

Laboratoire 1

Intro à Visual Studio et révision

Étape 1 : Comprendre et corriger un programme

Créer un projet vide du nom de calculSomme et y copier le programme ci-dessous. Ce programme lit une série d'entiers dans un fichier texte séquentiel, fait la somme de ceux-ci et l'affiche dans un fichier texte de sortie.

Il n'est pas fonctionnel pour le moment, vous devez le corriger. Pour le tester, **créez un fichier texte** dans le même répertoire que votre .cpp et mettez y des entiers séparés par des espaces ou des sauts de ligne et nommez-le **nombre.txt**. Choisissez des nombres pour avoir de bons tests.

```
/*
Programme:      calculSomme.cpp
Acteur:         Julie Gagnon
Date de création: 21/01/2019
But du programme: lit des nombres dans un fichier en input, calcul la somme et
                  l'affiche en console et dans un fichier en sortie.

#include <iostream>
#include <fstream>      //bibliothèque pour la manipulation de fichiers
using namespace std;

void main()
{
    int    nb,          //nombre lu dans le fichier
          total;        //total des nombres lu dans le fichier

    ifstream entree;      //variable ifstream
    entree.open("nombre.txt"); //ouverture du fichier

    ofstream sortie("sortie.txt"); //variable ofstream et ouverture du fichier

    while(entree)          //tant que le fichier n'est pas la fin
        entree >> nb;      //lecture dans le fichier
        total += nb;

    sortie << "La somme des nombres du fichier nombre.txt est" total;
    cout << "La somme des nombres du fichier nombre.txt est" total;

    entree.close();
    sortie.close();
}
```

Quand tout est fonctionnel, vous devez aussi ajouter 2 validations

- Tester si le fichier existe juste après avoir tenté de l'ouvrir.
- Tester si le fichier est vide après avoir tenté une première lecture.

Dans les 2 cas, vous devez afficher un message d'erreur complet et significatif dans le fichier texte de sortie et à l'écran. Ne pas afficher de somme dans ces 2 cas.

Dans le programme précédent, vous utiliser

- des flux (stream) d'entré/sortie dans les fichiers
 - **ifstream** pour input file stream. Pour les fichiers en lecture seulement
 - **ofstream** pour output file stream. Pour les fichiers en écriture seulement
 - y a aussi **ifstream** pour les fichiers input-output. Ceux où on lit et écrit.
- Les méthodes **open** et **close**
 - servent pour ouvrir et fermer les fichiers.
 - Voici d'autres méthodes utiles
 - **is_open**, **good**, **fail** sont pratique pour savoir si le fichier a été trouvé
 - **eof** pour savoir si on est à la fin de fichier.

```
while(!entree.eof()) //tant que cé pas la fin de fichier
entree >> nb;
```
 - On peut aussi tester l'état du flux, si l'extraction fonctionne, le flux est true.

```
entree >> nb;
while(entree) //tant que le stream est vrai, on
entree >> nb; //n'est pas rendu à la fin de fichier
```

Étape 2 : Ajouter des fonctionnalités au programme de l'étape 1

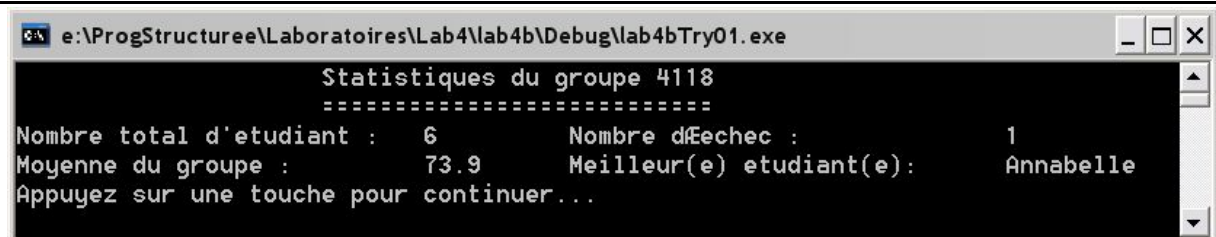
Modifier le programme précédent pour qu'il lise dans un fichier le prénom d'un étudiant ainsi que sa note finale (qui peut avoir des fractions). On veut afficher **à l'écran et dans un fichier** texte les statistiques sur les étudiants ci-dessous.

Créer le fichier `etudiant.txt` avec les données suivantes.

```
Roger 80
Bruno 65
Robert 82.5
Caroline 50
Annabelle 92.7
Sylvie 73.2
```

Le programme devrait afficher **à l'écran et dans un fichier** texte les renseignements suivants.

Statistiques du groupe 4118			
=====			
Nombre total d'etudiants :	6	Nombre d'echecs :	1
Moyenne du groupe :	73.9	Meilleur(e) etudiant (e) :	Anabelle



Reproduire intégralement l'affichage suivant à l'écran avec **des setw, left, right et setprecision** de la librairie `iomanip` pour établir les espaces vides entre les données à afficher. C'est un peu difficile à comprendre comment l'utiliser au début, mais c'est très simple et surtout très pratique si on veut rester en C++.

Vous devez faire une fonction pour

- ouvrir le fichier, tester s'il a été ouvert, écrire un message s'il n'est pas ouvert et quitter le programme s'il n'a pas été ouvert.
- afficher les résultats. Cette fonction sera appelée 2 fois dans le main comme les informations sont affichés 2 fois à des endroits différents.

Montrez-moi votre programme terminé avant la fin du cours ou avant le cours où sera présenté le lab. 2. (semaine 2)