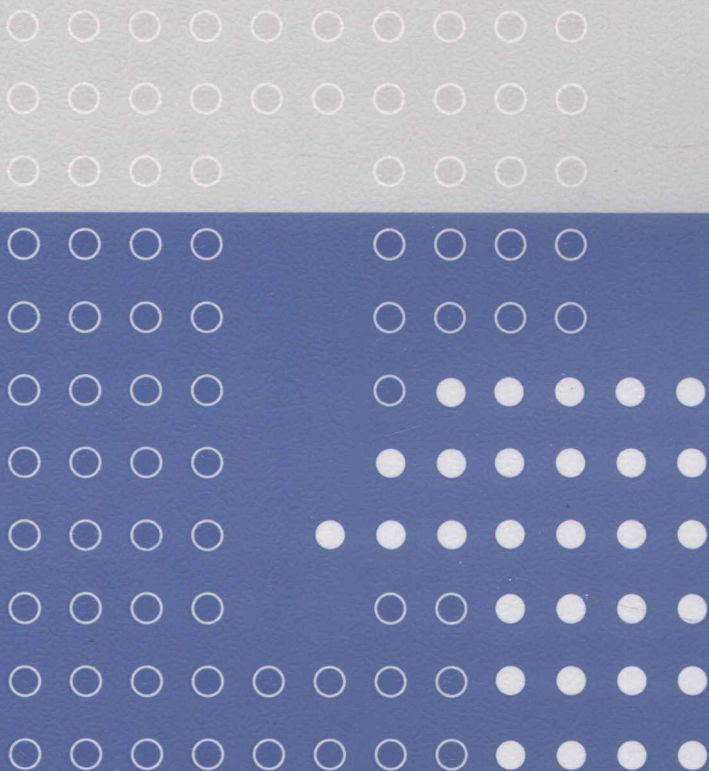




普通高等教育“十一五”国家级规划教材 计算机系列教材

软件工程



赖均 陶春梅 刘兆宏 胡峰 编著

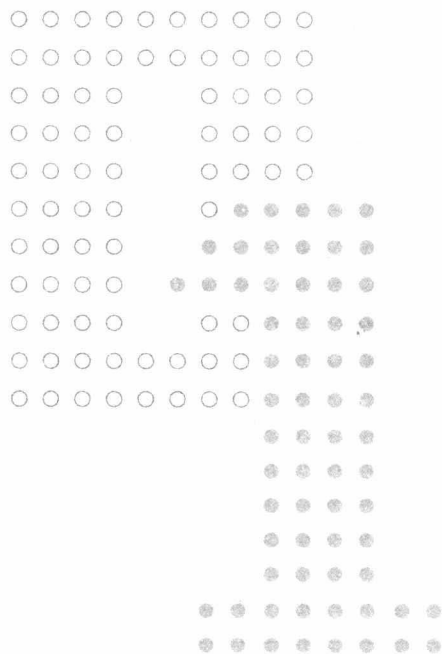
清华大学出版社



计算机系列教材

赖均 陶春梅 刘兆宏 胡峰 编著

软件工程



清华大学出版社
北京

内 容 简 介

本书简明扼要、思路清晰地介绍软件工程的基本技术,使读者在有限的课时里能够了解软件工程的主要框架知识,并掌握基本的软件工程方法。本书以广泛使用的结构化方法详细地阐述面向数据流的需求分析、系统设计、实现以及测试的开发方法和软件的维护;同时,也较为详细地介绍当前主流的面向对象的方法的需求分析、系统设计、实现和测试方法。本书着重于软件工程基本知识和理论的掌握,并同时强调软件工程实践能力的提高和培养。因此,在本书编写中主要关注工程应用知识的掌握和软件工程基本技能的学习,同时本书尽可能覆盖软件资格考试所需的基本软件工程理论知识,具有较大的实用性。

本书适合作为计算机和软件工程及相关专业的本、专科学生学习软件工程课程的教材,也可以作为软件工程实践者的参考资料。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

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/赖均等编著. --北京:清华大学出版社,2016

计算机系列教材

ISBN 978-7-302-41295-3

I. ①软… II. ①赖… III. ①软件工程—高等学校—教材 IV. ①TP311.5

中国版本图书馆CIP数据核字(2015)第196037号

责任编辑:白立军

封面设计:常雪影

责任校对:时翠兰

责任印制:杨 艳

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 装 者:清华大学印刷厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:26.75

