



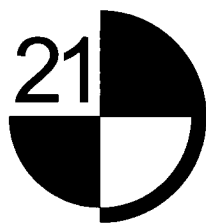
21世纪高职高专规划教材
软件专业系列

软件开发 生命周期

马在强 主编
李明龙 李驰 副主编



清华大学出版社



21世纪高职高专规划教材

软件专业系列

软件开发 生命周期

马在强 主编
李明龙 李驰 副主编

清华大学出版社
北京

内 容 简 介

本书以通俗易懂的语言,全面、系统地介绍了软件开发生命周期的各个环节。主要包括软件开发生命周期概述、软件过程的基本活动、软件过程的保护性活动、常用软件文档的书写四大部分。其中软件过程的基本活动既包括传统的结构化方法,也包括现在比较流行的面向对象的方法。为帮助读者进一步理解每一章的内容,我们专门为每一章设立了一个常见问题与回答,对读者在学习过程中可能会产生的疑问以自问自答的形式予以阐述。

本书可作为高职高专计算机软件专业的教材,也适合刚刚从事软件开发的技术人员阅读和参考。

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

软件开发生命周期/马在强主编. —北京:清华大学出版社,2009.1

21 世纪高职高专规划教材. 软件专业系列

ISBN 978-7-302-18613-7

I. 软… II. 马… III. 软件开发—高等学校:技术学校—教材 IV. TP311.52

中国版本图书馆 CIP 数据核字(2008)第 146087 号

责任编辑:孟毅新

责任校对:李 梅

责任印制:王秀菊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

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

印 刷 者:北京市清华园胶印厂

装 订 者:三河市溧源装订厂

经 销:全国新华书店

开 本:185×230 印 张:15.5 字 数:310 千字

版 次:2009 年 1 月第 1 版 印 次:2009 年 1 月第 1 次印刷

印 数:1~4000

定 价:23.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:027389-01

前言

软件开发生命周期

《软件开发生命周期》是高职高专计算机专业学生必修的一门专业课,是研究软件开发、维护和管理的一门工程性学科,是计算机应用及相关专业的主干课程,也是分析设计人员、程序开发人员、软件测试人员、软件管理人员、软件售前和售后工程师、软件高层决策者等相关人员必修的专业课程。它的研究内容非常广泛,主要研究软件的说明、开发、维护和管理等方面的内容。随着计算机技术的不断发展,这一学科的重要性已经逐渐为人所熟悉和重视。

随着我国软件业的迅速发展,软件开发生命周期在高职高专教育中占有十分重要的地位。高职高专教育的主要目标是培养具有一定专业理论水平与较强动手能力的应用型人才,而软件开发生命周期的实践性和应用性特点将为实现这一目标奠定良好的基础。因此,重视和加强软件开发生命周期教学对提高专业教学质量和高职高专学生的专业水平及能力具有非常重要的实际意义。

本书的首要特点就是通俗易懂。软件开发生命周期这门课程不可避免地会涉及很多专业术语和概念,大多数传统的软件开发生命周期书籍为追求概念准确,基本上是直接给出专业术语,这对高职高专这个层次的大多数学生有点勉为其难,本书力图多打一些比喻,多举一些实例,以通俗易懂的语言来阐述这些难懂的术语和概念。

本书另一个特点就是在软件开发生命周期的大框架上介绍得比较全面,而对某些具体的环节并不深入探讨,做到够用、实用就好,为其今后进一步就软件开发的某一个领域的深入学习打下基础。比如,对软件过程管理、配置管理、项目管理等都有介绍,但并不深入。这样做的目的主要是让学生对软件开发生命周期的整个过程有一个全局性的认识,并不要求学生马上成为每一个环节的专家,而是让学生学完本门课程后,根据自己的兴趣和具体情况来决定发展方向。例如,他想今后成为一名软件测试人员,就应专门阅读软件测试方面的专著,如果他想成为一名项目经理,就应该专门阅读软件项目管理方面的专著。

