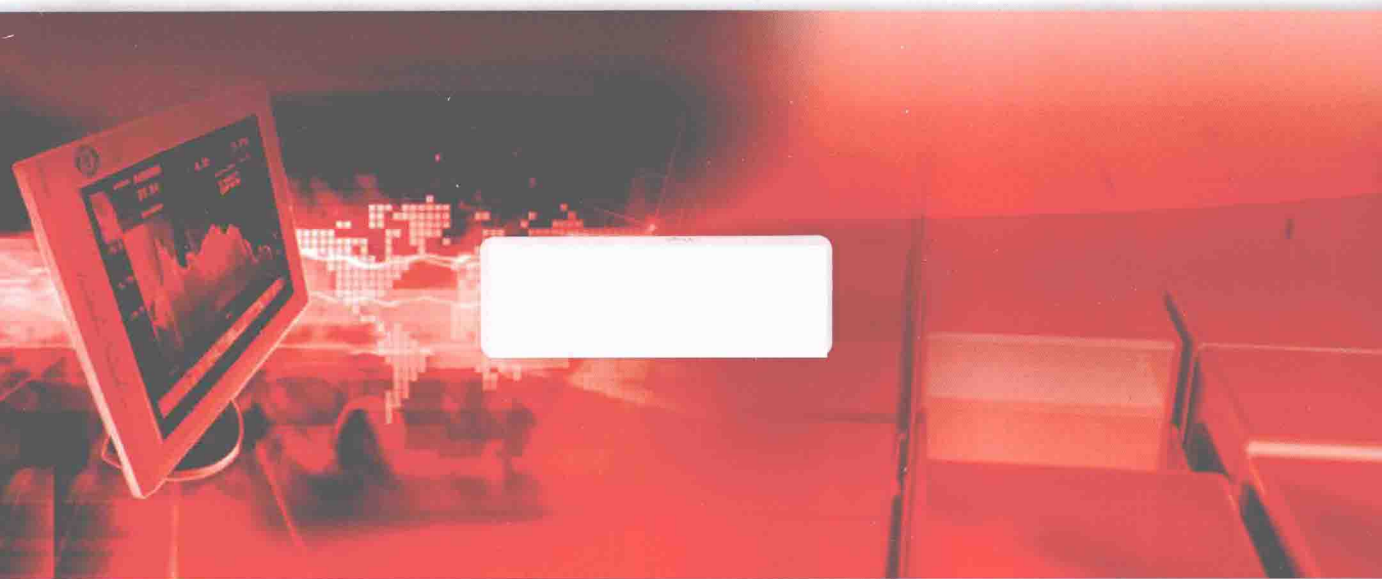




高等职业教育“十二五”规划教材（计算机类）

# 软件测试基础教程

牛红 刘卫宏 唐国平 编



 机械工业出版社  
CHINA MACHINE PRESS

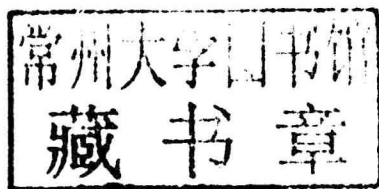


配电子课件

高等职业教育“十二五”规划教材（计算机类）

# 软件测试基础教程

牛 红 刘卫宏 唐国平 编



机械工业出版社

本书由多年从事“软件测试”课程教学工作的教师和从事软件测试与管理工作的工程师合作编写,期望既能把丰富的软件测试教学经验和实践成果融入书中,又能将一线公司测试专家多年积累的经验和操作技巧奉献给读者。

本书共8章,内容包括软件测试概述、软件测试基础、软件质量与测试、软件测试技术和方法、软件测试类型、软件测试管理、软件测试自动化工具以及单机版五子棋游戏测试实例。

本书在内容上既介绍了软件测试的基本理论和知识,又给出了很多具有实用价值的具体操作方法,做到了理论与实践深度融合。本书适合作为高职高专计算机专业相关课程的教材,也可以作为软件测试人员的参考书或培训用书。

为方便教学,本书配备电子课件等教学资源。凡选用本书作为教材的教师均可登录机械工业出版社教育服务网 [www.cmpedu.com](http://www.cmpedu.com) 免费下载。如有问题请致信 [cmpgaozhi@sina.com](mailto:cmpgaozhi@sina.com),或致电 010-88379375 联系营销人员。

## 图书在版编目(CIP)数据

软件测试基础教程 / 牛红, 刘卫宏, 唐国平编. —北京: 机械工业出版社, 2014.10  
高等职业教育“十二五”规划教材. 计算机类  
ISBN 978-7-111-48081-5

I. ①软… II. ①牛… ②刘… ③唐… III. ①软件—测试—高等  
职业教育—教材 IV. ①TP311.5

中国版本图书馆CIP数据核字(2014)第222505号

机械工业出版社(北京市百万庄大街22号 邮政编码100037)

策划编辑: 刘子峰 责任编辑: 刘子峰 陈瑞文

版式设计: 霍永明 责任校对: 路清双

封面设计: 陈沛 责任印制: 李洋

北京宝昌彩色印刷有限公司印刷

2014年11月第1版第1次印刷

184mm×260mm·12.75印张·309千字

0001—3000册

标准书号: ISBN 978-7-111-48081-5

定价: 28.00元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

社服务中心: (010) 88361066

教材网: <http://www.cmpedu.com>

销售一部: (010) 68326294

机工官网: <http://www.cmpbook.com>

销售二部: (010) 88379649

机工官博: <http://weibo.com/cmp1952>

读者购书热线: (010) 88379203

封面无防伪标均为盗版

# 前 言

当今社会不断在向信息化方向转型，同时也推动着软件产业的蓬勃发展。软件测试作为保证软件质量的重要手段，日益受到行业的关注。未来，软件产业对专业化的软件测试队伍的需求会越来越大，将会有更多的年轻人从事与软件测试相关的工作。因此，高校计算机专业的学生学好软件测试这门课程就显得愈加重要。

