



21st CENTURY
规划教材

面向21世纪高职高专计算机系列规划教材
COURSES FOR VOCATIONAL HIGHER EDUCATION, COMPUTER

数据结构

——C语言实现

张文明 主编



科学出版社
www.sciencep.com

NEW in 2006



面向21世纪高职高专计算机系列规划教材
COURSES FOR VOCATIONAL HIGHER EDUCATION: COMPUTER

数 据 结 构

——C 语言实现

张文明 主编

张海防 张军利 黄志良 副主编

科学出版社

北 京

内 容 简 介

本书简明扼要地介绍了各种常用的数据结构。主要内容包括：数据结构的基本概念和运算，算法分析初步；线性表、堆栈、队列、串、数组、树、图、邻接表、邻接矩阵、文件等基本数据结构和运算；查找、排序、图的简单算法等。全书采用目前广泛流行的 Turbo C 作为算法的描述语言。

本书注重概念叙述，在每章介绍完一种数据结构及基本运算后，都给出了相关的应用实例并在章后配有习题。为了加强实践教学，在第 11 章对线性表、链表、树、图等数据结构，排序、查找等算法设计都给出了相关的实验与上机指导，以提高学生编写和调试程序的能力。

本书可作为高职高专院校计算机应用、计算机信息管理、计算机通信等相关专业的教材，也可供从事计算机应用等工作的科技工作者参考使用。

图书在版编目 (CIP) 数据

数据结构: C 语言实现/张文明主编. —北京: 科学出版社, 2006

(面向 21 世纪高职高专计算机系列规划教材)

ISBN 7-03-016659-0

I. 数… II. 张… III. ①数据结构-高等学校: 技术学校-教材 ②C 语言-程序设计-高等学校: 技术学校-教材 IV. ①TP311.12 ②TP312

中国版本图书馆 CIP 数据核字 (2005) 第 154530 号

责任编辑: 万国清 孙露露/责任校对: 刘彦妮

责任印制: 吕春珉/封面设计: 飞天创意

科学出版社 出版

北京东黄城根北街 16 号

邮政编码: 100717

<http://www.sciencep.com>

铭浩彩色印装有限公司 印刷

科学出版社发行 各地新华书店经销

*

2006 年 2 月第 一 版 开本: 787×1092 1/16

2006 年 2 月第一次印刷 印张: 15 1/2

印数: 1—3 000 字数: 345

定价: 21.00 元

(如有印装质量问题, 我社负责调换<环伟>)

销售部电话 010-62136131 编辑部电话 010-62138978-8004 (V102)

面向 21 世纪高职高专规划教材专家委员会

主 任 李宗尧

副主任 (按姓氏笔画排序)

丁桂芝 叶小明 张和平 林 鹏

黄 藤 谢培苏

委 员 (略)

信息技术系列教材编委会

主 任 丁桂芝

副主任 (按姓氏笔画排序)

万金保 方风波 徐 红 鲍 泓

委 员 (按姓氏笔画排序)

于晓平	马国光	仁英才	王东红	王正洪
王 玉	王兴宝	王金库	王海春	王爱梅
邓 凯	付百文	史宝会	本柏忠	田 原
申 勇	任益夫	刘成章	刘克敏	刘甫迎
刘经玮	刘海军	刘敏涵	安志远	许殿生
何瑞麟	余少华	吴春英	吴家碚	吴瑞萍
宋士银	宋锦河	张红斌	张环中	张海鹏
张蒲生	张德实	李云程	李文森	李 洛
李德家	杨永生	杨 闯	杨得新	肖石明
肖洪生	陈 愚	周子亮	周云静	胡秀琴
赵从军	赵长旭	赵动庆	郝 梅	唐铸文
徐洪祥	徐晓明	袁德明	郭庚麒	高延武
高爱国	康桂花	戚长政	曹文济	黄小鸥
彭丽英	董振珂	蒋金丹	韩银峰	魏雪英

出版前言

随着世界经济的发展,人们越来越深刻地认识到经济发展需要的人才多元化、多层次的,既需要大批优秀的理论型、研究型的人才,也需要大批应用型人才。然而,我国传统的教育模式主要是培养理论型、研究型的人才。教育界在社会对应用型人才需求的推动下,专门研究了国外应用型人才教育的成功经验,结合国情大力度地改革我国的“高等职业教育”,制定了一系列的方针政策。联合国教科文组织1997年公布的教育分类中将这种教育称之为“高等技术与职业教育”,也就是我们通常所说的“高职高专”教育。

