



普通高等教育“十二五”规划教材

软件工程基础

主编 蒲天银

普通高等教育“十二五”规划教材

软件工程基础

主 编 蒲天银



 江苏大学出版社
JIANGSU UNIVERSITY PRESS

镇 江

内 容 提 要

本书根据作者多年从事软件开发及软件工程教学的经验编写而成。全书共 10 章,分别介绍了软件工程的基本知识,软件开发的可行性分析,软件开发过程中的需求分析、总体设计、详细设计、软件编码与测试、软件维护,以及面向对象的程序设计思想、软件项目管理、软件工程与软件产业发展趋势等。

本书内容实用、语言简炼、案例丰富,充分体现了理论与实践相结合的特点。本书既可作高等院校计算机专业的教材,也可供各类软件开发与维护人员阅读。

图书在版编目(CIP)数据

软件工程基础 / 蒲天银主编. — 镇江: 江苏大学出版社, 2013. 8

ISBN 978-7-81130-497-8

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

中国版本图书馆 CIP 数据核字(2013)第 203494 号

软件工程基础

Ruanjian Gongcheng Jichu

主 编 / 蒲天银

责任编辑 / 段学庆

出版发行 / 江苏大学出版社

地 址 / 江苏省镇江市梦溪园巷 30 号(邮编: 212003)

电 话 / 0511-84446464(传真)

网 址 / <http://press.ujs.edu.cn>

排 版 / 北京金企鹅文化发展中心

印 刷 / 北京忠信印刷有限责任公司

经 销 / 江苏省新华书店

开 本 / 787 mm×1 092 mm 1/16

印 张 / 13

字 数 / 300 千字

版 次 / 2013 年 8 月第 1 版 2013 年 8 月第 1 次印刷

书 号 / ISBN 978-7-81130-497-8

定 价 / 35.00 元

如有印装质量问题请与本社营销部联系(电话: 0511-84440882)



软件是信息化的核心之一，软件产业展现国家科技发展的核心竞争力，体现国家的综合实力。随着计算机应用的不断普及、互联网应用的不断深入和网络技术的不断发展，软件系统的规模和复杂度也不断增加，如何确保开发出符合用户预期的、质量有保证的软件系统仍然是一个巨大挑战，软件危机的存在仍然阻碍着软件的发展。

作为计算机科学技术的一个重要分支——软件工程学，成为软件需求、开发、维护、管理的普遍原理和技术相结合的、活跃的研究领域。随着软件工程的迅猛发展，新的技术、方法、工具不断涌现，为读者学习和研究这门学科打下了良好的基础，创造了难得的机遇。

作为软件工程学的入门介绍，本书立足于基本的原理、概念、方法和工具，将作者在教学活动中的很多案例从实用的角度引入到软件系统的需求、设计、实现、测试、维护和管理的内容中，同时兼顾对软件工程过程介绍的全面性和系统性。

本书根据编者多年从事软件工程课程教学和软件开发的实践经验，在介绍相关理论和过程的基础上，从3个层面介绍了软件项目开发：首先从传统的生命周期体系思想介绍开发软件的过程，然后介绍面向对象方法学开发软件过程和软件项目管理的体系框架，最后分析软件项目开发、软件教育工程的发展方向及趋势。本书的特点主要体现在：

(1) 案例贯穿于整个内容，在讲解过程中引入图书管理系统的设计过程分析案例，并与理论有机结合起来阐述。

(2) 介绍软件工程理论的基本概念和过程，它们对软件过程实践起着基石和指导的作用。

(3) 每章的小结对各章的主要内容进行总结，便于读者理解和掌握主要内容。

(4) 根据教学内容的总体安排，在部分章节后附上实验实训，让读者深切感受到书中介绍的理论是如何指导实践的。

本书分10个章节进行阐述，具体安排如下。

第1章回顾了软件危机的产生，介绍软件工程的产生和发展，包括软件工程的基本概念、目标和实施原则。通过对软件、软件生命周期和软件过程模型的介绍，让读者对软件工程的基本原理、方法、过程有一个基本认识。

