

Support de cours

Cours:

## Initiation à la programmation (en C++)

Vidéo:

### Exemples simples (vector) (partie 1)

Concepts (extraits des sous-titres générés automatiquement) :

**Tableaux dynamiques. Éléments du tableau. Boucle c. Variable i. Tableau dynamique. Premier élément. Structures de contrôle. Boucle for. Nom de variable élément. Contenu du tableau. Nom d'un élément. Séquence vidéo. Doubles tous nuls. Premier élément du tableau. Nom de la variable.**



[vers la recherche de séquences vidéo](#)  
(dans Initiation à la programmation (en C++).)



[vers la vidéo](#)

Center for Digital Education. Plus de matériel de soutien pédagogique ici :

<https://www.epfl.ch/education/educational-initiatives/cede/educational-technologies-gallery/boocs-en/>

# Tableaux : exemples simples (vector)

(Partie 1)

## Initiation à la programmation (C++)

Vincent Lepetit, Jean-Cédric Chappelier et Jamila Sam

...

notes

résumé

0m 0s



## Quelques exemples de manipulation de tableaux

Soit un tableau déclaré par:

```
vector<double> tab(10);
```

Affichage du tableau :

- ▶ si l'on n'a pas besoin d'explicitement les indices :

```
cout << "Le tableau contient : ";
for(auto element : tab) {
    cout << element << " ";
}
cout << endl;
```

- ▶ si l'on veut expliciter les indices :

```
for(size_t i(0); i < tab.size(); ++i) {
    cout << "L'élément " << i << " vaut " << tab[i] << endl;
}
```

Voyons dans cette séquence vidéo quelques cas d'application typique simple des concepts présentés jusqu'ici sur les tableaux dynamiques. Supposons, par exemple, que l'on ait un tableau dynamique qui contient des doubles que l'on appelle « tab » et qu'on initialise au départ avec 10 doubles tous nuls comment faire pour afficher les éléments du tableau? Si vous avez pas besoin d'afficher les indices, c'est-à-dire « l'élément à la position 0 » ou « le premier élément est », mais juste afficher le contenu du tableau, vous pouvez utiliser une itération à la C++ 2011; Donc, par exemple, commencez par afficher un message « Le tableau contient : » puis ensuite, donc, utilisez une boucle C++2011 avec les : ici, qui séparent d'un côté le tableau sur lequel on veut itérer qui s'appelle tab et puis, ensuite, choisir un nom d'un élément pour parcourir ce tableau donc on va ici l'appeler élément. On a une boucle ici qui va s'arrêter à cet endroit-là et qui utilise donc le nom de variable élément pour indiquer les différents éléments de ce tableau tab. Comme ici, parcourir le tableau pour l'afficher ne modifie pas le tableau, on n'a pas besoin d'ajouter le signe de référence entre auto et le nom de la variable. Si vous avez, par contre, besoin d'explicitement les indices, alors il faudra bien quelque part indiquer cet indice et donc avoir une variable supplémentaire. Pour ça on va donc déclarer une variable i. on va utiliser donc une boucle for comme on l'a vu au début, dans le cours sur les structures de contrôle qui va déclarer i de type size\_t démarrant à l'indice 0 premier élément du tableau est l'indice 0 séparez donc le for par les trois parties avec le ; Derrière la première partie, nous avons la condition. Ici, c'est tant que i est strictement inférieur à size, on va

notes

résumé

0m 1s



## Quelques exemples de manipulation de tableaux

Soit un tableau déclaré par:

→ `vector<double> tab(10);`

Affichage du tableau :

- ▶ si l'on n'a pas besoin d'expliquer les indices :

```
cout << "Le tableau contient : ";
for(auto element : tab) {
    cout << element << " ";
}
cout << endl;
```

