

21世纪
计算机
应用
技术



21世纪高校计算机应用技术系列规划教材
谭浩强 主编

软件测试技术

李庆义 岳俊梅 王爱乐 等编著



22

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE



21 世纪高校计算机应用技术系列规划教材
谭浩强 主编

软件测试技术

李庆义 岳俊梅 王爱乐 等编著

内 容 简 介

本书主要介绍软件适用的测试技术。全书内容共分为3部分。第1部分为基本概念、测试理论的背景及发展,用真实的案例引出软件测试的重要性,简要分析了当前软件测试技术的现状,指导当前即将毕业的大学生对测试的理解。第2部分介绍软件测试的程序分析技术和测试技术,软件测试的方法和策略,包括单元测试、集成测试、系统测试、验收测试、设计测试等内容,从不同的角度分析软件测试的方法,分析软件业在测试方面的研究成果,并总结测试的基本原则和经验。第3部分介绍两种测试工具软件——基于Windows的WinRunner和服务器负载测试软件WAS。

本书适合高校计算机专业的学生使用。不仅可作为计算机本科低年级的教学参考书和软件测试的培训教材,还可作为高职高专院校计算机软件专业的软件测试课程的教材。

图书在版编目(CIP)数据

软件测试技术/李庆义等编著. —北京:中国铁道出版社, 2006. 10

(21世纪高校计算机应用技术系列规划教材. 高职高专系列)

ISBN 7-113-07054-X

I. 软... II. 李... III. ①软件—测试—高等学校: 技术学校—教材 IV. TP311.5

中国版本图书馆CIP数据核字(2006)第129111号

书 名: 软件测试技术

作 者: 李庆义 岳俊梅 王爱乐 等

出版发行: 中国铁道出版社(100054,北京市宣武区右安门西街8号)

策划编辑: 严晓舟 秦绪好

责任编辑: 苏 茜 崔晓静

特邀编辑: 薛秋沛

封面设计: 薛 为

封面制作: 白 雪

责任校对: 郑 楠

印 刷: 河北省遵化市胶印厂

开 本: 787×1092 1/16 印张: 21.5 字数: 497千

版 本: 2006年11月第1版 2006年11月第1次印刷

印 数: 1~5 000册

书 号: ISBN 7-113-07054-X/TP·1806

定 价: 28.00元

版权所有 侵权必究

本书封面贴有中国铁道出版社激光防伪标签,无标签者不得销售

凡购买铁道版的图书,如有缺页、倒页、脱页者,请与本社计算机图书批销部调换。

21 世纪高校计算机应用技术系列规划教材

主 任：谭浩强

副主任：陈维兴 严晓舟

委 员：（按姓氏字母先后为序）

安淑芝	安志远	陈志泊	韩 劼	侯冬梅
李 宁	李雁翎	刘宇君	林成春	秦建中
秦绪好	曲建民	尚晓航	邵丽萍	宋 红
宋金珂	王兴玲	魏善沛	熊伟建	薛淑斌
张 玲	赵乃真	訾秀玲		

21 世纪是信息技术高度发展且得到广泛应用的时代,信息技术从多方面改变着人类的生活、工作和思维方式。每一个人都应当学习信息技术、应用信息技术。人们平常所说的计算机教育其内涵实际上已经发展为信息技术教育,内容主要包括计算机和网络的基本知识及应用。

对多数人来说,学习计算机的目的是为了利用这个现代化工具工作或处理面临的各种问题,使自己能够跟上时代前进的步伐,同时在学习的过程中努力培养自己的信息素养,使自己具有信息时代所要求的科学素质,站在信息技术发展和应用的前列,推动我国信息技术的发展。

学习计算机课程有两种不同的方法:一是从理论入手;一是从实际应用入手。不同的人有不同的学习内容和学习方法。大学生中的多数人将来是各行各业中的计算机应用人才。对他们来说,不仅需要“知道什么”,更重要的是“会做什么”。因此,在学习过程中要以应用为目的,注重培养应用能力,大力加强实践环节,激励创新意识。

