

适用于多模块量子程序的测试流程

龙沛洵* 赵建军†

*中国科学院软件研究所 †九州大学
联系方式: 龙沛洵 longpx@ios.ac.cn

1. 量子程序测试简介

- 随着量子硬件和软件的发展,量子计算机将逐渐走向实际应用,越来越多的研究领域开始编写量子程序来解决相关问题。
- 随着量子程序的复杂性不断增加,如何进行有效的量子软件开发以及提高量子软件质量等工程实践问题变得日益重要。(量子软件工程)
- 随着量子计算逐渐走向实际应用,量子程序的测试在实践中变得越来越关键。目前,对量子程序测试的研究还处于起步阶段,现有的大部分研究工作主要集中在单一且固定规模的量子程序上。



量子硬件

2. 多模块量子程序的测试难点

难点1: 制备所需的量子输入态可能很困难

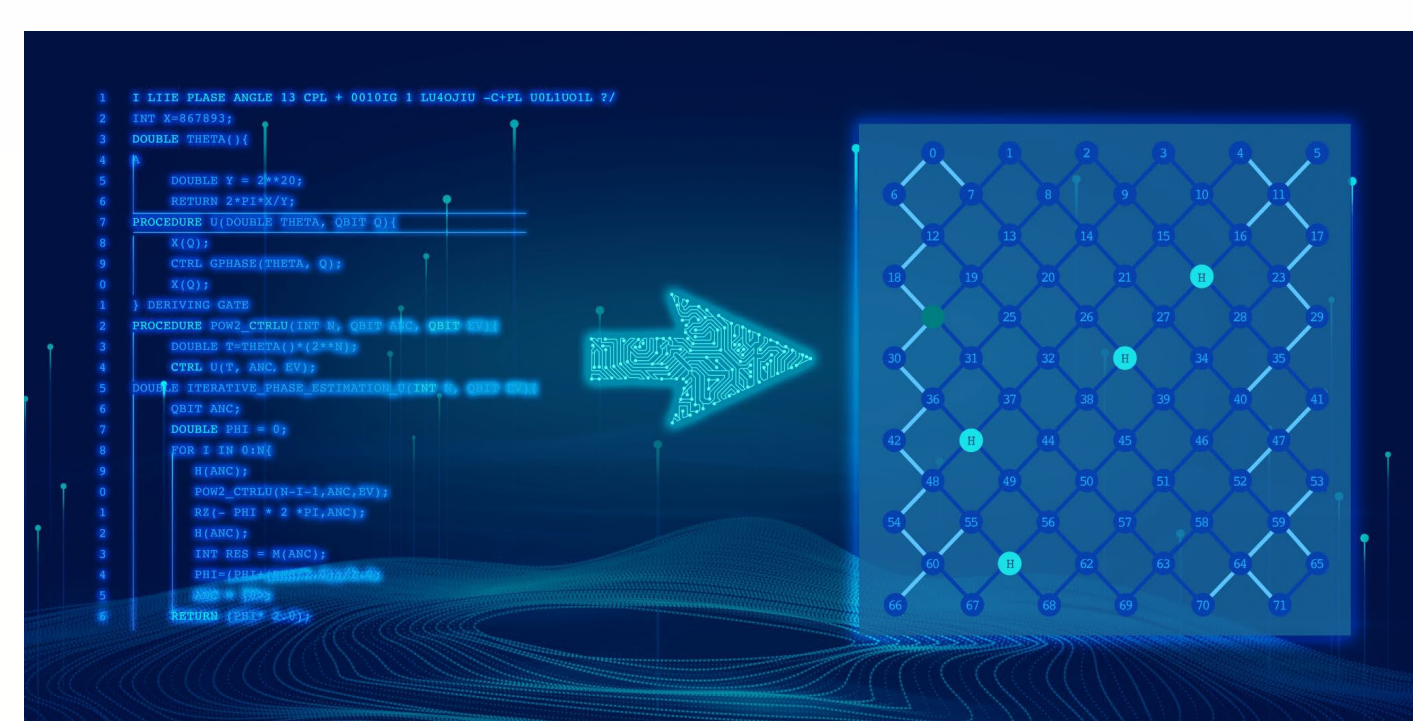
- 纸面上的公式表示 \rightarrow 实际的量子变量状态

难点2: 难以读取量子变量的内容

- 测量会导致量子态塌缩,破坏程序的执行,使得一些基于观测中间变量的经典测试方法失效。
- 检测量子态输出存在困难。

难点3: 量子硬件/模拟器的规模限制

- 量子硬件/模拟器的规模限制了问题的复杂性和程序的规模。

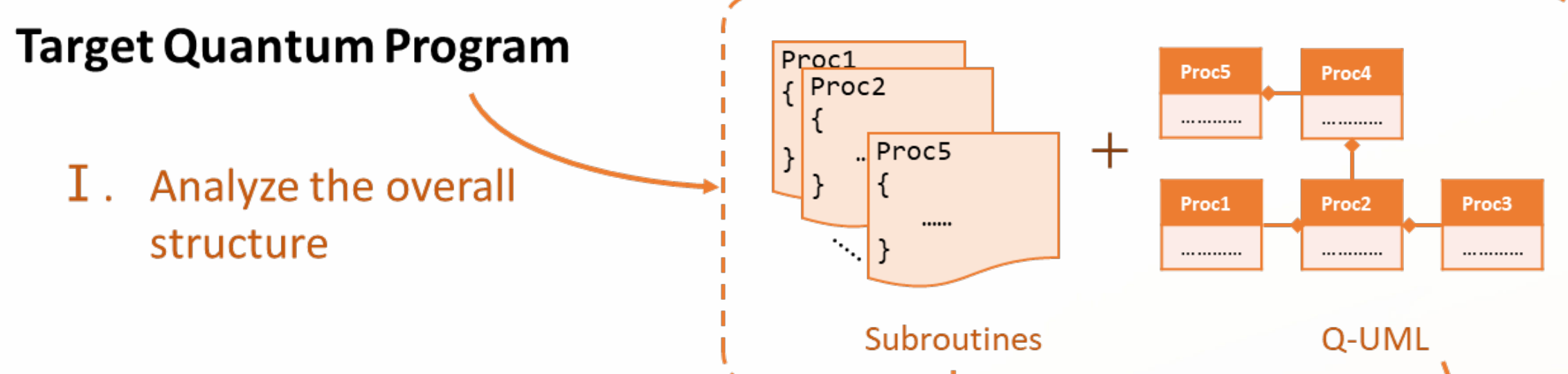


量子软件
(编译器、编程语言、模拟器等)

3. 推荐的多模块量子程序测试流程 (从单元测试到集成测试)

