

黄明 梁旭 周丽梅 编著

全国高等教育自学考试

软件工程

习题详解

计算机及应用专业
独立本科段

1.5-44



全国高等教育自学考试

软件工程习题详解

（计算机及应用专业 独立本科段）

黄 明 梁 旭 周丽梅 编著



机械工业出版社

本书是根据“全国自学考试（计算机及应用专业 独立本科段）考试大纲”以及历年考题编写的。本书共分为4部分：第1部分是笔试应试指南；第2部分是笔试题解；第3部分是模拟试卷及参考答案；最后是附录，包括考试大纲和2002年下半年试卷。

本书紧扣考试大纲，内容取舍得当，叙述通俗易懂，附有大量考试题型类似的习题及答案，以检查读者对考点的掌握程度。

本书适用于准备参加全国自学考试（计算机及应用专业 独立本科段）的考生，也可作为大专院校和培训班的教学参考书。

图书在版编目（CIP）数据

软件工程习题详解/黄明等编著. —北京：机械工业出版社，2003.8
（全国高等教育自学考试）

ISBN 7-111-12830-3

I. 软... II. 黄... III. 软件工程—高等教育—自学考试—解题
IV. TP311.5-44

中国版本图书馆CIP数据核字（2003）第068542号

机械工业出版社（北京市百万庄大街22号 邮政编码100037）

策 划：胡毓坚

责任编辑：孙 业

责任印制：路 琳

北京蓝海印刷有限公司印刷·新华书店北京发行所发行

2003年8月第1版·第1次印刷

787mm×1092mm 1/16·13.25印张·324千字

0001—5000册

定价：21.00元

凡购本图书，如有缺页、倒页、脱页，由本社发行部调换

本社购书热线电话（010）68993821、88379646

封面无防伪标均为盗版

出版说明

全国高等教育自学考试指导委员会推出面向社会的高等自学考试, 经过 10 多年的实践, 已建立起一整套较为完善的规章制度和操作规程, 考试组织严密规范, 考试纪律严格; 坚持考试标准, 实行教考分离, 确保了毕业生的质量。它为没有机会进入高等学校的中国公民提供了接受高等教育的机会, 并以严格的国家考试保证了毕业生的质量, 获得了普遍赞誉。国家自考中心于 2002 年开始执行新的考试计划。新计划中开设的专业共 224 个, 其中专科 141 个占 63%, 独立本科段 61 个占 27%, 专本衔接专业 22 个占 10%。为帮助、指导广大自考生深入理解计算机及相关专业考试的基本概念, 灵活运用基本知识, 掌握解题方法和技巧, 熟悉考试模式, 进一步提高应试能力和计算机水平, 特编写了以下专业的基础课与专业课主要课程的习题详解。

- ◇ 计算机及应用专业 独立本科段
- ◇ 计算机信息管理专业 独立本科段
- ◇ 计算机网络专业 独立本科段
- ◇ 计算机及应用专业 专科

丛书特点:

1. 以 2002 年最新考试大纲为基准

本丛书是根据 2002 年最新考试大纲, 为参加全国高等教育自学考试考生编写的一套习题详解教材。

2. 例题反映了历届考试中的难度和水平

书中对大量的例题进行了分析, 所选例题都是在对最近几年考题深入研究的基础上精心筛选的, 从深度和广度上反映了历届考试中的难度和水平。

3. 作者经验丰富

本丛书的作者都是多年从事全国高等教育自学考试辅导的高等院校的教师。

读者对象:

- ◇ 准备参加全国高等教育自学考试的考生。
- ◇ 计算机及相关专业的本专科生。

前言

自学考试是对自学者进行以学历考试为主的国家高等教育学历考试。本书是为帮助和指导广大考生深入理解考点涉及的基本概念，灵活运用基本知识，掌握解题方法和技巧，熟悉考试模式，进一步提高应试能力和计算机水平而编写的。

