



北京高等教育精品教材
BEIJING GAODENG JIAOYU JINGPIN JIAOCAI

全国优秀畅销书

全国高校出版社优秀畅销书

软件工程导论

(第5版)

张海藩 编著

清华大学出版社





北京高等教育精品教材
BEIJING GAODENG JIAOYU JINGPIN JIAOCAI

全国优秀畅销书
全国高校出版社优秀畅销书

张海藩 编著

软件工程导论

(第5版)

◎◎◎◎◎◎◎◎◎◎

◎◎◎◎◎◎◎◎◎◎

3

)

)

清华大学出版社

清华大学出版社

清华大学出版社

清华大学出版社

清华大学出版社

清华大学出版社

清华大学出版社

清华大学出版社

清华大学出版社

清华大学出版社

清华大学出版社

清华大学出版社

清华大学出版社

清华大学出版社
北京

内 容 简 介

本书的前4个版本累计销售近一百万册,已成为软件工程领域的经典教材,先后荣获全国普通高等学校工科电子类专业优秀教材二等奖、一等奖,并被评为全国优秀畅销书(前10名)、全国高校出版社优秀畅销书和北京高等教育精品教材。为了反映最近4年来软件工程的发展状况,作者对第四版作了下述的精心修改:增加了目前比较流行的 Rational 统一过程、以极限编程为杰出代表的敏捷过程以及微软过程的介绍,并且对书中的一些具体内容作了适当的增删或修改。

本书全面系统地讲述了软件工程的概念、原理和典型的方法学,并介绍了软件项目的管理技术。本书正文共13章,第1章是概述,第2章至第8章顺序讲述软件生命周期各阶段的任务、过程、结构化方法和工具,第9章至第12章分别讲述面向对象方法学引论、面向对象分析、面向对象设计和面向对象实现,第13章介绍软件项目管理。正文后面有两个附录,分别讲述了用面向对象方法和结构化方法开发两个软件的过程,对读者深入理解软件工程学很有帮助,也是上机实习的好材料。

本书可作为高等院校“软件工程”课程的教材或教学参考书,也可供有一定实际经验的软件工作人员和需要开发应用软件的广大计算机用户阅读参考。

本书扉页为防伪页,封面贴有清华大学出版社防伪标签,无上述标识者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

软件工程导论/张海藩编著. —5版. —北京:清华大学出版社,2008.2
ISBN 978-7-302-16478-4

I. 软… II. 张… III. 软件工程—高等学校—教材 IV. TP311.5

中国版本图书馆 CIP 数据核字(2007)第 176355 号

责任编辑:袁勤勇 李 晔

责任校对:白 蕾

责任印制:何 芊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

c-service@tup.tsinghua.edu.cn

社 总 机:010-62770175

邮购热线:010-62786544

投稿咨询:010-62772015

客户服务:010-62776969

印 刷 者:清华大学印刷厂

装 订 者:三河市金元印装有限公司

经 销:全国新华书店

开 本:185×260 印 张:26.25 插 页:1 字 数:610千字

版 次:2008年2月第5版 印 次:2008年3月第2次印刷

印 数:944001~984000

定 价:35.00元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:026013-01

第五版前言

INTRODUCTION

本书第四版出版后，受到广大读者的热烈欢迎，先后被评为全国优秀畅销书（前10名）、全国高校出版社优秀畅销书和北京高等教育精品教材。为了反映最近4年来软件工程的发展状况，作者对原书内容作了认真修改，写出了第五版。

鉴于先进适用的软件过程对提高软件生产率和确保软件产品质量有相当大的作用，第五版在保持原书结构和篇幅基本不变的前提下，主要增加了目前比较流行的 Rational 统一过程、以极限编程为杰出代表的敏捷过程以及微软过程的介绍，此外还对书中的一些具体内容作了适当的增删或修改。

