

## Objectifs

Analyser efficacement un code pour l'optimiser et le déboguer. Acquérir les meilleures méthodes pour une programmation robuste et fiable de vos programmes en C++.

## Participants

Toute personne ayant suivi le stage fonctions de base ou maîtriser l'ensemble des concepts.

## Prérequis

Connaître les principes de base de l'algorithme Disposer de connaissances du langage C++ et la programmation orientée objet

## Pédagogie

La pédagogie est basée sur le principe de la dynamique de groupe avec alternance d'apports théoriques, de phases de réflexion collectives et individuelles, d'exercices, d'études de cas et de mises en situations observées. Formation / Action participative et interactive : les participants sont acteurs de leur formation notamment lors des mises en situation car ils s'appuient sur leurs connaissances, les expériences et mettront en oeuvre les nouveaux outils présentés au cours de la session.

## Profil de l'intervenant

Consultant-formateur expert sur cette thématique. Suivi des compétences techniques et pédagogiques assurée par nos services.

## Moyens techniques

Encadrement complet des stagiaires durant la formation. Espace d'accueil, configuration technique des salles et matériel pédagogique dédié pour les formations en centre. Remise d'une documentation pédagogique papier ou numérique à échéance de la formation.

## Méthodes d'évaluation des acquis

Exercices individuels et collectifs durant la formation. Evaluation des acquis et attestation de fin de stage adressés avec la facture.

## Programme

### Rappels sur le C++

Typage

Héritage

Polymorphisme

Compilation modulaire

Abstraction

Classes abstraites

Constructeur

Constructeur de copie

Destructeur

Surcharge d'opérateur

Références et pointeurs

Pointeurs de pointeurs

## Langage C++ - Perfectionnement

2 jours - 14 heures



adhara France

Code formation : Log-187

[www.adhara.fr](http://www.adhara.fr)

Pièges liés à l'utilisation de pointeurs

Initialisation des pointeurs

Compilation

Allocation de mémoire dynamique