

Support de cours

Cours:

Initiation à la programmation (en C++)

Vidéo:

Expressions (partie 3)

Concepts (extraits des sous-titres générés automatiquement) :

Variables de type int. Valeurs décimales. Variable de type int. Règle générale. Piège de la division entière. Valeurs entières. Variables note1. Variables de type double d'un côté. Droite du signe égal. Type double. Type int. Valeur de l'expression. Valeur littérale. Valeur décimale. Moyenne égal moyenne.



[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/>

Expressions et opérateurs

(Partie 3)

Initiation à la programmation (C++)

Vincent Lepetit, Jean-Cédric Chappelier et Jamila Sam

...

notes

résumé

0m 0s



Affectation d'une valeur décimale à une variable entière

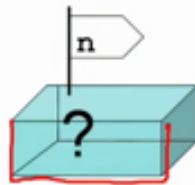
Quand on affecte une valeur décimale à une variable de type `int`, la partie fractionnaire est perdue.

Exemple:

```
double x(1.5);
```

```
→ int n;
```

```
→ n = 3 * x;
```



Jusqu'ici nous avons toujours vu des valeurs décimales et des variables de type `double` d'un côté, et d'un autre côté, des valeurs entières et des variables de type `int`. Que se passe-t-il quand, par exemple, on essaie d'affecter une valeur décimale à une variable de type `int` ? Par exemple, j'ai ici une variable `x` de type `double` qui contient la valeur 1,5 je déclare ici une variable `n` de type `int` que j'ai représenté ici, et dans cette affectation, j'essaie d'affecter la valeur de l'expression 3 fois `x`, c'est-à-dire... 4,5 à ma variable `n` de type `int` et qui est donc prévue que pour contenir des valeurs entières. Que se passe-t-il dans ce cas-là ? Le compilateur va convertir la valeur littérale 4,5 qui est de type `double`, en une valeur qui sera de type `int`.

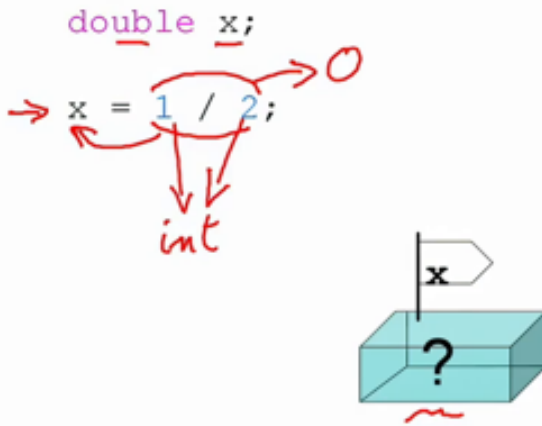
notes

résumé

0m 1s



La division entière



Cette conversion se fait tout simplement en perdant la partie fractionnaire c'est-à-dire que 4,5 va être converti en 4. Ce 4 est maintenant de type `int` et on va pouvoir effectuer l'affectation de ce 4 à la variable `n` de type `int`. Notez au passage que cette conversion de type `double` vers type `int` est un cas très particulier, on dispose également de la conversion de `int` vers `double` mais il s'agit là de deux exceptions. En règle générale, le C++ est ce qu'on appelle un langage fortement typé, qui va exiger que, dans une affectation par exemple, ce qui est à droite du signe égal soit du même type que ce qui est à gauche. Revenons maintenant sur le piège de la division entière ; j'ai déclaré ici une variable `x` de type `double`, que j'ai représentée ici, et dans cette affectation, j'affecte la valeur de l'expression `1 / 2` à ma variable `x`. A votre avis, que contient `x` après cette affectation ? Comme 1 et 2 sont tous les deux des valeurs littérales de type `int`, cette division est la division entière `1 / 2` dans le cas de la division entière vaut 0

notes

résumé

1m 13s



La division entière

Le problème peut se poser par exemple quand on calcule la moyenne de deux valeurs entières:

```
int note1(4), note2(5);  
  
double moyenne((note1 + note2) / 2);
```

Dans ce cas, moyenne vaut 4 au lieu de 4.5

Le problème peut se poser en pratique, quand par exemple, on essaie de calculer la moyenne de deux valeurs entières. J'ai par exemple ici déclaré deux variables note1 initialisée à 4 et note2 initialisée à 5 Elles sont toutes les deux de type int.

notes

résumé

2m 59s



La division entière

Une solution possible:

```
int note1(4), note2(5);  
  
double moyenne(note1 + note2);  
moyenne /= 2;
```

Je veux calculer leur moyenne, donc je voudrais obtenir 4,5 alors que si je fais cette déclaration, dans cette expression note1 et note2 sont donc de type int donc note1 + note2 est également de type int ; cette valeur littérale 2 est également de type int, il s'agit donc ici de la division entière. Je vais donc obtenir 9 divisé par 2 c'est-à-dire 4 alors que je voulais 4,5 Une solution possible, c'est de faire quelque chose comme ça. Ici, j'ai toujours note1 et note2 de type int initialisé toujours à 4 et 5,

notes

résumé

3m 13s



La division entière

Une solution possible:

```
int note1(4), note2(5);  
  
double moyenne(note1 + note2);  
→ moyenne /= 2;
```

