

全国高职高专计算机技能型人才培养系列规划教材

中国计算机学会教育专委会
推荐教材(高职高专类)

软件测试设计与实施



主 编 蒋方纯



北京大学出版社
PEKING UNIVERSITY PRESS

全国高职高专计算机技能型人才培养系列规划教材

中国计算机学会教育专委会推荐教材（高职高专类）

软件测试设计与实施



北京大学出版社
PEKING UNIVERSITY PRESS

内 容 简 介

本书基于工作过程的教学思想,将“软件测试”学习领域以学生可能的就业岗位所面对的“软件产品”为载体分为7种学习情境:单机软件测试的设计与实施、网络软件测试的设计与实施、游戏软件测试的设计与实施、数据仓库软件测试的设计与实施、软件安全测试的设计与实施、嵌入式软件测试的设计与实施、开源软件测试的设计与实施。本书知识具有系统性,兼顾职业资格证书、研究学习和虚拟实训等内容,为学习者继续深入学习和职业发展奠定了基础。

本书的特点是帮助学习者架构软件测试理论与实践基础,重点突出不同软件产品的测试设计与实施,同时兼顾学习者的职业发展与深入学习。本书不仅适合作为高职高专软件测试课程教材,也可作为从事软件测试开发人员的参考用书。

图书在版编目(CIP)数据

软件测试设计与实施/蒋方纯主编. —北京:北京大学出版社, 2010.9

(全国高职高专计算机技能型人才培养系列规划教材)

ISBN 978-7-301-17727-3

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

中国版本图书馆 CIP 数据核字(2010)第 169232 号

书 名: 软件测试设计与实施

著作责任者: 蒋方纯 主编

策 划 编 辑: 张荣琴

责 任 编 辑: 刘国明

标 准 书 号: ISBN 978-7-301-17727-3/TP · 1129

出 版 者: 北京大学出版社

地 址: 北京市海淀区成府路 205 号 100871

网 址: <http://www.pup.cn> <http://www.pup6.com>

电 话: 邮购部 62752015 发行部 62750672 编辑部 62750667 出版部 62754962

电 子 邮 箱: pup_6@163.com

印 刷 者: 三河市北燕印装有限公司

发 行 者: 北京大学出版社

经 销 者: 新华书店

787mm×1092mm 16 开本 18.5 印张 419 千字

2010 年 9 月第 1 版 2010 年 9 月第 1 次印刷

定 价: 33.00 元

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有 侵权必究

举报电话: 010-62752024

电子邮箱: fd@pup.pku.edu.cn

《全国高职高专计算机技能型人才培养系列规划教材》 编委会委员名单

主任委员：俞光昀

副主任委员：刘乃琦 张凌雯 顾 滨 蒋方纯

委 员：(按姓名拼音排序)

卜锡滨	陈书谦	程 刚	崔剑波
董汉丽	郝 梅	何文华	贺 平
连卫民	梁锦叶	刘甫迎	刘湘涛
李金祥	骆耀祖	聂 明	宋汉珍
苏传芳	田绍愧	佟伟光	王 伟
王秀平	吴小惠	谢 尧	徐建民
严学道	杨丽芳	杨 威	杨学全
袁启昌	朱乃立		

秘 书 长：张荣琴 刘 丽

出版说明

高技能人才是国家核心竞争力的体现,加快高技能人才的培养已经纳入国家人才强国战略的总体部署。正是国家急需高技能人才的客观要求推动了高等职业教育的飞速发展。今天,高职高专学生已经占据了高等教育的半壁江山。每年几百万新生的招生规模是一个何等惊人的数字,将如此众多的青年人培养成具有良好的道德素养、熟练的职业技能的高技能人才是多么伟大的工程!对于肩负着这一伟大使命的高职高专院校,既是难得的历史机遇,又是艰辛的任务和挑战。我们要从不断改革教学模式、教学方法等各方面努力,争取把我国的高职高专教育推向一个新的高度。

在这样伟大的历史任务面前,中国计算机学会教育专委会高职高专学组和北京大学出版社联手成立了《全国高职高专计算机技能型人才培养系列规划教材》编委会,致力于从教材编写的角度为培养高技能人才做出新贡献。

二十多年前,由全国几十所大专、成人高校、电视大学、职工大学和夜大等大专层次的学校在湖南长沙发起成立了“全国大专计算机教育研究会”,1986年全国大专计算机教育研究会加入中国计算机学会教育专委会,简单称大专学组,从此就在中国计算机学会教育专委会的指导下有计划地开始了大专层次的计算机专业的教育和教材建设的研究。同年,经原电子工业部批准,在全国大专计算机教育研究会的基础上,成立了“全国大专计算机专业教材编委会”。随着高职高专教育的发展,随着新世纪的来临,大专学组和全国大专计算机专业教材编委会分别更名为高职高专学组和全国高职高专计算机专业教材编委会。

二十多年来,高职高专学组和高职高专计算机专业教材编委会一方面不断研究改进高职高专计算机各专业方向的培养计划和教学方法,另一方面与出版社合作联合成立相关系列编委会致力于高职高专计算机专业系列教材的编写工作。二十多年来,共完成了五轮近三百种教材的编写工作。

