

# Παράλληλος Προγραμματισμός 2019

## Προγραμματιστική Εργασία #2

---

**Ονοματεπώνυμο:** Σπυρίδων Θεοδωρόπουλος **A.M.:** Π2015035

## ΣΥΝΟΨΗ

Στην δεύτερη άσκηση στα πλαίσια του μαθήματος «Παράλληλος προγραμματισμός» κατασκευάστηκε ένας quicksort αλγόριθμος με την βοήθεια μιας δεξαμενής threads σταθερού αριθμού thread (στην συγκεκριμένη εργασία τεσσάρων thread). Τα threads μας δημιουργούνται στην αρχή του προγράμματος και καταστρέφονται στο τέλος του . Κατά την διάρκεια της “ζωής” τους τα threads αναλαμβάνουν πακέτα εργασίας από μια «global» ουρά εργασιών . Ουσιαστικά η εργασία των threads είναι η παραλαβή η προώθηση και η ταξινόμηση των μηνυμάτων που έρχονται από την main. Η διαχείριση αυτών των μηνυμάτων από τα thread οδηγούν τελικά στην ταξινόμηση των στοιχείων του τυχαίου πίνακα που δημιουργούμε στην αρχή του προγράμματος .

## ΡΟΗ ΚΩΔΙΚΑ ΚΑΙ ΕΠΕΞΗΓΗΣΗ ΤΩΝ ΣΥΝΑΡΤΗΣΕΩΝ

Το πρόγραμμα μας ξεκινάει με την δημιουργία και την τυχαία γέμιση του πίνακα των οποίων τα στοιχεία θα ταξινομηθούν. Στην συνέχεια δημιουργούνται τα threads. Τα threads ξεκινούν να λαμβάνουν δεδομένα σε μορφή μηνυμάτων τα οποία υποδεικνύουν στα thread τι πρέπει να κάνουν . Τα μηνύματα αυτά αποτελούνται από τρία πεδία .

- Το typ δηλαδή τον τύπο του μηνύματος (work - 0 χρειάζεται επεξεργασία , finish-1 έχει ολοκληρωθεί η επεξεργασία , 2 – terminate όταν έχει τελειώσει η επεξεργασία όλου του πίνακα )
- Το start η θέση δηλαδή του πρώτου στοιχείου του υποσυνόλου του πίνακα προς ταξινόμηση
- Το end η θέση δηλαδή του τελευταίου στοιχείου του υποσυνόλου του πίνακα προς ταξινόμηση

Οι συναρτήσεις που χρησιμοποιούνται είναι οι εξής(με την σειρά που είναι γραμμένοι στον κώδικα ) :

Swap() - Μια απλή συνάρτηση για την εναλλαγή των τιμών δυο μεταβλητών. Χρησιμεύει για την ταξινόμηση των τιμών.

Partition() – Διαχωρίζει τα κομμάτια του πίνακα αν το κομμάτι που λάβαμε είναι μεγαλύτερο από την σταθερά threshold που ορίσαμε. Αποτελεί βοηθητική της quicksort και πραγματοποιεί μια μικρή ταξινόμηση η οποία θα βοηθήσει να ολοκληρωθεί η ταξινόμηση .

Send() – Διαχειρίζεται την αποστολή των μηνυμάτων.

Quicksort() – Ταξινόμηση των στοιχείων του πίνακα με την βοήθεια της partition()

Insert() – Αναλαμβάνει την ταξινόμηση αν το κομμάτι που λάβαμε είναι μικρότερο από την σταθερά threshold που ορίσαμε

Receive() - Διαχειρίζεται την λήψη των μηνυμάτων.

\*thread\_func() – Ο κώδικας που εκτελεί κάθε thread . Λαμβάνει συνεχώς μηνύματα και τα διαχειρίζεται ανάλογα με τον τύπο τους .Όταν λάβει το μήνυμα του τερματισμού τερματίζει.

Main () – Η main συνάρτηση του προγράμματος μας. Δημιουργεί τον πίνακα που θα ταξινομήσουμε . Ελέγχει για διαφορά τυχόν errors (π.Χ. allocation error). Τερματίζει τα thread και το πρόγραμμα όταν τελειώσει η ταξινόμηση του πίνακα .