

# Università degli studi di Roma “Tor Vergata”

**Insegnamento di Sistemi Operativi**  
**Corso di Laurea in Ingegneria Informatica**  
**Appello d’esame del 13/6/2022**  
**Docente: Francesco Quaglia**  
**Punteggio massimo raggiungibile: 21 punti**  
**Soglia per la sufficienza: 12 punti**

MATRICOLA \_\_\_\_\_ Cognome \_\_\_\_\_ Nome \_\_\_\_\_

*Si raccomanda di scrivere il proprio cognome e nome su questo foglio e di utilizzarlo come cartellina per contenere i fogli con le risposte. Se si considera ambigua una domanda, scrivere la propria interpretazione e rispondere conseguentemente.*

## **Domanda 1 (5.25 punti)**

Si descriva lo scheduling di CPU round-robin virtuale. Si consideri uno scenario in cui lo scheduler di CPU round-robin virtuale sia basato su un time-slice pari a  $K > 10$  millisecondi, esista un unico processo CPU-bound A di durata infinita ed esista un unico processo I/O-bound B di durata infinita. Il processo B ha CPU burst di lunghezza pari a 10 millisecondi. Si indichi il massimo valore di  $K$  che possa permettere al processo B di avere un tempo di attesa per l’esecuzione del CPU burst non più ampio del 250% rispetto alla durata del time-slice.

## **Domanda 2 (5.25 punti)**

Fornire una descrizione degli approcci e dei supporti per la protezione dei file utilizzati nei file system Unix.

## **Domanda 3 (5.25 punti)**

Si descriva il meccanismo di gestione della memoria basato sulla paginazione. Si consideri uno schema di gestione della memoria basato su paginazione, in cui gli indirizzi logici sono formati da 30 bit e l’indirizzamento della memoria è al singolo byte. Si consideri inoltre che il meccanismo di paginazione sia basato su due livelli, e che il primo livello sia basato su una tabella di 1024 elementi. Si calcoli il numero di possibili pagine costituenti l’address space dell’applicazione e il numero di elementi di una tabella di secondo livello nel caso in cui ciascuna delle pagine abbia taglia 4 KB oppure abbia taglia 8 KB.

## **Domanda 4 (5.25 punti)**

Si consideri un insieme di  $N$  processi ( $P_1, \dots, P_N$ ) ed un altro insieme di  $M$  processi ( $L_1, \dots, L_M$ ). Ogni processo  $P_i$  scrive periodicamente un nuovo messaggio in una memoria condivisa  $M$ , che deve essere letto una sola volta da tutti i processi  $L_j$ . Il processo  $P_i$  può scrivere un nuovo messaggio solo dopo che l’ultimo messaggio scritto sia stato letto da tutti i processi  $L_i$ . Altrimenti dovrà rimanere in attesa. Allo stesso tempo, un processo  $P_i$  che ha scritto un messaggio in  $M$  ne può scrivere un successivo solo dopo che anche tutti gli altri processi  $P_j$  (con  $j$  diverso da  $i$ ) abbiano scritto il loro messaggio in  $M$ . Altrimenti dovrà rimanere in attesa. Allo stesso tempo, ogni processo  $L_j$  che intende leggere dovrà rimanere in attesa che un messaggio non ancora letto da  $L_j$  sia reso disponibile. Si schematizzi la soluzione del suddetto problema di sincronizzazione, usando solo semafori, fornendo lo pseudo-codice delle procedure SCRIVI e LEGGI usate rispettivamente dai generici processi  $P_i$  e  $L_j$ .

La pubblicazione del risultato via Web avverrà in forma anonima utilizzando il numero di matricola. Per avere il proprio voto d’esame pubblicato tramite il sito Web del corso bisogna firmare la seguente autorizzazione.

Il Sottoscritto, ai sensi della legge 675 del 31/12/96, autorizza il Docente a pubblicare in bacheca e su Web i risultati della prova d’esame. In fede

Firma leggibile: \_\_\_\_\_