξισώσεων είτε γραμμικών είτε μη γραμμικών. Φυσικά σ' αυτήν την περίπτωση δίνουμε σαν πρώτο όρισμα της εντολής τη σχετική λίστα εξισώσεων (αντί απλή εξίσωση) και σαν δεύτερο όρισμα τη λίστα των αγνώστων που ζητάμε να βρούμε

```
In[10]:= SystemOfEquations = {a x^2 + b x == c, a y + 2 b y^2 == 3}; Solve[SystemOfEquations, {x, y}]
```

Out[10]= $\left\{\left\{x\to\frac{-b-\sqrt{b^2+4\,a\,c}}{2\,a},\ y\to\frac{-a-\sqrt{a^2+24\,b}}{4\,b}\right\},\ \left\{x\to\frac{-b-\sqrt{b^2+4\,a\,c}}{2\,a},\ y\to\frac{-a+\sqrt{a^2+24\,b}}{4\,b}\right\},\ \left\{x\to\frac{-b+\sqrt{b^2+4\,a\,c}}{2\,a},\ y\to\frac{-a-\sqrt{a^2+24\,b}}{4\,b}\right\},\ \left\{x\to\frac{-b+\sqrt{b^2+4\,a\,c}}{2\,a},\ y\to\frac{-a+\sqrt{a^2+24\,b}}{4\,b}\right\}\right\}$

```
In[11]:= ThreeLinearEquations = {2 x + 5 y - 3 z == 9, 4 x - 10 y + 7 z == -7, -3 x + 2 y + 8 z == -6};

In[12]:= {Solution1 = Solve[ThreeLinearEquations, {x, y, z}], Length[Solution1], N[Solution1]}
```

Out[12]= $\left\{\left\{\left\{x\to\frac{554}{387},\ y\to\frac{370}{387},\ z\to-\frac{175}{387}\right\}\right\},\ 1,\ \{\{x\to 1.43152,\ y\to 0.956072,\ z\to -0.452196\}\}\right\}$

Επαλήθευση (verification) της λύσεως που βρέθηκε:

```
In[13]:= VerificationOfTheSolution = ThreeLinearEquations /. Solution1

Out[13]= {{True, True, True}}
```

■ ΕΝΤΟΛΗ Ε2: ΕΠΙΛΥΣΗ ΠΟΛΥΩΝΥΜΙΚΩΝ ΕΞΙΣΩΣΕΩΝ

Roots[*ΠολυωνυμικήΕξίσωση, Άγνωστος*]

Η εντολή αυτή δίνει τη λύση μιας πολυωνυμικής εξισώσεως *(καμιάς άλλης μορφής!)* σε μορφή λογικής παραστάσεως σε αντίθεση με την προηγούμενη εντολή **Solve**, που τη δίνει σε μορφή λίστας κανόνων αντικαταστάσεως. Πέρα από αυτήν τη διαφορά οι λύσεις βέβαια συμπίπτουν. Παράδειγμα:

In[14]:= **{sol1 = Solve[x^3 + x + 2 == 0, x], Length[sol1], N[sol1], sol1[[2, 1, 2]]}**

Out[14]= $\left\{\left\{\{x\to -1\},\ \left\{x\to \frac{1}{2}\left(1-i\sqrt{7}\right)\right\},\ \left\{x\to \frac{1}{2}\left(1+i\sqrt{7}\right)\right\}\right\},\ 3,\ \{\{x\to -1.\},\ \{x\to 0.5-1.32288\,i\},\ \{x\to 0.5+1.32288\,i\}\},\ \frac{1}{2}\left(1-i\sqrt{7}\right)\right\}$

In[15]:= **{sol2 = Roots[x^3 + x + 2 == 0, x], N[sol2], sol2[[1, 2]], sol1[[2, 1, 2]] == sol2[[1, 2]]}**

Out[15]= $\left\{x == \frac{1}{2}\left(1-i\sqrt{7}\right) \,||\, x == \frac{1}{2}\left(1+i\sqrt{7}\right) \,||\, x == -1,\ x == 0.5-1.32288\,i \,||\, x == 0.5+1.32288\,i \,||\, x == -1.,\ \frac{1}{2}\left(1-i\sqrt{7}\right),\ \text{True}\right\}$

■ ΕΝΤΟΛΗ Ε3: ΕΠΙΛΥΣΗ ΓΡΑΜΜΙΚΩΝ ΕΞΙΣΩΣΕΩΝ

LinearSolve[*ΜητρώοΣυντελεστώνΤωνΑγνώστων, ΔιάνυσμαΔεξιώνΜελών*]

Η εντολή αυτή βρίσκει τη λύση ενός γραμμικού συστήματος αλγεβρικών εξισώσεων το οποίο είναι γραμμένο στη μητρωϊκή μορφή **AX** = **B** με το μητρώο **A** στο πρώτο όρισμά της και το διάνυσμα **B** στο δεύτερο όρισμά της. Το αποτέλεσμα είναι το διάνυσμα **X** των αγνώστων. Τέτοια γραμμικά συστήματα (σε μητρωϊκή μορφή, σε μορφή με μητρώα) απαντώνται πάρα πολύ συχνά στην Επιστήμη του Πολιτικού Μηχανικού. Εδώ θα επαναλάβουμε το σχετικό παράδειγμα της εντολής **Solve**

In[16]:= **ThreeLinearEquations = {2 x + 5 y − 3 z == 9, 4 x − 10 y + 7 z == −7, −3 x + 2 y + 8 z == −6};**

In[17]:= **{Solution1 = Solve[ThreeLinearEquations, {x, y, z}], Length[Solution1], N[Solution1]}**

Out[17]= $\left\{\left\{\left\{x\to \frac{554}{387},\ y\to \frac{370}{387},\ z\to -\frac{175}{387}\right\}\right\},\ 1,\ \{\{x\to 1.43152,\ y\to 0.956072,\ z\to -0.452196\}\}\right\}$

In[18]:= **{A = {{2, 5, −3}, {4, −10, 7}, {−3, 2, 8}}, B = {9, −7, −6}};**

In[19]:= **{Solution2 = LinearSolve[A, B], Table[Solution1[[1, k, 2]] == Solution2[[k]], {k, 1, 3}]}**

Out[19]= $\left\{\left\{\frac{554}{387},\ \frac{370}{387},\ -\frac{175}{387}\right\},\ \{\text{True, True, True}\}\right\}$

Πολύ σχετική με την παρούσα εντολή **LinearSolve** είναι και η εντολή **RowReduce** στο Notebook E8 (εντολή E8: M17) για την ανηγμένη κατά γραμμές μορφή μητρώου. Στο παράδειγμά μας οδηγεί κι αυτή στην ίδια ακριβώς λύση όπως και η εντολή **LinearSolve** γραμμένη όμως σε λίγο διαφορετική μορφή: με τη λύση να αποτελείται από την τελευταία στήλη (τη δεξιά στήλη) του μητρώου που προκύπτει:

In[20]:= **RowReduce[{{2, 5, −3, 9}, {4, −10, 7, −7}, {−3, 2, 8, −6}}]**

Out[20]= $\left\{\left\{1,\ 0,\ 0,\ \frac{554}{387}\right\},\ \left\{0,\ 1,\ 0,\ \frac{370}{387}\right\},\ \left\{0,\ 0,\ 1,\ -\frac{175}{387}\right\}\right\}$

■ ΕΝΤΟΛΗ Ε4: ΛΕΠΤΟΜΕΡΗΣ ΕΠΙΛΥΣΗ ΕΞΙΣΩΣΕΩΝ

Reduce[*Εξίσωση, Άγνωστος*]

Reduce[*ΛίσταΕξισώσεων, ΛίσταΑγνώστων*]

Η εντολή αυτή απλοποιεί την εξίσωση ή το σύστημα των εξισώσεων που δίνεται με μεγάλη προσοχή. Καταλήγει έτσι στη λεπτομερή λύση του, που παίρνει υπόψη της όλες τις περιπτώσεις που μπορούν να παρουσιασθούν. Αντίθετα η εντολή **Solve** κάνει υποθέσεις κατά την επίλυση χωρίς μάλιστα να τις αναφέρει. Επίσης τα αποτελέσματα της εντολής **Reduce** είναι σε μορφή εκφράσεων με τη χρήση των λογικών συμβόλων **And** (επίσης && ή ∧), **Or** (επίσης ¦¦ ή ∨) και **Not** (επίσης != ή ¬). Παραδείγματα:

Στο πιο κάτω πολύ απλό παράδειγμα της πρωτοβάθμιας αλγεβρικής εξισώσεως παρατηρούμε πως η εντολή **Solve** απλά υποθέτει τη σταθερά *a* μη μηδενική και διαιρεί μ' αυτήν. Αντίθετα η εντολή **Reduce** μας δίνει την πλήρη λύση της ίδιας εξισώσεως με τη διάκριση περιπτώσεων και τη χρήση λογικών τελεστών στη λύση που βρίσκει. Με την έννοια αυτή όσες φορές ενδιαφερόμαστε για τη λύση με διερεύνηση περιπτώσεων πρέπει να χρησιμοποιούμε την εντολή **Reduce** αντί για την εντολή **Solve**:

```
In[21]:= {Solve[a x == b, x], Reduce[a x == b, x], Reduce[a x == b, x] // TraditionalForm}
```

Out[21]= $\left\{\left\{\left\{x \to \frac{b}{a}\right\}\right\},\ a == 0\ \&\&\ b == 0\ ||\ x == \frac{b}{a}\ \&\&\ a \neq 0,\ a == 0 \land b == 0 \bigvee x == \frac{b}{a} \bigwedge a \neq 0\right\}$

Και το αντίστοιχο παράδειγμα της δευτεροβάθμιας αλγεβρικής εξισώσεως. Παρατηρούμε και πάλι με πόσο μεγάλη *(στ' αλήθεια!)* προσοχή έχει δουλέψει η εντολή **Reduce** σε αντίθεση με την εντολή **Solve**

```
In[22]:= TheTwoSolutions = {Solve[a x^2 + b x + c == 0, x], Reduce[a x^2 + b x + c == 0, x]}
```

Out[22]= $$\left\{\left\{\left\{x \to \frac{-b-\sqrt{b^2-4 a c}}{2 a}\right\},\ \left\{x \to \frac{-b+\sqrt{b^2-4 a c}}{2 a}\right\}\right\},\ x == \frac{-b-\sqrt{b^2-4 a c}}{2 a}\ \&\&\ a \neq 0\ ||\right.$$
$$\left. x == \frac{-b+\sqrt{b^2-4 a c}}{2 a}\ \&\&\ a \neq 0\ ||\ a == 0\ \&\&\ b == 0\ \&\&\ c == 0\ ||\ a == 0\ \&\&\ x == -\frac{c}{b}\ \&\&\ b \neq 0\right\}$$

```
In[23]:= TheTwoSolutions // TraditionalForm
```

Out[23]//TraditionalForm=

$$\left\{\left\{\left\{x \to \frac{-b-\sqrt{b^2-4 a c}}{2 a}\right\},\ \left\{x \to \frac{\sqrt{b^2-4 a c}-b}{2 a}\right\}\right\},\ x == \frac{-b-\sqrt{b^2-4 a c}}{2 a} \bigwedge a \neq 0 \bigvee\right.$$
$$\left. x == \frac{\sqrt{b^2-4 a c}-b}{2 a} \bigwedge a \neq 0 \bigvee a == 0 \land b == 0 \land c == 0 \bigvee a == 0 \bigwedge x == -\frac{c}{b} \bigwedge b \neq 0\right\}$$

■ ΕΝΤΟΛΗ Ε5: ΕΠΙΛΥΣΗ ΕΞΙΣΩΣΕΩΝ ΣΕ ΑΡΙΘΜΗΤΙΚΗ ΜΟΡΦΗ

NSolve[*Εξίσωση, Άγνωστος*]

NSolve[*ΛίσταΕξισώσεων, ΛίσταΑγνώστων*]

Πρόκειται για την ανάλογη εντολή της εντολής **Solve**, αλλα τώρα με τις λύσεις σε αριθμητική μορφή. Η εξίσωση μπορεί να περιέχει σύμβολα. Μερικές φορές μπορεί να χρησιμοποιηθεί σαν τρίτο όρισμα και η ακρίβεια των αριθμητικών προσεγγίσεων των ριζών. Ουσιαστικά τα ίδια αποτελέσματα τα παίρνουμε χρησιμοποιώντας την εντολή **Solve** και στη συνέχεια στη λύση ή τις λύσεις που αυτή δίνει την εντολή αριθμητικής προσεγγίσεως **N**.

Παραδείγματα:

```
In[24]:= {eq = x^4 - 10 == 0, Solve[eq, x], N[Solve[eq, x]], NSolve[eq, x]}

Out[24]= {-10 + x^4 == 0, {{x -> -10^(1/4)}, {x -> -ⅈ 10^(1/4)}, {x -> ⅈ 10^(1/4)}, {x -> 10^(1/4)}},
  {{x -> -1.77828}, {x -> 0. - 1.77828 ⅈ}, {x -> 0. + 1.77828 ⅈ}, {x -> 1.77828}},
  {{x -> -1.77828}, {x -> -1.77828 ⅈ}, {x -> 1.77828 ⅈ}, {x -> 1.77828}}}

In[25]:= {sol = NSolve[a x^3 + 3 == 0, x], Length[sol]}

Out[25]= {{{x -> -1.44225/a^(1/3)}, {x -> (0.721125 + 1.24902 ⅈ)/a^(1/3)}, {x -> (0.721125 - 1.24902 ⅈ)/a^(1/3)}}, 3}
```

Προφανώς σε υπερβατικές εξισώσεις (όπως είναι οι εξισώσεις με τριγωνομετρικές, υπερβολικές και εκθετικές συναρτήσεις και τις αντίστροφές τους συναρτήσεις) δεν προσδιορίζονται όλες οι λύσεις. Μερικές φορές μάλιστα η εντολή **NSolve** (όπως και η εντολή **Solve** βέβαια) δε μπορεί να προσδιορίσει καμία απολύτως λύση. Σ' αυτές τις περιπτώσεις εμφανίζονται προειδοποιητικά μηνύματα της *Mathematica*. Σε τέτοιες περιπτώσεις είναι σκόπιμη η χρήση της εντολής **FindRoot**, που κάνει καθαρά αριθμητική επίλυση της εξισώσεως. Την εντολή αυτή **FindRoot** θα τη δούμε αμέσως παρακάτω.

```
In[26]:= NSolve[Cos[z] + Sin[z] == -1, z, 18]

    Solve::ifun : Inverse functions are being used by Solve, so some solutions may not be found.

Out[26]= {{z -> -3.14159265358979324}, {z -> -1.57079632679489662}, {z -> 3.14159265358979324}}

In[27]:= NSolve[Cos[z] Cosh[z] == -1, z]

    Solve::tdep :
     The equations appear to involve the variables to be solved for in an essentially non-algebraic way.

Out[27]= NSolve[Cos[z] Cosh[z] == -1, z]
```

■ ΕΝΤΟΛΗ Ε6:

FindRoot[*Εξίσωση, {Άγνωστος, ΣημείοΕκκινήσεως}*]

FindRoot[*Εξίσωση, {Άγνωστος, {ΣημείοΕκκινήσεως-1, ΣημείοΕκκινήσεως-2*}}]

FindRoot[*Εξίσωση, {Άγνωστος, ΣημείοΕκκινήσεως, ΑρχήΔιαστήματος, ΤέλοςΔιαστήματος}*]

Στην πρώτη μορφή της η εντολή αυτή **FindRoot** προσδιορίζει αριθμητικά (με τη μέθοδο των Newton–Raphson) μία ρίζα της εξισώσεως που δίνεται ξεκινώντας από ένα σημείο εκκινήσεως. Στη δεύτερη μορφή της εργάζεται ανάλογα, αλλά τώρα με τη μέθοδο της τέμνουσας στο διάστημα $[a, b]$ το οποίο καθορίζεται με τη λίστα δεξιά στο δεύτερο όρισμα της εντολής. Τέλος στην τρίτη μορφή της η ίδια εντολή FindRoot προσπαθεί να βρει μία ρίζα ξεκινώντας πάλι από το σημείο εκκινήσεως που δίνεται, αλλά μόνο στο διάστημα $[c, d]$ που καθορίζεται μετά το σημείο εκκινήσεως. Παραδείγματα:

Θεωρούμε την υπερβατική εξίσωση που παρουσιάζεται στον προσδιορισμό των ιδιοσυχνοτήτων ω_k σε ιδιοταλαντώσεις συνήθους δοκού μήκους L, εδώ συγκεκριμένα μιας μονόπακτης δοκού με πακτωμένο το αριστερό άκρο της $x = 0$ και με κύλιση στο δεξιό $x = L$. Σ' αυτό το ενδιαφέρον πρόβλημα για τον Πολιτικό Μηχανικό, που το μελετήσαμε λεπτομερώς στην Παράγραφο B6.2.5, προκύπτει η υπερβατική εξίσωση (6.2.67), ισοδύναμα (6.2.74) και επίσης ισοδύναμα (6.2.85), που είναι ουσιαστικά η εξίσωση ιδιοσυχνοτήτων:

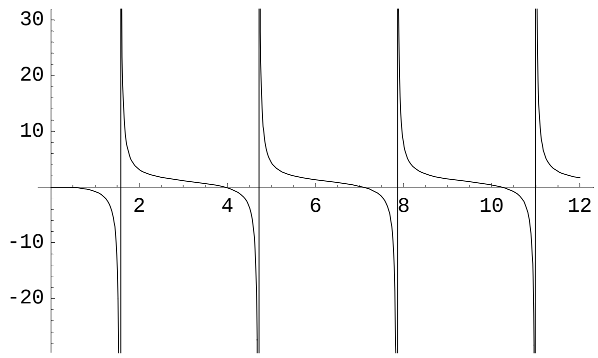
```
In[28]:= FrequencyEquation = Tanh[z] == Tan[z];
```

με $z = \beta L$ τον τελικό άγνωστο. Ούτε συζήτηση δε μπορεί να γίνει για την εντολή **Solve** ή την **NSolve** ή τη **Reduce** για μια τέτοια υπερβατική εξίσωση: αποτυγχάνουν πλήρως. Ας το δούμε αυτό για τη δεύτερη από τις εντολές αυτές: την εντολή **NSolve**. (Ανάλογα ισχύουν και για τις εντολές **Solve** και **Reduce**.)

```
In[29]:= NSolve[FrequencyEquation, z]

        Solve::tdep :
         The equations appear to involve the variables to be solved for in an essentially non-algebraic way.

Out[29]= NSolve[Tanh[z] == Tan[z], z]
```

Το μόνο που πήραμε ήταν το μήνυμα της αποτυχίας μαζί με την άλυτη εξίσωσή μας. (Και από "αριθμητική" εντολή: την **NSolve**.) Αυτό βέβαια δε συμβαίνει από κάποιον ελλιπή προγραμματισμό των εντολών αυτών. Συμβαίνει απλά και μόνο, επειδή η εξίσωσή μας δεν έχει κλειστή λύση. Θα τη λύσουμε λοιπόν αριθμητικά με την παρούσα εντολή **FindRoot**. Καταρχήν κάνουμε τη σχετική γραφική παράσταση:

```
In[30]:= Plot[Tanh[z] - Tan[z], {z, 0, 12}]
```




```
Out[30]= - Graphics -
```

Παρατηρούμε ότι εκτός από την προφανή ρίζα στο $z_0 = 0$ έχουμε ρίζες κοντά στο 4, στο 7 και στο 10. Αυτές είναι και οι ρίζες που αντιστοιχούν στις τρεις πρώτες ιδιοσυχνότητες ω_k της δοκού μας. Τις προσδιορίζουμε χρησιμοποιώντας εδώ την εντολή **FindRoot** τρεις φορες, πρώτα στην πρώτη της μορφή

```
In[31]:= {FindRoot[FrequencyEquation, {z, 4}],
          FindRoot[FrequencyEquation, {z, 7}],
          FindRoot[FrequencyEquation, {z, 10}]}

Out[31]= {{z → 3.9266}, {z → 7.06858}, {z → 10.2102}}
```

Μεγαλύτερη ακρίβεια πετυχαίνουμε εκτελώντας τις πράξεις στη *Mathematica* με μεγαλύτερη ακρίβεια **WorkingPrecision**, π.χ. με 35 σημαντικά ψηφία. Έτσι μπορούμε να πάρουμε άνετα πιο ακριβείς ρίζες, π.χ.

```
In[32]:= root1 = FindRoot[FrequencyEquation, {z, 4}, WorkingPrecision → 35]

Out[32]= {z → 3.9266023120479187782385333436270249}
```

Εναλλακτικά μπορούμε να χρησιμοποιήσουμε τη δεύτερη μορφή της εντολής **FindRoot**: με ένα διάστημα [*a*, *b*], εδώ το [3, 5], καθώς και την τρίτη μορφή της: με αρχική προσέγγιση και με διάστημα, π.χ. ως εξής:

```
In[33]:= {FindRoot[FrequencyEquation, {z, {3, 5}}], FindRoot[FrequencyEquation, {z, 4, 1, 6}]}

Out[33]= {{z → 3.9266}, {z → 3.9266}}
```

■ ΕΝΤΟΛΕΣ E7, E8, E9 ΚΑΙ E10: ΡΙΖΕΣ ΣΥΝΑΡΤΗΣΕΩΝ BESSEL

BesselJZeros[*ΤάξηΤηςΣυναρτήσεωςBessel, ΑριθμόςΖητούμενωνΡιζών*]

BesselYZeros[*ΤάξηΤηςΣυναρτήσεωςBessel, ΑριθμόςΖητούμενωνΡιζών*]

BesselJPrimeZeros[*ΤάξηΤηςΠαραγώγουΤηςΣυναρτήσεωςBessel, ΑριθμόςΖητούμενωνΡιζών*]

BesselYPrimeZeros[*ΤάξηΤηςΠαραγώγουΤηςΣυναρτήσεωςBessel, ΑριθμόςΖητούμενωνΡιζών*]

Για μη πολυωνυμικές εξισώσεις που δεν έχουν ρίζες (λύσεις) σε κλειστή μορφή η προηγούμενη εντολή **FindRoot** είναι η εντολή εκείνη που συνήθως χρησιμοποιείται για την εύρεση ριζών εξισώσεων. Οι τέσσερις παρούσες εντολές αποτελούν μια επέκταση της εντολής **FindRoot**, για την ακρίβεια χρησιμοποιούν την εντολή **FindRoot**, για να υπολογίσουν τις *n* πρώτες ρίζες των γνωστών συναρτήσεων Bessel $J_v(x)$ (πρώτου είδους, η πρώτη εντολή **BesselJZeros**), $Y_v(x)$ (δευτέρου είδους, η δεύτερη εντολή **BesselYZeros**) και των παραγώγων τους $J_v'(x)$ (η τρίτη εντολή **BesselJPrimeZeros**) και $Y_v'(x)$ (η τέταρτη εντολή **BesselYPrimeZeros**). Όλες τούτες οι εντολές ανήκουν στο πακέτο της *Mathematica* **NumericalMath‘BesselZeros‘**, που πρέπει να έχει κληθεί (φορτωθεί) πριν από τη χρήση των πιο πάνω εντολών. Τα πλεονεκτήματα των ειδικών αυτών εντολών έναντι της γενικής εντολής **FindRoot** είναι ότι υπολογίζουν με μία μόνο εντολή όλες μαζί τις ρίζες (και τις *n*) που ζητούνται. Και μάλιστα μεριμνούν οι ίδιες για τις κατάλληλες αρχικές προσεγγίσεις (που γενικά δεν είναι γνωστές) των ζητούμενων ριζών. Από πρακτικής απόψεως ρίζες συναρτήσεων Bessel παρουσιάζονται πάρα πολύ συχνά σαν ιδιοτιμές σε προβλήματα διαφορικών εξισώσεων με μερικές παραγώγους με αξονική (κυκλική ή κυλινδρική) συμμετρία. Παραδείγματα, αφού πρώτα φορτωθεί το πακέτο που προαναφέρθηκε, για τις εννέα πρώτες ρίζες των συναρτήσεων Bessel $J_0(x)$, $Y_0(x)$, $J_0'(x)$ και $Y_0'(x)$ (τάξεως $v = 0$) και μετά τις ανάλογες ρίζες για τις συναρτήσεις Bessel $J_1(x)$, $Y_1(x)$, $J_1'(x)$ και $Y_1'(x)$ (τάξεως $v = 1$):

```
In[34]:= Needs["NumericalMath`BesselZeros`"]

In[35]:= BesselJZeros[0, 9]

Out[35]= {2.40483, 5.52008, 8.65373, 11.7915, 14.9309, 18.0711, 21.2116, 24.3525, 27.4935}

In[36]:= BesselYZeros[0, 9]

Out[36]= {0.893577, 3.95768, 7.08605, 10.2223, 13.3611, 16.5009, 19.6413, 22.782, 25.923}

In[37]:= BesselJPrimeZeros[0, 9]

Out[37]= {3.83171, 7.01559, 10.1735, 13.3237, 16.4706, 19.6159, 22.7601, 25.9037, 29.0468}

In[38]:= BesselYPrimeZeros[0, 9]

Out[38]= {2.19714, 5.42968, 8.59601, 11.7492, 14.8974, 18.0434, 21.1881, 24.3319, 27.4753}

In[39]:= BesselJZeros[1, 9]

Out[39]= {3.83171, 7.01559, 10.1735, 13.3237, 16.4706, 19.6159, 22.7601, 25.9037, 29.0468}

In[40]:= BesselYZeros[1, 9]

Out[40]= {2.19714, 5.42968, 8.59601, 11.7492, 14.8974, 18.0434, 21.1881, 24.3319, 27.4753}
```

```
In[41]:= BesselJPrimeZeros[1, 9]

Out[41]= {1.84118, 5.33144, 8.53632, 11.706, 14.8636, 18.0155, 21.1644, 24.3113, 27.4571}

In[42]:= BesselYPrimeZeros[1, 9]

Out[42]= {3.68302, 6.9415, 10.1234, 13.2858, 16.4401, 19.5902, 22.738, 25.8843, 29.0296}
```

Παρατηρούμε πιο πάνω από τις ρίζες στις εντολές [37] και [39] ότι αυτές συμπίπτουν. Τό ίδιο ισχύει και για τις ρίζες στις εντολές [38] και [40]. Τούτα είναι εύλογα, επειδή ισχύουν οι εξής δύο γνωστές σχέσεις:

```
In[43]:= {D[BesselJ[0, x], x], D[BesselY[0, x], x]}

Out[43]= {-BesselJ[1, x], -BesselY[1, x]}
```

Με τη χρήση μεγάλης ακρίβειας στους υπολογισμούς στη *Mathematica* (με τη βοήθεια της επιλογής **WorkingPrecision**, που μας είναι ήδη γνωστή από την εντολή **FindRoot** προηγουμένως), πετυχαίνουμε την εύρεση ριζών συναρτήσεων Bessel με πάρα πολύ μεγάλη ακρίβεια, εδώ περίπου 75 σημαντικά ψηφία:

```
In[44]:= Off[General::spell1]

In[45]:= Jzeros = BesselJZeros[0, 10, WorkingPrecision → 75]

Out[45]= {2.40482555769577276862163187932645464312424490914596713570699909059676582,
          5.52007811028631064959660411281302742522186547878290985375755203814429082919,
          8.65372791291101221695419871266094668556579523127535561889147658302259995665,
          11.7915344390142816137430449119254589220229246996954467032505108790516465118,
          14.9309177084877859477625939973886822079158501156330281587741732188351933639,
          18.0710639679109225431478829756181765602489867470013260864233146352838205624,
          21.2116366298792589590783933505263068361818089759763998327382027337686882299,
          24.3524715307493027370579447631789071845693726751489270224060456377282775406,
          27.4934791320402547958772882346074145465295688605496220109388612300091310750,
          30.6346064684319751175495789268542327372735716291781471907550178997160244476}

In[46]:= Yzeros = BesselYZeros[0, 10, WorkingPrecision → 75]

Out[46]= {0.893576966279167521584887102058338241225146861930014487069228945110126188621,
          3.95767841931485786837567718691740128141860376556363062550751179484115237701,
          7.08605106030177269762362459682035246897151038117776446985516765235357776154,
          10.2223450434964170189920422763421871259940596131812411831190854840310423297,
          13.3610974738727634782676945857137864265791351748799264196287493067636445742,
          16.5009224415280907534211436664897741157513331047907082511444202634884373936,
          19.6413097008879397737760454722859800254415174918698689830372115929531567822,
          22.7820280472915593169320819683965166628163060018404614276284266796404574108,
          25.9229576531809227068721911462693733170525088745712452223305691111865223265,
          29.0640302527283980553047184051813443936050681215613445319582933552013663960}
```

Αυτήν την τόσο μεγάλη ακρίβεια την επιβεβαιώνουμε με την άμεση επαλήθευση των ριζών αυτών:

```
In[47]:= N[BesselJ[0, x] /. x → Jzeros]
```

Out[47]= $\{1.15426\times10^{-71}, -4.16165\times10^{-78}, -6.23862\times10^{-84}, 1.25326\times10^{-83}, -8.02914\times10^{-85}, -1.49441\times10^{-84}, -3.45534\times10^{-84}, -2.06531\times10^{-84}, 1.44314\times10^{-84}, -1.40893\times10^{-84}\}$

```
In[48]:= N[BesselY[0, x] /. x → Yzeros]
```

Out[48]= $\{1.46067\times10^{-89}, 1.57889\times10^{-83}, 5.9551\times10^{-88}, -4.6508\times10^{-88}, 1.78133\times10^{-89}, -3.15693\times10^{-89}, 1.48913\times10^{-88}, -7.25744\times10^{-89}, 3.13177\times10^{-88}, -9.62922\times10^{-89}\}$

■ *Notebook E14*
ΕΝΤΟΛΕΣ ΓΙΑ ΑΚΡΙΒΕΙΑ, ΠΑΡΕΜΒΟΛΗ ΚΑΙ ΠΡΟΣΕΓΓΙΣΕΙΣ
6 ΕΝΤΟΛΕΣ: N1. $MachinePrecision, N2. Precision, N3. Accuracy, N4. Interpolation, N5. InterpolatingPolynomial, N6. MiniMaxApproximation

ΠΑΡΑΤΗΡΗΣΗ: Για να αποφευχθούν τα δύο μηνύματα προειδοποιήσεως λαθών **spell** και **spell1** (που δεν παρουσιάζουν καμία χρησιμότητα εδώ), χρησιμοποιήθηκε η διπλή εντολή

```
In[1]:= Off[General::spell]; Off[General::spell1];
```

■ ΕΝΤΟΛΗ N1: ΑΚΡΙΒΕΙΑ ΜΗΧΑΝΗΣ

$MachinePrecision

Δίνει την ακρίβεια των αριθμητικών πράξεων στον ηλεκτρονικό υπολογιστή (στη "μηχανή" μας). Γενικά στους σημερινούς υπολογιστές αυτή είναι ίση με 16. Η εντολή αυτή δεν έχει κανένα απολύτως όρισμα. (Πρόκειται επομένως ουσιαστικά για ένα σύμβολο.) Η κλήση της είναι πάρα πολύ απλή:

```
In[2]:= $MachinePrecision

Out[2]= 16
```

■ ΕΝΤΟΛΗ N2: ΟΛΙΚΗ ΑΚΡΙΒΕΙΑ

Precision[*Αριθμός*]

Δίνει την ολική ακρίβεια ενός αριθμού, δηλαδή το συνολικό αριθμό των σωστών ψηφίων του αριθμού. Σ' αυτά περιλαμβάνονται τόσο τα ψηφία πριν την υποδιαστολή (το κόμμα) όσο και τα ψηφία μετά την υποδιαστολή: τα δεκαδικά ψηφία. Παραδείγματα παρουσιάζονται στην πιο κάτω εντολή **Accuracy**.

■ ΕΝΤΟΛΗ N3: ΔΕΚΑΔΙΚΗ ΑΚΡΙΒΕΙΑ

Accuracy[*Αριθμός*]

Δίνει τη δεκαδική ακρίβεια ενός αριθμού, δηλαδή τον αριθμό των σωστών δεκαδικών ψηφίων του αριθμού. Πρόκειται βέβαια για τα ψηφία μετά την υποδιαστολή (το κόμμα). Παραδείγματα:

Ο αριθμός π^{10} σε προσεγγιστική μορφή με συνολική ακρίβεια 20 ψηφίων:

```
In[3]:= Nπ = N[π^10, 20]

Out[3]= 93648.047476083020974
```

Η ολική ακρίβεια του αριθμού αυτού και η δεκαδική του ακρίβεια:

```
In[4]:= {Precision[Nπ], Accuracy[Nπ]}

Out[4]= {20, 15}
```

Προφανώς η πρώτη είναι 20 (μαζί με το ακέραιο μέρος του αριθμού), ενώ η δεύτερη 15 (μόνο τα δεκαδικά ψηφία του ίδιου αριθμού). Ανάλογα για τον αριθμό $10^6/3$, αυτός με έξι ακέραια ψηφία. Εδώ ο αριθμητικός υπολογισμός έγινε με την ακρίβεια του υπολογιστή **$MachinePrecision**, που είναι 16 συνολικά ψηφία:

```
In[5]:= {10^6 / 3., Precision[10^6 / 3.], Accuracy[10^6 / 3.]}

Out[5]= {333333., 16, 10}
```

■ ΕΝΤΟΛΗ N4: ΠΑΡΕΜΒΟΛΗ

Interpolation[*ΛίσταΣημείωνΠαρεμβολής*]

Δημιουργεί τη συνάρτηση παρεμβολής από τη λίστα των σημείων που δίνονται στο όρισμα. Η λίστα αυτή αποτελείται από τις *n* επιμέρους λίστες με δύο στοιχεία η καθεμιά τους, π.χ. **{x[k], y[k]}**, που καθορίζουν τα *n* σημεία παρεμβολής. Το αποτέλεσμα στη συνάρτηση παρεμβολής δίνεται στη μορφή καθαρής συναρτήσεως **InterpolationFunction** που μπορεί άμεσα να χρησιμοποιηθεί και σε παραπέρα υπολογισμούς. Παράδειγμα για τη συνάρτηση παρεμβολής y_1 σε λίστα **ListOfPoints** με επτά σημεία:

```
In[6]:= ListOfPoints = {{0, 3}, {1, 2}, {2, 7}, {3, 6}, {4, 5}, {5, 8}, {6, 10}};

In[7]:= y1 = Interpolation[ListOfPoints]

Out[7]= InterpolatingFunction[{{0, 6}}, <>]
```

Προέκυψε μια συνάρτηση παρεμβολής y_1 σε μορφή καθαρής συναρτήσεως. Μπορούμε όμως εύκολα να πάρουμε τις αριθμητικές τιμές της συναρτήσεως αυτής y_1 απλά γράφοντας **y1[x]** για το σημείο **x** για το οποίο ενδιαφερόμαστε, π.χ. εδώ για τα σημεία 0, 0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5, 5.5 και 6:

```
In[8]:= Table[{x, y1[x]}, {x, 0, 6, 0.5}]

Out[8]= {{0, 3}, {0.5, 1.}, {1., 2.}, {1.5, 4.5}, {2., 7.}, {2.5, 6.875}, {3., 6.},
         {3.5, 5.25}, {4., 5.}, {4.5, 6.3125}, {5., 8.}, {5.5, 9.4375}, {6., 10.}}
```

Φυσικά, αφού κάναμε παρεμβολή στα σημεία 0, 1, 2, 3, 4, 5 και 6, παίρνουμε εκεί ακριβώς τις τιμές εκείνες τις οποίες είχαμε δώσει για τη δημιουργία αυτής της συναρτήσεως παρεμβολής. Να τώρα και η γραφική παράσταση της ίδιας συναρτήσεως σ' ολόκληρο το διάστημα [0, 6]:

```
In[9]:= Plot[y1[x], {x, 0, 6}, PlotStyle → Thickness[0.009],
          AxesLabel → {"x", "y"}, PlotLabel -> "    ΣΥΝΑΡΤΗΣΗ ΠΑΡΕΜΒΟΛΗΣ",
          DefaultFont → {"Arial-Bold", 11}, ImageSize → 230];
```

Σημειώνουμε ότι γενικά η εντολή **Interpolation** χρησιμοποιεί πολυώνυμα τρίτου βαθμού για το σχηματισμό της σχετικής συναρτήσεως και σίγουρα δε χρησιμοποιεί ένα ενιαίο πολυώνυμο για την παρεμβολή σε όλα τα σημεία. (Αυτό όμως το κάνει η επόμενη εντολή **InterpolatingPolynomial**.) Σημειώνουμε τέλος ότι η αριθμητική λύση μιας συνήθους διαφορικής εξισώσεως με την αριθμητική εντολή **NDSolve**, που θα την αναφέρουμε στο αμέσως επόμενο notebook, δίνει το τελικό αποτέλεσμά της, την αριθμητική λύση της διαφορικής εξισώσεως, σε μορφή συναρτήσεως παρεμβολής **InterpolatingFunction**, ακριβώς όπως κάνει και η παρούσα εντολή **Interpolation**. (Γι' αυτόν ακριβώς το λόγο προηγήθηκε το παρόν notebook.)

■ ΕΝΤΟΛΗ N5: ΠΟΛΥΩΝΥΜΟ ΠΑΡΕΜΒΟΛΗΣ

InterpolatingPolynomial[*ΛίσταΣημείωνΠαρεμβολής, Μεταβλητή*]

Η εντολή αυτή δημιουργεί το πολυώνυμο παρεμβολής που βασίζεται στη λίστα των σημείων που δίνονται στο πρώτο όρισμά της με τη μεταβλητή να καθορίζεται στο δεύτερο όρισμά της. Η λίστα αυτή αποτελείται από τις n επιμέρους λίστες με δύο στοιχεία η καθεμιά τους, π.χ. **{x[k], y[k]}**, οι οποίες καθορίζουν τα n σημεία παρεμβολής. Αντίθετα με την προηγούμενη εντολή **Interpolation** το αποτέλεσμα στο παρόν πολυώνυμο παρεμβολής δίνεται στη μορφή συνηθισμένης συναρτήσεως. Επειδή μάλιστα τα σημεία παρεμβολής είναι n, το πολυώνυμο παρεμβολής που θα προκύψει θα είναι $n-1$ βαθμού. Παράδειγμα για το πολυώνυμο παρεμβολής της ίδιας λίστας **ListOfPoints** με επτά σημεία ($n = 7$), ακριβώς όπως και στην προηγούμενη εντολή **Interpolation**:

```
In[10]:= y2[x_] = InterpolatingPolynomial[ListOfPoints, x]
```

Out[10]= $3+\left(-1+\left(3+\left(-2+\left(\frac{3}{4}+\left(-\frac{1}{6}+\frac{13}{720}(-5+x)\right)(-4+x)\right)(-3+x)\right)(-2+x)\right)(-1+x)\right)x$

Φυσικά από οπτικής απόψεως είναι πολύ καλύτερη η γραφή του πολυωνύμου στη συνήθη μορφή του. Αυτή προφανώς είναι εδώ ένα πολυώνυμο έκτου βαθμού, απλά επειδή είχαμε επτά σημεία παρεμβολής:

```
In[11]:= y2[x_] = y2[x] // Expand
```

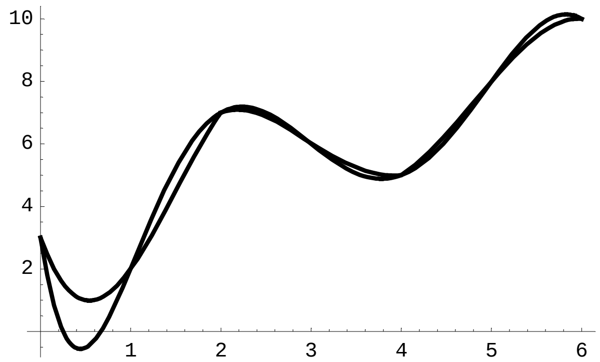
Out[11]= $3-\frac{56\,x}{3}+\frac{10991\,x^2}{360}-\frac{787\,x^3}{48}+\frac{569\,x^4}{144}-\frac{7\,x^5}{16}+\frac{13\,x^6}{720}$

Συγκρίνουμε τώρα τις αριθμητικές τιμές της συναρτήσεως παρεμβολής που βρέθηκε στην προηγούμενη εντολή με το πολυώνυμο παρεμβολής που υπολογίσθηκε στην παρούσα εντολή. Παρατηρούμε ότι ενώ βέβαια στα σημεία παρεμβολής τα αποτελέσματα συμπίπτουν (αφού και οι δύο αυτές συναρτήσεις από τον ορισμό τους περνούν από τα σημεία αυτά), γενικά δεν ισχύει το ίδιο σε άλλα σημεία, π.χ. εδώ στα σημεία 0.5, 1.5, 2.5, 3.5, 4.5 και 5.5:

```
In[12]:= Table[{x, y1[x], y2[x], y1[x] - y2[x]}, {x, 0, 6, 0.5}] // Chop

Out[12]= {{0, 3, 3, 0}, {0.5, 1., -0.516602, 1.5166},
          {1., 2., 2., 0}, {1.5, 4.5, 5.24512, -0.745117}, {2., 7., 7., 0},
          {2.5, 6.875, 6.99902, -0.124023}, {3., 6., 6., 0}, {3.5, 5.25, 5.05762, 0.192383},
          {4., 5., 5., 0}, {4.5, 6.3125, 6.1084, 0.204102}, {5., 8., 8., 0},
          {5.5, 9.4375, 9.71387, -0.276367}, {6., 10., 10., 0}}
```

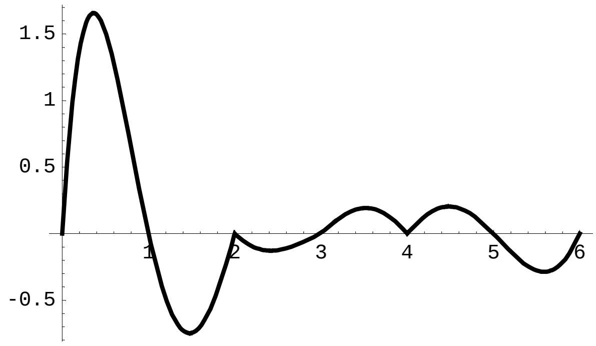
Οι διαφορές $y_1(x) - y_2(x)$ είναι προφανώς μηδενικές στα σημεία παρεμβολής 0, 1, 2, 3, 4, 5 και 6. Δεν είναι όμως μηδενικές στα σημεία 0.5, 1.5, 2.5, 3.5, 4.5 και 5.5, επειδή τα πολυώνυμα που ορίζουν τις y_1 και y_2 είναι διαφορετικά. Να τώρα και οι γραφικές παραστάσεις των δύο αυτών συναρτήσεων στο ίδιο σχήμα:

```
In[13]:= Plot[{y1[x], y2[x]}, {x, 0, 6}, PlotStyle → Thickness[0.008], PlotPoints → 40];
```



Επίσης και η γραφική παράσταση της διαφοράς τους $y_1 - y_2$:

```
In[14]:= Plot[y1[x] - y2[x], {x, 0, 6}, PlotStyle → {Thickness[0.008]}, PlotRange → All];
```



Οι γωνίες στη διαφορά αυτή (και στη συνάρτηση y_1 γενικότερα) οφείλονται απλά στο ότι η συνάρτηση παρεμβολής αποτελείται από διαφορετικά τριτοβάθμια πολυώνυμα και όχι από ένα ενιαίο πολυώνυμο, όπως είναι το πολυώνυμο παρεμβολής. Υπολογιστικά αυτό έχει και τα θετικά και τα αρνητικά του σημεία.

