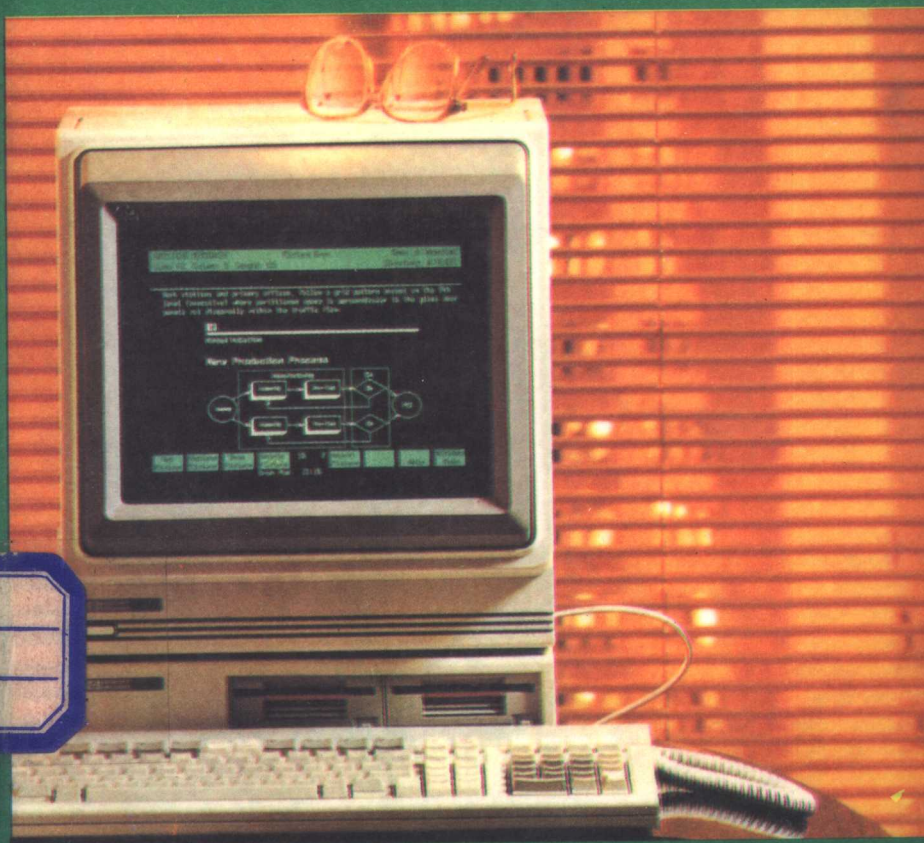


信息学(计算机) 奥林匹克

高级本

北京大学出版社



信息学(计算机)奥林匹克

(高级本)

吴文虎 吕 品 主编

北京大学出版社

新登字(京)159号

信息学(计算机)奥林匹克

(高级本)

吴文虎 吕 品 主编

责任编辑:郭佑民

*

北京大学出版社出版发行

(北京大学校内)

北京大学印刷厂印刷

各地新华书店经售

*

787×1092毫米 32开本 9.75印张 220千字

1992年12月第一版 1992年12月第一次印刷

印数:00001—11,000册

ISBN 7-301-01951-3/G·0130

定价:5.85元

序 言

信息学奥林匹克是计算机知识在世界范围的青少年中普及的产物。1987年,保加利亚的 Sendov 教授在联合国教科文组织第 24 届全体会议上,倡议举行国际信息学奥林匹克,定名为 International Olympiad in Informatics,简称 IOI。首届 IOI 于 1989 年 5 月在保加利亚首都索菲亚举行,13 个国家的 46 名青少年选手参赛,中国队的三名选手获三块铜牌,团体总分第二。IOI' 90(第二届)于 1990 年 7 月在原苏联的明斯克举行,26 个国家的 104 名选手参赛,中国队的四名选手获一块金牌、两块银牌、一块铜牌,团体总分第二。IOI' 91(第三届)于 1991 年 5 月在希腊首都雅典举行,23 个国家的 76 名选手参赛,中国队的三名选手获两块金牌、一块银牌,总分第一。IOI' 92(第四届)于 1992 年 7 月在德国波恩举行,中国队的四名选手获三块金牌、一块银牌,以总分 785 分的优异成绩名列 45 个参赛队之首。

举办国际信息学奥林匹克的目的是:通过竞赛形式对有才华的青少年起到激励作用,促其能力得以发展;让青少年彼此建立联系,推动知识与经验的交流,促进合作与理解;宣传信息学这一新兴学科,给学校这一类课程增加动力,启发新的思路;建立教育工作者与专家档次上的国际联系,推进学术思想交流。

