Επιστημονικός Υπολογισμός Ι
2η Εργαστηριακή Άσκηση

Ημερομηνία επιστροφής για πλήρη βαθμό: **11/12/2013** Ώρα: **23:59**
Δεν θα δοθεί καμία παράταση!!

Προσοχή: *Μπορείτε να συζητήσετε την άσκηση με συναδέλφους σας αλλά αν διαπιστωθεί αντιγραφή, θα* ***μηδενιστεί*** *ο βαθμός σας. Δείτε και τις οδηγίες που αναφέρονται στους κανόνες βαθμολογίας!*

Η παρούσα άσκηση αναφέρεται στο θέμα της αριθμητικής και των σφαλμάτων που γίνονται στον υπολογιστή. Στο μεγαλύτερο μέρος της άσκησης θα ασχοληθείτε με προσεγγίσεις και υπολογισμούς πολυωνύμων. Τα πολυώνυμα (όπως είδατε και στην πρώτη άσκηση) αποτελούν βασικό αλγεβρικό εργαλείο μοντελοποίησης πολύπλοκων ή άγνωστων συναρτήσεων. Στα ερωτήματα που ακολουθούν θα εξετάσετε θέματα τα οποία προκύπτουν κατά τη διαχείριση πολυωνύμων και τα οποία οφείλονται κατά βάση στην αριθμητική πεπερασμένης ακρίβειας καθώς και στη συμπεριφορά ισοδύναμων μαθηματικών αλλά υπολογιστικά διαφορετικών αναπαραστάσεων τους. Σκοπός της άσκησης είναι η εξοικείωση και η καλύτερη κατανόηση εννοιών που αφορούν το Αριθμητικό Μοντέλο.

Μην ξεχάσετε:
Σε μία παράγραφο να αναφέρετε όπως και στην πρώτη άσκηση τα χαρακτηριστικά του υπολογιστικού σας συστήματος.

Ερώτημα 1 - Χαρακτηριστικά Αριθμητικής στο MATLAB

Σκοπός του ερωτήματος είναι η βαθύτερη κατανόηση της αριθμητικής πεπερασμένης ακρίβειας την οποία χρησιμοποιεί το MATLAB.

(i) Να καταγράψετε σε έναν πίνακα το αποτέλεσμα της εκτέλεσης των εντολών του MATLAB `eps`, `realmax` και `realmin` για μονή (`single`) και διπλή (`double`) ακρίβεια. Επιπλέον, στον ίδιο πίνακα να συμπεριλάβετε και μία περιγραφή του τι αντιπροσωπεύουν τα αποτελέσματα που πήρατε.

(ii) Να αιτιολογήσετε το αποτέλεσμα του παρακάτω κώδικα MATLAB:

```
if (eps/2 ==0),
    disp(1)
end

if (1+eps > 1),
    disp(2)
end

if (1+realmin == 1),
    disp(3)
end

if (eps + realmax == realmax),
    disp(4)
end
```

Μία πλήρης αιτιολόγηση θα πρέπει να περιλαμβάνει σαφείς ενδείξεις για ποιο λόγο ικανοποιείται ή όχι κάθε μία από τις παραπάνω δομές επιλογής. Για αυτό χρήσιμο θα ήταν να υπολογίσετε ξεχωριστά τις τιμές των μελών κάθε συνθήκης. Για την αιτιολόγηση μπορείτε να χρησιμοποιήσετε τόσο τη δεκαεξαδική μορφή των αριθμών όσο και το εργαλείο bitgui[1].

[1] http://www.mathworks.com/matlabcentral/fileexchange/33874-bitgui-a-graphical-explorer-of-the-ieee-754-floating-point-formats

Ερώτημα 2 - Διερεύνηση Σύγλισης Σειρών

Στο δεύτερο ερώτημα θα ασχοληθείτε με απειροσειρές και το πως αυτές φαίνεται να συγκλίνουν ενώ θεωρητικά αποκλίνουν. Στη συνέχεια θα θεωρήσετε τη σειρά:

$$\sum_{n=1}^{\infty} \frac{1}{n^k}$$

Η σειρά αυτή σύμφωνα με τη θεωρία για θετικούς ακεραίους k συγκλίνει όταν $k > 1$ ενώ αποκλίνει όταν $k = 1$. Στην ειδική περίπτωση που $k = 1$ η σειρά ονομάζεται αρμονική[2].

