

SISTEMI OPERATIVI IIN/IEL/IDT
INFORMATICA INDUSTRIALE E SISTEMI OPERATIVI IDI
SISTEMI DI ELABORAZIONE P.O.
prova scritta del 22.06.2005

Nome: _____

Cognome: _____

Sia data una classe Sottoscrittore che definisce un thread come descritto nel seguito:

```
public class Utente extends Thread {
    private Erogatore e = null;
    public Utente ( Erogatore e ) { this.e = e; }
    /*    ...    completare se necessario    ...    */
    public void run() {
        /*    ...    completare se necessario    ...    */
        int i = 0;
        while ( true ) {
            i++;
            Object o = e.richiedi( i );
        }
    }
}
```

Come è possibile evincere dal codice riportato sopra, l'utente chiede ad un erogatore oggetti con un identificativo specificato, sempre crescente.

Tenendo presente che sul sistema in esame sono attivi N thread di tipo Utente, si realizzi la classe Erogatore in maniera tale che:

- eroghi l'oggetto i -esimo a tutti gli utenti che ne fanno richiesta;
- prima di procedere all'erogazione dell'oggetto $(i+1)$ -esimo, deve aver erogato l'oggetto i -esimo a tutti gli utenti.
- gli oggetti creati dall'erogatore siano accodati per la distribuzione;
- gli utenti che richiedono un oggetto che non è ancora in erogazione devono essere sospesi.

Soluzione

```
public class Erogatore {
    public Erogatore( int utenti ) {
        this.utenti = utenti;
    }

    public synchronized Object richiedi( int i ) {
        if ( i > inDistribuzione ) {
            try {
                wait();
            } catch ( InterruptedException ie ) {}
        }
        Object o = oggetto;
        distribuzioniEffettuate++;
        if ( utenti == distribuzioniEffettuate ) {
            inDistribuzione++;
            oggetto = new Object();
            distribuzioniEffettuate = 0;
            notifyAll();
        }
        return o;
    }

    private int inDistribuzione = 0;
    private Object oggetto = null;
    private int distribuzioniEffettuate = 0;
    private int utenti;
}
```