Programare orientată pe obiecte – Test de laborator

Restanta – 8 septembrie 2016

**Enunț**

Este un fapt binecunoscut că în SUA, după ”Labor Day” (prima zi de luni din septembrie) toată lumea își îndreaptă atenția către Halloween, Thanksgiving și, evident, Craciunul. În acest sens, fiecare companie, indiferent de domeniu, încearcă să-și prezinte oferta în modul cel mai atractiv (și profitabil).

Compania aviatică ”Wings – XYZ”, activă până în prezent doar în America de Nord și America Latină, a obținut licentă pentru zboruri Trans - Atlantice, astfel că, începând cu 8 septembrie, va avea curse regulate către Europa. Fiind într-o perioadă de monitorizare, pentru a fi sigură că toate cerintele sunt indeplinite, compania și-a propus să implementeze un program C++, orientat pe obiecte, cu ajutorul căruia să își gestioneze cursele de linie.

Fiecare zbor este caracterizat de un ID (ex. ”W123”), orașul și ziua plecării, destinația și durata. În preajma sărbătorilor importante (cum ar fi cele 3 anterior mentionate), compania poate să decidă suplimentarea numărului de zboruri, introducând curse temporare pe segmentele cele mai cerute (de exemplu, pe ruta New York – Londra). Evident, aceste curse sunt active pe o perioadă scurtă, maxim 3 luni. Compania poate, deasemenea, decide un discount (procent din preț) pentru unele zboruri, dacă biletele sunt achizitionate în anumite termene și condiții.

Programul C++ este construit astfel încât să ruleze în cloud și, implicit, să fie cât mai eficient posibil. În acest sens, ”Wings – XYZ” a găsit metoda de a își gestiona cursele folosind containere. Fiecare container stochează un anumit segment de zbor (rută), numărul de curse care-l efectuează și prețul unui bilet la clasa Economic. Cursele speciale (temporare) trebuie să prevadă și perioada efectuării care e diferită pentru fiecare segment de zbor în parte și care se calculează de la inregistrarea/adaugarea zborului în container. Pentru cursele cu bilete speciale, procentul de discount se aplică prețului înregistrat per segment de zbor.

De exemplu:

* pe data de 01.10.2016, se inregistrează pe ruta New York – Amsterdam, cursa temporară, cu o perioadă activă de 15 zile, cu prețul de 800 dolari / bilet.
* pe data de 15.09.2016, se inaugurează ruta Chicago – Varsovia, cu prețul de 1328 dolari / bilet;
* pe data de 30.10.2016, se inregistrează pe ruta New York – Amsterdam, cursa temporară de Halloween, cu o perioadă activă de 3 zile, la care compania decide să aplice un discount de 20%;
* pe data de 15.12.2016, se suplimentează ruta Chicago – Varsovia cu încă o cursă, pe o perioada de 1 lună.

Progamul trebuie să citească de la tastatură rutele (containerele care contin zborurile efectuate de companie), 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 departamentului IT. Acesta poate să opteze între:

1. a adăuga o rută nouă;
2. a afișa toate rutele gestionate de ”Wings – XYZ”;
3. a adăuga o cursă pe o anumită rută;
4. a afișa toate cursele care au fost inregistrate între două date calendaristice date
5. a afișa toate cursele active la data curentă;
6. a afisa toate rutele care nu au avut niciodată discount.

**Precizări**

1. Timpul de lucru este de 90 de minute.
2. Condițiile minimale de promovare a testului sunt ca programul să fie scris cu clase, să citească toate zborurile dintr-o anumita ruta, unul câte unul, să le memoreze și apoi să le afișeze cu informațiile aferente.
3. La sfarșitul 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.
4. 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.
5. Se acceptă și soluții parțiale, care nu respectă toate cerințele din enunț, dar sunt funcționale. Acestea vor fi depunctate corespunzător.
6. 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.
7. Orice tentativă de fraudă se va pedepsi conform regulamentelor Universității.