

Esercizio 1 Programmazione concorrente

Si consideri il codice dato.

Una classe "Deposito" gestisce due depositi di risorse: un deposito DA di risorse di tipo A e uno DB di risorse di tipo B. Ciascun deposito ha capienza MC=10.

Un certo numero di thread "client" creati dal main utilizza i depositi mediante il metodo switchElement messo a disposizione da un oggetto di classe Deposito. Ciascun client richiede iterativamente una operazione che richiede o di trasformare una risorsa di tipo A in una di tipo B (eliminando una risorsa dal deposito DA e aggiungendo una risorsa al deposito DB) o di trasformare una risorsa di tipo B in una di tipo A (eliminando una risorsa dal deposito DB e aggiungendo una risorsa al deposito DA).

Il codice dato non effettua alcuna sincronizzazione tra thread.

Si richiede di modificare il codice dato in modo che un client venga bloccato quando l'operazione richiesta non e' possibile, cioe' quando il deposito da cui prelevare e' vuoto o il deposito in cui aggiungere e' pieno.

Non si richiede che il codice prodotto garantisca l'assenza di deadlock. Tuttavia, se il deadlock e' possibile, occorre spiegare perche', e come si potrebbe evitare.

Si ricorda che bisogna caricare i file .java, NON i .class, meglio se raggruppati in un unico file zip.

Esercizio 2 Programmazione distribuita con socket

Si consideri il codice prodotto come soluzione all'esercizio di programmazione concorrente.

Si modifichi in modo che i client e il server siano processi che possono girare su macchine diverse.

Si utilizzino i socket per la comunicazione.

Si ricorda che bisogna caricare i file .java, NON i .class, meglio se raggruppati in un unico file zip.

Esercizio 3 Programmazione distribuita con RMI

Si consideri il codice prodotto come soluzione all'esercizio di programmazione concorrente.

Si modifichi in modo che i client e il server siano processi che possono girare su macchine diverse.

Si utilizzi RMI per la comunicazione.

Si ricorda che bisogna caricare i file .java, NON i .class, meglio se raggruppati in un unico file zip.