■ ΕΝΤΟΛΗ Ν6: ΠΡΟΣΕΓΓΙΣΗ MINIMAX

MiniMaxApproximation[*Συνάρτηση, {Μεταβλητή, {ΑρχικόΣημείο, ΤελικόΣημείο},*

***ΒαθμόςΠολυωνύμουΑριθμητή, ΒαθμόςΠολυωνύμουΠαρονομαστή*}]**

Η εντολή αυτή υπολογίζει την προσέγγιση minimax της συναρτήσεως η οποία δηλώνεται στο πρώτο όρισμά της υπό τη μορφή ρητής συναρτήσεως. Η μεταβλητή της συναρτήσεως, το διάστημα της προσεγγίσεως και οι βαθμοί των πολυωνύμων του αριθμητή και του παρονομαστή στη ρητή συνάρτηση της προσεγγίσεως minimax δηλώνονται στο δεύτερο όρισμα της εντολής. Εμείς εδώ θα χρησιμοποιήσουμε μόνο πολυωνυμικές αντί για ρητές προσεγγίσεις. Κατά συνέπεια ο βαθμός του πολυωνύμου του παρονομαστή της ρητής συναρτήσεως θα είναι μηδέν. Στο σημείο αυτό θα πρέπει να σημειώσουμε ότι η προσέγγιση minimax μιας συναρτήσεως είναι εκείνη που ελαχιστοποιεί το μέγιστο σφάλμα της προσεγγίσεως σε όλο το διάστημα που μας ενδιαφέρει. (Γι' αυτό και καλείται προσέγγιση minimax: το mini προέρχεται από την ελαχιστοποίηση και το max από το μέγιστο σφάλμα. Πρόκειται για μια πάρα πολύ ενδιαφέρουσα και την πιο ακριβή πολυωνυμική προσέγγιση.) Σημειώνουμε επίσης ότι στην προσέγγιση minimax αρκετές φορές αναφερόμαστε στο απόλυτο σφάλμα: τη διαφορά της συναρτήσεως που προσεγγίζεται μείον την προσέγγισή της. Εδώ όμως η *Mathematica* χρησιμοποιεί το αντίστοιχο σχετικό σφάλμα, κάτι που είναι και σωστότερο από απόψεως συνολικού αριθμού σημαντικών ψηφίων στην προσέγγιση minimax. Αναφέρουμε τέλος ότι η παρούσα εντολή **MiniMaxApproximation** είναι μια εντολή του πακέτου **NumericalMath‘Approximations‘**, το οποίο θα πρέπει βέβαια να έχει κληθεί (φορτωθεί) πριν από τη χρήση της εντολής **MiniMaxApproximation**. Πρώτα λοιπόν η κλήση του πακέτου:

```
In[15]:= Needs["NumericalMath`Approximations`"]
```

Αρχίζουμε με τη συνάρτηση σφάλματος (error function) erf *x* που την προσεγγίζουμε εδώ με πολυώνυμο minimax έκτου βαθμού στο διάστημα [1, 10]. Αυτό γίνεται απλά με την εντολή

```
In[16]:= MiniMaxErf = MiniMaxApproximation[Erf[x], {x, {1, 10}, 6, 0}]

Out[16]= {{1.000000000000000, 1.338065302154245, 2.285279345515466, 3.845303687904672,
           5.900197028385772, 7.950541368563328, 9.45089255707086, 10.00000000000000},
          {0.1534464645861447 + 1.188635468760697 x - 0.6429982570513566 x^2 +
            0.1723726570352321 x^3 - 0.02434568990317125 x^4 + 0.001732344134586652 x^5 -
            0.00004889737758623442 x^6, -0.007230676992129966}}
```

Στο αποτέλεσμα αυτό δίνεται πρώτα η λίστα των σημείων του διαστήματος [1, 10] όπου παρουσιάζεται το μέγιστο (κατ' απόλυτο τιμή) σχετικό σφάλμα της παρούσας προσεγγίσεως minimax. Στη συνέχεια δίνεται το σχετικό πολυώνυμο έκτου βαθμού (έχουμε ήδη καθορίσει το βαθμό: 6) και τέλος δίνεται και το αντίστοιχο ελάχιστο μέγιστο (minimax) σφάλμα. Από τα αποτελέσματα αυτά εμείς συχνά θέλουμε και εύκολα αποσπούμε από την πιο πάνω λίστα το ίδιο το πολυώνυμο minimax. Αυτό έχει εδώ τη μορφή:

```
In[17]:= erf[x_] = MiniMaxErf[[2, 1]]

Out[17]= 0.1534464645861447 + 1.188635468760697 x -
          0.6429982570513566 x^2 + 0.1723726570352321 x^3 - 0.02434568990317125 x^4 +
          0.001732344134586652 x^5 - 0.00004889737758623442 x^6
```

που τη δηλώνουμε με **erf** αντί **Erf**.Το μέγιστο σχετικό σφάλμα (που ελαχιστοποιήθηκε εδώ) έχει την τιμή

```
In[18]:= MaximumRelativeErrorErf = MiniMaxErf[[2, 2]]

Out[18]= -0.007230676992129966
```

Σαν δεύτερο παράδειγμα εξετάζουμε την προσέγγιση minimax της υπερβολικής συναρτήσεως cosh *x* τώρα με πολυώνυμο πέμπτου βαθμού και στο διάστημα [0, 3]. Με την ίδια εντολή προκύπτει

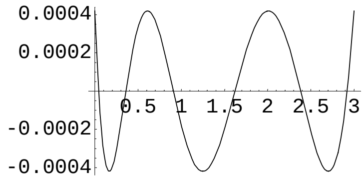
```
In[19]:= MiniMaxCosh = MiniMaxApproximation[Cosh[x], {x, {0, 3}, 5, 0}]

Out[19]= {{0, 0.1662185796142901, 0.6092114557497472, 1.249129439570007,
           2.010056077576799, 2.703283620756527, 3.000000000000000},
          {0.999581471947963 + 0.01194950238456329 x + 0.4458556265873798 x^2 +
            0.08601654627416872 x^3 - 0.01669306777373992 x^4 +
            0.01664612970656959 x^5, 0.0004185280520371935}}
```

με σχετική minimax προσέγγιση **cosh** της συναρτήσεως **Cosh** την ακόλουθη προσέγγιση:

```
In[20]:= cosh[x_] = MiniMaxCosh[[2, 1]]

Out[20]= 0.999581471947963 + 0.01194950238456329 x + 0.4458556265873798 x^2 +
          0.08601654627416872 x^3 - 0.01669306777373992 x^4 + 0.01664612970656959 x^5
```

Υπενθυμίζεται πως η *Mathematica* χρησιμοποιεί το σχετικό σφάλμα (δηλαδή εδώ με διαίρεση του απόλυτου σφάλματος δια cosh x) στην προσέγγιση minimax και όχι το απόλυτο σφάλμα. Για το λόγο αυτό και εμείς παρουσιάζουμε το σχετικό σφάλμα της παραπάνω προσεγγίσεως minimax στο επόμενο σχήμα:

```
In[21]:= Plot[(Cosh[x] - cosh[x]) / Cosh[x], {x, 0, 3}, ImageSize → 175];
```



Από το σχήμα αυτό φαίνεται καθαρά πως το σχετικό αυτό σφάλμα είναι το ίδιο κατ' απόλυτο τιμή σε όλα τα σημεία του διαστήματος [0, 3] που εξετάζουμε, όποτε παίρνει τοπικά μέγιστες και ελάχιστες τιμές. Εύκολα μπορούμε να προσδιορίσουμε όλα τα πολυώνυμα minimax που θέλουμε, π.χ. βαθμού μέχρι και 12:

```
In[22]:= Table[MMApproxCosh[n] = MiniMaxApproximation[Cosh[x], {x, {0, 3}, n, 0}], {n, 1, 12}];
```

Αν και αποκλειστικά για οικονομία χώρου δεν παρουσιάζουμε τούτα τα δώδεκα πολυώνυμα, εντούτοις παρουσιάζουμε τα μέγιστα σχετικά σφάλματα σ' αυτές τις προσεγγίσεις minimax:

```
In[23]:= Table[MMApproxCosh[n][[2, 2]], {n, 1, 12}]
```

```
Out[23]= {0.4455180001261270, -0.1027389239990284, 0.02218154579116696,
  -0.002942889616220822, 0.0004185280520371935, -0.00003985284899400468,
  4.223468603332318×10^-6, -3.137655894760941×10^-7, 2.648647290904025×10^-8,
  -1.613463053341952×10^-9, 1.131574563150643×10^-10, -5.842200400317100×10^-12}
```

Είναι προφανές ότι αυτά μειώνονται με την αύξηση του βαθμού n του πολυωνύμου της προσεγγίσεως minimax της συναρτήσεως cosh x. Να και το πολυώνυμο όγδοου βαθμού, το οποίο έχει ήδη υπολογισθεί:

```
In[24]:= CoshMiniMax8[x_] = MMApproxCosh[8][[2, 1]]
```

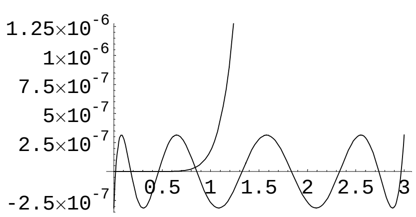
$$\text{Out[24]= } 1.000000313765589 - 0.00001907314178603718\,x + 0.5001894817300582\,x^2 - 0.0007131107047451936\,x^3 + 0.04298264158887492\,x^4 - 0.001327413022250424\,x^5 + 0.002145263670290803\,x^6 - 0.0002341161382805281\,x^7 + 0.00005705632837901748\,x^8$$

Φυσικά η πολυωνυμική προσέγγιση minimax μιας συναρτήσεως είναι πολύ καλύτερη από την αντίστοιχη προσέγγιση με σειρά Taylor (ή Maclaurin για $x = 0$) για τον ίδιο βαθμό πολυωνύμου n. Αυτό φαίνεται πολύ καθαρά παρακάτω με το σχετικό σφάλμα της προσεγγίσεως Maclaurin να εκτινάσσεται προς τα επάνω:

```
In[25]:= CoshMaclaurin8[x_] = Series[Cosh[x], {x, 0, 8}] // Normal // N
```

$$\text{Out[25]= } 1. + 0.5\,x^2 + 0.0416667\,x^4 + 0.00138889\,x^6 + 0.0000248016\,x^8$$

```
In[26]:= Plot[{1 - CoshMiniMax8[x] / Cosh[x], 1 - CoshMaclaurin8[x] / Cosh[x]}, {x, 0, 3}];
```



■ *Notebook E15*
ΕΝΤΟΛΕΣ ΓΙΑ ΤΗΝ ΕΠΙΛΥΣΗ ΔΙΑΦΟΡΙΚΩΝ ΕΞΙΣΩΣΕΩΝ
3 ΕΝΤΟΛΕΣ: D1. DSolve, D2. NDSolve, D3. PlotVectorField

■ ΕΝΤΟΛΗ D1: ΕΠΙΛΥΣΗ ΔΙΑΦΟΡΙΚΩΝ ΕΞΙΣΩΣΕΩΝ

DSolve[*ΣυνήθηςΔιαφορικήΕξίσωση, ΆγνωστηΣυνάρτηση, ΑνεξάρτητηΜεταβλητή*]

DSolve[{*ΣυνήθηςΔιαφορικήΕξίσωση, ΑρχικέςΉΣυνοριακέςΣυνθήκες*}, *Άγνωστη Συνάρτηση, ΑνεξάρτητηΜεταβλητή*]

DSolve[*ΛίσταΣυνήθωνΔιαφορικώνΕξισώσεων, ΛίσταΑγνώστωνΣυναρτήσεων, ΑνεξάρτητηΜεταβλητή*]

DSolve[*ΛίσταΣυνήθωνΔιαφορικώνΕξισώσεωνΚαιΑρχικώνΉΣυνοριακώνΣυνθηκών, ΛίσταΑγνώστωνΣυναρτήσεων, ΑνεξάρτητηΜεταβλητή*]

DSolve[*ΔιαφορικήΕξίσωσηΜεΜερικέςΠαραγώγους, ΆγνωστηΣυνάρτηση, ΛίσταΑνεξάρτητωνΜεταβλητών*]

DSolve[*ΣυνήθηςΔιαφορικήΕξίσωση, ΆγνωστηΣυνάρτηση, ΑνεξάρτητηΜεταβλητή*, DSolveConstants → *ΝέοΣύμβολοΓιαΤιςΑυθαίρετεςΣταθερές*]

Λύνει τη συνήθη διαφορική εξίσωση ή το σύστημα συνήθων διαφορικών εξισώσεων ή τη διαφορική εξίσωση με μερικές παραγώγους η οποία δίνεται στο πρώτο όρισμά της μαζί με τις σχετικές αρχικές ή συνοριακές συνθήκες, εφόσον υπάρχουν. (Υπενθυμίζεται ότι όλες οι εξισώσεις: αριθμητικές, αλγεβρικές, τριγωνομετρικές, λογικές, διαφορικές, ολοκληρωτικές, κλπ. πρέπει πάντοτε να γράφονται με διπλό ίσον == και όχι με απλό ίσον = . Στη *Mathematica* το απλό ίσον = δηλώνει καθορισμό τιμής σε σύμβολο, αλλ' όχι, με τίποτα εξίσωση.) Το αποτέλεσμα είναι η λύση της διαφορικής εξισώσεως σε μορφή κανόνα αντικαταστάσεως ή κανόνων αντικαταστάσεων, όχι με ίσον. Η χρήση της επιλογής **DSolveConstants** (που επιτρέπεται σε κάθε περίπτωση) μας επιτρέπει την αλλαγή του συμβόλου *C* για τις αυθαίρετες σταθερές (και στην περίπτωση διαφορικών εξισώσεων με μερικές παραγώγους για τις αυθαίρετες συναρτήσεις) σε κάποιο άλλο πιο κατάλληλο, πιο βολικό σύμβολο.

■ ΕΠΙΛΥΣΗ ΣΥΝΗΘΩΝ ΔΙΑΦΟΡΙΚΩΝ ΕΞΙΣΩΣΕΩΝ

Αρκεί να καλέσουμε την εντολή **DSolve** δίνοντάς της σαν πρώτο όρισμα τη διαφορική εξίσωση που θέλουμε να λύσουμε, σαν δεύτερο όρισμα την άγνωστη συνάρτηση (σαν συνάρτηση της ανεξάρτητης μεταβλητής) και σαν τρίτο όρισμα την ίδια την ανεξάρτητη μεταβλητή. Φυσικά στην παράγραφο αυτή χωρίς αρχικές ή συνοριακές συνθήκες παίρνουμε απλά τη γενική λύση της διαφορικής εξισώσεως. Αυτή περιλαμβάνει αριθμό αυθαίρετων σταθερών ίσο με την τάξη της διαφορικής εξισώσεως. Παραδείγματα:

```
In[1]:= DSolve[y'[x] == a y[x], y[x], x]
```

```
Out[1]= {{y[x] → ⅇ^(a x) C[1]}}
```

```
In[2]:= DSolve[a y''[x] + b y'[x] + c y[x] == 0, y[x], x]
```

Out[2]= $\left\{\left\{y[x] \to e^{\frac{\left(-b-\sqrt{b^2-4 a c}\right) x}{2 a}} C[1] + e^{\frac{\left(-b+\sqrt{b^2-4 a c}\right) x}{2 a}} C[2]\right\}\right\}$

```
In[3]:= DSolve[y'[x] - x y[x] == 1, y[x], x]
```

Out[3]= $\left\{\left\{y[x] \to e^{\frac{x^2}{2}} C[1] + e^{\frac{x^2}{2}} \sqrt{\frac{\pi}{2}}\, \mathrm{Erf}\left[\frac{x}{\sqrt{2}}\right]\right\}\right\}$

```
In[4]:= DSolve[y'[x]^2 + y[x]^2 == 10, y[x], x]
```

Out[4]= $\left\{\left\{y[x] \to -\sqrt{10}\, \mathrm{Sin}[x - C[1]]\right\}, \left\{y[x] \to \sqrt{10}\, \mathrm{Sin}[x + C[1]]\right\}\right\}$

Φυσικά η διαφορική εξίσωση μπορεί να έχει ορισθεί από πριν έξω από την εντολή **DSolve**, π.χ. με το σύμβολο **de** ή **deq**, που μπορεί να ερμηνευθεί σαν διαφορική εξίσωση (differential equation):

```
In[5]:= {de = y''[x] + 2 y'[x] - y[x] == e^(5 x), sol = DSolve[de, y[x], x]}
```

Out[5]= $\left\{-y[x] + 2 y'[x] + y''[x] == e^{5 x},\right.$
$\left\{\left\{y[x] \to \left(e^{-\left(-6+\sqrt{2}\right) x} \left(-e^{\left(-1+\sqrt{2}\right) x} - 3\sqrt{2}\, e^{\left(-1+\sqrt{2}\right) x} - e^{\left(-1-\sqrt{2}\right) x + \left(-6+\sqrt{2}\right) x + \left(6+\sqrt{2}\right) x} + 3\sqrt{2}\, e^{\left(-1-\sqrt{2}\right) x + \left(-6+\sqrt{2}\right) x + \left(6+\sqrt{2}\right) x}\right)\right) \middle/ \left(2\left(-6+\sqrt{2}\right)\left(6+\sqrt{2}\right)\right) + e^{\left(-1-\sqrt{2}\right) x} C[1] + e^{\left(-1+\sqrt{2}\right) x} C[2]\right\}\right\}\right\}$

Πολύ συχνά η απλοποίηση της λύσεως που βρέθηκε με την εντολή **Simplify** ή, αν χρειάζεται, με την εντολή **FullSimplify**, είναι χρήσιμη:

```
In[6]:= sol1 = sol // Simplify
```

Out[6]= $\left\{\left\{y[x] \to \frac{e^{5 x}}{34} + e^{-\left(1+\sqrt{2}\right) x} C[1] + e^{\left(-1+\sqrt{2}\right) x} C[2]\right\}\right\}$

Αυτή η απλοποίηση μπορεί να γίνει ακόμη καλύτερα κατευθείαν κατά την ώρα της επιλύσεως της διαφορικής εξισώσεως

```
In[7]:= {de = y''[x] + 2 y'[x] - y[x] == e^(5 x), sol2 = DSolve[de, y[x], x] // Simplify, sol2 == sol1}
```

Out[7]= $\left\{-y[x] + 2 y'[x] + y''[x] == e^{5 x}, \left\{\left\{y[x] \to \frac{e^{5 x}}{34} + e^{-\left(1+\sqrt{2}\right) x} C[1] + e^{\left(-1+\sqrt{2}\right) x} C[2]\right\}\right\}, \mathrm{True}\right\}$

Είναι προφανές ότι η γενική λύση της διαφορικής εξισώσεως που προκύπτει θα πρέπει να περιέχει έναν αριθμό αυθαίρετων σταθερών ίσο με την τάξη της. Αυτό φαίνεται στα πιο πάνω παραδείγματα επίσης και στα ακόλουθα παραδείγματα.

```
In[8]:= DSolve[y''''[x] + 2 y''[x] + y[x] == 1, y[x], x]
```

```
Out[8]= {{y[x] → 1 + C[1] Cos[x] + x C[2] Cos[x] + C[3] Sin[x] + x C[4] Sin[x]}}
```

```
In[9]:= DSolve[y''[x] + x y'[x] + x^2 y[x] == 0, y[x], x] // Simplify
```

Out[9]= $\left\{\left\{y[x] \to e^{-\frac{1}{4} i \left(-i+\sqrt{3}\right) x^2} \left(C[1]\, \mathrm{HermiteH}\left[\frac{1}{6} i \left(3 i + \sqrt{3}\right), \left(\frac{1}{2} + \frac{i}{2}\right) 3^{1/4} x\right] + C[2]\, \mathrm{Hypergeometric1F1}\left[\frac{1}{4} - \frac{i}{4\sqrt{3}}, \frac{1}{2}, \frac{1}{2} i \sqrt{3}\, x^2\right]\right)\right\}\right\}$

Το επόμενο παράδειγμα έχει σχέση με τη χωρική διαφορική εξίσωση που προκύπτει στις καμπτικές ιδιοταλαντώσεις συνήθους δοκού. Πρόκειται για γραμμική συνήθη διαφορική εξίσωση τετάρτης τάξεως:

```
In[10]:= DSolve[v''''[x] - β^4 v[x] == 0, v[x], x]

Out[10]= {{v[x] → e^(-x β) C[2] + e^(x β) C[4] + C[1] Cos[x β] + C[3] Sin[x β]}}
```

Παρατηρούμε πως αρκετά συχνά οι λύσεις συνήθων διαφορικών εξισώσεων περιέχουν και ειδικές συναρτήσεις και όχι μόνο αλγεβρικές παραστάσεις ή στοιχειώδεις υπερβατικές συναρτήσεις, όπως είναι η εκθετική συνάρτηση, η λογαριθμική συνάρτηση, οι υπερβολικές και οι τριγωνομετρικές συναρτήσεις και οι αντίστροφές τους υπερβολικές και τριγωνομετρικές συναρτήσεις.

Η *Mathematica* χρησιμοποιεί καταρχήν το σύμβολο *C* για να δηλώνει τις αυθαίρετες σταθερές στις λύσεις συνήθων διαφορικών εξισώσεων. Το ίδιο σύμβολο το χρησιμοποιεί, όπως θα δούμε παρακάτω, για να δηλώνει και τις αυθαίρετες συναρτήσεις στις λύσεις διαφορικών εξισώσεων με μερικές παραγώγους που διαθέτουν κλειστές λύσεις. Κάτι τέτοιο μπορεί να μη μας αρέσει και να προτιμάμε κάποιο άλλο σύμβολο. Ή να θέλουμε να χρησιμοποιήσουμε το σύμβολο *C* για κάποιον άλλο πιο σημαντικό σκοπό. Προς το σκοπό αυτό χρησιμοποιούμε την επιλογή **DSolveConstants** καθορίζοντας εμείς το σύμβολο το οποίο θα χρησιμοποιηθεί για τις αυθαίρετες σταθερές ή τις αυθαίρετες συναρτήσεις. Παραδείγματα:

```
In[11]:= solution1 = DSolve[v''''[x] - β^4 v[x] == 0, v[x], x]

Out[11]= {{v[x] → e^(-x β) C[2] + e^(x β) C[4] + C[1] Cos[x β] + C[3] Sin[x β]}}

In[12]:= solution2 = DSolve[v''''[x] - β^4 v[x] == 0, v[x], x, DSolveConstants → c]

Out[12]= {{v[x] → e^(-x β) c[2] + e^(x β) c[4] + c[1] Cos[x β] + c[3] Sin[x β]}}

In[13]:= solution3 = DSolve[v''''[x] - β^4 v[x] == 0, v[x], x, DSolveConstants → A]

Out[13]= {{v[x] → e^(-x β) A[2] + e^(x β) A[4] + A[1] Cos[x β] + A[3] Sin[x β]}}

In[14]:= {solution1 == solution3, solution1 == solution3 /. A → C}

Out[14]= {{{v[x] → e^(-x β) C[2] + e^(x β) C[4] + C[1] Cos[x β] + C[3] Sin[x β]}} ==
           {{v[x] → e^(-x β) A[2] + e^(x β) A[4] + A[1] Cos[x β] + A[3] Sin[x β]}}, True}
```

Από τη λύση που δίνει η εντολή **DSolve** (σε μορφή κανόνα ή κανόνων αντικαταστάσεως) μπορούμε εύκολα να ορίσουμε τη συνάρτηση που εκφράζει τη λύση αυτή παίρνοντας το κατάλληλο μέρος της λύσεως, συνήθως το ⟦1, 1, 2⟧, έτσι ώστε να φύγουν τα εξωτερικά άγκιστρα και επίσης να πάρουμε το δεξιό μέλος της καθαυτό λύσεως. Ένα πρώτο παράδειγμα από τη Δυναμική των Κατασκευών: Δοκοί

```
In[15]:= vs[x_] = solution1⟦1, 1, 2⟧

Out[15]= e^(-x β) C[2] + e^(x β) C[4] + C[1] Cos[x β] + C[3] Sin[x β]
```

Το δεύτερο πιο κάτω παράδειγμα προέρχεται από τις Ταλαντώσεις. Συγκεκριμένα αφορά στη μετατόπιση του υλικού σημείου (μάζας m) σε μονοβάθμιο μηχανικό σύστημα υλικού σημείου–ελατηρίου–αποσβεστήρα υπό σταθερή φόρτιση (εξωτερική δύναμη) p_0. Δίνουμε πρώτα τη γνωστή μας σχετική διαφορική εξίσωση και τη λύνουμε. Στη συνέχεια μπορούμε να ορίσουμε και τη σχετική συνάρτηση:

```
In[16]:= vde = u''[t] + 2 ξ ω0 u'[t] + ω0^2 u[t] == p0 / m; solution = DSolve[vde, u[t], t]
```

$$\text{Out[16]=}\ \left\{\left\{u[t] \to e^{t\left(-\xi\,\omega_0-\sqrt{-1+\xi^2}\,\omega_0\right)}\,C[1] + e^{t\left(-\xi\,\omega_0+\sqrt{-1+\xi^2}\,\omega_0\right)}\,C[2] + \frac{p_0}{m\,\omega_0^2}\right\}\right\}$$

```
In[17]:= us1[t_] = solution⟦1, 1, 2⟧
```

$$\text{Out[17]=}\ e^{t\left(-\xi\,\omega_0-\sqrt{-1+\xi^2}\,\omega_0\right)}\,C[1] + e^{t\left(-\xi\,\omega_0+\sqrt{-1+\xi^2}\,\omega_0\right)}\,C[2] + \frac{p_0}{m\,\omega_0^2}$$

Μια εναλλακτική δυνατότητα προσδιορισμού της λύσεως σε μορφή συναρτήσεως αποτελεί το να δώσουμε μία μόνο εντολή που και να λύνει τη διαφορική εξίσωση (με την εντολή **DSolve**) και αμέσως μετά να καθορίζει τη λύση σαν συνάρτηση συνήθως με την επιλογή του μέρους ⟦1, 1, 2⟧. Παράδειγμα:

```
In[18]:= us2[t_] = DSolve[vde, u[t], t][[1, 1, 2]]
```

$$\text{Out[18]=}\quad e^{t\left(-\xi\omega_0-\sqrt{-1+\xi^2}\,\omega_0\right)} C[1] + e^{t\left(-\xi\omega_0+\sqrt{-1+\xi^2}\,\omega_0\right)} C[2] + \frac{p_0}{m\,\omega_0^2}$$

Τέλος μια πολύ ενδιαφέρουσα τρίτη δυνατότητα προσδιορισμού της λύσεως σε μορφή συναρτήσεως είναι να χρησιμοποιήσουμε κατευθείαν τη λύση στη μορφή κανόνα αντικαταστάσεως, δηλαδή ακριβώς όπως τη δίνει η *Mathematica*

```
In[19]:= {vde, solution}
```

$$\text{Out[19]=}\quad \left\{\omega_0^2\, u[t] + 2\,\xi\,\omega_0\, u'[t] + u''[t] == \frac{p_0}{m},\right.$$
$$\left.\left\{\left\{u[t] \to e^{t\left(-\xi\omega_0-\sqrt{-1+\xi^2}\,\omega_0\right)} C[1] + e^{t\left(-\xi\omega_0+\sqrt{-1+\xi^2}\,\omega_0\right)} C[2] + \frac{p_0}{m\,\omega_0^2}\right\}\right\}\right\}$$

για τον ορισμό στη συνέχεια της συναρτήσεως–λύσεως της διαφορικής εξισώσεως. (Πρόκειται για μια ιδιαίτερα διαδεδομένη μέθοδο ορισμού της λύσεως αυτής.)

```
In[20]:= {us3[t_] = u[t] /. solution[[1]], us1[t] == us2[t] == us3[t]}
```

$$\text{Out[20]=}\quad \left\{e^{t\left(-\xi\omega_0-\sqrt{-1+\xi^2}\,\omega_0\right)} C[1] + e^{t\left(-\xi\omega_0+\sqrt{-1+\xi^2}\,\omega_0\right)} C[2] + \frac{p_0}{m\,\omega_0^2},\ \text{True}\right\}$$

Παρατηρούμε βέβαια ότι και οι τρεις αυτές μέθοδοι έδωσαν ακριβώς την ίδια συνάρτηση $u_s(t)$ για τη λύση που βρέθηκε με την εντολή **DSolve**. (Κι αλίμονό μας αν δεν ήταν έτσι η κατάσταση ...)

Φυσικά δεν είναι εύκολη η επίλυση διαφορικών εξισώσεων και οι αναλυτικές λύσεις τους (όσες φορές υπάρχουν ...) είναι συχνά πολύπλοκες. Καλό είναι επομένως να επαληθεύουμε τις λύσεις τις οποίες βρίσκουμε. Συνεχίζουμε το προηγούμενο παράδειγμα στις Ταλαντώσεις. Η επαλήθευση γίνεται, αφού ήδη έχει προσδιορισθεί η συνάρτηση που αντιστοιχεί στη λύση της διαφορικής εξισώσεως. Και είναι γενικά απλή η επαλήθευση αυτή, αν και συνήθως απαιτεί και απλοποίηση, για παράδειγμα

```
In[21]:= {vde, verification = vde /. u → us1 // Simplify}
```

$$\text{Out[21]=}\quad \left\{\omega_0^2\, u[t] + 2\,\xi\,\omega_0\, u'[t] + u''[t] == \frac{p_0}{m},\ \text{True}\right\}$$

Αυτό που πρέπει να τονισθεί έντονα στη *Mathematica* είναι ότι κατά την επαλήθευση λύσεων διαφορικών εξισώσεων (είτε συνήθων είτε με μερικές παραγώγους) στο σχετικό κανόνα αντικαταστάσεως, π.χ. στον **u → us1** πιο πάνω, θα πρέπει και οι δύο συναρτήσεις που εμφανίζονται να είναι σε "καθαρή" μορφή, δηλαδή χωρίς να αναφέρονται οι μεταβλητές τους. Αλλιώς δε θα γίνει δυνατή η αντικατάσταση με τον κανόνα αυτό των παραγώγων της άγνωστης (και τώρα πια γνωστής) συναρτήσεως στη διαφορική εξίσωση και έτσι η επαλήθευση που επιχειρείται θα αποτύχει (και αδικαιολόγητα μάλιστα, αφού η λύση της διαφορικής εξισώσεως έχει ήδη βρεθεί).

Πολύ συχνά προτιμάμε τη λύση μιας διαφορικής εξισώσεως που περιέχει την εκθετική συνάρτηση σε ισοδύναμη μορφή που να περιέχει αντί για την εκθετική συνάρτηση τις υπερβολικές συναρτήσεις συνημίτονο (cosh) και ημίτονο (sinh). Αυτό μπορούμε να το πετύχουμε εύκολα με τη χρήση της εντολής **ExpToTrig**, που μας είναι ήδη γνωστή (Εντολή E4:T1) στη λύση της διαφορικής εξισώσεως: η λύση να είναι γραμμένη υπό τη μορφή συναρτήσεως και όχι υπό τη μορφή κανόνα αντικαστάσεως. Παράδειγμα στη λύση $v_s(x)$ της χωρικής διαφορικής εξισώσεως των ιδιοταλαντώσεων συνήθους δοκού:

```
In[22]:= {ExpToTrig[vs[x]], vs[x] // ExpToTrig} // Simplify

Out[22]= {C[1] Cos[x β] + (C[2] + C[4]) Cosh[x β] + C[3] Sin[x β] + (-C[2] + C[4]) Sinh[x β],
          C[1] Cos[x β] + (C[2] + C[4]) Cosh[x β] + C[3] Sin[x β] + (-C[2] + C[4]) Sinh[x β]}
```

■ ΕΠΙΛΥΣΗ ΠΡΟΒΛΗΜΑΤΩΝ ΑΡΧΙΚΩΝ ΚΑΙ ΣΥΝΟΡΙΑΚΩΝ ΤΙΜΩΝ

Μέχρι τώρα κάναμε χρήση της εντολής **DSolve** για την επίλυση συνήθων διαφορικών εξισώσεων χωρίς αρχικές ή συνοριακές συνθήκες. Δηλαδή προσδιορίζαμε μόνο γενικές λύσεις διαφορικών εξισώσεων. Εδώ θα επιλύσουμε προβλήματα αρχικών και συνοριακών τιμών, δηλαδή διαφορικές εξισώσεις που συνοδεύονται από αρχικές συνθήκες (για τα προβλήματα αρχικών τιμών) και από συνοριακές συνθήκες (για τα προβλήματα συνοριακών τιμών). Προς το σκοπό αυτό οι συνθήκες θεωρούνται και αυτές εξισώσεις και μπαίνουν μαζί με την ίδια τη διαφορική εξίσωση στο πρώτο όρισμα της εντολής **DSolve**. Αυτό παίρνει έτσι τη μορφή μιας λίστας εξισώσεων. Ας δούμε τη σχετική διαδικασία σε δύο πολύ απλά παραδείγματα.

Θεωρούμε πρώτα τη διαφορική εξίσωση (differential equation) των ελεύθερων ταλαντώσεων του μονοβάθμιου μηχανικού συστήματος υλικού σημείου–ελατηρίου ή μάζας–ελατηρίου (εδώ χωρίς απόσβεση):

```
In[23]:= de = u''[t] + ω0^2 u[t] == 0

Out[23]= ω0^2 u[t] + u''[t] == 0
```

Αυτή η διαφορική εξίσωση **de** έχει γενική λύση τη

```
In[24]:= GeneralSolution = DSolve[de, u[t], t]

Out[24]= {{u[t] → C[1] Cos[t ω0] + C[2] Sin[t ω0]}}

In[25]:= ug[t_] = GeneralSolution[[1, 1, 2]]

Out[25]= C[1] Cos[t ω0] + C[2] Sin[t ω0]
```

Ωραία ως εδώ! Τώρα θεωρούμε τις ίδιες ελεύθερες ταλαντώσεις, αλλά με γνωστές τις δύο αρχικές συνθήκες (initial conditions), δηλαδή την αρχική θέση u_0 και την αρχική ταχύτητα v_0 του υλικού σημείου:

```
In[26]:= ics = {u[0] == u0, u'[0] == v0}

Out[26]= {u[0] == u0, u'[0] == v0}
```

Τώρα πια έχουμε ένα πρόβλημα αρχικών τιμών: μία διαφορική εξίσωση δευτέρας τάξεως μαζί με δύο αρχικές συνθήκες. Η εντολή **DSolve** μπορεί εύκολα να το λύσει, αρκεί στο πρώτο όρισμα να αναφέρουμε και τη διαφορική εξίσωση και τις δύο αρχικές συνθήκες που ισχύουν, π.χ. με την εντολή

```
In[27]:= PartialSolution = DSolve[{de, ics}, u[t], t] // Simplify

Out[27]= {{u[t] → u0 Cos[t ω0] + (v0 Sin[t ω0])/ω0}}

In[28]:= up[t_] = PartialSolution[[1, 1, 2]]

Out[28]= u0 Cos[t ω0] + (v0 Sin[t ω0])/ω0
```

Σημειώνουμε ότι στη *Mathematica* είναι πιο απλό από απόψεως εντολών να ξαναλύσουμε τη διαφορική εξίσωση (τώρα βέβαια μαζί με τις αρχικές συνθήκες της) παρά να προσδιορίσουμε τις δύο αυθαίρετες σταθερές στη γενική λύση $u_g(t)$, έτσι ώστε αυτή να πληροί και τις δύο αρχικές συνθήκες. Υπολογιστικά η δεύτερη αυτή δυνατότητα είναι πολύ πιο σωστή, αλλά πρακτικά (απλά για οικονομία σε εντολές) χρησιμοποιείται η πρώτη με επανάληψη της λύσεως της διαφορικής εξισώσεως.

