

## **Sujets : LES POINTEURS**

**Objectif : Savoir adresser indirectement la mémoire de l'ordinateur en plus de l'adressage direct.**

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;
- Lire le chapitre 9 du support principal (n°3) du cours, de la page **116** à la page **138** ;
- Lire le second support (n°2) du cours, de la page **51** à la page **56** ;
- Lire et faire des recherches complémentaires pour être capable d'expliquer et pratiquer les concepts du contenu ci-dessous.

Contenu :

- L'importance des pointeurs en C
- Adressage de variables (direct et indirect)
- Les pointeurs
  - Les opérateurs de base
  - Les opérations élémentaires sur pointeurs
- Pointeurs et tableaux
  - Adressage des composantes d'un tableau
  - Arithmétique des pointeurs
  - Pointeurs et chaînes de caractères
    - Pointeurs sur char et chaînes de caractères constantes
  - Pointeurs et tableaux à deux dimensions
- Tableaux de pointeurs
- Exercices d'application

Activités :

- Pratiquer en considérant tous les exemples et exercices des supports numéros 2 et 3 ;
- Faire le(s) devoir(s) de maison
- Partager les difficultés en groupe restreint avec des camarades proches ou plus éclairés ;
- Signaler les difficultés non réglées aux délégués pour une redirection à l'enseignant dans de meilleurs délais.

**Exercices pratiques****N°1**

Exercice 9.1 du support n°3.

Soit le programme suivant :

```

int 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	/
1	P2=&C					
2	*P1=(*P2)++					
3	P1=P2					
4	P2=&B					
5	*P1--=*P2					
6	++*P2					
7	*P1*=*P2					
8	A=++*P2**P1					
9	P1=&A					
10	*P2=*P1/=*P2					

N°2

Exercice 9.2 du support n°3.

N°3

Exercice 9.4 du support n°3.

N°4

Exercice 9.5 du support n°3.

N°5

Exercice 9.6 du support n°3.

N°6

Exercice 9.8 du support n°3.

N°7

Exercice 9.10 du support n°3.

N°8

Exercice 9.11 du support n°3.

N°9

Exercice 9.12 du support n°3.

N°10

Exercice 9.13 du support n°3.

N°11

Exercice 9.14 du support n°3.

N°12

Exercice 9.16 du support n°3.

N°13

Exercice 9.17 du support n°3.

N°14

Ecrire un programme qui effectue les traitements suivants. Après chaque traitement et pour chaque pointeur ou valeur, afficher le contenu du pointeur, son adresse, la valeur pointée et l'adresse de la valeur pointée. Le format d'affichage des pointeurs est %p.

- déclarer un pointeur  $p$  sans l'initialiser et un pointeur  $q$  initialisé à `NULL`;
- le pointeur  $p$  pointe sur une valeur entière  $v$  initialisée à 10 ;
- à l'aide du pointeur, modifier cette valeur entière  $v$  "à partir du clavier" ;
- utiliser seulement le pointeur  $p$  pour initialiser une valeur entière  $w$  à la valeur de  $v$ ;
- utiliser seulement un pointeur  $r$  qui pointe vers  $p$  pour modifier la valeur entière  $v$  "à partir du clavier".

NB :

- 1) Les projets doivent être ramassés et envoyés à l'enseignant au plus tard le lundi avant midi ;
- 2) Chaque apprenant doit respecter la procédure d'envoi ;
- 3) Les délégués sont chargés de la récupération à temps et de l'envoi à temps des fichiers compressés en un (1) ou deux (2) lots.

Pensées :

1. Henri Ford a dit un jour : *"L'échec n'est qu'une excellente occasion de recommencer de manière plus intelligente."*
2. Rick WARREN a dit : *« Si vous pratiquez régulièrement quelque chose, vous finirez par devenir très bon. L'entraînement mène à l'excellence. »* « Practices make perfect »
3. *« Connaissez-vous un homme habile dans ce qu'il fait ? Il ne restera pas au service de gens obscurs, mais il entrera au service des rois. »* Proverbes 22:29 BDS

L'adressage indirect :

- Déclaration du pointeur : `int *P ;` //réservation d'une zone mémoire pour contenir une adresse
- Initialisation du pointeur : `P = &X ;` // en considération la déclaration de la variable X ;
- Utilisation du pointeur : `*P = 10 ;` // on utilise l'adresse de la zone mémoire nommée X et dont l'adresse est contenu dans P ;