

21 世纪 高 职 高 专 规 划 教 材

计 算 机 专 业 基 础 系 列

21

# 数据结构实用教程

孙 湧 编 著

清华大学出版社



21 世纪 高 职 高 专 规 划 教 材

计 算 机 专 业 基 础 系 列

# 数据结构实用教程

孙 湧 编 著



清华大学出版社  
北京

## 内 容 简 介

本书是根据数据结构课程教学大纲的要求,结合作者多年实践经验积累而完成的具有工程实践价值的数据结构教材。全书共分8章,每章均先给出本章的教学重点和难点,明确理论和技能要求及教学方法,以方便教和学。其中,第1章说明开设数据结构课程的意义;第2章详细介绍了顺序表和链表结构及其实现,这是其他数据结构的两种实现基础;第3章采用顺序表和链表分别讲解最基本的线性数据结构——堆栈、队列和串;第4章和第5章介绍非线性数据结构——树和图。第6章提出了一种有工程应用价值的递归算法实现方法;第7章和第8章分别描述了多种查找和排序算法及其实现。本书配有电子课件,包括全部程序源代码。

本书适合大专层次的大类型学历教育,包括普通、成人和高职高专院校计算机相关专业的学生,也适合企业专业软件开发人员使用。

版权所有,翻印必究。举报电话:010-62782989 13501256678 13801310933

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。

本书防伪标签采用特殊防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

### 图书在版编目(CIP)数据

数据结构实用教程/孙湧编著. —北京:清华大学出版社,2006.1

(21世纪高职高专规划教材.计算机专业基础系列)

ISBN 7-302-12136-2

I. 数… II. 孙… III. 数据结构—高等学校:技术学校—教材 IV. TP311.12

中国版本图书馆CIP数据核字(2005)第136275号

出 版 者:清华大学出版社

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

社 总 机:010-62770175

地 址:北京清华大学学研大厦

邮 编:100084

客 户 服 务:010-62776969

责任编辑:曾 妍

印 刷 者:北京嘉实印刷有限公司

装 订 者:三河市化甲屯小学装订二厂

发 行 者:新华书店总店北京发行所

开 本:185×230 印 张:15.75 字 数:321千字

版 次:2006年1月第1版 2006年1月第1次印刷

书 号:ISBN 7-302-12136-2/TP·7835

印 数:1~5000

定 价:22.00元

# 出版说明

高职高专教育是我国高等教育的重要组成部分,担负着为国家培养并输送生产、建设、管理、服务第一线高素质技术应用型人才的重任。

进入 21 世纪后,高职高专教育的改革和发展呈现出前所未有的发展势头,学生规模已占我国高等教育的半壁江山,成为我国高等教育的一支重要的生力军;办学理念上,“以就业为导向”成为高等职业教育改革与发展的主旋律。近两年来,教育部召开了三次产学研交流会,并启动四个专业的“国家技能型紧缺人才培养项目”,同时成立了 35 所示范性软件职业技术学院,进行两年制教学改革试点。这些举措都表明国家正在推动高职高专教育进行深层次的重大改革,向培养生产、服务第一线真正需要的应用型人才的方向发展。

为了顺应当前我国高职高专教育的发展形势,配合高职高专院校的教学改革和教材建设,进一步提高我国高职高专教育教材质量,在教育部的指导下,清华大学出版社组织出版“21 世纪高职高专规划教材”。

为推动规划教材的建设,清华大学出版社组织并成立“高职高专教育教材编审委员会”,旨在对清华版的全国性高职高专教材及教材选题进行评审,并向清华大学出版社推荐各院校办学特色鲜明、内容质量优秀的教材选题。教材选题由个人或各院校推荐,经编审委员会认真评审,最后由清华大学出版社出版。编审委员会的成员皆来源于教改成效大、办学特色鲜明、师资实力强的高职高专院校、普通高校以及著名企业,教材的编写者和审定者都是从事高职高专教育第一线的骨干教师和专家。

编审委员会根据教育部最新文件政策,规划教材体系,比如部分专业的两年制教材;“以就业为导向”,以“专业技能体系”为主,突出人才培养的实践性、应用性的原则,重新组织系列课程的教材结构,整合课程体系;按照教育部制定的“高职高专教育基础课程教学基本要求”,教材的基础理论以“必要、够用”为度,突出基础理论的应用和实践技能的培养。

