

现代软件工程

孙 涌 等 编著
徐汀荣 审

突出应用主题

具备理论先进性与后续性

贯彻可行性与熟练技巧培养



北京希望电子出版社

Beijing Hope Electronic Press

www.bhp.com.cn

21世纪高等院校计

现代软件工程

孙 涌 等 编著
徐汀荣 审

突出应用主题

具备理论先进性与后续性

贯彻可行性与熟练技巧培养



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

内 容 简 介

本书是教育部世界银行贷款 21 世纪初高等理工科教育教学改革项目中《以培养创新人才为目标的地方院校计算机科学与技术专业教学改革的研究与实践》项目成果之一,是“21 世纪高等院校计算机技术教学”丛书中的一本。

《现代软件工程》一书是根据高等教育理工科计算机类软件工程课程教学要求编写的。本书共十四章,取材新颖,内容包括软件工程中面向过程、面向对象的开发方法、技术度量、质量保证及软件项目计划与管理、用统一建模语言 UML 开发软件的方法等等。另外,对软件工程的最新进展进行了讨论。本书的特点是既注重科学性和系统性,又很注重实用、新颖性。在详细论述概念和原理的同时,还介绍了典型的技术和例子。书后附录是两个软件的开发实例,对读者深入了解软件工程会起到一定的帮助作用。各章均配有适量的习题,以帮助学生加深对所学内容的消化和理解。

本书既是高等院校计算机及相关各专业本科生的教材或教学参考书,也可作为研究生的教材。同时亦适用于广大软件技术人员及有关人员在工作及学习中的参考用书。

系 列 书 名 : 21 世纪高等院校计算机技术教学丛书 (7)

书 名 : 现代软件工程

总 策 划 : 北京希望电子出版社

文本著作者 : 孙涌 田茵 尚鲜莲 卢晓东 李心科 编著 徐汀荣 审

责 任 编 辑 : 杨敏

出版、发行者 : 北京希望电子出版社

地 址 : 北京市海淀区知春路 63 号卫星大厦三层 100080

网址: www.bhp.com.cn

E-mail: lxr@bhp.com.cn

电话: 010-62520290, 62521724, 62528991, 62630301, 62524940, 62521921, 82610344 (发行)
010-82675588-202 (门市) 010-82675588-501, 82675588-201 (编辑部)

经 销 : 各地新华书店、软件连锁店

排 版 : 希望图书输出中心 杜海燕

文本印刷者 : 北京广益印刷有限公司

开本 / 规格 : 787 毫米×1092 毫米 16 开本 28 印张 631 千字

版次 / 印次 : 2002 年 8 月第 1 版 2002 年 8 月第 1 次印刷

印 数 : 0001~5000 册

本 版 号 : ISBN 7-900101-70 -5

定 价 : 35.00 元

说明: 凡我社产品如有残缺, 可执相关凭证与本社调换。

21世纪高等院校计算机技术教学丛书

编委会成员名单

顾问委员：（以姓氏笔划为序）

刘 璟 教授 南开大学

何炎祥 教授 武汉大学

施伯乐 教授 复旦大学

钱培德 教授 苏州大学

主任委员：左孝凌 教授 上海交通大学

副主任委员：徐汀荣 教授 苏州大学

委员：（以姓氏笔划为序）

左孝凌 教授 上海交通大学

朱巧明 教授 苏州大学

宋方敏 教授 南京大学

余雪丽 教授 太原理工大学

徐汀荣 教授 苏州大学

陶树平 教授 同济大学

崔志明 教授 苏州大学

曾 明 教授 西安交通大学

谢康林 教授 上海交通大学

序

本丛书是苏州大学主持的高等院校计算机技术教学丛书，其编委会聚集了国内很多院校的专家，编委会为本丛书的组织、审定起到集思广益、推波助澜的作用。

苏州大学建校百年，蜚声海内外，学校治学严谨、理论底蕴深厚，理科各系，久负盛名。近年来学校多次调整院系，全校各系蓬勃发展，尤其是电子信息各科发展迅速。当前信息时代，信息技术的应用已经深入各行各业，而苏州大学更以信息技术的应用作为各个专业的发展之本。

