

[illegible]

## Testo della prova

Si realizzi in linguaggio C/C++ la seguente applicazione basata su **code di messaggi UNIX**.

La applicazione include 2 **processi server**. Ogni server è dotato di una **risorsa logica**, rappresentata da una variabile intera. Al loro avvio, ogni processo server crea una propria **mailbox privata**, su cui ricevere le richieste dei client. Prima di porsi in attesa delle richieste, i server inviano ad un **processo registro** una richiesta di "*bind*", con cui inviano al registro un loro identificativo numerico ("0" oppure "1"), e lo ID della loro coda di messaggi UNIX privata.

Il **processo registro** contiene al suo interno un array di interi, in cui memorizza per ogni server lo ID della corrispondente coda di messaggi. Inoltre, il processo registro crea una coppia di mailbox, la prima per ricevere i messaggi di "*bind*" e "*query*", e la seconda per inviare i messaggi di "*result*".

I 3 **processi client** scelgono in modo casuale quale server contattare (un numero tra "0" e "1"). Essi inviano inizialmente un messaggio di "*query*" al **processo registro**, indicando il server che vogliono contattare, e ricevono un messaggio di "*result*" che indica la coda di messaggi con cui contattare il server che hanno scelto. Infine, ogni client invia al server 3 messaggi di "*service*" (attendendo 1 secondo ad ogni invio), contenenti un valore intero casuale tra 0 e 10. Ogni **processo server** riceve le richieste dei client, sovrascrivendo la propria risorsa logica con il valore ricevuto.

Si crei un unico eseguibile, in cui un processo padre avvia tutti gli altri processi. Il padre attende che tutti i processi client abbiano terminato dopo aver inviato ognuno 3 messaggi. Dopodiché, il padre invia un messaggio di "*exit*" al processo registro, che a sua volta invia due messaggi di "*exit*" ai due server.

