



主编 章维铁 徐滨
副主编 沈军 王敏

全国青少年信息学

奥林匹克联赛

培训习题与解答 (中学高级本)

南京大学出版社

全国青少年信息学奥林匹克竞赛培训丛书



本书附光盘壹张
含更多测试练习

《全国青少年信息学奥林匹克竞赛培训丛书》 已出版书目

- 青少年信息学(计算机)奥林匹克竞赛小学试题解析
- 全国青少年信息学(计算机)奥林匹克分区联赛试题解析(中学)
- 青少年信息学奥林匹克竞赛培训教材(小学)
- 青少年信息学奥林匹克竞赛培训习题与解答(小学)
- 全国青少年信息学奥林匹克联赛培训教材(中学)
- 全国青少年信息学奥林匹克联赛培训习题与解答(中学)
- 全国青少年计算机(信息学)奥林匹克分区联赛竞赛题集
- 全国青少年信息学奥林匹克联赛培训教材(中学高级本)
- 全国青少年信息学奥林匹克联赛培训习题与解答(中学高级本)(含光盘)

责任编辑 孙 楠

责任校对 王锡昌 装帧设计 杨小民

48.00
(含光盘)

ISBN 7-305-04246-3/TP · 277
总定价:78.00元(共二册)

ISBN 7-305-04246-3



9 787305 042461 >





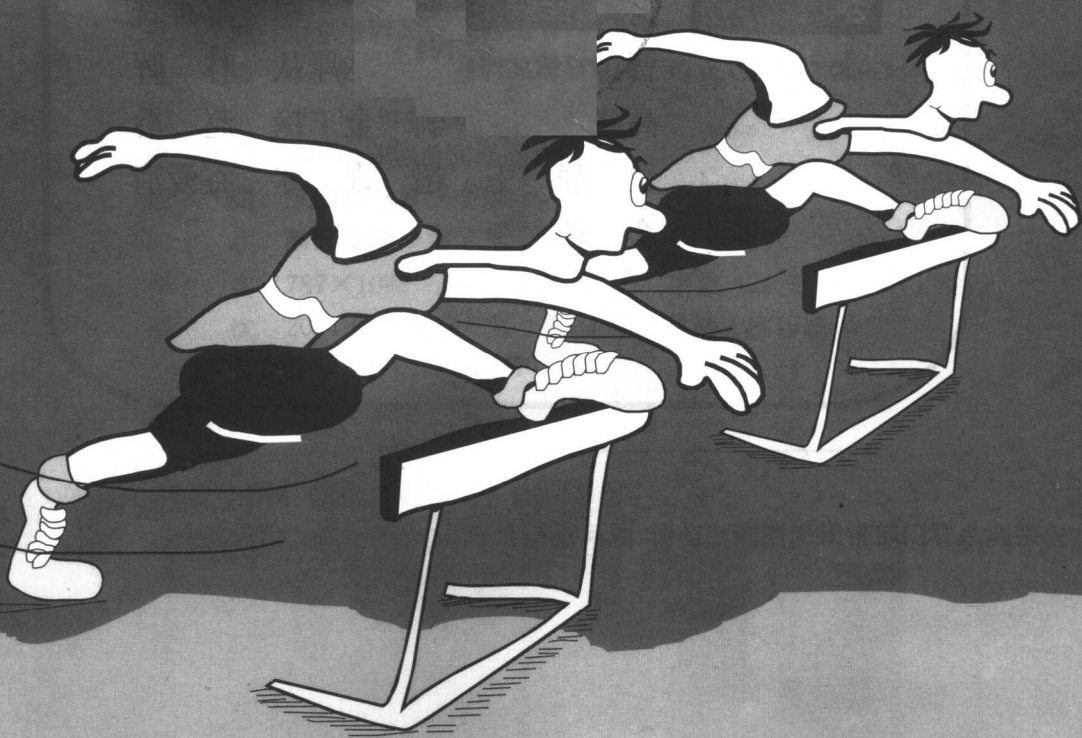
章维铁
林厚从
沈军
王晓敏
徐滨

主编
副主编
主审
主策

全国青少年信息学奥林匹克竞赛培训丛书

全国青少年信息学 奥林匹克联赛 培训习题与解答 (中学高级本)

