



2^η Ατομική Εργασία

Θέμα - Επίλυση προβλήματος παραγωγού - καταναλωτή

Όπως έχουμε ήδη παρατηρήσει στις πολυνηματικές εφαρμογές, τα νήματα διαμοιράζονται την ίδια περιοχή μνήμης. Αυτός ο μηχανισμός είναι σημαντικά αποδοτικός αλλά έχει ως αποτέλεσμα κάποιες φορές την εμφάνιση συνθηκών ανταγωνισμού (race conditions). Επίσης τα νήματα καλούνται να συντονίζονται (συγχρονίζονται) την εκτέλεση τους. Πολλές φορές καλείται ένα νήμα να αναστείλει την εκτέλεση του μέχρι να ικανοποιηθεί κάποια συνθήκη που είναι απαραίτητη για την εκτέλεση των λειτουργιών του. Όλα τα παραπάνω αντιμετωπίζονται με μηχανισμούς που παρέχονται από την Java όπως είναι ο συγχρονισμός (synchronized μέθοδοι και blocks) αλλά και η χρήση των μεθόδων **wait()** και **notify()** της κλάσης Object.

Στα πλαίσια της εργασίας αυτής καλείστε να υλοποιήσετε ένα κατανεμημένο σύστημα που να προσομοιώνει την λειτουργία ενός εστιατορίου κατά την οποία πολλές διαδικασίες γίνονται παράλληλα και εμφανίζονται συχνά θέματα συγχρονισμού. Στο εστιατόριο αυτό υπάρχουν 3 chef, ο καθένας από τους οποίους φτιάχνει και από ένα διαφορετικό είδος φαγητού. Ο 1^{ος} chef ασχολείται κυρίως με το ψήσιμο ψαριών στα κάρβουνα, ο 2^{ος} chef με τη δημιουργία θαλασσινών μεζέδων και ο 3^{ος} chef ασχολείται με ποικιλίες κρεατικών. Οι πελάτες του εστιατορίου μπορούν να ζητήσουν **ταυτόχρονα** κάποιες μερίδες από ένα από αυτά τα είδη.

Η εφαρμογή που θα υλοποιήσετε αποτελείται από έναν «εξυπηρετητή» που διαθέτει τρεις διαφορετικές θέσεις αποθήκευσης. Αυτές οι θέσεις χρησιμοποιούνται για την αποθήκευση των φαγητών ανά είδος που έχουν ήδη ετοιμασθεί στο εστιατόριο. Κάθε μία από αυτές τις θέσεις έχει χωρητικότητα 5 μερίδων. Οι «πελάτες» της εφαρμογής που μπορούν να επικοινωνήσουν με τον εξυπηρετητή μπορεί να είναι είτε οι chef είτε οι πελάτες του καταστήματος.

Όταν ένας πελάτης του καταστήματος ζητήσει κάποιες μερίδες φαγητού ανά είδος και δεν υπάρχουν διαθέσιμες όλες, θα πρέπει να «μπλοκάρει» το αίτημα μέχρι οι αντίστοιχοι chef φτιάξουν τις μερίδες που απαιτούνται. Τότε μόνο μπορεί να εξυπηρετηθεί ο πελάτης. Επίσης όταν γεμίσει μία από τις 3 θέσεις αποθήκευσης ο chef του συγκεκριμένου είδους θα πρέπει να «μπλοκάρει» και να περιμένει μέχρι κάποιος πελάτης να ζητήσει έστω μια μερίδα από το συγκεκριμένο είδος φαγητού.

Υλοποιήστε την παραπάνω κατανεμημένη εφαρμογή με χρήση Java Sockets.

Υποδείξεις για την υλοποίηση

- Για τις θέσεις αποθήκευσης των φαγητών χρησιμοποιείτε οποιαδήποτε δομή δεδομένων θεωρείτε ότι είναι κατάλληλη.
- Δημιουργήστε τυχαίους χρόνους καθυστέρησης ανάμεσα στις εκτελέσεις ενεργειών από τους σεφ και τους πελάτες του καταστήματος.
- Χαρακτηρίστε με δικό σας τρόπο τα διάφορα είδη που παράγονται από τους σεφ και καταναλώνονται από τους πελάτες.



Πανεπιστήμιο Αιγαίου
Τμήμα Μηχανικών Πληροφοριακών και Επικοινωνιακών
Συστημάτων

Κατανεμημένα Συστήματα – 2η Ατομική Εργασία
Ημερομηνία Παράδοσης : 19/04/2022

Εργαστηριακός Διδάσκοντας: Δούμα Αναστασία

- Εμφανίστε κατάλληλα μηνύματα για κάθε συμβάν που προκύπτει (π.χ. παραγωγή ψαριού από τον 1^ο σεφ, αίτημα για ποικιλία κρεατικών από πελάτη).
- Η υλοποίηση θα πρέπει να περιλαμβάνει δύο διαφορετικά projects, το πρόγραμμα του πελάτη και το πρόγραμμα του εξυπηρετητή.
- Δεν απαιτείται η υλοποίηση γραφικού περιβάλλοντος εκτός αν το επιθυμείτε.
- Εφόσον δεν έχετε δίκτυο υπολογιστών ώστε να τρέξετε τις διεργασίες πελάτη και εξυπηρετητή σε διαφορετικούς κόμβους μπορείτε να χρησιμοποιήσετε τη διεύθυνση localhost (IP address 127.0.0.1). Ανοίξτε διαφορετικά παράθυρα τερματικών για τον πελάτη και τον εξυπηρετητή.
- Διασφαλίστε ότι κάθε φορά που μια σύνδεση με τον εξυπηρετητή τερματίζεται απελευθερώνονται οι πόροι του συστήματος τόσο στην πλευρά του πελάτη, όσο και στην πλευρά του εξυπηρετητή (δυναμική μνήμη, socket descriptors, κ.α.).

Οδηγίες για παράδοση Ασκήσεων:

- Η παράδοση των εργασιών θα γίνει **ηλεκτρονικά** μέσω του **e-class** μέχρι την **Τρίτη 19/04/2022** στις **23.59** το βράδυ.
- Θα πρέπει να σταλεί ένα αρχείο rar/zip με όνομα ΑριθμόςΜητρώου_Lab02 (π.χ. icsd19002_Lab01). Στο αρχείο θα περιέχονται τα εξής:
 - Projects (ένα project για τον πελάτη και ένα για τον εξυπηρετητή) με τα αρχεία των ασκήσεων. **Επιβάλλεται η χρήση σχολίων στον κώδικα σας.**
 - Ένα **αρχείο .pdf** με οθόνες εκτέλεσης (screenshots) των προγραμμάτων σας **που να φαίνεται ξεκάθαρα το αποτέλεσμα εκτέλεσης του προγράμματος για διαφορετικά σενάρια**. Στην αρχή του αρχείου θα αναγράφεται ο αριθμός μητρώου και το ονοματεπώνυμο σας.
- **Καμία εργασία δεν θα διορθωθεί εάν δεν έχει ακριβώς αυτή τη μορφή.**
- Η υποβολή κοινών απαντήσεων από διαφορετικούς φοιτητές δεν επιτρέπεται και θεωρείται ως **ΑΝΤΙΓΡΑΦΗ**. Η αντιγραφή έχει ως αποτέλεσμα το **ΜΗΔΕΝΙΣΜΟ ΤΗΣ ΕΡΓΑΣΙΑΣ ΣΥΝΟΛΙΚΑ**.