- -

L'objectif de cette feuille est de développer la collection générique ***Ensemble*** du package ***_Maths***. Cette classe sera utilisée dans les feuilles d'exercices suivantes pour développer les classes ***AgenceBancaire*** et ***Banque*** du package***_Banque***.

Exercice 1. La classe *Collection* - Version 1.0.0 / Package _*Maths*.

Une collection est un agrégat d'éléments pointés. Aucune hypothèse restrictive sur le nombre, la nature, l'ordre ou l'identité éventuelle de deux éléments pointés (ou plus) ne peut être imposée. La notion est donc plus générale que celle d'ensemble (au sens mathématique du terme).

Installer avec succès les codes sources fournis en annexe.

Q1 / Pour quelle raison interdit on la duplication de toute collection ?
Q2 / Pour quelle raison la signature formelle de la méthode ***toString*** spécifie t-elle un paramètre ?
Q3 / Quel est la nature de ce paramètre ?
Q4 / Quel est l'usage de ce paramètre ?
Q5 / Quel est le rôle du service ***vider*** ?

Exercice 2. La classe *Collection* - Version 1.0.0 / Tests unitaires

Compléter le module de tests unitaires de l'opérateur += par le cas d'un ensemble de points du plan.

Exercice 3. La classe *Collection* - Version 1.0.0 / Surcharge de l'operateur []

Surcharger l'opérateur [] et exécuter avec succès le module de tests unitaires fourni en annexe.

Exercice 4. La classe *Ensemble* - Version 1.0.0

Un ensemble est une collection sur laquelle est définie une relation d'appartenance. Un ensemble ne peut pas contenir plusieurs fois le même élément.

Installer avec succès les codes sources fournis en annexe.

Exercice 5. La classe *Ensemble* - Version 1.0.0 / Constructeur de copie

Quelle est la définition du mécanisme de "*transtypage dynamique implicite ascendant*" ?

Analyser soigneusement les codes sources du constructeur de copie.

Quelles sont les lignes qui exploitent le mécanisme précité ?

Quelles restrictions d'usage imposent ce mécanisme pour l'objet transtypé ?

Exercice 6. La classe *Ensemble* - Version 1.0.0 / Surcharges des opérateurs += et -=

Justifier la surcharge de ces deux opérateurs.

Exercice 7. La classe *Ensemble* - Version 1.0.0 / Méthode *inclus*

Développer la méthode ***inclus*** et exécuter avec succès le module de tests unitaires fourni en annexe.

Exercice 8. La classe *Ensemble* - Version 1.0.0 / Méthode *intersection*

Développer la méthode ***intersection*** et exécuter avec succès le module de tests unitaires fourni en annexe.

Exercice 9. La classe *Ensemble* - Version 1.0.0 / Tests unitaires des cas d'anomalies

Compléter le module de tests unitaires des cas d'anomalies, fourni en annexe.