

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
CORSO DI SISTEMI OPERATIVI - ANNO ACCADEMICO 2005/2006
CONCORRENZA - 18 Settembre 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:

I Taxi che aspettano l'uscita dei passeggeri di fronte al terminal arrivi dell'aeroporto di Bologna sono disposti su quattro code. Quando un taxi arriva si colloca su una fila a caso. I passeggeri prendono il taxi dalla prima fila non vuota in ordine di vicinanza dall'uscita.

I tassisti chiamano la procedure entry

```
void blq.taxi-wait4cust(void)
```

i clienti

```
void blq.cust-wait4taxi(void)
```

Scrivere il monitor blq.

Esercizio 2:

N processi $P[0] \dots P[n-1]$ possono comunicare esclusivamente utilizzando la chiamata sincrona

```
msg exchange (msg m)
```

con le seguenti proprietà:

1. il messaggio inviato dal processo $P[i]$ tramite `exchange()` viene passato al processo $P[(i+1)\%N]$
2. i messaggi vengono consegnati a tutti i processi contemporaneamente

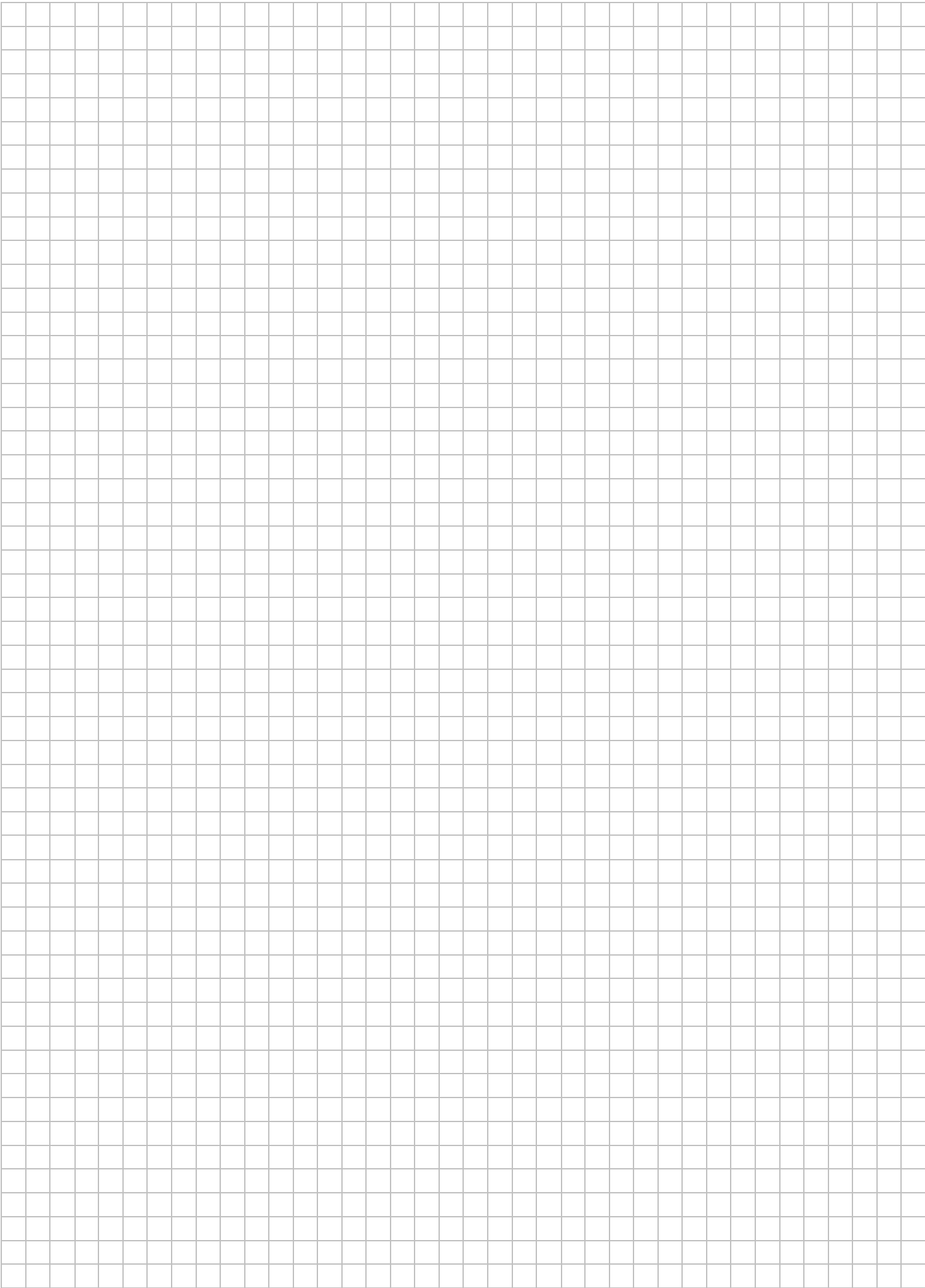
In altre parole, tutti i processi si debbono sincronizzare nell'invio di un messaggio al processo che li segue immediatamente in un ipotetico circolo.