南京大学出版社



图书在版编目(CIP)数据

全国青少年信息学奥林匹克联赛培训习题与解答。中学高级本/章维铎主编。—南京:南京大学出版社,2004.7
(全国青少年信息学奥林匹克联赛培训丛书)

ISBN 7-305-04246-3

I. 全... II. 曹... III. 计算机课—中学—教学参考资料 IV. G634.673

中国版本图书馆 CIP 数据核字(2004)第 036364 号

丛 书 名 全国青少年信息学奥林匹克竞赛培训丛书
书 名 全国青少年信息学奥林匹克联赛培训习题与解答(中学高级本)
主 编 章维铎
出版发行 南京大学出版社
社 址 南京市汉口路 22 号 邮编 210093
电 话 025-83596923 025-83592317 传真 025-83328362
网 址 <http://press.nju.edu.cn>
电子邮件 nupress1@public1.ptt.js.cn
经 销 全国各地新华书店
印 刷 南京曼德印刷有限公司
开 本 787×1092 1/16 印张 17.25 字数 424 千
版 次 2004 年 7 月第 1 版 2004 年 7 月第 1 次印刷
ISBN 7-305-04246-3/TP·277
定 价 48.00 元(含光盘)

* 版权所有,侵权必究

* 凡购买南大版图书,如有印装质量问题,请与所购图书销售部门联系调换

前 言

受中国计算机学会的委托,从1995年起,江苏省青少年科技中心已连续多年成功承办了全国信息学奥林匹克联赛(简称NOIP)活动,数以十万计的青少年从中受益。在这么多年的联赛活动中,参预此项工作的老师与专家积累了许多宝贵经验,从1999年起陆续撰写出版了一套青少年信息学奥林匹克竞赛丛书,包含初级、中级、高级本及全国青少年信息学奥林匹克联赛试题解析等。中国国家队总教练吴文虎教授给予该套丛书以极高评价,在为该套丛书作序中写到“该套丛书注重了系统性、入门性与实用性,始终围绕编程实践,以算法分析为主线,讲思想、讲方法,侧重基础训练,引导学生在参与的实践中掌握科学思维方法,提高使用计算机的能力。”

现根据活动普及与发展的需要及广大读者的强烈建议,我们将这套丛书中的初级本、中级本、提高本重新进行了编写,并各增加了一本相应的习题集与参考答案(现为初级、中级、高级本共3套6册书,详细书目请见封底)。以上各本辅导教材由南京大学出版社负责出版,均可作为广大青少年参与信息学奥林匹克活动的培训教材。

其中的“高级本”仍以PASCAL语言为载体,以算法介绍为主线,重在剖析思路与讨论研究思维方法,剖析了大量有关例题与习题(其中100多个例题与习题均为首次出现),对提高参赛选手的综合能力起着极为重要的引导作用。教材45万字、习题集40万字并配一张光盘(收录了:①FP软件;②教材中例题的源程序、测试数据;③习题集中的源程序、测试数据;④模拟试题10套)。

参加本书教材部分编写工作的有曹文、林厚从、许冰。其中第一章和第四章由许冰编写,第二、三、五、六章由林厚从编写,第七至第十一章由曹文编写。参加习题与解答部分编写工作的有:章维铎、岳军、林厚从、戴文渊。其中第一章至第四章由岳军、林厚从编写,第五至第九章由章维铎为主编写,戴文渊参与了部分章节的撰写工作,第十章由戴文渊编写。全书由南京航空航天大学李立新教授与东南大学计算机系的沈军老师统一审稿。

在成书的过程中,得到了江苏省青少年信息学奥赛委科学委员会的李立新、王晓敏、宋方敏、沈军、朱玉珑等专家教授以及江苏省青少年科技中心领导的大力帮助,同时还得到了南京大学出版社徐滨老师的大力支持,在此谨向他们表示感谢。

