



高职高专院校“十二五”精品示范系列教材

软件技术专业群

软件工程概论

主 编 倪天林 王伟娜
副主编 祁慧敏 罗东芳 郭 峰

教材特色：

- 平台课 + 模块课 搭建专业群课程
- 实例驱动 + 项目同步 优化内容 新颖实用
- 教材 + 案例 + 实战素材 立体化资源相结合



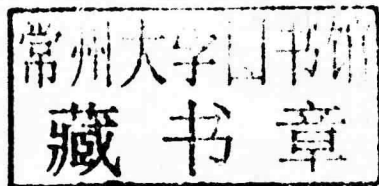
中国水利水电出版社
www.waterpub.com.cn

高职高专院校“十二五”精品示范系列教材（软件技术专业群）

软件工程概论

主 编 倪天林 王伟娜

副主编 祁慧敏 罗东芳 郭 峰



中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

软件工程是软件工程（或软件技术）专业的一门核心课程，也是计算机科学与技术、信息管理等相关专业的骨干课程，同时也是软件设计人员、程序开发人员、软件测试人员和软件项目管理人员等应具备的专门知识。

本书全面系统地讲授了软件工程的理论、方法和技术，书中运用大量的软件开发实例，采用图文并茂的形式来组织、理解知识内容。每章给出多种形式的习题练习，以巩固所学知识，书中安排有大量实训，以指导学生进行软件开发练习。全书共 13 章，内容包括软件工程概述、可行性研究、需求分析、概要设计、详细设计、程序编码、面向对象的分析与设计、统一建模语言 UML、统一软件开发过程 RUP、软件测试、软件维护、软件项目管理及软件复用技术。

本书可作为高等院校软件工程课程的教材或教学参考书，也可供有一定实际经验的软件工作人员和需要开发应用软件的广大计算机用户阅读参考。

图书在版编目（CIP）数据

软件工程概论 / 倪天林, 王伟娜主编. — 北京 :
中国水利水电出版社, 2014. 6

高职高专院校“十二五”精品示范系列教材. 软件技术专业群

ISBN 978-7-5170-2105-6

I. ①软… II. ①倪… ②王… III. ①软件工程—高等职业教育—教材 IV. ①TP311.5

中国版本图书馆CIP数据核字(2014)第118003号

策划编辑: 祝智敏 责任编辑: 陈 洁 加工编辑: 袁 慧 封面设计: 李 佳

书 名	高职高专院校“十二五”精品示范系列教材（软件技术专业群） 软件工程概论
作 者	主 编 倪天林 王伟娜 副主编 祁慧敏 罗东芳 郭 峰
出版发行	中国水利水电出版社 （北京市海淀区玉渊潭南路1号D座 100038） 网址: www.waterpub.com.cn E-mail: mchannel@263.net（万水） sales@waterpub.com.cn
经 售	电话: (010) 68367658（发行部）、82562819（万水） 北京科水图书销售中心（零售） 电话: (010) 88383994、63202643、68545874 全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京蓝空印刷厂
规 格	184mm×240mm 16开本 23印张 507千字
版 次	2014年6月第1版 2014年7月第1次印刷
印 数	0001—3000册
定 价	45.00元

凡购买我社图书，如有缺页、倒页、脱页的，本社发行部负责调换

版权所有·侵权必究

编审委员会

顾 问 段银田 甘 勇

主任委员 郝小会 郭长庚

副主任委员 连卫民 黄贻彬 苏 玉 丁爱萍 雷顺加

委 员 (按姓氏笔画排名)