第2章介绍软件项目可行性研究的任务、研究内容，以及可行性研究相关的一些关键技术。

第3章介绍软件需求分析的基本概念、任务和原则，并详细说明结构化分析和建模过



程，包括面向数据的数据建模、面向数据流的功能建模和面向状态的行为建模。

第 4、5 两章介绍软件设计的基本概念、任务和原则，以及目前主流的软件体系结构设计模型，它们分别是以数据为中心的数据仓库模型、客户端/服务器模式的分布式结构模型和层次模型。从应用角度出发，详细描述了结构化设计的设计方法。

第 6 章从软件工程范畴讨论程序实现和编码，包括程序设计语言的分类、特性、准则及程序编写规范等，介绍了进行软件测试的对象和测试技术。测试技术主要介绍了白盒测试和黑盒测试。

第 7 章介绍了软件项目中的一些维护技术，软件维护的内容和过程，以及如何提高软件的可维护性和软件再工程。

第 8 章介绍面向对象软件工程中的一些方法技术。

第 9 章介绍软件项目管理的基本内容，主要包括人员的组织与管理、软件度量、软件项目计划的内容与制订方法、软件项目估算、软件开发成本估算等。

第 10 章介绍了软件产业的发展态势，分析了软件行业未来发展的方向。

本书在编写过程中参考了与软件工程相关的大量文献，查阅了国内知名的软件工程网站、软件工程精品课程网站，得到了铜仁学院领导、同事及朋友们的大力支持和帮助，在此一并表示衷心的感谢！

由于作者水平有限，疏漏、欠妥之处在所难免，恳请读者指正。

编 者

2013 年 8 月



目 录

第 1 章 软件工程概述	1
1.1 软件知识回顾	1
1.2 软件危机	3
1.3 软件工程	5
1.3.1 软件工程定义	5
1.3.2 软件工程目标	5
1.3.3 软件工程过程	6
1.3.4 软件工程原则	6
1.3.5 软件工程与其他相关学科的关系	7
1.3.6 软件工程方法学	7
1.4 软件生命周期	8
1.5 软件开发过程模型	9
本章小结	16
练习题	17
第 2 章 可行性研究	18
2.1 可行性研究的目的及任务	18
2.2 可行性研究过程	19
2.3 系统流程图	21
2.3.1 符号	21
2.3.2 系统流程图实例分析	23
2.3.3 分层	23
2.4 成本/效益分析	24
2.4.1 成本估计	25
2.4.2 效益分析的方法	26
本章小结	28
练习题	28
实验实训一	29



第3章 需求分析	30
3.1 需求分析的任务	31
3.2 需求分析的原则	33
3.3 需求分析的方法	34
3.4 沟通获取需求	35
3.5 数据流图	36
3.5.1 数据流图符号	37
3.5.2 命名	38
3.5.3 数据流图实例	38
3.5.4 用途	39
3.6 建立数据模型：实体-联系图	39
3.6.1 数据对象	40
3.6.2 属性	40
3.6.3 联系	40
3.6.4 实体-联系图的符号	41
3.7 建立行为模型：状态转换图	42
3.8 数据字典	44
3.9 统一建模语言——UML 简介	45
3.10 数据规范化	47
3.11 软件需求说明书编写格式	47
本章小结	49
练习题	50
实验实训二	50
第4章 总体设计	52
4.1 总体设计步骤及主要过程	53
4.1.1 总体设计的主要任务	53
4.1.2 总体设计的过程	53
4.2 总体设计中的一些概念	55
4.2.1 模块化	55
4.2.2 模块独立性	56
4.2.3 抽象	60
4.2.4 逐步求精	61
4.2.5 信息隐藏	62
4.3 总体设计的原则	62

