高等院校信息技术规划教材

软件测试技术及实践

詹慧静 主编　陈　燕 段相勇 副主编

清华大学出版社

高等院校信息技术规划教材

软件测试技术及实践

詹慧静 主编　陈　燕 段相勇 副主编

清华大学出版社
北京

内 容 简 介

本书全面、系统地阐述了软件测试的基本理论和基本技术。全书共8章，内容包括软件测试基本知识、白盒测试技术、黑盒测试技术、软件生存周期的测试、缺陷报告与测试评估、测试管理、软件自动化测试工具以及自动化测试实例。本书精心安排了典型案例，介绍了不同测试方法中测试用例的设计过程及自动化功能、性能测试。本书既注重内容的先进性，又突出了教材的应用性和实践性，将软件测试与软件工程密切结合，强调将软件测试贯穿整个软件生存周期，使软件测试知识能迅速运用到软件工程实践中。

本书既可作为高等学校本科软件测试课程教材，也可以作为软件测试人员的参考书。

图书在版编目(CIP)数据

软件测试技术及实践/詹慧静主编. —北京：清华大学出版社，2015
高等院校信息技术规划教材
ISBN 978-7-302-42528-1

Ⅰ. ①软… Ⅱ. ①詹… Ⅲ. ①软件－测试－高等学校－教材 Ⅳ. TP311.5

中国版本图书馆 CIP 数据核字(2016)第 000833 号

责任编辑：焦　虹　战晓雷
封面设计：常雪影
责任校对：焦丽丽
责任印制：刘海龙

出版发行：清华大学出版社
网　　址：http://www.tup.com.cn，http://www.wqbook.com
地　　址：北京清华大学学研大厦 A 座　　**邮　　编**：100084
社 总 机：010-62770175　　**邮　　购**：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质量反馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
课件下载：http://www.tup.com.cn,010-62795954
印 刷 者：北京富博印刷有限公司
装 订 者：北京市密云县京文制本装订厂
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张**：18.25　　**字　　数**：422 千字
版　　次：2016 年 4 月第 1 版　　**印　　次**：2016 年 4 月第1次印刷
印　　数：1～2000
定　　价：39.00 元

产品编号：063557-01

前言 foreword

随着软件规模的不断增大和软件复杂性的日益提高，市场对软件质量的要求不断提高，如何保证软件质量已成为软件开发过程中越来越重要的问题。软件测试是保证软件质量的重要手段。软件测试直接决定软件产品的质量。

近年来，软件测试工作受到人们越来越多的重视，软件行业对进行专业化、高效率软件测试的要求也越来越高，越来越严格。要开发一个好的软件，需要有素质过硬的软件测试人员。国际化大型软件公司在软件测试上投入了大量的人力和物力，软件测试人才越来越受到重视。我国的软件测试工作远远落后于国外，软件测试人才的紧缺已是无法回避的事实，要让软件质量上台阶，需要更多合格的软件测试人才，这是促进我国软件产业成熟的一个亟待解决的问题。

软件测试是一项专业性较强的工作，除了要求软件测试人员有一定的实际开发经验，还要求测试人员掌握许多测试理论和实用的测试技术。作为高等学校计算机软件相关专业，软件测试是必须开设的一门专业课程。如何将软件测试教材的内容安排得既系统、合理、适用，又符合市场对软件测试人才的测试理论和测试技术的要求，是软件测试课程教师需要关心和思考的问题。为了满足教学需求，我们组织了有丰富软件开发经验及软件测试课程教学经验的人员共同编写了本教材。我们在编写中参阅了大量国内外相关文献资料，将软件开发及软件测试教学的经验融入教材中，在内容组织结构方面做了精心安排，设计了较多经典实例。

本书全面、系统地阐述了软件测试的基本理论和基本技术，全书共8章，第1章介绍软件测试基本知识；第2章和第3章介绍白盒及黑盒测试技术常用的测试方法，并通过典型案例介绍不同测试方法中测试用例的设计过程；第4章强调软件测试贯穿整个软件生存周期，介绍软件生存周期的不同测试阶段，即单元测试、集成测试、系统测试和验收测试4个阶段，还介绍性能测试和在各个测试阶段都可采用的回归测试；第5章介绍软件缺陷的基本概念及如何报