丰树谦 王 硕 王 辉 王东升 王惠斌

王德勇 牛军涛 尹新富 冯明卿 吕 争

齐英兰 许 磊 许绘香 孙 凌 杜永强

李 丹 李思广 杨梦龙 何 樱 谷海红

宋风忠 宋全有 张 洁 张 静 张 慧

张 巍 张志明 张俊才 张新成 张滨燕

陈利军 陈迎松 陈桂生 武凤翔 武俊琢

周观民 赵艳莉 侯 枫 袁芳文 倪天林

徐立新 徐钢涛 黄振中 曹 敏 商信华

董淑娟 韩应江 喻 林 谭建伟 谭营军

翟 慧 黎 娅

编委秘书 李井竹 武书彦 向 辉

序

为贯彻落实全国教育工作会议精神和《国家中长期教育改革和发展规划纲要（2010—2020年）》以及《关于“十二五”职业教育教材建设的若干意见》（教职成〔2012〕9号）文件精神，充分发挥教材建设在提高人才培养质量中的基础性作用，促进现代职业教育体系建设，全面提高职业教育教学质量，中国水利水电出版社在集合大批专家团队、一线教师和技术人员的基础上，组织出版“高职高专院校‘十二五’精品示范系列教材（软件技术专业群）”职业教育系列教材。

在高职示范校建设初期，教育部就曾提出：“形成500个以重点建设专业为龙头、相关专业为支撑的重点建设专业群，提高示范院校对经济社会发展的服务能力。”专业群建设一度成为示范性院校建设的重点，是学校整体水平和基本特色的集中体现，是学校发展的长期战略任务。专业群建设要以提高人才培养质量为目标，以一个或若干个重点建设专业为龙头，以人才培养模式构建、实训基地建设、教师团队建设、教学资源库建设为重点，积极探索工学结合教学模式。本系列教材正是配合专业群建设的开展推出，围绕软件技术这一核心专业，辐射学科基础相同的软件测试、移动互联应用和软件服务外包等专业，有利于学校创建共享型教学资源库、培养“双师型”教师团队、建设开放共享的实验实训环境。

此次精品示范系列教材的编写工作力求：集中整合专业群框架，优化体系结构；完善编者结构和组织方式，提升教材质量；项目任务驱动，内容结构创新；丰富配套资源，先进性、立体化和信息化并重。本系列教材的建设，有如下几个突出特点：

（1）集中整合专业群框架，优化体系结构。联合河南省高校计算机教育研究会高职教育专委会及二十余所高职院校专业教师共同研讨、制定专业群的体系框架。围绕软件技术专业，囊括具有相同的工程对象和相近的技术领域的软件测试、移动互联应用和软件服务外包等专业，采用“平台+模块”式的模式，构建专业群建设的课程体系。将各专业共性的专业基础课作为“平台”，各专业的核心专业技术课作为独立的“模块”。统一规划的优势在于，既能规避专业内多门课程中存在重复或遗漏知识点的问题；又能在同类专业间优化资源配置。

（2）专家名师带头，教产结合典范。课程教材研究专家和编者主要来自于软件技术教学领域的专家、教学名师、专业带头人，以最新的教学改革成果为基础，与企业技术人员合作共同设计课程，采用跨区域、跨学校联合的形式编写教材。编者队伍对教育部倡导的职业教育教学改革精神理解的透彻准确，并且具有多年的教育教学经验及教产结合经验，准确地对相关专业的知识点和技能点进行横向与纵向设计、把握创新型教材的定位。

（3）项目任务驱动，内容结构创新。软件技术专业群的课程设置以国家职业标准为基础，以软件技术行业工作岗位群中的典型事例提炼学习任务，体现重点突出、实用为主、够用为度

的原则，采用项目驱动的教学方式。项目实例典型、应用范围较广，体现技能训练的针对性，突出实用性，体现“学中做”、“做中学”，加强理论与实践的有机融合；文字叙述浅显易懂，增强了教学过程的互动性与趣味性，相应的提升教学效果。

(4) 资源优化配套，立体化信息化并重。每本教材编写出版的同时，都配套制作电子教案；大部分教材还相继推出补充性的教辅资料，包括专业设计、案例素材、项目仿真平台、模拟软件、拓展任务与习题集参考答案。这些动态、共享的教学资源都可以从中国水利水电出版社的网站上免费下载，为教师备课、教学以及学生自学提供更多更好的支持。

教材建设是提高职业教育人才培养质量的关键环节，本系列教材是近年来各位作者及所在学校、教学改革和科研成果的结晶，相信它的推出将对推动我国高职电子信息类软件技术专业群的课程改革和人才培养发挥积极的作用。我们感谢各位编者为教材的出版所作出的贡献，也感谢中国水利水电出版社为策划、编审所作出的努力！最后，由于该系列教材覆盖面广，在组织编写的过程中难免有不妥之处，恳请广大读者多提宝贵建议，使其不断完善。