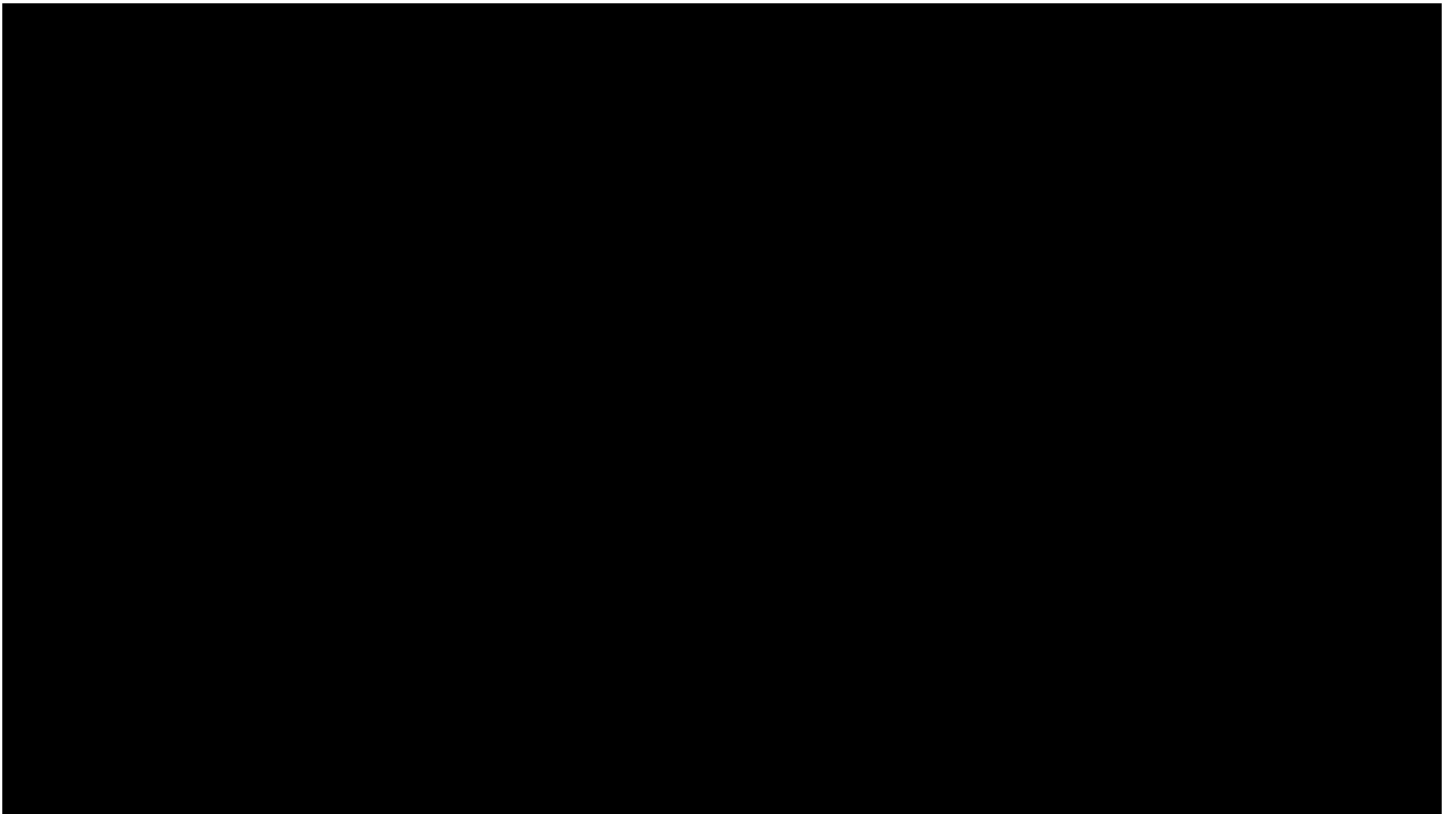
- ▶ si l'on veut expliciter les indices :

```
for(size_t i(0); i < tab.size(); ++i) {
    cout << "L'élément " << i << " vaut " << tab[i] << endl;
}
```

jusqu'à tab.size() moins 1

notes

résumé



et puis, ici de 1 en 1, ce qui nous permet donc, dans la boucle, d'afficher l'indice que l'on parcourt, de l'élément que l'on parcourt. Par exemple, ça va afficher « l'élément 0 vaut » et puis la valeur du premier élément stocké dans le tableau, stocké à l'indice 0. stocké à l'indice 0.

notes

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

résumé

2m 1s



.....

.....

.....

.....

.....