Αναφορά ΗΜΥ316

**Πανεπιστήμιο Κύπρου**

**Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών**

**Assignment 8** ( Due 31/03/2022 )

Ονόμα και Ταυτότητα :

**Εντουίνα Κάρουλλα 1042364**

Ομάδα : 7

Φοίβος Λύμπουρας                        1016477

Στέλιος Καραγίωργης                     1021340

Θεοδόσιος Ιωάννου                        1020844

**Άσκηση 1**

Στην πρώτη άσκηση έπρεπε να γράψουμε 3 pthreads τα οποία να τυπώνουν στην οθόνη συγχρονισμένα ένα μήνυμα που να περιέχει την ταυτότητα του νήματος και την τιμή ενός μετρητή ο οποίος αυξάνεται κάθε φορά κατά ένα.

Για τον σκοπό αυτό δημιουργήσαμε 3 pthreads και 3 functions (ένα για κάθε pthread). Το κάθε pthread στέλνει στην ανάλογη συνάρτηση το μήνυμα που θέλει να τυπώσει.

Για τον συγχρονισμό και για το τύπωμα της κάθε συνάρτησης με την σειρά στην οποία μας ζητήθηκε χρησιμοποιήσαμε mutex lock και cond\_wait, cond\_signal. To mutex lock εμποδίζει τα άλλα pthreads να τρέχουν ταυτόχρονα.

Για να σταματά το κάθε thread έπειτα από την τύπωση του μηνύματος χρησιμοποιήσαμε μια μεταβλητή count που παίρνει τις τιμές 1,2,3 ανάλογα με το thread στο οποίο θέλουμε να τυπώσουμε. Δηλαδή το count ξεκινά με την τιμή μηδέν και όταν καλείται η πρώτη συνάρτηση για να τυπωθεί το πρώτο thread (Tread 0) όταν η μεταβλητή γίνει ίση με 1 κάνει pthread\_cond\_wait(&cond1, &mutex1) και προχωρά στην επόμενη συνάρτηση με την χρήση της pthread\_cond\_signal(&cond2). Η συνάρτηση pthread\_cond\_signal είναι αυτή που θα ορίσει ποιο thread θα τυπωθεί στη συνέχεια.

A screenshot of a computer

Description automatically generated with low confidence**Αποτελέσματα:**

**Άσκηση 2**

Στην δεύτερη άσκηση μας ζητήθηκε να γράψουμε ένα χρονοδρομολογητή (scheduler) Feedback με κάποια χαρακτηριστικά.

Οι λειτουργία του χρονοδρομολογητή Round-Robin είναι η εξής επιλέγει ένα διάστημα q(quantum) και εκτελεί κάθε διεργασία για q χρονικές μονάδες και στη συνέχεια εναλλάσσει διεργασία. Τα πλεονεκτήματα αυτής της πολιτικής είναι ότι είναι απλή, είναι δίκαιη, καμιά διεργασία δεν υποφέρει από παρατεταμένη στέρηση και παρουσιάζει καλή απόκριση για μικρές διεργασίες (μικρές διεργασίες δεν περιμένουν πολύ πίσω από μεγάλες διεργασίες). Τα μειονεκτήματα αυτής της πολιτικής είναι ότι πιθανόν να έχει μειωμένη διεκπεραιωτική ικανότητα εάν το q είναι πολύ μικρό και δίνει προτεραιότητα στις διεργασίας με περισσότερους υπολογισμούς παρά σε διεργασίες με Ε/Ε.

Για την υλοποίηση της άσκησης 1 δημιουργήσαμε συνολικά 5 διεργασίες και ένα χρονοδρομολογητή.

Για την κάθε διεργασία χρησιμοποιήσαμε τις εξής εντολές:

**Fflush(stdout):** καθαρίζει το output buffer για να μπορέσουν να εκτυπωθούν τα μηνύματα

**Execl:** εκτέλεση μιας διεργασίας παιδί ως ξεχωριστό πρόγραμμα

**Atof(argc[1]):** για να δημιουργήσουμε το argv[1] από atoi σε float

**Signal(SIGCONT, signal\_cont\_handler):** δηλώνει το signal SIGCONT στο signal\_cont\_handler

**Usleep():** χρησιμοποιείται για καθυστέρηση

Για την κάθε διεργασία χρησιμοποιήσαμε τα εξής σήματα:

**SIGTSTP** για διακοπή της διεργασίας ενεργοποιώντας τον stop-handler

**SIGCONT** για επανεκκίνηση της διεργασίας ενεργοποιώντας τον signal\_condition

Για την κάθε διεργασία χρησιμοποιήσαμε τα εξής arguments:

Το πρώτο είναι το start time

Το δεύτερο είναι το execution time

Για τον χρονοδρομολογητή χρησιμοποιήσαμε τις εξής εντολές:

**Gettimeofday()** για την μέτρηση του χρόνου που πέρασε από την έναρξη της λειτουργίας του χρονοπρογραμματιστή

**atof(argv[1])** για να δημιουργήσουμε το argv[1] από atoi σε float

Text

Description automatically generated

Text

Description automatically generated Text

Description automatically generated

**Άσκηση 3**

Ο αλγόριθμος του τραπεζίτη λειτουργεί ως εξής: τα δημιουργημένα νήματα ζητούν πόρους και ελευθερώνουν πόρους με τυχαίο τρόπο. Έτσι κάθε δημιουργημένο νήμα ενημερώνει τον τραπεζίτη για τοn μέγιστο αριθμό πόρων που μπορεί να ζητήσει και τον αριθμό πόρων που χρειάζεται. Με αυτό τον τρόπο ο τραπεζίτης αποφασίζει αν θα αναθέσει πόρους ή όχι στο συγκεκριμένο νήμα, έτσι ώστε το σύστημα πάντα να παραμένει σε ασφαλή κατάσταση και να μην βρίσκετε σε αδιέξοδο. Αν ο τραπεζίτης δεν μπορεί να αναθέσει τους πόρους που ζήτησε ένα νήμα, τότε το νήμα μπαίνει σε κατάσταση αναμονής, μέχρι κάποιο άλλο νήμα να ελευθερώσει πόρους και να μπορεί ο τραπεζίτης να δώσει τους πόρους που ζήτησε το νήμα προηγουμένως, εφόσον δεν θα είναι τώρα σε ασφαλή κατάσταση.

Text

Description automatically generated