根据实际教学的需要,我们组织编写了这套“21 世纪高校计算机应用技术系列规划教材”。顾名思义,这套教材的特点是突出应用技术,面向实际应用。在选材上,根据实际应用的需要决定内容的取舍,坚决舍弃那些现在用不到、将来也用不到的内容。在叙述方法上,采取“提出问题——解决问题——归纳分析”的三部曲,这种从实际到理论、从具体到抽象、从个别到一般的方法,符合人们的认知规律,且在实践过程中已取得了很好的效果。

本套教材采取模块化的结构,根据需要确定一批书目,提供了一个课程菜单供各校选用,以后可根据信息技术的发展和教学的需要,不断地补充和调整。我们的指导思想是面向实际、面向应用、面向对象。只有这样,才能比较灵活地满足不同学校、不同专业的需要。在此,希望各校的老师把你们的要求反映给我们,我们将会尽最大努力满足大家的要求。

本套教材可以作为大学计算机应用技术课程的教材以及高职高专、成人高校和面向社会的培训班的教材,也可作为学习计算机的自学教材。

由于全国各地区、各高等院校的情况不同,因此需要有不同的特点的教材以满足不同学校、不同专业教学的需要,尤其是高职高专教育发展迅速,不能照搬普通高校的教材和教学方法,必须要针对它们的特点组织教材和教学。因此,我们在原有基础上,对这套教材做了进一步的规划。

本套教材包括以下 5 个系列:

- 基础教育系列
- 高职高专系列
- 实训教程系列
- 案例汇编系列
- 试题汇编系列

其中基础教育系列是面对应用型高校的教材，对象是普通高校的应用性专业的本科学生。高职高专系列是面向两年制或三年制的高职高专院校的学生的，突出实用技术和应用技能，不涉及过多的理论和概念，强调实践环节，学以致用。后面3个系列是辅助性的教材和参考书，可供应用型本科和高职学生选用。

本套教材自2003年出版以来，已出版了70多种，受到了许多高校师生的欢迎，其中有多种教材被国家教育部评为普通高等教育“十一五”国家级规划教材。《计算机应用基础》一书出版3年内发行了45万多册。这表示了读者和社会对本系列教材的充分肯定，对我们是有力的鞭策。

本套教材由浩强创作室与中国铁道出版社共同策划，选择有丰富教学经验的普通高校老师和高职高专院校的老师编写。中国铁道出版社以很高的热情和效率组织了这套教材的出版工作。在组织编写及出版的过程中，得到全国高等院校计算机基础教育研究会和各高等院校老师的热情鼓励和支持，对此谨表衷心的感谢。

本套教材如有不足之处，请各位专家、老师和广大读者不吝指正。希望通过本套教材的不断完善和出版，为我国计算机教育事业的发展和人才培养做出更大贡献。

全国高等院校计算机基础教育研究会会长
“21世纪高校计算机应用技术系列规划教材”丛书主编

谭浩强

前言

FOREWORD

本书是 21 世纪高校计算机应用技术系列教材之一。它在编写过程中贯彻 21 世纪高等教育应用型人才培养的方针,遵循“必要、适度、够用”的原则,注意在基础知识的建立和基本技能的应用上做到难易适度、灵活运用。

软件测试技术是一门新型的、独特的专门学科技术,它是软件项目从最初设计到最终使用的纽带和桥梁。大家应改变几种片面的观点:一是软件测试技术比较简单;二是测试不需要太多的工作,只要会编程,就会测试。其实,软件测试好入门,但测试成功或做得比较全面则是很困难的,它是有一定难度的事情。本书编写的出发点,就是教给读者软件测试的一些基础知识,提供给读者一个逐步获得成功测试的方法。

项目经理或部门单位的主管认为刚刚毕业的大学生应当具备软件测试的能力,事实上并非如此。从实践上讲,刚毕业的计算机专业的大学生缺乏项目工程经验,不了解软件项目开发的真实过程。从教学上讲,我国的计算机教育与社会的需求之间还存在一定的差距,需要彼此的沟通与联系。从工程上讲,我们的大学生距离技术人员、软件测试员、程序员都是有一定差距的。

