

国际大学生程序设计竞赛指南

ACM

Association for Computing Machinery

程序设计

曾棕根 编著



北京大学出版社
PEKING UNIVERSITY PRESS

国际大学生程序设计竞赛指南

ACM 程序设计

曾棕根 编著



北京大学出版社
PEKING UNIVERSITY PRESS

内 容 简 介

本书详细讲解了 ACM 国际大学生程序设计竞赛 (ACM/ICPC) 编程、调试方法, 以及提高时间、空间性能的策略, 并充分利用了 C++ 泛型编程的高效率、规范化的特性, 全部采用 C++ 泛型编程。

第 1 章讲解了 ACM 程序设计入门知识; 第 2 章讲解了 C++ 泛型编程的容器、迭代器和常用算法; 第 3 章讲解了 ACM 程序设计的基本编程技巧; 第 4 章讲解了 50 道原版 ACM 竞赛题的解题思路, 并配有 C++ 泛型编程参考答案和题目的中文翻译。

本书是一本专门针对 ACM 国际大学生程序设计竞赛而编写的入门教程, 适合参加 ACM/ICPC 的大学生和 C++ 编程爱好者学习, 对 ACM/ICPC 竞赛教练也具有一定的指导作用。

图书在版编目(CIP)数据

ACM 程序设计/曾棕根编著. —北京: 北京大学出版社, 2008.11
(国际大学生程序设计竞赛指南)

ISBN 978-7-301-14344-5

I. A… II. 曾… III. 程序设计—教材 IV. TP311.1

中国版本图书馆 CIP 数据核字(2008)第 158901 号

书 名: ACM 程序设计

著作责任者: 曾棕根 编著

策划编辑: 胡伟晔

责任编辑: 胡伟晔

标准书号: ISBN 978-7-301-14344-5/TP·0973

出 版 者: 北京大学出版社

地 址: 北京市海淀区成府路 205 号 100871

网 址: <http://www.pup.cn>

电 话: 邮购部 62752015 发行部 62750672 编辑部 62765126 出版部 62754962

电子邮箱: xxjs@pup.pku.edu.cn

印 刷 者: 河北滦县鑫华书刊印刷厂

发 行 者: 北京大学出版社

经 销 者: 新华书店

787 毫米×1092 毫米 16 开本 18 印张 427 千字

2008 年 11 月第 1 版 2008 年 11 月第 1 次印刷

定 价: 34.00 元

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有, 侵权必究

举报电话: 010-62752024

电子邮箱: fd@pup.pku.edu.cn

前 言

ACM 国际大学生程序设计竞赛 (ACM International Collegiate Programming Contest, ACM/ICPC) 由 ACM (Association for Computing Machinery, 美国计算机协会) 主办, 是世界上公认的规模最大、水平最高、影响最广的国际大学生程序设计竞赛。

由于 ACM 大学生程序设计竞赛能迅速提高学生的程序设计能力和团队协作水平, 又能有效地提升学校程序设计教学水平和质量, 促进校际交流与竞争, 近年来, ACM 大学生程序设计竞赛在我国得到了大规模的推广, 各个高校都十分重视。然而, 由于该赛事是个新鲜事物, 且 ACM 程序设计有其特殊的规律, 很多高校难以适应。

为了尽快解决这个问题, 笔者将近三年来潜心钻研 ACM 竞赛的相关资料整理出来, 供大家分享。经实践证明, 本书能在两个月内将程序设计的初学者培养成 ACM 竞赛高手。

本书以 C++ 泛型编程的应用为主线, 一步一步带领读者登上 ACM 程序设计的殿堂。第 1 章讲解了 ACM 程序设计入门知识; 第 2 章讲解了 C++ 泛型编程的容器、迭代器和常用算法; 第 3 章讲解了 ACM 程序设计的基本编程技巧; 第 4 章讲解了 50 道原版 ACM 竞赛题的解题思路, 并配有 C++ 泛型编程参考答案和题目的中文翻译。

本书的出版, 首先得感谢我的导师浙江师范大学数理信息学院王基一教授和丁革建高工、浙江工业大学信息学院钱能教授对笔者学术工作的不断启发与鼓励; 感谢北京大学出版社的大力支持!

