

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
 CORSO DI SISTEMI OPERATIVI - ANNO ACCADEMICO 2005/2006
 CONCORRENZA - 05 Luglio 2006

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

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

Esercizio 1:

```
monitor foo {
#define MAX 256
condition entercond[2];
int waiting[2];
int passing[2];
int direction= -1;

procedure entry enterfoo(int d)
{
    if (direction < 0) direction = d;
    waiting[d]++;
    if (passing[d] >= MAX || direction != d)
        entercond[d].wait();
    waiting[d]--; passing[d]++;
}

procedure entry exitfoo()
{
    passing[direction]--;
    entercond[direction].signal();
    if (passing[direction] == 0) {
        direction = 1 - direction;
        entercond[direction].signal();
        if (passing[direction] == 0)
            direction = -1;
    }
}
}
```

Questo monitor e' soggetto a starvation. Se ne scriva uno di funzionalita' equivalente ma privo del problema di starvation.

Esercizio 2:

Due mutex S e S' si dicono accoppiati quando vale l'invariante $S + S' = 1$. In particolare le P e le V su ciascun semaforo bloccano il processo invocante fino a quando non si raggiunge uno dei due seguenti stati:

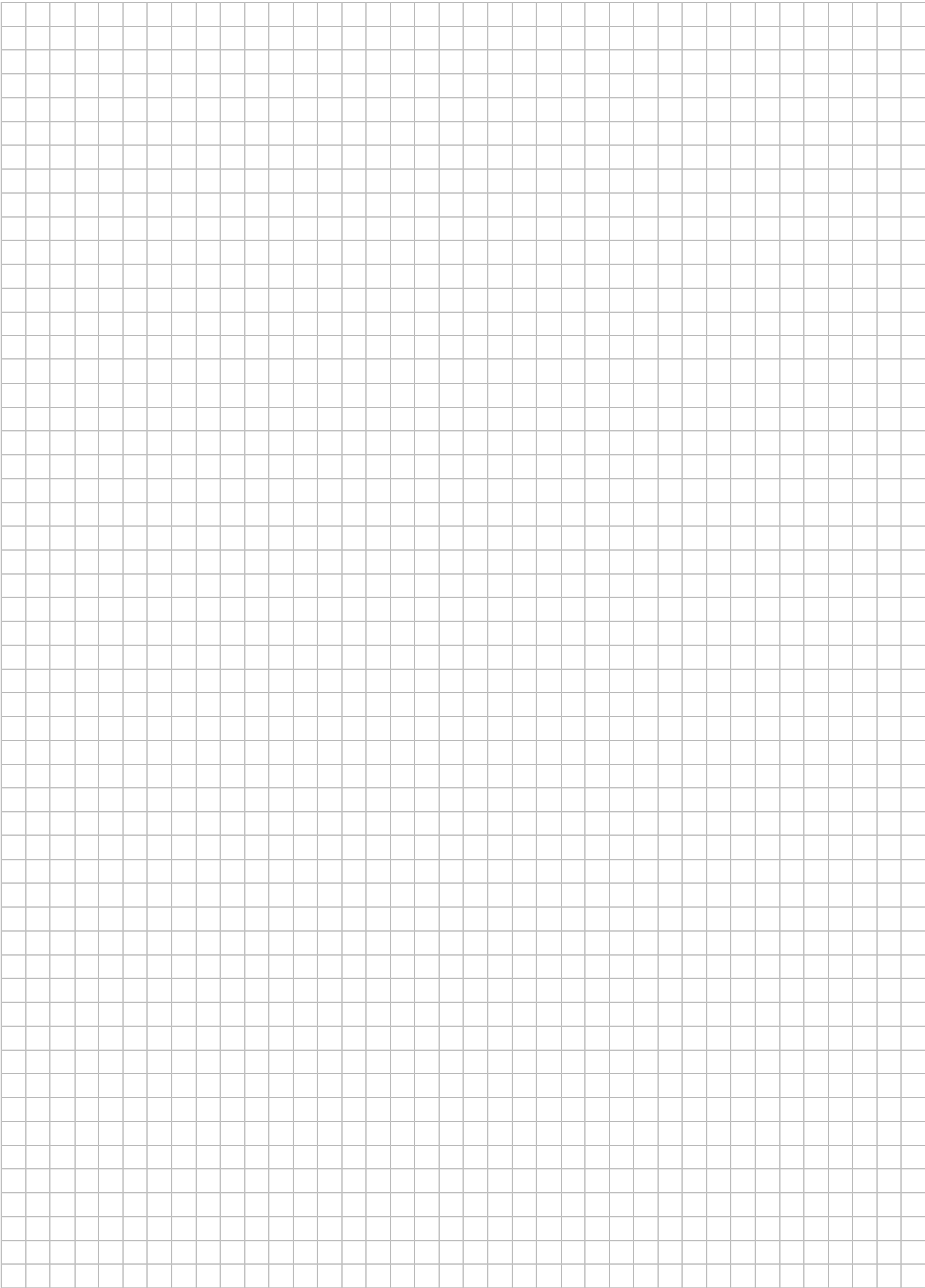
1. $S = 1, S' = 0$, un processo invoca P(S), un altro processo invoca V(S')
2. $S = 0, S' = 1$, un processo invoca V(S), un altro processo invoca P(S')