告软件缺陷和对缺陷进行评估；第6章介绍如何对测试项目实施各项管理活动；第7章介绍自动化测试的定义及发展、软件测试工具的作用和优势及软件测试工具的分类，并选择一些常用和主流的测试工具进行介绍；在第8章采用“任务驱动式”的编写模型，精心设计了WinRunner功能测试和LoadRunner负载测试两个实例。通过这两个自动化测试实例来带动WinRunner 8.2和LoadRunner 11.0自动化测试工具的学习。本书是一本非常实用的软件测试教材。

本书第1章、第2章、第3章、第7章由詹慧静编写，第8章由詹慧静、董坤共同编写，第4章、第6章的6.1节至6.4节和6.8节由陈燕编写，第5章、第6章的6.5节至6.7节由段相勇编写。全书由詹慧静、陈燕修改定稿。

李汶秋、王洋、杨丽萍、吴海峰、白玲、吴梅香、邓静参与了第5章的编写工作，并为本书编写工作收集了一些相关资料及进行了校稿工作。

真理是相对的，实践是多元的，读者是最好的老师，尽管编者以认真、严谨的态度来完成这本教材的策划和编写，但由于时间仓促，书中难免会存在疏漏之处，我们热切期待读者的批评指正。

编　者

目录 Contents

第1章 chapter 1

软件测试概述

本章学习目标

- 了解软件、软件危机和软件工程的基本知识。
- 了解软件错误、软件缺陷和软件故障的概念。
- 了解软件质量的含义及常用的质量模型。
- 了解软件测试基础知识。
- 熟练掌握指导测试过程的常用软件测试模型。
- 熟练掌握测试用例设计的基本知识。
- 了解软件测试的组织和对人员的要求。
- 了解软件测试的发展历史。

软件测试是软件工程的一个重要部分，是确保软件质量的重要手段。最近几年来，由于软件复杂度的不断增强，更由于软件的工业化发展趋势，软件测试得到广泛的重视。本章先向读者介绍软件测试需要的基本知识，再介绍指导衡量软件质量的常用质量模型和指导软件测试过程的常用软件测试模型，最后介绍软件测试的核心内容——测试用例及设计的相关知识。

1.1 软件、软件危机和软件工程

对于软件测试，需要了解如下软件及软件工程的内容：

- 软件及软件危机。
- 软件工程。
- 软件的开发模型。

1.1.1 软件及软件危机

计算机系统包括硬件系统和软件系统两大部分。随着电子技术的发展和进步，软件从规模、功能、应用范围上得到了很大的发展，人们对软件的需求和依赖越来越大，对软件的质量要求也越来越高。那么，什么是软件？软件有哪些特征呢？

1. 软件的定义及特征

软件是能够完成预定功能和性能的可执行的计算机程序，包括使程序正常执行所需要的数据，还包括在软件开发过程中记录的开发活动及为了维护和使用软件的一系列文档。

软件是一种特殊的产品，有以下特征：

(1) 软件是逻辑产品，它具有抽象性，与硬件产品有本质的区别。

(2) 软件没有明显的制造过程，要提高软件的质量，必须在软件开发方面下工夫。

(3) 软件在运行使用期间，没有像硬件那样的机械磨损、老化问题，但它存在退化问题，必须对其进行多次修改与维护。

(4) 软件的开发和运行受到计算机系统的限制，对计算机系统环境有着不同程度的依赖性。为了解决这种依赖性带来的问题，在软件开发中提出了软件的移植问题。

(5) 软件产品生产的成本主要是脑力劳动，在还未完全摆脱手工开发方式的情况下，大部分产品是“定做”的。

(6) 软件本身是复杂的。软件的复杂性可能来自它反映的实际问题的复杂性，也可能来自程序逻辑结构的复杂性。

(7) 软件成本相当昂贵。软件的开发需要高强度的、复杂的脑力劳动，因此成本昂贵。

(8) 软件的推广应用涉及社会因素。

2. 软件危机

软件危机是指落后的软件生产方式无法满足迅速增长的计算机软件需求，从而导致软件开发与维护过程中出现一系列严重问题的现象。

软件危机主要表现在以下几个方面：

(1) 软件功能与实际需求不符。原因是开发人员没有准确地理解用户的需求，一方面，许多用户在软件开发的初期不能准确完整地向开发人员表达他们的需求；另一方面，软件开发人员常常在对用户需求还没有正确全面认识的情况下，就急于编写程序。

