Support de cours

Cours:

Initiation à la programmation (en C++)

Vidéo:

Conditions (partie 2)

Concepts (extraits des sous-titres générés automatiquement) :

Termes de bonnes pratiques. Expressions de façon. Utilisation de conditions simples. Cas d'un petit programme. Valeurs initiales. Résultat de l'évaluation de la condition. Moyen d'une condition simple. Instruction de branchement conditionnel. Petits exemples. Moyen de l'opérateur de comparaison. Exécution des séquences. Résultat de l'évaluation de cette condition. Partie de l'expression. Affichage du message. Opérateurs logiques.

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

vers la vidéo

EPFL

Conditions
(Partie 2)

Initiation à la programmation (C++)

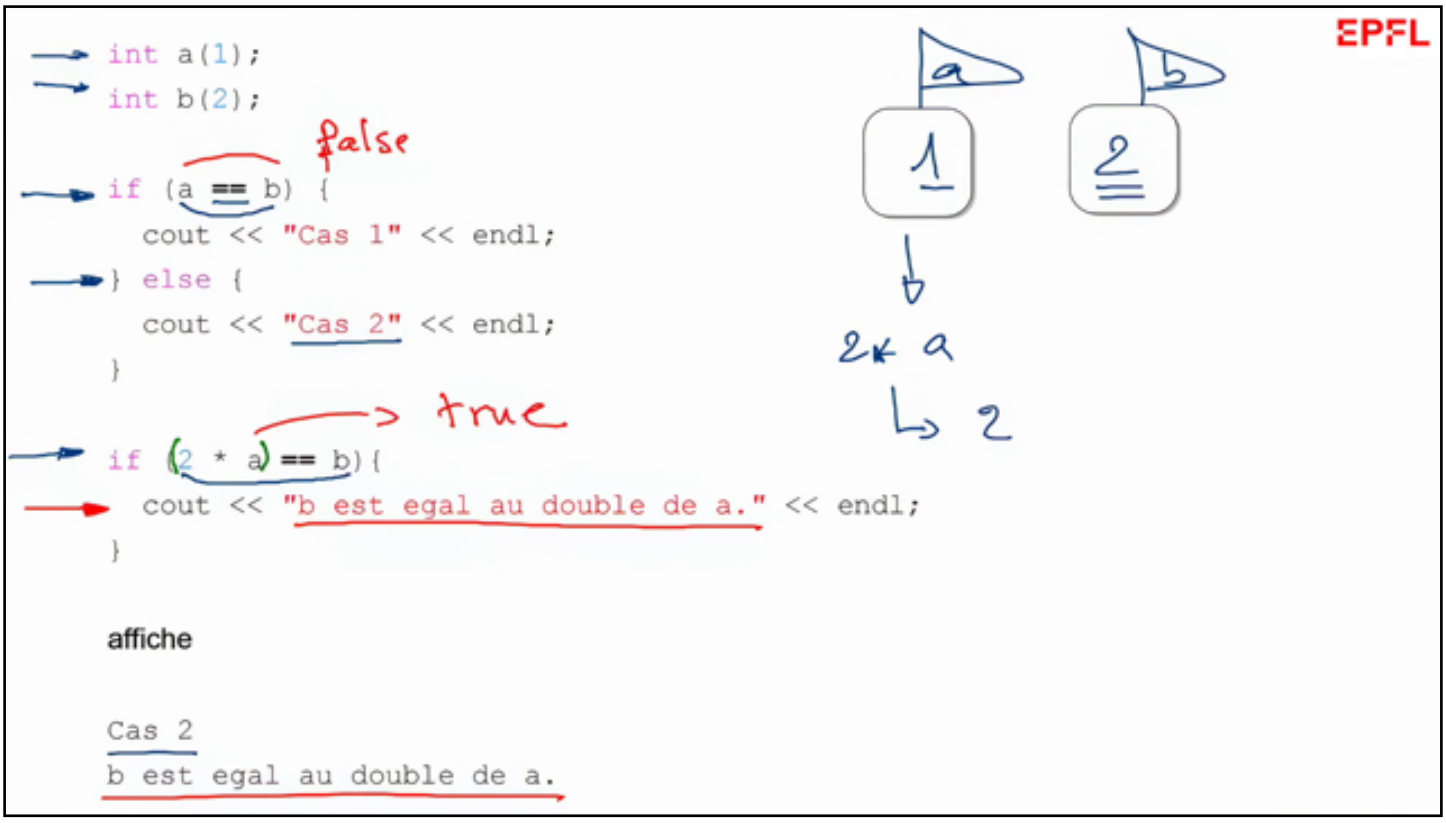
Vincent Lepetit, Jean-Cédric Chappelier et Jamila Sam