发展信息技术，苏州大学具有以下三个方面的独特优势：

一是苏州大学计算机工程系成立于1987年10月，至今已15年，这正是信息科学突飞猛进的时期。当前计算机技术的应用已经遍及生产、生活各个方面，苏州大学计算机工程系顺时应势，使全系的教学科研迅速崛起。目前系内除有211工程重点建设的计算机应用学科外，还有江苏省重点实验室（计算机信息处理技术实验室）以及纵横汉字信息技术研究所、多媒体应用技术研究所等，在国内外极具影响的研究机构。苏州大学计算机工程系现有博士点一个、硕士点两个，这些都显示了该系具有强劲的得天独厚的科研教育环境。

二是苏州大学地处“天堂”苏州，南邻上海，北依南京，近年新加坡和中国在苏州合建工业园区，而世界著名信息公司Motorola也移师毗邻，使苏州成为其主要研发基地。现在苏州已经成为世界信息技术产业中心之一，信息技术产业产值已达一千亿元以上。苏州大学身临其境，近水楼台，耳濡目染，凭借着培养人才的基础以及软件开发的优势，为新加坡工业园区的建设以及Motorola的芯片研发等项目进行了广泛的优势互补的合作。

三是十几年来信息技术一直是苏州大学发展重点，校长钱培德教授是国内著名信息技术学科的学术带头人，强将手下无弱兵。在苏州大学有一批奋发有为、敬业创新在教学科研上崭露头角的中青年教师队伍，他们把长期从事科学研究的经验与信息教学的实践紧密结合起来编写了一套以理论底蕴为基础，面向应用为目标，充分反映当前计算机进展的实用教材。本丛书即是重要组成部分。

本丛书包括现代软件工程、单片机技术、数据库系统及SYBASE技术、计算机组网技术、计算机图形图像技术、Delphi程序设计、Java程序设计、计算机网页设计等。这些教材有的是计算机专业核心教材，有的是电子类相关专业的实用性应用教材。这套丛书的立足点是以计算机教学应用为主体，也能反映当前的发展动向，在理论上具有先进性和后续性，而在教学实践中努力贯彻可行性和熟练技巧培养。为了改革精简教学内容，突出应用主题，我们把网络原理等大课，分解为组网技术和网页设计等不同主题，以便适应有关专业的不同需要。

我们感谢北京希望电子出版社领导对本套丛书的立题、审稿、编辑、出版等做的大量工作。编委会希望广大读者能够指出丛书中的各种疏漏，期望这套丛书能在信息技术日新月异飞速发展的年代起到一点卵石铺路的作用。

高等院校计算机技术教学丛书编委会 左孝凌

2002年8月

前 言

软件工程是计算机学科的主要科目之一。它是从 60 年代末期迅速发展起来的,随着计算机界发生了翻天覆地的变化,软件工程也成了一个异常活跃的研究领域。人们已经认识到,如果有哪个项目不遵循软件工程原则必定会受到实践的惩罚。当然,软件工程学研究范围非常广泛,包括技术方法、工具和管理等许多方面,新的技术方法和工具还在不断涌现。这本教材虽然不可能包含软件工程的全部内容,但是我们编著该教材是本着易懂、实用的原则并结合多年从事软件工程教学、科研实践进行的。另外考虑到软件工程的发展,编入了一些现代软件工程的观念、方法及技术。希望能为广大读者普及软件工程知识起到微薄的作用。

本书共十四章。第一章概括介绍了软件工程学的基本原理、概念和方法。第二章到第五章主要介绍了面向过程软件工程生命周期顺序的前几个阶段的任务、过程、方法和工具。第六到第八章比较完整地介绍了新型的面向对象的开发方法。第九章到第十一章介绍了软件工程中面向过程及面向对象的软件测试、技术度量、质量保证这几个阶段的任务、过程、方法和工具。第十三章着重讨论了软件工程管理技术。第十四章较详细地介绍了目前软件工程的新技术。正文后面有两个附录,较完整地介绍了两个实际软件的开发。

由于软件工程这门课程涉及面广,内容丰富,发展迅速,所以我们在取材方面,既考虑保持传统的内容,又充分将软件工程中的新技术、新发展补充、吸收到教材中。同时也兼顾到目前高校学生的实际情况。力争做到取材合理、内容新颖、理论为主、结合实际、重点突出、实用性强。