教材编审委员会

2013年12月

前 言

软件工程是研究软件开发技术和软件项目管理的一门工程学科，从工程化的角度来指导软件开发、测试和项目管理等活动。软件工程研究的范围非常广泛，包括技术方法、工具和管理等诸多方面，软件工程又是一门迅速发展的新兴学科，新的技术方法和工具不断涌现。本书是软件工程的入门教材，着重从实用角度讲述软件工程的基本原理、概念和方法，同时也力求做到知识的全面性和系统性。本书既适用于软件工程教学，又能对实际的软件开发工作提供指导与帮助。

软件工程是软件工程（或软件技术）专业的一门核心课程，也是计算机科学与技术、信息管理等相关专业的主干课程，同时也是软件设计人员、程序开发人员、软件测试人员和软件项目管理人员等应具备的专门知识。

高等职业教育是以就业为导向的职业能力教育，是培养数以万计高技能人才的主力军。高技能人才必须具有较强的技术应用能力，这就要求高职院校要加大对学生实际操作能力的培养。本书旨在借鉴国内外优秀教材的基础上，以多年的教学实践为基础，采用工学结合的方式，全面系统地组织教学内容，既注重知识的系统性和完整性，又突出技术能力的实践性。全书全面系统地讲授了软件工程的理论、方法和技术，书中运用大量的软件开发实例，采用图文并茂的形式来组织、理解知识内容。每章给出多种形式的习题练习，以巩固所学知识，书中安排有大量实训，以指导学生进行软件开发练习。

全书共 13 章，内容包括软件工程概述、可行性研究、需求分析、概要设计、详细设计、程序编码、面向对象的分析与设计、统一建模语言 UML、统一软件开发过程 RUP、软件测试、软件维护、软件项目管理及软件复用技术。

本书可作为高等院校软件工程课程的教材或教学参考书，也可供有一定实际经验的软件工作人员和需要开发应用程序的广大计算机用户阅读参考。

本书由倪天林、王伟娜任主编，负责设计编写大纲、修改定稿。各章的分工是：第 1 章、第 9 章由倪天林编写，第 2 章、第 3 章由王伟娜、张小红编写，第 4 章、第 5 章由罗东芳编写，第 6 章、第 7 章由郭峰、孙惠娟编写，第 8 章、第 13 章由祁慧敏编写，第 10 章由倪天林、董洁编写，第 11 章、12 章由王伟娜、张恩宾编写。

在编写过程中得到了中国水利水电出版社向辉同志、祝智敏同志的指导与支持，同时得到了责任编辑的认真审阅。此外，本教材还参考和借鉴了许多专家学者的研究成果，在此一并表示谢意。

