

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA  
PROVA SCRITTA DI SISTEMI OPERATIVI  
ANNO ACCADEMICO 2024/2025  
23 giugno 2025

Esercizio -1: Essere iscritti su AlmaEsami per svolgere questa prova.

Esercizio 0: Scrivere correttamente nome, cognome, matricola e posizione in tutti i fogli prima di svolgere ogni altro esercizio. Scrivere esclusivamente a penna senza abrasioni. E' vietato l'uso delle penne cancellabili, della matita, dei coprenti bianchi per la correzione (bianchetto) e la scrittura in colore rosso (riservato alla correzione).

Il compito è formato da tre fogli, sei facciate compresa questa. Le soluzioni che si vogliono sottoporre per la correzione devono essere scritte negli spazi bianchi di questi fogli. Non verranno corretti altri supporti.

E' obbligatorio consegnare il compito, è possibile chiedere che esso non venga valutato scrivendo "NON VALUTARE" in modo ben visibile nella prima facciata.

Per svolgere questo compito occorre solo una penna e un documento di identità valido. La consultazione o anche solo la disponibilità di altro materiale comporterà l'annullamento del compito (verrà automaticamente valutato gravemente insufficiente).

**Esercizio c.1:** Scrivere il monitor `minlenbb` che fornisce le seguenti procedure entry:

```
void minlenbb.add(objtype obj)
objtype minlenbb.get(int bufnum)
```

Il monitor gestisce `N` buffer ognuno dei quali contiene `NELEM` elementi.

La funzione `add` inserisce `obj` come elemento nel buffer che contiene meno elementi. In caso più buffer abbiano lo stesso numero minimo di elementi viene scelto il buffer di indice minore.

La funzione `get` prende un elemento dal buffer di indice `bufnum`

**Esercizio c.2:** Un servizio chiamato `threshlocking` ha le due seguenti funzioni di interfaccia:

```
void threshlock(int level)
void chthreshold(int newlevel)
```

Il servizio mantiene il valore di soglia corrente: un numero intero, con valore iniziale zero.

Se un processo chiama `threshlock` con parametro `level` maggiore o uguale alla soglia corrente viene bloccato, altrimenti no.

La funzione `chthreshold` cambia il valore di soglia portandolo a `newlevel`. Tutti i processi in attesa che avevano specificato un parametro `level` minore di `newlevel` devono essere sbloccati.

Implementare il servizio utilizzando semafori.

**Esercizio g.1:** L'algoritmo di rimpiazzamento `minpage` sceglie come pagina vittima quella avente il minimo numero di pagina.

1. mostrare una stringa di riferimenti di lunghezza infinita tale che `minpage`, LRU e MIN si comportino allo stesso modo. (NB una sequenza infinita deve generare infiniti page fault e riferirsi a un numero finito di pagine).
2. mostrare che `minpage` non è soggetto alla anomalia di Belady.
3. nonostante non sia soggetto ad anomalia di Belady (e questa è una caratteristica positiva), `minpage` non è un algoritmo di rimpiazzamento efficiente, perché?

**Esercizio g.2:** rispondere alle seguenti domande (*motivando opportunamente le risposte!*):

- a) L'algoritmo LOOK (detto anche dell'ascensore) è più efficiente di C-LOOK (che ha lo stesso principio di funzionamento del metodo LOOK, ma la scansione del disco avviene in una sola direzione). In quali casi di preferisce C-LOOK a LOOK?
- b) Spiegare quali attacchi sono possibili se non viene usato il meccanismo del "sale" (salt) nella memorizzazione delle password criptate..
- c) L'algoritmo del Banchiere evita il verificarsi di situazioni di deadlock ritardando le richieste che porterebbero il sistema in uno stato unsafe. Basta questa regola a evitare anche situazioni di starvation?
- d) Perché è necessario usare spinlock in sistemi multiprocessore per implementare kernel di tipo simmetrico (SMP)?