# Exercices sur les tableaux et les pointeurs

## 1 Triangle de Pascal

Créer un tableau triangulaire à deux dimensions dont l'élément d'indice $i, j$ est égal au coefficient binomial $\binom{i}{j}$ où $0 \leq i < n$, $0 \leq j \leq i$ et $n$ est saisi au clavier ; puis l'afficher à l'écran.

## 2 Concaténation de deux tableaux

1. Créer deux tableaux d'entiers de tailles respectives $n_1$ et $n_2$ où $n_1$ et $n_2$ sont deux entiers entrés au clavier.
2. Initialiser les éléments de ces tableaux avec des valeurs aléatoires. Pour plus de lisibilité, on prendra le reste du résultat de la fonction `rand` par une constante de taille raisonnable
3. Créer un troisième tableau de taille $n_1 + n_2$, correspondant à la concaténation des deux premiers.

## 3 Ajout d'un élément à un tableau

1. Créer un tableau d'entiers $T$ de taille $n$ où $n$ est entré au clavier. Initialiser chacun de ses éléments à une valeur aléatoire.
2. Créer un tableau de taille $n + 1$ correspondant au tableau précédent auquel on a ajouté un élément saisi au clavier.
3. Remplacer le premier tableau $T$ par le tableau à $n + 1$ éléments créé à l'étape précédente.

## 4 Transposition d'une matrice

1. Créer une matrice de taille $k \times n$ où $k$ et $n$ sont entrés au clavier. Initialiser cette matrice par des valeurs aléatoires (comme à l'exercice 2).
2. Créer une seconde matrice correspondant à la transposée de la première.

## 5 Produit de deux matrices

1. Créer deux matrices d'entiers de tailles respectives $n_1 \times n_2$ et $n_2 \times n_3$ où ces trois paramètres sont entrés au clavier. Initialiser ces deux matrices avec des valeurs aléatoires comme à l'exercice 2.
2. Créer une troisième matrice correspondant au produit des deux premières.

## 6 Combinaisons linéaires des lignes d'une matrice binaire

1. Créer un tableau à deux dimensions contenant les coefficients d'une matrice lue dans un fichier. La matrice sera décrite dans un fichier ayant la forme suivante : la première ligne contient le nombre de lignes de la matrice, la deuxième ligne son nombre de colonnes ; les lignes suivantes correspondent aux coefficients. On utilisera l'opérateur de redirection d'UNIX "<" pour lire le fichier.

2. Ecrire un programme qui affiche toutes les combinaisons linéaires des lignes de cette matrice. Pour énumérer les $2^k$ combinaisons linéaires des lignes de la matrice à $k$ lignes, on se servira de la méthode suivante utilisant le *code de Gray*. Cet algorithme permet d'énumérer tous les vecteurs de $\mathbf{F}_2^k$ en ne modifiant qu'une seule coordonnée à chaque étape. Les vecteurs de $\mathbf{F}_2^k$ sont représentés par $k$ bits, $(g_0, \ldots, g_{k-1})$. On ajoute à ce vecteur une composante $g_k$ vérifiant à tout moment $g_k = \sum_{i=0}^{k-1} g_i \bmod 2$.

   INITIALISATION :
   $$g_0 = g_1 = \ldots = g_{k-1} = g_k = 0 \ .$$

   A CHAQUE ÉTAPE :

   - Si $g_k = 0$, modifier $g_0$.
   - Sinon :
     - $i \leftarrow$ le premier indice tel que $g_i \neq 0$.
     - si $i \neq k-1$, modifier $g_{i+1}$.
     - si $i = k-1$, modifier $g_{k-1}$.

   Remettre à jour $g_k$.

   Ainsi, les vecteurs de $\mathbf{F}_2^2$ seront énumérés dans l'ordre suivant : (0,0) - (1,0) - (1,1) - (0,1)

   On peut utiliser cette méthode pour énumérer toutes les combinaisons linéaires de $k$ lignes. L'indice de la coordonnée à modifier à chaque étape de l'algorithme précédent correspond à l'indice de la ligne à ajouter à la combinaison linéaire précédente. Par exemple, pour $k = 2$, on énumérera les combinaisons linéaires dans l'ordre suivant : vecteur nul, ligne 1, ligne 1 + ligne 2 , ligne 2.