

Università degli Studi di Napoli Federico II
Corso di Laurea in Ingegneria Informatica
Esame di Sistemi Operativi
Proff. Cotroneo, Natella

Prova pratica del 14/02/2013
Durata della prova: 150 minuti

Cognome Nome Matr.

Lo studente legga attentamente il testo e produca il programma, il makefile, ed i casi di test necessari per dimostrarne il funzionamento. La mancata compilazione dell'elaborato, la compilazione con errori o l'esecuzione errata del programma daranno luogo alla valutazione come **prova non superata**. Ricordarsi di indicare Nome, Cognome e matricola su questo stesso foglio, che dovrà essere in ogni caso consegnato alla Commissione. Al termine della prova lo studente dovrà fare verificare il funzionamento del programma ad un membro della Commissione.

Testo della prova

Si realizzi in linguaggio C o C++ un programma multi-processo che simuli la gestione e prenotazione degli appelli di esame universitari. Il programma dovrà gestire le seguenti due risorse, da condividere tra i processi usando **shared memory UNIX**, e gestendo la sincronizzazione tramite **semafori UNIX**:

- Una variabile stringa che rappresenta la data del prossimo appello (ad esempio "14/02/2013");
- Una variabile intera che rappresenta il numero di studenti prenotati.

Il programma dovrà includere due tipi di processi:

- Un processo **docente**, che può aggiornare la variabile "prossimo appello", e che consulta periodicamente la variabile "numero prenotati";
- Un gruppo di processi **studente**, che può consultare la variabile "prossimo appello", e prenotarsi per l'appello corrente incrementando di 1 la variabile "numero prenotati".

Occorrerà inoltre garantire i seguenti vincoli:

- **Variabile "prossimo appello"**: Il processo docente e i processi studente dovranno accedere in mutua esclusione alla variabile "prossimo appello" (ad esempio, se un docente sta aggiornando la data, nessuno studente può consultarla; se uno studente sta consultando la data, il docente non può modificarla). Tuttavia, più studenti dovranno poter consultare contemporaneamente la data del prossimo appello.
- **Variabile "numero prenotati"**: Il processo docente e i processi studente dovranno accedere in mutua esclusione alla variabile "numero prenotati".

Si implementi il programma dividendo il codice in **tre eseguibili**, uno eseguito dal processo docente, uno eseguito dagli studenti, ed uno eseguito da un processo padre che istanzia 1 docente e 10 studenti e ne attende la terminazione. Per 3 volte, il processo docente dovrà aggiornare la data di esame (si usino delle stringhe a piacere), attendere 3 secondi, stampare ed azzerare il numero di prenotati (la stampa e l'azzeramento avvengano nella stessa sezione critica). Ciascun processo studente dovrà attendere per una quantità di tempo casuale (tra 0 e 8 secondi, usando la primitiva `sleep(int secondi)`), stampare la data dell'appello disponibile in quel momento, e prenotarsi all'appello incrementando la variabile del numero di prenotati.