(i) Να υλοποιήσετε συνάρτηση MATLAB η οποία υπολογίζει τα μερικά αθροίσματα της παραπάνω σειράς (για $k = 1$) και τερματίζει όταν ικανοποιείται ένα κριτήριο σύγκλισης. Το κριτήριο λαμβάνεται ως η διαφορά της νέας υπολογισμένης τιμής από την αμέσως προηγούμενη της να είναι 0.

(ii) Να ~~τρέξετε το πρόγραμμα και να εξηγήσετε αν προβλέπετε ότι~~ προβλέψετε αν το πρόγραμμα θα τερματίσει ~~ή όχι;~~. Ο Μαθηματικός προβλέπει ότι όχι. Εμείς γνωρίζουμε ότι θα τερματίσει. Να υπολογίσετε ~~πόσες~~ ένα φράγμα για το πόσες επαναλήψεις θα χρειαστούν.

(iii) Αν τρέξετε τον κώδικα (όπως θα έκανε κάποιος πρακτικός μηχανικός που μάλλον δεν έχει παρακολουθήσει ΕΥ) τι παρατηρείτε; Βλέπετε να τερματίζει; Αν όχι, πως το εξηγείτε; Υπολογίστε πόσο περίπου χρόνο θα πάρει για να τερματίσει η συνάρτησή σας.

(iv) Να επαναλάβετε τα παραπάνω και να απαντήσετε στις ίδιες ερωτήσεις χρησιμοποιώντας αποκλειστικά μονή ακρίβεια (single precision) όπως αυτή υλοποιείται στο MATLAB.

Ερώτημα 3 - Πράξεις με Πολυώνυμα

Το ερώτημα αυτό δομείται σε δύο μέρη, σε καθένα από τα οποία θα διερευνήσετε την επίδραση της αριθμητικής πεπερασμένης ακρίβειας πάνω σε πράξεις με πολυώνυμα. Αρχικά όμως θα πρέπει να μελετήσετε ορισμένες συναρτήσεις του MATLAB οι οποίες είναι σχεδιασμένες για τη διαχείριση πολυωνύμων.

Σε μία παράγραφο να εξηγήσετε για ποιο λόγο χρησιμοποιούνται οι συναρτήσεις: `poly`, `roots`,`polyval`. Εκτελώντας την εντολή `type` ή `open` θα παρατηρήσετε ότι μπορείτε να έχετε πρόσβαση στους κώδικες των παραπάνω συναρτήσεων. Να μελετήσετε τους κώδικες και να εξηγήσετε πώς ακριβώς λειτουργούν οι συναρτήσεις αυτές.

Στα επόμενα δύο μέρη θα ασχοληθείτε με πολυώνυμα της μορφής:

$$p(x) = \prod_{i=0}^{n} (x - i)$$

Τα πολυώνυμα αυτά φαίνονται πολύ απλά, αλλά (όπως και πολλά άλλα πολυώνυμα) είναι "δόλια", κατά τον πετυχημένο χαρακτηρισμό του διάσημου ερευνητή James Wilkinson, γιατί η διαχείρισή τους κρύβει παγίδες.

Μέρος Α

Στο πρώτο μέρος του ερωτήματος θα ασχοληθείτε με την επίδραση που έχουν μικρές διαταραχές των συντελεστών ενός πολυωνύμου στην εύρεση των ριζών του. Θα θεωρήσετε τρία είδη διαταραχών. Στο **πρώτο** είδος υπάρχει μόνο ένας συντελεστής που διαταράσσεται. Στο **δεύτερο** είδος διαταράσσονται όλοι οι συντελεστές βάσει κάποιας σταθεράς, ενώ στο **τρίτο** είδος διαταράσσονται όλοι οι συντελεστές βάσει κάποιας σχετικής διαταραχής η οποία θα εξαρτάται από τους ίδιους.

