Int a =0 ; int \*pa = null;

/\* Dangereux de ne pas initialise les pointeurs

Long b =0 ; int \*pb = null ;

Char c = 0 ; int \*pc = null;

pa (@adresse d’un entier)

|  |
| --- |
|  |

4octet

pb (@adresse d’un long)

|  |
| --- |
| 002000 |

4octet pb = &b ; adresse : 0105421

pc (@adresse d’un char)

|  |
| --- |
|  |

4octet

A (int)

|  |
| --- |
| 4octet |

B (long)

|  |
| --- |
| 10 |

002000 : adresse b =10 ;

C (char)

|  |
| --- |
| 1octet |

Printf( « b =%ld », b) ; 🡺 10

Printf (« @b =%ld », &b) ; 🡺 002000

Printf(« pb =%ld », pb) ; 🡺 002000 (donne le contenue de mon pointeur )

Printf(« @pb =%ld »,&pb) ; 🡺 0105421 (donne l’adresse de mon pointeur)

Printf( « lélément point par pb contient %ld », \*pb) 🡺10 (donne le contenue de l’adresse où je pointe )

|  |
| --- |
| 1 |
| 2 |
| 3 |
| 4 |
| 5 |
| 6 |

0002000 = adresse complet du tableau tab[1]

0002004 ⬄ 0002000 + sizeof(int) tab[2]

0002008

0002012

0002016

Printf (« tab =%ld », tab) ; ⬄ 0002000

‘’ (‘’tab[0] = %ld ‘’, tab[0]) ; 🡺1

‘’ (‘’@tab[0] =%ld’’,&tab[0]) ; 🡺 0002000

‘’ (‘’@tab[0] =%ld’’, tab) ; 🡺 0002000

‘’ ( ‘’ %ld’’, \*Tab) ; 🡺 1

‘’ (‘’%ld’’, tab +1) ; 🡺 0002004

‘’ (‘’%ld’’, \*(tab +1)) ; 🡺 2

‘’ (‘’%ld’’, \*(1+tab)) ; 🡺 2

Include <stdio.h>

Int main (){

Double t[10] ;

Int i  ;

For( i =0 ; i <10 ; i++){

Printf(‘’Entrez la valeur n°%d’’ , i+1) ;

Scanf(‘’%lf’’, tab+i ) ; 🡺 va a l’adresse et modifie la valeur par celle saisie au clavier

Ou scanf (‘’%lf’’, &tab[i]) ;

* &tab[i] ⬄ &\*(tab+i) ⬄ tab+i

}

}

Int main () {

Int T[5] ={} ;

Int \*p=null ;

Int i ;

p =T ;

for (i=0 ; i <=4 ; i++){

scanf(‘’%ld’’, p+i) ;

}

For(i=0 ; i <=4 ; i++){

Printf(‘’ %ld’’, \*(p+i)) ;

}

}

P 🡺 00500 = affiche l’adresse du tab

&p 🡺 700500 = affiche l’adresse de p

\*p 🡺 0 affiche la première valeur du tableau

P+1 🡺 00500 + sizeof(int) => 00504

#define taille 100 ;

Int tab[100] ;

Int T[taille] ;

Int Mat[4][3] = {{1,2,3},{4,5,6},{7,8,9},{10,11,12}} ;

Int Mat1[3][2][2] = {{{1,2},{3,4}}, {{5,6},{7,8}}, {{9,10},{11,12}}} :

Int i,j ;

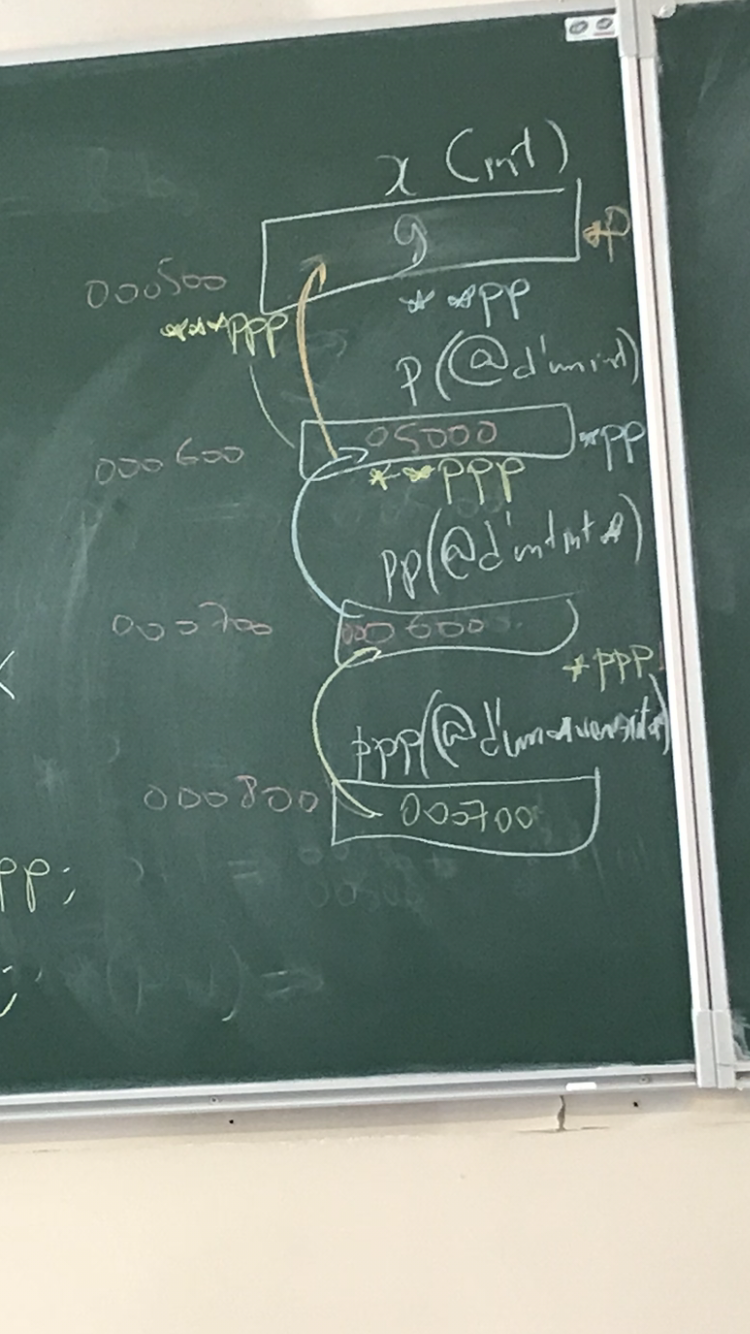
For (i=0 ; i <3 ; i++){

For (j =0 ; j<3, j++){

Scanf(‘’%d%, &Mat[i][j]) ;

}

}

Int x

Int \*p ;

Int \*\*pp ;

Int \*\*\*ppp ;

x = 5 ;

p = &x ;

\*p = 3 ;

pp = &p ;

\*\*pp = 3 ;

\*\*pp = 10 ;

ppp = &pp ;

\*\*\*ppp -- ;



