Université Hassan II
Faculté des Sciences Ain Chock – Casablanca
Département Mathématique et Informatique

Série de TD N° 7 Les fonctions & matrices	Informatique INFAPP S1

Exercice 1 :

Ecrire une fonction qui calcule le factoriel d'un entier passé en paramètre.
Ecrire une fonction qui calcule et retourne le **coefficient binomial C(n,p).** $\mathbf{C_n^p}$.
Ecrie une fonction main pour calculer C_n^p pour deux nombres entrés par l'utilisateur.

Solution :

```c
#include <stdio.h>

// Fonction pour calculer le factoriel d'un entier
long factoriel(int n) {
    long long result = 1;
    for (int i = 1; i <= n; i++) {
        result *= i;
    }
    return result;
}

// Fonction pour calculer C(n, p)
long coefficient_binomial(int n, int p) {
    if (p > n) {
        return 0;  // Le coefficient binomial est nul si p > n
    }
    return factoriel(n) / (factoriel(p) * factoriel(n - p));
}

int main() {
    int n, p;

    // Demander à l'utilisateur d'entrer les valeurs de n et p
    printf("Entrez un nombre entier n (n >= 0): ");
    scanf("%d", &n);

    printf("Entrez un nombre entier p (p >= 0 et p <= n): ");
    scanf("%d", &p);

    // Vérification des entrées
    if (p < 0 || p > n) {
        printf("Erreur : p doit être compris entre 0 et n.\n");
        return 1;
```

```
    }

    // Calcul du coefficient binomial C(n, p)
    long result = coefficient_binomial(n, p);

    // Affichage du résultat
    printf("Le coefficient binomial C(%d, %d) est : %ld\n", n, p, result);

    return 0;
}
```

Exercice 2 :

Écrire un programme en C qui fait ce qui suit :

1. Créez une fonction **int somme(int T[], int taille)** qui retourne la somme des éléments du tableau T de taille taille.
2. Créez une fonction **int moyenne(int T[], int taille)** qui retourne la moyenne des éléments du tableau T. La moyenne est calculée en appelant la fonction somme pour obtenir la somme des éléments et en la divisant par la taille du tableau.
3. Dans la fonction main **Déclare un tableau d'entiers** et initialise-le avec des valeurs que vous aurez choisies, appelez la fonction moyenne pour afficher la moyenne des éléments du tableau.

Solution :

```
int somme(int T[], int taille){
    int s=0;
    for(int i=0; i < taille; i++){
        s+= T[i];
    }
    return s;
}

int moyenne(int T[], int taille){
    return somme(T, taille)/taille;
}

int main(){
    int A[4] = {10,20,5,10};
    printf("La moyenne est %d", moyenne(A, 4));
    return 0;
}
```

Exercice 3 :

Ecrire un programme en C qui réalise les étapes suivantes :

1. **Déclaration et initialisation de deux matrices** de taille donnée par l'utilisateur.

2. **Calcul de la somme de ces deux matrices**. La somme de deux matrices `A` et `B` de même taille `n x m` est une matrice `C` telle que chaque élément `C[i][j]` est égal à la somme des éléments correspondants des matrices `A` et `B`, c'est-à-dire :

   C[i][j]=A[i][j]+B[i][j]

3. **Affichage de la matrice résultante.**

Solution :

```
#include <stdio.h>
#define L 2
#define C 3

int main() {

   int lignes, colonnes;

   int A[L][C], B[L][C], S[L][C];

   printf("Entrez les éléments de la matrice A :\n");
   for (int i = 0; i < L; i++) {
      for (int j = 0; j < C; j++) {
         printf("Element [%d][%d]: ", i+1, j+1);
         scanf("%d", &A[i][j]);
      }
   }

   printf("Entrez les éléments de la matrice B :\n");
   for (int i = 0; i < L; i++) {
      for (int j = 0; j < C; j++) {
         printf("Element [%d][%d]: ", i+1, j+1);
         scanf("%d", &B[i][j]);
      }
   }

   // Calculer la somme
   for (int i = 0; i < L; i++) {
      for (int j = 0; j < C; j++) {
         S[i][j] = A[i][j] + B[i][j];
      }
   }


   printf("La matrice résultat est  :\n");
   for (int i = 0; i < L; i++) {
      for (int j = 0; j < C; j++) {
         printf("%d ", S[i][j]);
      }
```

```
        printf("\n");
    }
    return 0;
}
```

Exercice 4 :

Ecrire un programme en C qui :

1. Demande à l'utilisateur de saisir les dimensions des deux matrices A et B.
2. Demande à l'utilisateur de saisir les éléments de ces matrices.
3. Calcule le produit matriciel P=A×B si les dimensions des matrices sont compatibles.
4. Affiche la matrice résultante du produit matriciel.

Solution :

```
#include <stdio.h>
#define L 2
#define C 3
#define M 3

int main() {

    int lignes, colonnes;

    int A[L][M], B[M][C], S[L][C];

    // Saisir les éléments de la matrice A
    printf("Entrez les éléments de la matrice A :\n");
    for (int i = 0; i < L; i++) {
        for (int j = 0; j < M; j++) {
            printf("Element [%d][%d]: ", i + 1, j + 1);
            scanf("%d", &A[i][j]);
        }
    }

    // Saisir les éléments de la matrice B
    printf("Entrez les éléments de la matrice B :\n");
    for (int i = 0; i < M; i++) {
        for (int j = 0; j < C; j++) {
            printf("Element [%d][%d]: ", i + 1, j + 1);
            scanf("%d", &B[i][j]);
        }
    }

    // Calculer le produit matriciel
    for (int i = 0; i < L; i++) {
        for (int j = 0; j < C; j++) {
            C[i][j] = 0; // Initialiser l'élément C[i][j] à 0
            for (int k = 0; k < M; k++) {
```

```
                C[i][j] += A[i][k] * B[k][j];
            }
        }
    }

    // Afficher le produit matriciel
    printf("Le produit matriciel C = A * B est :\n");
    for (int i = 0; i < L; i++) {
        for (int j = 0; j < C; j++) {
            printf("%d ", C[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```