由于编者水平所限，不足之处在所难免，敬请读者批评指正，以便在以后修订时加以改进和更正。

编者

2014 年 3 月

目 录

前言

第 1 章 软件工程概述	1	2.2.2 经济可行性	19
1.1 软件的概念和特点	1	2.2.3 操作可行性	19
1.1.1 计算机系统的构成及实现	1	2.2.4 法律可行性	19
1.1.2 软件的概念	1	2.3 可行性研究的步骤	19
1.1.3 软件的特点	2	2.3.1 确定项目规模和目标	19
1.1.4 计算机软件的分类	2	2.3.2 研究正在运行的系统	20
1.2 软件的发展和软件危机	3	2.3.3 建立目标系统的高层逻辑模型	20
1.2.1 计算机软件的发展过程	3	2.3.4 导出和评价各种方案	20
1.2.2 软件危机	3	2.3.5 推荐可行的方案	20
1.3 软件工程及其原理	5	2.3.6 编写可行性研究报告	20
1.3.1 软件工程的定义	5	2.4 系统流程图	21
1.3.2 软件工程的要素	5	2.4.1 系统流程图的作用	21
1.3.3 软件工程的目标	6	2.4.2 系统流程图的符号	21
1.3.4 软件工程的原理	6	2.5 成本—效益分析	22
1.3.5 软件工程的基本原理	7	2.5.1 成本估计	22
1.3.6 软件开发方法	8	2.5.2 两种成本估算技术	23
1.4 软件生存周期及其模型	9	2.5.3 度量效益的方法	24
1.4.1 软件定义时期	9	2.6 可行性研究报告	24
1.4.2 软件开发时期	10	2.7 项目开发计划	25
1.4.3 软件运行与维护时期	10	实训	25
1.4.4 软件生存周期模型	11	习题二	26
习题一	15	第 3 章 需求分析	28
第 2 章 可行性研究	17	3.1 需求分析任务	28
2.1 问题定义	17	3.1.1 需求分析的意义	28
2.1.1 问题定义的内容	17	3.1.2 需求分析的步骤	29
2.1.2 问题定义的步骤	17	3.1.3 需求分析的具体任务	31
2.1.3 问题定义报告	18	3.2 需求分析的基本原则	31
2.2 可行性研究的任务	18	3.3 需求分析过程	32
2.2.1 技术可行性	18	3.3.1 需求获取	32

3.3.2	需求建模	33	4.3.5	模块独立性	66
3.3.3	编写需求分析文档	34	4.4	体系结构设计工具	70
3.3.4	需求评审	34	4.4.1	结构图	70
3.4	需求获取方法	35	4.4.2	HIPO图	72
3.4.1	需求需要获取的内容	35	4.5	概要设计的启发式规则	74
3.4.2	需求获取的方法	36	4.5.1	提高模块独立性	74
3.4.3	需求调研的步骤	37	4.5.2	模块大小要适中	74
3.5	需求分析模型	39	4.5.3	模块应具有高扇入和适当的扇出	74
3.5.1	实体关系图	39	4.5.4	软件结构中的深度和宽度不宜过大	75
3.5.2	数据流图	41	4.5.5	模块的作用域应处于控制域之内	75
3.5.3	状态转换图	46	4.5.6	尽量降低模块的接口复杂度	77
3.6	数据字典	47	4.5.7	设计单入口、单出口的模块	77
3.6.1	数据字典的概念和组成	47	4.5.8	模块功能应该可以预测	77
3.6.2	数据字典各部分的描述	48	4.6	面向数据流的设计方法	77
3.6.3	词条描述	49	4.6.1	SD方法实施的步骤	78
3.6.4	数据字典中的符号	51	4.6.2	变换分析	79
3.7	需求规格说明书	52	4.6.3	事务分析	87
3.7.1	引言	53	4.6.4	软件模块结构的改进	88
3.7.2	任务概述	53	4.6.5	设计后的处理	90
3.7.3	需求规定	53	4.7	概要设计说明书	90
3.7.4	运行环境规定	54	实训		91
实训		54	习题四		93
习题三		56	第5章	详细设计	96
第4章	概要设计	60	5.1	详细设计的任务与原则	96
4.1	软件设计过程	60	5.1.1	详细设计的任务	96
4.1.1	软件设计过程	60	5.1.2	详细设计的原则	97
4.1.2	目标系统的运行环境	62	5.2	结构化程序设计	97
4.2	概要设计的目标和任务	62	5.2.1	结构化程序设计的概念	97
4.2.1	概要设计的目标	62	5.2.2	结构化和非结构化程序设计方法的比较	97
4.2.2	概要设计的任务	62	5.2.3	结构化程序设计的原则	98
4.3	概要设计原则	63	5.2.4	结构化程序设计的优点	99
4.3.1	模块化	63	5.3	过程设计工具	99
4.3.2	抽象	65	5.3.1	程序流程图	99
4.3.3	自顶向下, 逐步细化	65	5.3.2	盒图(N-S图)	102
4.3.4	信息隐蔽	66			

