

重点大学软件工程规划系列教材

软件工程实用教程

吕云翔 编著

清华大学出版社



重点大学软件工程规划系列教材

软件工程实用教程

吕云翔 编著

清华大学出版社

北京

内 容 简 介

本书按照典型的软件开发过程来组织内容,旨在培养读者具备软件工程思想及实际软件开发的能力。全书共12章,主要内容包括软件工程与软件过程、软件需求分析与建模、软件设计与建模、软件编程与软件测试以及软件工程的其他相关内容,如软件规模估算、进度计划、人员组织、软件开发风险管理等。

本书可以作为普通高校计算机相关专业“软件工程”课程的教材,也可以供学习软件工程的读者单独使用(包括参加计算机等级考试或相关专业自学考试)参考。

本书是高等院校计算机科学、软件工程及相关专业“软件工程”课程的理想教材,也可以供程序员、软件测试工程师、系统工程师以及软件项目经理等相关人员阅读参考。

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

软件工程实用教程/吕云翔编著.--北京:清华大学出版社,2015

重点大学软件工程规划系列教材

ISBN 978-7-302-39469-3

I. ①软… II. ①吕… 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036499号

责任编辑:魏江江 薛 阳

封面设计:傅瑞学

责任校对:时翠兰

责任印制:李红英

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:北京国马印刷厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:25 字 数:600千字

版 次:2015年4月第1版 印 次:2015年4月第1次印刷

印 数:1~2000

定 价:49.00元

产品编号:061696-01

前 言

软件工程是研究软件开发和管理的一门工程科学,是计算机学科及相关专业的主干课程,也是软件从业人员必不可少的专业知识。目前,软件工程的理论在我国各企业和部门的软件开发中得到了广泛的应用,软件工程课程也受到了各大专院校相关专业的重视,各种软件工程理论教材也很多。

软件工程是一门理论与实践并重的课程。本书在讲述软件工程的基本概念、原理和方法的基础上,详细而全面地介绍了可以实际用于软件开发实践的各种技能。旨在使学生通过有限课时的学习后,不仅能对软件工程的原理有所认识,而且能具备实际开发软件的各种技能,比如熟练使用各种软件工程工具,按照标准和规范编写文档等。

本书共分为 12 章,内容涉及软件工程的基本原理和概念、软件开发生命周期的各个阶段、软件工程管理的相关内容、如何使用各种自动化工具来辅助软件开发的过程,以及课程设计。

本书与其他书的不同之处在于,除了讲述必要的软件工程理论之外,还通过实验讲述比较流行的软件开发工具(如软件绘图工具 Microsoft Visio、面向对象建模工具 Rational Rose、软件实现工具 Visual Studio、软件测试工具 UnitTest、软件项目管理工具 Microsoft Project 等)是如何应用于“小型网上书店系统”的。书中还讲述了课程设计的指导,如怎样进行项目选题、组建团队、团队工作方式和项目进度安排等。本书中所讲述的“小型网上书店系统”是学生在做课程设计中可供模仿的一个项目。所有这些使得软件工程的三要素——过程、方法和工具在本书中能够充分地体现。

本书理论知识的教学安排建议如下。

章节	内 容	学时数
第 1 章	软件工程概述	2~4
第 2 章	软件过程	2
第 3 章	可行性研究及需求分析	2
第 4 章	结构化需求分析	2~4
第 5 章	面向对象分析	6~8
第 6 章	软件设计	2~4
第 7 章	结构化设计方法	4~6
第 8 章	面向对象设计	2
第 9 章	软件编程	1~2
第 10 章	软件测试概述	2
第 11 章	软件测试方法与过程	5~8
第 12 章	软件工程的其它相关内容	2~4