在当前软件产业发展的新形势下,软件测试的地位得到了应有的重视。随着计算机软件技术的教学和改革的发展,软件测试课程已列入教学计划并逐步展开,它将成为软件专业人才知识构架与技能培养的重要组成部分。国内的测试教学刚刚起步,缺乏教学经验,尤其是软件测试方面的教材比较少。本书是作者学习、研究测试技术的知识总结。它系统而全面地阐明了软件测试技术的概念、基本理论知识和适用的技术,为掌握软件测试的基本技能和实际工作的专业技能奠定了良好的基础。

本书的重点是介绍软件测试的一些基本概念和方法,涵盖了信息产业界的大部分测试知识。

软件测试是从软件工程中演化而来的一门学科,作为测试学,我们不能穷尽测试所有的方方面面,也不需要这样做,只要把握测试的核心就可以了。软件测试要求具备一定的程序设计经验和程序分析基础,所以掌握一门语言、了解一定的开发过程是非常必要的。

本书在编写的过程中得到了社会多方面的支持,在此表示由衷的感谢。全书共分为 10 章,第 1 章、第 2 章、第 4 章由李庆义负责完成,第 3 章由岳俊梅完成,第 5 章、第 7 章和附录部分由周贵全完成,第 6 章由武彩宏完成,第 8 章、第 9 章由王爱乐完成,第 10 章由邹俊完成。全书最后由李庆义修改定稿。

由于国内软件测试的针对性教材较少,加之编者水平有限,本书难免有不足之处,敬请广大的读者、老师和专家不吝赐教。

编 者

2006 年 10 月

目录

CONTENTS

第 1 章 软件测试概述	1
1.1 回顾软件测试的发展历史	1
1.1.1 软件缺陷产生的原因	1
1.1.2 软件缺陷事件	3
1.2 软件测试的概念	4
1.2.1 软件测试的定义	5
1.2.2 软件测试方法、对象和原则	6
1.2.3 软件测试技术的发展	7
1.3 软件产品及其软件项目工程	9
1.3.1 软件产品的组成	9
1.3.2 软件产品开发项目组成员	9
1.3.3 软件产品开发的方法和过程	10
1.4 软件测试与开发	14
1.4.1 测试与开发各阶段的关系	14
1.4.2 测试与开发的并行性	14
1.5 软件测试职业与测试员的必备素质	15
1.5.1 软件测试职业和职位	15
1.5.2 软件测试员的工作目标与必备素质	16
1.6 软件质量认证	16
1.6.1 软件质量简介	16
1.6.2 软件能力成熟度模型	28
1.6.3 ISO 9000 质量管理标准	32
小结与提高	35
思考与练习	35
第 2 章 软件测试方法	36
2.1 软件测试的复杂度	36
2.1.1 软件测试的不彻底性	36
2.1.2 潜在缺陷和故障的不确定	37
2.1.3 存在故障与发现故障成正比	37
2.1.4 软件的免疫力	38
2.1.5 有些软件故障不可能修正	38
2.1.6 软件测试的代价	38
2.2 常用的一些软件测试方法	39
2.2.1 静态与动态	39
2.2.2 常见的一些软件测试策略	49

2.3	单元测试	50
2.3.1	单元测试概述	51
2.3.2	进行单元测试的原因	54
2.3.3	单元测试的环境	56
2.3.4	单元测试的策略	57
2.3.5	单元测试分析	59
2.3.6	单元测试用例的设计思路	60
2.3.7	单元测试的过程	62
2.4	集成测试描述	69
2.4.1	非增量式综合测试	70
2.4.2	增量式综合测试	71
2.4.3	两种综合测试方法的比较	73
2.5	软件验证和确认测试	73
2.5.1	确认测试的原则	74
2.5.2	软件配置确认审查	75
2.6	系统测试	75
2.6.1	系统测试的定义	75
2.6.2	功能测试	76
2.6.3	协议的一致性测试	77
2.6.4	性能测试	79
2.6.5	恢复性测试	81
2.6.6	备份测试	81
2.6.7	安全性测试	82
2.6.8	压力测试	82
2.6.9	容量测试	83
2.6.10	GUI 测试	84
2.6.11	可靠性测试	86
2.6.12	可用性测试	87
2.6.13	兼容性测试	87
2.6.14	回归测试	89
2.6.15	可安装性测试	89
2.6.16	文档测试	92
2.6.17	在线帮助测试	93
2.6.18	Web 网站测试	93
2.6.19	数据转换测试	94
2.6.20	系统测试过程	95
2.7	验收测试	96
2.7.1	软件验收的内容	96
2.7.2	软件配置	97
2.7.3	软件调试	97

