

## TD 8

UFR Math-Info  
UE – L2/S4 – Programmation 4

### Question 1 – La classe Point

Compléter le squelette de code ci-dessous

```
public class Point {  
    //Attributs  
    private float x;  
    private float y;  
  
    //Constructeurs  
    public Point(float x, float y) {  
    }  
    public Point() {  
    }  
  
    //get  
    public float getX() {}  
  
    public float getY() {}  
  
    //Autres méthodes  
    //Distance entre le point courant et le point passé en argument  
    public float distance(Point p) {}  
  
    //Redéfinition de la méthode toString() (de la classe Object)  
    //pour avoir une chaine de la forme "(x,y)"  
    public String toString() {}  
  
    public void affiche() {}  
}
```

### Question 2 – La classe Polygone

Un polygone est défini par la liste de ses sommets successifs, par exemple un rectangle est défini par les coordonnées de ses quatre sommets (x1,y1), (x2,y2), (x3,y3) et (x4,y4)

Compléter le squelette de code ci-dessous :

```

public class Polygone {
    //Attribut
    private Point[] sommets;

    //Constructeurs
    public Polygone(Point[] sommets) {}

    public Polygone(int nbSommets) {}

    //Méthodes
    //Ajoute un sommet dans le tableau sommets
    public void setUnSommet(Point p, int pos){}

    //mettre à jour tous les sommets
    public void setSommets(Point[] sommets){}

    //Retourne le nom de la figure (Polygone dans le cas présent)
    protected String nomFigure() {}

    //Retourne le nombre de sommets
    public int nombreDeSommets() {}

    //retourne une chaine contenant les coordonnées des sommets
    //Elle se présente sous la forme : "Polygone : [(0.0,0.0) (2.0,0.0)
    (2.0,2.0) (0.0,2.0) ]"
    public String toString() {}

    //Affiche les coordonnées des sommets
    public void affiche(){}

    //Calcul le périmètre (somme des côtés) du polygone
    public float perimetre() {}
}

```

### Question 3 - La classe Rectangle

Pour simplifier le problème, on ne considère ici que les rectangles ayant des côtés parallèles aux axes. Chaque rectangle peut être défini par deux points opposés ou par un point plus la longueur et la largeur.

Ecrire la classe Rectangle, elle doit proposer deux constructeurs et doit permettre de calculer le périmètre et d'afficher les sommets.

### Question 4 – La classe Carre

Pour simplifier le problème, on ne considère ici que les carrés ayant des côtés parallèles aux axes. Chaque carré peut être défini par un point plus la longueur de son côté.