我国经济建设需要大批应用型人才,呼唤高职高专教育的崛起和成熟,寄希望于高职高专教育尽快向国家输送高质量的紧缺人才。近几年,高职高专教育发展迅速。目前,各类高职高专学校已占全国高等院校的近1/2,约有600所之多。教育部针对高职高专教育出台的一系列政策和改革方案主要体现在以下几个方面:

- “就业导向”成为高职高专教育的共识。高职高专院校在办学过程中充分考虑市场需求,用“就业导向”的思想制定招生和培养计划。
- 加快“双师型”教师队伍建设。已建立12个国家高职高专学生和教师的实训基地。
- 对学生实行“双认证”教育。学历文凭和职业资格“双认证”教育是高职高专教育特色之一。
- 高职高专教育以两年学制为主。从学制入手,加快高职高专教学方向的改革,充分办出高职高专教育特色,尽快完成紧缺人才的培养。
- 开展精品专业和精品教材建设。已建立科学的高职高专教育评估体系和评估专家队伍,指导、敦促不同层次、不同类型的学校办出一流的教育。

在教育部关于“高职高专”教育思想和方针指导下,科学出版社积极参与到高职高专教材的建设中去,在组织教材过程中采取了“请进来,走出去”的工作方法,即由教育界的专家、领导和一线的教师,以及企事业单位从事人力资源工作的人员组成顾问班子,充分分析我国各地区的经济发展、产业结构以及人才需求现状,研究培养国家紧缺人才的关键要素,寻求切实可行的教学方法、手段和途径。

通过研讨认识到,我国幅员辽阔,各地区的产业结构有明显的差异,经济发展也不平衡,各地区对人才的实际需求也有所不同。相应地,对相同专业和相近专业,不同地区的教学单位在培养目标和培养内容上也各有自己的定位。鉴于此,适应教育现状的教材建设应该具有多层次的设计。

为了使教材的编写能针对受教育者的培养目标,出版社的编辑分不同地区逐所学校拜访校长、系主任和老师,深入到高职高专学校及相关企事业单位,广泛、深入地教学和第

一线的老师、用人单位交流,掌握了不同地区、不同类型的高职高专院校的师生、学生和教学设施情况,清楚了各学校所设专业的培养目标和办学特点,明确了用人单位的需求条件。各区域编辑对采集的数据进行统计分析,在相互交流的基础上找出各地区、各学校之间的共性和个性,有的放矢地制定选题项目,并进一步向老师、教育管理者征询意见,在获得明确指导性意见后完成“高职高专规划教材”策划及教材的组织工作:

- 第一批“高职高专规划教材”包括三个学科大系:经济管理、信息技术、建筑。
- 第一批“高职高专规划教材”在注意学科建设完整性的同时,十分关注具有区域人才培养特色的教材。
- 第一批“高职高专规划教材”组织过程正值高职高专学制从3年制向2年制转轨,教材编写将其作为考虑因素,要求提示不同学制的讲授内容。
- 第一批“高职高专规划教材”编写强调
 - ◆ 以就业岗位对知识和技能需求下的教材体系的系统性、科学性和实用性。
 - ◆ 教材以实例为先,应用为目的,围绕应用讲理论,取舍适度,不追求理论的完整性。
 - ◆ 提出问题→解决问题→归纳问题的教、学法,培养学生触类旁通的实际工作能力。
 - ◆ 课后作业和练习(或实训)真正具有培养学生实践能力的作用。

在“高职高专规划教材”编委的总体指导下,第一批各科教材基本是由系主任或从教学一线中遴选的骨干教师执笔撰写。在每本书主编的严格审读及监控下,在各位老师的辛勤编撰下,这套凝聚了所有作者及参与研讨的老师们的经验、智慧和资源,涉及三个大的学科近200种的高职高专教材即将面世。我们希望经过近一年的努力,奉献给读者的这套书是他们渴望已久的适用教材。同时,我们也清醒地认识到,“高职高专”是正在探索中的教育,加之我们的水平和经验有限,教材的选题和编辑出版会存在一些不尽人意的地方,真诚地希望得到老师和学生的批评、建议,以利今后改进,为繁荣我国的高职高专教育不懈努力。