在每章后给出了一些习题,以供读者学习时参考。

本书由孙涌主编,田茵、尚鲜莲、卢晓东、李心科编著,徐汀荣教授主审。其中第一章、第五章、第十四章及 UML 部分的内容由孙涌编写,第二章、第三章、第四章由卢晓东编写,第六章、第七章、第八章由尚鲜莲编写,第九章、第十章、第十一章、第十三章由田茵编写,第十二章由李心科编写。完稿后,全书由孙涌进行统稿和整理工作。

本教材既可作为高等院校计算机及相关专业本、专科学生的教材或教学参考书,也可作为研究生的教材。同时,亦适用于广大专业技术人员及有关人员学习用的参考资料。读者可以根据自己的不同需求对本教材部分章节内容有所选择地进行学习。

本书是教育部世界银行贷款 21 世纪初高等理工科教育教学改革项目中《以培养创新人才为目标的地方院校计算机科学与技术专业教学改革的研究与发展》成果之一,是由苏州大学主持的高等院校计算机技术教学丛书中的一本。

在编写本教材的过程中崔志明教授、徐汀荣教授、艾兰虹女士曾经提供了许多很有价值的建议、材料及帮助。此外,还得到了宁以风、王红华、刘茗、赵晓静、杨文英的热情帮助,书中的附录就是她们根据软件工程的项目实践而编写的,或根据她们提供的资料完成的。最后,还要提到万勇平、张兴书、周渊华、钟铸、韩江标等同学对本书插图的绘制工作付出了辛勤的劳动,谨在此向他们深表谢意。

由于时间仓促,加之作者水平有限,书中肯定存在许多不足和疏漏之处,敬请广大读者不吝赐教。

编 者

2002 年 4 月

目 录

第1章 软件工程概述	1
1.1 软件工程的基本概念、特点、分类	1
1.1.1 软件的概念、特点	1
1.1.2 软件的分​​类	2
1.1.3 软件工程概述	4
1.2 软件发展和软件危机	6
1.2.1 软件的发展	6
1.2.2 软件危机过程	7
1.3 软件模型	10
1.4 软件工程的基本目标	14
1.5 小 结	15
习题 1	15
第2章 软件需求分析	16
2.1 需求分析的任务	16
2.1.1 基本原理	16
2.1.2 需求的层次	16
2.1.3 需求的开发与管理	17
2.2 需求获取的技术	19
2.2.1 需求分析人员的组成	19
2.2.2 需求的类型	19
2.2.3 获取需求的途径	20
2.3 需求规格说明书	20
2.3.1 需求说明的目的	20
2.3.2 需求说明的方法	20
2.3.3 数据流图	21
2.3.4 数据词典和加工说明	22
2.3.5 需求规格说明书格式	23
2.4 需求分析的过程	23
2.4.1 抽取现实问题的本质	23
2.4.2 改进和优化	24
2.4.3 需求分析的验证	25
2.5 结构化需求分析方法	26
2.5.1 画分层数据流图	26
2.5.2 确定数据定义和加工策略	28
2.5.3 需求分析的复审	29
2.6 原型化需求分析方法	30
2.7 小 结	30
习题 2	31
第3章 软件设计原理	33
3.1 软件设计的任务	33

3.1.1	基本原理	33
3.1.2	实际工作的误区	34
3.2	软件设计的步骤	34
3.2.1	两种设计思路	34
3.2.2	概要设计	35
3.2.3	详细设计	35
3.2.4	设计复审	36
3.3	软件设计的模块化思想	36
3.3.1	模块化	36
3.3.2	分解	38
3.3.3	信息隐藏和局部化	38
3.3.4	模块独立性 (modular independence)	38
3.4	软件设计的原则	46
3.4.1	提高模块独立性	46
3.4.2	调整模块的大小	47
3.4.3	适当的模块深度、宽度、扇出和扇入	47
3.4.4	模块的作用范围应在控制范围之内	48
3.4.5	简化模块接口的复杂程度	49
3.5	小结	49
	习题 3	49
第 4 章	软件设计方法	51
4.1	软件设计常用工具	51
4.1.1	概要设计工具	51
4.1.2	详细设计工具	53
4.2	结构化软件设计方法	55
4.2.1	概念	56
4.2.2	变换分析	58
4.2.3	事务分析	60
4.2.4	设计的后处理	61
4.3	Jackson 设计方法	62
4.3.1	面向数据结构的设计	62
4.3.2	表示数据结构的 Jackson 图	62
4.3.3	表示程序结构的 Jackson 图	65
4.3.4	Jackson 伪代码	65
4.3.5	Jackson 设计方法步骤	66
4.4	面向对象软件设计方法概述	67
4.4.1	什么是面向对象的软件设计方法	67
4.4.2	面向对象分析 OOA	67
4.4.3	面向对象设计 OOD	68
4.5	小结	68
	习题 4	68
第 5 章	编码设计	70
5.1	程序设计语言分类	70