本书还有一个特点就是指导学生书写一些常用的软件文档,这不仅有利于学生完成其毕业论文,也有利于学生今后走上工作岗位后能很快上手软件开发工作。

本书共分为4篇。第一篇是软件开发生命周期的基本概念,包括第1、2章的内容,主要讲解了包括软件、软件危机和软件开发生命周期学等在内的诸多相关概念以及常见的几种软件过程模型。第二篇是软件过程的基本活动,包括第3~10章的内容,分别从传统的结构化方法和现在比较流行的面向对象方法两个方面对软件的分析、设计、编码、测试和维护等软件过程中的基本活动进行了详细的讲解。第三篇是软件过程的保护性活动,包括第11~14章的内容,主要介绍了一些常见的保护性活动,如软件项目管理、软件配置管理、软件质量保证和软件过程管理,以使读者大致了解每个保护性活动是怎么一回事,都包含哪些内容,以便为今后就某一个方面的深入学习打下基础。第四篇是软件文档编写指南,即第15章的内容,主要对常见的一些文档的书写格式和要求作了简要描述。

本书由四川托普信息技术职业学院马在强教授担任主编,李明龙、李驰担任副主编,杜雪平、毛红霞、刘正龙参加编写。其中主编负责全书的编写大纲的制定和编写要求,提出样章编写要求和样章审定,二审参编各章,提出明确具体的修改意见。副主编一审参编各章,提出明确具体的修改意见并报送主编。各章具体分工如下:第1、2、5、11~14由李驰编写,第6~9章由杜雪平编写,第3、4章由毛红霞编写,第10章由刘正龙编写,第15章由李驰和刘正龙编写。

本书能够出版,是与主编所在学院对教材编写工作的支持分不开的,在此表示感谢。由于时间仓促,水平有限,书中难免有不足之处。欢迎广大读者把你们的意见、建议和要求反馈我们;使用本书的老师,把你们的教学意见反馈我们,以便再版时继续完善。主编的电子邮箱是 mazaiqiang@scsoftcollege.com。

作 者

2008年10月

目 录

软件开发生命周期

第一篇 软件开发生命周期概念

第 1 章 软件开发生命周期基本概念	3
1.1 软件的基本概念	3
1.1.1 软件定义	3
1.1.2 软件的特征	4
1.1.3 软件的分类	5
1.1.4 软件生产的发展	7
1.2 软件危机	7
1.2.1 软件危机的表现	7
1.2.2 软件危机的原因	8
1.2.3 软件危机的解决	11
1.3 软件开发生命周期	11
1.3.1 软件开发生命周期的定义	11
1.3.2 软件开发生命周期的基本原理	12
1.3.3 软件开发生命周期的作用	13
小结	14
常见问题与回答	15
习题	16
第 2 章 软件过程	17
2.1 软件过程的基本概念	17
2.1.1 软件过程的定义	17
2.1.2 软件过程的活动	18

2.2 软件开发生命周期	18
2.3 软件过程模型	21
2.3.1 瀑布模型	21
2.3.2 快速原型模型	23
2.3.3 增量模型	23
2.3.4 螺旋模型	24
2.3.5 喷泉模型	26
小结	27
常见问题与回答	27
习题	28

第二篇 软件过程的基本活动

第3章 结构化分析	31
3.1 可行性分析	31
3.1.1 可行性分析的任务	31
3.1.2 可行性分析的步骤	33
3.1.3 系统流程图	34
3.2 需求分析	35
3.2.1 需求分析的任务与步骤	36
3.2.2 实体—关系图	41
3.2.3 数据流图	43
3.2.4 状态变迁图	49
3.2.5 数据字典	50
3.2.6 需求规格说明与评审	53
小结	54
习题	55
第4章 结构化设计	56
4.1 概要设计	57
4.1.1 概要设计的任务与步骤	57
4.1.2 概要设计的基本原理	58
4.1.3 概要设计的启发式规则	61
4.1.4 概要设计的方法	62

