



Formation C & C++, programmer ses applications en Multicore

Vous découvrirez les architectures Multicore et leur programmation, les techniques de mise en œuvre d'une approche multithread ou multiprocesseurs et les langages dédiés à la programmation parallèle. Vous étudierez également les contraintes en matière de synchronisation d'accès aux données et les précautions à prendre.

Cette formation ne nécessite pas de prérequis.

Public et pré-requis

Objectifs

- Maîtriser les enjeux de la programmation Multicore
- Concevoir et développer des applications à base de threads et de processus
- Maîtriser les modèles de programmation parallèle et les bibliothèques disponibles
- Déboguer et profiler des applications Multicore

Moyens pédagogiques

- Présentation du formateur et du programme.
- Présentation et écoute de chacun de stagiaires.
- Apports didactiques pour apporter des connaissances communes.
- Mises en situation de réflexion sur le thème du stage et des cas concrets.
- Méthodologie d'apprentissage interactive et participative.
- Exercices et études de cas concrets.
- Temps d'échanges.
- Accompagnement pédagogique individualisé.

Formateur

Les formateurs de CROSSTHINK sont des experts de leur domaine, disposant d'une expérience terrain qu'ils enrichissent continuellement. Leurs connaissances techniques et pédagogiques sont rigoureusement validées en amont en interne.

Suivi de l'exécution et évaluation des résultats

- Accueil des stagiaires dans une salle dédiée à la formation / A distance
- Feuilles de présence.
- Documents supports de formation projetés.
- Mise à disposition du stagiaire des documents et supports de formation.
- Tout au long et/ou à l'issue de la formation : Evaluation des acquis des stagiaires via des exercices, des QCM, des QUIZZ, des mises en situation et/ou des cas pratiques.
- Enquête de satisfaction.
- Attestation de fin de formation.

Vous découvrirez les architectures Multicore et leur programmation, les techniques de mise en œuvre d'une approche multithread ou multiprocesseurs et les langages dédiés à la programmation parallèle. Vous étudierez également les contraintes en matière de synchronisation d'accès aux données et les précautions à prendre.

EN BREF

3 jours

1490€

CONTACT

contact@crossthink.fr

04.83.43.47.48

PROCHAINES DATES

[Nous consulter](#)



Formation C & C++, programmer ses applications en Multicore

Programme

Introduction

- Enjeux de la programmation Multicore.
- Tableau des technologies utilisables : processus, thread et parallélisme.
- Description du fonctionnement d'un processeur.
- Architecture en "Hyperthreading".
- Architectures des processeurs INTEL et AMD.
- Architectures NVidia et API.
- Architecture en mémoire partagée vs mémoire distribuée.

Modélisation des applications

- Importance des aspects modélisation.
- Patterns de mise en parallèle des traitements.
- Utilisation des mécanismes asynchrones.
- Développer une nouvelle application : précautions et modélisation. Eviter les "singletons".
- Modifier une application existante en Multicore.
- Choix d'architecture : un compromis synchronisation et performance. Choix multiprocessus/multithreads.

Threads

- Apport des threads dans une application industrielle.
- Ordonnement des threads.
- Gestion des stacks et "call stack" dans les threads.
- Débogueurs multithreads.
- Gestion des objets de synchronisation : sections critiques, Mutex et Sémaphores.
- Développer "thread safe".
- API de threads TBB, Clik++, C++11, boost threads, pthreads.

Travaux pratiques

Threads et synchronisation en C/C++.

Processus

- Espaces d'adressage des processus, organisation.
- Critères de choix d'une approche multiprocessus.
- Techniques de communication interprocessus (IPC).
- Outils de debugging multiprocessus.
- Avantage et inconvénients des techniques multiprocessus.

Travaux pratiques

Gestion de traitements asynchrones avec l'API C/C++.

La programmation parallèle

- Apport et objectifs de la programmation parallèle.
- La librairie "OpenMP" C++ (programmation mémoire partagée).
- La librairie "OpenMPI" (programmation mémoire distribuée).
- Utiliser les GPU des cartes graphiques pour le calcul.
- Kits de NVidia (CUDA) et ATI.
- La librairie "OpenAcc" pour la programmation GPU.
- La librairie "OpenCL" pour la programmation parallèle CPU et GPU.

Travaux pratiques

Paralléliser des algorithmes avec "OpenMP" en C++. Utilisation de l'API OpenCL.

Synthèse et conclusion

- Conclusion des techniques étudiées.
- Avenir du C++ avec le multicore.



CROSSTHINK

Formation C & C++, programmer ses applications en Multicore

Certification