Το πρόγραμμα από δύο πολυώνυμα της μορφής c0+c1x+... +ck*x^a1*y^a2 παράγει το μητρώο Sylvester και το πολυώνυμο Sylvester που προκύπτουν.

Τυπώνει στο stdout το σύστημα εξισώσεων. Δίαφορες πληροφορίες του συστήματος που θα λυθεί. Όπως τον πίνακα του μεγιστοβάθμιου όρου του πολυωνύμου Sylvester. Το κ που υπολογίζεται και το πρόβλημα που ανάγεται το σύστημα. Και τέλος τις λύσεις (ρίζες) του συστήματος καθώς και το αποτέλεσμα τις αντικατάστασης τους στα πολυώνυμα. (σε περιπτώσεις ιδιοτιμής με πολλαπλότητα απλά εκτυπώνεται η ιδιοτιμή και η πολλαπλότητα της, επίσης έαν υπολογισθεί ρίζα η οποία η αντικατάσταση της δεν θεώρειται αρκετά κοντά στο 0 και για τα δύο πολυώνυμα βγαίνει μήνυμα wrong solution).

/*Μετά τον υπολογισμό και την εκτύπωση των ριζών ρωτάται ο χρήστης εάν θέλει να προχωρίσει σε λύση ξανά μετά από αλλαγή μεταβλητής του πολυωνύμου sylvester.
*/

Αρχεία κώδικα .cpp :

1)alpha_main.cpp

2)poluonumo.cpp

3)mitroo_sylvester.cpp

4)poluonumo_sylvester.cpp

5)helping_functions.cpp

6)standard_eigen_problem.cpp

7) gen_eig_prob.cpp

8) roots.cpp

9)solve.cpp

10)change_poluonumo_sylvester.cpp

11)one_var_companion.cpp

12)interpolation.cpp

Αρχεία επικεφαλίδων .h :

1)poluonumo.h

2)mitroo_sylvester.h

3)poluonumo_sylvester.h

4)helping_functions.h

5)standard_eigen_problem.h

6) gen_eig_prob.h

7)roots.h

8)solve.h

9)change_poluonumo_sylvester.h

10)one_var_companion.h

11)interpolation.h

Αρχείο Makefile για μεταγλώτηση.

Περίεχονται στον φάκελο texts διάφορα παραδειγματικά αρχεία εισαγωγής πολυωνύμων.

Και δύο φάκελοι οι οποιοί είναι της βιβλιοθήκης Eigen. (ο Eigen και ο eigen-eigen-b30b87236a1b)

Περιγραφή αρχείων πηγαίου Κώδικα

1) main.cpp : Είναι η διεπαφή του χρήστη με το πρόγραμμα. Ο χρήστης έχει τρεις επιλογές για είσοδο των πολυωνύμων. Μπορεί να δημιουργήσει δυο τυχαία πολυώνυμα βαθμών δ1 και δ2,μπορεί να δώσει 2 πολυώνυμα ως είσοδο από αρχείο ή τέλος να δώσει τα πολυώνυμα από την πρότυπη είσοδο. Παρακάτω θα περιγραφεί και ο τρόπος πιο αναλυτικά.

2) poluonumo.cpp : Περιέχει όλες τις συναρτήσεις που αφορούν την κλάση πολυώνυμο καθώς και την κλάση (term) που είναι τα μονώνυμα του πολυωνύμου και έχουν άμεση σχέση με αυτό.

3) mitroo_sylvester.cpp : Περιέχει όλες τις συναρτήσεις που

αφορούν την κλάση mitroo_sylvester που αναπαριστά το Μητρώο Σιλβέστερ.

4) poluonumo_sylvester.cpp : Περιέχει όλες τις συναρτήσεις που αφορούν την κλάση poluonumo_sylvester που αναπαριστά το πολυώνυμο Σιλβέστερ.

5) helping_functions.cpp : Περιέχει τις συναρτήσεις οι οποίες χρησιμοποιήθηκαν για το parsing των πολυωνύμων από την είσοδο του χρήστη και την συνάρτηση δημιουργίας τυχαίων πολυωνύμων.

6) standard_eigen_problem.cpp:Περιέχει όλες τις συναρτήσεις που αφορούν την κλάση η οποία δημιουργήθηκε για την κατασκευή και την λύση του κλασικού προβλήματος.

7) gen_eig_prob.cpp:Περιέχει όλες τις συναρτήσεις που αφορούν την κλάση η οποία δημιουργήθηκε για την κατασκεύη και την λύση του γενικευμένου προβλήματος.