希望广大读者在使用本套教材时提出宝贵意见和建议,以便进一步修改,使之日臻完善。

编 者

2004年5月7日

目 录

全国青少年信息学奥林匹克联赛培训习题与解答(中学高级本)

前言

习 题 篇

第一章 回溯法

1.1	马拦过河卒	2
1.2	出栈序列统计	2
1.3	算 24 点	3
1.4	冗余依赖	4
1.5	走迷宫	5
1.6	单向双轨道	7
1.7	组合的输出	8
1.8	售货员的难题	9
1.9	驾车旅行	9
1.10	关路灯	10

第二章 递归与递推

2.1	遍历问题	12
2.2	产生数	13
2.3	出栈序列统计	13
2.4	计数器	14
2.5	诸侯安置	15
2.6	括号序列	16
2.7	新汉诺塔	17
2.8	排序集合	18
2.9	青蛙过河	19
2.10	电话号码	20
2.11	编码	21

第三章 贪心法

3.1	排队接水	22
3.2	智力大冲浪	22
3.3	取火柴游戏	23



3.4	加工生产调度	24
3.5	最大乘积	25
3.6	种树	26
3.7	餐巾	27
3.8	马拉松接力赛	28
3.9	线性存储问题	29
3.10	扇区填数	30
第四章 分治法		
4.1	取余运算	31
4.2	地毯填补	31
4.3	平面上的最接近点对	33
4.4	求方程的根	33
4.5	小车问题	34
4.6	黑白棋子的移动	34
4.7	麦森数	35
4.8	旅行家的预算	36
4.9	飞行计划	37
第五章 图		
5.1	医院设置	39
5.2	工程规划	40
5.3	服务器储存信息问题	41
5.4	间谍网络	43
5.5	宫廷守卫	44
5.6	K-联赛	45
5.7	机器调度	47
5.8	公路修建	48
5.9	速度限制	49
第六章 树		
6.1	排序二叉树	51
6.2	树的重量	53
6.3	信号放大器	54
6.4	“访问”艺术馆	56
6.5	聚会的快乐	57
6.6	重建道路	57
6.7	有线电视网	58
第七章 搜索		
7.1	最多因子数	61
7.2	黑白棋游戏	61



7.3	纵横填字游戏	62
7.4	魔术数字游戏	64
7.5	魔板	65
7.6	三维扫描	67
7.7	拼字游戏	68
7.8	小木棍	69
7.9	单词游戏	70
第八章 动态规划		
8.1	字符串距离	71
8.2	血缘关系	72
8.3	尼克的任务	73
8.4	书的复制	74
8.5	多米诺骨牌	75
8.6	平板涂色	76
8.7	三角形牧场	77
8.8	分组	78
第九章 数学问题		
9.1	多项式展开系数	79
9.2	两数之和	80
9.3	盒子与球	80
9.4	取数游戏	81
9.5	磁盘碎片整理	82
9.6	欧几里德的游戏	83
9.7	百事世界杯之旅	84
9.8	倒酒	85
9.9	班级聚会	86
第十章 杂题		
10.1	排序	88
10.2	木棍加工	89
10.3	三角形	89
10.4	多边形面积	90
10.5	网线切割	91
10.6	最接近的分数	92
10.7	切孔机	93
10.8	拴狗方案	94
10.9	城市街道交费系统	95
10.10	魔鬼之城	97
10.11	可见矩形	98



解析篇

第一章 回溯法

1.1 马拦过河卒 (简析)	101
1.2 栈 (简析)	102
1.3 算24点 (简析)	102
1.4 冗余依赖 (简析)	102
1.5 走迷宫 (详解)	105
1.6 单向双轨道 (简析)	108
1.7 组合的输出 (详解)	108
1.8 售货员的难题 (简析)	110
1.9 驾车旅行 (简析)	111
1.10 关路灯 (详解)	113

