EXEMPLU EXAMEN FINAL

JAVA 1 Associate

Un centru de inchirieri vehicule ofera doua mijloace de deplasare: masina si bicicleta.

Un vehicul are o marca, un pret de inchiriere si o disponibilitate(disponibil pentru inchirie sau nu).

O masina este un vehicul cu un tip de combustibil.

O bicicleta este un vehicul cu un echipament de protectie.

Avem un singur centru de inchirieri care poate contine mai multe vehicule(masini si biciclete). Centrul de inchirieri are urmatoarele metode:

         **adaugaVehicul** – adauga un nou vehicul in centrul de inchirieri;

         **afiseazaVehicule** – afiseaza toate vehiculele din centru;

         **afiseazaVehiculeDisponibile** – afiseaza toate vehiculele disponibile din centru;

         **stergeVehicul** – sterge un vehicul dupa un anumit index primit;

         **inchiriereRapida** – primeste ca parametru o marca si returneaza un vehicul ce are marca respectiva. Daca nu exista niciun vehicul disponibil, cu acea marca se arunca o exceptie creata de programator(MarcaIndisponibilaException);

Creati o aplicatie care primeste urmatoarele comenzi din consola:

         **AB** marcaBicicleta pret disponibilitate echipamentProtectie

o   Comanda ce adauga o bicicleta in centrul de inchirieri (1p)

         **AM** marcaMasina pret disponibilitate tipCombustibil

o   Comanda ce adauga o masina in centrul de inchirieri (1p)

         **AV**

o   Comanda ce afiseaza toate vehiculele(disponibile sau nu) din centrul de inchirieri (1p)

         **AVD**

o   Comanda ce afiseaza doar vehiculele disponibile din centrul de inchirieri (1.5p)

         **STERGE** index

o   Comanda ce sterge dupa index un vehicul din centrul de inchirieri (1p)

         **IR** marca

o   Comanda ce afiseaza primul vehicul disponibil cu acea marca (1p)

         **EXIT**

o   Comanda ce iese din program (0.5p)

\*marcaBicicleta, marcaMasina, pret, disponibilitate, echipamentProtectie, tipCombustibil sunt inlocuite in comenzi cu String-uri specific.

**Creati un thread care va afiseaza la fiecare 3s numarul de autovehicule care sunt adaugate in baza pana in acel moment.**

Observatii:

Clasele vor respecta principiul de encapsulare. Realizarea corecta a claselor: 3p.

Se va folosi Singleton Pattern acolo unde este necesar.

Se vor folosi expresii lambda oriunde este posibil.

Timp de lucru: 2h.