5.3.3	PAD图	102	6.3.4	输入/输出方式	126
5.3.4	判定表	103	6.4	程序效率	127
5.3.5	判定树	104	6.4.1	程序效率的准则	127
5.3.6	PDL语言	104	6.4.2	算法对效率的影响	127
5.4	用户界面设计	106	6.4.3	存储效率	127
5.4.1	用户界面设计的“黄金规则”	106	6.4.4	输入/输出效率	128
5.4.2	用户界面设计过程	106	6.5	冗余编程及容错技术	128
5.4.3	用户界面的主要形式	107	6.5.1	冗余编程	128
5.4.4	界面设计应考虑的因素	109	6.5.2	软件容错技术	129
5.4.5	用户界面设计分析	109	6.6	程序复杂性的度量	132
5.4.6	用户界面的质量要求	111	6.6.1	代码行度量法	132
5.5	数据库设计	111	6.6.2	McCabe度量法	133
5.5.1	数据需求分析	112	实训		134
5.5.2	概念结构设计	112	习题六		135
5.5.3	逻辑结构设计	112	第7章 面向对象的分析与设计		137
5.5.4	物理结构设计	113	7.1	面向对象概述	137
5.5.5	数据库的实施	113	7.1.1	面向对象方法	137
5.6	接口设计	113	7.1.2	面向对象的基本概念	139
5.6.1	软件接口设计的依据	113	7.1.3	面向对象的特性	145
5.6.2	软件接口的类型	113	7.2	面向对象分析与对象模型化技术	146
5.6.3	应用程序编程接口	114	7.2.1	面向对象的分析过程	146
5.7	详细设计说明书	114	7.2.2	面向对象的开发方法	148
实训		116	7.2.3	对象模型化技术	149
习题五		118	7.2.4	Coad&Yourdon面向对象分析	153
第6章 程序编码		120	7.3	面向对象设计	158
6.1	程序编码的目的和任务	120	7.3.1	面向对象设计准则	158
6.1.1	程序编码的目的	120	7.3.2	面向对象设计的启发式规则	160
6.1.2	程序编码的任务	121	7.3.3	Coad&Yourdon设计方法	161
6.2	程序设计语言	121	7.4	面向对象编程	164
6.2.1	程序设计语言的分类	121	7.4.1	面向对象语言的优点	164
6.2.2	程序设计语言的选择	122	7.4.2	面向对象语言的技术特点	165
6.3	编码风格	123	7.4.3	选择面向对象语言的现实因素	167
6.3.1	源程序文档化	123	7.4.4	程序设计风格	168
6.3.2	数据说明	124	案例：自动取款机(ATM)系统的分析		
6.3.3	语句结构	125	与设计		169

习题七	178	9.3.1 初始阶段	229
第 8 章 统一建模语言 UML	180	9.3.2 细化阶段	229
8.1 UML 概述	180	9.3.3 构造阶段	230
8.1.1 UML 的产生和发展	180	9.3.4 交付阶段	231
8.1.2 UML 的组成	181	9.4 RUP 的四个要素	232
8.1.3 UML 视图	182	9.4.1 人员至关重要	233
8.1.4 UML 的主要特点	184	9.4.2 项目创造产品	233
8.1.5 UML 的应用	185	9.4.3 产品不仅仅是代码	234
8.2 通用模型元素	186	9.4.4 过程指导项目	234
8.2.1 模型元素	186	9.4.5 工具对于过程不可或缺	235
8.2.2 约束	187	9.5 RUP 制品	235
8.2.3 依赖关系	187	9.5.1 模型	235
8.2.4 细化	188	9.5.2 其他制品	236
8.2.5 注释	188	9.6 RUP 的核心 workflow	236
8.3 静态建模机制	189	9.6.1 业务建模	237
8.3.1 用例图	189	9.6.2 需求	238
8.3.2 类图和对象图	193	9.6.3 分析	240
8.3.3 包图	200	9.6.4 设计	241
8.3.4 构件图	201	9.6.5 实现	244
8.3.5 配置图	202	9.6.6 测试	246
8.4 动态建模机制	203	9.6.7 部署	248
8.4.1 消息	203	9.6.8 配置和变更管理	249
8.4.2 顺序图	204	9.6.9 项目管理	249
8.4.3 协作图	205	9.6.10 环境	249
8.4.4 状态图	207	9.7 RUP 裁剪	251
8.4.5 活动图	210	9.8 RUP 的迭代开发模式	251
案例: 医院电话挂号系统	213	习题九	253
实训	217	第 10 章 软件测试	255
习题八	221	10.1 软件测试基础	255
第 9 章 统一软件开发过程 RUP	223	10.1.1 软件测试的意义	255
9.1 RUP 概述	223	10.1.2 软件测试的目标	256
9.1.1 RUP 的意义	223	10.1.3 软件测试的原则	256
9.1.2 RUP 的特点	224	10.1.4 软件测试的信息流	257
9.2 RUP 的二维开发模型	227	10.1.5 软件测试的对象	258
9.3 RUP 开发阶段及里程碑	228	10.1.6 软件测试与软件开发阶段的关系	258

