Programare orientată pe obiecte – Test de laborator

Seria 13 – 27 mai 2015

Enunț

Un magazin și-a propus să implementeze un program C++, orientat pe obiecte, cu ajutorul căruia să își gestioneze produsele. Fiecare produs comercializat de magazin are o unitate de măsura care poate fi bucată, greutate sau volum. Unele produse pot fi perisabile (au o perioadă de valabilitate). Unele produse pot fi în promoție (la prețul de vânzare se aplică un discount – procent din preț).

Magazinul primeste produsele sub forma de loturi. Un lot reprezintă o anumită cantitate dintr-un produs, primită la o anumită dată de magazin și care trebuie vândută la un anumit preț per unitatea de măsură a produsului. Două loturi diferite din același produs pot veni în magazin în cantități diferite sau la prețuri diferite. Produsele perisabile au în plus o perioadă de valabilitate care poate fi diferită pentru fiecare lot în parte și se calculează de la data intrării lotului în magazin. Produsele în promoție au în plus un procent de discount care poate fi diferit pentru fiecare lot în parte și se aplică prețului cu care a intrat în magazin acel lot din produs.

De exemplu:

* pe data de 03.05.2015, magazinul primește un lot de 150 kg de cireșe, cu termen de valabilitate de 5 zile, la prețul de 20 lei pe kg
* pe data de 05.05.2015, magazinul primește un lot de 500 de căciuli, la prețul de 15 lei bucata
* pe data de 16.05.2015, magazinul primește un lot de 300 kg de cireșe, cu termen de valabilitate de 3 zile, la prețul de 15 lei pe kg, la care decide să aplice un discount de 20%
* pe data de 19.05.2015, magazinul primește un lot de 100 de litri de rom, la prețul de 25 de lei pe litru, la care decide să aplice un discount de 10%.

Progamul trebuie să citească de la tastatură loturile de produse cu toate informațiile aferente, câte unul pe rând, în formatul cel mai convenabil, să creeze o structură de date în care să le adauge și să le afișeze împreună cu toate informațiile lor curente, în funcție de opțiunea vânzătorului.

Un vânzător din magazin poate să opteze între:

1. a adăuga un produs nou
2. a afișa toate produsele pe care magazinul le poate comercializa
3. a adăuga un lot nou dintr-un produs
4. a afișa toate loturile de produse care au fost primite de magazin între două date calendaristice date
5. a afișa toate loturile de produse care pot fi comercializate de magazin la data curentă (care sunt nevide și care nu sunt expirate)
6. a vinde o anumită cantitate cerută dintr-un produs pe care îl are în magazin.

Când primește o cerere de vânzare a unui produs, vânzătorul trebuie să verifice dacă are în magazin cel puțin un lot din acel produs, în cantitate cel puțin egală cu cererea și care nu a depășit data de valabilitate. Dacă în magazin există mai multe loturi din acel produs, în cantități mai mari sau egale decât cantitatea cerută și care sunt în termenul de valabilitate, atunci magazinul este obligat să vândă din lotul mai ieftin. Dacă vânzarea se poate face, aplicația trebuie să afișeze prețul total pentru cantitatea solicitată. Dacă există mai multe loturi din acel produs care pot fi comercializate, dar niciunul nu este într-o cantitate mai mare sau egală decât cea solicitată, atunci programul trebuie să afișeze cantitățile disponibile în fiecare lot comercializabil, cu prețul corespunzător. O vânzare nu se poate face din mai multe loturi disponibile; cantitatea vândută o dată trebuie să fie dintr-un singur lot. După finalizarea vânzării, cantitatea vândută se scade din cantitatea disponibilă în stocul din care s-a făcut vânzarea.

Precizări

1. Timpul de lucru este de 90 de minute.
2. La sfărșitul timpului de lucru, studenții vor salva pe stick-ul de memorie al profesorului supraveghetor fișierul sursă cu extensia cpp. Acesta trebuie să conțină pe primul rând un comentariu cu numele și prenumele studentului, grupa și compilatorul folosit.
3. Sursa predată trebuie să compileze. Sursele care au erori de compilare nu vor fi luate în considerare. Înainte de predarea surselor, studenții vor pune în comentariu eventualele părți din program care au erori de compilare sau nu funcționează corespunzător.
4. Se acceptă și soluții parțiale, care nu respectă toate cerințele din enunț, dar sunt funcționale. Acestea vor fi depunctate corespunzător.
5. În implementarea programului se vor utiliza cât mai multe dintre noțiunile de programare orientată pe obiecte, care au fost studiate pe parcursul semestrului și care se potrivesc cerințelor din enunț.
6. Condițiile minimale de promovare a testului sunt ca programul să fie scris cu clase, să citească produsele unul câte unul, să le memoreze și apoi să le afișeze cu informațiile aferente.
7. Orice tentativă de fraudă se va pedepsi conform regulamentelor Universității.

