

高 等 学 校 计 算 机 课 程 规 划 教 材

软件工程

宋 雨 编著



清华大学出版社



高等学校计算机课程规划教材

软件工程

宋雨 编著

清华大学出版社
北京

内 容 简 介

本书以软件开发过程为主线,系统地介绍了软件工程的基本原理、概念和技术方法。全书分16章,包括基本概念、软件计划、软件需求分析、软件设计、程序编码、软件测试、软件维护、软件复用、面向对象的软件工程、软件质量保证、软件的技术度量、软件开发工具与环境、软件工程管理、软件过程管理、基于构件的软件工程(CBSE)和应用 Web 工程,每章后面都附有习题供读者练习之用。

本书的编写基于中国软件工程学科教程的要求,结合作者多年的教学和科研实践并参考了同类文献,包含了软件工程学科的重要分支。本书体系完整,可作为本科生或研究生软件工程课程的教材或参考书,也可供对该学科感兴趣的读者阅读。

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/宋雨编著. —北京:清华大学出版社,2012.9

(高等学校计算机课程规划教材)

ISBN 978-7-302-27758-3

I. ①软… II. ①宋… III. ①软件工程—教材 IV. ①TP311.5

中国版本图书馆 CIP 数据核字(2011)第 280163 号

责任编辑:汪汉友 徐跃进

封面设计:傅瑞学

责任校对:李建庄

责任印制:何 芊

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:三河市金元印装有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:27.25

字 数:679 千字

版 次:2012 年 9 月第 1 版

印 次:2012 年 9 月第 1 次印刷

印 数:1~3000

定 价:46.00 元

产品编号:036722-01

前言

几十年来,伴随着计算机科学技术的进步,软件工程得到了飞速发展,可以说,现在人们已经离不开软件,软件无处不在,这样,如何用工程化的方法设计和开发软件就显得十分重要。在 20 世纪 90 年代初,中国计算机学会教育委员会及全国高等学校计算机教育研究会就把软件工程列为计算机学科中的专业必修课和主干课。之后,中国计算机科学与技术学科教程研究组将其列为核心课程,可见软件工程课程的重要程度和作用是很大的,现在各高校信息类专业基本都设软件工程课程,有的学校甚至设为全校公共选修课,说明对其重视程度是很高的。

为了适应软件工程教育日益重要的形势,作者依据软件工程课程体系及学科要求按照软件的开发过程进行了全书组织。本书体系完整,讲述上尽量使其符合学生学习知识的规律,避免引入高深的理论,每章后都附有习题,学生在认真学完每章内容后,再完成后面的习题,即可达到教学目的并得到较好的学习效果。

本书的知识结构大体上可分为 3 个层次,可用图 1 表示。第一层次是课程的核心,建议重点讲授和学习。第二层次+第一层次构成学科的基础,包括基础理论、实用方法和基本技术。第三层次为学科扩展内容,在学时数能满足的情况下,可安排学习这部分内容。介于第二和第三层次之间的中间层内容既可放在第二层次也可放在第三层次,根据具体情况安排,甚至也可只选择指定章节中的部分内容。

教学建议:对于学时数较少或设为公共选修课的学校,可采用第一层次实施教学的内容,这是该课程的核心,建议教学学时数为 32~40。若教学学时数在 40 以上,建议采用第一层次+第二层次实施教学的内容,这两部分是软件工程的基础。若教学学时数能达到 60 以上,可采用全部内容。