字 数:610千字

版 次:2016年1月第1版

印 次:2016年1月第1次印刷

印 数:1~2000

定 价:49.00元

产品编号:058551-01

信息技术的快速发展为社会各个方面带来了深刻的变革,而各种应用软件则是各种信息系统的智能中枢,软件开发、运营管理能力与技术直接影响各行业各方面的信息化建设、发展和应用。为了应对社会发展对软件应用的要求,提出了软件工程,以使软件产业能够良性发展,满足社会进一步发展的要求。随着各种应用系统的快速发展和广泛应用,对软件的需求获取、系统的维护和管理提出了更高的要求。为了不断提高软件开发的质量和软件维护与管理的水平,应系统地学习并掌握、运用软件工程的基本理论和技术,才能使开发出的软件满足当前社会的全面的信息化建设、发展的要求,促进社会经济的全面转型升级和发展。

软件工程是一门指导计算机软件进行开发、维护和管理的新兴学科,主要利用工程概念、原理、技术和方法,在计划、开发、运行、维护与管理软件过程中,将科学的管理和最佳的技术方法紧密结合,从而以比较经济的手段获得满足用户需求的可靠软件的一系列方法。软件工程主要内容涉及如何应用软件开发的基本理论和工程技术来指导对软件系统的开发、维护和管理。软件工程不仅具有一般工程学科的共性,还具有抽象性、复杂性等特性;同时,又具有软件工程学科的系统性、工程化、综合性和学科交叉性的基本特点。软件工程课程是高等学校计算机及信息类相关专业的一门专业主干课程。

软件工程知识涉及计算机科学、工程科学、管理科学、数学等综合交叉多学科的领域,不仅涵盖软件系统的开发方法和技术、维护与管理技术,还包括软件工具、环境及软件开发的规范。目前,软件工程主要包括软件工程原理、软件工程过程、软件工程方法、软件工程技术、软件工程模型、软件工程管理、软件工程度量、软件工程环境、软件工程应用等多方面的内容。多年来,编写者在高等院校和信息技术企业一直从事计算机及信息领域的教学、科研和学科专业管理工作,多次主持或参加了多个项目软件系统开发和相关的研究,积累了大量经验,为满足高校计算机及相关专业人才培养的需要,特编写了本教材。

本书从实用的角度出发,根据教育部高教司审定的《中国计算机科学与技术学科教程2002》中对软件工程课程的要求编写,并参照美国 ACM 和 IEEE Computing Curricula 2007 以及 SWEBOK V3 等关于软件工程方面的内容,吸取了国内外软件工程的工程理论和实用技术、方法。全书共分 12 章,内容涵盖软件工程概述、可行性分析、需求分析、总体设计、软件详细设计、软件实现、软件测试;软件维护、面向对象分析、面向对象设计、面向对象的实现与测试、软件项目管理等。书中列出了很多在校学生相对熟悉的典型例子,以求学生来学习、掌握理论、技术和方法的同时,把它们转化为实践分析和设计能力。课后附有适当的习题,以便课后练习,复习巩固所学知识。

本书重点介绍软件工程技术、方法和实际应用等方面的较为完整的软件工程实践所需的基本知识,为读者以后在这些知识的某一方面进行深入的研究建立一个相对完整的框架。本书的主要特点如下。

(1) 内容先进,结构新颖,吸收了国内外大量的知识、新技术、新方法和国际通用准则。注重科学性、先进性、操作性,图文并茂、学以致用。

(2) 注重实用和特色,坚持实用、规范的原则,突出实用及素质能力培养,列举了大量实际案例,在内容安排上将理论知识与实际应用有机结合。

(3) 资源配套,便于教学。为了方便教师教学,本书配有电子教案并附有部分习题答案。

本书以软件生命周期过程为主线,全书可分为五个部分:第一部分——软件工程概述(第1章)对软件工程产生的背景、软件工程的目标与任务以及常用软件工程模型和常用的开发方法进行了介绍;第二部分——可行性分析与项目立项(第2章)对软件的可行性分析的基本技术进行了阐述;第三部分——软件工程基本框架(第3~8章)较为系统地阐述了软件开发的各阶段的方法和过程以及工具和软件的维护;第四部分——面向对象的软件工程对面向对象的软件开发方法进行了系统的介绍(第9~11章);第五部分——项目管理(第12章)简要介绍了项目管理的基本过程和相关概念。书中还包含部分学生比较熟悉的项目实例和适量必要的课后习题,这些都是编者在软件开发和教学实践经验的沉淀。

本书由赖均主编、统稿并编写第2~6章以及第9章和第10章。陶春梅编写第7章和第11章,刘兆宏编写第8章和第12章,胡峰编写第1章,王莹、赵正敏参加了编著审校和修改等工作,汪俊、邢浩然、李鸿垚对全书的文字、图表进行了核对编排及查阅资料等,并完成了大部分插图的制作。

非常感谢清华大学出版社的广大员工,为本书的编写提供了帮助并使本书得以及时出版。同时,感谢对本书编写给予大力支持与帮助的各位技术专家、教师和研究人員以及相关的院校和相关企业单位。对编写过程中参阅的大量重要文献资料难以完全准确注明,在此深表诚挚谢意!

