**实验十四 软件测试**

实验目的：

1. 了解软件测试
2. 深入理解白盒测试和黑盒测试
3. 了解符号测试
4. 了解差分测试

实验内容：

1. 阅读下面软件测试相关资料（或查阅其它相关资料），了解软件测试的基本概念、主要技术、重要挑战等

Software Testing-A Research Travelogue (2000–2014).pdf

1. 阅读下面白盒测试和黑盒测试相关资料（或查阅其它相关资料），深入理解白盒测试和黑盒测试，总结其特点 （保存到每个小组选定的协作开发平台上，以组为单位）

WhiteBox.pdf

BlackBox.pdf

1. 阅读下面符号测试 （Symbolic Testing）相关资料（或查阅其它相关资料），了解符号测试的基本概念、主要技术、重要挑战等

A Survey of Symbolic Execution Techniques.pdf

Symbolic Execution and Program Testing.pdf

Symbolic Execution for Software Testing-Three Decades Later.pdf

1. 阅读下面差分测试 （Differential Testing）相关资料（或查阅其它相关资料），了解差分测试的基本原理、主要应用等

Differential Testing for Software.pdf

Feedback-Directed Differential Testing of Interactive Debuggers.pdf

项目跟踪，建立能反映项目及小组每个人工作的进度、里程碑、工作量的跟踪图或表，将其保存到每个小组选定的协作开发平台上，每周更新。

1.软件测试的经典定义：

在规定的条件下对程序进行操作，以发现程序错误，衡量软件质量，并对其是否能满足设计要求进行评估的过程。

2.标准定义(IEEE)  
软件测试是使用人工或自动的手段来运行或测定某个软件系统的过程，其目的在于检验它是否满足规定的需求或弄清预期结果与实际结果之间的差别。  
3.测试目的  
软件测试的目的是发现问题，发现至今未发现的问题。检查系统是否满足需求。  
Grenford J.Myers观点（1）测试是程序的执行过程，目的在于发现错误；  
（2）一个好的测试用例在于能发现至今未发现的错误；  
（3）一个成功的测试是发现了至今未发现的错误的测试；

4.软件测试范围：

对软件形成中的文档、数据以及程序进行的测试，而不仅仅是对程序进行的测试。软件测试贯穿于整个软件生命周期中。

5.软件测试的重要性：

60%以上的软件错误并不是程序错误，而是分析和设计错误，因此做好软件需求和设计阶段的测试工作就显得非常重要。

6.软件测试的目的：

不仅仅为了发现软件缺陷与错误，而且也要对软件质量进行度量和评估，以提高软件的质量。测试是为发现错误而执行一个程序或者系统的过程，测试是为了证明程序有错，而不是证明程序无错误。