建议先修课程:计算机导论、面向对象程序设计、数据结构、数据库原理等。

建议理论教学时数:32~48 学时。

建议实验(实践)教学时数:16~32 学时。

教师可以按照自己对软件工程的理解适当地删除一些章节,也可以根据教学目标,灵活地调整章节的顺序,增减各章的学时数。

由于软件工程是一门新兴学科,软件工程的教學方法本身还在探索之中,加之作者的水平有限,本书内容难免有疏漏之处。恳请各位同仁和广大读者给予批评指正,也希望各位能将实践过程中的经验和心得与作者交流(yunxianglu@hotmail.com)。

编者

2015 年 1 月

第一部分 软件工程与软件过程

第 1 章 软件工程概述	3
1.1 软件	3
1.1.1 软件的概念	3
1.1.2 软件的特点	4
1.1.3 软件分类	5
1.2 软件危机	5
1.2.1 软件危机的表现	5
1.2.2 软件危机的原因	6
1.3 软件工程	7
1.3.1 软件工程的定义	7
1.3.2 软件工程的基本内容	7
1.3.3 软件工程目标和原则	9
1.3.4 软件工程的知识体系	10
1.4 软件工程的相关工具	11
1.5 软件工程的常用信息源	12
1.6 “小型网上书店系统”案例介绍	12
小结	12
习题	13
第 2 章 软件过程	14
2.1 软件过程概述	14
2.2 软件生命周期	14
2.2.1 软件生命周期的概念	14
2.2.2 传统软件生命周期的各个阶段	15
2.3 软件过程模型	16
2.3.1 瀑布模型	16
2.3.2 快速原型模型	17
2.3.3 增量模型	18
2.3.4 螺旋模型	19

2.3.5 喷泉模型	19
2.3.6 基于组件的开发模型	20
2.3.7 统一软件开发过程模型	21
2.3.8 敏捷模型	22
小结	29
习题	30

第二部分 软件需求分析与建模

第3章 可行性研究及需求分析	35
3.1 可行性研究	35
3.1.1 可行性研究的目的与意义	35
3.1.2 可行性研究的内容	35
3.1.3 可行性研究的步骤	36
3.2 需求分析	37
3.2.1 需求分析的目的与意义	37
3.2.2 需求分析的步骤	38
3.2.3 需求管理	40
3.3 软件开发计划书编写指南	40
3.4 需求规格说明书编写指南	45
3.5 实验	50
3.5.1 Visio 的功能及使用方法介绍	50
3.5.2 Rose 的功能及使用方法介绍	56
小结	62
习题	62
第4章 结构化需求分析	64
4.1 结构化需求分析的方法	64
4.2 结构化需求分析的工具	65
4.2.1 实体关系(E-R)图	65
4.2.2 数据流图	66
4.2.3 状态转换图	67
4.2.4 数据字典	68
4.3 实验: 利用 Visio 绘制“小型网上书店系统”的数据流图	69
小结	73
习题	74
第5章 面向对象分析	76
5.1 面向对象的软件工程方法	76

5.1.1	面向对象的基本概念	76
5.1.2	面向对象的软件工程方法的特征与优势	78
5.2	UML 简介	79
5.2.1	用例图	80
5.2.2	类图和对象图	81
5.2.3	状态图	85
5.2.4	顺序图	86
5.2.5	活动图	87
5.2.6	协作图	89
5.2.7	构件图	89
5.2.8	部署图	90
5.2.9	包图	90
5.3	面向对象分析方法	91
5.3.1	面向对象分析过程	91
5.3.2	面向对象分析原则	92
5.4	面向对象建模	93
5.4.1	建立对象模型	93
5.4.2	建立动态模型	96
5.4.3	建立功能模型	98
5.4.4	三种模型之间的关系	99
5.5	实验	103
5.5.1	利用 Rose 绘制“小型网上书店系统”的用例图	103
5.5.2	利用 Rose 绘制“小型网上书店系统”的类图	108
5.5.3	利用 Rose 绘制“小型网上书店系统”的对象图	111
5.5.4	利用 Rose 绘制“小型网上书店系统”的状态图	112
5.5.5	利用 Rose 绘制“小型网上书店系统”的顺序图	114
5.6	案例：“小型网上书店系统”的需求规格说明书	116
	小结	132
	习题	133