本书根据市场对软件测试专业人才的需要，结合高职高专院校计算机专业教学大纲“软件测试”课程的教学特点编写而成，旨在使读者可以较全面地学习软件测试的基础知识和应用技能。在内容上，既介绍了软件测试的基本理论和知识，又给出了很多具有实用价值的具体操作方法，做到了理论与实践的深度融合。在内容安排上，力争做到由浅入深、循序渐进，同时还提供了例题、案例和习题，便于教师组织教学及学生自学。

本书共 8 章，建议学时数为 64 课时，具体内容如下：

第 1 章主要介绍软件测试基础知识、软件测试职业和职位以及测试人员的素质要求。

第 2 章主要介绍软件测试过程模型，从不同的角度介绍软件测试的分类，最后介绍软件测试流程。

第 3 章主要介绍软件的质量特性及子特性、软件质量的度量、基于软件质量特性的测试、ISO 9000 质量体系标准以及软件能力成熟度模型（CMM）。

第 4 章主要介绍黑盒测试和白盒测试等一些通用的测试技术以及测试技巧。

第 5 章主要介绍单元测试、集成测试、确认测试、系统测试等软件测试类型和测试实例。

第 6 章主要介绍测试流程管理、测试资源管理、测试技术管理以及测试风险管理。

第 7 章主要介绍软件测试自动化以及常用的自动化测试工具。

第 8 章将通过一个五子棋游戏介绍软件测试实例。

本书由多年从事软件测试课程教学工作的北京培黎职业学院的牛红副教授、刘卫宏副教授和北京联想利泰软件有限公司的唐国平高级工程师合作编写而成。其中，第 1~4 章由牛红、唐国平编写，第 5~8 章由刘卫宏、唐国平编写，全书由牛红统稿。在本书的编写过程中，北京培黎职业学院的关仲和副院长给予了大力支持和帮助，计算机系主任尤克教授提出了宝贵意见，付强、闫树涛、何建成为编写团队提供了帮助，常敏慧、王辉、何元娇为本书的编写提出了很好的建议。在此，向所有关心和支持本书出版的同志表示衷心的感谢！

由于作者水平有限，书中难免存在不足之处，恳请广大读者批评指正。

编 者

# 目 录

## 前言

<b>第 1 章 软件测试概述</b> .....	1	<b>第 4 章 软件测试技术和方法</b> .....	37
1.1 什么是软件测试.....	1	4.1 静态测试和动态测试.....	37
1.2 软件测试的目的.....	1	4.1.1 静态测试.....	37
1.3 对软件测试的理解.....	2	4.1.2 动态测试.....	38
1.4 软件测试的原则.....	4	4.2 黑盒和白盒测试概述.....	38
1.5 测试人员的职责.....	5	4.3 黑盒测试技术.....	39
1.6 测试人员的素质要求.....	7	4.3.1 等价类划分.....	40
1.7 软件测试职业岗位分析.....	8	4.3.2 边值分析.....	43
习题.....	9	4.3.3 因果图法.....	46
<b>第 2 章 软件测试基础</b> .....	10	4.3.4 ALAC 测试.....	49
2.1 软件测试模型.....	10	4.3.5 正交实验设计法.....	50
2.1.1 V 模型.....	10	4.3.6 决策表驱动测试.....	52
2.1.2 h 模型.....	11	4.3.7 错误推测法.....	53
2.1.3 模型小结.....	12	4.4 白盒测试技术.....	54
2.2 确认和验证.....	13	4.4.1 白盒测试简介.....	54
2.3 软件测试分类.....	13	4.4.2 程序结构分析测试.....	54
2.4 软件测试流程概述.....	15	4.4.3 逻辑覆盖测试.....	56
习题.....	20	4.4.4 路径分析测试.....	60
<b>第 3 章 软件质量与测试</b> .....	22	4.4.5 程序插装测试.....	61
3.1 软件质量的重要性.....	22	4.4.6 程序变异测试.....	62
3.2 软件质量问题的原因.....	24	4.5 实例设计.....	62
3.3 对软件质量特性的理解.....	25	4.6 测试技术和方法的应用原则与技巧.....	67
3.3.1 软件质量特性的定义.....	25	4.6.1 应用原则.....	67
3.3.2 软件质量特性对于测试人员的意义.....	27	4.6.2 应用技巧.....	67
3.4 基于软件质量特性的测试.....	27	4.6.3 对黑盒、白盒测试方法的总结.....	68
3.4.1 功能性测试.....	27	习题.....	69
3.4.2 可靠性测试.....	28	<b>第 5 章 软件测试类型</b> .....	71
3.4.3 易用性测试.....	29	5.1 单元测试.....	71
3.4.4 兼容性测试.....	31	5.1.1 单元测试的步骤.....	71
3.5 ISO 9000 标准.....	33	5.1.2 单元测试的内容.....	72
3.6 软件能力成熟度模型 (CMM).....	34	5.1.3 单元测试用例的设计.....	74
习题.....	35	5.1.4 单元测试的规则.....	75
		5.2 集成测试.....	75