ACM 程序设计要求很高, 由于时间仓促, 笔者水平有限, 书中定有许多不足甚至错误之处, 恳请广大专家、读者批评指正! 笔者邮箱 zjnuken@126.com, 个人主页 <http://www.zzg.3126.net>。

曾棕根

2008 年 10 月

目 录

第 1 章 ACM 程序设计入门	1
1.1 ACM/ICPC 简介	1
1.1.1 历史	1
1.1.2 简要规则	1
1.1.3 区域和全球决赛	2
1.1.4 历届冠军	2
1.1.5 在线评测系统 (Online Judge)	3
1.1.6 试题样例	4
1.2 用 VC++ 编写控制台程序	4
1.2.1 例题	4
1.2.2 操作	4
1.3 ACM 竞赛本机调试方法	10
1.3.1 竞赛样题	10
1.3.2 本机调试步骤	10
1.4 浙江大学在线评测系统 (ZOJ) 在线实验	12
1.4.1 竞赛样题	12
1.4.2 提交代码	13
第 2 章 C++STL 泛型编程	16
2.1 C++STL 概述	16
2.1.1 C++STL 的实现版本	16
2.1.2 C++STL 组件	16
2.1.3 C++STL 泛型编程示例	17
2.1.4 VC++6.0 泛型编程	18
2.2 vector 向量容器	18
2.2.1 创建 vector 对象	19
2.2.2 尾部元素扩张	19
2.2.3 下标方式访问 vector 元素	20
2.2.4 用迭代器访问 vector 元素	20
2.2.5 元素的插入	21
2.2.6 元素的删除	21

2.2.7 使用 reverse 反向排列算法	22
2.2.8 使用 sort 算法对向量元素 排序	23
2.2.9 向量的大小	25
2.3 string 基本字符系列容器	26
2.3.1 创建 string 对象	26
2.3.2 给 string 对象赋值	26
2.3.3 从 string 对象尾部添加字符	27
2.3.4 从 string 对象尾部追加 字符串	28
2.3.5 给 string 对象插入字符	28
2.3.6 访问 string 对象的元素	29
2.3.7 删除 string 对象的元素	29
2.3.8 返回 string 对象的长度	30
2.3.9 替换 string 对象的字符	31
2.3.10 搜索 string 对象的元素或 子串	31
2.3.11 string 对象的比较	32
2.3.12 用 reverse 反向排序 string 对象	32
2.3.13 string 对象作为 vector 元素	33
2.3.14 string 类型的数字化处理	33
2.3.15 string 对象与字符数组互 操作	34
2.3.16 string 对象与 sscanf 函数	35
2.3.17 string 对象与数值相互转换	36
2.4 set 集合容器	37
2.4.1 创建 set 集合对象	37
2.4.2 元素的插入与中序遍历	38
2.4.3 元素的反向遍历	38
2.4.4 元素的删除	39
2.4.5 元素的检索	40
2.4.6 自定义比较函数	41



2.5	multiset 多重集合容器.....	43	2.12	queue 队列容器.....	77
2.5.1	multiset 元素的插入.....	43	2.13	priority_queue 优先队列容器.....	78
2.5.2	multiset 元素的删除.....	44	2.13.1	优先队列的使用方法.....	79
2.5.3	查找元素.....	45	2.13.2	重载“<”操作符来定义 优先级.....	79
2.6	map 映照容器.....	46	2.13.3	重载“()”操作符来定义 优先级.....	81
2.6.1	map 创建、元素插入和遍历 访问.....	46	第3章 ACM 程序设计基础.....	82	
2.6.2	删除元素.....	47	3.1	读入一个参数.....	82
2.6.3	元素反向遍历.....	48	3.1.1	链接地址.....	82
2.6.4	元素的搜索.....	49	3.1.2	题目内容.....	82
2.6.5	自定义比较函数.....	49	3.1.3	参考答案.....	82
2.6.6	用 map 实现数字分离.....	51	3.2	读入两个参数.....	83
2.6.7	数字映照字符的 map 写法.....	52	3.2.1	链接地址.....	83
2.7	multimap 多重映照容器.....	53	3.2.2	题目内容.....	83
2.7.1	multimap 对象创建、元素 插入.....	53	3.2.3	参考答案.....	83
2.7.2	元素的删除.....	54	3.3	1!到 n!的和.....	83
2.7.3	元素的查找.....	55	3.3.1	链接地址.....	83
2.8	deque 双端队列容器.....	56	3.3.2	题目内容.....	84
2.8.1	创建 deque 对象.....	57	3.3.3	参考答案.....	84
2.8.2	插入元素.....	57	3.4	等比数列.....	84
2.8.3	前向遍历.....	59	3.4.1	链接地址.....	84
2.8.4	反向遍历.....	60	3.4.2	题目内容.....	84
2.8.5	删除元素.....	60	3.4.3	参考答案.....	85
2.9	list 双向链表容器.....	63	3.5	菲波那契数.....	86
2.9.1	创建 list 对象.....	64	3.5.1	链接地址.....	86
2.9.2	元素插入和遍历.....	64	3.5.2	题目内容.....	86
2.9.3	反向遍历.....	65	3.5.3	参考答案.....	86
2.9.4	元素删除.....	65	3.6	最大公约数.....	87
2.9.5	元素查找.....	68	3.6.1	链接地址.....	87
2.9.6	元素排序.....	70	3.6.2	题目内容.....	88
2.9.7	剔除连续重复元素.....	70	3.6.3	参考答案.....	88
2.10	bitset 位集合容器.....	71	3.7	最小公倍数.....	89
2.10.1	创建 bitset 对象.....	72	3.7.1	链接地址.....	89
2.10.2	设置元素值.....	72	3.7.2	题目内容.....	89
2.10.3	输出元素.....	74	3.7.3	参考答案.....	89
2.11	stack 堆栈容器.....	76			



