## Università di Napoli Federico II Corso di Laurea in Ingegneria Informatica e Elettronica Esame di Sistemi Operativi Proff. De Carlini, Cotroneo, Cinque

## Prova pratica del 13/09/2010 Durata della prova: 150 minuti

Cognome	Nome	Matr	••••••
_			

Lo studente legga attentamente il testo e produca il programma, il makefile, ed i casi di test necessari per dimostrarne il funzionamento. La mancata compilazione dell'elaborato, la compilazione con errori o l'esecuzione errata del programma daranno luogo alla valutazione come **prova non superata**. Ricordarsi di indicare Nome, Cognome e matricola su questo stesso foglio, che dovrà essere in ogni caso consegnato alla Commissione. Al termine della prova lo studente dovrà fare verificare il funzionamento del programma ad un membro della Commissione.

## Testo della prova

Si realizzi in linguaggio C/C++ un'applicazione **multiprocesso** per la simulazione di un magazzino per lo stoccaggio di merci. L'applicazione è costituita da 20 processi, 10 *fornitori* e 10 *clienti*. Il magazzino è rappresentato da un vettore di 100 elementi, ognuno contenente la seguente struttura **scaffale**:

```
typedef struct {
     unsigned int id_fornitore;
     unsigned int stato;
} scaffale;
```

dove id\_fornitore è il PID dell'ultimo fornitore che ha usato lo scaffale (inizialmente 0), e stato indica se lo scaffale è *libero* (0), *occupato* (1), o *in uso* (2), ovvero correntemente usato da un fornitore o cliente. Una variabile globale livello\_scorte indica il numero di prodotti presenti nel magazzino (livello max: 100 prodotti, inizialmente 0).

Ognuno dei processi (fornitori e clienti) effettua 15 accessi al magazzino, uno ogni secondo. Ad ogni accesso, il fornitore stabilisce se c'è spazio per effettuare la fornitura, in base al livello\_scorte, altrimenti si sospende in attesa che qualche cliente liberi spazio. In seguito, pone in uso il primo scaffale libero che trova ed effettua la fornitura ponendo a occupato lo stato dello scaffale e il proprio PID in id\_fornitore (si simuli una durata di 2 secondi per la fornitura, utilizzando una sleep()). Infine, aggiorna il livello scorte.

Allo stesso modo, il cliente stabilisce se ci sono prodotti, in base al livello\_scorte, altrimenti si sospende in attesa di fornitori. In seguito, pone in uso il primo scaffale occupato che trova ed effettua l'acquisto (si simuli una durata dell'acquisito di 2 secondi). Infine pone a libero lo stato dello scaffale, pone a 0 la variabile id\_fornitore, e aggiorna il livello\_scorte.

Il magazzino e la variabile livello\_scorte sono variabili condivise, e l'accesso a tali variabili da parte dei processi deve essere disciplinato attraverso il costrutto **Monitor**.

I processi fornitori e clienti sono generati dal programma principale attraverso la primitiva fork(). Una volta generati i processi, il programma principale ne attende la terminazione e termina a sua volta.