

TP 1
Structure de données (8INF259)

Instructions

- Pour faciliter la correction de vos programmes, il est recommandé de bien commenter vos programmes.
- Travaillez en équipe **au plus de quatre étudiant(e)s**, remettez une seule copie par équipe.
- Il est interdit pour une équipe de copier le travail d'une autre équipe.

=====
Objectifs du travail demandé

Le principal but de ce travail est de se familiariser avec les pointeurs et les classes dans le langage C++.

Il est donc demandé dans ce TP de concevoir un programme, écrit en C++, créant une **liste chaînée** en mémoire à partir de données contenues dans un fichier texte appelé '**donnees.txt**'. Ce dernier contient de l'information concernant des dossiers de professeurs.

Chaque dossier de professeur contient son id (identifiant), son nom, son prénom, une liste de cours qu'il souhaite enseigner et une liste de noms d'étudiants qu'il souhaite diriger.

Pour ce faire, il est demandé de programmer une classe **DossierProfesseur** qui crée, à partir des données contenues dans le fichier texte '**donnees.txt**', une liste chaînée (dont la structure est décrite plus loin). Cette classe permettra d'effectuer les opérations suivantes sur cette liste :

1. Charger les informations du fichier '**donnees.txt**'.
2. Supprimer un dossier professeur.
3. Afficher le profil d'un professeur avec toutes les informations : id, nom, prénom, les cours et les étudiant(e)s.
4. Afficher le nom du professeur ou des professeurs ayant le plus d'étudiants.
5. Afficher le cours ou les cours les plus demandés.

Les informations contenues dans le fichier texte '**donnees.txt**' ont le format suivant :

```
id
nom
prenom
cours 1
cours 2
.
.
cours n
# // ici # sert à séparer la liste des cours de la liste des étudiants
Etudiant 1
Etudiant 2
.
.
# //cette fois-ci # sert à passer au professeur suivant
```

- Le **id** est un entier pour identifier un professeur.
- Le **nom** et prénom sont des chaînes de caractères.
- Les **cours 1, cours 2, ...** etc sont des chaînes de caractères identifiant des sigles de cours.
- Utiliser les **sigles** suivants: INF100, INF101, INF102, INF103, ..., etc.
- Les valeurs de **Etudiant 1, Etudiant 2, ...**, etc sont des chaînes de caractères identifiant le nom des étudiants que souhaite diriger le professeur en question.
- Le caractère # sert la première fois à séparer les sigles de cours des noms des étudiants et la deuxième fois à séparer les dossiers de professeurs.

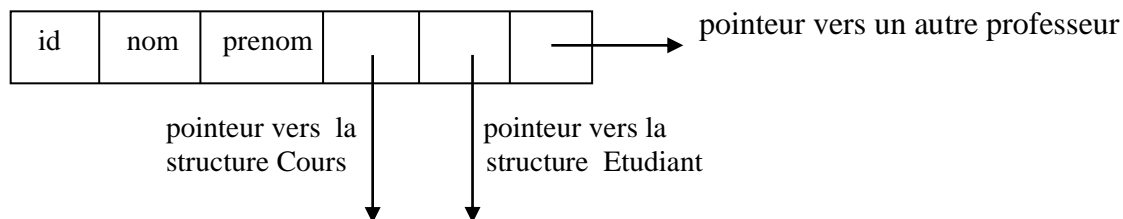
Votre programme lira, à partir d'un autre fichier texte **operations.txt**, les transactions à effectuer par votre programme dont le format est: op [param1]. Les opérations possibles sont comme ci-dessous.

- **id** : supprimer un professeur de la liste chaînée
- + **id** : afficher le profil du professeur
- >> : afficher le nom du ou des professeurs ayant le plus d'étudiants dans la liste chaînée.
- * : afficher le cours ou les cours les plus demandés dans la liste chaînée.

Les structures de données à utiliser sont les suivantes :

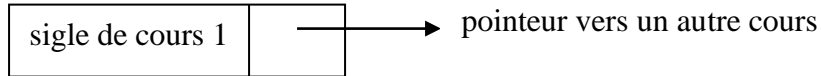
```
class Professeur {
public:
    int id;
    std::string nom;
    std::string prenom;
    Cours *listeCours;
    Etudiant *listeEtudiants;
    Professeur *suivant;
};
```

Graphiquement, cette structure est comme suit :



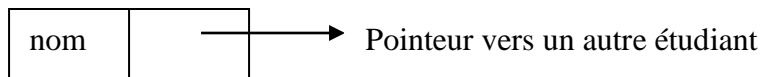
```
class Cours {
public:
    std::string sigle;
    Cours *suivant;
};
```

Graphiquement, cette structure est comme suit :



```
class Etudiant {  
    public:  
        std::string nom;  
        Etudiant *suivant;  
};
```

Graphiquement, la structure Etudiant est comme suit :



```
class DossierProfesseur {  
    public:  
        Professeur *tete;           // début de la liste chaînée  
        DossierProfesseur();         // Appele la fonction chargerProfesseur  
        ~DossierProfesseur();        // Détruit de la liste chaînée existant en mémoire.  
        void chargerProfesseurs();    // Construit la structure de la liste chaînée en mémoire à  
                                     // partir du fichier donnees.txt  
        void supprimer(int id);       // supprime de la liste chaînée un professeur donné  
        void profil(int id) const;    // affiche le profil du professeur identifié pour id  
        void afficherProfPlusEtudiants() const; // affiche les noms des professeur ayant le plus  
                                               // d'étudiants  
        void afficherCoursPlusDemande() const; // affiche les cours les plus demandés de la liste  
};
```