全书共分 4 部分，即笔试应试指南、笔试题解、模拟试卷及参考答案和附录。书中所选试题均是对历年真题深入研究的基础上经过精心筛选的，从深度和广度上反映了考试的难度和水平。模拟试卷的题型分配与真题一致，这些题目是考试指导教师的多年积累，且在辅导班中实际多次使用过。

书中附录给出了“全国自学考试（计算机及应用专业 独立本科段）软件工程考试大纲”，以及“2002 年下半年全国自学考试软件工程试卷及参考答案”。

本书由黄明、梁旭、周丽梅编写。

由于编者水平有限，编写时间仓促，书中错误和不妥之处在所难免，请读者和专家批评指正。

读者在使用本书的过程中如有问题，可通过 E_mail 与我们联系：

dlhm@263.net

编 者

目 录

出版说明

前言

第 1 部分 笔试应试指南

1.1 笔试应试策略	2
1.2 笔试考点归纳	3
1.2.1 绪论	3
1.2.2 软件可行性研究与项目开发计划	5
1.2.3 软件需求分析	6
1.2.4 软件概要设计	10
1.2.5 软件详细设计	13
1.2.6 软件编码	15
1.2.7 软件测试	17
1.2.8 软件维护	21
1.2.9 软件开发的增量模型	24
1.2.10 面向对象的方法	27
1.2.11 软件质量与质量保证	34
1.2.12 软件工程管理	37
1.2.13 软件开发环境	40

第 2 部分 笔 试 题 解

2.1 绪论	44
2.1.1 单项选择题	44
2.1.2 填空题	45
2.1.3 名词解释	46
2.1.4 简答题	47
2.1.5 习题	49
2.2 软件可行性研究与项目开发计划	51
2.2.1 单项选择题	51
2.2.2 填空题	52
2.2.3 名词解释	53
2.2.4 简答题	53
2.2.5 应用题	55
2.2.6 习题	55

2.3	软件需求分析	57
2.3.1	单项选择题	57
2.3.2	填空题	58
2.3.3	名词解释	59
2.3.4	简答题	60
2.3.5	应用题	61
2.3.6	习题	64
2.4	软件概要设计	66
2.4.1	单项选择题	66
2.4.2	填空题	68
2.4.3	名词解释	69
2.4.4	简答题	70
2.4.5	应用题	72
2.4.6	习题	73
2.5	软件详细设计	75
2.5.1	单项选择题	75
2.5.2	填空题	77
2.5.3	名词解释	78
2.5.4	简答题	78
2.5.5	应用题	79
2.5.6	习题	80
2.6	软件编码	81
2.6.1	单项选择题	81
2.6.2	填空题	83
2.6.3	名词解释	84
2.6.4	简答题	84
2.6.5	习题	86
2.7	软件测试	87
2.7.1	单项选择题	87
2.7.2	填空题	89
2.7.3	名词解释	90
2.7.4	简答题	90
2.7.5	应用题	92
2.7.6	习题	93
2.8	软件维护	95
2.8.1	单项选择题	95
2.8.2	填空题	96
2.8.3	名词解释	98
2.8.4	简答题	98

2.8.5 习题	99
2.9 软件开发的增量模型	101
2.9.1 单项选择题	101
2.9.2 填空题	103
2.9.3 名词解释	104
2.9.4 简答题	104
2.9.5 习题	105
2.10 面向对象的方法	107
2.10.1 单项选择题	107
2.10.2 填空题	108
2.10.3 名词解释	109
2.10.4 简答题	110
2.10.5 应用题	111
2.10.6 习题	114
2.11 软件质量与质量保证	116
2.11.1 单项选择题	116
2.11.2 填空题	118
2.11.3 名词解释	119
2.11.4 简答题	120
2.11.5 习题	120
2.12 软件工程管理	122
2.12.1 单项选择题	122
2.12.2 填空题	123
2.12.3 名词解释	124
2.12.4 简答题	125
2.12.5 习题	126
2.13 软件开发环境	128
2.13.1 单项选择题	128
2.13.2 填空题	130
2.13.3 名词解释	131
2.13.4 简答题	131
2.13.5 习题	133
2.14 习题参考答案	134

