

C++

Des string et des stream

Entrainement

Exercise 1 Lecture

Soit la variable `name` de type `string`. La commande `cin >> name` permet de lire un `string` sur l'entrée standard et de le ranger dans `name`. Lors de la lecture d'un `string`, les caractères blancs en début de chaînes sont éliminés (espaces, tabulation, fin de fichier ...). La chaîne de caractères est lue jusqu'au prochain blanc ou la fin de fichier.

Prédire le comportement du programme suivant lorsque l'on entre deux noms (par exemple : "Rolling Stones"), puis essayer-le.

```
#include <iostream>
#include <string>

int main()
{
    std::cout << "Comment vous appelez vous ? ";
    std::string name;
    std::cin >> name;

    std::cout << "Salut " << name << std::endl
              << "Et quel est le votre ? ";
    std::cin >> name;
    std::cout << "Enchante " << name << std::endl;

    return 0;
}
```

Exercise 2 Adieu printf et scanf

Ecrire le programme suivant (correct en C comme en C++), en ne faisant appel qu'aux entrée/sorties de C++.

```
#include <stdio.h>
int main()
{
    int n; float y;
    printf("Donnez un entier et un flottant\n");
    scanf("%d %e", &n, &y);
    printf("Le produit de %d par %e\n est : %e", n, y, n*y);
    return 0;
}
```

Exercice 3 Calculer une moyenne

Écrire un programme qui calcule la moyenne à une UE d'un étudiant. Le programme demandera le nom de l'étudiant puis sa note de partiel, sa note de l'examen final et son ensemble de notes de devoirs maisons. Il affichera ensuite le résultat tel que la note de partiel compte pour 20%, l'examen pour 40% et la moyenne des devoirs pour 40% de la note finale.

1. Ecrire le programme sans tenir compte des devoirs maison (en prenant 40% partiel, 60% exam).
2. Modifier la précision du résultat à 4 chiffres en utilisant la fonction `setprecision` de la bibliothèque `iomanip`
3. Ajouter la saisie et le calcul de la moyenne des devoirs maisons. Le nombre de devoir maisons n'est pas connu a priori (et on a pas besoin de le demander).

Remarque : C-d représente une fin de flux.

Exercice 4 Lecture/Ecriture dans un fichier

La bibliothèque `fstream` donne la possibilité d'ouvrir (`open(nom, mode d'ouverture)`) ou de fermer (`close()`) un fichier et de l'utiliser comme un stream. On peut préciser le mode d'ouverture :

- lecture : `fstream::in`
- écriture : `fstream::out`

1. Ecrire un programme qui compte le nombre de mots dans le fichier `travail.txt`. Vérifier le résultat avec la commande UNIX `wc`.
2. Ecrire un programme qui lit le fichier `prenom.txt` et qui recopie les noms contenant la lettre "a" dans le fichier `prenom_A.txt`.