

Support de cours

Cours:

Initiation à la programmation (en C++)

Vidéo:

Blocs d'instructions (partie 2)

Concepts (extraits des sous-titres générés automatiquement) :

Variables globales. Exemple de variables globales. Ensemble du programme. Variable j. Bloc d'instructions. Exemple du if précédent. Ligne de déclaration. Variables. Variable globale. Première fois. Bloc correspondant. Valeur de cette variable. Reste du programme. Règle d'or. Première instruction.



[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/>

Fonctions : blocs

(Partie 2)

Initiation à la programmation (C++)

Vincent Lepetit, Jean-Cédric Chappelier et Jamila Sam

...

notes

résumé

0m 0s



Notion de portée

- ▶ Les variables déclarées à l'intérieur d'un bloc sont appelées **variables locales** (au bloc). Elles ne sont accessibles qu'à l'intérieur du bloc.
- ▶ Les variables déclarées en dehors de tout bloc (même du bloc `main() {}`) sont appelées **variables globales** (au programme). Elles sont accessibles dans l'ensemble du programme.

Bonnes pratiques:

1. **Ne jamais utiliser** de variables globales (sauf peut-être pour certaines constantes).

Les variables qui sont déclarées hors de tout bloc, même y compris hors du `main`, on a le droit de déclarer des variables comme ça hors de tout bloc, hors du `main`, c'est ce qu'on appelle des variables globales et je ne vous donnerais pas d'exemple de variables globales parce que la règle d'or c'est de ne jamais déclarer de variables globales, ne jamais utiliser de variables globales. Pourquoi?

notes

résumé

0m 1s





Parce que une variable globale justement, est accessible à l'ensemble du programme, ce qui fait que n'importe quelle instruction, n'importe où dans le programme peut, par exemple, modifier cette variable et donc il devient

notes

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

résumé

0m 25s



.....

.....

.....

.....

.....

Notion de portée

- ▶ Les variables déclarées à l'intérieur d'un bloc sont appelées **variables locales** (au bloc). Elles ne sont accessibles qu'à l'intérieur du bloc.
- ▶ Les variables déclarées en dehors de tout bloc (même du bloc `main() {}`) sont appelées **variables globales** (au programme). Elles sont accessibles dans l'ensemble du programme.

Bonnes pratiques:

1. **Ne jamais utiliser** de variables globales (sauf peut-être pour certaines constantes).