从学科国际奥林匹克来看,只有信息学奥林匹克中国是

首届就派队参赛的,其中的一个重要原因是计算机在中国青少年中的普及受到党和政府的重视,得到了老师、家长和社会各界的支持,许多有识之士认识到“计算机的普及要从娃娃做起”的战略意义。科学技术是第一生产力,未来人才的全面素质,包括科学素养,是发展科学技术,增强综合国力的重要因素。计算机在青少年中的普及是多方面分层次的,有课堂教学也有课外活动,竞赛也是其中的一种形式。在国内各区县、各省市所举行的竞赛的目的,说到底是为了推动普及,使学生开阔眼界,扩大知识面,了解计算机科学基本知识、应用领域、发展状况,培养逻辑思维、创造性思维,以及应用计算机解决实际问题的初步能力。要普及就要有教材,课堂教学要有课本,课外活动也要有辅导材料。为了推动计算机的普及,北京市信息学(计算机)奥林匹克学校的几位热心而又有经验的老师集体编写了这套课外读物:《信息学(计算机)奥林匹克》,包括初级本(适合于小学高年级和初中低年级学生)、中级本(适合于初中高年级和高中低年级学生)、高级本(适合于高中高年级学生)、LOGO 本(适合于学过 LOGO 语言的中、小学生)和习题集上、下册(包括各册中的习题和解答)。前三册供学过 BASIC 语言的同学选用。这套读物涉及计算机的典型算法和基本的数据结构知识,重在讲解编程解题的思路与技巧,有丰富的例题、习题和透彻的讲解,希望能使读者感到亲切爱读。这套读物由北京大学出版社出版发行,可供各地“信息学奥林匹克学校”选用。这里需要说明的是,计算机是实践性很强的学科,阅读教材应和上机实践密切结合才能真正学懂,学到手。另外,书上给出的方法也只是为了抛砖引玉。小读者中有许多是善于独立思考的人,我们鼓励创造性的思维,希望你编写的

程序更有效、更高明。

信息学竞赛是智力与计算机解题能力的竞赛。中国的青少年业已在世界大赛中显露出自己的才华,他们没有辜负党和人民的期望。社会主义祖国重视青少年的全面发展与科学素养的提高。中华民族有志气有能力自立于世界民族之林。作为教师,我们有责任自觉地将普及现代科技知识与中华民族光辉灿烂的未来联系起来。“精心育桃李,切望青胜蓝”是我们写这套书的初衷。

**中国计算机学会普及委员会主任
国际信息学奥林匹克中国队总教练**

吴文虎

1992年8月于清华园

前 言

信息学(计算机)科学正以前所未有的、日新月异的速度向前发展,它有综合性强、应用极其广泛等特点。作为未来 21 世纪的人才,当今的中学生,很有必要学习其中一些基本知识和技能,提高自身素质,以适应将来的社会发展需要。目前各地区、各学校一些计算机课外小组、奥林匹克学校雨后春笋般地出现,正是适应了这种需要。这些学生迫切希望有适合他们学习的教材和参考书籍。本套图书就是为此目的编写的。

本书写作的宗旨是向读者介绍有关的入门知识,力求通俗易懂,便于理解。因此讲解以实例为主,理论上不刻求严格。大部分理论只给出原理,作一些必要的解释,不加以证明。读者如有兴趣,想进一步深入地探讨,请阅读有关专门的书籍。

这本书中的程序,遵循普及和提高相结合的原则,一般用 BASIC 和 PASCAL 两种语言写出,其中语句尽量用通用的基本语句。因此,有的程序从技巧上讲不是最简的。但我们这里主要研究的是算法,语言的技巧是次要的。读者也可以根据你使用的语言自己加以优化。

学习计算机的一条捷径就是多上机实践。因此这本书每章后面都有一定量的习题,提供给读者作为练习用。希望读者尽可能地给予完成,并通过上机实践,真正提高自己的分析问题和解决问题的能力。

本书第一章由吴文虎编写,第二章由俞咪华编写,第五章

由陈星火编写,第十章由庄燕文编写,附录一由赵志方编写,第三、四、六、七、八、九章和附录二由吕品编写。全书由全国中小学计算机教育研究中心兼职教研员、北京市信息学(计算机)奥林匹克学校副校长吕品老师统稿,最后由吴文虎教授审定。

由于编者水平有限,书中难免存在不少缺点和错误,殷切希望广大读者给予批评和指正。

编 者

1992年9月

目 录

第一章 结构化程序设计	1
§ 1.1 结构化程序设计的概念	1
§ 1.2 三种基本结构	2
§ 1.3 结构化流程图	9
§ 1.4 使用结构化流程图举例	12
习题一	14
第二章 筛选法	16
§ 2.1 用筛选法求某范围的素数	16
§ 2.2 不相同的余数问题	21
§ 2.3 其他问题	28
§ 2.4 小结	32
习题二	33
第三章 递归	35
§ 3.1 概述	35
§ 3.2 两个典型例子	40
§ 3.3 希尔伯特曲线	45
§ 3.4 八皇后问题	52
§ 3.5 表达式树	62
§ 3.6 消除递归	71
习题三	76
第四章 算法设计的常用策略	81