- a) Implementare un mutex tradizionale dati dei mutex accoppiati.
- b) Implementare due mutex accoppiati dati dei mutex tradizionali.

Esercizio 3:

La primitiva di comunicazione **msg sendreceive(int dest, msg messaggio)** viene utilizzata da un processo mittente per sincronizzarsi e scambiarsi un messaggio con il destinatario remoto. In particolare, la **res1 = sendreceive(P2,msg1)** effettuata da un processo P1 risulta bloccante fino a quando il processo P2 non cerca di effettuare una **res2 = sendreceive(P1,msg2)**; al risveglio dei due processi **res1** varra' **msg2** e **res2** varra' **msg1**.

Implementare la sendreceive utilizzando le usuali primitive di comunicazione sincrona.



UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
CORSO DI SISTEMI OPERATIVI - ANNO ACCADEMICO 2005/2006
PARTE GENERALE - 05 Luglio 2006

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

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

Esercizio 1:

Quali sono gli eventi rilevanti per un device driver per una periferica che operi in modalita' DMA? (e.g. quali sono gli eventi causati dal software o dall'hardware che pongono in esecuzione le funzioni di un device driver)

Per ogni evento identificato scrivere la pseudo codifica delle azioni svolte dal device driver per la gestione dell'evento.

Esercizio 2:

Sia data una memoria formata da 32KB inizialmente interamente non allocata.

Mostrare una (una sola!) sequenza di allocazione e deallocazione che contemporaneamente:

1. se gestita mediante paginazione con frame di 4KB provochi una perdita di spazio per frammentazione interna di 6KB (in totale)
2. se gestita mediante partizionamento dinamico con politica di allocazione first fit causi una perdita di spazio per frammentazione esterna sempre di 6KB

Esercizio 3:

Sia x l'ultima e y la penultima cifra del vostro numero di matricola. Rispondete alla domanda $(y*10+x+3)\%7$
0) Illustrate i concetti di scheduling preemptive e di scheduling cooperativo, evidenziando anche i vincoli posti sull'hardware dai due tipi di scheduling.

1) Spiegate i concetti di loading e linking dinamici

2) Descrivete concisamente l'algoritmo di scheduling SJF, inclusa la formula per il calcolo approssimato dei CPU burst.

3) Spiegate il concetto di Memory-Mapped I/O, evidenziando anche i vincoli posti sull'hardware e sul sistema operativo dal MMIO.

4) Descrivete le tecniche per verificare la coerenza di un file system.

5) Spiegare un livello RAID a vostra scelta, evidenziando le situazioni in cui e' opportuno adottare tale livello.

6) Descrivete i principali meccanismi per la realizzazione di directory basate su grafi aciclici.