第三部分 软件设计与建模

第 6 章	软件设计	139
6.1	软件设计的基本概念	139
6.1.1	软件设计的意义和目标	139
6.1.2	软件设计的分类	140
6.1.3	软件设计的原则	140
6.2	软件的体系结构	143
6.2.1	软件的体系结构概述	143

6.2.2	软件系统的设计模式	145
6.3	分布式系统结构	148
6.3.1	多处理器体系结构	148
6.3.2	客户/服务器体系结构	148
6.3.3	分布式对象体系结构	151
6.3.4	对等端体系结构	151
6.3.5	代理	152
6.4	体系结构框架	152
6.4.1	模型-视图-控制器	152
6.4.2	模型-视图-表示器	153
6.4.3	J2EE 体系结构框架	154
6.4.4	PCMEF	155
6.4.5	PCBMER	155
6.5	数据库结构设计	156
6.6	人机界面设计	158
6.7	软件设计说明书编写指南	159
	小结	162
	习题	163
第 7 章	结构化设计方法	165
7.1	结构化软件设计方法	165
7.1.1	表示软件结构的图形工具	165
7.1.2	面向数据流的设计方法	166
7.1.3	面向数据结构的设计方法	169
7.2	结构化软件设计工具	171
7.2.1	流程图	171
7.2.2	N-S 图	172
7.2.3	PAD	173
7.2.4	判定表	174
7.2.5	判定树	175
7.2.6	过程设计语言	175
7.3	实验: 利用 Visio 绘制“小型网上书店系统”的结构图	176
	小结	178
	习题	178
第 8 章	面向对象设计	180
8.1	面向对象设计方法	180
8.1.1	面向对象的设计原则	180
8.1.2	面向对象设计的启发规则	181

8.1.3	系统设计	181
8.1.4	对象设计	182
8.2	实验	182
8.2.1	利用 Rose 绘制“小型网上书店系统”的活动图	182
8.2.2	利用 Rose 绘制“小型网上书店系统”的协作图	184
8.2.3	利用 Rose 绘制“小型网上书店系统”的构件图	186
8.2.4	利用 Rose 绘制“小型网上书店系统”的部署图	187
8.3	案例：“小型网上书店系统”的软件设计说明书	189
	小结	206
	习题	206

第四部分 软件编程与软件测试

第 9 章	软件编程	211
9.1	编程语言	211
9.1.1	编程语言的发展与分类	211
9.1.2	选择编程语言需考虑的因素	215
9.2	编程风格	216
9.3	编程人员的基本素质	218
9.4	实验	219
9.4.1	Visual Studio 使用方法介绍	219
9.4.2	利用 Visual Studio 实现“小型网上书店系统”的用户登录模块	235
	小结	241
	习题	242
第 10 章	软件测试概述	243
10.1	软件测试的基本概念	243
10.1.1	软件测试的原则	243
10.1.2	软件测试模型	245
10.2	软件测试的分类	247
10.3	测试用例	248
10.3.1	测试用例编写	248
10.3.2	测试用例设计	248
10.3.3	测试用例场景	248
10.4	测试分析报告编写指南	249
	小结	251
	习题	252

第 11 章 软件测试方法与过程	253
11.1 软件测试方法	253
11.2 黑盒测试	254
11.2.1 等价类划分法	254
11.2.2 边界值分析法	256
11.2.3 错误推测法	257
11.2.4 因果图法	257
11.2.5 决策表法	259
11.2.6 场景法	260
11.2.7 黑盒测试选择	262
11.3 白盒测试	262
11.3.1 代码检查法	263
11.3.2 静态结构分析法	263
11.3.3 程序插桩技术	263
11.3.4 逻辑覆盖法	264
11.3.5 基本路径法	266
11.3.6 白盒测试方法选择	267
11.3.7 白盒测试和黑盒测试比较	268
11.4 软件测试的一般步骤	268
11.5 单元测试	269
11.5.1 单元测试概述	269
11.5.2 单元测试内容	269
11.5.3 单元测试方法	270
11.6 集成测试	270
11.6.1 集成测试概述	270
11.6.2 集成测试分析	271
11.6.3 集成测试策略	271
11.7 系统测试	275
11.7.1 系统测试概述	275
11.7.2 系统测试类型	275
11.8 验收测试	277
11.8.1 验收测试概述	277
11.8.2 验收测试内容	277
11.8.3 α 测试和 β 测试	278
11.9 回归测试	278
11.10 面向对象的软件测试	279
11.11 软件调试	280
11.11.1 调试过程	280