4.4 总体设计描述工具	64
4.4.1 系统结构图	64
4.4.2 层次图和 HIPO 图	65
4.5 面向数据流的设计方法	66
本章小结	67
练习题	68
实验实训三	68
第 5 章 详细设计	70
5.1 详细设计的目标、任务与原则	70
5.2 结构程序设计	71
5.3 详细设计工具	72
5.3.1 程序流程图	73
5.3.2 盒图	73
5.3.3 PAD 图	74
5.3.4 过程设计语言	76
5.3.5 判定表	76
5.3.6 判定树	77
5.4 程序复杂程度的定量度量	78
5.4.1 McCabe 方法	78
5.4.2 Halstead 方法	81
5.5 详细设计规格说明与复审	81
5.5.1 详细设计说明书	81
5.5.2 设计复审	81
5.6 人机界面设计	82
5.6.1 设计问题	82
5.6.2 设计过程	83
5.6.3 人机界面设计指南	83
本章小结	84
练习题	85
实验实训四	85
第 6 章 软件编码与测试	87
6.1 编码	88
6.1.1 程序设计语言	88
6.1.2 编码风格	89



6.1.3 常用程序设计工具简介	92
6.2 软件测试	94
6.2.1 软件测试的术语和原则	94
6.2.2 软件测试中的误区	96
6.2.3 软件测试的方法	97
6.2.4 软件测试过程及策略	99
6.3 软件测试技术及用例	105
6.3.1 白盒测试法用例的设计	105
6.3.2 黑盒测试法分析	110
6.4 软件测试的经验教训	113
6.5 软件调试	115
6.5.1 调试原则	116
6.5.2 调试步骤	117
6.5.3 调试方法	118
6.6 软件可靠性	119
6.6.1 软件可靠性概念	120
6.6.2 软件测试中可靠性分析	121
6.7 主流软件测试工具 Win Runner 介绍	123
本章小结	125
练习题	126
实验实训五	126
实验实训六	127
第7章 维护	128
7.1 软件维护的定义与分类	128
7.2 软件维护的特点	129
7.3 软件维护内容	130
7.4 软件维护步骤	131
7.5 软件的可维护性	131
7.6 预防性维护	132
7.7 软件维护的副作用	133
7.8 软件再工程过程	134
本章小结	135
练习题	136



第 8 章 面向对象设计	137
8.1 面向对象思想的基本概念	137
8.2 面向对象的三大要素	141
8.3 面向对象与系统设计	143
8.3.1 用面向对象的方法分析项目需求	143
8.3.2 用面向对象的方法设计系统	149
本章小结	151
练习题	151
第 9 章 软件项目管理	153
9.1 软件项目管理概述	153
9.1.1 软件项目管理的提出及特点	153
9.1.2 软件项目管理的内容	154
9.1.3 软件项目管理的原则	154
9.2 人员的组织与管理	155
9.2.1 组织模式	155
9.2.2 组织管理	156
9.3 软件度量	158
9.3.1 软件度量流程	158
9.3.2 软件度量三维度	159
9.3.3 软件度量工具	160
9.3.4 软件度量的目标	160
9.3.5 软件度量的方法体系	161
9.4 软件项目计划	166
9.4.1 软件项目计划的内容	166
9.4.2 制订软件项目计划的方法与策略	167
9.4.3 软件项目计划的编制方针	170
9.4.4 软件项目计划模板	171
9.5 软件项目计划及管理涉及的相关问题	173
9.5.1 软件项目估算	173
9.5.2 软件项目进度计划	173
9.5.3 软件开发成本估算	174
9.5.4 软件项目风险管理	177
9.5.5 软件质量保证与配置管理	182
本章小结	182



练习题	183
第 10 章 软件产业前景	184
10.1 全球软件工程发展趋势	184
10.2 国内软件产业发展趋势	187
10.2.1 软件产业发展特征	187
10.2.2 软件产业发展中的关键技术	189
10.3 软件工程教育	191
本章小结	194
练习题	194
参考文献	195

第 1 章 软件工程概述

【学习目标】

- ◆ 掌握软件、软件工程的观念
- ◆ 了解软件危机产生的原因、表现形式、应对办法
- ◆ 熟悉软件生命周期
- ◆ 熟悉几种软件开发模型

计算机软件是计算机得以广泛应用的灵魂，到 20 世纪 70 年代初，随着软件产业的发展，计算机应用涉及社会的方方面面，计算机软件业的推进成为计算机应用推广的重要部分，但在发展过程中也出现了很多问题，如典型的软件危机问题。为了更有效地开发与维护软件，软件工作者认真研究消除软件危机的途径，并因此产生了一门新兴的工程学科——软件工程学，即软件工程。因为有了软件后才有了软件工程，所以为了更有效地学习软件工程技术，要从软件本身的特性开始。

