

R 6 . A . 0 5 / C + +

T P 3

**L a c l a s s e s t r i n g –
f o r m a t a g e – I / O f i c h i e r s**

2 0 2 4 - 2 0 2 5

Pour ce TP3, on créera sous *VSCode* un nouvel espace de développement « *ProjetTP3* » avec l'arborescence habituelle :

/ProjetTP3

/		\
/src	/ws	/bin
*.cpp	Makefile	*.bin
*.h		*.o

Exercice N°1 (durée 30 minutes)

Ecrire une classe *CConvert* permettant de manipuler des chaînes de caractères et des réels. Le but est de stocker comme attributs le réel à la fois sous forme numérique ET sous forme de chaîne de caractères : utiliser obligatoirement les classes *istringstream*, *ostringstream* vues en cours.

Cette classe prendra comme attributs :

<i>string</i>	<i>m_str;</i>
<i>double</i>	<i>m_dbl;</i>

Ces attributs DOIVENT être initialisés à la même valeur (tous les 2) à chaque construction.

Cette classe contiendra les méthodes et constructeurs ci-dessous :

CConvert (string theCh); // initialiser *m_str* ET *m_dbl* à la valeur passée en paramètre

CConvert (double theD); // initialiser *m_str* ET *m_dbl* à la valeur passée en paramètre

CConvert (); // ce constructeur met le réel à zéro et la chaîne de caractères à "" (chaîne vide)

double getReel () ; // accesseur

string getString () ; // accesseur

Le test unitaire de la classe se fera dans un fichier *testCConvert.cpp* qui contiendra un lanceur et toutes les fonctions de test unitaire des méthodes (cas normaux uniquement).

Exercice N°2 (durée 60 minutes)

Ecrire une classe *CManFch* de manipulation d'un fichier de données. Le nom complet d'un fichier est constitué du chemin d'accès (absolu) et de son nom (y compris l'extension). On considère que l'extension *.txt* désigne un fichier texte tandis que toute autre extension est un cas d'erreur. Utiliser obligatoirement la classe *string* pour extraire les différentes parties du nom complet d'un fichier. Le test unitaire de cette classe se fera comme pour la classe *CConvert* (fichier *testCManFch.cpp*).

Cette classe prendra comme attributs :

```
string    m_nomFich;           // uniquement le nom (y compris l'extension)
string    m_emplcmtFich;       // le chemin d'accès absolu (sans le nom de
fichier)
```

Elle contiendra les méthodes et constructeurs ci-dessous :

```
CManFch ( string str );        // str = nom complet du fichier (y compris le
chemin)
CManFch () ;                  // chaînes de caractères à ""
void setNomFichier ( string str ); // str = nom complet du fichier (y compris le
chemin)
void afficherAttributs ();      // affiche les valeurs des attributs de l'objet
CManFch
void afficherFichier ();        // explication ci-dessous
string getLigne ( ... );        // explication ci-dessous
```

La méthode *void afficherFichier ()*

Effectue l'affichage du fichier ligne par ligne en numérotant les lignes selon le format ci-dessous.

```
1 :   xxxxxxxxxxxxxxxxxxxxxxxxx
2 :   yyyyyyyyyyyyyyyyyyyyyyy
3 :   zzzzzzzzzzzzzzzzz
etc...
```

La méthode *string getLigne (int numLign1, int numLign2)*

Retourne une seule ligne de texte. Cette ligne correspond à la concaténation des lignes numérotées *numLign1* et *numLign2* dans le fichier texte (*1 <= numLign1* et *1 <= numLign2*).

Méthodes de la classe *string* très utiles

- *size_t find (const string& str)* ou *size_t find (const char* s)* : renvoie la constante prédéfinie *string::npos* si et seulement si la chaîne *str* ou *s* passée en paramètre ne se trouve PAS dans l'objet *string*. Il faut donc comparer le résultat renvoyé avec *string::npos*. Sinon, la méthode renvoie la position (entre zéro et *size()-1*) de la première occurrence de la chaîne *str* ou *s* dans l'objet *string*.
- *size_t find_last_of (const string& str)* ou *size_t find_last_of (const char* s)* : renvoie la **dernière** position (≥ 0) dans l'objet *string* de la chaîne *str* (ou *s*) passée en paramètre. Renvoie la constante prédéfinie *string::npos* si et seulement si la chaîne *str* ou *s* passée en paramètre ne se trouve PAS dans l'objet *string*.
- *string substr (int first, int length)* : renvoie une **sous-chaîne** de l'objet *string*. Cette sous-chaîne commence à l'indice *first* (≥ 0) et est de longueur égale à *length* caractères.

Références nécessaires

- Les flux d'entrées/sorties « cin » et « cout » instances respectives des classes de la bibliothèque standard C++, « istream » et « ostream ».
- La définition de classe en C++.
- Le constructeur de copie.
- Le cours C++ **et en particulier l'usage de la classe *ifstream***.

De l'aide sur C++ est disponible sur les sites suivants :

<http://www.cplusplus.com/ref>
<http://www.cppreference.com/cppstring>
<http://www.cplusplus.com/reference/iostream>