2.8 面向对象的软件测试方法	98
2.8.1 面向对象的描述	98
2.8.2 与传统软件测试的比较	100
2.8.3 面向对象的单元测试	107
2.8.4 面向对象的集成测试	109
2.9 其他的软件专项性测试	110
2.9.1 可接受性测试	110
2.9.2 Alpha 测试	110
2.9.3 Beta 测试	111
2.9.4 标杆测试	112
2.9.5 配置测试	112
2.9.6 外场测试	113
2.9.7 SQL 测试	113
2.9.8 千年测试	113
小结与提高	113
思考与练习	114
第 3 章 设计测试	115
3.1 设计测试概述	115
3.1.1 什么是设计	115
3.1.2 软件构架设计	117
3.1.3 概要设计和详细设计	121
3.2 设计的评审	122
3.2.1 设计检查要素	122
3.2.2 构架设计的评审方法	125
3.2.3 软件构架评价的最佳工业实践	129
3.3 实时系统的规格描述语言及相关测试	136
3.3.1 SDL 简介	136
3.3.2 SDL 的基本概念	136
3.3.3 SDL 的结构	139
3.3.4 SDL 测试	139
小结与提高	144
思考与练习	145
第 4 章 程序分析技术	146
4.1 文本视角和句法视角	146
4.1.1 文本视角	146
4.1.2 句法视角	147
4.2 控制流视角	152
4.3 数据流视角	153

4.4. 计算流视角.....	156
4.4.1 故障插入.....	156
4.4.2 变体分析.....	157
4.4.3 敏感性分析.....	160
4.5 功能视角	160
小结与提高	161
思考与练习	161
第5章 测试分析技术	162
5.1 面向规格的测试.....	162
5.1.1 测试独立于规格技术	163
5.1.2 测试依赖于规格技术	165
5.2 面向实现的测试.....	172
5.2.1 面向结构的测试.....	172
5.2.2 面向影响的测试.....	173
5.2.3 面向传递的测试.....	175
5.3 面向错误的测试.....	181
5.3.1 基于错误的测试.....	181
5.3.2 基于故障的测试.....	182
5.3.3 基于风险的测试.....	183
5.3.4 可能的正确性.....	184
5.4 混合测试技术.....	184
小结与提高	184
思考与练习	185
第6章 自动化测试.....	186
6.1 自动化测试的相关知识.....	186
6.1.1 自动化测试的背景.....	187
6.1.2 自动化测试的概念.....	187
6.1.3 选择自动化测试工具的优越性	188
6.1.4 自动化测试案例.....	188
6.2 自动化测试生存周期学.....	189
6.3 自动化测试生存周期的应用	191
6.3.1 必须选择正确的测试对象	191
6.3.2 测试自动化与测试工具存在的缺陷	192
6.4 常用的自动化测试工具.....	194
6.4.1 白盒测试工具.....	194
6.4.2 黑盒测试工具.....	195
6.4.3 常用的一些软件测试工具	195
小结与提高	200
思考与练习	200