3.8	平均数	90	3.17.2	题目内容	102
3.8.1	链接地址	90	3.17.3	参考答案	103
3.8.2	题目内容	90	第 4 章 ACM 程序设计实战 104		
3.8.3	参考答案	90	4.1	Quicksum	104
3.9	对称三位数素数	91	4.1.1	链接地址	104
3.9.1	链接地址	91	4.1.2	时空限制	104
3.9.2	题目内容	91	4.1.3	题目内容	104
3.9.3	参考答案	91	4.1.4	题目来源	105
3.10	十进制转换为二进制	92	4.1.5	解题思路	105
3.10.1	链接地址	92	4.1.6	参考答案	105
3.10.2	题目内容	92	4.1.7	汉语翻译	106
3.10.3	参考答案	93	4.2	IBM Minus One	107
3.11	列出完数	93	4.2.1	链接地址	107
3.11.1	链接地址	93	4.2.2	时空限制	107
3.11.2	题目内容	93	4.2.3	题目内容	107
3.11.3	参考答案	94	4.2.4	题目来源	108
3.12	12! 配对	95	4.2.5	解题思路	108
3.12.1	链接地址	95	4.2.6	参考答案	108
3.12.2	题目内容	95	4.2.7	汉语翻译	109
3.12.3	参考答案	95	4.3	Binary Numbers	110
3.13	五位以内的对称素数	96	4.3.1	链接地址	110
3.13.1	链接地址	96	4.3.2	时空限制	110
3.13.2	题目内容	96	4.3.3	题目内容	110
3.13.3	参考答案	96	4.3.4	题目来源	111
3.14	01 串排序	97	4.3.5	解题思路	111
3.14.1	链接地址	97	4.3.6	参考答案	111
3.14.2	题目内容	97	4.3.7	汉语翻译	112
3.14.3	参考答案	98	4.4	Encoding	112
3.15	排列对称串	98	4.4.1	链接地址	112
3.15.1	链接地址	98	4.4.2	时空限制	112
3.15.2	题目内容	99	4.4.3	题目内容	113
3.15.3	参考答案	99	4.4.4	题目来源	113
3.16	按绩点排名	100	4.4.5	解题思路	113
3.16.1	链接地址	100	4.4.6	参考答案	113
3.16.2	题目内容	100	4.4.7	汉语翻译	114
3.16.3	参考答案	101	4.5	Look and Say	115
3.17	按 1 的个数排序	102	4.5.1	链接地址	115
3.17.1	链接地址	102			



4.5.2	时空限制	115	4.10	Geometry Made Simple	131
4.5.3	题目内容	115	4.10.1	链接地址	131
4.5.4	题目来源	116	4.10.2	时空限制	131
4.5.5	解题思路	116	4.10.3	题目内容	131
4.5.6	参考答案	116	4.10.4	题目来源	132
4.5.7	汉语翻译	117	4.10.5	解题思路	132
4.6	Abbreviation	118	4.10.6	参考答案	132
4.6.1	链接地址	118	4.10.7	汉语翻译	134
4.6.2	时空限制	118	4.11	Reverse Text	135
4.6.3	题目内容	118	4.11.1	链接地址	135
4.6.4	题目来源	119	4.11.2	时空限制	135
4.6.5	解题思路	119	4.11.3	题目内容	135
4.6.6	参考答案	119	4.11.4	题目来源	135
4.6.7	汉语翻译	120	4.11.5	解题思路	136
4.7	The Seven Percent Solution	122	4.11.6	参考答案	136
4.7.1	链接地址	122	4.11.7	汉语翻译	136
4.7.2	时空限制	122	4.12	Word Reversal	137
4.7.3	题目内容	122	4.12.1	链接地址	137
4.7.4	题目来源	123	4.12.2	时空限制	137
4.7.5	解题思路	123	4.12.3	题目内容	137
4.7.6	参考答案	123	4.12.4	题目来源	138
4.7.7	汉语翻译	124	4.12.5	解题思路	138
4.8	Digital Roots	125	4.12.6	参考答案	138
4.8.1	链接地址	125	4.12.7	汉语翻译	139
4.8.2	时空限制	125	4.13	A Simple Question of Chemistry	140
4.8.3	题目内容	125	4.13.1	链接地址	140
4.8.4	题目来源	126	4.13.2	时空限制	140
4.8.5	解题思路	126	4.13.3	题目内容	140
4.8.6	参考答案	126	4.13.4	题目来源	141
4.8.7	汉语翻译	127	4.13.5	解题思路	141
4.9	Box of Bricks	128	4.13.6	参考答案	141
4.9.1	链接地址	128	4.13.7	汉语翻译	142
4.9.2	时空限制	128	4.14	Adding Reversed Numbers	143
4.9.3	题目内容	128	4.14.1	链接地址	143
4.9.4	题目来源	129	4.14.2	时空限制	143
4.9.5	解题思路	129	4.14.3	题目内容	143
4.9.6	参考答案	129	4.14.4	题目来源	144
4.9.7	汉语翻译	130	4.14.5	解题思路	144

