

Programmazione di Sistemi in Rete (A.A. 2017/18)

Esercitazione n. 2 del 10 ottobre 2017

Prof. Eugenio Zimeo

Esercizio 2.1

Scrivere un'applicazione client/server per realizzare in remoto la somma di due numeri interi. Il client dovrà leggere da standard input due numeri interi ed inviarli al server. Il server, una volta ricevuti i due numeri, ne effettua la somma ed invia il risultato al client. Si usi il modello di comunicazione orientato ai flussi (protocollo di trasporto TCP).

Esercizio 2.2

Scrivere un'applicazione client/server per realizzare in remoto la somma di due numeri interi. Il client dovrà leggere da standard input due numeri interi ed inviarli al server. Il server, una volta ricevuti i due numeri, ne effettua la somma ed invia il risultato al client. Si usi il modello di comunicazione orientato ai datagram (protocollo di trasporto UDP).

Esercizio 2.3

Scrivere un'applicazione per consentire la chat tra due utenti che interagiscono attraverso due processi. Si utilizzi il modello di comunicazione orientato ai flussi (protocollo di trasporto TCP).

Un'applicazione per la chat è per sua natura peer-to-peer (ciascun processo può inviare e ricevere stringhe di caratteri). Il canale di comunicazione orientato ai flussi costruito sul protocollo TCP è full-duplex. Pertanto la comunicazione può avvenire indifferentemente nei due versi. La particolare organizzazione dell'interfaccia delle socket orientate ai flussi prevede però che un processo si predisponga al contatto (listen) e l'altro promuova il contatto (connect). Queste due modalità di funzionamento conferiscono ai due processi i ruoli di server e client, rispettivamente, solo per la costruzione del canale di comunicazione. Una volta creato il canale, i due processi possono assumere in momenti diversi i ruoli di server e di client (peer). Si tenga in considerazione la semantica bloccante della funzione di lettura (read()). Realizzare allo scopo un protocollo applicativo di tipo half-duplex.