Moderne C++-Programmierung («CPROGR»)

C++ has changed a lot in recent years and has become much more powerful. New language features allow an easier and therefore safer development. These advantages are used more and more in the embedded environment. Get fit with this training!

Duration: 4 days
Price: 3'200.–
Course documents: Digicomp Courseware

Content

The course consists of the following modules. The weighting and selection of the modules is adapted to the knowledge and preferences of the participants.

1. Basic features
   - Namespace
   - References
   - Overloading functions
2. Important standard classes
   - Use of streams
   - String
   - Vector
3. Classes
   - Class definition and instantiation
   - Attributes (data elements)
   - Accessing class elements
   - Methods (element functions)
   - Information hiding
   - Constructor
4. Modules and interfaces
   - Declaration and definition
   - Inline functions
   - Static class elements
5. Class derivation, inheritance
   - Inheritance and visibility of elements
   - Initialization of base classes
   - Inheritance and polymorphism
6. Dynamic memory management
   - New operator
   - Delete operator
   - Using Dynamic Instances
7. Normal form for classes
   - Copy constructor
   - Deep copy
   - Prevent copying
   - Destructor
   - Assignment operator
   - Move, Rvalue references
8. Smart pointer
   - Unique_ptr
   - Shared_ptr
9. Virtual methods

    - Static binding
    - Dynamic binding
    - Virtual methods
    - Abstract classes
10. Input and output to files
11. Templates
    - Define templates
    - Applying templates
    - Function templates
12. The Standard Template Library (STL)
13. Container
    - Vector, deque
    - List
    - Set
    - Map
14. Functions and function objects
    - Lambda expression
    - Predicate, equality and equivalence
15. Algorithms
16. Exceptions
    - Throw / try / catch
    - Exception hierarchies

Key Learnings

- Using class libraries in your programs
- Understanding and using scopes and namespaces
- Using the string class to handle character strings
- Defining classes with their attributes and methods
- Understanding classes as a tool for information hiding
- Implementing and inserting constructors
- Practical modularization of code
- Defining a class hierarchy
- Knowledge of the different visibility of class elements in relation to derivation
- Users of the operators new and delete
- Creating a program with dynamic memory management
- Knowing the advantages of smart pointers
- Understanding unique_ptr and shared_pr
- Defining virtual methods for dynamic binding
- Defining interfaces with purely virtual functions and abstract classes
- Access files with streams
- Basic knowledge of templates
- Correct application of the containers vector, deque, list, set map, unordered_set
- Efficient searching and sorting of data due to STL algorithms
- Understanding the functionality of error handling with exceptions

Target audience

C/C++ programmers who want to develop object-oriented C++ programs. Project managers in the C++ environment who want to use C++ productively and efficiently.

Requirements

You must at least know the basics of the languages C or C++. We recommend basic knowledge of object-oriented programming according to the following courses:

- Basics of the Languages C and C++ («CPG»)

Any questions?

We are happy to advise you on +41 44 447 21 21 or info@digicomp.ch. You can find detailed information about dates on www.digicomp.ch/courses-software-engineering/programming-languages/c-c/course-moderne-c-programmierung