

Initialisation tableau

```
#define N 4
```

double x[N]; ← non initialisé.

$x[0] \dots [N-1]$: valeurs
sont déclarées

① Init du tableau à Ø

double x[N] = { Ø., Ø., Ø., Ø. };

double x[N] = { Ø. };

$x[0] \dots x[N-1]$ sont à Ø.

② Init du tableau avec des valeurs spécifiques

double x[N] = { -3.14, +2.7, 92.Ø, -1. };

③ Init avec qq valeurs

double x[N] = { x[0], x[1], x[2] };

$x[2]$ et $x[3]$ ne sont pas initialisés,

TD 2020/11/23

- Ⓐ 3 tableaux de 5 cases chacun
 t_1 , t_2 et t_3 .
chacun case est un char

$$t_1[\emptyset] = 1 \dots t_1[1] = 2 \dots t_1[4] = 5$$

$$t_2[\emptyset] = 6 \quad \dots \quad t_2[4] = 10$$

$$t_3[\emptyset] = 11 \quad \dots \quad t_3[4] = 15.$$

- Ⓑ Affichez les adresses et les valeurs
de $t_1[k]$ $k : [\emptyset \dots N-1]$
de $t_2[k]$ "
de $t_3[k]$ "

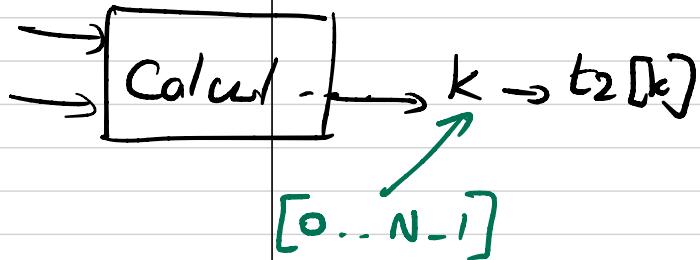
→ image "Vionelli" du contenu
des 3 tableaux
et de l'emplacement en mémoire
des cellules de ces tableaux.

13^h53

$t_1[\phi] \dots t_1[9]$

$t_2[\phi] \dots t_2[4]$

$t_3[\phi] \dots t_3[1]$



(A)

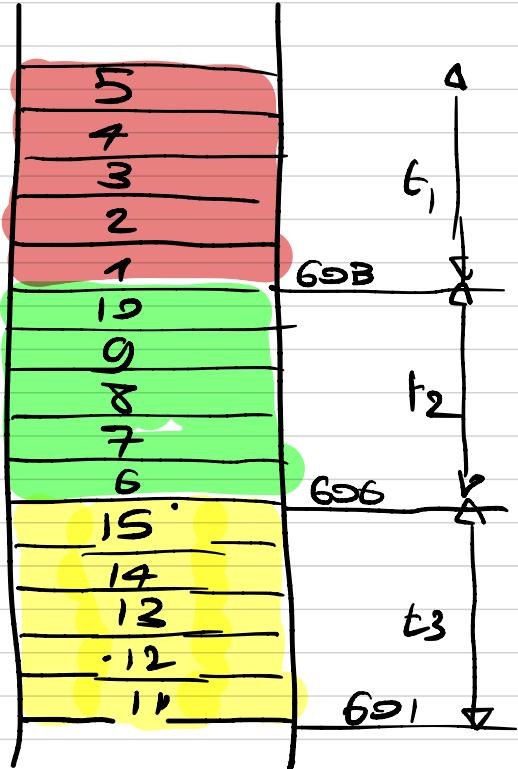
$k = \phi$

modifier $t_2[k-1]$ 42

(B)

$k = N-1$

modifier $t_2[k+2]$ 99



$t_1[0]$	0x7ffeeb2b160b	1
$t_1[1]$	0x7ffeeb2b160c	2
$t_1[2]$	0x7ffeeb2b160d	3
$t_1[3]$	0x7ffeeb2b160e	4
$t_1[4]$	0x7ffeeb2b160f	5
$t_2[0]$	0x7ffeeb2b1606	6
$t_2[1]$	0x7ffeeb2b1607	7
$t_2[2]$	0x7ffeeb2b1608	8
$t_2[3]$	0x7ffeeb2b1609	9
$t_2[4]$	0x7ffeeb2b160a	10
$t_3[0]$	0x7ffeeb2b1601	11
$t_3[1]$	0x7ffeeb2b1602	12
$t_3[2]$	0x7ffeeb2b1603	13
$t_3[3]$	0x7ffeeb2b1604	14
$t_3[4]$	0x7ffeeb2b1605	15

↳ renvoie $t_1/t_2/t_3$
adresse et contenu

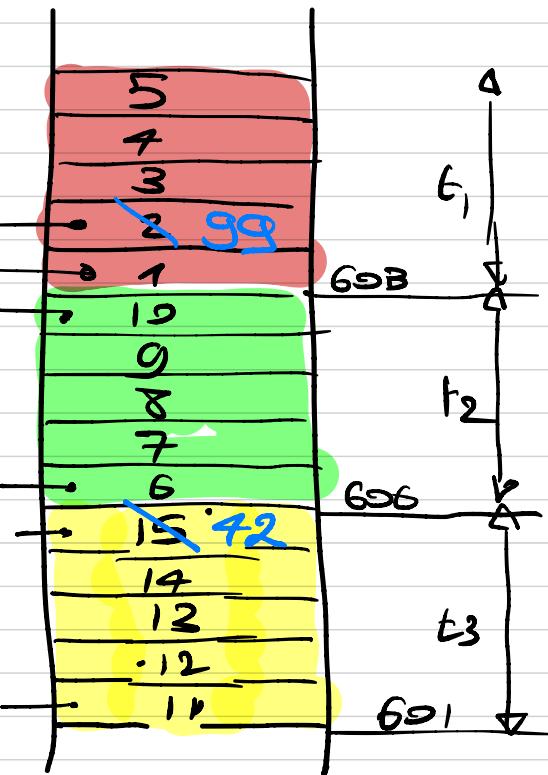
= 14^n ID

(A)