4.14.6	参考答案	144	4.19.4	题目来源	162
4.14.7	汉语翻译	146	4.19.5	解题思路	162
4.15	Image Transformation	146	4.19.6	参考答案	162
4.15.1	链接地址	146	4.19.7	汉语翻译	163
4.15.2	时空限制	147	4.20	Martian Addition	164
4.15.3	题目内容	147	4.20.1	链接地址	164
4.15.4	题目来源	148	4.20.2	时空限制	165
4.15.5	解题思路	148	4.20.3	题目内容	165
4.15.6	参考答案	148	4.20.4	题目来源	165
4.15.7	汉语翻译	149	4.20.5	解题思路	165
4.16	Beautiful Meadow	150	4.20.6	参考答案	166
4.16.1	链接地址	150	4.20.7	汉语翻译	168
4.16.2	时空限制	150	4.21	FatMouse' Trade	169
4.16.3	题目内容	150	4.21.1	链接地址	169
4.16.4	题目来源	152	4.21.2	时空限制	169
4.16.5	解题思路	152	4.21.3	题目内容	169
4.16.6	参考答案	152	4.21.4	题目来源	170
4.16.7	汉语翻译	153	4.21.5	解题指导	170
4.17	DNA Sorting	155	4.21.6	参考答案	170
4.17.1	链接地址	155	4.21.7	汉语翻译	171
4.17.2	时空限制	155	4.22	List the Books	172
4.17.3	题目内容	155	4.22.1	链接地址	172
4.17.4	题目来源	156	4.22.2	时空限制	172
4.17.5	解题思路	156	4.22.3	题目内容	172
4.17.6	参考答案	156	4.22.4	题目来源	173
4.17.7	汉语翻译	157	4.22.5	解题指导	173
4.18	Daffodil Number	158	4.22.6	参考答案	174
4.18.1	链接地址	158	4.22.7	汉语翻译	175
4.18.2	时空限制	159	4.23	Head-to-Head Match	176
4.18.3	题目内容	159	4.23.1	链接地址	176
4.18.4	题目来源	159	4.23.2	时空限制	177
4.18.5	解题指导	159	4.23.3	题目内容	177
4.18.6	参考答案	159	4.23.4	题目来源	177
4.18.7	汉语翻译	160	4.23.5	解题指导	177
4.19	Error Correction	160	4.23.6	参考答案	178
4.19.1	链接地址	160	4.23.7	汉语翻译	178
4.19.2	时空限制	161	4.24	Windows Message Queue	179
4.19.3	题目内容	161	4.24.1	链接地址	179



4.24.2	时空限制	179	4.28.7	汉语翻译	194
4.24.3	题目内容	179	4.29	Semi-Prime	195
4.24.4	题目来源	180	4.29.1	链接地址	195
4.24.5	解题指导	180	4.29.2	时空限制	195
4.24.6	参考答案	180	4.29.3	题目内容	196
4.24.7	汉语翻译	181	4.29.4	题目来源	196
4.25	Language of FatMouse	182	4.29.5	解题思路	196
4.25.1	链接地址	182	4.29.6	参考答案	197
4.25.2	时空限制	182	4.29.7	汉语翻译	198
4.25.3	题目内容	182	4.30	Beautiful Number	199
4.25.4	题目来源	183	4.30.1	链接地址	199
4.25.5	解题思路	183	4.30.2	时空限制	199
4.25.6	参考答案	183	4.30.3	题目内容	199
4.25.7	汉语翻译	184	4.30.4	题目来源	199
4.26	Palindromes	185	4.30.5	解题思路	199
4.26.1	链接地址	185	4.30.6	参考答案	200
4.26.2	时空限制	185	4.30.7	汉语翻译	200
4.26.3	题目内容	185	4.31	Phone List	201
4.26.4	题目来源	186	4.31.1	链接地址	201
4.26.5	解题思路	186	4.31.2	时空限制	201
4.26.6	参考答案	187	4.31.3	题目内容	201
4.26.7	汉语翻译	188	4.31.4	题目来源	202
4.27	Root of the Problem	189	4.31.5	解题思路	202
4.27.1	链接地址	189	4.31.6	参考答案	202
4.27.2	时空限制	189	4.31.7	汉语翻译	205
4.27.3	题目内容	189	4.32	Calendar	206
4.27.4	题目来源	189	4.32.1	链接地址	206
4.27.5	解题思路	189	4.32.2	时空限制	206
4.27.6	参考答案	190	4.32.3	题目内容	206
4.27.7	汉语翻译	190	4.32.4	题目来源	207
4.28	Magic Square	191	4.32.5	解题思路	207
4.28.1	链接地址	191	4.32.6	参考答案	207
4.28.2	时空限制	191	4.32.7	汉语翻译	210
4.28.3	题目内容	191	4.33	No Brainer	211
4.28.4	题目来源	192	4.33.1	链接地址	211
4.28.5	解题思路	192	4.33.2	时空限制	211
4.28.6	参考答案	193	4.33.3	题目内容	211
			4.33.4	题目来源	211