I. 总体结构分析

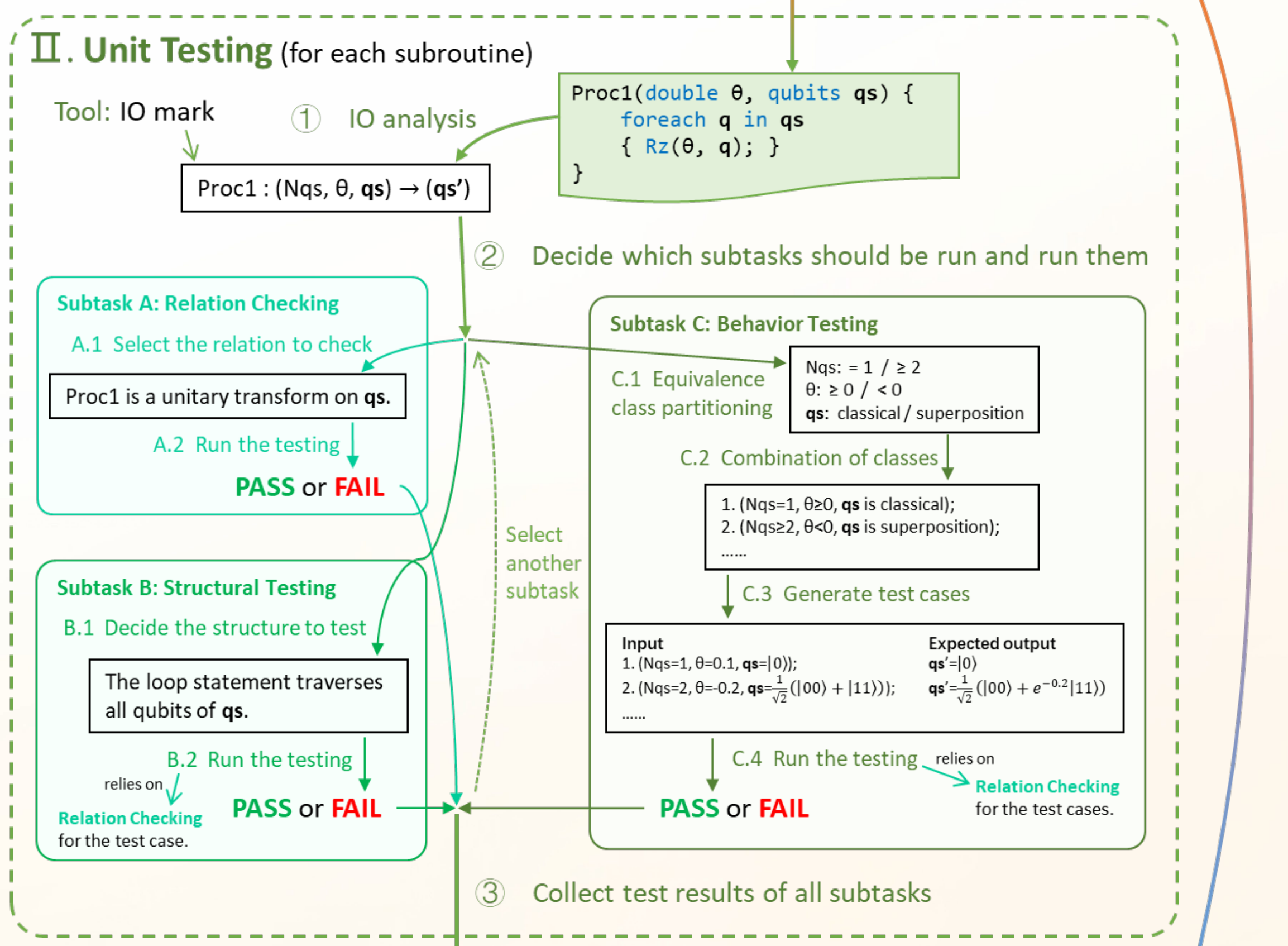
- 获得: 子程序及其组织结构
Q-UML (UML语言的量子扩展)