第二章 递归与递推

2.1 遍历问题 (详解)	117
2.2 产生数 (详解)	120
2.3 出栈序列统计 (详解)	123
2.4 计数器 (详解)	125
2.5 诸侯安置 (详解)	128
2.6 括号序列 (简析)	130
2.7 新汉诺塔 (简析)	131
2.8 排序集合 (简析)	133
2.9 青蛙过河 (简析)	133
2.10 电话号码 (简析)	134
2.11 编码 (简析)	136

第三章 贪心法

3.1 排队接水 (详解)	138
3.2 智力大冲浪 (详解)	139
3.3 取火柴游戏 (详解)	142
3.4 加工生产调度 (详解)	144
3.5 最大乘积 (详解)	148
3.6 种树 (简析)	152
3.7 餐巾 (简析)	152
3.8 马拉松接力赛 (简析)	153
3.9 线性存储问题 (简析)	154
3.10 扇区填数 (简析)	154



第四章 分治法

4.1 取余运算 (详解)	155
4.2 地毯填补 (详解)	156
4.3 平面上的最接近点对 (详解)	159
4.4 求方程的根 (简析)	168
4.5 小车问题 (简析)	168
4.6 黑白棋子的移动 (简析)	168
4.7 麦森数 (简析)	169
4.8 旅行家的预算 (简析)	169
4.9 飞行计划 (简析)	170

第五章 图

5.1 医院设置 (详解)	172
5.2 工程规划 (详解)	174
5.3 服务器储存信息问题 (详解)	177
5.4 间谍网络 (简析)	183
5.5 宫廷守卫 (简析)	183
5.6 K-联赛 (简析)	183
5.7 机器调度 (简析)	183
5.8 公路修建 (简析)	184
5.9 速度限制 (简析)	184

第六章 树

6.1 排序二叉树 (详解)	185
6.2 树的重量 (详解)	190
6.3 信号放大器 (简析)	194
6.4 “访问”艺术馆 (简析)	194
6.5 聚会的快乐 (简析)	196
6.6 重建道路 (简析)	196
6.7 有线电视网 (简析)	196
6.8 Two (简析)	196

第七章 搜索

7.1 最多因子数 (详解)	197
7.2 黑白棋游戏 (详解)	201
7.3 纵横填字游戏 (详解)	205
7.4 魔术数字游戏 (简析)	211
7.5 魔板 (简析)	214
7.6 三维扫描 (简析)	214
7.7 拼字游戏 (简析)	214
7.8 小木棍 (简析)	214



7.9	单词游戏 (简析)	215
第八章 动态规划		
8.1	字串距离 (详解)	218
8.2	血缘关系 (详解)	221
8.3	尼克的任务 (详解)	226
8.4	书的复制 (简析)	228
8.5	多米诺骨牌 (简析)	231
8.6	平板涂色 (简析)	232
8.7	三角形牧场 (简析)	233
8.8	分组 (简析)	233
第九章 数学问题		
9.1	多项式展开系数 (详解)	234
9.2	两数之和 (详解)	235
9.3	盒子与球 (详解)	239
9.4	取数游戏 (简析)	241
9.5	磁盘碎片整理 (简析)	241
9.6	欧几里德的游戏 (简析)	241
9.7	百事世界杯之旅 (简析)	243
9.8	倒酒 (简析)	243
9.9	班级聚会 (简析)	243
第十章 杂题		
10.1	排序 (详解)	246
10.2	木棍加工 (详解)	249
10.3	三角形 (详解)	251
10.4	多边形面积 (简析)	255
10.5	网线切割 (简析)	255
10.6	最接近的分数 (简析)	256
10.7	切孔机 (简析)	259
10.8	拴狗方案 (简析)	259
10.9	城市街道交费系统 (简析)	261
10.10	魔鬼之城 (简析)	261
10.11	可见矩形 (简析)	261

