

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ, ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΚΑΙ ΔΙΚΤΥΩΝ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΗΥ200: ΕΠΙΣΤΗΜΟΝΙΚΟΣ ΥΠΟΛΟΓΙΣΜΟΣ

ΕΡΓΑΣΙΑ 1: Αριθμητική Γραμμική Άλγεβρα.

(Ημερομηνία Παράδοσης: Κυριακή 10 Απριλίου 2005, (Ώρα: 23:55))

(Η παράδοση της εργασίας θα γίνει μέσα από την σελίδα του μαθήματος στο eClass.)

ΑΣΚΗΣΗ 1.

ΑΣΚΗΣΗ 2.

n_x	n_y	c_1	c_2	c	χρόνος	σφάλμα
3	10	0	0	0	0.1250	$8.0005e - 015$
10	3	0	0	0	0.0470	$1.6852e - 014$
5	5	0	0	0	0.0310	$3.7321e - 015$
5	5	100	0	0	0.0310	$2.6899e - 015$
5	5	0	100	0	0.0310	$3.2501e - 015$
5	5	0	0	100	0.0150	$6.3273e - 015$
10	10	0	0	0	0.2500	$2.6232e - 014$
20	20	0	0	0	3.9850	$2.5990e - 013$
30	30	0	0	0	21.0940	$1.0038e - 012$
40	40	0	0	0	68.4530	$2.6837e - 012$
50	50	0	0	0	180.4220	$7.0978e - 012$
10	10	0	0	100	0.3750	$4.9728e - 014$
20	20	0	0	100	4.9370	$2.5738e - 013$
30	30	0	0	100	21.5620	$1.0141e - 012$
40	40	0	0	100	74.3440	$2.8267e - 012$
50	50	0	0	100	197.2970	$7.4974e - 012$

Πίνακας 1. Αποτελέσματα με χρήση της `lu`.

Στην πρώτη άσκηση και συγκεκριμένα στα (α),(β) παρατηρούμε ότι ενώ το μέγεθος του πίνακα παραμένει ίδιο με $c_1=c_2=c=0$, όταν το $n_x < n_y$ τότε ο χρόνος εκτέλεσης της `lu_run` είναι μεγαλύτερος και το σφάλμα μικρότερο απ'οτι όταν έχω $n_x > n_y$. ο χρόνος εκτέλεσης. Στα ερωτήματα (γ),(δ),(ε),(στ) έχω και εδώ ίδιο μέγεθος πίνακα με $n_x = n_y$ αλλά μεταβάλλοντας τα c_1, c_2, c παρατηρούμε ότι παραμένει ίδιος όταν $c=0$ ανεξαρτήτως των c_1, c_2 ενώ το σφάλμα διαφοροποιείται. Στην περίπτωση όπου το c μεταβάλλεται παρατηρούμε ο χρόνος εκτέλεσης να μειώνεται δραματικά ενώ το σφάλμα να αυξάνεται σε σχέση με τις προηγούμενες περιπτώσεις.

Στην δεύτερη άσκηση παρατηρούμε ότι όσο το μέγεθος του πίνακα αυξάνεται,έχοντας $c_1=c_2=c=0$, ο χρόνος εκτέλεσης της `lu_run` αυξάνεται δραματικά καθώς και το σφάλμα γίνεται ολοένα και μεγαλύτερο.Το ίδιο παρατηρείται και όταν $c_1=c_2=0$ αλλά $c=100$ αυξανόμενου πάντα του μεγέθους του πίνακα,με τη διαφορά ότι υπάρχει μία μικρή απόκλιση των μετρήσεων μας προς τα επάνω.

ΑΣΚΗΣΗ 3.

ΑΣΚΗΣΗ 4.

ΑΣΚΗΣΗ 5.

ΑΣΚΗΣΗ 6.