11.11.2 调试途径	281
11.12 实验	281
11.12.1 利用 Visual Studio 中的工具进行单元测试	281
11.12.2 利用 Visual Studio 对“小型网上书店系统”的用户登录模块 进行单元测试	293
11.13 案例：“小型网上书店系统”的测试分析报告	295
小结	305
习题	306

第五部分 软件工程的其他相关内容

第 12 章 软件工程的其他相关内容	311
12.1 软件估算	311
12.1.1 软件估算的概念	311
12.1.2 软件估算的方法	312
12.1.3 软件估算的原则与技巧	314
12.2 软件开发进度计划	314
12.2.1 Gantt 图	315
12.2.2 PERT 图	315
12.3 软件开发人员组织	316
12.3.1 民主制程序员组	316
12.3.2 主程序员组	316
12.3.3 现代程序员组	317
12.4 软件开发风险管理	317
12.4.1 软件开发风险	317
12.4.2 软件开发风险管理	318
12.5 软件质量保证	319
12.5.1 软件质量的基本概念	319
12.5.2 软件质量保证的措施	320
12.6 软件配置管理概述	321
12.6.1 软件配置管理术语	321
12.6.2 配置管理的过程	324
12.6.3 配置管理的角色划分	325
12.7 软件维护	326
12.7.1 软件维护的过程	326
12.7.2 软件维护的分类	328
12.7.3 软件的可维护性	329
12.7.4 软件维护的副作用	329
12.7.5 软件再工程技术	330

12.8	软件工程标准与软件文档	331
12.8.1	软件工程标准	331
12.8.2	软件文档	333
12.9	软件过程能力成熟度模型	335
12.10	软件项目管理	336
12.10.1	软件项目管理概述	336
12.10.2	软件项目管理与软件工程的关系	337
12.11	软件复用	337
12.12	实验	339
12.12.1	Project 的功能及使用方法介绍	339
12.12.2	利用 Project 管理“小型网上书店系统”的开发过程	347
	小结	351
	习题	352
附录 A	课程设计指导	355
附录 B	模拟考试题与参考答案	360
附录 C	部分习题参考答案	373
参考文献	386

第一部分

软件工程与软件过程

第 1 章 软件工程概述

第 2 章 软件过程

软件工程概述

本章目标

- 了解软件的概念,理解软件的特点。
- 了解软件危机产生的原因和表现,明确软件工程的重要性。
- 掌握软件工程的基本内容。
- 熟悉软件开发的方法。
- 熟悉与软件开发项目相关的常用工具。
- 了解软件工程的常用信息源。

1.1 软 件

1.1.1 软件的概念

软件是计算机系统的“思维中枢”,在计算机系统中起着举足轻重的作用。它与计算机硬件相互作用,互相配合,从而实现了特定的系统功能。计算机软件的概念是随着计算机技术的发展而发展的。

在计算机发展初期,软件就是指程序,即计算机可以识别的源代码或机器可直接执行的代码。当时,软件的作用并没有得到足够的重视。使用者一般需要直接操纵计算机硬件,程序是为某个特定问题而专门设计的。

随着计算机技术的发展,人们越来越充分认识到高质量的软件会使计算机系统的功能和效率大大地提高。高质量、多功能的软件使得计算机的应用从单一的科学计算扩展到多个领域,比如数据处理、实时控制等。随着计算机应用的日益广泛,软件日益复杂,规模日益增大,人们开始意识到软件并不等于程序。

