

**ΟΙΚΟΝΟΜΙΚΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΑΘΗΝΩΝ**



ATHENS UNIVERSITY
OF ECONOMICS
AND BUSINESS

Λειτουργικά Συστήματα

Ακαδ. Έτος 2021-2022, Εαρινό Εξάμηνο

Προγραμματιστική Εργασία (30%)



Όνομα	Αριθμός Μητρώου	E- mail
Νεφέλη Ελένη Πολιού	3200243	p3200243@aueb.gr
Μιχάλης Χασάπης	3200218	p3200218@aueb.gr

Δομή Κώδικα Εργασίας

Αρχικά στις πέντε (5) πρώτες γραμμές του κώδικα περιλαμβάνουμε (include) τα απαραίτητα header files. Επιπλέον, σε μια global μεταβλητή αποθηκεύουμε το seed που θα χρησιμοποιηθεί για την συνάρτηση rand_r. Στις επόμενες σειρές δημιουργούμε μεταβλητές τύπου timespec για την clock_gettime, pthread conditions για τους τηλεφωνητές και τους ταμίες (availability_caller, availability_cashier) και τα mutexes του προγράμματος (cashiers_mutex, caller_mutex, seats_mutex, timetofinm, timetowaitm, percentm, accm). Ακόμα, φτιάχνουμε μεταβλητές για τους πελάτες που εξυπηρετούνται από τηλεφωνητές και ταμίες (slot, slot2). Τέλος, κατασκευάζουμε έναν διδιάστατο πίνακα που περιέχει το πλάνο του θεάτρου και αρχικοποιούμε τις μεταβλητές για τα ποσοστά ολοκλήρωσης και χρόνου αναμονής (success_rate, payment_fail_rate, unavailable_seats_rate, time_of_wait, time_of_finish).

Στην σειρά 32 αρχίζει η μέθοδος με ονομασία Buying_phase, η οποία υλοποιεί την κράτηση και αγορά θέσεων ενός πελάτη. Ειδικότερα, αποθηκεύουμε την ταυτότητα του πελάτη (id), την χρονική στιγμή, κλειδώνουμε το mutex των τηλεφωνητών και ελέγχουμε την διαθεσιμότητα τους. Σε περίπτωση που όλοι οι τηλεφωνητές είναι απασχολημένοι περιμένουμε ώστε να λάβουμε σήμα(availability_caller). Διαφορετικά επιλέγεται μια τυχαία ζώνη μεταξύ των «Α» και «Β». Έπειτα, κλειδώνουμε το mutex του πίνακα των θέσεων, επιλέγει ο πελάτης τυχαίο αριθμό θέσεων και αναμένει μέχρι να πάρει απάντηση από τον τηλεφωνητή (sleep). Αυξάνουμε την μεταβλητή χρόνου αναμονής και ξεκινάει η επιλογή των θέσεων. Αναζητούμε ανά σειρά την ύπαρξη των συνεχόμενων θέσεων που ζήτησε ο πελάτης. Στο ενδεχόμενο μη διαθεσιμότητας τους αυξάνουμε την μεταβλητή απόρριψης λόγω θέσεων και τελειώνει το thread, αλλιώς ξεκλειδώνουμε το mutex του τηλεφωνητή και των θέσεων και δίνουμε σήμα. Ομοίως προχωράμε την διαδικασία με τους ταμίες. Κλειδώνουμε το mutex των ταμιών, ελέγχουμε την διαθεσιμότητα τους, εάν δεν είναι εφικτό περιμένουμε να ληφθεί το κατάλληλο σήμα (availability_cashier). Σε περίπτωση ελεύθερου ταμιά περιμένουμε μέχρι να ανταποκριθεί. Ακολουθώντας με την βοήθεια της rand_r υπολογίζεται η πιθανότητα να γίνει δεκτή η πληρωμή. Αν γίνει αποδεκτή ο πελάτης πληρώνει, το ποσό προστίθεται στον λογαριασμό της εταιρίας τυπώνοντας το κατάλληλο μήνυμα για τις θέσεις που κλείστηκαν και το κόστος τους, ενώ ταυτόχρονα αυξάνεται το ποσοστό επιτυχίας. Από την άλλη πλευρά εάν δεν γίνει αποδεκτή η συναλλαγή οι θέσεις γίνονται πάλι διαθέσιμες αυξάνοντας

το ποσοστό απόρριψης λόγω πληρωμής. Ξεκλειδώνουμε τα mutexes θέσεων και ταμίων και τελειώνει το thread.

Εν συνεχεία ακολουθεί η μέθοδος Start_Buying (γραμμή 241), όπου σώζουμε τον αριθμό πελατών (count), δημιουργούμε πίνακα με τα id αυτών (count_cust) και δεσμεύουμε μνήμη για τα threads των πελατών. Πιο αναλυτικά, για κάθε πελάτη εκτός του πρώτου περιμένουμε (sleep) τυχαίο αριθμό δευτερολέπτων και κατασκευάζουμε τα νήματα τους και τα κάνουμε join. Ακόμα, τυπώνουμε τα id των πελατών, τις θέσεις που δεσμεύονται, καθώς και τα συνολικά έσοδα, όλα τα ποσοστά και τους χρόνους αναμονής. Τέλος, ελευθερώνουμε την μνήμη που έχουμε δεσμεύσει.

Στις τελευταίες σειρές βρίσκεται η μέθοδος main, κατά την οποία ελέγχουμε αν είναι σωστός ο αριθμός των ορισμάτων, αποθηκεύουμε στην global μεταβλητή το seed για το rand_r και αρχικοποιούμε όλα τα mutexes και conditions που χρησιμοποιούμε. Δημιουργούμε το αρχικό thread, το κάνουμε join και καταστρέφουμε όλα τα mutexes και conditions.

Τέλος θέλουμε να ενημερώσουμε ότι δοκιμάσαμε διάφορους τρόπους (με διαφορετικούς mutexes στον πίνακα θέσεων για ταμίες και τηλεφωνητές, χωρίς mutexes στους χρόνους) και καταλήξαμε πως πιο λογικό και ορθό αποτέλεσμα προκύπτει από αυτή την υλοποίηση (mutexes στις μεταβλητές χρόνου και κοινό mutex πίνακα θέσεων για ταμίες και τηλεφωνητές) Με αυτό τον τρόπο υπάρχει η λογική συσσώρευση πελατών στους ταμίες .