5.2.1 集成测试概述.....	75	7.5 常用测试工具介绍.....	133
5.2.2 集成测试的策略和方法.....	76	7.5.1 LoadRunner.....	135
5.2.3 集成测试案例.....	80	7.5.2 ClearQuest.....	143
5.2.4 Counter 软件集成测试实例.....	83	7.5.3 单元测试工具 JUnit.....	154
5.3 确认测试.....	89	习题 .....	162
5.3.1 确认测试概述.....	89	<b>第 8 章 单机版五子棋游戏测试实例 .....</b>	<b>164</b>
5.3.2 确认测试策略.....	91	8.1 五子棋游戏简介.....	164
5.3.3 确认测试的设计方法.....	91	8.2 单机版五子棋游戏系统介绍.....	165
5.3.4 确认测试实践.....	94	8.2.1 系统概述.....	165
5.3.5 确认测试的其他有关内容.....	95	8.2.2 系统运行环境.....	165
5.3.6 确认测试实例.....	96	8.2.3 功能需求描述.....	165
5.4 系统测试.....	100	8.3 单机版五子棋游戏的测试要求.....	167
5.4.1 系统测试概述.....	100	8.3.1 测试范围.....	167
5.4.2 系统测试内容.....	100	8.3.2 测试任务.....	167
5.4.3 系统测试的技术与工具.....	102	8.3.3 测试清单.....	168
5.4.4 系统测试实例.....	103	8.4 单元测试案例.....	169
习题 .....	110	8.4.1 单元测试计划.....	170
<b>第 6 章 软件测试管理 .....</b>	<b>111</b>	8.4.2 单元测试的设计与执行.....	171
6.1 测试流程管理.....	111	8.4.3 单元测试报告.....	182
6.1.1 测试准备.....	112	8.5 集成测试案例.....	182
6.1.2 测试实施.....	115	8.5.1 FiveStone 类中各方法的调用关系.....	183
6.1.3 测试结束.....	120	8.5.2 集成测试计划.....	183
6.2 测试资源管理.....	122	8.5.3 集成测试设计.....	184
6.3 测试技术管理.....	124	8.5.4 集成测试报告.....	187
6.4 测试风险管理.....	125	8.6 系统测试案例.....	187
习题 .....	126	8.6.1 系统测试计划.....	187
<b>第 7 章 软件测试自动化和工具 .....</b>	<b>128</b>	8.6.2 系统测试的设计与执行.....	188
7.1 测试自动化和工具概述.....	128	8.6.3 系统测试报告.....	194
7.2 测试脚本技术.....	130	<b>附录 软件测试术语及定义.....</b>	<b>195</b>
7.3 测试比较.....	132	<b>参考文献.....</b>	<b>198</b>
7.4 使用测试自动化和工具要考虑的因素.....	132		

# 第 1 章 软件测试概述

本章主要介绍软件测试方面的基础知识。通过对本章的学习，可以了解软件测试的含义以及测试人员的职责和素质要求，能够对软件测试工作有初步的理解和认识。

本章要点：

- 1) 软件测试的目的。
- 2) 对软件测试的理解。
- 3) 软件测试的原则。
- 4) 测试人员的职责。
- 5) 测试人员的素质要求。
- 6) 软件测试职业岗位分析。

## 1.1 什么是软件测试

目前，业界对软件测试的看法不尽相同，甚至对软件测试的定义也不完全一致。其中，比较公认的定义有以下 3 个。

广义的软件测试定义：贯穿于软件开发各阶段的复查、评估与检验活动，这远远超出了程序测试的范围，可以统称为确认、验证与测试活动（Validation, Verification and Testing）。

狭义的软件测试定义：软件测试是为了发现错误而执行程序的过程。软件测试是根据软件开发各阶段的规格说明和程序的内部结构而精心设计一批测试用例，并利用这些测试用例去运行程序，以发现程序错误的过程。

IEEE 在 1983 年对软件测试的定义：使用人工或自动手段来进行或测定某个系统的过程，其目的在于检验它是否满足规定的需求，或是弄清预期结果与实际结果之间的差别。软件测试以检验是否满足需求为目标。

不必追究到底哪个定义更正确、更科学，但至少可以得出以下结论：

- 1) 软件测试要发现软件的错误。
- 2) 软件测试最终要以软件满足用户需求为目标。

## 1.2 软件测试的目的

最初的软件测试是为了证明程序的正确性，但这只能在有限的情况下证明程序的正确性，对于今天的软件规模和开发环境来讲，证明程序的正确性实际上是不可能的。所以，测试人员不可能在测试报告中描述：“经过测试，该程序是没有问题的！”其他人员也不要寄希望于测试人员来担保程序或软件产品是完全正确的。

软件测试最直接的目的是发现软件中的缺陷，包括需求、设计方面的缺陷和程序中包含

的缺陷（BUG）。这里所说的缺陷是一种泛指，它可以指软件功能的错误，也可以指性能低下、易用性差以及其他软件产品中的缺陷等。

测试人员总是先假设程序中存在缺陷，再通过执行测试活动来发现并最终改正缺陷。理解测试的目的是个很重要的意识问题。如果说测试的目的是为了说明程序中没有缺陷，那么测试人员就会向这个目标靠拢，因而会下意识地选用一些不易暴露错误的测试示例，这样的测试是虚假的、没有意义的。

软件测试最终的目的是检查软件是否满足用户的需求，其中包括用户的隐含需求和潜在需求。只有满足用户需求的软件才能成为“好”的软件产品，才能得到用户的认可和好评。

Glen Myers 曾提出关于测试目标的规则：

- 1) 测试是一个为了寻找错误而运行程序的过程。
- 2) 一个好的测试用例是指很可能找到迄今为止尚未发现的错误的用例。
- 3) 一个成功的测试是指揭示了迄今为止尚未发现的错误的测试。

以上 3 条规则表明了两种涵义：一是软件测试的直接目的，即发现软件中的错误；二是测试工作的职责就是要发现软件中的错误。

发现软件中的错误和检查软件是否满足用户的需求，这两点都是软件测试工作的目的，但是否就局限于此呢？当然不是。

测试工作要讲究工作效率，需要尽早地、不断地进行测试。越早发现缺陷，越能降低软件的质量风险，减少修复缺陷的花费；测试工作越充分，软件中的 BUG 会越少，软件的质量会越高，软件的维护费用就会越少。所以，测试一定要尽早和充分。

