

Esercizio chat tramite socket TCP

Si progetti ed implementi un'applicazione client-server TCP che realizzi una semplice chat con le seguenti caratteristiche:

- Il server riceve le richieste di connessione da parte dei client e funziona come punto di raccolta e smistamento (a seguito di elaborazione) di messaggi.
- Ogni client, al momento della connessione, fornisce al server un nickname con cui l'utente comparirà in chat. Successivamente, il server, in risposta ad un messaggio ricevuto da un dato client, invierà a tutti gli altri client un nuovo messaggio, antepoendo ad esso il nickname del client che ha inviato. Il server stampa a video un messaggio contenente il nickname di ogni client al momento della connessione. Per esempio, il client Pippo si `e connesso.
- Il messaggio che il server invia agli altri client, in risposta ad un messaggio ricevuto, `e frutto di una computazione. Precisamente, alla ricezione di ogni messaggio da parte di un client, il server genera un numero pseudo-casuale: se il numero `e pari, calcola ed invia il reverse del messaggio ricevuto. Per esempio, se il client Pippo invia il messaggio "ciao" al server e il numero pseudo-casuale generato dal server `e 10602, il server invierà a tutti gli altri client in chat il messaggio "Pippo:10602:oaic". Se, invece, il numero `e dispari, allora invia il messaggio inalterato. Per esempio, se il client Pippo invia il messaggio "bravo" al server e il numero pseudo-casuale che il server ha generato `e 12403, il server invierà a tutti gli altri client in chat il messaggio "Pippo:12403:bravo".
- Quando il client si disconnette, il server stampa a video un messaggio di disconnessione. Per esempio, "il client Pippo ha lasciato la chat".
- Infine, quando il server viene arrestato, prima di terminare definitivamente la propria esecuzione, stampa a video un messaggio di terminazione. Per esempio, "la chat è stata terminata."