Langage C++

- CIR2 -

ISEN ALL IS DIGITAL! NANTES | yncréa

TP6 : Composition

Ayoub KARINE (ayoub.karine@isen-ouest.yncrea.fr)

Exercice 1:

Partie 1 : Classe Point

1. Implémenter une classe Point comprenant :
    a. deux attributs : x (abscisses), y (ordonnées)
    b. un constructeur de deux arguments
    c. un constructeur de copie
    d. les getters et les setters

Contrainte : L'utilisation d'un constructeur par défaut est interdite dans la classe Point

Partie 2 : Classe Vehicle

1. Implémenter une classe Vehicle ayant :
    a. deux attributs

Nom	Type	Description
identifier	char*	Numéro d'immatriculation du véhicule

position	Point	Position du véhicule

b. un constructeur par défaut (valeurs par défaut : (0,0) pour la position, "" pour l'identifiant). **Contrainte** : L'utilisation de Point::setX() ou Point::setY() est interdite

```
Vehicule::Vehicule() : position(0,0){
   // 10 est la taille max des caractères d'une plaque
   identifier = new char[10*sizeof(char)];
   strcpy(identifier,"");
}
```

c. un constructeur de deux arguments. **Contrainte** : L'utilisation de Point::setX() ou Point::setY() est interdite
d. un destructeur. **Contrainte** : L'utilisation de Point::setX() ou Point::setY() est interdite
e. un constructeur de copie. **Contrainte** : L'utilisation de Point::setX() ou Point::setY() est interdite
f. les getters et les setters
g. une méthode float moveTo(p) qui déplace le véhicule au point p et renvoie la distance parcourue (choisir le bon type d'argument)
h. une méthode float moveTo(v) qui déplace le véhicule à la position du véhicule v et renvoie la distance parcourue (choisir le bon type d'argument).
i. une méthode float moveTo(x,y) qui déplace le véhicule aux coordonnées (x,y) et renvoie la distance parcourue (choisir le bon type d'argument)
j. une méthode void print() qui affiche les caractéristiques du point sous la forme : [identifier] (x,y).
   Exemple : [JK-857-FA] position : (34,10)
k. une méthode void reset() qui initialise la position à (0,0)

Partie 3 : Tests

1. Soit la fonction de teste suivante :

```
void vehiculeIdTest() {
   Point position(0,10);
   char identifier[] = "ML-888-VK";
   Vehicule vehicle(identifier, position);
   vehicle.print(); // affichage 1
   char * retrieveId = vehicle.getIdentifier();
   retrieveId[0] = 'X';
   vehicle.print(); // affichage 2
}
```

Si votre code compile, vous obtiendrez l'affichage 1 :
[ML-888-VK] Position : (0,10)

Mais l'affichage 2 produira :
[XL-888-VK] Position : (0,10)

Avez-vous une explication ?

Changez le prototype de getIdentifier() en :
const char* getIdentifier();

Ceci résout-il votre problème ? Pourquoi ?

2. Soit la fonction de test suivante :

```
void vehiculeIdTest2() {
   Point position(0,10);
   char identifier[] = "ML-888-VK";
   Vehicule vehicle(identifier, position);
   vehicle.print(); // affichage 1
   identifier[0] = 'X';
   vehicle.print(); // affichage 2
}
```

Vérifier si l'affichage est bien :
[ML-888-VK] Position : (0,10)

3. Essayer de prévoir l'affichage de fonction de test suivante et exécuter-là par la suite sur CLion pour vérifier vos réponses

```
void vehiculeCopyTest(){
    Vehicule v1("JK-857-FA", Point(0,0));
    Vehicule v2 = v1;
    v1.setIdentifier("XXX");
    v1.print();
    v2.print();
}
```

4. Essayer de prévoir l'affichage de fonction de test suivante et exécuter-là par la suite sur CLion pour vérifier vos réponses

```
void vehicleMoveTests(){
    Vehicule v1("JK-857-FA", Point(0,0));
    Vehicule v2("ML-888-VK", Point(0,10));

    Point p = v1.getPosition();
    float distance = 0;

    distance = v1.moveTo(v2);
    std::cout << "distance : " << distance << std::endl; // affichage 1
    v1.print(); // affichage 2

    p.setY(50);
    std::cout << p.getY();
    v1.print(); // affichage 3
    distance = v1.moveTo(p);

    std::cout << "distance : " << distance << std::endl; // affichage 4
    v1.print(); // affichage 5
    distance = v1.moveTo(0, 80);

    std::cout << "distance : " << distance << std::endl; // affichage 6
    v1.print(); // affichage 7
```

```
    v2.print(); // affichage 8
}
```