



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

ΣΥΣΤΗΜΑΤΑ ΠΑΡΑΛΛΗΛΗΣ ΕΠΕΞΕΡΓΑΣΙΑΣ

9ο εξάμηνο ΗΜΜΥ, ακαδημαϊκό έτος 2020-21

ΑΣΚΗΣΗ 1 - Παραλληλοποίηση αλγορίθμων σε πολypύρηνες αρχιτεκτονικές κοινής μνήμης

Παράδοση Ενδιάμεσης Αναφοράς: **10 Νοεμβρίου**¹
Παράδοση Τελικής Αναφοράς: **24 Νοεμβρίου**²

1 Conway's Game of Life

1.1 Σκοπός της ενότητας

Σκοπός της συγκεκριμένης ενότητας της άσκησης είναι η εξοικείωση με τις υποδομές του εργαστηρίου (πρόσβαση στα συστήματα, μεταγλώττιση προγραμμάτων, υποβολή εργασιών κλπ) μέσα από την παραλληλοποίηση ενός απλού προβλήματος σε αρχιτεκτονικές κοινής μνήμης.

1.2 Το Παιχνίδι της Ζωής

Το *Παιχνίδι της Ζωής* (*Conway's Game of Life*) λαμβάνει χώρα σε ένα ταμπλό με κελιά δύο διαστάσεων. Το περιεχόμενο κάθε κελιού μπορεί να είναι γεμάτο (alive) ή κενό (dead), αντικατοπτρίζοντας την ύπαρξη ή όχι ζωντανού οργανισμού σε αυτό, και μπορεί να μεταβεί από τη μία κατάσταση στην άλλη μία φορά εντός συγκεκριμένου χρονικού διαστήματος. Σε κάθε βήμα (χρονικό διάστημα), κάθε κελί εξετάζει την κατάστασή του και αυτή των γειτόνων του (δεξιά, αριστερά, πάνω, κάτω και διαγώνια) και ακολουθεί τους παρακάτω κανόνες για να ενημερώσει την κατάστασή του:

- Αν ένα κελί είναι ζωντανό και έχει λιγότερους από 2 γείτονες πεθαίνει από μοναξιά.
- Αν ένα κελί είναι ζωντανό και έχει περισσότερους από 3 γείτονες πεθαίνει λόγω υπερπληθυσμού.
- Αν ένα κελί είναι ζωντανό και έχει 2 ή 3 γείτονες επιβιώνει μέχρι την επόμενη γενιά.
- Αν ένα κελί είναι νεκρό και έχει ακριβώς 3 γείτονες γίνεται ζωντανό (λόγω αναπαραγωγής).

¹ mail-to: nikela@cslab.ece.ntua.gr, cc: goumas@cslab.ece.ntua.gr,
filename: a1-parlabXX-interim.pdf, mail subject: a1-parlabXX-interim

² mail-to: nikela@cslab.ece.ntua.gr, cc: goumas@cslab.ece.ntua.gr,
filename: a1-parlabXX-final.pdf, mail subject: a1-parlabXX-final

1.3 Δεδομένα

Σας δίνεται η σειριακή υλοποίηση του παιχνιδιού της ζωής στο αρχείο *Game_Of_Life.c*, στο φάκελο /home/parallel/pps/2020-2021/a1 στον scirouter.

1.4 Ζητούμενα

1. Εξοικειωθείτε με τον τρόπο μεταγλώττισης και υποβολής εργασιών στις συστοιχίες. Για οδηγίες σύνδεσης, μεταγλώττισης, εκτέλεσης κ.λ.π. των προγραμμάτων σας συμβουλευτείτε τις "ΟΔΗΓΙΕΣ ΕΡΓΑΣΤΗΡΙΟΥ", που είναι διαθέσιμες στο: <http://www.cslab.ece.ntua.gr/courses/pps/files/fall2020/pps-lab-guide.pdf>. Η υποβολή εργασιών μεταγλώττισης και εκτέλεσης θα γίνει στην ουρά **parlab**, όπως παρουσιάστηκε στον οδηγό του εργαστηρίου.
2. Αναπτύξτε παράλληλο πρόγραμμα στο μοντέλο κοινού χώρου διευθύνσεων (shared address space) με τη χρήση του OpenMP.
3. Πραγματοποιείτε μετρήσεις επίδοσης σε έναν από τους κόμβους της συστοιχίας των clones για 1,2,4,6,8 πυρήνες και μεγέθη ταμπλώ 64×64 , 1024×1024 και 4096×4096 (σε όλες τις περιπτώσεις τρέξτε το παιχνίδι για 1000 γενιές). *Απενεργοποιήστε την εκτύπωση της εξόδου πριν πραγματοποιήσετε μετρήσεις!*
4. Συγκεντρώστε τα αποτελέσματά σας σε γραφήματα για το χρόνο και το speedup, καθώς και τα σχόλιά σας, στην **ενδιάμεση αναφορά**.
(Ακολουθήστε τις υποδείξεις στη σελίδα 1 για την υποβολή της Ενδιάμεσης Αναφοράς.)
5. **Προαιρετικά:** Αναζητήστε στο διαδίκτυο ειδικές αρχικοποιήσεις του ταμπλώ που οδηγούν σε ενδιαφέροντα οπτικά αποτελέσματα και δώστε την εξέλιξη των γενιών σε μορφή κινούμενης εικόνας.

