

Παράλληλος Προγραμματισμός 2022

Προγραμματιστική Εργασία #1

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

Θέμα

Στην επεξεργασία εικόνας συχνά εμφανίζεται ο μετασχηματισμός, κατά τον οποίο κάθε **pixel** αντικαθίσταται από μια νέα τιμή υπολογισμένη από **τα 8 γειτονικά pixels** (και το ίδιο). Κατά τον μετασχηματισμό, κάθε αρχικό pixel πολλαπλασιάζεται με μια **σταθερά**.

Έτσι, αν έχουμε τα **γειτονικά pixels**:

P0 P1 P2

P3 P4 P5

P6 P7 P8

και τις **σταθερές μετασχηματισμού**:

K0 K1 K2

K3 K4 K5

K6 K7 K8

τότε, για παράδειγμα, η νέα τιμή για το pixel P4 ισούται με:

$$\text{newP4} = P0 \cdot K0 + P1 \cdot K1 + P2 \cdot K2 + P3 \cdot K3 + P4 \cdot K4 + \\ + P5 \cdot K5 + P6 \cdot K6 + P7 \cdot K7 + P8 \cdot K8$$

Η διαδικασία επαναλαμβάνεται **για όλα τα pixels, για όλες τις γραμμές της εικόνας**.

Για τις ανάγκες της άσκησης θεωρήστε ότι:

- Η εικόνα είναι **μονόχρωμη** (ένας **float** αριθμός ανά **pixel**).
- Η εικόνα έχει ήδη μετατραπεί και φορτωθεί σε έναν **πίνακα float NxM** (αρχικοποιήστε με **τυχαίες τιμές** τον πίνακα πριν αρχίσετε τη μέτρηση του χρόνου).
- Οι νέες τιμές που υπολογίζετε θα αποθηκεύονται **σε δεύτερο πίνακα** με τις ίδιες διαστάσεις (αρχικοποιήστε και αυτόν τον πίνακα για να μεταφερθεί στην κρυφή μνήμη πριν αρχίσετε να μετράτε χρόνο).
- Τα **M** και **N** είναι τέτοια ώστε να βολεύουν στις πράξεις σας, π.χ. μπορείτε να θεωρήσετε ότι το N είναι πολλαπλάσιο του 4 ή οτιδήποτε άλλο σας βολεύει.
- Μπορείτε, αν θέλετε, **να αγνοήσετε τον υπολογισμό των pixels στο περίγραμμα της εικόνας**.
- Για τις ανάγκες της άσκησης, τα K0, K1, K2, K3, K5, K6, K7, K8 ισούνται με **0.5**, ενώ το K4 είναι ίσο με **5.0** (θα χρησιμοποιήσετε δηλαδή **2 σταθερές μόνο!**).

Ζητείται η επιλογή όσο το δυνατόν αποδοτικότερων SIMD πράξεων (από το σεντ SSE/SSE2) για την υλοποίηση του μετασχηματισμού. Σε αντίθεση με παραδείγματα που θα βρείτε on-line, τα οποία

χρησιμοποιούν 3 ανεξάρτητες συνιστώσες χρώματος (RGB) και εξάγουν την παραλληλία από αυτό ακριβώς το γεγονός, εδώ η εικόνα είναι μονόχρωμη και ο παραλληλισμός δυσκολότερος! **Βεβαιωθείτε ότι το τροποποιημένο πρόγραμμά σας δεν είναι αργότερο από το αρχικό!**

Ζητούμενο

α) Κατασκευάστε δοκιμαστικό πρόγραμμα σε C χωρίς εντολές SSE, έτσι ώστε να μελετήσετε τον χρόνο εκτέλεσης χωρίς πρόσθετες βελτιώσεις. Χρησιμοποιήστε πίνακες που χωράνε στην κρυφή μνήμη (L2) του συστήματός σας, έως π.χ. 1000x1000.

β) Κατασκευάστε το αντίστοιχο πρόγραμμα με εντολές SSE. Χρησιμοποιήστε ως βάση τον κώδικα που δόθηκε στο εργαστήριο (συμβουλευθείτε το site με το υλικό του μαθήματος). Μετρήστε τους αντίστοιχους χρόνους εκτέλεσης και εισάγετε περαιτέρω βελτιώσεις, αν χρειάζεται. Βεβαιωθείτε ότι οι αναγνώσεις και εγγραφές είναι ευθυγραμμισμένες στα 16 bytes!

γ) Ετοιμάστε αναφορά σε μορφή pdf, η οποία θα περιέχει:

- Σύντομη περιγραφή της μεθόδου αξιοποίησης των εντολών SSE που εφαρμόζετε.
- Πίνακες αποτελεσμάτων ανά παραλλαγή του προγράμματος για ενδεικτικά μεγέθη εικόνας.
- Σύντομο σχολιασμό των αποτελεσμάτων.
- Αναφορές σε πηγές που χρησιμοποιήσατε.

Παραδοτέο

Η παράδοση θα γίνει μέσω opencourses:

1. Στο μάθημα του Παράλληλου Προγραμματισμού στο opencourses επισκεφτείτε την ενότητα «**Εργασίες**».
2. Η κατάθεση του παραδοτέου σας θα γίνει στην εργασία «**Προγραμματιστική Εργασία #1**».
3. Τοποθετήστε **την αναφορά σας** (αρχείο pdf) και **τον κώδικά σας** (2 αρχεία C, το πρόγραμμά σας χωρίς και με εντολές SSE) σε ένα (και μοναδικό) αρχείο zip.
4. Ανεβάστε το αρχείο zip στο opencourses.

Η εργασία είναι αυστηρά ατομική.

Προθεσμία παράδοσης: Δευτέρα 28/3/2022.