5.2	程序设计语言的特点	71
5.3	编码的目的	73
5.4	写程序的风格	74
5.5	程序设计方法论	77
5.6	面向对象编程概念	77
5.7	面向对象编程语言	81
5.8	小 结	86
	习题 5	86
第 6 章	面向对象技术	87
6.1	面向对象的概念	87
6.1.1	面向对象方法概述	87
6.1.2	面向对象的软件工程	88
6.1.3	面向对象的基本概念	90
6.2	对象的本质及表示	97
6.3	面向对象软件的开发过程	99
6.3.1	应用生存期	99
6.3.2	类生存期	100
6.3.3	面向对象软件的开发过程	104
6.3.4	系统体系结构	105
6.4	面向对象软件项目的管理	106
6.5	统一建模语言 UML 简介 (一)	108
6.5.1	发展历史	108
6.5.2	UML 简介	109
6.5.3	UML 视图简介	110
6.5.4	视图	111
6.5.5	UML 类、构件、部署和协作图中的图标	112
6.5.6	扩展组件	112
6.5.7	各种视图间的关系	113
6.6	小 结	114
	习题 6	114
第 7 章	面向对象分析	115
7.1	面向对象分析 (OOA) 基本内容	115
7.1.1	OOA 的基本原则和任务	116
7.1.2	常用的 OOA 方法	116
7.1.3	OOA 概述	118
7.2	论域分析	119
7.2.1	论域分析介绍	119
7.2.2	论域分析过程	122
7.3	系统分析过程	123
7.3.1	使用实例	123
7.3.2	类—责任—协作者建模	125
7.3.3	定义结构和层次	127
7.3.4	定义主题和子系统	128

7.4	OOA 的 Yourdon 方法	130
7.4.1	面向对象分析的考虑	130
7.4.2	标识类和对象	131
7.4.3	细化类和对象	133
7.4.4	标识属性	134
7.4.5	标识服务	136
7.4.6	标识结构	138
7.4.7	标识主题	139
7.5	统一建模语言 UML 简介(二)	140
7.5.1	静态视图	140
7.5.2	用例图	141
7.5.3	交互视图	142
7.5.4	状态图	144
7.5.5	活动视图	145
7.5.6	物理视图	146
7.5.7	模型管理视图	148
7.6	小 结	148
	习题 7	148
第 8 章	面向对象设计	150
8.1	面向对象设计 (OOD) 概述	150
8.1.1	高层设计	150
8.1.2	类设计的目标和方针	153
8.1.3	通过复用设计类	154
8.1.4	类设计方法	156
8.1.5	类设计的例子	158
8.2	系统设计过程	162
8.2.1	划分分析模型	162
8.2.2	并发性和子系统分配	163
8.2.3	任务管理构件	163
8.2.4	数据管理构件	164
8.2.5	资源管理构件	164
8.2.6	人机界面构件	164
8.2.7	子系统间通信	164
8.3	OOD 的 Yourdon 方法	165
8.3.1	问题论域部分(PDC, Problem Domain Component)的设计	166
8.3.2	用户界面部分的设计	168
8.3.3	任务管理部分的设计	170
8.3.4	数据管理部分的设计	171
8.3.5	程序设计语言的影响	172
8.4	Booch 的方法	172
8.4.1	Booch 方法的设计过程	173
8.4.2	Booch 方法的基本的模型	173
8.5	统一建模语言 UML 简介 (三)	177