§ 4.1	归纳策略	81
§ 4.2	分治策略	89
§ 4.3	最优策略	96
§ 4.4	穷举策略	103
§ 4.5	产生式系统	111
习题四		118
第五章 深度优先搜索		122
§ 5.1	引入	122
§ 5.2	深度优先搜索基本算法(一)	129
§ 5.3	深度优先搜索实例	133
§ 5.4	深度优先搜索基本算法(二)	150
§ 5.5	小结	155
习题五		158
第六章 广度优先搜索法		164
§ 6.1	引入	164
§ 6.2	应用	174
§ 6.3	小结	184
习题六		188
第七章 启发式搜索		190
§ 7.1	启发函数	190
§ 7.2	A 算法	202
§ 7.3	A* 算法	211
习题七		213
第八章 分枝定界		215
§ 8.1	旅行问题	215
§ 8.2	任务安排问题	228
§ 8.3	任务的最佳排序	237
习题八		244

第九章 动态规划	246
§ 9.1 最佳原理	246
§ 9.2 生产规划	251
§ 9.3 0/1 背包问题	257
§ 9.4 最短路径	260
习题九	263
第十章 博弈	266
附录一 常用国家标准程序流程图符号及说明	292
附录二 算法描述语言	294

第一章 结构化程序设计

§ 1.1 结构化程序设计的概念

凡是学过程序设计的都会感觉到,有时看懂别人编的程序要比自己动手编,花的时间多得多。其中重要的一点就是在编写程序时规范化不够,编程思路表述不清楚,缺乏必要的注释。这样,不同的编程者之间就缺乏共同语言,很难交流。不规范的程序设计文本有时就像“天书”,让人望而生畏。杂乱无章的后果,隐藏着大量的估计不到的错误,使程序的可靠性成了问题。在软件急需纳入大生产的轨道的形势下,人们希望能够遵循一种系统的,严格的,规范化的设计方法,使设计出的程序结构清晰,易于理解,便于维护。

为了提高程序的易读性,确保程序质量,降低软件开发成本,荷兰学者 Dijkstra 等人提出了“结构化程序设计方法”,其基本点是:

(1) 程序的质量标准应该是“清晰第一,效率第二”,不清晰,意味着思路紊乱,埋藏着出错隐患,效率可能是虚假的;

(2) 要求编程者按工程化、规范化要求书写程序,使生产出的软件有统一的标准,统一的风格;

(3) 规定三种“基本结构”,并以这三种基本结构作为构筑程序的“构件”。这三种基本结构是:顺序结构,选择结构,重复结构。任何一个复杂的程序都可以用这三种基本结构经过组合与嵌套构造出来。

(4) 采取自顶向下的分层次设计方法, 一个大的任务先分成若干个子任务, 每个子任务就是一个模块。每个模块又可分细为小的模块, 这样逐级细分直至可以用基本结构来表述为止。要求模块之间具有一定的逻辑独立性。

(5) 只有必要时才对程序进行优化, 譬如内存紧张或运行时间过长等等。

结构化程序设计是一种程序设计思想, 无论用哪种语言编程都应该遵循这种思想。不过用 PASCAL 编程就较用 BASIC 编程容易些。因为前者属于结构化的高级语言, 其结构和语句都与三种基本结构相对应。用 PASCAL 语言设计的程序可以不含 GOTO 语句, 而一般 BASIC 语言就较难做到不出现 GOTO。为了改变这种情况而保留 BASIC 的优点, 1985 年出现了一种结构化的 BASIC 最新版本——TRUE BASIC。用这个新版本可以实现结构化、模块化的程序设计。

§ 1.2 三种基本结构

一、顺序结构

这是最简单的一种结构。如图 1.1。

图中画了三个模块, 自上而下先执行 A 块, 再执行 B 块, 最后执行 C 块, 按模块出现的先后顺序执行。这种顺序结构有一个入口和一个出口。其中的每个模块都是由若干条(至少一条)语句构成的。

