

信息学奥林匹克竞赛指导丛书

吴文虎摇主编

信息学奥林匹克竞赛指导

——~~国际~~竞赛试题解析

吴文虎摇王建德摇编著

清华大学出版社
北京

内 容 简 介

本书收集了 1994 年国际、国内有关信息学奥林匹克竞赛试题,重点在于分析解题思路和方法上,其中包括数学模型的构建、相应的算法分析以及程序的编写等,这些试题有相当的难度,是训练思维、提高解题技巧的很好参考资料。

书中对试题类型进行了归纳,增强了本书的可读性,既便于教师对参赛学生的辅导,又便于学生自学,所以本书既是参赛选手的必读书,也可作为理工院校编程爱好者提高分析问题、解决问题能力的参考书。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

信息学奥林匹克竞赛指导——1994 年竞赛试题解析 / 文虎, 王建德编著. —北京: 清华大学出版社, 1994. 10

(信息学奥林匹克竞赛指导丛书 文虎主编)

ISBN 7-302-01400-0

I 援信... 摇 II 援①吴... ②王... 摇 III 援计算机课 原中小学 原解答题 摇 IV 援O21 摇 O21 摇 O21

中国版本图书馆 CIP 数据核字(94)第 000000 号

出 版 者: 清华大学出版社

地 址: 北京清华大学学研大厦

社 总 机: 010-62770175

邮 编: 100084

客 户 服 务: 010-62776969

责任编辑: 宋摇方

印 刷 者: 清华大学印刷厂

装 订 者: 三河市新茂装订有限公司

发 行 者: 新华书店总店北京发行所

开 本: 185mm×260mm 印 张: 10.5 字 数: 250 千字

版 次: 1994 年 1 月第 1 版 1994 年 1 月第 1 次印刷

书 号: 7-302-01400-0

印 数: 1~1000

定 价: 15.00 元

前摇摇头

信息科学与技术正在对人类社会的发展产生难以估量的深远影响,已经成为新世纪的一个标志。作为人类总体智慧的结晶,电脑已经成为一种新的现代文化。“计算机的普及要从娃娃抓起”已经成为“科教兴国”的一项重要内容。

一个国家、一个民族要想不落伍,要想跻身于世界先进民族之林,关键在于拥有高素质的人才,综合国力的竞争,说到底也是人才的竞争。电子计算机是现代科学与技术的基础和核心,它的飞速发展,把社会生产力水平提到前所未有的高度,人类进入了信息时代。电脑对人类社会的发展所起的巨大作用,特别是对人类智能的发展所起的促进作用,已为人们普遍认识。计算跟语言一样是人类社会每时每刻都不可缺少的。现在,人类已经拥有了帮助自己进行复杂计算与思维的工具,电子计算机起到了人脑延伸的作用。以往历史上的技术革命,只能起到创造和改进工具,用机器代替人的体力的作用,而计算机则是把人从重复性的或有固定程式的脑力劳动中解放出来,使自己的智能获得空前的发展。作为“人类通用智力工具”的计算机在开发人类智能方面所起的无与伦比的作用不容忽视。信息技术是以计算机为龙头的。这也就是信息技术与基础教育相结合,能够成为当今世界的大趋势的一个原因。从信息社会要求人才具备的科学素养看,数学、物理学、化学、生命科学和信息科学是五大支柱,这正是联合国教科文组织倡导举行五项国际学科奥林匹克竞赛的内容。

国际信息学奥林匹克(国际数学奥林匹克的继兄弟)始于1989年,到1994年已成功地举办了5届。这是一种智力与应用计算机能力的大赛。从益智的角度看,是用电脑帮助开发人脑,重在提高思维能力,培养创新意识。在中国队的训练中强调德智体美全面发展,心态上自立、自尊、自信、自强,要怀着中华民族的自豪感和自信心去参赛,这种心态是学习、训练和取胜的重要条件。5届比赛,中国队每届都取得了名列前茅的好成绩,300人次参赛,夺得3块奖牌,其中金牌1块、银牌1块、铜牌1块。特别是1993年(荷兰)突破了前4届比赛女孩与金奖无缘的纪录,两名中国女选手荣登金牌领奖台。在1994年(匈牙利)上中国队又实现了全金的突破,四名选手,每人获得一块金牌。

从大局看,竞赛不是目的,是推动普及的手段,我们的目的只有一个:科教兴国。竞赛活动带有因材施教、因材施教测的特点。普及是有层次的,与学科竞赛有关的普及活动,对青少年而言属于比较高的层次,当然就有相当的难度。我们编写的这套竞赛指导丛书,涉及程序设计语言、常用算法、组合数学、图论、人工智能搜索等的基本知识和基本方法。这些理论知识往往都是通过竞赛当中的一些实例来讲解的,重点放在解题思路,书中有许多题目比较新颖,很难去套固定算法或固定模式,这中间有些招数