[2] http://en.wikipedia.org/wiki/Harmonic_series_(mathematics)

Έστω το πολυώνυμο p_{20} με ρίζες τις τιμές $z = [1 : 20]$ και συντελεστές (τους οποίους θα πρέπει να υπολογίσετε μέσω της κατάλληλης συνάρτησης MATLAB) $\{\alpha_{20}, \alpha_{19}, \ldots, \alpha_1, \alpha_0\}$. Έστω επίσης ένα δεύτερο πολυώνυμο g_{20} το οποίο έχει συντελεστές $\{\beta_{20}, \beta_{19}, \ldots, \beta_1, \beta_0\}$, καθένας από τους οποίους θα υπολογίζεται σύμφωνα με ερωτήματα που ακολουθούν. Το πολυώνυμο g_{20} θα αποτελεί το πολυώνυμο εκείνο το οποίο θα επιτρέπει την εισαγωγή της ζητούμενη διαταραχή (έστω ϵ) στους αντίστοιχους συντελεστές του p_{20}. Θα πρέπει δηλαδή κάθε φορά να κατασκευάζετε κατάλληλα διανύσματα συντελεστών του g_{20} έτσι ώστε οι συντελεστές του πολυωνύμου $f_{20} = p_{20} + \epsilon g_{20}$ να είναι οι διαταραγμένοι ή όχι συντελεστές του πολυωνύμου p_{20}.

(Α΄) Να υλοποιήσετε script MATLAB το οποίο:

- (i) Θα υπολογίζει τους συντελεστές του πολυωνύμου p_{20}.
- (ii) Θα υπολογίζει πέντε διαφορετικά σύνολα συντελεστών του πολυωνύμου g_{20} (ένα για κάθε είδος διαταραχής) ως εξής (**Προσοχή:** Ο συντελεστής α_{20} δεν πρέπει να επηρεάζεται σε καμία περίπτωση και θα πρέπει να παραμένει πάντα 1):
  - (α΄) Τα πρώτα τρία σύνολα διαταραχών θα αφορούν μόνο ένα συντελεστή του πολυωνύμου p_{20}. Οι διαταραχές $\cancel{\epsilon^{(20)}}$ $\epsilon^{(19)} = 10^{-8}$, $\epsilon^{(11)} = -10^3$, $\epsilon^{(1)} = 10^{10}$ θα πρέπει να εφαρμόζονται στους συντελεστές $\cancel{\alpha_{20}}$ $\alpha_{19}, \alpha_{11}, \alpha_1$ αντίστοιχα.
  - (β΄) Στο ~~τρίτο~~ τέταρτο σύνολο διαταραχών θα πρέπει να διαταράξετε όλους τους συντελεστές α_i με τη διαταραχή $\epsilon = 10^{-8}$.
  - (γ΄) Στο τελευταίο σύνολο διαταραχών θα πρέπει να διαταράξετε όλους τους συντελεστές α_i με μία σχετική διαταραχή της τάξης του $\epsilon^{(i)} = 2^{-52}\alpha_i$.
- (iii) Για κάθε σύνολο διαταραχών:
  - (α΄) Θα υπολογίζει το πολυώνυμο f_{20}.
  - (β΄) Θα υπολογίζει τις ρίζες του πολυωνύμου f_{20}.
  - (γ΄) Θα ιεραρχεί τις ρίζες του πολυωνύμου f_{20} . Για την ιεραρχία θα πρέπει να λάβουμε υπόψιν τις εξής δύο περιπτώσεις: *(α΄)* Οι ρίζες να είναι πραγματικές. Στην περίπτωση αυτή οι ρίζες θα ιεραρχούνται βάσει του σχετικού σφάλματος κάθε ρίζας με την πλησιέστερη σε αυτή γνωστή ρίζα. Για παράδειγμα αν η υπολογισμένη ρίζα είναι 1.01 τότε η πιο κοντινή γνωστή ρίζα θα είναι η 1.0 και το σχετικό σφάλμα θα αντιστοιχεί στο $\frac{|1.01-1.0|}{|1.0|} = 10^{-2}$. *(β΄)* Οι ρίζες να είναι πραγματικές και μιγαδικές. Στην περίπτωση αυτή θα κάνετε ένα sorting στις ρίζες (προσέχτε ότι η συνάρτηση του MATLAB `sort` για μιγαδικά δεδομένα χρησιμοποιεί τη συνάρτηση `abs`-δηλαδή κάνει sorting βάσει του μέτρου) και θα αντιστοιχίσετε σε αυτές τις γνωστές ρίζες υπολογίζοντας στη συνέχεια το σχετικό σφάλμα. Δηλαδή στην περίπτωση αυτή θεωρείτε ότι την πιο κοντινή γνωστή ρίζα σας τη δίνει ένα sorting των υπολογιζόμενων ριζών. Η ιεραρχία θα είναι φθίνουσα ως προς το σχετικό σφάλμα.
  - (δ΄) Θα οπτικοποιεί με μία γραφική παράσταση στο μιγαδικό επίπεδο τις υπολογιζόμενες και τις γνωστές ρίζες του πολυωνύμου.

