

Σετ προγραμμάτων #2
(OpenMP)**Πίνακας επί πίνακα, με παραλληλοποίηση βρόχων (35%)**

Πρέπει να υλοποιήσετε και να χρονομετρήσετε παράλληλο πρόγραμμα για τον πολλαπλασιασμό τετραγωνικών πινάκων, χρησιμοποιώντας OpenMP και παραλληλοποιώντας (έναν κάθε φορά) τους βρόχους του κλασσικού σειριακού προγράμματος (βλ. λεπτομέρειες παρακάτω).

Παραλληλοποίηση εύρεσης πρώτων αριθμών (30%)

Σας δίνεται ένα σειριακό πρόγραμμα στο οποίο, δεδομένου του N , η συνάρτηση `serial_primes()` υπολογίζει το πλήθος των πρώτων αριθμών καθώς και τον μεγαλύτερο πρώτο αριθμό μέχρι και το N . Σας ζητείται να συμπληρώσετε τη συνάρτηση `openmp_primes()` ώστε να κάνει τους ίδιους υπολογισμούς παράλληλα, χρησιμοποιώντας το OpenMP. Δεν επιτρέπεται να αλλάξετε τον αλγόριθμο, απλά να μοιράσετε σωστά τη δουλειά.

- Πρέπει να χρονομετρήσετε και να συγκρίνετε με την σειριακή έκδοση
- Πρέπει να δοκιμάσετε εναλλακτικούς τρόπους διαμοιρασμού της δουλειάς μεταξύ των νημάτων και να καταλήξετε (αιτιολογώντας) στον καλύτερο δυνατό.

Πίνακας επί πίνακα, με tasks (35%)

Πρέπει να υλοποιήσετε και να χρονομετρήσετε παράλληλο πρόγραμμα για τον πολλαπλασιασμό τετραγωνικών πινάκων, χρησιμοποιώντας εργασίες του OpenMP (tasks). Η κάθε εργασία θα πρέπει να είναι ο υπολογισμός ενός block (υποπίνακα) του αποτελέσματος, μεγέθους $S \times S$.

- Το μέγεθος S του block (και άρα το πλήθος των εργασιών) θα πρέπει να είναι παραμετροποιήσιμο.
- Θα πρέπει να πειραματιστείτε με διάφορες τιμές για το S (βλ. παρακάτω).

Λεπτομέρειες**Απαιτούμενα**

- Θα πρέπει να παραδώσετε πλήρη αναφορά, περιλαμβάνοντας και γραφικές παραστάσεις χρονομετρήσεων καθώς και συζήτηση γύρω από τα αποτελέσματα.
- Τα προγράμματά σας (πηγαίοι κώδικες + αναφορά) θα πρέπει να τα παραδώσετε με `turnin set2@mye023`. Πληροφορίες στην ιστοσελίδα του μαθήματος.
- Για τη χρονομέτρηση θα πρέπει να χρησιμοποιηθούν οι κλήσεις χρονομέτρησης που παρέχει το ίδιο το OpenMP (`omp_get_wtime()` κλπ).
- Τα προγράμματά σας για τον απλό πολλαπλασιασμό πινάκων να τα δοκιμάσετε με **4 νήματα** (και να τα συγκρίνετε με τον καθαρό σειριακό κώδικα). **Προσοχή:** θα πρέπει να παραλληλοποιήσετε κάθε φορά **μόνο έναν βρόχο**, χρησιμοποιώντας `static` και `dynamic` schedules για το `for`. Θα πρέπει να δοκιμάσετε να παραλληλοποιήσετε (έναν τη φορά) και τους 3 βρόχους `for`. Προσέξτε διότι ανάλογα με το ποιον βρόχο παραλληλοποιείτε, ίσως απαιτείται αμοιβαίος αποκλεισμός! Ουσιαστικά θα φτιάξετε 3 προγράμματα OpenMP, καθένα με παραλληλοποιημένο έναν από τους βρόχους και για καθένα από αυτά θα δοκιμάσετε και το `static` και το `dynamic` schedule (συνολικά, δηλαδή, 6 περιπτώσεις).

- Τα προγράμματά σας για τον πολλαπλασιασμό πινάκων με tasks να τα δοκιμάσετε με **4 νήματα** (και να τα συγκρίνετε με τον καθαρό σειριακό κώδικα) και να δοκιμάσετε 3 διαφορετικά πλήθη εργασιών (16, 256, 1024).
- Για κάθε περίπτωση, το πρόγραμμα θα εκτελείται τουλάχιστον 4 φορές και ο τελικός χρόνος θα είναι ο μέσος όρος των τεσσάρων χρόνων.

Παρατηρήσεις

1. Η ανάπτυξη των προγραμμάτων σας μπορεί να γίνει οπουδήποτε αλλά η εκτέλεση και χρονομέτρηση των πειραμάτων σας θα πρέπει να γίνει σε υπολογιστές του τμήματος οι οποίοι διαθέτουν 4-πύρηνους επεξεργαστές (π.χ. opti7020ws08).
2. Τα αντίστοιχα σειριακά προγράμματα μπορείτε να τα βρείτε στην ιστοσελίδα του μαθήματος.
3. Για τον πολλαπλασιασμό πινάκων, δοκιμάστε πίνακες ακεραίων 1024×1024 στοιχείων (δίνονται στην ιστοσελίδα του μαθήματος). Τα στοιχεία τους θα πρέπει να τα αποθηκεύσετε σε αρχεία ώστε να μπορέσετε να ελέγξετε τη σωστή λειτουργία των προγραμμάτων σας συγκρίνοντας με τα αποτελέσματα του σειριακού (εγγυημένα σωστού) προγράμματος. **Προσοχή:** μην χρονομετρήσετε το μέρος του κώδικα που διαβάζει / γράφει στα αρχεία.

Προθεσμία παράδοσης:

Δευτέρα, 1 Μαΐου 2017

Βασίλειος Δημακόπουλος