

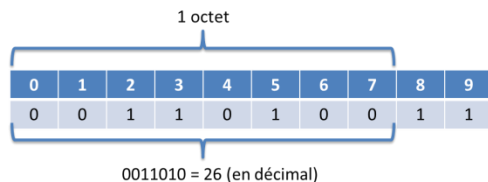
Les pointeurs en C

Ecrit par Sylyon dans le cadre d'une activité sur le site openclassrooms.com

Bonjours, pour voir ce qu'est un pointeur on doit expliquer comment fonctionne la mémoire.

La mémoire

La mémoire d'un PC peut être regroupée en deux grandes catégories, la mémoire vive et la mémoire morte. On peut résumer ça à deux entités dans le PC : la RAM et le disque dur. Le disque dur contient des fichiers et la RAM des « informations ». Pour ce résumé seul la RAM nous intéresse (la mémoire vive). Tu peux voir la RAM comme un grand tableau contenant des 0 et des 1.



Un ensemble de 8 valeur forme un octet, la RAM est donc constituer d'un ensemble d'octet. On peut écrire n'importe quel type de variable C dans la RAM. Elles prendront plus ou moins d'octet en fonction de la variable (un *char* prend 1 octet, un *int* en prend 4 octets, ...).

Finalement les variables que tu crées en C sont sauvegardées dans un grand tableau et elles prennent un nombre fixes de case dans ce tableau. Ce nombre est définit par le type de la variable (cad tous les *int* prennent 4 octets).

Les pointeurs

Les pointeurs en C sont des variables C. Voilà c'est fini.

Bon je vais expliquer un peu plus. Un pointeur est une variable dont la valeur est une adresse mémoire. Un exemple vaut mille mots :

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main()
5  {
6      int myInt=42;
7      int* myPoint= &myInt;
8      // &myInt = myPoint
9      // myInt = *myPoint
10     // *myInt = ???
11     // &myPoint = ???
12     // myPoint = &myInt
13     // *myPoint = myInt
14     return 0;
15 }
16
```

Dans ce code *myInt* est un *int* et il a comme valeur 42 et *myPoint* est une variable dont la valeur est l'adresse mémoire à laquelle *myInt* est enregistré. Pour information :

- *&* avant une variable renvoie l'adresse en mémoire de cette variable.
- *** avant un variable renvoi la valeur de ce qui est enregistré à l'adresse indiqué. (voir le code ci-contre pour exemple).

Finalement un pointeur est une variable qui contient une adresse mémoire. Ce peut être l'adresse de n'importe quoi (une variable, un autre pointeur, une fonction,...).