10.1.7	软件测试有关概念	259	11.2.3	完善性维护	310
10.2	软件测试方法	260	11.3	软件维护成本	310
10.2.1	静态测试和动态测试	260	11.4	软件维护的实施	310
10.2.2	白盒测试和黑盒测试	260	11.4.1	软件维护机构	310
10.2.3	测试用例的设计	262	11.4.2	软件维护报告	311
10.3	软件测试策略	281	11.4.3	软件维护工作流程	312
10.4	软件测试过程	281	11.4.4	软件维护步骤	312
10.4.1	单元测试	282	11.4.5	编写维护档案记录	313
10.4.2	集成测试	283	11.4.6	维护评价	314
10.4.3	确认测试	285	11.5	软件的可维护性	314
10.4.4	系统测试	287	11.5.1	影响软件可维护性的因素	314
10.4.5	调试	288	11.5.2	软件可维护性的度量	315
10.5	面向对象的测试	290	11.5.3	提高软件可维护性的方法	318
10.5.1	面向对象测试模型	291	11.6	软件维护的副作用	321
10.5.2	面向对象分析的测试	291	11.6.1	软件维护副作用的类型	321
10.5.3	面向对象设计的测试	293	11.6.2	控制软件维护副作用的策略	322
10.5.4	面向对象编程的测试	294	习题十一		322
10.5.5	面向对象的单元测试	294	第 12 章	软件项目管理	324
10.5.6	面向对象的集成测试	295	12.1	软件项目管理概述	324
10.5.7	面向对象的系统测试	295	12.1.1	软件项目管理的意义	324
10.6	测试计划和分析报告	295	12.1.2	软件项目的特点	325
10.6.1	软件测试计划	296	12.1.3	软件项目管理的职能	325
10.6.2	测试分析报告	298	12.2	软件项目计划	325
实训		300	12.2.1	软件项目计划的目标和风险分析	325
习题十		302	12.2.2	软件项目计划的类型	325
第 11 章	软件维护	305	12.2.3	项目实施计划中任务的划分	326
11.1	软件维护概述	305	12.2.4	项目的进度安排	326
11.1.1	软件维护的定义	305	12.3	软件项目组织	328
11.1.2	软件维护的类型	306	12.3.1	组织原则	329
11.1.3	影响软件维护工作量的因素	307	12.3.2	组织结构的模式	329
11.1.4	造成软件维护困难的因素	308	12.3.3	程序设计小组的组织形式	330
11.1.5	结构化维护与非结构化维护	308	12.4	软件项目的人员配备	331
11.2	软件维护策略	309	12.4.1	项目开发各阶段所需人员	332
11.2.1	改正性维护	309	12.4.2	配备人员的原则	332
11.2.2	适应性维护	309	12.4.3	对项目经理的要求	333

12.4.4	评价软件开发人员的标准	333	习题十二	344
12.5	软件配置管理	333	第 13 章 软件复用技术	346
12.5.1	软件配置管理的概念	333	13.1 软件复用概述	346
12.5.2	软件配置管理的基本目标	334	13.1.1 软件复用的意义	346
12.5.3	基线	334	13.1.2 软件复用的层次	347
12.5.4	软件配置项	334	13.1.3 软件成分的重用级别	348
12.5.5	版本控制	335	13.2 可复用的软件成分	348
12.5.6	变更控制	336	13.3 软件复用过程	349
12.5.7	软件配置管理应注意的问题	337	13.4 软件复用环境	350
12.6	软件质量管理	337	13.5 面向对象的复用技术	351
12.6.1	软件质量的含义	337	13.5.1 类构件的复用	351
12.6.2	影响软件质量的因素	337	13.5.2 类库	352
12.6.3	软件质量保证措施	339	13.5.3 产生所需类的次序	353
12.7	软件能力成熟度模型	340	13.5.4 既存类的复用方法	353
12.7.1	软件过程的度量	340	习题十三	354
12.7.2	关键过程域	340	参考文献	356
实训		342		

