

## TD 4 PROGRAMMATION ORIENTÉE OBJET EN C++

### Introduction

Le but de cette séance est d'étudier le mécanisme d'héritage et les méthodes virtuelles. On reprendra l'exemple de l'application de calcul des notes et on lui ajoutera un menu utilisant ces mécanismes.

On rappelle les différents cas d'utilisation identifiés pour l'application :

1. définition des matières que suit l'étudiant,
2. saisie des notes pour l'ensemble des matières,
3. calcul de la moyenne générale.

Vous devez d'abord récupérer les fichiers sources des classes *Matiere* et *Semestre* que vous avez créés dans le TD3. Ces fichiers seront utilisés dans la partie 2. Vous devez également **télécharger les fichiers sources** des classes gérant le menu qui se trouvent sur le portail (fichier TD4.zip). Ces fichiers seront utilisés dans les parties 1 et 2. A la fin de la séance, **vous devrez déposer l'ensemble des fichiers sources créés ou modifiés** sur le portail (les diagrammes ne seront pas déposés sur le portail).

### Partie 1 : analyse des classes gérant le menu

Décompresser les fichiers de l'archive TD4.zip dans votre dossier de travail, puis créer un projet *test\_menu* permettant de compiler le programme de test *test\_menu.cpp* (pour créer le projet et la solution correspondante dans *Visual .net*, utiliser l'option *Fichier-Nouveau-Projet à partir de code existant*).

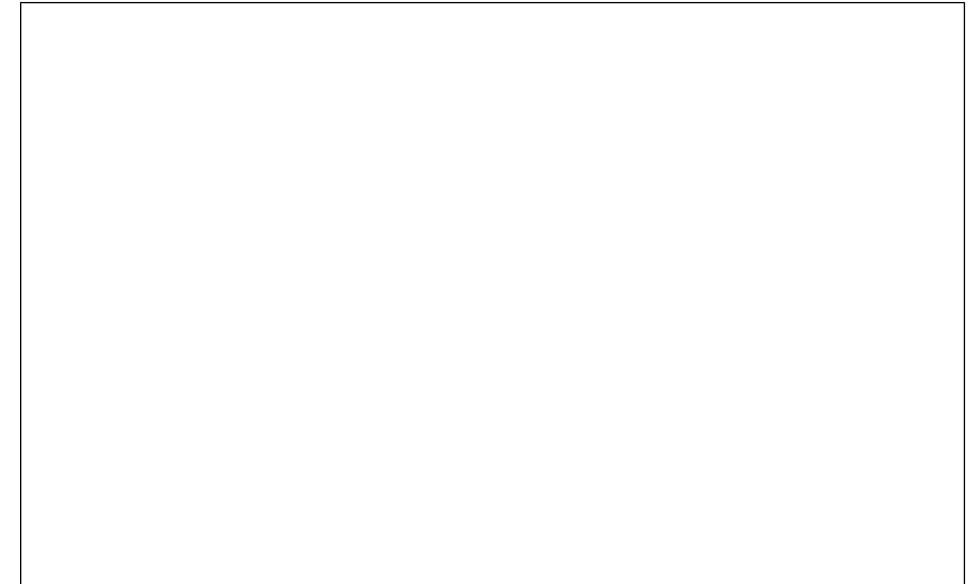
a) Compiler et exécuter ce programme de test.

b) Examiner les fichiers de déclaration et de définition des classes *Menu*, *OptionMenu* et *ExempleMenu* (fichiers *menu.h*, *menu.cpp*, *exemplmenu.h*, *exemplmenu.cpp*) et le programme de test (fichier *test\_menu.cpp*).

c) Quelle relation y a-t-il entre les classes *Menu* et *ExempleMenu* ? Préciser les méthodes redéfinies et les nouvelles méthodes.



d) Faire le diagramme des classes de l'application dans le cadre suivant en faisant figurer toutes les relations entre ces classes.



e) Analyser les méthodes qui sont appelées à l'exécution de l'instruction *m.executer()* du programme *main*.



f) Pourquoi la méthode *executerOption* de la classe *Menu* est virtuelle ? En déduire l'intérêt d'avoir choisi de faire dériver *ExempleMenu* de *Menu*.



