



TP N° 6 : Pointeurs & fonctions

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;
}
```

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

	A	B	C	P1	P2
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 une fonction **swap** qui a comme paramètres deux pointeurs vers des entiers et qui échange le contenu des deux entiers pointés. Tester cette fonction en écrivant un programme qui échange le contenu de deux entiers a et b en appelant cette fonction.

Exercice 3 :

Ecrire une fonction qui a comme paramètres un tableau d'entiers de taille quelconque, la taille du tableau et 2 pointeurs vers des entiers min et max. La fonction doit renvoyer dans les entiers pointés par min et max respectivement les plus petits et les plus grands entiers du tableau.

Exercice 4 :

Ecrire un programme en C qui définit sept fonctions pour effectuer les tâches suivantes:

1. void saisirDimensions(int * L, int* C) : Saisie des dimensions d'une matrice ;
2. int** allouerEspace(int L, int C) : Réservation de l'espace mémoire pour une matrice de L lignes et de C colonnes;
3. int** libererEspace(int L, int C) : Libération de l'espace mémoire occupé;
4. void saisirMatrice(int** M, int L, int C) : Saisie des éléments d'une matrice ;
5. void afficherMatrice(int** M, int L, int C) : Affichage des éléments d'une matrice ;
6. int** produitMatrice(int** A, int** B, int M, int N, int L) : Calcul du produit de deux matrices ;
7. main() : fonction principale pour tester le programme.

Exercice 5 :

Écrivez une fonction qui compte le nombre d'occurrences d'un caractère c dans une chaîne s. La fonction pourra être **récursive**. Tester cette fonction.

Prototype : int compte (char c, char * s);