(2) 软件生产率随着软件规模与复杂性提高而下降，软件生产不能满足日益增长的软件需求。

(3) 软件开发费用和进度失控。软件开发中费用超支，开发周期大大超过规定日期的情况经常发生，有时为了赶进度和节约成本采取一些权宜之计，这样又往往严重损害了软件产品的质量。

(4) 软件难以修改、维护。很多程序缺乏相应的文档资料，程序中的错误难以定位，难以改正，有时改正了已有的错误又引入新的错误。

(5) 软件的可靠性差。尽管软件开发耗费了大量的人力物力，而系统的正确性却越来越难以保证，出错率大大增加，由于软件错误而造成的损失十分惊人。

(6) 软件产品质量难以保证。开发团队缺乏严密有效的质量检测手段，以及缺少完善的软件质量保证评审体系，使最终的软件产品存在很多缺陷。

(7) 软件文档配置没有受到足够的重视。软件文档不完备,并且存在文档内容与软件产品不符的情况。

软件危机产生的原因有两方面:一是软件产品的固有特性;二是软件专业人员自身的缺陷。

软件的不可预见性和逻辑结构复杂是软件产品固有特性,软件不同于硬件,它是逻辑产品而不是物理部件,随着软件的发展,软件规模越来越大并且逻辑越来越复杂。在写出程序代码并在计算机上试运行之前,软件开发过程的进展情况较难衡量,软件质量也较难评价,因此管理和控制软件开发过程十分困难。

来自软件专业人员自身的缺陷主要体现在以下方面。其一,软件产品是人的思维结果,因此软件生产水平最终在相当程度上取决于软件人员的教育、训练和经验的积累;其二,大型软件往往需要很多人合作开发,甚至要求软件开发人员深入应用领域的问题研究,这样就需要在用户与软件人员之间以及软件开发人员之间相互沟通和交流,在此过程中难免发生理解的差异,从而导致后续错误的设计或实现,而要消除这些误解和错误往往需要付出巨大的代价;其三,由于计算机技术和应用发展迅速,知识更新周期加快,软件开发人员经常处在变化之中,不仅需要适应硬件更新的变化,而且还要涉及日益扩大的应用领域问题研究;软件开发人员在每一项软件开发中几乎都必须调整自身的知识结构以适应新的问题求解的需要,而这种调整是人所固有的学习行为,难以用工具来代替。

解决软件危机既要有技术措施,又要有必要的组织管理措施。软件工程就是从技术和管理两个方面研究如何更好地开发和维护计算机软件的一门学科。

1.1.2 软件工程

1. 软件工程的定义及目标

1) 软件工程的定义

"软件工程"一词,首先是 1968 年北大西洋公约组织(NATO)在联邦德国召开的一次会议上提出的。它反映了软件人员认识到软件危机的出现,以及为谋求解决这一危机而做的一种努力。对于软件工程,人们从不同的角度给其下过各种定义。

- 定义一:软件工程是将系统化的、严格约束的、可量化的方法应用于软件的开发、运行和维护,即将工程化应用于软件。
- 定义二:软件工程是建立并使用完善的工程化原则,以较经济的手段获得能在实际机器上有效运行的可靠软件的一系列方法。
- 定义三:软件工程是应用计算机科学、数学、逻辑学及管理科学等原理,开发软件的工程。软件工程借鉴传统工程的原则、方法,以提高质量、降低成本和改进算法。其中,计算机科学、数学用于构建模型与算法,工程科学用于制定规范、设计范型(paradigm)、评估成本及确定权衡,管理科学用于计划、资源、质量、成本等管理。

软件工程除以上 3 种定义,还有很多种定义,但不论有多少种定义,它的中心思想都

是把软件当作一种工业产品，要求“采用工程化的原理与方法对软件进行计划、开发和维护”。这样做的目的，不仅是为了实现按预期的进度和经费完成软件生成计划，也是为了提高软件的生成率和软件的质量。