8) roots.cpp:Περιέχει τις συναρτήσεις που αφορούν την κλάση roots η οποία κρατάει τις λύσεις που τις δίνονται.Χρησιμοποιεί και μια κλάση riza η οποία αναπαρίστα μια πραγματική ρίζα των πολυωνύμων και έχει άμεση σχέση με την κλάση roots.

9) solve.cpp:Είναι η διεπαφή της alpha_main για την λύση του κατάλληλου προβλήματος. Καθώς και τον υπολογισμό του κ και τον υπολογισμό της ορίζουσας.

10) change_poluonumo_sylvester.cpp: περιέχει όλες τις συναρτήσεις οι οποίες χρησιμοποιούνται για την υλοποιήση της αλλαγή μεταβλητής στο πολυώνυμο sylvester. Σημειώνεται ότι έχει υλοποιηθεί συνάρτηση η οποία μετατρέπει μόνο τον max σταθερό όρο του πολυωνύμου, υπολογίζοντας το κ μετά. Έτσι ώστε να συγκρίνει το παλιό κ με το νέο κ και αν το νέο κ είναι μικρότερο τότε προχωράει το πρόγραμμα στο να καλέσει την συνάρτηση που αφορά την αλλαγή μεταβλητής ολόκληρου του πολυωνύμου Sylvester.

11) one_var_companion.cpp: περιέχει όλες τις συναρτήσεις οι οποίες χρησιμοποιούνται για την υλοποιήση της εύρεσης και της δεύτερης συντεταγμένης της ρίζας όταν υπάρχει πολλαπλότητα. Η κλάση αυτή που έχει υλοποιηθεί χρησιμοποιείται μέσα σε συναρτήσεις της roots.cpp στην περίπτωση πολλαπλότητας.

12) Interpolation.cpp: περιέχει όλες τις συναρτήσεις της κλάσης interpolation που υλοποιεί τον αντίστοιχο αλγόριθμο και κράταει τις απαραίτητες πληροφορίες.

Οδηγίες Μεταγλώττισης του Προγράμματος

Περιέχεται αρχείο Makefile για εύκολη μεταγλώττιση του προγράμματος πληκτρολογώντας την εντολή make στο terminal παράγεται το εκτελέσιμο αρχείο run.

Οδηγίες Χρήσης του Προγράμματος

Έχουν γίνει οι παραδοχές ότι το πολυώνυμο που δίνεται ως είσοδος δεν θα περιέχει κενά και αν υπάρχουν και οι δύο μεταβλητές σε κάποιο μονώνυμο η μεταβλητή x προηγείται της y. Η σειρά με την οποία δίνονται τα μονώνυμα δεν παίζει κανέναν ρόλο.

(παράδειγμα εισόδου: 2*x^4*y^3-4*x^2*y^2+3*x^2*y+2*x^2+2*x*y^3-y+5)

Πλέον το πρόγραμμα μετατράπηκε και μπορουν τα πολυώνυμα καθώς και όλες οι υπόλοιπες κλάσεις που τα χρησιμοποιούνε να δέχονται double όρους.

Running the program:

Ο χρήστης επιλέγει έναν από τους παραπάνω τρόπους που περιγράφηκαν στο main.cpp από την γραμμή εντολών με τις παρακάτω εντολές:

1) ./run -generate -d1 # -d2 # -solve #

για δημιουργία δύο τυχαίων πολυωνύμων βαθμού d1 και d2, όπου # είναι ο βαθμός του πολυωνύμου που θα δημιουργηθεί. Εάν δεν δωθεί αριθμός μετα το solve τότε θεωρείται το 7.

2) ./run -read -console

για να δώσει ο χρήστης τα πολυώνυμα από την πρότυπη είσοδο.

3) ./run -read -i [filepath] -d1 # -d2 # -solve #

για να δώσει ο χρήστης μέσω αρχείου 2 πολυώνυμα βαθμού d1 και d2, όπου # είναι ο βαθμός του πολυωνύμου και όπου [filepath] το path για το αρχείο.

το αρχείο θα πρέπει να περιέχει 2 γραμμές, μία γραμμή για κάθε πολυώνυμο.Εάν δεν δωθεί αριθμός μετα το solve τότε θεωρείται το 7.

Μόλις υπολογιστούν και εκτυπωθούν οι λύσεις το πρόγραμμα ρωτάει τον χρήστη εάν θέλει να προχωρίσει στην διαδικασία αλλαγής μεταβλητης του πολυωνύμου Sylvester.