



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ
www.cslab.ece.ntua.gr

16 Μαΐου 2014

ΣΥΣΤΗΜΑΤΑ ΠΑΡΑΛΛΗΛΗΣ ΕΠΕΞΕΡΓΑΣΙΑΣ
Εξετάσεις Κανονικής Περιόδου Ακ. Έτους 2013-2014

Η εξέταση γίνεται με κλειστά βιβλία και σημειώσεις. Μπορείτε να έχετε μαζί σας μόνο μία κόλλα Α4. Διάρκεια εξέτασης $2^{1/2}$ ώρες.

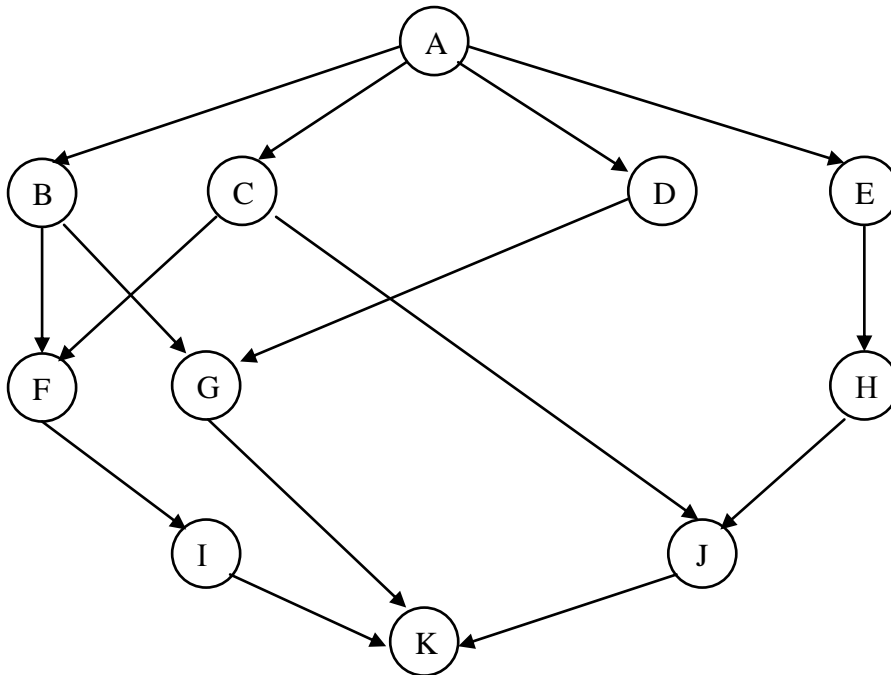
Θέμα 1ο (15%):

Θέλουμε να διασυνδέσουμε 6000 υπολογιστικούς κόμβους. Καταστρώστε τις τοπολογίες 2D-torus, 3D-torus, υπερκύβο και επιλέξτε την καταλληλότερη στις εξής περιπτώσεις:

- Αν βασικό κριτήριο επιλογής είναι η υψηλή επίδοση (ελαχιστοποίηση χρόνου επικοινωνίας με τον πιο απομακρυσμένο κόμβο).
- Αν βασικό κριτήριο επιλογής είναι το χαμηλό κόστος.
- Αν υιοθετηθεί τεχνολογία wormhole routing που μειώνει σημαντικά το χρόνο πρόσβασης στους πιο απομακρυσμένους κόμβους και δοθεί παρόμοια βαρύτητα στα κριτήρια επίδοση και κόστος.

Θέμα 2ο (15%):

Δίνεται ο παρακάτω γράφος εξαρτήσεων:



- Δώστε ψευδοκώδικα που παραλληλοποιεί τον παραπάνω γράφο χρησιμοποιώντας fork-join (εναλλακτικά μπορείτε να χρησιμοποιήσετε τα constructs των OpenMP, Cilk, ή TBBs). (5%)
- Επιτύχατε το μέγιστο δυνατό παραλληλισμό; Περιγράψτε σχήμα με ψευδοκώδικα (κατά προτίμηση) ή με λόγια που να αξιοποιεί όλο τον παραλληλισμό του παραπάνω γράφου. (10%)

Θέμα 3^ο (10%):

