

信息学(计算机)奥林匹克丛书

信息学(计算机)奥林匹克

(基础篇下册)

戚小玲 俞咪华 编著
李维诩 李玉敏

北 京 大 学 出 版 社
北 京

内 容 简 介

本书是信息学(计算机)判断与推理、排序、字符串处理方法与技巧、文本图形和综合举例。本书介绍了程序设计的基本方法以及编程技巧,内容丰富、取材新颖、富有趣味性,叙述深入浅出,语言通俗易懂。书中所有程序均用 GW BASIC 语言编写,并在 PC 机上通过。

本书适合作为中小学计算机课外活动小组的教材,也可作为广大青少年计算机爱好者深入学习计算机知识的自学读物。对于从事中小学计算机教育的广大教师,本书也有较好的参考价值。

图书在版编目(CIP)数据

信息学(计算机)奥林匹克:基础篇 下册/戚小玲等编. —北京:北京大学出版社,1996.6
(信息学(计算机)奥林匹克丛书/吴文虎主编)
ISBN 7-301-03078-9

. 信... . 戚... . 电子计算机-程序设计-基本知识 计算机课-中学-补充教材
.G634.672

中国版本图书馆 CIP 数据核字(96)第 09701 号

书 名: 信息学

著作责任者: 戚小玲等

责任编辑: 郭佑民

标准书号: ISBN 7-301-03078-9 TP·286

出版者: 北京大学出版社

地 址: 北京市海淀区中关村北京大学校内 100871

电 话: 出版部 62752015 发行部 62559712 编辑部 62752032

排 印 者: 北京市蓝地公司激光照排

发 行 者: 北京大学出版社

经 销 者: 新华书店

787 × 1092 毫米 16 开本 7.625 印张 200 千字

1996 年 7 月第一版 1996 年 7 月第一次印刷

定 价: 13.00 元

序 言

科学技术普及工作是传播科学知识、科学思想和科学方法，提高全民科学文化素质的关键措施，是实施“科教兴国”战略，加速科学技术进步，促进社会主义物质文明和精神文明建设的重要内容。在科技竞争、人才竞争已成为世界各国竞争焦点的今天，科学技术的普及活动也已成为衡量综合国力的一个重要尺度。从培养 21 世纪人才着眼，抓好青少年科技教育，是一件十分有意义的事情，这也是我们编写这套丛书的初衷。

众所周知，现代信息技术正在对人类社会产生着难以估量的深远影响，并将成为新世纪的一个标志。将有关信息科学知识纳入到未来人才的知识结构中来，是时代的需要。人们越来越认识到：计算，跟语言一样是人类社会每时每刻都不可缺少的事情；计算已经成为与理论研究和实验研究并列的第三种研究方法；计算机成了“人类通用智力工具”；计算机和现代通信网络结合成了教育面向世界、面向未来、面向现代化的支撑环境；信息技术与基础教育相结合已经成为当今世界的大趋势。国家教委副主任柳斌同志在论述“为什么要重视计算机教育”时说：“经验证明，计算机技术越是高度发展，计算机人才就越重要，计算机教育就越重要。只有培养一批又一批掌握现代化已经成熟的电子计算机技术人才，并不断发展和提高我国的计算机技术水平，我们才能加速我国走向现代化，走向世界，走向未来的历史进程。”

信息学奥林匹克竞赛最根本的一条宗旨是推动计算机在青少年中的普及。从 1989 年到 1995 年已经成功地举行过七届世界大赛。在这七届大赛中，我国的青少年选手取得了名列前茅的好成绩。这表明，中国孩子有志向、有能力、有条件掌握先进的信息科学知识，具备成为合格的跨世纪人才的素质条件。学科奥林匹克对青少年树立自立、自信、自尊、自强精神，起到了激励和鼓舞作用。