是选手们想出来的。从中可以看出信息学奥林匹克要求创新,鼓励创新。当然,书中给出的解法,对青少年读者而言,我们希望仅仅起到抛砖引玉的作用,并且热切盼望引出更多的“玉”来。作为老师,我和王建德都这样想;“精心育桃李,热望青胜蓝”就是我们编写这套丛书的初衷。

国际信息学奥林匹克中国队总教练
清华大学计算机系教授博士生导师

吴文虎

二〇〇四年 八月

目摇摇录

第 员章摇与课本知识相关的例题	员
员源摇级数求和	员
员圆摇自由落体	圆
第 圆章摇数据结构类的例题	源
圆源摇有关字符串处理的例题	源
圆苑摇并查集与路径压缩	苑
第 猿章摇数论类的例题	员源
猿源摇荒岛野人	员远
第 源章摇组合分析类的例题	圆园
源源摇机器人皂号	圆园
第 缘章摇计算几何学类的例题	圆园
缘源摇矩形覆盖	圆园
第 远章摇图论类的例题	猿猿
远源摇玩具兵	猿猿
第 苑章摇搜索策略类的例题	源源
苑源摇枚举法	源源
苑圆摇回溯法	缘
第 愿章摇动态程序设计方法类的例题	缘怨
愿源摇过河卒	苑园
愿源摇工作安排	苑圆
愿源摇烦人的青蛙	苑远
愿源摇颁奖典礼	愿园
愿源摇贪吃的九头龙	愿苑
第 怨章摇模拟策略类的例题	怨猿
怨源摇灭鼠行动	怨猿

怨谣调皮的小孩	员源
怨谣两支竿	员圆
第 员章谣“贪心法”类的例题	员圆
员谣均分纸牌	员圆
员谣月亮森林	员原
第 员章谣“构造法”类的例题	员怨
员谣乌托邦	员圆
员谣载的压缩	员源
员谣贝奇方块	员员
员谣新俄罗斯方块	员愿

自由落体

问题描述

在高为 H 的天花板上有 n 个小球, 体积不计, 位置分别为 $0, 1, 2, \dots, n-1$ 。地面上有一个小车(长为 L , 高为 k , 距原点距离为 s_1)。已知小球下落距离计算公式为 $s = \frac{1}{2}gt^2$, 其中 t 为下落时间。地面上的小车以速度 v 前进。如图 1 所示。

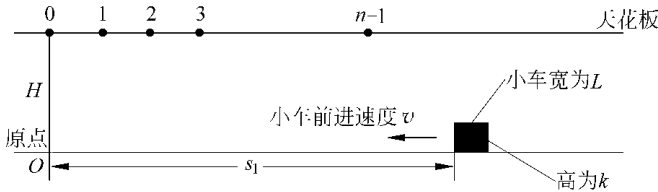


图 1

小车与所有小球同时开始运动, 当小球距小车的距离小于等于 L 时, 即认为被小车接收(小球落到地面后不能被接收)。请你计算出小车能接收到多少个小球。

输入

第一行: H, v, L, k, n, s_1

输出

小车能接收到的小球个数。

算法分析

这是分区联赛最容易失误的一道试题, 关键是弄清小车行程的物理意义和计算的精度误差。由题意可知, 车顶与天花板的距离为 $H - k$, 小球由天花板落至车顶所花费的时间为 $\sqrt{\frac{2(H-k)}{g}}$, 由天花板落至地面的时间为 $\sqrt{\frac{2H}{g}}$ 。小车与所有小球同时开始运动, 当小球距小车的距离小于等于 L 时, 即认为小球被小车接收(小球落到地面后不能被接收)。

小车在行驶了 $s_1 - Lv$ 路程后开始接收第 i 个小球, 由于第 i 个小球的编号上限为 $n-1$, 且在行驶 L 距离的过程中, 该小球都会落至车上, 因此第 i 个小球的编号 i 越自 $s_1 - Lv$ 至 $s_1 - Lv + L$, 如图 2 所示。公式中取下整的目的是为了舍去无用的小数部分。

小车在行驶了 $s_1 - Lv$ 路程后小球落至地面, 由于小车接收的最后一个小球的编号下限为 0 , 因此其编号为 i 越自 0 至 $s_1 - Lv + L$, 如图 3 所示。公式中加 L 且取上整的目的是为了得出小球落地前小车接收的最后一个小球编号。