由于水平有限,书中难免存在不妥之处,敬请见谅! 欢迎提出宝贵意见和建议。

欢迎联系,作者邮箱: laijun@163.com。

赖均

2015年6月于重庆

第一部分 软件工程概述

第 1 章 软件工程概述 /3

- 1.1 软件与软件危机 /3
 - 1.1.1 软件技术的变化与发展 /3
 - 1.1.2 软件与软件危机的产生 /4
 - 1.1.3 解决软件危机的途径 /9
- 1.2 软件工程 /10
 - 1.2.1 软件工程的定义和研究对象 /10
 - 1.2.2 软件工程的基本目标与原理 /14
 - 1.2.3 软件工程的发展 /15
- 1.3 软件生存周期与软件开发 /16
 - 1.3.1 软件生存周期与软件过程 /16
 - 1.3.2 软件过程模型 /19
 - 1.3.3 软件开发模型的选定 /27
 - 1.3.4 软件开发方法简述 /29
- 1.4 本章小结 /30
- 1.5 习题 /31

第二部分 可行性分析与项目立项

第 2 章 可行性分析 /37

- 2.1 问题定义与可行性分析 /37
 - 2.1.1 软件问题定义 /37
 - 2.1.2 可行性分析的任务 /41
 - 2.1.3 可行性分析的步骤 /45
 - 2.1.4 软件立项及合同 /47
- 2.2 系统流程图 /49
 - 2.2.1 系统流程图的使用符号 /49
 - 2.2.2 系统流程图应用举例 /50

- 2.3 成本与效益分析 /51
 - 2.3.1 成本估计 /51
 - 2.3.2 可参考的成本计算案例 /53
- 2.4 软件可行分析报告概述 /54
- 2.5 本章小结 /55
- 2.6 习题 /56

第三部分 软件工程基本框架

第3章 需求分析 /61

- 3.1 需求分析的概念和任务 /61
 - 3.1.1 需求分析的概念 /61
 - 3.1.2 需求分析的目标和任务 /62
 - 3.1.3 需求分析的原则 /65
- 3.2 获取需求的方法 /66
 - 3.2.1 常用需求获取技巧与方法 /66
 - 3.2.2 需求分析方法分类 /67
 - 3.2.3 需求分析的过程与具体步骤 /70
- 3.3 需求描述工具与方法 /74
 - 3.3.1 需求描述概述 /74
 - 3.3.2 数据流图及数据字典 /74
 - 3.3.3 处理或加工描述 /80
- 3.4 验证软件需求 /81
 - 3.4.1 需求验证与评审 /81
 - 3.4.2 验证软件需求的方法 /82
- 3.5 需求实例与规格说明 /83
 - 3.5.1 需求实例 /83
 - 3.5.2 需求规格说明书概述 /86
- 3.6 本章小结 /87
- 3.7 习题 /87

第4章 总体设计 /92

- 4.1 软件总体设计的任务和步骤 /92
 - 4.1.1 软件总体设计的任务 /92
 - 4.1.2 总体设计步骤 /93
- 4.2 总体设计的原则和方法 /95
 - 4.2.1 总体设计的原则 /95
 - 4.2.2 描述软件结构的方法 /101
 - 4.2.3 软件总体设计的常用方法 /103
- 4.3 总体设计过程 /107
 - 4.3.1 软件总体设计过程 /107
 - 4.3.2 模块接口设计 /116
 - 4.3.3 结构设计优化规则 /117
- 4.4 数据库设计概述 /119
 - 4.4.1 数据库设计步骤 /119
 - 4.4.2 概念数据模型设计 /119
 - 4.4.3 逻辑结构及物理结构设计 /120
- 4.5 用户界面设计 /122
 - 4.5.1 界面设计的意义 /122
 - 4.5.2 界面设计的任务和原则 /123
 - 4.5.3 界面设计的基本方法 /125
- 4.6 总体设计说明书概述 /128
- 4.7 本章小结 /130
- 4.8 习题 /130

第5章 软件详细设计 /133

- 5.1 详细设计的任务和原则 /133
- 5.2 详细设计的方法和工具 /134
 - 5.2.1 详细设计的方法 /134
 - 5.2.2 详细设计的描述工具 /136
 - 5.2.3 详细设计工具选择 /141
- 5.3 程序复杂程度的度量方法 /141
 - 5.3.1 McCabe 方法 /142