2）软件工程的目标

从狭义上说，软件工程的目标是生产出满足预算、按期交付、用户满意的无缺陷的软件，进而当用户需求改变时，所生产的软件必须易于修改。从广义上说，软件工程的目标就是提高软件的质量与生产率，最终实现软件的工业化生产。

要达到软件工程的目标，在软件开发时必须注重考虑下面几个方面的问题：

（1）可修改性。允许对系统进行修改，而不增加系统的复杂性。

（2）有效性。软件系统能在一定的时间资源和空间资源环境下完成规定的任务。

（3）正确性。软件能够准确无误地执行用户需求的各种功能，满足用户要求的各种性能指标。

（4）可靠性。有时也称为健壮性，就是在硬件、操作系统出现小故障，或者人为操作不当的情况下，不会导致软件系统失效。如对卫星导航系统，可靠性要求就特别高。

（5）可理解性。包括两个方面的内容，一是软件系统结构清晰，容易理解；二是程序算法功能清晰，容易读懂。可理解性有助于控制软件系统的复杂性，提高软件的可维护性。

（6）可复用性。软件中的某个部分可以在系统的多处重复使用，或者在多个系统中使用。

（7）可适应性。体现软件在不同的硬件和操作系统环境下的适应程度。

（8）可移植性。体现了软件从一种计算机环境移动到另一种计算机环境下的难易程度。

（9）可跟踪性。包括两个方面，一是可以根据软件开发的文档对设计过程进行正向跟踪或逆向跟踪；二是软件测试和维护过程中，对程序的执行进行跟踪，根据跟踪情况，分析程序执行的因果关系。

（10）互操作性。多个软件相互通信，协作完成任务的能力。

2. 软件的生存周期

一个软件从定义到开发、使用和维护，直到最终被废弃，要经历一个漫长的时期，通常把软件经历的这个漫长的时期称为软件的生存周期。

软件工程强调使用生存周期方法，从时间的角度对软件开发和维护的复杂问题进行分解，把软件生存的漫长周期依次划分为若干阶段，每个阶段有相对独立的任务，最后逐步完成每个阶段的任务。软件工程采用生存周期的方法，对软件产品开发过程的管理、软件开发工具的组织及软件质量保障具有重要意义。首先，由于把整个开发工作划分成若干开发阶段来完成，这样能把复杂的问题分解并分派到不同阶段加以解决，使得每阶段工作目标明确，工作相对简单。其次，可为每一阶段的中间产品提供了检验的依据。

一般，软件生存周期包括软件定义、软件开发、软件使用与维护 3 个阶段，每个阶段又可细分为不同的子阶段。软件生存周期一般由制订计划、需求分析、软件设计、程序编

写、软件测试、运行维护 6 个子阶段组成，如图 1-1 所示。下面简单介绍各阶段的主要任务。

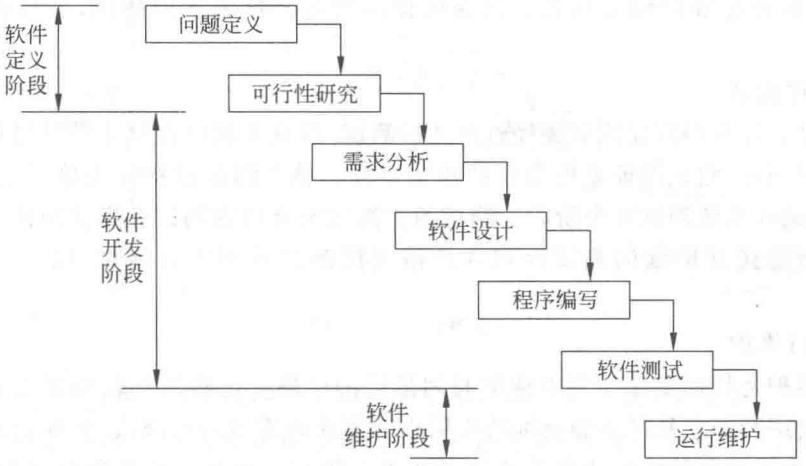

图 1-1 软件生存周期

1）制订计划

此阶段由软件开发人员和用户通过沟通、讨论，弄清楚用户需要解决的问题。由系统分析员根据对问题的理解，提出系统目标与范围，请用户审查并认可。之后根据系统目标及实现环境，从技术、经济和社会等几个方面研究并论证开发软件系统的可行性，并写出可行性论证报告。如果结论认为该软件项目值得进行，应制订出完成开发任务的项目实施计划，否则应提出项目终止的建议。同时，此阶段还应制定出人力、资源及进度计划。