Φυσικά η παραπάνω μερική λύση $u_p(t)$, που είναι η λύση του προβλήματος αρχικών τιμών που προαναφέραμε, επαληθεύει τόσο τη διαφορική εξίσωση όσο και τις δύο αρχικές συνθήκες, όπως πολύ εύκολα παρατηρούμε. (Εδώ ειδικά για τη διαφορική εξίσωση χρειάζεται και απλοποίηση του αποτελέσματος.)

```
In[29]:= verification = {de, ics} /. u → up // Simplify

Out[29]= {True, {True, True}}
```

Σαν δεύτερο και τελευταίο παράδειγμα θα επιλύσουμε την τόσο απλή διαφορική εξίσωση της συνήθους δοκού δυσκαμψίας *EI* υπό σταθερή κάθετη κατανεμημένη φόρτιση p_0 σε όλο το μήκος της

```
In[30]:= BeamDE = v''''[x] == p0 / EI;
```

Αυτή έχει ιδιαίτερα απλή γενική λύση, που είναι ένα πολυώνυμο τετάρτου βαθμού με τέσσερις σταθερές:

```
In[31]:= SolutionBeamDE = DSolve[BeamDE, v[x], x]
```

Out[31]= $\left\{\left\{v[x] \to C[1] + x\,C[2] + x^2\,C[3] + x^3\,C[4] + \frac{x^4\,p_0}{24\,EI}\right\}\right\}$

```
In[32]:= vg[x_] = SolutionBeamDE[[1, 1, 2]]
```

Out[32]= $C[1] + x\,C[2] + x^2\,C[3] + x^3\,C[4] + \frac{x^4\,p_0}{24\,EI}$

Ας θεωρήσουμε τώρα ότι η δοκός μας είναι μήκους *L* και αμφίπακτη, οπότε ισχύουν οι ακόλουθες τέσσερις συνοριακές συνθήκες (ανά δύο σε κάθε άκρο της):

```
In[33]:= BeamBoundaryConditions1 = {v[0] == 0, v'[0] == 0, v[L] == 0, v'[L] == 0};
```

Τότε η λύση του σχετικού προβλήματος συνοριακών τιμών, η οποία προφανώς παριστάνει το βέλος κάμψεως της δοκού (ή σχεδόν ισοδύναμα την ελαστική γραμμή που σχηματίζει η δοκός) θα έχει τη μορφή

```
In[34]:= BeamSolution1 = DSolve[{BeamDE, BeamBoundaryConditions1}, v[x], x] // Simplify
```

Out[34]= $\left\{\left\{v[x] \to \frac{(L-x)^2\,x^2\,p_0}{24\,EI}\right\}\right\}$

```
In[35]:= v1[x_] = BeamSolution1[[1, 1, 2]]
```

Out[35]= $\frac{(L-x)^2\,x^2\,p_0}{24\,EI}$

Αυτή φυσικά επαληθεύει όλες τις εξισώσεις: και τη διαφορική εξίσωση και τις συνοριακές συνθήκες:

```
In[36]:= BeamVerification1 = {BeamDE, BeamBoundaryConditions1} /. v → v1

Out[36]= {True, {True, True, True, True}}
```

Ας λύσουμε πολύ σύντομα και το αντίστοιχο πρόβλημα του προβόλου με πάκτωση δεξιά με εξισώσεις:

```
In[37]:= CantileverEquations = {BeamDE, v''[0] == 0, v'''[0] == 0, v[L] == 0, v'[L] == 0};
```

Η λύση όλων αυτών των εξισώσεων προκύπτει και πάλι εύκολα με την εντολή **DSolve** σε συνδυασμό με την εντολή **Simplify** ή με την εντολή **Factor** (ανάλογα με το πώς εμείς θέλουμε το τελικό αποτέλεσμα):

```
In[38]:= {v2[x_] = DSolve[CantileverEquations, v[x], x][[1, 1, 2]] // Simplify, v2[x] // Factor}
```

Out[38]= $\left\{\frac{(3 L^4 - 4 L^3 x + x^4)\, p_0}{24\, EI}, \frac{(L - x)^2 (3 L^2 + 2 L x + x^2)\, p_0}{24\, EI}\right\}$

■ ΕΠΙΛΥΣΗ ΣΥΣΤΗΜΑΤΩΝ ΣΥΝΗΘΩΝ ΔΙΑΦΟΡΙΚΩΝ ΕΞΙΣΩΣΕΩΝ

Μέχρι στιγμής χρησιμοποιήσαμε την εντολή **DSolve** για την επίλυση συνήθων διαφορικών εξισώσεων. Εντούτοις ή ίδια εντολή **DSolve** είναι εφαρμόσιμη και σε συστήματα συνήθων διαφορικών εξισώσεων. Συνήθως μάλιστα χρησιμοποιείται σε γραμμικά συστήματα συνήθων διαφορικών εξισώσεων με σταθερούς συντελεστές. Αυτά είναι και τα πιο εύκολα στην επίλυσή τους (διαθέτοντας αναλυτικές λύσεις) και τα πιο χρήσιμα στην πράξη. Εδώ θα αναφερθούμε στις ιδιοταλαντώσεις ενός απλού διώροφου ιδεατού κτιρίου διατμήσεως το οποίο προσεγγίζεται από διώροφο επίπεδο πλαίσιο. Ορίζουμε πρώτα τα μητρώα μάζας **M** και δυσκαμψίας **K** και το διάνυσμα των αγνώστων $\mathbf{u}(t)$. Έπειτα προσδιορίζουμε το σχετικό σύστημα διαφορικών εξισώσεων και το λύνουμε, αφού ορίσουμε πρώτα τις τέσσερις αρχικές συνθήκες:

```
In[39]:= {M = {{3 m, 0}, {0, 2 m}}; M // MatrixForm, K = {{5 k, -k}, {-k, 3 k}}; K // MatrixForm}
```

Out[39]= $\left\{\begin{pmatrix} 3 m & 0 \\ 0 & 2 m \end{pmatrix}, \begin{pmatrix} 5 k & -k \\ -k & 3 k \end{pmatrix}\right\}$

```
In[40]:= u[t_] = {{u1[t]}, {u2[t]}}
Out[40]= {{u1[t]}, {u2[t]}}
In[41]:= SystemOfODEs1 = M.u''[t] + K.u[t] == {{0}, {0}}
Out[41]= {{5 k u1[t] - k u2[t] + 3 m u1''[t]}, {-k u1[t] + 3 k u2[t] + 2 m u2''[t]}} == {{0}, {0}}
In[42]:= SystemOfODEs2 = Table[SystemOfODEs1[[1, n, 1]] / m == 0, {n, 1, 2}] /. k → m ω^2 // Simplify
Out[42]= {5 ω^2 u1[t] + 3 u1''[t] == ω^2 u2[t], 3 ω^2 u2[t] + 2 u2''[t] == ω^2 u1[t]}
In[43]:= InitialConditions = {u1[0] == u10, u1'[0] == v10, u2[0] == u20, u2'[0] == v20}
Out[43]= {u1[0] == u10, u1'[0] == v10, u2[0] == u20, u2'[0] == v20}
In[44]:= solution =
           DSolve[{SystemOfODEs2, InitialConditions}, {u1[t], u2[t]}, t] // FullSimplify;
In[45]:= {us1[t_] = solution[[1, 1, 2]], us2[t_] = solution[[1, 2, 2]]}
```

Out[45]= $\left\{\frac{1}{70\,\omega}\left(14\,\omega\,\mathrm{Cos}\left[\sqrt{2}\,t\,\omega\right](3 u_{10} - 2 u_{20}) + 28\,\omega\,\mathrm{Cos}\left[\sqrt{\frac{7}{6}}\,t\,\omega\right](u_{10} + u_{20}) + 7\sqrt{2}\,\mathrm{Sin}\left[\sqrt{2}\,t\,\omega\right](3 v_{10} - 2 v_{20}) + 4\sqrt{42}\,\mathrm{Sin}\left[\sqrt{\frac{7}{6}}\,t\,\omega\right](v_{10} + v_{20})\right),\right.$

$\left.\frac{1}{70\,\omega}\left(42\,\omega\,\mathrm{Cos}\left[\sqrt{\frac{7}{6}}\,t\,\omega\right](u_{10} + u_{20}) + 14\,\omega\,\mathrm{Cos}\left[\sqrt{2}\,t\,\omega\right](-3 u_{10} + 2 u_{20}) + 6\sqrt{42}\,\mathrm{Sin}\left[\sqrt{\frac{7}{6}}\,t\,\omega\right](v_{10} + v_{20}) + 7\sqrt{2}\,\mathrm{Sin}\left[\sqrt{2}\,t\,\omega\right](-3 v_{10} + 2 v_{20})\right)\right\}$

Και τώρα θα κάνουμε την επαλήθευση της λύσεως τόσο ως προς το σύστημα των δύο συνήθων διαφορικών εξισώσεων **SystemOfODEs2** όσο και ως προς τις τέσσερις αρχικές συνθήκες **InitialConditions**:

```
In[46]:= verification = {SystemOfODEs2, InitialConditions} /. {u1 → us1, u2 → us2} // Simplify

Out[46]= {{True, True}, {True, True, True, True}}
```

Μπορούμε να βεβαιωθούμε για την ορθότητα των δύο ιδιοσυχνοτήτων που εμφανίζονται στην πιο πάνω λύση του συστήματός μας υπολογίζοντάς τις και με έναν εντελώς διαφορετικό τρόπο:

```
In[47]:= {Eigenfrequencies = Solve[Det[K - ω^2 M] == 0, ω],
          Eigenfrequencies /. k → m ω0^2 // PowerExpand}
```

Out[47]= $\left\{\left\{\left\{\omega \to -\frac{\sqrt{\frac{7}{6}}\sqrt{k}}{\sqrt{m}}\right\}, \left\{\omega \to \frac{\sqrt{\frac{7}{6}}\sqrt{k}}{\sqrt{m}}\right\}, \left\{\omega \to -\frac{\sqrt{2}\sqrt{k}}{\sqrt{m}}\right\}, \left\{\omega \to \frac{\sqrt{2}\sqrt{k}}{\sqrt{m}}\right\}\right\}, \left\{\left\{\omega \to -\sqrt{\frac{7}{6}}\,\omega_0\right\}, \left\{\omega \to \sqrt{\frac{7}{6}}\,\omega_0\right\}, \left\{\omega \to -\sqrt{2}\,\omega_0\right\}, \left\{\omega \to \sqrt{2}\,\omega_0\right\}\right\}\right\}$

Στις Ταλαντώσεις (π.χ. μηχανικών συστημάτων, δοκών, κλπ.) καθώς και στη Δυναμική των Κατασκευών γενικότερα είναι αυτονόητο πως μονάχα οι θετικές πιο πάνω λύσεις της εξισώσεως ιδιοσυχνοτήτων έχουν φυσική έννοια. (Οι αρνητικές λύσεις περιττεύουν. Δε χρειάζεται να ληφθούν καθόλου υπόψη.)

■ ΕΠΙΛΥΣΗ ΔΙΑΦΟΡΙΚΩΝ ΕΞΙΣΩΣΕΩΝ ΜΕ ΜΕΡΙΚΕΣ ΠΑΡΑΓΩΓΟΥΣ

Η παρούσα εντολή επιλύσεως διαφορικών εξισώσεων **DSolve** μπορεί να οδηγήσει στη λύση και μερικών απλών γραμμικών διαφορικών εξισώσεων με μερικές παραγώγους. Εδώ ξεκινάμε με τη μονοδιάστατη εξίσωση του κύματος, η οποία καλείται συνήθως και εξίσωση της χορδής. Επειδή στην Επιστήμη του Πολιτικού Μηχανικού η εξίσωση αυτή εφαρμόζεται προσεγγιστικά και σε καλώδια (στις καλωδιωτές και στις κρεμαστές γέφυρες), την καλούμε εδώ εξίσωση του καλωδίου (cable equation). Αυτή η διαφορική εξίσωση με μερικές παραγώγους έχει μια απλή κλειστή λύση η οποία εκφράζεται με τη βοήθεια δύο αυθαίρετων, αλλά δύο φορές παραγωγίσιμων, συναρτήσεων. Στις επόμενες εντολές ορίζουμε την εξίσωση του καλωδίου, τη λύνουμε με τη χρήση της εντολής **DSolve**, ορίζουμε την αντίστοιχη συνάρτηση και επαληθεύουμε πως αυτή είναι στ' αλήθεια λύση της εξισώσεως του καλωδίου (με απλοποίηση):

```
In[48]:= CableEquation = D[u[x, t], {x, 2}] == (1 / c^2) D[u[x, t], {t, 2}]
```

Out[48]= $u^{(2,0)}[x, t] == \frac{u^{(0,2)}[x, t]}{c^2}$

```
In[49]:= sol1 = DSolve[CableEquation, u[x, t], {x, t}] // PowerExpand
```

Out[49]= $\left\{\left\{u[x, t] \to C[1]\left[t + \frac{x}{c}\right] + C[2]\left[t - \frac{x}{c}\right]\right\}\right\}$

```
In[50]:= us[x_, t_] = sol1[[1, 1, 2]]
```

Out[50]= $C[1]\left[t + \frac{x}{c}\right] + C[2]\left[t - \frac{x}{c}\right]$

```
In[51]:= verification1 = CableEquation /. u → us // Simplify

Out[51]= True
```

Απόλυτα ανάλογη εργασία μπορεί να γίνει (και με επιτυχία μάλιστα!) για τη διδιάστατη εξίσωση του Laplace. Εδώ όμως η λύση περιέχει και τη φανταστική μονάδα *i*. Για να είναι η τελική λύση πραγματική, όπως στ' αλήθεια επιθυμεί ο Πολιτικός Μηχανικός στην επιστήμη του, οι δύο αυθαίρετες συναρτήσεις πρέπει να είναι συζυγείς μιγαδικές, δηλαδή όπου η μία έχει *i* η άλλη να έχει $-i$. Σημειώνεται ότι ειδικά εδώ η επαλήθευση δεν απαιτεί τη χρήση της εντολής **Simplify**:

```
In[52]:= TwoDimensionalLaplaceEquation = D[u[x, y], {x, 2}] + D[u[x, y], {y, 2}] == 0
```

```
Out[52]= u^(0,2)[x, y] + u^(2,0)[x, y] == 0
```

```
In[53]:= sol2 = DSolve[TwoDimensionalLaplaceEquation, u[x, y], {x, y}]
```

```
Out[53]= {{u[x, y] → C[1][ⅈ x + y] + C[2][-ⅈ x + y]}}
```

Πολύ συχνά δε θέλουμε οι αυθαίρετες συναρτήσεις στις λύσεις διαφορικών εξισώσεων με μερικές παραγώγους να δηλώνονται με το σύμβολο *C* που γενικά αναφέρεται σε σταθερές. Προς το σκοπό αυτό, όπως έχουμε ήδη δει και στις συνήθεις διαφορικές εξισώσεις, αυτό το σύμβολο *C* μπορούμε να το αλλάξουμε στη *Mathematica* χρησιμοποιώντας την επιλογή **DSolveConstants** της εντολής **DSolve.** Αυτό το κάνουμε αμέσως παρακάτω δηλώνοντας τις δύο αυθαίρετες συναρτήσεις είτε με *f* είτε με *φ* την ώρα ακριβώς που χρησιμοποιούμε την εντολή **DSolve**:

```
In[54]:= sol3 = DSolve[TwoDimensionalLaplaceEquation, u[x, y], {x, y}, DSolveConstants → f]
```

```
Out[54]= {{u[x, y] → f[1][ⅈ x + y] + f[2][-ⅈ x + y]}}
```

```
In[55]:= sol4 = DSolve[TwoDimensionalLaplaceEquation, u[x, y], {x, y}, DSolveConstants → ϕ]
```

```
Out[55]= {{u[x, y] → ϕ[1][ⅈ x + y] + ϕ[2][-ⅈ x + y]}}
```

```
In[56]:= us[x_, y_] = sol4[[1, 1, 2]]
```

```
Out[56]= ϕ[1][ⅈ x + y] + ϕ[2][-ⅈ x + y]
```

```
In[57]:= verification2 = TwoDimensionalLaplaceEquation /. u → us
```

```
Out[57]= True
```

Αντίθετα η μονοδιάστατη εξίσωση της διαχύσεως (η οποία παρουσιάζεται στη Μετάδοση Θερμότητας, στην Εδαφομηχανική: Στερεοποίηση, στην Περιβαλλοντική Μηχανική, κλπ.) δεν έχει κλειστή λύση με τη χρήση δύο αυθαίρετων συναρτήσεων, γιατί η χρονική μερική παράγωγός της είναι πρώτης τάξεως, ενώ η χωρική είναι δευτέρας τάξεως. Κι αφού δεν έχει τέτοια κλειστή λύση, με κανέναν τρόπο δε μπορούμε να περιμένουμε από τη *Mathematica* να τη βρει με τη χρήση της εντολής **DSolve**. Άρα δεν είναι κάποια αδυναμία της εντολής **DSolve**. Είναι απλά η ανυπαρξία τέτοιας κλειστής λύσεως: δεν υπάρχει λύση σε αναλυτική μορφή. Ας τα δούμε λοιπόν αυτά:

```
In[58]:= OneDimensionalDiffusionEquation = D[u[x, t], {x, 2}] == (1 / a^2) D[u[x, t], t]
```

```
Out[58]= u^(2,0)[x, t] == u^(0,1)[x, t] / a^2
```

```
In[59]:= sol5 = DSolve[OneDimensionalDiffusionEquation, u[x, t], {x, t}]
```

```
Out[59]= DSolve[u^(2,0)[x, t] == u^(0,1)[x, t] / a^2, u[x, t], {x, t}]
```

Από την αντίθετη όψη η εντολή **DSolve** της *Mathematica* δε μπορεί να λύσει τη διαρμονική εξίσωση (στην τασική συνάρτηση του Airy στην Επίπεδη Ελαστικότητα στη Μηχανική των Υλικών), παρόλο που αυτή διαθέτει απλή κλειστή λύση με την παρουσία τεσσάρων (όχι μόνο δύο) αυθαίρετων συναρτήσεων. Αυτό ίσως συμβαίνει, επειδή η διαρμονική εξίσωση είναι τετάρτης τάξεως και η εντολή **DSolve** ίσως δεν έχει προγραμματισθεί αρκετά, ώστε να λύνει και τέτοιες εξισώσεις. Ας το δούμε κι αυτό, αφού βέβαια καλέσουμε πρώτα το πακέτο **Calculus'VectorAnalysis'** για τη Διανυσματική Ανάλυση:

```
In[60]:= << Calculus`VectorAnalysis`
```

```
In[61]:= BiharmonicPDE = Biharmonic[u[x, y], Cartesian[x, y, z]] == 0

Out[61]= u^(0,4)[x, y] + 2 u^(2,2)[x, y] + u^(4,0)[x, y] == 0

In[62]:= DSolve[BiharmonicPDE, u[x, y], {x, y}]

Out[62]= DSolve[u^(0,4)[x, y] + 2 u^(2,2)[x, y] + u^(4,0)[x, y] == 0, u[x, y], {x, y}]
```

Άρα η *Mathematica* δεν κατάφερε να βρει την κλειστή λύση της διαρμονικής εξισώσεως, αν και υπάρχει. (Η λύση αυτή περιέχει τέσσερις αυθαίρετες συναρτήσεις.) Μια που την ξέρουμε, τη δίνουμε λοιπόν εμείς οι ίδιοι με το χέρι και απλά την επαληθεύουμε με τη *Mathematica*, που εκτελεί έτσι μόνο τις σχετικές πολύ απλές παραγωγίσεις (με την υπόθεση βέβαια ότι οι αυθαίρετες συναρτήσεις είναι παραγωγίσιμες):

```
In[63]:= us[x_, y_] = (x - ⅈ y) f1[x + ⅈ y] + (x + ⅈ y) f2[x - ⅈ y] + f3[x + ⅈ y] + f4[x - ⅈ y];

In[64]:= verification = BiharmonicPDE /. u → us // Simplify

Out[64]= True
```

Ας γίνει στο σημείο αυτό η υπενθύμιση ότι ο Πολιτικός Μηχανικός ενδιαφέρεται συνήθως μόνο για πραγματικές συναρτήσεις, όπως είναι η τασική συνάρτηση του Airy, που είναι όχι μόνο διαρμονική, αλλά και πραγματική. Στην περίπτωση αυτή οι συναρτήσεις f_1 και f_2 θα πρέπει να είναι συζυγείς μιγαδικές συναρτήσεις, δηλαδή όπου η μία έχει i η άλλη να έχει $-i$. Το ίδιο και οι άλλες δύο συναρτήσεις f_3 και f_4.

■ ΕΝΤΟΛΗ D2: ΑΡΙΘΜΗΤΙΚΗ ΕΠΙΛΥΣΗ ΔΙΑΦΟΡΙΚΩΝ ΕΞΙΣΩΣΕΩΝ

NDSolve[{*ΣυνήθηςΔιαφορικήΕξίσωση,*
***ΑρχικέςΣυνθήκεςΉΣυνοριακέςΣυνθήκεςΉΚαιΆλλεςΣυνθήκες*},**
***Άγνωστη Συνάρτηση*, {*ΑνεξάρτητηΜεταβλητή, ΑρχήΔιαστήματος, ΤέλοςΔιαστήματος* }]**

NDSolve[*ΛίσταΣυνήθωνΔιαφορικώνΕξισώσεωνΚαιΑρχικώνΣυνθηκώνΉΣυνοριακώνΣυνθηκών-ΉΚαιΆλλωνΣυνθηκών, ΛίσταΑγνώστωνΣυναρτήσεων,*
{*ΑνεξάρτητηΜεταβλητή, ΑρχήΔιαστήματος, ΤέλοςΔιαστήματος* }]

NDSolve[{*ΔιαφορικήΕξίσωσηΜεΜερικέςΠαραγώγους,*
***ΑρχικέςΣυνθήκεςΉΣυνοριακέςΣυνθήκεςΉΚαιΆλλεςΣυνθήκες*}, *ΆγνωστηΣυνάρτηση*,**
{*ΑνεξάρτητηΜεταβλητή-1, ΑρχήΔιαστήματος-1, ΤέλοςΔιαστήματος-1* },
{*ΑνεξάρτητηΜεταβλητή-2, ΑρχήΔιαστήματος-2, ΤέλοςΔιαστήματος-2* }]

Πάρα πολύ συχνά μια συνήθης διαφορική εξίσωση, μια διαφορική εξίσωση με μερικές παραγώγους ή ένα σύστημα διαφορικών εξισώσεων (είτε συνήθων είτε με μερικές παραγώγους) δεν έχουν κλειστή λύση. Επομένως σ' αυτές τις περιπτώσεις πρέπει να χρησιμοποιηθεί αριθμητική μέθοδος επιλύσεως. Η παρούσα εντολή **NDSolve** λύνει πράγματι αριθμητικά μια συνήθη διαφορική εξίσωση ή ένα σύστημα συνήθων διαφορικών εξισώσεων ή μια διαφορική εξίσωση με μερικές παραγώγους που δίνεται στη λίστα του πρώτου ορίσματός της μαζί με τις σχετικές αρχικές συνθήκες ή συνοριακές συνθήκες ή/και άλλες συνθήκες, που δίνονται επίσης στην ίδια λίστα. Το αποτέλεσμα είναι η αριθμητική επίλυση της διαφορικής εξισώσεως (είτε συνήθους είτε με μερικές παραγώγους) σε μορφή συναρτήσεως παρεμβολής ή (σε περίπτωση συστήματος διαφορικών εξισώσεων) συναρτήσεων παρεμβολής. Σημειώνεται ότι στην παρούσα αριθμητική εντολή **NDSolve** το διάστημα εργασίας στην αριθμητική επίλυση θα πρέπει οπωσδήποτε να αναφέρεται μετά από κάθε μεταβλητή. Η παρούσα εντολή **NDSolve** είναι εφαρμόσιμη τόσο σε συνήθεις διαφορικές εξισώσεις και σε σχετικά συστήματα όσο και σε διαφορικές εξισώσεις με μερικές παραγώγους. Εντούτοις και στις δύο αυτές περιπτώσεις είναι απόλυτα αναγκαίο να είναι γνωστές και να δίνονται και οι σχετικές αρχικές ή/και συνοριακές συνθήκες. Παραδείγματα:

■ ΑΡΙΘΜΗΤΙΚΗ ΕΠΙΛΥΣΗ ΣΥΝΗΘΩΝ ΔΙΑΦΟΡΙΚΩΝ ΕΞΙΣΩΣΕΩΝ ΜΕ ΑΡΧΙΚΕΣ/ΣΥΝΟΡΙΑΚΕΣ ΣΥΝΘΗΚΕΣ

Ξεκινάμε με ένα παράδειγμα συνήθους διαφορικής εξισώσεως (με μία αρχική συνθήκη) η οποία μάλλον δεν έχει κλειστή λύση (ή έστω η εντολή **DSolve** της *Mathematica* δε μπορεί να τη βρει):

```
In[65]:= ClosedFormSolution = DSolve[{y'[x] + Sin[x] y[x]^2 == 1, y[0] == 1}, y[x], x]

        Solve::ifun : Inverse functions are being used by Solve, so some solutions may not be found.

Out[65]= DSolve[{Sin[x] y[x]^2 + y′[x] == 1, y[0] == 1}, y[x], x]
```

Προχωράμε τώρα στην αριθμητική επίλυση της συνήθους αυτής διαφορικής εξισώσεως. Η λύση που προκύπτει με την εντολή αυτή **NDSolve** είναι σε μορφή συναρτήσεως παρεμβολής

```
In[66]:= NumericalSolution = NDSolve[{y'[x] + Sin[x] y[x]^2 == 1, y[0] == 1}, y[x], {x, 0, 3}]

Out[66]= {{y[x] → InterpolatingFunction[{{0., 3.}}, <>][x]}}
```

Από τη συνάρτηση παρεμβολής που βρήκε η εντολή **NDSolve** μπορεί εύκολα να ορισθεί και η σχετική συνάρτηση για την αριθμητική λύση της διαφορικής εξισώσεως (μαζί βέβαια με την αρχική συνθήκη της)

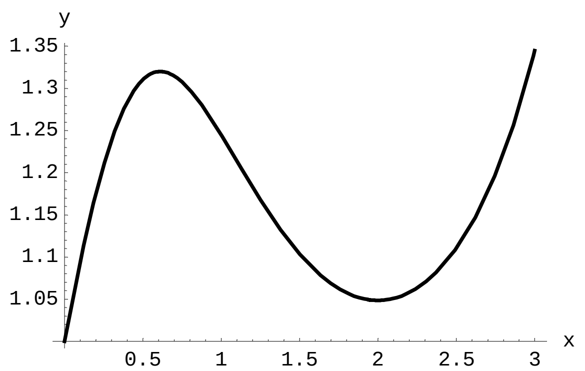
```
In[67]:= ys[x_] = NumericalSolution〚1, 1, 2〛

Out[67]= InterpolatingFunction[{{0., 3.}}, <>][x]
```

Η αριθμητική αυτή λύση (στη μορφή συναρτήσεως παρεμβολής) μπορεί να δώσει τις προσεγγιστικές τιμές της συναρτήσεως σε οποιοδήποτε σημείο του διαστήματος αριθμητικής επιλύσεως είναι επιθυμητό:

```
In[68]:= {ys[0.2], ys[1.4], Table[ys[k], {k, 0, 3}]}

Out[68]= {1.17491, 1.12682, {1., 1.24504, 1.04859, 1.34484}}
```

Η ίδια αριθμητική λύση $y_s(x)$ μπορεί επίσης να σχεδιασθεί εύκολα με τη χρήση της εντολής **Plot** βέβαια μέσα στο διάστημα αριθμητικής επιλύσεως, συγκεκριμένα εδώ στο διάστημα [0, 3], και όχι έξω από αυτό:

```
In[69]:= Plot[ys[x], {x, 0, 3}, PlotStyle → Thickness[0.008], AxesLabel → {"x", "y"}];
```



Πιο κάτω καλό είναι να έχουν εκτελεσθεί οι δύο εντολές μη εμφανίσεως πιθανών ορθογραφικών λαθών:

```
In[70]:= Off[General::spell]; Off[General::spell1]
```

■ ΑΡΙΘΜΗΤΙΚΗ ΕΠΙΛΥΣΗ ΣΥΣΤΗΜΑΤΩΝ ΣΥΝΗΘΩΝ ΔΙΑΦΟΡΙΚΩΝ ΕΞΙΣΩΣΕΩΝ

Ανάλογα μπορούν να επιλυθούν αριθμητικά με την ίδια εντολή **NDSolve** και συστήματα συνήθων διαφορικών εξισώσεων (systems of ordinary differential equations) μαζί βέβαια με τις σχετικές συνθήκες (εδώ πολύ συχνά αρχικές συνθήκες: initial conditions). Παράδειγμα τέτοιου συστήματος:

```
In[71]:= Clear[u]; SystemOfDifferentialEquations = {u''[t] - v[t] == t, u[t] + 2 v'[t] == Cos[t]};

In[72]:= InitialConditions = {u[0] == 1, u'[0] == 2, v[0] == -3};
```

Επίλυση με την εντολή **NDSolve** ως προς τις δύο άγνωστες συναρτήσεις στο διάστημα [0, 10]

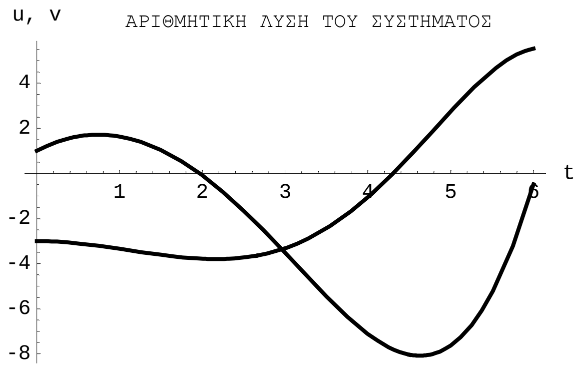
```
In[73]:= Nsolution = NDSolve[{SystemOfDifferentialEquations,
                  InitialConditions}, {u[t], v[t]}, {t, 0, 10}]

Out[73]= {{u[t] → InterpolatingFunction[{{0., 10.}}, <>][t],
          v[t] → InterpolatingFunction[{{0., 10.}}, <>][t]}}
```

Και τώρα ο ορισμός των δύο σχετικών συναρτήσεων για την αριθμητική αυτή λύση **Nsolution**

```
In[74]:= {us[t_] = Nsolution[[1, 1, 2]], vs[t_] = Nsolution[[1, 2, 2]]}

Out[74]= {InterpolatingFunction[{{0., 10.}}, <>][t],
         InterpolatingFunction[{{0., 10.}}, <>][t]}
```

Σχεδίαση της αριθμητικής λύσεως που βρέθηκε σε τμήμα [0, 6] του διαστήματος αριθμητικής επιλύσεως [0, 10], που χρησιμοποιήθηκε στην εντολή **NDSolve**:

```
In[75]:= Plot[{us[t], vs[t]}, {t, 0, 6}, PlotStyle → Thickness[0.008],
          PlotLabel -> "    ΑΡΙΘΜΗΤΙΚΗ ΛΥΣΗ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ", AxesLabel → {"t", "u, v"}];
```




```
In[76]:=
```

Αριθμητική επαλήθευση των δύο διαφορικών εξισώσεων του συστήματος και των αρχικών συνθηκών:

```
In[77]:= Table[us''[t] - vs[t] - t, {t, 0, 10}]

Out[77]= {-17.993, -0.0000400987, -1.28032×10^-6, -0.000261487, -0.0000541757,
         0.000639855, 0.00015022, 0.000354712, -0.00122065, 0.00292976, 1.49795×10^-8}
```

```
In[78]:= Table[us[t] + 2 vs'[t] - Cos[t], {t, 0, 10}]

Out[78]= {-0.0148941, 2.51129×10^-6, -5.97268×10^-6, 9.50563×10^-6, -0.0000111401,
          0.0000351841, 0.00010109, 7.05166×10^-6, -3.76863×10^-6, 0.000084709, -1.77626×10^-8}

In[79]:= InitialConditions /. {u → us, v → vs}

Out[79]= {True, True, True}
```

■ ΑΡΙΘΜΗΤΙΚΗ ΕΠΙΛΥΣΗ ΔΙΑΦΟΡΙΚΩΝ ΕΞΙΣΩΣΕΩΝ ΜΕ ΜΕΡΙΚΕΣ ΠΑΡΑΓΩΓΟΥΣ

Τέλος με την ίδια πάλι εντολή **NDSolve** μπορούν να επιλυθούν αριθμητικά και διαφορικές εξισώσεις με μερικές παραγώγους (partial differential equations) μαζί βέβαια και με τις σχετικές συνθήκες (αρχικές συνθήκες: initial conditions ή/και συνοριακές συνθήκες: boundary conditions). Παράδειγμα μιας τέτοιας εξισώσεως είναι η τόσο γνωστή μας εξίσωση της μεταγωγής–διαχύσεως (convection–diffusion) στην Περιβαλλοντική Υδραυλική στη μεταφορά ρύπου (και με μεταγωγή και με διάχυση–διασπορά) σε ένα υδατόρρευμα από τη χρονική στιγμή $t = 0$ και μετά. Έχουμε λοιπόν στο παράδειγμα αυτό τη γνωστή μας διαφορική εξίσωση με μερικές παραγώγους της μεταγωγής–διαχύσεως:

```
In[80]:= ConvectionDiffusionPDE = D[c[x, t], t] == D0 D[c[x, t], {x, 2}] - V D[c[x, t], x]

Out[80]= c^(0,1)[x, t] == -V c^(1,0)[x, t] + D0 c^(2,0)[x, t]
```

και τις σχετικές τρεις συνθήκες: μία αρχική συνθήκη και δύο συνοριακές συνθήκες καθώς και τις αριθμητικές τιμές των παραμέτρων που υποθέτουμε εδώ:

```
In[81]:= conditions = {c[x, 0] == 0, c[0, t] == 5 t Exp[t / 3], c[10, t] == 2 Sin[3 t]};
         values = {D0 → 1, V → 3};
```

Φυσικά η πιο πάνω διαφορική εξίσωση με μερικές παραγώγους, η εξίσωση της μεταγωγής–διαχύσεως, δε μπορεί να επιλυθεί αναλυτικά με τη συνηθισμένη εντολή **DSolve** είτε χωρίς τις τρεις συνθήκες

```
In[82]:= DSolve[ConvectionDiffusionPDE, c[x, t], {x, t}]

Out[82]= DSolve[c^(0,1)[x, t] == -V c^(1,0)[x, t] + D0 c^(2,0)[x, t], c[x, t], {x, t}]
```

είτε μαζί και με τις τρεις συνθήκες (την αρχική συνθήκη και τις δύο συνοριακές συνθήκες):

```
In[83]:= DSolve[{ConvectionDiffusionPDE, conditions}, c[x, t], {x, t}]

Out[83]= DSolve[{c^(0,1)[x, t] == -V c^(1,0)[x, t] + D0 c^(2,0)[x, t],
           {c[x, 0] == 0, c[0, t] == 5 e^(t/3) t, c[10, t] == 2 Sin[3 t]}}, c[x, t], {x, t}]
```

Είναι δυνατόν όμως να επιλυθεί αριθμητικά με την εντολή **NDSolve** αριθμητικής επιλύσεως διαφορικών εξισώσεων: εδώ μιας διαφορικής εξισώσεως με μερικές παραγώγους, συγκεκριμένα της εξισώσεως της μεταγωγής–διαχύσεως, και των τριών συνθηκών που τη συνοδεύουν:

```
In[84]:= Nsolution = NDSolve[{ConvectionDiffusionPDE /. values, conditions},
           c[x, t], {x, 0, 10}, {t, 0, 5}]

Out[84]= {{c[x, t] → InterpolatingFunction[{{0., 10.}, {0., 5.}}, <>][x, t]}}
```

Η λύση αυτή έχει υπολογισθεί σαν συνάρτηση παρεμβολής (interpolating function). Εύκολα ορίζουμε και τη σχετική συνάρτηση για τη συγκέντρωση του ρύπου στο υδατόρρευμα (σαν συνάρτηση παρεμβολής)

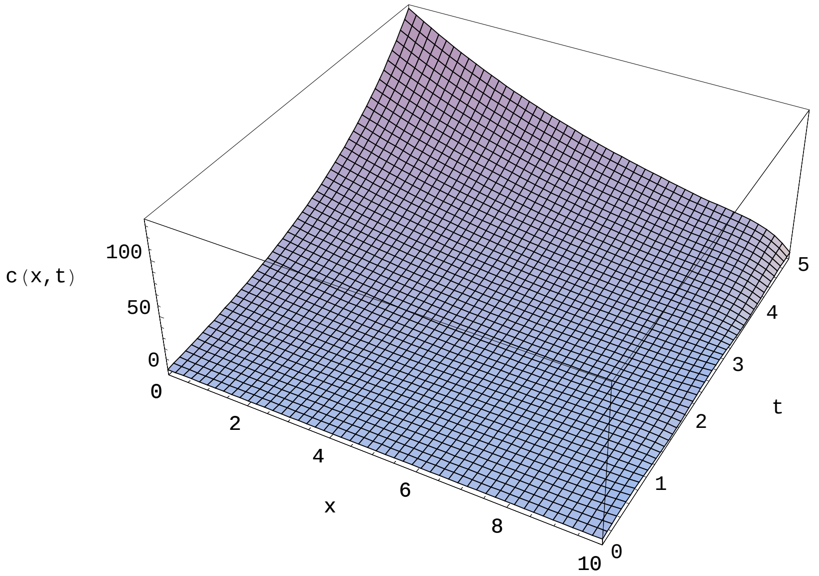
```
In[85]:= cn[x_, t_] = Nsolution[[1, 1, 2]]

Out[85]= InterpolatingFunction[{{0., 10.}, {0., 5.}}, <>][x, t]
```

Μπορούμε τώρα θαυμάσια να παίρνουμε αριθμητικές τιμές για τη λύση που βρήκαμε, π.χ. στη θέση $x = 2$

```
In[86]:= Table[cn[2, t], {t, 0, 5}] // Chop

Out[86]= {0, 2.31819, 10.8579, 26.0751, 51.7009, 93.5075}
```

ή να κάνουμε τη σχετική γραφική παράσταση με την εντολή **Plot3D** στο διάστημα [0, 10] για τη θέση *x* και [0, 5] για το χρόνο *t* που μας ενδιαφέρουν εδώ με μεγάλη ακρίβεια χάρη στην επιλογή **PlotPoints→50**

```
In[87]:= Plot3D[cn[x, t], {x, 0, 10}, {t, 0, 5},
           AxesLabel → {" x", " t", "c(x,t)  "}, PlotRange → All, PlotPoints → 50];
```



Και ένα δεύτερο παράδειγμα, όπου η εντολή **NDSolve** είναι και πάλι επιτυχής στην επίλυση διαφορικών εξισώσεων με μερικές παραγώγους. (Δεν είναι πάντοτε επιτυχής: έχει κι αυτή τους περιορισμούς της!) Το παράδειγμα αυτό αφορά στο ενδιαφέρον φαινόμενο της στερεοποιήσεως (consolidation) στην Εδαφομηχανική με τη γνωστή θεωρία της στερεοποιήσεως του εδάφους που διατυπώθηκε από τον καλούμενο πατέρα της Εδαφομηχανικής Karl Terzaghi. Κι εδώ έχουμε τη μονοδιάστατη εξίσωση της διαχύσεως

```
In[88]:= ConsolidationPDE = D[u[z, t], {z, 2}] == (1 / cv) D[u[z, t], t]
```

Out[88]= $u^{(2,0)}[z, t] == \frac{u^{(0,1)}[z, t]}{c_v}$

εδώ βέβαια στη θεωρία της στερεοποιήσεως με άγνωστη συνάρτηση $u(z, t)$ την πρόσθετη υδροστατική πίεση και τη σταθερά c_v να δηλώνει το συντελεστή στερεοποιήσεως. Θεωρούμε ότι η στερεοποίηση του εδάφους που μελετάμε γίνεται στο στρώμα εδάφους [0, 10] με δύο μηδενικές συνοριακές συνθήκες και με μία συγκεκριμένη (μη μηδενική) αρχική συνθήκη. Έχουμε έτσι συνολικά τρεις συνθήκες, τις εξής:

```
In[89]:= BoundaryConditions = {u[0, t] == 0, u[10, t] == 0};
         InitialCondition = u[z, 0] == Sin[π z / 10];
```

Και πάλι η εντολή **DSolve** δε μπορεί να λύσει το παρόν πρόβλημα. Αποτυγχάνει ξανά:

```
In[90]:= DSolve[{ConsolidationPDE, BoundaryConditions, InitialCondition}, u[z, t], {z, t}]
```

Out[90]= $\text{DSolve}\left[\left\{u^{(2,0)}[z,t]==\frac{u^{(0,1)}[z,t]}{c_v},\ \{u[0,t]==0,\ u[10,t]==0\},\ u[z,0]==\text{Sin}\left[\frac{\pi z}{10}\right]\right\},\ u[z,t],\ \{z,t\}\right]$

ενώ αντίθετα η αριθμητική εναλλακτική εντολή **NDSolve** μπορεί θαυμάσια να λύσει το πρόβλημα αυτό, αρκεί βέβαια να δώσουμε αριθμητική τιμή και στο συντελεστή στερεοποιήσεως:

```
In[91]:= NumericalSolution = NDSolve[{ConsolidationPDE /. cv → 2,
            BoundaryConditions, InitialCondition}, u[z, t], {z, 0, 10}, {t, 0, 8}]

Out[91]= {{u[z, t] → InterpolatingFunction[{{0., 10.}, {0., 8.}}, <>][z, t]}}
```

Υπολογίσθηκε λοιπόν η λύση, αλλά αριθμητικά, όχι αναλυτικά εδώ, στη μορφή συναρτήσεως παρεμβολής. Ορίζουμε τώρα και τη σχετική συνάρτηση για την προσέγγιση της πρόσθετης υδροστατικής πιέσεως που ζητάμε να προσδιορίσουμε (δυστυχώς μόνο αριθμητικά) και ήδη βρήκαμε:

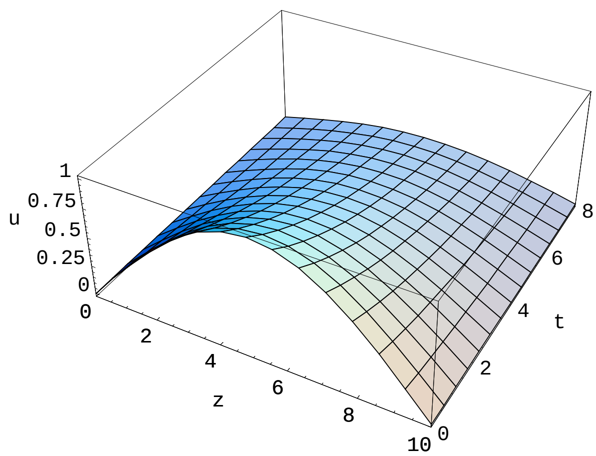
```
In[92]:= un[z_, t_] = NumericalSolution[[1, 1, 2]]

Out[92]= InterpolatingFunction[{{0., 10.}, {0., 8.}}, <>][z, t]
```

Διαθέτοντας πλέον καθαρά την αριθμητική αυτή λύση σε μορφή συναρτήσεως παρεμβολής, μπορούμε εύκολα να παίρνουμε αριθμητικές τιμές της. Για παράδειγμα:

```
In[93]:= Table[{un[0, t], un[10, t]}, {t, 0, 8}] // Chop

Out[93]= {{0, 0}, {0, 0}, {0, 0}, {0, 0}, {0, 0}, {0, 0}, {0, 0}, {0, 0}, {0, 0}}

In[94]:= Chop[Table[{un[z, 0], un[z, 0] - Sin[π z / 10] // N}, {z, 0, 10}], 10^-5]

Out[94]= {{0, 0}, {0.309012, 0}, {0.58779, 0}, {0.809012, 0}, {0.951049, 0}, {0.999991, 0},
          {0.951049, 0}, {0.809012, 0}, {0.58779, 0}, {0.309012, 0}, {0, 0}}
```

Βλέπουμε λοιπόν πως η αριθμητική λύση μας επαληθεύει τις δύο συνοριακές συνθήκες και τη μία αρχική συνθήκη. Φυσικά έχουμε επίσης τη δυνατότητα να κάνουμε και τη σχετική γραφική παράσταση με την εντολή **Plot3D** στο διάστημα [0, 10] για τη θέση *x* και [0, 8] για το χρόνο *t* που μας ενδιαφέρουν εδώ:

```
In[95]:= Plot3D[un[z, t], {z, 0, 10}, {t, 0, 8}, AxesLabel → {"z", "t", "u  "}, ImageSize → 290];
```



■ ΕΝΤΟΛΗ D3: ΣΧΕΔΙΑΣΗ ΠΕΔΙΟΥ ΚΑΤΕΥΘΥΝΣΕΩΝ

PlotVectorField[{1, *ΈκφρασηΤηςΠαραγώγουΧωρίςΣυναρτήσεις*}, {*Μεταβλητή-x, ΑρχικήΤιμή-x, ΤελικήΤιμή-x*}, {*Μεταβλητή-y, ΑρχικήΤιμή-y, ΤελικήΤιμή-y*}, *Επιλογή-1, Επιλογή-2*, ...]