科学出版社

前 言

程序设计的核心是数据结构和算法。作为程序设计的基础,“数据结构”是计算机应用与计算机信息管理等专业的基础课程之一。

全书共分 11 章,其中:第 1 章介绍数据、数据结构、算法及其复杂性分析初步和 C 语言的数据类型等基本概念;第 2~7 章及第 10 章介绍线性表、堆栈和队列、串、数组和广义表、树、图、文件等基本数据结构的概念、运算和应用;第 8、9 两章介绍信息处理中广泛使用的算法——排序和查找;第 11 章是上机实验指导,对线性表、链表、树、图等基本数据结构,排序、查找等算法都给出了相关的实验与上机指导,以提高学生编写和调试程序的能力。

本书主编张文明教授,长期从事计算机算法理论的研究与教学,近年来一直担任数据结构与 C 语言等课程的教学,积累了丰富的教学经验。由于数据结构比较抽象、难学,结合高职高专学生的实际情况,在本书编写过程中,几位作者反复研究编写大纲,决定以目前学生们熟悉的 C 语言作为算法的描述性语言,将抽象数据类型用 C 语言的数据类型代替;对数据结构中普遍使用的结构体类型、用户自定义类型作了进一步的讨论,省略大部分算法的复杂性证明;对于所给算法总是从具体例子入手,并给出相应的数据结构,结合具体算法采用通俗易懂的语言或用图示的办法,详细描述算法执行的每一步,以达到帮助学生理解和掌握所给算法的目的。

本书编写分工如下:第 1~4 章由张海防执笔;第 5 章第 1、2 节,第 7、8 章由张文明执笔;第 5 章第 3 节,第 9、10 章由黄志良执笔;第 6 章、第 11 章由张军利编写。全书由张海防统一整理。

刘继章教授、华继学副教授在百忙之中抽出时间,仔细审阅了编写大纲和书稿,并提出了宝贵的修改意见;冯刚等老师在本书的编写过程中也给予了大力支持,在此表示衷心的感谢。

由于作者水平有限、时间仓促,书中难免有疏漏和不足之处,恳请广大读者批评指正。

目 录

第 1 章 绪论	1
1.1 基本概念和术语	1
1.1.1 引言	1
1.1.2 基本概念和术语	5
1.1.3 C 语言的数据类型	5
1.1.4 动态存储分配	6
1.2 算法的描述和分析	6
1.2.1 什么是算法	6
1.2.2 几个数学符号	7
1.2.3 算法分析	7
1.2.4 简单的算法分析举例	8
习题 1	10
第 2 章 线性表	11
2.1 线性表的定义及其运算	11
2.1.1 线性表的定义	11
2.1.2 各种运算简介	11
2.2 线性表的顺序存储结构	12
2.2.1 顺序表	12
2.2.2 顺序表上实现的基本运算	12
2.3 线性表的链式存储结构	15
2.3.1 线性表的链接分配——单链表	15
2.3.2 单链表的基本运算	16
2.4 循环链表和双向链表	25
2.4.1 循环链表	25
2.4.2 双向链表	26
2.5 顺序表和链表的比较	28
2.5.1 基于空间的考虑	28
2.5.2 基于时间的考虑	28
习题 2	29
第 3 章 栈和队列	31
3.1 栈	31
3.1.1 栈的定义及基本运算	31
3.1.2 顺序栈及基本运算	31
3.1.3 双向栈	33

3.1.4 栈的链表存储结构	34
3.2 队列	35
3.2.1 队列的定义及运算	35
3.2.2 队列的顺序存储结构(向量)	36
3.2.3 队列的链表存储结构	39
3.3 栈和队列的应用	41
3.3.1 函数的嵌套调用	41
3.3.2 函数的递归调用	41
3.3.3 栈与队列的应用	42
习题 3	43
第 4 章 串	46
4.1 字符串及其运算	46
4.1.1 字符串的基本概念	46
4.1.2 串的运算简介	46
4.2 串的存储结构	49
4.2.1 串的顺序存储结构	49
4.2.2 串的链式存储	50
4.2.3 串运算的实现	52
习题 4	55
第 5 章 多维数组和广义表	57
5.1 多维数组	57
5.2 矩阵的压缩存储	59
5.2.1 特殊矩阵	59
5.2.2 稀疏矩阵	62
5.3 广义表的概念和特性	65
习题 5	66
第 6 章 树	68
6.1 树的概念	68
6.1.1 树的定义	68
6.1.2 基本术语	69
6.2 二叉树	70
6.2.1 二叉树的定义	70
6.2.2 二叉树的性质	71
6.2.3 二叉树的存储结构	72
6.3 二叉树的遍历	75
6.3.1 中序遍历二叉树的递归算法	76
6.3.2 中序遍历二叉树的非递归算法	77