第 10 章 字符串处理类的例题

数据结构专门研究各种数据的表示、数据的类型以及它们之间关系的集合,其研究范围主要包括各种数据结构的性质,即它们的逻辑结构、物理结构以及施于其上的操作。虽然 1994 年全国和国际信息学奥赛的每一道题都有如何为解题选择合适的数据结构的问题,都与数据结构相关,但是其中也有一些重点考核数据结构知识的试题,这些知识包括字符串处理和并查集与路径压缩。

10.1 字符串处理类的例题

在实际生活中我们经常要对一串元素进行操作。如信息检查系统、文字编辑系统、自然语言翻译系统以及音乐分析程序等,都是以字符串数据作为处理对象的。随着非数值的广泛应用,字符串的处理将越来越重要,因此竞赛中也经常出现一些字符串处理的试题。为此,我们不仅应该熟悉字符串处理的经典算法,例如 KMP 等,而且还应该熟悉 C++ 系统提供的库函数,并能驾轻就熟地应用于解题过程。

10.1.1 字符串近似查找

问题描述

设有 n 个单词的字典表($1 \leq n \leq 1000$)。计算某单词在字典表中的 1 种匹配情况(字典表中的单词和待匹配单词的长度上限为 100) :

1. 该单词在字典表中的序号 ;

2. 在字典表中仅有一个字符不匹配的单词序号 ;

3. 在字典表中多或少一个字符(其余字符匹配)的单词序号 ;

4. 其他情况。

当查找时有多个单词符合条件,仅要求第 1 个单词的序号即可。

输入文件

输入文件名为 10.1.in 文件的格式如下 :

1. 字典表的单词数)

2. 行,每行一个单词

3. 待匹配单词

输出文件

输出文件名 10.1.out 格式如下 :

1.

2.

3.

· 源

路径压缩后,寻找并查集元素的时间复杂度降低为 $O(n)$ 。

1995年的全国赛曾经出过一道并查集与路径压缩的试题(食物链),引起了了对于并查集的讨论。无独有偶,1996年的全国赛又出了同类型的一道试题,但增加了计算并查集中元素间相对位置关系的要求,把讨论又引深了一步。

1.1 摇摇银河英雄传说

问题描述

公元 1500 年,地球居民迁移至金牛座 α 第二行星,在那里发表银河联邦创立宣言,同年改元为宇宙历元年,并开始向银河系深处拓展。

宇宙历 795 年,银河系的两大军事集团在巴米利恩星域爆发战争。泰山压顶集团派宇宙舰队司令莱因哈特率领 10 万余艘战舰出征,气吞山河集团点名将杨威利组织麾下 10 万艘战舰迎敌。

杨威利擅长排兵布阵,巧妙运用各种战术屡次以少胜多,难免滋生骄气。在这次决战中,他将巴米利恩星域战场划分成 10 列,每列依次编号为 1, 2, ..., 10。之后,他把自己的战舰也依次编号为 1, 2, ..., 10, 让第 1 号战舰处于第 1 列(1 超越 1, 2, ..., 10)形成“一字长蛇阵”,诱敌深入,这是初始阵形。当进犯之敌到达时,杨威利会多次发布合并指令,将大部分战舰集中在某几列上,实施密集攻击。合并指令为 $M: a, b$, 含义为让第 a 号战舰所在的整个战舰队列,作为一个整体(头在前尾在后)接至第 b 号战舰所在的战舰队列的尾部。显然战舰队列是由处于同一列的一个或多个战舰组成的。合并指令的执行结果会使队列增大。

然而,老谋深算的莱因哈特早已在战略上取得了主动。在交战中,他可以通过庞大的情报网络随时监听杨威利的舰队调动指令。

在杨威利发布指令调动舰队的同时,莱因哈特为了及时了解当前杨威利的战舰分布情况,也会发出一些询问指令 $A: a, b$ 。该指令的意思是,询问电脑,杨威利的第 a 号战舰与第 b 号战舰当前是否在同一列中,如果在同一列中,那么它们之间布置有多少战舰。

作为一个资深的高级程序设计员,你被要求编写程序分析杨威利的指令,以及回答莱因哈特的询问。

最终的决战已经展开,银河系的历史又翻过了一页……

输入文件

输入文件 `input.txt` 的第 1 行有一个整数 N ($1 \leq N \leq 10000$), 表示总共有 N 条指令。

以下有 N 行,每行有一条指令。指令有如下两种格式。

$M: a, b$ 和 $A: a, b$ 是两个整数 ($1 \leq a, b \leq N$), 表示指令涉及的战舰编号。该指令是莱因哈特窃听到的杨威利发布的舰队调动指令,并且保证第 a 号战舰与第 b 号战舰不在同一列。

