

Support de cours

Cours:

Introduction à la programmation orientée objet (en C++)

Vidéo:

W14-03-2-heritageconstr-CPP-pt1

Concepts (extraits des sous-titres générés automatiquement) :

Ordre d'appel des constructeurs. Hiérarchie de classe. Constructeurs de cette classe. Toute première chose. Exécution du constructeur. Propres attributs. Première chose. Exécution de ce constructeur. Construction de sa partie. Liste d'initialisation. Constructeur de la classe. Suite tous. Initialisation des attributs. Sous partie. Travers de la relation d'héritage.



[vers la recherche de séquences vidéo](#)

(dans Introduction à la programmation orientée objet (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/>

Héritage : constructeurs (2)

(Partie 1)

Introduction à la programmation orientée objet (en C++)

Jean-Cédric Chappelier, Jamila Sam et Vincent Lepetit

...

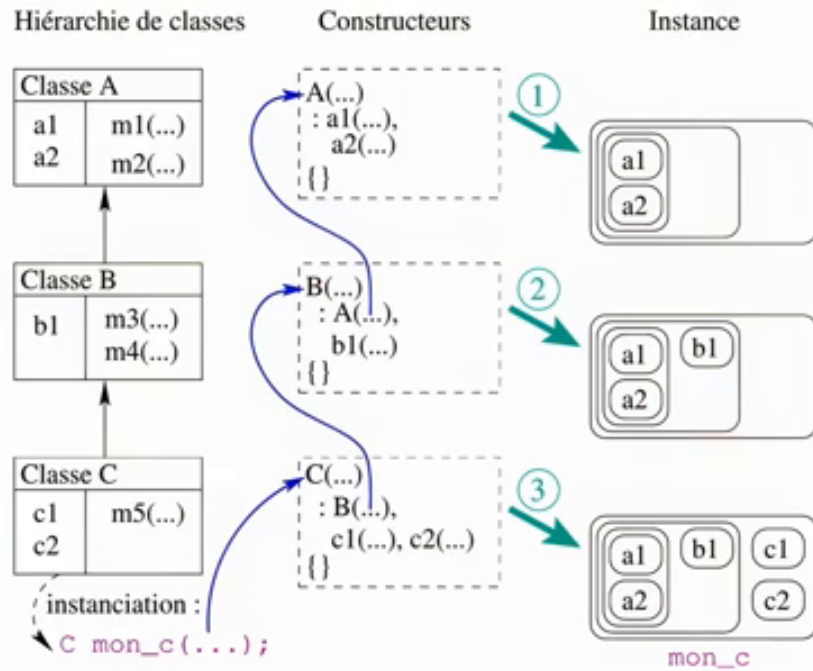
notes

résumé

0m 0s



Ordre d'appel des constructeurs



Revenons maintenant en détails sur l'ordre d'appel des constructeurs dans une hiérarchie de classe. Supposons par exemple que nous ayons défini une classe « C », qui a certains attributs, certaines méthodes, et qui hérite d'une classe « B », laquelle a ses propres attributs, ses propres méthodes ; et laquelle classe « B » hérite elle-même d'une classe « A » avec ses attributs et ses méthodes. Et nous déclarons donc une instance ici « mon_c » de la classe « C », avec un appel ici à un des constructeurs de cette classe « C ». Que se passe-t-il lorsque l'on déclare, donc comme ceci une instance, lors de l'appel donc de ce constructeur ? Ce constructeur ici, dans sa section liste d'initialisation, aurait un appel au constructeur de « B », que cette écriture soit explicite, ou que ce soit un appel au constructeur par défaut. Et puis ensuite bien sûr, l'initialisation des attributs. Donc lorsque l'on appelle ici le constructeur, on va commencer par appeler le constructeur de la classe « B ». Le constructeur de la classe « B » a lui-même dans sa liste d'initialisation, l'appel à un des constructeurs de sa superclasse. Et donc dans cet appel ici du constructeur de la classe « B » dans le constructeur de la classe « C », on a d'abord en tout premier, un appel au constructeur de la super-super-classe « A ». Donc la toute première chose qui va se faire, lors de l'appel du constructeur de « C », c'est l'exécution du constructeur de la super-super-classe, de la classe la plus haute dont on dérive. Et donc l'exécution de ce constructeur va commencer par initialiser des attributs « a1 » et « a2 ». Et donc dans la construction d'un « C », on aura lancé la construction d'un « B », lequel « B » aura lancé la construction d'un « A ».

