

ATELIER 0: REVISION

Exercice 1:

Ecrire une fonction qui calcule le cardinal (n, p)

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

Exercice 2:

Ecrire la fonction **MAJ_CH** qui convertit toutes les lettres d'une chaîne en majuscules.

Exercice 3:

Ecrire une fonction **Inverser** qui consiste à inverser un tableau d'entiers.

1 2 3 4 5 6 → 6 5 4 3 2 1

Exercice 4:

Soit un tableau **U1** composé de **N** entiers non nuls et un tableau **L** de même longueur dont les composantes sont **0** ou **1**.

L'objectif de l'exercice est de faire la compression de **U1** par **L** : le résultat est un tableau **U2** dont les composantes sont, dans l'ordre, celles de **U1** pour lesquelles la composante de **L** vaut **1**.

Pour le faire on vous demande d'écrire les fonctions suivantes:

- Ecrire une fonction **AfficherT (int T[], int n)** permettant d'afficher un tableau **T** de **n** entiers.
- Ecrire une fonction **SaisirU (int U1[], int n)** permettant de saisir dans le tableau **U1** **n** entiers.
- Ecrire une fonction **SaisirL (int L[], int n)** permettant de saisir dans le tableau **L** **n** entiers (on accepte uniquement des 0 ou bien 1).
- Ecrire une fonction **ChargerU2 (int U1[], int L[], int U2[], int n)** permettant de compresser **U1** par **L** : le résultat est un tableau **U2** dont les composantes sont, dans l'ordre, celles de **U1** pour lesquelles la composante de **L** vaut **1**.

Exemple: N=7

U1 :

1	2	3	4	5	6	7
---	---	---	---	---	---	---

L :

1	0	1	1	0	0	1
---	---	---	---	---	---	---

U2 :

1	3	4	7	0	0	0
---	---	---	---	---	---	---

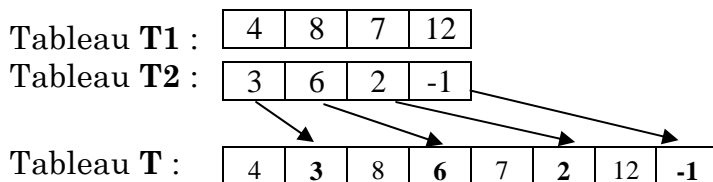
Ecrire un programme principal permettant de:

- Lire la dimension **N** de deux tableaux **U1** et **L** du type **entier** (dimension 10).
- Remplir **U1** et **L** par des valeurs entrées au clavier (il faut contrôler la saisie).
- Créer et afficher ensuite le tableau **U2** composé par les éléments de **U1** qui correspondre à 1 dans L et le reste par des 0.

Exercice 5:

Ecrire une fonction **void ChargerT (int T1[], int T2[], int T[], int n)** permettant de charger **T** par une alternance entre les valeurs de **T1** et celles de **T2**.

Exemple :



Ecrire un programme principal permettant de saisir deux tableaux T1 et T2 et de tester la fonction **ChargerT**.

Exercice 6:

Un polynôme $P(X) = \sum_{k=0}^n a_k X^k$ est représenté par un tableau de ses coefficients :

$$P = \begin{array}{|c|c|c|c|c|} \hline a_0 & a_1 & a_2 & \dots & a_n \\ \hline \end{array}$$

Exemple: le polynôme $P(x) = 2 + 3x^2 + 5x^3$ (taille =10) sera présenté par le tableau P suivant :

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

1. Ecrire la fonction **void afficherP(int P[], int n)** qui affiche le polynôme **P** de taille **n** sous la forme: $a_0 + a_1 X^1 + \dots + a_n X^n$
2. Ecrire la fonction **float valeurX (int P[], int n, float x)** qui calcule et retourne la valeur d'un réel **X** avec le polynôme.

Exemple:

Pour le polynôme $P(x) = 2+3x^2+5x^3$ et $X=2$, la fonction retourne $P(2)=54$

3. Ecrire la fonction **void SaisirP(int P[], int n)** permettant de saisir les coefficients d'un polynôme **P** de taille **n**.
4. Ecrire un programme principal permettant de tester ces fonctions.

Exercice 7:

Ecrire une fonction qui détermine les indices de la plus grande valeur dans **imax** et la plus petite valeur dans **imin** d'un tableau d'entiers.

```
void maxima (int tab[], int n, int *imax, int * imin);
```

Si le tableau contient plusieurs maxima ou minima, la fonction retiendra la position du premier maximum ou minimum rencontré.

Exercice 8:

Soit le programme C suivant:

```
#include <stdio.h>
int Increment1(int n)
{
    n=n+1;
    return n;
}
int Increment2(int *n)
{
    *n=*n+1;
    return *n;
}
int main()
{
    int n=2, n1, n2;
    n1=Increment1(n);
    n2=n1+n;
    printf("n2 =%d\n",n2);
    n1= Increment2(&n);
    n2=n1+n;
    printf("n2 =%d\n",n2);
    return 0;
}
```

Donner le résultat d'exécution de ce programme.

Exercice 9:

- 1- Écrire une fonction est **estVoyelle** qui prend en paramètre un caractère et retourne 1 si ce caractère est une voyelle et 0 si non.
- 2- Écrire une fonction **nbVoyelle**:

```
void nbVoyelles(char ch[], int *v, int *s)
```

À travers **v**, la fonction sauvegarde le nombre de voyelles dans la chaîne **ch**. Dans **c**, la fonction sauvegarde le nombre de consonnes.

Exercice 10:

Écrire une fonction qui permet de rechercher dans un tableau d'entiers **tab** de taille **n** les deux plus grandes valeurs :

```
void deux_max (int T[ ], int n, int *max1, int *max2);
```

Exemple: si T=[3, 1, 5, 5, 3, 6, 2, 6] alors **max1=6** et **max2 = 5**.