第3部分 模拟试卷及参考答案

3.1 模拟试卷一及参考答案	168
3.1.1 模拟试卷一	168
3.1.2 参考答案	171
3.2 模拟试卷二及参考答案	173

3.2.1 模拟试卷二	173
3.2.2 参考答案	176
附录	181
附录 A 全国自学考试（计算机及应用专业 本科独立段）软件工程考试大纲	182
附录 B 2002 年下半年全国自学考试 软件工程试卷及参考答案	195
参考文献	201

第1部分

笔试应试指南

笔试应试策略

笔试考点归纳

1.1 笔试应试策略

全国自学考试（计算机及应用专业 独立本科段）软件工程考试大纲涵盖了绪论、软件可行性研究与项目开发计划、软件需求分析、软件概要设计、软件详细设计、软件编码、软件测试、软件维护、软件开发的增量模型、面向对象的方法、软件质量与质量保证、软件工程管理和软件开发环境 13 章内容。使用的教材是由全国高等教育自学考试指导委员会组编，陆丽娜编著的《软件工程》，2000 年 3 月由经济科学出版社出版。考试复习的过程中要紧紧围绕大纲的知识点，首先要对大纲涉及的各章基本概念熟练掌握。

第 1 章为绪论，属于基础知识，主要考察的是对一些概念的理解和记忆。没有难点，相对的重点是软件危机产生的原因以及软件工程面临的各种问题。占分量约为 4 分左右。

第 2 章为软件可行性研究与项目开发计划，主要深刻理解可行性研究的必要性，掌握可行性研究的任务及可行性研究的具体步骤。占分量为 5 分左右。

第 3 章为软件需求分析，是重点考察的内容，深刻理解需求分析阶段的概念及任务；数据流图、数据字典的作用及应用。熟练掌握结构化分析方法；对于某一个小型项目，能够使用数据流图、数据字典构造它的逻辑模型。占分量为 20 分左右。

第 4 章为软件概要设计，深刻理解模块化、抽象、信息隐蔽、模块独立性等概念，熟练掌握数据流类型的划分，将数据流图转换成软件结构图，并能根据优化准则将其优化。占分量在 20 分左右。

第 5 章为软件详细设计，理解 Jackson 方法的概念，学会使用 Jackson 方法设计输入输出数据结构和程序结构。占分量为 10 分左右。

第 6 章为软件编码，掌握几种常用的程序设计语言的特点，领会程序设计中应注意的问题。占分量为 15 分左右。

第 7 章为软件测试，掌握测试阶段的任务、测试方法及测试步骤，能够针对某一问题采用白盒法或黑盒法进行测试用例的设计。占分量为 10 分左右。

第 8 章为软件维护，掌握软件维护的内容、特点、方法、技术、文档。深刻理解如何提高软件可维护性。占分量为 5 分左右。

第 9 章为软件开发的增量模型，了解增量模型的基本思想、特点、模型的分类和各种模型的表示、快速原型的基本原理，深刻理解快速原型开发过程和步骤。占分量为 5 分左右。

第 10 章为面向对象的方法，掌握面向对象的基本思想、基本概念和两种模型的基本概念、构造方法以及面向对象的分析、面向对象的设计、面向对象的实现的过程；掌握如何画对象图、画状态图和画数据流程图。占分量为 5 分左右。

第 11 章为软件质量与质量保证，了解软件质量的定义，软件质量的度量与评价、软件质量保证的基本概念；软件质量度量模型、软件复杂性、软件可靠性、设计质量的评审和程序质量评审的具体内容，软件容错技术的概念。理解软件复杂性的几种常用度量方法；软件质量度量模型；实现容错软件的一般方法和容错软件的实现过程。占分量为 5 分左右。

第 12 章为软件工程管理，了解软件产品的各种特点与软件工程管理的内容、软件项目计划内容、软件开发成本估算、软件项目进度安排、软件配置管理等概念。深入理解软件项目计划的重要性及各个环节之间的联系、软件工程标准化的意义。占分量为 7 分左右。