全面地讲,软件由如图 1-1 所示的三部分组成。

(1) 计算机程序,即人们为了完成特定的功能而编制的一组指令集。

(2) 数据,即程序能处理的具有一定数据结构的信息。

(3) 软件文档,它是与程序的开发、维护和使用有关的图文资料,如软件开发计划书、需求规格说明书、设计说明书、测

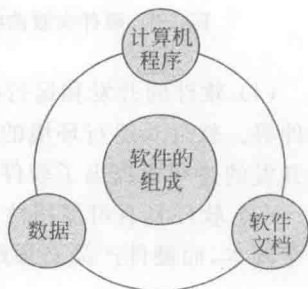


图 1-1 软件的组成

试分析报告和用户手册等。

1.1.2 软件的特点

任何事物都有自己的特点,这是区别于其他事物的根本。理解事物的特点有利于人们更加深刻、更加准确地认识事物的本质。作为计算机系统的重要组成部分,计算机软件的功能依赖于计算机硬件的支持。与硬件相比,计算机软件具有以下一些特点。

(1) 计算机硬件是实物产品,是有形的设备,具有明显的可见性。但是,人们却无法直接观察计算机软件的物理形态,只能通过观察它的实际运行情况来了解它的功能、特性和质量等。

(2) 人们在分析、设计、开发、测试软件产品的过程中,以及在软件开发项目的管理过程中,渗透了大量的脑力劳动。可以说,人类的逻辑思维、智能活动和技术水平是生产软件产品的关键。而传统意义上的硬件制造,除了人类的脑力劳动外,还需要大量的体力劳动。

(3) 在计算机系统运行的过程中,计算机硬件存在着磨损和老化的现象,这也是一切物理器件都存在的普遍现象。但是,硬件设备磨损后,人们可以简单地用另一个硬件设备替换。对于计算机软件而言,不存在像硬件一样的磨损和老化现象,因为它不会受到引起硬件磨损的环境因素(比如温度、振动、灰尘和阳光等)的影响。但是软件却存在着缺陷维护和技术更新的问题。人们对软件的缺陷进行维护或者对技术进行更新的时候,往往要对软件的设计和编码进行改动,这个过程会比简单的硬件替换复杂得多。图 1-2 和图 1-3 分别展示了硬件的失效率与使用时间的关系以及软件的失效率与使用时间的关系。

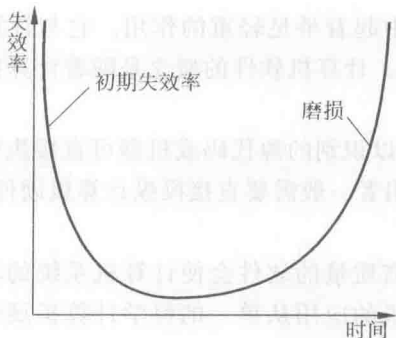


图 1-2 硬件失效率曲线图

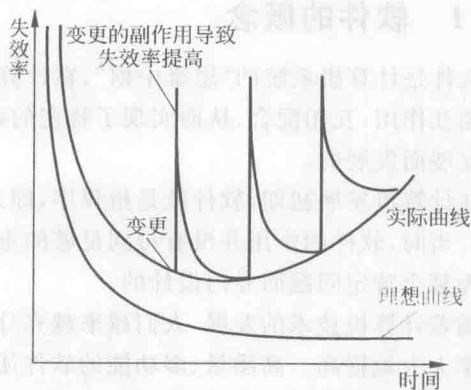


图 1-3 软件失效率曲线图

(4) 软件的开发和运行必须依赖于特定的计算机系统环境,比如硬件、网络配置和支撑软件等。软件对运行环境的这种依赖性是一般产品所没有的。为了减少这种依赖性,在软件开发的过程中提出了软件的可移植性。

(5) 软件具有可复用性。软件一旦被开发出来,便可以很容易地被大量复制,从而形成多个副本,而硬件产品必须经过完整的生产周期才能得到。