

Stage pratique de 5 jour(s)
Réf : LGC

Participants

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

Pré-requis

Connaissances de bases en programmation.

Prix 2012 : 2315€ HT

Dates des sessions

Paris

13 fév. 2012, 12 mar. 2012
16 avr. 2012, 21 mai 2012
18 juin 2012, 9 juil. 2012
20 août. 2012, 17 sep. 2012
15 oct. 2012, 12 nov. 2012
10 déc. 2012

Aix

16 avr. 2012, 11 juin 2012
10 sep. 2012, 19 nov. 2012

Bordeaux

2 avr. 2012, 4 juin 2012
8 oct. 2012, 3 déc. 2012

Geneve

2 avr. 2012, 4 juin 2012
8 oct. 2012, 3 déc. 2012

Lille

2 avr. 2012, 4 juin 2012
8 oct. 2012, 3 déc. 2012

Luxembourg

2 avr. 2012, 4 juin 2012
8 oct. 2012, 3 déc. 2012

Lyon

16 avr. 2012, 11 juin 2012
10 sep. 2012, 19 nov. 2012

Nantes

23 avr. 2012, 18 juin 2012
24 sep. 2012, 26 nov. 2012

Rennes

23 avr. 2012, 18 juin 2012
24 sep. 2012, 26 nov. 2012

Sophia-antipolis

16 avr. 2012, 11 juin 2012
10 sep. 2012, 19 nov. 2012

Strasbourg

2 avr. 2012, 4 juin 2012
8 oct. 2012, 3 déc. 2012

Toulouse

2 avr. 2012, 4 juin 2012
8 oct. 2012, 3 déc. 2012

Programmation en C

OBJECTIFS

Ce stage intensif vous permettra d'acquérir une connaissance réellement opérationnelle du langage. Il vous expliquera le fonctionnement des différents mécanismes et vous montrera leur mise en œuvre grâce à de nombreux exercices pratiques. A l'issue de ce stage, vous serez en mesure d'écrire des programmes C robustes et portables.

1) Première approche pratique

2) Mécanismes

3) Compléments

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.

1) 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.

2) 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.

3) 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 macroprocesseur, 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.
- Etude de cas.

Outils associés à la chaîne de production