计算机高职高专教材的出版,解决了大专计算机教学过度依赖本科教材的问题,一轮又一轮,一批又一批教材的相继出版,不但使高职高专教材的质量与时俱进,同时还推动了高职高专院校师资队伍的成长。

但是,由于我国职业教育起步较晚,至今还没有形成西方发达国家那样完整的职业教育体系,因此在职业教育的许多方面,包括教材建设方面还存在着相对落后的方面和诸多不足。就教材而言,存在着部分新专业没有教材;教材内容陈旧,不适应新技术发展的需要;实践技能教材严重缺乏;教材内容和职业资格证书制度衔接不足等。

我国社会主义现代化建设需要大批高技能人才,而高技能人才的培养需要科学的、合理的教材。《全国高职高专计算机技能型人才培养系列规划教材》旨在在教材建设中引进国内外成熟的经验,同时适应高等职业教育不断改革的需要,在教材内容和教材风格上有所创新。

本套教材计划按照每门课程的不同特点,分别采用任务驱动法、项目教学法或案例教学法。

在教材内容上,本套教材力图将最新的知识、最新的技术写进教材;着重讲解技能型人才培养所需的内容和关键点,突出实用性和可操作性;尽量采用综合性的实例来讲解理论知识的

综合运用,“以例释理”,将理论讲解简单化,从而锻炼学生的思维能力以及运用概念解决问题的能力;要设计具备真实性的实践操作训练项目,加强学生对工程实践的兴趣,提高他们的实践操作技能;为了满足学有余力的学生深入学习的需要,我们提倡模块化编写方法,有些科目需要编写提高模块。

在编写风格上,本套教材将努力学习和借鉴国内外优秀教材的写作思路、写作方法和章节安排;作为工科教材,本套教材也将借鉴人文学科教材的写作模式,体现清新活泼的风格;部分教材还将采用学校教师任主编,企业高工任主审的方式,依托行业和企业共同进行编写;在出版纸质教材的同时,还将编写网络课件、CAI 课件、教学素材库、电子教案、试题库及考试系统和多媒体教学软件。

本套教材不仅适合高职高专院校计算机及相近专业的学生使用,也适用于企事业单位从业人员的在职培训,对于社会上广大自学人员的素质提高也具有实用价值和参考作用。

中国计算机学会教育专委会高职高专学组
《全国高职高专计算机技能型人才培养系列规划教材》编委会
2008 年 8 月

前 言

在软件业发达的国家,软件测试不仅早已成为软件开发的一个有机组成部分,而且在整个软件开发的系统工程中占据着相当大的比重。与此同步,软件测试市场已成为软件产业中的一个独特市场,凡是软件开发企业或者设有软件开发部门的公司,都有专门的软件测试单位,其中软件测试人员的数量相当于软件开发工程师的 3/4。因此,软件测试及产业的发展离不开软件测试人员的教育与培训。

本书基于工作过程的教学思想,将“软件测试”学习领域以学生可能的就业岗位所面对的“软件产品”为载体分为 7 种学习情境:第 4 章单机软件测试的设计与实施、第 5 章网络软件测试的设计与实施、第 6 章游戏软件测试的设计与实施、第 7 章数据仓库软件测试的设计与实施、第 8 章软件安全测试的设计与实施、第 9 章嵌入式软件测试的设计与实施、第 10 章开源软件测试的设计与实施。

本书兼顾了知识的系统性,第 1 章软件测试基础知识、第 2 章软件测试设计与实施、第 3 章软件测试实施与管理,奠定了学生学习的理论基础与实践基础。

在本书的最后还有 3 章:第 11 章软件测试职业资格证书、第 12 章软件测试的拓展学习与研究学习、第 13 章软件测试虚拟实训,分别讲解了职业资格证书、研究学习和虚拟实训的内容,为学生的深入学习和职业发展奠定了基础。

本书的特点是帮助学生架构软件测试理论与实践基础,重点突出不同软件产品的测试设计与实施,同时兼顾学生的职业发展与深入学习。因此,本书在每章后面的习题中,并没有沿用传统的习题方式,重复教材中讲述的概念、方法等知识,而是分为两个方面,一方面是知识的拓展与学习,另一方面是能力的拓展与训练,强调到企业去调研、到网上去查找、在小组间讨论、知识的总结与比较、技能的综合与灵活运用等。

本书是深圳信息职业技术学院基于工作过程教学改革教材建设项目,也是广东省高等教育改革项目《高职软件教育课程体系改革研究》的建设成果之一。本书将校内教师与企业技术人员、课程教学与真实案例、虚拟实训与职业发展有机结合起来,形成教材内容。

本书由蒋方纯主编,负责编写大纲与统稿,并编写第 1、2、12、7 章,许志良编写第 4、5 章,谢晓勇编写第 6、8 章,王建华编写第 9、10 章,陆云帆编写第 3、11、13 章。在此要感谢深圳信息职业技术学院的领导与有关老师、赛宝认证中心的有关技术人员、北京大学出版社的老师,以及有关参考书籍作者、网站,对本书的完成给予的支持与帮助。

