

ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ

Εργασία 1^η

ΕΥΑΓΓΕΛΟΣ-ΑΝΔΡΕΑΣ ΒΕΝΕΤΣΑΝΟΣ 3180019

ΔΗΜΗΤΡΙΟΣ ΔΕΛΗΚΟΥΡΑΣ 3170034

Ανάλυση των 3 αρχείων του project:

Αρχείο .h

Σε αυτό το αρχείο γίνονται όλα τα απαραίτητα “include”, καθώς και οι αρχικοποιήσεις των αναγκαιών σταθερών.

Αρχείο .c

Περιέχει όλες τις μεθόδους που υλοποιούν το πρόγραμμα.

Main

Η main μέθοδος δέχεται τις παραμέτρους nCust και seed. Το πλήθος των πελατών προς εξυπηρέτηση και τον τυχαίο σπόρο για τη γεννήτρια των τυχαίων αριθμών αντίστοιχα. Έπειτα από τους κατάλληλους ελέγχους για την εγκυρότητα των παραμέτρων αυτών γίνεται memory allocation του πίνακα threads, ο οποίος περιέχει στοιχεία τύπου pthread_t. Ακόμα, γίνονται οι απαραίτητοι έλεγχοι για την σωστή αρχικοποίηση των μεταβλητών τύπου pthread_mutex_t.

Η δημιουργία των νημάτων πραγματοποιείται μέσα σε ένα for loop από την pthread_create. Επιπλέον, σε κάθε επανάληψη υπολογίζονται δυο “τυχαίες” τιμές μέσω της συνάρτησης rand_r, ο αριθμός από πίτσες της παραγγελίας και ο χρόνος μέχρι να πραγματοποιηθεί η επόμενη παραγγελία. Η δεύτερη τιμή δεν υπολογίζεται για την τελευταία παραγγελία. Τα νήματα εκτελούν την συνάρτηση order η οποία προετοιμάζει την συγκεκριμένη παραγγελία και ως παράμετρο δέχεται τον αριθμό από πίτσες.

Στην συνέχεια, με ένα δεύτερο for loop μεταφέρεται ο έλεγχος σταδιακά σε όλα τα νήματα. Με αυτό τον τρόπο υπολογίζεται ο μέγιστος καθώς και ο μέσος χρόνος ολοκλήρωσης παραγγελίας. Αυτό επιτυγχάνεται μέσω της pthread_join.

Τέλος εκτυπώνονται ο μέγιστος καθώς και ο μέσος χρόνος και το πρόγραμμα τερματίζεται.

Order

Η συνάρτηση order είναι υπεύθυνη για την προετοιμασία των παραγγελιών. Ως παράμετρο δέχεται το πλήθος από πίτσες της εκάστοτε παραγγελίας.

Για τον σωστό υπολογισμό του χρόνου προετοιμασίας της παραγγελίας, ο οποίος αποθηκεύεται στην μεταβλητή latency, χρησιμοποιείται η συνάρτηση clock_gettime. Η κλήση της γίνεται έπειτα από κλείδωμα mutex με την μεταβλητή uLatency τύπου pthread_mutex_t.

Ακόμα, γίνεται σύνδεση της κάθε παραγγελίας με ένα μοναδικό αριθμό ταυτότητας oid.

Για την “προετοιμασία” της παραγγελίας χρησιμοποιείται η συνάρτηση sleep, με παράμετρο αρχικά τον χρόνο προετοιμασίας και στην συνέχεια τον χρόνο ψησίματος της παραγγελίας. Πριν την κλήση της sleep και στις δυο περιπτώσεις θα πρέπει να πραγματοποιηθεί κλείδωμα mutex με τις μεταβλητές uCooks και uOvens τύπου pthread_mutex_t αντίστοιχα.

Σε αυτό το σημείο χρησιμοποιούνται και οι συναρτήσεις acquire_cook, acquire_oven, release_cook, release_oven. Με αυτό τον τρόπο πραγματοποιείται ο σωστός υπολογισμός των διαθεσίμων παρασκευαστών και φούρνων.

Μετά την “ολοκλήρωση” της παραγγελίας υπολογίζεται ο χρόνος προετοιμασίας της και εκτυπώνεται ένα μήνυμα τύπου:

Η παραγγελία με αριθμό <oid>, για <pizzas> πίτσες, ετοιμάστηκε σε <latency> λεπτά.

Τέλος η συνάρτηση order επιστρέφει τον χρόνο προετοιμασίας latency πίσω στην main με σκοπό τον υπολογισμό του μέγιστου και μέσου χρόνου προετοιμασίας ολοκλήρωσης παραγγελίας.

Μέθοδοι acquire και release

acquire_cook, acquire_oven, release_cook, release_oven

Οι παραπάνω συναρτήσεις χρησιμοποιούνται για τον σωστό υπολογισμό των διαθεσίμων κοινών πόρων. Αυτοί οι πόροι είναι οι παρασκευαστές και οι φούρνοι. Οι μέθοδοι acquire μειώνουν τους διαθέσιμους πόρους ενώ οι release τους απελευθερώνουν.

Όταν δεν υπάρχει κάποιος διαθέσιμος πόρος, μέσω ενός while loop, περιμένουμε μέχρι να απελευθερωθεί ο αντίστοιχος πόρος.

Αρχείο .sh

Αρχικά αποθηκεύεται το path στο οποίο βρίσκονται όλα τα αρχεία του project στην μεταβλητή DIR και μέσω της εντολής cd μεταφερόμαστε σε αυτό το directory.

Στην συνέχεια γίνεται compile το αρχείο πηγαίου κώδικα, δημιουργώντας ένα εκτελέσιμο αρχείο ίδιου ονόματος.

Τέλος, γίνεται εκτέλεση αυτού του αρχείου για πλήθος πελατών ίσο με 100 και γεννήτρια των τυχαίων αριθμών ίση με 1000.

