



Your Gateway to Excellence

Formation Programmation en C

Description de la formation Langage C

Inventé au début des années 1970 pour réécrire UNIX, le langage C est un langage de programmation toujours beaucoup utilisé aujourd'hui. On peut le considérer comme un langage de haut niveau "classique" car il en propose tous les mécanismes standards (boucle, test, fonctions, types de base, etc.) et à ce titre il est plutôt facile à apprendre. Mais il a des caractéristiques bas-niveau également car il permet une gestion fine de la mémoire et des variables (au bit près), issue de ses origines, et qui lui confère une place de choix dans le domaine de l'embarqué (système dédié, IoT, drivers, etc.). Par bien des aspects il est également extrêmement pédagogique sur les bases de fonctionnement d'une application en mémoire (cycle de vie et localisation des variables en pile mémoire, code retour des fonctions, etc.) et donne donc implicitement de très bonnes bases en sécurité du développement (débordement de pile, injection de caractères, etc.).

De nombreux langages plus modernes comme C++, C#, Java et PHP ou encore Javascript, ont repris une syntaxe similaire au langage C. Ils reprennent en partie sa logique et proposent tous d'utiliser des bibliothèques écrites en C.

Objectifs

Objectifs Opérationnels:

Savoir écrire des programmes robustes et portables en langage C

Objectifs Pédagogiques:

Cette formation Langage C vous apportera les connaissances et compétences nécessaires pour:

- ✓ Expliquer l'allure général d'une application écrite en C
- ✓ Savoir programmer en C (opérateurs, expressions, tests, boucle, ...)
- ✓ Utiliser les pointeurs avec aisance
- ✓ Maîtriser l'utilisation des chaînes de caractères
- ✓ Manipuler les tableaux avec aisance

Capital Social: 50000 DT **MF:** 1425253/M/A/M/000 **RC:** B91211472015

Tél. / Fax.: +216 73362 100 **Email:** contact@itgate-training.com **Web:** www.itgate-training.com

Adresse : 12 Rue Abdelkadeur Daghrrir - Hammam Sousse 4011 – Tunisie

- ✓ Ecrire ses propres fonctions
- ✓ Manipuler les structures de données avec aisance
- ✓ Savoir gérer les flux d'entrées et de sorties
- ✓ Ecrire un programme C modulaire (écriture et utilisation de bibliothèques systèmes ou métiers, fichier header, makefile)

À qui s'adresse cette formation ?

Public :

Les ingénieurs et les informaticiens qui souhaitent acquérir une formation complète et opérationnelle sur le langage C.

Prérequis :

Les participants doivent avoir des connaissances générales en informatique et maîtriser au moins Contenu du cours Langage C un langage de programmation.

Contenu du cours Langage C

Première approche pratique

Organisation d'un programme

- ✓ Composants de base du langage
- ✓ Types et constantes de base
- ✓ Stockage et passage de paramètres
- ✓ Variables globales et locales
- ✓ Arguments de fonction : passage par valeur
- ✓ Code de retour des fonctions
- ✓ Opérateurs d'adressage
- ✓ Passage d'arguments par adresse
- ✓ Entrées/sorties formatées
- ✓ Utilisation élémentaire de la chaîne de production

Mécanismes

Opérateurs, expressions numériques et opérations sur les nombres

- ✓ Opérateurs arithmétiques
- ✓ Mécanismes d'évaluation des expressions
- ✓ Post et pré-incrémentation de décrémentation
- ✓ Précédence et associativité des opérateurs

Opérateurs de base et expressions logiques

- ✓ Instruction d'affectation
- ✓ Instruction while
- ✓ Mécanismes de fonctionnement des expressions logiques
- ✓ Opérateurs de comparaison
- ✓ Gestion du flux d'entrée/sortie standard
- ✓ Prototype de filtre
- ✓ Instruction if
- ✓ Opérateurs logiques

Opérations numériques : rappels et compléments

- ✓ Les types numériques composés. Règle de conversion dans les expressions mixtes
- ✓ Initialisation des variables
- ✓ Arithmétique sur les adresses
- ✓ Formats d'entrée/sortie associés aux types numériques

Tableaux, pointeurs et chaînes de caractères

- ✓ Définition et utilisation classique des tableaux
- ✓ Equivalences pointeurs/tableaux
- ✓ Chaînes de caractères
- ✓ Initialisation des tableaux
- ✓ Exemples de manipulation de chaînes de caractères
- ✓ Rappels et compléments sur for, while et l'opérateur ", "

Compilation séparée, classe d'allocation

- ✓ Mécanisme de fonctionnement de la chaîne de production
- ✓ Utilisation de bibliothèque de sources
- ✓ Différents cas de figure de la compilation séparée
- ✓ Notion d'objet externe
- ✓ Cas des données globales et statiques
- ✓ Cas des données locales
- ✓ Règle de visibilité
- ✓ Cas des fonctions
- ✓ Compléments sur les fonctions et les initialisations

Opérateurs travaillant au niveau du bit

- ✓ ET, OU, OU exclusif, complément à 1
- ✓ Décalages

Compléments

Compléments sur les opérateurs et les instructions

- ✓ Opérateur ternaire d'alternative
- ✓ L'instruction switch
- ✓ L'instruction do while
- ✓ Instructions associées aux boucles

Les structures

- ✓ Généralités
- ✓ Les structures emboîtées
- ✓ Les structures chaînées
- ✓ Les champs de bits
- ✓ Les unions
- ✓ Les énumérations

Le macro processeur, la portabilité

- ✓ Les directives de compilation : constantes symboliques, macros, inclusions, compilation conditionnelle
- ✓ Construction de surtypes : typedef
- ✓ Conversion explicite de types : casting
- ✓ Maîtrise des types complexes

Les entrées/sorties de la bibliothèque

- ✓ Généralités
- ✓ Ouverture et fermeture de fichiers
- ✓ Entrées et sorties : caractères, lignes, enregistrements
- ✓ Outils généraux
- ✓ Accès direct

Gestion des entrées/sorties par les appels système

- ✓ Généralités
- ✓ Opérations classiques
- ✓ Accès direct
- ✓ Destruction
- ✓ Étude de cas

Outils de base associés à la chaîne de production

- ✓ Philosophie et exemples de makefile
- ✓ Principes de la gestion de version (exemples avec Git)

Travaux Pratiques

Des machines sous-système Unix ou Windows (PC) seront mises à la disposition des participants de manière à mettre en pratique les notions présentées.