6.4 线索二叉树	78
6.4.1 中序线索化	79
6.4.2 线索二叉树上的运算	80
6.4.3 遍历线索二叉树	81
6.5 树和森林	81
6.5.1 树、森林与二叉树的转换	82
6.5.2 树的存储结构	84
6.5.3 树和森林的遍历	86
6.6 赫夫曼树及其应用	87
6.6.1 最优二叉树(赫夫曼树)	87
6.6.2 赫夫曼编码	93
习题 6	95
第 7 章 图	98
7.1 图的概念	98
7.2 图的存储结构	101
7.2.1 邻接矩阵表示法	101
7.2.2 邻接表表示法	103
7.3 图的遍历	105
7.3.1 深度优先搜索	105
7.3.2 广度优先遍历	109
7.4 图的生成树	111
7.4.1 生成树的概念	111
7.4.2 最小生成树	111
7.5 最短路径	114
7.5.1 单源最短路径	114
7.5.2 每对顶点之间的最短路径	115
7.6 拓扑排序	116
习题 7	121
第 8 章 排序	124
8.1 三个简单的排序算法	124
8.1.1 冒泡排序	124
8.1.2 直接选择排序	126
8.1.3 直接插入排序	127
8.2 快速排序	128
8.3 谢尔排序	133
8.3.1 方法	133
8.3.2 增量的选择	135

8.4 堆排序	136
8.5 基数排序	138
8.6 合并已整序的表	142
8.6.1 简单的合并算法	142
8.6.2 二路归并排序	143
8.7 字符串排序	144
习题 8	149
第 9 章 查找	153
9.1 基本概念	153
9.2 静态查找表	153
9.2.1 顺序表的查找	153
9.2.2 有序表的二分查找	154
9.2.3 索引顺序表的查找	155
9.3 动态查找表	156
9.3.1 二叉排序树和二叉平衡树	156
9.3.2 B-树和 B+树	159
9.4 哈希表及其查找	163
9.4.1 哈希表与哈希函数	163
9.4.2 构造哈希函数的常用方法	164
9.4.3 解决冲突的主要方法	165
习题 9	170
第 10 章 文件	173
10.1 文件的基本概念	173
10.1.1 文件	173
10.1.2 外存储器及信息特点	175
10.2 文件的组织	177
10.2.1 顺序文件	177
10.2.2 散列文件	178
10.2.3 索引文件	179
10.2.4 索引顺序文件	181
10.3 多关键字文件	183
10.3.1 多重表文件	183
10.3.2 倒排文件	185
习题 10	185
第 11 章 实验内容与上机指导	187
11.1 线性表及其运算	187
11.2 链表及其运算	190
11.3 二叉树的存储与遍历	196

11.4 图的存储与遍历	202
11.5 排序	206
11.6 查找	211
附录 A	216
附录 B	225
参考文献	233

第 1 章 绪 论

随着计算机技术的飞速发展，计算机应用领域迅速扩展，计算机应用已不再局限于科学计算，而广泛应用于数据处理、信息管理等非数值计算的各个方面。数据结构主要研究非数值处理中数据之间的逻辑关系、存储结构和对数据的运算。本章将介绍贯穿全书的基本概念和术语，以及算法描述和算法分析初步。

1.1 基本概念和术语

1.1.1 引言

早期的计算机主要用于数值计算。解决问题的一般步骤是：分析问题，确定数学模型；编写程序，反复调试程序直到得到正确的结果。随着计算机的发展，其应用已经深入到人类社会的各个方面，出现了大量的非数值计算的问题。对于复杂的非数值计算的问题，需要解决怎样合理地组织数据，建立合适的数据结构，选择算法等。数据结构就是在此背景下逐步形成和发展起来的。

