

Durée : 3 jours

Réf : C++ Moderne

C++ moderne : versions 11 à 20

Cette formation présente les nouvelles fonctionnalités de C++ 11 à 17 et certaines de C++ 20. Elle couvre les nombreuses modifications qui permettent d'écrire rapidement du code plus sûr et plus rapide (constructeur move, fonctions lambda, inférence de type, smart-pointers, threads, range, ...)

Objectifs

- Développer du code de meilleure qualité en utilisant les nouvelles fonctionnalités de C++
- Savoir utiliser les nouveautés de la bibliothèque standard du C++.

Public

- Développeurs C++

Pré requis

- Connaissance et pratique du C++ (syntaxe de base, programmation orientée objet)

Evaluation des acquis

- Exercices pendant la formation

Méthodes et moyens

- Support pdf
- Exercices et corrigés en ligne

Points forts

- Nombreux exercices
- Alternance théorie/pratique
- Assistance après la formation

Contact

- 04 58 00 02 22
- contact@webformation.fr

1 Introduction aux versions récentes de C++

- Historique C++98, C++11, C++14 et 17, 20
- Éléments obsolètes ou redéfinis (auto, register, auto_ptr, ...)
- Le futur avec C++23

2 Nouvelles fonctionnalités du langage

- Pointeur null : nullptr
- Listes d'initialisation
- Parcours d'un conteneur

3 Typage par inférence

- auto, decltype
- Syntaxe de définition d'une fonction
- Structured binding

4 Fonctions

- Fonctions constantes à la compilation
- Propagation des exceptions
- Support des fonctions dans la STL : std::function, std::bind,
- Définition d'une fonction anonyme
- Définition d'une fermeture

5 Classes

- Initialisation de variables
- Appels de constructeurs
- Spécifications des constructeurs par défaut (delete, default)
- Héritage des constructeurs
- Blocage de l'héritage
- Énumérations typées

6 Variables

- Alignement
- Nouveaux types

7 Contrôle du compilateur

- Static_assert
- If constexpr

8 Nouveautés de syntaxe

- Chevrons
- Using
- if et switch avec initialisation (C++ 17)
- Espace de nom inline et nested (C++ 17)

9 Concept et contraintes (C++20)

- Erreurs de compilation lors d'une mauvaise utilisation de template
- Présentation de la bibliothèque Concept
- Création de contraintes

10 Template

- Rappels sur les templates de classes et de fonctions
- Définition externe
- Template variadiques
- Définition de types partiels
- Inférence de type (C++ 17)
- Évaluation des expressions (C++ 17)

11 Référence vers des Rvalue et sémantique du move

- Définition d'une Rvalue
- Constructeur move
- Opérateur d'affection
- std::move

12 Smart Pointers

- Dépréciation de auto_ptr
- unique_ptr,
- shared_ptr
- weak_ptr

13 Multithreading

- Présentation
- Attribut thread_local
- Classes thread et jthread
- mutex, sémaphore
- Conditions, Verrous, future et promise

14 Range Library (C++20)

- Présentation
- Range adapter et generator
- Enchaînement
- Filter
- Transform

15 Nouvelles bibliothèques de la librairie standard

- <regex>
- <chrono>
- Fonctions mathématiques
- Opérations atomiques
- Nouveaux types (optional, any, variant) (C++17)
- Bibliothèque Filesystem (C++ 17)
- Parallélisation des algorithmes (C++ 17)

16 Nouveaux conteneurs de la librairie standard

- array
- tuple - pair
- unordered_map, unordered_set, unordered_multimap, unordered_multiset