(Β΄) Να συγκρίνετε τις γραφικές παραστάσεις.

Μέρος Β

Στο δεύτερο μέρος θα διερευνήσετε την πίσω ευστάθεια του αλγορίθμου Horner και θα την χρησιμοποιήσετε για να φράξετε το εμπρός σφάλμα. Το πολυώνυμο με το οποίο θα πειραματιστείτε θα είναι το:

$$p(x) = \prod_{j=1}^{10} (x - j)$$

για το οποίο θα υπολογίσετε τις τιμές του (μέσω κατάλληλης εντολής MATLAB) στα σημεία:

$$x = [1 + \texttt{eps}, 1 + 10^{-8}, 1.5, 5.5, 9 + 10^{-8}, 9 + \texttt{eps}(9)]$$

Στη συνέχεια για το διάνυσμα x θα κατασκευάσετε πίνακα κάθε γραμμή του οποίου θα περιέχει τα στοιχεία:

ξ_i	$p(\xi_i)$	$p(\xi_i)$ από **μορφή γινομένου**	σχετικό σφάλμα	δείκτης κατάστασης προβλήματος	δείκτης κατάστασης προβλήματος $\times$ πίσω σφάλμα

Η τιμή ξ_i της ~~δεύτερης~~ τρίτης στήλης να υπολογιστεί από την παραπάνω μορφή γινομένου του πολυωνύμου. Θα θεωρήσουμε την τιμή αυτή ως ακριβή.

Για το δείκτη κατάστασης της τιμής του πολυωνύμου σε **δυναμομορφή** στο ξ (το οποίο δεν αποτελεί ρίζα) καθώς και για το πίσω σφάλμα της μεθόδου Horner θα πρέπει να ανατρέξετε στη θεωρία που έγινε στην τάξη.

Τρόπος Παράδοσης Εργασίας

Παραδοτέα: Αναφορά (σε μορφή pdf) και κώδικας της άσκησης συμπιεσμένα σε αρχείο zip με ονομασία **AM_prb2_2014** π.χ. **3948_prb2_2014**.

Παράδοση & Αποστολή: Το συμπιεσμένο αρχείο παραδίδεται μέσω της πλατφόρμας e-class ενώ υποχρεούστε να παραδώσετε και εκτυπωμένη αναφορά, **για την παράδοση της οποία θα υπάρξει ειδική ανακοίνωση προς το τέλος του εξαμήνου.**

Ιδιαίτερη έμφαση θα δοθεί στον τρόπο παρουσίασης της εργασίας και των αποτελεσμάτων. Για τη συγγραφή της αναφοράς μπορείτε εκτός των γνωστών εργαλείων να πειραματιστείτε και με άλλα όπως το LaTeX ή το εργαλείο του MATLAB `publish`.