本套规划教材的编写原则如下:

- (1) 根据岗位群设置教材系列,并成立系列教材编审委员会;
- (2) 由编审委员会规划教材、评审教材;
- (3) 重点课程进行立体化建设,突出案例式教学体系,加强实训教材的出版,完善教学服务体系;
- (4) 教材编写者由具有丰富教学经验和多年实践经验的教师共同组成,建立“双师

型”编者体系。

本套规划教材涵盖了公共基础课、计算机、电子信息、机械、经济管理以及服务等大类的主要课程,包括专业基础课和专业主干课。目前已经规划的教材系列名称如下:

#### • 公共基础课

公共基础课系列

#### • 计算机类

计算机基础教育系列

计算机专业基础系列

计算机应用系列

网络专业系列

软件专业系列

电子商务专业系列

#### • 电子信息类

电子信息基础系列

微电子技术系列

通信技术系列

电气、自动化、应用电子技术系列

#### • 机械类

机械基础系列

机械设计与制造专业系列

数控技术系列

模具设计与制造系列

#### • 经济管理类

经济管理基础系列

市场营销系列

财务会计系列

企业管理系列

物流管理系列

财政金融系列

#### • 服务类

旅游系列

艺术设计系列

本套规划教材的系列名称根据学科基础和岗位群方向设置,为各高职高专院校提供“自助餐”形式的教材。各院校在选择课程需要的教材时,专业课程可以根据岗位群选择系列;专业基础课程可以根据学科方向选择各类的基础课系列。例如,数控技术方向的专业课程可以在“数控技术系列”选择;数控技术专业需要的基础课程,属于计算机类课程的可以在“计算机基础教育系列”和“计算机应用系列”选择,属于机械类课程的可以在“机械基础系列”选择,属于电子信息类课程的可以在“电子信息基础系列”选择。依此类推。

为方便教师授课和学生学习,清华大学出版社正在建设本套教材的教学服务体系。本套教材先期选择重点课程和专业主干课程,进行立体化教材建设:加强多媒体教学课件或电子教案、素材库、学习盘、学习指导书等形式的制作和出版,开发网络课程。学校在选用教材时,可通过邮件或电话与我们联系获取相关服务,并通过与各院校的密切交流,使其日臻完善。

高职高专教育正处于新一轮改革时期,从专业设置、课程体系建设到教材编写,依然是新课题。希望各高职高专院校在教学实践中积极提出意见和建议,并向我们推荐优秀选题。反馈意见请发送到 E-mail:gzgz@tup.tsinghua.edu.cn。清华大学出版社将对已出版的教材不断地修订、完善,提高教材质量,完善教材服务体系,为我国的高职高专教育出版优秀的高质量的教材。

高职高专教育教材编审委员会

# 前言

## 数据结构实用教程

如果希望成为一名优秀的程序设计师,不学或没学好数据结构是不可想像的事情。也许有人会认为,在可视化程序设计普及的今天,只要掌握好几种软件开发工具,就能很快借助于软件集成开发环境成为软件编程高手,这绝对是一种误解。

一个优秀的专业软件开发人员至少需要具备以下 3 项能力:

(1) 至少熟练掌握一门程序设计语言,能够具体从事软件系统开发;

(2) 熟悉系统最终应用领域的相关背景知识,能够充分、合理地描述、归纳客观事物需求;

(3) 能够熟练选择和设计各种数据结构和算法,从而有效地描述、组织和操作数据。

其中,前两个能力比较容易具备,而第三个能力需要花费相当的时间和精力才能够逐步具备,它是判定程序设计人员水平高低的一个重要的标志性数据。

程序设计不仅涉及计算机编程语言,如何有效描述、组织和操作数据也是编程人员所关心和了解的。描述和组织数据是数据结构设计,操作数据则是算法设计,数据结构加上算法构成程序设计的整个过程,这是《数据结构》课程所要讲授的内容。如果数据结构和算法功底不深,将很难设计出专业水准的应用程序。《数据结构》在软件编程能力提高过程中,起着承前启后的作用,并成为计算机软件等相关专业的最重要的专业基础课程。

