

Résumé: Le tableau dynamique

- Un tableau dynamique peut avoir une taille [] à chaque exécution
- L'espace mémoire est réservé au moment de l' [] et non à la [] comme un tableau automatique
- Le tableau est alloué sur le tas et non sur la []
- L'utilisation d'un tableau dynamique se fait en 5 étapes:
 1. Déclaration d'un [] pour contenir l' [] (le point de départ) du tableau
 2. Conserver la taille du tableau dans une variable
 3. Réservation de l'espace mémoire avec l'opérateur []
 4. Utiliser le tableau comme s'il était un tableau automatique
 5. Lorsque le tableau n'est plus requis, libérer l'espace en mémoire avec l'opérateur []

- Différents types de tableau dynamique

Les tableaux dynamiques

- 

```
int* Tab = new int [10]; ... delete [] Tab;
```

- De type utilisateurs

- De structures

```
Point* Flocs = new Point [10]; ... delete [] Flocs;
```

- 

```
CTasse* Tasses = new CTasse[10]; ... delete [] Tasses;
```

- De pointeurs!



```
Tasses = new  [10]; ... delete [] Tasses;
```

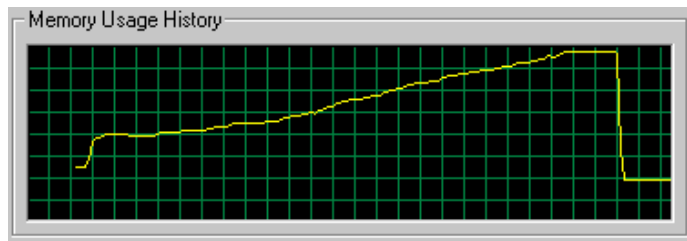
➤ C'est toujours la même syntaxe:



Si je fais l'exemple de code suivant :

```
void Test()  
{  
    int* Tab = new int[1000000000];  
    // ...  
}
```

Je vais obtenir ce résultat :



Nommez ce phénomène et expliquez la cause dans l'exemple de code.