第 13 章为软件开发环境，了解软件开发环境的概念、软件开发环境的分类、软件工具、软件工具分类、工具间集成、计算机辅助软件工程 CASE 的基本概念。理解软件环境的特点和分类，软件工具的分类与评价，工具间集成性与灵活性，CASE 的分类与生命周期；理解集成化 CASE、CASE 生存周期、软件工程环境；软件工程方法的研究在软件开发中的作用。占分量为 4 分左右。

考生要根据大纲里提供的考核点和考核要求进行复习，这样才能抓住重点，进行有效复习。在做练习时，要根据考试的题型进行练习，在掌握基本概念的基础上，掌握一定的解题技巧。软件工程的考试题型有：单项选择题、填空题、名词解释、简答题和应用题等题型。对于不同题型，要采用不同的答题方法。

单项选择题：这种题型考查考生的理解、推理分析、综合比较能力，评分客观。在答题时，可以直接得出正确答案，对于没有太大把握的试题，也可以采用排除法，经过分析比较逐步排除错误答案，最终选定正确答案。

填空题：这种题型常用于考核考生观察能力与运用有关公式、原理的能力。在答题时，无论有几个空，回答都应明确、肯定，考生在复习中最好的应对办法是对学科知识中最基本的知识、概念、原理等要牢记。

名词解释：这种题型着重考核考生对概念、基本原理理解的准确性，在复习的过程中对考试大纲涉及的一些基本概念和原理要熟练掌握。

简答题：这种题型一般围绕基本概念、原理及其联系进行命题，着重考核考生对概念、知识、原理的掌握、辨别和理解能力。在答题时，答案要有层次性、条理清晰、列出要点，同时加以简要扩展就可以。

应用题：这种题型着重考核考生分析、解决实际问题的能力，考核考生综合应用能力和创见性。在答题时，要综合运用所学知识进行分析和设计。

考生复习时在掌握知识点的同时也应抓住这些题型的特点，这样才能达到好的应试效果。

1.2 笔试考点归纳

1.2.1 绪论

1. 软件工程的产生

软件的生产经过了三个阶段：程序设计时代，程序系统时代，软件工程时代。

软件发展的第二阶段的末期，一些复杂的、大型的软件开发项目被提出来，但是，软件开发技术的进步并没有新的突破，开发人员素质又差，以上这些使在软件开发中遇到的问题找不到解决的办法，问题不断地积累，形成了尖锐的矛盾，导致了软件危机。

软件危机的主要表现是：

- (1) 对软件开发成本和进度的估计常常很不准确；
- (2) 用户对“已完成的”软件系统不满意的现象经常发生；
- (3) 软件产品的质量往往靠不住；
- (4) 软件通常没有适当的文档资料；
- (5) 软件成本在计算机系统总成本中所占的比例逐年上升；

(6) 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趋势;

(7) 软件常常不可维护。

为了克服软件危机, 人们提出了“软件工程”的概念。软件工程是指导计算机软件开发和维护的工程学科。采用工程的概念、原理、技术和方法来开发与维护软件, 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, 这就是软件工程。

2. 软件工程的过程和软件生存周期

软件工程包含七个过程:

(1) 获取过程。定义需方按合同获取一个系统、软件产品或服务的活动。

(2) 供应过程。定义供方向需方提供合同中的系统、软件产品或服务所需的活动。

(3) 开发过程。定义开发者和机构为了定义和开发软件或提供服务所需的活动。此过程包括需求分析、设计、编码、集成、测试、软件安装和验收等活动。

(4) 操作过程。定义操作者和机构为了在规定的运行环境中为其用户运行一个计算机系统所需要的活动。

(5) 维护过程。定义维护者和机构为了修改和管理软件, 使它处于良好运行状态所需要的活动。

(6) 管理过程。定义软件工程过程中各项管理活动, 包括: 项目开始和范围定义、项目管理计划、实施和控制、评审和评价、项目完成。

(7) 支持过程。支持过程对项目的生存周期过程给予支持。它有助于项目的成功并能提高项目的质量。

软件生命周期是软件工程最基本的重要概念, 它是指一个软件从提出开发要求开始直到该软件报废为止的整个时期。把整个生存周期划分为若干阶段, 使得每个阶段有明确的任务, 使规模大、结构复杂和管理复杂的软件开发变得容易控制和管理。