倪宁对第五版应增加的内容提出了有益的建议，谨在此向他表示感谢。

编者

2008年1月

第四版前言

PREFACE

光阴荏苒，本书第三版已经出版 5 年多了。在此期间软件工程又有了很大发展，为了跟踪学科发展方向，更好地为广大读者服务，编者对原书内容作了认真修改，写出了第四版。

在保持原书的结构和篇幅基本不变的前提下，第四版主要对原书内容作了下述修改：

(1) 删掉了一些较陈旧的或较次要的内容。删掉的内容主要有：Warnier 程序设计方法，程序设计语言概述，程序设计途径，日立预测法，自动测试工具，COCOMO 模型，估算成本的标准值法，软件管理工具。

(2) 增加了一些较新颖的或较重要的内容。增加的内容主要有：软件过程，与用户沟通获取需求的方法，形式化说明技术，逐步求精，人机界面设计，回归测试，控制结构测试，预防性维护与软件再工程，面向对象测试策略及设计测试用例的方法，COCOMO2 模型，能力成熟度模型 (CMM)。

(3) 用统一建模语言 (UML) 的概念与符号重新改写了讲述面向对象方法学的第 9、10、11、12 章和附录 A。

此外，还对书中许多具体内容作了修改或更新，对文字叙述作了进一步的加工和润色。

与第四版配套出版的还有《软件工程导论学习辅导》，该书共分 10 章，涵盖了教材的主要内容。每章均由三部分组成：第一部分系统扼要地复习本知识单元的重点内容；第二部分给出了与本单元内容密切配合的习题；第三部分是习题解答，对典型题目还详细分析了解题思路。附录给出了三套模拟试题以及参考答案，可供读者在课程学习之后检验学习效果。

为便于教学，本书制作了电子教案。采用本书作为教材的教师，可以从清华大学出版社免费获取电子教案。联系方法请参阅本书后面的

“读者意见反馈卡”。

我的学生张劲松和张展新参与了附录 A 所述的 C++ 类库管理系统的设计和实现工作，张雯和张杰为本书出版做了许多具体工作。谨在此向他们表示感谢！

编者

2003 年 8 月

第三版前言

PREFACE

本书修订版(即第二版)自 1992 年 6 月出版以来,受到更多读者的欢迎,在不足 5 年的时间内共印刷了 13 次,台湾于 1993 年 2 月出版了本书(修订版)的繁体字版。有关部门领导和同行专家也对本书给予了充分肯定和热情鼓励,于 1995 年 12 月把本书(修订版)评为第三届全国普通高等学校工科电子类专业优秀教材一等奖。

5 年来计算机软件技术又有了突飞猛进的进步,特别是面向对象的软件技术在我国已经逐渐进入实用阶段,越来越多的软件工作者开始使用面向对象技术开发自己的应用系统。为了跟踪学科发展方向,更好地为广大读者服务,编者再次对原书作了认真修订,写出了第三版。

在第三版中,编者吸取国内外论述面向对象方法学众多书刊的精华,并结合自己使用面向对象方法从事科学研究、开发软件的经验 and 体会,对面向对象方法作了全面、系统、深入的介绍。第三版新增加了 4 章新内容,结合丰富的实例集中讲述面向对象方法学。此外,新编写的附录 A,讲述了用面向对象方法分析、设计一个简化的 C++ 类库管理系统的过程。读者通过这个实例,不仅可以对面向对象分析和面向对象设计有更具体、更深入的认识,还可以在这份材料的基础上进行用面向对象方法分析、设计和实现软件的实习。

在努力推广新技术的同时,编者认为对传统的结构分析和结构设计技术也应该“一分为二”。事实上,结构分析和结构设计技术现在仍然是人们在开发软件过程中使用得非常广泛的方法,在相当长一段时间内这种方法还会有生命力。特别是作为一本教材来说,更应该把基本概念、原理、技术和方法讲深讲透,因此,在第三版中保留了第二版的基本内容,并且根据编者多年从事软件工程教学和研究的体会,补充了一些必要的内容(例如,软件工程的基本原理,信息建模的概念和方法等)。对于第二版中已经是陈旧了的内容,则坚决删掉。近年来软件蓝图方法应用得并不广泛,因此在第三版中删掉了