习题篇

1.1 马过河卒

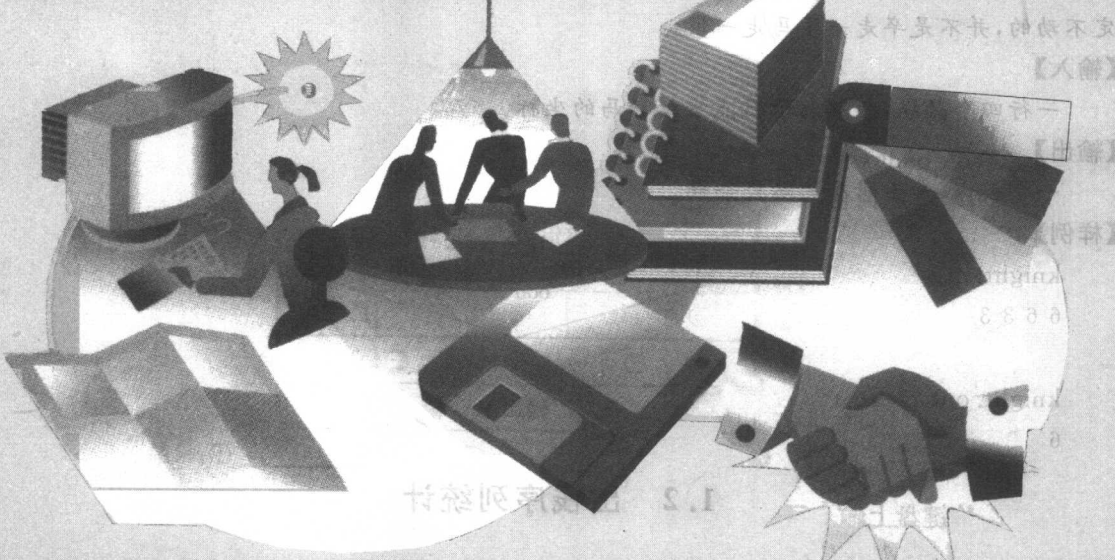
骑士??? (PAS.C, CFP)	源程序名
knight.exe	可执行文件名
knight.in	输入文件名
knight.out	输出文件名

【问题描述】

棋盘上A点有一个过河卒，需要走到目标B点。卒行走的规则：可以向下、向左或向右。卒在行走过程中可能会遇到对方马的拦截。卒从A点走到B点，最短的路线有多少条？

输入：A点坐标(x1, y1)，B点坐标(x2, y2)。马的坐标(x3, y3)。

输出：最短路线的条数。



stackl ??? (PAS.C, CFP)	源程序名
stackl.exe	可执行文件名
stackl.in	输入文件名
stackl.out	输出文件名

第一章 回溯法

1.1 马拦过河卒

源程序名	knight. ??? (PAS,C, CPP)
可执行文件名	knight. exe
输入文件名	knight. in
输出文件名	knight. out

【问题描述】

棋盘上 A 点有一个过河卒,需要走到目标 B 点。卒行走的规则:可以向下、或者向右。同时在棋盘上 C 点有一个对方的马,该马所在的点和所有跳跃一步可达的点称为对方马的控制点。因此称之为“马拦过河卒”。

棋盘用坐标表示,A 点(0,0)、B 点(n,m) (n,m 为不超过 15 的整数),同样马的位置坐标是需要给出的。现在要求你计算出卒从 A 点能够到达 B 点的路径的条数,假设马的位置是固定不动的,并不是卒走一步马走一步。

【输入】

一行四个数据,分别表示 B 点坐标和马的坐标。

【输出】

一个数据,表示所有的路径条数。

【样例】

knight. in
6 6 3 3

knight. out
6

1.2 出栈序列统计

源程序名	stack1. ??? (PAS,C, CPP)
可执行文件名	stack1. exe
输入文件名	stack1. in
输出文件名	stack1. out



【问题描述】

栈是常用的一种数据结构,有 n 个元素在栈顶端一侧等待进栈,栈顶端另一侧是出栈序列。你已经知道栈的操作有两种:push 和 pop,前者是将一个元素进栈,后者是将栈顶元素弹出。现在要使用这两种操作,由一个操作序列可以得到一系列的输出序列。请你编程求出对于给定的 n ,计算并输出由操作数序列 $1, 2, \dots, n$, 经过一系列操作可能得到的输出序列总数。