9 int
moye

moyen
9.0

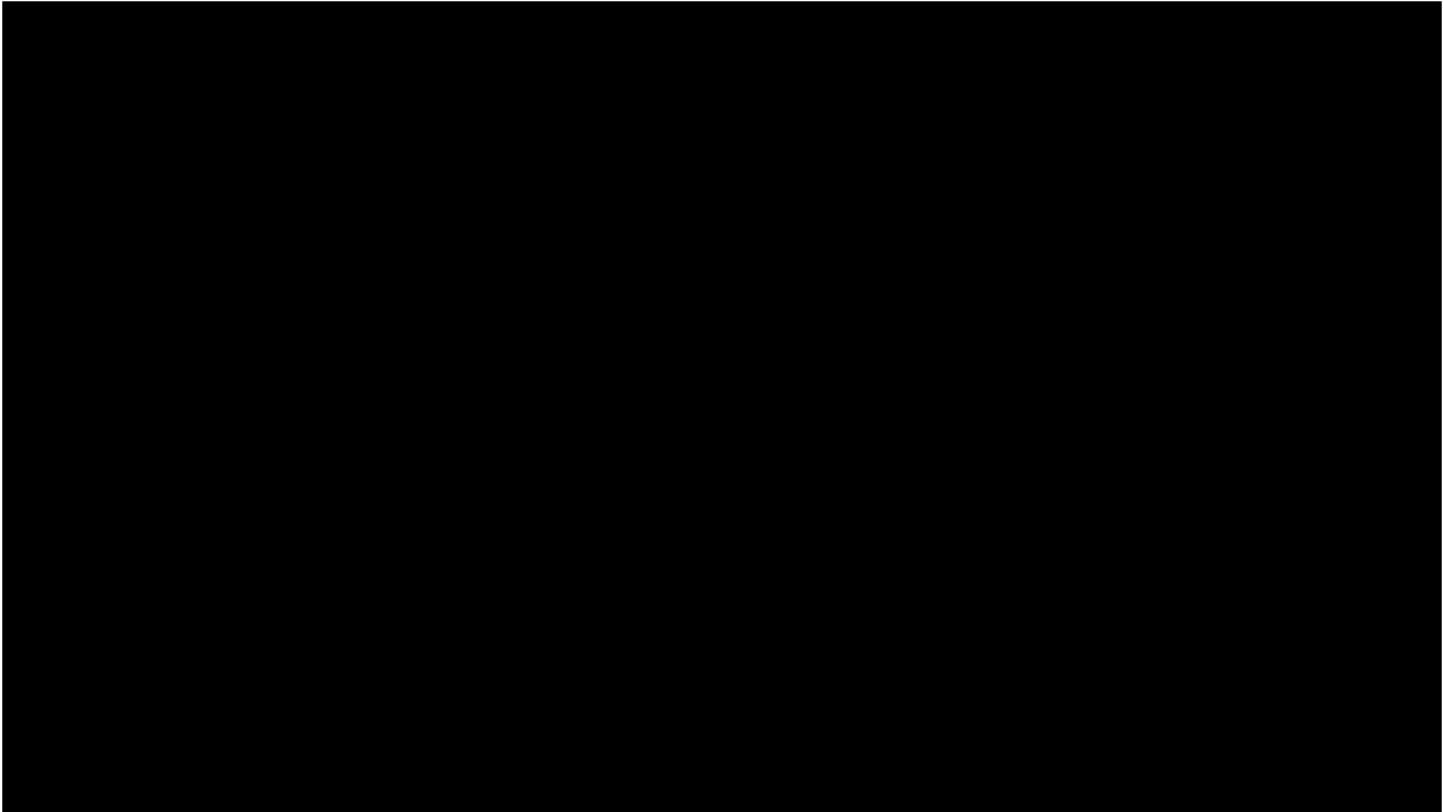
j'initialise moyenne au résultat de note1 + note2 note1 + note2 vaut 9 de type int mais je l'affecte à moyenne, qui a été déclarée de type double. Donc, moyenne va contenir 9 que je pourrais écrire 9.0 pour bien montrer qu'il est de type double. Cette fois-ci, quand j'écris moyenne divisé égal 2 qui est, je vous rappelle,

notes

résumé

4m 1s





équivalent à moyenne égal moyenne divisé par 2. Comme cette fois-ci moyenne est de type double. Même si 2 est de type int et va être converti en une valeur de type double, que je peux écrire 2.0 on va donc calculer 9.0 divisé par 2.0 qui est la division classique, mais cette fois-ci on va bien obtenir 4,5 et affecter 4,5 à la variable moyenne. et affecter 4,5 à la variable moyenne.

notes

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

résumé

4m 37s



.....