## 1.3 对软件测试的理解

很多人对软件测试的认识存有误区，而且对软件测试的看法也不尽相同，不同的角色、不同的文化环境使很多人对软件测试存有争议。在此探讨以下几个问题。

**问题 1：需求→设计→编码→测试，软件测试工作在编码完成后才开始？**

实际上，软件测试工作要贯穿于整个软件产品的生命周期。一方面，软件测试实施前有测试计划、测试用例的设计和实现等，而且其效果会直接影响后期测试实施的效率和进度。因此，早在项目立项和策划阶段，软件测试工作就应该开始了。另一方面，软件测试越早进行越好，因为越早发现 BUG，其造成的影响和修改的代价就越小。而且，软件测试工作不仅是针对程序，还包括对软件的需求、设计等进行同行评审（这些属于广义上的测试范畴）。

**问题 2：软件测试能否确保软件质量？**

这一点至今仍存在争议，因为很多人把测试人员和质量保证人员（QA）等同起来，但是从软件测试的目的来讲，的确没有说明软件测试能确保软件质量。目前的理解是，软件测试本身不能确保软件质量，但它却是保证软件质量的重要而关键的技术手段，因为软件经过测试后，质量一般都有提升。

**问题 3：软件发布后出现了质量问题，这是测试人员的责任？**

软件发布后出现了问题，尤其是遭到用户的抱怨或投诉，测试人员一般要负有一定的责任，但是软件测试并不能 100%地发现软件中存在的所有缺陷，即使测试已投入了充分的资源。高质量的软件是开发出来的，并不是测试出来的。



**问题 4: 软件测试工作到底难不难?**

绝大多数公司的测试人员都比开发人员技术水平差,有些测试人员甚至只有简单的计算机基础,有些编码人员技术水平较低,所以可能会转为测试人员。其实,这些事实既是无可奈何的,又是无可厚非的。一方面,很多具体的测试实施工作并不需要具有很高的技术知识,只需严格执行测试计划和测试用例,然后认真记录测试结果就可以了;另一方面,测试用例的设计与开发、测试工作的管理都需要测试人员具有深厚的技术功底和丰富的测试经验,从这一点上讲,测试效果和软件质量是与测试人员的技术经验水平成正比的。另外,测试人员在测试工作中要及时、有效地进行缺陷分析,这一点对项目的益处是不言而喻的,而缺陷分析工作来源于测试人员的缺陷分析技术。所以说,高质量的软件测试工作是很困难的。

**问题 5: 软件测试工作是否也像设计工作那样具有开拓性和创新性?**

软件测试工作并不是简单地运行程序和执行测试用例或发现程序中的 BUG 即可,很多测试工作都需要测试人员的技术能力,需要测试人员具有开拓性和创新性。例如:

- 1) 设计和开发大量优秀的测试用例。
- 2) 分析和把握用户的需求,制订出高“性价比”的测试策略。
- 3) 拥有切实可行的测试计划和 BUG 跟踪。
- 4) 不断学习和发明测试的新技术和方法。

由此可见,软件测试工作具有很高的开拓性和创新性。

**问题 6: 软件测试对于软件开发是建设性的,还是摧毁性的?**

测试相对于设计和编码阶段的审查和评审工作来说,是一种“事后诸葛”,不断提交大量且较为严重的缺陷会令开发人员讨厌或心烦,甚至会使项目进度处于尴尬的局面。一遍又一遍的回归测试和新的 BUG 列表,会摧毁开发人员和管理者对软件质量的信心。

软件测试虽然导致项目成本大量的增加,但暴露出的是项目开发阶段中工作的不足,这样有利于及时认识到哪些工作需要改进和调整,测试工作毕竟可使软件质量逐渐提高,缺陷越来越少。所以,软件测试对软件开发是有建设性意义的。

**问题 7: 软件测试是测试人员的事,与开发人员无关?**

为了减少相互影响,一般要求开发和测试相对独立,但这只是分工上的不同。开发和测试是软件项目相辅相成的两个过程,人员之间的交流、协作和配合是提高整体效率的重要因素,而且在实际操作中,开发人员也会有一些测试工作。

**问题 8: 软件测试与调试工作类似?**

很多开发人员会把自己的调试工作认为就是测试,认为是自己在测试自己的程序,而且觉得测试与调试本身也没什么差别。其实,软件测试与调试是完全不同的工作,具体区别如下:

- 1) 目的。软件测试的目的是尽可能多地发现程序中的错误,而调试的目的是确定错误的原因和位置,并改正错误,调试也被称为纠错。
- 2) 工作性质。测试是测试人员针对被测软件产品执行的检查和确认,属于测试范畴;而调试是开发人员在发现程序中的 BUG 后开始的发现和改正 BUG 的工作,属于开发范畴。
- 3) 内容与方法。测试是按照计划执行的,需要测试计划、设计开发、测试执行和测试评估等阶段;而调试只是针对程序中出现 BUG 的开发工作,是“BUG 驱动”类型的工作。



## 1.4 软件测试的原则

任何工作都是讲究原则的，软件测试工作更是如此。由于软件测试工作的特殊性（目的是为了发现错误），其原则很多，下面从两方面进行讲解。

### 1. 测试技术和策略方面

1) 测试工作要尽可能地找出关键性的错误，因为这些错误很可能会限制用户使用此软件产品完成工作的能力，从而直接影响客户对质量的评价。

2) 把 Pareto 原则应用于软件测试。因为测试有群集现象，所以适用于 Pareto 原则。事实证明，一段程序中存在的错误数与其中已发现的错误数是成正比的。一般情况下，在设计、实现阶段的复审和测试工作能够发现和避免 80% 的 BUG，而系统测试又能找出其余 BUG 中的 80%，最后的 4% 的 BUG 可能只有在用户大范围、长时间的使用后才会曝露出来。因此测试只能保证尽可能多地发现错误，无法保证能够发现所有错误。

