

Εργασία εξαμήνου στα λειτουργικά συστήματα  
Βρουσάι Νγκελίνα – p3210024  
Ρούση Αικατερίνη – t8210130 (ΔΕΤ)

Το πρόγραμμα υλοποιεί τις βασικές λειτουργίες μιας πιτσαρίας.

Αρχικά, το κυρίως πρόγραμμα (main), ελέγχει αν το input που δόθηκε από το χρήστη είναι το αναμενόμενο. Αν όχι, εμφανίζει κατάλληλο ειδοποιητικό μήνυμα και τερματίζει.

Αν τα ορίσματα είναι σωστά, αυτά μετατρέπονται σε integers και αποθηκεύονται στις αντίστοιχες μεταβλητές. Δημιουργείται πίνακας για την αποθήκευση των threads και μεταβλητή για να καταγράφει τα return codes.

Στη συνέχεια αρχικοποιούνται όλα τα mutexes και οι condition variables με τους κατάλληλους ελέγχους και, με τη χρήση ενός for loop, γίνεται η αρχικοποίηση των νημάτων με τις κατάλληλες παραμέτρους και η κλήση της βασικής λειτουργικής συνάρτησης του προγράμματος που διαχειρίζεται τα threads (manage\_order).

Η συνάρτηση εκκινεί ανανεώνοντας το πλήθος των πελατών και αρχικοποιώντας ορισμένες μεταβλητές. Καταγράφει τον χρόνο εκκίνησης της παραγγελίας και ξεκινά να τρέχει τη διαδικασία στη φάση της τηλεφωνικής επικοινωνίας. Εκεί, αφού αποφασιστεί με τη χρήση της rand\_r (και μέσα στα καθορισμένα όρια) το πόσες πίτσες επιλέγει ο πελάτης, κλειδώνει το mutex του τηλεφωνητή και περιμένει να βρεθεί κάποιος διαθέσιμος με χρήση condition variable. Με το που γίνει διαθέσιμος κάποιος τηλεφωνητής (av\_tel), το mutex ξεκλειδώνεται και πάλι με χρήση πιθανοτήτων ο πελάτης επιλέγει είδος και ποσότητες από κάθε πίτσα.

Στη συνέχεια υπολογίζεται ο χρόνος διεκπεραίωσης της πληρωμής και δηλώνεται fail με χρήση τυχαίας μεταβλητής is\_failure.

Αν αποτύχει όντως η παραγγελία, εμφανίζεται κατάλληλο μήνυμα στο χρήστη (κλειδώνεται το Mutex που διαχειρίζεται τα μηνύματα), και αυξάνονται οι διαθέσιμοι τηλεφωνητές και ο counter για τις αποτυχημένες παραγγελίες (πάλι με χρήση mutex και condition variable). Το thread τερματίζει.

Αν τώρα η παραγγελία πετύχει, αυξάνεται το συνολικό κέρδος με βάση τα επιμέρους κόστη με κλείδωμα και ξεκλείδωμα του mutex που διαχειρίζεται το revenue. Εμφανίζεται μήνυμα στο χρήστη πως η παραγγελία πέτυχε, απελευθερώνεται ένας τηλεφωνητής και το πρόγραμμα συνεχίζει στη φάση του μαγειρέματος.

Εδώ, το πρόγραμμα ελέγχει αν υπάρχουν διαθέσιμοι παρασκευαστές, και αν όχι, περιμένει μέχρι να ισχύει η συνθήκη.

Μόλις γίνει αυτό, το thread περιμένει για χρόνο ίσο με tprep\*αριθμόΠίτσες και στη συνέχεια προχωράει να ελέγξει αν υπάρχουν διαθέσιμοι φούρνοι. Αν όχι, περιμένει να απελευθερωθούν (τόσοι ώστε να χωρούν οι πίτσες), απελευθερώνει τον παρασκευαστή (αφού βάζει τις πίτσες για ψήσιμο) και κοιμάται μέχρι αυτές να ψηθούν (χρήση mutexOven και mutexCook).

Αφού τελειώσει ο χρόνος ψησίματος, τη διαδικασία αναλαμβάνει η λειτουργία του delivery. Η condition variable ελέγχει αν υπάρχουν διαθέσιμοι deliverers, και μόλις βρεί, απελευθερώνει τον φούρνο και κοιμάται για το χρόνο που διαρκεί το πακετάρισμα.

Στη συνέχεια, υπολογίζεται πάλι μέσω της rand\_r ο χρόνος delivery και το thread κοιμάται στη διάρκειά του. Αμέσως μετά καταγράφεται το end\_time της παραγγελίας και υπολογίζονται ο χρόνος που διήρκεσε και ο χρόνος που έκανε για να κρυώσει (elapsed\_total, cooling\_time).

Εμφανίζεται μήνυμα στο χρήστη, πως η παραγγελία με το συγκεκριμένο id παραδόθηκε σε τόσο χρόνο (mutexMess), και στη συνέχεια με χρήση mutexDel απελευθερώνεται ο διανομέας. Παρακάτω υπολογίζονται ορισμένες τιμές που θα γίνουν έξοδοι στο τελικό πρόγραμμα και μετά το thread (και άρα η παραγγελία) τερματίζουν.

Μόλις επιστρέψουμε στο κυρίως πρόγραμμα, τα threads που χρησιμοποιήσαμε γίνονται join και εμφανίζονται τα απαραίτητα στατιστικά στην οθόνη του χρήστη (αθροίσματα και μο). Τα Mutexes και οι condition variables καταστρέφονται και το πρόγραμμα τερματίζει.