

Formation Maîtriser le langage C++



La mise en pratique des concepts de modélisation Objet nécessite des langages de programmation adaptés. Le langage C++, digne successeur du langage C, fût l'un des premiers acteurs à voir le jour sur le marché du développement. Depuis une vingtaine d'années, ce langage a prouvé à de très nombreuses reprises son fort potentiel au niveau de la performance et de la stabilité des applications

Objectifs

- Acquérir les principes de base de la programmation Objet (polymorphisme, héritage, encapsulation)
- Maîtriser le langage C++
- Apprendre à réaliser des applications réutilisables
- Connaître les meilleures pratiques de tests de qualité en C++

Public concerné

- Développeurs
- Concepteurs

Prérequis

- Avoir suivi la formation Développer avec le langage C (LA100) ou connaissances équivalentes

Une formation de 5 jours

Caractéristiques

Tarif : 2290 euros HT par personne
Numéro de formateur : 11754730575
Nombre d'heures : 35
Référence : LA200
Contact : Jean JUILLET
Telephone : 01.42.62.91.86
Email : contact@kaptive.com

Paris

13/01/2014
07/04/2014
30/06/2014
27/10/2014

Description des modules

Du procédural à l'objet

- Bref historique des langages
- Critères de qualité dans un développement logiciel
- Aperçu général des langages orientés Objet

Les concepts objet

- Classes, objets (constructeurs et paramètres par défaut)
- Encapsulation (visibilité public-private-protected, namespaces)
- Membres et méthodes de classes (static)
- Héritage simple
- Héritage multiple (classe de base virtuelle)
- Polymorphisme (virtuel), Classes abstraites (virtuel pures)
- Interfaces

Syntaxe C++

- Fonctions Friends
- Fonctions Inline
- Paramètre caché this
- Point d'entrée main avec arguments
- Type références
- Classes imbriquées (Inner class)
- Inclusion des headers et références multiples
- Utilisation du qualificateur const
- Bibliothèque standard d'E/S
- Gestion dynamique de la mémoire
- Destructeurs virtuels
- Typage dynamique avec RTTI

Traitement des exceptions

- Traitement des erreurs dans les programmes
- Traitement des erreurs en C++
- Traitement des exceptions imbriquées
- Classes d'exception applicatives

Surcharge

- Surcharge des fonctions
- Surcharge d'opérateurs
- Surcharge de l'opérateur new (avec nothrow) et set-new_handler

Patrons - Templates

- Définitions de patrons, syntaxe et instanciation
- Patrons de fonctions
- Template de classes (exemple avec la classe smart pointer)

Points clé de la STL

- Présentation des principaux conteneurs (vector, list, set, map, deque)
- Critères de choix pour un conteneur STL
- Les itérateurs
- Les algorithmes génériques

Introduction aux Design Pattern

- Pattern singleton
- Modèle observateur MVC avec exemple d'implémentation en C++

Qualité logicielle

- Les best practices en C++

- Règles de conception et astuces de codage
- Conclusion avec comparatif Java / C++