第 7 章 测试计划与测试标准	201
7.1 测试计划	201
7.2 测试标准	205
7.2.1 IEEE/ANSI 测试文档概述	206
7.2.2 软件各阶段的测试任务和需要交付的文档	207
7.2.3 测试文档的类型	209
小结与提高	218
思考与练习	218
第 8 章 企业级测试工具——WinRunner	219
8.1 WinRunner 简介	219
8.1.1 测试模式	220
8.1.2 测试步骤	221
8.1.3 测试实例软件	222
8.1.4 配套的测试软件	222
8.2 运用 WinRunner	222
8.2.1 启动 WinRunner	222
8.2.2 WinRunner 的主窗口	223
8.2.3 WinRunner 的测试窗口	224
8.2.4 WinRunner 的插件	224
8.3 WinRunner 识别 GUI 对象	224
8.3.1 识别 GUI 对象	225
8.3.2 测试中识别 GUI 对象	225
8.3.3 物理描述	225
8.3.4 Logic name	226
8.3.5 GUI Map	226
8.3.6 Window Context	226
8.4 GUI Map 的相关知识	227
8.4.1 GUI Map 概述	227
8.4.2 对象属性的描述	227
8.4.3 使用 WinRunner	229
8.4.4 在 GUI Map 中找到窗体	229
8.4.5 对 GUI Map File 的理解	229
8.4.6 比较 GUI Map File 的两种工作模式	229
8.5 Global GUI Map File 的使用	230
8.5.1 关于 Global GUI Map File 的使用方法	230
8.5.2 GUI Map File 共享	231
8.5.3 RapidTest Script Wizard 的使用方法	231
8.5.4 GUI Map 的保存方法	233
8.5.5 GUI Map 文件的载入	234

8.6	GUI Map File Per Test 的使用	236
8.6.1	GUI Map File Per Test 的使用方法	236
8.6.2	GUI Map File Per Test 的工作	237
8.6.3	GUI Map File Per Test 的关键要素	237
8.7	GUI Map	237
8.7.1	GUI Map 的编辑	237
8.7.2	Run Wizard	237
8.7.3	“GUI Map Editor” 窗口	238
8.7.4	修改对象	238
8.7.5	窗体卷标	239
8.7.6	物理描述使用的表达式	240
8.7.7	在文件之间复制和移动对象	240
8.7.8	GUI Map File 的对象描述	241
8.7.9	快速找到 GUI Map File 对象	241
8.7.10	添加对象	242
8.7.11	删除对象	242
8.7.12	清除文件	242
8.7.13	筛选对象	243
8.7.14	保存结果	243
8.8	GUI Map 文件合并工具	243
8.8.1	合并 GUI Map File	243
8.8.2	事先准备	244
8.8.3	解决合并文件的冲突	244
8.8.4	手工合并文件	246
8.8.5	变换到 Global GUI Map File 模式	247
8.9	设置 GUI Map 的方法	247
8.9.1	关于设置 GUI Map	247
8.9.2	GUI Map 的默认设置	248
8.9.3	映射标准的类	248
8.9.4	设置标准的类	249
8.10	为类创建 GUI Map 配置	252
8.10.1	删除类	252
8.10.2	类的属性	252
8.10.3	WinRunner 在 Context sensitive 测试中使用的属性	253
8.10.4	各个类默认学习的属性	255
8.10.5	Visual Basic 属性	255
8.10.6	PowerBuilder 属性	255
8.11	虚拟对象	256
8.11.1	虚拟对象的操作方法	257
8.11.2	虚拟对象的物理描述	257

8.12	用录制或编程的方式创建测试	258
8.12.1	创建测试	258
8.12.2	解决常见问题	258
8.12.3	WinRunner 录制的脚本	259
8.12.4	在脚本中插入检查点	259
8.12.5	数据驱动测试	259
8.12.6	同步点	260
8.12.7	计划测试	260
8.12.8	信息标准化	260
8.12.9	相关插件	261
8.12.10	录制要点	262
8.12.11	用热键激活测试命令	262
8.12.12	创建编程脚本	263
8.12.13	编辑菜单命令	264
8.13	检查被测软件中的 GUI 对象	264
8.13.1	检查 GUI 对象	264
8.13.2	检查对象的属性值	265
8.13.3	检查单个对象的属性	265
8.13.4	检查多个对象的属性	267
8.13.5	检查全部对象的属性	267
8.13.6	检查语句	268
8.13.7	检查清单	269
8.13.8	对话框	270
8.13.9	属性和默认检查	270
8.13.10	检查指定变量	273
8.13.11	常规表达式	273
8.13.12	时间格式	273
8.13.13	关闭属性	273
8.13.14	期望值	273
8.13.15	修改期望结果	274
	小结与提高	275
	思考与练习	276
第 9 章	负载测试软件——WAS	277
9.1	负载测试软件的介绍	277
9.1.1	WAS 简介	277
9.1.2	Web Application Stress Tool 系统的安装	277
9.2	Web Application Stress Tool 的操作	278
9.2.1	主界面	278
9.2.2	制作脚本	278

