
ΠΑΡΑΛΛΗΛΑ ΣΥΣΤΗΜΑΤΑ ΑΣΚΗΣΗ-2Α

ΕΡΓΑΣΤΗΡΙΟΥ 2024-25 (15%)

Γράψτε ένα πρόγραμμα σε **OpenMP** το οποίο να ταξινομεί παράλληλα μία ακολουθία N ακεραίων $A[0...N-1]$, σύμφωνα με τον αναδρομικό αλγόριθμο ταξινόμησης '**multisort**' που παρατίθεται στις διαφάνειες #49-50 του αρχείου **OpenMP.ppt**.

[Συμβουλευτείτε μεταξύ των άλλων και στηριχτείτε για την υλοποίησή σας στον κώδικα του αρχείου **OpenMP_msor.c** (ενδεικτική υλοποίηση του αλγόριθμου ταξινόμησης mergesort σε OpenMP) που είναι αναρτημένο στο Eclass.]

Μετρήστε την απόδοση της υλοποίησής σας για μεγάλες τιμές του ' N ' και για διαφορετικό αριθμό νημάτων (threads), και συγκρίνετε μεταξύ τους σε σχέση με την επιτάχυνση που επιτυγχάνεται σε κάθε περίπτωση.

Παραδοτέα: κώδικας, σχολιασμός/τεκμηρίωση, ενδεικτικά τρεξίματα/αποτελέσματα.

ΠΑΡΑΛΛΗΛΑ ΣΥΣΤΗΜΑΤΑ ΑΣΚΗΣΗ-2Β

ΕΡΓΑΣΤΗΡΙΟΥ 2024-25 (25%)

1. Γράψτε ένα πρόγραμμα σε **CUDA** το οποίο (δοθέντων ως είσοδο των ' A ', ' N '), να βρίσκει για έναν δυσδιάστατο πίνακα ακεραίων $A(N \times N)$, τη μέση τιμή των στοιχείων του (m), και το μέγιστο (a_{max}) στοιχείο του. Ακολουθώντας να ελέγχει αν το μέγιστο στοιχείο του είναι μεγαλύτερο από το N -πλάσιο της μέσης τιμής του ($a_{max} > N * m$), και ανάλογα:

- i. Εάν αυτό ισχύει, να φτιάχνει έναν νέο πίνακα $B(N \times N)$ (τον οποίον να τυπώνει στο τέλος στην οθόνη) όπου:

$$B_{ij} = (m - A_{ij}) / a_{max}$$

Για τον παραπάνω πίνακα B ζητείται επίσης να υπολογίζεται και να τυπώνεται στην οθόνη, το ελάχιστο σε τιμή στοιχείο του (a_{min}).

- ii. Εάν αυτό δεν ισχύει, να φτιάχνει έναν νέο πίνακα $C(N \times N)$ (τον οποίον θα τυπώνει στο τέλος στην οθόνη) όπου:

$$C_{ij} = (A_{ij} + A_{i(j+1)} + A_{i(j-1)}) / 3$$

Θεωρήστε επίσης πως εάν $j+1=N$ τότε $A_{i(j+1)}=A_{i0}$, ενώ εάν $j-1=-1$ τότε $A_{i(j-1)}=A_{i(N-1)}$.

2. Γράψτε ένα πρόγραμμα σε **CUDA** το οποίο δοθέντος ενός δυσδιάστατου πίνακα ακεραίων $A(N \times N)$ υπολογίζει το μητρώο συνδιακύμανσης του A ως εξής:

- i. Για κάθε στήλη του πίνακα A υπολογίζει τον μέσο όρο των στοιχείων της στήλης.
- ii. Από κάθε στοιχείο μιας στήλης του πίνακα A αφαιρεί τον μέσο όρο της αντίστοιχης στήλης που υπολογίστηκε στο προηγούμενο βήμα.
- iii. Πολλαπλασιάζει τον πίνακα A του προηγούμενου βήματος με τον ανάστροφό του. Επειδή ο πίνακας που προκύπτει είναι συμμετρικός, αρκεί να υπολογιστεί είτε το άνω είτε το κάτω τριγωνικό μέρος του αποτελέσματος.

Άλλες Απαιτήσεις: Μετρήστε την απόδοση της υλοποίησής σας για διαφορετικές τιμές του 'N', διαφορετικό συνολικό αριθμό threads (π.χ. N, N² κ.α.), και διαφορετικό αριθμό blocks και threads/block. Εξετάστε επίσης τη δυνατότητα χρήσης shared memory σε κάθε υποερώτημα. Τέλος, όπου απαιτείται προστασία κρίσιμης περιοχής / reduction εξετάστε τη δυνατότητα χρήσης τόσο ατομικών εντολών όσο και τη δυνατότητα εφαρμογής αλγόριθμου δυαδικού δέντρου.

Παραδοτέα: Κώδικας, σχολιασμός/τεκμηρίωση, ενδεικτικά τρεξίματα/αποτελέσματα.

Καταληκτική Ημερομηνία Υποβολής: 12/1/2025 (σσ. καλείστε να ασχοληθείτε με το ένα μόνο κατ' επιλογή σας – ή αν το επιθυμείτε και με τα δύο – από τα δύο μέρη της Άσκησης – βλ. για περισσότερες λεπτομέρειες τη σχετική Ανακοίνωση που έχει αναρτηθεί στο Eclass). Οι Ασκήσεις που θα παραδώσετε θα εξεταστούν προφορικά στο τέλος του εξαμήνου.