通常, 软件生命周期包括: 软件可行性研究与项目开发计划、需求分析、概要设计、详细设计、编码、测试和维护等活动, 可以将这些活动以适当方式分配到不同的阶段去完成。

3. 软件生存周期模型、方法和工具

模型是为了理解事物而对事物做出一种抽象, 它忽略不必要的细节, 它也是事物的一种抽象形式、一个规划、一个程式。

软件生存周期模型是描述软件开发过程中各种活动如何执行的模型。

常见的软件生存周期模型有: 瀑布模型、增量模型、螺旋模型、喷泉模型、变换模型以及基于知识的模型。

软件开发方法是一种使用早已定义好的技术集及符号表示习惯来组织软件生产的过程。

软件开发的目标是在规定的投资和时间内, 开发出符合用户需求的高质量的软件。为了达到此目的, 需要成功的开发方法。软件开发方法是克服软件危机的重要方面之一。从软件工程诞生以来, 人们很重视软件开发方法的研究, 已经提出了多种软件开发方法和技术。软件开发方法主要有: 结构化方法、Jackson 方法、维也纳开发方法(VDM)和面向对象的开发方法。

软件工具一般是指为了支持软件人员开发和维护活动而使用的软件。它可以放大人类的智力。

最初的软件工具是以工具箱的形式出现的, 一种工具支持一种开发活动, 然后将各种工

具简单组合起来,就构成工具箱。

软件开发环境的目的是使软件工具支持整个生存周期,不仅能支持软件开发和维护中的个别阶段,而且能支持从项目开发计划、需要分析、设计、编码、测试到维护等所有阶段,做到不仅支持各阶段中的技术工作,还要支持管理和操作工作,保持项目开发的高度可见性、可控制性和可追踪性。

计算机辅助软件工程可以简单地定义为软件开发的自动化,通常简称为 CASE。CASE 的实质是为软件开发提供一组优化集成的且能大量节省人力的软件开发工具,其目的是实现软件生存周期各环节的自动化并使之成为一个整体。

1.2.2 软件可行性研究与项目开发计划

1. 可行性研究

可行性研究的任务:用最小的代价确定在问题定义阶段所确定的系统的目标和规模是否能实现,所确定的问题是否可以解决,系统方案在经济上、技术上和操作上是否可以接受。

典型的可行性研究有下列具体步骤:

- (1) 确定项目规模和目标;
- (2) 研究正在运行的系统;
- (3) 建立新系统的高层逻辑模型;
- (4) 导出和评价各种方案;
- (5) 推荐可行的方案;
- (6) 编写可行性研究报告。

可行性研究报告是指将可行性研究各个步骤的结果写成清晰的文档,请用户和使用部门的负责人仔细审查,以决定是否继续这项工程以及是否接受分析员推荐的方案。

2. 系统流程图

系统流程图是描绘物理系统的传统工具,它用图形符号来表示系统中的各个元素,例如人工处理、数据处理、数据库、文件和设备等。它表达了系统中各个元素之间的信息流动的情况。

3. 成本-效益分析

- (1) 货币的时间价值。

成本估算的目的是进行项目投资。经过成本估算后,得到项目开发时需要的费用,该费用就是项目的投资。项目开发后,应取得相应的效益,有多少效益才合算,这就要考虑货币的时间价值。通常用利率表示货币的时间价值。

