

CORSO DI SISTEMI OPERATIVI A Prova del 19/6/2003

MATR. Cognome Nome

Username

NOTE

Il presente foglio va immediatamente compilato con le proprie generalità e matricola. Esso deve essere restituito al termine della prova. In caso di mancata restituzione, la prova dello studente non verrà presa in considerazione per la correzione.

IMPORTANTE

Tutti i file sorgenti prodotti dallo studente per l'esame devono essere memorizzati in un direttorio denominato **soa-190603-x** nella propria home, dove **x** rappresenta il carattere **i** per gli Informatici, **t** per i Telecomunicazionisti, e per gli Elettronici. Soluzioni contenute in altri direttori non verranno prese in considerazione per la correzione.

Prova UNIX-C1

Si realizzi in ambiente Unix/C la seguente interazione tra processi:

- il sistema consiste di due processi: un processo P_{padre} che crea un processo P_{figlio} ;
- il processo P_{figlio} visualizza la progressione dei numeri interi;
- per N volte (con $1 < N < 5$ costante) P_{padre} attende un intervallo di durata casuale e invia al processo P_{figlio} un segnale SIGUSR1 ;
- alla N -esima notifica del segnale SIGUSR1 il processo P_{figlio} deve terminare.

Devono essere utilizzate le primitive per la gestione affidabile dei segnali.

Prova UNIX-C2

Si realizzi in ambiente Unix/C la seguente interazione tra processi:

- il sistema consiste di due tipi di processi: un processo *server* remoto P_s e i processi clienti P_{ci} ;
- per la comunicazione tra P_s e i processi clienti P_{ci} vengono utilizzate socket **Stream**;
- il server P_s offre un servizio concorrente (un figlio per ogni connessione) di esecuzione comandi alla porta N (dove N è ottenuto dalle 5 cifre meno significative della vostra matricola);
- ogni processo cliente P_{ci} richiede all'utente il nome di un comando e lo invia al server ;

- i processi figli del server P_s , ricevuto il messaggio contenente il nome del comando, devono attivare un nuovo figlio per l'esecuzione del comando e attenderne la terminazione, trasmettendo il valore di uscita al processo cliente ;
- ogni processo cliente P_{ci} deve memorizzare quanto ricevuto dalla socket connessa in un file denominato "session.log".