

Objectifs du TD

Ce TD à pour but de présenter le concept de mutabilité des variables en Java, ainsi que les itérations (boucles for).

Exercice 1 : Echange du contenu de deux variables

Ecrire une classe Echange réalisant l'échange de :

- a) 2 entiers passés par la ligne de commande,
- b) 2 objets de type Entier (cf TD2) construits au moyen des entiers passés par la ligne de commande.

Rappel : la classe Entier contient un attribut `private int Val` ;, un constructeur, ainsi que les méthodes `copy(Entier a)`, `getVal()`, et `affiche()`.

Tirez les conclusions sur la manipulation de *valeurs* et *références* en Java.

Exercice 2 : Portée des variables

Lisez le code qui suit et répondez aux questions insérées dans les commentaires :

```
public class test { // On utilise cette classe dans la classe suivante
    private int x = 10; // Déclaration de variables
    public int y = 20;
    test(){
        // Constructeur de la classe
    }
}

public class portee {
    static test e,e1; // 1. A quoi sert la déclaration static ?
    private int x=1; // 2. Quelle est la portée de x ?
    public int y=2; // 3. Quelle est la portée de y ?

    public portee() {
        int x=5; // 4. Quelle est la portée de x ?
        A(); // 5. Qu'affiche t-on avec la méthode A ?
        B(); // 6. Qu'affiche t-on avec la méthode B ?
        A(); // 7. Qu'affiche t-on avec la méthode A ?
        B(); // 8. Qu'affiche t-on avec la méthode B ?
        System.out.println(x); // 9. Quelle est la valeur de x?
    }

    public void A() { // Méthode A
        int x =25;
        System.out.println("Méthode A avant:"+x);
    }
}
```

```

    ++x;
    System.out.println("Méthode A après:"+x);
}

public void B() { // Méthode B
    System.out.println("Méthode B avant:"+x);
    x*=10;
    System.out.println("Méthode B après :"+x);
}

public static void main(String[] args) { // Méthode main
    int x,y;
    portee p = new portee();
    e = new test();
    e1 = new test();

    // 10. Qu'obtient-on ici ?
    // 11. Proposez une solution au problème de la question 10.

    System.out.println("e avant:"+e.x);
    e.x = 50;
    System.out.println("e apres:"+e.x);

    // 12. Qu'obtient-on ici ?

    System.out.println("e avant:"+e.y);
    e.y = 60;
    System.out.println("e apres:"+e.y);
}
}

```

Exercice 3 : Calculs itératifs

L'énoncé itératif est souvent appelé boucle. On entre dans la boucle et à chaque tour de boucle les énoncés qui la composent sont exécutés. L'arrêt de la boucle se produit lorsque le prédicat d'achèvement est vérifié.

A/ Factorielle

Ecrire sous forme itérative l'algorithme de la fonction factorielle $n!$. A savoir que cette fonction s'exprime par récurrence comme suit :

```

Factorielle(0)=1
Factorielle(n)=n*Factorielle(n-1)

```

B/ Fibonacci

Ecrire sous forme itérative l'algorithme de la fonction fibonacci (fonction proposée en 1202 par le mathématicien italien LEONARDO PISANO). A savoir que cette fonction s'exprime par récurrence comme suit :

```
Fib(0)=0
Fib(1)=1
Fib(n)=Fib(n-1)*Fib(n-2)
```

C/ Le plus grand diviseur

Le plus grand diviseur de deux nombres naturels est l'entier naturel le plus grand qui les divisent tous les deux. Il est tel que :

```
pgcd(a,b) = pgcd(a-b,b) si a>b
pgcd(a,b) = pgcd(a,b-a) si a<b
```

Ecrivez le code qui réalise ce calcul.

Exercice 4 : Manipulation des tableaux

A/ Conversion

Ecrire une méthode qui transforme un tableau de chaînes de caractères chainetab en tableau d'entiers :
`public int[] chaineversentier(String [] chainetab).`

B/ Insertion

Déclarez un tableau comportant les éléments 2,56,87,102,932 dans la fonction `main()`, puis appelez la méthode `insere()` qui va insérer le nombre 203.

La méthode `insere()` devra :

- Trouver la place où insérer l'élément,
- Puis décaler tous les éléments supérieurs d'une place avant d'insérer le nombre désiré.

Ecrire cette méthode `insere()` dans un premier temps, puis la méthode `main()` (attention à la déclaration des tableaux).

Exercice 5 : Classe Somme

Ecrire une classe `Somme` qui permet de calculer la somme de n entiers. Cette classe contient les méthodes suivantes :

- `Somme()`
- `calculSomme(int n)`
- `afficheResultat()`

Ecrire ensuite la classe `UtiliseSomme` qui comporte une méthode `main()` et une méthode `lire()` permettant de lire un entier saisi au clavier. Cette classe va demander à l'utilisateur d'entrer un entier n et appellera `Somme` pour calculer la somme des n premiers entiers et l'afficher.