本书的课程网站为 <http://jpkc.sziit.edu.cn/software/www/st/index.html>,上面有丰富的教学资源供教材选用者使用。

本书中所述 7 个教学情境中的案例,分别选自下列著作,并在此基础上进行编写而成,它们是:赵斌编著的《软件测试技术经典教程》、佟伟光主编的《软件测试》、周学毛等翻译的国外教材《游戏测试精要》、兰雨晴等翻译的国外参考书《软件测试的有效方法》、康一梅等编著的《嵌入式软件测试》等,在此表示衷心感谢。

由于编者水平有限,疏漏之处在所难免,恳请广大读者及使用本书的师生批评指正,联系电子邮箱 psp_st@sziit.com.cn。

编 者
2010 年 9 月

目 录

第 1 章 软件测试基础知识	1		
1.1 软件及软件测试的发展	1	1.7.1 软件测试部门的组织结构	21
1.1.1 软件的定义	1	1.7.2 软件测试团队	22
1.1.2 软件与软件测试的发展	2	1.7.3 软件测试人员的基本素质与技能	22
1.1.3 软件测试的现状与发展趋势	5	1.7.4 软件测试人员的职业发展	26
1.2 软件分类及测试要求	6	第 2 章 软件测试设计与实施	29
1.3 软件测试定义及软件测试的过程	7	2.1 软件测试流程	29
1.3.1 软件测试的定义	7	2.1.1 制订测试计划	29
1.3.2 软件测试的过程	8	2.1.2 编制测试大纲	30
1.3.3 软件测试的目的	9	2.1.3 设计测试方案	30
1.3.4 软件测试的原则	9	2.1.4 准备测试及搭建测试环境	30
1.4 软件缺陷	10	2.1.5 执行测试	31
1.4.1 软件缺陷的定义	10	2.1.6 评估测试	31
1.4.2 软件缺陷产生的原因	12	2.1.7 总结测试	32
1.4.3 软件缺陷与测试的关系	12	2.2 软件测试计划	32
1.5 软件测试分类	13	2.2.1 制订测试计划的原则	32
1.5.1 黑盒测试、白盒测试和灰盒测试	13	2.2.2 制订测试计划可能面对的问题	33
1.5.2 静态测试和动态测试	14	2.2.3 测试计划的标准	33
1.5.3 单元测试、集成测试、确认测试、系统测试和验收测试	14	2.2.4 制订测试计划	34
1.5.4 功能测试和性能测试	16	2.3 软件测试环境	39
1.5.5 回归测试、冒烟测试和随机测试	17	2.3.1 什么是测试环境	39
1.5.6 不同软件测试分类之间的关系	17	2.3.2 软件环境的分类	40
1.6 软件测试过程模型	18	2.3.3 怎样搭建测试环境	40
1.6.1 V 模型	18	2.3.4 测试环境的维护和管理	41
1.6.2 W 模型	18	2.4 软件测试用例	42
1.6.3 X 模型	19	2.4.1 什么是测试用例	42
1.6.4 H 模型	20	2.4.2 编写测试用例的注意事项	44
1.6.5 测试成熟度模型	20	2.4.3 测试用例的组织和跟踪	45
1.6.6 软件测试过程模型的选择	20	2.4.4 测试用例案例	47
1.7 软件测试岗位	21	第 3 章 软件测试实施与管理	50
		3.1 软件缺陷管理	50
		3.1.1 软件缺陷的属性	50
		3.1.2 软件缺陷报告	54

3.1.3 缺陷处理流程	57	5.2.2 定义	98
3.1.4 缺陷的跟踪和管理	59	5.2.3 质量风险摘要	98
3.1.5 常用缺陷管理工具	61	5.2.4 测试进度计划	99
3.2 软件测试管理	63	5.2.5 进入标准及退出标准	99
3.2.1 建立软件测试管理体系	63	5.2.6 测试配置和环境	100
3.2.2 软件测试管理的基本内容	64	5.2.7 测试开发及预算	100
3.2.3 软件测试管理原则	65	5.2.8 关键参与者及参考文档	100
3.2.4 常用软件测试管理工具	65	5.3 HIS 测试过程概述	101
3.3 软件测试工具	68	5.3.1 单元测试	101
3.3.1 软件测试工具分类	68	5.3.2 集成测试	102
3.3.2 常用软件测试工具	69	5.3.3 系统测试	103
3.3.3 如何选择软件测试工具	71	5.3.4 验收测试	103
3.4 软件测试文档	71	5.4 测试用例设计	103
3.4.1 软件测试文档的作用	72	5.4.1 门诊挂号管理子系统	
3.4.2 软件测试文档的类型	72	测试大纲	103
3.4.3 主要软件测试文档	73	5.4.2 其他可用性测试检查标准	105
第 4 章 单机软件测试的设计与实施	76	5.4.3 功能测试用例	105
4.1 单机软件案例分析	76	5.4.4 性能测试用例	114
4.1.1 项目及被测软件简介	76	5.5 缺陷报告	114
4.1.2 测试目的与要求	77	5.5.1 建立缺陷报告数据库	114
4.2 测试知识扩展	77	5.5.2 编写缺陷报告	115
4.2.1 已学相关知识回顾	77	5.6 测试结果总结分析	116
4.2.2 黑盒测试技术	78	5.6.1 测试总结报告	116
4.2.3 黑盒测试技术的综合运用	79	5.6.2 测试用例分析	117
4.2.4 白盒测试技术	81	5.6.3 软件测试结果统计分析	117
4.3 软件测试计划	83	5.7 软件测试自动化工具	120
4.4 软件测试的实施	87	5.7.1 黑盒测试工具介绍	120
4.5 测试总结	89	5.7.2 黑盒功能测试工具——	
第 5 章 网络软件测试的设计与实施	93	WinRunner	121
5.1 被测试软件项目介绍	93	5.7.3 在项目中应用	123
5.1.1 被测试软件项目背景	93	5.8 文档测试	124
5.1.2 门诊挂号管理子系统简介	94	第 6 章 游戏软件测试的设计与实施	126
5.1.3 门诊挂号管理子系统的		6.1 组合测试	126
功能需要分析	95	6.1.1 问题的提出	126
5.1.4 门诊挂号管理子系统的性能		6.1.2 组合测试的概念	126
及可用性要求	97	6.1.3 组合测试表格的生成	127
5.2 测试计划	97	6.1.4 组合测试的分析	129
5.2.1 概述	98	6.2 TFD 的要素	131
		6.2.1 流程	131

