

# Programmation avancée en C

3 j (21 heures)

Ref : LACP

## Public

Ingénieurs, Analystes programmeurs désirant se perfectionner au langage C

## Pré-requis

Avoir suivi la formation "Langage C" ou posséder les connaissances équivalentes

## Moyens pédagogiques

Formation réalisée en présentiel ou à distance selon la formule retenue  
Présentation des concepts, discussion technique, démonstrations, exercices simples et TP  
Un poste informatique par stagiaire connecté à internet, à une imprimante en réseau et au réseau informatique  
Les salles sont équipées d'un tableau interactif ou d'un vidéoprojecteur et d'un paperboard  
Support de cours fourni à chaque stagiaire

## Modalités de suivi et d'évaluation

Feuille de présence émargée par demi-journée par les stagiaires et le formateur  
Exercices de mise en pratique ou quiz de connaissances tout au long de la formation permettant de mesurer la progression des stagiaires  
Questionnaire d'évaluation de la satisfaction en fin de stage  
Auto-évaluation des acquis de la formation par les stagiaires  
Attestation de fin de formation

## Objectifs

- Connaître les fonctions appartenant aux principales bibliothèques standards du langage C
- Approfondir les concepts de gestion de mémoire
- Utiliser les tableaux et les listes chaînées
- Apprendre à travailler en utilisant les bibliothèques dynamiques
- Compiler sous Windows et linux

## Programme détaillé

### RAPPEL DE BASE

---

- Architecture d'un programme
- Types de données

Structures de contrôle

Fonctions et visibilité

Pointeurs et allocation dynamique

## **POINTEURS**

---

Pointeurs de fonction

Pointeurs de tableau

Indirections à plusieurs niveaux

## **FONCTIONS**

---

Récurtivité

Nombre de paramètres variables

## **APPELS SYSTEME**

---

Processus et fork()

Lecture d'un répertoire

Le réseau (sockets)

## **GESTION DE LA MEMOIRE**

---

Tableaux / listes chaînées

Gestion du tas

## **DECLARATIONS SPECIALES**

---

Variables externes

Fonctions exportées et importées de DLL

Variables statiques

Fonctions inline

## **CONSTITUTION ET APPEL DE FONCTIONS ASSEMBLEUR**

---

Faut-il encore utiliser l'assembleur ?

Mise en œuvre

Edition des liens

## **OPTIMISATION**

---

De la vitesse

De la mémoire

## **PROFILER**

---

Objectif du profiling

Mise en oeuvre

Interpréter le résultat

## **PORTABILITE**

---

Standard ANSI C

Standard POSIX

Unix / Windows

Bibliothèques partagées (SO, DLL)

---