信息学奥林匹克竞赛属于课外因材施教活动，重在培养能力，特别是创造能力。按照教育心理学的观点，创造能力的培养必须以实践为基础。在计算机上编程解题是一项极好的实践活动，它可以训练观察能力、逻辑思维能力、抽象化形式化描述问题能力、规划能力、动手动脑分析问题和解决问题的能力。总之，这是一项用电脑帮助开发人脑的有益活动。正因为如此，热心于普及教育的老师们集体编著了这套丛书。这套丛书按不同年龄段设计了不同的内容，分层次由浅入深，介绍计算机的典型算法和基本数据结构知识，重点讲解编程解题的思路与技巧，有丰富的例题和习题，许多都是能够引发兴趣的题目，有的有相当难度。这里需要强调：计算机是实践性很强的学科，不上机是学不会的。阅读丛书应和上机实践紧密配合才能真正学懂、学到手。另外，书上给出的方法也只是为了抛砖引玉。我们鼓励创造，尤其希望读者编出的程序更有效、更高明。“精心育桃李，切望青胜蓝”是我们的座右铭。

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

1995 年 12 月于清华园

前 言

自 1989 年开始,我国已连续七届派出选手参加国际信息学奥林匹克竞赛,并都载誉而归。信息学奥林匹克竞赛推动了计算机知识在中小学的普及,使学生开阔了眼界,扩大了知识面,提高了逻辑思维、创造性思维以及用计算机解决实际问题的能力。随着计算机技术的迅速发展,中小学使用的计算机已逐渐由 APPLE 机转为 PC 机。为了适应计算机机型的变化,使广大青少年计算机爱好者能够较为系统地学习计算机程序设计的基本方法,了解信息学奥林匹克竞赛的内容和形式,我们组织力量重新编写了“信息学丛书”。

本书的作者都是多年来从事中小学计算机奥林匹克竞赛训练、辅导工作的教师,具有丰富的教学经验,并在培养计算机优秀后备人材方面作出过积极的贡献。本书是他们多年经验与心血的结晶。

参加本书编写工作的有戚小玲(第一章、第七章)、俞咪华(第二章)、李玉敏(第三章)、李维翎(第四章、第五章、第六章)。全书最后由戚小玲老师统稿。

由于编写时间仓促,编者水平有限,书中难免有不妥之处,恳请广大读者批评指正。

编 者

1996 年 1 月

目 录

第一章 数值计算	(1)
第一节 趣味数学计算	(1)
第二节 一般数学问题	(8)
习题一	(23)
第二章 排列和组合	(26)
第一节 排列和组合	(26)
第二节 排列组合综合举例	(38)
第三节 产生排列与组合的其他方法	(43)
习题二	(45)
第三章 逻辑判断与推理	(47)
第一节 基本概念	(47)
第二节 逻辑判断与推理题	(48)
习题三	(57)
第四章 字符串处理的方法与技巧	(60)
第一节 字符串的比较	(60)
第二节 字符串函数及其应用	(63)
第三节 综合举例	(70)
习题四	(72)
第五章 文本图形	(74)
第一节 单一符号组成的图形	(74)
第二节 有规律的数字组成的图形	(80)
第三节 有规律的字符组成的图形	(84)
第四节 特殊文本图形	(86)
习题五	(88)
第六章 排序	(91)
第一节 选择法	(91)
第二节 最大（小）元素法	(93)
第三节 双数组法	(94)
第四节 冒泡法（沉淀法）	(95)
第五节 穿梭法	(96)
第六节 插入法	(98)
第七节 折半插入法	(99)
习题六	(100)
第七章 综合举例	(101)

第一章 数值计算

数值计算是计算机一个重要的应用领域。许多数学问题由人工处理费时、费力，而且出错率较高；用计算机处理，可以快速准确地得出结果，使问题迎刃而解。下面，将通过一些实例介绍用计算机进行数值计算的方法。

第一节 趣味数学计算

【例 1】 巧填数字：

将 1 - 6 这六个数字分别填到图 1.1 的圆圈中，使三角形每条边上的三个数之和相等，一共有多少种方案？编程打印出这些方案。

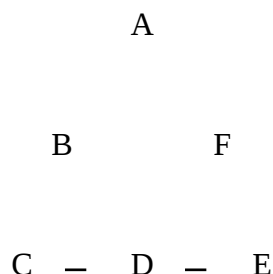


图 1.1

方法一：

分析：设填入六个圆圈中的数字分别为 A, B, C, D, E, F (如图 1.1)，如果满足条件，则需有：

- (1) A, B, C, D, E, F 的取值范围均为 1 - 6 的整数；
- (2) A, B, C, D, E, F 互不相同；
- (3) $A + B + C = C + D + E = E + F + A$ 。

