

TP6 – Les objets :

héritage, polymorphisme, virtual, héritage multiple

Exercice 1 :

Soit la classe Vecteur définie par deux composantes réelles x et y . Un Vecteur possède un constructeur par défaut qui crée un vecteur nul (x et y valent 0) ou qui crée un vecteur à partir de deux réels passés en paramètres.

On peut afficher un vecteur sous cette forme : (2.5 , 6).

On peut aussi déplacer un vecteur suivant l'axe des abscisses et l'axe des ordonnées.

Un vecteur3D est un vecteur ayant une troisième composante réelle z .

1. Ecrire la classe Vecteur et la classe Vecteur3D en prenant soins de redéfinir les méthodes nécessaires.
2. Ecrire la fonction principale main, qui :
 - Crée d'une façon statique un vecteur nul et l'affiche.
 - Déplace le vecteur créé et le réaffiche
 - Crée d'une façon dynamique un vecteur3D

Exercice 2 :

On se propose de modéliser un Robot qui a pour objectif de traquer et de tuer un autre Robot. Un Robot est caractérisé par un nom, une position sur un plan orthonormé, une portée de son radar et un statut (bool estMort).

Un RobotChasseur est un Robot qui peut chasser et tuer d'autres Robot.

On peut :

- Déplacer un Robot d'une position (x, y) à une autre position (v, w) en fournissant un pas de déplacement selon l'axe des abscisses et un autre pas selon l'axe des ordonnées.
- Afficher un Robot en affichant toutes ses informations.
- un RobotChasseur peut tester si la position d'un autre Robot est à sa portée (la distance entre la position d'un robot et son rival est inférieure à sa portée)
- un RobotChasseur peut tuer un autre robot s'il se trouve à sa portée (modifier l'attribut "estMort" du robot rival à *true*) et afficher le nom du robot qui est mort.

1. Ecrire le prototype de la classe Robot (Robot.h) et RobotChasseur (RobotChasseur.h)
2. Implémenter le constructeur et les méthodes identifiées pour un Robot et un RobotChasseur
3. Dans le film "The transformers" le robot "Optimus" a pour objectif de traquer et de tuer

son rival le robot "Megatron" afin de sauver la terre.

Dans la fonction main (main.cpp):

- Créer un RobotChasseur (robotChass1) nommé "Optimus" à la position (2, 3) ayant une portée égale à 25
- Déplacer "Optimus" vers la position (35,22) et afficher les informations de ce robot.
- Créer 5 autres Robots et les placer dans des positions différentes (Vector).
- « Optimus » est constamment à la recherche de ses ennemies alors chaque Robot détecté dans sa portée sera abattue.

Exercice 3 :

Un parc auto se compose de voitures et de camions qui ont des caractéristiques communes regroupées dans la classe Véhicule.

- Chaque véhicule est caractérisé par un numéro de série, un numéro de véhicule dans la série, l'année de son modèle, son prix.
- Le numéro de série est incrémenté lorsque le numéro de véhicule atteint 10000.
- Le numéro de véhicule dans la série est incrémenté automatiquement à chaque création d'un véhicule et lorsqu'il atteint 10000 alors il sera réinitialisé à 1.
- la classe Véhicule possède un constructeur qui prend deux paramètres (année et prix) et génère les valeurs de la série et le numéro de véhicule.
- La classe Véhicule possède également une méthode accélérer(). On ne connaît pas comment un

Véhicule accélère sauf si on sait que c'est une voiture ou un camion.

- La méthode afficher() de la classe Véhicule affiche une chaîne de caractères qui contient les valeurs de la matricule (ex: 142 TUN 5555), de l'année du modèle et du prix.

Les classes Voiture et Camion étendent la classe Véhicule en définissant concrètement la méthodes accélérer().

Une voiture qui se déplace affiche "Je peux atteindre 200km/h" alors qu'un Camion affiche "Je peux atteindre 120km/h". Un camion possède en plus une charge maximale alors que la voiture possède un nombre de places.

Définissez les classes nécessaires et testez les dans une fonction main.

Exercice 4 :

Un jeu d'échec se compose de plusieurs pièces de deux couleurs différentes et chaque pièce se déplace selon une façon donnée sur un échiquier d'une taille 8x8.

Une pièce est caractérisée par une couleur (blanc ou noir), une position sur l'échiquier (x et

y). On peut afficher une pièce en indiquant son type, sa position et sa couleur.

Un pion est une pièce qui se déplace en avançant d'un pas (y augmentera de 1) et si sa position dépasse la limite de l'échiquier (8) alors un message sera affiché.

Un Roi est une pièce qui se déplace d'un pas selon une des 8 directions possibles.

Pour simplifier l'exercice on se limitera à ces deux pièces.

- Définissez les classes nécessaires

Ecrire une fonction main qui crée un vecteur contenant quelques Pions et un Roi. Puis afficher ces pièces.

Exercice 5 :

Dans le but de créer une application de gestion des ressources humaines d'un établissement universitaire, nous avons identifié la classe Enseignant, Etudiant, Contractuel. Tous ces objets partagent des caractéristiques communes suivantes : le nom, l'âge et le sexe.

Un enseignant est une personne ayant en particulier une charge à assurer.

Un étudiant est une personne ayant une filière.

Un contractuel est un étudiant-enseignant.

1. Ecrire la classe Personne et y définir les méthodes accesseurs, modificateurs et `getPersonne()` qui retourne une chaîne de caractère contenant ses informations.
2. Ecrire les classes Enseignant et Etudiant en y ajoutant et y redéfinissant les méthodes nécessaires
3. Ecrire la classe Contractuel en utilisant le concept d'héritage multiple.
4. Dans une fonction principale main, tester les classes que vous avez défini