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

Prova pratica del 21/12/2020 - TURNO 4 Durata della prova: 75 minuti

Lo studente completi il programma a corredo di questo documento, seguendo le seguenti indicazioni.

La prova sarà valutata come segue:

- A: Prova svolta correttamente.
- **B**: Il programma non esegue correttamente, con errori minori di programmazione o di concorrenza.
- C: Il programma non esegue correttamente, con errori significativi (voto max: 22).
- **INSUFFICIENTE**: Il programma non compila o non esegue, con errori gravi di sincronizzazione.

Istruzioni per la consegna dell'elaborato

L'elaborato dovrà essere svolto in una cartella dal nome: Cognome Nome Matricola Docente

Esempio:

In alternativa, è consentito creare il file compresso "tar" tramite l'interfaccia grafica.



All'interno della macchina virtuale, aprire il browser all'indirizzo: https://tinyurl.com/y97qfmx9

Nel form, cliccare su "aggiungi file", selezionare il file compresso contenente il proprio svolgimento, e indicare il proprio nome ed email @studenti.unina.it.

Attendere una notifica del docente, e quindi scollegarsi dalla piattaforma di VirtualClassroom.

Testo della prova

Si realizzi in linguaggio C/C++ un'applicazione basata su **code di messaggi UNIX** e **thread**, che simuli la lettura di dati da parte di un sensore, e la distribuzione dei dati a più destinazioni. Un processo **sensore** dovrà periodicamente (ogni secondo) generare un valore numerico casuale, tra 0 e 10, e inviarlo tramite una prima coda di messaggi. Un processo **aggregatore** dovrà ricevere i dati, e inoltrare una copia a tutti e tre i processi **collettori**, attraverso delle code di messaggi riservate ad ogni collettore.

Il processo aggregatore dovrà includere 4 thread. Il thread **scrittore** dovrà prelevare i messaggi dal processo sensore, e scrivere il valore ricevuto su una variabile condivisa tra i thread. I thread **lettori** dovranno leggere periodicamente (ogni secondo) il contenuto della variabile condivisa, e inviarlo ai processi collettori. Come nello schema generale dei lettori-scrittori, i thread lettori devono poter **leggere in concorrenza** la variabile condivisa.

Un processo padre dovrà creare gli altri processi attraverso **fork()**. Il processo sensore dovrà generare 10 messaggi in un ciclo, attendendo 1 secondo tra le iterazioni. Il thread scrittore effettuerà una scrittura ad ogni messaggio. I thread lettori dovranno leggere la variabile condivisa per 10 volte in un ciclo, attendendo un secondo tra le iterazioni, ed inviando altrettanti messaggi ai collettori. A loro volta, i collettori faranno 10 ricezioni, stampando a video i valori ricevuti, per poi terminare.