8.5.1	项目概述	178
8.5.2	静态分析和设计	178
8.5.3	持久对象设计	180
8.5.4	动态对象设计	181
8.5.5	通用界面设计	183
8.5.6	体系结构设计	186
8.6	小 结	189
	习题 8	189
第 9 章	软件测试	190
9.1	软件测试概述	190
9.1.1	软件测试目标	190
9.1.2	软件测试原则	191
9.1.3	软件测试方法	192
9.1.4	软件测试与软件开发各阶段的关系	192
9.1.5	测试信息流	193
9.1.6	错误分类	193
9.2	软件测试过程与策略	195
9.2.1	单元测试	195
9.2.2	集成测试	197
9.2.3	确认测试	199
9.2.4	平行运行	201
9.3	设计测试方案	201
9.3.1	逻辑覆盖	202
9.3.2	等价划分	206
9.3.3	边界值分析	208
9.3.4	错误推测	209
9.3.5	实用测试策略	210
9.4	对 OOA 和 OOD 模型的测试	213
9.4.1	扩大测试的视角	213
9.4.2	测试 OOA 和 OOD 模型	214
9.5	面向对象的测试策略	215
9.5.1	在 OO 语境中的单元测试	216
9.5.2	在 OO 语境中的集成测试	216
9.5.3	在 OO 语境中的有效性测试	216
9.6	OO 软件的测试用例设计	216
9.6.1	OO 概念的测试用例设计的含义	217
9.6.2	传统测试用例设计方法的可用性	217
9.6.3	基于故障的测试	217
9.6.4	OO 编程对测试的影响	218
9.6.5	测试用例和类层次	219
9.6.6	基于场景的测试设计	219
9.6.7	测试表层结构和深层结构	220
9.7	其他专门环境要求的测试	221

9.7.1	GUI 测试	221
9.7.2	测试文档和帮助设施	222
9.7.3	实时系统测试	223
9.8	小 结	223
习题 9		224
第 10 章	软件的技术度量	226
10.1	软件质量	226
10.1.1	McCall 的质量因素	227
10.1.2	FURPS	229
10.2	软件技术度量框架	230
10.2.1	技术度量的挑战	230
10.2.2	测度原则	230
10.2.3	有效软件度量的属性	231
10.3	分析模型的度量	231
10.3.1	基于功能的度量	231
10.3.2	“撞击值”度量	233
10.3.3	规约质量的度量	234
10.4	设计模型的度量	235
10.4.1	高层设计度量	235
10.4.2	构件级设计度量	237
10.4.3	界面设计度量	239
10.4.4	源代码度量	239
10.4.5	对测试的度量	241
10.4.6	对维护的度量	242
10.5	面向对象度量概述	243
10.5.1	面向对象度量的目的	243
10.5.2	区别性的特征	243
10.6	度量特征	243
10.6.1	局部化	243
10.6.2	封装	243
10.6.3	信息隐蔽	244
10.6.4	继承	244
10.6.5	抽象	244
10.7	对 OO 设计模型的度量	244
10.8	对类设计的度量	244
10.8.1	CK 度量套件	245
10.8.2	Lorenz 和 Kidd 建议的度量	246
10.9	对操作的度量	247
10.10	对面向对象测试的度量	248
10.11	对面向对象项目的度量	248
10.12	小 结	249
习题 10		249
第 11 章	软件维护	251

11.1	系统的变化	251
11.1.1	系统的类型	252
11.1.2	系统生命周期中的变化	253
11.1.3	系统的生命范围	254
11.1.4	开发时间和维护时间	254
11.2	软件的可维护性	256
11.2.1	决定软件可维护性的因素	256
11.2.2	文档	257
11.2.3	可维护性复审	258
11.2.4	提高可维护性的方法	258
11.3	软件维护的任务及过程	260
11.3.1	维护机构	260
11.3.2	编写报告	261
11.3.3	维护流程	261
11.3.4	记录保存	263
11.3.5	评价	264
11.4	软件维护的副作用	265
11.4.1	修改代码的副作用	265
11.4.2	修改数据的副作用	265
11.4.3	修改文档的副作用	266
11.5	反推工程和再生工程	266
11.6	预防性维护	267
11.7	小 结	267
	习题 11	268
第 12 章	软件质量保证	269
12.1	软件质量概述	269
12.1.1	软件质量的定义	269
12.1.2	软件质量评价	270
12.1.3	软件开发中的质量控制	270
12.2	软件质量保证	271
12.2.1	SQA 计划	272
12.2.2	软件质量代价	273
12.2.3	软件工程与软件质量保证	274
12.3	软件质量度量模型	275
12.3.1	有关定义	275
12.3.2	质量度量模型	275
12.3.3	3 种度量模型的比较	277
12.4	软件可靠性	278
12.4.1	基本概念	279
12.4.2	影响软件可靠性的原因	279
12.4.3	软件可靠性模型	280
12.4.4	软件可靠性工程	282
12.5	质量体系的建立和实施	282

