

Module : Structures de données

**TD4 (Les pointeurs)**

**Exercice 1 :**

```
main()
{
    int A = 1;
    int B = 2;
    int C = 3;
    int *P1, *P2;
    P1=&A;
    P2=&C;
    *P1=(*P2)++;
    P1=P2;
    P2=&B;
    *P1--=*P2;
    ++*P2;
    *P1*=*P2;
    A=++*P2**P1;
    P1=&A;
    *P2=*P1/=*P2;
    return 0;
}
```

Copiez le tableau suivant et complétez-le pour chaque instruction du programme ci-dessus.

	<u>A</u>	<u>B</u>	<u>C</u>	<u>P1</u>	<u>P2</u>
Init.	1	2	3	/	/
P1=&A	1	2	3	&A	/
P2=&C					
*P1=(*P2)++					
P1=P2					
P2=&B					
*P1--=*P2					
++*P2					
*P1*=*P2					
A=++*P2**P1					
P1=&A					
*P2=*P1/=*P2					

### **Exercice 2**

Ecrire un programme qui lit un entier X et un tableau A du type **int** au clavier et élimine toutes les occurrences de X dans A en tassant les éléments restants. Le programme utilisera les pointeurs P1 et P2 pour parcourir le tableau.

### **Exercice 3**

Ecrire un programme qui lit deux tableaux A et B et leurs dimensions N et M au clavier et qui ajoute les éléments de B à la fin de A. Utiliser le formalisme pointeur à chaque fois que cela est possible.

### **Exercice 4**

Ecrire un programme qui lit une chaîne de caractères CH et détermine la longueur de la chaîne à l'aide d'un pointeur P. Le programme n'utilisera pas de variables numériques.

### **Exercice 5**

Ecrire un programme qui lit un caractère C et une chaîne de caractères CH au clavier. Ensuite toutes les occurrences de C dans CH seront éliminées. Le reste des caractères dans CH sera tassé à l'aide d'un pointeur et de la fonction **strcpy**.