9.3 参数设置	281
9.3.1 设置 Content Tree	281
9.3.2 设置 Settings	281
9.3.3 性能分析	283
9.3.4 用户单击的分布	284
9.3.5 用户登录名和密码	285
9.3.6 客户机	285
9.3.7 设置 Cookies	286
9.4 运行负载测试	286
9.5 结果分析	286
9.5.1 测试摘要	286
9.5.2 测试结果代码	287
9.5.3 统计信息	288
9.5.4 设置脚本	288
9.5.5 测试客户机	288
9.5.6 页面概要	288
9.5.7 页面用户组结果	288
9.5.8 测试的页面数据	289
9.6 编写测试脚本的方法	289
9.6.1 手工编写	289
9.6.2 导入 IIS 日志的方法	289
9.6.3 导入网站上具体的文件的方法	290
9.7 设计测试方案要素	291
9.8 Web 应用系统工具的评价	291
小结与提高	292
思考与练习	292
第 10 章 软件测试的经验总结	293
10.1 软件测试的语境驱动方法	293
10.2 软件测试的十大原则	294
10.3 软件测试的十个最佳实践	297
小结与提高	303
思考与练习	303
附录 A 常见的测试术语	304
附录 B 测试技术的分类	318
附录 C 常见的编码错误	321
附录 D 有关的测试网站	324
参考文献	328

第 1 章 | 软件测试概述

学习目标

- ☑ 了解软件测试的发展历史、软件缺陷的案例及其描述。
- ☑ 掌握软件测试的概念。
- ☑ 掌握软件测试与软件开发过程的联系。
- ☑ 了解一名软件测试员应具备的素质要求。
- ☑ 理解软件质量的概念。
- ☑ 了解国际质量标准认证 ISO 9000。

1.1 回顾软件测试的发展历史

测试涉及我们生活的各个方面，它是一个非常广泛的概念。一件产品生产出来后需要经过各种测试才能最终投放到商品市场中。作为质检人员，使用什么方法才能检验出产品质量的优劣呢？最简单的方法是通过建立一些不同条件下的环境，看产品是否能正常工作，这就是一种产品测试。

测试是随着现代化发展而产生的。本书讨论的测试主要是软件测试，它是软件工程的一部分。测试活动是一项工程性的活动，而不是一项简单的、孤立的活动。

软件测试最早可以追溯到软件开发的初期。在计算机问世之初，由于计算机的运行性能低，软件的编程范围也比较小，软件的错误也主要是因为硬件不稳定，这时候的测试也主要是指一些调试性的工作，测试的设计和选取也主要是在随机的基础上凭借一些个人的经验来进行的。

20 世纪 50 年代后期到 20 世纪 60 年代，随着计算机高级语言的应用，测试的重点转为高级语言编写的系统。程序的复杂性虽然增强了，但是由于程序受到硬件系统的制约，软件相对而言居于次要位置，软件的正确性也主要依赖于编程人员的水平。测试理论发展比较缓慢。

20 世纪 70 年代以后，随着计算机处理速度的不断提高以及存储介质的性能和存储容量的不断提高，软件在整个系统中的重要性变得越来越高，软件的规模越来越大，软件的开发也形成了一套体系，如可视化编程环境、面向对象的分析设计概念、软件开发模型（螺旋模型、增量模型）等。在软件的规模和复杂度增加的同时，软件的可靠性也面临着危机，软件测试承受了严峻的挑战。许多软件测试理论和方法相继诞生，逐步形成了一套完整的理论。

1.1.1 软件缺陷产生的原因

随着软件产业化发展，用户对软件的质量、成本和进度都提出了较高的要求，那么为什么还会产生软件缺陷呢？产业界（如 TRW、Nippon Electric 和 Mitre Corp 公司）研究表明，软件故障不一定是由编码所引起的，大部分是因为在详细设计阶段、概要设计阶段甚至是在需求分析阶段所存在的问题引起的。问题的根源可能存在于软件开发前期的各个阶段。

