FEABHAS

C++ Software Testing Fundamentals

Course category	Testing
Training area	Quality
Course code	TC++-201
Duration	2 days
Additional information	For price: Contact Feabhas

Testing is a vital aspect of verifying the design of a software system. This is especially true in a real-time embedded environment, where the system may be subject to strict safety or reliability requirements.

This course introduces the concepts and practice of testing embedded software across the entire development lifecycle. It covers testing software in an embedded environment and is a mixture of lectures and practical exercises.

Attendees will also perform practical exercises in software verification and Agile development practices.

Course objectives:

- To understand the need for testing in an embedded environment
- To give an understanding of the test process
- To understand the objectives of Agile Programming methodologies
- The importance of testing in Refactoring
- To identify the different types of testing and where and when they should be used

Delegates will learn:

- How to create their own test harnesses
- The use of test harnesses for embedded systems
- Integration testing using test doubles
- How to develop in a Test-Driven Development style.

Pre-requisites:

Attendees should have a good working knowledge of the C++ language.

Who should attend:

The course is designed for software engineers working in an embedded, real-time environment. The target audience is engineers who have to create and also test their own code. The course is also highly beneficial for test engineers new to software testing in a real-time environment.

Duration:

- Two days

Course materials:

- Delegate handbook

Course workshop:

During the practical exercises, delegates will gain hands-on experience with development environments and Open Source testing tools and frameworks.

Why do we test?

- Measuring confidence in a product.
- How and why errors occur in software.
- Testing as quality improvement.

Terminology

- The language of testing

The Swiss Cheese model

- Being effective in your testing.
- A layered approach to testing.

Test Harnesses

- The principles of test harness construction.
- Test harness tools

Using Google Test (gtest)

- Introduction to the gtest test suite
- Functional testing with gtest

Black Box Testing

- The principles of black-box testing
- Test Conditions
- Test Cases
- Test Procedures

Systematic Black Box Testing

- The problems with ad-hoc test creation
- The Classification Tree Analysis method

Test Doubles

- Software integration techniques
- The problems with black-box testing during integration
- Test stubs
- Mock objects
- Wrappers

Using Google Mock (gmock)

- Introduction to the gmock test suite
- Create mock objects
- Integration testing with mocks

Coverage Testing

- The principles of white-box testing
- Coverage metrics
- The problems of white-box testing

Static Analysis

- Types of static analysis tool
- Problems with SA tools

Test Planning

- When to perform testing
- Create a test plan

Agile Processes and Testing

- Agile manifesto and principals
- Scrum Lean and Kanban
- Test-driven development (TDD)
- Continuous Integration and delivery

Appendices:

Static Analysis – The Compiler

- The compilation process
- The compiler's semantic analysis capabilities.
- The limitations of the compiler

Static Analysis – Data Flow Analysis

- Data usage in software
- D-U Path analysis

Static Analysis – Coding Standards

- Coding Standards
- Style guides
- MISRA-C++

Feabhas Ltd - UK Office 15-17 Lotmead Business Park, Wanborough, Swindon, SN4 0UY, UK •

+44 (0) 1793 792909 • info@feabhas.com • www.feabhas.com