# 软件测试

## 第一章

导致软件缺陷的最大原因是产品说明书；

开发小组把软件的初期版本发给小部分客户进行试用，叫beta测试；

软件测试员的目标是尽可能早地找出软件缺陷，并确保其得到修复；

## 第二章

### 软件开发生命周期模式

**大爆炸模式：**优点是简单，计划、进度安排和正规开发过程几乎没有，所有精力花在开发软件和编码上。**要尽量避免在此模式下进行测试。**

**边写边改模式：**测试在此模式中未特别强调，但是在编写代码和修复缺陷过程中举足轻重。

**瀑布模式：**无法回溯

**螺旋模式：**一开始不必详细定义所有细节。测试人员参与最初的设计，可以尽早影响产品、了解产品的来龙去脉；在产品末期，不至于匆忙进行所有的测试，测试一直在进行，最后只需要进行最后的验证。

**敏捷开发（极限编程）：**在敏捷开发中，软件项目在构建初期被切分成多个子项目，各个子项目的成果都经过测试，具备可视、可集成和可运行使用的特征。换言之，就是把一个大项目分为多个相互联系，但也可独立运行的小项目，并分别完成，在此过程中软件一直处于可使用状态。

## 第三章

### 测试的原则

1 完全测试是不可逆的**：**

输入量太大；输出量太多；软件执行路径太多；软件说明书是主观的；

2 软件测试是有风险的行为，因为只能进行不完全测试；

3 可以报告软件缺陷存在，不能报告软件缺陷不存在，因为这种情况根本不存在；

4 一个软件缺陷表明周围可能还有更多的软件缺陷；

5 软件测试越多，缺陷对测试的免疫力越强，所以要编写不同的测试用例，对程序的不同部分进行测试；

6 并非所有的软件缺陷都要修复。考虑到时间和修复可能带来更大的风险、不值得修复等。

### 软件测试的术语和定义：

**精确与准确：**准确是要得出正确答案，精确是偏离正确答案多少

**验证与确认：**确认是保证软件符合说明书的过程；验证时软件满足用户要求的过程；

**质量和可靠性：**用户眼里，可靠性是质量的一部分；

**测试和质量保证：**

软件测试员的目标是尽可能早地找出软件缺陷，并确保其得到修复；

软件质量保证人员的主要职责是创建和执行改进软件开发过程并防止软件缺陷发生的标准和方法；

## 第四章

### 黑盒测试和白盒测试

黑盒测试：又叫功能性测试或者行为测试，只需知道软件做什么，进行一些输入，得到一些输出结果，无法看到盒子里的软件是如何运行的，

白盒测试：透明盒测试，可以访问程序员的代码，分析代码协助测试，要冒风险，因为为了适应代码定制输入，测试可能不客观。

### 静态测试和动态测试

静态测试指测试不运行的部分，知识检查和审查；动态测试是一般意义的测试，使用运行软件

### 静态黑盒测试、测试产品说明书

产品说明书不能运行，所以是静态的，不要了解说明书信息获取原因和途径只要知道最后的说明书，所以是黑盒

### 对产品说明书进行高级审查

1 假设自己是客户

2 研究现有的标准和规范

3 审查和测试类似软件

### 产品说明书低层次测试技术

1 产品说明书属性检查清单

2 产品说明书属于检查清单

## 第五章

### 动态黑盒测试

### 通过性测试和失效性测试

通过性测试：确认软件至少能做什么

失效性测试：纯粹为了破坏软件而设计和执行的测试用例

### 等价类划分

分步骤地把海量测试用例集减得很小，但是过程同样有效

在寻找等价划分时，考虑把软件具有相似输入相似输相似操作的分在一组，这一组就是等价划分

### 数据测试

根据关键原则进行等价类划分，包括**边界条件，次边界条件，空值和无效数据**

提出边界条件时，一定要测试临近边界的有效数据，测试最后一个可能有效地数据，同时测试超过边界的无效数据

次边界条件，如2的幂，ASCII集，Unicode

默认、空白、空值、零值和无

非法、错误、不正确和垃圾数据

失败状态测试：竞争条件和时序错乱，重复（检测内存泄漏）、压迫（软件在不够理想的条件下运行）和重负

## 第六章

数据引用错误是造成缓冲区溢出的主要原因

### 静态白盒测试（结构化分析）

在不执行软件的条件下有条件地仔细审查软件设计、体系结构和代码，从而找出缺陷的过程，**也叫结构化分析**

### 正式审查

同事审查（非正式）、走查（编写代码的程序员向小组做正式）、检验（最正式，高度组织化，表述者不是原来的程序员，检验员从不同的角度审查代码）

