

## 1 Création d'un module

On désire implémenter une bibliothèque permettant de représenter des objets géométriques dans un repère cartésien orthonormé. Le but de cette activité est de commencer l'écriture d'un module répondant à ce désir, et de le documenter.

### 1.1 Écriture du module

Créer un module `geometrie.py` qui implémente les ressources suivantes :

1. **constante `PI`** (importée depuis le module `math` : `math.pi`)
2. **classe `Point`** :  
**Attributs** : `x` et `y` : ce sont les coordonnées cartésiennes du point.  
**Méthodes** :
  - (a) `__str__` : renvoie une chaîne de caractères donnant les coordonnées du point sous la forme `(x, y)`.  
*Remarque* : `__str__` est une méthode spéciale appelée lorsqu'on affiche l'objet avec `print`.
  - (b) `__eq__` : prend un autre point en paramètre et renvoie un booléen indiquant si les points sont égaux (mêmes coordonnées).  
*Remarque* : `__eq__` est une méthode spéciale appelée lorsqu'on utilise l'opérateur d'égalité `==`.  
Ainsi, avec cette méthode, on pourra écrire `A == B` (renvoie `True` si `A` et `B` représentent le même point).
3. **classe `Rectangle`** :  
**Attributs** : `so` et `ne` : ce sont les sommets inférieur droit et supérieur gauche (Sud-Ouest, Nord-Est) du rectangle.  
*Remarque* : Lorsqu'on instancie un rectangle, on donne 2 points en paramètres et le rectangle créé passe par ces 2 points.  
**Méthode** :
  - (a) `__str__` : renvoie une chaîne de caractères donnant les coordonnées des points `SO` et `NE` sous la forme `[(xSO, ySO), (xNE, yNE)]`.
4. **classe `Cercle`** :  
**Attributs** : `c` et `r` : `c` est le Point centre et `r` le rayon du cercle.  
**Méthodes** :
  - (a) `rayon` : renvoie le rayon du cercle.
  - (b) `centre` : renvoie le centre du cercle.
  - (c) `__str__` : renvoie une chaîne de caractères donnant les coordonnées du centre et le rayon `r` sous la forme `[(xC, yC), r]`.
  - (d) `perimetre` : renvoie le périmètre du cercle.
  - (e) `surface` : renvoie la surface du disque défini par le cercle.
5. **fonction `distance`** : prend en paramètre 2 Points et renvoie la distance séparant ces points.  
*Rappel* : la distance entre 2 points de coordonnées  $(x_1, y_1)$  et  $(x_2, y_2)$  est :
$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

## 1.2 Documentation, spécification

Si ce n'est déjà fait, documenter et spécifier l'ensemble des fonctions et classes du module.

## 1.3 Tests

Ajouter librement quelques fonctionnalités permettant de tester le module, en l'exécutant comme un programme indépendant.

## 2 Exploitation du module

Créer un programme `geo1.py` qui s'appuiera sur le module de `geometrie.py` pour réaliser les actions suivantes :

1. Créer 5 points A, B, C, D et E de coordonnées respectives (1, 2) , (3, 7) , (-2, 3) , (1, 2) et (3, -1).
2. Vérifier que A et D sont égaux, et que A et B sont différents.
3. Déterminer la longueur du segment AB.
4. Créer un rectangle passant par C et D.
5. Créer un cercle de centre E et de rayon 2,5234 (en unités du repère).
6. Tester si la surface du rectangle est égale à celle du disque défini par ce cercle.  
Attention : on se souviendra qu'on ne teste jamais l'égalité entre des nombres flottants mais on vérifie que leur différence est « suffisamment petite ». On décidera ici que 2 flottants sont égaux s'ils le sont à  $10^{-6}$  près.
7. Créer une fonction qui renvoie le symétrique d'une forme (cercle ou rectangle) par rapport à l'axe des abscisses. Utiliser cette fonction pour obtenir le symétrique du rectangle et du cercle précédemment définis.