$A: a, b$ 和 $A: b, a$ 是两个整数 ($1 \leq a, b \leq N$), 表示指令涉及的战舰编号。该指令是莱因哈特发布的询问指令。

输出文件

输出文件为 `output.txt`。你的程序应当依次对输入的每一条指令进行分析和处理:

如果是杨威利发布的舰队调动指令,则表示舰队排列发生了变化,你的程序要注意到这一点,但是不要输出任何信息;

如果是莱因哈特发布的询问指令,你的程序要输出一行,仅包含一个整数,表示在同一列上,第 猿号战舰与第 猿号战舰之间布置的战舰数目。如果第 猿号战舰与第 猿号战舰当前不在同一列上,则输出 原猿

输入输出样例

输入

源

酝猿猿

悦猿猿

酝猿源

悦源猿

输出

原猿

猿

样例说明

战舰位置图:如表 猿猿所示。表格中阿拉伯数字表示战舰编号。

摇摇表猿猿

	第 猿列	第 猿列	第 猿列	第 源列
初始时	猿	猿	猿	源
酝猿猿		猿	猿 猿	源
悦猿猿	猿号战舰与 猿号战舰不在同一列,因此输出 原猿				
酝猿源	猿			源 猿 猿
悦源猿	源号战舰与 猿号战舰之间仅布置了一艘战舰,编号为 猿输出 猿				

算法分析

同一列的战舰组成一个并查集,集合中的一个结点对应一艘战舰。在集合中,我们以当前列的第一艘战舰作为集合的代表元。并查集的数据类型采用树型,树的根结点即为集合的代表元。为了使查询的效率达到最优,我们进行了路径压缩的优化:首先找到树根,然后将路径上所有结点的父结点改为根,使得树的深度为 猿

问题是,试题不仅要求判别两个结点是否在同一个集合(即两艘战舰是否在同一列),而且还要求计算结点在有序集合的位置(即每一艘战舰相隔同列的第一艘战舰几个位置,简称相对位置)。设

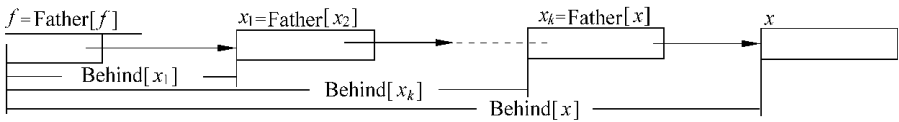


图 10.10

合并指令

按照题意,合并指令为 $\text{Merge}(u, v)$, 它的含义是让战舰 u 所在的整个战舰队列, 作为一个整体(头在前,尾在后)接至战舰 v 所在的战舰队列的尾部。显然两个队列合并成同一列后,其集合的代表元为结点 v 所在树的根结点 u (即 u 所在树的根结点 u 的父指针指向 u)。由于合并后 v 所在的集合在前, u 所在的集合在后, 因此 u 的相对位置为合并前 v 所在集合的结点数 $\text{Count}(v)$, 新集合的结点数为原来两个集合结点数的和 $(\text{Count}(u) + \text{Count}(v))$, 如图 10.11 所示。

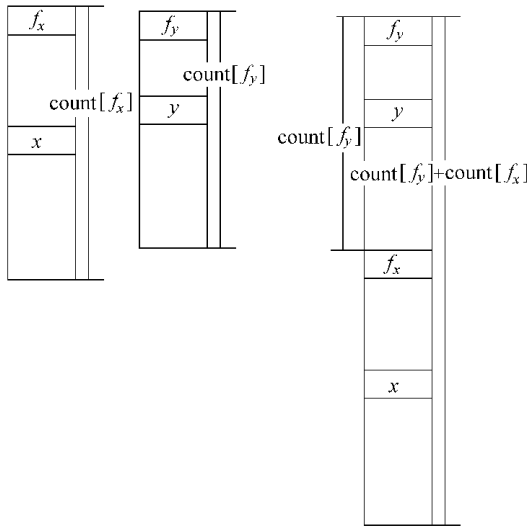


图 10.11

$\text{Merge}(u, v)$;	{把 u 所在的集合合并入 v 所在的集合}
$\text{Count}(u)$;	{计算 u 所在树的根结点 u }
$\text{Count}(v)$;	{计算 v 所在树的根结点 v }
$\text{Father}(u) \leftarrow v$;	{将 u 的父结点设为 v }
$\text{Count}(u) \leftarrow \text{Count}(u) + \text{Count}(v)$;	{计算 u 的相对位置为 $\text{Count}(v)$ }
$\text{Count}(v) \leftarrow \text{Count}(u)$;	{计算新集合的结点数}
$\text{Merge}(u, v)$;	