

link testo esame

- Esercizio c1
- Esercizio c2
- Esercizio g1 - Soluzione
- Esercizio g2

Esercizio c1

Una gara di lancio del giavellotto prevede che tutti gli atleti in gara facciano tre tiri. Lancia un atleta alla volta seguendo l'ordine numerico.

```
Atleta: process i=0..NATLETI-1:
    while giav.pronto(i)
        // lancia
        giav.lanciato(i);
Giudice: process
    while (i=giav.lanciofatto()) >= 0:
        m = misuralancio()
        giav.registraechiama(i, m)
        v = giav.classifica().
```

Il giudice di volta in volta attende in “lanciofatto” fino a quando il giocatore corrente non ha lanciato, misura la distanza del lancio e la registra, solo a quel punto il prossimo giocatore può lanciare. Scrivere il monitor giav.

```
int n;
#define true 1
#define false 0
#define NATLETI n

monitor giavellotto{
    list atleti = []; // atleti pronti a lanciare
    int lanci = 0; // contatore di lanci per atleta [max 3]
    condition atterrato; // il giavellotto è stato lanciato e il giudice può misurare
    condition pronto; // l'atleta è pronto a lanciare

    procedure entry pronto(int num_atleta)->true{
        atleti.append(num_atleta);
        if lanci == 3: // l'atleta ha finito i suoi lanci
            atleti.next(); // tocca al prossimo lanciare
        return true; //
    }
    procedure entry lanciato(int num_atleta){
        // segnala che è stato lanciato
        atterrato.signal();
        lanci++;
    }
}
```

```

}
procedure entry lanciofatto()->int num_atleta{
    if lanci == 0:
        atterrato.wait();
}
procedure entry registraechiama(int num_atleta, int misura){}
procedure entry classifica()->lista punteggi{}
}

// non è finito, è in pausa diciamo

```

Esercizio c2

Quali fra le seguenti funzioni possono essere usate al posto della test&set per realizzare un servizio di spinlock in un sistema nel quale la scrittura di una variabile globale non sia una funzione atomica?

- i) `foo(a,b,c): <if (random(0,1)) {a=b; b=c} else {c=b; b=a}>` (random(0,1) genera a caso un v
- ii) `bar(a,b): <x=a & 1; a = (a >> 1) | b << 31; b = x>`

In tutte le funzioni i parametri sono passati per indirizzo.

```

import math

def foo (a,b,c):
    if math.random(0,1):
        # è uscito 1
        a = b
        b = c
    else:
        # è uscito 0
        c = b
        b = a

def bar (a,b):
    x = a & 1
    a = (a >> 1) | b << 31
    b = x

```

Esercizio g1

Siano dati due processi in esecuzione in un sistema monoprocesso e gestiti da uno scheduler round-robin. I due processi sono gli unici nel sistema e usano la

stessa unità di I/O gestita in modo FIFO.

PA: 1ms CPU, 2ms I/O, 1ms CPU, 6ms I/O, 1ms CPU

PB: 2ms CPU, 1ms I/O, 6ms CPU, 1ms I/O, 1ms CPU

Quale è il tempo minimo e quale il tempo massimo impiegato dal sistema per completare l'elaborazione dei due processi al variare della lunghezza del quanto di tempo e della posizione iniziale dei processi nella ready queue (PA precede PB o viceversa).

Soluzione

risolto, vedi foglio Priorità: PA poi PB: - $t = 6\text{ms} \rightarrow \text{tot} = 17\text{ms} - t = 10\text{ms} \rightarrow \text{tot} = 12\text{ms}$

Priorità: PB poi PA: - $t = 6\text{ms} \rightarrow \text{tot} = 15\text{ms} - t = 10\text{ms} \rightarrow \text{tot} = 17\text{ms}$

Esercizio g2

Rispondere alle domande seguenti: - a) perché aumentando la dimensione dell'area di memoria secondaria usata dalla memoria virtuale si corre rischio di trashing? - b) cosa provoca la scelta errata della durata del quanto di tempo in uno scheduler round robin? - c) quali sono le condizioni necessarie per poter applicare l'algoritmo del banchiere - d) quali sono i vantaggi di usare una FAT rispetto ad una allocazione concatenata dove l'indirizzo del prossimo blocco dati viene memorizzato nel blocco dati stesso?

- a.
- b.
- c.
- d.