Την εντολή αυτή **PlotVectorField** την έχουμε ήδη δει στο Notebook E12 για τη Διανυσματική Ανάλυση (Εντολή Δ9) και αφορά στη σχεδίαση ενός διανυσματικού πεδίου στις δύο διαστάσεις (x, y). Εδώ θα τη χρησιμοποιήσουμε αποκλειστικά για να σχεδιάσουμε το πεδίο κατευθύνσεων μιας συνήθους διαφορικής εξισώσεως πρώτης τάξεως της μορφής $y' = f(x, y)$. Ακολουθούν οι σχετικές λεπτομέρειες.

Εδώ η ίδια εντολή **PlotVectorField** θα χρησιμοποιηθεί σε διαφορικές εξισώσεις πρώτης τάξεως και μάλιστα αποκλειστικά της μορφής $y' = f(x, y)$ με άγνωστη συνάρτηση την $y = y(x)$ και τη συνάρτηση $f(x, y)$ στο δεξιό μέλος, δηλαδή την παράγωγο $y'(x)$, γνωστή. Το πεδίο κατευθύνσεων είναι απλά ένα σχήμα, όπου σχεδιάζουμε σε πάρα πολλά σημεία (x, y) ένα ευθύγραμμο τμήμα με αρχή το σημείο αυτό και μήκος ίσο με την παράγωγο $y' = y'(x)$, δηλαδή ίσο με τη συνάρτηση του δεξιού μέλους $f(x, y)$. Αφού λοιπόν $y' = f(x, y)$, θα έχουμε και $\mathrm{d}y = f(x, y)\,\mathrm{d}x$. Δηλαδή σε κάθε μεταβολή $\mathrm{d}x$ της μεταβλητής x θα αντιστοιχεί μεταβολή $f(x, y)\,\mathrm{d}x$ της μεταβλητής y. Επομένως το βελάκι στο παρόν πεδίο κατευθύνσεων θα δείχνει τη διεύθυνση $(\mathrm{d}x, \mathrm{d}y)$ ή ισοδύναμα $(\mathrm{d}x,\ f(x, y)\,\mathrm{d}x)$ ή τελικά $(1,\ f(x, y))$. Κατά συνέπεια εδώ αρκεί στο διανυσματικό πεδίο που θα σχεδιάσουμε με την παρούσα εντολή **PlotVectorField** να θέσουμε $(1,\ f(x, y))$ για το διάνυσμα που θα σχεδιασθεί. Τόσο απλά! Βέβαια το πεδίο κατευθύνσεων δε μας δίνει τις λύσεις της διαφορικής εξισώσεως. Δείχνει απλά τις κατευθύνσεις, τις διευθύνσεις που πρέπει να έχουν οι λύσεις της. Όμως με τον τρόπο αυτό μας επιτρέπει να πάρουμε μια πρώτη, πρόχειρη εικόνα της μορφής των λύσεων. Συγκεκριμένα οι λύσεις θα πρέπει να εφάπτονται στα βελάκια που δείχνουν την κλίση τους $y' = f(x, y)$ και που έχουν σχεδιασθεί άφθονα στο πεδίο κατευθύνσεων της διαφορικής εξισώσεως. Αυτά θα φανούν καλύτερα στα δύο παραδείγματα τα οποία ακολουθούν. (Σημειώνεται εδώ ότι στο πεδίο κατευθύνσεων σε διαφορικές εξισώσεις πρώτης τάξεως γίνεται επίσης αναφορά και στην Παράγραφο A20.1.1 στο Μέρος A των διδακτικών βιβλίων με συγκεκριμένο παράδειγμα από τη Δυναμική και με σχήμα με τη *Mathematica*.) Εδώ το πρώτο παράδειγμα είναι ένα γενικό παράδειγμα στις διαφορικές εξισώσεις, ενώ το δεύτερο παράδειγμα αποτελεί ένα παράδειγμα το οποίο αφορά σε ένα συγκεκριμένο πρόβλημα διδιάστατης ροής ιδεατού ρευστού. Θα πρέπει να σημειώσουμε στο σημείο αυτό ότι η παρούσα εντολή **PlotVectorField** είναι εντολή του πακέτου **Graphics‘PlotField‘**. Ασφαλώς αυτό το πακέτο θα πρέπει να έχει ήδη κληθεί πριν από οποιαδήποτε χρήση της παρούσας εντολής **PlotVectorField** ως εξής:

```
In[96]:= Needs["Graphics`PlotField`"]
```

ή εναλλακτικά με την ισοδύναμη και λίγο πιο σύντομη εντολή <<. Στο πρώτο μας παράδειγμα θεωρούμε μια απλή συνήθη διαφορική εξίσωση πρώτης τάξεως της μορφής $y' = f(x, y)$, εδώ με $f(x, y) = (x-2)\,y^2$, μαζί με την αρχική συνθήκη της $y(0) = 3$ για $x = 0$. Επομένως έχουμε ένα απλό πρόβλημα αρχικής τιμής:

```
In[97]:= equations = {de = y'[x] == (x - 2) y[x]^2, ic = y[0] == 3};
```

Η αναλυτική λύση **solution** του προβλήματος αυτού προσδιορίζεται εύκολα με την εντολή **DSolve**. Αυτό συμβαίνει, γιατί, όπως μπορεί να διαπιστωθεί, η διαφορική αυτή εξίσωση είναι χωριζόμενων μεταβλητών:

```
In[98]:= solution = DSolve[equations, y[x], x]
```

Out[98]= $\left\{\left\{y[x] \to -\dfrac{6}{-2 - 12\,x + 3\,x^2}\right\}\right\}$

Στη συνέχεια από τη λύση αυτή **solution** μπορεί αμέσως να ορισθεί η σχετική συνάρτηση $y_s(x)$:

```
In[99]:= ys[x_] = solution[[1, 1, 2]]
```

Out[99]= $-\frac{6}{-2-12\,x+3\,x^2}$

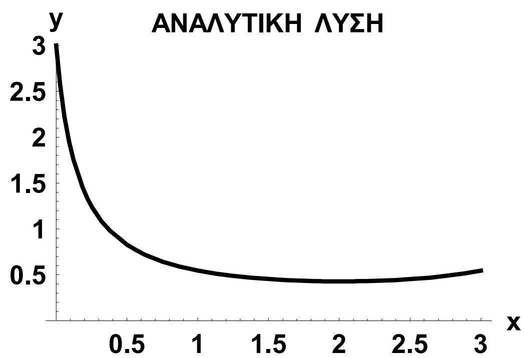
Αυτή επαληθεύεται βέβαια (σωστή!) και ως προς τη διαφορική εξίσωση και ως προς την αρχική συνθήκη:

```
In[100]:= verification = equations /. y → ys // Simplify
```

```
Out[100]= {True, True}
```

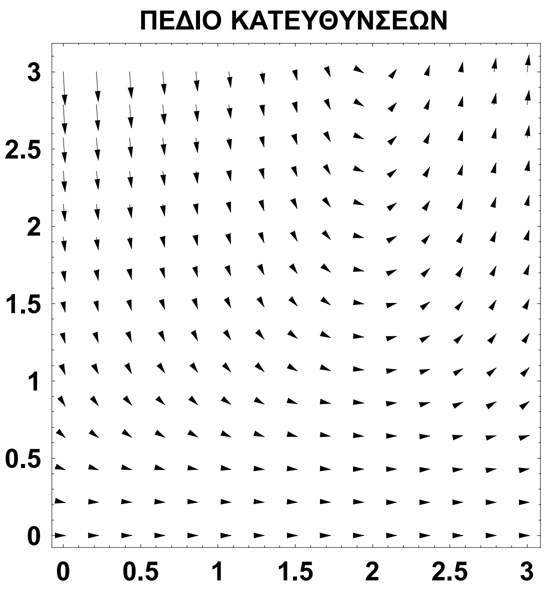
Ακολουθεί τώρα η γραφική παράσταση της λύσεως $y_s(x)$ που μόλις προσδιορίσθηκε:

```
In[101]:= Figure1 = Plot[ys[x], {x, 0, 3}, PlotRange → {0, 3},
            PlotStyle → Thickness[0.01], AxesLabel → {"x", "y"},
            PlotLabel -> "ΑΝΑΛΥΤΙΚΗ ΛΥΣΗ", DefaultFont → {"Arial-Bold", 13}];
```



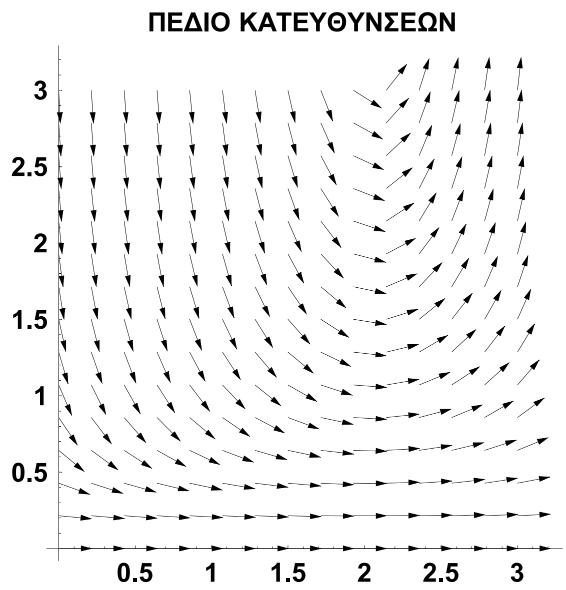
Ας δούμε τώρα και το πεδίο κατευθύνσεων της ίδιας διαφορικής εξισώσεως. Αυτό γίνεται στο επόμενο σχήμα φυσικά με τη χρήση της εντολής **PlotVectorField** ακριβώς όπως την εκθέσαμε αμέσως πιο πάνω. Τα βελάκια στο σχήμα αυτό δείχνουν την κλίση την οποία πρέπει να έχει κάθε λύση της διαφορικής μας εξισώσεως στην αρχή του καθενός από τα βελάκια αυτά:

```
In[102]:= Figure2 = PlotVectorField[{1, (x - 2) y^2}, {x, 0, 3}, {y, 0, 3}, Frame → True,
            PlotLabel -> "ΠΕΔΙΟ ΚΑΤΕΥΘΥΝΣΕΩΝ", DefaultFont → {"Arial-Bold", 13}];
```



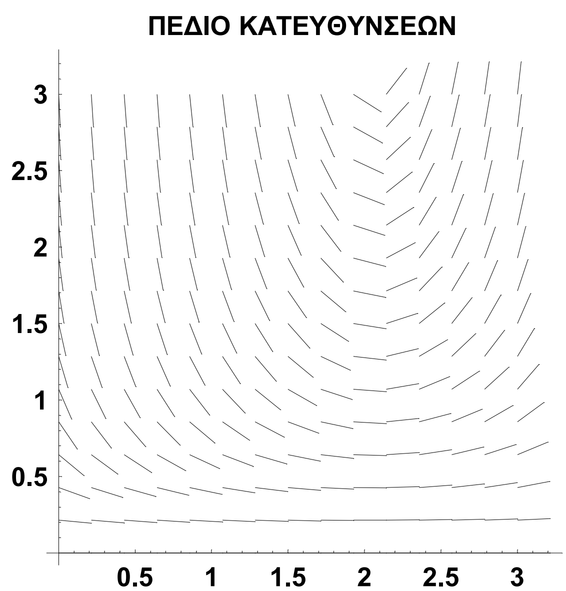
Παρατηρούμε όμως ότι τα μήκη στα βελάκια αυτά δεν είναι ίσα μεταξύ τους και αυτό μας ενοχλεί κάπως. Με την επιλογή **ScaleFunction → Function[t, 1]** μπορούμε να έχουμε όλα τα βελάκια με το ίδιο μήκος. Με την επιλογή **HeadLength** μπορούμε επίσης να καθορίζουμε και το μήκος της κεφαλής στο κάθε βελάκι:

```
In[103]:= Figure3 = PlotVectorField[{1, (x - 2) y^2}, {x, 0, 3}, {y, 0, 3},
    Axes → Automatic, ScaleFunction → Function[t, 1], HeadLength → 0.02,
    PlotLabel -> "ΠΕΔΙΟ ΚΑΤΕΥΘΥΝΣΕΩΝ", DefaultFont → {"Arial-Bold", 13}];
```



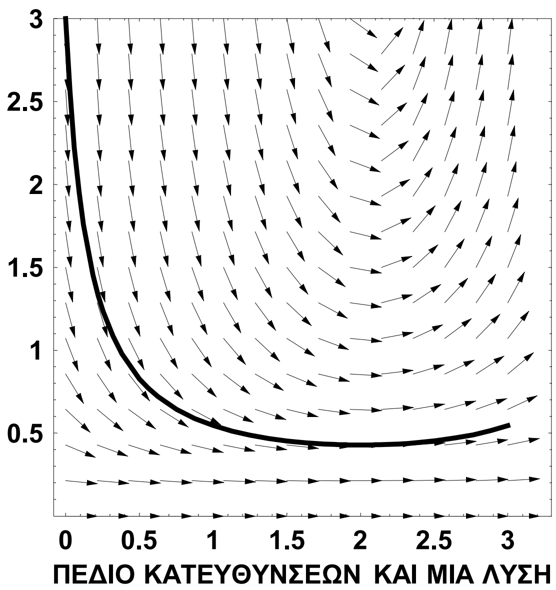
Σχεδόν πάντοτε στα Μαθηματικά στις Διαφορικές Εξισώσεις αντί για βελάκια χρησιμοποιούνται απλά ευθύγραμμα τμήματα σε πεδία κατευθύνσεων. Αυτό το πετυχαίνουμε και εδώ θέτοντας **HeadLength → 0**:

```
In[104]:= Figure4 = PlotVectorField[{1, (x - 2) y^2}, {x, 0, 3}, {y, 0, 3},
    Axes → Automatic, ScaleFunction → Function[t, 1], HeadLength → 0,
    PlotLabel -> "ΠΕΔΙΟ ΚΑΤΕΥΘΥΝΣΕΩΝ", DefaultFont → {"Arial-Bold", 13}];
```



Από τα πιο πάνω τρία σχήματα για το πεδίο κατευθύνσεων της παρούσας διαφορικής εξισώσεως παρατηρούμε ποια περίπου μορφή πρέπει να έχουν οι λύσεις της (που καλούνται μερικές φορές και ολοκληρωτικές καμπύλες). Πρέπει να έχουν σε κάθε σημείο κλίση που να συμπίπτει με την κατεύθυνση που δείχνει το αντίστοιχο βελάκι στο ίδιο σημείο. Και αυτό σε κάθε σημείο! Παίρνουμε έτσι μια προσεγγιστική γραφική μέθοδο επιλύσεως διαφορικών εξισώσεων. Εδώ με τη *Mathematica* δείχνουμε μια τέτοια λύση, συγκεκριμένα τη λύση που ξεκινάει από το σημείο (0, 3), δηλαδή την αρχική συνθήκη. Είναι η λύση που εδώ είχαμε ήδη βρει με αναλυτικό τρόπο. Εντούτοις η μέθοδος του πεδίου κατευθύνσεων ισχύει και σε συνήθεις διαφορικές εξισώσεις πρώτης τάξεως της μορφής $y' = f(x, y)$ που δεν έχουν αναλυτική λύση. Να λοιπόν ταυτόχρονα τόσο το πεδίο κατευθύνσεων όσο και η συγκεκριμένη λύση στο παρόν πρόβλημα αρχικής τιμής με βάση τα δύο προηγούμενα σχήματα **Figure1** και **Figure3** και χρήση της εντολής **Show**:

```
In[105]:= Figure5 = Show[Figure1, Figure3, AspectRatio → 1, Frame → True, PlotLabel → "",
            FrameLabel → {"ΠΕΔΙΟ ΚΑΤΕΥΘΥΝΣΕΩΝ ΚΑΙ ΜΙΑ ΛΥΣΗ", "", "", ""}, ImageSize → 290];
```



Το πιο πάνω σχήμα **Figure5** δείχνει καθαρά τη χρησιμότητα του πεδίου κατευθύνσεων στην προσεγγιστική γραφική εύρεση λύσεων συνήθων διαφορικών εξισώσεων πρώτης τάξεως της μορφής $y' = f(x, y)$.

Προχωράμε τώρα σε ένα δεύτερο (και το τελευταίο μας) παράδειγμα. Αυτό αφορά στο διδιάστατο πεδίο μόνιμης αστρόβιλης ροής ιδεατού ρευστού στη Ρευστομηχανική. (Σημειώνουμε ότι μόνιμη ροή είναι η σταθερή ροή, η ροή που δεν εξαρτάται από το χρόνο. Υπενθυμίζουμε επίσης ότι ιδεατό ρευστό είναι ένα ρευστό (α) ασυμπίεστο, δηλαδή με σταθερή πυκνότητα ρ, και (β) χωρίς συνεκτικότητα, ιξώδες: με $\mu = 0$.) Στο συγκεκριμένο πρόβλημα θεωρούμε τη ροή με συνιστώσες της ταχύτητας

```
In[106]:= {u[x_, y_] = Sinh[x] Cos[y], v[x_, y_] = -Cosh[x] Sin[y]};
```

Επαληθεύονται βέβαια τόσο η εξίσωση της συνεχείας όσο και η εξίσωση του αστρόβιλου της ροής:

```
In[107]:= {D[u[x, y], x] + D[v[x, y], y] == 0, D[u[x, y], y] == D[v[x, y], x]}

Out[107]= {True, True}
```

Προφανώς οι γραμμές ροής θα επαληθεύουν την εξής διαφορική εξίσωση:

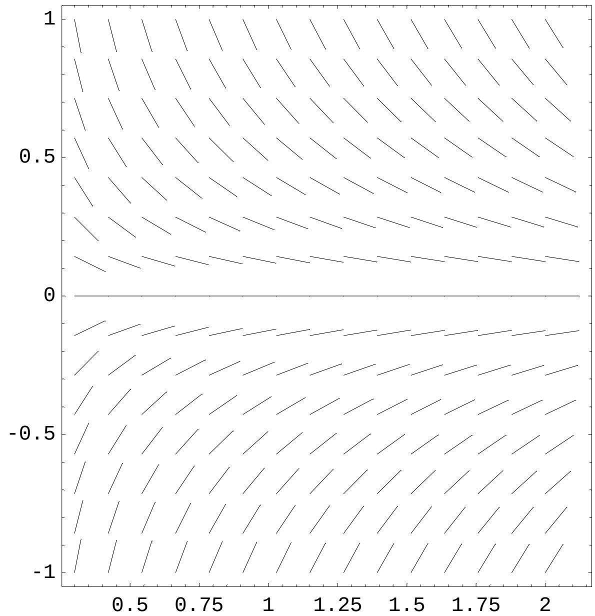
```
In[108]:= FlowDifferentialEquation = y'[x] == v[x, y] / u[x, y]

Out[108]= y′[x] == -Coth[x] Tan[y]
```

Επομένως πρόκειται πραγματικά για διαφορική εξίσωση πρώτης τάξεως της μορφής $y' = f(x, y)$, εδώ με

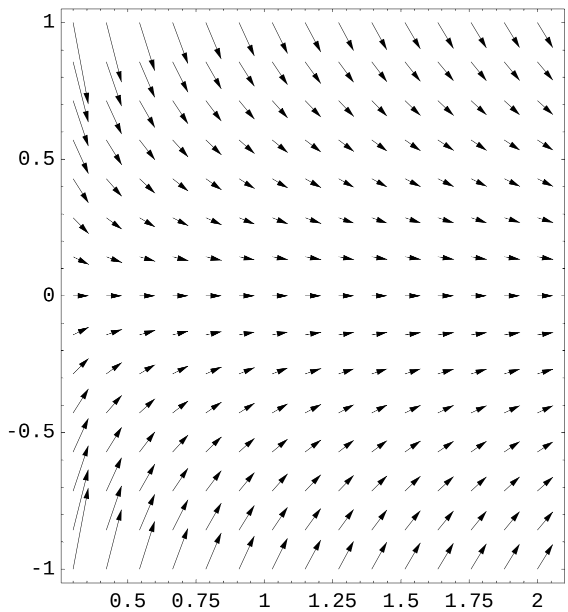
```
In[109]:= f[x_, y_] = FlowDifferentialEquation[[2]]

Out[109]= -Coth[x] Tan[y]
```

Μπορούμε λοιπόν τώρα να σχεδιάσουμε το σχετικό πεδίο κατευθύνσεων στα δύο πιο κάτω σχήματα. Στο πρώτο έχουμε ευθύγραμμα τμήματα και μάλιστα με ίσα μήκη. Αντίθετα στο δεύτερο έχουμε βελάκια και μάλιστα με άνισα μήκη. Στις Διαφορικές Εξισώσεις προτιμάμε βέβαια το πρώτο πεδίο κατευθύνσεων, ενώ στη Ρευστομηχανική (ή Μηχανική των Ρευστών) το δεύτερο. Ουσιαστικά όμως είναι ίδια!

```
In[110]:= FlowFigure1a = PlotVectorField[{1, f[x, y]}, {x, 0.3, 2}, {y, -1, 1},
            HeadLength → 0, ScaleFunction → Function[x, 1], Frame → True];
```




```
In[111]:= FlowFigure1b = PlotVectorField[{1, f[x, y]},
            {x, 0.3, 2}, {y, -1, 1}, ScaleFactor → 0.3, Frame → True];
```



Ας προχωρήσουμε τώρα και στην αναλυτική επίλυση της διαφορικής εξισώσεως των γραμμών ροής

```
In[112]:= FlowODE = y'[x] == -Coth[x] Tan[y[x]]

Out[112]= y′[x] == -Coth[x] Tan[y[x]]
```

Αυτή μπορεί να επιτευχθεί στο παρόν απλό πρόβλημα ροής ιδεατού ρευστού με την εντολή **DSolve**:

```
In[113]:= SolutionFlowODE = DSolve[FlowODE, y[x], x]

        Solve::ifun : Inverse functions are being used by Solve, so some solutions may not be found.

Out[113]= {{y[x] → ArcSin[ⅇ^C[1] Csch[x]]}}
```

(Υπάρχει βέβαια μια προειδοποίηση, αλλ' εντούτοις η λύση που πήραμε είναι σωστή.) Η συνάρτηση $y_s(x)$ που αντιστοιχεί στη λύση αυτή **SolutionFlowODE** είναι προφανώς η εξής:

```
In[114]:= ys[x_] = SolutionFlowODE[[1, 1, 2]]

Out[114]= ArcSin[ⅇ^C[1] Csch[x]]
```

Επαλήθευση **FlowODEVerification** της λύσεως της διαφορικής εξισώσεως **FlowODE** της παρούσας ροής:

```
In[115]:= FlowODEVerification = FlowODE /. y → ys

Out[115]= True
```

Παίρνοντας το ημίτονο και των δύο μελών της λύσεως και αλλάζοντας λιγάκι την εμφάνιση της σταθεράς, έχουμε μια καλύτερη έκφραση των γραμμών ροής:

```
In[116]:= FlowLine1[c_] = Sin[y] == Sin[ys[x]] /. ⅇ^C[1] → c

Out[116]= Sin[y] == c Csch[x]
```

Στο τέλος πολλαπλασιάζουμε και τα δύο μέλη αυτής της παραμετρικής εξισώσεως **FlowLine1[c_]** (με παράμετρο το c) επί sinh x. Προκύπτει έτσι η εξής ακόμη πιο απλή τελική έκφραση των γραμμών ροής:

```
In[117]:= FlowLine2[c_] = FlowLine1[c][[1]] Sinh[x] == FlowLine1[c][[2]] Sinh[x]

Out[117]= Sin[y] Sinh[x] == c
```

Τώρα για κάθε τιμή της σταθεράς c παίρνουμε και μια διαφορετική γραμμή ροής. Να και μερικές από τις γραμμές ροής (για ορισμένες τιμές της σταθεράς c) με τη χρήση της εντολής **Table**:

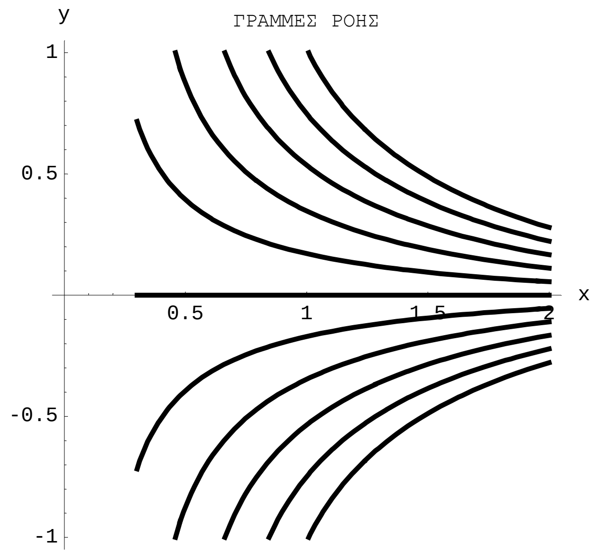
```
In[118]:= FlowLinesTable = Table[FlowLine2[n / 5], {n, -5, 5}]

Out[118]= {Sin[y] Sinh[x] == -1, Sin[y] Sinh[x] == -4/5,
  Sin[y] Sinh[x] == -3/5, Sin[y] Sinh[x] == -2/5, Sin[y] Sinh[x] == -1/5,
  Sin[y] Sinh[x] == 0, Sin[y] Sinh[x] == 1/5, Sin[y] Sinh[x] == 2/5,
  Sin[y] Sinh[x] == 3/5, Sin[y] Sinh[x] == 4/5, Sin[y] Sinh[x] == 1}
```

Αυτές τις γραμμές ροής μπορούμε φυσικά να τις σχεδιάσουμε. Η χρήση της εντολής **ImplicitPlot** αποτελεί μια τέτοια δυνατότητα. Αυτή όμως χρειάζεται πρώτα την κλήση του πακέτου **Graphics‘ImplicitPlot‘**:

```
In[119]:= Needs["Graphics`ImplicitPlot`"]
```

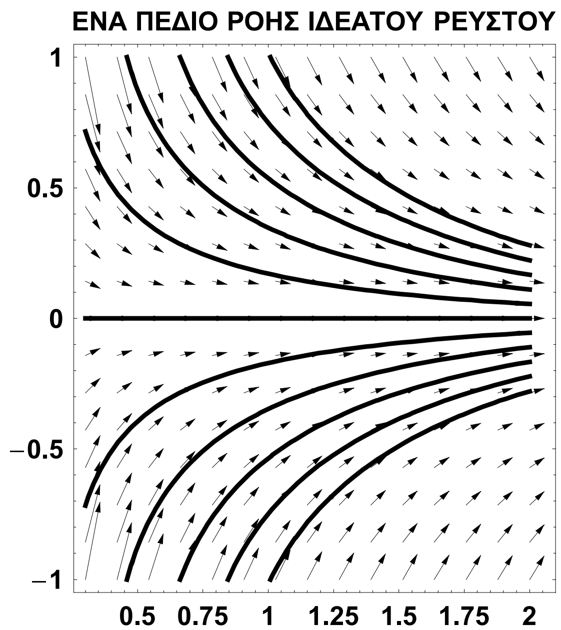
Τώρα μπορούμε χωρίς καμία δυσκολία να σχεδιάσουμε όλες αυτές τις γραμμές ροής **FlowLinesTable**:

```
In[120]:= FlowFigure2 = ImplicitPlot[FlowLinesTable, {x, 0.3, 2}, {y, -1, 1},
    PlotStyle → Thickness[0.01], AxesLabel → {x, y}, PlotLabel → "ΓΡΑΜΜΕΣ ΡΟΗΣ"];
```



Τέλος με υπέρθεση του πεδίου κατευθύνσεων του σχήματος **FlowFigure1b** (με τα βελάκια) και των αληθινών γραμμών ροής **FlowFigure2** παρατηρούμε ότι πράγματι το πεδίο κατευθύνσεων επιτρέπει τη σωστή γραφική σχεδίαση των γραμμών ροής (προσεγγιστικά βέβαια, άλλο που εδώ τις είχαμε υπολογίσει):

```
In[121]:= FlowFigure3 = Show[FlowFigure1b, FlowFigure2, ImageSize → 274,
    PlotLabel → "ΕΝΑ ΠΕΔΙΟ ΡΟΗΣ ΙΔΕΑΤΟΥ ΡΕΥΣΤΟΥ", DefaultFont → {"Arial-Bold", 13}];
```



Περισσότερα για το διδιάστατο (επίπεδο) πεδίο μόνιμης (σταθερής) αστρόβιλης ροής ιδεατού ρευστού αναφέρθηκαν ήδη στις τρεις εντολές Δ9, Δ10 και Δ11 του Notebook E12 για τη Διανυσματική Ανάλυση. Αναφέρονται επίσης και στις δύο εντολές C9 και C10 του Notebook E18 για τις Μιγαδικές Συναρτήσεις.

■ *Notebook E16*
ΕΝΤΟΛΕΣ ΓΙΑ ΣΕΙΡΕΣ FOURIER

10 ΕΝΤΟΛΕΣ: **F1. FourierTrigSeries, F2. NFourierTrigSeries,**
F3. FourierSeries, F4. NFourierSeries,
F5. FourierCosCoefficient, F6. NFourierCosCoefficient,
F7. FourierSinCoefficient, F8. NFourierSinCoefficient,
F9. FourierCoefficient, F10. NFourierCoefficient

■ ΕΝΤΟΛΗ F1: ΤΡΙΓΩΝΟΜΕΤΡΙΚΗ ΣΕΙΡΑ FOURIER

**FourierTrigSeries[*Συνάρτηση, Μεταβλητή, ΤάξηΠροσεγγίσεωςΤηςΣειράςFourier,*
FourierParameters → {−1, 1/*Περίοδος*}]**

Δημιουργεί την τριγωνομετρική σειρά Fourier της συναρτήσεως η οποία δίνεται στο πρώτο όρισμα ως προς τη μεταβλητή (π.χ. t ή x) που δίνεται στο δεύτερο όρισμα με όρους τάξεως μέχρι και N (μεταβλητή που δίνεται στο τρίτο όρισμα). Ακολουθεί η επιλογή **FourierParameters** με λίστα με δύο στοιχεία: το πρώτο στοιχείο είναι το -1 και το δεύτερο στοιχείο το αντίστροφο $1/T$ της περιόδου T της σειράς Fourier. Εάν η συνάρτηση η οποία προσεγγίζεται με την παρούσα σειρά Fourier δεν είναι περιοδική με περίοδο T, τότε η σειρά Fourier αφορά στο διάστημα $[-T/2, T/2]$. Σημειώνεται ότι για τη χρήση οποιασδήποτε εντολής αυτού του notebook πρέπει να έχει προηγουμένως κληθεί (φορτωθεί) το πακέτο **Calculus‘FourierTransform‘** της *Mathematica*. Αυτή η κλήση μπορεί να γίνει είτε με την πρώτη είτε με τη δεύτερη από τις δύο παρακάτω ισοδύναμες εντολές κλήσεως ενός πακέτου της *Mathematica.* Προηγούνται όμως από την κλήση αυτή οι δύο γνωστές μας εντολές μη εμφανίσεως μηνυμάτων για πιθανά ορθογραφικά λάθη. Αυτές είναι αρκετά χρήσιμες εδώ:

```
In[1]:= Off[General::spell]; Off[General::spell1];
```

```
In[2]:= Needs["Calculus`FourierTransform`"]
```

```
In[3]:= << Calculus`FourierTransform`
```

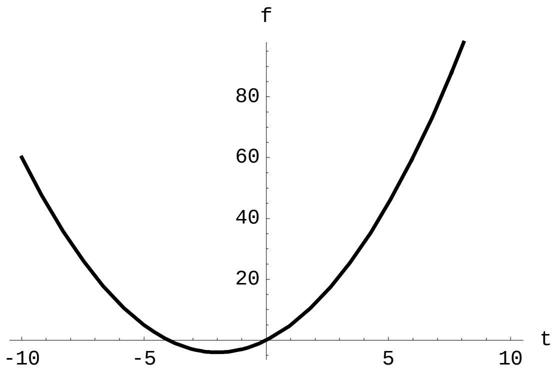
Σημειώνεται ότι λίγο παράδοξα οι εντολές υπολογισμού του μετασχηματισμού Fourier και του αντιστρόφου του δεν απαιτούν την κλήση αυτού του πακέτου **Calculus‘FourierTransform‘**. Έχουν μεταφερθεί στον πυρήνα (kernel) της *Mathematica.* Επομένως η ονομασία **Calculus‘FourierTransform‘** του παρόντος πακέτου ίσως πρέπει να θεωρείται πια λίγο άστοχη. Το πακέτο είναι τώρα εστιασμένο περισσότερο στις σειρές Fourier και όχι στους μετασχηματισμούς Fourier.

Και τώρα παράδειγμα τριγωνομετρικής σειράς Fourier με συνάρτηση την $f(t) = 4t + t^2$. Πρόκειται για μια συνάρτηση που έχει και άρτιο μέρος (το t^2) και περιττό μέρος (το $4t$). Άρα δεν είναι ούτε άρτια ούτε περιττή συνάρτηση και επομένως η σειρά Fourier της θα έχει και συνημιτονικούς και ημιτονικούς όρους.

```
In[4]:= f[t_] = 4 t + t^2;
```

Αυτή η συνάρτηση έχει την πιο κάτω πολύ απλή γραφική παράσταση στο διάστημα [−10, 10]:

```
In[5]:= Plot[f[t], {t, -10, 10}, PlotStyle -> Thickness[0.008], AxesLabel → {t, f}];
```



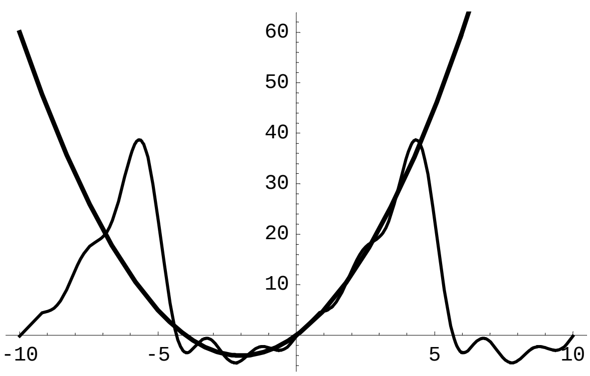
Τώρα που έχει ήδη κληθεί (φορτωθεί) το πακέτο **Calculus‘FourierTransform‘** μπορούμε πια εύκολα να υπολογίσουμε τη σχετική σειρά Fourier. Εδώ επιλέγουμε περίοδο $T = 10$ και το δηλώνουμε ρητά αυτό στη σχετική εντολή **FourierTrigSeries** (στον παρονομαστή T του δευτέρου στοιχείου $1/T$ της λίστας με δύο στοιχεία της επιλογής **FourierParameters** στο τέλος της εντολής **FourierTrigSeries**). Δηλώνουμε επίσης και την τάξη N της σειράς Fourier στο τρίτο όρισμα: εδώ $N = 5$. Προκύπτει έτσι η ακόλουθη απλοποιημένη τριγωνομετρική σειρά Fourier:

```
In[6]:= FTS1[t_] = FourierTrigSeries[f[t], t, 5, FourierParameters → {-1, 1/10}] // Simplify
```

Out[6]=
$$\frac{1}{36\pi^2}\left(300\pi^2 - 3600\,\mathrm{Cos}\left[\frac{\pi t}{5}\right] + 900\,\mathrm{Cos}\left[\frac{2\pi t}{5}\right] - 400\,\mathrm{Cos}\left[\frac{3\pi t}{5}\right] + 225\,\mathrm{Cos}\left[\frac{4\pi t}{5}\right] - 144\,\mathrm{Cos}[\pi t] + 1440\pi\,\mathrm{Sin}\left[\frac{\pi t}{5}\right] - 720\pi\,\mathrm{Sin}\left[\frac{2\pi t}{5}\right] + 480\pi\,\mathrm{Sin}\left[\frac{3\pi t}{5}\right] - 360\pi\,\mathrm{Sin}\left[\frac{4\pi t}{5}\right] + 288\pi\,\mathrm{Sin}[\pi t]\right)$$

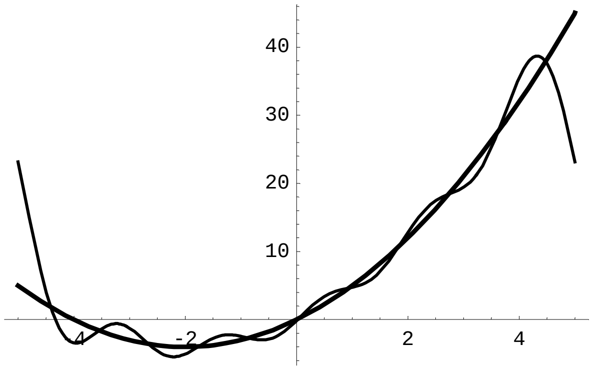
Βλέπουμε πως πρόκειται για μια πλήρη τριγωνομετρική σειρά Fourier με σταθερό όρο (τον πρώτο όρο), συνημιτονικούς όρους (τους επόμενους όρους) και ημιτονικούς όρους (τους τελευταίους όρους) μέχρι την τάξη $N = 5$ της προσεγγίσεώς της. Δείχνουμε και τη γραφική παράστασή της στο διάστημα [−10, 10], δηλαδή στο διάστημα $[-T, T]$ μήκους $2T$, αφού είχαμε επιλέξει την περίοδο $T = 10$. Είναι προφανές (και φαίνεται καθαρά αυτό στην παρακάτω γραφική παράσταση) ότι η σειρά Fourier που βρήκαμε προσεγγίζει ικανοποιητικά τη συνάρτηση μόνο στο διάστημα [−5, 5] (μήκους μιας περιόδου $T = 10$) και όχι και έξω από το διάστημα αυτό: $T < -5$ και $T > 5$. Τούτο ήταν βέβαια αναμενόμενο με την περίοδο $T = 10$ που επιλέξαμε:

```
In[7]:= Plot[{f[t], FTS1[t]}, {t, -10, 10}, PlotStyle → {Thickness[0.008], Thickness[0.006]}];
```



Και τώρα πιο καθαρά το ίδιο σχήμα στο διάστημα [−5, 5] μήκους μιας περιόδου $T = 10$ μόνο:

```
In[8]:= Plot[{f[t], FTS1[t]}, {t, -5, 5}, PlotStyle → {Thickness[0.008], Thickness[0.006]}];
```



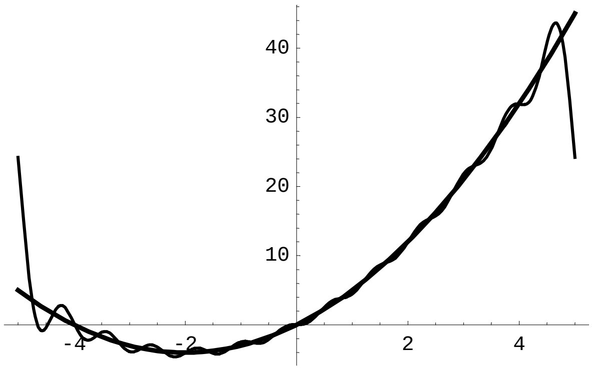
Μπορούμε βέβαια να πετύχουμε μεγαλύτερη ακρίβεια απλά με πιο πολλούς όρους σ' αυτήν την τριγωνομετρική σειρά Fourier (δηλαδή με μεγαλύτερη ανώτερη τάξη *N* στους όρους που περιλαμβάνονται), ας πούμε εδώ με $N = 12$. Ακολουθούν η σειρά Fourier και η γραφική παράστασή της:

```
In[9]:= FTS2[t_] = FourierTrigSeries[f[t], t, 12, FourierParameters → {-1, 1/10}] // Simplify
```

Out[9]=

$$\frac{25}{3} - \frac{100\,\mathrm{Cos}[\frac{\pi t}{5}]}{\pi^2} + \frac{25\,\mathrm{Cos}[\frac{2\pi t}{5}]}{\pi^2} - \frac{100\,\mathrm{Cos}[\frac{3\pi t}{5}]}{9\pi^2} + \frac{25\,\mathrm{Cos}[\frac{4\pi t}{5}]}{4\pi^2} - \frac{4\,\mathrm{Cos}[\pi t]}{\pi^2} + \frac{25\,\mathrm{Cos}[\frac{6\pi t}{5}]}{9\pi^2} - \frac{100\,\mathrm{Cos}[\frac{7\pi t}{5}]}{49\pi^2} + \frac{25\,\mathrm{Cos}[\frac{8\pi t}{5}]}{16\pi^2} - \frac{100\,\mathrm{Cos}[\frac{9\pi t}{5}]}{81\pi^2} + \frac{\mathrm{Cos}[2\pi t]}{\pi^2} - \frac{100\,\mathrm{Cos}[\frac{11\pi t}{5}]}{121\pi^2} + \frac{25\,\mathrm{Cos}[\frac{12\pi t}{5}]}{36\pi^2} + \frac{40\,\mathrm{Sin}[\frac{\pi t}{5}]}{\pi} - \frac{20\,\mathrm{Sin}[\frac{2\pi t}{5}]}{\pi} + \frac{40\,\mathrm{Sin}[\frac{3\pi t}{5}]}{3\pi} - \frac{10\,\mathrm{Sin}[\frac{4\pi t}{5}]}{\pi} + \frac{8\,\mathrm{Sin}[\pi t]}{\pi} - \frac{20\,\mathrm{Sin}[\frac{6\pi t}{5}]}{3\pi} + \frac{40\,\mathrm{Sin}[\frac{7\pi t}{5}]}{7\pi} - \frac{5\,\mathrm{Sin}[\frac{8\pi t}{5}]}{\pi} + \frac{40\,\mathrm{Sin}[\frac{9\pi t}{5}]}{9\pi} - \frac{4\,\mathrm{Sin}[2\pi t]}{\pi} + \frac{40\,\mathrm{Sin}[\frac{11\pi t}{5}]}{11\pi} - \frac{10\,\mathrm{Sin}[\frac{12\pi t}{5}]}{3\pi}$$

```
In[10]:= Plot[{f[t], FTS2[t]}, {t, -5, 5}, PlotStyle → {Thickness[0.008], Thickness[0.006]}];
```



Παρατηρούμε ότι στα σημεία $t = \mp 5$ της ασυνέχειας η σειρά Fourier συγκλίνει στη μέση τιμή των δύο σχετικών οριακών τιμών της συναρτήσεώς μας $f(t)$. Αυτά είναι τα άκρα του διαστήματος $[-T/2,\ T/2]$, εδώ $[-5, 5]$. Δυστυχώς δεν υπάρχει καλύτερη δυνατότητα για τα σημεία ασυνέχειας μιας σειράς Fourier.