3) 100% 测试覆盖率。只有达到 100% 测试覆盖率才能最大限度地降低软件的质量风险，这就要求在实际测试过程中，要 100% 地执行设计的测试用例。

4) 所有的测试都应追溯到用户需求。软件测试的最终目的就是验证软件是否满足用户的需求，所以测试工作要以用户需求为根本出发点，以用户的实际应用环境为判断依据。

5) 应当尽早地、不断地进行软件测试。尽早测试可以保证项目的开发进度，减少项目开销；不断进行测试可以使测试更充分，可以更多、更有效地发现软件中的缺陷。

6) 总假定程序是有错误的。这一原则非常有效，它涉及人的心理问题。因为只有抱着“程序有错”的心理，才能更精心地设计出测试用例，才能发现更多的缺陷；否则，测试人员想当然地认为程序没有问题，那么主观心理必然会使测试不充分。

7) 彻底检查和仔细分析每一个测试结果。在软件测试过程中，测试人员有时会忽略对测试结果的检查和分析。不彻底检查会使某些“隐藏”的 BUG 没有被发现，不仔细分析会使开发人员很难对 BUG 进行定位和修复。

8) 不断提高测试策略和技巧。测试的策略和技巧包含很多方面。例如，测试一般讲究从“小规模”开始，逐步转向“大规模”；穷举测试是不可能的，所以要采用科学的方法，同时根据用户的实际使用情况综合分析测试内容和步骤；测试用例的设计与选择既要尽可能地发现程序中的错误，又要满足测试策略。

### 2. 测试管理方面

1) 测试必须是有计划、有组织、有准备的，其中包括：确定测试任务、时间、人员职责及分工、资源设备、方法与工具、输入和输出准则等。

2) 严格执行测试计划并及时进行修订。严格执行测试计划，意味着要严格遵守测试的工作时间表及具体安排，排除测试的随意性，如果实际情况发生了变化，则要及时对测试计划进行修订。这样测试计划才能更好地达到测试和管理的目的。

3) 有效的 BUG 跟踪和管理。建立 BUG 管理流程，使软件中的 BUG 能够按统一的标准和流程得到解决。另外，测试人员要对 BUG 进行坚决跟踪和监控，使软件质量能够得到保证和控制，为项目管理人员提供量化的软件质量状态。

4) 由独立的第三方来完成测试工作。尤其是确认测试和系统测试，要由独立的测试机

构完成，以达到更好的测试效果。对于单元测试经常是由开发人员自己完成，但是开发人员要避免测试自己开发的程序。另外，为了保证测试工作能够高效执行，需要有很好的团队交流机制，以及快速的反应和反馈能力。

那么，到底什么样的测试才算是好的测试呢？一般公认至少具有以下5个属性：

- ① 发现错误的可能性很高。
- ② 不冗余。
- ③ 是“最佳品种”。
- ④ 不会太简单，也不会太复杂。
- ⑤ 揭示了迄今为止尚未发现的错误。

总体来说，就是保证测试的有效性和测试的恰到好处。

## 1.5 测试人员的职责

测试人员的职责从本质上讲就是做好项目的测试工作，达到软件测试的目的。

测试经理和测试主管是测试小组对外（主要是项目组）的接口，对内负责组员的工作安排、工作检查和进度管理，同时也要承担重要项目的测试工作。

### 1. 测试经理和测试主管的职责

一般来说，测试经理和测试主管都重视测试技术工作，并经常与测试人员共同完成测试任务。对于测试经理和测试主管来讲，最重要的工作是提供和协调测试资源，帮助测试人员提高测试工作的质量，其中主要体现在以下几个方面：

1) 招聘合适的测试人员。这是首要的，但往往事与愿违。一方面是由于很多人对软件测试有偏见，觉得水平和能力较差的人才做测试工作，所以很难招聘到高水平的测试工程师；另一方面，测试经理和测试主管在面试时，往往忽略了其他的重要因素，如是否有较强的工作责任心，是否乐于做测试工作等。

当然，招聘测试人员还要结合现有的实际情况，考虑需要何种类型的员工，是活力四射的，还是稳重诚实的。

2) 建立测试技术模型和培训机制。软件测试本身就是一种技术，但很可惜，目前国内软件测试水平还很有限，测试技术和管理书籍很少。那么，在测试组织内部建立有效的测试技术模型，可以弥补这方面的不足。测试技术模型是根据组织实际情况建立起来的一套测试技术体系，包括测试技术和测试流程两大方面，还有就是测试方法和经验的共享。

当然，测试技术模型需要不断地改进，需要测试人员具有学习和创新能力。在测试技术模型建立并改进后，需要进行推广和实施，最主要的方法是组织各种培训，包括正式培训、技术交流研讨会和文档共享等形式。因为软件测试涉及的技术很广，所以需要进行技术分工，使测试人员各攻所长，然后通过培训提高整个测试组织的水平。因此，建立有效、实用的培训机制非常重要，培训机制可以根据测试组织的实际情况建立，如定期或不定期，指定培训人选或大家毛遂自荐等。

3) 定期与测试人员进行正式交谈。与员工进行沟通的好处很多，如可以了解员工的工作情况和工作心态，也可以为员工提供意见和建议，还可以增进与员工之间的感情等。

但是，仍然有很多测试经理和测试主管因为工作繁忙而忽略了这件重要的工作。所以，

建议测试经理和测试主管们给自己制订一个定期计划，每周或几周内与所有员工进行单独而正式的交谈，一旦形成了这个习惯，就会发现受益匪浅。

4) 对员工工作的充分信任。在做了几年的测试管理工作后，测试经理和测试主管会发现，很多骨干员工的测试技术和水平要强于自己，这是很正常的现象。基于此原因，测试经理和测试主管在把工作分配给员工后，就要充分信任测试人员的能力，尤其是测试负责人，而自己主要应做好宏观的管理和测试支持工作。同时，还要鼓励员工试用更科学的测试技术、方法或工具，并进行知识共享。