4.33.5	解题思路	211	4.38.3	题目内容	226
4.33.6	参考答案	212	4.38.4	题目来源	227
4.33.7	汉语翻译	212	4.38.5	解题思路	227
4.34	Quick Change	213	4.38.6	参考答案	227
4.34.1	链接地址	213	4.38.7	汉语翻译	228
4.34.2	时空限制	213	4.39	Champion of the Swordsmanship	229
4.34.3	题目内容	213	4.39.1	链接地址	229
4.34.4	题目来源	214	4.39.2	时空限制	229
4.34.5	解题思路	214	4.39.3	题目内容	229
4.34.6	参考答案	214	4.39.4	题目来源	230
4.34.7	汉语翻译	215	4.39.5	解题思路	230
4.35	Total Amount	215	4.39.6	参考答案	230
4.35.1	链接地址	215	4.39.7	汉语翻译	231
4.35.2	时空限制	215	4.40	Doubles	231
4.35.3	题目内容	216	4.40.1	链接地址	231
4.35.4	题目来源	216	4.40.2	时空限制	232
4.35.5	解题思路	216	4.40.3	题目内容	232
4.35.6	参考答案	217	4.40.4	题目来源	232
4.35.7	汉语翻译	219	4.40.5	解题思路	232
4.36	Electrical Outlets	220	4.40.6	参考答案	233
4.36.1	链接地址	220	4.40.7	汉语翻译	233
4.36.2	时空限制	220	4.41	File Searching	234
4.36.3	题目内容	220	4.41.1	链接地址	234
4.36.4	题目来源	221	4.41.2	时空限制	234
4.36.5	解题思路	221	4.41.3	题目内容	234
4.36.6	参考答案	221	4.41.4	题目来源	236
4.36.7	汉语翻译	222	4.41.5	解题思路	236
4.37	Speed Limit	223	4.41.6	参考答案	237
4.37.1	链接地址	223	4.41.7	汉语翻译	239
4.37.2	时空限制	223	4.42	Old Bill	240
4.37.3	题目内容	223	4.42.1	链接地址	240
4.37.4	题目来源	224	4.42.2	时空限制	240
4.37.5	解题思路	224	4.42.3	题目内容	240
4.37.6	参考答案	224	4.42.4	题目来源	241
4.37.7	汉语翻译	225	4.42.5	解题思路	241
4.38	Beat the Spread!	226	4.42.6	参考答案	241
4.38.1	链接地址	226	4.42.7	汉语翻译	243
4.38.2	时空限制	226			



4.43	Divisor Summation	244	4.47.2	时空限制	258
4.43.1	链接地址	244	4.47.3	题目内容	258
4.43.2	时空限制	244	4.47.4	题目来源	259
4.43.3	题目内容	244	4.47.5	解题思路	259
4.43.4	题目来源	244	4.47.6	参考答案	260
4.43.5	解题思路	245	4.47.7	汉语翻译	261
4.43.6	参考答案	245	4.48	Excuses, Excuses!	262
4.43.7	汉语翻译	246	4.48.1	链接地址	262
4.44	Easier Done Than Said?	246	4.48.2	时空限制	262
4.44.1	链接地址	246	4.48.3	题目内容	262
4.44.2	时空限制	246	4.48.4	题目来源	264
4.44.3	题目内容	246	4.48.5	解题思路	264
4.44.4	题目来源	247	4.48.6	参考答案	264
4.44.5	解题思路	248	4.48.7	汉语翻译	266
4.44.6	参考答案	248	4.49	Lowest Bit	268
4.44.7	汉语翻译	250	4.49.1	链接地址	268
4.45	Let the Balloon Rise	251	4.49.2	时空限制	268
4.45.1	链接地址	251	4.49.3	题目内容	268
4.45.2	时空限制	251	4.49.4	题目来源	268
4.45.3	题目内容	251	4.49.5	解题思路	269
4.45.4	题目来源	252	4.49.6	参考答案	269
4.45.5	解题思路	252	4.49.7	汉语翻译	269
4.45.6	参考答案	252	4.50	Longest Ordered Subsequence	270
4.45.7	汉语翻译	253	4.50.1	链接地址	270
4.46	The Hardest Problem Ever	254	4.50.2	时空限制	270
4.46.1	链接地址	254	4.50.3	题目内容	270
4.46.2	时空限制	254	4.50.4	题目来源	271
4.46.3	题目内容	254	4.50.5	解题思路	271
4.46.4	题目来源	255	4.50.6	参考答案	272
4.46.5	解题思路	255	4.50.7	汉语翻译	274
4.46.6	参考答案	255	参考文献	275	
4.46.7	汉语翻译	257			
4.47	Fibonacci Again	258			
4.47.1	链接地址	258			