## 第七章

### 动态白盒测试（结构化测试）

利用查看代码功能和实现方式得到的信息来确定哪些需要测试，哪些不需要测试，如何开展测试。

### 动态白盒测试和调试

动态白盒测试的目标是寻找缺陷，调试的目的是修复缺陷

### 分段测试

**单元测试和集成测试**

底层进行的测试称为单元测试或者模块测试，对模块的组合进行集合测试，对产品或产品的主要部分进行的测试叫做系统测试。

分段测试很容易隔离软件缺陷。

**这种递增测试有两种途径：自底向上和自顶向下**

自底向上，要编写测试驱动模块调动正在测试的模块，，这样可以对整个软件进行非常全面的测试，为他提供全部类型和数量的数据，甚至高层难以发送的数据。

自顶向下，有时候为了方便，编写小段下层测试桩来对高层进行测试

***在进行白盒测试之前一定要根据说明书建立黑盒测试用例。***

### 数据覆盖

数据流、次边界、公式和等式、错误强制

### 代码覆盖

设法进入和退出每一个模块，执行每一条代码，进入软件每一条逻辑和决策分支。

对于小程序或单独模块，使用调试器一般足够，但是对大多数程序进行代码覆盖要用到代码覆盖率分析器的专用工具。[注:python中使用了coverage，显示覆盖文件和覆盖率，如果覆盖了会绿色高亮显示，未覆盖用红色标注，coverage集成在pycharm中]

**程序语句覆盖和代码行覆盖**

注意：即使全部语句都背覆盖了，也不能说走遍了软件的所有路径。

**分支覆盖**

试图覆盖软件中的所有路径，

**条件覆盖**

## 第八章 配置测试

## 第九章 兼容性测试

**软件兼容性测试**是指检查软件之间是否能够正确地交互和共享信息；

### 平台和应用程序版本

**向后兼容：**可以使用软件的以前版本

**向前兼容：**可以使用软件的未来版本

测试多个版本的影响，关键是“重要”，可以按流行***程度、年头、类型、生产产商***

**数据共享兼容性：**A文件保存和文件读取；B文件导出和文件导入 C剪切复制粘贴

## 第十章 外国语言测试

使软件使用特定地域特征，照顾到语言、方言、地区习俗和文化的过程称为本地化或者国际化，测试此类软件称为本地化测试

### 翻译问题

文本扩展，如英文button翻译成中文button，按钮边长或者变短

ASCII DBCS Unicode

热键和快捷键

扩展字符

从左向右和从右像左读：如希伯来文和阿拉伯文

图形中的文字

让文本与代码脱离:将文本字符串、错误提示等放在资源文件中，不要放在代码中，也不要用代码来连接字符串

### 本地化

内容要确保适应使用该软件的地区

数据格式

## 第十一章 易用性测试

### 优秀的UI构成

符合标准和规范

直观：界面洁净不唐突不拥挤、UI组织布局合理性、信息是否太庞杂、

一致：快捷键和菜单选项是否一致，术语和命名，听众，ok和cancel按钮位置

灵活：允许用户选择想要做的和怎么做

舒适：恰当、错误处理、性能

正确：注意市场偏差、语言和拼写、不良媒体和所见即所得

实用：具体特性是否实用

### 为有残疾障碍的人员测试—辅助选项测试

视力损伤、听力损伤、运动损伤、认知和语言障碍

## 第十四章 网站测试

灰盒测试：介于白盒和黑盒之间，HTML和网页可以视为灰盒测试。灰盒测试是可以边看着代码、边利用代码的信息帮助测试。它不像白盒测试一样详细地检查代码。代码用来协助测试，但是测试并不完全基于代码。

## 第十五章 自动测试和测试工具

工具和自动化的好处：速度、效率、准确度和精确度、节省资源、防真和模拟、坚持不懈

### 测试工具分类

**查看器和监视器：**能够看到正常情况下看不到的软件运行的细节。（如看覆盖率的coverage）

**驱动程序：**控制和操作被测试软件的工具。

**桩（仿真器）：**不控制或者操作被测试软件，相反，它接收或者响应软件发送的数据

**压力和负载工具：**向被测试软件增加压力和负载

**干扰注入器和噪声发生器：**类似压力和负载工具，但是在行为上更具有随机性

**分析工具：**如电子表格，数据库软件。文件比较软件

### 软件测试自动化

宏录制和回放

可编程的宏

完全可编程的自动测试工具

### 随机测试（模拟非专业用户）

猴子程序、半聪明的猴子程序、聪明的猴子程序