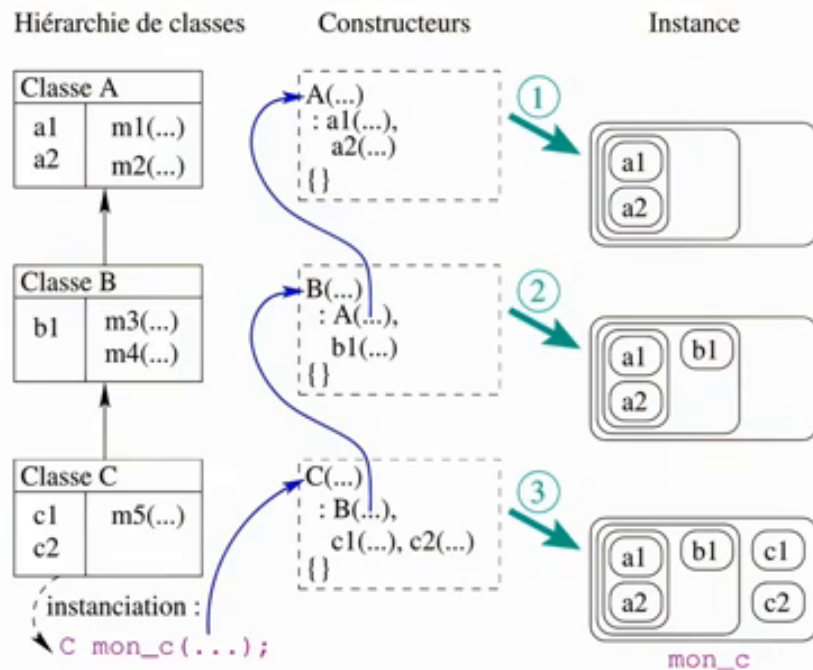
notes

résumé

0m 1s



Ordre d'appel des constructeurs



». La première chose qui se passe dans cette construction du « C », c'est la construction de sa partie « A », je vous rappelle qu'au travers de la relation d'héritage, « C » hérite tous les attributs de « B », « B » hérite tous les attributs de « A », donc dans « C », on a effectivement un attribut « a1 » et un attribut « a2 ». Donc dans la construction de ce « C », qui n'est pas encore terminée, on a commencé par construire la sous-partie « A ». C'est la première chose qui se passe. Une fois qu'on a terminé toute l'exécution du constructeur de « A », on revient ici, donc dans la liste d'initialisation du constructeur de « B », pour initialiser ses propres attributs, les propres attributs de la classe « B », et donc construire ici l'attribut « b1 » de la classe « B ». Et puis on termine donc l'exécution du constructeur de « B » et on est donc à ce stade là, dans l'exécution du constructeur de « C », on termine l'exécution de la liste d'initialisation du constructeur de « C », qui va donc construire la partie propre à « C », les attributs propres à « C », exécuter son corps et ceci termine la construction de « C ». Donc dans la construction d'une sous-sous-classe, on commence d'abord par appeler le constructeur de la classe la plus élevée dont on dérive puis ainsi de suite tous les constructeurs des super classes intermédiaires. Enfin on termine donc par la partie spécifique au constructeur de la classe, dont on est en train de créer une instance. de créer une instance.

notes

résumé