1.1 软件知识回顾

1. 软件的概念

软件这个概念很多人都不会陌生，简单地说，软件是程序及其他相关文档的集合。人们在计算机上操作的对象就是软件。程序是实现相应功能的一些指令代码序列；文档是与软件开发、维护和使用有关的图文材料。

这里要注意一点：程序和软件不是一个概念，程序只是软件的组成部分。

2. 软件的特点

软件与硬件等相比较有其独特的特征，下面列举分析。

(1) 软件是一种逻辑实体。当启动计算机后，在各种操作过程中只能感知软件的存在，无法看到其具体的形态，只能通过观察、分析、思考、判断等方式去了解它的特性、功能。



(2) 软件是通过人的智力的高度发挥开发而成的。和传统意义上的硬件制造不同，它虽然也有研制过程，但整个过程完全是一种通过人们的智力活动，把知识与技术转化为一种信息产品的过程。

(3) 软件的开发对计算机系统有着不同程度的依赖性，有时甚至还需考虑社会因素。软件开发是在硬件及软件相关基础之上进行的，必须在一定的计算机环境中进行。如高校财务管理信息系统的开发和运行就涉及当前高校财务管理机制、体制模式等方面的问题。

(4) 软件的开发是一个复杂的过程。软件是人脑力劳动的智能产品，由于人类思维上的复杂性，因此软件开发过程也就来自于开发程序逻辑结构的复杂度。

(5) 软件在维护上也有本质差异。软件在运行过程中不存在硬件的机器磨损、老化问题，但同样存在退化问题，随硬件的不断发展也需要维护。

(6) 软件的价值无法估量。由于软件开发需要投入大量的、高强度的脑力劳动，在成本、风险等方面存在一定的问题，因此，一旦软件开发成功后，其价值有时无法准确估量。

3. 软件的类型

软件的分类方法有很多，常用的有以下几种。

(1) 按软件的功能进行划分：分为系统软件和应用软件两大类。

这是一种常规的分类方式，系统软件有操作系统、数据库管理系统等，如 Windows 7.0、Visual FoxPro 8.0 等。

应用软件有商业数据处理软件、工程与科学计算软件、计算机辅助设计/制造软件、系统仿真软件等，如 Office 2010 就是典型的应用软件。

(2) 按软件规模进行划分：分为微型、小型、中型、大型，分别从参加人员数、研制期限、源程序行数等考虑，见表 1-1。

表 1-1 按软件规模划分

类别	参加人员数	研制期限
微型	1	1~4 周
小型	2-5	1~12 月
中型	5-20	1~2 年
大型	>20	>2 年

(3) 按软件工作方式划分：分为实时处理软件、分时处理软件、交互软件、批处理软件。

(4) 按软件服务对象的范围划分：分为项目软件（定制软件）、产品软件（通用软件）。



4. 软件发展史

第一阶段（20世纪60年代中期以前）：程序设计阶段。第一代软件主要是用机器语言编写的，在这个时代的末期出现了汇编语言，使用汇编器编写程序的程序员是最初的系统程序员。在20世纪60年代末期出现了“软件危机”。

第二代（20世纪60年代中期~70年代中期）：程序系统阶段——“软件工程”学科诞生。

第三代（20世纪70年代中期~80年代中期）：软件工程阶段。

第四代（20世纪80年代中期至今）：软件过程化、软件产业在世界经济中已经占有举足轻重的地位。

1.2 软件危机

1. 软件危机概述

20世纪60年代以前，计算机刚刚投入实际使用，软件设计往往只是为了一个特定的应用而在指定的计算机上设计和编制，采用密切依赖于计算机的机器代码或汇编语言，软件的规模比较小，文档资料通常也不存在，很少使用系统化的开发方法。设计软件往往等同于编制程序，基本上个人设计、个人使用、个人操作。

