

Plan de cours N° : 614

Durée : 5 jours (35h)

Participants

Tous développeurs
désireux d'apprendre le
langage C++ ISO 2014.

Pré-Requis

Connaître le C

Méthode pédagogique

Alternance entre apports
théoriques et exercices
pratiques

Les moyens d'encadrement mis en oeuvre

1 à 8 personnes maximum par session
1 poste informatique par personne
Une assistance post-formation, d'une
durée d'un an, sur le contenu de la
formation

Les moyens permettant de suivre son exécution et d'en apprécier les résultats

Emergence par demi-journée
Evaluation des acquis par mise en
situation de travail
Evaluation qualitative de fin de stage
Remise d'une attestation individuelle
de formation en fin de stage

Assistance

formateurs@atp-formation.com

Objectifs

Comprendre les concepts fondamentaux de la programmation orientée objets
avec le langage C++ ISO 2014

Programme :

- **Introduction**
 - Historique et version du standard C++ ISO
 - Mise en oeuvre de l'environnement de développement
- **Extensions procédurales du langage**
 - Le fameux HelloWorld en C++
 - Une nouvelle librairie d'entrées/sorties, Une nouvelle forme de commentaires
 - Un contrôle de typage renforcé en rapport à C
 - Principe de localité
 - Gestion dynamique de la mémoire
 - Mode de passage de paramètres par référence
 - Définition de fonctions inlines
 - Surcharge et valeurs de paramètres par défaut
 - valeurs de paramètres par défaut, Surcharge de fonctions, Surcharge d'opérateurs
 - Définition de fonctions membres d'une structure
- **Introduction aux concepts d'objets et de classes**
 - Quelques points de vocabulaires
 - Définitions de classes et instanciations d'objets
 - Définitions de membres à l'intérieur d'une classe
 - Définitions d'attributs de classes, Définitions de méthodes, Comparatifs entre méthodes et fonctions
 - Concept d'encapsulation
 - Définition de méthodes d'accès aux attributs, Niveaux de visibilité des membres d'une classe, Intérêts de l'encapsulation
 - Concept de composition
 - Les constructeurs et les destructeurs
 - Le constructeur par défaut, Le constructeur par copie, Le destructeur
 - Comment bien prototyper nos méthodes
 - Définition d'opérateurs membres d'une classe
 - Les nouveautés C++ 2011 relatives à la mise en oeuvre de classes
- **La généricité (Les templates)**
 - Buts et intérêts de la généricité (factorisation de code)
 - Les fonctions génériques
 - Aspects syntaxiques, Où localiser le code d'une fonction générique, Pièges à éviter
 - Définitions de classes génériques
 - Aspects syntaxiques, Implémentation des méthodes génériques, Recopie d'objets contenant des pointeurs
 - Les nouveautés C++ 2011 relatives aux templates

Plan de cours N° : 614

- **Introduction au concept d'héritage**
 - Pourquoi mettre en oeuvre l'héritage ?
 - Aspects syntaxiques
 - Constructions et destructions d'objets basés sur l'héritage
 - Cycle de vie d'un objet, Cascade des appels de constructeur, Cascade des appels de destructeur
 - L'héritage induit le polymorphisme
 - Modes de liaisons d'appels de méthodes
 - La liaison statique (early binding), La liaison dynamique (late binding),
 - Fonctionnement interne de la liaison dynamique, Le piège des destructeurs
 - Les nouveautés C++ 2011 relatives à l'héritage
- **Gestion des exceptions**
 - Rappels sur la gestion des erreurs en C
 - Valeurs de retour identifiant le problème survenu, Utilisation de l'entête <errno.h>, Limites de ce modèle
 - Qu'est qu'une exception ?
 - Types d'exceptions, Définition d'une classe d'exception, Les classes d'exceptions ANSI
 - Déclenchement d'exceptions dans un système informatique
 - Modes de traitement d'une exception
 - Capturer l'exception - instruction try ... catch
 - Relayer l'exception à la méthode appelante
 - Cas subtiles d'utilisations des exceptions
- **Gestions des entrées/sorties sur fichiers**
 - Hiérarchie des classes d'entrées/sorties ANSI
 - L'entête <fstream>
 - La classe ifstream pour les accès en lecture, La classe ofstream pour les accès en écriture