Φυσικά για άρτιες συναρτήσεις οι σειρές Fourier περιλαμβάνουν μόνο συνημιτονικούς όρους (και το σταθερό όρο βέβαια, αν υπάρχει και γενικά υπάρχει σταθερός όρος), ενώ για περιττές συναρτήσεις περιλαμβάνουν μόνο ημιτονικούς όρους χωρίς σταθερό όρο.

Ακολουθούν δύο απλά παραδείγματα: ένα για άρτια συνάρτηση (επομένως και σειρά Fourier: με συνημιτονικούς μόνο όρους) και ένα για περιττή συνάρτηση (άρα και σειρά Fourier: με ημιτονικούς μόνο όρους):

```
In[11]:= FTSC[t_] =
          FourierTrigSeries[Cosh[t], t, 6, FourierParameters → {-1, 1 / (2 π)}] // FullSimplify
```

$$\text{Out[11]=}\ \frac{1}{40885\,\pi}\left(\left(37\left(-221\left(-5+5\,\text{Cos}[t]-2\,\text{Cos}[2\,t]+\text{Cos}[3\,t]\right)+130\,\text{Cos}[4\,t]-85\,\text{Cos}[5\,t]\right)+2210\,\text{Cos}[6\,t]\right)\text{Sinh}[\pi]\right)$$

```
In[12]:= FTSS[t_] =
          FourierTrigSeries[Sinh[t], t, 6, FourierParameters → {-1, 1 / (2 π)}] // FullSimplify
```

$$\text{Out[12]=}\ \frac{1}{40885\,\pi}\left(\left(481\left(85\,\text{Sin}[t]-68\,\text{Sin}[2\,t]+51\,\text{Sin}[3\,t]-40\,\text{Sin}[4\,t]\right)+15725\,\text{Sin}[5\,t]-13260\,\text{Sin}[6\,t]\right)\text{Sinh}[\pi]\right)$$

■ ΕΝΤΟΛΗ F2: ΤΡΙΓΩΝΟΜΕΤΡΙΚΗ ΣΕΙΡΑ FOURIER (ΑΡΙΘΜΗΤΙΚΑ)

NFourierTrigSeries[*Συνάρτηση, Μεταβλητή, ΤάξηΠροσεγγίσεωςΤηςΣειράςFourier,*

FourierParameters → {−1, 1/*Περίοδος*}]

Πρόκειται για την απόλυτα αντίστοιχη εντολή της προηγούμενης εντολής **FourierTrigSeries**, αλλά τώρα για τον κατευθείαν αριθμητικό υπολογισμό της τριγωνομετρικής σειράς Fourier χωρίς καν να χρειάζεται η εντολή **N**. Αυτό είναι χρήσιμο σε περιπτώσεις που δεν είναι δυνατόν να υπολογισθεί αναλυτικά η σειρά Fourier, δηλαδή δε μπορούν να υπολογισθούν αναλυτικά τα ολοκληρώματα στους συντελεστές της. Παράδειγμα: Αρχικά αναλυτικός υπολογισμός της τριγωνομετρικής σειράς Fourier της εκθετικής συναρτήσεως e^t στο διάστημα $[-\pi, \pi]$, έπειτα από την αρχή ο ίδιος υπολογισμός, αλλά τελικά με αριθμητικό αποτέλεσμα με την εντολή **N** και τέλος πάλι ο αριθμητικός υπολογισμός της ίδιας σειράς Fourier, αλλά με διαφορετικό τρόπο: με τη χρήση αυτής της εντολής **NFourierTrigSeries**:

```
In[13]:= TrigFSE[t_] =
          FourierTrigSeries[Exp[t], t, 6, FourierParameters → {-1, 1 / (2 π)}] // FullSimplify
```

$$\text{Out[13]=}\ \frac{1}{40885\,\pi}\left(\left(-40885\,\text{Cos}[t]+16354\,\text{Cos}[2\,t]-8177\,\text{Cos}[3\,t]+4810\,\text{Cos}[4\,t]-3145\,\text{Cos}[5\,t]+2210\,\text{Cos}[6\,t]+481\left(85+85\,\text{Sin}[t]-68\,\text{Sin}[2\,t]+51\,\text{Sin}[3\,t]-40\,\text{Sin}[4\,t]\right)+15725\,\text{Sin}[5\,t]-13260\,\text{Sin}[6\,t]\right)\text{Sinh}[\pi]\right)$$

```
In[14]:= NTrigFSE1[t_] =
          FourierTrigSeries[Exp[t], t, 6, FourierParameters → {-1, 1 / (2 π)}] // N

Out[14]= 3.67608 - 3.67608 Cos[t] + 1.47043 Cos[2. t] - 0.735216 Cos[3. t] + 0.43248 Cos[4. t] -
          0.282775 Cos[5. t] + 0.198707 Cos[6. t] + 3.67608 Sin[t] - 2.94086 Sin[2. t] +
          2.20565 Sin[3. t] - 1.72992 Sin[4. t] + 1.41388 Sin[5. t] - 1.19224 Sin[6. t]

In[15]:= NTrigFSE2[t_] = NFourierTrigSeries[Exp[t], t, 6, FourierParameters → {-1, 1 / (2 π)}]

Out[15]= 3.67608 - 3.67608 Cos[t] + 1.47043 Cos[2 t] - 0.735216 Cos[3 t] + 0.43248 Cos[4 t] -
          0.282775 Cos[5 t] + 0.198707 Cos[6 t] + 3.67608 Sin[t] - 2.94086 Sin[2 t] +
          2.20565 Sin[3 t] - 1.72992 Sin[4 t] + 1.41388 Sin[5 t] - 1.19224 Sin[6 t]
```

Η διαφορά **difference1** των δύο αριθμητικών τριγωνομετρικών σειρών Fourier είναι εδώ απειροελάχιστη (της τάξεως της ακρίβειας των αριθμητικών υπολογισμών με τη *Mathematica* στον υπολογιστή: περίπου 15 σημαντικά ψηφία) και μάλιστα μηδενίζεται απλά με τη χρήση της εντολής **Chop**:

```
In[16]:= difference1 = NTrigFSE1[t] - NTrigFSE2[t] // N // Chop

Out[16]= 0
```

■ ΕΝΤΟΛΗ F3: ΕΚΘΕΤΙΚΗ Ή ΜΙΓΑΔΙΚΗ ΣΕΙΡΑ FOURIER

FourierSeries[*Συνάρτηση, Μεταβλητή, ΤάξηΠροσεγγίσεωςΤηςΣειράςFourier,*

FourierParameters → {−1, 1/*Περίοδος*}]

Δημιουργεί την εκθετική ή μιγαδική σειρά Fourier της συναρτήσεως η οποία δίνεται στο πρώτο όρισμα ως προς τη μεταβλητή (π.χ. *t* ή *x*) που δίνεται στο δεύτερο όρισμα με όρους τάξεως μέχρι και *N* (μεταβλητή η οποία δίνεται στο τρίτο όρισμα). Ακολουθεί η επιλογή **FourierParameters** με λίστα δύο στοιχείων: το πρώτο στοιχείο είναι το -1 και το δεύτερο στοιχείο το αντίστροφο $1/T$ της περιόδου T της σειράς Fourier. Σημειώνεται πως πρόκειται για μια εντολή απόλυτα ανάλογη της προπροηγούμενης εντολής **FourierTrigSeries**, αλλά τώρα για την εκθετική ή μιγαδική σειρά Fourier. Υπενθυμίζεται επίσης ότι για τη χρήση οποιασδήποτε εντολής αυτού του notebook πρέπει να έχει προηγουμένως κληθεί (να έχει φορτωθεί) το πακέτο **Calculus'FourierTransform'** της *Mathematica*, όπως έχει ήδη αναφερθεί. Παράδειγμα θα δοθεί στην αμέσως επόμενη εντολή: την αριθμητική μορφή της παρούσας εντολής.

■ ΕΝΤΟΛΗ F4: ΕΚΘΕΤΙΚΗ Ή ΜΙΓΑΔΙΚΗ ΣΕΙΡΑ FOURIER (ΑΡΙΘΜΗΤΙΚΑ)

NFourierSeries[*Συνάρτηση, Μεταβλητή, ΤάξηΠροσεγγίσεωςΤηςΣειράςFourier,*

FourierParameters → {−1, 1/*Περίοδος*}]

Πρόκειται για την απόλυτα αντίστοιχη εντολή της προηγούμενης εντολής **FourierSeries**, αλλά τώρα για τον κατευθείαν αριθμητικό υπολογισμό της εκθετικής ή μιγαδικής σειράς Fourier χωρίς καν να χρειάζεται η εντολή **N**. Παράδειγμα: Αρχικά αναλυτικός υπολογισμός της σειράς Fourier της εκθετικής συναρτήσεως e^t στο διάστημα $[-\pi, \pi]$, έπειτα από την αρχή ο ίδιος υπολογισμός, αλλά με αριθμητικό αποτέλεσμα στο τέλος με την εντολή **N** και τέλος πάλι ο αριθμητικός υπολογισμός της ίδιας σειράς Fourier, αλλά με διαφορετικό τρόπο: με τη χρήση της παρούσας εντολής **NFourierTrigSeries**:

```
In[17]:= ExpFSE[t_] =
          FourierSeries[Exp[t], t, 4, FourierParameters → {-1, 1 / (2 π)}] // Simplify
```

Out[17]=

$$\frac{1}{\pi}\left(\left(\frac{1}{340}+\frac{i}{340}\right) e^{-\pi-4 i t}\left(-1+e^{2\pi}\right)\left((-15-25 i)+(17+34 i) e^{i t}-(17+51 i) e^{2 i t}+85 i e^{3 i t}+(85-85 i) e^{4 i t}-85 e^{5 i t}+(51+17 i) e^{6 i t}-(34+17 i) e^{7 i t}+(25+15 i) e^{8 i t}\right)\right)$$

```
In[18]:= NExpFSE1[t_] =
          FourierSeries[Exp[t], t, 4, FourierParameters → {-1, 1 / (2 π)}] // N // Chop
```

Out[18]=

$$3.67608-(1.83804-1.83804 i)\, 2.71828^{-1. i t}-(1.83804+1.83804 i)\, 2.71828^{1. i t}+(0.735216-1.47043 i)\, 2.71828^{-2. i t}+(0.735216+1.47043 i)\, 2.71828^{2. i t}-(0.367608-1.10282 i)\, 2.71828^{-3. i t}-(0.367608+1.10282 i)\, 2.71828^{3. i t}+(0.21624-0.86496 i)\, 2.71828^{-4. i t}+(0.21624+0.86496 i)\, 2.71828^{4. i t}$$

```
In[19]:= NExpFSE2[t_] = NFourierSeries[Exp[t], t, 4, FourierParameters → {-1, 1 / (2 π)}]
```

```
Out[19]= 3.67608 - (1.83804 - 1.83804 ⅈ) ⅇ^(-ⅈ t) - (1.83804 + 1.83804 ⅈ) ⅇ^(ⅈ t) +
  (0.735216 - 1.47043 ⅈ) ⅇ^(-2 ⅈ t) + (0.735216 + 1.47043 ⅈ) ⅇ^(2 ⅈ t) - (0.367608 - 1.10282 ⅈ) ⅇ^(-3 ⅈ t) -
  (0.367608 + 1.10282 ⅈ) ⅇ^(3 ⅈ t) + (0.21624 - 0.86496 ⅈ) ⅇ^(-4 ⅈ t) + (0.21624 + 0.86496 ⅈ) ⅇ^(4 ⅈ t)
```

Η διαφορά **difference2** των δύο αριθμητικών αποτελεσμάτων είναι και εδώ απειροελάχιστη (της τάξεως της ακρίβειας των υπολογισμών της *Mathematica*) και μηδενίζεται με τη χρήση και της εντολής **Chop**:

```
In[20]:= difference2 = NExpFSE1[t] - NExpFSE2[t] // N // FullSimplify
```

```
Out[20]= -4.44089×10^-16 + (0. + 0. ⅈ) Cos[1. t] + (6.66134×10^-16 + 0. ⅈ) Cos[2. t] -
  (9.99201×10^-16 + 0. ⅈ) Cos[3. t] + (8.32667×10^-16 + 0. ⅈ) Cos[4. t] -
  (1.77636×10^-15 + 0. ⅈ) Sin[1. t] + (0. + 0. ⅈ) Sin[2. t] -
  (8.88178×10^-16 + 0. ⅈ) Sin[3. t] + (6.66134×10^-16 + 0. ⅈ) Sin[4. t]
```

```
In[21]:= {Chop[difference2], difference2 // Chop, difference2 == 0 // Chop}
```

```
Out[21]= {0, 0, True}
```

Ας παρατηρήσουμε τέλος την ουσιαστική σύμπτωση της παρούσας εκθετικής ή μιγαδικής σειράς Fourier **ExpFSE[t]** και της αντίστοιχης τριγωνομετρικής σειράς Fourier, εδώ όμως με $N = 4$:

```
In[22]:= TrigFSE[t_] =
   FourierTrigSeries[Exp[t], t, 4, FourierParameters → {-1, 1 / (2 π)}] // FullSimplify
```

```
Out[22]= 1/(85 π) ((85 - 85 Cos[t] + 34 Cos[2 t] - 17 Cos[3 t] +
     10 Cos[4 t] + 85 Sin[t] - 68 Sin[2 t] + 51 Sin[3 t] - 40 Sin[4 t]) Sinh[π])
```

Πραγματικά έχουμε μηδενική διαφορά

```
In[23]:= difference3a = ExpFSE[t] - TrigFSE[t] // ExpToTrig // Simplify
```

```
Out[23]= 0
```

αφού βέβαια μετατρέψαμε όλους τους όρους σε τριγωνομετρική μορφή και απλοποιήσαμε το αποτέλεσμα. Εναλλακτικά και χωρίς μάλιστα η *Mathematica* να χρειάζεται καν την εντολή **Simplify**

```
In[24]:= difference3b = ExpFSE[t] - TrigFSE[t] // TrigToExp
```

```
Out[24]= 0
```

Τα ίδια συμβαίνουν και αν εργασθούμε με τις καθαρά αριθμητικές αντίστοιχες σειρές **NExpFSE2[t]** και

```
In[25]:= NTrigFSE2[t_] = NFourierTrigSeries[Exp[t], t, 4, FourierParameters → {-1, 1 / (2 π)}]
```

```
Out[25]= 3.67608 - 3.67608 Cos[t] + 1.47043 Cos[2 t] - 0.735216 Cos[3 t] + 0.43248 Cos[4 t] +
  3.67608 Sin[t] - 2.94086 Sin[2 t] + 2.20565 Sin[3 t] - 1.72992 Sin[4 t]
```

με διαφορά μηδενική, αφού βέβαια χρησιμοποιηθεί και η εντολή **Chop**, για να μηδενισθούν τα σφάλματα αριθμητικών στρογγυλεύσεων. Αυτό ισχύει με μετατροπή της διαφοράς είτε σε τριγωνομετρική μορφή με την εντολή **ExpToTrig** είτε σε εκθετική ή μιγαδική μορφή με την αντίστροφη εντολή **TrigToExp**:

```
In[26]:= difference4a = NExpFSE2[t] - NTrigFSE2[t] // ExpToTrig // Chop
```

```
Out[26]= 0
```

```
In[27]:= difference4b = NExpFSE2[t] - NTrigFSE2[t] // TrigToExp // Chop
Out[27]= 0
```

■ ΕΝΤΟΛΗ F5: ΣΥΝΗΜΙΤΟΝΙΚΟΣ ΣΥΝΤΕΛΕΣΤΗΣ ΣΕΙΡΑΣ FOURIER

FourierCosCoefficient[*Συνάρτηση, Μεταβλητή, ΤάξηΤουΣυνημιτονικούΣυντελεστήΤηςΣειράςFourier,* FourierParameters → {−1, 1 / *Περίοδος*}]

Υπολογίζει το συνημιτονικό συντελεστή τάξεως n που δίνεται στο τρίτο όρισμα στην τριγωνομετρική σειρά Fourier της συναρτήσεως που δίνεται στο πρώτο όρισμα ως προς τη μεταβλητή (π.χ. t ή x) που δίνεται στο δεύτερο όρισμα. Ακολουθεί πάλι η επιλογή **FourierParameters** με λίστα από δύο στοιχεία: το πρώτο στοιχείο είναι το -1 και το δεύτερο στοιχείο το αντίστροφο $1/T$ της περιόδου T της τριγωνομετρικής σειράς Fourier. Αντίστοιχη είναι και η μεθεπόμενη εντολή **FourierSinCoefficient**, η οποία περιλαμβάνει και τα σχετικά παραδείγματα.

■ ΕΝΤΟΛΗ F6: ΣΥΝΗΜΙΤΟΝΙΚΟΣ ΣΥΝΤΕΛΕΣΤΗΣ ΣΕΙΡΑΣ FOURIER (ΑΡΙΘΜΗΤΙΚΑ)

NFourierCosCoefficient[*Συνάρτηση, Μεταβλητή, ΤάξηΤουΣυνημιτονικούΣυντελεστήΤηςΣειράςFourier,* FourierParameters → {−1, 1 / *Περίοδος*}]

Πρόκειται για την απόλυτα αντίστοιχη εντολή της προηγούμενης εντολής **FourierCosCoefficient**, αλλά τώρα για τον κατευθείαν αριθμητικό υπολογισμό του συνημιτονικού συντελεστή τάξεως n. Παραδείγματα θα δοθούν στη μεθεπόμενη εντολή **NFourierSinCoefficient**.

■ ΕΝΤΟΛΗ F7: ΗΜΙΤΟΝΙΚΟΣ ΣΥΝΤΕΛΕΣΤΗΣ ΣΕΙΡΑΣ FOURIER

FourierSinCoefficient[*Συνάρτηση, Μεταβλητή, ΤάξηΤουΗμιτονικούΣυντελεστήΤηςΣειράςFourier,* FourierParameters → {−1, 1 / *Περίοδος*}]

Υπολογίζει τον ημιτονικό συντελεστή τάξεως n που δίνεται στο τρίτο όρισμα στην τριγωνομετρική σειρά Fourier της συναρτήσεως που δίνεται στο πρώτο όρισμα ως προς τη μεταβλητή (π.χ. t ή x) που δίνεται στο δεύτερο όρισμα. Ακολουθεί και πάλι η επιλογή **FourierParameters** με λίστα δύο στοιχείων: το πρώτο στοιχείο είναι το -1 και το δεύτερο στοιχείο είναι το αντίστροφο $1/T$ της περιόδου T της σειράς Fourier. Επανερχόμαστε στο παράδειγμα της συναρτήσεως $f(t) = 4t + t^2$, που είχαμε στην πρώτη εντολή **FourierTrigSeries** αυτού του notebook (με $T = 10$) για την τριγωνομετρική σειρά Fourier:

```
In[28]:= FTS1[t]
```

Out[28]=
$$\frac{1}{36\pi^2}\left(300\pi^2 - 3600\,\mathrm{Cos}\left[\frac{\pi t}{5}\right] + 900\,\mathrm{Cos}\left[\frac{2\pi t}{5}\right] - 400\,\mathrm{Cos}\left[\frac{3\pi t}{5}\right] + 225\,\mathrm{Cos}\left[\frac{4\pi t}{5}\right] - 144\,\mathrm{Cos}[\pi t] + 1440\pi\,\mathrm{Sin}\left[\frac{\pi t}{5}\right] - 720\pi\,\mathrm{Sin}\left[\frac{2\pi t}{5}\right] + 480\pi\,\mathrm{Sin}\left[\frac{3\pi t}{5}\right] - 360\pi\,\mathrm{Sin}\left[\frac{4\pi t}{5}\right] + 288\pi\,\mathrm{Sin}[\pi t]\right)$$

Παραδείγματα (σε δύο λίστες με την εντολή **Table**) για συγκεκριμένους συνημιτονικούς (λίστα **tb1**) και ημιτονικούς (λίστα **tb2**) συντελεστές αυτής της τριγωνομετρικής σειράς Fourier. (Δε θεωρείται γνωστή!)

```
In[29]:= tb1 = Table[
           a[n] = FourierCosCoefficient[f[t], t, n, FourierParameters → {-1, 1/10}], {n, 0, 5}]
```

Out[29]= $\left\{\frac{25}{3}, -\frac{100}{\pi^2}, \frac{25}{\pi^2}, -\frac{100}{9\pi^2}, \frac{25}{4\pi^2}, -\frac{4}{\pi^2}\right\}$

```
In[30]:= tb2 = Table[
           b[n] = FourierSinCoefficient[f[t], t, n, FourierParameters → {-1, 1/10}], {n, 1, 5}]
```

Out[30]= $\left\{\frac{40}{\pi}, -\frac{20}{\pi}, \frac{40}{3\pi}, -\frac{10}{\pi}, \frac{8}{\pi}\right\}$

Και φυσικά υπό αυτές τις συνθήκες μπορούμε να επαληθεύσουμε και όλους αυτούς τους συντελεστές ότι μας οδηγούν πραγματικά στην τριγωνομετρική σειρά Fourier **FTS1[t]**, την οποία έχουμε ήδη βρει, χωρίς βέβαια να λησμονηθεί κανένας από τους αναγκαίους τριγωνομετρικούς όρους ούτε και ο σταθερός όρος:

```
In[31]:= FTS1[t] == a[0] + Sum[a[n] Cos[n π t / 5] + b[n] Sin[n π t / 5], {n, 1, 5}] // Simplify

Out[31]= True
```

■ ΕΝΤΟΛΗ F8: ΗΜΙΤΟΝΙΚΟΣ ΣΥΝΤΕΛΕΣΤΗΣ ΣΕΙΡΑΣ FOURIER (ΑΡΙΘΜΗΤΙΚΑ)

NFourierSinCoefficient[*Συνάρτηση, Μεταβλητή, ΤάξηΤουΗμιτονικούΣυντελεστήΤηςΣειράςFourier,* FourierParameters → {−1, 1/*Περίοδος*}]

Πρόκειται για την απόλυτα αντίστοιχη εντολή της προηγούμενης εντολής **FourierSinCoefficient**. Τώρα όμως γίνεται ο κατευθείαν αριθμητικός υπολογισμός του ημιτονικού συντελεστή τάξεως *n*. Παραδείγματα των δύο εντολών **NFourierCosCoefficient** και **NFourierSinCoefficient** για την αρχική μας συνάρτηση $f(t) = 4\,t + t^2$, όπου μάλιστα παρατηρούμε τη συμφωνία των αποτελεσμάτων με τα αντίστοιχα αναλυτικά αποτελέσματα **tb1** και **tb2** στα παραδείγματα της αμέσως προηγούμενης εντολής, αφού βέβαια τα μετατρέψουμε και αυτά σε αριθμητικά με τη χρήση της γνωστής μας εντολής **N**:

```
In[32]:= tb1n =
          Table[NFourierCosCoefficient[f[t], t, n, FourierParameters → {-1, 1/10}], {n, 0, 5}]

Out[32]= {8.33333, -10.1321, 2.53303, -1.12579, 0.633257, -0.405285}

In[33]:= {tb1 // N, N[tb1] == tb1n}

Out[33]= {{8.33333, -10.1321, 2.53303, -1.12579, 0.633257, -0.405285}, True}

In[34]:= tb2n =
          Table[NFourierSinCoefficient[f[t], t, n, FourierParameters → {-1, 1/10}], {n, 1, 5}]

Out[34]= {12.7324, -6.3662, 4.24413, -3.1831, 2.54648}

In[35]:= {tb2 // N, N[tb2] == tb2n}

Out[35]= {{12.7324, -6.3662, 4.24413, -3.1831, 2.54648}, True}
```

■ ΕΝΤΟΛΗ F9: ΣΥΝΤΕΛΕΣΤΗΣ ΕΚΘΕΤΙΚΗΣ ΣΕΙΡΑΣ FOURIER

FourierCoefficient[*Συνάρτηση, Μεταβλητή, −ΤάξηΤουΣυντελεστήΤηςΕκθετικήςΣειράςFourier,*

FourierParameters → {−1, 1 / *Περίοδος*}]

Υπολογίζει το συντελεστή της εκθετικής ή μιγαδικής σειράς Fourier τάξεως n που δίνεται στο τρίτο όρισμα (αλλά με μείον: $-n$ αντί για n εξαιτίας ελαφρά διαφορετικού ορισμού: με $-n$ στον εκθέτη, που ακολουθεί εδώ η *Mathematica*) της συναρτήσεως που δίνεται στο πρώτο όρισμα ως προς τη μεταβλητή (π.χ. t ή x) η οποία δίνεται στο δεύτερο όρισμα. Ακολουθεί ξανά η επιλογή **FourierParameters** με λίστα δύο στοιχείων: το πρώτο στοιχείο είναι το -1 και το δεύτερο στοιχείο το αντίστροφο $1/T$ της περιόδου T της εκθετικής ή μιγαδικής σειράς Fourier. Παράδειγμα για τη συνάρτηση $f(t) = 4t + t^2$:

```
In[36]:= tb3 = Table[
           c[n] = FourierCoefficient[f[t], t, -n, FourierParameters → {-1, 1 / 10}] // Simplify,
           {n, -5, 5}]
```

Out[36]= $\left\{\frac{-2+4 i \pi}{\pi^2}, \frac{5(5-8 i \pi)}{8\pi^2}, \frac{10 i (5 i+6\pi)}{9\pi^2}, \frac{5(5-4 i \pi)}{2\pi^2}, \frac{-50+20 i \pi}{\pi^2}, \frac{25}{3}, -\frac{50+20 i \pi}{\pi^2}, \frac{5(5+4 i \pi)}{2\pi^2}, -\frac{50+60 i \pi}{9\pi^2}, \frac{5(5+8 i \pi)}{8\pi^2}, -\frac{2+4 i \pi}{\pi^2}\right\}$

ενώ για την ίδια ακριβώς συνάρτηση $f(t)$ είχαμε βρει προηγουμένως: στην προπροηγούμενη εντολή:

```
In[37]:= tb1
```

Out[37]= $\left\{\frac{25}{3}, -\frac{100}{\pi^2}, \frac{25}{\pi^2}, -\frac{100}{9\pi^2}, \frac{25}{4\pi^2}, -\frac{4}{\pi^2}\right\}$

```
In[38]:= tb2
```

Out[38]= $\left\{\frac{40}{\pi}, -\frac{20}{\pi}, \frac{40}{3\pi}, -\frac{10}{\pi}, \frac{8}{\pi}\right\}$

Μπορούμε έτσι να υπολογίσουμε με δεύτερο τρόπο τους συντελεστές της εκθετικής ή μιγαδικής σειράς Fourier που εξετάζουμε. (Δεν ξεχνάμε βέβαια και τους τύπους για τους συντελεστές της c_n από την Ενότητα A17.3 του Κεφαλαίου A17 του Μέρους A των διδακτικών βιβλίων.)

```
In[39]:= tb4a = Table[d[n] = (a[n] - ⅈ b[n]) / 2 // Simplify, {n, 5, 1, -1}]
```

Out[39]= $\left\{-\frac{2+4 i \pi}{\pi^2}, \frac{5(5+8 i \pi)}{8\pi^2}, -\frac{50+60 i \pi}{9\pi^2}, \frac{5(5+4 i \pi)}{2\pi^2}, -\frac{50+20 i \pi}{\pi^2}\right\}$

και παρατηρούμε εύλογα ότι

```
In[40]:= verification1 = Table[c[n] == d[n], {n, 1, 5}]

Out[40]= {True, True, True, True, True}
```

Και παραπέρα

```
In[41]:= tb4b = Table[d[-n] = (a[n] + ⅈ b[n]) / 2 // Simplify, {n, 1, 5}]
```

Out[41]= $\left\{\frac{-50+20 i \pi}{\pi^2}, \frac{5(5-4 i \pi)}{2\pi^2}, \frac{10 i (5 i+6\pi)}{9\pi^2}, \frac{5(5-8 i \pi)}{8\pi^2}, \frac{-2+4 i \pi}{\pi^2}\right\}$

και παρατηρούμε και πάλι ότι

```
In[42]:= verification2 = Table[c[n] == d[n], {n, -5, -1}]

Out[42]= {True, True, True, True, True}
```

Τέλος για το σταθερό όρο (με $n = 0$) έχουμε

```
In[43]:= verification3 = c[0] == a[0]

Out[43]= True
```

Ολοκληρώθηκε λοιπόν η επαλήθευση της πλήρους ισοδυναμίας της εκθετικής ή μιγαδικής σειράς Fourier με την αντίστοιχη τριγωνομετρική σειρά Fourier στο παρόν παράδειγμα ως προς τους συντελεστές τους για n από -5 έως 5 στη μιγαδική ή εκθετική σειρά Fourier.

■ ΕΝΤΟΛΗ F10: ΣΥΝΤΕΛΕΣΤΗΣ ΕΚΘΕΤΙΚΗΣ ΣΕΙΡΑΣ FOURIER (ΑΡΙΘΜΗΤΙΚΑ)

NFourierCoefficient[*Συνάρτηση, Μεταβλητή, −ΤάξηΤουΣυντελεστήΤηςΕκθετικήςΣειράςFourier,* FourierParameters → {−1, 1/*Περίοδος*}]

Πρόκειται για την απόλυτα αντίστοιχη εντολή της προηγούμενης εντολής **FourierCoefficient**, αλλά τώρα για τον κατευθείαν αριθμητικό υπολογισμό του συντελεστή τάξεως n της εκθετικής ή μιγαδικής σειράς Fourier. Ακολουθεί το παράδειγμα της προηγούμενης εντολής, τώρα όμως καθαρά αριθμητικά:

```
In[44]:= tb5 = Table[
           c[n] = NFourierCoefficient[f[t], t, -n, FourierParameters → {-1, 1/10}], {n, -5, 5}]

Out[44]= {-0.202642 + 1.27324 i, 0.316629 - 1.59155 i, -0.562895 + 2.12207 i, 1.26651 - 3.1831 i,
          -5.06606 + 6.3662 i, 8.33333, -5.06606 - 6.3662 i, 1.26651 + 3.1831 i,
          -0.562895 - 2.12207 i, 0.316629 + 1.59155 i, -0.202642 - 1.27324 i}
```

ενώ προηγουμένως είχαμε βρει για το ίδιο ακριβώς παράδειγμα

```
In[45]:= tb3
```

Out[45]= $\left\{\frac{-2+4\,i\,\pi}{\pi^2}, \frac{5\,(5-8\,i\,\pi)}{8\,\pi^2}, \frac{10\,i\,(5\,i+6\,\pi)}{9\,\pi^2}, \frac{5\,(5-4\,i\,\pi)}{2\,\pi^2}, \frac{-50+20\,i\,\pi}{\pi^2}, \frac{25}{3}, -\frac{50+20\,i\,\pi}{\pi^2}, \frac{5\,(5+4\,i\,\pi)}{2\,\pi^2}, -\frac{50+60\,i\,\pi}{9\,\pi^2}, \frac{5\,(5+8\,i\,\pi)}{8\,\pi^2}, -\frac{2+4\,i\,\pi}{\pi^2}\right\}$

Επομένως οι διαφορές των αντίστοιχων συντελεστών c_n πρέπει να είναι μηδενικές. Πραγματικά έχουμε διαφορές που οφείλονται αποκλειστικά στα σφάλματα στρογγυλεύσεως, όπως εύκολα παρατηρούμε:

```
In[46]:= differences1 = tb5 - N[tb3]
```

Out[46]= $\{1.63758\times10^{-15} - 1.11022\times10^{-15}\,i,\ -1.27676\times10^{-15} - 6.66134\times10^{-16}\,i,\ -1.11022\times10^{-16} + 1.33227\times10^{-15}\,i,\ -6.66134\times10^{-16} + 0.\,i,\ 0. + 1.77636\times10^{-15}\,i,\ 1.77636\times10^{-15},\ 0. - 1.77636\times10^{-15}\,i,\ -6.66134\times10^{-16} + 0.\,i,\ -1.11022\times10^{-16} - 1.33227\times10^{-15}\,i,\ -1.27676\times10^{-15} + 6.66134\times10^{-16}\,i,\ 1.63758\times10^{-15} + 1.11022\times10^{-15}\,i\}$

```
In[47]:= {differences2 = tb5 - N[tb3] // Chop, "και απλούστερα", tb5 == tb3}

Out[47]= {{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}, και απλούστερα, True}
```

■ *Notebook E17*
ΕΝΤΟΛΕΣ ΓΙΑ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΥΣ LAPLACE ΚΑΙ FOURIER

4 ΕΝΤΟΛΕΣ: O1. LaplaceTransform, O2. InverseLaplaceTransform, O3. FourierTransform, O4. InverseFourierTransform

■ ΕΝΤΟΛΗ O1: ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ LAPLACE

LaplaceTransform[*Συνάρτηση ή ΛίσταΣυναρτήσεων ή Εξίσωση ή ΛίσταΕξισώσεων, ΑρχικήΜεταβλητή, ΜεταβλητήΜετασχηματισμούLaplace*]

Υπολογίζει το μετασχηματισμό Laplace μιας συναρτήσεως ή λίστας συναρτήσεων ή εξισώσεως ή λίστας εξισώσεων. Συχνά η αρχική μεταβλητή είναι το *t* ή το *x* και η μεταβλητή του μετασχηματισμού Laplace είναι το *s*. Η μέθοδος του μετασχηματισμού Laplace είναι χρήσιμη: (α) Στις γραμμικές συνήθεις διαφορικές εξισώσεις, (β) Στις γραμμικές διαφορικές εξισώσεις με μερικές παραγώγους και (γ) σε μερικές γραμμικές ολοκληρωτικές εξισώσεις Volterra. Παραδείγματα απλών μετασχηματισμών Laplace:

In[1]:= `{LaplaceTransform[Cos[ω t], t, s] , LaplaceTransform[t^3 Cos[ω t] Sinh[ω t], t, s]}`

Out[1]= $\left\{\frac{s}{s^2+\omega^2}, \frac{24\, s\, \omega\, (s^{10} - 10\, s^8\, \omega^2 - 40\, s^6\, \omega^4 + 80\, s^4\, \omega^6 + 80\, s^2\, \omega^8 - 32\, \omega^{10})}{(s^4 + 4\, \omega^4)^4}\right\}$

Με το μετασχηματισμό Laplace επιτυγχάνεται η μετατροπή μιας συνήθους γραμμικής διαφορικής εξισώσεως (είτε ομογενούς είτε μη ομογενούς) με σταθερούς συντελεστές σε γραμμική (πρωτοβάθμια) αλγεβρική εξίσωση ως προς το μετασχηματισμό Laplace της άγνωστης συναρτήσεως της συνήθους διαφορικής εξισώσεως. Το πιο κάτω σχετικό παράδειγμα αφορά στην κλασική διαφορική εξίσωση του αρμονικού ταλαντωτή (το πιο απλό και μάλιστα μονοβάθμιο μηχανικό σύστημα του Πολιτικού Μηχανικού!) με απόσβεση σε εξαναγκασμένες ταλαντώσεις (με την ανηγμένη στη μάζα φόρτιση στο δεξιό μέλος):

In[2]:= `LT = LaplaceTransform[u''[t] + 2 ξ ω0 u'[t] + ω0^2 u[t] == p[t] / m, t, s]`

Out[2]= $s^2\, \text{LaplaceTransform}[u[t], t, s] + \text{LaplaceTransform}[u[t], t, s]\, \omega_0^2 + 2\, \xi\, \omega_0\, (s\, \text{LaplaceTransform}[u[t], t, s] - u[0]) - s\, u[0] - u'[0] == \frac{\text{LaplaceTransform}[p[t], t, s]}{m}$

Μερικές φορές η χρήση ενός απλούστερου συμβόλου για ένα μετασχηματισμό Laplace είναι χρήσιμη για την οπτικά απλούστερη εμφάνιση της μετασχηματισμένης κατά Laplace εξισώσεως:

In[3]:= `LTs = LT /. {LaplaceTransform[u[t], t, s] → U[s], LaplaceTransform[p[t], t, s] → P[s]}`

Out[3]= $-s\, u[0] + s^2\, U[s] + \omega_0^2\, U[s] + 2\, \xi\, \omega_0\, (-u[0] + s\, U[s]) - u'[0] == \frac{P[s]}{m}$

Και η ακόμη πιο απλή τελική μορφή της ίδιας γραμμικής αλγεβρικής εξισώσεως:

In[4]:= `LTs1 = Collect[LTs[[1]], U[s]] == LTs[[2]] // Simplify`

Out[4]= $(s^2 + 2\, s\, \xi\, \omega_0 + \omega_0^2)\, U[s] == \frac{P[s]}{m} + s\, u[0] + 2\, \xi\, \omega_0\, u[0] + u'[0]$

καθώς και η λύση της

```
In[5]:= sol = Solve[LTs1, U[s]]
```

$$\text{Out[5]=}\ \left\{\left\{U[s] \to \frac{\frac{P[s]}{m} + s\,u[0] + 2\,\xi\,\omega_0\,u[0] + u'[0]}{s^2 + 2\,s\,\xi\,\omega_0 + \omega_0^2}\right\}\right\}$$

