

TP 09 Pointeurs

À retenir

```
int age;  
int* pointeurAge;  
pointeurAge = &age;
```

- sur une variable, comme la variable `age`:
 - `age` signifie : « Je veux la valeur de la variable `age` »,
 - `&age` signifie : « Je veux l'adresse à laquelle se trouve la variable `age` » ;
- sur un pointeur, comme `pointeurSurAge`:
 - `pointeurSurAge` signifie : « Je veux la valeur de `pointeurSurAge` » (cette valeur étant une adresse),
 - `*pointeurSurAge` signifie : « Je veux la valeur de la variable qui se trouve à l'adresse contenue dans `pointeurSurAge` ».

Exercice 1

- 1) Essayez de deviner le résultat de l'exécution du programme ci-dessous :

```
1  #include <stdio.h>  
2  int main() {  
3      int i = 20;  
4      int* p;  
5      p = &i;  
6      *p = 10;  
7      printf("%d", *p );  
8      return 0;  
9  }  
10
```

- 2) Essayez de deviner le résultat de l'exécution du programme ci-dessous :

```
1  #include <stdio.h>  
2  int main() {  
3      int i = 20;  
4      int* p;  
5      p = &i;  
6      *p = 10;  
7      printf("%d",p);  
8      return(0);  
9  }  
10
```

Exercice 2

Essayez de deviner le résultat de l'exécution du programme ci-dessous :

```

1 void main(void)
2 {
3     short int a,b ;
4     short int* ptr ;
5     ptr = &a ; /* ptr pointe sur a */
6     a = 10 ;
7     b = *ptr ; /* met dans b le contenu de l'adresse ptr (10) */
8     *ptr = b*2 ; /* met la valeur b*2 à l'adresse pointée par ptr (attention aux 2 sens différents de *) */
9     printf(" adresse de a : %p (%u en décimal)", &a, &a) ;
10    printf("\n adresse de b : %p (%u en décimal)", &b, &b) ;
11    printf("\n adresse de ptr : %p (%u en décimal)", &ptr, &ptr) ;
12    printf("\n\n contenu de ptr : %p", ptr) ;
13    printf("\n\n contenu de a : %hd \n contenu de b : %hd", a, b) ;
14    printf("\n contenu de la variable pointée par ptr : %hd", *ptr) ;
15    getch() ; /* attend la frappe d'une touche */
16 }

```

Exercice 3: opérations élémentaires sur pointeurs

```

1  #include <stdio.h>
2  main()
3  {
4      int A = 1;
5      int B = 2;
6      int C = 3;
7      int *P1, *P2;
8
9      P1=&A;
10     P2=&C;
11     *P1=(*P2)++;
12     P1=P2;
13     P2=&B;
14     *P1-=*P2;
15     ++*P2;
16     *P1*=*P2;
17     A=++*P2**P1;
18     P1=&A;
19     *P2=*P1/=*P2;
20     return 0;
21 }
22

```

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					

<code>++*P2</code>					
<code>*P1*==*P2</code>					
<code>A=++*P2**P1</code>					
<code>P1=&A</code>					
<code>*P2=*P1/=*P2</code>					

Exercice 4

Soit P un pointeur qui 'pointe' sur un tableau A:

```
int A[] = {12, 23, 34, 45, 56, 67, 78, 89, 90};
int *P;
P = A;
```

Quelles valeurs ou adresses fournissent ces expressions:

- a) `*P+2`
- b) `*(P+2)`
- c) `&P+1`
- d) `&A[4]-3`
- e) `A+3`
- f) `&A[7]-P`
- g) `P+(*P-10)`
- h) `*(P+*(P+8)-A[7])`

Exercice 5

Ecrire un programme C qui utilise la notion de pointeur pour lire deux entiers et calculer leur somme.

Exercice 6

Ecrire un programme C qui utilise la notion de pointeur pour permuter le contenu de deux variables de type *char*.

Exercice 7

Ecrire un programme C qui remplit un tableau d'entiers et calcule la somme de ses éléments en utilisant un pointeur pour son parcours.

Exercice 8

Ecrire un programme qui lit deux tableaux d'entiers A et B et leurs dimensions N et M au clavier et qui ajoute les éléments de B à la fin de A. Utiliser deux pointeurs PA et PB pour le transfert et pour afficher le tableau résultant A.

Exercice 9

Ecrire un programme C qui lit une chaîne de caractères et affiche cette chaîne à partir de la première occurrence d'un caractère entré par l'utilisateur. En utilisant pour ceci la fonction *strchr* et un pointeur pour le parcours de la chaîne.

Exercice 10

Ecrire un programme C qui définit une structure permettant de stocker le nom, le prénom et l'âge d'une personne. Lit ensuite ces informations pour deux personnes et affiche le nom complet de la moins âgée d'entre elles en utilisant une seule fonction *printf* pour l'affichage du résultat.

Exercice 11

Ecrire un programme C qui vérifie si une chaîne de caractères est palindrome en utilisant deux pointeurs pour son parcours.

une chaîne de caractères est dite palindrome si elle se lit de gauche à droite et de droite à gauche de la même façon. Par exemple, otto est palindrome. (23532)

Exercice 12

On souhaite écrire une fonction qui permet de résoudre une équation du premier degré : $ax+b=0$. Voici le prototype de la fonction : `int resoudre1(int a, int b, float *x);`

la fonction retourne le nombre de solution trouvé (0: pas de solution, 1: une solution, -1: tout x est solution). Dans le cas où l'équation a une solution, la fonction retourne la solution dans x.

Exercice 13

1) Ecrire un programme qui demande les éléments d'un tableau T, et une valeur A puis élimine toutes les occurrences de A dans ce tableau. Utiliser deux pointeurs P1 et P2.

soit le tableau T suivant

La valeur de la variable A est: A=4

5	4	15	-2	88	4
---	---	----	----	----	---

Donc le résultat sera:

5	15	-2	88
---	----	----	----

2) Ecrire un programme qui inverse les éléments d'un tableau T. Utiliser deux pointeurs P1 et P2.

Premier tableau:

5	15	-2	88
---	----	----	----

Le résultat est :

88	-2	15	5
----	----	----	---