

PRG303 : Advanced C++ Programmeren

Code : PRG303 Duur : 3 dagen

Categorie : Programming

Doelgroep :

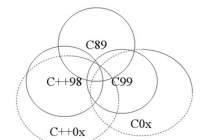
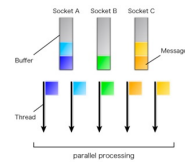
Deze cursus is bedoeld voor developers die bekend zijn met C++ maar die zich willen verdiepen in de meer geavanceerde technieken en mogelijkheden van de taal.

Voorkennis :

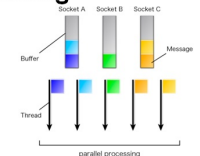
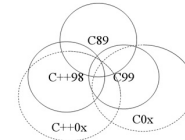
Om aan deze cursus deel te nemen is kennis van de basis concepten van C++ en uitgebreide ervaring met programmeren in C++ vereist. De deelnemers moeten vertrouwd zijn met onderwerpen als operator overloading, templates, virtual functions en polymorfisme.

Uitvoering :

De theorie wordt behandeld aan de hand van presentatie slides. Illustratieve demo's worden gebruikt om de behandelde concepten te verhelderen. Er is een sterke afwisseling van theorie en praktijk en ruime gelegenheid om te oefenen.



Advanced C++ Programming



Inhoud :

In de cursus Advanced C++ komen geavanceerde aspecten van de C++ taal aan de orde. De algemene structuur van de STL en het gebruik ervan in de praktijk wordt behandeld. Er wordt aandacht besteed aan de details van containers, iterators en algoritmen. Bovendien komen geavanceerde STL features zoals function objects, binders, comparators en adapters aan de orde. De cursus behandelt ook de meer geavanceerde mogelijkheden van C++ zoals de overloading van operators als \rightarrow , \rightarrow^* en $()$, member templates en pointers naar member functies. Daarnaast wordt in detail gekeken naar de interne werking van smart pointers, een techniek die het mogelijk maakt om objecten te alloceren zonder ze expliciet op te moeten ruimen. De combinatie van smart pointers met STL en het gebruik van hash tables en function objects wordt ook belicht bij de behandeling van de geavanceerde aspecten van STL.

Module 1 : Templates Review

- Templates review
- Template specializations and partial specializations
- Non-type template parameters
- Default template parameters
- Template member functions

Module 2 : STL Intro

- Introduction to the STL
- Purpose and structure of the STL
- Containers
- Algorithms
- Iterators
- Container classification
- Characteristics and content

Module 3 : Iterators and Algorithms

- Iterator classification and behavior
- Generic algorithms and iterators
- Design of generic algorithms
- Reverse iterators
- iostream iterators

Module 4 : STL Containers

- Vectors
- Maps
- Hash Maps
- Strings Bitsets
- Function objects
- Predicates, and comparators

Module 5 : Adapters

- Container adapters
- Iterator adapters
- Function adapters
- Iterator traits and tags
- Adapting existing code to STL

Module 6 : Operator Overloading

- Advanced Operator Overloading
- Functors Or Function Objects
- Pointers To Member Functions
- Member Templates
- Reference Counting

Module 7 : Smart Pointers

- Smart Pointers
- Smart Pointers and STL