2）需求分析

需求分析的任务是确定所要开发软件的功能需求、性能需求、运行环境约束和外部接口等描述。编制软件需求规格说明书、软件系统的确认测试准则。软件的性能需求包括软件的适应性、安全性、可靠性、可维护性、错误处理等。为了使需求规格说明书更直观，易于理解，可使用需求模型。

3）软件设计

这一阶段主要根据需求分析的结果，对整个软件系统进行设计，如系统框架设计、数据库设计等。软件设计一般分为概要设计和详细设计，主要是根据软件需求规格说明书建立软件系统的结构，明确算法、数据结构和各程序模块之间的接口信息，规定设计约束，并为编写源代码提供必要的说明。概要设计即把确定的各项功能需求转换成需要的体系结构，在该系统结构中，每个成分都是意义明确的模块，而每个模块都和某些需求相对应。详细设计就是为每个模块完成的功能进行具体描述，要把功能描述转变成精确的、结构化的过程描述，即该模块的控制结构是怎样的，应该先做什么、后做什么，有什么样的条件判定，有什么重复处理等，并用相应的表示工具把这些控制结构表示出来。软件设计是软件工程的技术核心，好的软件设计将为软件程序编写打下良好的基础。

4）程序编写

此阶段是将软件设计的结果转换成计算机可运行的程序代码。在程序编码中必须制定统一、符合标准的编写规范。以保证程序的可读性和易维护性，提高程序的运行效率。

5）软件测试

在软件设计及编码完成后要经过严密的测试，以发现软件在整个设计过程中存在的问题并加以纠正，它是保证软件质量的重要手段。整个测试过程分为单元测试、集成测试、确认测试和系统测试4个阶段。测试的方法主要有白盒测试和黑盒测试两种。在测试过程中需要建立详细的测试计划并严格按照测试计划进行测试，以减少测试的随意性。

6）运行维护

软件维护是软件生存周期中持续时间最长的阶段。该阶段在软件开发完成并投入使用后开始进行。在软件不能继续适应用户的要求而需要进行修改（比如运行中发现了软件故障，为了增强软件的功能需要进行变更等等）时，必须对软件进行维护。另外，要延续软件的使用寿命，也需要对软件进行维护。

3. 软件过程

什么是过程？广义地说，人们随时间的流逝而进行的各种活动均可称为过程（process，或译为流程）。软件过程可理解为围绕软件开发所进行的一系列活动。软件过程也称为软件开发模型。早期软件开发中，软件过程与软件生存周期过程常常不加区分。但是随着软件的发展，软件的规模越来越大，逻辑结构也越来越复杂，对于大型、复杂的软件系统，采用这种线性模型显然是不合适的。除传统的线性开发模型外，又陆续涌现了一批新的、允许在开发过程中任意回溯和迭代的过程模型。常用的软件开发模型有瀑布模型、快速原型模型、螺旋模型、增量模型、构件集成模型、转换模型、净室模型、统一过程等。

1.1.3 软件的开发模型

1. 瀑布模型

瀑布模型（也称线性顺序模型或软件生存周期模型）是温斯顿·罗伊斯（Winston Royce）在1970年提出的。瀑布模型遵循软件生存周期的划分，明确规定各个阶段的任务，各个阶段的工作自上而下、顺序展开，如同瀑布流水，逐级下落。

瀑布模型把软件生存周期划分为计划时期（或定义时期）、开发时期和维护时期，这3个时期又分别细分为若干个阶段。瀑布模型如图1-2所示。

瀑布模型的特点如下：

（1）顺序性。只有等前一阶段的工作完成以后，后一阶段的工作才能开始；前一阶段的输出文档就是后一阶段的输入文档。

（2）依赖性。只有前一阶段有正确的输出时，后一阶段才可能有正确的结果。