	$n_x = n_y$	c	χρόνος	επαναλήψεις	σφάλμα	c	χρόνος	επαναλήψεις	σφάλμα
<i>luinc</i>	10	0	0	—	$2.6472e - 014$	100	0	—	$4.6573e - 014$
	20	0	0.0160	—	$2.8339e - 013$	100	0.0160	—	$2.7947e - 013$
	30	0	0.0310	—	$1.0588e - 012$	100	0.0470	—	$1.1591e - 012$
	40	0	0.1250	—	$2.8510e - 012$	100	0.1250	—	$3.0609e - 012$
	50	0	0.2810	—	$7.4437e - 012$	100	0.2340	—	$7.6968e - 012$
<i>Jacobi</i>	10	0	0	100	$2.4734e - 004$	100	0	43	$4.9531e - 006$
	20	0	0.0470	400	$8.8342e - 005$	100	0.0160	139	$4.7231e - 006$
	30	0	0.2030	900	$5.1935e - 005$	100	0.0780	285	$4.9409e - 006$
	40	0	0.5780	1600	$3.6468e - 005$	100	0.2030	480	$4.9761e - 006$
	50	0	1.5470	2500	$2.8015e - 005$	100	0.5000	722	$4.9764e - 006$
<i>G - S</i>	10	0	0.0470	100	$8.4316e - 006$	100	0	25	$4.3281e - 006$
	20	0	0.1720	360	$4.9394e - 006$	100	0.0310	77	$4.3936e - 006$
	30	0	0.6090	746	$4.9818e - 006$	100	0.1250	156	$4.7805e - 006$
	40	0	1.9070	1257	$4.9941e - 006$	100	0.4060	262	$4.8521e - 006$
	50	0	4.4370	1888	$4.9846e - 006$	100	1	393	$4.9679e - 006$
<i>CG</i>	10	0	0.0150	14	$2.5440e - 007$	100	0.0320	12	$9.5757e - 007$
	20	0	0.0460	29	$2.8347e - 006$	100	0.0160	24	$2.7685e - 006$
	30	0	0.0470	44	$3.6850e - 006$	100	0.0310	36	$4.4713e - 006$
	40	0	0.0620	59	$3.9598e - 006$	100	0.0620	49	$3.7638e - 006$
	50	0	0.1090	75	$4.2605e - 006$	100	0.0930	62	$3.7857e - 006$

Πίνακας 2. Αποτελέσματα με χρήση λογισμικού για αραιούς πίνακες και επαναληπτικές μεθόδους.

Για την `luinc_run` ως προς το χρόνο εκτέλεσης παρατηρούμε ότι αυξάνοντας το μέγεθος του πίνακα αυξάνεται και ο χρόνος εκτέλεσής της.

Για την `jacobi_run` ως προς το χρόνο εκτέλεσης παρατηρούμε ότι αυξάνοντας το μέγεθος του πίνακα αυξάνεται και ο χρόνος εκτέλεσής της.Ως προς το πλήθος των επαναλήψεων διαπιστώνω ότι και αυτός αυξάνεται όσο μεγαλύτερος είναι ο πίνακας μου με μικρότερο όμως ρυθμό όσο το `c`» ενώ για την ακρίβεια των λύσεων μου σημαντικό ρόλο παίζει η παράμετρος `c(c » ==> σφάλμα »)`.

Για την `gs_run` ως προς το χρόνο εκτέλεσης παρατηρούμε ότι αυξάνοντας το μέγεθος του πίνακα αυξάνεται και ο χρόνος εκτέλεσής της.Για την ακρίβεια των λύσεων μου σημαντικό ρόλο παίζει η παράμετρος `c (c » ==> σφάλμα »)`ενώ ως προς το πλήθος των επαναλήψεων διαπιστώνω ότι και αυτός αυξάνεται όσο μεγαλύτερος είναι ο πίνακας μου,με μικρότερο όμως ρυθμό όσο το `c`».

Για την `cg_run` ως προς το πλήθος των επαναλήψεων διαπιστώνω ότι και αυτός αυξάνεται όσο μεγαλύτερος είναι ο πίνακας μου.Ως προς το χρόνο εκτέλεσης παρατηρούμε ότι αυξάνοντας το μέγεθος του πίνακα αυξάνεται και ο χρόνος εκτέλεσής της ενώ για την ακρίβεια των λύσεων μου σημαντικό ρόλο παίζει η παράμετρος `c(c » ==> σφάλμα »)`.

Γενικά πρέπει να επισημανθεί ότι όσο προχωρούμε απ'την `luinc_run` προς την `cg_run` τόσο η ακρίβεια της λύσης μας όσο και ο χρόνος εκτέλεσης γίνονται καλύτερα για τη μηχανή καθώς γίνεται καλύτερος και ο αλγόριθμος που χρησιμοποιούμε σε κάθε περίπτωση.Το ίδιο βέβαια ισχύει και για τον αριθμό των επαναλήψεων.