第 1 章 ACM 程序设计入门

1.1 ACM/ICPC 简介

ACM 国际大学生程序设计大赛 (ACM International Collegiate Programming Contest, ACM/ICPC) 是由世界计算机权威组织——ACM (Association for Computing Machinery, 美国计算机协会) 主办, 是世界上公认的规模最大、水平最高的国际大学生程序设计竞赛, 素来被冠以“程序设计的奥林匹克”的尊称。

1.1.1 历史

竞赛的历史可以上溯到 1970 年, 当时在美国德克萨斯 A&M 大学举办了首届比赛。当时的主办方是 the Alpha Chapter of the UPE Computer Science Honor Society。作为一种全新的发现和培养计算机科学顶尖学生的方式, 竞赛很快得到美国和加拿大各大学的积极响应。1977 年, 在 ACM 计算机科学会议期间举办了首次总决赛, 并演变成为目前的一年一届的多国参与的国际性比赛。

最初几届比赛的参赛队伍主要来自美国和加拿大, 后来逐渐发展成为一项世界范围内的竞赛。特别是自 1997 年 IBM 开始赞助赛事之后, 赛事规模增长迅速。1997 年, 总共有来自 560 所大学的 840 支队伍参加比赛。而到了 2004 年, 这一数字迅速增加到 840 所大学的 4109 支队伍并以每年 10%~20% 的速度在增长。

1980 年, ACM 将竞赛的总部设在位于美国德克萨斯州的贝勒大学。

在赛事的早期, 冠军多为美国和加拿大的大学获得。而进入 20 世纪 90 年代后期以来, 俄罗斯和其他一些东欧国家的大学连夺数次冠军。来自中国大陆的上海交通大学代表队在 2002 年美国夏威夷第 26 届和 2005 年上海举行的第 29 届全球总决赛上两次夺得冠军。这也是目前为止亚洲大学在该竞赛上取得的最好成绩。赛事的竞争格局已经由最初的北美大学的一枝独秀演变成目前的亚欧对抗的局面。

1.1.2 简要规则

ACM/ICPC 以团队的形式代表各学校参赛, 每队由 3 名队员组成。每位队员必须是入校 5 年内的在校大学生, 最多可以参加 2 次全球总决赛和 4 次区域选拔赛。比赛期间, 每队使用 1 台电脑, 在 5 个小时内使用 C、C++、Pascal 或 Java 中的一种语言编写程序解决 8 或 10 个问题 (区域选拔赛通常是 8 题, 全球总决赛是 10 题)。程序完成之后提交给在线评测系统 (Online Judge, OJ 系统) 去运行, 运行的结果会判定为正确或错误并及时通知参赛队。每队在正确完成一题后, 组织者将在其位置上升起一只代表该题颜色的气球。

最后的获胜者为正确解答题目最多且总用时最少的队伍。每道试题用时将从竞赛开始



到试题解答被判定为正确为止,其间每一次提交运行结果被判错误的话将被加罚 20 分钟时间,未正确解答的试题不记时。例如:A、B 两队都正确完成两道题目,其中 A 队提交这两题的时间分别是比赛开始后 1h 和 2h45min, B 队为 1h20min 和 2h,但 B 队有一题提交了两次。这样 A 队的总用时为 3h45min,而 B 队为 3h40min,所以 B 队以总用时少而获胜。

1.1.3 区域和全球决赛

与其他计算机程序竞赛(例如国际信息学奥林匹克,IOI)相比,ACM/ICPC 的特点在于其题量大,每队需要 5 小时内完成 8 道题目,甚至更多。另外一支队伍 3 名队员却只有 1 台电脑,使得时间显得更为紧张。因此除了扎实的专业水平外,良好的团队协作和心理素质同样是获胜的关键。

赛事由各大洲区域预赛和全球总决赛两个阶段组成。各预赛区第一名自动获得参加全球总决赛的资格。决赛安排在每年的 3—4 月举行,而区域预赛一般安排在上一年度的 9—12 月举行。一个大学可以有多个队伍参加区域预赛,但只能有一支队伍参加全球总决赛。