II. 单元测试

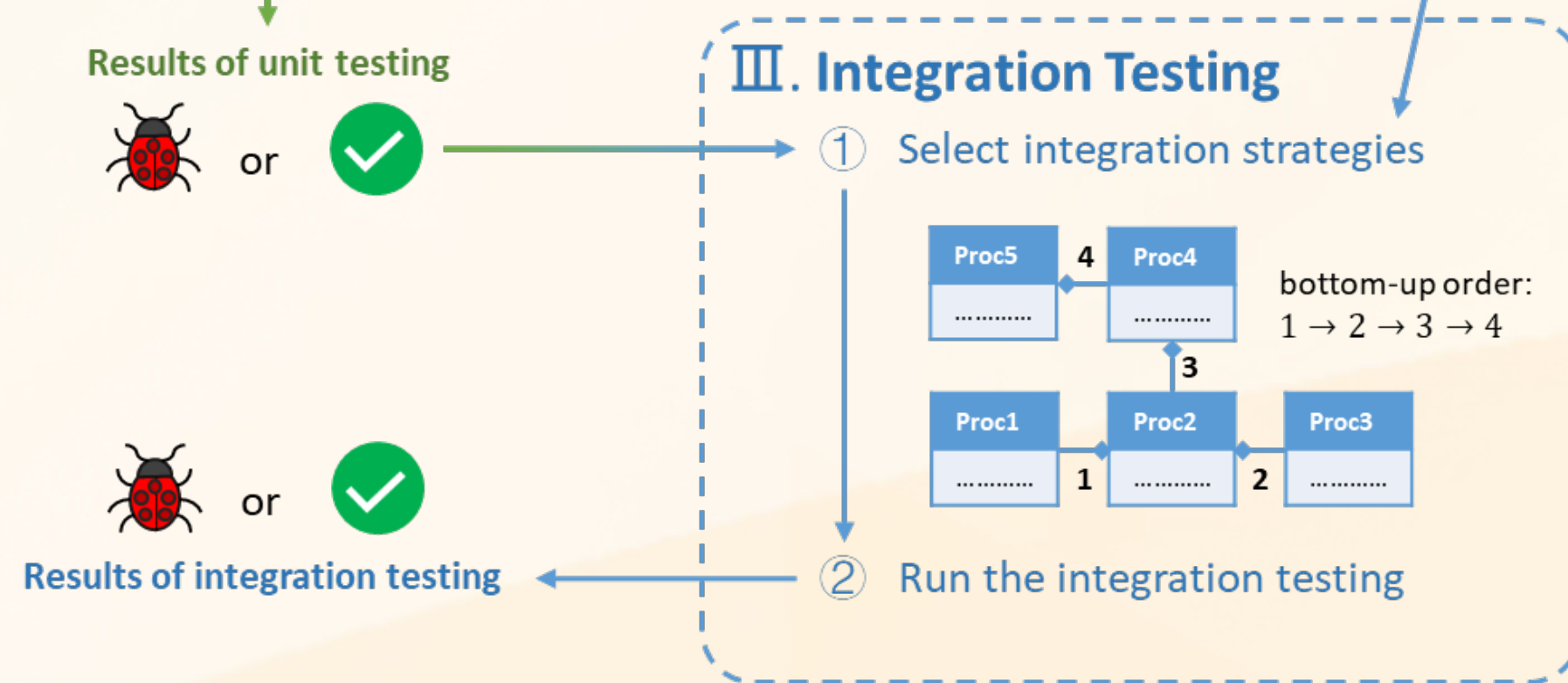
对每个子程序执行:

- 输入-输出分析
 - 识别该子程序的输入输出变量
 - 工具: IO标记
- 确定要执行的子任务,并执行
 - 量子关系检验
 - 量子结构测试
 - 量子行为测试
 - 设计测试用例并执行
- 收集测试结果



III. 集成测试

- 集成策略
- 测试替身 (Test double) 的使用
- 推荐使用“测试驱动”(Test-driven)方法



4. 与经典程序测试对比

子任务	经典对应	主要采用的方法
A. 量子关系检验	蜕变测试	黑盒
B. 量子结构测试	结构测试	白盒
C. 量子行为测试	行为测试	黑盒

- 在经典程序的单元测试中,行为测试(黑盒测试)和结构测试(白盒测试)是主要的测试任务,而蜕变测试通常只被用作辅助手段。
- 然而,在量子程序测试中,由于检测量子态输出的困难性,测试任务通常需要通过量子关系检验来间接完成。这使得量子关系检验(类似于经典蜕变测试)在量子程序单元测试中扮演更为核心的角色,而量子行为测试和量子结构测试则通常依赖于它。

