

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
PROVA SCRITTA DI SISTEMI OPERATIVI
ANNO ACCADEMICO 2023/2024
22 luglio 2024

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 oddblock che ha le due seguenti procedure entry:

```
int oddstop(int key)
```

```
void evengo(void)
```

I processi che chiamano la oddstop si fermano. Detto $n(key)$ il numero di processi bloccati che hanno chiamato oddblock con parametro key, la funzione evengo deve riattivare tutti i processi che sono fermi per effetto di una oddstop per i quali $n(key)$ sia pari. oddstop deve restituire il valore $n(key)$.

Date le definizioni di questo testo ci possono essere casi di deadlock o di starvation? perché?

Esercizio c.2: Usando i semafori **implementare** la classe nas712 (not a semaphore 7 1/2).

```
class nas712:
```

```
    __init__(value)
```

```
    naP()
```

```
    naV()
```

nas712 si comporta come un semaforo ma ci possono essere al più 7 processi bloccati. Quando ci sono 7 processi bloccati una ulteriore naP non è bloccante e causa la riattivazione di tutti i (7) processi bloccati.

Esercizio g.1: Sia dato il seguente processo:

```
process P {
    char *a;
    int i;
    for (i=0; i<15; i++)
        a[i * 1024] = a[(i+2) * 1024];
}
```

In un sistema con pagine da 1K. Nell'ipotesi che sia il codice sia lo stack siano in segmenti non gestiti dalla memoria virtuale mostrare la lista dei riferimenti e calcolare il comportamento dell'algoritmo di rimpiazzamento MIN data una memoria di 3 frame? Quanti page fault vengono generati? E quale è il comportamento di MIN con 4 frame. Quanti page fault vengono generati?

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

- la ricostruzione della coerenza mediante tecnica di journaling risolve solo alcuni tipi di incoerenze, quali non vengono gestite?
- A cosa serve partizionare un disco? Fornire esempi pratici di uso di dischi partizionati.
- Il tempo rimanente calcolato dall'algoritmo SRTF (shortest remaining time first) può diventare negativo? (sì o no, perché?)
- I sorgenti del sistema operativo Linux possono essere scaricati e il kernel può essere ricompilato. In quali casi pensate sia necessario compilare un kernel specifico?