

O banca proceseaza tranzactii venite de la diferiti clienti in timp real.

Fiecare client cand initiaza o tranzactie (de tip credit, sau debit), emite un mesaj ce este plasat intr-o coada pana cand este procesat de banca.

Structura mesajelor - (client, tip\_tranzactie, suma)

**E.g: ("client\_1", "credit", 100) ("client\_1", "debit", 50) ("client\_2", "debit", 50)**

Banca isi gestioneaza aceste tranzactii intr-o structura de tip dictionar, ce are urmatoarea structura: {

```
    client_id: { credit: total,  
                debit: total }  
}
```

Pentru lista de tranzactii de mai sus, structura finala ar fi urmatoarea:

```
{  
  client_1: {  
    credit: 100 debit: 50  
  },  
  client_2: {  
    debit: 50  
  }  
}
```

Clasa Dictionar va avea urmatoarele metode:

Descriere:

K - tip Cheie (String) V - tip Valoare (Dictionar (pentru primul nivel)/Intreg (pentru al 2 lea nivel))

D - dictionar gol

**- verifyKey(k)**

- pre: D valid - post: True - daca exista k de tip K

False - contrar

**- insertKey(k):**

- pre: verifyKey(k) - False
- post: k este inserat in D

**- pushValue(k,v):**

- pre: verifyKey(k) - True
- post: valoarea v este inserat in V corespondent lui k

**- deleteValue(k,v):**

- pre: verifyKey(k) - True
- post: valoarea v este stearsa din V respondent lui k

**- iterator**

- pre: D valid
- post: it - iterator peste dictionar

**Se cere implementarea unui mecanism de sincronizare la nivelul structurii de date prezentate mai sus.**

**PRECESUL DE ITERARE VA BLOCA DE ASEMENEA INTREAGA STRUCTURA IN MOMENTUL ITERARII.**

Test:

Thread-ul 1 va insera tranzactii de tipul debit pentru client\_1 cu valori random .

Thread-ul 2 va insera tranzactii de tipul debit pentru client\_2 cu valori random .

Thread-ul 3 va insera tranzactii de tipul credit pentru client\_1 cu valori random .

Thread-ul 4 va insera tranzactii de tipul credit pentru client\_2 cu valori random .

Thread 1, 2, 3 si 4 insereaza **N=100** tranzactii fiecare.

Thread-ul 5 va itera de fiecare data peste structura de date si va afisa structura de gestiune a bancii la momentul respectiv de fiecare data cand thread-urile 1, 2, 3 si 4 termina de inserat cumulat in coada **50 tranzactii**. Se va folosi un mecanism de **wait-notify** pentru implementarea acestui proces de iterare.

Thread-urile vor trebui sa isi inregistreze actiunile intr-un fisier de tranzactii.log.