如果软件需求说明书写得不清楚、不全面，在开发过程中经常被更改，或开发组的成员

之间没有很好地进行交流和沟通，都会导致软件缺陷。在图 1-1 中，软件需求说明书产生的缺陷最大，其次是设计阶段产生的软件缺陷，由源代码引起的软件缺陷只占 7%，其他原因引起的软件缺陷占 10%。

软件从设计、编写、测试，直到用户公开使用的过程中，都有可能产生软件缺陷，随着软件整个开发过程的推移，软件修正的费用呈几何倍数增长，如图 1-2 所示。如果软件缺陷是在软件编写时发现的，那么修正费用按元计算，要上升几十倍；如果软件缺陷是在软件发布以后由用户发现的，那么修正费用可能要上升上百倍。

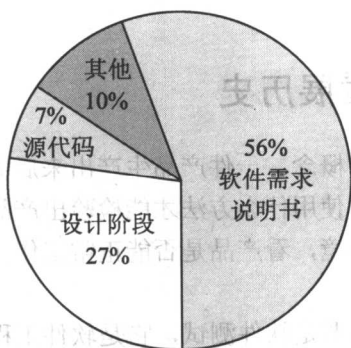


图 1-1 软件缺陷的原因分布图

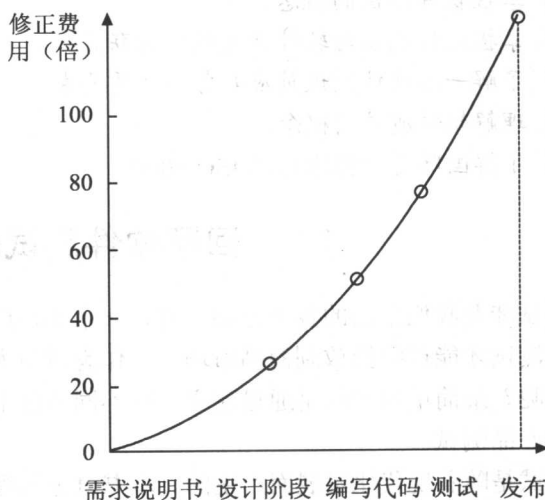


图 1-2 不同阶段的软件缺陷修正费用示意图

IBM 公司的研究结果表明，软件缺陷存在放大的趋势。如果在需求阶段漏过一个错误，则该错误可能会引起 K 个设计错误， K 称为放大系数。不同阶段的 K 的数值不同。经验表明，从概要设计到详细设计阶段的缺陷放大系数约为 1.5，从详细设计到编码阶段的缺陷放大系数约为 3。如图 1-3 所示即为缺陷放大的大致状况。

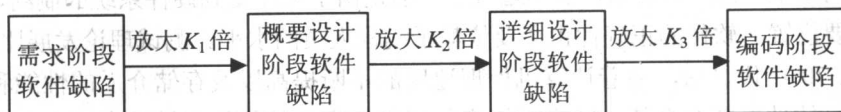


图 1-3 缺陷放大模型图

由于上述因素的制约，不难想象为什么虽然很多软件产品在其开发过程中的系统测试上投入了大量的时间和金钱，而最后得到的产品却是低质量的软件。在经过了大量的失败之后，人们慢慢明白，一个软件的成功不可能仅仅依赖于软件开发技术是否先进。软件的产业化要求有一个规范的软件开发过程，一个全局的质量控制体系。一个好的软件开发过程为软件的开发指明了一条通向成功的捷径。在此要求下，许多优秀的软件开发过程、开发规范应运而生，如 CMM（软件能力成熟度模型）、IPD（Integrated Product Development，集成产品开发是一套产品开发的模式、理念与方法）、RUP（Rational Unified Process，IBM 旗下 Rational 子公司的商业性软件开发过程，是对 UP 的一种商业扩展）等。在这些过程中，测试已经不再是编码后才可进行的活动，而是基于软件开发整个生命周期的质量控制活动。