C++语言单元测试

课程适合人群：

使用C++语言开发产品代码的软件工程师、白盒测试工程师

本课程重点解答&解决：

 介绍单元测试的基本概念

 全面介绍白盒覆盖率内容，包括：语句、分支、条件、MC/DC、路径以及控制流覆盖

 介绍静态代码扫描技术

 介绍如何在Linux环境搭建C++语言编译环境

 介绍C++语言单元测试工具：Google Test

 介绍C++语言覆盖率分析工具GCOV、LCOV

 介绍如何在Window和Linux下搭建C++语言质量扫描工具SonarQube

 了解TDD技术并且对TDD技术进行实战

本课程亮点+核心价值介绍：

 全面介绍白盒覆盖率

 介绍C++语言单元测试工具：Google Test

 介绍如何在Linux下搭建质量扫描工具SonarQube

 了解TDD技术并且对TDD技术进行实战

课程大纲：

 课长3天

1.单元测试概要

 介绍单元测试

 单元测试在传统开发模式中的地位

 单元测试在敏捷开发模式中的地位

 TDD

 测试桩和驱动

 测试应该贯彻始终

 敏捷与DevOps下的单元测试

 代码提交

 嵌入式软件测试的难点

 嵌入式软件测试的方法

2.白盒测试基本概念

 静态测试与动态测试

 精准测试

 白盒测试基本概念-动态分析

 语句覆盖

 分支覆盖

 条件覆盖

 条件/分支覆盖

 MC/DC覆盖

 路径覆盖

 控制流覆盖

 白盒测试基本概念-静态态分析

 MISRA C/C++

 代码错误

 功能层面错误

 有特征的错误

 语法特征错误

 边界行为特征错误

 经验特征错误

 无特征的错误

 算法错误

 部分算法错误

 性能层面的错误

 时间性能层面问题

 空间性能层面问题

3 C++语言单元测试工具

 Google Test

 安装

 准备条件

 安装CMake

 建立google Test

 引言

 Google Test简介

 基本概念

 断言

 书写测试

 简单测试（TEST()宏）

 测试夹具（TEST\_F()宏）

 参数化(TEST\_P()宏)

 如何测试静态、保护和私有变量和方法

 如何测试静态变量和方法

 如何测试保护变量和方法

 如何测试私有变量和方法

 Google Mock

 引言

 Google Mock简介

 为什么选择Google Mock

 Google Mock基础

 测试的重要性

 Google Mock的安装和配置

 测试用例的结构

 编写测试用例的步骤

 创建测试用例

 测试用例的重要性

 测试用例的基本原则

 测试用例的结构

 测试用例的参数化

 测试用例的组织

 示例：测试复杂逻辑

 Mocking的基本概念

 什么是Mocking？

 Mocking与Stub的区别

 为什么使用Mocking？

 使用Google Mock进行Mocking

 示例：Mock一个简单的依赖

 使用Mock验证方法调用顺序

 任意次数的调用

 动作：被调用时的行为

 设置多个期望

 组合Mock和Stub

 Mocking复杂类型（ NiceMock 和StrictMock ）

 Google Mocking的局限性 高级Mocking技巧

 引言

 使用ON\_CALL自定义Mock行为

 模拟异常

 使用Invoke回调函数

 模拟复杂的数据结构

 模拟方法调用的副作用

 模拟方法的多次调用

 使用UnorderedElementsAre匹配容器

 Google Mock的断言和期望

 断言的重要性

 基本断言Google Test

 匹配器：验证参数

 组合断言

 断言动作（Actions）

 验证器（Validators）

 期望和断言的高级用法

4.白盒测试覆盖率工具

 圈复杂度

 扇入扇出

 GCOV、LCOV安装

 GCOV、LCOV运行

 测试报告

 LCOV工具在Jenkins中的应用

5.语法规范检查SonaQube

 JDK 11的安装

 SonaQube支持的数据库

 MS SQL Server安装

 postgresql安装

 SonaQube安装

 SonaQube配置

 sonar-scanner安装及运行

6.TDD实践

 TDD简介

 需求变更带来的