...

notes

résumé

0m 0s



Illustrons maintenant sur de petits exemples l'utilisation de conditions simples en C++. Nous avons ici le cas d'un petit programme qui commence par déclarer deux variables a et b en leur affectant des valeurs initiales, respectivement 1 et 2. Nous rencontrons ensuite une instruction de branchement conditionnel qui teste au moyen d'une condition simple si la valeur de a est la même que la valeur de b. Ceci se fait au moyen de l'opérateur de comparaison ==. Comme cette condition n'est pas vérifiée, le résultat de l'évaluation de la condition sera la valeur "false", ce qui aura pour conséquence de faire brancher l'exécution sur le bloc else de l'instruction if et donc va causer l'affichage du message "cas 2", que l'on voit ici. On continue ensuite l'exécution des séquences, et on rencontre une deuxième instruction de branchement conditionnel qui cette fois va comparer la valeur de deux fois a avec la valeur de b. On remarquera ici au passage que les opérateurs de comparaison permettent non seulement de comparer les valeurs de deux variables, comme c'est la cas ici, mais également les valeurs de deux expressions de façon plus générale. Si on évalue ici deux fois a, on obtient la valeur 2, qui est la même que la valeur de b, et donc le résultat de l'évaluation de cette condition sera "true" ce qui aura pour conséquence de faire brancher l'exécution sur le bloc positif, le bloc vrai de l'instruction if. Ceci provoquera l'affichage du message "b est égal au double de a", comme on peut le constater ici. En termes de bonnes pratiques, nous avons ici affaire à une expression qui est relativement simple, vous noterez cependant qu'il est conseillé, lorsque l'expression se complexifie, de parenthéser ces termes, donc ici on pourrait déjà commencer à le faire, en parenthésant par exemple

notes

résumé

0m 1s

cette partie de l'expression, ce qui rend l'expression beaucoup plus lisible. Nous venons de voir comment utiliser des

notes

résumé

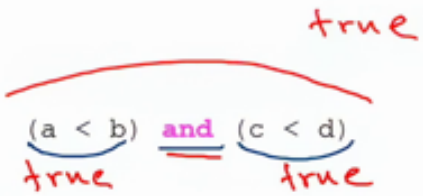
2m 1s

Les opérateurs logiques

On peut relier des conditions simples par des opérateurs logiques.

L'opérateur logique `and` (ET):

par exemple, la condition



est vraie **uniquement** si les deux conditions `(a < b)` et `(c < d)` sont toutes les deux vraies.

L'opérateur `and` peut aussi s'écrire `&&`: On aurait pu écrire

```
(a < b) && (c < d)
```

conditions simples en utiliser des opérateurs de comparaison. En pratique, il est souvent nécessaire de combiner plusieurs de ces conditions simples pour formuler une condition plus complexe. Pour combiner des expressions simples nous avons besoin d'utiliser un autre type d'opérateur, les opérateurs logiques. Par exemple, l'opérateur logique ET, qui en C++ s'exprime soit au moyen du mot-clé "and", soit au moyen du symbole "&&", est un opérateur fonctionnant avec deux opérandes qui sont deux expressions logiques, c'est-à-dire des expressions retournant "true" ou "false", le résultat de l'évaluation impliquant le and est "true" uniquement dans le cas où chacun des opérandes vaut "true".

notes

résumé

2m 6s

Dans tous les autres cas, imaginons par exemple que l'un des deux opérandes et a fortiori les deux, soient "false", le résultat de l'évaluation de l'expression globale sera "false" également, ici. "false" également, ici.

notes

résumé

2m 49s