4.2 详细设计	65
4.2.1 详细设计的基本概念	65
4.2.2 详细设计的工具	66
小结	72
习题	72
第5章 结构化实现	73
5.1 编码	73
5.1.1 源程序的质量要求	73
5.1.2 源程序的编码风格	74
5.1.3 程序设计语言	79
5.2 测试	82
5.2.1 软件测试的基础	82
5.2.2 软件测试的方法	86
5.2.3 黑盒测试的用例设计	87
5.2.4 白盒测试的用例设计	91
5.2.5 测试策略	97
小结	106
常见问题与回答	107
习题	108
第6章 面向对象的概念	109
6.1 面向对象思想产生	109
6.1.1 传统方法存在的问题	109
6.1.2 产生问题的原因	110
6.1.3 面向对象的主要思想	110
6.2 基本概念	111
6.2.1 对象与类	111
6.2.2 其他概念	114
6.3 UML 基本概念和描述元素	117
6.3.1 UML 简介	117
6.3.2 UML 组成及应用	118
6.3.3 类与对象的表示	118
6.4 面向对象建模	119

6.4.1	对象模型	119
6.4.2	动态模型	121
6.4.3	功能模型	122
6.4.4	三种模型之间的关系	123
小结		124
习题		124
第7章	面向对象的分析	125
7.1	概述	125
7.2	问题域建模	126
7.3	用例建模	135
7.3.1	用例图表示	135
7.3.2	用例分析	137
7.3.3	用例图和用例描述设计实例	139
小结		140
习题		141
第8章	面向对象设计	142
8.1	面向对象系统设计基础	142
8.1.1	面向对象设计的准则	142
8.1.2	面向对象设计的启发规则	143
8.2	子系统设计	144
8.2.1	问题域子系统的设计	145
8.2.2	设计人机交互子系统	145
8.2.3	应用控制子系统的设计	146
8.2.4	设计数据管理子系统	147
8.2.5	与问题域有关的设计	148
8.3	通过MVC设计模式设计	149
小结		150
习题		151
第9章	面向对象的实现	152
9.1	程序设计语言	152
9.1.1	面向对象语言特点	152

9.1.2 面向对象语言选择	153
9.1.3 实际因素	155
9.1.4 程序设计风格	155
9.2 测试策略及测试用例的设计	157
9.2.1 面向对象分析的测试	157
9.2.2 面向对象设计的测试	158
9.2.3 面向对象实现的测试	158
9.2.4 面向对象单元测试	159
9.2.5 面向对象集成测试	159
9.2.6 面向对象确认测试	160
小结	160
习题	160
第 10 章 软件维护	161
10.1 软件维护概述	161
10.2 软件维护的特点	162
10.2.1 结构化维护和非结构化维护	162
10.2.2 软件维护的成本	162
10.2.3 软件维护的副作用	163
10.3 软件维护活动	163
10.3.1 软件维护机构	163
10.3.2 软件维护报告	164
10.3.3 软件维护工作流程	164
10.3.4 软件维护档案记录	165
10.3.5 软件维护评价	165
小结	165
常见问题与回答	166
习题	166

第三篇 软件过程的保护性活动

第 11 章 软件项目管理	169
11.1 软件项目管理概述	169
11.2 软件度量	170

11.2.1	软件代码行度量	171
11.2.2	功能度量	171
11.3	软件项目估算	174
11.3.1	软件项目估算概述	174
11.3.2	软件开发成本和工作量估算方法	175
11.4	软件项目风险	177
11.4.1	风险的分类	178
11.4.2	风险识别	179
11.4.3	风险估计与评价	180
11.4.4	风险缓解、监控和管理	181
11.5	软件项目进度	183
11.5.1	估算软件开发时间	184
11.5.2	进度安排和跟踪	184
11.6	软件项目组织及人员配备	186
11.6.1	软件项目组织	186
11.6.2	人员配备	188
	小结	188
	常见问题与回答	189
	习题	190
第 12 章	软件配置管理	191
12.1	软件配置概述	191
12.1.1	软件配置项	192
12.1.2	基线	193
12.2	软件配置管理过程	194
12.2.1	配置标识	195
12.2.2	版本控制	195
12.2.3	变更控制	195
12.2.4	配置审计	197
12.2.5	配置状态报告	197
	小结	198
	常见问题与回答	198
	习题	199
第 13 章	软件质量保证	200
13.1	软件质量的基本概念	200