Η μοναδιαία βηματική συνάρτηση του Heaviside $H(t)$ (που συμβολίζεται με **UnitStep[t]** στη *Mathematica*) καθώς και η ωστική (ή κρουστική) συνάρτηση δέλτα του Dirac $\delta(t)$ (που συμβολίζεται με **DiracDelta[t]** στη *Mathematica*) παρουσιάζουν ενδιαφέρον στο μετασχηματισμό Laplace, όπως συμβαίνει και σε ποικίλες εφαρμογές του Πολιτικού Μηχανικού. Οι μετασχηματισμοί Laplace τους βρίσκονται με την εντολή

```
In[6]:= LaplaceTransform[{UnitStep[t], DiracDelta[t]}, t, s]
```

$$\text{Out[6]=}\ \left\{\frac{1}{s},\ 1\right\}$$

Είδαμε ήδη πως με το μετασχηματισμό Laplace επιτυγχάνεται η μετατροπή μιας συνήθους γραμμικής διαφορικής εξισώσεως (είτε ομογενούς είτε μη ομογενούς) με σταθερούς συντελεστές σε γραμμική αλγεβρική εξίσωση. Ανάλογα κατορθώνεται και η μετατροπή μιας γραμμικής διαφορικής εξισώσεως με μερικές παραγώγους με σταθερούς συντελεστές και δύο ανεξάρτητες μεταβλητές σε συνήθη διαφορική εξίσωση ως προς το μετασχηματισμό Laplace της άγνωστης συναρτήσεως της διαφορικής εξισώσεως με μερικές παραγώγους. Το πιο κάτω σχετικό παράδειγμα αφορά στην τόσο γνωστή μας μονοδιάστατη εξίσωση της διαχύσεως. Αυτή είναι η εξίσωση:

```
In[7]:= pde = D[u[t, x], {x, 2}] == (1 / c^2) D[u[t, x], {t, 2}]
```

$$\text{Out[7]=}\ u^{(0,2)}[t, x] == \frac{u^{(2,0)}[t, x]}{c^2}$$

Γενικά μιλώντας, ο μετασχηματισμός Laplace μπορεί να γίνει εδώ είτε ως προς το χρόνο t είτε ως προς τη θέση x. Φυσικά θα γίνει τελικά ως προς τη μεταβλητή εκείνη που μεταβάλλεται στο ημιάπειρο διάστημα $[0, \infty)$ του μετασχηματισμού Laplace:

```
In[8]:= LT1 = LaplaceTransform[pde, t, s]
```

$$\text{Out[8]=}\ \text{LaplaceTransform}[u^{(0,2)}[t, x], t, s] == \frac{s^2\,\text{LaplaceTransform}[u[t, x], t, s] - s\,u[0, x] - u^{(1,0)}[0, x]}{c^2}$$

```
In[9]:= LT2 = LaplaceTransform[pde, x, σ]
```

$$\text{Out[9]=}\ \sigma^2\,\text{LaplaceTransform}[u[t, x], x, \sigma] - \sigma\,u[t, 0] - u^{(0,1)}[t, 0] == \frac{\text{LaplaceTransform}[u^{(2,0)}[t, x], x, \sigma]}{c^2}$$

Ο μετασχηματισμός Laplace είναι εφαρμόσιμος και σε γραμμικές ολοκληρωτικές εξισώσεις (linear integral equations) Volterra με συνελικτικό πυρήνα μετατρέποντας μια τέτοια ολοκληρωτική εξίσωση σε μια γραμμική αλγεβρική εξίσωση. Ένα σχετικό παράδειγμα αποτελεί η ακόλουθη ολοκληρωτική εξίσωση του μηχανικού συστήματος μάζας–ελατηρίου (χωρίς απόσβεση) υπό συγκεκριμένη αρμονική φόρτιση:

```
In[10]:= ie = u[t] + ω0^2 Integrate[(t - τ) u[τ], {τ, 0, t}] == u0 + (p0 / ω0^2) (1 - Cos[ω0 t])
```

$$\text{Out[10]=}\ \left(\int_0^t (t-\tau)\,u[\tau]\,d\tau\right)\omega_0^2 + u[t] == u0 + \frac{(1 - \text{Cos}[t\,\omega_0])\,p_0}{\omega_0^2}$$

Με την εφαρμογή του μετασχηματισμού Laplace αυτή παίρνει την εξής μορφή:

```
In[11]:= LTie = LaplaceTransform[ie, t, s] // Simplify
```

Out[11]= $\frac{\text{LaplaceTransform}[u[t], t, s]\,(s^2+\omega_0^2)}{s^2} == \frac{s^2\,u0 + p_0 + u0\,\omega_0^2}{s^3 + s\,\omega_0^2}$

με λύση της ως προς το μετασχηματισμό Laplace της άγνωστης συναρτήσεως την ακόλουθη:

```
In[12]:= solLTie = Solve[LTie, LaplaceTransform[u[t], t, s]] // Simplify
```

Out[12]= $\left\{\left\{\text{LaplaceTransform}[u[t], t, s] \to \frac{s\,(p_0 + u0\,(s^2+\omega_0^2))}{(s^2+\omega_0^2)^2}\right\}\right\}$

■ ΕΝΤΟΛΗ Ο2: ΑΝΤΙΣΤΡΟΦΟΣ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ LAPLACE

InverseLaplaceTransform[*Συνάρτηση ή ΛίσταΣυναρτήσεων ή Εξίσωση ή ΛίσταΕξισώσεων, ΜεταβλητήΜετασχηματισμούLaplace, ΑρχικήΜεταβλητή*]

Υπολογίζει τον αντίστροφο μετασχηματισμό Laplace μιας συναρτήσεως ή λίστας συναρτήσεων ή εξισώσεως ή λίστας εξισώσεων. (Δηλαδή είναι η αντίστροφη εντολή της εντολής **LaplaceTransform**). Συχνά η μεταβλητή του μετασχηματισμού Laplace είναι το *s* και η αρχική μεταβλητή (όπου επιστρέφουμε) είναι το *t* (συνήθως για το χρόνο) ή το *x* (συνήθως για τη θέση). Πρώτα ένα απλό παράδειγμα:

```
In[13]:= InverseLaplaceTransform[1/(s^2 + a^2)^5, s, t]
```

Out[13]= $\frac{5\,a\,t\,(-21 + 2\,a^2\,t^2)\,\text{Cos}[a\,t] + (105 - 45\,a^2\,t^2 + a^4\,t^4)\,\text{Sin}[a\,t]}{384\,a^9}$

και έπειτα ένα δυσκολότερο παράδειγμα που σχετίζεται με δοκό επί ελαστικής βάσεως (π.χ. πεδιλοδοκό):

```
In[14]:= InverseLaplaceTransform[1/(s^4 + 4 a^4), s, t] // ExpToTrig // Simplify
```

Out[14]= $\frac{\text{Cosh}[a\,t]\,\text{Sin}[a\,t] - \text{Cos}[a\,t]\,\text{Sinh}[a\,t]}{4\,a^3}$

Φυσικά και η εντολή αυτή είναι εφαρμόσιμη και σε ολόκληρη λίστα συναρτήσεων καθώς και σε εξισώσεις:

```
In[15]:= InverseLaplaceTransform[
           {1/(s - a), 1/(s^2 + a^2), s/(s^2 + a^2), 1/(s^2 - a^2), s/(s^2 - a^2)}, s, t] // FullSimplify
```

Out[15]= $\left\{e^{a\,t}, \frac{\text{Sin}[a\,t]}{a}, \text{Cos}[a\,t], \frac{\text{Sinh}[a\,t]}{a}, \text{Cosh}[a\,t]\right\}$

```
In[16]:= InverseLaplaceTransform[U[s] == 1/(s^2 + a^2), s, t] /.
          InverseLaplaceTransform[U[s], s, t] → u[t]
```

Out[16]= $u[t] == \frac{\text{Sin}[a\,t]}{a}$

Ασφαλώς οι εντολές **LaplaceTransform** και **InverseLaplaceTransform** είναι αντίστροφες μεταξύ τους

```
In[17]:= InverseLaplaceTransform[LaplaceTransform[u[x], x, s], s, x]

Out[17]= u[x]

In[18]:= LaplaceTransform[InverseLaplaceTransform[U[s], s, x], x, s]

Out[18]= U[s]
```

■ ΕΝΤΟΛΗ O3: ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ FOURIER

FourierTransform[*Συνάρτηση ή ΛίσταΣυναρτήσεων ή Εξίσωση ή ΛίσταΕξισώσεων,*

***ΑρχικήΜεταβλητή, ΜεταβλητήΜετασχηματισμούFourier*]**

Εντελώς ανάλογα με το μετασχηματισμό Laplace η εντολή αυτή υπολογίζει το μετασχηματισμό Fourier μιας συναρτήσεως ή λίστας συναρτήσεων ή εξισώσεως ή λίστας εξισώσεων. Πολύ συχνά αρχική μεταβλητή είναι το t ή το x και η μεταβλητή του μετασχηματισμού Fourier το ω. Τονίζεται όμως πως η *Mathematica* δε χρησιμοποιεί τον ορισμό του μετασχηματισμού Fourier που χρησιμοποιεί ο Πολιτικός Μηχανικός. Γι' αυτό γίνεται αλλαγή των παραμέτρων της με την επιλογή **FourierParameters → {1, –1}**

```
In[19]:= SetOptions[{FourierTransform, InverseFourierTransform},
           FourierParameters → {1, -1}];
```

Έτσι γραμμένη η εντολή αυτή αφορά και στον ευθύ και στον αντίστροφο μετασχηματισμό Fourier. Τώρα

```
In[20]:= FourierTransform[{DiracDelta[t], DiracDelta[t - a]}, t, ω]
```

Out[20]= $\{1, e^{-i a \omega}\}$

```
In[21]:= FourierTransform[p0 (UnitStep[t] - UnitStep[t - a]), t, ω] // Simplify
```

Out[21]= $-\frac{i\, p_0}{\omega} + \frac{i\, e^{-i a \omega}\, p_0}{\omega}$

Ο μετασχηματισμός Fourier της δευτέρας παραγώγου μιας συναρτήσεως είναι γνωστός στη *Mathematica.* Το ίδιο και οι μετασχηματισμοί Fourier όλων των παραγώγων μιας παραγωγίσιμης συναρτήσεως:

```
In[22]:= {FourierTransform[u''[t], t, ω], FourierTransform[u''''[t], t, ω]}
```

Out[22]= $\{-\omega^2$ FourierTransform[u[t], t, ω], ω^4 FourierTransform[u[t], t, ω]$\}$

■ ΕΝΤΟΛΗ O4: ΑΝΤΙΣΤΡΟΦΟΣ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ FOURIER

InverseFourierTransform[*Συνάρτηση ή ΛίσταΣυναρτήσεων ή Εξίσωση ή ΛίσταΕξισώσεων,*

***ΜεταβλητήΜετασχηματισμούFourier, ΑρχικήΜεταβλητή*]**

Ανάλογα με το μετασχηματισμό Laplace υπολογίζει τον αντίστροφο μετασχηματισμό Fourier μιας συναρτήσεως ή λίστας συναρτήσεων ή εξισώσεως ή λίστας εξισώσεων. Συχνά η μεταβλητή του μετασχηματισμού Fourier είναι το ω και η αρχική μεταβλητή (όπου γυρίζουμε) είναι το t ή το x. Παράδειγμα:

```
In[23]:= InverseFourierTransform[1, ω, t]

Out[23]= DiracDelta[t]
```

Όπως στο μετασχηματισμό Laplace, έτσι και στο μετασχηματισμό Fourier οι εντολές **FourierTransform** και **InverseFourierTransform** είναι αντίστροφες μεταξύ τους. Επομένως η ακόλουθη σχέση είναι αληθής:

```
In[24]:= InverseFourierTransform[FourierTransform[u[t], t, ω], ω, t] == u[t]

Out[24]= True
```

■ *Notebook E18*
ΕΝΤΟΛΕΣ ΓΙΑ ΜΙΓΑΔΙΚΕΣ ΣΥΝΑΡΤΗΣΕΙΣ

10 ΕΝΤΟΛΕΣ: **C1. ComplexExpand, C2. Conjugate, C3. Re, C4. Im, C5. Arg, C6. RealValued, C7. Residue, C8. NResidue, C9. CartesianMap, C10. PolarMap**

■ ΕΝΤΟΛΗ C1: ΑΝΑΠΤΥΓΜΑ ΜΕ ΠΡΑΓΜΑΤΙΚΕΣ ΜΕΤΑΒΛΗΤΕΣ

ComplexExpand[*ΜιγαδικήΠαράσταση*]

Αναπτύσσει μια μιγαδική παράσταση υποθέτοντας ότι όλες οι μεταβλητές (όλα τα σύμβολα) που υπεισέρχονται στην παράσταση είναι πραγματικές. Πολύ συχνά η εντολή αυτή **ComplexExpand** μπαίνει μετά την παράσταση με το σύμβολο // μπροστά της. Ένα πρώτο παράδειγμα:

```
In[1]:= {Expand[(-1)^(1/5)], e1 = ComplexExpand[(-1)^(1/5)], e2 = (-1)^(1/5) // ComplexExpand, e1 == e2}
```

Out[1]= $\left\{(-1)^{1/5},\ \frac{1}{4}+\frac{\sqrt{5}}{4}+\frac{1}{2}\,i\sqrt{\frac{1}{2}\left(5-\sqrt{5}\right)},\ \frac{1}{4}+\frac{\sqrt{5}}{4}+\frac{1}{2}\,i\sqrt{\frac{1}{2}\left(5-\sqrt{5}\right)},\ \text{True}\right\}$

Το ανάπτυγμα δύο τριγωνομετρικών συναρτήσεων με την υπόθεση πραγματικών μεταβλητών x και y:

```
In[2]:= {Sin[x + i y] // Expand, Sin[x + i y] // ComplexExpand, Tanh[x + i y] // ComplexExpand}
```

Out[2]= $\left\{\text{Sin}[x+i\,y],\ \text{Cosh}[y]\,\text{Sin}[x]+i\,\text{Cos}[x]\,\text{Sinh}[y],\ \frac{i\,\text{Sin}[2\,y]}{\text{Cos}[2\,y]+\text{Cosh}[2\,x]}+\frac{\text{Sinh}[2\,x]}{\text{Cos}[2\,y]+\text{Cosh}[2\,x]}\right\}$

Αρκετές εφαρμογές της εντολής **ComplexExpand** (βέβαια με την υπόθεση πραγματικών μεταβλητών) ακολουθούν στις πέντε επόμενες εντολές για μιγαδικές συναρτήσεις.

■ ΕΝΤΟΛΗ C2: ΣΥΖΥΓΗΣ ΜΙΓΑΔΙΚΗ ΠΑΡΑΣΤΑΣΗ

Conjugate[*ΜιγαδικήΠαράσταση*]

Υπολογίζει τη συζυγή μιγαδική παράσταση μιας μιγαδικής παραστάσεως, ειδικότερα ενός μιγαδικού αριθμού. Αρκετές φορές η παρούσα εντολή **Conjugate** καθώς και οι εντολές **Re, Im, Abs** και **Arg** (επίσης για μιγαδικούς αριθμούς και μιγαδικές παραστάσεις) δίνουν πολύ καλύτερα αποτελέσματα μετά την κλήση (το φόρτωμα) του πακέτου **Algebra'ReIm'**. Επιπλέον η εντολή **RealValued** πιο κάτω (για τη δήλωση πραγματικών συναρτήσεων) είναι αποκλειστικά εντολή του ίδιου πακέτου. Αυτή η κλήση γίνεται με έναν από τους δύο πιο κάτω ισοδύναμους τρόπους:

```
In[3]:= Needs["Algebra`ReIm`"]
```

```
In[4]:= << Algebra`ReIm`
```

Υπενθυμίζεται ότι το **Im** στο σύμβολο **ReIm** αποτελεί σύντμηση της Αγγλικής λέξεως imaginary (φανταστικός). Δεν πρέπει επίσης να λησμονείται ότι η προηγούμενη εντολή **ComplexExpand** επιτρέπει την υπόθεση πραγματικών μεταβλητών και πρέπει να χρησιμοποιείται σ' αυτήν την περίπτωση, που δεν είναι δα και τόσο σπάνια περίπτωση. Παραδείγματα:

```
In[5]:= {Conjugate[1 + ⅈ], Conjugate[2^(3-4 ⅈ)],
         Conjugate[a + ⅈ b], Conjugate[a + ⅈ b] // ComplexExpand}

Out[5]= {1 - ⅈ, 2^(3+4 ⅈ), Conjugate[a] - ⅈ Conjugate[b], a - ⅈ b}

In[6]:= {Conjugate[Coth[x + ⅈ y]], Conjugate[Coth[x + ⅈ y]] // ComplexExpand}

Out[6]= {Coth[Conjugate[x] - ⅈ Conjugate[y]], -(ⅈ Sin[2 y])/(Cos[2 y] - Cosh[2 x]) - Sinh[2 x]/(Cos[2 y] - Cosh[2 x])}
```

Και τώρα μια πολύ γνωστή ταυτότητα για μιγαδικούς αριθμούς, που απαιτεί όμως για την απόδειξή της πλήρη απλοποίηση με την εντολή **FullSimplify**:

```
In[7]:= {ταυτότητα = z Conjugate[z] == Abs[z]^2, ταυτότητα // Simplify, ταυτότητα // FullSimplify}

Out[7]= {z Conjugate[z] == Abs[z]^2, z Conjugate[z] == Abs[z]^2, True}
```

■ ΕΝΤΟΛΗ C3: ΠΡΑΓΜΑΤΙΚΟ ΜΕΡΟΣ

Re[*ΜιγαδικήΠαράσταση*]

Υπολογίζει το πραγματικό μέρος μιας μιγαδικής παραστάσεως, ειδικότερα ενός μιγαδικού αριθμού. Δεν πρέπει να λησμονείται η χρήση και της εντολής **ComplexExpand** για την υπόθεση πραγματικών μεταβλητών. Παραδείγματα:

```
In[8]:= {Re[2 + 3 ⅈ], Re[ⅈ (4 + 5 ⅈ)], Re[ⅈ^2], Re[ⅈ^3], Re[Sqrt[ⅈ]], r = Re[Exp[3 + 2 ⅈ]], N[r, 50]}

Out[8]= {2, -5, -1, 0, 1/√2, ⅇ^3 Cos[2], -8.3585326509353715808873680784697285721955110088860}

In[9]:= {Re[Cosh[α + ⅈ β]], Re[Cosh[α + ⅈ β]] // ComplexExpand}

Out[9]= {Cos[Im[α] + Re[β]] Cosh[Im[β] - Re[α]], Cos[β] Cosh[α]}
```

■ ΕΝΤΟΛΗ C4: ΦΑΝΤΑΣΤΙΚΟ ΜΕΡΟΣ

Im[*ΜιγαδικήΠαράσταση*]

Υπολογίζει το φανταστικό μέρος μιας μιγαδικής παραστάσεως, ειδικότερα ενός μιγαδικού αριθμού. Παραδείγματα:

```
In[10]:= {Im[2 + 3 ⅈ], Im[ⅈ (4 + 5 ⅈ)], Im[ⅈ^2], Im[ⅈ^3], Im[Sqrt[ⅈ]], s = Im[Exp[3 + 2 ⅈ]], N[s, 50]}

Out[10]= {3, 4, 0, -1, 1/√2, ⅇ^3 Sin[2], 18.263727040666766171446496807388694680491480807685}

In[11]:= {Im[Cosh[α + ⅈ β]], Im[Cosh[α + ⅈ β]] // ComplexExpand}

Out[11]= {-Sin[Im[α] + Re[β]] Sinh[Im[β] - Re[α]], Sin[β] Sinh[α]}

In[12]:= {ex = Cosh[α + ⅈ β], re = Re[ex], im = Im[ex], ex == re + ⅈ im} // ComplexExpand

Out[12]= {Cos[β] Cosh[α] + ⅈ Sin[β] Sinh[α], Cos[β] Cosh[α], Sin[β] Sinh[α], True}
```

■ ΕΝΤΟΛΗ C5: ΟΡΙΣΜΑ

Arg[*ΜιγαδικήΠαράσταση*]

Υπολογίζει το πρωτεύον όρισμα μιας μιγαδικής παραστάσεως. Για το μέτρο ή απόλυτο τιμή συνεχίζει να ισχύει η εντολή **Abs**, ακριβώς όπως συμβαίνει και στους πραγματικούς αριθμούς. Παραδείγματα:

```
In[13]:= {Arg[1 + ⅈ], Arg[1 + 2 ⅈ], N[Arg[1 + 2 ⅈ], 40], Arg[ⅇ^(1+2 ⅈ)], Arg[ⅈ], Arg[-ⅈ], Arg[Sin[ⅈ]]}
```

$$\text{Out[13]= } \left\{\frac{\pi}{4},\ \text{ArcTan[2]},\ 1.107148717794090503017065460178537040070,\ 2,\ \frac{\pi}{2},\ -\frac{\pi}{2},\ \frac{\pi}{2}\right\}$$

```
In[14]:= {{Abs[ⅇ^(x+ⅈ y)], Arg[ⅇ^(x+ⅈ y)]}, {Abs[ⅇ^(x+ⅈ y)], Arg[ⅇ^(x+ⅈ y)]} // ComplexExpand}

Out[14]= {{ⅇ^(-Im[y]+Re[x]), Arg[ⅇ^(x+ⅈ y)]}, {ⅇ^x, ArcTan[Cos[y], Sin[y]]}}

In[15]:= {p = ⅇ^(x+ⅈ y) == Abs[ⅇ^(x+ⅈ y)] ⅇ^(ⅈ Arg[ⅇ^(x+ⅈ y)]) // ComplexExpand // Simplify, p // FullSimplify}

Out[15]= {ⅇ^x (Cos[y] - Cos[ArcTan[Cos[y], Sin[y]]] + ⅈ (Sin[y] - Sin[ArcTan[Cos[y], Sin[y]]])) ==
           0, True}
```

■ ΕΝΤΟΛΗ C6: ΔΗΛΩΣΗ ΠΡΑΓΜΑΤΙΚΩΝ ΣΥΝΑΡΤΗΣΕΩΝ

RealValued[*ΠραγματικήΣυνάρτηση-1, ΠραγματικήΣυνάρτηση-2, . . .*]

Με την εντολή αυτή, η οποία είναι μέρος του πακέτου **ReIm**, γίνεται η υπόθεση ότι η συνάρτηση στο όρισμά της/οι συναρτήσεις στα ορίσματά της είναι πραγματική/πραγματικές. Δηλαδή δεν περιλαμβάνει/περιλαμβάνουν τη φανταστική μονάδα *i*. Με άλλα λόγια, εφόσον η μεταβλητή στο όρισμα της συναρτήσεως είναι πραγματική, το αποτέλεσμα θα είναι επίσης πραγματικό. Ανάλογα ισχύουν και για όλες τις συναρτήσεις. Παράδειγμα δηλώσεως τριών συναρτήσεων σαν πραγματικών συναρτήσεων:

```
In[16]:= RealValued[f, g, h]

Out[16]= {f, g, h}
```

Τώρα η *Mathematica* γνωρίζει ότι η συνάρτηση *f* είναι πραγματική (όπως και οι συναρτήσεις *g* και *h*). Εντούτοις δίνει

```
In[17]:= {Re[f[x]], Im[f[x]]}

Out[17]= {Re[f[x]], Im[f[x]]}
```

γιατί δε γνωρίζει τί συμβαίνει με τη μεταβλητή *x*: είναι πραγματική ή μιγαδική; Αυτή η μεταβλητή μπορεί επίσης να υποτεθεί πραγματική με τη χρήση της εντολής **ComplexExpand**, η οποία έχει ήδη αναφερθεί στην αρχή αυτού του notebook για τις μιγαδικές συναρτήσεις. Συγκεκριμένα τώρα παίρνουμε

```
In[18]:= {Re[f[x]], Im[f[x]]} // ComplexExpand

Out[18]= {f[x], 0}

In[19]:= {Re[f[x + y] + g[y + z] + h[z + x]], Im[f[x + y] + g[y + z] + h[z + x]]} // ComplexExpand

Out[19]= {f[x + y] + g[y + z] + h[x + z], 0}
```

■ ΕΝΤΟΛΗ C7: ΟΛΟΚΛΗΡΩΤΙΚΟ ΥΠΟΛΟΙΠΟ

Residue[*ΜιγαδικήΣυνάρτηση, {Μεταβλητή, Σημείο*}]

Υπολογίζει το ολοκληρωτικό υπόλοιπο μιας μιγαδικής συναρτήσεως (για την ακρίβεια συνήθως μιγαδικής), που καθορίζεται στο πρώτο όρισμα ως προς τη μεταβλητή και στο σημείο που καθορίζονται στη λίστα του δεύτερου ορίσματος, που έχει έτσι δύο στοιχεία. Υπενθυμίζεται πως το ολοκληρωτικό υπόλοιπο μιας συναρτήσεως (συνήθως μιγαδικής) σε ένα σημείο z είναι απλά ο συντελεστής της δυνάμεως με εκθέτη -1 της σειράς Laurent της συναρτήσεως στο σημείο αυτό. Σημειώνεται επίσης ότι η εντολή αυτή **Residue** ανήκει στον πυρήνα (kernel) της *Mathematica*. Επομένως δεν απαιτεί την κλήση (το φόρτωμα) του πακέτου **ReIm**. Παραδείγματα:

```
In[20]:= {Residue[1 / z, {z, 0}], Residue[A / (z - z0), {z, z0}], Residue[B / (w - w0)^2, {w, w0}]}

Out[20]= {1, A, 0}

In[21]:= {Residue[f[z] / (z - z0)^30, {z, z0}], Residue[f[ξ] / ((ξ - ξ0)^2 g[ξ]), {ξ, ξ0}],
          Residue[ Exp[1 / z], {z, 0}], Residue[ Sin[1 / z], {z, 0}], Residue[Cot[z], {z, 0}]}
```

Out[21]= $\left\{\dfrac{f^{(29)}[z0]}{8841761993739701954543616000000}, \dfrac{g[\xi0]\, f'[\xi0] - f[\xi0]\, g'[\xi0]}{g[\xi0]^2}, 1, 1, 1\right\}$

■ ΕΝΤΟΛΗ C8: ΑΡΙΘΜΗΤΙΚΟΣ ΥΠΟΛΟΓΙΣΜΟΣ ΟΛΟΚΛΗΡΩΤΙΚΟΥ ΥΠΟΛΟΙΠΟΥ

NResidue[*ΜιγαδικήΣυνάρτηση, {Μεταβλητή, Σημείο*}]

Υπολογίζει αριθμητικά το ολοκληρωτικό υπόλοιπο μιας μιγαδικής συναρτήσεως (για την ακρίβεια συνήθως μιγαδικής), η οποία καθορίζεται στο πρώτο όρισμα, ως προς τη μεταβλητή και στο σημείο που καθορίζονται στη λίστα του δεύτερου ορίσματος, το οποίο έχει έτσι δύο στοιχεία. Σημειώνεται ότι η εντολή αυτή **NResidue** απαιτεί για τη χρήση της την κλήση (το φόρτωμα) ενός ειδικού πακέτου: του πακέτου **NumericalMath‘NResidue‘** με μοναδική εντολή του αυτήν εδώ την εντολή: **NResidue**. Κλήση (φόρτωμα) του πακέτου με έναν από τους δύο γνωστούς μας ισοδύναμους τρόπους και παραδείγματα:

```
In[22]:= Needs["NumericalMath`NResidue`"]

In[23]:= << NumericalMath`NResidue`

In[24]:= {NResidue[1 / z, {z, 0}], NResidue[5 / (z - 3), {z, 3}], NResidue[10 / (ξ + 8), {ξ, -8}]}

Out[24]= {1., 5., 10.}

In[25]:= {NResidue[1 / z^2, {z, 0}], NResidue[Exp[z] / (z - 5)^4, {z, 5}], NResidue[Cot[z], {z, 0}]}
```

Out[25]= $\{-3.55271\times10^{-15} + 0.\,i,\ 24.7355 + 0.\,i,\ 1. + 1.61614\times10^{-17}\,i\}$

Η παρούσα αριθμητική εντολή **NResidue** κάνει τους υπολογισμούς της με βάση αριθμητική ολοκλήρωση στην περιφέρεια ενός μικρού κύκλου που περιβάλλει το υπόψη σημείο: εργάζεται εντελώς διαφορετικά από την αντίστοιχη αναλυτική πιο πάνω εντολή **Residue**. Πολύ συχνά είναι σκόπιμο να χρησιμοποιείται και η εντολή **Chop**, ώστε πολύ μικροί αριθμοί (αποτελέσματα των σφαλμάτων των αριθμητικών υπολογισμών και εδώ ειδικότερα της αριθμητικής ολοκληρώσεως) να εξαλείφονται στο τέλος. Με αυτόν τον τρόπο στο τελευταίο πιο πάνω παράδειγμα παίρνουμε τελικά

```
In[26]:= {NResidue[1 / z^2, {z, 0}],
          NResidue[Exp[z] / (z - 5)^4, {z, 5}], NResidue[Cot[z], {z, 0}]} // Chop

Out[26]= {0, 24.7355, 1.}
```

Σημειώνεται τέλος ότι με τη μέθοδο της αριθμητικής ολοκληρώσεως, με την οποία κάνει τους υπολογισμούς της η παρούσα αριθμητική εντολή **NResidue**, σε πολύ σπάνιες περιπτώσεις, συγκεκριμένα όταν μέσα στον κύκλο ο οποίος έχει επιλεγεί η μιγαδική συνάρτηση έχει πάνω από έναν πόλους με ολοκληρωτικά υπόλοιπα, η εντολή αυτή **NResidue** θα βγάλει λάθος αποτέλεσμα εξαιτίας των άνω του ενός πόλων. Στην περίπτωση αυτή πρέπει να μειωθεί κατάλληλα η ακτίνα της περιφέρειας του κύκλου όπου γίνεται η αριθμητική ολοκλήρωση. Για το σκοπό αυτό υπάρχει διαθέσιμη σχετική επιλογή **Radius**. Τέλος σε δύσκολες περιπτώσεις, όχι όμως και τόσο σπάνια, η αριθμητική ολοκλήρωση στην περιφέρεια του κύκλου μπορεί να συγκλίνει πάρα πολύ αργά. (Αυτά συμβαίνουν μερικές φορές στους αριθμητικούς υπολογισμούς ...) Επομένως πρέπει να δίνεται προτίμηση στους αναλυτικούς υπολογισμούς, όπως εδώ με τη συνήθη εντολή **Residue**, όσες φορές βέβαια αυτό είναι δυνατόν.

■ ΕΝΤΟΛΗ C9: ΚΑΡΤΕΣΙΑΝΗ ΑΠΕΙΚΟΝΙΣΗ ΜΙΓΑΔΙΚΗΣ ΣΥΝΑΡΤΗΣΕΩΣ

CartesianMap[*ΟνομασίαΜιγαδικήςΣυναρτήσεως*, {*ΑρχικήΤιμή-x*, *ΤελικήΤιμή-x*},
{*ΑρχικήΤιμή-y*, *ΤελικήΤιμή-y*}]

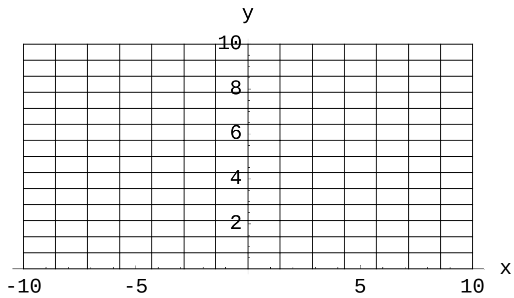
Κάνει τη γραφική παράσταση, καλύτερα την απεικόνιση μιας μιγαδικής συναρτήσεως, σε Καρτεσιανές συντεταγμένες στο διδιάστατο χώρο. Συγκεκριμένα για τη μεταβλητή x κατά τον άξονα Ox χωρίζει το σχετικό διάστημα $[x_1, x_2]$ σε ίσα τμήματα και απεικονίζει τις σχετικές κατακόρυφες ευθείες στο νέο επίπεδο $[u, v]$ με βάση τη μιγαδική συνάρτηση που δίνεται: $w = f(z)$ με $z = x + iy$ και $w = u + iv$. Ανάλογα συμβαίνουν και για τη μεταβλητή y. Για την εκτέλεση της εντολής αυτής απαιτείται προηγουμένως η κλήση (το φόρτωμα) του σχετικού ειδικού πακέτου Graphics'ComplexMap'. Αυτή η κλήση μπορεί να γίνει με την πρώτη ή τη δεύτερη από τις παρακάτω δύο ισοδύναμες εντολές:

```
In[27]:= Needs["Graphics`ComplexMap`"]

In[28]:= << Graphics`ComplexMap`
```

Αρχικά σημειώνουμε πως στην παρούσα εντολή **CartesianMap** (εδώ για Καρτεσιανές συντεταγμένες), ακριβώς όπως και στην επόμενη εντολή **PolarMap** (για πολικές συντεταγμένες) η συνάρτηση που απεικονίζεται αναγράφεται στο πρώτο όρισμα της εντολής μόνο με την ονομασία της: χωρίς τη μιγαδική μεταβλητή της. Εναλλακτικά μπορεί να χρησιμοποιηθεί καθαρή συνάρτηση, όπως θα εξηγηθεί παρακάτω.

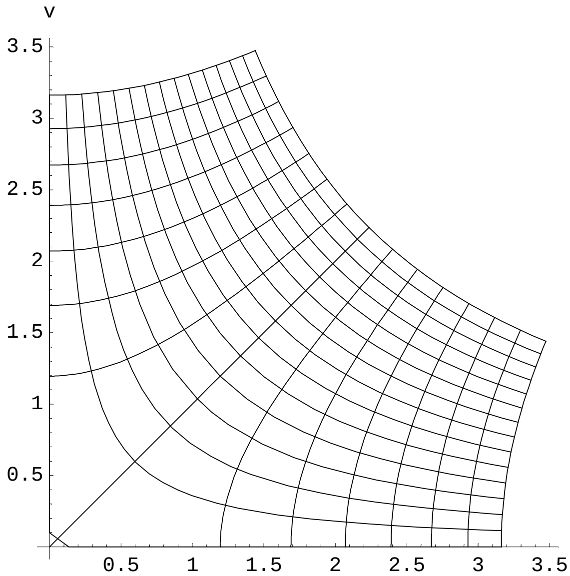
Τώρα σαν πρώτο, εισαγωγικό παράδειγμα χρησιμοποιούμε την ταυτοτική συνάρτηση (που δηλώνεται στη *Mathematica* με το σύμβολο **Identity**). Η σχετική γραφική παράσταση μας δείχνει ουσιαστικά ποιες ευθείες (κατακόρυφες: κάθετες στον άξονα x και οριζόντιες: κάθετες στον άξονα y) θα απεικονισθούν στο νέο επίπεδο Ouv με την παραπέρα χρήση της παρούσας εντολής **CartesianMap**. Εδώ που χρησιμοποιούμε την ταυτοτική συνάρτηση **Identity** γίνεται μόνο η ταυτοτική απεικόνιση από το επίπεδο Oxy στο επίπεδο Ouv, δηλαδή ουσιαστικά δε γίνεται καμία απεικόνιση. Ας το δούμε αυτό χρησιμοποιώντας διαστήματα $[-10, 10]$ στον οριζόντιο άξονα x και $[0, 10]$ στον κατακόρυφο άξονα y:

In[29]:= **CartesianMap[Identity, {-10, 10}, {0, 10}, AxesLabel → {x, y}];**



Σαν δεύτερο παράδειγμα δείχνουμε την απεικόνιση της συναρτήσεως τετραγωνική ρίζα (**Sqrt**) για τα ίδια διαστήματα $[-10, 10]$ στον άξονα *x* και $[0, 10]$ στον άξονα *y*. Παρατηρούμε ότι τώρα ο θετικός ημιάξονας *x* απεικονίζεται στο θετικό ημιάξονα *u*, ενώ ο αρνητικός ημιάξονας *x* απεικονίζεται στο θετικό ημιάξονα *v*. Μπορεί επίσης να αναφερθεί ότι εύλογα ο θετικός ημιάξονας *y* απεικονίζεται στη διχοτόμο της ορθής γωνίας των δύο θετικών ημιαξόνων *u* και *v*, οι οποίοι σχηματίζουν το πιο κάτω τεταρτημόριο. Να τη λοιπόν ολόκληρη η απεικόνιση:

In[30]:= **CartesianMap[Sqrt, {-10, 10}, {0, 10}, AxesLabel → {u, v}];**

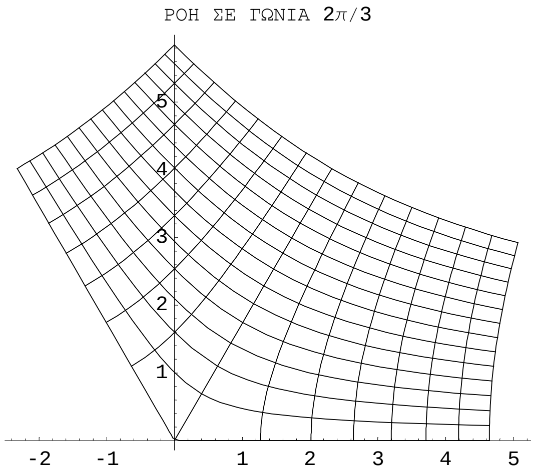


Στο σημείο αυτό ο παρατηρητικός φοιτητής όπως και η παρατηρητική φοιτήτρια Πολιτικός Μηχανικός διαπιστώνει ότι το προπροηγούμενο σχήμα (το σχήμα με τις οριζόντιες και τις κατακόρυφες ευθείες γραμμές) σχετίζεται με τη διδιάστατη μόνιμη ροή ιδεατού ρευστού στη Ρευστομηχανική. Συγκεκριμένα αφορά σε ομοιόμορφη ροή στο πάνω ημιεπίπεδο $y \geq 0$. Οι γραμμές ροής είναι οι οριζόντιες ευθείες, ενώ οι ισοδυναμικές γραμμές είναι οι κατακόρυφες ευθείες (ή πολύ πιο σωστά ημιευθείες). Το ρευστό ρέει οριζόντια. Ο άξονας *x* είναι το σύνορο της ροής. Ωραία, πολύ ωραία ως εδώ!

Ερχόμαστε τώρα στο κάτω σχήμα, στο προηγούμενο σχήμα διατηρώντας όμως στη σκέψη μας τη διδιάστατη μόνιμη ροή ιδεατού ρευστού στη Ρευστομηχανική. Έχοντας υπόψη μας την εφαρμογή στην Παράγραφο Δ3.6.1 του Μέρους Δ των διδακτικών βιβλίων για τις Μιγαδικές Συναρτήσεις για τη ροή ιδεατού ρευστού σε γωνία, παρατηρούμε ότι το κάτω σχήμα, το προηγούμενο σχήμα μας δείχνει απλά τη ροή ρευστού σε ορθή γωνία. Οι γραμμές ροής είναι οι υπερβολές που δεν τέμνουν τους δύο ημιάξονες *u* και *v* (συν τους ίδιους τους δύο ημιάξονες σαν μία επιπλέον γραμμή ροής). Επίσης οι ισοδυναμικές γραμμές είναι οι άλλες υπερβολές, αυτές που τέμνουν κάθετα τους δύο ημιάξονες *u* και *v*. Είναι λοιπόν χρήσιμη η εντολή **CartesianMap** όχι μόνο θεωρητικά αλλά και πρακτικά στον/στην Πολιτικό Μηχανικό.

Στο σημείο αυτό πρέπει να σημειωθεί ότι όταν η μιγαδική συνάρτηση που θέλουμε να απεικονίσουμε με την εντολή **CartesianMap** δεν έχει ειδική ονομασία στη *Mathematica*, τότε μπορούμε να χρησιμοποιήσουμε την ειδική εντολή **Function** της *Mathematica* για "καθαρές" συναρτήσεις. Άρα την ίδια ακριβώς απεικόνιση μπορούμε να την πάρουμε και με την πιο πάνω εντολή θέτοντας απλά **Function[*z*, *z*$^{1/2}$]** ή **Function[*w*, *w*$^{1/2}$]** σαν πρώτο όρισμα στη θέση του **Sqrt**. Και τώρα ένα παράδειγμα με τη χρήση της εντολής **Function** για τη συνάρτηση $w^{2/3}$. Αυτή αφορά στη ροή σε γωνία 2π/3 rad, ακτίνια (120 μοίρες):

In[31]:= **CartesianMap[Function[w, w$^{2/3}$], {-10, 10}, {0, 10}, PlotLabel → "ΡΟΗ ΣΕ ΓΩΝΙΑ 2π/3"];**

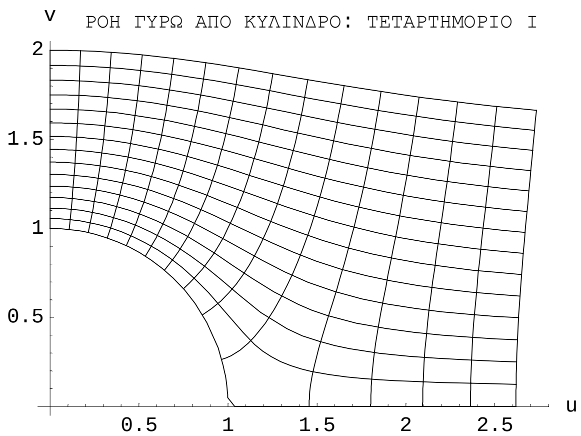


Κι ένα ακόμη παράδειγμα χρήσεως της εντολής **CartesianMap** για τη συνάρτηση που αφορά στη ροή γύρω από κυκλικό κύλινδρο, αν και εδώ θα γίνει η παράσταση της ροής μόνο στο πρώτο τεταρτημόριο:

In[32]:= **{equation = w == V$_0$ (z + a^2 / z) /. {V$_0$ → 1, a → 1}, solution = Solve[equation, z]}**

Out[32]= $\left\{w == \frac{1}{z} + z, \left\{\left\{z \to \frac{1}{2}\left(w - \sqrt{-4 + w^2}\right)\right\}, \left\{z \to \frac{1}{2}\left(w + \sqrt{-4 + w^2}\right)\right\}\right\}\right\}$

In[33]:= **CartesianMap[Function[w, (1 / 2) (w + Sqrt[-4 + w^2])], {0, 3}, {0, 1.5},
AxesLabel → {u, v}, PlotLabel → "   ΡΟΗ ΓΥΡΩ ΑΠΟ ΚΥΛΙΝΔΡΟ: ΤΕΤΑΡΤΗΜΟΡΙΟ Ι"];**



Υπενθυμίζεται τέλος ότι με την Αγγλική άνω τελεία ; (το Ελληνικό ερωτηματικό ;) στο τέλος εντολών για γραφικές παραστάσεις, όπως είναι η εντολή **Plot** και η παρούσα εντολή **CartesianMap**, αποφεύγεται η εμφάνιση της λέξεως **- Graphics -** μετά από ένα σχήμα. Έτσι επιτυγχάνεται λίγη οικονομία χώρου.