5) 以员工期待的方式善待员工。测试工作量大，又容易被人误解。如果测试经理和测试主管能以恰到好处的方式对待员工，那么就会对测试人员的心理起到良好的调和作用。另外，测试人员的工作风格也会有所差异，所以布置和指导工作方式也要因人而异。对员工的奖励和批评也是如此，物质奖励或惩罚并不是万能的，而“一分钟夸奖”往往能取得事半功倍的效果。

6) 评价实事求是、以事论事。无论是测试工作，还是测试人员本身，都需要不断地进行评价和总结，以便后续改进，并且使所有员工都能共享经验和教训。评价的原则有两点，即实事求是和以事论事。实事求是需要测试人员有勇气揭示自己的不足，以事论事则需要测试经理和测试主管不能把工作的缺陷与员工自身联系在一起，否则会影响评价工作的正常开展。

评价也包括测试经理和测试主管要勇于承认自己的错误。“金无足赤，人无完人”，否认和忽略自己的错误属不明智之举。若想改进不足，需要多倾听员工的意见和建议。

7) 规划和开展测试管理工作。作为软件公司或测试组织的经理，都希望通过软件测试来不断提高软件产品的质量，所以测试组织必须要不断提高技术水平、不断发展，作为测试经理和测试主管，需要每年或每季度规划测试组织工作，为测试组织定制发展目标和愿景。

当前的项目规模越来越大，产品交付市场的时间点越提前越好。所以，软件测试工作作为软件交付的最后一个环节，其质量风险是不容忽视的。因此在测试实施前，就要做好充分的计划，采取高效的测试策略并进行评估，使项目中严重的缺陷尽早发现，以降低整个项目的开发风险。

测试经理和测试主管可能每天都会有很多的时间花在开各种会议、参加培训、接打电话或收发邮件上，但是作为测试负责人，应该把更多的精力放在提高测试工作效果和避免测试风险的发生上，而不能只完成属于自己的部分工作。

## 2. 测试工程师的职责

作为软件测试工程师，要能够完成所有的测试工作，如从测试计划到测试总结，从软件测试活动到与测试相关的其他活动等，具体说来主要包括以下工作：

1) 制订测试计划。在项目做项目策划时，测试的计划工作即开始，这个时期测试工程师的职责不仅是完成一篇《测试计划》，而是与项目经理一起规划好测试活动所涉及的资源、花费和关键时间点，以及制订测试风险管理计划等。另外，每一个测试阶段的测试计划还要包括具体的测试内容、输入和输出准则以及具体测试活动的安排等。简而言之，测试工程师要制订测试计划，为以后的测试实施活动做好充分的准备。

2) 设计与编写测试用例。制订测试计划是对测试活动的资源、花费等的准备，而具体测试执行的内容和方法则需要测试工程师编写测试用例文档（即《测试说明》）。测试用例的设计与编写是测试工程师技术与经验的综合体现，是测试实施前最重要、最关键的活动。

3) 实施测试。测试工作的实施是测试工程师最重要的测试活动，也是软件开发过程中一个很重要的步骤。实施测试需要测试工程师既要保证测试的充分，又要讲究工作效率和测试策略与方法。因为测试工作具有风险性，所以对测试负责人来说，实施测试非常具有挑战意义。

4) BUG 跟踪。有效的 BUG 跟踪是提高测试效率、改进产品质量和保证项目进度的重要策略之一。测试工程师在测试过程中，不仅是发现和验证软件的 BUG，而且要对其进行跟踪，使其得到尽快修改和控制。

5) 测试报告与总结。软件测试的结束并不表示整个测试工作的结束，测试工程师需要把软件测试的结果经过统计和分析汇报出来，一方面使项目管理者能够清晰地了解软件当前的质量，另一方面也能为项目以后的改进和测试工作的改进提供重要的参考资料。

6) 其他软件工程活动，包括需求、概要设计的同行评审、项目会议以及相关产品的确认和验证活动。

## 1.6 测试人员的素质要求

人是测试工作中最有价值也是最重要的资源，没有一个合格的、积极的测试小组，测试就不可能很好地实现。然而，在软件企业中有一种非常普遍的习惯，那就是让经验较少的新手、没有效率的开发者或不适合于其他工作的人去做测试工作。这绝对是一种目光短浅的行为，因为对一个系统进行有效地测试所需要的技能绝对不比进行软件开发需要的少。事实上，测试人员的技术与经验与软件质量和测试工作的效率有非常大的关系。

### 1. 测试人员的技术素质要求

测试人员的技术要求与开发人员相比要更全面，因为测试人员要有软件开发技术与经验，还要具有软件测试技术以及软件工程方面的能力，当然也包括软件应用领域的行业知识。

1) 软件开发技术。做软件测试工作当然要掌握一定程度的测试软件应用的开发技术。一方面，对开发技术比较了解能使测试人员更有效地发现程序中的 BUG，并进行有效地分析，以帮助开发人员更快地进行定位和修改；另一方面，很多测试工作本身需要编码技术。例如，单元测试和集成测试经常需要构造驱动和桩程序，如果测试人员的开发技术水平很低，就根本无法胜任这些测试工作。

另外，软件测试经常需要自主开发测试工具，一些测试中心的实用工具都是测试人员开发出来的。

2) 软件测试技术。软件测试本身就是一门学问，它是软件测试人员的“独门专业”。只有掌握了软件测试技术，测试工作才能更有效地进行；只有掌握了软件测试技术，才能更早、更快、更有效地找出软件中的缺陷；只有掌握了软件测试技术，才能用更科学的方法验证软件是否满足用户的需求。