设年利率为 i , 现存入 P 元, n 年后可得钱数为 F , 若不计复利则:

$$F = P \times (1 + n \times i)$$

F 就是 P 元在 n 年后的价值。反之,若 n 年能收入 F 元,那么这些钱现在的价值是:

$$P = F / (1 + n \times i)$$

- (2) 投资回收期。

通常用投资回收期衡量一个开发项目的价值。投资回收期就是使累计的经济效益等于最初的投资费用所需的时间。投资回收期越短,就越快获得利润,则该项目就越值得开发。

- (3) 纯收入。

衡量项目价值的另一个经济指标是项目的纯收入，也就是在整个生存周期之内的累计经济效益(折合成现在值)与投资之差。这相当于投资开发一个项目与把钱存入银行中进行比较，看这两种方案的优劣。若纯收入为零，则项目的预期效益和在银行存款一样，但是开发一个项目要冒风险，因此，从经济观点看这个项目可能是不值得投资开发的。若纯收入小于零，那么这个项目显然不值得投资开发。

4. 项目开发计划

项目开发计划是一个管理性的文档，它的主要内容如下：

(1) 项目概述。

说明项目的各项主要工作；说明软件的功能、性能；为完成项目应具备的条件；用户及合同承包者承担的工作，完成期限及其他条件限制；应交付的程序名称，所使用的语言及存储形式；应交付的文档。

(2) 实施计划。

说明任务的划分，各项任务的责任人；说明项目开发进度，按阶段应完成的任务，用图表说明每项任务的开始时间和完成时间；说明项目的预算，各阶段的费用支出预算。

(3) 人员组织及分工。

说明开发该项目所需人员的类型、组成结构、数量等。

(4) 交付期限。

说明项目最后完工交付的日期。

1.2.3 软件需求分析

1. 需求分析的任务

需求分析是指开发人员要准确理解用户的要求，进行细致的调查分析，将用户非形式的需求陈述转化为完整的需求定义，再由需求定义转换到相应的形式功能规定(需求规格说明)的过程。需求分析虽处于软件开发过程的开始阶段，但它对于整个软件开发过程以及软件产品质量是至关重要的。

需求分析的基本任务是要准确地定义新系统的目标，回答系统必须“做什么”的问题。

2. 结构化分析方法

结构化分析(Structured Analysis, 简称 SA)，是面向数据流进行需求分析的方法。SA也是一种建模活动，该方法使用简单易读的符号，根据软件内部数据传递、变换的关系，自顶向下逐层分解，描绘出满足功能要求的软件模型。

SA 分析步骤：

(1) 了解当前系统的工作流程，获得当前系统的物理模型。当前系统指目前正在运行的系统，可能是需要改进的正在计算机上运行的软件系统。也可能是人工处理系统。

(2) 抽象出当前系统的逻辑模型。物理模型反映了系统“怎样做”的具体实现，去掉物理模型中非本质的因素，抽象出本质的因素。所谓的本质因素是指系统固有的、不依赖运行环境变化而变化的因素，任何实现均可这样做。非本质因素不是固有的，而是随实现不同而不同。

(3) 建立目标系统的逻辑模型。目标系统指待开发的系统。分析、比较目标系统与当前系统逻辑上的差别，然后对“变化的部分”重新分解，分析人员根据自己的经验，采用自顶

向下逐步求精的分析策略，逐步确定变化部分的内部结构，从而建立目标系统的逻辑模型。

(4) 作进一步补充和优化。为了完整描述目标系统，还要作一些补充：说明目标系统的人机界面；说明至今尚未详细考虑的细节。

3. 数据流图(DFD)

数据流图，简称 DFD，是 SA 方法中用于表示系统逻辑模型的一种工具。它以图形的方式描绘数据在系统中流动和处理的过程，由于它只反映系统必须完成的逻辑功能，所以它是一种功能模型。

