



Ecole Polytechnique Sousse

Département Informatique

SECTION : GÉNIE INFORMATIQUE & TÉLÉCOM ET RÉSEAUX

NIVEAU : 3<sup>ème</sup> ANNÉE

AU : 2020-2021

## Programmation C

### Travaux Pratiques N°3

#### Exercice 1

1. Écrire une fonction TRANSFERE qui permet à transférer les éléments de T1 dans T2 de telle sorte :
  - Les éléments du rang impair de T1 sont rangés dans T2 en ordre inverse en commençant par la fin de T2.
  - Les éléments du rang pair de T1 sont rangés dans T2 dans le même ordre.
2. Écrire une fonction SYMETRIQUE permettant de retourner :
  - 1 si T1 est symétrique
  - 0 si non
3. Écrire une fonction INVERSE qui permet de ranger les éléments du tableau T1 dans l'ordre inverse sans utiliser un tableau supplémentaire.
4. Écrire les fonctions nécessaires pour d'effectuer  $k$  ( $0 < k < n$ ) rotations entre les éléments de T.  
Une rotation entre les éléments de T s'effectue de la manière suivante : l'élément T[1] prend la valeur de T[0], T[2] prend la valeur de T[1], ..... et le premier élément prend la valeur du dernier.
5. Écrire une fonction INSERTION permettant d'insérer une valeur val (introduite à partir du clavier) à une position pos (introduite à partir du clavier)
6. Écrire les fonctions nécessaires pour construire un tableau V1 comportant les entiers présents dans T1 et dans T2. (Sans répétition).
7. Écrire une fonction main qui permet de tester les fonctions ci-dessus.

#### Exercice 2

Soit **T** un tableau contenant des entiers et de dimension maximale **50**.

1. Écrire les fonctions nécessaires permettant de :
  - saisir une valeur A quelconque,
  - chercher et afficher tous les éléments de T qui sont supérieurs à A et
  - chercher et afficher le plus petit des éléments qui sont supérieurs à A.

2. Écrire les fonctions nécessaires pour chercher et afficher l'élément qui apparaît le plus dans le tableau T de taille N, ainsi que son nombre d'occurrences.  
Si plusieurs éléments différents répondent à la condition, la fonction doit en fournir le premier.
3. Un plateau dans T est une suite d'éléments consécutifs et égaux. Écrire une fonction qui permet de :  
— chercher la longueur, l'indice début et fin du plus long plateau de T.

**Exemple :**

Pour N = 15

T	2	2	6	6	6	6	3	3	6	3	3	3	2	3	3
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

**Résultat :**

Le plus long plateau commence par 3 se termine par 6 et de longueur 4

4. Écrire une fonction main qui permet de tester les fonctions ci-dessus.

### Exercice 3

Soit T un tableau de N entiers strictement positifs et distincts.

Le but de cet exercice est de trier le tableau T dans un autre tableau B par ordre décroissant en utilisant le principe suivant :

**Principe :**

1. chercher le maximum de T
2. placer ce maximum dans B
3. remplacer le maximum par -1 dans T
4. refaire les étapes 1, 2 et 3 jusqu'à ce que le tableau T soit entièrement composé de -1.

### Exercice 4

Soit T un tableau de taille N (taille maximale 50) et contenant des entiers positifs et de 3 chiffres.

Un nombre est dit k-pp (presque premier) s'il s'écrit sous la forme d'un produit de  $k$  ( $2 \leq k \leq 5$ ) nombres premiers non nécessairement distincts.

**Exemples :**

- $243 = 3 * 73$  est un 2-pp car il est le produit de deux nombres premiers.
- $32 = 2 * 2 * 2 * 2 * 2$  est un nombre 5-pp car il est le produit de cinq nombres premiers.

Écrire un programme en C (en utilisant des fonctions) qui permet de chercher et afficher les k-pp (presque premiers) nombres du tableau T.

### Exercice 5

Soit M1 une matrice carrée contenant des valeurs numériques quelconques. Écrire les modules nécessaires pour vérifier et afficher si la matrice M1 est symétrique par rapport à la première diagonale.

**Exemple :**

M1 :

1	2	3	4
2	0	5	6
3	5	9	7
4	6	7	8

La matrice M1 est symétrique par rapport à la première diagonale.

## Exercice 6

Soit M2 une matrice de L lignes et C colonnes contenant des valeurs numériques quelconques. On appelle COL de la matrice l'élément qui est le plus petit de sa ligne et le plus grand de sa colonne.

**Exemple :**

M2

4	8	5
9	8	8
3	4	5
8	1	5

**Résultats :**

M2(1,1) = 8 est un point COL .

M2(1,2) = 8 est un point COL .

Écrire les fonctions que vous jugez nécessaires pour chercher et afficher les COL de M2.

## Exercice 7

Une matrice carrée, de dimension impaire, est dite magique si : la somme de chaque ligne = la somme de chaque colonne = la somme de chaque diagonale.

**Exemple :**

A :

4	9	2
3	5	7
8	1	6

Cette Matrice est magique, car toutes les sommes sont égales à 15.

Écrire les fonctions que vous jugez nécessaires pour vérifier si une matrice donnée est magique ou non.

# CORRECTION

## Atelier de Programmation 1

### Exercice 3

```
#include <stdio.h>
        /* Fonction SAISIE_TAILLE */
int  SAISIE_TAILLE()
{
    int x;
    do
    {
        printf ("Donner la taille du tableau : ");
        scanf("%d", &x);
    }while((x<=1) || (x>50));
    return x;
}
/* Fonction VERIF */
int verif (int t[],int i,int v)
{
    int j;
    for(j=0;j<i;j++)
        if(t[j]==v)
            return 1;
    return 0;
}
/* Remplir un tableau par des entiers strictement positifs et distincts*/
void REMPLIR (int t[],int n )
{
    int i ;
    for(i=0;i<n;i++)
        do
        {
            printf("%c1%cment[%d] : ",130,130,i);
            scanf("%d",&t[i]);
        }while(!(t[i]>0 && verif(t,i,t[i])==0));
}
/* Fonction AFFICHER */
void AFFICHER(int T[],int N)
{
    int i;
    for(i=0; i<N;i++)
        printf("%d\t", T[i]);
}
/*Fonction Maximum*/
int maximum(int t[],int n)
{
    int max=t[0],i,pos=0;
    for(i=0;i<n;i++)
        if(t[i]>max)
        {
```

```
        pos=i;
        max=t[i];
    }
    return pos;
}

/*Fonction Tri_Remplacer*/
void tri_replacer(int t[],int B[],int n)
{
    int i,pos ;
    for(i=0;i<n;i++)
    {
        pos=maximum(t,n);
        B[i]=t[pos];
        t[pos]=-1;
    }
}

void main()
{
    int T[50],B[50],n;
    n=SAISIE_TAILLE();
    REMPLIR(T,n);
    printf("**CONTENU DE T***\n");
    AFFICHER(T,n);
    tri_replacer(T,B,n);
    printf("\n**CONTENU DE B***\n");
    AFFICHER(B,n);
}
```