

Sistemi Operativi e in Tempo Reale

Esercizio: Sportelli

Si realizzi in ambiente Unix/C l'interazione tra processi descritta nel seguito. Il sistema consiste di tre tipi di processi: un processo server S, processi utenti U e processi impiegati I. Per la comunicazione tra il processo server e i processi cliente vengono utilizzate socket di tipo Stream.

Il processo S gestisce il sistema di sportelli di un ufficio pubblico. I processi U rappresentano gli utenti identificati dal loro univoco nome (una stringa di max. 20 char), che richiedono uno dei K servizi ($K=3$) erogati dall'ufficio pubblico. I servizi forniti hanno un codice numerico rappresentato da un intero compreso fra 0 e $K-1$. I processi I rappresentano gli impiegati, ciascuno dei quali è abilitato a fornire almeno uno dei K servizi.

Ciascun processo U, ad ogni invocazione, si registra presso il server comunicando il proprio nome ed il codice del servizio richiesto. Successivamente si mette in attesa della disponibilità di uno degli impiegati abilitati a svolgere il servizio richiesto. Il processo U, infine, riceve la convocazione da parte di un impiegato che eroga il servizio oppure l'impossibilità di ricevere il servizio e termina la sua esecuzione.

Ciascun processo I contatta il server comunicando i servizi che è in grado di erogare. Successivamente riceve la lista degli utenti, che hanno richiesto uno dei servizi che l'impiegato è in grado di svolgere, ordinata in base all'ordine di arrivo degli utenti. Invia, quindi, al server il nominativo del primo utente tra quelli che è in grado di servire e termina la sua esecuzione.

Il processo server S mantiene la lista o le liste degli utenti U in coda per ciascuno dei K servizi ordinati in base al loro ordine di arrivo. Quando un processo U comunica il proprio nome ed il servizio richiesto, il server lo inserisce in coda e lo mette in attesa. Quando un processo impiegato I invia un messaggio al server comunicando i servizi che è in grado di erogare, il server risponde inviando all'impiegato la lista degli utenti in attesa dell'erogazione di uno di tali servizi ordinata secondo l'ordine di arrivo. Successivamente, l'impiegato invia il nominativo dell'utente che intende servire e termina la sua esecuzione. Il server comunica, quindi, al processo utente U servito e termina la sua esecuzione.

L'ufficio adotta, inoltre, una politica (non equa e poco razionale) per ridurre la lunghezza delle code. Il server, dopo aver composto ed inviato la lista degli utenti al processo impiegato corrente come illustrato precedentemente, elimina da tale lista tutti i processi utente U dalla $(L+1)$ -esima posizione della lista in avanti ($L=4$). In particolare, invia a ciascuno di tali utenti in attesa un messaggio di mancata disponibilità del servizio e termina la comunicazione.