775 route de l'aérodrome
ZA Agroparc
84000 Avignon
Tél 04 90 85 98 12
Fax 04 90 86 26 24
info@cit.fr
www.cit.fr

LANGAGE C++ - Les fondamentaux

Objectif : Cette formation permettra aux participants d' appréhender les principes fondamentaux de la conception objet et de les appliquer de façon effective au travers d'une étude de cas incrémentale qui intégrera de nombreux idiomes C++.

À l'issue de la formation, le stagiaire sera capable de :

- 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++

Prérequis : Connaître les principes de la programmation orientée objet / Disposer d'une expérience d'un langage de programmation.

- 5 jours -

Introduction à la conception orientée objet

- Les paradigmes de développement (procédural, objet, fonctionnel, déclaratif)
- Principes de conception objet : Abstraction, Encapsulation, Héritage et Polymorphisme
- Généricité

Syntaxe et modèle de C++

- Point d'entrée d'une application
- Utilisation de la console, objets d'entrée/sortie (streams)
- Typologie des données et initialisation uniforme (C++11)
- Inférence automatique des types avec le spécificateur auto (C++11)
- Les structures de contrôle de flux (if, switch, for, while) et leurs améliorations (C++17/20)
- Les énumérations et leurs nouveautés (C++11/20)
- Fonctions, paramètres et spécificateurs
- Les fonctions lambda (C++11/14/17)
- Fonctions inline et fonctions constexpr (C++11 à C++20)
- Les tableaux et la classe std::array (C++11)
- Introduction à la bibliothèque standard
- Les modèles mémoire (Data segment, Heap, Stack)
- Constantes, pointeurs et références.
- La classe std::string
- Allocation et désallocation dynamiques
- Allocation et désallocation dynamiques Les espaces de noms (namespaces)
- Organisation du code au sein d'un projet (fichiers d'entête et fichiers d'implémentation)

Déroulé de l'action

- **Modalités**

  Session Inter/Intra
  En présentiel/Classe virtuelle

- **Horaires**

  9H00-12H30 /13H30-17H00

- **Méthode pédagogique**

  Alternance exposés théoriques et exercices pratiques (80% de pratique)

- **Suivi et assistance**

  Support de cours adapté au logiciel étudié et au niveau suivi
  Assistance téléphonique gratuite et illimitée

- **Modalité d'évaluation**

  Passage de la certification TOSA en fin de formation
  Attestation de stage
  Emargement quotidien d'une feuille de présence

- **Accessibilité aux personnes handicapées**

  Pour tout besoin d'adaptation, retrouver le contact de notre référent handicap et les modalités d'accueil sur la page : Infos pratiques/Situation de Handicap

775 route de l'aérodrome
ZA Agroparc
84000 Avignon
Tél 04 90 85 98 12
Fax 04 90 86 26 24
info@cit.fr
www.cit.fr

LANGAGE C++ - Les fondamentaux

Les concepts objet appliqués au C++

- Classes, instances, cycle de vie et cinématique d'un objet
- Éléments de notation UML (modélisation statique et dynamique)
- Encapsulation et visibilité (public-private-protected)
- Méthodes, constructeurs, destructeurs
- La zone d'initialisation des membres (ZIM)
- Fonctions et classes amies (friendship)
- Surcharge (overloading) des méthodes et opérateurs
- Membres de classe (ou statiques)
- Mise en oeuvre des relations (agrégation / composition)
- Robustesse via les spécificateurs de déclaration (default, delete, override, final) de C++11
- Héritage, classes abstraites et concrètes
- Polymorphisme et interfaces
- Destructeur virtuel
- Héritage privé et protégé
- Clonage d'objets (construction par recopie / opérateur d'affectation)

Robustesse dynamique et traitement des exceptions

- Prise en compte des erreurs avec les assertions et les exceptions
- Lancement (throw), propagation et interception d'une exception via un gestionnaire (bloc try/catch)
- Conception d'une classe d'exception personnalisée
- Contrôle dynamique d'une exception avec la clause noexcept de C++11

Points clés de la Standard Template Library (STL)

- Présentation des principaux conteneurs (vector, list, set, map, deque) et des critères de choix
- Insertion efficace via l'emplacement (C++11)
- Les itérateurs
- Les algorithmes génériques et leurs nouveautés (C++17/20)

Généricité - Templates

- Définitions de patrons de fonctions et de classes génériques, syntaxe et instanciation
- Spécialisation générique
- Polymorphisme statique versus polymorphisme dynamique
- Introduction à la métaprogrammation

Qualité logicielle et tests d'applications

- Les best practices en C++
- Idiomes et règles de conception
- Introduction aux design patterns
- Le développement guidé par les tests (TDD – Test Driven Development)
- Présentation succincte de GoogleTest

Travaux pratiques

- **Pour chaque point du programme abordé :**
  Présentation et démonstration par le formateur, mise en pratique sur les fonctionnalités abordées par le stagiaire avec l'appui du formateur et du groupe, feedbacks du formateur tout au long de l'activité.

Déroulé de l'action

- **Modalités**
  Session Inter/Intra
  En présentiel/Classe virtuelle
- **Horaires**
  9H00-12H30 /13H30-17H00
- **Méthode pédagogique**
  Alternance exposés théoriques et exercices pratiques (80% de pratique)
- **Suivi et assistance**
  Support de cours adapté au logiciel étudié et au niveau suivi
  Assistance téléphonique gratuite et illimitée
- **Modalité d'évaluation**
  Passage de la certification TOSA en fin de formation
  Attestation de stage
  Emargement quotidien d'une feuille de présence
- **Accessibilité aux personnes handicapées**
  Pour tout besoin d'adaptation, retrouver le contact de notre référent handicap et les modalités d'accueil sur la page : Infos pratiques/Situation de Handicap