

Παράλληλος Προγραμματισμός 2019

Προγραμματιστική Εργασία #2

Ονοματεπώνυμο: Τουλάκης Αλέξανδρος

AM: Π2015067

Κώδικας Ουράς μηνυμάτων:

Η ουρά των μηνυμάτων είναι ένας πίνακας σταθερών θέσεων που προσδιορίζεται από την σταθερά στο πρόγραμμα με όνομα `QUEUE_SIZE`, η ουρά αυτήν είναι κυκλική και δηλώνεται καθολικά στον κώδικα μας. Ο τύπος του πίνακα αυτού είναι μια δομή που ονομάζεται "message" και τα συστατικά στοιχεία της είναι ο τύπος του μηνύματος, η αρχή και το τέλος για τον πίνακα. Για την διαχείριση της αναπτύχθηκαν οι συναρτήσεις `send` και `recv`. Η συνάρτηση `send` χρησιμοποιείται για να εισάγουμε δεδομένα στην ουρά, ενώ η `recv` λαμβάνει το τρέχων στοιχείο το οποίο βρίσκεται στο πρώτο στοιχείο της ουράς. Αξιοσημείωτο είναι το γεγονός ότι επειδή και οι δύο συναρτήσεις τροποποιούν παράλληλα έναν κοινό διαμοιραζόμενο πόρο έγινε χρήση `Mutexes` προκειμένου να αποφευχθεί η πολλαπλή πρόσβαση σε αυτό καθώς και χρήση `conditional variables` προκειμένου να τοποθετήσουμε threads σε `sleep` κατάσταση. Η συνθήκη για την συνάρτηση `send` είναι όταν η ουρά έχει γεμίσει, ενώ στην `recv` όταν η ουρά δεν έχει στοιχεία μέσα και οι έλεγχοι αυτοί γίνονται μέσα σε `while loop` για να αποφύγουμε το πρόβλημα του spurious wakeup όπου ένα thread μπορεί να ξυπνήσει ενώ δεν θα έπρεπε. Η πληροφορία σχετικά με τα πόσα στοιχεία έχουν εισαχθεί στην ουρά προσδιορίζεται από την μεταβλητή `m_count`.

Τύποι Μηνυμάτων:

Οι τύποι μηνυμάτων που ανταλλάσσονται είναι τρεις :

SORT: ο τύπος αυτός μηνύματος υποδηλώνει στο πρόγραμμα να γίνει ταξινόμηση των στοιχείων του πίνακα που ξεκινάει από την θέση `Low` και τελειώνει στην θέση `High`

SORTED: Παρόμοιο μήνυμα με αυτό της `SORT` με την διαφορά ότι τα στοιχεία αυτά ταξινομήθηκαν.

SHUTDOWN: μήνυμα για τον τερματισμό των thread pools

THREAD POOLS

Ο αριθμός προσδιορίζεται από την σταθερά THREADS. Κάθε thread εκτελεί την συνάρτηση thread_func η οποία τροφοδοτείται με τον πίνακα arr ο οποίος πρέπει να ταξινομηθεί. Στην συνέχεια επαναληπτικά γίνεται εξαγωγή των μηνυμάτων. Τα μηνύματα για τα οποία εκτελεί κάποιο είδος επεξεργασίας είναι αυτά που έχουν τύπο ταξινόμησης. Στην περίπτωση αυτή γίνεται έλεγχος για το μέγεθος άμα είναι κάτω από το όριο του LIMIT και άμα είναι τότε γίνεται εκτέλεση της Insertion_sort. Ειδικά γίνεται κλήση στην συνάρτηση partition και με βάση αυτήν στέλνονται δύο μηνύματα για το αριστερό και το δεξί υπομέρος το οποίο οροθετείται από την θέση του partition. Σε περίπτωση που η thread_func πάρει τον τύπο μηνύματος SORTED από sort απλά ξαναστέλνει το μήνυμα που αφαίρεσε από την ουρά με σκοπό να την αναλάβει η main. Αυτή η συνάρτηση εκτελείται επαναληπτικά μέχρι το επόμενο μήνυμα να είναι τύπου shutdown, τότε σε αυτήν την περίπτωση γίνεται έξοδος του thread αυτού και στέλνεται ένα ακόμα μήνυμα τερματισμού προκειμένου να τον πάρει το επόμενο thread για να τερματίσει και αυτό.

Main Συνάρτηση:

Αρχικά στην συνάρτηση της main γίνεται μια δυναμική δέσμευση ενός πίνακα στο heap με χρήση της συνάρτησης malloc, στην συνέχεια γίνεται αρχικοποίηση του με τυχαίους αριθμούς, αρχικοποίηση ενός pool threads με αριθμό THREADS νήματα. Σε περίπτωση αποτυχίας γίνεται αποδέσμευση της μνήμης που δεσμεύτηκε. Μετά στέλνεται το πρώτο μήνυμα ταξινόμησης με δείκτες από 0 έως N. Η συνάρτηση main στην συνέχεια εκτελείται επαναληπτικά μέχρι τα εύρη που θα λάβει με μηνύματα από έως που έχουν ταξινομηθεί να συμπληρώσει το μέγεθος του πίνακα. Σε περίπτωση που λάβει μηνύματα άλλων τύπων εκτός από SORTED τα ξαναεισάγει στην ουρά για να τα διαχειριστεί η thread_func. Μετέπειτα αφού έχει ολοκληρωθεί η διαδικασία ταξινόμησης των στοιχείων στέλνεται μήνυμα τερματισμού το οποίο θα οδηγήσει στον τερματισμό των threads. Τελικά γίνεται επαλήθευση για να είμαστε σίγουροι ότι η ταξινόμησή είναι σωστή καθώς και απελευθέρωση όλων των απαραίτητων πόρων δηλαδή τα mutexes, conditional variables και ο δεσμευμένος πίνακας στο heap.