

**Durée : 3 jours**

Réf : C++20

**C++ 20**

Cette formation présente les nouvelles fonctionnalités de C++ 20. Elle s'adresse à des développeurs maîtrisant déjà le C++ moderne (au moins C++ 11).

**Objectifs**

- Développer du code de meilleure qualité en utilisant les nouvelles fonctionnalités de la version 20
- Savoir utiliser les nouveautés de la version 20 de la bibliothèque standard du C++.

**Pré requis**

- Connaissance et pratique du C++ 11

**Méthodes et moyens**

- Support papier et électronique

**Points forts**

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

**Contact**

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

**1 Rappels sur les versions de C++**

- C++ 98
- Historique des versions C++11, C++14 et 17
- Présentation des nouveautés majeures de C++20 (module, concept, range et coroutines)

**2 Modules**

- Contraintes de la compilation et du préprocesseur
- Présentation des modules
- Implémentation des modules
- Utilisation

**3 Concepts**

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

**4 Range Library**

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

**5 Coroutines**

- Programmation asynchrone
- Présentation des coroutines
- Générateur
- Exemples d'utilisation

**6 Nouvel opérateur de comparaison**

- Opérateur <=>
- Opérateur <=> par défaut
- Impact sur les comparaisons

**7 Améliorations du langage**

- using enum
- amélioration des expressions lambda
- Initialiseurs désignés
- Extensions de constexpr : consteval et constinit
- Attributs likely,unlikely,no\_unique\_address

**8 Formatage de texte**

- Présentation de la bibliothèque
- Syntaxe et utilisation de std::format
- Syntaxe et utilisation de format\_to et format\_to\_n
- Extensions de format

## 9 Multithread

- Rappels sur les threads depuis C++11
- jthread
- Sémaphores
- Latch et Barrier
- Smart pointeurs atomiques

## 10 Autres extensions de la STL

- Calendriers
- `std::span`