Langace C - Travaux Dirigés n° 2

— Licence Informatique —

▶ **Exercice 1 (Recherche du maximum)**
Écrire une fonction `max` qui recherche la valeur maximale contenue dans un tableau de flottants (`float`). La fonction prendra en paramètre le tableau et sa taille. Elle retourna la valeur maximale. Inclure cette fonction dans un programme qui remplit un tableau avec des valeurs pseudo-aléatoires et qui l'utilise.

▶ **Exercice 2 (Recherche d'un élément)**
Écrire une fonction `search` qui recherche une valeur particulière dans un tableau de flottants (`float`). La fonction prendra en paramètre le tableau, sa taille et la valeur à rechercher. Elle retourna une valeur entière qui sera une position de la valeur dans le tableau si elle est présente dans le tableau et -1 sinon. Inclure cette fonction dans un programme qui saisit les valeurs du tableau et la valeur à rechercher.

▶ **Exercice 3 (Inversion d'un tableau)**
Écrire une fonction `inverse` qui renverse un tableau de flottants. Cette fonction prendra en paramètre le tableau et sa taille. Après l'appel à la fonction `inverse`, le tableau contient les mêmes valeurs mais placées dans l'ordre inverse. Ainsi si le tableau contient les valeurs $\{2.1, 3.4, 5.6, 7.8, 9.1, 7.4\}$, il devra ensuite contenir $\{7.4, 9.1, 7.8, 5.6, 3.4, 2.1\}$.

▶ **Exercice 4 (Insertion d'un élément)**
Écrire une fonction `insert` qui insère une valeur dans un tableau de flottants triés. Cette fonction prend en paramètre un tableau, le nombre d'éléments dans le tableau et une valeur v. Elle suppose que les éléments du tableau sont triés par ordre croissant. Elle insère alors la valeur v dans le tableau de telle sorte que les éléments soient encore triés par ordre croissant dans le tableau.

▶ **Exercice 5 (Triangle de Pascal)**
On s'intéresse encore aux coefficients du binôme

$$C_n^p = \frac{n!}{(n-p)! \times p!}$$

Ceux-ci vérifient par la relation de récurrence suivante

$$C_n^p = \begin{cases} 1 & \text{si } p = 0 \text{ ou } p = n \\ 0 & \text{si } p < 0 \text{ ou } p > n \\ C_{n-1}^p + C_{n-1}^{p-1} & \text{sinon} \end{cases}$$

On veut afficher le triangle de Pascal complet jusqu'à l'ordre N. Résoudre ce problème en utilisant chacune des trois structures suivantes :

- un tableau `T` à deux dimensions de taille $N \times N$ (`T[i][j]` contient C_i^j).
- deux tableaux simples de taille N.
- un tableau simple de taille N.