条件 (1) 确定了变量 A, B, C, D, E, F 的取值范围，在此范围内，同时满足条件 (2)、(3) 的即是解。

对于多重循环，尤其是各循环变量的取值范围较大时，如果减少一重循环，就可以使程序的运行速度提高几倍。由条件 (1)、(2) 可得，A, B, C, D, E, F 六个变量的和为 21，即 $A + B + C + D + E + F = 21$ ，当其中的五个变量的值已经确定，且互不相同时，第六个变量的值就可用减法求出，从而可减少一重循环。

程序及运行结果如下：

```
5 REM "QIAOTIANSUANSI1"
10 FOR A = 1 TO 6
20 FOR B = 1 TO 6
25 IF A = B THEN 110
30 FOR C = 1 TO 6
35 IF C = A OR C = B THEN 100
```

```

40 FOR D = 1 TO 6
45 IF D = A OR D = B OR D = C THEN 90
50 FOR E = 1 TO 6
55 IF E = A OR E = B OR E = C OR E = D THEN 80
60 F = 21 - A - B - C - D - E
65 IF D = E OR D = C OR D = A THEN 120
70 IF A + B + C = C + D + E AND C + D + E = A + F + E THEN GOSUB 200: N = N + 1
75 IF F = E OR F = D OR F = C OR F = B OR F = A THEN 120
80 NEXT E
90 NEXT D
100 NEXT C
105 PRINT
110 NEXT B
120 NEXT A
130 PRINT "h = "; N
140 END

200 PRINT TAB (10); A
210 PRINT TAB (8); B; TAB (12); F
220 PRINT TAB (6); C; TAB (10); D; TAB (14); E
230 PRINT
240 RETURN
RUN

```

```

      1
    4  6
  5  2  3
      1
    5  6
  3  4  2
      1
    6  5
  2  4  3
      1
    6  4
  3  2  5
      2
    3  5
  6  1  4
      ...
      6
    3  1
  2  5  4

```

n = 24

以上方法是解决此类问题的一种基本方法，思路简单，但程序较长，运行速度较慢，配

合数学方法，可以对程序加以改进。

方法二：

分析：设三角形每边上的三个数之和为 X ，则有：

$$(A + B + C) + (C + D + E) + (E + F + A) = 3X$$

$$A + B + C + D + E + F = 21$$

$$X = (21 + A + C + E) / 3$$

$$\text{因而, } B = X - A - C$$

$$D = X - C - E$$

$$F = X - A - E$$

这样，在 A, B, C, D, E, F 六个变量中，只要确定了 A, C, E 三个变量的值，其他变量的值即可确定。因此可以采用三重循环结构编写程序。

当 A, B, C, D, E, F 均为 1 - 6 的整数， $A + B + C + D + E + F = 21$ 且 $A * B * C * D * E * F = 72$ 时，即可保证 A, B, C, D, E, F 互不相同。

程序如下：

```
5 REM "QIAOTIANSUANSI2"
10 FOR A = 1 TO 6
20 FOR C = 1 TO 6
25 IF C = A THEN 130
30 FOR E = 1 TO 6
35 IF E = C OR E = A THEN 120
40 X = (21 + A + C + E) / 3
45 IF X < > INT(X) THEN 120
50 B = X - A - C
60 D = X - C - E
70 F = X - A - E
75 IF A * B * C * D * E * F < > 720 THEN 120
80 PRINT TAB(10);A
90 PRINT TAB(8);B;TAB(12);F
100 PRINT TAB(6);C;TAB(10);D;TAB(14);E
105 PRINT
110 N = N + 1
120 NEXT E
130 NEXT C
140 NEXT A
150 PRINT "h = ";N
160 END
```

【例 2】 四则等式：

图 1.2 由加、减、乘、除四个等式拚搭而成。编程将 1 - 8 这八个数字填到算式的空格中去，使四个等式都成立（数字不能重复使用）。

分析：设填入空格中的数字分别为 I, J, K, M, P, T, N, L （如图 1.2）。在上述八个变量中，只要确定 I, J, L, M 四个变量的值，其他变量的值即可确定。因此可以采用