【输入】

一个整数 $n(1 \leq n \leq 15)$ 。

【输出】

一个整数,即可能输出序列的总数目。

【样例】

stack1.in

3

stack1.out

5

1.3 算 24 点

源程序名	point24.??? (PAS,C, CPP)
可执行文件名	point24.exe
输入文件名	point24.in
输出文件名	point24.out

【问题描述】

几十年前全世界就流行一种数字游戏,至今仍有人乐此不疲。在中国我们把这种游戏称为“算 24 点”。您作为游戏者将得到 4 个 1~9 之间的自然数作为操作数,而您的任务是对这 4 个操作数进行适当的算术运算,要求运算结果等于 24。

您可以使用的运算只有: +, -, *, /, 您还可以使用 () 来改变运算顺序。注意:所有的中间结果必须是整数,所以一些除法运算是不允许的(例如, $(2 * 2) / 4$ 是合法的, $2 * (2 / 4)$ 是不合法的)。下面我们给出一个游戏的具体例子:

若给出的 4 个操作数是: 1、2、3、7, 则一种可能的解答是 $1 + 2 + 3 * 7 = 24$ 。

【输入】

只有一行,四个 1 到 9 之间的自然数。

【输出】

如果有解的话,只要输出一个解,输出的是三行数据,分别表示运算的步骤。其中第一行是输入的两个数和一个运算符和运算后的结果,第二行是第一行的结果和一个输入的数据、运算符、运算后的结果;第三行是第二行的结果和输入的一个数、运算符和“=24”。如果两个操作数有大小的话则先输出大的。

如果没有解则输出“No answer!”



【样例】

```
point24.in
1 2 3 7
```

```
point24.out
2+1=3
7*3=21
21+3=24
```

1.4 冗余依赖

源程序名	redund. ??? (PAS,C, CPP)
可执行文件名	redund. exe
输入文件名	redund . in
输出文件名	redund. out

【问题描述】

在设计关系数据库的表格时,术语“函数依赖”(FD)被用来表示不同域之间的关系。函数依赖是描述一个集合中的域的值与另一个集合中的域的值之间的关系。记号 $X \rightarrow Y$ 被用来表示当集合 X 中的域被赋值后,集合 Y 的域就可以确定相应的值。例如,一个数据表格包含“社会治安编号”(S)、“姓名”(N)、“地址”(A)、“电话”(P)的域,并且每个人都与某个特定的互不相同的 S 值相对应,根据域 S 就可以确定域 N, A, P 的值。这就记作 $S \rightarrow NAP$ 。

写一个程序以找出一组依赖中所有的冗余依赖。一个依赖是冗余的是指它可以通过组里的其他依赖得到。例如,如果组里包括依赖 $A \rightarrow B, B \rightarrow C$ 和 $A \rightarrow C$,那么第三个依赖是冗余的,因为域 C 可以用前两个依赖得到(域 A 确定了域 B 的值,同样域 B 确定了域 C 的值)。在 $A \rightarrow B, B \rightarrow C, C \rightarrow A, A \rightarrow C, C \rightarrow B$ 和 $B \rightarrow A$ 中,所有的依赖都是冗余的。

现在要求你编写一个程序,从给定的依赖关系中找出冗余的。

【输入】

输入的文件第一行是一个不超过 100 的整数 n ,它表示文件中函数依赖的个数。从第二行起每一行是一个函数依赖且互不重复,每行包含用字符“ \rightarrow ”和“ $>$ ”隔开的非空域列表。域列表只包含大写的字母,函数依赖的数据行中不包括空格和制表符,不会出现“平凡”冗余依赖(如 $A \rightarrow A$)。虽然文件中没有对函数依赖编号,但其顺序就是编号 1 到 n 。

【输出】

每一个冗余依赖,以及其他依赖的一个序列以说明该依赖是冗余的,先是一个 FD,然后是依赖函数号,接着是“is redundant using FDs:”最后是说明的序列号。

