

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΠΟΛΟΓΙΣΤΩΝ

ΠΑΡΑΛΛΗΛΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΣΕ ΠΡΟΒΛΗΜΑΤΑ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

2η Εργαστηριακή Άσκηση Ο αλγόριθμος των κ-μέσων “K-MEANS” με openMP

ΠΡΟΕΤΟΙΜΑΣΙΑ

Στα πλαίσια αυτής της άσκησης πρέπει να βελτιστοποιήσετε με openMP τον χρόνο εκτέλεσης του προγράμματος που υλοποιήσετε στην προηγούμενη άσκηση και είχε σαν θέμα τον αλγόριθμο k-means. Επειδή πρέπει να τον βελτιστοποιήσετε χρησιμοποιώντας το API της openMP και τις δυνατότητες παράλληλης επεξεργασίας σε επίπεδο διανυσματικής επεξεργασίας όπως αυτή προσφέρεται από την δήλωση `#pragma omp simd`, μελετήστε το εγχειρίδιο χρήσης και βρείτε από το διαδίκτυο και το e-class του μαθήματος παραδείγματα χρήσης της openMP 4.0+.

ΕΚΦΩΝΗΣΗ

Εργασία 1

Μελετήστε ξανά τον αλγόριθμο k-means, πάρτε μια καλή σειριακή έκδοση του προγράμματός σας από την προηγούμενη άσκηση και αποφασίστε ποιά τμήματα του προγράμματος μπορείτε να παραλληλοποιήσετε, ή κάντε τις απαραίτητες τροποποιήσεις στο πρόγραμμά σας έτσι ώστε να μπορεί να παραλληλοποιηθεί καλύτερα.

Στην συνέχεια με την βοήθεια του profiler εντοπίστε ποιές θα είναι οι βασικές συναρτήσεις που θα εκτελέσουν τους πλέον χρονοβόρους υπολογισμούς που χρειάζονται. Μεταγλωττίστε το πρόγραμμα σας και υπολογίστε τον χρόνο εκτέλεσής του.

Εργασία 2

Προσθέστε τις κατάλληλες δηλώσεις στον προεπεξεργαστή (`#pragma omp parallel`, `#pragma omp parallel for`) στα τμήματα που θέλετε να παραλληλοποιήσετε με τις κατάλληλες προσθήκες `private()` `reduction()`. Επιβεβαιώστε ότι τα αποτελέσματα των υπολογισμών του προγράμματός σας είναι σωστά. Προσοχή γιατί είναι πολύ εύκολο ένα σωστό πρόγραμμα σειριακής εκτέλεσης να δώσει λανθασμένα αποτελέσματα με απρόσεκτο παραλληλισμό. Μεταγλωττίστε το πρόγραμμα σας και υπολογίστε τον χρόνο εκτέλεσής του.

Εργασία 3

Τροποποιείτε τις δηλώσεις `private()` `reduction()` και προσθέστε δηλώσεις

`schedule()` για να βελτιώσετε τους χρόνους εκτέλεσης του προγράμματος. Μεταγλωττίστε το πρόγραμμα σας και υπολογίστε τον νέο χρόνο εκτέλεσής του.

Εργασία 4

Προσπαθείστε να εκμεταλευτείτε τις δυνατότητες παράλληλης επεξεργασίας SIMD (vector processing) της openMP που υλοποιείται με την δήλωση `#pragma omp simd`, εκμεταλευόμενοι τις δυνατότητες της αρχιτεκτονικής των καταχωρητών του επεξεργαστή που διαθέτετε. Προσθέστε στα `for` που μπορούν να παραλληλοποιηθούν με διανυσματική επεξεργασία τα κατάλληλα `#pragma omp simd` και μεταγλωττίστε με `-O3 -fort-info` για να πληροφορηθείτε ποιά τμήματα του προγράμματός σας παραλληλοποιήθηκαν. Εκτελέστε το πρόγραμμα για να δείτε τι βελτιώσεις στους χρόνους υπολογισμού πετύχατε.

Επιτέλους τελειώσατε....

ΠΑΡΑΔΟΤΕΑ

- Για την εργασία 1 στείλτε τον κώδικα σε `c` ή `c++` και σχολιάστε τις αλλαγές που κάνατε (αν κάνατε) στις συναρτήσεις του προγράμματος. Να βάλετε σχόλια. Και να αποτυπώσετε την έξοδο του προγράμματος τυπώνοντας την αθροιστική ελάχιστη απόσταση των διανυσμάτων από το κοντινότερο κέντρο για κάθε βήμα του αλγόριθμου: `time ./myprogram`
- Για την εργασία 2 στείλτε τον κώδικα σε `c` ή `c++` υλοποιήσατε και να τυπώσετε τα αποτελέσματα του προγράμματος εκτελώντας την ίσια με την εργασία εντολή: `time ./myprogram`
Να βάλετε σχόλια στα σημεία παραλληλοποίησης αιτιολογώντας τις επιλογές σας.
- Για την εργασία 3 να στείλτε τις διαφορετικές εκδόσεις του προγράμματός σας. Και να κάνετε ότι και στην εργασία 2.
- Για την εργασία 4. Οτι και στην εργασία 3, δηλαδή να στείλτε τις εκδόσεις του κώδικα σε `c` ή `c++` που υλοποιήσατε με σχόλια στα σημεία που προσπαθείσατε να κάνετε αυτόματη παραλληλοποίηση με διανυσματική επεξεργασία. Στείλτε τα μηνύματα του μεταγλωττιστή και σχολιάστε τα αποτελέσματα της προσπάθειάς σας. Τυπώσετε τα αποτελέσματα του προγράμματος εκτελώντας την ίσια με την εργασία εντολή: `time ./myprogram`