介绍这个方法的原第 6 章，并对附录 B 作了相应的修改，不再使用软件蓝图描述设计结果。

我的学生张劲松和张展新用 Visual C++ 4.0 成功地实现了附录 A 中所述的 C++ 类库管理系统。谨在此向他们表示衷心的感谢！

编者

1997 年 6 月

修订版（第二版）前言

PREFACE

本书第一版自 1987 年 6 月出版以来，颇受读者欢迎，不少高校用其作为“软件工程概论”课的教材。然而第一版的内容主要取材于 1985 年以前出版的书籍或文章和著者的实践总结，经过 5 年多的时间，这一学科又有了不少新的进展，为了跟踪学科发展方向，更好地为广大读者服务，编者对原书作了认真修订。

这次修订的指导思想是，保持原书系统性较强，内容比较全面，有丰富的实例与原理性论述紧密配合的特点，增加了反映学科最新发展方向的新内容，同时又适当压缩全书的篇幅，以降低成本，减轻读者负担。为此在修订时对书的取材作了十分慎重的斟酌，删掉了前后重复的内容，去掉了对理解本书的内容不是十分必要的图表和文字，占篇幅比较多的原附录 A 也被割爱了；增加了近年来比较流行的原型法、面向对象的程序设计和软件再用技术的介绍，强调了软件工程与人工智能相结合、互相促进的发展趋势。对附录 B 介绍的编辑程序的功能作了扩充，不仅增加了编辑命令，还增加了输入和编辑汉字的功能。

此次修订还适当加强了对日本软件工程方法的介绍，例如，第 5 章增加了对日立公司发明的 PAD 图的介绍，第 8 章讲述了日立预测法的内容。

另一个比较重要的改动，是使书中使用的术语与近年来国内的习惯用法保持一致，对其他文字也作了进一步的加工和润色。

为便于教学，此次修订在每章后面都附上了适当数量的习题。

编者

1990 年 11 月
于北京信息工程学院

第一版前言

PREFACE

计算机工业发达国家在发展软件的过程中曾经走过不少弯路，受过许多挫折，至今仍然经受着“软件危机”的困扰。人们开发优质软件的能力大大落后于计算机硬件日新月异的进展和社会对计算机软件不断增长的需求，这种状况已经严重妨碍了计算机技术的进步。

为了摆脱软件危机的困扰，一门研究软件开发与维护的普遍原理和技术的工程学科——软件工程学——从20世纪60年代末期开始迅速发展起来了，现在它已经成为计算机科学技术的一个重要分支，一个异常活跃的研究领域。严格遵循软件工程方法论可以大大提高软件开发的成功率，能够显著减少软件开发和维护中的问题。

软件工程学(通常简称软件工程)研究的范围非常广泛，包括技术方法、工具和管理等许多方面，软件工程又是一门迅速发展的新兴学科，新的技术方法和工具不断涌现，真可谓五花八门层出不穷。因此，在一本书中不可能包含软件工程的全部内容。本书是工程学的入门介绍，着重从实用角度讲述软件工程的基本原理、概念和技术方法，同时也尽量注意书的全面性和系统性。希望本书既能对实际的软件开发工作有所帮助，又能为读者在今后深入研究这门学科奠定良好的基础。

本书正文共10章。第1章概括介绍软件工程学产生的历史背景以及它的基本原理、概念和方法。第2章到第9章按软件生存周期的顺序介绍各个阶段的任务、过程、方法和工具。第10章着重讨论软件工程的管理技术。正文后面有两个附录，比较完整地介绍了两个实际软件的开发。附录A着重讲述从问题定义到实现的过程，把这个具体例子和课文前几章的内容结合起来学习，有助于加深对一些基本概念和方法的理解。附录B讲述一个交互式正文编辑程序的设计和描述，它是上机实习的好材料(例如，可以实习把设计翻译成程序、设计测试方案以及维护的方法)。

《软件工程导论》可以为多种读者服务。本书通俗易懂，实例丰富，既有对多种常见方法的全面概括介绍，又有对一种典型方法的深入

详尽介绍，很适合于有一定实践经验的软件工作者和广大计算机用户参考或自学；对于高等院校计算机系高年级本科生和研究生来说，本书可以作为软件工程课程的教材。

本书根据编者最近几年在北京大学校外多次讲授“软件工程概论”课所用的讲义改写而成，改写时充分考虑了在校内外讲授这门课程时广大学员的建议和要求，并且增加了最近收集到的大量新材料。