6.2.2 事件	131	9.1.3 嵌入式系统的分类	188
6.2.3 行动	132	9.1.4 嵌入式软件测试的方法	189
6.2.4 状态	132	9.2 嵌入式软件测试	191
6.2.5 基本要素	132	9.2.1 嵌入式软件测试的特点	191
6.2.6 终结器	132	9.2.2 嵌入式软件统一测试模型	192
6.3 TFD 设计活动	132	9.2.3 嵌入式软件目标机环境测试和 宿主机环境测试	192
6.3.1 准备	132	9.2.4 嵌入式软件的测试步骤概述	192
6.3.2 配置	133	9.2.5 嵌入式软件测试和普通 软件测试的区别	195
6.3.3 构建	133	9.2.6 嵌入式软件测试策略总结	196
6.4 一个 TFD 的例子	134	9.3 嵌入式软件测试环境	197
6.5 数据词典	139	9.3.1 嵌入式软件测试环境综述	197
6.5.1 数据词典的应用	139	9.3.2 嵌入式软件测试环境的建立	199
6.5.2 数据词典的重复使用	140	9.3.3 嵌入式软件测试环境建立 实例	200
6.5.3 数据词典的例子	140	9.4 嵌入式软件的特殊测试技术	203
6.6 TFD 路径	142	9.4.1 状态转换测试	203
6.6.1 最小值路径的产生	142	9.4.2 控制流测试	206
6.6.2 基线路径的生成	143	9.5 嵌入式软件测试工具	209
6.6.3 专家结构路径	144	9.5.1 LOGISCOPE	209
6.6.4 组合路径策略	145	9.5.2 CodeTest	210
6.7 由路径创建测试案例	146	9.5.3 CRESTS/ATAT	212
6.8 使用 TFD 或不使用 TFD	148	9.5.4 TestManager	213
第 7 章 数据仓库软件测试的 设计与实施	150	第 10 章 开源软件测试的设计与实施	215
7.1 数据仓库测试项目概述	150	10.1 开源代码的有关概念	215
7.2 学习数据仓库知识	151	10.2 开源软件测试模型	217
7.3 数据仓库测试工作流程	155	10.2.1 开源软件测试模型概述	217
7.4 数据仓库测试实施	156	10.2.2 测试环境	218
7.5 有关工作表	162	10.2.3 产品元素	219
7.6 总结	172	10.2.4 质量准则	221
第 8 章 软件安全测试的设计与实施	173	10.2.5 测试技术选择	222
8.1 软件安全测试	173	10.2.6 通用测试技术	222
8.2 软件安全测试项目概述	177	10.3 开源软件测试模型常用工具	224
8.3 软件安全测试的实施	178	10.4 JUnit 工具简介	224
8.4 总结	185	10.5 Selenium 工具	228
第 9 章 嵌入式软件测试的设计与实施	187	第 11 章 软件测试职业资格证书	236
9.1 嵌入式系统及测试	187	11.1 ISTQB 职业资格证书	236
9.1.1 嵌入式系统定义	187		
9.1.2 嵌入式系统的特点	187		

