

C++

Programmer en objet avec C++

Après 30 ans d'existence, le langage C++ reste l'un des langages les plus populaires.

En offrant le meilleur compromis entre proximité du système et concepts objets de haut niveau, il reste un langage de choix pour adresser tout type de développement informatique.

Cette formation permettra aux participants de concevoir des applications C++ utilisant des classes, de devenir architectes de classes et de créer leur propre architecture.

Détails

- Code : PR-CPP
- Durée : 5 jours (35 heures)

Public

- Chefs de projets
- Développeurs

Pré-requis

- Maîtriser les concepts de développement du langage C

Objectifs

- Maîtriser les bases du langage
- Appréhender ses mécanismes puissants
- Maîtriser la programmation par objet

Programme

Les concepts de la Programmation objet essentiel pour bien commencer en C++

- Les paradigmes de la P.O.O.
- La classification
- L'encapsulation
- L'agrégation, la composition de classes
- L'héritage, la dérivation, la sous classification
- L'abstraction, la généricité

Le langage C++, le langage

- Les types de données, les opérateurs
- Le type référence et le pointeur
- Les structures de contrôles
- Les fonctions et les procédures
- L'opérateur de résolution de portée
- Liaison C - C++ Les paramètres par défaut des fonctions
- Les fonctions " inline " Comparaison avec Java/C#

Les classes en C++

- Définition d'une classe
- Les qualificatifs " public " et " private "
- Les constructeurs par défauts
- Les constructeurs surchargés
- Les constructeurs par copie
- Les destructeurs
- Les fonctions membres de classes et les fonctions membres d'objets

L'agrégation en C++

- Principe
- Ordre d'appel des constructeurs et destructeurs

- La liste d'initialisation de constructeur

L'héritage, la dérivation

- L'héritage public simple
- Règle des constructeurs destructeurs
- Les règles de conversion
- Le qualificatif d'accès " protected "
- Diverses syntaxes et accès
- L'héritage privé
- L'héritage multiple
- Comparaison avec Java

Le polymorphisme

- Les fonctions virtuelles
- Le mécanisme du polymorphisme
- Les destructeurs virtuels
- Les classes abstraites et les fonctions virtuelles pures
- Comparaison avec Java

La gestion des exceptions

- Principe du traitement des anomalies
- Try, catch et throw
- Traiter une anomalie
- Lever une anomalie
- Comparaison avec Java

Les templates en C++

- Principe de la généricité
- Les classes templates
- Les fonctions templates
- Exemple d'un template

Modalités

- **Type d'action** : Acquisition des connaissances
- **Moyens de la formation** : Formation présentielle - 1 poste par stagiaire - 1 vidéo projecteur - Support de cours fourni à chaque stagiaire
- **Modalités pédagogiques** : Exposés - Cas pratiques - Synthèse
- **Validation** : Exercices de validation - Attestation de stages
- **Profils des intervenants** : Toutes nos formations sont animées par des consultants-formateurs expérimentés et reconnus par leurs pairs.