5.3.2	Halstead 方法	/143
5.4	详细设计说明书及评审	/145
5.5	本章小结	/146
5.6	习题	/146
第 6 章	软件实现	/150
6.1	软件实现技术概述	/150
6.1.1	软件实现的过程及任务	/150
6.1.2	软件实现的准则	/151
6.1.3	软件实现策略与管理	/152
6.2	软件程序设计语言	/154
6.2.1	程序设计语言的发展及分类	/154
6.2.2	选择程序设计语言的标准	/157
6.3	程序设计风格	/161
6.3.1	文档化源程序	/161
6.3.2	数据定义	/162
6.3.3	语句结构和输入输出	/163
6.4	程序效率	/164
6.4.1	程序的效率介绍	/164
6.4.2	存储和输入输出效率	/165
6.5	常用软件实现方法	/165
6.5.1	模块化程序设计	/165
6.5.2	结构化程序设计	/166
6.5.3	面向对象程序设计	/166
6.5.4	常用程序设计工具	/167
6.5.5	软件实现的文档	/170
6.6	本章小结	/171
6.7	习题	/171
第 7 章	软件测试	/174
7.1	软件测试的目的和原则	/174
7.1.1	软件测试的意义	/174

- 7.1.2 软件测试的概念及内容 /174
- 7.1.3 软件测试的目的和原则 /176
- 7.1.4 软件测试涉及的对象和测试信息流 /176
- 7.1.5 测试与软件开发各阶段的关系 /178
- 7.2 软件测试的方法和过程 /179
 - 7.2.1 静态测试和动态测试 /179
 - 7.2.2 软件测试用例设计 /180
 - 7.2.3 软件测试的过程 /204
 - 7.2.4 软件测试策略 /205
- 7.3 软件测试的步骤 /206
 - 7.3.1 单元测试 /207
 - 7.3.2 集成测试与集成 /209
 - 7.3.3 确认测试及内容 /213
 - 7.3.4 系统测试及验收 /214
 - 7.3.5 其他测试种类 /216
- 7.4 软件测试用例与测试标准 /219
 - 7.4.1 软件测试用例设计及方法 /219
 - 7.4.2 软件测试标准和工具 /222
 - 7.4.3 软件测试文档与可测试性 /223
 - 7.4.4 测试的可靠性分析 /224
- 7.5 本章小结 /225
- 7.6 习题 /225

第8章 软件维护 /232

- 8.1 软件维护的概述 /232
 - 8.1.1 软件维护的定义与分类 /232
 - 8.1.2 软件维护代价与工作量 /233
- 8.2 软件维护的特点和过程 /234
- 8.3 软件维护的任务与策略 /237
- 8.4 可维护性 /239
 - 8.4.1 可维护性因素 /239

- 8.4.2 提高软件的可维护性方法 /240
- 8.5 自动化运维 /241
 - 8.5.1 自动化与问题 /241
 - 8.5.2 运维自动化迫切性 /243
 - 8.5.3 运维自动化的内容与工具 /244
 - 8.5.4 建立运维自动化管理的步骤 /245
- 8.6 逆向工程和再工程 /246
 - 8.6.1 软件的逆向工程和再工程 /246
 - 8.6.2 软件再工程过程与技术 /249
- 8.7 本章小结 /251
- 8.8 习题 /251

第四部分 面向对象的软件工程

第9章 面向对象分析 /255

- 9.1 面向对象分析概述 /255
 - 9.1.1 面向对象方法学 /255
 - 9.1.2 面向对象的概念 /263
 - 9.1.3 面向对象建模基础 /268
 - 9.1.4 面向对象分析方法 /281
- 9.2 建立需求模型 /283
 - 9.2.1 用例描述与模型建立 /283
 - 9.2.2 用例关系与细化 /287
- 9.3 建立对象模型 /287
 - 9.3.1 系统划分和对象识别 /287
 - 9.3.2 识别类关系 /290
- 9.4 建立动态模型和定义操作 /292
 - 9.4.1 设想用户界面 /292
 - 9.4.2 建立动态模型 /292
 - 9.4.3 面向对象分析案例 /294
- 9.5 本章小结 /297
- 9.6 习题 /298

