

## Travaux Pratiques 2 de la Programmation Avancée en C

### Exercice 1 :

Écrire une fonction *distance* ayant comme paramètres 4 doubles  $xa, ya$  et  $xb, yb$  qui représentent les coordonnées de deux points  $A$  et  $B$  et qui renvoie la distance  $AB$ . Tester cette fonction.

### Exercice 2 :

Ecrire deux fonctions C, l'une utilisant un algorithme itératif, l'autre un algorithme récursif, permettant de calculer, l'entier naturel  $n$  étant donné en entrée, la somme  $1 \times 2 + 2 \times 3 + 3 \times 4 + \dots + n \times (n + 1)$ .

### Exercice 3 :

Écrivez un programme en C qui utilise les pointeurs pour manipuler un tableau d'entiers. Le programme doit inclure les fonctions suivantes :

1. `void remplir(int* tab, int taille)` : une fonction qui remplit le tableau avec des valeurs entières de 1 à `taille`.
2. `void afficher(int* tab, int taille)` : une fonction qui affiche le contenu du tableau.
3. `void doubler(int* tab, int taille)` : une fonction qui double les valeurs de chaque élément du tableau.
4. `int somme(int* tab, int taille)` : une fonction qui retourne la somme des éléments du tableau.

### Exercice 4 : (*Algorithme de cryptage*)

On choisit un décalage (par exemple 5), et un  $a$  sera remplacé par un  $f$ , un  $b$  par  $g$ , un  $c$  par un  $h$ , etc.. On ne cryptera que les lettres majuscules et minuscules sans toucher ni à la ponctuation, ni à la mise en page. On supposera que les codes des lettres se suivent de  $a$  à  $z$  et de  $A$  à  $Z$ .

1. Déclarer un tableau de caractères *Message* initialisé avec le message en clair.
2. Ecrire une fonction *crypt* de cryptage d'une chaîne de caractères.
3. Ecrire une fonction *decrypt* de décryptage de message.
4. Ecrire le programme principal *main* qui permet de tester les deux fonctions.