

# C++, programmation Objet

## INFORMATIONS GÉNÉRALES

**Type de formation :** Formation continue

**Éligible au CPF :** Non

**Domaine :** Développement

**Action collective :** Non

**Filière :** Open Source - LAMP : Linux Apache PHP

**Rubrique :** Langages : Python, Scala, GO,...

**Code de formation :** AS552

## € Tarifs

**Prix public :** 2770 €

## 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](mailto:contact@aston-institut.com)

## PRÉSENTATION

### Objectifs & compétences

A l'issue de la formation, le stagiaire sera capable de mettre en œuvre les principes fondamentaux de la conception orientée objet et de concevoir des applications en C++. Appliquer les principes de la Conception Orientée Objet Maîtriser la syntaxe du langage C++ Concevoir des applications C++ utilisant des classes Utiliser les outils de développement associés au langage C++ Maîtriser les ajouts majeurs de la norme C++ 11

### Public visé

Développeurs, ingénieurs, chefs de projets proches du développement.

### Pré-requis

Connaître les principes de la programmation orientée objet et disposer d'une expérience d'un langage de programmation...

## PROGRAMME

Mise en place de plusieurs projets mettant en exergue les acquis tout au long de la formation

### Chapitre 1 :

#### Les fondamentaux :

- De C à C++ • Installation des outils
- Premier programme
- Les variables
- Les commentaires
- Les mots-clés du langage
- Opérateurs
- Chaînes de caractères
- Tableaux statiques
- Console
- Conditions
- Constantes
- Boucles
- Références
- Utilisation d'espaces de noms
- Déduction automatique de type
- Tableaux dynamiques
- Fonctions

Atelier pratique :

Calculatrice dans la console pour les opérateurs.

Gestion d'un stock de produits.

### Chapitre 2 :

#### Structure d'un programme C++ • Les espaces de noms

- Les fonctions

## 📍 Lieux & Horaires

**Campus :** Ensemble des sites

**Durée :** 35 heures

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

**Distanciel possible :** Oui

## 📅 Prochaines sessions

Cliquez sur la date choisie pour vous inscrire :

■ 07 / 07 / 2025

📍 : Ensemble des sites

✓ : Distanciel possible

⌚ : 35 heures

📅 : 5 jours

■ 06 / 10 / 2025

📍 : Ensemble des sites

✓ : Distanciel possible

⌚ : 35 heures

📅 : 5 jours

■ 01 / 12 / 2025

📍 : Ensemble des sites

✓ : Distanciel possible

⌚ : 35 heures

📅 : 5 jours

- Les fonctions in line
- Surcharge de fonctions
- Paramètres en nombre variable

Atelier pratique :

Nous reprenons les exercices précédents et analysons les espaces de noms.

La calculatrice se verra agrémentée de surcharge de fonctions sur certaines opérations, de même pour l'exercice « Gestion d'un stock de produits ».

### **Chapitre 3 :**

#### **Approche objet**

- Inconvénients de l'approche procédurale
- Origines de l'approche objet
- Encapsulation
- Abstraction
- Héritage
- Polymorphisme

• Panorama des principaux langages objet

Atelier pratique : Cour théorique basé sur des exemples de code. Les exercices auront lieu dans le chapitre « Classes et Objets »

### **Chapitre 4 :**

#### **Le pré-compilateur et compilation séparée**

- Principe de la compilation séparée
- Rôles de fichiers d'entête
- L'inclusion de fichiers
- Principe de la compilation conditionnelle
- Les constantes
- Les macro-instructions

Atelier pratique : Nous faisons une analyse de la compilation et mettons en place des « macro » pour envoyer des logs de l'état de la compilation dans un fichier.

### **Chapitre 5 :**

#### **Les classes et les objets**

- Généralités
- Déclaration de classe
- Les membres de la classe
- Contrôle d'accès aux membres
- Constructeur et destructeur
- Instanciation
- Accès aux attributs et méthodes
- Membres spéciaux : this
- Les membres amis
- Les membres static
- Les méthodes const
- Classes composées d'objets
- Destruction des instances

Atelier pratique :

Nous reprenons l'exercice « Gestion d'un stock de produits » et nous le transformons en projet purement objet.

### **Chapitre 6 :**

#### **Concepts avancés**

- Rôle du constructeur de copie
- Surcharge de l'opérateur d'affectation

Atelier pratique :

Ajout d'un constructeur de copie et surcharge de l'opérateur d'affectation dans les classes développées dans le précédent exercice et permettons la copie des produits.

### **Chapitre 7 :**

#### **La surcharge des opérateurs**

- Sommaire des opérateurs
- Conversion de type implicite
- Fonction amies pour la surcharge d'opérateurs
- Opérateur de conversion de type

Atelier pratique :

Ajout d'opérateurs surchargés comme méthodes et comme fonctions amies

### **Chapitre 8 :**

#### **L'héritage et le polymorphisme**

- Types d'héritage

- Mécanisme d'héritage
- Classes dérivées
- Hiérarchie de classes
- Polymorphisme et méthodes virtuelles
- Héritage multiple
- Classes de base virtuelles

Atelier pratique :

Mise en œuvre d'un héritage public simple entre trois classes, du polymorphisme et d'un héritage multiple répété en se basant sur l'exercice « Gestion d'un stock de produits ».

### **Chapitre 9 :**

#### **Gestion des exceptions**

- Gestion sur place avec try, catch, finally
- Propagation • Lever une exception avec throw

Atelier pratique :

En se basant sur l'exercice « Gestion d'un stock de produits », mise en place d'une gestion d'exceptions pour traiter des anomalies lors du passage de paramètres

### **Chapitre 10 :**

#### **Introduction aux pointeurs intelligent**

- Inconvénients des pointeurs nus
- Principe des pointeurs smart pointers
- Mise en œuvre de unique\_ptr et shared\_ptr

Atelier pratique :

Mise en œuvre de pointeurs intelligents pour remplacer des pointeurs nus

### **Chapitre 11 :**

#### **Introduction aux templates**

- Principe des templates
- Instanciation des templates
- Template de fonctions
- Template de classes

Atelier pratique :

En se basant sur l'exercice « Gestion d'un stock de produits » : création et utilisation d'un template de fonction et d'un template de classe simple

### **Chapitre 12 :**

#### **La bibliothèque STL**

- Présentation de la Standard Template Library STL
- Présentation de quelques conteneurs
- Présentation des templates d'entrées/sorties

Atelier pratique :

Mise en œuvre du conteneur std::vector

Mise en œuvre des templates d'entrées-sorties sur fichiers

## **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.

**POUR ALLER PLUS LOIN**

**Et après**

N