第 10 章	面向对象设计	/302
10.1	面向对象设计的原则、任务及过程	/302
10.1.1	面向对象设计原则的应用	/302
10.1.2	面向对象设计的基本任务	/304
10.1.3	面向对象的设计过程和 基本规则	/305
10.2	软件重用	/307
10.3	面向对象设计的系统分解	/309
10.4	设计问题域与人-机交互部分	/312
10.4.1	类的调整与修改	/312
10.4.2	设计问题域部分	/315
10.4.3	设计人-机交互部分	/316
10.5	设计任务管理部分	/318
10.6	设计数据管理部分	/320
10.6.1	选择数据存储管理模式	/320
10.6.2	设计数据管理部分	/321
10.7	对象设计的过程	/322
10.7.1	对象描述与类设计	/322
10.7.2	设计类中的关联	/324
10.8	设计优化和设计实例	/325
10.8.1	设计优化	/325
10.8.2	设计实例	/328
10.9	设计模式简介	/331
10.9.1	设计模式的设计原则	/331
10.9.2	模式命名与基本模式	/332
10.9.3	设计模式应用步骤	/334
10.10	本章小结	/335
10.11	习题	/335

第 11 章	面向对象实现与测试	/339
11.1	面向对象程序设计语言	/339
11.1.1	面向对象设计的实现	/339
11.1.2	面向对象语言的技术特点	/340
11.1.3	面向对象语言的选择	/343
11.2	面向对象程序设计风格	/344
11.2.1	提高可重用性	/344
11.2.2	提高可扩充性和健壮性	/346
11.3	面向对象测试	/347
11.3.1	面向对象的测试与传统测试 的比较	/347
11.3.2	面向对象测试模型	/348
11.3.3	面向对象模型的检查与评估	/351
11.3.4	面向对象的单元测试	/352
11.3.5	面向对象的集成测试与 确认测试	/354
11.4	本章小结	/358
11.5	习题	/359

第五部分 项目管理

第 12 章	软件项目管理	/363
12.1	软件项目管理概述	/363
12.1.1	软件项目管理的特点	/363
12.1.2	软件项目管理及过程	/366
12.2	项目管理内容及常见工具简介	/367
12.2.1	软件项目管理的主要内容	/367
12.2.2	常用工具简介	/368
12.3	软件项目管理过程	/369
12.3.1	项目初始阶段	/369

12.3.2	项目计划阶段	/370
12.3.3	项目执行控制	/394
12.3.4	软件项目结束	/399
12.4	软件能力成熟度模型简介	/401
12.5	本章小结	/409
12.6	习题	/409

参考文献	/414
------	------

第一部分

软件工程概述

第 1 章 软件工程概述

本章学习目标

- (1) 了解软件工程的产生和发展。
- (2) 掌握软件工程的概念、内容和原理。
- (3) 熟悉软件生存周期及阶段任务。
- (4) 掌握软件开发模型。
- (5) 了解几种常见的软件开发方法。

本章首先介绍软件与软件危机,对软件技术的发展、软件危机的出现和解决软件危机的方法和途径进行了阐述;然后对软件工程的出现、软件工程的定义和研究内容以及软件工程的基本目标、原理和原则进行了介绍;接着对软件过程和软件生命周期进行了阐述,并简要介绍几种常见的软件开发过程模型;最后,对几种广泛应用的软件开发方法进行简述。

1.1 软件与软件危机

1.1.1 软件技术的变化与发展

纵观软件技术的发展历史,软件技术发展与计算机硬件平台、人的思维方式和问题呈现的基本特征密切相关,同时它们也驱动软件技术不断向前发展。可以这样说,高级语言的发展是为了避免程序的设计涉及计算机硬件平台的复杂性问题,结构程序设计的发展是为了控制程序开发过程和执行过程的复杂性,面向对象方法则是为了控制系统需求易变所导致的复杂性。

结构化程序设计技术的出现,使得任意计算流程均可用顺序、循环和条件 3 种基本控制结构来描述,而函数或过程等程序元素的出现又为信息隐蔽和封装提供了机制,这样的软件模型不仅使基本计算单位增大,而且也使控制流程结构化。它是在较长一段时间内在软件界占据主流的软件模型。模块化、抽象数据类型等概念的出现也是人们在提高基本计算单位大小和抽象度方面努力的结果,但是这些概念并没有在编程语言层次上的对应体。在这一时期,虽然一些个别的研究探索性语言提供了模块或抽象数据类型的机制,但对大多数语言而言,还只能用其本身的基本结构来模拟这种机制。

面向对象技术目前之所以流行是因为它较好地综合了软件开发的 3 个要素:计算机平台、人的思维模式和问题的特征。面向对象技术的基本建模元素是对象及对象间的交互,这种看待现实世界的观点既符合人的思维模式,又符合客观世界的构成规律,因此,能够达到问题空间、设计空间和程序空间之间的直接映射,从而能更容易地控制需求变化所