Les pointeurs

Un pointeur est une variable qui contient l’adresse en mémoire d’une variable.

Pour retrouver la variable, il faut connaitre l’adresse de l’octet ou elle est stockée.

Si un pointeur Pt contient l’adresse d’une variable nombre (nbr), on dit que Pt pointe sur nbr.

Il est limité à un type de donnée.

Pour utiliser une variable, il existe deux manières :

* Soit comme habituellement une réservation en mémoire 🡪 int nbr = 100 ;
* Soit par adresse 🡪
  + - Int nbr=100 ;
    - Int \*PointeurNbr ;
    - PointeurNbr =&nbr ; //PointeurNbr pointe vers nbr

Déclaration

Un pointeur doit être déclaré comme toute autre variable classique :

Type\_du\_pointeur \* nom\_du\_pointeur

Exemples :

* int \*PointeurNbr ;
* float \*PointeurPrix ;
* char \*PointeurChoix ;

Initialisation

Pour initialiser un pointeur, il faut créer un lien entre le pointeur et la variable !

Exemple :

Int \*PointeurNbr ; //Déclaration

PointeurNbr = &Nbr ; //Lien entre le pointeur et la variable.

Exemple d’utilisation des pointeurs avec un programme qui demande à l’utilisateur d’entrer 10 nombres. Et qui en ressort un tableau à l’aide d’un pointeur.

#include <stdio.h>

main()

{

/\* Déclarations \*/

int TabEntier[10]; /\* tableau \*/

int \*PtTab; /\* pointeur dans A \*/

/\* Saisie des données \*/

for (PtTab=TabEntier; PtTab<TabEntier+10; PtTab++) {

printf("Introduire votre nombre : \n");

scanf("%d", PtTab);

}

/\* Affichage du tableau \*/

printf("\n\tTableau TabEntier :\n");

for (PtTab=TabEntier; PtTab<TabEntier+10; PtTab++)

printf("%d ", \*PtTab);

return 0;

}

**Allocation dynamique.**

Il existe un moyen de réserver dynamiquement une zone mémoire pour un pointeur, cette fonction s’appelel malloc.

Elle s’utilise avec un opérateur sizeof qui donne la grandeur de la zone…

Voici un exemple :

Int \*PointeurA ;

PointeurA = &A ;

PointeurA = (int\*)malloc(sizeof(int)) ; // Crée le lien entre PointeurA et l’espace mémoire

Pour libérer cette esapce mémoire 🡪 free(PointeurA).

Autre exemple (avec un memset) 🡪 Initialisatin d’une zone à une valeur:

Int \*Pointeur ;

Pointeur = (int\*)malloc(5\*sizeof(int)) ; //Pointeur pointe sur une zone de 5 entiers

memset(Pointeur,0,5\*sizeof(int)) ; // Initialisation de 5 entiers à 0 dans la zone du pointeur

Pointeur[0] = 100 ;

Pointeur[1] = 200 ;

Pointeur

100

200

0

Les 3 valeurs à 0

0

0

memset 🡪 fonction

Pointeur 🡪 Nom\_Du\_Pointeur

0 🡪 Valeur de la variable

5 🡪 Nombre « D’adresse » à initialiser à 0

\* 🡪 Siginifie qu’on utilise un pointeur

sizeof(int))  🡪 Zone dynamique d’entiers

Pour libérer cette allocation il y a une fonction FREE qui est à utiliser !

On l’utilise tout simplement en mettant entre parenthèses le nom du pointeur qui est utilisé.

* Free(NomDuPointeur) ;