在数据流程图中一般包括四种基本的图形符号，见下表。

图形符号的名称	代表的意义
圆或椭圆	变换 / 加工
方框	数据源点或终点
箭头	数据流
双杠	数据存储

数据流图由数据流、加工(又称数据处理)、数据存储(又称文件)、数据源点或终点四种基本成分组成。

(1) 数据流。数据流是数据在系统内传播的路径。

(2) 加工(又称数据处理)。对数据流进行某些操作或变换。

(3) 数据存储(又称文件)。指暂时保存的数据，它可以是数据库文件或任何形式的数据组织。

(4) 数据源点或终点。是本软件系统外部环境中的实体(包括人员、组织或其他软件系统)，统称为外部实体。

画数据流图时应注意的事项：

(1) 命名。不论数据流、数据存储还是加工，合适的命名使人们易于理解其含义。

(2) 画数据流而不是控制流。

(3) 一般不画物质流。

(4) 每个加工至少有一个输入数据流和一个输出数据流，反映出此加工数据的来源与加工的结果。

(5) 编号。如果一张数据流图中的某个加工分解成另一张数据流图时，则上层图为父图，直接下层图为子图。子图应有编号，子图上的所有加工也应编号，子图的编号就是父图中相应加工的编号，加工的编号是由子图号、小数点及局部号组成。

(6) 父图与子图的平衡。子图的输入输出数据流同父图相应加工的输入输出数据流必须一致。

(7) 局部数据存储。当某些数据流图中的数据存储不是父图中相应加工的外部接口，而只是本图中某些加工之间的数据接口，则称这些数据存储为局部数据存储。一个局部数据存储只要当它作为某些加工的数据接口或某个加工特定的输入或输出时，就把它画出来，这样有助于实现信息隐藏。

(8) 提高数据流图的易理解性。

4. 数据字典(DD)

数据字典 (Data Dictionary, 简称 DD) 就是用来定义数据流图中的各个成分的具体含义的, 它以一种准确的、无二义性的说明方式为系统的分析、设计及维护提供了有关元素的一致性的定义和详细的描述。它和数据流图共同构成了系统的逻辑模型, 是需求规格说明书的主要组成部分。

数据字典是为分析人员查找数据流图中有关名字的详细定义而服务的。因此也像普通字典一样, 要把所有条目按一定的次序排列起来, 以便查阅。数据字典有四类条目: 数据流、数据项、数据存储和基本加工。

通常在数据字典的定义中可能出现的符号及其含义是 (设 x 和 a 、 b 都是数据元素):

$x=a+b$	表示 x 是由 a 和 b 构成
$x=[alb]$	表示 x 是由 a 或 b 构成
$x=\{a\}$	表示 x 是由 0 个或多个 a 组成
$x=m\{a\}n$	表示 x 中最少出现 m 次 a , 最多出现 n 次 a 。
$x=(a)$	表示 a 可在 x 中出现, 也可不出现
$x="a"$	表示 x 是取值为字符 a 的数据元素
$x=1..9$	表示 x 可取 1 到 9 中任意一个值

5. 加工逻辑的描述

加工逻辑也称为“小说明”, 描述加工逻辑一般用以下三种工具: 结构化语言、判定表、判定树。

(1) 结构化语言。

结构化语言是介于自然语言和形式语言之间的一种半形式语言。结构化语言是在自然语言的基础上加上一些限制, 使用有限的词汇和有限的语句来描述加工逻辑, 它的结构可分成外层和内层两层:

1) 外层: 用来描述控制结构, 采用顺序、选择、重复三种基本结构。

2) 内层: 一般采用祈使语句的自然语言短语, 使用数据字典中的名词和有限的自定义词, 其动词含义要具体, 尽量不用形容词和副词来修饰。还可使用一些简单的算术运算和逻辑运算符号。

(2) 判定表。

判定表能够清楚地表示复杂条件组合与应做的动作之间的对应关系。它能够把在什么条件下系统应做什么动作准确无误地表示出来。

判定表由四部分组成, 用双线分割开四个区域, 如下图所示, 各部分的含义在图上标出。

条件定义	条件取值的组合
动作定义	在各种取值的组合下应执行的动作

(3) 判定树。

判定树是判断表的变形, 一般情况下, 比判断表更直观, 且易于理解和使用。用它来描述具有多个条件的数据加工更容易被用户接受。

下面举例说明判定树的构成, 如下图。