Αναφορά εργασίας 2 Παράλληλος προγραμματισμός 2017-2018.

Βασίλης Χαρίσης Π2013013

Αρχικά, θα πρέπει να αναφέρω ότι με δυσκόλεψε αρκετά η εργασία 2.Την 1 δυστυχώς δεν την έκανα δεν πρόλαβα την προθεσμία.Θα ήθελα να ρωτήσω εδώ, αν θα μετρήσει καθόλου αυτή η προσπάθεια στο βαθμό του μαθήματος όταν θα έρθω για γραπτή εξέταση τον Ιούνιο.

Επείδη μερικά κομμάτια δεν μου βγαίναν στον compiler δεν τα πρόσθεσα στην main() για να τρέχει τουλάχιστον το πρόγραμμα.Μια περίληψη του προγράμματος θα ήταν ότι δέχεται από τον χρήστη το μήκος της ουράς εργασιών που θέλει ο ίδιος και το μέγιστο όριο από την συνάρτηση rand()που επιθυμεί για να γεμίσει η qeueue με τυχαία integer στοιχεία,που με την σειρά τους, στο πρόγραμμα δημιουργούνται 4 threads που τα 2 από αυτά βγάζουν από την qeueue στοιχεία, απελευθαιρώνοντας και τον εκάστοτε χώρο,τα άλλα 2 thread ταξινομούν τα στοιχεία που παίρνουν τα πρώτα 2.Το thread 3 παίρνει το left και το thread 4 το right.Στο τέλος στον πίνακα arr βλέπουμε τα στοιχεία ταξινομημένα.Κλείνουμε τα threads.

Αναλυτικά μέσα στον κώδικα quickshort.c

1. Δήλωση μεταβλητών-συναρτήσεων

Ορίζω την welcomeMessage(*nptr) η οποία δέχεται έναν pointer για την μεταβλητή n που της ζητείται στην main.Το n global variable αντιπροσοπεύει το μήκος της ουράς.Ο λόγος που δεν επέλεξα να δεχθεί μια άλλη μεταβλητή και μετά να την περάσω στην n είναι ότι η τοπική μεταβλητή που θα δημιουργούσα θα παρέμενε και μετά το τέλος της συνάρτησης welcomeMessage. Μέσα στην συνάρτηση ορίζεται και η maxrand ή μέγιστη τιμή που βάλει η συνάρτηση rand() στην ουρά.

Fillitem() Γεμίζει integer μεταβλητές με τυχαίους αριθμούς σύμφωνα με την maxrand του χρήστη.

typedef struct cell{} Ο κόμβος της ουράς.Περιέχει έναν int και μια μεταβλητή δείκτη για τον επόμενο κόμβο που θα δείξει.

Typedef struct cell queue{} Η queue είναι μια μια δομή που περιέχει stuct cell δόμες.

Typedef queue *queueptr Δείκτης που δείχνει την queue

Void enqueue Φτίαχνει την ουρά.Δύο δέικτες queuptr ενας για την header της ουρας και ένας για την tail της ουράς και ένα int value για την τίμη που θα περιέχει η ούρα.

Int arr[Maxsize]; Ο Πίνακας που θα γίνει το quickshort και η συναλάγη μεταξύ των threads.Προσπάθησα να βάλω το n της ουράς ως maxsize αλλά αντιμετώπισα πρόβλημα στον compiling.Για αυτό τον λόγο έβαλα έναν σχετικά ψηλό όριο.Μπορεί να αλλάξει βέβαια.

Void swap() αλλάζει τις τιμές στον arr.Δεν χρησιμοποίησα την τιμή του κάθε arr[i] στις αλλαγές για λόγους μνήμης,ταχύτητας,και δεν ήξερα να θα χανόταν η μεταβλητή που θα περιέχει κατά την μεταφορά.

Typedef stuct task_data{}

Δομή με τα στοιχεία που θα περιέχει ο thread όπως πχ.id

Typedef struct thread_data

{μηνύματα μεταξύ των thread.Περιέχει και semaphore για συχρονιμσό των μηνυμάτων.

}

Thread_data * thread_pool[4] δήλωση της pool με 4 thread

Δ΄ήλωση mutex kai flags για την ούρα των threads(Δεν τα έβαλα στην εργασία είχα πρόβλημα με compiling)

Jobpool Ουρά για την ανάθεση εργασιών.

==========

Main(){

Κώδικας

}

==========

WELCOMEMESSAGE()

{

{ζητάει το length και το maxrand

}

==========

Enqueue{

Δήλωση newptr ως queueptr

newptr=malloc(sizeof(queue))

Δημιουργία του κόμβου διεύθυνσης της μνήμης στο newptr

Αν headerptr==0

Θέσε το στο newptr

Αλλίως αφού δεν είναι 0 το header

Θέσε στο cellptr to 0

Και tailptr το newptr

}

---

printQueue{

εμφανίζει την queue

}

Quickshort{

Ανθέτει στον πίνακα arr

Βάζει για pivot το πρώτο στοίχειο του πίνακα και κάνει τις απαράιτητες συγκρίσεις.

}