extrêmement difficile dans le programme de suivre la valeur de cette variable,

notes

résumé

0m 37s



Notion de portée

Déclarer les variables au plus près de leur utilisation

Par exemple, si la variable `j` n'est pas utilisée après la condition,

écrivez:

```
if (i != 0) {
    int j(0);
    ...
    j = 2 * i;
    ...
}
```

plutôt que:

```
int j(0);
if (i != 0) {
    ...
    j = 2 * i;
    ...
}
```

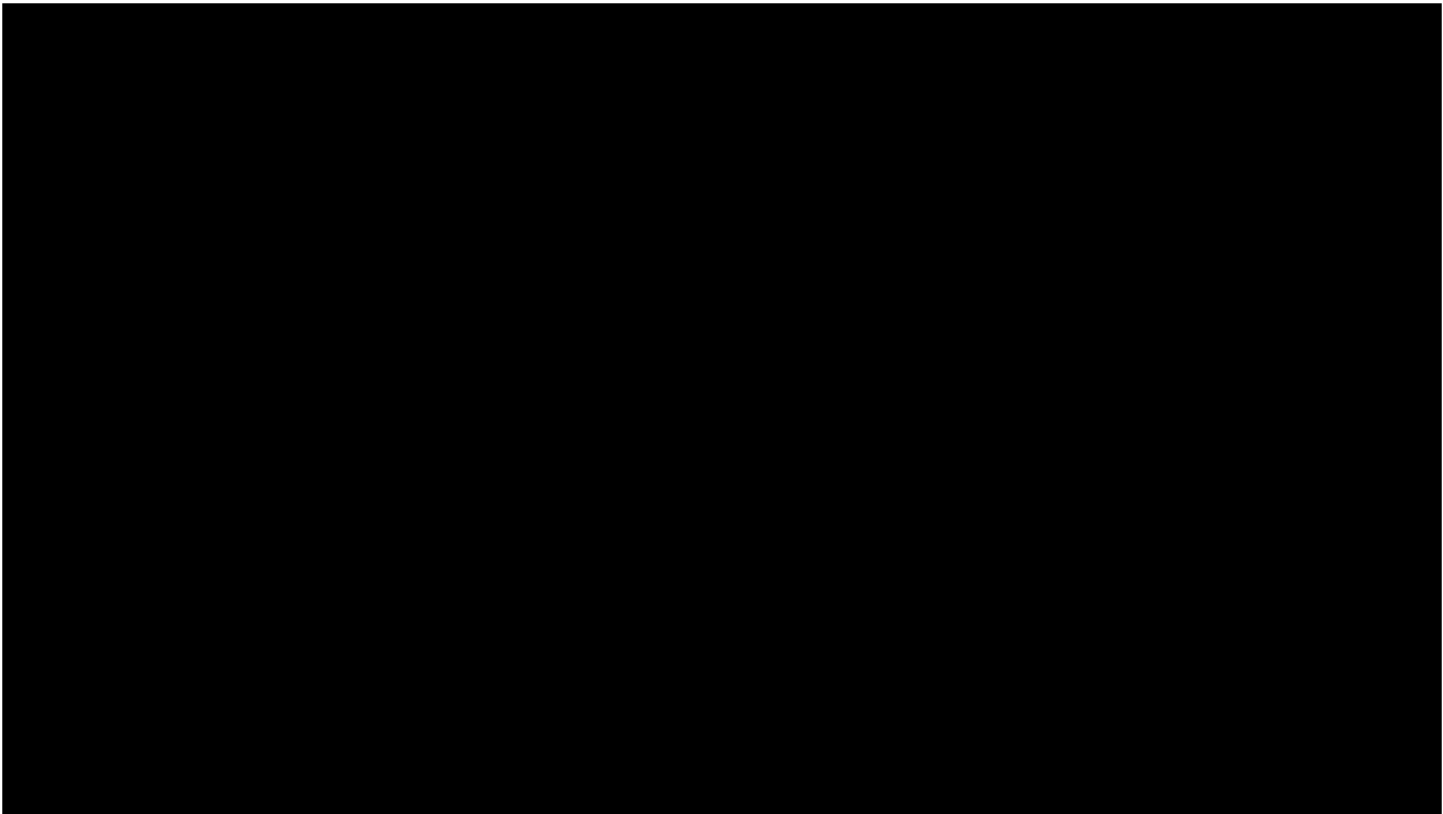
c'est pour cela qu'il ne faut absolument jamais utiliser de variables globales. D'ailleurs on vous conseille aussi de déclarer les variables au plus près de leur utilisation, c'est à dire déclarer, mettre la ligne de déclaration le plus proche possible de la ligne où on va, la première fois, utiliser la variable. Par exemple, dans l'exemple du if précédent, nous avons ici une variable `j` que l'on utilisait dans le bloc d'instructions contrôlé par le if et nous l'avons donc déclarée ici dans le bloc correspondant. Si on n'utilisait pas `j` dans la partie ici correspondante, on aurait même pu déplacer la ligne encore plus près ici, plus près de son utilisation. En tout cas, comme `j` ici, n'est utilisé que dans le bloc, on suppose qu'on n'utilise pas `j` dans le reste du programme, là en bas, alors une instruction comme ceci, qui

notes

résumé

0m 42s





déclarerait la variable hors du bloc, ne suit pas la recommandation de déclarer au plus près de l'utilisation de la variable et on la mettra donc au plus près de la première instruction qui utilisera le j; étant entendu ici que donc j n'est utilisé que dans ce bloc et n'est plus utilisé par la suite. la suite.

notes

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

résumé

1m 37s



.....

.....

.....

.....

.....