

UNIVERSITÀ DEGLI STUDI DI BOLOGNA – CORSO DI LAUREA IN INFORMATICA  
CORSO DI SISTEMI OPERATIVI A-L  
PRIMA PROVA PARZIALE – ANNO ACCADEMICO 2001/2002  
08 ottobre 2001

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

Esercizio 0: Scrivere correttamente il proprio nome, cognome e numero di matricola in ogni foglio prima di svolgere ogni altro esercizio seguente.

Esercizio 1: Per percorrere un tratto di strada a senso unico alternato le auto utilizzano il seguente protocollo:

```
altoneway.enter(direction)
..... attraversa la parte a senso unico alternato
altoneway.leave(direction)
```

`direction` è una variabile che può assumere valore 0 o 1 e rappresenta il senso di attraversamento del tratto di strada.

Più veicoli che intendono attraversare il senso unico alternato nella stessa direzione possono farlo contemporaneamente. Per poter ammettere un veicolo ad attraversare il senso unico alternato occorre che tutti i veicoli che lo hanno attraversato in senso contrario abbiano completato l'operazione.

Al fine di evitare problemi di starvation viene inoltre deciso che se un veicolo è in attesa si consente al più a 10 veicoli di passare in senso contrario.

Scrivere il monitor `altoneway`.

---

Esercizio 2a: Dato un meccanismo di message passing sincrono (dotato delle chiamate `ssend` `sreceive` viste a lezione) implementare un sistema di supporto per il message passing sincrono secondo Thoth con le chiamate `tsend`, `treceive` e `treply`. (ricordo che `treceive` non sblocca il processo che ha fatto la `tsend` corrispondente. È la `treply` a sbloccare il processo, `treply` spedisce anche un valore quale risposta al messaggio. `tsend` ha come parametri il processo destinazione e il messaggio da inviare e restituisce il valore specificato nella `treply` dal destinatario).

Esercizio 2b: Il servizio di message passing sincrono e di message passing secondo Thoth hanno lo stesso potere espressivo? (SI/NO)

Esercizio 2c: Motivare la risposta di 2b.

Esercizio 3:

Sia dato il seguente monitor:

```
monitor x
var c1, c2: condition;
procedure entry p1()
begin
    c1.wait;
    c2.signal;
end;

procedure entry p2()
begin
    c1.signal;
    c2.wait;
end;

begin
end.
```

Ponendo l'esistenza di più processi che chiamano la procedura p1 e di molteplici processi che chiamano p2 spiegare brevemente il comportamento del monitor.

Esercizio 4: Sia dato un sistema dove oltre alle azioni atomiche di lettura e scrittura in memoria sia data la seguente operazione atomica che opera su due variabili booleane:

```
f(a,b)=<if a {toggle(a);toggle(b)}>
```

dove la funzione `toggle` corrisponde alla negazione logica. Scrivere un meccanismo di supporto di sezioni critiche in modo simile al `test&set` usando la funzione `f`.