60年代中期，大存储容量、高速度计算机出现，使计算机的应用范围迅速扩大，软件开发急剧增长。这时期高级语言开始出现，操作系统的发展引起了计算机应用方式的变化，大量数据处理导致第一代数据库管理系统的诞生。软件系统的规模越来越大，复杂程度越来越高，软件可靠性问题也越来越突出。原来的个人设计、个人使用的方式不再能满足要求，迫切需要改变软件生产方式，提高软件生产率，软件危机开始爆发。

1968年，北大西洋公约组织的计算机科学家在联邦德国召开国际会议，第一次讨论软件危机问题，并正式提出“软件工程”一词。从此，一门新兴的工程学科——软件工程学，为研究和克服软件危机应运而生。

因此，一般认为软件危机是指在计算机软件开发和维护过程中碰到问题的集合。所碰到的问题主要表现在两个方面：一是面对硬件技术不断发展，如何开发软件；二是以前开发的一些软件如何维护。

2. 软件危机典型表现

软件危机主要表现在以下几个方面。

(1) 软件开发费用和进度失控。由于在开发前期对软件规模、费用成本、技术支持



等需求方面分析不到位,因此超支、进度拖延的情况屡屡发生。有时为了赶进度或压缩成本不得不采取一些权宜之计,这样又往往严重损害了软件产品的质量。

(2) 软件的可靠性差。尽管耗费了大量的人力物力,但系统的正确性却越来越难以保证,出错率大大增加,由于软件错误而造成的损失十分惊人。

(3) 生产出来的软件难以维护。之前开发的很多软件由于缺乏相应的文档资料,程序中的错误难以定位、难以改正,有时改正了已有的错误又引入新的错误,因此基本难以维护。

20 世纪 80 年代以来,尽管软件工程研究与实践取得了可喜的成就,软件技术水平有了长足的进展,但众所周知,在 80 年代后期计算机硬件技术发展速度惊人。在这一段时期,软件危机不仅没有消失,还有加剧之势,主要表现在以下两方面:一方面软件成本在计算机系统总成本中所占的比例居高不下,且逐年上升。由于微电子学技术的进步和硬件生产自动化程度不断提高,硬件成本逐年下降,性能和产量迅速提高。然而软件开发需要大量人力,软件成本随着软件规模和数量的剧增而持续上升。从部分国家的统计数字表明,1985 年软件成本大约占总成本的 90%。另一方面是软件开发生产率提高的速度远远跟不上计算机应用迅速普及深入的需要,软件产品供不应求的状况使得人们不能充分利用现代计算机硬件所能提供的巨大潜力。

3. 软件危机产生的原因

软件工作者经过长时间研究,结果表明,软件危机的原因主要有两方面:一方面是软件本身的特点,另一方面是由于软件开发和维护的方法不规范、不正确,其具体原因如下。

(1) 忽略软件需求分析。这是一种通病,许多软件开发人员不能正确地理解用户的需求,甚至根本不去深入了解用户的需求。

(2) 开发过程没有统一、规范的方法做指导,文档资料不齐全。一个软件从开始计划到使用和维护要经历一个漫长的过程,这个时期一般称为软件的生命周期,这个生命周期从大的方面分为软件定义、软件开发、软件运行三个时期,每一个时期又可分为若干小的阶段,而编程只是软件开发过程的一个小阶段,一般占软件总成本的 10%~30%,甚至更少。缺少规范而盲目上阵编写程序是不可取的。

(3) 忽视测试工作。在很多人的眼中,代码编写工作是软件开发中最具实力的工作,而软件测试一般都不愿意干,这在软件开发中存在很大的误区。一般来讲,软件测试占软件开发总成本的 30%~50%。

(4) 轻视软件维护。在一个软件漫长的维护期中,必须及时改正软件在使用中出现的错误,给用户一个满意的回答。软件维护工作是极其复杂艰巨的,其工作量往往随系统使用环境、用户需求的变化而变化。



1.3 软件工程

软件工程 (Software Engineering, SE) 是一门研究用工程化方法构建和维护有效、实用和高质量软件的学科。这个概念最早在 1968 年秋季北约 (NATO) 的科技委员会上提出, 它涉及程序设计语言、数据库、软件开发工具、系统平台、设计、模式等多个方面。