12.5.1	ISO9000 标准的产生	282
12.5.2	ISO9000 系列标准的内容	283
12.5.3	ISO9000: 2000 版的特点	284
12.6	CMM: 软件能力成熟度模型	285
12.6.1	CMM 的发展	285
12.6.2	基本概念	286
12.6.3	SW-CMM 的用途	287
12.6.4	CMM 的五级	287
12.6.5	CMM 的内部结构	290
12.6.6	软件过程改进	292
12.7	程序正确性证明	294
12.8	小 结	294
	习题 12	295
第 13 章	软件项目管理与计划	296
13.1	项目管理的概念	296
13.1.1	项目管理过程	296
13.1.2	项目管理的范围	297
13.2	软件过程和项目的度量	297
13.2.1	测度、度量和指标	298
13.2.2	过程和项目领域中的度量	298
13.2.3	软件度量	301
13.2.4	调和不同的度量方法	305
13.2.5	软件质量的度量	305
13.2.6	在软件过程中集成度量	306
13.3	可行性研究	307
13.3.1	可行性研究的任务及过程	307
13.3.2	技术可行性研究	307
13.3.3	经济可行性研究	309
13.3.4	运行可行性研究	311
13.4	软件项目估算	312
13.4.1	对估算的考虑	312
13.4.2	项目计划目标	312
13.4.3	软件的范围	313
13.4.4	资源	313
13.4.5	分解技术	315
13.5	软件开发成本估算	317
13.5.1	软件开发成本估算方法	317
13.5.2	专家判定技术	318
13.5.3	软件开发成本估算的早期经验模型	319
13.6	软件风险管理	321
13.6.1	风险识别	321
13.6.2	风险估计	323
13.6.3	风险评价	324

13.6.4	风险驾驭和监控.....	325
13.7	进度安排.....	326
13.7.1	软件开发小组人数与软件生产率.....	326
13.7.2	任务的确定与并行性.....	327
13.7.3	制定开发进度计划.....	327
13.7.4	项目的追踪和控制.....	329
13.8	软件项目的组织与计划.....	329
13.8.1	软件项目管理的特点.....	329
13.8.2	制定计划.....	330
13.8.3	软件项目组织的建立.....	331
13.8.4	人员配备.....	332
13.9	小 结.....	333
	习题 13	333
第 14 章	软件工程新技术概述	335
14.1	客户/服务器软件工程.....	335
14.1.1	客户/服务器系统的结构.....	335
14.1.2	对 C/S 系统的软件工程.....	338
14.1.3	分析建模问题.....	338
14.1.4	对 C/S 系统的设计.....	338
14.1.5	测试问题.....	341
14.2	构件接口技术.....	343
14.2.1	概述	343
14.2.2	EJB/J2EE 技术	344
14.2.3	COM+技术	352
14.2.4	CORBA.....	362
14.3	软件复用.....	373
14.3.1	管理问题.....	374
14.3.2	复用过程.....	376
14.3.3	领域工程.....	378
14.3.4	建造可复用构件.....	381
14.3.5	分类和检索构件.....	383
14.3.6	软件复用经济学.....	386
14.3.7	复用实例研究.....	387
14.4	小 结.....	389
	习题 14	390
	附录 1	391
	附录 2	407
	参考文献.....	425

第1章

软件工程概述

我们知道，计算机软件是整个计算机系统中具体实现各种功能和操作的核心部分。软件工程即采用工程的概念、原理、技术和方法来开发和维护软件，将工程管理技术成功的经验和思想与具体软件的开发过程、研究技术相结合，形成一整套适合于计算机软件开发的方法、规范和技术。因此，软件工程这门课程，对于从事软件开发研究的专业人员，特别是高层次的管理、分析、开发人员，显得比以往更加重要。

