

## Παράλληλος και Κατανεμημένος Υπολογισμός

### Εργαστήριο 10

#### 1. Μελετήστε τον κώδικα Code/Count-client-server-tcp

1.1 Υλοποιείτε ένα σύστημα πελάτη διακομιστή όπου ο διακομιστής εξυπηρετεί πολλούς πελάτες ταυτόχρονα. Ο κάθε πελάτης στέλνει έναν ακέραιο αριθμό  $n$ , πχ 100000 και ο διακομιστής εκτελεί στο πρωτόκολλό του τον υπολογισμό του  $\pi$  (δηλαδή τον ακολουθιακό κώδικα NumInt) και επιστρέφει την τιμή στον πελάτη.

*Για την εφαρμογή 3.1 δεν απαιτείται μοιραζόμενη κατάσταση. Αρκεί ένας πολυνηματικός διακομιστής όπως αυτός της εφαρμογής 1. Ο πελάτης μπορεί να τερματίζεται αν πχ επιλέξετε `NumInt == -1`.*

1.2 Ο διακομιστής αποθηκεύει το  $n$  και το  $\pi$  που υπολογίζει για κάθε πελάτη σε μια δομή (προφανώς ελέγχει αν υπάρχει ήδη καταχώρηση για το συγκεκριμένο  $n$ .) Πριν ξεκινήσει τον υπολογισμό νέου  $\pi$  ελέγχει αν ο υπολογισμός έχει ήδη γίνει. Αν ναι στέλνει το αποτέλεσμα στο πελάτη χωρίς να το υπολογίσει ξανά.

*Για την εφαρμογή 1.2 απαιτείται μια μοιραζόμενη δομή. Η προσπέλαση (αναζήτηση, ανάγνωση και πιθανή τροποποίηση της δομής) πρέπει να προστατεύεται με αμοιβαίο αποκλεισμό, ή η δομή να είναι ταυτόχρονη.*

#### 2. Μελετήστε τον κώδικα Code/Chat

2.1 Επεκτείνετε τον κώδικα ώστε να εξυπηρετεί  $n$  ταυτόχρονους συνομιλητές. Πρέπει να δημιουργήσετε μια μοιραζόμενη δομή που να κρατά τα στοιχεία των συνδέσεων.

2.2 Προαιρετική επέκταση: Προσθέστε λειτουργικότητα σύνδεσης και αποσύνδεσης συνομιλητών κατά τη λειτουργία. Πρέπει να προστατεύετε όλες τις λειτουργίες ανάγνωσης και εγγραφής της μοιραζόμενης δομής.