11.2	ISTQB 软件测试初级认证大纲	238	12.3.5	结束语	268
11.3	ISTQB 软件测试高级认证大纲	239	第 13 章	软件测试虚拟实训	269
11.4	模拟试题	239	13.1	虚拟实训	269
第 12 章	软件测试的拓展学习与 研究学习	252	13.1.1	传统实训存在的主要问题	269
12.1	软件测试理论中的阴阳学说	252	13.1.2	虚拟实验室	270
12.1.1	阴阳学说的基本内容及其 相互关系	253	13.1.3	虚拟实验室国内外现状	270
12.1.2	软件测试理论中的 阴阳学说	254	13.1.4	虚拟实训室的功能特点	271
12.1.3	结论	256	13.1.5	虚拟实验室的构建使用与 管理	271
12.2	游戏软件测试模式选择与 测试估计研究	257	13.2	虚拟现实技术	272
12.2.1	软件测试估计相关研究	257	13.2.1	虚拟现实技术的发展	272
12.2.2	游戏特有测试与功能点 分析技术	259	13.2.2	虚拟现实的概念	272
12.2.3	游戏软件测试模式设计与 测试估计	261	13.2.3	虚拟现实面临的主要问题	273
12.2.4	实验与结论	262	13.2.4	虚拟现实关键技术	273
12.3	基于净室软件工程的 游戏软件测试技术研究与分析	262	13.2.5	虚拟现实技术的重要技术 特征	275
12.3.1	前言	262	13.2.6	虚拟现实系统的构成与 应用	275
12.3.2	净室方法	263	13.3	虚拟企业简介	275
12.3.3	净室组合测试	264	13.3.1	虚拟企业的组成	275
12.3.4	净室测试分析	267	13.3.2	虚拟企业的特点	276
			13.3.3	虚拟企业的优势	276
			13.3.4	虚拟企业软件的开发语言与 运行环境	276
			参考文献	278	

第 1 章

软件测试基础知识

1.1 软件及软件测试的发展

计算机在 1946 年出现,软件开发到现在已有 60 多年的历史,在软件逐步发展的过程中,产生了软件测试技术。

1.1.1 软件的定义

软件的定义并不是固定不变的,不同的国家和组织出于各自的理解与利益,产生了不同的软件定义。同时,随着软件产业的不断发展,软件与传统产业的融合日益加深,传统的软件概念和定义也在随着时代的发展而发生变化,以适应软件产业自身发展的需要。

软件可以从多方面去理解,有的从学科方面理解,有的从软件系统方面理解。下面介绍国际组织、世界有关国家和我国对软件的定义,最后给出从软件测试角度对软件的理解与概念定义。

1978 年,世界知识产权组织 WIPO(World Intellectual Property Organization)发表的《保护计算机软件示范条例》中将计算机软件的概念阐述为:“计算机软件包括程序、程序说明和程序使用指导三项内容”。

美国电子电气工程师协会 IEEE(Institute of Electrical and Electronics Engineers)定义:软件是计算机程序及其说明程序的各种文档。

美国在 1980 年修改的著作权法中的第 101 条将计算机程序定义为“是一组旨在直接或间接用于计算机以取得一定结果的语句或指令”。此后,又通过联邦法院的判例,把源程序、目标程序、固化在只读存储器中的程序、系统程序和应用程序都归为计算机程序,并纳入著作权法的保护范围。该法并未对计算机软件做出一个明确的定义。而根据微软计算机百科词典的解释:软件是指计算机程序或能够使硬件工作的指令。

欧盟在 1991 年所颁布的计算机程序保护指令中将计算机软件的概念阐述为:计算机软件包括先前准备的程序设计资料和计算机程序。

日本在 1985 年颁布的著作权法修改草案中对计算机程序作出了如下的定义:“能使计算机完成某种功能的一组指令”,并明确规定,计算机程序“不包括为完成程序作品而使用的程

序语言、规则和方法”，其中，程序语言是表达程序所用的文字、符号或文字和符号的组合。日本没有把文档包括在计算机软件之内。

印度把软件称为“IT 软件”，指以计算机可读形式记录的，通过被归类为“IT 产品”的自动数据处理机进行处理并向用户提供交互性服务的所有指令、数据、声音和图像，包括原码或目标码表达的任何形式。印度对软件定义的突破源于其软件产业的发展需要，源于其深厚的软件外包背景。印度的软件外包很大部分是业务流程外包 BPO(Business Process Outsourcing)的内容，印度本土称为 ITES(IT Enabled Services)，即基于 IT 的服务，其涉及的内容很宽泛。印度的软件产业对于软件服务外包的依存度很大，所以在定义软件概念时充分考虑了整个软件产业的需求。

我国的学术界一直遵循程度加文档的定义方法，学术公认的权威软件定义来自国际电工委员会 IEC(International Electro-technical)制定的国际标准中的定义：“软件是与数据处理系统的操作有关的计算机程序、过程、规则和有关的文件集的总和。”中国对软件概念的界定也要根据中国软件产业的特点，把握国际软件产业的发展趋势，进行概念的创新和突破。

通常讲软件就是程序加文档，这也是软件概念的经典定义。从软件测试的角度来看，可以将软件定义为：软件=程序+数据(库)+文档+服务。后面对软件测试的学习与实践将围绕软件所包含的程序、数据(库)、文档和服务这几个方面进行。

1.1.2 软件与软件测试的发展

第一个编写软件的人是英国数学家奥古斯塔·艾达·洛夫莱斯(Augusta Ada LoveLace)，在 1836 年她开始尝试为查尔斯·巴贝奇(Charles Babbage)的机械式计算机写软件。尽管他们的努力失败了，但他们的名字永远载入了计算机发展的史册。20 世纪 60 年代美国大学里开始授予计算机专业的学位，教授学生编写软件。

