

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
CORSO DI SISTEMI OPERATIVI - ANNO ACCADEMICO 2007/2008
CONCORRENZA – 30 gennaio 2009

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: Scrivere un monitor sbb che implementi un buffer limitato di caratteri.

Il monitor deve implementare le seguenti procedure entry:

```
int sbb.enqueue(char *s);
```

```
int sbb.dequeue(char *buf, len l);
```

Il buffer contiene MAX caratteri. Le stringhe passate alla enqueue sono classiche stringhe del linguaggio C con il carattere nullo come terminatore e devono avere lunghezza minore di MAX. Nel caso la stringa s abbia lunghezza maggiore o uguale a MAX la stringa non viene caricata nel buffer. Il valore di ritorno e' la lunghezza della stringa inserita nel buffer, -1 in caso di errore.

La dequeue deve restituire una stringa alla volta scrivendola nel buffer. Il parametro l contiene l'ampiezza del buffer (in byte in modo simile alla system call "read").

Se la stringa non puo' essere memorizzata interamente nel buffer perche' il buffer e' di lunghezza insufficiente viene tornato un errore (-1), in modo che il processo chiamante provveda a chiamare nuovamente la funzione specificando un buffer di ampiezza maggiore. (N.B. In caso di errore di buffer insufficiente la stringa non deve essere perduta!).

Il valore di ritorno e' la lunghezza effettiva della stringa presa dal buffer.

Esercizio 2 (Test&Set): Mostrare se, e nel caso come, le seguenti funzioni possano essere usate per implementare sezioni critiche.

```
boh1(bool x, bool y) = < y = x = !x >
```

```
boh2(int x, int y) = < xx = x, yy = y, x = (xx + yy) % 3, y = (xx * yy) % 3 >
```

```
boh3(int x, int y) = < y = (x * y) % 3, x = y % 2 >
```

Esercizio 3: Scrivere un servizio di message passing sincrono (quello classico che consenta alla **srecv** di aspettare un messaggio da un utente specifico o da tutti *****) sulla base di un servizio sincrono **ssendx srecvx**, dove la **srecvx** non ha parametri e riceve FIFO tutti i messaggi (come se fosse **srecv** *****).

UNIVERSITA' DEGLI STUDI DI BOLOGNA - CORSO DI LAUREA IN INFORMATICA
CORSO DI SISTEMI OPERATIVI - ANNO ACCADEMICO 2007/2008
PARTE GENERALE – 30 Gennaio 2009

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:

Disegnare un grafo di holt (connesso) che dopo due passi di riduzione risulti irriducibile.

Quale e' il numero minimo di processi e di classi di risorse necessari per creare questa situazione?

Quale e' il numero di nodi del knot?

Esercizio 2: Sia data una memoria con 4 frame: scrivere una stringa di riferimenti infinita tale che LRU e MIN coincidano, cioe' lo stato della memoria in ogni istante sia lo stesso sia che venga utilizzato LRU sia che venga utilizzato MIN.

La stringa di riferimenti infinita deve generare infiniti page fault.

Esercizio 3: Dimostrare il seguente Teorema:

Se un algoritmo di rimpiazzamento e' a stack, allora non puo' essere soggetto alla anomalia di Belady.