为了增加对数据结构的感性认识，下面举例说明有关数据结构的概念。

表 1.1 所示是一个学生学籍表。

表 1.1 学生学籍表

学 号	姓 名	性 别	年 龄	数 学 分 析	普 通 物 理	高 等 代 数
880001	丁一	男	19	90	85	95
880002	马二	女	20	80	85	90
880003	张三	女	19	95	91	99
880004	李四	男	19	70	84	86
880005	王五	男	20	91	84	92
⋮	⋮	⋮	⋮	⋮	⋮	⋮

这里我们把表 1.1 称为一个数据结构，表中的每一行是一个结点（或记录），它由学号、姓名、性别、年龄和各科成绩等数据项组成。该表中数据元素之间的逻辑关系是：对表中任意一个结点，与它相邻且在它前面的结点（亦称为直接前趋，immediate predecessor）最多只有一个；与表中任意一个结点相邻且在其后的结点（亦称为直接后继，immediate successor）也最多只有一个。表中只有第一个结点没有直接前趋，故称为开始结点；也只有最后一个结点没有直接后继，故称为终端结点。例如，表 1.1 中“马二”所在结点的直接前趋结点和直接后继结点分别是“丁一”和“张三”所在的结点，上述结点间的关系构成了这张学生学籍表的逻辑结构。

该表的存储结构则是指用计算机语言如何表示结点之间的这种关系，即表中的结点

是顺序邻接地存储在一片连续的单元之中，还是用指针将这些结点链接在一起。在这张表中，可能要经常查看某一学生的成绩，当学生退学时要删除相应的结点，进来新学生时要增加结点。究竟怎样进行查找、删除、插入操作，这就是数据的运算问题。弄清楚了上述三个问题，也就弄清了学生学籍表这个数据结构。

在不会产生混淆的前提下，我们常常将数据的逻辑结构简称为数据结构。数据的逻辑结构有以下两大类。

(1) 线性结构

线性结构的逻辑特征是：若结构是非空集，则有且仅有一个开始结点和一个终端结点，并且所有结点都最多只有一个直接前趋和一个直接后继。线性表就是一个典型的线性结构。本书第 1 章到第 4 章介绍的都是线性结构。

(2) 非线性结构

非线性结构的逻辑特征是一个结点可能有多个直接前趋和直接后继。第 5 章到第 7 章讨论的数据结构都是非线性结构。

数据的存储结构可用以下四种基本的存储方法得到。

(1) 顺序存储方法

该方法就是把逻辑上相邻的结点存储在物理位置上相邻的存储单元里，结点间的逻辑关系由存储单元的邻接关系来体现。由此得到的存储表示称为顺序存储结构 (sequential storage structure)，通常顺序存储结构是借助数组来描述的。表 1.1 学生学籍表就可以用结构体类型数组，以顺序存储方法存储在一片连续的内存单元里。

(2) 链接存储方法

该方法不要求逻辑上相邻的结点在物理位置上也相邻，结点间的逻辑关系是由附加的指针字段表示的。由此得到的存储表示称为链式存储结构 (linked storage structure)，通常要借助于指针类型来描述它。

(3) 索引存储方法

该方法通常是在存储结点信息的同时，还建立附加的索引表。索引表中的每一项称为索引项，索引项的一般形式是：(关键字，地址)。关键字是能唯一标识一个结点的某些数据项。下面将要讲到的例 1.2 电话号码查询，就是索引存储方法的例子。

(4) 散列存储方法

该方法的基本思想是根据结点的关键字直接计算出该结点的存储地址。

上述四种基本的存储方法，既可以单独使用，也可以组合起来对数据结构进行存储映像。同一种逻辑结构采用不同的存储方法，可以得到不同的存储结构。选择何种存储结构来表示相应的逻辑结构，要视具体要求而定，主要考虑的是运算方便及算法的时间和空间复杂度要求。