从目前的软件测试技术来说，包括业界公认的测试技术（如等价类划分、因果图分析等黑盒测试方法），也包括软件测试中心自己总结出的一些通用的测试方法，还包括不同类型项目的技术总结。

所以，测试人员除了要掌握和了解测试行业本身需要的测试技术外，还要针对不同项目的需求掌握一些通用的测试方法和技巧，而且要不断地总结经验，共享技术与经验，以使测试中心的整体技术水平得到迅速提升。

3) 软件工程方面的能力。软件测试活动几乎贯穿于软件开发的全过程,那么就要求测试人员,尤其测试负责人,要对软件工程有较深的认识。例如,某软件设计中心对软件研发管理非常重视,软件工程活动采用能力成熟度模型(CMM),这就要求测试人员要对软件研发和测试流程以及规范非常重视,而且还要灵活地运用,使软件测试工作既严格遵从项目计划,又符合软件开发和测试的流程规范。

4) 行业知识。既然非常重视用户需求,并把软件质量的重点放在了用户的满意程度上,那么,了解用户需求就显得尤为重要。目前,软件越来越多地应用在了企业管理和教育等行业上,因此一定要了解软件应用领域的行业知识,如 ERP、CRM 等业务流程以及教育理论与实现方式等。

## 2. 测试人员的非技术素质要求

1) 沟通能力。一名优秀的测试人员必须具有与技术(开发人员)和非技术人员(客户、管理人员)交流的能力,既要和用户谈得来,又能同开发人员说得上话。不幸的是,这两类人没有共同语言。和用户谈话的重点必须放在系统可以正确地处理什么和不可以处理什么上;而和开发人员谈相同的话题时,就必须将这些话重新组织,以另一种方式表达出来。测试小组的成员必须能够同用户和开发者进行沟通。

2) 移情能力。和系统开发有关的所有人员都处在一种既关心又担心的状态中。用户担心将来使用一个不符合自己要求的系统,开发者则担心由于系统不正确而导致不得不重新开发整个系统,管理部门则担心这个系统突然崩溃而导致声誉受损。测试者必须和每一类人打交道,因此需要测试小组的成员对他们每个人都具有足够的理解和同情,具备了这种能力可以将测试人员与相关人员之间的冲突和对抗降到最低程度。

3) 自信心。开发者指责测试者出了错是常有的事,测试者必须对自己的观点有足够的自信心。

4) 幽默感。与开发人员意见相左、僵持不下时,一句幽默的点评可以有效缓和气氛。

5) 外交能力。机智、老练的外交手法有助于维护与开发人员的协作关系。

6) 超强的记忆力。一个优秀的测试人员应该有能力将以前曾经遇到过的类似的错误从记忆深处挖掘出来,这一能力在测试过程中的价值是无法衡量的,因为许多新出现的问题和已经发现的问题相差无几。

7) 足够的耐心。很多软件测试工作都需要有足够多的耐心。有时需要花费惊人的时间去分离、识别和分析一个错误。

8) 怀疑精神。可以预料,开发人员会尽他们最大的努力将所有的错误“绕”过去,测试人员必须听每个人的说明,但必须保持怀疑直到看过以后。

9) 自我督促。做测试工作的人很容易变得懒散。只有那些具有自我督促能力的人,才能做到每天都能正常地工作。

10) 洞察力。一个好的测试工程师要具有“测试是为了破坏”的观点,同时还要有捕获用户观点的能力,以及强烈的质量追求和对细节的关注能力。拥有高风险区的判断能力可以将有限的测试针对到重点环节。

## 1.7 软件测试职业岗位分析

随着软件业的迅猛发展,软件产品的质量控制与管理正逐渐成为软件企业生存与发展的

关键。为保证软件出厂时的“健康状态”，几乎所有的IT企业在软件产品发布前，都要进行大量的软件测试，以对软件质量进行严格控制。这也引发了软件测试人才的走俏。目前软件测试职业的岗位基本具有以下特点：

1) 工作起点高。小公司对软件质量的重视程度没有大公司高，一般较大规模的软件企业需要大量的测试工程师。

2) 发展空间大。进入企业后，随着业务越来越熟练，有可能晋升为软件测试主管、项目主管、行业专家、自动化测试专家、QA、需求分析师、客户服务与支持等。

3) 职业寿命长。这个岗位是针对程序员来讲的，软件测试工程师经常使用专业测试工具，经验价值会逐步上升，对公司的业务也会越来越精、越来越“资深”。另外，这个岗位对年龄没有限制，工作更稳定。

4) 薪水待遇好。真正资深的测试工程师凤毛麟角，所以薪酬空间很大，甚至超出相同服务年限的软件开发人员的薪资水平。

## 习 题

### 一、选择题

- 软件测试的目的是（ ）。
  - 表明软件的正确性
  - 评价软件质量
  - 尽可能发现软件中的错误
  - 判定软件是否合格
- 下面关于软件测试的说法，错误的是（ ）。
  - 软件测试是程序测试
  - 软件测试贯穿于软件定义和开发的整个期间
  - 需求规格说明和设计规格说明都是软件测试的对象
  - 程序是软件测试的对象
- 某软件公司在招聘软件评测师时，应聘者甲向公司作如下保证：
  - 经过自己测试的软件今后不会再出现问题；
  - 在工作中对所有程序员一视同仁，不会因为某个程序员编写的程序问题多，就重点审查该程序，以免不利于团结；
  - 承诺不需要其他人员，自己就可以独立进行测试工作；
  - 发扬咬定青山不放松的精神，不把所有问题都找出来，决不罢休。
 你认为应聘者甲的保证（ ）。
  - ①、④是正确的
  - ②是正确的
  - 都是正确的
  - 都不正确

