

Formation C++: C++ avancé: C++17 et plus (CPPBOOSTFR, 4 jours)

Description

Le cours Formation C++ : C++ avancé : C++17 et plus repousse les limites du développement C++ selon les normes C++17 et C++11. La formation comprend l'utilisation du C++ moderne, y compris les structures de données, la gestion de la mémoire, la programmation multithread, les locales, les templates et plus encore. L'ensemble du cours est consacré aux techniques de programmation avancées qui vous permettent de créer un code C++ puissant et flexible. Le cours comprend en option Boost, STL ou ATL.

Tarifs

- Tarification: \$3,750/person
- Rabais de 10% lorsque vous inscrivez 3 personnes.

Plan de cours

Premiers pas avec C++11

La norme C++03 et C++11

Architecture de la mémoire

Utiliser le mot-clé automatique

Syntaxe alternative des fonctions C++

Opérateurs de moulage

La nouvelle boucle for basée sur l'intervalle

Fonctions et expressions lambda

Boost.Lambda - Boost.Phoenix

Assertions statiques

Utilisation de const et constexpr

Structures de données

Comprendre les Plain Old Data (POD) triviaux et à présentation standard

Énumération faiblement et fortement typée

Unions non affectées

Tuples

Hashmap et ensembles

Liste des initialisateurs

Comprendre et utiliser les traits de type dans C++11 et Boost

Gestion de la mémoire

Architecture de la mémoire

Utilisation de unique_ptr, shared_ptr et weak_ptr

Utilisation d'une référence wrapper

Les opérateurs sizeof, alignas et alignof

Utilisation de set_new_handler()

Différence entre NULL et nullptr

Mise en œuvre de la collecte des déchets

Programmation multithread

Architecture mémoire multithread

Installations d'enfilage

Stockage local du fil

Comprendre les opérations atomiques

Utilisation de `std::mutex`, `std::lock_guard` et `std::unique_lock`

Utilisation de `std::future`, `std::packaged_task` et `std::async`

Chaînes de caractères

Chaînes ascii, UTF et larges

Les littéraux définis par l'utilisateur

Expressions régulières avec C++11 et Boost.Regex

Boost.Xpressive

Analyse syntaxique LL avec Boost.Spirit

Locales

Comprendre les locales

Facettes locales

Travailler avec des chiffres

Formatage de la date, de l'heure et de la devise

Utilisation des messages et des catalogues de messages

Classes

Changements dans le constructeur de C++03 à C++11

Utilisation des constructeurs pairs

Hériter des constructeurs

Constructeurs et membres par défaut

Membres du groupe supprimés

Remplacement explicite de la fonction

Création des classes finales

Opérateurs de conversion explicites

Utilisation des modèles

Modèle variadique

Utilisation de modèles externes

Bibliothèque Boost

Qu'est-ce que Boost

Bibliothèques Boost faisant partie de C++11

Utilisation de Boost TR1

Boost.Utility

Conteneurs tels que Array, Circular Buffer et Dynamic Bitset

Meilleure gestion temporelle avec Timer, Date-Time et Chrono

Utilisation de threads et de pools de threads

Meilleures entrées-sorties : Système de fichiers, sérialisation et ASIO

Accumulateurs Boost.

Boost.Bind et Boost.Function

Boost.Iterator et Boost.Range/Range_ex

Options du programme Boost.

Signalisation interprocessus à l'aide de Boost.Signal et Boost.Signal2

Tests unitaires avec Boost.Tests