《数据结构》是一门理论性极强的课程,高职高专院校学生学习《数据结构》课程,并非进行数据结构和算法的理论研究,而是要能灵活运用数据结构和算法知识去解决实际问题,因此学生要有良好的 C 语言基础,C 语言是帮助读者理解和使用数据结构的最有效工具。本教材在所附电子课件中提供了一个基于 Windows 环境的 C 语言编程环境 Win-TC 1.8,供读者选用,可从网站([www.tup.tsinghua.edu.cn/sub-press/6](http://www.tup.tsinghua.edu.cn/sub-press/6))获得。

目前一些高职高专院校所用的《数据结构》教材只是本科教材的压缩饼干,习惯于就事论事的算法简单实现,与企业实际问题解决模式还有较大距离。学生无法有效建立面

向过程的编程思维方式,模块编写缺乏合理的编程风格和规范,没能考虑程序的健壮性和模块的可重用性,数据处理一旦出错就会导致系统崩溃;习惯在主函数中编写程序,没有重用已有功能函数的意识,喜欢一切从头开始,并因此经常发生同一种错误,导致程序质量和编程效率较差。

针对以上情况,本书希望能够帮助读者解决以下几类问题。

(1) 在学习《数据结构》课程的同时,强化 C 语言实战特征,强化 C 语言编程能力。

(2) 掌握合理的模块代码编写风格和规范,确保代码易读和易交流,以便为开发团队所认可。

(3) 建立模块可重用性概念,一个功能一个函数,函数封装可以大大减少不同因素间的相互影响和耦合度,降低问题复杂度。重用已有函数还可避免低水平重复和编程低效率的弊端。就像调用 printf 函数,因为无需了解其代码实现,从而可有效提升程序代码编写质量和编写效率。

(4) 建立模块合法性检查意识,有效提升系统健壮性。

(5) 建立数据结构的逻辑描述与物理实现无关意识,了解不同数据结构间的共性和个性,将千变万化的物理模型抽象成单一的逻辑模型,便于借鉴前人的知识积累,提高工作效率。

本书遵循课程教学大纲,以满足企业从业需求为依托进行教学内容的组织。

全书分成 8 章,第 1 章是数据结构概论,着重讲解数据结构课程的重要性;第 2 章介绍线性表,着重分析、描述其他数据结构的构造基础——顺序表和链表;第 3 章介绍堆栈、队列和串,是第 2 章线性表的内涵引申和应用特例;第 4 章全面分析了二叉排序树和 Huffman 树的模块实现,以及二叉树各种遍历的输出序列和工程化代码实现;第 5 章详细介绍了图的基本概念、图的最小生成树、AOV 网和 AOE 网;第 6 章在介绍传统递归实现算法的同时,还详细介绍了工程性实用递归算法实现;第 7 章在查找多种结点方案的基础上,提出应用实例;第 8 章介绍了最常用的排序方法,以使学生会精益求精,追求卓越。

尽管不同数据结构的数据操作存在差异性,但这些数据结构在结点插入、修改、查询、删除等方面的操作思路却是一样的,只要理解一个,即可一通百通。对学生的最低要求是会调用,最高要求是知其然还要知其所以然,真正掌握《数据结构》的知识要点。任课教师可根据实际教学安排裁减教学内容,培养学生的自学能力。

本书由孙湧主编,并负责全书的编程调试,王梅参与了第 3 章、第 5 章、第 7 章的编写工作,并提供了各章习题。在编写过程中,作者认真听取了深圳职业技术学院计算机应用



工程系众多教师提出的宝贵意见,特别是钟剑龙老师对数据结构分层描述设计提出了独到的见解,在此一并表示衷心感谢。

本书是作者多年工作经验的总结,将有利于培养企业所需的软件人才。但本教材仍会有不少缺点错误,欢迎大家批评指正。联系方式:深圳职业技术学院计算机应用工程系孙湧。E-mail: sunyong@oa.szpt.net。