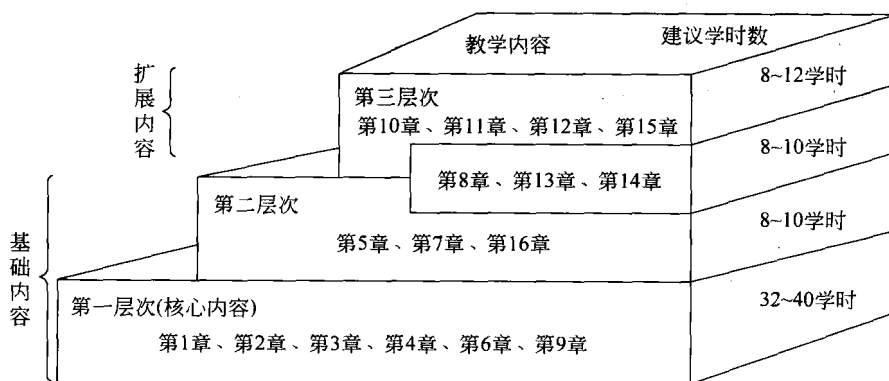


图 1 软件工程知识结构及参考教学学时数

全书共 16 章,第 1 章是概述,从计算机系统的总体开发过程出发,介绍软件、软件工程的概

围、确定软件开发中的资源、估算软件成本(介绍了具体模型和技术)和安排软件开发进度;第3章是软件需求分析,讲解结构化需求分析方法、原型化需求分析方法以及面向对象方法;第4章是软件设计,介绍软件设计的有关概念,讲解了结构化设计方法、面向数据结构的设计方法以及面向对象的设计方法,简要介绍其他新的设计方法,本章还讲解了软件设计工具及程序复杂性的度量等相关内容;第5章介绍程序设计语言的分类、编码风格以及面向对象的编程语言;第6章讲解软件测试的步骤、策略、方法,面向对象的测试以及软件可靠性;第7章介绍软件维护的概念,讲解了软件的可维护性含义、提高可维护性的方法以及软件再工程方法;第8章是软件复用,介绍软件复用概念、领域工程、可复用构件的建造和复用以及面向对象的软件复用技术;第9章是面向对象的软件工程,系统地讲解了面向对象的基本概念,介绍面向对象建模及统一建模语言 UML,讲述了面向对象的软件需求分析及设计过程和方法,给出了2个实例;第10章讲述软件质量保证的有关内容,介绍了 ISO9000 质量标准,给出软件质量预测模型及应用;第11章是软件的技术度量,包括软件分析模型的度量、软件设计模型的度量、源代码度量、软件测试的度量、软件维护活动的度量以及面向对象的度量;第12章介绍软件开发工具与环境;第13章是软件工程管理,讨论软件工程文化、现代件、软件风险分析与管理、软件配置管理、软件工程标准化以及软件的知识产权保护等课题,本章还介绍金尊和先生所提出的软件工程 36 计;第14章是软件过程管理,讲解由 SEI 提出的软件能力成熟度模型(CMM)以及个体软件过程(PSP);第15章讲述基于构件的软件工程(CBSE),包括 CBSE 基本概念、可复用构件的分类与查询、基于构件的软件重用成熟度以及构件技术的应用;第16章是应用 Web 工程,介绍 WebApp 项目的计划、分析、设计及测试。每章最后都有该章内容的小结和习题,便于读者总结和练习。

本书系统地介绍了软件工程学科的重要内容,既能满足软件工程课程的教学需求,同时又能为工程技术人员及管理人员提供参考。在本书编写过程中,郑州轻工业学院的张超钦、尚展垒参加了部分章节的写作。本书作者长期从事软件工程教学和科研,因而,本书也是作者教学和科研成果的总结。由于软件工程是一门快速发展的学科,先进技术不断出现,因而,虽然是基础教程但在书中也尽量对发展中的新技术有所反映,以便学生在学习该课程的过程中能感受到该学科的“脉搏”跳动,使学生能适应学科的发展。书中除参考了教学计划和教学大纲外,也参考了国内外大量同类文献,在此对所有参考文献的作者表示感谢,对华北电力大学的支持表示感谢,感谢清华大学出版社的支持和帮助。

宋 雨
2012 年

目 录

第 1 章 概述	1
1.1 计算机系统的开发过程	1
1.2 软件及其分类	3
1.3 软件工程的由来和发展	4
1.4 软件的生命周期	6
1.5 软件生存期模型	7
1.6 小结.....	13
习题 1	13
第 2 章 软件计划	15
2.1 确定软件范围.....	15
2.2 软件开发中的资源需求.....	15
2.3 软件成本估算.....	17
2.3.1 估算方法	17
2.3.2 估算模型	20
2.3.3 面向对象项目的估算	30
2.3.4 自行开发或购买的决策	31
2.4 软件开发进度的安排.....	32
2.4.1 甘特图	32
2.4.2 PERT 图	33
2.4.3 两种图的结合	34
2.5 小结.....	36
习题 2	36
第 3 章 软件需求分析	41
3.1 需求分析的任务.....	41
3.2 需求规约说明书.....	42
3.3 结构化需求分析方法.....	44
3.3.1 数据流图	44
3.3.2 数据词典	50
3.3.3 功能说明	52
3.4 原型化需求分析方法.....	55
3.4.1 原型开发的分类	56
3.4.2 快速原型技术	58
3.4.3 用户界面开发	61

