# C++ Programming Language Certified

## Objectifs

- Utiliser les aspects fortement typés du langage
- Concevoir des classes en respectant les bonnes pratiques
- Surcharger les opérateurs du langage
- Construire des applications en utilisant l'héritage et de polymorphisme
- Manipuler les différentes collections
- Gérer les cas d'erreurs à l'aide des exceptions
- Déboguer vos programmes
- Utiliser les principales collections de la stl
- Intégrer dans vos développements les nouveautés de la norme C++ 11

## Prérequis

- Avoir suivi la formation Programmation en C, ou posséder les connaissances et compétences équivalentes
- Avoir une expérience de développement en langage C. Les aspects de base du langage (modularité, codage d'une fonction, les pointeurs, …) seront considérés comme acquis, car largement exploités durant la formation
- Idéalement, avoir suivi la formation Conception et programmation objet, ou posséder les connaissances et compétences équivalentes

## Programme

Les extensions du langage

- L'historique et les versions du langage C++
- Un rappel sur le langage C
- Le passage de paramètres par référence
- La surcharge des fonctions
- Les valeurs de paramètres par défaut
- La nouvelle librairie d'entrées/sorties «iostream»
- La gestion de la mémoire dynamique
- Travaux pratiques :
    - Réaliser un programme qui permet de trouver un nombre tiré de façon aléatoire en utilisant les extensions du langage

Les classes

- La conception de classes (attributs et méthodes)
- L'instanciation d'objets
- L'encapsulation (les attributs d'accès private et public)
- Les accesseurs
- Les méthodes « inline »
- Les constructeurs et les destructeurs
- Les attributs et les méthodes de classes
- La surcharge d'opérateurs

- Les relations d'associations
- Travaux pratiques :
    - Réaliser une classe représentant un compte bancaire et tester son bon fonctionnement

L'héritage et le polymorphisme

- La conception d'une classe dérivée
- Les constructeurs et les destructeurs
- L'attribut protected
- La redéfinition de méthodes
- Les méthodes virtuelles
- Le destructeur virtuel
- L'héritage multiple
- Les classes abstraites
- Travaux pratiques :
    - Réaliser une classe CompteEpargne par héritage de la classe Compte

La gestion des exceptions

- Un rappel sur la gestion des erreurs en C
- La levée d'exception
- Le traitement d'exception avec le bloc try … catch
- La spécification d'exception
- La création de classe exception
- Travaux pratiques :
    - Mettre en place des traitements d'erreurs sur les méthodes de la classe Compte à l'aide des exceptions

Les templates et la stl

- Les objectifs et les intérêts de la généricité
- La conception et l'utilisation de classes génériques
- La présentation de la stl
- La classe String
- La manipulation de vecteurs
- Les autres collections de la stl
- Travaux pratiques :
    - Réaliser une classe Banque qui gère un ensemble de classes en utilisant les collections de la stl

Les principaux apports de C++ 11

- Les inférences de type (auto)
- Initialisation d'un pointeur nul : le nullptr
- Les tableaux à taille fixe
- Notion d'intervalle pour les boucles
- Les énumérations fortement typées
- Les directives de constructeurs default et delete
- Les qualificatifs de méthodes override, final

- Présentation de la RAI et des pointeurs intelligents
- Travaux pratiques :
    - Evolution de l'application bancaire en intégrant les différentes nouveautés