作 者

2005年11月



# 目 录

数据结构实用教程

<b>第 1 章 数据结构概论</b> .....	1
1.1 数据结构与软件从业人员的未来发展 .....	2
1.1.1 软件的可重用模型.....	2
1.1.2 数据结构在软件项目开发中的关键作用.....	2
1.1.3 数据结构与岗位分工.....	3
1.1.4 数据结构的重要地位与课程学习方法.....	4
1.2 数据结构综述 .....	6
1.2.1 数据结构基础.....	6
1.2.2 常用数据结构说明.....	7
1.2.3 数据结构的实现基础.....	9
1.3 算法综述.....	10
1.3.1 算法初步 .....	10
1.3.2 算法的构造 .....	11
1.3.3 算法复杂度 .....	13
1.3.4 基于递归的算法设计思想 .....	14
1.4 数据结构与算法存在互为因果的辩证关系.....	15
习题 .....	16
<b>第 2 章 线性表</b> .....	18
2.1 线性表的概念及其基本运算.....	19
2.1.1 线性表的逻辑描述 .....	19
2.1.2 线性表的基本运算 .....	19
2.2 顺序表——线性表的顺序存储方式.....	20
2.2.1 顺序存储结构 .....	20

2.2.2	顺序表基本操作的算法实现 .....	23
2.2.3	顺序表性能小结 .....	33
2.3	链表——线性表的链接存储方式 .....	33
2.3.1	单向链表 .....	33
2.3.2	单向链表的基本操作 .....	36
2.3.3	单向循环链表 .....	51
2.3.4	双向链表和双向循环链表 .....	52
2.3.5	链表性能小结 .....	55
2.4	二维数组的数据压缩处理 .....	55
2.4.1	有规律二维数组的一维化映射 .....	55
2.4.2	无规律稀疏矩阵的数据压缩处理 .....	57
	习题 .....	58
<b>第3章</b>	<b>堆栈、队列和串 .....</b>	<b>62</b>
3.1	堆栈 .....	63
3.1.1	堆栈的概念及其操作 .....	63
3.1.2	堆栈的输出序列分析 .....	64
3.1.3	基于顺序表的堆栈实现 .....	64
3.1.4	基于链表的堆栈实现 .....	65
3.1.5	堆栈应用 .....	66
3.2	队列 .....	68
3.2.1	队列的概念及其操作 .....	68
3.2.2	循环队列 .....	68
3.2.3	基于顺序表的队列实现 .....	69
3.2.4	基于链表的队列实现 .....	70
3.2.5	队列应用 .....	71
3.3	串 .....	72
3.3.1	串的概念及其操作 .....	73
3.3.2	串存储结构描述 .....	74
3.3.3	文本编辑 .....	75
	习题 .....	76
<b>第4章</b>	<b>树与二叉树 .....</b>	<b>78</b>
4.1	树与森林 .....	79

4.1.1	基本概念与属性 .....	79
4.1.2	树的存储结构 .....	81
4.2	二叉树 .....	82
4.2.1	基本概念 .....	82
4.2.2	二叉树的存储结构 .....	84
4.3	二叉树遍历 .....	85
4.3.1	二叉树的深度优先遍历 .....	86
4.3.2	二叉树的宽度优先遍历 .....	89
4.3.3	二叉树遍历的工程化实现算法 .....	91
4.4	树与森林的基本操作 .....	92
4.4.1	树与二叉树的相互转换 .....	92
4.4.2	森林与二叉树的相互转换 .....	93
4.4.3	树的深度优先遍历 .....	94
4.4.4	树的宽度优先遍历 .....	94
4.5	二叉树应用之一——二叉排序树 .....	95
4.5.1	二叉排序树的定义和遍历特征 .....	95
4.5.2	二叉排序树基本操作的功能实现 .....	95
4.6	二叉树应用之二——Huffman 树 .....	106
4.6.1	Huffman 树的定义和特征 .....	106
4.6.2	生成 Huffman 树的算法思路 .....	107
4.6.3	Huffman 树基本操作的功能实现 .....	109
习题	.....	116
<b>第 5 章</b>	<b>图 .....</b>	<b>118</b>
5.1	基本概念 .....	118
5.1.1	无向图 .....	118
5.1.2	有向图 .....	120
5.2	图的存储结构 .....	121
5.2.1	邻接矩阵表示法 .....	121
5.2.2	邻接表表示法 .....	122
5.3	图的遍历 .....	123
5.3.1	图的深度优先遍历 .....	123
5.3.2	图的宽度优先遍历 .....	124
5.4	生成树和最小生成树 .....	125