1

软件工程概述

软件工程是研究软件开发技术和软件项目管理的一门工程学科，从工程化的角度来指导软件开发、测试和项目管理等活动。

本章主要讲授软件工程的基本概念和知识，内容包括软件的概念和特点、软件危机、软件工程的观念及原理、软件生存周期及其模型等，本章的内容是学习以后各章的基础。

1.1 软件的概念和特点

1.1.1 计算机系统的构成及实现

计算机系统由硬件系统及软件系统构成，硬件系统的实现靠硬件工程，软件系统的实现靠软件工程，如图 1-1 所示。

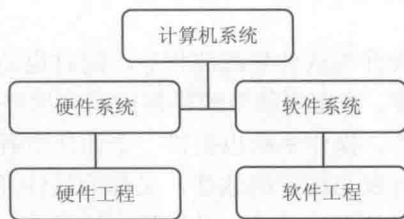


图 1-1 计算机系统的构成及实现

1.1.2 软件的概念

软件 (Software) 是计算机系统中与硬件相互依存的另一部分，它包括程序、数据及其相

关文档的完整集合。

软件=程序+数据+文档

程序是为实现软件的功能和性能要求而编写的指令序列。

数据是指使程序能够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资料。

1.1.3 软件的特点

与计算机硬件相比，计算机软件具有如下特点：

- (1) 软件是一种逻辑实体而非物理实体，因而软件具有抽象性。
- (2) 软件的开发是人智力的高度发挥，而不是传统意义上的硬件制造。
- (3) 软件可能被废弃，但不会用坏，不存在磨损、消耗问题。
- (4) 软件的开发和运行常常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。
- (5) 软件的开发至今尚未完全摆脱手工艺的开发方式，使软件的开发效率受到很大限制。
- (6) 软件开发是一个复杂的过程。主要表现在实际问题的复杂性和程序逻辑结构的复杂性两个方面。
- (7) 软件成本非常高昂。

1.1.4 计算机软件的分类

计算机软件一般分为系统软件和应用软件两大类。

1. 系统软件

系统软件为计算机使用提供最基本的功能，可分为操作系统和支撑软件，其中操作系统是最基本的软件。系统软件是负责管理计算机系统中各种独立的硬件，使得它们可以协调工作。系统软件使得计算机使用者和其他软件将计算机当作一个整体而不需要顾及到底层每个硬件是如何工作的。

操作系统是用来管理计算机硬件与软件资源的程序，同时也是计算机系统的内核与基石。操作系统身负诸如管理与配置内存、决定系统资源供需的优先次序、控制输入与输出设备、操作网络与管理文件系统等基本事务。操作系统也提供一个让使用者与系统交互的操作接口。

支撑软件是支撑各种软件的开发与维护的软件，又称为软件开发环境（SDE）。它主要包括环境数据库、各种接口软件和工具组；包括一系列基本的工具，比如编译器、数据库管理、存储器格式化、文件系统管理、用户身份验证、驱动管理、网络连接等方面的工具。著名的软件开发环境有 IBM 公司的 Web Sphere、微软公司的 Studio.NET 等。

2. 应用软件

系统软件并不针对某一特定应用领域，而应用软件则相反，不同的应用软件根据用户和

所服务的领域提供不同的功能。

应用软件是为了某种特定的用途而被开发的软件。它可以是一个特定的程序，比如一个图像浏览器；也可以是一组功能联系紧密，可以互相协作的程序的集合，比如微软的 Office 软件；也可以是一个由众多独立程序组成的庞大的软件系统，比如数据库管理系统。