Εκτελείτε τους παρακάτω τρεις υπολογιστικούς πυρήνες (A, B, C) και λαμβάνετε τα διαγράμματα επιτάχυνσης που δίνονται στη συνέχεια (1, 2, 3). Αντιστοιχείστε τα διαγράμματα με τους υπολογιστικούς πυρήνες και δικαιολογήστε την απάντησή σας. Θεωρήστε ότι το N είναι πολύ μεγάλο.

| | | |
|---|---|---|
| <pre>pragma omp parallel for for (i = 0; i < N; i++){ double a = 0x42; a = a & A[i]; }</pre> | <pre>pragma omp parallel for for (i = 0; i < N; i++){ #pragma omp atomic a = a + A[i]; }</pre> | <pre>pragma omp parallel for for (i = 0; i < N; i++){ double a = 0x42; a += exp(sqrt(A[i])); }</pre> |
| A | B | C |
| | | |
| 1 | 2 | 3 |

Θέμα 4^ο (40%):

Δίνεται το παρακάτω κομμάτι κώδικα:

```
for (k = 0; k < STEPS; k++){
    ...
    for (i = 0; i < N; i++)
        for (j = 0; j < N; j++)
            y[i] += A[i][j] * x[j];
    ...
    for (i=0; i<N; i++)
        r = y[i] * y[i];
    ...
    for (i=0; i<N; i++)
        x[i] = r * x[i];
}
```

όπου ο πίνακας A και τα διανύσματα x, y περιέχουν πραγματικούς αριθμούς διπλής ακρίβειας (8 bytes).

A. Ο παραπάνω κώδικας εκτελείται σε επεξεργαστή με ισχύ 4GFLOP/sec, bandwidth διαδρόμου μνήμης 6GByte/sec και μέγεθος cache 2MB. Τα δεδομένα είναι αριθμοί κινητής υποδιαστολής διπλής ακρίβειας (8bytes) Θεωρήστε N=2000, STEPS=100. Μπορείτε να αγνοήσετε τις προσθέσεις.

- i. Χρησιμοποιήστε το κλασικό μοντέλο πρόβλεψης και υπολογίστε το χρόνο εκτέλεσης του αλγορίθμου. (5%)
- ii. Ομοίως με πριν, χρησιμοποιώντας το μοντέλο roofline. Συγκρίνετε το αποτέλεσμα σε σχέση με το κλασικό μοντέλο και σχολιάστε ποιο θεωρείτε πιο αξιόπιστο (5%)

B. Δώστε παράλληλη υλοποίηση σε ψευδοκώδικα του παραπάνω αλγορίθμου στο μοντέλο της κοινής μνήμης. Με βάση τα αποτελέσματα του ερωτήματος A, τι αναμένετε για την κλιμάκωση του αλγορίθμου; (5%)

Γ. Για το μοντέλο ανταλλαγής μηνυμάτων, επιλέξτε και σχεδιάστε μία κατανομή δεδομένων σε επεξεργαστές και επισημάνετε τα σημεία και τον τρόπο επικοινωνίας (10%).

E. Δώστε παράλληλη υλοποίηση σε ψευδοκώδικα του παραπάνω αλγορίθμου στο μοντέλο της ανταλλαγής μηνυμάτων. (15%)

Θέμα 5^ο (20%):

A. Εξηγήστε για ποιο λόγο τα κλειδώματα του Peterson και του Lamport δεν χρησιμοποιούνται στην πράξη. Σε ποια λύση καταφεύγουν οι σύγχρονοι επεξεργαστές για την υλοποίηση κλειδωμάτων (4%);

B. Εξηγήστε πώς η κίνηση στο μοιραζόμενο διάδρομο εξαιτίας του πρωτοκόλλου συνάφειας κρυφής μνήμης επηρεάζουν την επίδοση των κλειδωμάτων TAS, TTAS και Queue (4%).

Γ. Αναφέρετε πλεονεκτήματα και μειονεκτήματα κάθε μιας από τις στρατηγικές ταυτόχρονης πρόσβασης σε κοινές δομές δεδομένων coarse-grain locking, fine-grain locking, optimistic synchronization, lazy synchronization και non-blocking (4%).

Δ. Συγκρίνετε τις στρατηγικές blocking και non-blocking synchronization (4%).

E. Εξηγήστε τους όρους deadlock-free, starvation-free, lock-free και wait-free (4%).