I	-	J	=	K
÷				+
L				M
=				=
N	×	T	=	P

图 1.2

四重循环结构编写程序。

对于嵌套层数较少的多重循环，可以用条件语句直接判断循环变量的值是否相等，以排除循环变量的值相等的情况（即排除有重复数字的情况）。当嵌套层数较多时，用这种方法，会使程序很长，且影响运行速度。下面介绍一种使用标志数组判重的方法。

建立数组 A(8), 数组中变量的意义如下：

A(I) = 1 数 I 已被赋给循环变量
0 数 I 还未使用

在内层循环体中, 首先要判断循环变量的当前值是否已被外层循环使用。如果已被使用 (A(I) = 1), 就转到 NEXT 语句去寻找下一个循环变量值。如果未被使用 (A(I) = 0), 把相应的标志变量置为 1, 表示现在这个值已被使用, 然后再进行下一步处理。当某种方案不符合题意, 重新选择方案时, 需要把相应的标志变量清零。

程序及运行结果如下：

```

3 REM "SISUANSHI "
5 DIM A(8)
10 FOR I= 2 TO 8
15 A(I) = 1
20 FOR J= 1 TO I - 1
30 K = I - J: IF A(K) = 1 THEN 185
45 A(J) = 1: A(K) = 1
50 FOR L = 1 TO I - 1: IF A(L) = 1 THEN 175
60 N = I \ L
65 IF N < INT(N) OR A(N) = 1 THEN 175
70 A(L) = 1: A(N) = 1
80 FOR M = 1 TO 7: IF A(M) = 1 THEN 165
90 P = K + M: IF P > 8 THEN 165
95 IF A(P) = 1 THEN 165
100 A(M) = 1 : A(P) = 1
105 T = P \ N: IF T < > INT(T) THEN 160
110 IF A(T) = 1 THEN 160
115 PRINT I; "- "; J; "= "; K
120 PRINT "/ " + "
125 PRINT L; " "; M
130 PRINT " = " = "

```

```

140 PRINT N; "*" ;T; "= ";P
150 PRINT
160 A(M) = 0:A(P) = 0
165 NEXT M
170 A(L) = 0:A(N) = 0
175 NEXT L
180 A(J) = 0:A(K) = 0
185 NEXT J
190 A(I) = 0
200 NEXT I:END
RUN
6 - 5 = 1
/      +
3      7
=      =
2 * 4 = 8
8 - 7 = 1
/      +
4      5
=      =
2 * 3 = 6

```

【例 3】 素数乘法算式：

在下面的乘法算式(图 1.3)中,“ * ”号只能用素数 2,3,5,7 代替,因此称为素数乘法算式。编程找出素数乘法算式所有可能的方案,并按算式的格式打印出来。

$$\begin{array}{rcccc}
 & & * & * & * \\
 & \times & & * & * \\
 \hline
 & * & * & * & * \\
 * & * & * & * & \\
 \hline
 * & * & * & * & *
 \end{array}$$

图 1.3

分析：算式中被乘数的数字可以是 2, 3, 5, 7, 而乘数的数字只能是 3, 5, 7, 否则会出现非素数数字。

本题的关键在于如何判断运算过程中所得的数字是否为素数。我们可采用以下方法：对于运算过程中产生的每一个整数 X, 用分离数字的方法取出 X 的各位数字, 然后判断是否为 2, 3, 5, 7 中的数。

由于需要多次判断整数 X 中的数字是否为素数, 因此采用编写子程序的方法。在子程序中, 设立标志变量 T, 当 T = 0 时, 表示 X 的各位数字不全是素数, 需要转到 NEXT 语句去寻找新的乘数或被乘数; 当 T = 1 时, 表示 X 的各位数字均为素数, 可以给 X 赋新值继续判断 X 中的数字是否为素数, 或对满足条件的方案进行打印。