全球总决赛第一名将获得奖杯一座。另外,成绩靠前的参赛队伍也将获得金、银和铜牌。而解题数在中等以下的队伍会得到确认但不会进行排名。

1.1.4 历届冠军

下表列出了自 1977 年以来,截至 2005 年历年全球总决赛的冠军。

ACM 竞赛历年全球冠军

年份	总决赛地点	冠军大学	国家
2005	中国上海	上海交通大学	中国
2004	捷克布拉格	圣彼得堡光学与精密仪器学院	俄罗斯
2003	美国洛杉矶	华沙大学	波兰
2002	美国夏威夷	上海交通大学	中国
2001	加拿大温哥华	国立圣彼得堡大学	俄罗斯
2000	美国奥兰多	国立圣彼得堡大学	俄罗斯
1999	荷兰爱因霍温	滑铁卢大学	加拿大
1998	美国亚特兰大	查尔斯大学	捷克
1997	美国圣何塞	哈维玛德大学	美国
1996	美国费城	加州大学伯克利分校	美国
1995	美国纳什维尔	Albert-Ludwigs-Universitat Freiburg (弗莱堡大学)	德国
1994	美国菲尼克斯	滑铁卢大学	加拿大
1993	美国印第安纳波利斯	哈佛大学	美国

续表

年份	总决赛地点	冠军大学	国家
1992	美国堪萨斯城	墨尔本大学	澳大利亚
1991	美国圣安东尼奥	斯坦福大学	美国
1990	美国华盛顿	奥塔哥大学	新西兰
1989	美国路易斯维尔	加州大学洛杉矶分校	美国
1988	美国亚特兰大	加州理工学院	美国
1987	美国圣路易斯	斯坦福大学	美国
1986	美国辛辛那提	加州理工学院	美国
1985	美国新奥尔良	斯坦福大学	美国
1984	美国费城	约翰霍普金斯大学	美国
1983	美国墨尔本	内布拉斯加大学	美国
1982	美国印第安纳波利斯	贝勒大学	美国
1981	美国圣路易斯	密苏里大学罗拉分校	美国
1980	美国堪萨斯城	华盛顿大学圣路易斯分校	美国
1979	美国代顿	华盛顿大学圣路易斯分校	美国
1978	美国底特律	麻省理工学院	美国
1977	美国亚特兰大	密歇根州立大学	美国

1.1.5 在线评测系统 (Online Judge)

在线评测系统 (Online Judge, OJ) 上有大量的试题, 只需在在线评测系统上免费注册一个账号即可做题。

竞赛试题涵盖的范围很广, 大致划分如下: Direct (简单题), Computational Geometry (计算几何), Number Theory (数论), Combinatorics (组合数学), Search Techniques (搜索技术), Dynamic Programming (动态规划), Graph Theory (图论), Other (其他)。

国内比较著名的 OJ 系统有:

浙江大学 ACM 网站 (ZOJ): <http://acm.zju.edu.cn/>, 是国内最早也是最有名气的 OJ, 特点是数据比较刁钻, 经常会有你想不到的边界数据, 很能考验思维的全面性。

北京大学 ACM 网站 (POJ): <http://acm.pku.edu.cn/>, 建站较 ZOJ 晚一些, 但题目增加得很快, 可申请下载 POJ 源代码。

西班牙 Universidad de Valladolid 的 ACM 网站: <http://acm.uva.es>, 是世界上最大最有名的 OJ, 题目多, 类型杂, 数据也很刁钻。

俄罗斯乌拉尔大学的 ACM 网站: <http://acm.timus.ru>, 是一个老牌的 OJ, 题目不多, 但比较经典。

ICPC 官方网站: <http://icpc.baylor.edu/icpc/>, 上面会公布每年世界总决赛的排名及试题。



1.1.6 试题样例

题目名称: A + B Problem
链接地址: http://acm.zju.edu.cn/ 网上第 1001 题
Time limit: 1 Seconds Memory limit: 32768K
Calculate a + b
Input
The input will consist of a series of pairs of integers a and b, separated by a space, one pair of integers per line.
Output
For each pair of input integers a and b you should output the sum of a and b in one line, and with one line of output for each line in input.
Sample Input
1 5
Sample Output
6
Hint
Use + operator

1.2 用 VC++ 编写控制台程序

在 ACM 竞赛中, 一般使用 C++ 语言来编制程序。C++ 编译器有很多, 在 Windows 平台上有 Borland C++ Builder 5/6/2007, Bloodshed Dev-C++, Microsoft Visual C++ 6.0 (SP6), Microsoft Visual Studio .NET 中的 Visual C++ 等, 用不同的平台编写的程序, 一般在各个编译器上都能编译通过。所以, 选用何种 C++ 编译器, 纯属于个人喜好。

