

**Εργαστήριο Δικτύων Υπολογιστών**  
**Δεύτερη Άσκηση**  
**6/12/13**

**Σειμένης Σπύρος, 5070, netlab209**

**1. Συνοπτικός τρόπος λειτουργίας server/client**

**Server.c**

Ο TCP server αρχικά αναλαμβάνει να ανοίξει ένα socket για tcp επικοινωνία στο port που του δόθηκε. Στη συνέχεια επεξεργάζεται κάθε σύνδεση client που δέχεται ως εξής: Αρχικά διαβάζει απο τον client την udp port που θα χρησιμοποιήσουν οι clients για την μεταξύ τους επικοινωνία και αφού επιτρέπονται 10 συνδέσεις (για ports 10080-10089 για την ομάδα 209 συγκεκριμένα) διατηρεί σε 2 στατικούς πίνακες τις πληροφορίες του. Για κάθε επόμενο client που έρχεται, εάν το udp port είναι διαφορετικό και κανένας client δεν έχει ζητήσει επικοινωνία με αυτό επαναλαμβάνει την παραπάνω διαδικασία. Εάν κάποιος client έχει udp port ίδιο με κάποιον που έχει ήδη έρθει τότε γίνεται επιλογή του ποιος θα είναι server και ποιος client και αποστέλλονται οι κατάλληλες πληροφορίες σε αυτούς ενώ κλείνει η σύνδεση τους με τον server.

**client.c**

Αρχικά ο client ανοίγει επικοινωνία με τον server και του στέλνει την udp port που θέλει να χρησιμοποιήσει. Στη συνέχεια αν είναι ο πρώτος που θέλει να χρησιμοποιήσει το συγκεκριμένο port περιμένει εως ότου να έρθει στον server άλλος ένας client. Μόλις δυο client είναι έτοιμοι για επικοινωνία ο server στέλνει και στους 2 τις πληροφορίες για την σύνδεση<sup>1</sup>. Έπειτα ο client που λειτουργεί ως udp server ανοίγει ένα udp socket στο κατάλληλο port ώστε ο client να μπορεί να συνδεθεί και να του στείλει μηνύματα.

**Σημειώσεις**

Στην έκδοση που έχει παραδοθεί, οι clients που είτε έχουν επιλεγεί ως server είτε ως client κάνουν και οι δύο bind ένα udp socket, δηλαδή λειτουργούν και οι δύο ως udp servers. Αυτό είναι μια μικρή αλλαγή η οποία επιτρέπει στους δυο clients να ανταλλάσσουν μηνύματα σύγχρονα. Οι αλλαγές στον κύριο κώδικα που επιτυγχάνουν αυτή την συμπεριφορά είναι:

- 1) ο tcp server στέλνει και στους 2 clients τις πληροφορίες σύνδεσης
- 2) αρχικοποιούνται 2 udp sockets ένα με την udp port που έχει δοθεί και ένα (για τον "client") στην port που δόθηκε +10, για να μην επηρεαστεί η υπόλοιπη συμπεριφορά του προγράμματος.
- 3) και ο client κάνει bind ένα socket στο udp port+10 για να λαμβάνει μηνύματα απο τον server.

## 2. Αναλυτικός τρόπος λειτουργίας μια ολοκληρωμένης σύνδεσης

Ο server διατηρεί 2 στατικούς πίνακες αποθήκευσης πληροφοριών. Ένα για τα sockets των πρώτων clients που φθάνουν και θα πρέπει να περιμένουν και έναν με structs settings που δηλώνεται στο common.h και περιέχει τις πληροφορίες σύνδεσης.

Ένας client με udp port 10080 επικοινωνεί με τον server και του στέλνει το udp port. Ο server εξετάζει τον στατικό πίνακα των settings για το αν έχει τεθεί το πεδίο του port. Εφόσον δεν έχει τεθεί εκτελείται το else. Τίθεται το πεδίο του port στην κατάλληλη θέση του πίνακα(το offset χειρίζεται την αριθμητική πίσω από αυτό) και λαμβάνεται και μια απόφαση για το εάν ο client θα λειτουργήσει ως server ή client(όχι απαραίτητο με την τρέχουσα έκδοση που λειτουργούν και οι δύο ως server). Στη συνέχεια ενημερώνει τον client ότι είναι ο πρώτος που έφτασε και θα πρέπει να περιμένει και ανάλογα με τον τύπο του client αποθηκεύει τις πληροφορίες του στο struct και κρατάει το socket του.

Ένας δεύτερος client με udp port 10080 επικοινωνεί με τον server. Αυτή την φορά το πεδίο του πορτ στην θέση 10080 έχει τεθεί που σημαίνει ότι ήδη ένας client περιμένει για σύνδεση άρα και εκτελείται το if. Εάν ο προηγούμενος client ήταν server ενημερώνονται τα πεδία του client στο struct και αντίστροφα. Στέλνονται και στους 2 clients οι πληροφορίες της σύνδεσης μέσω του struct με μόνη διαφορά το πεδίο type που στον έναν δηλώνει server(0) και στον άλλον client(1). Μηδενίζεται το πεδίο στον στατικό πίνακα για να δηλώσει πως αυτό το port είναι πλέον ξανά διαθέσιμο και κλείνουν τα tcp sockets.

Από την μεριά των clients, ο udp server κάνει bind στο κατάλληλο port ένα socket και περιμένει μηνύματα από τον client ενώ ο client απλά ξεκινά να στέλνει μηνύματα στον udp server. Όπως προανέφερα, για να επιτραπεί επικοινωνία και από τον server στον client έχει γίνει μια μικρή τροποποίηση στον αρχικό κώδικα. Εκτός από το socket του udp server αρχικοποιείται και ένα socket στην udp port+10 ώστε και ο client να κάνει bind σε αυτό και να λειτουργεί και αυτός ως server.