程序及运行结果如下:

```
5 REM "SUSHUSUANSHI"
10 DIM A(4)
20 A(1) = 2:A(2) = 3:A(3) = 5:A(4) = 7
30 FOR I = 1 TO 4
40 FOR J = 1 TO 4
50 FOR K = 1 TO 4
60 FOR M = 2 TO 4
70 FOR N = 2 TO 4
80 A = A(I) * 100 + A(J) * 10 + A(K)
90 B = A(M) * 10 + A(N)
100 C = A(N) * A:D = A(M) * A
110 X = C:GOSUB 500
120 IF T = 0 THEN 260
130 X = D:GOSUB 500
140 IF T = 0 THEN 250
145 E = 10 * D + C
150 X = E:GOSUB 500
160 IF T = 0 THEN 250
170 PRINT TAB(5);A
180 PRINT TAB(3);"*";TAB(6);B
190 PRINT TAB(3);"_____"
200 PRINT TAB(4);C
210 PRINT TAB(3);D
220 PRINT TAB(3);"_____"
230 PRINT TAB(3);10 * D + C
240 PRINT
250 NEXT N
260 NEXT M
270 NEXT K
280 NEXT J
290 NEXT I
300 END

500 X$ = STR$(X):L = LEN(X$)
510 FOR P = 2 TO L
520 Y = VAL(MID$(X$,P,1))
530 IF Y < > 2 AND Y < > 3 AND Y < > 5 AND Y < > 7 THEN 560
540 NEXT P
550 T = 1:RETURN
560 T = 0:RETURN

RUN
```

	775
*	33
	2325
	2325
	25575

程序中 I, J, K 循环是在寻找被乘数, 因此, 循环变量的值由 1 取到 4, 即被乘数的数字可以取变量 A(1), A(2), A(3), A(4) 中的数; M, N 循环是在寻找乘数, 因此, 循环变量的值由 2 取到 4, 即乘数中的数字只能取变量 A(2), A(3), A(4) 中的数。变量 A, B, C, D, E 分别为被乘数、乘数、被乘数与乘数的个位相乘的积、被乘数与乘数的十位相乘的积、被乘数与乘数相乘的积。将 C, D, E 的值分别送到子程序中检验, 如果其中的数字均为 2, 3, 5, 7 中的数, 就打印出结果。

【例 4】 数字游戏:

教室内有一块黑板, 上面写着 1995。教室外有 7 个学生, 其编号分别为 1, 2, 3, 4, 5, 6, 7。他们按编号次序依次走进教室, 每次走进一个人。此人首先把自己的编号写在黑板上, 又把黑板上原来的那个数(注: 不是他刚写到黑板上的数), 记为自己的编号, 然后把黑板擦干净, 再把自己原编号与新编号相加后的结果写在黑板上后, 离开教室。下一个人进入教室, 按同样的做法进行, 直到第 7 个人完成以上任务。请你编程计算最后黑板上写着的数是多少? 每个人新的编号是多少?

分析: 建立数组 A, 记录每个人的编号, 变量 N 表示黑板上的数, 变量 T 表示新的编号。每个人进入教室时黑板上的数, 即是这个人新的编号。用一重循环完成给每个人赋新编号的任务。

程序及运行结果如下:

```
3 REM "SHUZIYOUXI "
5 N = 1995: DIM A(7)
10 FOR I = 1 TO 7: A(I) = I: NEXT I
20 FOR I = 1 TO 7
30 T = N: N = A(I) + N: A(I) = T
50 NEXT I
60 PRINT "Number on board is "; N
70 FOR I = 1 TO 7
80 PRINT "Student . "; I; " num is "; A(I)
90 NEXT I
RUN
Number on board is 2023
Student . 1 num is 1995
Student . 2 num is 1996
Student . 3 num is 1998
Student . 4 num is 2001
Student . 5 num is 2005
```

Student . 6 num is 2010

Student . 7 num is 2016

【例 5】 大王应放在哪个位置：

51 张扑克牌排成一排, 其中有一张大王。从第一张开始 1, 2, 3, 4, 5, ..., 50, 51 地数, 把数单数的牌都取走, 然后又从 1 开始数, 再把数单数的牌都取走 直到最后剩一张牌为大王。编程找出大王开始时应放的位置。

