

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
 PROVA SCRITTA DI SISTEMI OPERATIVI
 ANNO ACCADEMICO 2022/2023
 22-30 maggio
 1 giugno 2023

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 e' 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, e' 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 cs che fornisca un servizio di elaborazione client-server.

I molteplici "clienti" chiedono elaborazioni ai server eseguendo la seguente funzione:

```
def service_request(data):
    return cs.request(data)
```

mentre i server eseguono il codice:

```
process server(i: i = 0, ..., NSERVER-1):
    while True:
        data = cs.get_request(i)
        cs.send_result(i, process(data))
```

Quando un server è libero chiede una nuova richiesta da elaborare (funzione get_request), se non ci sono richieste da elaborare attende che un cliente ne sottoponga una (tramite la funzione request). Se vi sono uno o più richieste in attesa di essere elaborate get_request restituisce i dati (argomento data) della prima.

Dopo che il server ha elaborato la richiesta (funzione process) il risultato viene passato al monitor tramite la funzione send_result che lo restituisce al cliente come valore di ritorno della funzione request.

Esercizio c.2: Scrivere, facendo uso di semafori, la funzione syncvalue che ha la seguente dichiarazione:

```
void syncvalue(int key);
```

I processi che chiamano la syncvalue si bloccano sempre. Quando il valore del parametro key è diverso da quello della precedente chiamata il processo prima di bloccarsi riattiva tutti i processi in attesa. Per esempio:

P chiama syncvalue(42), si blocca.

Q chiama syncvalue(42), si blocca.

R chiama syncvalue(44) sblocca P e Q poi si blocca.

T chiama syncvalue(46), sblocca R e si blocca.

P chiama syncvalue(46), si blocca.

V chiama syncvalue(0), sblocca T e P poi si blocca...

Esercizio g.1: Un sistema applica l'algoritmo del Banchiere a tre valute. La disponibilità iniziale del banchiere (IC) è (4,3,4).

Inizialmente non c'è alcun processo attivo. Ad un determinato istante $t=0$ sono attivi 3 processi:

P1 che ha un credito (2,2,3) e ha allocato le risorse (prestito attuale) (1,2,3),

P2 che ha un credito (2,3,0) e ha allocato (0,1,0)

P3 che ha un credito (3,0,3) e ha allocato (2,0,0)

Mostrare che a $t=0$ lo stato è safe. Successivamente il sistema evolve:

$t=1$: Viene chiesta l'attivazione del processo P4 che ha un credito (1,0,0) e chiede per iniziare l'esecuzione (1,0,0)

$t=2$: Termina il processo P4

$t=3$: P3 richiede ulteriori risorse (1,0,0)

$t=4$: Viene chiesta l'attivazione del processo P5 che ha un credito (1,0,0) e chiede per iniziare l'esecuzione (1,0,0)

$t=5$: Il processo P1 termina.

Mostrare l'evoluzione delle strutture dati dell'algoritmo del banchiere e le decisioni prese dall'algoritmo.

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

a) Nei file system UNIX il nome del file non è memorizzato nell'I-node. Dove è memorizzato? Quale operazione non sarebbe possibile se il nome fosse memorizzato nell'I-node? Perché?

b) Da quale tipo di attacco protegge il meccanismo del sale (salt) nella memorizzazione delle password?

c) Per migliorare la gestione dei processi interattivi da parte dello scheduler round robin viene proposto di assegnare quanti di tempo (time slice) diversi ai processi: più corti ai processi interattivi e più lunghi ai processi batch (che fanno molti calcoli). E' una buona idea o no?

d) Per applicare l'algoritmo di rimpiazzamento LRU (Least Recently Used) occorrerebbe un hardware molto complesso e costoso, perché?