

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
PROVA SCRITTA DI SISTEMI OPERATIVI
ANNO ACCADEMICO 2024/2025
21 luglio 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 nmeet che ha una funzione di interfaccia:

```
float nmeet(unsigned int n, float val)
```

La funzione nmeet blocca il processo chiamante fino a quando n processi abbiano richiamato nmeet con parametro n. (es due processi che hanno chiamato la nmeet con n=3 rimangono bloccati, quando un terzo processo chiama nmeet con n=3, i tre processi continuano l'esecuzione.) Il valore di ritorno di nmeet e' la media dei valori 'val' passati dagli n processi che hanno sbloccato la chiamata nmeet.

Esercizio c.2: Sia dato un servizio di message passing asincrono. Implementare senza fare uso di processi server un servizio di message passing col segno sincrono mpss che fornisce le due seguenti funzioni:

```
void mpssend(msg_t msg, pid_t destination)
```

```
msg_t mpssrecv(pid_t sender, int num)
```

Il parametro num della funzione di ricezione indica con il segno se la chiamata e' bloccante o meno (numer positivo=chiamata bloccante, numero negativo=chiamata non bloccante). Il valore assoluto indica il numero d'ordine fra i messaggi in attesa di essere ricevuti. Ad esempio se num=3 la chiamata mpssrecv restituisce il terzo messaggio in attesa se vi sono almeno tre messaggi in attesa di essere consegnati dal mittente indicato, altrimenti attende che arrivi il terzo messaggio. Se num=-2 la funzione mpssrecv restituisce NULL se non vi sono almeno due messaggi in attesa di esser consegnati dal mittente specificato, altrimenti restituisce il secondo.

Esercizio g.1: Ci siano in un sistema tre processi P1 P2 e P3 identici che iniziano contemporaneamente. Tutti i processi computano per 4 ms, fanno I/O per 4 ms e poi elaborano nuovamente per 4 ms. L'I/O dei due processi avviene sulla stessa unita' con accesso FIFO. Il sistema usa uno scheduler di tipo round robin. Calcolare lo schedule per ogni scelta della lunghezza del quanto di tempo e quale sia la scelta (o siano le scelte) migliori. E quale è il limite massimo di durata dello schedule (considerando il time slice come numero intero positivo di millisecondi)?

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

- Perché durante la compattazione di memoria occorre sospendere l'esecuzione dei processi coinvolti?
- Perché nell'i-node non è presente il nome del file?
- Perché il microkernel è meno efficiente del Kernel monolitico? Perché è più flessibile e sicuro?
- Nella memorizzazione da parte dei processi utente delle capability per il controllo di accesso puo' essere usato un algoritmo di crittografia a singola chiave (chiave privata) oppure occorre un algoritmo a doppia chiave (chiave pubblica)? Perché?