1.3.1 软件工程定义

软件工程一直以来都缺乏一个统一的定义, 在现实应用中, 很多学者、组织机构根据需要给出了自己的定义。

(1) Barry Boehm: 运用现代科学技术知识来设计并构造计算机程序及为开发、运行和维护这些程序所必需的相关文件资料。

(2) IEEE 在软件工程术语汇编中的定义: 软件工程是将系统化的、严格约束的、可量化的方法应用于软件的开发、运行和维护, 即将工程化应用于软件。

(3) Fritz Bauer 在 NATO 会议上给出的定义: 建立并使用完善的工程化原则, 以较经济的手段获得能在实际机器上有效运行的可靠软件的一系列方法。

(4) 《计算机科学技术百科全书》中的定义: 软件工程是应用计算机科学、数学及管理科学等原理开发软件的工程。软件工程借鉴传统工程的原则、方法, 以提高质量、降低成本。其中, 计算机科学、数学用于构建模型与算法, 工程科学用于制定规范、设计范型 (Paradigm)、评估成本及确定权衡, 管理科学用于计划、资源、质量、成本等管理。

目前比较认可的一种定义: 软件工程是研究和应用如何以系统性的、规范化的、可定量的过程化方法去开发和维护软件, 以及如何把经过时间考验而证明正确的管理技术和当前能够得到的最好的技术方法结合起来。

1.3.2 软件工程项目目标

通过前面介绍软件、软件危机, 我们知道软件工程是为了克服软件危机而提出的。概括地讲, 软件项目的目标是解决如下问题: 在给定成本、进度的前提下, 开发出具有可修改性、有效性、可靠性、可理解性、可维护性、可重用性、可适应性、可移植性、可追踪性和可互操作性并且满足用户需求的软件产品。

软件工程追求这些目标有助于提高软件产品的质量和开发效率, 减少维护的困难, 便于缓解软件危机问题。



1.3.3 软件工程过程

软件工程过程是指生产一个最终能满足需求且达到工程目标的软件产品所需要的步骤。软件工程过程主要包括开发过程、运作过程、维护过程。它们覆盖了需求、设计、实现、确认及维护等活动。

需求活动包括问题分析和需求分析。问题分析获取需求定义，又称软件需求规约。需求分析生成功能规约。设计活动一般包括概要设计和详细设计。

概要设计建立整个软件系统结构，包括子系统、模块，以及相关层次的说明、每一模块的接口定义。详细设计产生程序员可用的模块说明，包括每一模块中的数据结构说明及加工描述。

实现活动把设计结果转换为可执行的程序代码。

确认活动贯穿于整个开发过程，实现完成后的确认，保证最终产品满足用户的要求。

维护活动包括使用过程中的扩充、修改与完善。

伴随以上过程，软件工程过程还有管理过程、支持过程、培训过程等。

1.3.4 软件工程原则

软件工程的原则是指围绕工程设计、工程支持及工程管理在软件开发过程中必须遵循的原则，具体包括以下四项基本原则。

(1) 选取适宜开发模型

该原则与系统设计有关。在系统设计中，软件需求、硬件需求及其他因素之间是相互制约、相互影响的，经常需要权衡。因此，必须认识需求定义的易变性，采用适宜的开发范型予以控制，以保证软件产品满足用户的要求。

(2) 采用合适的设计方法

在软件设计中，通常要考虑软件的模块化、抽象与信息隐蔽、局部化、一致性及适应性等特征。合适的设计方法有助于这些特征的实现，以达到软件工程的目标。

(3) 提供高质量的工程支持

在软件工程中，软件工具与环境对软件过程的支持颇为重要。软件工程项目的质量与开销直接取决于对软件工程所提供的支撑质量和效用。

(4) 重视开发过程的管理

软件工程的管理直接影响可用资源的有效利用，生产满足目标的软件产品，提高软件组织的生产能力等问题。因此，仅当软件过程得以有效管理时，才能实现有效的软件工程。

这一软件工程框架告诉我们，软件工程的目标是可用性、正确性和合算性；实施一