

Documenti autorizzati: due fogli manoscritti A4 fronte retro.

## 1 Esercizio 1

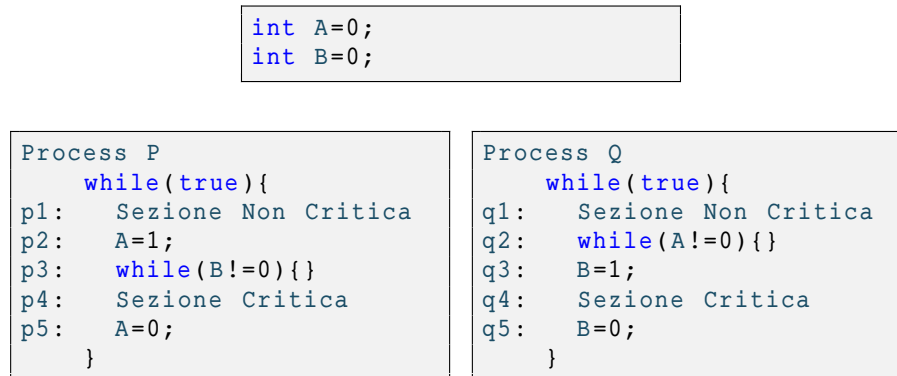


Figura 1: Algoritmo di mutua esclusione

Consideriamo l'algoritmo di mutua esclusione proposto alla figura 1.

1. Disegnare l'inizio dello diagramma degli stati per questo algoritmo (massimo 10 stati).
2. Questo algoritmo verifica la proprietà di mutua esclusione? Giustificare.
3. Questo algoritmo verifica la proprietà d'assenza di deadlock? Giustificare.
4. Questo algoritmo verifica la proprietà d'assenza di starvation? Giustificare.

## 2 Esercizio 2

Consideriamo un sistema fatto di due recipienti A e B e dei produttori e dei consumatori. Il recipiente A è di capacità a e il recipiente B è di capacità b dove a e b sono degli interi strettamente superiori a 0. I produttori eseguono in ciclo le azioni seguente:

- Se A e B sono pieni, aspettano.
- Se A è pieno ma non B, aggiungono un elemento a B altrimenti aggiungono un elemento a A.

I consumatori eseguono in ciclo le azioni seguente:

- Se A e B sono vuoti, aspettano.
- Se B non è vuoto, ritirano un elemento di B altrimenti ritirano un elemento di A.

Proponete in JAVA, C o pseudo-codice una modellazione di questo sistema, prendendo in considerazione che vogliamo un processo per produttore e un processo per consumatore, che prenderemo tre produttore e tre consumatore e che vogliamo evitare le attese attive. Per facilitare il lavoro, potete assumere che i recipienti sono rappresentati da variabile interi condivisi (produrre aumenta di 1 il valore e consumare lo sminuisce di 1).

## 3 Esercizio 3

Consideriamo l'algoritmo con message passing proposto sulla figura 2. Lo scopo di questo algoritmo è di garantire la mutua esclusione per due processi.

1. Descrivere in qualche righe come funziona questo algoritmo.
2. Questo algoritmo verifica la mutua esclusione? Giustificare.
3. Questo algoritmo verifica l'assenza di starvation? Giustificare e nel caso di una risposta negativa, proporre un ipotesi per avere l'assenza di starvation.
4. Come modificare l'algoritmo per avere un algoritmo di mutua esclusione per 3 processi (lavorando con l'ipotesi trovata nella domanda precedente)?

```

Process Pi per i uguale a 0 o 1
boolean B0=true; //variabile di P0
boolean B1=false; //variabile di P1

while(true){
p1: Sezione Non Critica
p2: while(Bi==false){}
p3: Sezione Critica
p4: Bi=false;
p5: send(ok)->P(1-i)
}

--Gestione dei messaggi di Pi per i uguale a 0 o 1
case ok: Bi=true;

```

Figura 2: Algoritmo di mutua esclusione con message passing

## 4 Esercizio 4

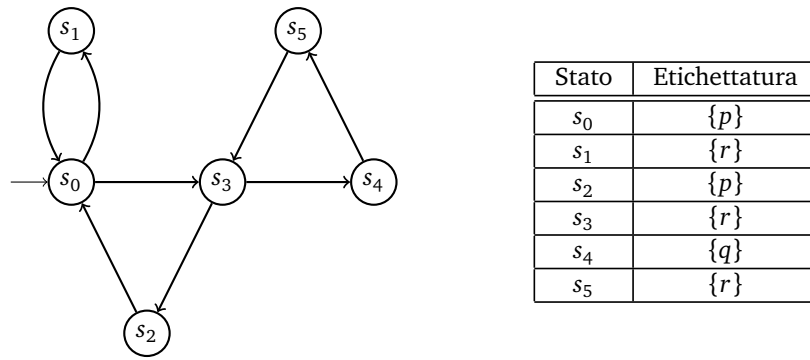


Figura 3: Struttura di Kripke  $\mathcal{S}_1$

Consideriamo la struttura di Kripke  $\mathcal{S}_1$  rappresentata sulla figura 3 (la tabella a destra indica per ogni stato l'insieme delle proposte atomiche associate) su l'insieme di proposte atomiche  $PA = \{p, q, r\}$ . Per ciascuna formula  $\phi$  di LTL che seguono, dire (giustificando) se sono soddisfatte dalla struttura (i.e. se abbiamo  $S_1 \models \phi$ ).

1.  $\phi = Fq$
2.  $\phi = GFr$
3.  $\phi = G(p \Rightarrow (Xr))$
4.  $\phi = (p \text{ U } r)$
5.  $\phi = FGp$