3.5	面向对象方法	62
3.5.1	Coad/Yourdon 方法	62
3.5.2	Booch 方法	63
3.5.3	Rumbaugh 的 OMT 方法	63
3.5.4	Jacobson 方法	63
3.5.5	UML 方法	64
3.6	小结	67
	习题 3	68
第 4 章	软件设计	70
4.1	软件设计的任务	70
4.1.1	概要设计	70
4.1.2	详细设计	71
4.1.3	软件设计的目标和准则	71
4.2	软件设计的概念和原则	71
4.2.1	软件体系结构	71
4.2.2	软件结构中的若干概念	72
4.2.3	软件的模块化	73
4.3	软件概要设计	76
4.3.1	充分理解 SRS, 确定设计策略	76
4.3.2	模块化准则	77
4.3.3	模块设计中的具体方法	77
4.3.4	概要设计文档	80
4.4	软件详细设计	81
4.4.1	图示工具	81
4.4.2	表格工具	85
4.4.3	语言工具	85
4.4.4	工具的比较	85
4.4.5	程序复杂性的度量	87
4.4.6	详细设计文档	89
4.5	结构化设计方法	91
4.5.1	软件结构图	92
4.5.2	从 DFD 导出软件结构图	92
4.5.3	软件设计的评判	98
4.6	面向数据结构的设计方法	100
4.6.1	Jackson 方法	101
4.6.2	LCP 方法	105
4.7	面向对象的设计方法	107
4.7.1	OOD 基本概念	108
4.7.2	OOD 任务	110

4.7.3	Coad 与 Yourdon 的 OOD 方法	113
4.7.4	层次化 OOD 方法	114
4.8	其他设计方法	116
4.8.1	面向方面程序设计	116
4.8.2	面向 Agent 的设计方法	117
4.8.3	泛型程序设计	117
4.8.4	面向构件的技术	118
4.8.5	敏捷方法	118
4.8.6	Rational 统一过程	118
4.8.7	功能驱动开发模式	118
4.8.8	极端编程	119
4.9	小结	119
	习题 4	121
第 5 章	程序编码	124
5.1	程序设计语言的分类	124
5.1.1	第一代语言——机器语言时代	124
5.1.2	第二代语言——汇编语言时代	124
5.1.3	第三代语言——高级语言时代	124
5.1.4	第四代语言——非过程语言时代	125
5.2	编码风格	125
5.2.1	语句结构	125
5.2.2	源程序文档化	127
5.2.3	数据说明	128
5.2.4	输入和输出	129
5.3	面向对象的编程语言	130
5.3.1	Smalltalk	130
5.3.2	Eiffel	130
5.3.3	SDL	130
5.3.4	C++	131
5.3.5	面向对象的 Pascal	131
5.3.6	Java	131
5.3.7	C#	131
5.4	小结	132
	习题 5	132
第 6 章	软件测试	133
6.1	软件测试基础	133
6.1.1	测试目标	133
6.1.2	测试原则	133

6.1.3	可测试性	134
6.2	测试步骤和策略	134
6.2.1	软件测试过程与开发过程的对应关系	134
6.2.2	测试信息流	135
6.2.3	单元测试	135
6.2.4	集成测试	136
6.2.5	确认测试	141
6.2.6	系统测试	142
6.3	测试用例设计	143
6.3.1	概述	143
6.3.2	逻辑覆盖	144
6.3.3	等价类划分	147
6.3.4	边界值分析	148
6.3.5	划分测试与随机测试	148
6.3.6	错误推测法	148
6.3.7	因果图法	149
6.3.8	人工测试	149
6.3.9	调试	150
6.4	软件可靠性	151
6.4.1	可靠性和可用性	152
6.4.2	影响软件可靠性的因素	152
6.4.3	软件可靠性模型	153
6.4.4	软件可靠性工程	155
6.5	面向对象的测试	155
6.5.1	OO 软件测试策略	155
6.5.2	OO 类测试方法	156
6.5.3	继承层次的测试	158
6.6	小结	160
习题 6		161
第 7 章	软件维护	163
7.1	软件维护的概念	163
7.1.1	改正性维护	164
7.1.2	适应性维护	164
7.1.3	完善性维护	164
7.1.4	预防性维护	164
7.2	软件的可维护性	164
7.2.1	可理解性	165
7.2.2	可测试性	165
7.2.3	可修改性	165