1.1 软件工程的基本概念、特点、分类

1.1.1 软件的概念、特点

计算机软件是程序、数据及相关文档的完整集合。其中，程序是按事先设计的功能和性能要求执行的指令序列，数据是使程序能正常操纵信息的数据结构，文档是与程序开发、维护和使用有关的图文材料。

要深入进行计算机软件的开发和研究，首先要了解计算机软件的特点和计算机软件开发规律。计算机软件可归结具有如下几个共同特点：

(1) 软件是一种逻辑实体，而不是具体的物理实体。因此，它具有抽象性。

(2) 软件的生产与硬件不同，软件是由开发或工程化而形成的，它没有明显的制造过程。对软件的质量控制，必须立足于软件开发方面。软件成为产品之后，其制造只是简单的拷贝而已。

(3) 任何机械、电子设备在运行和使用过程中，其失效率大致遵循如图 1-1 所示的 U 型曲线(即浴盆曲线)。软件的情况与此不同，它不存在磨损和老化问题。然而，它存在退化问题，设计人员必须多次修改(维护)软件，图 1-2(a)给出了软件故障率的理想曲线，图 1-2(b)给出了实际的软件故障率曲线。

(4) 软件的开发和运行往往受到计算机系统的限制，对计算机系统有着不同程度的依赖性。为了解除这种依赖性，在软件开发中提出了软件移植的问题。

(5) 迄今为止，软件的开发尚未完全摆脱手工艺的方式。

(6) 软件本身是复杂的。软件的复杂性可能来自它所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) 软件的成本相当昂贵。软件的研制工作需要投入大量的、复杂的、高强度的脑力劳动，它投入的成本是比较高的。

(8) 相当多的软件工作涉及到社会因素。许多软件的开发和运行涉及机构设置、体制运作及管理方式等问题，甚至涉及到人们的观念和心理，这些因素直接影响到项目的成败。

(9) 从市场上买到的软件，它本身就是一个完整的软件，而不能作为构件再组装成新的

程序。但目前已有大量的支持“软件复用”的软件和中间件作为相对独立的构件。

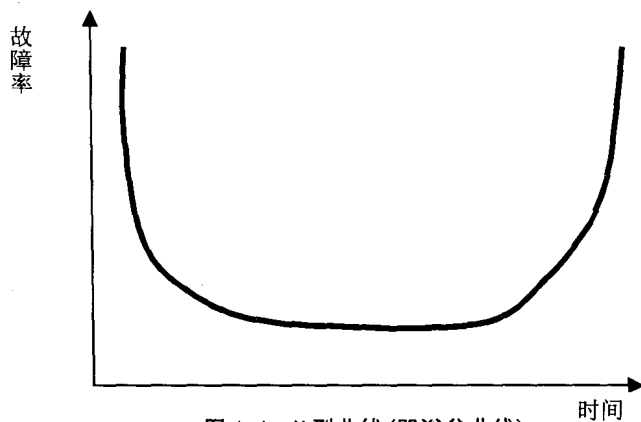


图 1-1 U型曲线(即浴盆曲线)



图 1-2(a) 软件的故障率曲线(理想情况)

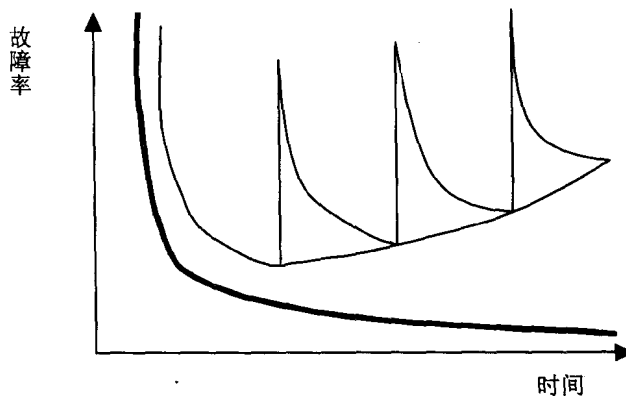


图 1-2(b) 软件的故障率曲线(实际情况)

1.1.2 软件的分类

软件按其功能进行划分,可分为如下几大类: