

Programmation 1

Examen

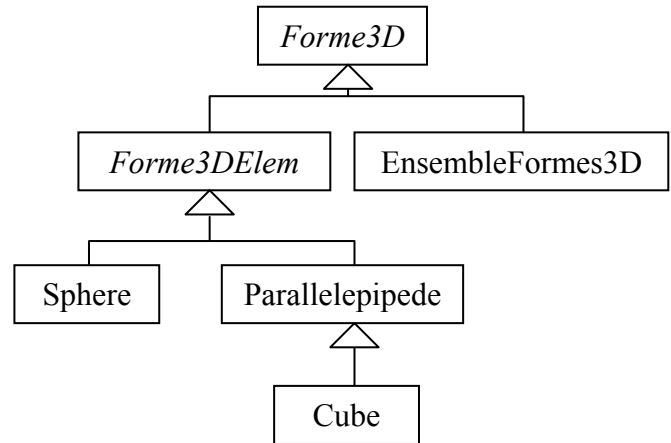
Nous nous proposons de définir des classes qui permettent de manipuler des formes dans un espace à trois dimensions. Les opérations souhaitées sont le déplacement des formes et le calcul de la surface, du volume et du poids des formes. Pour cela nous définissons la hiérarchie suivante :

La classe *Forme3D* est une classe abstraite définie par :

```
abstract public class Forme3D {  
    abstract public void deplace(double dx, double dy, double dz);  
    abstract public double surface();  
    abstract public double volume();  
    abstract public double poids();  
}
```

La classe *Forme3DElem* est une classe abstraite, super classe des formes élémentaires. Les formes élémentaires ont un centre et une densité.

La classe *EnsembleFormes3D* est une forme en trois dimensions composées de plusieurs formes, élémentaires ou non.



Question 1 (3 points)

Définir les constructeurs des classes *Forme3DElem*, *Sphere*, *Parallelepipede*, *Cube*.

Question 2 (3 points)

Définir la méthode *void deplace(double dx, double dy, double dz)* des formes élémentaires. Pour déplacer une forme, il suffit de déplacer son centre.

Question 3 (3 points)

Définir les méthodes *double surface()* et *double volume()* des formes élémentaires. La surface d'une sphère est $4 \times \pi \times \text{rayon}^2$ et son volume est $4/3 \times \pi \times \text{rayon}^3$.

Question 4 (3 points)

Définir les méthodes *double poids()* des formes élémentaires. Le poids d'une forme est égal à son volume multiplié par sa densité.

La classe *EnsembleFormes3D* permet de représenter un ensemble de formes tridimensionnelles. Cet ensemble est représenté à l'aide d'un *ArrayList*.

Question 5 (1 point)

Définir le constructeur de la classe *EnsembleFormes3D*.

Question 6 (3 points)

Définir les méthodes :

- *void ajouter(Forme3D f)* : ajoute la forme *f* à l'ensemble des formes.
- *void enlever(Forme3D f)* : enlève la forme *f* de l'ensemble des formes.
- *enleverTout()* : enlève toutes les formes de l'ensemble des formes.

Question 7 (4 points)

Définir les méthodes :

- *void deplace(double dx, double dy, double dz)* : déplace toutes les formes de l'ensemble de formes.
- *double volume()* : retourne la somme des volumes des formes de l'ensemble.
- *double surface()* : retourne la somme des surfaces des formes de l'ensemble.
- *double poids()* : retourne la somme des poids des formes de l'ensemble.

```
public abstract class Forme3DElem extends Forme3D {
    protected Point3D centre;
    protected double densite;
    public Forme3DElem ( Point3D c, double d){...}
    . . .
}
```

```
public class Sphere extends Forme3DElem {
    private double rayon;
    public Sphere( Point3D c, double d, double rayon){...}
    public Sphere( Point3D c, double rayon){... }
    . . .
}
```

```
public class Parallelepipede extends Forme3DElem {
    private double largeur, profondeur, hauteur ;
    public Parallelepipede ( Point3D c, double d, double l, double p, double h)
    {...}
    public Parallelepipede ( Point3D c, double l, double p, double h){...}
    . . .
}
```

```
public class Cube extends Parallelepipede {
    public Cube ( Point3D c, double d, double cote){...}
    public Cube ( Point3D c, double cote){... }
    . . .
}
```

```
public class EnsembleFormes3D extends Forme3D {
    private ArrayList<Forme3D> lesFormes;

    public EnsembleFormes3D(){...}

    public void ajouter( Forme3D f){...}
    public void enlever( Forme3D f){...}
    public void enleverTout(){...}

    public double surface() {...}
    public double volume() {...}
    public double poids() {...}

    public void deplace (double dx, double dy, double dz) {...}
}
```

```
public class Point3D {
    private double x, y, z;

    public Point3D( double x, double y, double z){
        this.x = x; this.y = y; this.z = z;
    }

    void deplace(double dx, double dy, double dz){
        x+=dx; y +=dy; z+=dz;
    }

    public double getX(){return x;}
    public double getY(){return y;}
    public double getZ(){return z;}

    public void setX(double x){this.x = x;}
    public void setY(double y){this.y = y;}
    public void setZ(double z){this.z = z;}
    public void setXYZ(double x, double y, double z){
        this.x = x; this.y = y; this.z = z;
    }
    public String toString(){
        return (" +x+", " +y+", " +z+");
    }
}
```