本书的程序都是采用 Microsoft Visual C++ 6.0 (SP6) 来编写的。下面讲讲 VC++ 6.0 的控制台程序的编写方法。

1.2.1 例题

编制一个 C++ 程序, 输入 a 和 b 两个整数, 输出这两个整数的和。

1.2.2 操作

(1) 运行 VC++ 6.0, 单击 File | New 菜单项, 再在弹出的 New 对话框的 Projects 选项卡中单击 Win32 Console Application (Win32 控制台应用程序), 如图 1-1 所示。

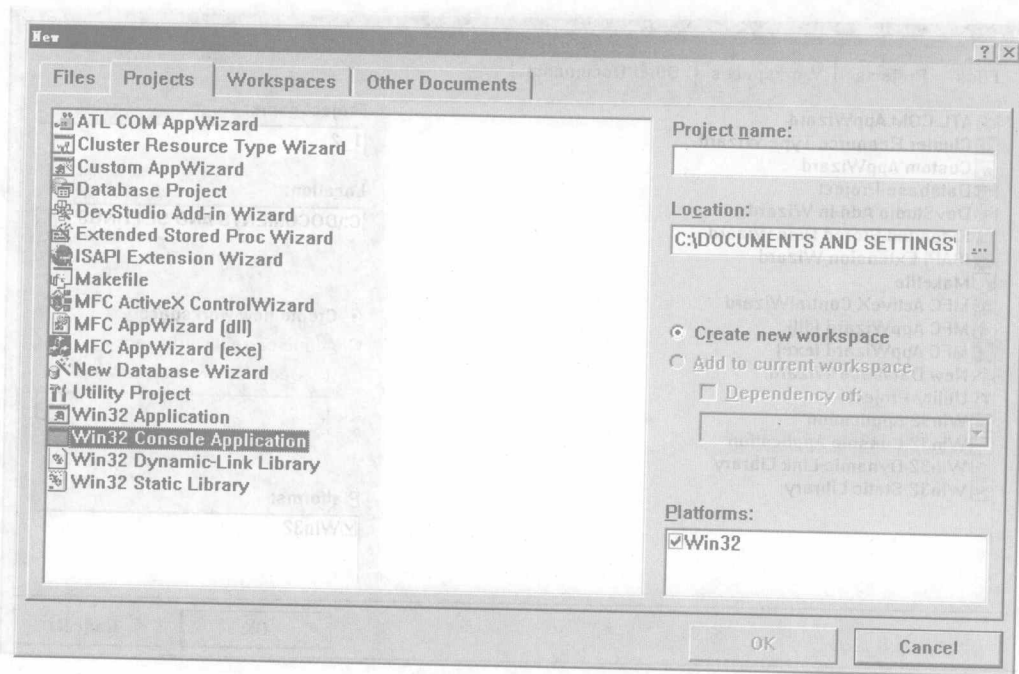


图 1-1 选定 Win32 控制台应用程序

↑(2)单击 New 对话框右上方 Location(位置)下边的...按钮,在弹出的 Choose Directory 对话框中单击文件夹列表中的“桌面”后再单击 OK 按钮,就把工程的位置设定在桌面上,即 C:\DOCUMENTS AND SETTINGS\ADMINISTRATOR\桌面\,如图 1-2 所示。

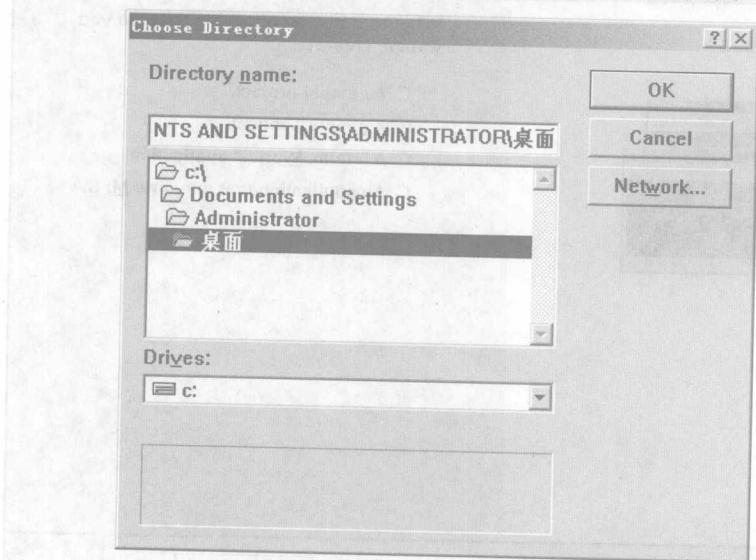


图 1-2 选择桌面位置

(3) 再在 New 对话框的右上方的 Project name 文本框中输入工程名称“1_2”,如图 1-3 所示。