7.软件测试的分类：

软件测试方法和技术的分类与软件开发过程相关联，它贯穿了整个软件生命周期。

7.1 按照开发阶段划分软件测试：单元测试、集成测试、系统测试、验收测试。

7.2 按照测试实施组织划分软件测试：开发方测试、用户测试(alpha测试，Beta测试)、第三方测试。

7.3 按照测试技术划分：白盒测试、灰盒测试、黑盒测试。

8.软件测试风险：

软件测试中的软件风险分析是根据预测软件将出现的风险，制定软件测试计划并排列优先等级，风险分析是对软件中潜在的问题进行识别、估计和评价的过程。

软件测试风险也包括进度风险、质量风险、人员风险、变更风险、成本风险等。

9.软件测试原则：

基于测试是为了寻找软件的错误与缺陷，评估与提高软件质量，测试原则,如下所示。

9.1 所有的软件测试都应追溯到用户需求。

9.2 应当把“尽早地和不断地进行软件测试”作为软件测试者的座右铭。

9.3 完全测试是不可能的，测试需要终止。

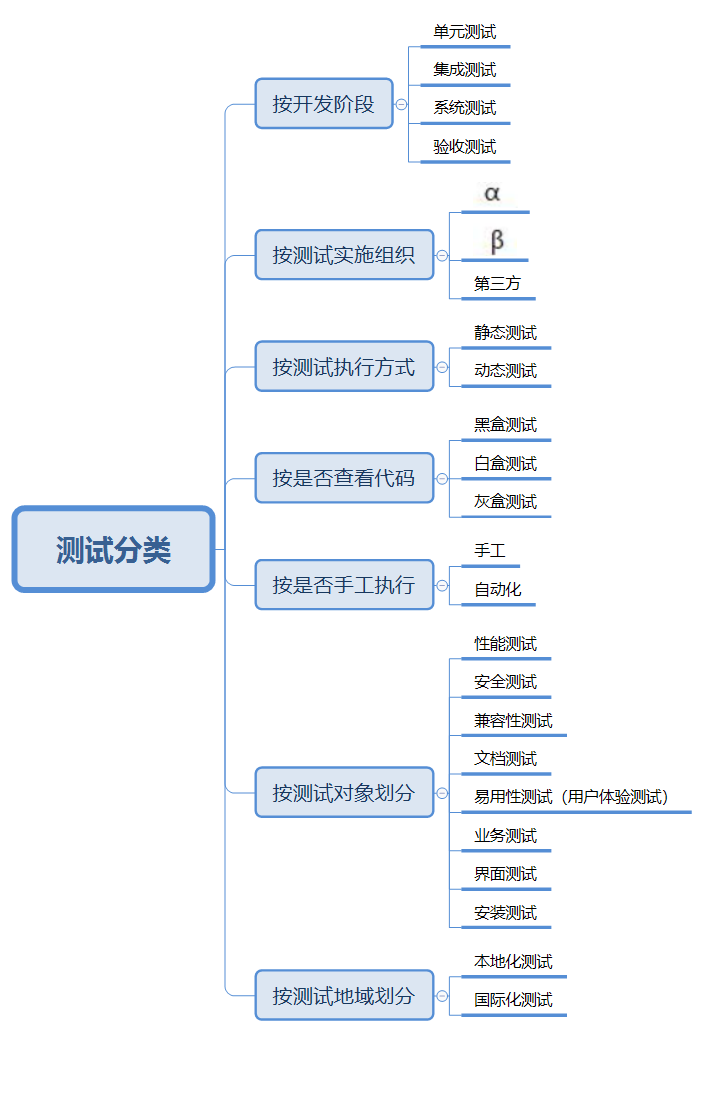
9.4 测试无法显示软件潜在的缺陷。

9.5 充分注意测试中的群集现象。

9.6 程序员应避免检查自己的程序。

9.7 尽量避免测试的随意性。

10.测试分类



白盒测试

1、白盒测试的定义

白盒测试又称为结构测试或逻辑驱动测试，它是把测试对象看成一个透明的盒子，它允许测试人员利用程序内部的逻辑结构设计测试用例，对程序所有逻辑路径进行测试。

2、白盒测试的测试对象

白盒测试的测试对象是基于被测试程序的源代码，而不是软件的需求规格说明书。

使用白盒测试方法时，测试人员必须全面了解程序内部逻辑结构，检查程序的内部结构，从检查程序的逻辑着手，对相关的逻辑路径进行测试，最后得出测试结果。

3、白盒测试的原则

采用白盒测试方法必须遵循以下原则：

保证一个模块中的所有独立路径至少被测试一次。

对所有的逻辑判定均需测试取真和取假两种情况。

在上下边界及可操作范围内运行所有循环。

检查程序的内部数据结构，保证其结构的有效性。

4、白盒测试的分类

白盒测试方法有两大类：静态测试方法和动态测试方法。

静态测试： 不要求在计算机上实际执行所测试的程序，主要以一些人工的模拟技术对软件进行分析和测试，如代码检查法、静态结构分析法等；

动态测试： 是通过输入一组预先按照一定的测试准则构造实际数据来动态运行程序，达到发现程序错误的过程。白盒测试中的动态分析技术主要有逻辑覆盖法和基本路径测试法。

黑盒测试

1.定义

黑盒测试又称为功能测试，它是通过测试来检验程序的每个功能是否能正常使用。在测试中，将程序看成一个不能打开的黑盒子，在完全不考虑内部结构的情况下，在 程序接口 进行测试 ，检查程序是否能适当的接受输入数据从而产生正确的输出信息。黑盒测试主要针对 功能测试 和 软件界面测试 。

