

Παράλληλος Προγραμματισμός 2019

Προγραμματιστική Εργασία #1

(Προσοχή: η παράδοση της άσκησης θα γίνει μέσω *github*. Διαβάστε τις οδηγίες στο τέλος της εκφώνησης)

Θέμα

Σκοπός της άσκησης είναι να προσπαθήσετε να **αυξήσετε την απόδοση** της απλής μεθόδου πολλαπλασιασμού πινάκων 2 διαστάσεων (με τρία loops, το ένα μέσα στο άλλο, με πολυπλοκότητα $O(n^3)$) **χρησιμοποιώντας εντολές SSE2**.

- Τα προγράμματα που θα κατασκευάσετε θα εκτελούν τον πολλαπλασιασμό των πινάκων $C=A*B$. Για τις ανάγκες της άσκησης όλοι οι πίνακες είναι **τύπου float** διαστάσεων $N \times N$ (ορίστε το με $-DN=...$), όπου το N **διαίρεται δια 4**.
- Η δέσμευση κάθε πίνακα δύο διαστάσεων θα πρέπει υποχρεωτικά να γίνεται δυναμικά (με `malloc` ή `posix_memalign`) **ως μονοδιάστατος πίνακας σε συνεχόμενες θέσεις μνήμης** – δείτε το υλικό του εργαστηρίου:
<https://gist.github.com/mixstef/71913bae3dc774d52b2beecd58985160>
- Προσοχή:** για να επιτύχετε ευνοϊκή προσπέλαση της κρυφής μνήμης, θεωρήστε ότι έχει ήδη γίνει **αντιμετάθεση γραμμών – στηλών** στον πίνακα B: η αποθήκευση σε αυτόν τον πίνακα είναι **κατά στήλες**.

Ζητούμενο

Υλοποιήστε τα εξής:

A) **Κατασκευάστε πρόγραμμα** σε C για τον πολλαπλασιασμό δύο float πινάκων $N \times N$ και την αποθήκευση του αποτελέσματος σε τρίτο πίνακα $N \times N$, χωρίς τη χρήση εντολών SSE2.

- Ακολουθήστε την τακτική των παραδειγμάτων που έγιναν στο εργαστήριο: δέσμευση πινάκων, αρχικοποίηση, μέτρηση χρόνου (αρχή), φορτίο, μέτρηση χρόνου (τέλος), έλεγχος αποτελέσματος, αποδέσμευση πινάκων.
- Στον κώδικα του φορτίου **χρησιμοποιήστε δείκτες** για την προσπέλαση των δεδομένων.
- Χρησιμοποιήστε επίπεδο βελτιστοποίησης $-O2$ αλλά **φροντίστε να αποφύγετε την απαλοιφή των loops του φορτίου από τον μεταγλωττιστή!**

B) **Ξαναγράψτε νέα εκδοχή του προγράμματος**, χρησιμοποιώντας αυτή τη φορά εντολές SSE2. Το ζητούμενο είναι το ίδιο με την περίπτωση (A).

- Θα πρέπει να χρησιμοποιηθούν **4-άδες floats**, συνεπώς θα πρέπει να χρησιμοποιήσετε τον τύπο `__m128` (ή δείκτες αυτού του τύπου), καθώς και `intrinsics` της μορφής `_mm_????_ps`. Συμβουλευτείτε τα παραδείγματα που έγιναν στο εργαστήριο:
<https://gist.github.com/mixstef/de602d20c196d2234d8ab88d283be5a9>
 - Σε 64-bit συστήματα έχετε 16 SSE καταχωρητές στη διάθεσή σας. Αν οι μεταβλητές τύπου `__m128` (όχι οι δείκτες) που χρησιμοποιείτε είναι τόσες ή λιγότερες, ο μεταγλωττιστής θα τις κρατήσει με μεγάλη πιθανότητα σε καταχωρητές, αυξάνοντας την απόδοση.
- Ξεδιπλώστε (νοερά) το εσωτερικό loop του φορτίου (loop unrolling) για να βρείτε πού και πώς θα χρησιμοποιήσετε τις εντολές SSE2.

- Θυμηθείτε ότι για να παραχθεί το στοιχείο C_{ij} θα πρέπει να βρείτε το εσωτερικό γινόμενο (dot product) της γραμμής A_i με τη στήλη B_j . Σε περίπτωση που υπολογίζετε το γινόμενο με εντολές SSE2 ανά τετράδες, θα πρέπει να προσέξετε να αθροίσετε μαζί την τελική τετράδα που προκύπτει, σε έναν μόνο αριθμό. Προσπαθήστε να σκεφτείτε έναν τρόπο για να γίνει αυτό, με ή χωρίς εντολές SSE2 (δείτε και το παράδειγμα “οριζόντιας άθροισης” στο υλικό του εργαστηρίου).

Γ) **Δοκιμάστε τις δύο παραλλαγές** για πίνακες με $N = 4, 40, 400, 4000$ (πριν προχωρήσετε σε μεγαλύτερες διαστάσεις, σκεφτείτε την πολυπλοκότητα $O(n^3)$). Μετρήστε την απόδοση για τις δύο περιπτώσεις (χωρίς και με εντολές SSE2). Παραθέστε τα αποτελέσματα των μετρήσεών σας.

Δ) **Εξηγήστε τα αποτελέσματα** που λαμβάνετε, αποδίδοντάς τα στα χαρακτηριστικά της αρχιτεκτονικής του υπολογιστή και στη χρήση των εντολών SSE2.

Παραδοτέο

Η παράδοση θα γίνει μέσω github. Οδηγίες:

1. Αντιγράψτε (**fork**) το repository <https://github.com/mixstef/parprog1819a1> στο δικό σας repository. Βεβαιωθείτε ότι δουλεύετε αποκλειστικά στο **master branch**.
2. Τροποποιήστε κατάλληλα τα αρχεία που περιέχονται στο repository σας με το δικό σας περιεχόμενο:
 - Συμπληρώστε τα στοιχεία σας στο αρχείο **README.md**.
 - Βάλτε τον κώδικά σας στα αρχεία **matmul-normal.c** και **matmul-sse.c** (οι δύο παραλλαγές που εξετάζετε).
 - Προσθέστε την αναφορά σας ως **report.pdf**.
 - **Προσοχή: πρέπει να διατηρήσετε τα ονόματα των παραπάνω αρχείων!**
3. Ενημερώστε το repository σας στο github εντός προθεσμίας. **Μην κάνετε pull request!**

Η εργασία είναι αυστηρά ατομική. Για την εγκυρότητα της υποβολής σας θα χρησιμοποιηθεί η χρονοσήμανση των αλλαγών (commits) των αρχείων σας.

Προθεσμία παράδοσης: Τρίτη 19/3/2019 15:00.