Cours Programmation C, C++ et Java
Magistère de mathématiques, 2e année
—
TD 9 : la bibliothèque

Luc Bougé, Hervé Jégou

22 novembre 2005

Ce TD est noté. Il est à rendre pour le 29 novembre dernier délai. Vous pouvez le rendre soit en version imprimée, soit par e-mail à l'adresse Herve.Jegou@irisa.fr en pièce jointe. Dans ce cas, j'accuserai réception de votre e-mail. Votre code doit être écrit dans un fichier unique. Le nom du fichier doit *impérativement* contenir les noms de votre binôme.

Pour la présentation, écrivez des commentaires dans votre code pour que je comprenne vos choix. Les commentaires sont entre les signes /* et */ en C et C++. Puisque le TD porte sur le C++, vous pouvez également commenter en utilisant le double slash // en début de ligne. Votre code source devra être indenté.

Les critères suivants seront largement pris en compte dans la notation :
- la qualité du code,
- la pertinence de vos choix,
- les tests effectués pour valider votre code,
- sa présentation, y compris les commentaires.

Remarque : pour toutes les méthodes que vous allez écrire, on vous demande d'utiliser le mot clé const *dès que son usage vous semble approprié.*

I. Modélisation d'une bibliothèque

Le but de ce TP est de modéliser les données relatives à une bibliothèque ainsi que les opérations associées. Pour cela, vous allez définir trois classes :
- la classe Abonne,
- la classe Livre,
- la classe Bibli.

Les livres et les abonnés seront repérés par un identifiant unique. Ces identificateurs seront respectivement identifiés par des variables de types Abonne_id et Livre_id. Vous les définirez au moyen de l'instruction typedef :

```
typedef int Abonne_id;
typedef int Livre_id;
```

II. La classe Livre

La classe Livre devra contenir les informations suivantes :
- l'identificateur du livre
- le titre du livre,
- le nom de l'auteur,
- l'identificateur abonné de l'emprunteur,
- le nombre de jour depuis le début de l'emprunt.

Dans le cas où le livre n'est pas emprunté, vous utiliserez un identificateur factice pour l'emprunteur :

```
const Abonne_id ABONNE_FACTICE = -9;
```

Question II.1. *Définissez pour la classe* Livre*,*
- *les attributs,*
- *le constructeur par défaut (sans attribut),*
- *un constructeur qui initialise l'objet à partir de ses attributs,*
- *les extracteurs.*

Remarque *: pour le constructeur qui initialise l'object à partir de ses attributs, utilisez une variable qui s'incrémente toute seule pour affecter l'identificateur du livre. Pour des raisons de simplicité, cette variable sera initialisée à 0.*

Question II.2. *Écrivez une méthode* tic() *qui augmente de* 1 *le nombre de jours d'emprunt du livre considéré.*

Question II.3. *Écrivez une méthode* en_retard *qui renvoie* true *si le livre est en retard et* false *sinon. Un livre est en retard si il est emprunté depuis un nombre de jours supérieur à une constante fixée.*

Question II.4. *Écrivez une fonction externe permettant l'affichage des informations d'un livre.*

III. La classe Abonne

La classe Abonne devra contenir les informations suivantes
- un numéro d'abonné,
- le nom du titulaire,
- le nombre de livre empruntés,
- la liste des livres empruntés.

La liste des livres empruntés sera défini comme un tableau de Livre_id et de taille MAX_EMPRUNTS, où MAX_EMPRUNTS sera défini à 3 en utilisant une directive du précompilateur. De la même manière que vous avez défini un abonné factice, déclarez un identificateur LIVRE_FACTICE pour signaler l'absence de livre.

Question III.1. *En suivant la même démarche que pour la classe* Livre*, définissez les attributs, constructeurs et extracteurs de la classe* Abonne*.*

Remarque *: l'extraction de l'élément i de la liste des livres empruntés se fera au moyen d'un extracteur prenant cet indice i en paramètre.*

Question III.2. *Écrivez une fonction qui affiche tous les renseignements ayant trait à un abonné.*

IV. La classe Bibli

La classe Bibli devra contenir les informations suivantes
- un nombre d'abonnés,
- un nombre de livres,
- la liste des abonnés,
- la liste des livres.

Les nombres d'abonnés et de livres seront contraints à être inférieurs ou égaux à des constantes, respectivement MAX_ABONNES et MAX_LIVRES.

Question IV.1. *Écrivez les attributs, extracteurs et constructeur de la classe* Bibli. *Par défaut, la bibliothèque ne contient ni livre ni abonné.*

Question IV.2. *Écrivez une fonction qui écrit les données relatives à une bibliothèque.*

Question IV.3. *Écrivez les fonctions* ajoute_livre *et* ajoute_abonne*, qui ajoutent respectivement un livre et un abonné à la bibliothèque. Ces fonctions pourront être appelées avec un appel de la forme*

```
int lid = B.ajoute_livre(Livre( "UNIX", "M. Wielsh", 12 ) );
```

où lid *représente l'identificateur du livre nouvellement ajouté.*

Question IV.4. *Écrivez les fonctions* print_list_livres() *et* print_list_abonnes() *qui affichent les noms des livres et abonnés de la bibliothèque.*

Dans la suite, les abonnés et les livres seront gérés, la plupart du temps, par leur numéro/identificateur.

Question IV.5. *Écrivez une fonction qui indique si un livre, -identifié par sa référence-, existe dans la bibliothèque.*

Question IV.6. *Écrivez une fonction* tic *qui incrémente le nombre de jour d'emprunt des livres de la bibliothèque qui sont effectivement empruntés.*

Note *: dans vos tests, utilisez cette fonction pour incrémenter le temps.*

Question IV.7. *Écrivez une fonction qui renvoie* true *si un abonné est autorisé à emprunter.*

Question IV.8. *Écrivez des fonctions* emprunte *et* retour *pour gérer les opérations d'emprunt et de retour de livre des abonnés.*

Question IV.9. *Écrivez des fonctions qui affichent respectivement la liste des livres en retard et la liste des abonnés non autorisés à emprunter. La fonction devra au moins afficher les identificateurs et les titres/noms des livres/abonnés concernés.*

V. Bandes dessinées

On souhaite désormais pouvoir définir des bandes dessinées. Elles se caractérisent par une durée d'emprunt autorisée plus courte que celle des livres normaux, et elles comptent pour deux livres lors du calcul du nombre de livres empruntés.

Question V.1. *En utilisant un mécanisme d'héritage, proposez et implantez une solution pour permettre la gestion de ce nouveau type de livre.*