【例 1.1】 我们可以把一所高校看成一棵树的根，把下设的若干个系看成它的树枝，再把每个系的各个专业看成树叶。这样就形成一个树型结构，如图 1.1 所示。

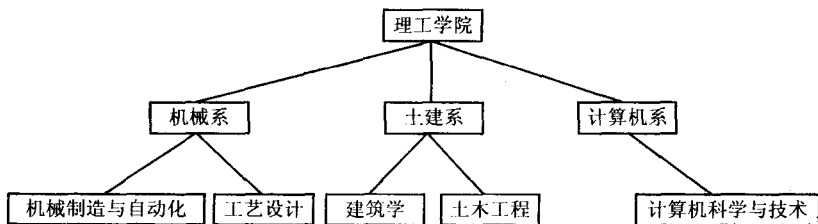


图 1.1 树型结构的例子

树中每个结点可以包含较多的信息，结点之间的关系不再是顺序的，而是层次的、分叉的树型结构。对于树型结构的主要操作是遍历、查找、插入或删除等。

在计算机发展初期，人们使用计算机主要是处理数值计算问题。由于当时所涉及的运算对象是简单的整型、实型或布尔型数据，所以程序设计者的主要精力集中在程序设计的技巧上，而无需重视数据结构。随着计算机应用领域的扩大和软、硬件的发展，“非数值性问题”越来越显得重要。据统计，当今处理非数值性问题占用了 90% 以上的机器时间，这类问题涉及到的数据结构更为复杂，数据元素之间的相互关系一般无法用数学方程式加以描述。因此，解决此类问题的关键已不再是数学分析和计算方法，而是要设计出合适的数据结构。

著名的瑞士计算机科学家 N. 沃思 (N. Wirth) 教授曾提出：算法+数据结构=程序。这里的数据结构是指数据的逻辑结构和存储结构，而算法是对数据运算的描述。由此可见，程序设计的实质是对实际问题选择一种好的数据结构和设计一个好的算法，而好的算法在很大程度上取决于描述实际问题的数据结构。请看下面的两个例子。

【例 1.2】电话号码查询问题。

假定要编写一个程序，查询某个城市或单位的私人电话号码。对任意给出的一个姓名，若该人有电话号码，则要迅速地找到其电话号码；否则，指出该人没有电话号码。解决此问题首先要构造一张电话号码登记表，表中每个结点存放两个数据项：姓名和电话号码。要写出好的查找算法，取决于这张表的结构及存储方式的好坏。最简单的方式是将表中结点顺序地存储在计算机中，查找时从头开始依次查对姓名，直到找出正确的姓名或是找遍整个表均没有找到为止。这种查找算法对于一个不大的单位或许是可行的，但对一个有成千上万私人电话的城市就不适用了。然而，若这张表是按姓氏排列的，则我们可以另造一张姓氏索引表，采用如图 1.2 所示的存储结构。

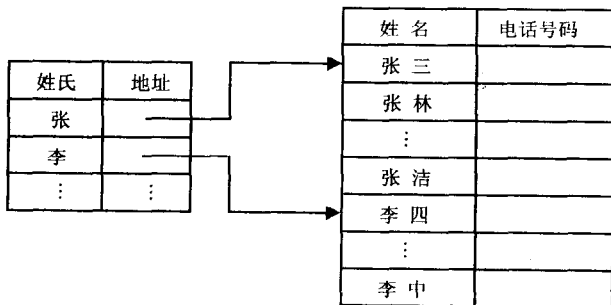


图 1.2 电话号码查询问题的索引存储

那么查找过程是先在索引表中查对姓氏,然后根据索引表中的地址到电话号码登记表中核查姓名,这样查找登记表时就无需查找其他姓氏的名字了。因此,在这种新的结构上产生的查找算法就更为有效。在第 9 章中我们还要讨论有关的查找策略。

【例 1.3】 田径赛的时间安排问题。

假设某校的田径选拔赛共设六个项目的比赛,即跳高、跳远、标枪、铅球、100 米和 200 米短跑,规定每个选手至多参加三个项目的比赛。现有五名选手报名参赛,选手所选择的项目如表 1.2 所示。

表 1.2 参赛选手比赛项目表

姓名	项目 1	项目 2	项目 3
丁一	跳高	跳远	100 米
马二	标枪	铅球	—
张三	标枪	100 米	200 米
李四	铅球	200 米	跳高
王五	跳远	200 米	—