分析：

用数组元素分别表示 51 张牌。扑克牌有被取走和未被取走两种状态, 我们可以用设标记的方法区分这两种状态。例如, 用“0”表示牌已被取走, 用“1”表示牌未被取走。每数一次牌, 都将被取走的牌作上“0”标记, 最后没有该标记的那张牌所在的位置即为解。

建立数组 A, 记录每张牌是否被取走的情况。数组元素的下标表示开始时每张牌摆放的位置号。当 $A(I) = 0$ 时, 表示第 I 张牌已被取走; 当 $A(I) = 1$ 时, 表示第 I 张牌未被取走。

设立计数器 S, 用来计数数的情况。S = 1 时表示所数的数为单数。S = 0 时表示所数的数为双数。开始, 把所有数组元素置为 1 (表示所有的牌均未被取走), 当所数的数为单数时, 将以这个数为下标的元素置 0 (表示这张牌已被取走), 总牌数 K 减 1, 并将计数器 S 置为 - 1 (以保证 S 的值 1, 0 交替变化, 即单数、双数交替出现), 否则, 不作任何处理, 继续数下一张牌。

当总牌数大于 1 时, 重复以上过程。当总牌数等于 1 时, 打印出没有“0”标记 (即 $A(I) < > 0$) 的牌的下标。这张牌的下标就是开始时大王应放的位置。

程序及运行结果如下：

```
3 REM "DAWANGDEWEIZHI"
5 K = 51
10 DIM A(51):FOR I = 1 TO 51:A(I) = 1:NEXT I
15 S = 0
20 FOR I = 1 TO 51
25 S = S + A(I)
30 IF S = 1 THEN S = - 1:K = K - 1:A(I) = 0
40 NEXT I
50 IF K > 1 THEN 15
60 FOR I = 1 TO 51
70 IF A(I) < > 0 THEN PRINT "King is at ";I:END
80 NEXT I
RUN
King is at 32
```

第二节 一般数学问题

【例 6】 军事演习：

一次军事演习, A, B 两队同时从相距 100 公里的两驻地相向而行。A 队行进速度为 10 公里/小时, B 队行进速度为 8 公里/小时。一通信员骑摩托车从 A 队的驻地同时出发为行进中的两队传递消息。摩托车的速度为 60 公里/小时。通信员往返于两队之间, 每遇一队立即返

回,驶向另一队。当两队距离小于 0.5 公里时,通信员停下来,不再传消息。试编程计算通信员跑了多少趟(从一队驶向另一队叫一趟)。

分析:

设: D 表示 A, B 两队之间的距离;

F 代表通信员行进的方向,当 $F = 0$ 时表示向 B 队行进,当 $F = 1$ 时表示向 A 队行进;

S0 为摩托车的速度;

S1 为 A 队的速度;

S2 为 B 队的速度;

TIME 表示通信员从一队出发到达另一队所用的时间;

T 表示通信员跑的趟数。

通信员从 A 队向 B 队行进时,从出发到与 B 队相遇所用的时间 $TIME = D / (S0 + S2)$;从 B 队向 A 队行进时,从出发到与 A 队相遇所用的时间 $TIME = D / (S0 + S1)$ 。相遇时两队之间的距离为原两队之间的距离减去两队速度之和与本趟所用时间之积的差,即 $D = D - (S1 + S2) * TIME$ 。用计数器 T 统计通信员跑的趟数。当 A, B 两队之间的距离 D 大于 0.5 公里时重复以上过程,否则 T 的值即为所求。

程序及运行结果如下:

```
5 REM "TONGXUNYUAN"
10 D = 100:T = 0:F = 0:S0 = 60:S1 = 10:S2 = 8
20 IF F = 0 THEN TIME = D / (S0 + S2):D = D - S1 * TIME:T = T + 1:F = 1
30 IF F = 1 THEN TIME = D / (S0 + S1):D = D - S2 * TIME:T = T + 1:F = 0
35 PRINT D
40 IF D > .5 THEN 20
50 PRINT "T = ";T
60 END
RUN
T = 18
```

【例 7】 绕圈数:

