

# C++ Programmation avancée

## INFORMATIONS GÉNÉRALES

**Type de formation :** Formation continue

**Éligible au CPF :** Non

**Domaine :** Développement

**Action collective :** Non

**Filière :** Action collective ATLAS "Java, .Net, C++"

**Rubrique :** C++

**Code de formation :** F28030

## € Tarifs

**Prix public :** 2420 €

## Tarif & financement :

Nous vous accompagnons pour trouver la meilleure solution de financement parmi les suivantes :

**Le plan de développement des compétences de votre entreprise :** rapprochez-vous de votre service RH.

**Le dispositif FNE-Formation.**

**L'OPCO** (opérateurs de compétences) de votre entreprise.

**France Travail:** sous réserve de l'acceptation de votre dossier par votre conseiller Pôle Emploi.

**CPF -MonCompteFormation**

Contactez nous pour plus d'information : contact@aston-institut.com

## PRÉSENTATION

### Objectifs & compétences

Maîtriser les concepts avancés du C++ Connaître les principales nouveautés du C++ 11

### Public visé

Développeur C++

### Pré-requis

Une bonne connaissance du langage C++ est nécessaire

## 📍 Lieux & Horaires

**Durée :** 28 heures

**Délai d'accès :** Jusqu'à 8 jours avant le début de la formation, sous condition d'un dossier d'inscription complet

## PROGRAMME

### Chapitre 1 : Rappels sur les classes en C++

Rôle du constructeur de copie

Rôle de l'opérateur d'affectation surchargé

Intérêt du destructeur virtuel

Exercice : mise en oeuvre du const\_cast et static\_cast

### 📅 Prochaines sessions

Consultez-nous pour les prochaines sessions.

### Chapitre 2 : Les conversions en C++

Présentation const\_cast static\_cast reinterpret\_cast dynamic\_cast

Exercice : mise en oeuvre du const\_cast et static\_cast

Les

c

### Chapitre 3 : Les pointeurs sur membres Syntaxe

Mise en oeuvre

Exercice : Utilisation de pointeurs sur méthodes pour effectuer des calculs mathématiques

C

### Chapitre 4 : Les pointeurs intelligents

Principe de la gestion de ressources

Intérêt des smart pointers unique\_ptr et shared\_ptr

Exercice : remplacement de pointeurs nus par des pointeurs intelligents

Chapitre 5 : L'identification de type à l'exécution (RTTI) Principe Fonctionnement du dynamic\_cast La classe type\_info Exercice : mise en oeuvre de RTTI pour détecter un type particulier dans une hiérarchie de classes

Chapitre 6 : Les exceptions Principe Les classes d'exceptions Lever une exception Gestionnaires d'exceptions Liste d'exceptions Hiérarchies d'exceptions Exceptions standards Exceptions et constructeurs Exceptions et gestion des ressources Bonnes pratiques Exercice : Mise en place d'une gestion d'exceptions dans une application en remplacement des messages d'erreur

C

Chapitre 7 : Les templates Présentation Syntaxe Template de fonctions Template de classes Template de méthodes Instanciation des templates Spécialisation des templates Instanciations partielles Avantages / inconvénients Classe/typename/template Exercice : Mise en oeuvre de fonctions template de fonctions, de classes

Chapitre 8 : Les foncteurs Présentation Foncteurs prédéfinis Adaptateurs Intérêt des foncteurs Exercice : Mise en oeuvre d'un foncteur avec un algorithme de la STL

Chapitre 9 : La bibliothèque STL Présentation Les conteneurs Les allocateurs Les itérateurs Les algorithmes Les entrées-sorties Exercice : Mise en oeuvre de quelques conteneurs, d'algorithmes et template d'entrées-sorties Complément d'E-Learning avec Lynda sur la gestion des exceptions

Chapitre 10 : Les idiomes Traits Policy SFINAE (substitution failure is not an error) CRTP (curiously recurring template pattern) Exercices : mise en oeuvre de chacun des idiomes

Chapitre 11 : La métaprogrammation Présentation Avantages/inconvénients Optimisations Exercices : mise en oeuvre de la métaprogrammation pour effectuer des calculs par le compilateur

Chapitre 11 : Présentation de la bibliothèque Boost avec les tests unitaires Présentation des différents modules Static\_assert Property\_map Tuple Variant Threads BoostTest : librairie permettant de faire les tests unitaires Exercices : mise en oeuvre de quelques template de Boost avec tests unitaires

C

Chapitre 12 : Principales nouveautés du C++11 Les mots-clés auto, decltype et constexpr La constante nullptr La nouvelle boucle for Références rvalue Copie par déplacement Transfert parfait Les fonctions lambda Exercices : Mise en œuvre des mot-clés auto, d'decltype et constexpr Mise en œuvre des références rvalue pour la création et la copie d'objet par déplacement Mise en œuvre des expressions lambdas Complément d'E-Learning avec Lynda sur l'essentiel de C++11

## MODALITÉS

### Modalités

**Modalités :** en présentiel, distanciel ou mixte . Toutes les formations sont en présentiel par défaut mais les salles sont équipées pour faire de l'hybride. – Horaires de 9H à 12H30 et de 14H à 17H30 soit 7H – Intra et Inter entreprise.

**Pédagogie :** essentiellement participative et ludique, centrée sur l'expérience, l'immersion et la mise en pratique. Alternance d'apports théoriques et d'outils pratiques.

**Ressources techniques et pédagogiques :** Support de formation au format PDF ou PPT Ordinateur, vidéoprojecteur, Tableau blanc, Visioconférence : Cisco Webex / Teams / Zoom.

**Pendant la formation :** mises en situation, autodiagnostic, travail individuel ou en sous-groupe sur des cas réels.

### Méthode

**Fin de formation :** entretien individuel.

**Satisfaction des participants :** questionnaire de satisfaction réalisé en fin de formation.

**Assiduité :** certificat de réalisation.

**Validations des acquis :** grille d'évaluation des acquis établie par le formateur en fin de formation.