

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ, ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΚΑΙ ΔΙΚΤΥΩΝ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
HY200: ΕΠΙΣΤΗΜΟΝΙΚΟΣ ΥΠΟΛΟΓΙΣΜΟΣ

ΕΡΓΑΣΙΑ 1: Αριθμητική Γραμμική Άλγεβρα.

ΑΣΚΗΣΗ 1. Σ'αυτές τις 2 αρχικές περιπτώσεις έχουμε τα n_x, n_y τα οποία δεν πληρούν την προϋπόθεση $n_x, n_y \leq 5$ και δε φαίνεται το figure1. Η λύση παρουσιάζει πιο μικρά σφάλματα όταν έχουμε $n_x, n_y = 5$ και κυρίως για $c_1 = 100$ που το σφάλμα είναι $2.6899e-015$.

ΑΣΚΗΣΗ 2. Καθώς αυξάνονται τα n_x, n_y αυξάνεται και ο χρόνος εκτέλεσης. Όμως όταν έχουμε $c = 0$ τις περισσότερες φορές υπάρχει πιο μεγάλη καθυστέρηση από αυτές που έχουμε όταν $c = 100$. Το πιο μικρό σφάλμα έχουμε όταν $n_x, n_y = 10$ και $c_1, c_2, c = 0$. Ισχύει ότι τα σφάλματα είναι μικρότερα σε κάθε περίπτωση όταν $c = 0$ αντί για εκείνες τις φορές που έχουμε $c = 100$.

ΑΣΚΗΣΗ 3. Άυξηση των n_x, n_y σημαίνει αύξηση και του χρόνου εκτέλεσης. Όταν $c = 0$ συνήθως υπάρχει πιο μεγάλη καθυστέρηση από τις φορές που είναι $c = 100$. Το ελάχιστο σφάλμα υπάρχει όταν έχουμε $n_x, n_y = 10$ και $c_1, c_2, c = 0$. Τα σφάλματα είναι πιο μικρά όταν $c = 0$ αντί για όταν έχουμε $c = 100$.

ΑΣΚΗΣΗ 4. Για $c = 0$ οι επαναλήψεις είναι πιο πολλές απ'ότι για $c = 100$. Άυξηση των n_x, n_y σημαίνει αύξηση και του χρόνου εκτέλεσης. Όταν $c = 0$ συνήθως υπάρχει πιο μεγάλη καθυστέρηση από τις φορές που $c = 100$. Όταν $c = 0$ συνήθως υπάρχει πιο μεγάλη καθυστέρηση από όταν $c = 100$. Έλαχιστο σφάλμα για $n_x, n_y = 20$ και $c_1, c_2 = 0$ $c = 100$.

ΑΣΚΗΣΗ 5. Για $c = 100$ έχουμε λιγότερες επαναλήψεις από όταν $c = 0$. χρόνος είναι πιο μεγάλος όταν $c = 0$ και το σφάλμα είναι λίγο μεγαλύτερο για $c = 0$. Μικρότερο σφάλμα έχουμε για $n_x, n_y = 10$ και $c_1, c_2, c = 0$.

ΑΣΚΗΣΗ 6. Χρόνος και επαναλήψεις όταν $c = 0$ είναι λίγο μεγαλύτερα. Οι διαφορές μεταξύ των σφαλμάτων για $c = 0$ και $c = 100$ είναι μικρές. Πιο σφάλμα έχουμε για $c = 0$, ενώ πιο μικρό σφάλμα έχουμε για $n_x, n_y = 10$ και $c_1, c_2, c = 0$.

ΓΕΝΙΚΟ ΣΥΜΠΕΡΑΣΜΑ: Η cg είναι η καλύτερη επαναληπτική μέθοδος γιατί σ'αυτήν υπάρχουν πιο λίγες επαναλήψεις και μικρότερο χρόνο εκτέλεσης και σφάλμα.

n_x	n_y	c_1	c_2	c	χρόνος	σφάλμα
3	10	0	0	0	0	$8.000542e - 015$
10	3	0	0	0	0	$1.6852e - 014$
5	5	0	0	0	0	$3.7321e - 015$
5	5	100	0	0	0	$2.6899e - 015$
5	5	0	100	0	0	$3.2501e - 015$
5	5	0	0	100	0	$6.3273e - 015$
10	10	0	0	0	0.0310	$2.632e - 015$
20	20	0	0	0	0.0310	$2.5990e - 013$
30	30	0	0	0	0.3600	$1.0038e - 012$
40	40	0	0	0	1.8440	$2.6837e - 012$
50	50	0	0	0	8.53100	$7.0978e - 012$
10	10	0	0	100	0	$4.9728e - 014$
20	20	0	0	100	0.0940	$2.5738e - 013$
30	30	0	0	100	0.3590	$1.0141e - 012$
40	40	0	0	100	1.828	$2.8267e - 012$
50	50	0	0	100	7.484	$7.4974e - 012$

Πίνακας 1. Αποτελέσματα χρησιμοποιώντας την `lu`.

	$n_x = n_y$	c	χρόνος	επαναλήψεις	σφάλμα	c	χρόνος	επαναλήψεις	σφάλμα
<i>luinc</i>	10	0	0.0630	—	$2.6479e - 014$	100	0	—	$4.6573e - 014$
	20	0	0.0310	—	$2.8339e - 013$	100	0	—	$2.7947e - 013$
	30	0	0.0470	—	$1.0588e - 012$	100	0.0470	—	$1.1591e - 012$
	40	0	0.1100	—	$2.8510e - 012$	100	0.1100	—	$3.0609e - 012$
	50	0	0.2340	—	$7.4437e - 012$	100	0.2350	—	$7.6968e - 012$
<i>Jacobi</i>	10	0	0.0150	100	0.00025	100	0	43	$4.95311e - 006$
	20	0	0.4530	400	0.00009	100	0.188	139	$4.7231e - 006$
	30	0	5.9530	900	0.0005	100	1.7040	285	$4.9409e - 006$
	40	0	26.813	1600	0.0004	100	8.3440	480	$4.9761e - 006$
	50	0	112.047	2500	0.00003	100	38.42200	722	$4.9764e - 006$
<i>G - S</i>	10	0	0.3100	100	0.00199	100	0.17200	77	0.00384
	20	0	0.860	360	0.00433	100	0.2030	77	0.00384
	30	0	8.42200	746	0.00955	100	1.84400	156	0.00916
	40	0	41.375	1257	0.01676	100	8.81200	262	0.01629
	50	0	166.06200	1888	0.02590	100	33.65600	393	0.02582
<i>CG</i>	10	0	0.07800	14	$2.5440e - 007$	100	0.07800	12	$9.5757e - 006$
	20	0	0.09300	29	$2.8347e - 006$	100	0.07800	24	$2.7685e - 006$
	30	0	0.51600	44	$3.6850e - 006$	100	0.42200	36	$4.4713e - 006$
	40	0	2	59	$3.9598e - 006$	100	1.65700	49	$3.7638e - 006$
	50	0	6.6560	75	$4.2605e - 006$	100	5.5000	62	$3.7857e - 006$

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