



TP 1 : Programmation C++

Exercice 1 :

Écrire un programme qui affiche la moyenne entre 2 point

- Dans cet exemple, la fonction b a 2 paramètres de type double, nommées x et y. La fonction b renvoie un double. Cette fonction renvoie la moyenne de x et de y. La variable m est une variable locale à la fonction b.
- Dans le programme principal, on récupère dans a la moyenne de 3.2 et 4.2 en appelant la fonction b.

Attention n'oublier pas !

```
#include<iostream>
using namespace std;
```

Exercice 2 :

Écrire un programme qui remplit un tableau de 4 cases et lui affiché

- Lors de l'appel saisir(a), il y a identification du tableau a et du paramètre t de la fonction saisir : toute modification de t modifie le tableau a. La fonction saisir a permet de demander à l'utilisateur de saisir une à une toutes les cases d'un tableau de n cases.
- La fonction affiche à un tableau de n entiers en paramètres et affiche toutes les cases de ce tableau.
- Le programme est structuré : il est constitué d'un ensemble de fonctions courtes dont le rôle peut être facilement identifié.
- Le main() devient un programme très court !

Exercice 3 :

Écrire un programme qui fait l'échange entre la valeur de deux réels.

- Premièrement écrire la fonction echange(a,b) qui prend en paramètres deux variables de type double
- La fonction main() ou il a l'initialisation des deux variables et un affichage avant et après l'échange()
- Que vous remarquez ? Refait le même programme en utilisant passage par référence.

Exercice 4 :

Écrire un programme qui affiche la note moyenne d'après des notes stockées dans un tableau statique

- Déclarer un tableau statique de 6 cases
- Une fonction moyenne qui calcule et affiche la moyenne du tableau.

Exercice 5 :

Refait l'exercice 4 avec les tableaux dynamique

- Déclarer un tableau dynamique vide.
- Ajouter des cases avec les notes.
- Une fonction moyenne qui calcule et affiche la moyenne du tableau.

Attention n'oublier pas

```
#include <vector>
```

Exercice 6 :

Écrire un programme qui calcule milieu entre deux points.

- Dans cet exemple, nous définissons une structure point composée de 2 réels x et y.
- Dans la fonction main, on définit 3 points a, b et c.
- On fixe les coordonnées du point a à (3.2, 6.4).
- On demande à l'utilisateur de saisir les coordonnées du point b.
- On calcule dans c les coordonnées du milieu du segment [ab].
- On affiche finalement les coordonnées du point c.

Exercice 7 :

Fonctions manipulant des points

Sur le type point, on peut envisager d'effectuer les opérations suivantes :

- saisir les coordonnées d'un point au clavier.
- afficher les coordonnées d'un point.
- calculer la distance entre 2 points.

```
dx = a.x - b.x;  
dy = a.y - b.y;  
sqrt( dx*dx + dy*dy );
```

- calculer le milieu de 2 points.

```
m.x = (a.x + b.x) / 2;  
m.y = (a.y + b.y) / 2;
```

- Nous allons donc créer 4 fonctions nous permettant d'effectuer des manipulations sur les points.

Exercice 8 :

On va maintenant réaliser un programme plus complexe : un service du personnel veut gérer des employés qui possèdent chacun un nom, un prénom et un salaire. On veut manipuler une liste d'employés grâce au menu suivant :

1. Ajouter un employé
2. Afficher la liste
3. Rechercher un employé
4. Quitter

Explication :

Un employé est défini par un nom, un prénom et un salaire. Le nom et le prénom comporteront au maximum 9 caractères utiles. Lorsqu'on rajoute un employé dans la liste, on le rajoute à la fin de la liste.

Lorsqu'on affiche la liste des employés, on l'affiche dans le même ordre que celui de la saisie.

Lorsqu'on recherche un employé, on saisit un nom, et on affiche toutes les caractéristiques de tous les employés portant ce nom.

Structure du programme :

Il y aura 2 fonctions : une pour saisir les caractéristiques d'un employé et une autre pour les afficher à l'écran.

On définira également 4 fonctions pour manipuler cette liste :

- une fonction pour initialiser la liste à la liste vide.
- une fonction pour afficher une liste d'employés.
- une fonction pour ajouter un employé dans la liste.
- une fonction pour rechercher tous les employés portant un nom donné.

Aussi une interface de gestion'

- une fonction menu qui contient la boucle principale de notre menu.
- une fonction choix qui affiche le menu et saisit le choix de l'utilisateur.
- une fonction traiter_choix qui effectue la manipulation souhaitée par l'utilisateur sur la liste.