## Partie 2 : construction de l'application

On veut utiliser la classe *Menu* et les classes *Matiere* et *Semestre* du TD3 pour faire l'application présentée en introduction. Pour cela, il nous faudra créer une classe dérivée de *Menu* et spécialiser cette classe pour la gestion du menu de notre application. La classe dérivée sera appelée *MenuMoyenne*. Elle sera faite sur le même modèle que la classe *ExempleMenu*. Dans toute cette partie, vous ne devrez **jamais modifier** les fichiers *menu.h* et *menu.cpp*.

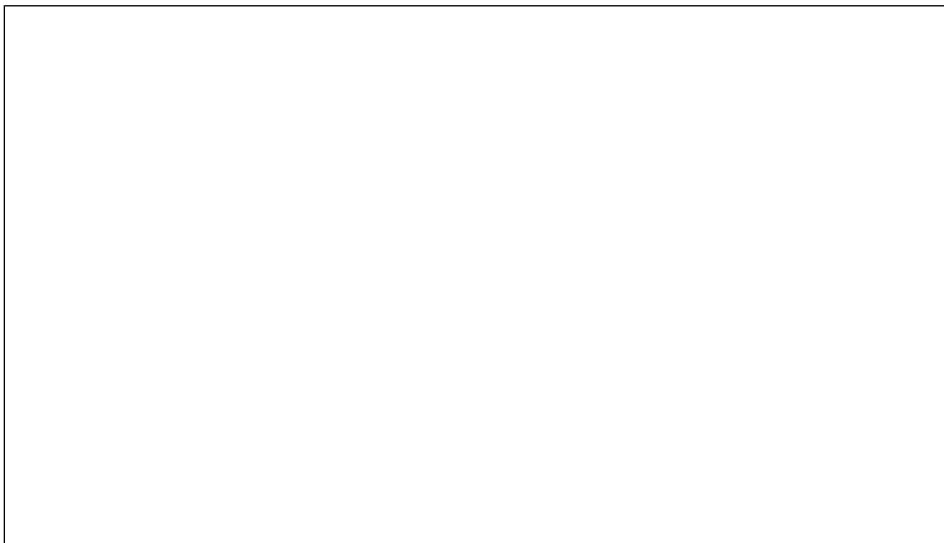
Copier les fichiers des classes *Matiere* et *Semestre* du TD3 dans un nouveau dossier de travail. Copier ensuite dans ce dossier les fichiers *menu.h*, *menu.cpp*, *exemplmenu.h*, *exemplmenu.cpp*, *test\_menu.cpp*. Renommer les fichiers *exemplmenu.h* et *exemplmenu.cpp* en *menumoyenne.h* et *menumoyenne.cpp* respectivement. Enfin, renommer le fichier *test\_menu.cpp* en *main.cpp*.

Créer un nouveau projet intitulé *application\_moyenne* contenant les fichiers précédents (utiliser à nouveau l'option *Fichier-Nouveau-Projet à partir de code existant*).

a) Renommer la classe *ExempleMenu* en *MenuMoyenne* et adapter les appels aux fichiers *.h*. Compiler et vérifier que l'exécution est identique à la partie précédente.

