# 四、测试（第14至23章）

## 1. 技术评审（在建模编码阶段）

找出错误和发现可能对将要部署的软件产生负面影响的问题

## 1. 验证与确认

验证：确保软件正确地实现某一特定功能的一系列活动

确认：确保开发的软件课追溯到客户需求的另一系列活动

## 测试的目标和衡量标准。

测试目标：发现错误。

衡量标准：只有当发现了错误时，测试才被认为是成功的。

## 2. 测试的分类（或组织）。各种类型的测试的主要任务及所依赖的文档。

单元测试：将每个程序构件与系统中其他构件隔离，对其本身进行测试。文档：构件代码。

集成测试：验证系统构件是否能够按照系统和程序设计规格说明中描的那样共同工作。文档：设计规格说明。

功能测试：对系统进行评估，以确定集成的系统是否确实执行了需求规格说明书中描述的功能。文档：系统功能需求。

性能测试：将系统与这些软件和硬件需求的剩余部分进行比较。当测试在客户的实际工作环境中成功时，它会产生一个确认的系统。文档：其他软件需求。

验收测试：与客户一起执行验收测试，其中根据客户的需求描述对系统进行检查。文档：客户需求规格说明书。

安装测试：确保系统将按照它应该的方式来运行。文档：用户环境。

## 3. 黑盒测试和白盒测试的思想，了解白盒测试中的基本路径测试等方法。

闭盒或黑盒: 测试对象的功能。如果从外部观察测试对象，将其看作一个不了解其内容的黑盒，那么，我们的测试就是向闭盒提供输入数据，并记录产生的输出。在这种情况下，测试的目标是确保每一种输入都被提交，并且观察到的输出与预期的输出相匹配。（程序控制结构）

开盒或白盒: 测试对象的结构。对于某些测试对象，测试小组不可能生成一组证明所有情况下功能正确的、有代表性的测试用例，为了克服这个难题，我们将测试对象看做一个开盒，然后可以根据测试对象的结构用不同的方式来进行调试。（确认功能需求）

基本路径测试法是在程序控制流图的基础上，通过分析控制构造的环路复杂性，导出基本可执行路径集合，从而设计测试用例的方法。

！计算环复杂度（课本P348）

## 4. 单元测试的主要内容。

边界测试、错误处理测试、路径测试、局部数据结构测试、模块接口测试。

**检查代码**

* 代码走查
* 代码审查

**证明代码的正确性**

* 形式化证明技术
* 符号执行
* 自动定理证明

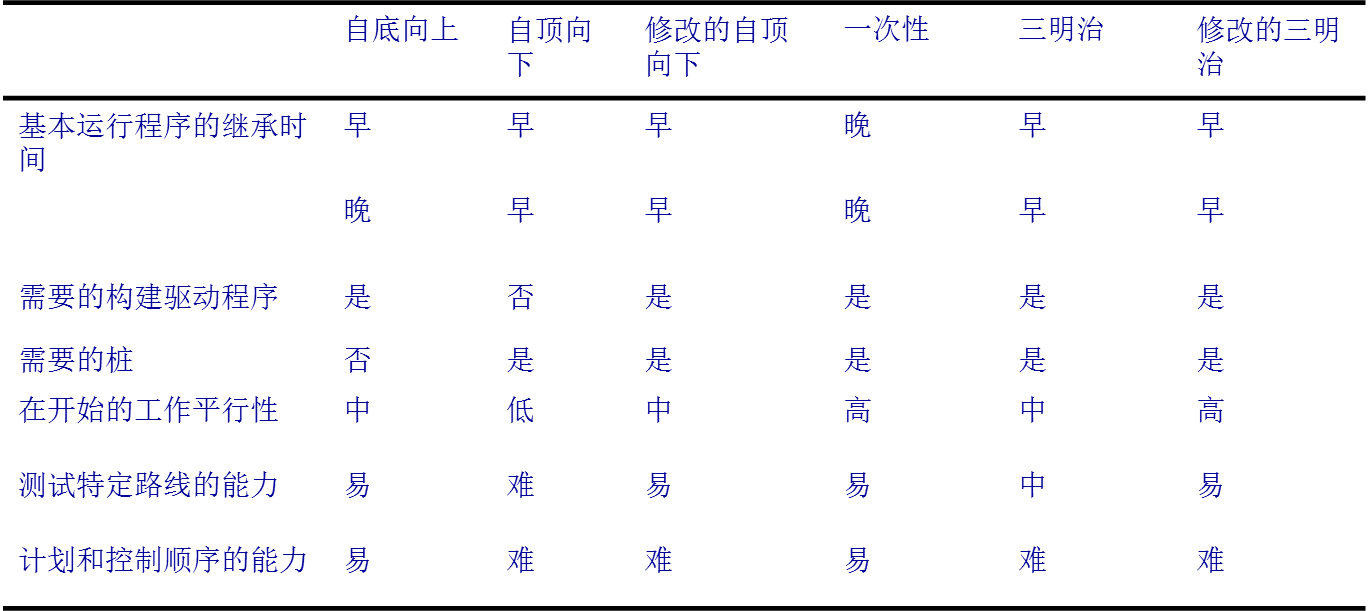
**测试程序构件**

* 测试与证明
* 选择测试用例
* 测试的完全性

**技术比较**

## 5. 集成测试的类型及主要的测试策略。

* 自底向上的测试：主控模块—原子模块
* 自顶向下测试：原子—主控
* 一次性测试
* 三明治测试
* 改进的自顶向下测试
* 改进的三明治测试
* **策略比较**



## 6. 确认测试的内容。

确认测试的目的是向未来的用户表明系统能够像预定要求那样工作。经[集成测试](http://baike.baidu.com/view/106652.htm)后，已经按照设计把所有的模块组装成一个完整的[软件系统](http://baike.baidu.com/view/8343.htm)，接口错误也已经基本排除了，接着就应该进一步验证软件的有效性，这就是确认测试的任务，即软件的功能和性能如同用户所合理期待的那样。

α测试：软件开发公司组织内部人员模拟各类用户对即将面市软件产品（称为α版本）进行测试，试图发现错误并修正。在受控环境下进行

β测试：软件的多个用户在实际使用环境下进行的测试，这些用户返回有关错误信息给开发者。（又称客户验收测试）

## 7. 了解测试计划的主要内容。

* 构建测试目标
* 设计测试用例
* 编写测试用例
* 测试测试用例
* 执行测试
* 评估测试结果

## 8. 测试系统中的测试过程：功能测试、性能测试、验收（或确认）测试、安装测试，及它们的内容。

* 功能测试: 集成系统是否按照需求规格说明执行它的功能?
* 性能测试: 是否满足非功能需求? 用于检查响应速度、结果的精确性、数据的可访问性
* 验收测试: 系统是客户期望的吗？让客户和用户能够确定我们构建的系统满足了他们的期望
* 安装测试: 系统能在客户端运行吗 ? 配置系统，将正确的数量和种类的设备连接到主处理器上，与其他系统建立通信