1.2 软件的发展和软件危机

1.2.1 计算机软件的发展过程

软件是由计算机程序和程序设计的概念发展演化而来的，是在程序和程序设计发展到一定规模并且逐步商品化的过程中形成的。软件开发经历了程序设计阶段、软件设计阶段和软件工程阶段的演变过程。

1. 程序设计阶段

程序设计阶段出现在 1946~1955 年。此阶段的特点是：尚无软件的概念，程序设计主要围绕硬件进行开发，规模很小，工具简单，无明确分工（开发者和用户），程序设计追求节省空间和编程技巧，无文档资料（除程序清单外），主要用于科学计算。

2. 软件设计阶段

软件设计阶段出现在 1956~1970 年。此阶段的特点是：硬件环境相对稳定，出现了“软件作坊”的开发组织形式。开始广泛使用产品软件（可购买），从而建立了软件的概念。随着计算机技术的发展和计算机应用的日益普及，软件系统的规模越来越庞大，高级编程语言层出不穷，应用领域不断拓宽，开发者和用户有了明确的分工，社会对软件的需求量剧增。但软件开发技术没有重大突破，软件产品的质量不高，生产效率低下，从而导致了“软件危机”的产生。

3. 软件工程阶段

自 1970 年起，软件开发进入了软件工程阶段。由于“软件危机”的产生，迫使人们不得不研究、改变软件开发的技术手段和管理方法。从此软件发展进入了软件工程时代。此阶段的特点是：硬件已向巨型化、微型化、网络化和智能化四个方向发展，数据库技术已成熟并广泛应用，第三代、第四代语言出现；第一代软件技术——结构化程序设计在数值计算领域取得优异成绩；第二代软件技术——软件测试技术、方法、原理用于软件生产过程；第三代软件技术——处理需求定义技术用于软件需求分析和描述；第四代软件技术——使软件工程公认的模块化、信息隐蔽、抽象、局部化、软件重用等原则在面向对象机制下得到了充分的体现。

每个发展阶段都具有不同的特点，见表 1-1 所示。

1.2.2 软件危机

20 世纪 60 年代末 70 年代初，西方工业发达国家经历了一场“软件危机”。这场软件危机表现在：一方面软件十分复杂，价格昂贵，供需差日益增大，另一方面软件开发时又常常受挫，

质量差,指定的进度和完成日期很少能按时实现,研制过程很难管理,即软件的研制往往失去控制。

表 1-1 计算机软件发展的三个阶段及其特点

阶段	程序设计	软件设计	软件工程
特点			
软件所指	程序	程序及说明书	程序+数据+文档
主要程序设计语言	汇编及机器语言	高级语言	软件语言
软件工作范围	程序编写	设计和测试	整个软件生命周期
需求者	程序设计者本人	少数用户	市场用户
开发软件的组织	个人	开发小组	开发小组及大、中型开发机构
软件规模	小型	中、小型	大、中、小型
决定质量的因素	个人技术	小组技术水平	技术与管理水平
开发技术和手段	子程序、程序库	结构化程序设计	数据库、开发工具、集成开发环境、工程化开发方法、标准和规范、网络及分布式开发、面向对象技术、计算机辅助软件工程
维护责任者	程序设计者	开发小组	专职维护人员
硬件的特征	高价、存储量小、可靠性差	降价,速度、容量和可靠性明显提高	向超高速、大容量、网络化、微型化方向发展
软件的特征	完全不受重视	软件的技术发展不能满足需求,出现软件危机	开发技术有进步,但仍未完全摆脱软件危机

落后的软件生产方式无法满足迅速增长的计算机软件需求,从而导致软件开发与维护过程中出现一系列严重问题的现象称为软件危机。

1. 软件危机的表现

- (1) 经费预算经常突破,完成时间一再拖延。
- (2) 开发的软件不能满足用户要求。
- (3) 开发的软件可靠性差。
- (4) 开发的软件可维护性差。

2. 软件危机产生的原因

- (1) 软件规模越来越大,结构越来越复杂。
- (2) 软件开发管理困难。
- (3) 软件开发费用不断增加。
- (4) 软件开发技术落后。
- (5) 生产方式落后。