计算机软件发展历史可以分成三个阶段：1955 年到 1965 年的开创阶段，1965 年到 1985 年的稳定阶段，1985 年到现在是发展阶段。计算机软件发展历史可以从不同的角度去划分，但上述的划分方法可以更好地帮助理解软件测试的形成与发展。

1955—1965 年，计算机运算速度越来越快、价格越来越便宜的新计算机不断涌现，软件工作人员需要不断针对不同计算机写出新的软件，这种变化速度令软件编写人员应接不暇。1965—1985 年，此阶段计算机硬件变化节奏缓慢一些，属于较平稳的年代，随着计算机软件的平稳发展，确立了软件在市场中的重要地位，软件成为商品并逐渐被人们理解和接受。1985 年到现在，是软件发展过程中最重要的时期。因为 PC 和工作站以半年更新一代的令人目不暇接的速度，势不可挡地入侵小型机、中型机甚至大型机领域，从而使计算机无处不在，计算机走出了象牙塔，走进了平常百姓家庭，走进了普通人的办公室。在家里、办公室、银行、邮局等生活工作的周围，处处可见计算机的应用成果，普及流行速度极快。

软件测试是伴随着软件的发展而产生的。在早期的软件开发过程中，软件规模都很小、复杂程度低，软件开发的过程混乱无序、相当随意，测试的含义比较狭窄，开发人员将测试等同于“调试”，目的是纠正软件中已经知道的故障，常常由开发人员自己完成这部分工作。对测试的投入极少，测试介入也晚，常常是等到形成代码，产品已经基本完成时才进行软件测试。

直到 1957 年，软件测试才开始与调试区别开来，作为一种发现软件缺陷的活动。由于一直存在着“为了让我们看到产品在工作，就得将测试工作往后推一点”的思想，因此潜意识里对测试的目的就理解为“使自己确信产品能工作”。测试活动始终后于开发的活动，测试通常

被作为软件生命周期中最后一项活动而进行。当时也缺乏有效的测试方法，主要依靠“错误推测 Error Guessing”来寻找软件中的缺陷。因此，大量软件交付后，仍存在很多问题，软件产品的质量无法保证。

到了 20 世纪 70 年代，开发的软件仍然不复杂，但人们已开始思考软件开发流程的问题，尽管对“软件测试”的真正含义还缺乏共识，但这一词条已经频繁出现。一些软件测试的探索者们建议在软件生命周期的开始阶段就根据需求制订测试计划，这时也涌现出一批软件测试的宗师，Bill Hetzel 博士就是其中的领导者。1972 年，软件测试领域的先驱 Bill Hetzel 博士(代表论著 *The Complete Guide to Software Testing*)，在美国的北卡罗来纳大学组织了历史上第一次正式的关于软件测试的会议。1973 年，他首先给软件测试下了一个这样的定义：“就是建立一种信心，认为程序能够按预期的设想运行(Establish confidence that a program does what it is supposed to do)”。1983 年他又将定义修订为：“评价一个程序和系统的特性或能力，并确定它是否达到预期的结果。软件测试就是以此为目的的任何行为(Any activities aimed at evaluating an attribute or capability of a program or system)”。在他的定义中的“设想”和“预期的结果”其实就是我们现在所说的用户需求或功能设计。他还把软件的质量定义为“符合要求”。他的思想的核心观点是：测试方法是试图验证软件是“工作的”，所谓“工作的”就是指软件的功能是按照预先的设计执行的，以正向思维，针对软件系统的所有功能点，逐个验证其正确性。软件测试业界把这种方法看作是软件测试的第一类方法。

尽管如此，这一方法还是受到很多业界权威的质疑和挑战。代表人物是 Glenford J. Myers(代表论著 *The Art of Software Testing*)。他认为测试不应该着眼于验证软件是工作的，相反应该首先认定软件是有错误的，然后用逆向思维去发现尽可能多的错误。他还从人的心理学的角度论证，如果将“验证软件是工作的”作为测试的目的，非常不利于测试人员发现软件的错误。于是他于 1979 年提出了他对软件测试的定义：“测试是为发现错误而执行的一个程序或者系统的过程(The process of executing a program or system with the intent of finding errors)”。这个定义，也被业界所认可，经常被引用。除此之外，Myers 还给出了与测试相关的三个重要观点，即测试是为了证明程序有错，而不是证明程序无错误；一个好的测试用例在于它能发现至今未发现的错误；一个成功的测试是发现了至今未发现的错误的测试。这就是软件测试的第二类方法，简单地说就是验证软件是“不工作的”，或者说是有错误的。Myers 认为，一个成功的测试必须是发现 Bug 的测试，不然就没有价值。Myers 提出的“测试的目的是证伪”这一概念，推翻了过去“为表明软件正确而进行测试”的错误认识，为软件测试的发展指出了方向，软件测试的理论、方法在之后得到了长足的发展。第二类软件测试方法在业界也很流行，受到很多学术界专家的支持。