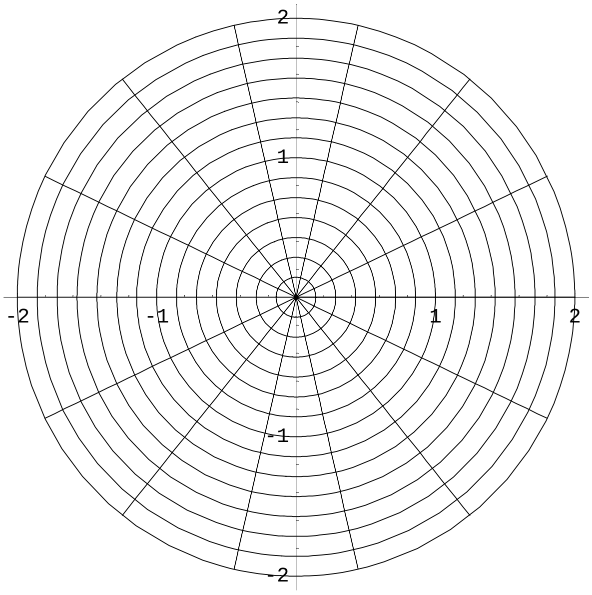
■ ΕΝΤΟΛΗ C10: ΠΟΛΙΚΗ ΑΠΕΙΚΟΝΙΣΗ ΜΙΓΑΔΙΚΗΣ ΣΥΝΑΡΤΗΣΕΩΣ

PolarMap[*ΟνομασίαΜιγαδικήςΣυναρτήσεως*, {*ΑρχικήΤιμή-r*, *ΤελικήΤιμή-r*},

{*ΑρχικήΤιμή-θ*, *ΤελικήΤιμή-θ*}]

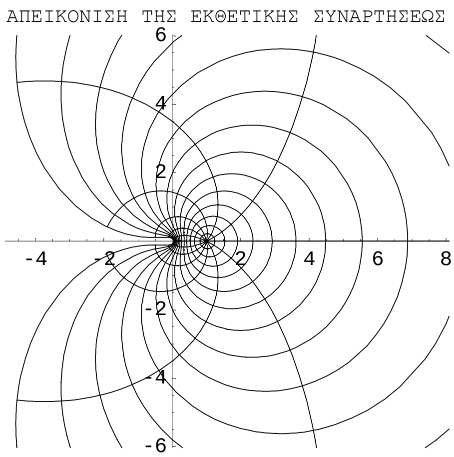
Πρόκειται για την απόλυτα ανάλογη εντολή της αμέσως προηγούμενης εντολής **CartesianMap**, αλλά τώρα σε πολικές συντεταγμένες (r, θ) αντί σε Καρτεσιανές συντεταγμένες (x, y). Φυσικά και η παρούσα εντολή **PolarMap** απαιτεί κι αυτή την κλήση (το φόρτωμα) του ίδιου ακριβώς ειδικού πακέτου **Graphics'ComplexMap'**, του οποίου αποτελεί μέρος. Τώρα η ταυτοτική συνάρτηση **Identity** μας δίνει περιφέρειες που έχουν σταθερή πολική ακτίνα r, και ημιευθείες που έχουν σταθερή πολική γωνία θ:

```
In[34]:= PolarMap[Identity, {0, 2}, {0, 2 π}];
```



Ας σημειωθεί εδώ ότι το πιο πάνω σχήμα αντιστοιχεί στο πεδίο ροής σημειακής πηγής με γραμμές ροής τις ημιευθείες και ισοδυναμικές γραμμές τις περιφέρειες. Ανάλογα ισχύουν για το πεδίο ροής σημειακής δίνης. Άλλο παράδειγμα με την απεικόνιση σε πολικές συντεταγμένες της εκθετικής συναρτήσεως **Exp**:

```
In[35]:= PolarMap[Exp, {0, 3}, {0, 2 π}, PlotLabel -> "ΑΠΕΙΚΟΝΙΣΗ ΤΗΣ ΕΚΘΕΤΙΚΗΣ ΣΥΝΑΡΤΗΣΕΩΣ"];
```



■ *Notebook E19*

ΕΝΤΟΛΕΣ ΕΙΣΟΔΟΥ–ΕΞΟΔΟΥ

5 ΕΝΤΟΛΕΣ: **I1. Import, I2. Export, I3. FortranForm, I4. CForm, I5. TeXForm**

■ ΕΝΤΟΛΗ Ι1: ΕΙΣΑΓΩΓΗ ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΑΡΧΕΙΟ

Import[*"Αρχείο"*]

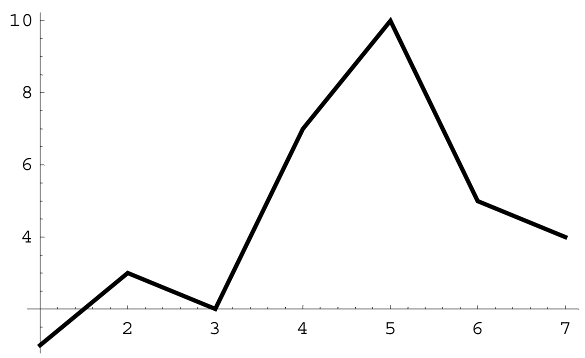
Import[*"Αρχείο"*, *"ΤύποςΔεδομένων"*]

Στην πρώτη μορφή της η εντολή αυτή εισάγει δεδομένα από το αρχείο που δίνεται στην πλήρη μορφή του στο όρισμά της και τα μετατρέπει σε αντίστοιχη παράσταση της *Mathematica*. Ο τύπος των δεδομένων καθορίζεται από την επέκταση του ονόματος του αρχείου (μετά την τελεία), π.χ. **dat** για την εισαγωγή των στοιχείων μητρώου. Στη δεύτερη μορφή της εντολής ο τύπος των δεδομένων καθορίζεται πιο ρητά από το δεύτερο όρισμά της, π.χ. "**Text**", "**List**", "**Table**", "**GIF**", "**EPS**", "**JPEG**", κλπ. Παράδειγμα εισαγωγής των στοιχείων λίστας από το αρχείο **file1.txt** στο φάκελο **math** του δίσκου **C**:

```
In[1]:= lst = Import["c:/math/file1.txt", "List"]
Out[1]= {1, 3, 2, 7, 10, 5, 4}
```

Και η γραφική παράσταση της λίστας αυτής που εισαγάγαμε:

```
In[2]:= pl = ListPlot[lst, PlotStyle → Thickness[0.008], PlotJoined → True, ImageSize → 280];
```



■ ΕΝΤΟΛΗ Ι2: ΕΞΑΓΩΓΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ ΣΕ ΑΡΧΕΙΟ

Export[*"Αρχείο", Αποτελέσματα*]

Εξάγει τα αποτελέσματα που δηλώνονται στο δεύτερο όρισμα στο αρχείο που δηλώνεται στο πρώτο όρισμα (με την πλήρη θέση του). Η μορφή των αποτελεσμάτων στο αρχείο αυτό δηλώνεται με την κατάλληλη επέκτασή του, π.χ. **txt, dat, eps, gif, jpg,** κλπ. Παράδειγμα για τη γραφική παράσταση **pl**:

```
In[3]:= Export["c:\\math\\Figure1.eps", pl]
Out[3]= c:\math\Figure1.eps
```

■ ΕΝΤΟΛΗ Ι3: ΕΜΦΑΝΙΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ ΣΕ ΜΟΡΦΗ FORTRAN

FortranForm[*Παράσταση*]

Εμφανίζει την παράσταση στο όρισμά της σε μορφή γλώσσας Fortran. Αυτή μπορεί έτσι να χρησιμοποιηθεί κατευθείαν στη γλώσσα Fortran έξω από τη *Mathematica*. Σαν παράδειγμα θεωρούμε τη γενική λύση της διαφορικής εξισώσεως των εξαναγκασμένων ταλαντώσεων στο κλασικό μηχανικό σύστημα μάζας–ελατηρίου με σταθερή θετική φόρτιση (εξωτερική δύναμη) $p(t) = a^2$. Αυτήν τη λύση τη μετατρέπουμε εύκολα σε γλώσσα Fortran, για να τη χρησιμοποιήσουμε στη γλώσσα Fortran:

In[4]:= **us[t_] = DSolve[m u''[t] + k u[t] == a^2, u[t], t][[1, 1, 2]]**

Out[4]= $\frac{a^2}{k} + C[1] \operatorname{Cos}\left[\frac{\sqrt{k}\, t}{\sqrt{m}}\right] + C[2] \operatorname{Sin}\left[\frac{\sqrt{k}\, t}{\sqrt{m}}\right]$

In[5]:= **FortranForm[us[t]]**

Out[5]//FortranForm=
```
          a**2/k + C(1)*
     -     Cos((Sqrt(k)*t)/Sqrt(m)) +
     -    C(2)*Sin((Sqrt(k)*t)/Sqrt(m))
```

■ ΕΝΤΟΛΗ Ι4: ΕΜΦΑΝΙΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ ΣΕ ΜΟΡΦΗ C

CForm[*Παράσταση*]

Εμφανίζει την παράσταση στο όρισμά της σε μορφή γλώσσας C. Αυτή μπορεί έτσι να χρησιμοποιηθεί κατευθείαν στη γλώσσα C έξω από τη *Mathematica*. Συνεχίζουμε το προηγούμενο παράδειγμα:

In[6]:= **CForm[us[t]]**

Out[6]//CForm=
```
Power(a,2)/k + C(1)*Cos((Sqrt(k)*t)/Sqrt(m)) +
   C(2)*Sin((Sqrt(k)*t)/Sqrt(m))
```

■ ΕΝΤΟΛΗ Ι5: ΕΜΦΑΝΙΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ ΣΕ ΜΟΡΦΗ TEX

TeXForm[*Παράσταση*]

Εμφανίζει την παράσταση στο όρισμά της σε μορφή γλώσσας TEX, που είναι επίσης κατάλληλη για το LATEX. Αυτή μπορεί έτσι να χρησιμοποιηθεί κατευθείαν στο πρόγραμμα στοιχειοθεσίας TEX (ή στο LATEX) έξω από τη *Mathematica* για τη στοιχειοθεσία καλαίσθητου μαθηματικού κειμένου. Συνεχίζουμε κι εδώ το προηγούμενο παράδειγμα για το μηχανικό σύστημα μάζας–ελατηρίου:

In[7]:= **TeXForm[us[t]]**

Out[7]//TeXForm=
```
\frac{a^2}{k} + \Mfunction{C}(1)\,
   \cos (\frac{{\sqrt{k}}\,t}{{\sqrt{m}}}) +
  \Mfunction{C}(2)\,
   \sin (\frac{{\sqrt{k}}\,t}{{\sqrt{m}}})
```

■ *Notebook E20*
ΕΝΤΟΛΕΣ ΔΙΑΔΙΚΑΣΤΙΚΟΥ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ
6 ΕΝΤΟΛΕΣ: P1. If, P2. Which, P3. Do, P4. While, P5. Module, P6. Print

■ ΕΝΤΟΛΗ P1: ΕΛΕΓΧΟΣ ΣΥΝΘΗΚΗΣ ΜΕ ΔΙΑΚΡΙΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ

If[*Συνθήκη, ΑποτέλεσμαΓιαΑληθήΣυνθήκη*]

If[*Συνθήκη, ΑποτέλεσμαΓιαΑληθήΣυνθήκη, ΑποτέλεσμαΓιαΨευδήΣυνθήκη*]

If[*Συνθήκη, ΑποτέλεσμαΓιαΑληθήΣυνθήκη, ΑποτέλεσμαΓιαΨευδήΣυνθήκη, ΆλλοΑποτέλεσμα*]

Στην πρώτη μορφή της ελέγχει τη συνθήκη που δίνεται στο πρώτο όρισμα και αν η συνθήκη αυτή είναι σίγουρα αληθής (ισχύει), τότε δίνει το αποτέλεσμα που ορίζεται στο δεύτερο όρισμα. Στη δεύτερη μορφή της κάνει ακριβώς τα ίδια, αλλά αν η συνθήκη στο πρώτο όρισμα είναι σίγουρα ψευδής (δεν ισχύει), τότε δίνει το αποτέλεσμα που ορίζεται στο τρίτο όρισμα. Τέλος στην τρίτη μορφή της κάνει ό,τι και στη δεύτερη μορφή της, αλλά αν δε μπορεί με σιγουριά να αποφανθεί αν η συνθήκη στο πρώτο όρισμα είναι αληθής ή ψευδής, τότε δίνει το αποτέλεσμα που ορίζεται στο τέταρτο όρισμα. Σαν παράδειγμα, θεωρούμε τη φόρτιση δοκού $p(x)$ που είναι p_1 για $x < 1/2$ και p_2 για $x \geq 1/2$:

```
In[1]:= p[x_] = If[x < 1 / 2, p1, p2]

Out[1]= If[x < 1/2, p1, p2]
```

Με τον ορισμό της συναρτήσεως $p(x)$ μπορούμε να υπολογίζουμε τις τιμές της κατά μήκος της δοκού

```
In[2]:= values1 = {p[0], p[1 / 4], p[0.40], p[1 / 2], p[0.60], p[Sqrt[2]], p[3 / 4], p[1]}

Out[2]= {p1, p1, p1, p2, p2, p2, p2, p2}
```

Προφανώς δε μπορούμε να υπολογίσουμε την τιμή της για τη φανταστική μονάδα *i*

```
In[3]:= p[ⅈ]

        Less::nord : Invalid comparison with ⅈ attempted.

Out[3]= If[ⅈ < 1/2, p1, p2]
```

Κι ένα δεύτερο παράδειγμα: Για μια δοκό *b* ορίζουμε γενικά τη δυσκαμψία της (flexural rigidity) ίση με *EI*

```
In[4]:= FlexuralRigidity[b_] = If[b == beam, EI]

Out[4]= If[b == beam, EI]
```

Παίρνουμε έτσι το αποτέλεσμα *EI* για τη δυσκαμψία, αν βέβαια το πρώτο όρισμα είναι σίγουρα δοκός

```
In[5]:= bm = beam; FlexuralRigidity[bm]

Out[5]= EI
```

Αυτό όμως προφανώς δε συμβαίνει, αν το πρώτο όρισμα δεν είναι σίγουρα δοκός

```
In[6]:= FlexuralRigidity[plate]

Out[6]= If[plate == beam, EI]
```

■ ΕΝΤΟΛΗ P2: ΕΛΕΓΧΟΣ ΣΕΙΡΑΣ ΣΥΝΘΗΚΩΝ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ

**Which[*Συνθήκη-1, ΑποτέλεσμαΓιαΑληθήΤηΣυνθήκη-1,*
Συνθήκη-2, ΑποτέλεσμαΓιαΑληθήΤηΣυνθήκη-2ΌμωςΜεΤηΣυνθήκη-1ΜηΑληθή, . . .]**

Ελέγχει τη συνθήκη στο πρώτο όρισμα και αν είναι σίγουρα αληθής (ισχύει), δίνει το αποτέλεσμα στο δεύτερο όρισμα. Αν δε συμβαίνει αυτό, τότε ελέγχει τη δεύτερη συνθήκη στο τρίτο όρισμα και αν αυτή η δεύτερη συνθήκη είναι σίγουρα αληθής, δίνει το αποτέλεσμα στο τέταρτο όρισμα. Αν και η δεύτερη συνθήκη δεν είναι σίγουρα αληθής, προχωράει ανάλογα με τις επόμενες συνθήκες (αν βέβαια υπάρχουν) στα περιττά ορίσματα και για όποια πρώτη βρεθεί να είναι αληθής (αν φυσικά βρεθεί) δίνει το αποτέλεσμα που την ακολουθεί. Σαν παράδειγμα, θεωρούμε την ίδια ασυνεχή φόρτιση δοκού $p(x)$ της προηγούμενης εντολής **If**, εδώ όμως με τη χρήση της παρούσας εντολής **Which**

```
In[7]:= p[x_] = Which[x < 1 / 2, p1, x ≥ 1 / 2, p2];

In[8]:= values2 = {p[0], p[1 / 4], p[0.40], p[1 / 2], p[0.60], p[Sqrt[2]], p[3 / 4], p[1]}

Out[8]= {p1, p1, p1, p2, p2, p2, p2, p2}
```

Προφανώς πήραμε την ίδια λίστα τιμών με εκείνη που είχαμε πάρει στην προηγούμενη εντολή **If**

```
In[9]:= verification = values1 == values2

Out[9]= True
```

Και ένα κάπως πιο σύνθετο παράδειγμα χρήσεως της ίδιας εντολής **Which**, εδώ με πέντε συνθήκες και πέντε αντίστοιχα αποτελέσματα που τα υπολογίζουμε πολύ εύκολα με τη χρήση της συναρτήσεως $p(x)$

```
In[10]:= p[x_] = Which[x < 0.2, p1, x < 0.4, p2, x < 0.6, p3, x < 0.8, p4, x ≥ 0.8, p5];

In[11]:= Table[{x, p[x]}, {x, 0, 1, 0.15}]

Out[11]= {{0, p1}, {0.15, p1}, {0.3, p2}, {0.45, p3}, {0.6, p4}, {0.75, p4}, {0.9, p5}}
```

■ ΕΝΤΟΛΗ P3: ΕΠΑΝΑΛΗΨΗ ΔΙΑΔΙΚΑΣΙΑΣ ΟΡΙΣΜΕΝΕΣ ΦΟΡΕΣ

Do[*Διαδικασία*, {*ΑριθμόςΕπαναλήψεων*}]

Do[*Διαδικασία*, {*Μεταβλητή, ΤελικήΤιμήΜεταβλητής*}]

Do[*Διαδικασία*, {*Μεταβλητή, ΑρχικήΤιμήΜεταβλητής, ΤελικήΤιμήΜεταβλητής*}]

Do[*Διαδικασία*, {*Μεταβλητή, ΑρχικήΤιμήΜεταβλητής, ΤελικήΤιμήΜεταβλητής, ΒήμαΜεταβολής*}]

Στην πρώτη μορφή εκτελεί τη διαδικασία στο πρώτο όρισμα τόσες φορές όσες δηλώνει ο αριθμός των επαναλήψεων στο δεύτερο όρισμα. Στη δεύτερη μορφή στο δεύτερο όρισμα υπάρχει και μια μεταβλητή (π.χ. i ή j ή k) που παίρνει τιμές από το 1 μέχρι την τελική τιμή της μεταβλητής (η οποία καθορίζεται επίσης στη λίστα του δευτέρου ορίσματος) ανά μονάδα και για κάθε τιμή της μεταβλητής αυτής εκτελείται η διαδικασία στο πρώτο όρισμα. Στην τρίτη μορφή η μεταβλητή παίρνει τιμές από την αρχική τιμή της μέχρι και την τελική τιμή της (που καθορίζονται στην λίστα του δευτέρου ορίσματος) πάλι ανά μονάδα. Απλά τώρα σαν αρχική τιμή της μεταβλητής δεν υποτίθεται αυτόματα το 1. Τέλος στην τέταρτη μορφή της καθορίζεται επιπλέον και το βήμα μεταβολής της μεταβλητής από τη μια τιμή της στην επόμενη (και αυτό στη λίστα του δευτέρου ορίσματος). Σαν παράδειγμα, θεωρούμε τη γνωστή διαφορική εξίσωση της αποδομήσεως ρύπου με αρχική συνθήκη: πρόβλημα αρχικής τιμής:

```
In[12]:= {ode = c'[x] == -k c[x], ic = c[0] == c0};
```

Με αρχική προσέγγιση της συγκεντρώσεως του ρύπου την αρχική τιμή της

```
In[13]:= c[0, x_] = c0;
```

εφαρμόζουμε τη μέθοδο των διαδοχικών προσεγγίσεων του Picard για τις επόμενες προσεγγίσεις

```
In[14]:= Do[c[n, x_] = c0 - k Integrate[c[n - 1, ξ], {ξ, 0, x}] // Factor;
          Print["c[", n, ", x] = ", c[n, x]], {n, 1, 4}]
```

c[1, x] = $-c0\,(-1 + k\,x)$

c[2, x] = $\frac{1}{2}\,c0\,(2 - 2\,k\,x + k^2\,x^2)$

c[3, x] = $-\frac{1}{6}\,c0\,(-6 + 6\,k\,x - 3\,k^2\,x^2 + k^3\,x^3)$

c[4, x] = $\frac{1}{24}\,c0\,(24 - 24\,k\,x + 12\,k^2\,x^2 - 4\,k^3\,x^3 + k^4\,x^4)$

■ ΕΝΤΟΛΗ P4: ΕΠΑΝΑΛΗΨΗ ΔΙΑΔΙΚΑΣΙΑΣ ΟΣΟ ΙΣΧΥΕΙ ΜΙΑ ΣΥΝΘΗΚΗ

While[*Συνθήκη, Διαδικασία*]

Ελέγχει τη συνθήκη στο πρώτο όρισμα και αν είναι σίγουρα αληθής (ισχύει), τότε εκτελεί τη διαδικασία στο δεύτερο όρισμα. Μετά ξαναελέγχει τη συνθήκη στο πρώτο όρισμα (που μπορεί να έχει αλλάξει εξαιτίας της διαδικασίας που προηγήθηκε) και αν συνεχίζει να είναι σίγουρα αληθής, τότε ξαναεκτελεί τη διαδικασία. Αυτό συνεχίζεται όσο η συνθήκη στο πρώτο όρισμα συνεχίζει να είναι σίγουρα αληθής. Μόλις παύσει να είναι, διακόπτεται η εκτέλεση της εντολής. Ακολουθεί το ίδιο παράδειγμα με εκείνο στην προηγούμενη εντολή **Do**, δηλαδή το παράδειγμα για την αποδόμηση ρύπου, τώρα όμως με τη χρήση της παρούσας εντολής **While** και για $n < 5$, με τα ίδια ακριβώς αποτελέσματα όπως και προηγουμένως με την εντολή **Do**

```
In[15]:= n = 1; While[n < 5, c[n, x_] = c0 - k Integrate[c[n - 1, ξ], {ξ, 0, x}] // Factor;
          Print["c[", n, ", x] = ", c[n, x]]; n = n + 1]
```

c[1, x] = $-c0\,(-1 + k\,x)$

c[2, x] = $\frac{1}{2}\,c0\,(2 - 2\,k\,x + k^2\,x^2)$

c[3, x] = $-\frac{1}{6}\,c0\,(-6 + 6\,k\,x - 3\,k^2\,x^2 + k^3\,x^3)$

c[4, x] = $\frac{1}{24}\,c0\,(24 - 24\,k\,x + 12\,k^2\,x^2 - 4\,k^3\,x^3 + k^4\,x^4)$

■ ΕΝΤΟΛΗ P5: ΧΡΗΣΗ ΤΟΠΙΚΩΝ ΜΕΤΑΒΛΗΤΩΝ ΣΕ ΔΙΑΔΙΚΑΣΙΑ

Module[*ΛίσταΤοπικώνΜεταβλητών, Διαδικασία*]

Πρόκειται για μια διαδικασία συγκεντρωμένη σε μία εντολή με τη χρήση τοπικών μεταβλητών. Αυτές αναφέρονται στη λίστα του πρώτου ορίσματος της εντολής είτε απλά σαν σύμβολα είτε μαζί με τις αρχικές τιμές τους. Οι τιμές των τοπικών μεταβλητών είναι εσωτερικές στην εντολή **Module** και δεν είναι διαθέσιμες στο notebook έξω από την εντολή **Module**. Γι' αυτό και τις αποκαλούμε τοπικές μεταβλητές (local variables). Σημειώνουμε επίσης ότι οι επιμέρους εντολές στη διαδικασία (στο δεύτερο όρισμα) χωρίζονται μεταξύ τους με Αγγλική άνω τελεία (το ;) αποτελώντας ένα μόνο όρισμα. Ξανακάνουμε το παράδειγμα της συγκεντρώσεως ρύπου των δύο προηγούμενων εντολών, τώρα με τοπική μεταβλητή την *m* στην εντολή **Module**, εδώ για *m* < 4. (Η εξωτερική τιμή της *m*, αν υπάρχει, δεν αλλάζει!):

```
In[16]:= Module[{m}, m = 1;
          While[m < 4, c[m, x_] = c0 - k Integrate[c[m - 1, ξ], {ξ, 0, x}] // Factor;
           Print["c[", m, ", x] = ", c[m, x]]; m = m + 1]]
```

c[1, x] = $-\text{c0}\,(-1+k\,x)$

c[2, x] = $\frac{1}{2}\,\text{c0}\,(2-2\,k\,x+k^2\,x^2)$

c[3, x] = $-\frac{1}{6}\,\text{c0}\,(-6+6\,k\,x-3\,k^2\,x^2+k^3\,x^3)$

Αυτό γίνεται λίγο καλύτερα με την αρχική τιμή *m* = 1 της τοπικής μεταβλητής *m* να ορίζεται στο πρώτο όρισμα της εντολής, δηλαδή στην ίδια τη λίστα των τοπικών μεταβλητών (εδώ μιας τοπικής μεταβλητής):

```
In[17]:= Module[{m = 1}, While[m < 4, c[m, x_] = c0 - k Integrate[c[m - 1, ξ], {ξ, 0, x}] // Factor;
           Print["c[", m, ", x] = ", c[m, x]]; m = m + 1]]
```

c[1, x] = $-\text{c0}\,(-1+k\,x)$

c[2, x] = $\frac{1}{2}\,\text{c0}\,(2-2\,k\,x+k^2\,x^2)$

c[3, x] = $-\frac{1}{6}\,\text{c0}\,(-6+6\,k\,x-3\,k^2\,x^2+k^3\,x^3)$

Εδώ έξω από την εντολή **Module** η τοπική μεταβλητή *m* δεν έχει καμία τιμή, είναι ένα απλό σύμβολο:

```
In[18]:= m

Out[18]= m
```

■ ΕΝΤΟΛΗ P6: ΕΜΦΑΝΙΣΗ ΠΑΡΑΣΤΑΣΕΩΣ ΣΤΗΝ ΟΘΟΝΗ

Print[*Παράσταση-1, Παράσταση-2, Παράσταση-3, . . .*]

Δεν πρόκειται για εντολή προγραμματισμού. Απλά εμφανίζει στην οθόνη τις παραστάσεις (κάθε μορφής, ακόμη και συμβολοσειρές, κείμενο) στα ορίσματά της τη μία μετά την άλλη και σε διαφορετικές γραμμές. Αυτό δε χρειάζεται συνήθως. Είναι όμως χρήσιμο στις εντολές διαδικαστικού προγραμματισμού **Do**, **While** και **Module** που προηγήθηκαν. Παράδειγμα αναφέρθηκε ήδη στις τρεις αυτές εντολές.

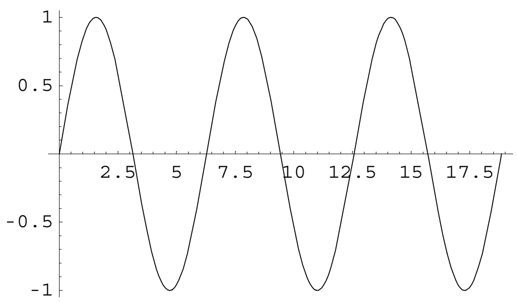
■ *Notebook AN: Animations-Introduction*
ΚΙΝΗΣΗ ΣΕ ΣΧΗΜΑΤΑ (ANIMATIONS) ΜΕ ΤΗ *MATHEMATICA*

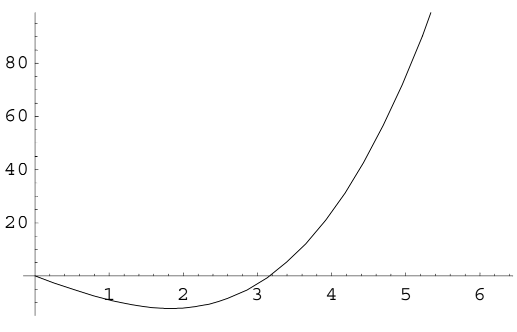
■ ΓΕΝΙΚΑ ΓΙΑ ANIMATIONS ΜΕ ΤΗ *MATHEMATICA*

Όπως ήδη γνωρίζουμε, με τη *Mathematica* μπορούμε να δημιουργήσουμε ποικίλα σχήματα, για παράδειγμα γραφικές παραστάσεις συναρτήσεων που είναι συχνά χρήσιμες στον Πολιτικό Μηχανικό. Στο notebook αυτό θα δείξουμε πώς ακριβώς μπορούμε να δημιουργούμε κίνηση σε σχήματα ή κινούμενα σχήματα, "ζωντανά" σχήματα, στα Αγγλικά animations. (Αυτός ό Αγγλικός όρος, animations, είναι ένας όρος που θα το χρησιμοποιούμε και εδώ, επειδή είναι μονολεκτικός, άρα και εύχρηστος. Είναι επίσης και καθιερωμένος όρος.) Όπως συμβαίνει και στον κινηματογράφο, τέτοια δυνατότητα δεν υπάρχει απευθείας. Υπάρχει όμως η δυνατότητα σε μια συγκεκριμένη θέση της οθόνης να παρουσιάζονται διαφορετικά σχήματα το ένα αμέσως μετά το άλλο. Αυτό το υποστηρίζει η *Mathematica* με τη χρήση της επιλογής **Cell -> Animate Selected Graphics** από τις επιλογές **Cell** των menus της. Απόλυτα ισοδύναμη και κάπως πιο εύχρηστη είναι η συντόμευση **Ctrl Y** της επιλογής αυτής **Cell -> Animate Selected Graphics**. Είναι στ' αλήθεια τόσο εύκολο! Μετά τη δημιουργία μιας animation με τη *Mathematica* μπορούμε να τη σώζουμε σε αρχείο animated **gif** και να την καλούμε εντελώς έξω από τη *Mathematica* ή από το διαδίκτυο.

Για να εργασθεί η *Mathematica* με τον πιο πάνω τρόπο θέτοντας σε μια θέση της οθόνης περισσότερα από ένα σχήματα, βέβαια το ένα μετά το άλλο (όχι ταυτόχρονα!), μας ζητάει πρώτα να έχουμε δημιουργήσει τα σχήματα αυτά και να τα έχουμε σαν αποτέλεσμα μιας εντολής της, που συχνά (αλλ' όχι βέβαια αναγκαστικά) βασίζεται στην εντολή **Plot**. Ας δούμε ένα σχετικό απλό παράδειγμα για την ημιτονική συνάρτηση sin x και για την πολυωνυμική συνάρτηση $x^3 - 10\,x$ σε λίστα με την εντολή **Plot**:

```
In[1]:= {Plot[Sin[x], {x, 0, 6 π}, ImageSize → 250], Plot[x^3 - 10 x, {x, 0, 2 π}, ImageSize → 250]}
```






```
Out[1]= {- Graphics -, - Graphics -}
```

Άρα δημιουργήσαμε δύο χωριστές γραφικές παραστάσεις (με δύο κλήσεις της εντολής **Plot**, αλλά στην ίδια σύνθετη εντολή): μία για την ημιτονική συνάρτηση sin x και μία για την πολυωνυμική συνάρτηση $x^3 - 10\,x$. Αυτές παρουσιάζονται η μία κάτω από την άλλη. Αν τώρα εμείς πατήσουμε με το ποντίκι απλά **Cell -> Animate Selected Graphics** από τις επιλογές ή πολύ πιο απλά **Ctrl Y** από το πληκτρολόγιο, τίποτε απολύτως δε θα συμβεί. Αν τώρα επιλέξουμε την πρώτη γραφική παράσταση μαυρίζοντας με πάτημα του ποντικού την κατακόρυφη μπλε γραμμή ακριβώς δεξιά της (δεξιά στην πρώτη πιο πάνω γραφική παράσταση: την εσωτερική μπλε γραμμή) και πατήσουμε **Ctrl Y**, πάλι τίποτε δε θα συμβεί. Το ίδιο κι αν μαυρίσουμε την εσωτερική μπλε γραμμή που αντιστοιχεί στη δεύτερη γραφική παράσταση: πάλι τίποτε δε θα συμβεί. Είναι λογικά αυτά, γιατί δε μπορούμε να πάρουμε κίνηση με ένα μόνο σχήμα.

Τώρα ας επιλέξουμε και τις δύο πιο πάνω γραφικές παραστάσεις (ταυτόχρονα και για την ημιτονική και για την πολυωνυμική συνάρτηση) μαυρίζοντας τη λίγο πιο έξω κατακόρυφη μπλε γραμμή που αντιστοιχεί και στις δύο μαζί αυτές γραφικές παραστάσεις, δηλαδή σε ολόκληρη την έξοδο της παραπάνω σύνθετης εντολής **Plot** (λίστα με χρήση δύο φορές της εντολής **Plot**). Έτσι "σημαδέψαμε" ταυτόχρονα και τις δύο πιο πάνω γραφικές παραστάσεις: και τα δύο σχήματα. Πατάμε τώρα **Cell -> Animate Selected Graphics** ή πιο απλά **Ctrl Y** και το θαύμα έγινε: τη θέση της πρώτης γραφικής παραστάσεως (αν είναι ορατή στην οθόνη μας) την παίρνει μια η ημιτονική συνάρτηση (η πρώτη γραφική παράσταση) και μια η πολυωνυμική συνάρτηση (η δεύτερη γραφική παράσταση). Εναλλάσσονται λοιπόν οι δύο αυτές γραφικές παραστάσεις στην ίδια ακριβώς θέση της οθόνης. Το σχήμα κινείται! Αρκετά ωραία. (Ας σημειώσουμε εδώ πως αν η πρώτη γραφική παράσταση δεν είναι ορατή στην οθόνη, δηλαδή είναι λίγο πιο πάνω, τότε θα αρχίσει να κινείται το σχήμα που αφορά στην πρώτη ορατή στην οθόνη γραφική παράσταση. Λεπτομέρεια βέβαια, αλλ' ας την έχουμε υπόψη μας.) Αν τέλος επιθυμούμε να διακόψουμε αυτήν την τόσο μικρή "ταινία" που δημιουργήσαμε μέχρι ένα μικρό βαθμό (και εδώ με δύο μόνο γραφικές παραστάσεις που εναλλάσσονται στην ίδια θέση της οθόνης), απλά πατάμε το ποντίκι κάπου στην οθόνη και τέρμα η οπτική αυτή παράσταση. Τέλος! Μπορούμε να την ξαναρχίσουμε με **Cell -> Animate Selected Graphics** ή πιο απλά **Ctrl Y**, όπως και πριν. Από 'δώ και πέρα θα γράφουμε μόνο **Ctrl Y** εννοώντας βέβαια ότι μπορεί εναλλακτικά να χρησιμοποιηθεί και η επιλογή **Cell -> Animate Selected Graphics** από τα menus.

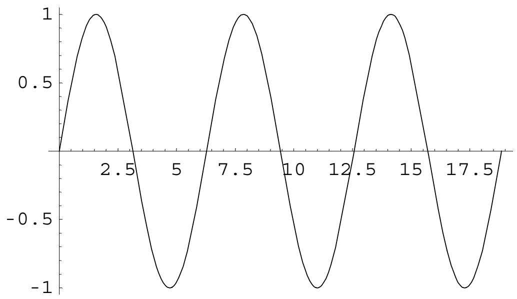
Ας σημειώσουμε στο σημείο αυτό ότι αν μαυρίσουμε την ακόμη πιο μεγάλη κατακόρυφη μπλε γραμμή που αφορά σε ολόκληρη την πιο πάνω εντολή, δηλαδή περιλαμβάνει και όλη την είσοδο (input) και όλη την έξοδο (output) της εντολής, ακόμη και τις ενδείξεις **- Graphics -** στη λίστα στο τέλος της εντολής, και πάλι όλα πάνε μια χαρά. Η κίνηση του ενός σχήματος (συνήθως του πρώτου) είναι και πάλι γεγονός. Το δίδαγμα είναι πως πατάμε όποια κατακόρυφη μπλε γραμμή μας αρέσει δεξιά στην οθόνη μας, αρκεί αυτή η κατακόρυφη μπλε γραμμή να περιλαμβάνει οπωσδήποτε πάνω από μία γραφικές παραστάσεις. (Με μία μόνο γραφική παράσταση δεν υπάρχει κίνηση!) Τότε η πρώτη στην οθόνη (η πιο ψηλά στην οθόνη) γραφική παράσταση θα αρχίσει να εναλλάσσεται με όλες τις υπόλοιπες γραφικές παραστάσεις που έχουμε σημαδέψει με το μαύρισμα της κατάλληλης κατακόρυφης μπλε γραμμής δεξιά στο notebook.

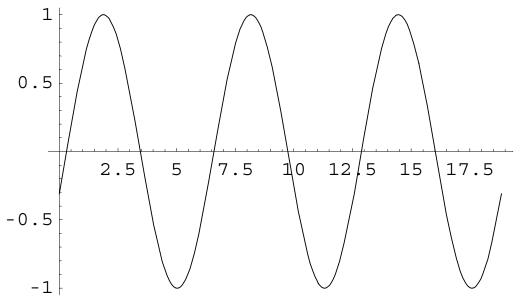
■ ΚΙΝΗΣΗ ΗΜΙΤΟΝΙΚΗΣ ΚΑΜΠΥΛΗΣ

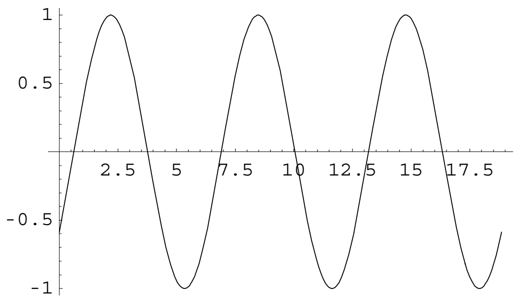
Ωραία ώς εδώ, αλλά τί μας νοιάζουν όλα αυτά; Δεν καταφέραμε και τίποτε το σημαντικό μέχρι τώρα! Μάλλον ζαλιστήκαμε με δύο εντελώς διαφορετικές γραφικές παραστάσεις (η πρώτη για μια ημιτονική συνάρτηση, ενώ η δεύτερη για μια πολυωνυμική συνάρτηση) να εναλλάσσονται στο ίδιο σχήμα. Σε τί μας βοηθάει μια τέτοια εναλλαγή πέρα από το να μας ζαλίζει; Η απάντηση είναι απλή: το πιο πάνω τόσο απλοϊκό και ουσιαστικά άχρηστο παράδειγμα μας έδειξε καθαρά τον τρόπο εργασίας της *Mathematica* στην κίνηση σε σχήματα: στην animation. Από 'δώ και κάτω η ιδέα είναι τα σχήματα που εμφανίζουμε στην ίδια θέση το ένα αμέσως μετά το άλλο με την επιλογή **Ctrl Y** να μην είναι άσχετα μεταξύ τους, αλλά σχετικά με πολύ μικρές διαφορές το ένα από το αμέσως επόμενό του (ή αμέσως προηγούμενό του).

