

## Lucrul cu colecții de obiecte Java

Într-un program avem mai multe feluri de obiecte de tip *valoare*. Pe orice astfel de obiect putem apela metoda *conține(long)* ce întoarce o listă de obiecte *valoare* a cărei conținut se va construi după cum e detaliat în cele ce urmează. Tipurile concrete de valori sunt:

*ValoareNumerică* - un astfel de obiect memorează în starea sa un *long* setat la crearea obiectului. Două obiecte *ValoareNumerică* sunt egale dacă memorează același *long* în starea lor. Un astfel de obiect se reprezintă sub formă de șir de caractere prin numeralul corespunzător *long*-ului conținut. Metoda *conține(long)* întoarce o listă *LinkedList* în care se pune referință la obiectul curent dacă *long*-ul său e egal cu parametrul metodei. Altfel metoda întoarce o listă goală.

*PerecheDeValori* - un astfel de obiect memorează în starea sa două obiecte *valoare* de orice fel și în orice combinație (ex. două valori numerice, două perechi de valori, o valoare numerică și o pereche de valori, etc). Două perechi de valori sunt egale dacă valorile conținute de ele sunt egale, în aceeași ordine. O pereche de valori se reprezintă sub forma "(Membrul1, Membrul2)" unde Membrul1 și Membrul2 sunt reprezentarea sub formă de șir de caractere a membrului unu a perechii, respectiv a membrului doi. Metoda *conține(long)* returnează un *ArrayList* care conține rezultatele apelării metodei *conține(long)* pe fiecare membru al perechii.

*Grup* - un astfel de obiect memorează în starea sa o mulțime (obligatoriu *HashSet*) de obiecte de tip *valoare*. Clasa definește metoda *adaugă(Valoare...)* cu număr variabil de parametri care introduce parametrii în mulțimea conținută de obiectul grup. Două grupuri sunt egale dacă obiectele lor mulțime sunt egale. Un grup se reprezintă sub formă de șir de caractere sub forma "[Membru1, Membru2, ... ,MembruN]" unde MembruX e reprezentarea sub formă de șir de caractere a membrului X al perechii. Metoda *conține(long)* returnează un *ArrayList* care conține rezultatele apelării metodei *conține(long)* pe fiecare membru al grupului (implementarea acestei metode realizându-se obligatoriu folosind un *Iterator* ca modalitate de traversare a elementelor mulțimii).

Se va implementa o metodă *main* în care să avem un grup a cărei reprezentare sub formă de șir de caractere este "[(1, 2), (3, [3])]". grupul se va tipări la ieșirea standard. Se va încerca apoi adăugarea în grup a unui obiect pereche "(1, 2)" diferit de obiectul utilizat la crearea grupului anterior și se va retipări la ieșirea standard grupul inițial. Apoi se va tipări la ieșirea standard, pe câte o linie diferită, rezultatul apelării metodei *conține* cu parametrul 3 pe grupul creat. Pentru fiecare obiect din rezultat se tipărește și *hashCode*-ul său și *hashCode*-ul de identitate a acelui obiect (folosind metoda statică *identityHashCode()* din clasa *System*). Se folosește obligatoriu ca modalitate de iterare a listei un *for* normal făcând uz de metodele *get()* / *size()* a unei liste.