

Realizzare in Java il seguente sistema:

- Sono presenti 4 risorse di tipo A, 7 risorse di tipo B e 4 risorse di tipo C accessibili in modo esclusivo a N thread Master (N chiesto in input).
- Sono presenti 6 thread Worker utilizzabili dai Master per svolgere alcune operazioni, ogni Worker è in attesa della richiesta da parte di un thread Master, elabora la richiesta e produce un risultato entro 500 millisecondi, quindi incrementa una variabile che indica quanti servizi ha offerto e quindi rientra in attesa di richiesta.
- Il thread Master periodicamente acquisisce una risorsa di tipo A ed una di tipo C e due risorse di tipo B (se disponibili) o se non disponibile una sola risorsa B (gli vengono concesse risorse solo se tutte le risorse richieste sono disponibili altrimenti attende) quindi richiede due thread Worker per elaborare i dati (attende se tutti i thread sono occupati), fa 'sbloccare' i due thread Worker (che opereranno in parallelo) e attende la notifica della generazione del risultato, quindi rilascia le risorse acquisite ed incrementa una variabile che indica il numero di cicli completati dal singolo thread Master.
- Il programma principale fa partire gli N thread quindi attende 1 minuto e termina i thread Master, ed i thread Worker (tramite metodo *interrupt()*) e stampa il numero di iterazioni completate da ogni Master ed il numero di servizi svolti da ogni Worker, si deve garantire che le risorse acquisite dai Master siano rilasciate quando vengono interrotti e in caso di un qualsiasi altro fallimento.

Implementare il sistema usando i metodi **sincronizzati**.