2.作用

黑盒测试方法着重测试软件的功能需求，是在程序的接口上进行测试，主要发现以下错误：

2.1.是否有功能错误，是否有功能遗漏。

2.2.是否能正确的接受输入数据达到正确的输出信息。

2.3.是否有数据结构错误或外部信息访问错误。

白盒测试是指实际运行被测程序，通过程序的源代码进行测试而不使用用户界面。  
黑盒测试又称功能测试、数据驱动测试或基于规格说明的测试，是通过使用整个软件或某种软件功能来严格地测试，而并没有通过检查程序的源代码，或者很清楚地了解该软件的源代码程序具体是怎样设计的。  
白盒测试特点：  
可检测代码中的每条分支和路径 ，揭示隐藏在代码中的错误；昂贵且无法检测代码中遗漏的路径和数据敏感性错误，不验证规格的正确性。  
黑盒测试的特点：  
测试人员不需要了解实现的细节（特定编程语言）；测试人员和编程人员彼此独立；有助于暴露规格的不一致或有歧义的问题；测试用例可以在规格完成后马上进行；只有一小部分输入被测试到，要测试每个可能的输入几乎不可能。  
如果测试人员不被告知开发人员已经执行过的用例，在测试数据上会存在不必要的重复；不能直接针对特定程序段测试，而这些程序段可能很复杂，有可能隐藏更多的问题。

符号测试技术主要包括：变量替换、表达式简化和约束条件求解。可用于程序路径检查，程序证明，程序约简等。是程序静态测试方法之一，不需要运行被测程序，通过程序源代码进行词法、语法分析以查找程序中的缺陷。可以精确地找出满足约束条件的测试用例或找出不可达路径。当执行完源程序后，得出不等式组，有多少路径就有多少不等式组，不等式组中的变量全部为初始输入变量，解空间是满足当前路径的测试用例。如果不等式组解空间不存在，则当前路径不可达。因此将测试用例生成问题变成了约束求解问题。

1.概述：普通测试执行的是算术运算，符号测试则是执行代数运算，可以代表一类的测试。

符号值可以是初等符号值，也可以是表达式。初等符号是任何变量值的字符串，表达式则是数字、算术运算符和符号值的组合。

2.符号执行树

在条件语句中，判断条件就是谓词，可能是符号表达式，可取真假。不断构造下去，构成了一个二叉树，称为符号执行树。将各个分支点的谓词条件累积在一起，用逻辑乘符号联接在一起，得到的这个逻辑表达式称为路径条件。

3.目前挑战  
分支问题、二义性问题、大程序问题。

4.基本思想

允许程序的输入不仅仅是具体的数值数据，而且包括符号值，符号值可以是基本的符号变量值，也可以是符号变量值的表达式。

5.优点

1）符号测试执行的是代数运算，可以作为普通测试的扩充。

2）符号测试可以看做是程序测试和程序验证的一种折中办法。

3）符号测试程序中仅有有限的几条执行路径。

差分测试通过比较每个实现的输出来交叉引用同一功能的多个实现。 假设我们有一个函数规范 F(X)，以及该规范的两个实现：f1(X) 和f2(X)。 对于存在于适当输入空间中的所有 x，我们期望 f1(x) == f2(x)。 如果 f1(x) != f2(x)，我们知道至少有一个函数错误地实现了 F(X)。 这种测试等效和识别差异的过程是差异测试的核心。

差异模糊测试是差异测试的扩展。 差异模糊以编程方式生成许多x值，以发现手动选择的输入可能无法揭示的差异和边缘情况。

这种类型的测试在现实生活中的一些用途包括：

* 将升级后的实施与其之前版本进行比较
* 针对已知参考实现测试代码
* 确认与第三方工具和依赖项的兼容性