**ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ**

**ΕΡΓΑΣΙΑ 2η**

Πάνος Φώτιος – 3180141

Παπαχρήστου Ιωάννης – 3180150

Πρόφης Ιωάννης – 3180161

Αρχικά δηλώνουμε τα mutex και τα conditions που θα χρησιμοποιήσουμε μέσα στον κώδικα. Χρησιμοποιήσαμε mutexes και τα αντίστοιχα conditions για τους παρασκευαστές, τους φούρνους, τους διανομείς, την εκτύπωση του μηνύματος στην οθόνη, τον συνολικό χρόνο αναμονής, τον συνολικό χρόνο κρυώματος, τον μέγιστο χρόνο αναμονής και τον μέγιστο χρόνο κρυώματος.

Στη συνέχεια δηλώνουμε τις μεταβλητές όπου θα αποθηκεύουμε τα δεδομένα για τον αριθμό των διαθέσιμων παρασκευαστών, των διαθέσιμων φούρνων, των διαθέσιμων διανομέων, τον συνολικό χρόνο αναμονής και κρυώματος που θα μας βοηθήσει για τον υπολογισμό του μέσου όρου, τον μέγιστο χρόνο αναμονής, τον μέγιστο χρόνο κρυώματος και τον σπόρο της γεννήτριας τυχαίων αριθμών.

Ξεκινάμε δηλώνοντας τη συνάρτηση successful\_mutex\_action. Δεχόμενη σαν όρισμα το response\_code που επιστρέφεται μετά από το κλείδωμα ή το ξεκλείδωμα ενός mutex, ελέγχει αν η λειτουργία πραγματοποιήθηκε όπως αναμενόταν. Διαφορετικά τερματίζει την εκτέλεση του προγράμματος επιστρέφοντας τον κωδικό σφάλματος. Η ιδέα για τη χρήση της συνάρτησης αυτής προέκυψε μετά από έρευνα στο διαδίκτυο σχετικά με τη χρήση mutexes και την υιοθετήσαμε καθώς θεωρούμε ότι αποτελεί καλή τεχνική.

Συνεχίζουμε υλοποιώντας τη συνάρτηση order, που είναι και η βασική συνάρτηση του προγράμματός μας. Αφού κάνουμε τις απαραίτητες δηλώσεις ξεκινάμε την καταγραφή του χρόνου χρησιμοποιώντας τη συνάρτηση clock\_gettime. Έπειτα, με κλείδωμα του κατάλληλου mutex ελέγχουμε αν υπάρχει διαθέσιμος παρασκευαστής να αναλάβει την τρέχουσα παραγγελία. Αν όχι εκτυπώνουμε μήνυμα αναμονής και περιμένουμε μέχρι κάποιος παρασκευαστής να γίνει διαθέσιμος. Όταν υπάρχει διαθέσιμος παρασκευαστής εκτυπώνουμε ένα αντίστοιχο μήνυμα, ενημερώνουμε τον αριθμό διαθέσιμων παρασκευαστών και ξεκλειδώνουμε το mutex. Με χρήση της rand\_r παίρνουμε έναν τυχαίο αριθμό από πίτσες που θα αποτελέσει την παραγγελία, περιμένουμε όσο χρόνο χρειαστεί για την προετοιμασία της παραγγελίας και στη συνέχεια πάλι κλειδώνουμε το κατάλληλο mutex, εκτυπώνουμε τα αντίστοιχα μηνύματα και όταν υπάρξει διαθέσιμος φούρνος ενημερώνουμε τη μεταβλητή, απελευθερώνουμε το mutex και αντίστοιχα αποδεσμεύουμε τον παρασκευαστή ενώ περιμένουμε να ψηθούν οι πίτσες τις παραγγελίας. Όταν γίνει αυτό, περιμένουμε μέχρι κάποιος διανομέας να γίνει διαθέσιμος και να παραλάβει την πίτσα, με τα κατάλληλα signals και ενημέρωση των μεταβλητών αποδεσμεύουμε τον φούρνο και δεσμεύουμε τον διανομέα. Όταν ο διανομέας παραδώσει την παραγγελία τερματίζουμε το χρονόμετρο και περιμένουμε να επιστρέψει στο κατάστημα για να ενημερώσουμε το πλήθος των διαθέσιμων διανομέων. Με κατάλληλη χρήση των mutexes εκτυπώνουμε στην οθόνη το ζητούμενο μήνυμα ολοκλήρωσης της παραγγελίας, ενημερώνουμε τον συνολικό χρόνο αναμονής, το συνολικό χρόνο κρυώματος, τον μέγιστο χρόνο αναμονής, τον μέγιστο χρόνο κρυώματος και τερματίζουμε το thread.

Τέλος υλοποιήσαμε την main μέθοδο του προγράμματός μας. Αφού κάνουμε τους απαραίτητους ελέγχους για την ορθότητα των δοθέντων παραμέτρων και αρχικοποιήσουμε κατάλληλα τις μεταβλητές μας, περνάμε στη δημιουργία των threads για κάθε νέα εισερχόμενη παραγγελία η οποία έρχεται μετά από τυχαίο χρόνο βασισμένη στη γεννήτρια rand\_r. Αφού ολοκληρωθούν όλα τα νήματα εκτυπώνουμε τα ζητούμενα δεδομένα για τον μέσο και τον μέγιστο χρόνο αναμονής, καταστρέφουμε όλα τα mutexes και τα conditions και τερματίζουμε το πρόγραμμά μας.