



JAVA

Tableau à plusieurs indices

Tableau à plusieurs indices

Premier exemple. Ces trois déclarations sont équivalentes pour un tableau à 2 dimensions :

```
int t [] [];  
int [] t [];  
int [] [] t;
```

```
int t [] [] = { new int [3], new int [2] };
```

`t` comporte deux éléments de sorte que le premier soit un tableau de 3 entiers et que le deuxième soit un tableau de 2 entiers. On aboutit à la situation suivante (les éléments des tableaux d'entiers sont comme d'habitude initialisés à 0) :

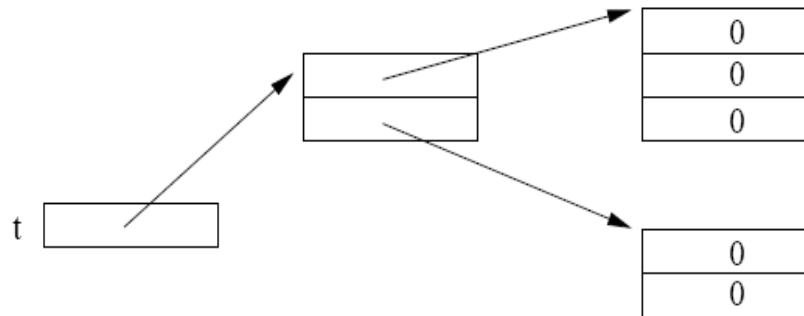


Tableau à plusieurs indices

On constate alors que :

- la notation `t[0]` désigne la référence au premier tableau de 3 entiers,
- la notation `t[0][1]` désigne le deuxième élément de ce tableau (pour rappel les indices commencent à 0),
- la notation `t[1]` désigne la référence au deuxième tableau de 2 entiers,
- la notation `t[0][i-1]` désigne le *i*ème élément de ce tableau pour *i* compris entre 1 et 2,
- l'expression `t.length` vaut 2,
- l'expression `t[0].length` vaut 3,
- l'expression `t[1].length` vaut 2.

Tableau à plusieurs indices

Second exemple.

```
int t [] [];  
t = new int [2] [];  
int [] t1 = new int[3];  
int [] t2 = new int[2];  
t[0] = t1;  
t[1] = t2;
```

La situation peut s'illustrer comme ceci :

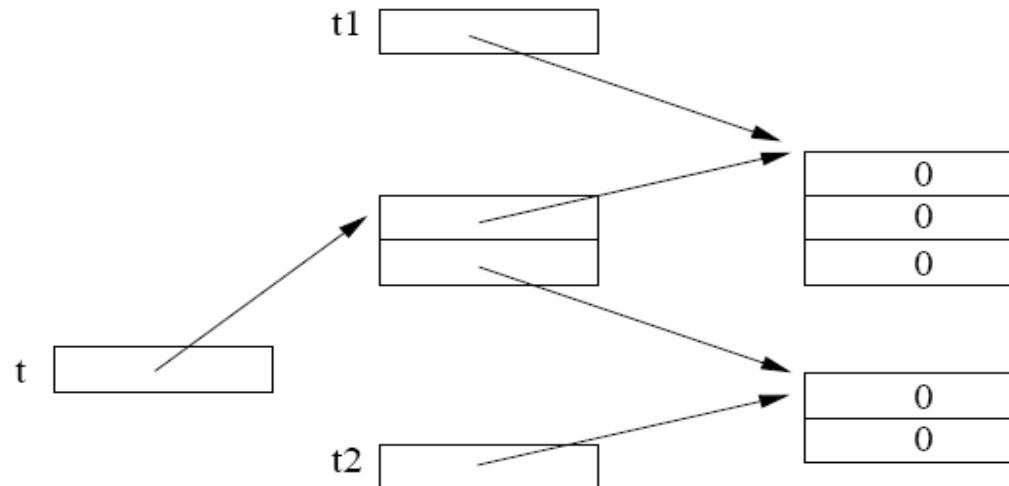


Tableau à plusieurs indices

```
class Util
{
static void raz(int t[] []) {
int i,j;
for(i=0;i<t.length;i++)
for(j=0;j<t[i].length;j++) t[i][j]=0; }
static void affiche(int t[] [])
{ int i,j;
for(i=0;i<t.length;i++) {
System.out.println("ligne de rang "+i+" = ");
for(j=0;j<t[i].length;j++) System.out.println(t[i][j] + " ");
System.out.println();
}
} }
```

Tester