7.2.4	可靠性	166
7.2.5	可移植性	166
7.2.6	可使用性	166
7.2.7	效率	166
7.3	提高可维护性的方法	166
7.3.1	提供完整和一致的文档	167
7.3.2	建立明确的软件质量目标和优先级	167
7.3.3	使用现代化的开发技术和工具	168
7.3.4	进行明确的质量保证审查	168
7.3.5	选择可维护性好的程序设计语言	169
7.3.6	采用软件维护的新方法	169
7.4	软件再工程	170
7.4.1	业务过程再工程	170
7.4.2	软件再工程过程模型	171
7.4.3	逆向工程	172
7.4.4	软件重构	173
7.4.5	正向工程	174
7.4.6	再工程经济学	174
7.5	小结	175
	习题 7	175
第 8 章	软件复用	177
8.1	软件复用概念	177
8.1.1	软件复用的类型	177
8.1.2	软件复用的粒度	178
8.1.3	软件复用机会的识别	179
8.2	领域工程	181
8.2.1	领域分析	181
8.2.2	领域特征	182
8.3	可复用构件的建造及复用	183
8.3.1	构件应具有的特征	183
8.3.2	领域构件的设计框架	184
8.3.3	几种流行的构件技术	184
8.3.4	建立可复用的构件库	186
8.3.5	软件构件的复用	187
8.3.6	基于 Agent 的软件复用	188
8.3.7	基于复用的数据挖掘系统	189
8.4	面向对象的软件复用技术	190
8.4.1	复用技术对 OO 方法的支持	190
8.4.2	OO 软件复用技术	191

8.5 小结	192
习题 8	193
第 9 章 面向对象的软件工程	194
9.1 面向对象的基本概念	194
9.1.1 对象	195
9.1.2 类	196
9.1.3 消息、方法和变量	197
9.1.4 面向对象的基本特征	198
9.2 面向对象建模及统一建模语言 UML	203
9.2.1 面向对象建模	203
9.2.2 统一建模语言 UML	204
9.2.3 UML 模型	208
9.2.4 UML 的扩展机制	216
9.2.5 UML 建模工具	219
9.3 面向对象的软件需求分析及设计	220
9.3.1 常用的面向对象软件工程方法	220
9.3.2 基于 UML 的软件分析与设计	222
9.4 面向对象的软件需求分析方法及其在图书馆系统的应用	223
9.4.1 系统需求	224
9.4.2 用面向对象方法进行系统需求分析	224
9.5 基于 UML 的网络管理平台的分析与设计	228
9.5.1 系统开发过程	228
9.5.2 系统的实现	231
9.6 小结	232
习题 9	233
第 10 章 软件质量保证	235
10.1 软件质量的概念	235
10.2 软件质量保证	236
10.2.1 SQA 活动	236
10.2.2 软件可靠性与软件质量保证	237
10.2.3 软件质量检验与软件质量保证	238
10.3 软件复审	238
10.3.1 软件错误的危害性	238
10.3.2 软件复审与审核计划	239
10.3.3 正式技术复审	239
10.4 统计质量保证	240
10.5 ISO9000 质量标准	242
10.5.1 ISO9000 简介	242

10.5.2	ISO9000 认证标准	243
10.5.3	我国对 ISO9000 标准的使用	246
10.6	软件质量预测模型及应用	247
10.6.1	软件质量度量和质量预测模型的关系	247
10.6.2	广义回归神经网络结构及特点	248
10.6.3	软件质量预测模型的应用	249
10.7	小结	250
习题 10		251
第 11 章	软件的技术度量	252
11.1	软件技术度量概述	252
11.1.1	软件技术度量要解决的问题	252
11.1.2	软件技术度量考虑的软件质量因素	252
11.1.3	软件技术度量的原则	256
11.2	软件分析模型的度量	256
11.2.1	基于功能点的度量	256
11.2.2	基于原语的度量——Bang 度量	258
11.2.3	SRS 质量的度量	259
11.3	软件设计模型的度量	259
11.3.1	体系结构设计度量	259
11.3.2	构件级度量	262
11.3.3	界面设计度量	265
11.4	源代码度量	266
11.4.1	Halstead 度量方法	266
11.4.2	Thayer 复杂性度量	266
11.5	软件测试的度量	267
11.5.1	测试过程 S 曲线	267
11.5.2	缺陷追踪和管理	268
11.5.3	外包软件验收测试中的度量	268
11.6	软件维护的度量	269
11.6.1	UKSMA 和 NESMA 标准	269
11.6.2	软件维护的度量	269
11.7	面向对象的度量	270
11.7.1	面向对象度量方法	270
11.7.2	面向对象系统中考虑结构因素的类的内聚度量	274
11.8	小结	276
习题 11		276
第 12 章	软件开发工具与环境	279
12.1	软件开发工具	279