到了 20 世纪 80 年代初期，软件和 IT 行业进入了大发展阶段，软件趋向大型化、高度复杂化，软件的质量越来越重要。这时，一些软件测试的基础理论和实用技术开始形成，并且人们为软件开发设计了各种流程和管理方法，软件开发的方式也逐渐由混乱无序的开发过程过渡到结构化的开发过程，以结构化分析与设计、结构化评审、结构化程序设计以及结构化测试为特征。人们还将“质量”的概念融入其中，软件测试定义发生了改变，测试不单纯是一个发现错误的过程，而且将测试作为软件质量保证(SQA)的主要职能，包含软件质量评价的内容。Bill Hetzel 在《软件测试完全指南》(*Complete Guide of Software Testing*)一书中指出：“测试是以评价一个程序或者系统属性为目标的任何一种活动。测试是对软件质量的度量。”这个定义至今仍被引用。软件开发人员和测试人员开始坐在一起探讨软件工程和测试问题。软件测试已有

了行业标准(IEEE/ANSI), 1983 年, 在 IEEE 提出的软件工程术语中给软件测试下了定义: 软件测试的目的是为了检验软件系统是否满足需求。它再也不是一个一次性的、而且只是开发后期的活动, 而是与整个开发流程融合成一体的活动。软件测试已成为一个专业, 需要运用专门的方法和手段, 需要专门的人才和专家来承担。

随着软件产业界对软件过程的不断研究, 并伴随着过程成熟度模型 CMM(Capability Maturity Model)的提出, 许多研究机构和测试服务机构从不同角度提出了有关软件测试方面的能力成熟度模型, 以作为 SEI-CMM 的有效补充。其中比较有代表性的有: 美国国防部提出的一个 CMM 软件评估和测试 KPA 建议; Gelper 博士提出了一个测试支持模型(Testing Support Model, TSM)来评估测试小组所处环境对于他们的支持程度; Burgess/Drabick I.T.I.公司提出的测试能力成熟度模型(Testing Capability Maturity Model, TCMM)则提供了与 CMM 完全一样的 5 级模型; Burnstein 博士提出了测试成熟度模型(Testing Maturity Model, TMM), 依据 CMM 的框架提出测试的 5 个不同级别, 关注于测试的成熟度模型, 它描述了测试过程, 是项目测试部分得到良好计划和控制的基础。

进入 20 世纪 90 年代, 软件行业开始迅猛发展, 软件的规模变得非常大, 在一些大型软件开发过程中, 测试活动需要花费大量的时间和成本, 而当时测试的手段几乎完全都是手工测试, 测试的效率非常低; 并且随着软件复杂度的提高, 出现了很多通过手工方式无法完成测试的情况, 尽管在一些大型软件的开发过程中, 人们尝试编写了一些小程序来辅助测试, 但是这还是不能满足大多数软件项目的统一需要。于是, 很多测试实践者开始尝试开发商业的测试工具来支持测试, 辅助测试人员完成某一类型或某一领域内的测试工作, 因而测试工具逐渐盛行起来。人们普遍意识到, 工具不仅是有用的, 而且要对今天的软件系统进行充分的测试, 工具是必不可少的。测试工具可以进行部分的测试设计、实现、执行和比较的工作。运用测试工具, 可以达到提高测试效率的目的。测试工具的发展, 大大提高了软件测试的自动化程度, 让测试人员从烦琐和重复的测试活动中解脱出来, 专心从事有意义的测试设计等活动。采用自动比较技术, 还可以自动完成对测试用例执行结果的判断, 从而避免人工比对存在的疏漏问题。设计良好的自动化测试, 在某些情况下可以实现“夜间测试”和“无人测试”。在大多数情况下, 软件测试自动化可以减少开支, 增加有限时间内可执行的测试, 在执行相同数量测试时节约测试时间。

测试工具的选择和推广也越来越受到重视。在软件测试工具平台方面, 商业化的软件测试工具已经很多, 如捕获/回放工具、Web 测试工具、性能测试工具、测试管理工具、代码测试工具, 等等, 这些都有严格的版权限制且价格较为昂贵, 因此无法自由使用。然而, 有一些软件测试工具开发商对某些测试工具提供了 Beta 测试版本以供用户有限次数使用。在开放源码社区中也出现了许多软件测试工具, 并已得到广泛应用且相当成熟和完善。

综上所述, 对软件与软件测试发展的简单概括见表 1.1。

表 1.1 软件开发与软件测试的关系

特 点	年 代	20 世纪 70 年代及以前	20 世纪 80 年代	20 世纪 90 年代后
软件规模		小	适中	超大
软件复杂性		低	中等	高
开发队伍规模		小	中等	大
开发方法及标准		个别	适中	复杂

续表

年 代	20 世纪 70 年代及以前	20 世纪 80 年代	20 世纪 90 年代后
特 点			
测试重要性认识	很少	有些	重要
测试方法及标准	个别	早期	正在形成
独立测试组织	很少	有些	许多
测试从业人员	很少	很少	许多