Implementare la chiamata `exchange` utilizzando le usuali primitive di comunicazione asincrona.

Esercizio 3:

Con quali delle seguenti funzioni è possibile implementare un meccanismo di entrata e uscita da una sezione critica? Motivare la risposta con un frammento di codice o una spiegazione per l'impossibilità.

1. $f1(x,y) = \langle x|=y ; y\&=x \rangle$
2. $f2(x,y) = \langle y\&=x ; x|=y \rangle$
3. $f3(x,y) = \langle x1=x\&y ; y1=x|y ; x=x1 ; y=y1 \rangle$



UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
 CORSO DI SISTEMI OPERATIVI - ANNO ACCADEMICO 2005/2006
 PARTE GENERALE - 18 Settembre 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:

Si consideri il seguente scenario per un banchiere multivaluta.

(a) Per quali valori di x lo stato e' safe? Motivare la risposta.

(b1) Scegliete un valore x per il quale lo stato sia safe e mostrate una sequenza di allocazioni che porterebbe il sistema in uno stato unsafe in due mosse (allocazioni). Cosa accade in un sistema reale se tale sequenza e' ricevuta dal banchiere?

(b2) Scegliere un valore x per il quale lo stato sia unsafe e mostrare una sequenza di deallocazioni che porterebbe il sistema in uno stato safe in due mosse (deallocazioni).

Valuta 1

COH=5

Valuta 2

COH=25

i	Ci Pi Ni			Ci Pi Ni		
	Max	Curr.	Residuo	Max	Curr.	Residuo
1	15	5	10	30	5	25
2	12	6	6	35+x	x	35
3	7	2	5	20	10	10
4	3	1	2	75	30	45

Esercizio 2:

Un algoritmo di rimpiazzamento sceglie la prima pagina non acceduta in scrittura secondo la politica LRU.

Se tutte le pagine sono state modificate sceglie fra queste utilizzando la politica LRU.

Rispondere alle domande seguenti:

- Indicare almeno un pro e un contro del metodo
- Data la seguente stringa di riferimenti mostrare il funzionamento dell'algoritmo (situazione della memoria ad ogni istante e page fault). La memoria ha 3 frame.
- [punteggio extra] E' a stack?

Esercizio 3:

Sia x l'ultima e y la penultima cifra del vostro numero di matricola. Rispondete alla domanda

$(y*10+x+3)\%4$

0. Spiegare in dettaglio tutto quello che accade in risposta a una richiesta di accesso a una pagina non presente in memoria.
1. Spiegare in dettaglio tutto quello che accade in risposta a una richiesta di scrittura di una parola su un file risiedente su disco rigido gia' aperto in precedenza.
2. Spiegare in dettaglio tutto quello che accade quando un processo in ambiente Unix chiede di aprire in modalita' append il file "/mnt/floppy/README.txt". Supporre che in precedenza il filesystem di un floppy disk sia stato mounted tramite il comando "mount /dev/hdb1 /mnt".
3. Spiegare in dettaglio la differenza fra thread e processi valutandone pregi e difetti. Soffermarsi sulle problematiche di implementazione dei thread nei sistemi operativi.