5.4.1	生成树	126
5.4.2	最小生成树	126
5.4.3	基于知识的最小生成树	128
5.5	拓扑排序	132
5.6	关键路径法	135
5.7	最短路径	139
5.7.1	单点出发的最短路径问题	139
5.7.2	多点出发的最短路径问题	140
	习题	145
<b>第 6 章</b>	<b>基于树的工程性实用递归算法</b>	<b>147</b>
6.1	算法的递归和非递归实现的性能分析	147
6.1.1	算法的递归实现方法性能分析	147
6.1.2	算法的非递归实现方法性能分析	150
6.1.3	算法的递归实现与非递归实现的比较	151
6.2	工程性实用递归算法解决方案	152
6.2.1	算法描述	152
6.2.2	基于树的工程性实用递归算法实现	152
6.3	新算法应用举例	155
6.3.1	fibonacci 数列	155
6.3.2	汉诺塔	157
6.3.3	N 皇后问题	160
	习题	165
<b>第 7 章</b>	<b>查找</b>	<b>167</b>
7.1	基本概念和意义	167
7.2	线性表查找	168
7.2.1	线性表的顺序查找	168
7.2.2	顺序表的折半查找	170
7.2.3	顺序表的分块查找	172
7.2.4	线性表的 Hash 散列查找	173
7.3	基于树的结点查找	176
7.3.1	二叉排序树	176
7.3.2	平衡二叉树	177

7.3.3 一般二叉树的结点查找	182
习题	183
<b>第8章 排序</b>	<b>185</b>
8.1 基本概念	185
8.2 插入排序	186
8.2.1 直接插入排序	186
8.2.2 Shell 排序	188
8.3 交换排序	190
8.3.1 冒泡排序	190
8.3.2 快速排序	191
8.4 选择排序	195
8.4.1 直接选择排序	195
8.4.2 二次选择排序	197
8.4.3 堆排序	199
8.5 其他归类排序方法	202
8.5.1 二路归并排序	202
8.5.2 二叉排序树排序	205
8.5.3 基数排序	205
8.6 排序小结	207
习题	208
<b>附录 A 实训项目</b>	<b>210</b>
<b>附录 B 基于数组的函数原型定义和功能说明 array.hc</b>	<b>217</b>
<b>附录 C 基于链表的函数原型定义和功能说明 chain.hc</b>	<b>222</b>
<b>附录 D 基于链表的 Huffman 树函数原型定义和功能说明 Huffer.hc</b>	<b>230</b>
<b>附录 E 教材电子课件所含文件清单及其运行环境说明</b>	<b>234</b>
<b>参考文献</b>	<b>237</b>

# 数据结构概论

重点	了解计算机课程体系中数据结构的重要性及其课程学习内容,明确学习目标,增强学习兴趣
难点	明白算法复杂度的分析方法,初步理解本章提及的概念
理论要点	<ul style="list-style-type: none"> <li>(1) 了解瀑布模型和模块可重用模型</li> <li>(2) 理解并掌握数据结构和算法的概念和分类,能够正确区分线性表和非线性表</li> <li>(3) 掌握算法描述及空间、时间复杂度分析</li> <li>(4) 理解数据结构和算法间的辩证关系</li> </ul>
技能要求	<ul style="list-style-type: none"> <li>(1) 能按照可重用模块编程思想进行实践</li> <li>(2) 掌握数据的组织方法,即能简单介绍堆栈、队列、串、树、图等传统的数据结构,以及这些数据结构基于线性表、链表和 STL 规范的基本实现方法</li> <li>(3) 掌握算法复杂度的分析方法</li> </ul>
教学方法	<ul style="list-style-type: none"> <li>(1) 以故事形式简介软件发展史,让读者明白数据结构出现的必然,对于提升个人角色重要性、改善产品开发能力具有重要意义</li> <li>(2) 比照分析,让读者了解本课程讲解的各类数据结构的宏观特征及属性的异同性</li> </ul>

随着软硬件的迅猛发展,计算机已逐渐应用于工业、农业、商业、军事、文教卫生、家庭等社会的各个领域,所开发的应用系统层出不穷,正在有效地改变着人们的日常生活和工作方式。计算机不再只是用于早期的简单海量和快速数值计算,现已逐步延伸至图像、声音等复杂的非数值性数据处理和维护,同时,待处理信息量也越来越大,需要形成具有特定关系的复杂的数据形式,以方便存储和处理。

