Travaux dirigés n°3 : Héritage

EXERCICE 1

- Réaliser une classe **point** permettant de manipuler un point d'un plan défini par ses coordonnées (réel) et une couleur (entier).

Cette classe définit les opérations suivantes:

Un constructeur par défaut

Un constructeur à 3 paramètres

afficher coordonnées

changer ordonnée

changer coordonnée

changer couleur

afficher nom (elle affiche je suis un point)

déplace effectuant une translation définie par deux arguments

- Utiliser la classe Point pour définir la classe suivante:

Point_mathématique:

Booléen Coïncide(point) vérifie si deux points coïncident ou non

Booléen symétrique(point) vérifie si deux points sont symétriques ou non

Exercice 2

- Créer une classe compte d'épargne contenant les membres suivants:

Numéro de compte

Taux d'intérêt annuel

Solde

Constructeur (numéro)

Calcul_intérêt: calcul l'intérêt annuel en multipliant le solde par le taux d'intérêt annuel

Modifier taux (réel) :qui ajuste le taux d'intérêt annuel

Ajouter intérêt : qui ajoute le montant d'intérêt au solde

Déposer montant

Retirer montant

Afficher solde

- Créer une classe Ccompteplafonné gérant des dépôts d'argent avec un plafond (montant maximum), héritant la classe compte d'épargne ayant l'attribut Plafond.

Et les opérations suivantes:

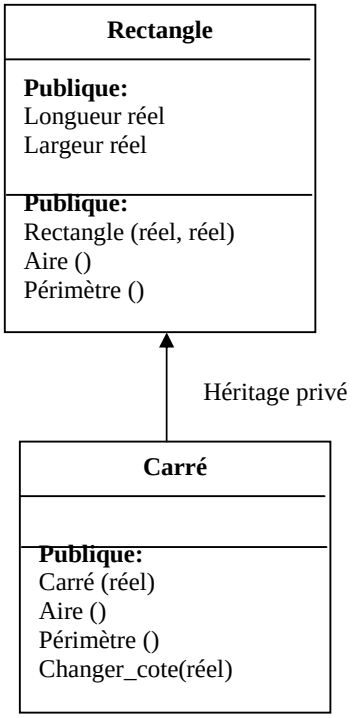
Constructeur (numero, plafond)

Déposer(montant)

Afficher solde

Exercice 3 examen

- Définir les classes Rectangle et Carré



- corriger le programme suivant:

```
#include <iostream.h>

int main()

{Rectangle Or;
```

```
Carre Oc1,Oc2(13.f,14.f);


Or.Longueur =10.5f;

Or.Largeur = 11.2f;

Oc1. Longueur =22.f;

Oc1.Largeur = 22.f;

cout <<Oc1.Aire();

Or.changer_cote(32.f);

cout <<Oc1.Perimetre();

cout <<Or.Perimetre();

return 0;

}
```

Remarque il y a 4 erreurs, il faut justifier chaque erreur