

Exercices sur les références

Exercice 3.1.1 *dessin*

Pour cet exercice, vous allez dessiner des structures de données au moyen de petits schémas analogues à ceux du cours, comportant la pile et le tas. Les références pourront être représentées par des flèches ou des adresses.

Question 1 *tableaux*

Dessinez l'état de la mémoire à la fin de l'exécution de ce programme.

```
class Exo_16_1_1{
    public static void main(String[] args){
        int[] t1, t2, t3, t4, t5;
        t2 = new int[4];
        t3 = new int[4];
        t4 = new int[4];
        for (int i=0; i<t3.length; i++){
            t3[i] = 10+i;
            t4[i] = 10+i;
        }
        t5 = t4;
    }
}
```

Question 2 *objets*

Même question pour le programme suivant.

```
class Exo_16_1_2{
    public static void main(String[] args){
        C1 var1, var2, var3;
        C2 cou1, cou2, cou3, cou4, cou5, cou6;
        var1 = new C1(0);
        var2 = new C1(1);
        cou2 = new C2();
        cou3 = new C2();
        cou4 = new C2();
        cou6 = new C2();
        cou3.premier = var1;
        cou3.second = var2;
        cou4.premier = var1;
    }
}
```

```

        cou4.second = var1;
        cou5 = cou3;
        cou6.premier = var1;
        cou6.second = new C1(2);
    }
}
class C1{
    int x;
    C1(int i){
        x=i;
    }
}
class C2{
    C1 premier, second;
}

```

Question 3 *tableaux d'objets*

Même question pour le programme suivant.

```

class Exo_16_1_3{
    public static void main(String[] args){
        C1[] t1, t2, t3;
        C1 var = new C1(0);
        t1=new C1[3];
        t2=new C1[3];
        t3=new C1[3];
        for (int i=0; i<t1.length; i++){
            t1[i] = var;
            t2[i] = new C1(0);
        }
        t1[1].x = 1;
        t2[1].x = 1;
        t3 = t2;
    }
}

```

Question 4 *objets contenant des tableaux*

Même question pour le programme suivant.

```

class Exo_16_1_4{
    public static void main(String[] args){
        C3 p1, p2, p3;
        C1 c1;
        c1 = new C1(0);
        p1 = new C3();
        p2 = new C3();
        p2.tab = new C1[3];
        p3 = new C3();
        p3.tab = new C1[3];
        p3.tab[1] = c1;
    }
}

```

```
        pl3.tab[2] = c11;
    }
}
class C3{
    C1[] tab;
}
```

Exercice 3.1.2 *comptes bancaires*

On va représenter des comptes bancaires et des titulaires de comptes. Un même compte peut avoir plusieurs titulaires (c'est le cas par exemple des comptes joints pour un couple). Dans ce cas, le même objet sera partagé entre les titulaires. Un titulaire peut avoir plusieurs comptes.

On vous propose le squelette de classe suivant pour implémenter les comptes et les titulaires.

```
class TableauCompte{
    Compte[] tab;
    int longueur;
    TableauCompte(int n){
        tab = new Compte[n];
    }
    void ajouter(Compte c) throws NonInitialise{
        if (c == null){
            throw new NonInitialise();
        }
        if (longueur < tab.length){
            tab[longueur]=c;
            longueur++;
        }
    }
}
class Banque{
    String nom;
    TableauCompte tous = new TableauCompte(50);
    Banque(String n){
        nom = n;
    }
}
class Titulaire{
    String nom;
    Titulaire(String n){
        nom = n;
    }
    TableauCompte mesComptes = new TableauCompte(10);
}
class Compte{
    int numero;
    int solde;
    void depot(int n){
        solde = solde + n;
    }
    void retrait(int n){
```

```
        solde = solde -n;
    }
    void afficher(){
        Terminal.ecrireString("solde_du_compte_numero_" + numero + ":\n");
        Terminal.ecrireInt(solde);
    }
}
class NonInitialise extends Exception{ }
```

Question 1 création de comptes

Ajouter à ce squelette de classes des méthodes. Seule la banque peut créer des nouveaux comptes. Quand elle crée un nouveau compte, elle l'ajoute à la liste des comptes de chacun des titulaires de ce compte et elle l'ajoute aussi à sa propre liste de comptes. Par exemple, si une banque *b* crée un compte *c* pour les titulaires *t1* et *t2*, le même objet *c* sera ajouté à trois tableaux : *b.tous*, *t1.mesComptes* et *t2.mesComptes*. Dans chacun de ces trois tableaux, il y aura la même adresse, celle de l'objet *c*.

Pour réaliser cette création de comptes, il faut une méthode dans *Banque* qui crée un compte avec en paramètre la liste des titulaires (par exemple sous forme de tableau). Il peut également être utile d'écrire une méthode permettant d'ajouter un nouveau compte chez un titulaire.

Question 2 variables

Ecrivez une méthode *main* qui réalise une situation où trois titulaires, Paul, Pierre et Fatima ont des comptes dans une banque *BNP*. Paul et Fatima ont un compte joint. Fatima a en plus un compte personnel, de même que Pierre.

Dessinez la situation en faisant apparaître sur un schéma tous les objets et tableaux.

Question 3 virements

On désire écrire une méthode qui permette de faire un virement depuis un compte dont on est titulaire vers un compte dont on ne connaît que le numéro (et dont on n'est pas forcément titulaire) et la banque (autrement dit, on ne connaît pas son adresse). Dans quelle classe faut-il ajouter cette méthode ? Ecrivez cette méthode.

Exercice 3.1.3 procurations

Nous allons développer l'exemple des comptes bancaires en introduisant une nouvelle notion : celle de procuration. En plus des comptes dont elle est titulaire, une personne peut avoir une procuration sur le compte d'une autre personne. Cela lui permet de faire des opérations sur le compte (par exemple des dépôts et des retraits), mais elle n'a pas le droit de donner à son tour procuration sur ce compte. Seul un des titulaires peut donner procuration sur un compte.