编者在美国马里蓝大学进修期间，听过该校朱耀汉(Yaohan Zhu)教授讲授的软件工程课，并且在朱教授指导下从事过一些软件开发工作，这些都为本书的编写奠定了基础。本书编写前征求了复旦大学计算机系副教授潘锦平同志对写作大纲的意见，初稿写出后又征求了潘锦平和北大计算机系副主任许卓群等同志对内容编排的意见，承蒙他们给予真诚的鼓励并且提出了许多宝贵的具体建议。此外，本书编写过程中董士海副教授曾经提供了许多很有价值的材料。谨在此向他们表示衷心的感谢！

本书不当之处敬请广大读者不吝赐教。

编者

1986年

目 录

CONTENTS

第 1 章 软件工程学概述	1
1.1 软件危机	1
1.1.1 软件危机的介绍	1
1.1.2 产生软件危机的原因	3
1.1.3 消除软件危机的途径	4
1.2 软件工程	5
1.2.1 软件工程的介绍	5
1.2.2 软件工程的基本原理	7
1.2.3 软件工程方法学	9
1.3 软件生命周期	11
1.4 软件过程	14
1.4.1 瀑布模型	15
1.4.2 快速原型模型	16
1.4.3 增量模型	17
1.4.4 螺旋模型	19
1.4.5 喷泉模型	21
1.4.6 Rational 统一过程	22
1.4.7 敏捷过程与极限编程	25
1.4.8 微软过程	29
1.5 小结	31
习题 1	32
第 2 章 可行性研究	35
2.1 可行性研究的任务	35
2.2 可行性研究过程	36
2.3 系统流程图	38

2.3.1	符号	38
2.3.2	例子	38
2.3.3	分层	40
2.4	数据流图	40
2.4.1	符号	40
2.4.2	例子	42
2.4.3	命名	44
2.4.4	用途	45
2.5	数据字典	47
2.5.1	数据字典的内容	47
2.5.2	定义数据的方法	47
2.5.3	数据字典的用途	48
2.5.4	数据字典的实现	49
2.6	成本/效益分析	49
2.6.1	成本估计	50
2.6.2	成本/效益分析的方法	51
2.7	小结	53
	习题 2	53
第 3 章	需求分析	55
3.1	需求分析的任务	56
3.1.1	确定对系统的综合要求	56
3.1.2	分析系统的数据要求	57
3.1.3	导出系统的逻辑模型	58
3.1.4	修正系统开发计划	58
3.2	与用户沟通获取需求的方法	58
3.2.1	访谈	58
3.2.2	面向数据流自顶向下求精	59
3.2.3	简易的应用规格说明技术	59
3.2.4	快速建立软件原型	61
3.3	分析建模与规格说明	62
3.3.1	分析建模	62
3.3.2	软件需求规格说明	62
3.4	实体-联系图	62
3.4.1	数据对象	63
3.4.2	属性	63
3.4.3	联系	63
3.4.4	实体-联系图的符号	64

3.5	数据规范化	64
3.6	状态转换图	65
3.6.1	状态	65
3.6.2	事件	65
3.6.3	符号	66
3.6.4	例子	66
3.7	其他图形工具	67
3.7.1	层次方框图	68
3.7.2	Warnier 图	68
3.7.3	IPO 图	69
3.8	验证软件需求	70
3.8.1	从哪些方面验证软件需求的正确性	70
3.8.2	验证软件需求的方法	70
3.8.3	用于需求分析的软件工具	71
3.9	小结	72
	习题 3	73
第 4 章	形式化说明技术	75
4.1	概述	75
4.1.1	非形式化方法的缺点	75
4.1.2	形式化方法的优点	76
4.1.3	应用形式化方法的准则	76
4.2	有穷状态机	77
4.2.1	概念	77
4.2.2	例子	79
4.2.3	评价	82
4.3	Petri 网	82
4.3.1	概念	82
4.3.2	例子	84
4.4	Z 语言	85
4.4.1	简介	85
4.4.2	评价	88
4.5	小结	88
	习题 4	89
第 5 章	总体设计	91
5.1	设计过程	91
5.2	设计原理	94