13.2 软件质量的度量和评价	202
13.3 软件质量的保证方法	206
13.3.1 SQA 活动	206
13.3.2 软件评审	206
小结	208
常见问题与回答	208
习题	209
第 14 章 软件过程管理	210
14.1 CMM 基本概念	210
14.1.1 软件过程管理的概念	210
14.1.2 能力成熟度等级	211
14.1.3 能力成熟度模型的体系结构	212
14.1.4 成熟度提问单	214
14.2 CMM 实施步骤	214
小结	216
常见问题与回答	217
习题	217

第四篇 软件文档编写指南

第 15 章 软件文档的编写	221
15.1 《需求规格说明书》的撰写	221
15.2 《概要设计说明书》的撰写	224
15.3 《详细设计说明书》的撰写	226
15.4 《测试报告》的撰写	228
15.5 软件文档编写的注意事项	231
15.6 软件文档编写技巧	233
小结	234
常见问题与回答	234
习题	234
参考文献	235

第一篇

软件开发生命周期概念

第 1 章 软件开发生命周期基本概念

第 2 章 软件过程

软件开发生命周期基本概念

本章目标

- 了解软件的定义、特征和分类
- 了解软件产生的发展阶段
- 理解软件危机的表现、原因和解决途径
- 了解软件开发生命周期的含义、基本原理和作用

本章导读

什么是软件呢？难道仅仅就是指我们平时看到的一张光碟、一张磁盘或者是一堆代码吗？很多人都听说过经济危机，那软件危机又是什么，有经济危机那么可怕吗？又该如何解决软件危机呢？同样，很多人也听说过水利工程、电力工程、建筑工程，那软件开发生命周期又是指什么呢？它究竟是一门技术学还是管理学呢？这一连串的疑问将通过本章的学习找到答案。

1.1 软件的基本概念

1.1.1 软件定义

什么是软件呢？这个看似简单的问题，直到 20 世纪 70 年代以后才有了比较科学的认识，而在此之前人们的认识都很片面、模糊甚至荒谬。有人说软件就是一张光碟、一张磁盘，因为他每次买软件其实就是买光碟。这种说法显然是荒谬的，因为光碟、磁盘这些只不过是程序、软件的载体，而并不是软件本身。另一些人似乎高明一些，说软件就是一堆程序，因为平时大家都把软件开发直接说成程序开发。这种说法也是片面的，应该说软件确实离不开程序，但是程序也仅仅是软件这个概念的一部分，而不是全部。

“软件”这一名词是在 20 世纪 60 年代初从国外传来的。由于“software”一词由 soft 和 ware 两部分组成，所以有人译作“软制品”，也有人译作“软体”，现在比较统一的说法叫

“软件”。对于它的一种公认的解释为：软件是计算机系统中与硬件相互依存的另一部分，它是包括程序、数据和相关文档的完整集合。

- 程序一般认为是能够完成预定功能和性能的可执行的指令。对于面向过程的程序，我们一般把它理解为“算法”结合“数据结构”；而对于面向对象的程序，我们一般把它理解为“对象”结合“消息”。
- 数据一般认为是使程序能够适当地操作信息的数据结构。
- 文档是与程序开发、维护和使用有关的图文材料。文档在软件开发生命周期中特别重要，文档是否规范与齐全，是衡量软件企业是否成熟的重要标志之一。软件文档分为开发文档和管理文档两大类。开发文档主要由项目组书写，用于指导软件开发；管理文档主要由软件开发生命周期管理部门书写，用于指导软件管理和决策。

1.1.2 软件的特征

为了能够全面、正确地理解软件，让我们来看看软件的特点。

(1) 软件是一种逻辑实体，无形态，具有抽象性。人们可以把它记录在纸面上，存储在计算机的硬盘上，又或者刻在光碟上，但纸、硬盘和光碟都只不过是它的载体，而不是软件本身。软件这种无形体只能通过分析、思考和判断去理解它。

