

单元测试

1、单元测试又称模块测试,是针对软件设计的最小单位(程序模块),进行正确性检验的测试工作。

单元测试主要需要测试者非常清楚代码内部结构

单元测试是软件开发人员的职责,测试人员一般不参与单元测试。

(1) 一般单元的选择依据几个条件

单元必须是可测试的

单元的行为或输出是可观测的

有一个明确的可定义的边界或接口

(2) 单元: 单元能够实现需求规格的最小组件, 可以是: 函数、过程、类、页面、4GL(第四代语言)的菜单和显示

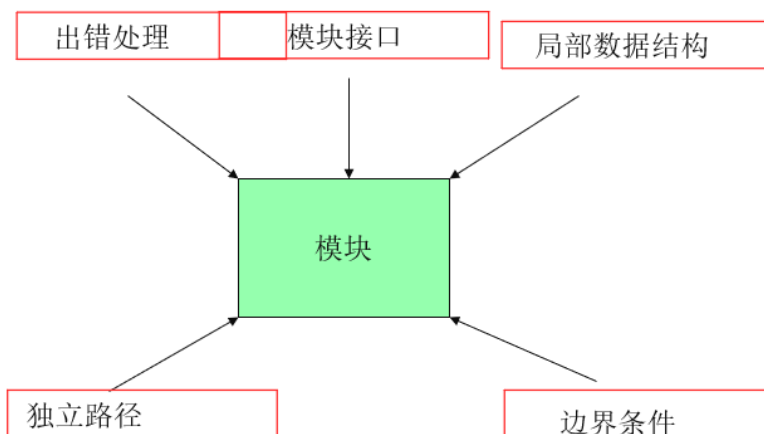
(3) 单元测试的主要目的有:

验证代码和相似设计相符合;(不能检查设计本身)

发现设计中存在的错误;

发现在编码过程中引入的错误;

2、单元测试的任务



- (1) 模块接口测试：进出模块的数据是否正确
- (2) 模块局部数据处理测试：主要检查局部数据结构能否保持完整性
- (3) 模块边界条件测试：检查临界数据是否正确
- (4) 模块独立执行通路测试：检查由于计算机错误、判定错误、控制流错误导致的程序错误
- (5) 模块内部错误处理测试：检查内部错误处理设施是否有效

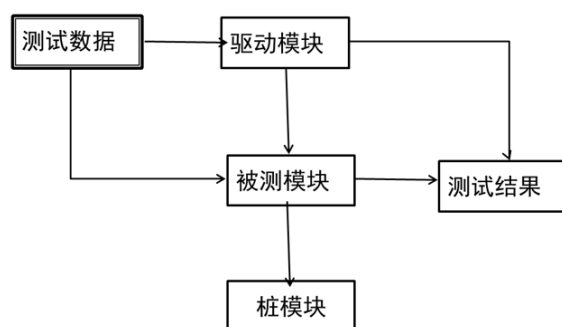
3、单元测试的环境

一、驱动模块与桩模块的概念

驱动模块(driver)是一种用于模拟被测试模块的上一级模块，相当于被测模块的主程序，主要用来接收测试数据，将相关数据传送给被测试模块，并调用被测试模块，打印执行结果。设计驱动模块的目的就是为了访问类库的属性和方法，检测类库的功能是否正确

桩模块(Stub)是模拟被测试模块所调用的模块，它不是软件产品的组成部分。在集成测试前要为被测模块编制一些模拟其下级模块功能的“替身”模块，以代替被测模块的接口，接受或传递被测模块的数据，这些专供测试用的“假”模块称为被测试模块的桩模块。

单元模块测试环境如下



二、驱动模块与桩模块的设计

驱动模块满足的条件：

- (1) 必须能驱动被测试模块的执行
- (2) 能够接收要传递给被测试模块的各项参数，判断其正确性；并将正确的接收数据传送给被测试数据
- (3) 能接收到被测试模块的执行结果，并对结构的正确性进行判断
- (4) 能将判断结果作为测试用例结果并输出测试报告

桩模块满足的条件

- (1) 被测试模块必须调用桩模块
- (2) 桩模块必须能正确地接受来自被测试模块传递的各项参数，并对参数进行正确性判断，并返回执行结果
- (3) 桩模块对外的接口的定义必须与被测试模块调用模块的接口一致

4、单元测试的过程

单元测试一般由编程人员完成，测试人员可以辅助开发人员进行单元测试

具体过程分为：测试计划阶段、测试设计阶段、测试执行阶段和生成测试报告

测试计划阶段

根据被测软件的详细设计说明书、代码及测试任务书，对被测单元进行分析，并确定如下内容：

- (1) 确定被测试单元的目标、范围和约束条件

- (2) 确定被测软件采用的覆盖程度及覆盖的方法和技术
- (3) 确定被测单元的环境，包括软件、硬件、网络、人员配备等
- (4) 确定被测单元的测试结束的要求
- (5) 确定单元测试活动的进度

测试设计阶段

根据测试计划与要求，对被测试单元设计测试用例，一般由测试人员和测试程序员共同完成，主要工作内容：

- (1) 设计测试用例
- (2) 获取测试用例的数据
- (3) 确定测试的顺序
- (4) 获取测试资源，建立测试环境
- (5) 编写测试程序及测试说明文档。

测试执行阶段

根据设计阶段设计好的测试用例，由测试人员对指定的单元进行测试，记录测试步骤及测试结果。主要内容如下：

- (1) 配置单元测试环境
- (2) 执行设计阶段的测试用例，并记录执行过程
- (3) 记录执行测试结果

生成测试报告

根据执行阶段产生的测试结果，由测试分析人员进行分析、总结，得到测试结论并写出测试报告。

主要完成两个方面的工作：

(1) 根据测试设计中的期望值与实际测试执行结果比较,判定该测试能否通过,并记录结果

(2) 若测试不能通过,分析其不能通过的原因,填写软件问题报告,并提出相关建议。

5、单元测试的策略与方法

(1) 静态测试与动态测试相结合

单元测试是一种静态与动态相结合的测试。在执行动态测试之前,经过编译后的单元测试内容,先进行静态测试代码复审,找出其中的错误,可以由程序设计人员、程序编写人员和程序测试人员参与,由软件设计能力较强的高级程序员任组长,在研究软件设计文档基础上召开审查会议,分析程序逻辑与错误清单,测试语言,人工测试,代码复审后在进行计算机代码执行活动的动态测试。所以说是静态与动态相结合的测试。

(2) 白盒测试与黑盒测试相结合

单元测试主要采用白盒测试方法,辅以黑盒测试方法。其中白盒测试应用于代码评审、单元程序执行。在白盒测试方法中,以路径覆盖为最佳准则,且系统内多个模块可以并行进行测试。而黑盒测试方法则应用于模块、组件等大单元的功能测试。

(3) 人工测试与自动化测试相结合

人工测试是由测试人员手工逐步执行所有的活动,并观察每一步是否成功完成。人工测试是任何测试活动的一部分,在开发初始阶段软件及其用户接口还未足够稳定时尤其有效。

在不能使用自动化测试工具时，必须采用人工测试的方法对单元相关内容进行测试。

自动化测试是把人为驱动测试行为转化为机器执行的一种过程。在设计了测试用例并通过评审之后，由测试人员采用自动化测试工具，根据测试用例中描述的规程一步步执行测试，得到实际结果与期望结果的比较。在此过程中，节省人力、时间或硬件资源，提高测试效率。