### 二、简答题

- 简述软件测试的定义。
- 简述软件测试的目的。
- 简述软件测试的原则。
- 谈谈你对今后从事软件测试职业的打算。
- 简述软件测试工程师应具备的素质。

## 第 2 章 软件测试基础

通过第 1 章的学习，可以对软件测试有一个大致的认识，但对一些基本概念，如软件测试的模型、分类等并不十分清楚。本章将讨论几个重要的软件测试过程模型，从不同的角度介绍软件测试的分类，最后介绍软件测试的流程。在测试分类中介绍的一些主要的测试方法和技术，如白盒测试、黑盒测试、单元测试等是本书的重点内容，将在后续章节详细介绍。

本章要点：

- 1) 软件测试模型。
- 2) 确认和验证的关系。
- 3) 软件测试的分类。
- 4) 软件测试的流程。

### 2.1 软件测试模型

随着测试过程管理的不断发展，测试人员通过大量的实践，总结出了很多很好的测试过程模型。这些模型将测试活动进行了抽象，并与开发活动进行了有机的结合，是测试过程管理的重要依据。

#### 2.1.1 V 模型

V 模型主要应用于项目的测试工作中，它强调了测试阶段与开发阶段的对应关系以及测试工作的及早准备和进行。V 模型的示意图如图 2-1 所示。

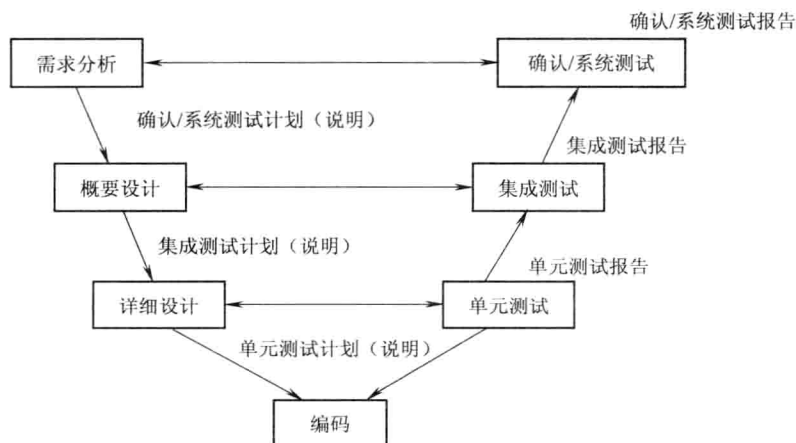


图 2-1 V 模型示意图



V模型揭示了软件测试活动分层和分阶段的本质特性。例如，集成测试对应概要设计，那么，集成测试计划和集成测试说明文档的编写就可以在概要设计阶段开始编写，只要在集成测试实施前完成即可。

V模型还有一点意义：在需求分析阶段编写测试用例，可以发现需求文档本身的缺陷，这样就能尽早把需求的缺陷消除，避免使缺陷遗留到下一个阶段中。同时，在概要设计阶段编写集成测试用例也会间接地提高软件设计的质量。

### 2.1.2 h模型

V模型从某种意义上讲，把软件的开发视为需求、设计、编码等一系列的串行活动。而事实上，虽然这些活动之间存在着相互牵制的关系，但在大部分时间里，它们是相互独立的，是可以并发进行的。虽然软件开发期望有清晰的需求、设计和编码等阶段，但实践告诉我们，严格的阶段之分只是一种理想情况。很多测试工作并没有严格的先后顺序，只要测试条件满足，就可以进行测试。各个不同层次之间的测试除了简单的时间上的先后关系外，还存在着触发、反复、迭代和增量关系。另外，V模型并没有表示出测试流程的完整性。测试流程大致分为两类活动，一类是测试准备活动，包括测试需求分析、测试计划、测试用例设计与实现、测试环境准备、培训学习等；另一类是测试执行活动，包括测试实施、BUG提交与验证等。h模型作为一个新的理念，包含了以上不足。

h模型的示意图如图2-2所示。

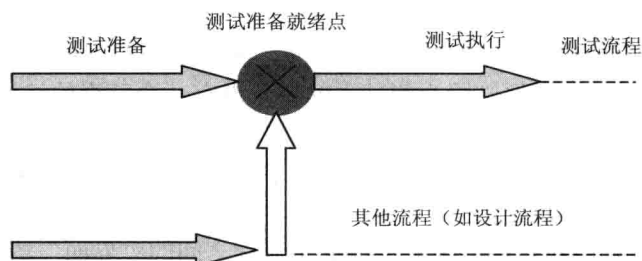


图 2-2 h模型示意图

这个示意图仅仅演示了在整个生产周期中，某个测试层次上的一次测试“微循环”。图中的“其他流程”可以是任意的开发流程，如设计流程和编码流程，也可以是其他的非开发流程，甚至是测试流程本身。向上的空心箭头表示，在某个时间点，由于“其他流程”的进展而（由于先后关系）引发或者（由于因果关系）触发了测试就绪点，此时只要测试准备活动完成，测试执行活动就可以（或者说是需要）进行。概括来说，h模型揭示了以下几点内容：

- 1) 软件测试不仅指测试的执行，还包括很多其他的活动。
- 2) 软件测试是一个独立的过程，贯穿产品的整个周期，与其他流程并发进行。
- 3) 软件测试要尽早准备，尽早执行。
- 4) 软件测试根据被测物的不同是分层次的。不同层次的测试活动可以按照某个次序先后进行，也可以反复进行。

由上可以看出，软件测试是一个独立的流程，贯穿于整个产品周期，与其他流程并发进行。当某个测试时间点就绪时，软件测试即从测试准备阶段进入测试执行阶段。h模型兼顾