

# ΚΑΤΑΝΕΜΗΜΕΝΑ ΣΥΣΤΗΜΑΤΑ (Εργαστήριο)

## Άσκηση #1 2022-23

Ημ. Παράδοσης: 30/4/2023

Σας ζητείται να φτιάξετε σε C έναν concurrent<sup>1</sup> server (διεργασία εξυπηρετητή) ο οποίος ως έργο εξυπηρέτησης θα επιτελεί τους ακόλουθους υπολογισμούς (λαμβάνοντας ως εισόδους έναν πραγματικό αριθμό  $r$  και δύο διανύσματα ακεραίων  $X (x_1, x_2, \dots, x_n)$ ,  $Y (y_1, y_2, \dots, y_n)$  μήκους  $n$  όπου το  $n$  θα το ορίζει ο χρήστης, και τις οποίες θα μπορούν να στέλνουν επαναληπτικά ένας ή περισσότεροι clients<sup>2</sup> / διεργασίες πελατών):

1. Το εσωτερικό γινόμενο των δύο διανυσμάτων  $X \cdot Y$  (επιστροφή: ένας ακέραιος αριθμός)
2. Τη μέση τιμή κάθε διανύσματος:  $E_X, E_Y$  (επιστροφή: ένας πίνακας 2 πραγματικών αριθμών)
3. Το γινόμενο  $r \cdot (X+Y)$  (επιστροφή: ένα διάνυσμα-πίνακας πραγματικών αριθμών μήκους  $n$ )

Η επικοινωνία θα πρέπει να γίνεται μέσω TCP AF\_INET (Internet Domain) sockets. Η κάθε διεργασία socket-Client θα διαβάζει από το πληκτρολόγιο (επαναληπτικά, μέχρι να δηλώσει ο χρήστης ότι δεν επιθυμεί να συνεχίσει) (α) την επιλογή του υπολογισμού που επιθυμεί ο χρήστης να γίνει (1,2,3) και (β) τα αντίστοιχα-απαραίτητα κατά περίπτωση δεδομένα ( $n, X, Y, r$ ), θα τα διοχετεύει στη διεργασία socket-Server και θα περιμένει να λάβει από αυτήν το αποτέλεσμα για να το τυπώσει στην οθόνη.

Η διεργασία socket-Server θα δέχεται τα δεδομένα προς επεξεργασία από τις διεργασίες socket-Clients, και θα παράγει το αντίστοιχο αποτέλεσμα ΟΧΙ μέσω δικιάς του (τοπικής) συνάρτησης-υπολογισμού ΑΛΛΑ μέσω κατάλληλου Remote Procedure Call που θα υλοποιήσετε με τη βοήθεια του ONC RPC implementation. Θα πρέπει δηλαδή η διεργασία socket-Server (λειτουργώντας παράλληλα και ως RPC-Client) να καλεί (ανάλογα με την τιμή υπολογισμού που έστειλε ο χρήστης - 1,2,3) την αντίστοιχη ρουτίνα από έναν RPC-Server και να περιμένει το αντίστοιχο αποτέλεσμα από αυτόν (προκειμένου να το διοχετεύσει στη συνέχεια στον αντίστοιχο socket-Client).

Όσον αφορά το RPC-based μέρος της επικοινωνίας, θα πρέπει πρώτα να ορίσετε σωστά το απαιτούμενο ('.x') interface file (ορίζοντας μέσα σε αυτό τρεις ξεχωριστές συναρτήσεις-διαδικασίες (μία για κάθε έναν από τους τρεις υπολογισμούς που ζητούνται παραπάνω), στη συνέχεια να παράγετε αυτοματοποιημένα μέσω του *rpcgen* utility (και με βάση τα όσα διδαχθήκατε στο εργαστήριο) τόσο τα απαιτούμενα system modules (RPC-server-stub module και RPC-client-stub module) για την υλοποίηση των ζητούμενων RPCs, όσο και τα έτοιμα templates για τα δύο application modules της εφαρμογής σας (RPC-server-application module και RPC-client-application module), και ακολουθώντας:

(α) να ολοκληρώσετε κατάλληλα το RPC-server-application module (το οποίο θα επιτελεί τις βασικές εργασίες εξυπηρέτησης πάνω στα δεδομένα-παραμέτρους που θα στέλνει απομεμακρυσμένα ο RPC-Client/socket-Server) και

(β) να ολοκληρώσετε κατάλληλα επίσης το RPC-client-application module (μέσω του οποίου θα επιτελείται επί της ουσίας η κλήση της εκάστοτε απομεμακρυσμένης διαδικασίας) ενσωματώνοντας/συγχωνεύοντάς το μεταξύ άλλων με τη βασική διεργασία socket-Server που περιγράφηκε παραπάνω.

**Παραδοτέα:** κώδικας, σχολιασμός/τεκμηρίωση, ενδεικτικά τρεξίματα

<sup>1</sup> με δυνατότητα δηλαδή ταυτόχρονης εξυπηρέτησης πολλαπλών αιτήσεων.

<sup>2</sup> τη διεργασία εκτέλεσης των οποίων επίσης καλείστε να φτιάξετε.