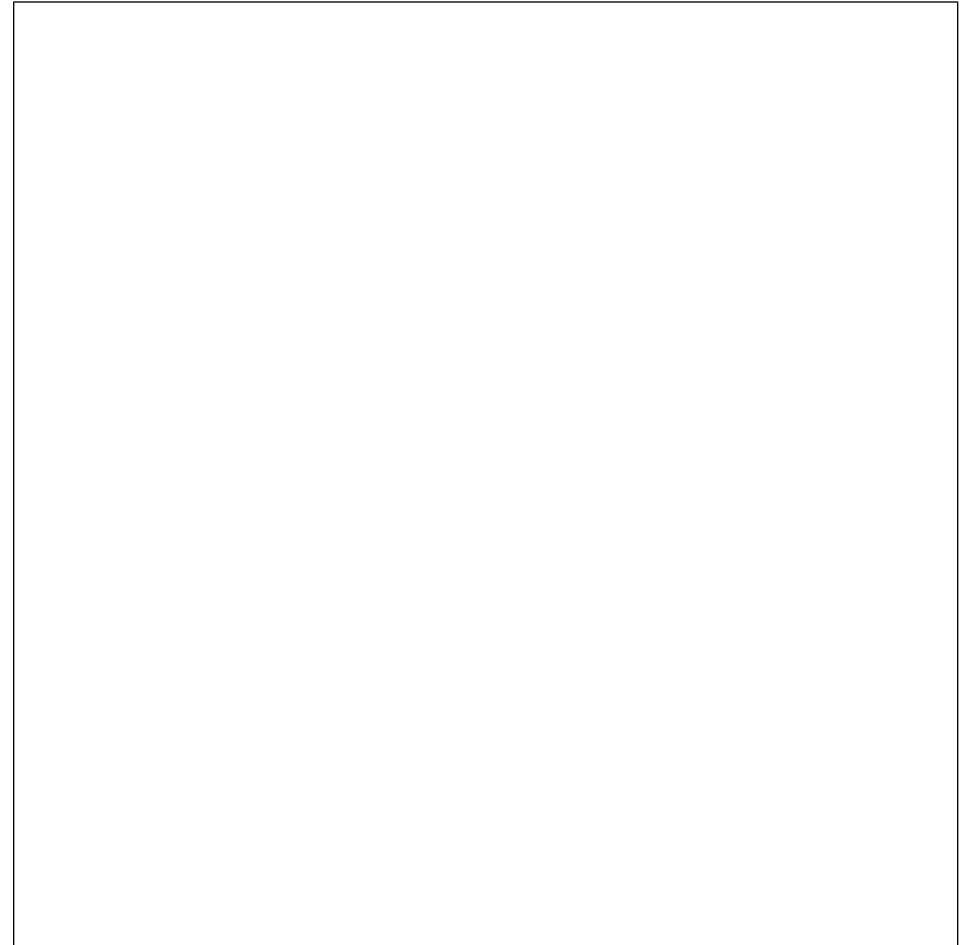
b) Nous allons maintenant personnaliser la classe *MenuMoyenne* pour mettre en place une nouvelle option pour la saisie des matières (cas d'utilisation 1). Pour cela, nous allons remplacer la méthode *nouvelleOption* par une méthode *saisirMatiere* chargée de la saisie des matières d'un objet *Semestre*. Faire les modifications correspondantes dans la classe *MenuMoyenne* et vérifier l'exécution.

c) Pour pouvoir utiliser un objet *Semestre* commun pour toutes les méthodes de la classe *MenuMoyenne*, nous allons mettre en place une association entre *MenuMoyenne* et *Semestre* : un objet *Semestre* sera déclaré dans le programme *main* et la classe *MenuMoyenne* aura un champ qui pointera sur cet objet *Semestre*. Faire le diagramme de classe de *MenuMoyenne* traduisant la relation d'association. Y faire figurer les champs et les méthodes de cette classe.



d) Modifier la classe *MenuMoyenne* et le programme *main* pour mettre en place cette association. Compiler et tester.

e) On veut maintenant ajouter les options permettant de gérer les autres cas d'utilisation (saisie des notes et calcul de la moyenne). Pour cela, il faudra créer de nouvelles méthodes dans *MenuMoyenne*. Faire le diagramme de classes complet de *MenuMoyenne* en faisant figurer ces nouvelles méthodes.



f) Déclarer et définir les nouvelles méthodes de la classe *MenuMoyenne*. Compiler et tester.