如果许多函数依赖的序列都能被用来说明一个依赖是冗余的,则输出其中最短的证明序列。如果这些函数依赖中不包含冗余依赖,则输出“No redundant FDs”信息。

【样例 1】

```
redund.in
```



```
3
A->BD
BD->C
A->C
```

redund. out

FD 3 is redundant using FDs:1 2 {即 C 可以通过 1,2 得到}

【样例 2】

```
redund. in
6
P->RST
VRT->SQP
PS->T
Q->TR
QS->P
SR->V
```

redund. out

FD 3 is redundant using FDs:1

FD 5 is redundant using FDs:4 6 2

1.5 走迷宫

源程序名	maze. ??? (PAS,C, CPP)
可执行文件名	maze. exe
输入文件名	maze. in
输出文件名	maze. out

【问题描述】

有一个 $m * n$ 格的迷宫(表示有 m 行、 n 列),其中有可走的也有不可走的,如果用 1 表示可以走,0 表示不可以走,文件读入这 $m * n$ 个数据和起始点、结束点(起始点和结束点都是用两个数据来描述的,分别表示这个点的行号和列号)。现在要你编程找出所有可行的道路,要求所走的路中没有重复的点,走时只能是上下左右四个方向。如果一条路都不可行,则输出相应信息(用-1 表示无路。)

【输入】

第一行是两个数 $m, n(1 < m, n < 15)$,接下来是 m 行 n 列由 1 和 0 组成的数据,最后两行是起始点和结束点。

【输出】

所有可行的路径,描述一个点时用 (x, y) 的形式,除开始点外,其他的都要用“->”表示



方向。

如果没有一条可行的路则输出-1。

【样例】

maze.in

```

5 6
1 0 0 1 0 1
1 1 1 1 1 1
0 0 1 1 1 0
1 1 1 1 1 0
1 1 1 0 1 1
1 1
5 6

```

maze.out

```

(1,1)->(2,1)->(2,2)->(2,3)->(2,4)->(2,5)->(3,5)->(3,4)->(3,
3)->(4,3)->(4,4)->(4,5)->(5,5)->(5,6)
(1,1)->(2,1)->(2,2)->(2,3)->(2,4)->(2,5)->(3,5)->(3,4)->(4,
4)->(4,5)->(5,5)->(5,6)
(1,1)->(2,1)->(2,2)->(2,3)->(2,4)->(2,5)->(3,5)->(4,5)->(5,
5)->(5,6)
(1,1)->(2,1)->(2,2)->(2,3)->(2,4)->(3,4)->(3,3)->(4,3)->(4,
4)->(4,5)->(5,5)->(5,6)
(1,1)->(2,1)->(2,2)->(2,3)->(2,4)->(3,4)->(3,5)->(4,5)->(5,
5)->(5,6)
(1,1)->(2,1)->(2,2)->(2,3)->(2,4)->(3,4)->(4,4)->(4,5)->(5,
5)->(5,6)
(1,1)->(2,1)->(2,2)->(2,3)->(3,3)->(3,4)->(2,4)->(2,5)->(3,
5)->(4,5)->(5,5)->(5,6)
(1,1)->(2,1)->(2,2)->(2,3)->(3,3)->(3,4)->(3,5)->(4,5)->(5,
5)->(5,6)
(1,1)->(2,1)->(2,2)->(2,3)->(3,3)->(3,4)->(4,4)->(4,5)->(5,
5)->(5,6)
(1,1)->(2,1)->(2,2)->(2,3)->(3,3)->(4,3)->(4,4)->(3,4)->(2,
4)->(2,5)->(3,5)->(4,5)->(5,5)->(5,6)
(1,1)->(2,1)->(2,2)->(2,3)->(3,3)->(4,3)->(4,4)->(3,4)->(3,
5)->(4,5)->(5,5)->(5,6)
(1,1)->(2,1)->(2,2)->(2,3)->(3,3)->(4,3)->(4,4)->(4,5)->(5,
5)->(5,6)

```