

Ecole Polytechnique de Sousse	
Examen pratique (Sujet 1/6)	
<i>Classe : 4 Info</i> <i>Matière : POO C++</i> <i>Enseignant : Sakka Rouis Taoufik</i> <i>Documents Autorisés : Non</i>	Session : Dec. 2024 A.U. : 2024/2025 Durée : 2H00 Nombre Total de Pages : 1




### Exercice 1 :

1) Écrivez une classe Cylindre comportant :

- comme données membres privées, le rayon, la hauteur et un donnée membre **statique** nb représentant le numéro du cylindre (le premier créé portera le numéro 1, le suivant le numéro 2, et ainsi de suite, ...)
- un constructeur simple et en ligne, qui permet l'initialisation de ses données membres non statiques. En outre, le constructeur affichera le numéro du cylindre créé ;
- un constructeur par recopie qui permet l'initialisation de ses données membres non statiques à partir d'un autre objet. En outre, le constructeur affichera le numéro du cylindre créé ;
- un destructeur qui affiche le numéro du cylindre détruit ;
- une fonction **amie** appelée surface, renvoyant un réel qui représente la surface de la base du cylindre ( $\pi * R^2$ );
- une méthode **amie** appelée volume, renvoyant un réel qui représente le volume du cylindre ( $\pi * R^2 * H$ );
- une fonction affiche qui affiche le rayon, la hauteur, la surface de la base et le volume d'un cylindre. Attention cette fonction n'est pas une fonction amie.

2) Écrivez une fonction main qui déclare 3 instances de type cylindre puis affiche leurs informations

NB. Vous devez définir la constante PI.

Ecole Polytechnique de Sousse	
Examen pratique (Sujet 2/6)	
<i>Classe : 4 Info</i> <i>Matière : POO C++</i> <i>Enseignant : Sakka Rouis Taoufik</i> <i>Documents Autorisés : Non</i>	Session : Dec. 2024 A.U. : 2024/2025 Durée : 2H00 Nombre Total de Pages : 1 

### Exercice 1 :

Un segment de droite est caractérisé par deux points qui représentent ses extrémités (extrémité gauche et extrémité droite). Un segment de droite peut avoir plusieurs comportements dont on cite:

- une méthode **SeDeplacer** qui déplace le segment selon les pas communiquées à la méthode comme arguments (sans changement de longueur)
- **une fonction amie** appelée **longueur**, renvoyant un réel qui représente la longueur d'un segment selon la formule suivante :

$$\sqrt{(\text{extrDroite.x} - \text{extrGauche.x})^2 + (\text{extrDroite.y} - \text{extrGauche.y})^2}$$

**NB.** **sqrt (x)** retourne la racine carré et **pow (x, y)** retourne  $x^y$

- une fonction **afficher ( )** qui affiche toutes ses caractéristiques (y compris sa longueur)
- 1) Définir la Classe Point (définir les attributs et les méthodes nécessaires)
  - 2) Définir la Classe Segment en implémentant les méthodes suivantes :
    - un constructeur en ligne qui prend en paramètre deux Points représentant ses extrémités. En outre, le constructeur affichera le numéro du segment créé ;
    - un constructeur par recopie qui permet l'initialisation de ses données membres non statiques à partir d'un autre objet. En outre, le constructeur affichera le numéro du segment créé ;
    - un destructeur qui affiche le numéro du segment détruit ;
  - 3) Écrivez une fonction main qui déclare 3 instances de type Segment puis affiche leurs informations

Ecole Polytechnique de Sousse	
Examen pratique (Sujet 3/6)	
<i>Classe : 4 Info</i> <i>Matière : POO C++</i> <i>Enseignant : Sakka Rouis Taoufik</i> <i>Documents Autorisés : Non</i>	Session : Dec. 2024 A.U. : 2024/2025 Durée : 2H00 Nombre Total de Pages : 1



### Exercice 1 :

Écrivez une classe Vecteur comportant :

- comme membres données privées : trois composantes de type double, et un donnée membre **statique** nb représentant le numéro du vecteur (le premier créé portera le numéro 1, le suivant le numéro 2, et ainsi de suite, ...)
- trois constructeurs :
  - l'un, sans arguments, initialisant chaque composante à 0,
  - l'autre, avec 3 arguments, représentant les composantes,
  - le dernier par recopie
- un destructeur qui affiche le numéro du Vecteur détruit ;
- une fonction homothétie pour multiplier les composants par une valeur fournie en argument.
- une fonction prod-scal qui fournit en résultat le produit scalaire de deux vecteurs,
- une fonction somme permettant de calculer la somme de deux vecteurs.
- une fonction amie module qui fournit le module d'un vecteur  $\sqrt{x^2 + y^2 + z^2}$
- une fonction affiche qui affiche le numéro, les coordonnées et le module d'un vecteur

Écrivez ensuite une fonction main qui permet de les tester.