Barem

1. Se acordă 1 punct din oficiu.
2. Dacă sursa nu compilează, nu se mai acordă niciun alt punct (se acordă nota 1).
3. Se acordă 4 puncte pentru respectarea condițiilor minimale, precizate în enunț, astfel:
   1. 1 punct dacă produsele sunt definite prin clase
   2. 1 punct dacă produsele sunt citite corect, cu toate informațiile aferente
   3. 1 punct dacă produsele sunt memorate corect într-o listă sau într-o altă structură
   4. 1 punct dacă produsele sunt afișate corect, cu toate informațiile aferente

Pentru oricare dintre cerințele a., b., c. și d., dacă cerința este îndeplinită parțial, se acordă 0,5 puncte.

1. Se acordă câte 0,5 puncte pentru rezolvarea corectă a cerințelor 1, 2, 4 și 5 din enunț (cerința 3 este complet inclusă în condițiile minimale de mai sus, deci pentru ea nu se acordă puncte suplimentare). Nu se acordă punctaje parțiale.
2. Se acordă 2 puncte pentru rezolvarea corectă a cerinței 6 din enunț (vânzarea), astfel:
   1. 0,5 puncte dacă vânzarea se face dintr-un lot de produs valabil (care conține o cantitate cel puțin egală cu cea solicitată și care este în termenul de valabilitate)
   2. 0,5 puncte dacă în cazul existenței mai multor loturi de produs valabile, vânzarea se face din cel de preț minim
   3. 0,5 puncte dacă vânzarea se finalizează corect, adică prețul total și cantitatea rămasă în lot după vânzare sunt calculate corect
   4. 0,5 puncte dacă mesajele afișate în cazul în care nu se poate face vânzarea sunt corecte (produsul nu este disponibil, respectiv produsul este disponibil în cantitatea x, mai mică decât cea solicitată, cu x maximal, cf. cerinței din enunț).

Nu se acordă punctaje parțiale.

1. Se acordă 1 punct pentru definirea unei ierarhii de clase (produs -> produs perisabil, produs în promoție, produs perisabil în promoție). Dacă cerința este îndeplinită parțial, se acordă 0,5 puncte.
2. Se scade 1 punct pentru utilizarea incorectă a constructorilor și destructorilor. Dacă cerința este încălcată parțial, se scad 0,5 puncte.
3. Se scade 1 punct pentru utilizarea incorectă a încapsulării (câmpurile private și metode publice). Dacă cerința este încălcată parțial, se scad 0,5 puncte.
4. Se acordă 1 punct pentru definirea corectă a unei ierarhii de clase de tip romb (moștenire multiplă și virtuală). Nu se acordă punctaje parțiale.
5. Se acordă 1 punct pentru folosirea corectă în contextul dat a supraîncărcării operatorilor. Nu se acordă punctaje parțiale.
6. Se acordă 1 punct pentru folosirea corectă în contextul dat a claselor șablon. Nu se acordă punctaje parțiale.
7. Se acordă 1 punct pentru folosirea corectă în contextul dat a metodelor virtuale. Nu se acordă punctaje parțiale.

Observații de evaluare

1. Punctele de la 7. și 8. nu se scad din cele acordate pentru îndeplinirea condițiilor minimale. În consecință, dacă sunt respectate toate condițiile minimale, nota nu poate scădea sub 5.
2. Nota maximă este 12. Punctele care sunt peste se anulează.
3. Erorile minore, care sunt cauzate de neatenție și care nu se repetă în mai multe locuri (de ex., a uitat să scrie un caracter și nu îi compilează, a greșit un semn sau un coeficient într-o formulă, nu afișează primul sau ultimul element dintr-o listă etc) pot fi corectate și trecute cu vederea (se poate acorda punctajul maxim prevăzut pentru o cerință neîndeplinită corect din cauza unei erori minore).