12.2	软件开发环境	280
12.2.1	软件开发环境的组成	280
12.2.2	软件开发环境的分类	281
12.3	计算机辅助软件工程	282
12.3.1	CASE 环境	283
12.3.2	软件开发 CASE 工具简介	283
12.4	小结	289
	习题 12	289
第 13 章	软件工程管理	290
13.1	软件工程文化	290
13.1.1	软件文化概念	290
13.1.2	软件项目的五要素	291
13.1.3	肯定开发者的成就	292
13.1.4	不断学习新知识	292
13.1.5	寻找卓越的需求	293
13.1.6	质量文化	293
13.1.7	方法、测量和工具	294
13.2	现代人件	295
13.2.1	团队开发	295
13.2.2	程序员个人对软件系统的影响	296
13.2.3	工作组织	297
13.3	软件工程 36 计	302
13.3.1	方法篇	302
13.3.2	设计篇	305
13.3.3	实现篇	306
13.3.4	管理篇	308
13.4	软件风险分析与管理	309
13.4.1	软件风险及分类	309
13.4.2	风险预测	310
13.4.3	风险的缓解、监控和管理	312
13.5	软件配置管理	314
13.5.1	软件配置	314
13.5.2	配置标识	315
13.5.3	里程碑与变更控制	315
13.5.4	配置状态登录	316
13.5.5	配置审查	316
13.6	软件工程标准化	317
13.6.1	软件工程标准化的意义	317
13.6.2	软件工程标准化的研究内容	317

13.6.3	软件工程国家标准	318
13.7	软件的知识产权保护	319
13.7.1	软件知识产权的法律保护	319
13.7.2	计算机软件是著作权法保护的客体	320
13.7.3	软件著作权人享有的专有权利	321
13.7.4	软件著作权的登记	323
13.8	小结	323
习题 13		324
第 14 章	软件过程管理	326
14.1	软件能力成熟度模型	326
14.1.1	CMM 的必要性	326
14.1.2	CMM 的提出与发展	327
14.1.3	CMM 的作用	328
14.2	CMM 的主要内容	329
14.2.1	CMM 的专业术语	329
14.2.2	CMM 的内容	330
14.2.3	CMM 的内部结构	332
14.3	CMM 各级之间的关系	336
14.3.1	评估软件企业的 CMM 级别	337
14.3.2	从初始级向可重复级过渡	337
14.3.3	从可重复级向已定义级过渡	338
14.3.4	向可管理级和优化级过渡	338
14.4	CMM 实施的人员构成和组织机构的划分	339
14.4.1	人员的构成	339
14.4.2	组织机构的划分	340
14.5	个体软件过程	342
14.5.1	PSP 概述	342
14.5.2	PSP 的设计原则	343
14.5.3	PSP 进化的框架	343
14.5.4	PSP 的应用	345
14.5.5	PSP 的扩展——团队软件过程	347
14.6	小结	349
习题 14		349
第 15 章	基于构件的软件工程	351
15.1	基本概念	351
15.1.1	CBSE 过程	351
15.1.2	领域工程	352
15.1.3	基于构件的开发	352

