

Note sullo svolgimento della prova

- Non è consentito l'uso di vecchi testi d'esame, libri, o appunti.
- Utilizzare NetBeans per editare, compilare ed eseguire i programmi.
- Creare un file **Cognome_matricola.zip** contenente i soli file *.java* da consegnare.
- Al termine della prova, dopo il caricamento degli elaborati, il docente mostrerà una possibile soluzione. Dopo aver visto la soluzione, gli studenti avranno la possibilità di riprendere il proprio elaborato.

Esercizio

Per creare un prodotto finito sono necessari un pezzo di tipo X e un pezzo di tipo Y. Un deposito può contenere pezzi di entrambi i tipi. Ogni pezzo è caratterizzato da un livello di qualità, espresso come un numero compreso tra 0 e 1. Il deposito ha una capienza limitata. Un primo produttore immette nel deposito solo pezzi di tipo X con un livello di qualità a caso. Un secondo produttore immette nel deposito solo pezzi di tipo Y, sempre con un livello di qualità a caso. Un consumatore estrae pezzi dal deposito solo quando può estrarre contemporaneamente un pezzo di tipo X e un pezzo di tipo Y. Un controllore esegue periodiche operazioni di verifica della qualità. Quando viene eseguito un controllo tutti i pezzi contenuti nel deposito che hanno un livello di qualità minore di 0.2 vengono eliminati. Realizzare una classe *Deposito* dotata almeno dei seguenti costruttori e metodi:

- *Deposito(int dim)* crea un deposito in grado di contenere al più *dim* pezzi.
- *void inserisci(Pezzo p)* inserisce il pezzo *p* nel deposito. Il metodo è bloccante nel caso in cui nel deposito non ci sia spazio oppure se è rimasto un solo posto disponibile e i pezzi attualmente contenuti sono tutti dello stesso tipo del pezzo che deve essere inserito (per esempio se sto inserendo un pezzo di tipo X, c'è un solo posto disponibile e ci sono solo pezzi di tipo X).
- *Pezzo[] estrai()* estrae simultaneamente dal deposito un pezzo di tipo X e un pezzo di tipo Y. L'estrazione avviene secondo una logica FIFO compatibilmente con il precedente vincolo. Il metodo è bloccante nel caso in cui non sia possibile estrarre i due pezzi. I due pezzi vengono restituiti con il valore di ritorno.
- *int controlla()* rimuove dal deposito tutti i pezzi con un livello di qualità minore di 0.2. Restituisce il numero di pezzi rimossi.

Realizzare anche le classi *Pezzo*, *Produttore*, *Consumatore* e *Controllore*. La classe *Controllore* deve eseguire un'operazione di controllo sul deposito ogni secondo.