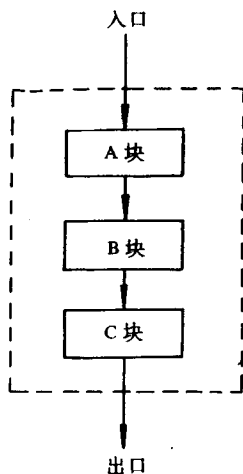


图 1.1 顺序结构

二、选择结构

也称分支结构。如图 1.2。

图 1.2(a)是：当条件 A 得到满足，程序执行 B 块，否则（不满足）执行 C 块。图 1.2(b)是：满足条件 A 则执行 B 块，否则（不满足）什么也不做就退出选择结构。这两种选择结构的差别是，图 1.2(a)用语句

```
if 条件 A
then
    B 块
else
    C 块
```

图 1.2(b) 用语句

```
if 条件 A
then
    B 块
```

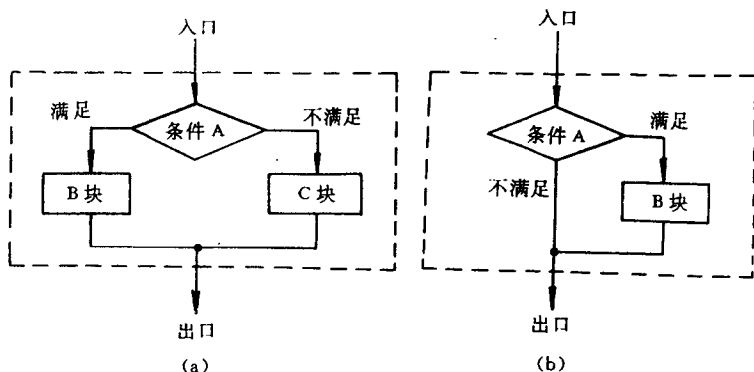


图 1.2 选择结构

选择结构也是一个入口,一个出口。

为了使两个以上的分支的结构更为清晰,在 BASIC 语言和 Pascal 语言中设计了 CASE 语句,其结构如图 1.3 所示。这种结构根据 A 的取值(A 在 $L_1, L_2, \dots, L_n$ 中取值)决定执行 $B_1, B_2, \dots, B_n$ 中的一个。这种多分支结构的语句形式为:

```
case A of
    L1 : B1 块 ;
    L2 : B2 块 ;
    :
    Ln : Bn 块 。
```

多分支结构仍然可以保证一个入口,一个出口。

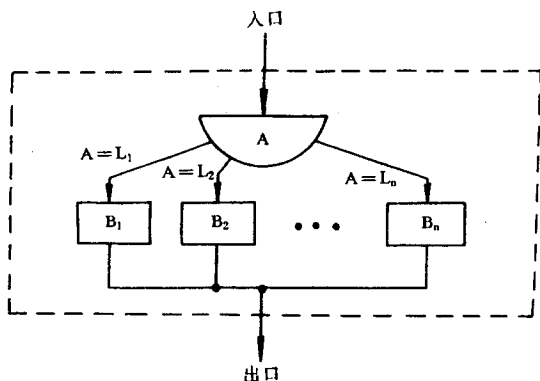


图 1.3 多分支结构

三、重复结构

重复结构有三种基本形式:

(1) For-Do 循环。这是很有用的一种循环结构,在 BASIC 语言版本中用 FOR NEXT 语句。在 Turbo Pascal 中用如下形式的语句结构:

```
For a:=i To e Do  
  Begin  
    S  
  End
```

其中 a 为循环控制变量; i 为初值, e 为终值, i 与 e 均在 For-Do 语句之前定义好; S 为循环体。这里 a 起计数器的作用。第一次执行循环时,置 a 等于初值 i 。执行完循环体 S 的最后一

条语句后,计数器 a 自动加 1,只要 a 小于终值 e ,就不断地执行循环体 S ;一旦 a 大于终值 e ,则不再执行循环体,退出循环结构。图 1.4 为 For-do 循环结构的框图。

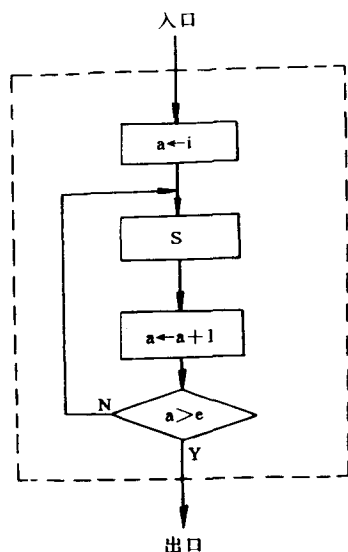


图 1.4 For-Do 循环

(2) Repeat-Until 循环。先看一下 Turbo Pascal 语言给出的这种循环的通用格式：

```

Repeat
    S;
Until C;

```

其中 Repeat 和 Until 是这种循环的标示符; S 是循环体; C 是控制循环退出的条件。这种循环可解释成:重复做 S ,直到条