10 个数字排成一圈,如图 1.4 所示。分别把每个数与周围的两个数相加,得到 10 组和。编程求出这 10 组和的最大值与最小值之差。

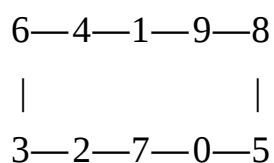


图 1.4

分析: 本题首先要求出 10 组和,然后再求 10 组和中的最大值与最小值,最后求出最大值与最小值之差。

求 10 组和可用一重循环完成。建立数组 A(10),存放图 1.4 中的 10 个数字。因 10 个数字环形排列,可由任何一个数字开始将它们顺序装入数组。观察可得,从第二组到第 9 组,每组和为 $A(I) + A(I - 1) + A(I + 1)$ 。第一组和第 10 组不符合以上规律,可用以下方法求和:

方法一:建立数组 A(11),若 $A(1) = 6, A(10) = 3$,则令 $A(0) = A(10) = 3, A(11) = A(1) = 6$ 。

方法二:设 X, Y 分别为第 $I - 1$ 项和第 $I + 1$ 项的下标,当 $X < 1$ 时,令 $X = X + 10$,当 $Y > 10$ 时,令 $Y = Y - 10$ 。

采用方法一,程序及运行结果如下:

```
5 REM "RAOQUANSHU1"
10 DIM A(11):M = 0: N = 30
20 FOR I = 1 TO 10
30 READ A(I)
40 NEXT I
50 FOR I = 1 TO 10
60 S = A(I - 1) + A(I) + A(I + 1)
70 IF M < S THEN M = S
80 IF N > S THEN N = S
90 NEXT I
100 PRINT "max - min = ";M - N
110 DATA 3,6,4,1,9,8,5,0,7,2,3,6
120 END
RUN
max - min = 13
```

采用方法二,程序及运行结果如下:

```
5 REM "RAOQUANSHU2"
10 DIM A(11):M = 0: N = 30
20 FOR I = 1 TO 10
30 READ A(I)
40 NEXT I
50 FOR I = 1 TO 10
60 X = I - 1:Y = I + 1
70 IF X < 1 THEN X = X + 10
80 IF Y > 10 THEN Y = Y - 10
90 S = A(X) + A(I) + A(Y)
100 IF M < S THEN M = S
110 IF N > S THEN N = S
120 NEXT I
130 PRINT "max - min = ";M - N
140 DATA 6,4,1,9,8,5,0,7,2,3
150 END
RUN
max - min = 13
```

采用方法二编写的程序比方法一复杂,但它的处理技巧在许多问题中需要使用。因此,我们必须掌握这种处理技巧。

【例 8】 求出满足下述条件的两位数,当它们乘以 $2, 3, 4, 5, 6, 7, 8, 9$ 时,所得积的各位数字之和保持不变。

分析:本题需将每一个两位数分别乘以 $2, 3, 4, 5, 6, 7, 8, 9$,把所得积分离数字后求和,然

后判断其和是否保持不变,保持不变的即是解。因需要多次进行分离数字并求和,采用编写子程序的方法。

程序及运行结果如下:

```
5 REM "SHUZIZHIHEBUBIAN"
10 FOR I=10 TO 99
20 Q=I*2
30 GOSUB 500:W=S
40 FOR J=3 TO 9
50 Q=I*J:GOSUB 500
60 IF S<>W THEN 90
70 NEXT J
80 PRINT I
90 NEXT I
100 END
500 S=0
510 A$=STR$(Q)
520 L=LEN(A$)
530 FOR K=2 TO L
540 S=S+VAL(MID$(A$,K,1))
550 NEXT K
560 RETURN
RUN
18
45
90
99
```

在子程序(500—560 语句行)中,首先将数转换为字符串,分离数字后,再将其转换为数,最后进行累加。将数转换为字符串时,数的符号占一个字符的位置,因此分离数字时从第二位开始。

分离数字时,还可先测得字符串的长度 L,然后右取 L-1 个字符,去掉符号位后,再从第一位开始顺序分离。

程序如下:

```
10 FOR I=10 TO 99
20 Q=I*2
30 GOSUB 500:W=S
40 FOR J=3 TO 9
50 Q=I*J:GOSUB 500
60 IF S<>W THEN 90
70 NEXT J
80 PRINT I
90 NEXT I
100 END
```