1.1.3 软件测试的现状与发展趋势

在软件业较发达的国家，无论从投入的人力上来看还是从时间上来看，软件测试都受到软件公司的极大重视。相比较而言，国内的软件测试还属于起步阶段，缺乏专业的第三方软件测试公司。

1. 国外现状

在软件业发达的国家，软件测试不仅早已成为软件开发的一个有机组成部分，而且在整个软件开发的系统工程中占据着相当大的比重。以美国的软件开发和生产的平均资金投入为例，通常，“需求分析”和“规划确定”各占 3%，“设计”占 5%，“编程”占 7%，“测试”占 15%，“投产和维护”占 67%。测试在软件开发中的地位，由此可见一斑。

与此同步的是，软件测试市场已成为软件产业中的一个独特市场。在美国硅谷地区，凡是软件开发企业或者设有软件开发部门的公司，都有专门的软件测试单位，其中软件测试人员的数量相当于软件开发工程师的 3/4。在这些公司或部门中，负责软件测试质量保证工作的经理其职位与软件开发主管往往是平行的。据了解，在软件产业发展较快的印度，软件测试在软件企业中同样拥有举足轻重的地位。

2. 国内现状

目前，国内软件测试市场发展比较缓慢，国内市场中的软件开发公司比比皆是，但是软件测试公司则凤毛麟角。

为什么国内的软件测试市场发展会比较缓慢，到现在企业才开始关注呢？主要原因是企业对软件测试的重要性理解不够。很多人认为程序能运行基本上就已经成功，没有必要成立专门的测试部门或设立测试岗位。

另一方面，软件开发企业在为软件开发支付费用后，就不希望再为软件的测试支付新的成本了，而项目甲方则往往认为开发合格的软件是软件开发企业的责任。即使有些项目的开发方或委托方有意对软件进行第三方测试，也会考虑到在测试过程中往往需要软件开发商提供源代码，担心其知识产权遭到侵犯，这是软件测试市场无法壮大的又一个重要原因。此外，软件开发企业严重缺乏专业测试力量也是因素之一。

3. 发展趋势

国际的测试领域已基本成熟，而我国的测试领域才刚刚开始。我们有很多的东西要去学习。如何更好地将软件项目管理和软件质量保障结合起来，让项目管理带动软件测试业的发展和成熟，将是项目管理中不可缺少的一部分。

4. 面临的挑战

近几年软件测试取得了较大的发展，但仍落后于软件开发的发展水平，它面临着很大的挑战。

(1) 软件测试理论不成熟的挑战。软件测试行业的兴起将在很大程度上取决于测试理论的成熟度。目前软件测试过程中还有一些问题没有定论或没有明确的定论，如软件测试的终止标准、如何评价测试价值等。

(2) 测试技术有待提高的挑战。目前国内软件测试技术比较落后，手工测试比重较大，自动化的性能测试、白盒测试、代码测试、安全测试等都处于初级阶段，软件测试的质量、进度、成本和风险都未得到有效的保证和控制。

(3) 软件测试人才缺乏的挑战。我国软件产业已经获得了长足的进步，但软件测试人才的缺乏在很大程度上制约了软件产业的发展。加紧建立和健全软件测试人才培养体系正面临着挑战。

(4) 软件测试工程师素质的挑战。软件测试工程师的综合素质高低体现在：责任心、综合素质、学习能力、解决问题的能力以及对软件业发展趋势的了解。软件测试团队规模会越来越大，在一个测试团队中能否形成以核心人物为支柱的强有力的测试团队成为关键。

1.2 软件分类及测试要求

从不同角度去分类，可以将软件分为不同的种类，不同种类的软件，其测试要求和侧重点也各不相同。

1. 按功能分类及测试要求

按功能划分软件可分为4类：固件、系统软件、中间件和应用软件。

(1) 固件(Firm Ware)就是直接写在芯片组或集成电路里的一段程序。其原来的定义是指具有软件功能的硬件，早期的这类器件一般是存有软件的ROM或EPROM，这些硬件中保存的程序无法被用户直接读出或修改。目前固件有了新的定义，它是最贴近计算机硬件的一些小的软件，可以看作是一个系统中最低层、最基础的工作软件。固件中软件的测试都是由计算机设计厂家，根据设计及硬件情况来完成其中固化的软件测试工作的，一般的用户很少需要进行该类软件的测试工作。

(2) 系统软件能直接操作底层硬件，并为上层软件提供支撑，如操作系统、硬件驱动程序等。系统软件和硬件共同提供一个平台，管理和优化计算机硬件资源的分配与使用。这类软件的测试需要结合底层硬件来完成。

(3) 中间件在应用软件和平台之间建立一种桥梁，常见的中间件包括数据库和互联网服务器等。这类软件的测试除进行一些一般软件共同需要的测试外，一般都有专门的测试方法和工具。

(4) 应用软件能为用户提供某种特定的应用服务，如办公软件、通信软件、行业软件、游戏软件等。这类软件是大部分公司测试的重点。本书主要讨论这类软件的测试设计与实施。

2. 按用户分类及测试要求

按用户分类可将软件分为产品软件和项目软件。