

— COMPITO CON SOLUZIONE —

Sistemi Operativi (modulo II / B)

5 Giugno 2012

Esercizio 1

Si considerino 3 processi P1, P2, P3 che si sincronizzano tramite i semafori contatori S1, S2, S3 inizializzati a 0:

```
process P1 {          process P2 {          process P3 {
    P(S1);              <B>                  <D>
    P(S2);              V(S1);              V(S2);
    <A>                  P(S3);              V(S3);
                        <C>
}                      }                  }
```

In quale ordine vengono eseguiti i blocchi di codice A,B,C,D? Elencare tutti i casi possibili spiegando brevemente (risposte senza spiegazione non verranno valutate).

I semafori sono tutti rossi quindi le P sono bloccanti. Di conseguenza sappiamo che:

A viene eseguito sempre dopo B e D (semafori S1 e S2),

C viene eseguito sempre dopo D (semaforo S3) e comunque dopo B.

Otteniamo quindi i seguenti casi:

BDAC
BDCA
DBAC
DBCA

Esercizio 2

Considerare la seguente soluzione **software** (errata) al problema della sezione critica

```
while (locked) {}
locked = true;

< sezione critica >

locked = false;
```

in cui la variabile condivisa **locked** è inizializzata a **false**. Mostrare una esecuzione con 2 processi in cui NON viene garantita la mutua esclusione, spiegando brevemente.

Supponiamo che il primo processo P1 superi il while ma venga interrotto prima di **locked = true**. Viene eseguito P2 che supera il while, assegna **locked = true** e entra nella sezione critica. Se P1 viene rischedulato prima che P2 sia uscito dalla sezione critica, P1 riassegna **locked = true** e entra in sezione critica rompendo la mutua esclusione.

Intuitivamente, il problema è la presenza di un test e un assegnamento della stessa variabile in due momenti successivi. Una possibile soluzione, infatti, è l'utilizzo di istruzioni macchina atomiche quali la **test&set**.