15.1.4	CBSE 经济学	354
15.1.5	软构件的可测试性	355
15.2	可复用构件的分类与查询	357
15.2.1	主要的分类方法	357
15.2.2	基于刻面的检索	358
15.3	基于构件的软件重用成熟度	359
15.3.1	重用成熟度	359
15.3.2	构件重用与软件成熟度的关系	360
15.4	构件技术应用	361
15.4.1	CORBA 构件模型 CCM	361
15.4.2	基于构件技术的移动通信监控系统	362
15.4.3	构件技术在电力信息化中的应用	363
15.4.4	基于演化构件的软件开发	364
15.4.5	基于复用技术的仿真系统软件	364
15.5	小结	367
习题 15		368
第 16 章	应用 Web 工程	369
16.1	Web 工程	369
16.1.1	基于 Web 的系统及应用的特点	369
16.1.2	WebApp 工程的层次	371
16.1.3	Web 工程过程	371
16.1.4	Web 工程的最佳实践	373
16.2	WebApp 项目计划	374
16.2.1	表述 WebApp 项目	374
16.2.2	制定 WebApp 项目计划	376
16.2.3	组建 Web 工程团队	377
16.2.4	Web 工程的项目管理	378
16.3	WebApp 分析	382
16.3.1	WebApp 的需求分析	382
16.3.2	WebApp 的分析模型	385
16.3.3	内容模型	385
16.3.4	交互模型	387
16.3.5	功能模型	388
16.3.6	配置模型	389
16.3.7	关系导航分析	390
16.4	WebApp 设计	392
16.4.1	Web 工程设计	392
16.4.2	WebApp 界面设计	395
16.4.3	美学设计	399

16.4.4	内容设计·····	400
16.4.5	体系结构设计·····	400
16.4.6	导航设计·····	403
16.4.7	构件级设计·····	404
16.4.8	超媒体设计模式·····	404
16.4.9	面向对象的超媒体设计方法·····	405
16.5	WebApp 测试 ·····	406
16.5.1	WebApp 测试概念 ·····	406
16.5.2	WebApp 测试过程 ·····	408
16.5.3	WebApp 测试策略 ·····	409
16.6	小结·····	411
习题 16	·····	413
参考文献	·····	415

第 1 章 概 述

计算机是现代人都熟悉的一个词汇,也有人把它叫做电脑,还有人说“网络就是计算机”。从宏观上讲,一个计算机系统应该由 3 部分组成:硬件(Hardware)、软件(Software)和人件(Peopleware),它们互相联系,互相依存,硬件和软件是“经济基础”,人件是“上层建筑”。软件的角色很特殊,有时起到“经济基础”的作用,有时又起到“上层建筑”的作用,这 3 部分都非常重要,缺一不可。

1.1 计算机系统的开发过程

计算机系统的开发过程是指从提出项目开始,经过分析、论证、设计、实施到交付使用的全过程,其开发流程如图 1-1 所示。

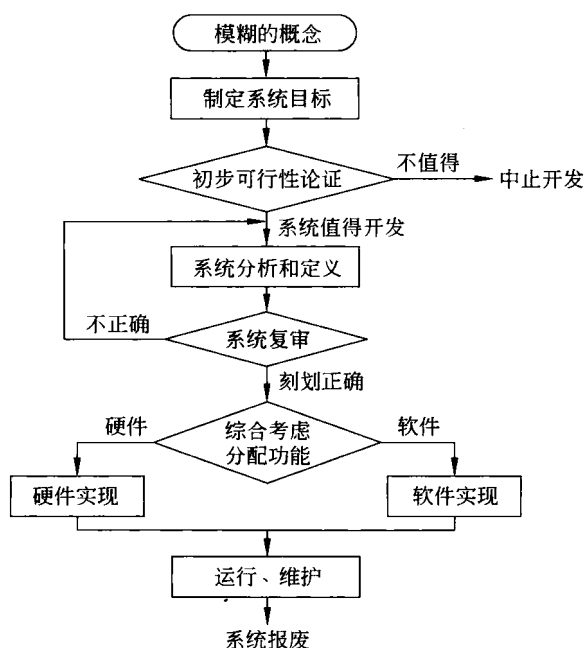


图 1-1 计算机系统开发流程

从图 1-1 可以看出,最初对未来的计算机系统我们只能有一个模糊的概念,根据这个概念来制定系统的初始目标,主要应该明确建立计算机系统的目的是什么,适应于哪些用户对象,期望的性能价格比是多少,开发周期多长,总的开发费用是多少等。这些问题要尽量定量描述,对于无法定量描述的东西,要仔细研究和推敲,用严密精确的概念和术语去描述。

初步可行性论证主要从 3 个方面讨论该计算机系统是否值得开发:一是开发该计算机系统的先决条件是否具备,二是成功的可能性有多大,三是考虑是否有社会效益和经济效益。