5.2.1	模块化	94
5.2.2	抽象	95
5.2.3	逐步求精	95
5.2.4	信息隐藏和局部化	96
5.2.5	模块独立	97
5.3	启发规则	99
5.4	描绘软件结构的图形工具	102
5.4.1	层次图和 HIPO 图	102
5.4.2	结构图	103
5.5	面向数据流的设计方法	104
5.5.1	概念	104
5.5.2	变换分析	105
5.5.3	事务分析	111
5.5.4	设计优化	112
5.6	小结	113
习题 5	114
第 6 章	详细设计	117
6.1	结构程序设计	117
6.2	人机界面设计	119
6.2.1	设计问题	119
6.2.2	设计过程	121
6.2.3	人机界面设计指南	122
6.3	过程设计的工具	124
6.3.1	程序流程图	124
6.3.2	盒图	125
6.3.3	PAD 图	126
6.3.4	判定表	127
6.3.5	判定树	128
6.3.6	过程设计语言	128
6.4	面向数据结构的设计方法	129
6.4.1	Jackson 图	130
6.4.2	改进的 Jackson 图	131
6.4.3	Jackson 方法	132
6.5	程序复杂程度的定量度量	136
6.5.1	McCabe 方法	137
6.5.2	Halstead 方法	139
6.6	小结	140

习题 6	140
第 7 章 实现	145
7.1 编码	146
7.1.1 选择程序设计语言	146
7.1.2 编码风格	147
7.2 软件测试基础	149
7.2.1 软件测试的目标	150
7.2.2 软件测试准则	150
7.2.3 测试方法	151
7.2.4 测试步骤	151
7.2.5 测试阶段的信息流	152
7.3 单元测试	153
7.3.1 测试重点	153
7.3.2 代码审查	154
7.3.3 计算机测试	155
7.4 集成测试	156
7.4.1 自顶向下集成	157
7.4.2 自底向上集成	158
7.4.3 不同集成测试策略的比较	159
7.4.4 回归测试	160
7.5 确认测试	160
7.5.1 确认测试的范围	160
7.5.2 软件配置复查	161
7.5.3 Alpha 和 Beta 测试	161
7.6 白盒测试技术	162
7.6.1 逻辑覆盖	162
7.6.2 控制结构测试	165
7.7 黑盒测试技术	171
7.7.1 等价划分	172
7.7.2 边界值分析	175
7.7.3 错误推测	175
7.8 调试	176
7.8.1 调试过程	176
7.8.2 调试途径	178
7.9 软件可靠性	179
7.9.1 基本概念	179
7.9.2 估算平均无故障时间的方法	180

7.10 小结	182
习题 7	183
第 8 章 维护	189
8.1 软件维护的定义	189
8.2 软件维护的特点	190
8.2.1 结构化维护与非结构化维护差别巨大	190
8.2.2 维护的代价高昂	190
8.2.3 维护的问题很多	191
8.3 软件维护过程	192
8.4 软件的可维护性	194
8.4.1 决定软件可维护性的因素	194
8.4.2 文档	195
8.4.3 可维护性复审	196
8.5 预防性维护	197
8.6 软件再工程过程	198
8.7 小结	200
习题 8	201
第 9 章 面向对象方法学引论	203
9.1 面向对象方法学概述	203
9.1.1 面向对象方法学的要点	203
9.1.2 面向对象方法学的优点	205
9.2 面向对象的概念	209
9.2.1 对象	209
9.2.2 其他概念	211
9.3 面向对象建模	215
9.4 对象模型	216
9.4.1 类图的基本符号	217
9.4.2 表示关系的符号	218
9.5 动态模型	223
9.6 功能模型	224
9.6.1 用例图	224
9.6.2 用例建模	227
9.7 3 种模型之间的关系	228
9.8 小结	229
习题 9	229