《数据结构》(Data Structure)就是这样一门随着计算机科学的发展而逐渐形成的学科,其核心和实质就是如何有效组织数据和如何高效处理数据。

## 1.1 数据结构与软件从业人员的未来发展

本节介绍两个软件工程基本模型,是为了多角度阐述数据结构在软件项目开发中的独特地位和重要作用,以使读者掌握正确的工程化编程风格,从而明白代码编写只是软件开发过程中的一个中间环节而非全部,代码编写工作量只占软件开发总量的10%~20%,如果只会代码编写将缺乏发展后劲和上升空间。

### 1.1.1 软件的可重用模型

#### 1. 思想简介

现代软件项目开发一般采用模块化设计方法,即每个功能设计成一个模块,尽量减少系统不同功能代码间的相互影响。这种系统设计思想是构造软件重用库的基础。在模块设计时要充分考虑界面的数据接口规格,并以编译前源代码和编译后目标程序两种形式存放于库中,以便软件开发时能够重复使用。软件重用库中的模块不仅要便于选择使用,还应允许系统开发者不断扩充和修改。

#### 2. 意义

C是基于函数的结构化程序语言,只要按照规范将功能模块开发成自定义函数,就能反复调用。而不必关心该函数的实现方法。调用软件重用模块可以大大减少软件开发中存在的低级重复性编程工作,以及由此引发的语法、逻辑错误,从而可有效提高系统的开发效率,缩短项目开发周期,降低开发成本。

那种在main函数中书写所有程序代码的编程风格是错误的,学生在校期间就要养成符合企业规范的良好工作习惯和工作方式。本教材是基于模块的可重用思路来组织教学内容的,读者在学习过程中要学会修改和调用相关模块。

### 1.1.2 数据结构在软件项目开发中的关键作用

如同人一般要经历出生、儿童、青年、中年、老年到死亡的过程一样,每个软件

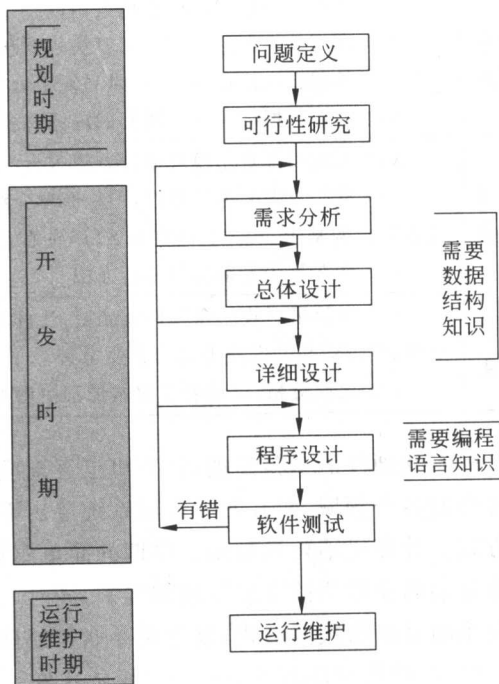


图 1-1 适用于结构化开发技术的典型瀑布模型



项目也同样存在定义、开发、使用、维护到停用的过程,这一过程可简单归纳为软件规划、开发和运行维护三个阶段,称为软件生命周期(Software Life Cycle)。

软件生命周期的各阶段工作恰如奔流不息、拾级而下的瀑布,如图 1-1 所示。

### 1. 软件规划时期

在这一时期所做的工作是:明确待开发系统所要解决的实际问题,通过可行性研究来评价系统的建设目标和规模的科学性及效益回报,最终确定系统是否值得开发。

### 2. 软件开发时期

在需求分析、总体设计、详细设计、程序设计、软件测试五个阶段中,前三个阶段属于系统逻辑设计,与具体编程语言无关,需要有丰富的行业背景经验并精通数据结构、数据库原理等方面的知识;程序设计是系统编码,依赖于具体的编程语言。系统逻辑设计越充分,编码过程就越轻松,也越容易得到用户的认可。

