

Εργασία #1
(OpenMP)**Παραλληλοποίηση εύρεσης πρώτων αριθμών (35%)**

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

Θόλωση εικόνων (35%)

Σας δίνεται ένα σειριακό πρόγραμμα το οποίο εφαρμόζει Gaussian blur προκειμένου να θολώσει (ή να ομαλοποιήσει) μία εικόνα. Η συνάρτηση που κάνει τη θόλωση είναι η `gaussian_blur_serial()`, η οποία παίρνει μία εικόνα `imgin` και παράγει τη θολωμένη της εκδοχή `imgout`, βάσει μίας ακτίνας θόλωσης `radius` (όσο μεγαλύτερη η ακτίνα, τόσο πιο έντονο το θόλωμα). Σας ζητείται να γίνεται η θόλωση παράλληλα χρησιμοποιώντας το OpenMP ως εξής (χωρίς να αλλάξετε τον αλγόριθμο, απλά να μοιράσετε σωστά τη δουλειά):

- να συμπληρώσετε τη συνάρτηση `gaussian_blur_omp_loops()` παραλληλοποιώντας κατάλληλα `loop(s)`.
- να συμπληρώσετε τη συνάρτηση `gaussian_blur_omp_tasks()` όπου για την παραλληλοποίηση γίνεται χρήση `tasks`. Θεωρείστε ότι μία γραμμή της εικόνας αποτελεί ένα `task`.

Task dependencies (30%)

Από την έκδοση 4.0 και μετά, το OpenMP υποστηρίζει εξαρτήσεις μεταξύ `tasks` (task dependencies), οι οποίες καθορίζουν τη σειρά με την οποία θα εκτελεστούν τα `tasks` που δημιουργούνται.

Μελετήστε τι είναι τα `task dependencies`, που χρησιμεύουν και με ποιο τρόπο μπορούν να εκφραστούν στο OpenMP, έκδοση 4.0. Πρέπει να παραδώσετε γραπτή περιγραφή (1–2 σελίδες, όχι παραπάνω) μαζί με ένα μικρό πρόγραμμα επίδειξης.

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 set1@mye023`. Πληροφορίες στην ιστοσελίδα του μαθήματος.
- Για τη χρονομέτρηση μπορείτε να χρησιμοποιήσετε κλήσεις χρονομέτρησης που παρέχει το ίδιο το OpenMP (`omp_get_wtime()` κλπ) ή την `gettimeofday()`.
- Τα προγράμματά σας να τα δοκιμάσετε με **1, 2, 3 και 4 νήματα** (και να τα συγκρίνετε με τον καθαρό σειριακό κώδικα).
- Για κάθε περίπτωση, ένα 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 opti7020 του τμήματος οι οποίοι διαθέτουν 4-πύργους επεξεργαστές (π.χ. opti7020ws08).
2. Τα αντίστοιχα σειριακά προγράμματα μπορείτε να τα βρείτε στην ιστοσελίδα του μαθήματος.
3. Για τη θόλωση εικόνων δίνεται στην ιστοσελίδα του μαθήματος εικόνα με ανάλυση 500×500 , 1000×1000 και 1500×1500 . Πειραματιστείτε με όλες, αλλά δώστε αποτελέσματα μόνο για την μεγαλύτερη εικόνα και για ακτίνα θόλωσης ίση με 8.

Προθεσμία παράδοσης:

Τετάρτη, 24 Απριλίου 2024

Βασίλειος Δημακόπουλος