现在要求设计一个竞赛日程安排表,使得选手在尽可能短的时间内安排完比赛。为了较好地解决这个问题,首先应该选择一个合适的数据结构来表示它。为此,我们可以设计这样的一个图(见图 1.3),图中顶点代表竞赛项目,在所有的两个不能同时进行的比赛项目之间连上一条边。显然同一个选手选择的几个项目是不能在同一时间内比赛的,因此该选手所选择的项目中应该两两有边相连。由此可得到如图 1.3 所示的数据结构模型,图中 A, B, ..., F 分别对应六个竞赛项目。例如,丁一选择的三个项目是 A, B 和 E,因而 A 与 B 之间, A 与 E 之间及 B 与 E 之间均有边相连。

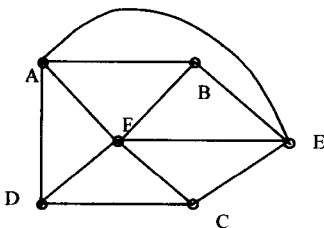


图 1.3 安排竞赛项目的数据结构模型

上述由顶点和边组成的无向图是数据结构中非线性数据结构中的一类。竞赛项目的时间安排问题可以抽象为对该无向图进行“着色”操作,即使用尽可能少的颜色去给图中每个顶点着色,使得任意两个有边连接的相邻顶点着上不同的颜色。每一种颜色表示一个比赛时间,着上同一颜色的顶点是可以安排在同一时间内竞赛的项目。例如,图中顶点 A 和 C 不相邻,可以选取颜色 1 为它们着色;同理, B 和 D 可着同一种颜色 2; E 和 F 有边相连,它们是相邻的,应该分别着上颜色 3 和颜色 4。也就是说,只要安排 4 个不同的时间竞赛即可。时间 1 内可以比赛跳高(A)和标枪(C),时间 2 内可以比赛跳远(B)和铅球(D),时间 3 和时间 4 内分别比赛 100 米(E)和 200 米(F)。

从上述的例子不难看出，解决问题的一个关键步骤是，选取合适的数据结构表示该问题，然后才能写出有效的算法。

1.1.2 基本概念和术语

数据 (data) 是信息的载体，它能被计算机存储、识别、加工和处理。

数据元素 (data element) 是数据的基本单位。有时数据元素也称为结点或记录。下面的学生类型结构体变量 `struct student x, y` 等就是两个结点或记录。

数据项 (成员、字段、域) 是具有独立含义的数据元素的最小标识单位。由它组成数据元素。

数据结构 (data structure) 指的是数据之间的相互关系，即数据的组织形式。虽然至今没有一个关于数据结构的定义，但它一般包括三个方面的内容：

- 1) 数据元素之间的逻辑关系，也称为数据的逻辑结构 (logical structure)。
- 2) 数据元素及其关系在计算机存储器内的表示，称为数据的存储结构 (storage structure)。
- 3) 数据的运算，即对数据施加的操作。

数据的逻辑结构是从逻辑关系上描述数据，它与数据的存储无关，是独立于计算机的。因此，数据的逻辑结构可以看做是从具体问题抽象出来的数学模型。数据的存储结构是逻辑结构用计算机语言的实现 (亦称为映像)，它是依赖于计算机语言的，对机器语言而言，存储结构是具体的，但我们只在高级语言的层次上来讨论存储结构。数据的运算是定义在数据的逻辑结构上的，每种逻辑结构都有一个运算的集合。最常用的运算有检索、插入、删除、更新、排序等。这些运算实际上是在抽象的数据上所施加的一系列抽象的操作。所谓抽象的操作，是指我们只知道这些操作是“做什么”，而无需考虑“如何做”。只有确定了存储结构之后，我们才考虑如何具体实现这些运算。本书中讨论的数据运算，均以 C 语言描述的算法来实现。

1.1.3 C 语言的数据类型

C 语言的数据类型有整型 (int)、实型 (float)、字符型 (char)、数组 (array[N])、结构体 (struct)、共用体 (union)、指针类型、枚举 (enum) 类型、用户自定义类型等，是非常丰富的。

为了存放一个学生的学号、姓名、性别、年龄及三门课的成绩，我们可以定义如下的结构体类型：

```
struct student
{
    long    number ;
    char    name[20] ;
    char    sex ;
    int     age ;
    float   score[3] ;
};
```

有了类型定义，就可以定义变量、数组和指向该类型变量的指针变量。如：

```
struct student x, y;
```