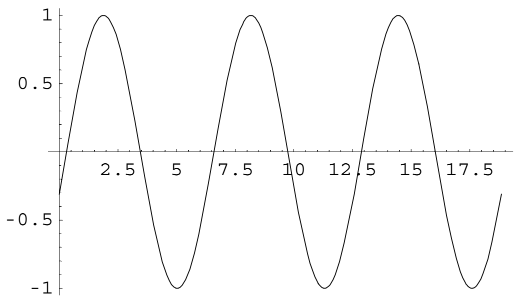
Τότε αντί απλά να ζαλιζόμαστε θα έχουμε την αίσθηση της κινήσεως του σχήματος. Δε θα είναι βέβαια αληθινή κίνηση, θα συνεχίσει να είναι απλά εναλλαγή σχημάτων, αλλά θα μας δίνει την ψευδαίσθηση κινήσεως, ακριβώς όπως συμβαίνει και στον κινηματογράφο. Ας δούμε ένα σχετικό, αλλά και πάλι απλό πάραδειγμα, που αφορά ξανά στην ημιτονική συνάρτηση, τώρα όμως χωρίς πια παρέα της μια πολυωνυμική συνάρτηση. (Σημειώνουμε τέλος τον καθορισμό μεγέθους των σχημάτων με την επιλογή **ImageSize**.)

```
In[2]:= {Plot[Sin[x], {x, 0, 6 π}, ImageSize → 250],
          Plot[Sin[x - π / 10], {x, 0, 6 π}, ImageSize → 250],
          Plot[Sin[x - 2 π / 10], {x, 0, 6 π}, ImageSize → 250],
          Plot[Sin[x - π / 10], {x, 0, 6 π}, ImageSize → 250]};
```





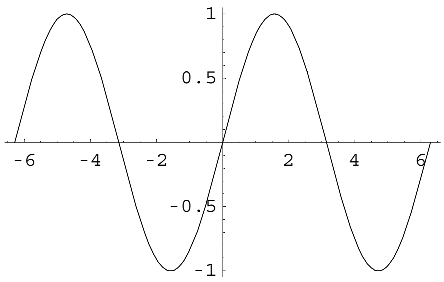


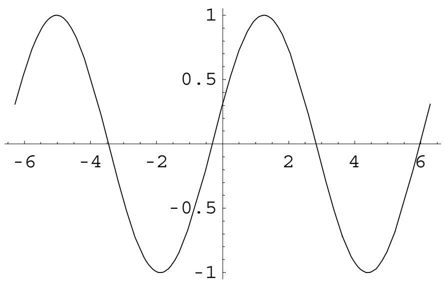


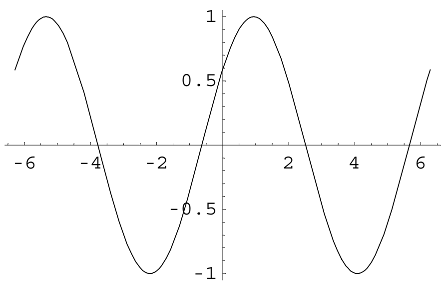
Τώρα έχουμε πάρει τέσσερις γραφικές παραστάσεις: της αρχικής ημιτονικής συναρτήσεως η πρώτη και με μετατοπίσεις της κατά π/ 10 (σαν να εκινείτο προς τα δεξιά) και πίσω μετά. Επομένως ας μαυρίσουμε τη σχετική κατακόρυφη μπλέ γραμμή δεξιά με το ποντίκι (είτε τη μπλε γραμμή μόνο με τα τέσσερα αυτά σχήματα είτε την αμέσως επόμενη προς τα έξω μπλε γραμμή με ολόκληρη την πιο πάνω εντολή και την είσοδό της και ας πατήσουμε **Ctrl Y**. Τότε θα παρατηρήσουμε την κίνηση της γραφικής παραστάσεως (της πρώτης γραφικής παραστάσεως στην οθόνη μας) προς τα δεξιά και μετά προς τα αριστερά. Έχουμε λοιπόν κίνηση στη γραφική παράσταση; Όχι βέβαια, με τίποτε δεν έχουμε αληθινή κίνηση, αλλ' έχουμε μια ψευδαίσθηση κινήσεως εξαιτίας της εναλλαγής των τεσσάρων παραπάνω γραφικών παραστάσεων (η δεύτερη μάλιστα εκ προθέσεως είναι ίδια με την τέταρτη) με γρήγορο ρυθμό πάνω στην οθόνη μας. Έχουμε λοιπόν πετύχει την ψευδαίσθηση κινήσεως της γραφικής μας παραστάσεως κι αυτήν την ψευδαίσθηση θα την αποκαλούμε από 'δώ και μπρος animation ή κίνηση σε σχήμα (εδώ γραφική παράσταση).

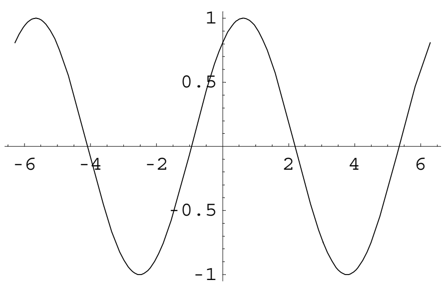
Στο σημείο αυτό ας παρατηρήσουμε ότι είναι μάλλον κουτό να δίνουμε μακριές σύνθετες εντολές στη *Mathematica*, όπως η πιο πάνω εντολή σε λίστα με τέσσερα στοιχεία: τέσσερις χωριστές γραφές της εντολής **Plot**. Ακριβώς το ίδιο μπορούμε να το πετύχουμε με τη χρήση μία ή το πολύ δύο φορές της εντολής **Table**. Αυτή δημιουργεί τα ίδια ακριβώς σχήματα με μία ή δύο μόνο εμφανίσεις της εντολής **Plot**. Ας κάνουμετώρα ένα σοβαρό παράδειγμα κινήσεως προς τα αριστερά της ημιτονικής συναρτήσεως:

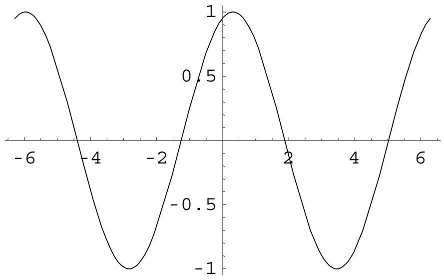
```
In[3]:= SinAnimation = Table[Plot[Sin[x + k π / 10], {x, -2 π, 2 π}, ImageSize → 220], {k, 0, 19}];
```

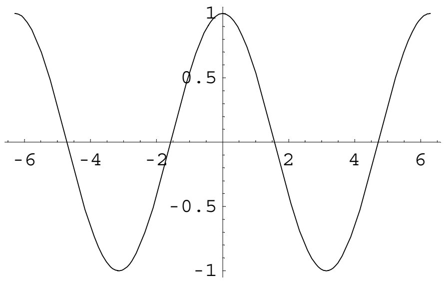


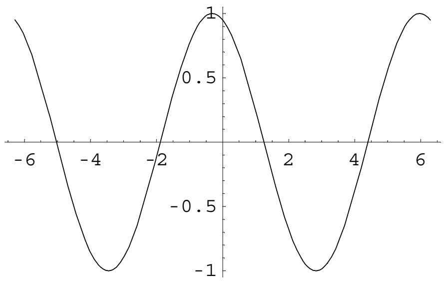


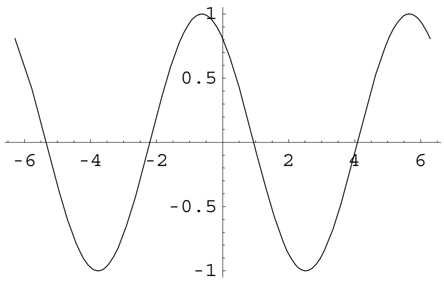


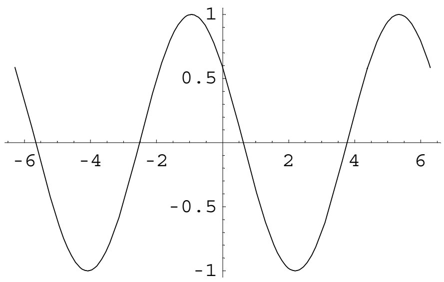


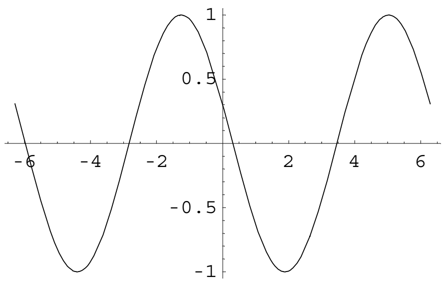

1
0.5
-6 -4 -2 2 4 6
-0.5
-1

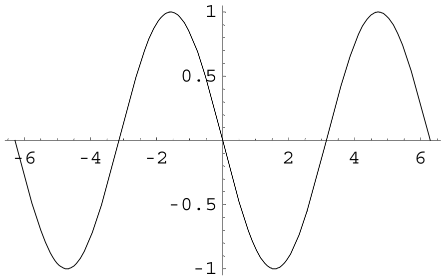

1
0.5
-6 -4 -2 2 4 6
-0.5
-1

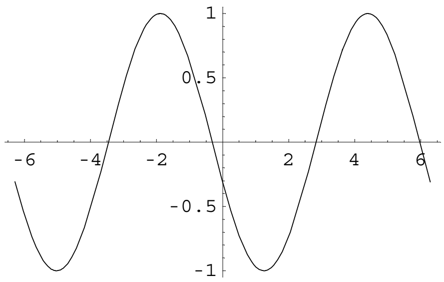

1
0.5
-6 -4 -2 2 4 6
-0.5
-1

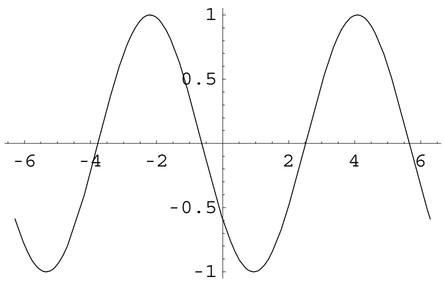

1
0.5
-6 -4 -2 2 4 6
-0.5
-1

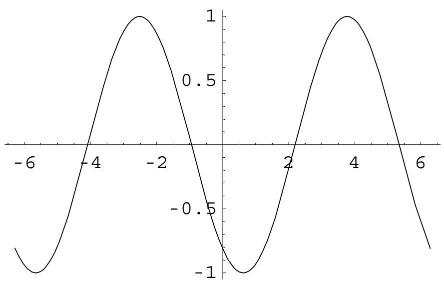

1
0.5
-6 -4 -2 2 4 6
-0.5
-1

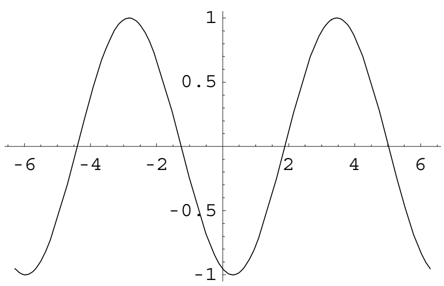

1
0.5
-6 -4 -2 2 4 6
-0.5
-1

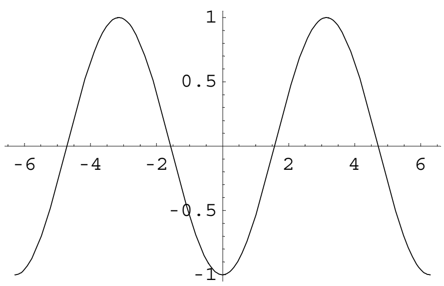

1
0.5
-6 -4 -2 2 4 6
-0.5
-1

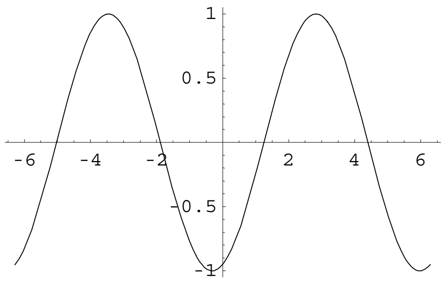

1
0.5
-6 -4 -2 2 4 6
-0.5
-1

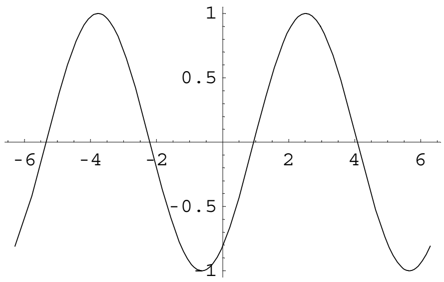

1
0.5
-6 -4 -2 2 4 6
-0.5
-1

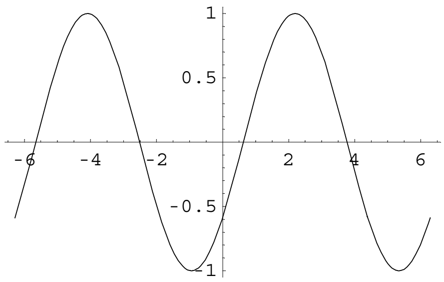

1
0.5
-6 -4 -2 2 4 6
-0.5
-1

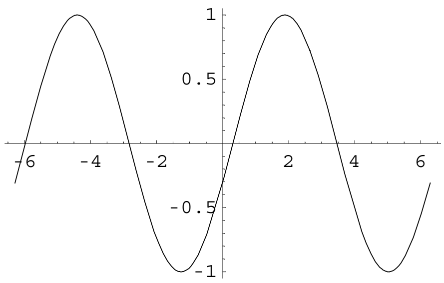

1
0.5
-6 -4 -2 2 4 6
-0.5
-1


1
0.5
-6 -4 -2 2 4 6
-0.5
-1







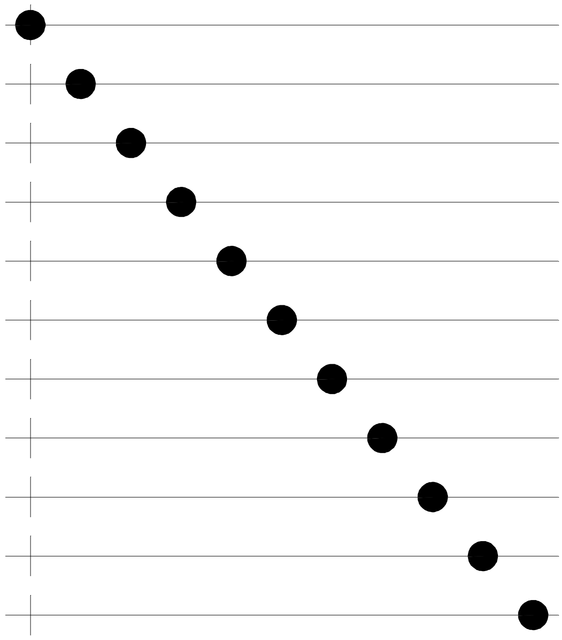


Παρατηρούμε ότι προέκυψαν 20 σχήματα, λίγο διαφορετικά το ένα από το άλλο, και ότι η εντολή **Ctrl Y** για την κίνηση της γραφικής παραστάσεως (εδώ της ημιτονικής συναρτήσεως sin x) δουλεύει μια χαρά. Έχουμε κίνηση της ημιτονικής καμπύλης προς τα αριστερά. Αν όμως είχαμε βάλει – αντί για + στον όρο $k\pi/10$, ε τότε η κίνηση θα ήταν προς τα δεξιά. Για να μη σπαταλάμε χώρο, θα σβήνουμε συχνά στα notebooks εφαρμογών της animation πολλά από τα σχήματα που θα προκύπτουν από τις ανάλογες εντολές.

■ ΕΥΘΥΓΡΑΜΜΗ ΚΙΝΗΣΗ ΚΥΚΛΟΥ

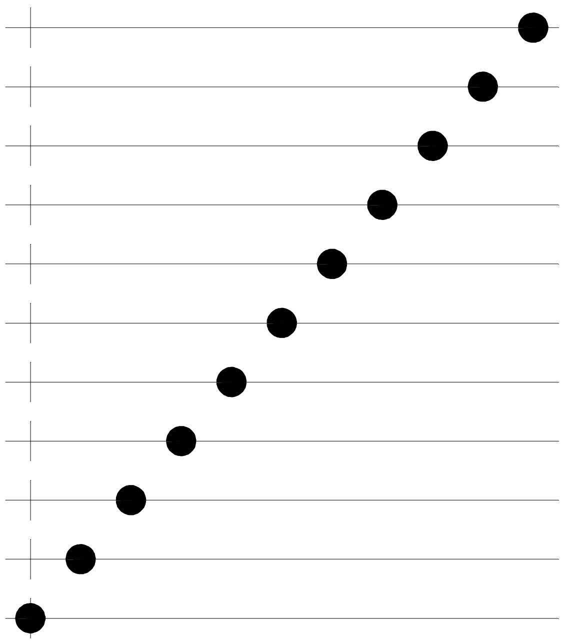
Το επόμενο σχήμα μας δείχνει την κίνηση ενός μικρού κύκλου ακτίνας $a = 0.03$ πάνω στον άξονα Ox από $x = 0$ μέχρι $x = 1$ με σταθερή ταχύτητα. Χρησιμοποιούμε πάλι την εντολή **Table**, εδώ μαζί με τις εντολές **Disk**, **Graphics** και **Show.** Πάει λοιπόν με **Ctrl Y** δεξιά το κυκλάκι, κινείται! Όταν βέβαια τελειώσει αυτή η παράσταση και το κυκλάκι φθάσει στο σημείο $x = 1$, τότε γυρίζει απότομα πίσω και ξαναρχίζει να κινείται από την αρχή: από το σημείο $x = 0$. Δοκιμάστε το με μαύρισμα (με πάτημα του ποντικιού) της κατακόρυφης μπλε γραμμής δεξιά στο notebook που περιλαμβάνει όλα τα πιο κάτω σχήματα και μετά απλά **Ctrl Y**:

```
In[4]:= tb1 = Table[Show[Graphics[Disk[{x, 0}, 0.03], AspectRatio → 0.08 / 1.1, PlotRange →
          {{-0.05, 1.05}, {-0.04, 0.04}}, Axes → True, Ticks → None]], {x, 0, 1, 0.1}];
```



Να και η αντίστοιχη κίνηση προς τα αριστερά. Και αυτή ενεργοποιείται με **Ctrl Y**, αρκεί βέβαια να επιλεγούν όλα τα σχήματα, όλες οι θέσεις του κύκλου. Αυτό μπορεί να γίνει, ξαναλέμε, είτε (α) με μαύρισμα της κατακόρυφης μπλε γραμμής στην έξοδο της πιο κάτω εντολής είτε (β) με μαύρισμα της επόμενής της (προς τα έξω) κατακόρυφης μπλε γραμμής, η οποία περιλαμβάνει και την είσοδο της ίδιας εντολής.

```
In[5]:= tb2 = Table[Show[Graphics[Disk[{x, 0}, 0.03], AspectRatio → 0.08 / 1.1, PlotRange →
          {{-0.05, 1.05}, {-0.04, 0.04}}, Axes → True, Ticks → None]], {x, 1, 0, -0.1}];
```



Με λίγη προσοχή μπορούμε (α) να μαυρίσουμε και τις δύο πιο πάνω εντολές με τα αποτελέσματα **tb1** και **tb2** ταυτόχρονα και την ενδιάμεση παράγραφο κειμένου. (Δεν πειράζει, επειδή η ενδιάμεση παράγραφος κειμένου δεν έχει κανένα σχήμα!) Άρα μαυρίζουμε συνολικά τρεις διαδοχικές κατακόρυφες μπλέ γραμμές δεξιά στο notebook και πατάμε και πάλι **Ctrl Y**. Τώρα η κίνηση θα είναι και προς τα δεξιά και προς τα αριστερά: συνεχής κίνηση, η σωστή κίνηση. (Ένα μικρό διαλειμματάκι που γίνεται στην κίνηση αυτή στα σημεία $x = 0$ και $x = 1$, όπου ***κακώς*** πήραμε διπλές καμπύλες, ξεφεύγει συνήθως από την προσοχή μας. Αυτό όμως δε θα το ξανακάνουμε! Δε θα το κάνουμε στις εφαρμογές του Πολιτικού Μηχανικού!) Αυτά!

■ ΤΙ ΜΑΘΑΜΕ ΓΙΑ ΤΗ ΔΗΜΙΟΥΡΓΙΑ ANIMATIONS ΜΕ ΤΗ MATHEMATICA

Μέχρι τώρα στο notebook αυτό για animations (κινήσεις σε σχήματα) με τη *Mathematica*

1. Μάθαμε τί ακριβώς είναι μια animation: κίνηση σε σχήμα, ένα κινούμενο σχήμα, ένα "ζωντανό" σχήμα.

2. Μάθαμε πώς να δημιουργούμε την ακολουθία των διαδοχικών σχημάτων που χρειάζεται μια animation: συνήθως με μία ή δύο το πολύ εφαρμογές της εντολής **Table**. Ανάλογα βέβαια ισχύουν με την εφαρμογή της παραπλήσιας εντολής **Do** και άλλων ειδικών εντολών για animations, όπως είναι η εντολή **Animate**.

3. Μάθαμε επίσης να "σημαδεύουμε" τα σχήματα αυτά στο notebook που εργαζόμαστε απλά μαυρίζοντας με το ποντίκι όχι τα ίδια τα σχήματα: την κατάλληλη ή τις κατάλληλες κατακόρυφες μπλε γραμμές τους δεξιά στο notebook. Οι γραμμές αυτές πρέπει οπωσδήποτε (α) να περιλαμβάνουν αριστερά τους όλα τα σχήματα που χρειαζόμαστε σε μια animation και (β) να μην περιλαμβάνουν κανένα απολύτως παράσιτο (παρασιτικό) σχήμα που δεν αφορά στην animation που δημιουργούμε. Τέλος ο αληθινός θρίαμβός μας: μάθαμε να πατάμε **Ctrl Y** (από το πληκτρολόγιο) ή ισοδύναμα **Cell -> Animate Selected Graphics** (από τις επιλογές, τα menus, με το ποντίκι) και να βλέπουμε την animation που ήδη δημιουργήσαμε να κινείται μπροστά μας. Όταν βαρεθούμε να παρατηρούμε αυτό το δημιούργημά μας, την animation, πατάμε κάπου στην τύχη το ποντίκι μέσα στο notebook και η παράσταση τελειώνει. Ίσως για να αρχίσει μια άλλη καλύτερη με μια καινούργια animation. Πολύ ωραία ώς εδώ! Και προχωράμε λίγο ακόμη: στο τελικό αρχείο.

■ Η ΔΗΜΙΟΥΡΓΙΑ ΤΟΥ ΑΡΧΕΙΟΥ GIF: Graphics Interchange File

Το τελευταίο μας βήμα, στο οποίο αφιερώνουμε αυτήν την ενότητα του notebook είναι η "εξαγωγή" της animation την οποία ήδη ετοιμάσαμε με τη *Mathematica* στον έξω κόσμο: στον κόσμο που γενικά δε χρησιμοποιεί τη *Mathematica*. Αυτό το πετυχαίνουμε με τη δημιουργία και τη χρήση ενός αρχείου **gif**. Στην κίνηση της ημιτονικής καμπύλης στην έξοδο της εντολής [3] έχουμε δημιουργήσει με την εντολή **Table** τον πίνακα **SinAnimation** με 20 γραφικές παραστάσεις της ημιτονικής συναρτήσεως sin, τη μία ελαφρά μετατοπισμένη ως προς την άλλη. Αυτόν τον πίνακα **SinAnimation** μπορούμε τώρα να τον "εξαγάγουμε" από τη *Mathematica* χρησιμοποιώντας μια ειδική εντολή εξόδου της *Mathematica*: την εντολή **Export**. Σκοπεύουμε έτσι να δημιουργήσουμε ένα αρχείο **gif** (**GIF**) με τα σχήματα που ετοιμάσαμε και ήδη διαθέτουμε στον πίνακα **SinAnimation**. Η σύνταξη αυτής της εντολής στην περίπτωσή μας είναι

Export["*ΠλήρεςΌνομαΑρχείου*", *ΤαΣχήματαΤηςAnimation*, ConversionOptions → {Loop → True}]

Στην περίπτωσή μας ο πίνακας με όλα τα σχήματα της animation που ήδη δημιουργήσαμε για την κίνηση της ημιτονικής καμπύλης είναι ο πίνακας **SinAnimation**. Το όνομα του αρχείου όπου θα βάλουμε όλα αυτά τα σχήματα δεν είναι ιδιαίτερα σημαντικό, πρέπει όμως οπωσδήποτε να τελειώνει σε **gif** (ή **GIF**, το ίδιο κάνει!). Εδώ επιλέχθηκε το όνομα **SineCurveAnimation.gif**. Το αρχείο αυτό το θέτουμε εδώ στο σκληρό δίσκο **D:** και μάλιστα στο φάκελο (directory) **cemb\animations**. Δεν έχουν ιδιαίτερη σημασία αυτά. Δεν ξεχνάμε βέβαια τα εισαγωγικά που περικλείουν το πλήρες όνομα του αρχείου. Το σημαντικό είναι να μην ξεχασθεί στο τέλος η επιλογή **ConversionOptions → {Loop → True}** για το αρχείο **gif** που εξάγουμε από τη *Mathematica*. Με την επιλογή αυτή δηλώνουμε στη *Mathematica* ότι αυτό το αρχείο **gif** (και είναι εδώ αρχείο **gif**, επειδή φτειάχθηκε σαν αρχείο **gif**) δεν είναι ένα κοινό, στατικό αρχείο **gif**. Είναι ένα αρχείο **gif** με animation: με κινούμενο σχήμα σ' αυτό. Φθάνουν οι επεξηγήσεις ώς εδώ. Να και η εντολή που δώσαμε:

```
In[6]:= Export["D:\cemb\animations\SineCurveAnimation.gif",
          SinAnimation, ConversionOptions → {Loop → True}]

Out[6]= D:\cemb\animations\SineCurveAnimation.gif
```

Με τον τρόπο αυτό ο πίνακας **SinAnimation** των γραφικών παραστάσεών μας φεύγει (εξάγεται: εντολή **Export** της *Mathematica* για την εξαγωγή αποτελεσμάτων: εξάγαγε, να εξαγάγεις σημαίνει) από τη *Mathematica* προς τον έξω κόσμο. Και τώρα βγαίνουμε (προσωρινά ή οριστικά) από τη *Mathematica* και επαληθεύουμε από τον υπολογιστή μας (π.χ. μέσω των εικονιδίων **My Computer**, **Local Disk (D:)**, κλπ.) πως πραγματικά, στ' αλήθεια υπάρχει το αρχείο **gif** που με τόσο κόπο δημιουργήσαμε στο σκληρό δίσκο **D**: και μάλιστα στο directory **D:\cemb\animations.** Μας έφυγε λίγο η ανησυχία μήπως κάτι δεν έχει πάει καλά. Ευτυχώς όλα πήγανε καλά ώς εδώ και το αρχείο **gif** που με τόσο κόπο φτειάξαμε (χάρη στη *Mathematica* βέβαια) πραγματικά υπάρχει εκεί όπου ζητήσαμε να πάει. Μπράβο μας λοιπόν! Και τί κάνουμε τώρα;

Τώρα απλά πατάμε με το ποντίκι πάνω στο εικονίδιο αυτού του αρχείου **gif**, δηλαδή του αρχείου **SineCurveAnimation.gif** και, ω του θαύματος, βλέπουμε μπροστά μας να ανοίγει ένα παράθυρο του υπολογιστή μας και να παρουσιάζεται η κίνηση της ημιτονικής μας καμπύλης έξω από τη *Mathematica*: δεν τη χρειαζόμαστε πια! Να λοιπόν που η animation που φτειάξαμε λειτουργεί τέλεια. Με τις επιλογές που έχουμε στα μικρά εικονίδια κάτω στο παράθυρο αυτό μπορούμε να μεγαλώσουμε το σχήμα με την animation με **Zoom In (+)**, να τη μικρύνουμε με **Zoom Out (−)**, να την κάνουμε slide, να την αποθηκεύσουμε, κλπ. Καμαρώνουμε βέβαια για τη δουλειά μας, τη βλέπουμε και την ξαναβλέπουμε. Κίνηση σαν μια πολύ μικρή ταινία κινηματογράφου. Όταν τελειώσουμε, κλείνουμε το παράθυρο που είχε ανοιχθεί πατώντας το κόκκινο X πάνω δεξιά στην οθόνη και επιτρέφουμε πίσω στον υπολογιστή μας με καινούργια ίσως σχέδια δημιουργίας animations με τη *Mathematica* (ή και με άλλα προγράμματα που δημιουργούν animations στο μέλλον, γιατί όχι;) σε αρχεία animated **gif** (ή και σε άλλων ειδών αρχεία στο μέλλον, γιατί όχι;).

Τώρα που πήραμε θάρρος μπορούμε να δημιουργήσουμε κι ένα αρχείο **gif** για την κίνηση του κύκλου στο διάστημα [0, 1] στο δεύτερο παράδειγμα. Απόλυτα ανάλογα με προηγουμένως δημιουργούμε το αρχείο:

```
In[7]:= Export["C:\CircleMovement.gif", {tb1, tb2}, ConversionOptions → {Loop → True}];
```

Στο αρχείο αυτό βάλαμε όλα τα σχήματα: και τα σχήματα **tb1** και τα σχήματα **tb2** που δημιουργήσαμε πιο πάνω, ώστε να έχουμε πλήρη κίνηση του κύκλου και προς δεξιά και προς τα αριστερά. Και πάλι, αφού επαληθεύσουμε ότι κι αυτό αρχείο **gif**: το **CircleMovement.gif** είναι πράγματι στη θέση του, το καλούμε (πατώντας το εικονίδιό του) και παρατηρούμε την κίνηση του κύκλου στην animation την οποία φτειάξαμε.

Τις animations που φτειάξαμε και που θα φτειάξουμε σε εφαρμογές του Πολιτικού Μηχανικού (ιδίως σε προβλήματα ταλαντώσεων: μηχανικών συστημάτων, δοκών, πλακών, κτιρίων και επίσης λυγισμού, ροής ρευστών, κλπ.) μπορούμε να τις στείλουμε σε φίλους ή να τις βάλουμε σε μια ιστοσελίδα. Έτσι θα μπορέσει ίσως να δημιουργηθεί μια "βιβλιοθήκη" με animations ενδιαφέροντος του Πολιτικού Μηχανικού.

Σημειώνεται τέλος ότι στο Τεύχος 2 των ***Εφαρμοσμένων Μαθηματικών ΙΙ για Πολιτικούς Μηχανικούς*** υπάρχουν δύο notebooks για animations που αφορούν σε ταλαντώσεις: συστήματος μάζας–ελατηρίου και μονώροφου κτιρίου. Και ανάλογα στο Τεύχος 2 των ***Εφαρμοσμένων Μαθηματικών ΙΙΙ για Πολιτικούς Μηχανικούς*** υπάρχουν επίσης δύο notebooks για animations που αφορούν σε ταλαντώσεις (ιδιοταλαντώσεις): αμφιέρειστης δοκού και ορθογωνικής πλάκας. Βέβαια η μελέτη του παρόντος εισαγωγικού notebook για τις animations είναι αναγκαία, ώστε να προχωρήσει κανείς πιο πέρα στις εφαρμογές αυτές.

ΒΙΒΛΙΟΓΡΑΦΙΑ

• Στα Ελληνικά το χρονικά πρώτο βιβλίο για τη *Mathematica* είναι εκείνο του Κ. Ε. Παπαδάκη. Πρόκειται για ένα άριστα γραμμένο και καλοτυπωμένο βιβλίο. Αυτό καλύπτει εκτενώς τις περισσότερες χρήσιμες εντολές της *Mathematica* και περιέχει πάρα πολλά παραδείγματα και εφαρμογές:

1. Παπαδάκης, Κ. Ε. (2002), *Εισαγωγή στο Mathematica*, 2η Έκδοση. Εκδόσεις Τζιόλα, Θεσσαλονίκη.

• Εκτός από το βιβλίο αυτό υπάρχουν και πολύ λίγα ακόμη αξιόλογα βιβλία για τη *Mathematica* στα Ελληνικά. Βασικά συνιστώνται τα εξής τέσσερα βιβλία:

1. Don, E. (2005), *Mathematica*. Μετάφραση από την Αγγλική Έκδοση με τον ίδιο τίτλο, Schaum's Outlines, McGraw-Hill, New York. Ελληνική Έκδοση: Κλειδάριθμος, Αθήνα.
2. Θεοδώρου, Γ., Θεοδώρου, Χ. (2005), *Mathematica και Εφαρμογές*. Εκδόσεις Γιαχούδη, Θεσσαλονίκη.
3. Καραμπετάκης, Ν., Σταματάκης, Σ., Ψωμόπουλος, Ε. (2004), *Μαθηματικά & Προγραμματισμός στο Mathematica*. Εκδόσεις Ζήτη, Θεσσαλονίκη.
4. Τραχανάς, Σ. (2001), *Mathematica και Εφαρμογές για Μαθηματικούς, Φυσικούς και Μηχανικούς*. Πανεπιστημιακές Εκδόσεις Κρήτης, Ηράκλειο. *(Σημειώνεται ότι σ' αυτό το βιβλίο χρησιμοποιείται ο όρος «η Mathematica» που υιοθετήθηκε και στο φάκελο αυτό.)*

• Επίσης περιλαμβάνουν σημαντικό μέρος για τη *Mathematica* και τα εξής δύο βιβλία:

1. Βλάχος, Λ. (2008), *Διαφορικός Λογισμός Πολλών Μεταβλητών με Σύντομη Εισαγωγή στο Mathematica*. Εκδόσεις Τζιόλα, Θεσσαλονίκη.
2. Παπαγεωργίου, Γ. Σ., Τσίτουρας, Χ. Γ., Φαμέλης, Ι. Θ. (2004), *Σύγχρονο Μαθηματικό Λογισμικό: MATLAB – MATHEMATICA: Εισαγωγή και Εφαρμογές*. Εκδόσεις Συμεών, Αθήνα.

• Στα Αγγλικά υπάρχει ένας τεράστιος όγκος βιβλίων για τη *Mathematica*. Την πρώτη θέση την κατέχει προφανώς το σχετικό εκτενέστατο εγχειρίδιο της *Mathematica* από τον ίδιο το δημιουργό της *Mathematica* Stephen Wolfram και επίσης το συνοδό βιβλίο για τα πακέτα της *Mathematica*:

1. Wolfram, S. (1999), *The Mathematica Book*, 4η Έκδοση: *Mathematica* Version 4. Wolfram Media, Champaign, Illinois, and Cambridge University Press, Cambridge. *(Σημειώνεται πως αυτή είναι η έκδοση (η version) της* Mathematica *που χρησιμοποιείται στο φάκελο αυτό.)*
2. Wolfram Research (1999), *Mathematica 4 Standard Add-on Packages*. Wolfram Media, Champaign, Illinois.

• Ανάμεσα στα εκατοντάδες ίσως επιπλέον βιβλία για τη *Mathematica* εδώ περιοριζόμαστε να αναφέρουμε ένα κλασικό βιβλίο για τις Διαφορικές Εξισώσεις με τη *Mathematica*, ένα πρόσφατο βιβλίο για τις Μιγαδικές Συναρτήσεις πάλι με τη *Mathematica* και τέλος ένα βιβλίο για τις animations (τα κινούμενα σχήματα) που μπορούν κι αυτές να δημιουργηθούν εύκολα με τη *Mathematica*:

1. Abell, M. L., Braselton, J. P. (2004), *Differential Equations with Mathematica*, 3η Έκδοση. Academic Press, San Diego, California.
2. Shaw, W. T. (2006), *Complex Analysis with Mathematica*. Cambridge University Press, Cambridge.
3. Franke, H. W. (2002), *Animation mit Mathematica*. Springer, Berlin.

- Ο φοιτητής και η φοιτήτρια Πολιτικός Μηχανικός που ενδιαφέρονται παραπέρα για τη *Mathematica* και τις εφαρμογές της μπορούν να βρουν πολύ περισσότερα βιβλία σε εξειδικευμένα τεχνικά βιβλιοπωλεία και στο διαδίκτυο (π.χ. στη γνωστή ηλεκτρονική διεύθυνση http://www.amazon.com). Υπάρχουν επίσης και πάρα πολλές ιστοσελίδες με πληροφορίες και notebooks για τη *Mathematica*. Η πιο βασική σχετική ιστοσελίδα είναι φυσικά η ιστοσελίδα http://www.wolfram.com της ίδιας της εταιρείας Wolfram Research που διαθέτει τη *Mathematica*.

ΕΥΡΕΤΗΡΙΟ ΕΠΙΛΟΓΩΝ ΤΗΣ MATHEMATICA

• ***ΤΟ ΕΥΡΕΤΗΡΙΟ ΑΥΤΟ:*** Στο δισέλιδο αυτό ευρετήριο αναφέρουμε μερικές (τις πιο χρήσιμες 30) επιλογές που είναι προσιτές από τα menus της *Mathematica*, πώς καλείται η καθεμιά τους από το πληκτρολόγιο (χωρίς τη χρήση των menus), όταν αυτό είναι εφικτό, και τί ακριβώς κάνει. Φυσικά υπάρχουν πάρα πολλές ακόμη ενδιαφέρουσες επιλογές!

1. Από το menu File

- File → New [ή Ctrl N]: Δημιουργεί ένα εντελώς καινούργιο notebook
- File → Open ... [ή Ctrl o]: Ανοίγει ένα notebook
- File → Save ... [ή Ctrl s]: Αποθηκεύει το notebook στην προκαθορισμένη θέση του
- File → Save As ... [ή Shift Ctrl s]: Αποθηκεύει το notebook στη θέση που καθορίζουμε
- File → Palettes: Οδηγεί στις παλέττες για την επιλογή ενός συμβόλου ή μιας εντολής
- File → Notebooks: Οδηγεί στα πιο πρόσφατα notebooks, ώστε να καλέσουμε ένα
- File → Print ... [ή Ctrl p]: Τυπώνει το notebook στον εκτυπωτή
- File → Print Selection ... [ή Shift Ctrl p]: Τυπώνει το μέρος του notebook που μαυρίσαμε

2. Από το menu Edit

- Edit → Undo [ή Ctrl z]: Αναιρεί την προηγούμενη εντολή στο notebook
- Edit → Cut [ή Ctrl x]: Απαλείφει (κόβει) το τμήμα του notebook που έχουμε μαρκάρει
- Edit → Copy [ή Ctrl c]: Μεταφέρει το μαρκαρισμένο τμήμα του notebook στον clipboard
- Edit → Paste [ή Ctrl v]: Αντιγράφει ό,τι βρίσκεται στον clipboard πάνω στο notebook
- Edit → Select All [ή Ctrl a]: Επιλέγει (μαρκάρει, μαυρίζει) ολόκληρο το notebook
- Edit → Expression Input: Επιτρέπει τον καθορισμό εισόδου σαν εκθέτη, δείκτη, ρίζα, κλπ.
- Edit → Preferences: Οδηγεί σε μια τεράστια συλλογή όλων των δυνατοτήτων εμφανίσεως του notebook για επιλογή, έτσι ώστε να επιτευχθεί η βέλτιστη εμφάνισή του

3. Από το menu Cell

- Cell → Animate Selected Graphics [ή Ctrl Y]: Δημιουργεί animation (κινούμενο σχήμα)

4. Από το menu Format

- Format → Style → Title [ή Alt 1]: Γραφή σε μορφή αρχικού τίτλου
- Format → Style → Section [ή Alt 4]: Γραφή σε μορφή τίτλου ενότητας
- Format → Style → Text [ή Alt 7]: Γραφή σε μορφή παραγράφου κειμένου
- Format → Style → Input [ή Alt 9]: Γραφή σε μορφή εισόδου εντολής, όταν χρειάζεται
- Format → Font: Επιλογή γραμματοσειράς
- Format → Face: Επιλογή τρόπου γραφής: όρθια, παχιά, πλάγια, υπογράμμιση

- Format → Size: Επιλογή μεγέθους χαρακτήρων
- Format → Text Justification: Επιλογή τρόπου στοιχίσεως των παραγράφων
- Format → Magnification: Επιλογή ποσοστού μεγενθύνσεως του notebook στην οθόνη

5. Από το menu Kernel

- Kernel → Evaluation → Evaluate Notebook: Εκτέλεση όλων των εντολών του notebook
- Kernel → Abort Evaluation [ή Alt .]: Οριστική διακοπή της εκτελέσεως μιας εντολής
- Kernel → Quit Kernel → Local → Quit: Πλήρης διακοπή υπολογισμών με τη *Mathematica*

6. Από το menu Find

- Find → Find ... [ή Ctrl F]: Οδηγεί σε αναζήτηση (και αν χρειάζεται) και σε αντικατάσταση

7. Από το menu Help

- Help → Help Browser [ή Shift F1]: Καλεί τη βοήθεια της *Mathematica* για τις εντολές της

ΕΥΡΕΤΗΡΙΟ ΕΝΤΟΛΩΝ ΤΗΣ MATHEMATICA

• ***ΟΔΗΓΙΕΣ ΧΡΗΣΕΩΣ:*** Το πεντασέλιδο αυτό ευρετήριο αφορά σε όλες (224) τις εντολές της *Mathematica* που έχουν ήδη αναφερθεί κατ' αλφαβητική σειρά. Πρώτα αναφέρεται η ίδια η εντολή. Ακολουθεί η σελίδα όπου αυτή βρίσκεται (π.χ. 113 για την εντολή DSolve), άνω και κάτω τελεία (:) και ο αριθμός της εντολής (π.χ. D1) στο σχετικό notebook. *(Συνήθως μόνο η σελίδα αρκεί!)* Τέλος αναφέρεται πολύ περιληπτικά τί κάνει η καθεμία εντολή. Για τη σύνταξη, περισσότερες πληροφορίες και παραδείγματα ο χρήστης και η χρήστρια της *Mathematica* Πολιτικός Μηχανικός πρέπει να ανατρέξουν στο ίδιο το notebook όπου παραπέμπει (με τον τρόπο που εξηγήθηκε) αυτό το *Ευρετήριο Εντολών της Mathematica*.