

Conçu dans les années soixante-dix, le langage C reste encore de nos jours l'un des piliers de la programmation dans de nombreux domaines : industriel, scientifique, réseau, bas-niveau, etc.

Puissant, efficace, épuré, le C est à l'origine de nombreux langages dérivés (C++, java, C#...), et une bonne connaissance de ce langage est un atout réel pour la maîtrise du développement logiciel.

Organisation

Durée : 4 jours (28 heures).

Pré-requis : le stagiaire doit avoir des notions générales de programmation.

Conseil cursus : en préambule à cette formation, nous vous conseillons notre formation « Introduction à la Programmation ».

Thèmes abordés

- Concepts du langage C : présentation, avantages et inconvénients, utilisations typiques.
- Outils de développement : environnement, compilateur, débogueur.
- Structures des programmes : exemple, vocabulaire, fonctions et variables, constantes, types et expressions.
- Les variables du C : globales et locales, détail des types scalaires, tableaux.
- Les fonctions : passage d'arguments, types de retour, passage par valeur, passage par référence.
- Les pointeurs : déclarations, pièges, arithmétique des pointeurs, initialisation et utilisation.
- Structures de contrôle : tests et itérations, sélections.
- Chaînes de caractères : pointeurs et tableaux de caractères, fonctions standard de manipulation de chaînes.
- Expressions du C : opérateurs mathématiques, logiques et binaires.
- Structures de données : principes et déclaration, utilisation, tableaux de structures, pointeurs de fonction, programmation objet.
- Allocation dynamique de mémoire : allocation et libération, organisation des données.

Travaux pratiques

Les travaux pratiques de ce cours peuvent se dérouler, au choix du stagiaire, sur système Linux avec Eclipse-CDT ou sur système Windows avec Visual Studio C++.

Plan détaillé au verso ►

Plan détaillé

Concepts du langage C

Présentation

Premières versions du langage C, évolutions, standards et normes

Avantages et inconvénients

Efficacité et performance, proximité du matériel et du processeur, portabilité, laxisme du langage, exemples de bogues courants et des conséquences désastreuses.

Utilisations typiques

Systèmes industriels et embarqués, serveurs et réseaux, télécommunication, gestion de données.

Outils de développement

Environnements

Présentation de l'environnement de développement, prise en main d'Eclipse sous Linux ou Visual C++ sous Windows

Compilation

Création d'un projet et compilation d'un premier programme, exécution et résultat. Rôle de la bibliothèque C.

Débogage

Exécution du programme en pas à pas, points d'arrêts.

Structures des programmes

Aspect et vocabulaire du C

Présentation de code, indentation, caractères spéciaux, mots-clés, définition et déclaration de fonctions, fichier d'entête, fonctions de bibliothèque.

Fonctions

Ecriture des fonctions, différences entre déclaration et définition, paramètres.

Variables

Types des variables, déclaration et utilisation, expression et constantes.

Les variables du C

Portée et persistance

Stockage des variables en pile, variables globales et variables locales, variables automatiques et statiques.

Les types scalaires

Format et représentation interne, utilisation.

Les tableaux

Tableaux d'entiers, déclaration, initialisation et utilisation.

Les fonctions

Passage d'arguments

Utilisation de la pile, types des paramètres, modification des paramètres formels, passage par valeur, réservation de l'espace dans la pile.

Types de retour

Utilisation des valeurs de retour des fonctions. Limitation à une valeur.

Passage d'argument par référence

Nécessité de passer un pointeur dans la pile, modification des paramètres d'appel.

Les pointeurs

Déclarations des pointeurs

Exemple de déclaration des types, affichage des valeurs des pointeurs et des contenus pointés. Risques liés à l'utilisation des pointeurs.

Arithmétique des pointeurs

Concepts, vérification des pas d'incrémentations en fonction des types de données.

Structures de contrôle

Structures de test

Structure if / else. Imbrications et indentation. Expression testée. Confusion entre égalité et affectation.

Structure de sélection

Construction switch / case. Utilisation du break.

Itération conditionnelle

Boucles while et until. Boucle infinie. Rupture de séquence.

Enumération

Boucle for. Syntaxe usuelle et variantes. Rupture de séquence.

Chaînes de caractères

Principes des chaînes en C

Pointeurs et tableaux de caractères, représentation, caractère nul.

Fonctions standards de la bibliothèque C

Longueur, copie, concaténation, recherche de sous-chaîne.

Expressions du C

Opérateurs arithmétique

Opérateurs classiques, notations condensées, pré- ou post- incrémentation et décrémentation.

Opérateurs logiques

Valeurs de vérité, composition d'expressions booléennes

Opérateurs binaires

ET, OU, OU exclusif, négation, décalage, masques

Structures de données

Principes

Utilités des structures, déclaration, et utilisation des structures

Tableaux

Tableaux et pointeurs de structures, initialisation et utilisation.

Programmation objet

Pointeurs de fonction, encapsulation, attributs et méthodes, conception orientée objet.

Allocation dynamique de mémoire

Principes

Utilité des allocations dynamiques, fonctions d'allocation et de libération.

Fuites mémoire

Risques des allocations, règles de bonne conduite, outils de débogage

Structures de données dynamique

Implémentation des listes chaînées et doublement chaînées, utilisation.

Conclusion

Discussions et expérimentations libres sur l'ensemble des thèmes abordés.