2 Παραλληλοποίηση και βελτιστοποίηση του αλγορίθμου Floyd-Warshall σε αρχιτεκτονικές κοινής μνήμης

2.1 Σκοπός της ενότητας

Η ενότητα αυτή αποτελεί το βασικό κορμό της άσκησης. Στόχος της ενότητας είναι να αναπτύξετε διαφορετικές παράλληλες εκδόσεις του αλγορίθμου Floyd-Warshall, να αξιολογήσετε την παραγωγικότητα (productivity) ανάπτυξης παράλληλου κώδικα και την τελική επίδοση του παράλληλου προγράμματος, επιλέγοντας ένα από τα δύο προγραμματιστικά εργαλεία για αρχιτεκτονικής κοινής μνήμης: OpenMP ή Threading Building Blocks (TBBs).

2.2 Ο αλγόριθμος Floyd-Warshall

Ο αλγόριθμος των Floyd-Warshall (FW) υπολογίζει τα ελάχιστα μονοπάτια ανάμεσα σε όλα τα ζεύγη των N κόμβων ενός γράφου (all-pairs shortest path). Θεωρώντας το γράφο αποθηκευμένο στον πίνακα γειτνίασης A , ο αλγόριθμος έχει ως εξής:

```
for (k=0; k<N; k++)  
  for (i=0; i<N; i++)  
    for (j=0; j<N; j++)  
       $A[i][j] = \min(A[i][j], A[i][k] + A[k][j]);$ 
```

Εκτός από την *standard* έκδοση του αλγορίθμου, έχουν προταθεί άλλες δύο εκδόσεις, μία *αναδρομική* (recursive) και μία *tiled*, προκειμένου να αξιοποιείται καλύτερα η κρυφή μνήμη (βλ. παρουσίαση άσκησης).

2.3 Δεδομένα

Οι τρεις σειριακές εκδόσεις του αλγορίθμου Floyd-Warshall σάς δίνονται στον scirouter στο φάκελο /home/parallel/pps/2020-2021/a1/FW-serial.

2.4 Ζητούμενα

Για τις εκδόσεις του αλγορίθμου Floyd-Warshall που σας δίνονται:

1. Ανακαλύψτε τον παραλληλισμό του αλγορίθμου σε κάθε έκδοση και σχεδιάστε την παραλληλοποίησή του. Περιγράψτε τη στρατηγική παραλληλοποίησης στην **ενδιάμεση αναφορά**.
2. Υλοποιήστε παράλληλες εκδόσεις του αλγορίθμου Floyd-Warshall (τουλάχιστον μία για κάθε σειριακή έκδοση) χρησιμοποιώντας το OpenMP ή τα Threading Building Blocks και πραγματοποιήστε μετρήσεις για μεγέθη πινάκων 1024x1024, 2048x2048 και 4096x4096 για 1, 2, 4, 8, 16, 32 και 64 threads στο μηχάνημα *sandman*.
3. Αξιολογήστε τις παρατηρήσεις σας και τις μετρήσεις σας από τις πρώτες υλοποιήσεις, την κατανόηση της αρχιτεκτονικής του μηχανήματος *sandman*, τις δομές παράλληλου προγραμματισμού που προσφέρει κάθε εργαλείο παραλληλοποίησης και τις δυνατότητες του μεταγλωττιστή για να βελτιώσετε την παράλληλη επίδοση του αλγορίθμου Floyd-Warshall.
4. Πραγματοποιήστε μετρήσεις για τις καλύτερες παράλληλες εκδόσεις, για μεγέθη πινάκων 1024x1024 2048x2048 και 4096x4096, για 1, 2, 4, 8, 16, 32 και 64 threads στο μηχάνημα *sandman*.
5. Περιγράψτε τις καλύτερες παράλληλες εκδόσεις σας και παρουσιάστε τις μετρήσεις σας στην **τελική αναφορά**. Αναφέρετε τον καλύτερο χρόνο που επιτύχατε!

2.5 Περιβάλλον εκτέλεσης

- Για την ενότητα αυτή, θα χρησιμοποιήσετε το μηχάνημα *sandman*, που ανήκει στην ουρά *serial* (βλ. παρουσίαση άσκησης). Για να υποβάλλετε ένα script, έστω `script.sh`, στο μηχάνημα, δίνετε την εντολή `qsub` ως εξής:

```
$ qsub -q serial -l nodes=sandman:ppn=64 script.sh
```
- ΠΡΟΣΟΧΗ: Μπορείτε να χρησιμοποιείτε τα μηχανήματα της συστοιχίας *parlab* για την ανάπτυξη και τον έλεγχο παράλληλων προγραμμάτων.
- Για να ρυθμίσετε το περιβάλλον μεταγλώττισης και εκτέλεσης για OpenMP και TBBs, θα χρησιμοποιήσετε τα `modules`. Λεπτομέρειες για τη χρήση των `modules` μπορείτε να βρείτε εδώ: <https://admin.cslab.ece.ntua.gr/trac/wiki/EnvModulesUsage>. Φορτώνετε το αντίστοιχο `module` ως εξής:
 - Για το OpenMP: `module load openmp`
 - Για τα TBBs: `module load tbbz`
- Στον scirouter, στο φάκελο /home/parallel/pps/2020-2021/a1 θα βρείτε το φάκελο `tbb-workspace` με οδηγίες για τη μεταγλώττιση και την εκτέλεση προγραμμάτων σε TBBs και παραδείγματα.