

## Héritage comme spécialisation

### Exercice 1 : Comprendre la classe PointNommé

À partir du texte source de la classe PointNommé, expliquer les différents éléments de cette classe et dessiner le diagramme de classes UML avec cette classe PointNommé.

### Exercice 2 : Comprendre la relation d'héritage entre Point et PointNommé

Intéressons nous au programme de test TestPolymorphisme.

**2.1.** Lire le texte du programme TestPolymorphisme et répondre aux questions posées.

**2.2.** Exécuter le programme TestPolymorphisme pour vérifier les réponses.

**2.3.** Ajouter le mot-clé **final** devant la déclaration de la méthode afficher dans la classe Point. Compiler Point, puis PointNommé et expliquer.

**2.4.** Ajouter **final** devant la déclaration de la classe Point. Compiler Point et PointNommé.

**2.5.** Expliquer pourquoi on peut calculer la distance entre un point et un point nommé ? Et entre un point nommé et un point ?

**2.6.** Pourquoi peut-on attacher à q aussi bien un point qu'un point nommé ? Que se passe-t-il lorsque q est affiché ? Quel est ce concept ?

### Exercice 3 : Segments et points nommés

Intéressons nous aux segments et points nommés. Remarquons que la classe Segment définie ne fait référence dans son texte qu'à la classe Point et jamais à PointNommé.

**3.1.** Peut-on créer un segment à partir d'un point et d'un point nommé ? Comment serait affiché un tel segment ? Justifier les réponses.

**3.2.** Créer le programme ExempleSchema2 pour représenter le schéma de la figure 1. On partira du programme ExempleSchema1. Indiquer les modifications à apporter.

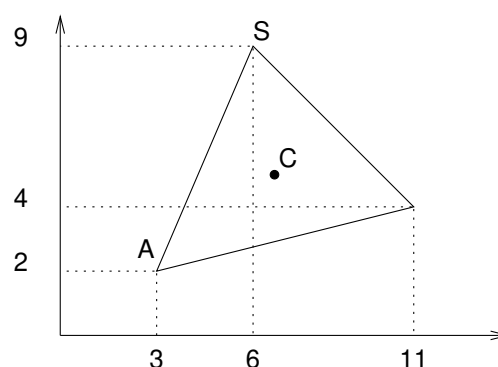


FIGURE 1 – Un exemple de schéma mathématique avec des points nommés