

ΟΝ/ΜΟ : ΠΑΝΟΠΟΥΛΟΣ ΓΕΩΡΓΙΟΣ

ΑΜ: Π2014062

Συνοπτική περιγραφή κώδικα:

Και στις 2 περιπτώσεις χρησιμοποιούμε την δομή `get_walltime` για να μετρήσουμε τον χρόνο προσπέλασης

Ο πίνακας υλοποιείται με συνεχόμενες θέσεις μνήμης και παίρνει την τιμή 2. από τη θέση 0 έως την θέση που αντιστοιχεί στον

αριθμό των γραμμών επί των στηλών, η οποία τιμή μας βοηθά στο να ποσοτικοποιήσουμε την απόδοση της κάθε περίπτωσης.

Η ποσοτικοποίηση της απόδοσης γίνεται με το άθροισμα όλων των στοιχείων των πινάκων (που ορίζουμε να είναι τα ίδια), μέσα στην αντίστοιχη για κάθε μία από τις 2 περιπτώσεις `for loop`(μία για την περίπτωση γραμμή προς γραμμή και μία για την περίπτωση στήλη προς στήλη), και με την χρονομέτρηση των `for loops` αυτών

`matrix1`: Εδώ γίνεται η χρονομέτρηση της περίπτωσης γραμμή προς γραμμή

`matrix2` :Εδώ γίνεται η χρονομέτρηση της περίπτωσης στήλη προς στήλη

Πίνακας αποτελεσμάτων:

`matrix1`:

για `DNROWS=100`

`time took :0.000040`

`sum= 2000000.000000`(είναι πάντα σταθερό)

για DNROWS=1000

time took :0.000449

για DNROWS=10000

time took :0.001824

για DNROWS=100000

time took :0.017900

matrix2:

για DNROWS=100

time took :0.000039

για DNROWS=1000

time took :0.000706

για DNROWS=10000

time took : 0.004886

για DNROWS=100000

time took : 0.083979

Εξήγηση αποτελεσμάτων:

Σύμφωνα με τα αποτελέσματα ο matrix1 είναι πιο αποδοτικός από τον matrix2 έκτος από την περίπτωση που έχουμε ως είσοδο 100 γραμμές

Αυτό σημαίνει η προσπέλαση γραμμή προς γραμμή είναι κατάλληλη για δισδιάστατους πίνακες με μεγάλο αριθμό γραμμών ενώ η προσπέλαση στήλη προς στήλη για μικρό αριθμό γραμμών