

## Objet : pratique du principe de délégation.

En programmation objet, le programme fonctionne via les instances qui sont créées, chacune d'entre-elles « offrant » ses propres « services ». C'est la méthode `main` de la classe principale (celle qui représente le programme) qui crée les premières instances et articule les services qu'elles apportent. Dans ce cadre, le principe de délégation consiste à ne faire effectuer par une instance d'une classe que ce qu'elle « sait faire » mieux que les instances d'autres classes. C'est notamment le cas des méthodes qui utilisent les valeurs de ses attributs. De même, elle ne doit pas empiéter sur les « compétences » des autres instances et doit leur déléguer ce qu'elles « savent faire » mieux qu'elle.

## Objets géométriques

Il s'agit de représenter des objets géométriques dans un espace euclidien de dimension 2. Chaque type d'objet est représenté par une classe. Recensez les attributs et méthodes (les membres) de ces classes, ainsi, éventuellement, que les procédures ou fonctions annexes nécessaires. Il n'y a pas de représentation graphique. On s'intéresse aux propriétés suivantes :

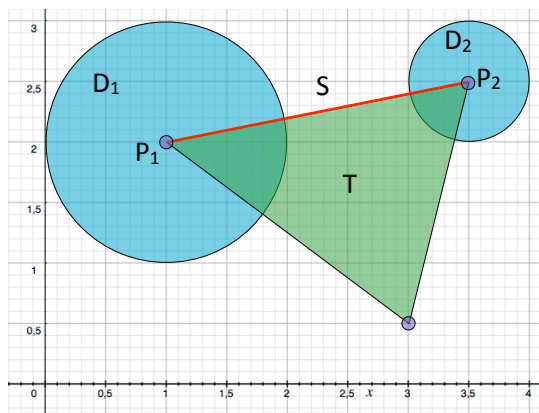
1. Pour un point, son abscisse et son ordonnée.
2. Pour un disque, son centre, son rayon, son périmètre et son aire. On doit pouvoir déterminer qu'un point fait ou non partie de la surface du disque.
3. Pour le segment, ses deux points extrémités, sa longueur<sup>1</sup>, et son milieu.
4. Pour un triangle, ses sommets, ses arêtes, son périmètre et son aire.

Il doit être possible de contrôler les valeurs proposées pour les attributs, par exemple que le rayon d'un disque soit positif, que son centre soit bien défini (différent de `null`), que les deux points d'un segment soient distincts,...

Faites en sorte qu'un point puisse être créé à partir de deux coordonnées ou d'un autre point par clonage, qu'un disque puisse être créé à partir d'un centre et d'un rayon ou à partir d'un autre disque par clonage. Quelle question doit-on alors se poser à propos du centre du disque ? Un triangle doit pouvoir être défini à partir de trois points ou d'un point et d'un segment.

En utilisant les constructeurs définis dans la question précédente, écrivez les instructions qui permettent de créer la figure ci-dessus.

L'équivalence entre deux objets du même type devra pouvoir être vérifiée (`equals`).



<sup>1</sup> Distance euclidienne entre les extrémités.