Ecole Polytechnique de Sousse	
Examen pratique (Sujet 4/6)	
<i>Classe : 4 Info</i> <i>Matière : POO C++</i> <i>Enseignant : Sakka Rouis Taoufik</i> <i>Documents Autorisés : Non</i>	Session : Dec. 2024 A.U. : 2024/2025 Durée : 2H00 Nombre Total de Pages : 1



### Exercice 1 :

On vous demande de développer un programme en C++ pour gérer des comptes bancaires. Chaque compte est identifié par un numéro unique, possède un nom de titulaire, un solde, et un taux d'intérêt. Certains aspects doivent être implémentés avec des concepts spécifiques :

1. **Constructeur simple** : Permet de créer un compte bancaire avec un numéro de compte, un nom de titulaire, un solde initial et un taux d'intérêt.
2. **Constructeur par recopie** : Permet de copier un compte existant, tout en incrémentant le nombre total de comptes créés (à l'aide d'un attribut `static`).
3. **Destructeur** : qui affiche le numéro du Compte détruit ;
4. **Attribut `static`** : Utilisez un attribut `static` pour suivre le nombre total de comptes créés.
5. **Fonction amie** : Implémentez une fonction amie nommée `afficherSolde` qui affiche le solde d'un compte, même si cette fonction n'est pas une méthode de la classe.
6. **Fonction membre** : Implémentez une méthode `afficherDetails` qui affiche les détails d'un compte (numéro, titulaire, solde, taux d'intérêt).
7. Dans le `main`, créez au moins deux comptes avec le constructeur simple, copiez un compte à l'aide du constructeur par recopie, puis utilisez la fonction amie pour afficher le solde d'un compte.

Examen pratique (Sujet 5/6)	
<i>Classe : 4 Info</i> <i>Matière : POO C++</i> <i>Enseignant : Sakka Rouis Taoufik</i> <i>Documents Autorisés : Non</i>	Session : Dec. 2024 A.U. : 2024/2025 Durée : 2H00 Nombre Total de Pages : 1 <div>  </div>

### Exercice 1 :

On souhaite développer une application C++ pour gérer les produits d'un magasin. Chaque produit a un code unique, un nom, un prix et une quantité en stock. L'exercice doit exploiter les concepts suivants :

1. **Constructeur simple** : Pour créer un produit avec un code, un nom, un prix et une quantité initiale.
2. **Constructeur par recopie** : Permet de dupliquer un produit existant.
3. **Destructeur** : qui affiche le numéro du Produit détruit ;
4. **Attribut static** : Utilisé pour compter le nombre total de produits créés.
5. **Fonction amie** : Implémentez une fonction amie nommée `appliquerRemise` qui permet de réduire le prix d'un produit d'un pourcentage donné.
6. **Fonction membre** : Implémentez une méthode `afficherDetails` pour afficher les informations d'un produit.
7. Dans la fonction main, créez plusieurs produits, utilisez le constructeur par recopie pour dupliquer un produit, et appliquez une remise à un produit en utilisant la fonction amie.

Ecole Polytechnique de Sousse	
Examen pratique (Sujet 6/6)	
<i>Classe : 4 Info</i> <i>Matière : POO C++</i> <i>Enseignant : Sakka Rouis Taoufik</i> <i>Documents Autorisés : Non</i>	Session : Dec. 2024 A.U. : 2024/2025 Durée : 2H00 Nombre Total de Pages : 1



### Exercice 1 :

Vous êtes chargé de développer un programme en C++ pour gérer des étudiants et leurs notes. Chaque étudiant a un identifiant unique, un nom, une note moyenne, et un statut d'admission (admis ou non admis) déterminé en fonction de la note moyenne. Un compteur `static` pour compter le nombre total d'étudiants créés.

La classe `Etudiant` doit inclure les concepts suivants :

1. **Constructeur simple** : Pour initialiser les informations d'un étudiant.
2. **Constructeur par recopie** : Permet de copier un étudiant en conservant ses données.
3. **Destructeur** : qui affiche le numéro de l'étudiant détruit ;
4. **Attribut `static`** : Utilisé pour compter le nombre total d'étudiants créés.
5. **Fonction amie** : Implémentez une fonction amie nommée `comparerEtudiants` qui compare les moyennes de deux étudiants et affiche celui qui a la meilleure note.
6. **Fonction membre** : Implémentez une méthode `afficherDetails` pour afficher les informations d'un étudiant (y compris le statut admis ou non admis).

Dans le programme principal :

- Créez au moins deux étudiants avec le constructeur simple.
- Copiez un étudiant en utilisant le constructeur par recopie.
- Comparez les moyennes des deux premiers étudiants à l'aide de la fonction amie.
- Affichez le nombre total d'étudiants créés.