

C++

PROGRAMMATION D'UNE CLASSE (PART1)

Table des matières

<u>Exercice 1: expérimentation</u>	3
<u>Q1 : Définissez une instance de la classe Demo que vous appellerez D</u>	3
<u>Q2 : Dans le main, essayez successivement d'accéder à toutes les données et les fonctions définies dans la classe. Notez soigneusement ce à quoi vous avez accès ou pas et les messages d'erreur à la compilation</u>	3
<u>Q3 : Dans la définition de la classe, à l'intérieur de chacune des fonctions, appelez les données et les autres fonctions membres</u>	4
<u>Exercice 2: Programmation de la classe Date</u>	4
Q1. Définition des données membres d'une classe.....	4
Q2. Définition des constructeurs.....	4
Q3. Définition du destructeur.....	4
Q4. Définition de fonctions membres.....	4
Q5. Définition de fonction non membres.....	4
Q6. Et les vérifications ?.....	4

EXERCICE 1: EXPERIMENTATION

Q1 : Définissez une instance de la classe Demo que vous appellerez D

Q2 : Dans le main, essayez successivement d'accéder à toutes les données et les fonctions définies dans la classe. Notez soigneusement ce à quoi vous avez accès ou pas et les messages d'erreur à la compilation.

Exemple: dans le main essayez de modifier la donnée `donnee_private` et d'appeler la fonction `fonction_private()`.

lorsque nous tentons d'accéder aux données membre et aux fonctions privées, g++ indique :

```
experimentation.cpp:7:9: error: 'int Demo::donnee_private' is private
    int donnee_private;
    ^
experimentation.cpp:38:7: error: within this context
    D.donnee_private = 1;
    ^
experimentation.cpp:9:10: error: 'void Demo::fonction_private()' is private
    void fonction_private()
    ^
experimentation.cpp:40:24: error: within this context
    D.fonction_private();
```

quand in cherche à faire de même avec les données et fonction protected :

```
experimentation.cpp:15:9: error: 'int Demo::donnee_protected' is protected
    int donnee_protected ;
    ^
experimentation.cpp:44:7: error: within this context
    D.donnee_protected = 1;
    ^
experimentation.cpp:17:10: error: 'void Demo::fonction_protected()' is protected
    void fonction_protected()
    ^
experimentation.cpp:46:26: error: within this context
    D.fonction_protected();
    ^
```

quand aux données publiques :

ça marche !!!

Q3 : Dans la définition de la classe, à l'intérieur de chacune des fonctions, appelez les données et les autres fonctions membres.

Exemple: dans la fonction `fonction_protected` essayez de modifier la donnée `donnee_private` et d'appeler la fonction `fonction_private()`.

essai de modification de `donnee_private` dans `fonction_protected()` : ça fonctionne

essai d'appel de `fonction_private()` dans `fonction_public()` : ça marche

modification de `donnee_private` dans `fonction_public()` : ça marche

etc. tant qu'on reste au sein de la classe. il semble que nous ayons accès à tout

EXERCICE 2: PROGRAMMATION DE LA CLASSE DATE

Q1. Définition des données membres d'une classe

Une date est définie par un jour, un mois et une année qui seront représentés par des entiers.

Créez la classe `Date` qui contient ces données membres privées.

Q2. Définition des constructeurs

Définissez le constructeur par défaut qui met une date à l'origine du calendrier soit au 1er Janvier de l'an 00.

Définissez ensuite le constructeur par copie.

Enfin créez un constructeur avec paramètres dont au moins un des paramètres est un paramètre par défaut.

Q3. Définition du destructeur

Créez le destructeur de la classe `Date`.

Q4. Définition de fonctions membres

Définissez la fonction `Afficher` qui vous permet d'afficher la date au format : jour/mois/an

Définissez la fonction `Saisie` qui vous permet de saisir une date.

Créez la fonction `Modifier` qui vous permet de modifier la valeur d'une date par celle d'une autre date précisée en paramètre.

Créez ensuite la fonction `Comparer` qui indique si deux dates sont similaires ou différentes.

Q5. Définition de fonction non membres

Créez la fonction `Estbissextile` qui, pour une date donnée, indique si elle appartient à une année bissextile.

Q6. Et les vérifications ?

Avez-vous compilé le programme entre chaque question à la recherche d'erreur de syntaxe ?

Toutes les fonctions que vous avez codées, les avez-vous systématiquement appelées dans le main ?

Avez-vous pensé au cas limites ? Par exemple si le mois saisi vaut 13.

Envoyez-moi ce que vous avez fait à la fin de la séance et profitez de l'autoformation pour finir les questions qui vous manquent.