

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
 CORSO DI SISTEMI OPERATIVI - ANNO ACCADEMICO 2002/2003
 CONCORRENZA – 22 settembre 2003

Esercizio -1: essersi iscritti correttamente per svolgere questa prova.

Esercizio 0: Scrivere correttamente nome, cognome e n. di matricola prima di svolgere ogni altro esercizio seguente.

Esercizio 1: Un motore a quattro tempi ha un funzionamento in quattro fasi: aspirazione, compressione, scoppio e scarico. La camera di scoppio e' composta da un cilindro ed un pistone che scorre all'interno del cilindro. In testa ad ogni cilindro ci sono due valvole (di aspirazione e di scarico) e una candela. Durante la fase di aspirazione occorre aprire la valvola di aspirazione e il pistone scende, durante la fase di compressione il pistone sale con entrambe le valvole chiuse, durante la fase di scoppio la candela produce una scintilla e il pistone scende, durante la fase di scarico deve essere aperta la valvola di scarico e il pistone sale. Scrivere le sincronizzazioni fra pistone, valvole e candela sotto forma di **monitor**.

```
Pistone: process
while(true) {
    cyl.iniziaascendere()
    ...
    cyl.sceso()
    cyl.iniziaasalire()
    ...
    cyl.salito()
}
```

```
Valvola: array i=[a,s] of process {
while (true) {
    cyl.iniziaadaprire(i)
    ...
    cyl.aperta(i)
    cyl.iniziaachiodere(i)
    .....
    cyl.chiusa(i)
}
```

```
Candela: process {
while (true) {
    cyl.iniziaascintillare()
    /* scintilla */
    cyl.finescintilla()
}
```

Ci sono problemi di deadlock? Di starvation?

Esercizio 2:

P1: S1.V(); S1.V(); S1.P(); S1.P(); S2.P(); S2.P(); print(A); S1.V();S1.V()

P2: S1.P();print(B);S1.V();S1.P();print(C);S1.V();S2.V();

P3: S1.P();print(D);S1.V();S1.P();print(E);S1.V();S2.V();

Scrivere tutte le possibili sequenze generate dall'esecuzione del programma composto dai processi P1,P2 e P3.

I semafori sono tutti inizializzati a zero. Sono possibili casi di deadlock? Se si', quali? Se no, perche'?

Esercizio 3: Sia data una funzione $x=f(n,y)$ che atomicamente fa lo shift a destra di un numero binario. Il bit meno significativo prima dello shift (quello che verrebbe eliminato) viene restituito dopo l'operazione; il bit più significativo viene posto uguale a y. Può questa funzione essere usata come la test&set? In caso negativo, spiegare perchè; in caso positivo, illustrare il codice delle funzioni `cs_enter` e `cs_exit`.