(2) 软件的运行和使用期间，没有硬件那样的机械、磨损、老化问题。任何硬件产品一开始并不好使用，随着磨合调整时期一过，产品达到了最佳的使用状态；但随着时间的推移，由于磨损等原因又变得很难使用甚至不可使用，如图 1-1(a) 所示。软件则不同，它不存在磨损和老化问题，但它存在退化问题。理想状态，随着软件的不断维护，软件会越来越好用。而实际情况往往是，随着软件的不断维护修改，每次修改都必不可少地引入新的错误，这样一次又一次修改，导致软件失效率升高，如图 1-1(b) 所示。

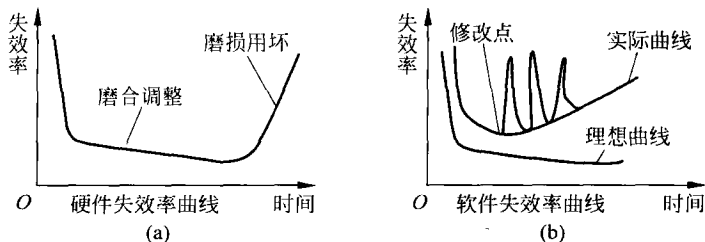


图 1-1 硬件与软件失效率对比

(3) 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。软件不能完全摆脱硬件单独活动，在开发和运行中必须以硬件提供的条件为依据。有的软件只能在某种型号的计算机上运行，而有的软件只能在某种操作系统上运行。

为了解除这种依赖性,在软件开发中提出了“软件移植”问题,并且把软件的可移植性作为衡量软件质量的因素之一。

(4) 软件的开发至今尚未完全摆脱手工的方式。人们一直期待软件的生产能够像硬件的生产那样实现全自动,将原料放入一个全自动机器,从另一端就可以输出成品。近年来软件技术虽然取得了不少进展,提出了许多新的开发方法,如组件技术、复用技术,也研制了一些高效的开发工具和开发环境,但这种期待软件生产实现全自动的梦想至今未能实现。软件的开发仍然以手工开发方式为主,开发的效率自然受到很大的限制。

(5) 软件是复杂的。开发软件是一种高强度的脑力劳动,没有哪一个软件人员认为这是一项轻松的工作。软件的复杂性主要表现在两个方面。一方面,可能来自程序逻辑结构的复杂性,如像操作系统这样复杂的系统软件,需要处理各种可能的情况,其复杂度是不言而喻的。另一方面,也可能来自它所反映的实际问题的复杂性,如开发一个图书馆管理信息系统,要求软件人员对图书领域的专门知识,诸如图书的编码规则和借还书的流程等有所了解。图 1-2 表明尽管软件技术有了不少的发展,但是发展的速度一直落后于软件需求的发展,而且这种差距越拉越大。

(6) 软件成本相当昂贵。大家知道,很多正版软件居然要卖到几千甚至上万元,似乎不可思议。这主要是因为软件的研制需要投入大量的、复杂的、高强度的脑力劳动,成本自然比较高。从图 1-3 可以看出一个戏剧性的变化趋势。随着时间的推移,花费在软件上的成本所占整个系统成本的比例逐年增加,在 20 世纪 80 年代以后,软件的开销就开始大大超过硬件的开销了,时至今日,这种趋势更为明显。

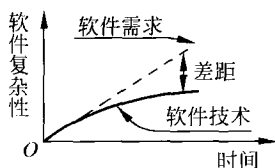


图 1-2 软件技术的发展落后于需求

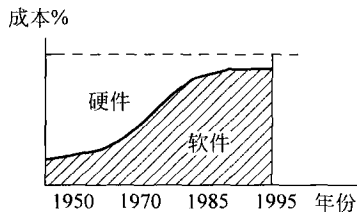


图 1-3 硬件、软件成本的比例变化

1.1.3 软件的分类

大家可能经常听到有人说这是“系统软件”，那是“大型软件”、“定制软件”等，这其实是对软件类型的一种描述。站在不同的角度可以对软件做出不同的分类，就像以身高可以将人分为高人和矮人，以体重可以将人分为胖人和瘦人。以下介绍一些常用的分类方式。