千奇百怪、缺乏共性的客观物理世界需要归纳成抽象的数学模型,才能让计算机处理。这个过程其实就是思考如何组织数据和操作数据,这些都是《数据结构》这门课程的学习内容,学好数据结构和数据库原理将有助于快速提升软件项目开发的水平。

### 3. 软件运行维护时期

软件交付使用后,需要及时修改错误,以维护系统的正常运行。

## 1.1.3 数据结构与岗位分工

人尽其才和各尽所能是目前社会化分工大协作下的工作原则,软件项目开发的人员构成也符合金字塔模式,系统分析员、高级程序员、程序员和初级程序员等四个软件人才层次与角色分工如图 1-2 所示。

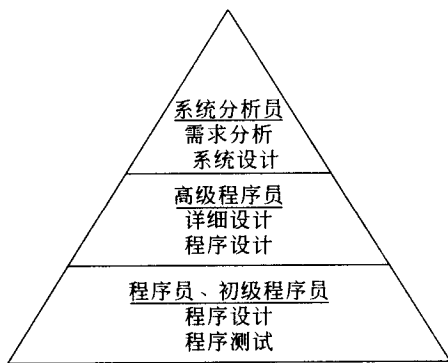


图 1-2 软件人才金字塔与分工

### 1. 软件人才金字塔

#### (1) 系统分析员

系统分析员能全面理解公司具体的业务管理流程,并充分利用丰富的行业背景知识和项目经验积累,通过用户需求分析和系统主体设计,完成公司业务流程再造,全面提升用户的管理水平和档次。他们处于软件人才金字塔的塔尖,是竞争公司和猎头公司的主要关注对象。

要成为优秀的系统分析员就要先成为相关应用领域的业务流程专家,并在精通某个领域业务的同时,尽可能积累其他领域的行业背景

知识。精通《数据结构》和《数据库原理》这两门课程是成为并发挥系统分析员作用的重要



基础。

### (2) 高级程序员

高级程序员作为公司的技术骨干,从事模块详细设计和程序代码设计,处于软件人才金字塔的中部,是竞争公司和猎头公司的关注目标和成为系统分析员的中间阶段。

要成为优秀的高级程序员,首先不能满足于按照详细设计的要求编写程序代码,要能利用数据结构知识,设计系统功能模块的有效实现算法,确保模块的数据流程和数据操作尽可能高效运转。其次,还要坚持积累不同领域的行业背景知识,努力成为相关应用领域的业务流程专家,有意识地向系统分析员的角色转变。认真钻研《数据结构》和《数据库原理》课程将是一个行之有效的手段。

### (3) 程序员和初级程序员

程序员和初级程序员的界线并不明显,他们主要从事模块编程和程序测试,决定着软件项目的质量和生命力,处于软件人才金字塔的塔基,企业需求量较大,由刚出道技术人员构成。

程序员绝对不能满足于按照详细设计要求编写模块代码,要有意识地强化数据结构知识,逐步提高系统功能模块的数据结构和算法设计能力,以求编写出高质、高效的模块代码。

## 2. 社会化分工大协作

在西方发达国家和软件大国印度,软件开发的专业分工较为明显,如果要求不高,只按照详细设计的要求完成程序模块代码编写工作是可行的。

在中国,软件企业正在逐步从手工作坊向现代化大生产模式转变,绝对的社会化分工体系远未建立,如果只能按照详细设计的要求编写程序模块代码,是满足不了软件企业的要求的,所以读者要高度重视《数据结构》课程的学习,以增强发展后劲。

## 1.1.4 数据结构的重要地位与课程学习方法

### 1. 数据结构课程的地位与作用

计算机需要结合系统程序和应用程序才能真正有效地发挥作用,计算机能否高效运转,与程序设计所采用的数据结构及算法有密切关系,而好算法又在很大程度上取决于描述实际问题的数据结构。《数据结构》是一门兼顾数学、计算机硬件和计算机软件相关知识的综合性专业核心课程,数据结构和算法已成为计算机程序设计技术的两大支柱,即程序=算法+数据结构,如图 1-3 所示。

学好数据结构课程将有助于提升从业人员的软件系统设计和编程能力,这种能力对从业人员的未来发展具有深远影响。

### 2. 数据结构的学习方法

《数据结构》是理论性和实践性都很强的课程,其中的许多概念十分抽象,需要有认真