 $k = \phi$ modifier $t_2[k-1] = 72$

(B)

 $k = N - 1$ modifier $t_2[k+2] = 99$
 $k = N - 1$
 $k + 2$
 $t_2[6] = 99$
 $k = \phi$
 $t_2[k-1] = 72$
 $t_2[-1] = 72$

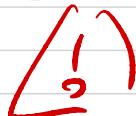
 $t_1[1]$
 $t_1[\phi]$
 $t_2[1]$
 $t_2[1]$
 $t_2[\phi]$
 $t_3[9]$
 $t_3[\phi]$ 

$t_1[0]$	<code>0x7ffee21cf60b</code>	1
$t_1[1]$	<code>0x7ffee21cf60c</code>	99
$t_1[2]$	<code>0x7ffee21cf60d</code>	3
$t_1[3]$	<code>0x7ffee21cf60e</code>	4
$t_1[4]$	<code>0x7ffee21cf60f</code>	5
$t_2[0]$	<code>0x7ffee21cf606</code>	6
$t_2[1]$	<code>0x7ffee21cf607</code>	7
$t_2[2]$	<code>0x7ffee21cf608</code>	8
$t_2[3]$	<code>0x7ffee21cf609</code>	9
$t_2[4]$	<code>0x7ffee21cf60a</code>	10
$t_3[0]$	<code>0x7ffee21cf601</code>	11
$t_3[1]$	<code>0x7ffee21cf602</code>	12
$t_3[2]$	<code>0x7ffee21cf603</code>	13
$t_3[3]$	<code>0x7ffee21cf604</code>	14
$t_3[4]$	<code>0x7ffee21cf605</code>	42

Pas de limitation pour la valeur de l'index à un tableau de taille N

$< \phi$

$\geq N$



JARDINAGE



Tableau à 1D

#define N 4

uint8_t $\times [N] = \{ 10, 15, 3, 8 \}$

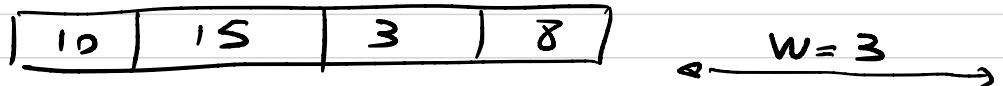
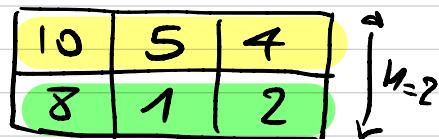


Tableau à 2D

① le nombre de lignes : 2

② le nombre de colonnes : 3

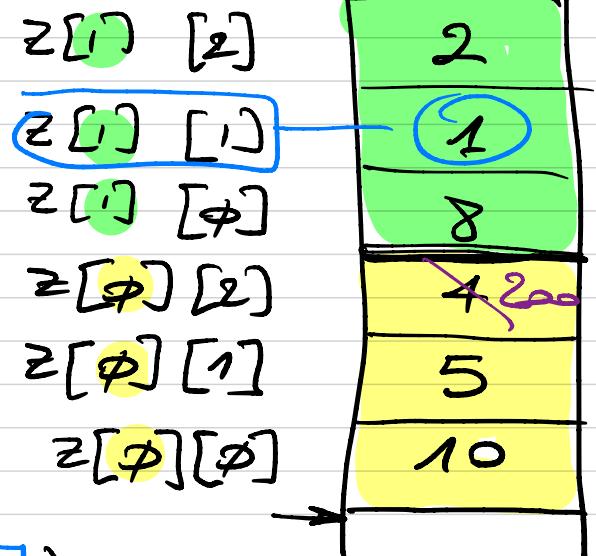


#define WIDTH 3

#define HEIGHT 2

uint8_t $\equiv [HEIGHT][WIDTH];$
 $\uparrow \quad \uparrow$
lignes colonnes

l'organisation du
tableau en
mémoire n'est pas
par lignes successives.



printf("%d", $z[1][1]$); // =

$z[2][2] = 200;$ ligne colonne,

Init + tableau 2D

w = 3		
10	5	4
8	1	2

$h=2$

$$\text{vint8-t} \approx [\text{HEIGHT}][\text{WIDTH}] =$$

f

{ 10, 5, 4 } , 11 lign 1
{ 8, 1, 2 } , 11 lign 2
}; —

TD 2020/123b

①

Tableau 2D avec des vint8-t
 $\leftarrow 640 \xrightarrow{\text{colonnes}}$

1	2	
3	4	

480 lignes

②

Initialisation du tableau: + rectangles aux 1, 2, 3 et 4

③

Couper le tableau moyenne du tableau

17h42..

```
#define HEIGHT 480
#define WIDTH 640
```

init [1] for ligne : [0 → HEIGHT/2] {
 col. [0 → WIDTH/2]

[2] for ligne : [0 → HEIGHT/2] [
 col. [WIDTH/2 → WIDTH]