

图灵程序设计丛书

TURING

挑战 程序设计竞赛

算法和数据结构

网罗算法和数据结构的**关键**知识点

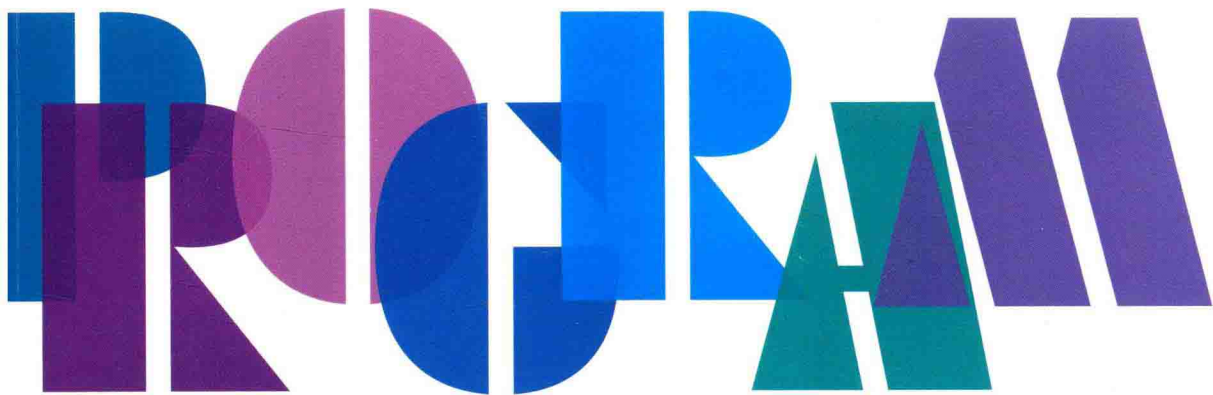
[日] 渡部有隆 / 著

[日] Ozy 秋叶拓哉 / 审

短码高手 Ozy (冈田佑一)

TopCoder Open 2013 Algorithm 第4名 iwi (秋叶拓哉)

支鹏浩 / 译



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

图灵程序设计丛书

TURING

挑战 程序设计竞赛 2

算法和数据结构

网罗算法和数据结构的关键知识点

[日] 渡部有隆 / 著

[日] Ozy 秋叶拓哉 / 审

短码高手 Ozy (冈田佑一)

TopCoder Open 2013 Algorithm 第4名 iwi (秋叶拓哉)

支鹏浩 / 译

PROGRAM

人民邮电出版社
北京

图书在版编目(CIP)数据

挑战程序设计竞赛. 2, 算法和数据结构 / (日) 渡部有隆著; 支鹏浩译. -- 北京: 人民邮电出版社, 2016.9

(图灵程序设计丛书)

ISBN 978-7-115-43161-5

I. ①挑… II. ①渡… ②支… III. ①程序设计
IV. ①TP311.1

中国版本图书馆CIP数据核字(2016)第180248号

Original Japanese title: PUROGURAMINGU KONTESUTO KOURYAKU
NO TAME NO ARUGORIZUMU TO DETA KOUZOU

© Yutaka Watanabe 2014

© Mynavi Publishing Corporation 2014

Original Japanese edition published by Mynavi Publishing Corporation.

Simplified Chinese translation rights arranged with Mynavi Publishing Corporation.
through The English Agency (Japan) Ltd. and Eric Yang Agency

本书中文简体字版由 Mynavi Publishing Corporation 授权人民邮电出版社独家出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

版权所有, 侵权必究。

内 容 提 要

本书分为准备篇、基础篇和应用篇三大部分, 借助在线评测系统 Aizu Online Judge 以及大量例题, 详细讲解了算法与复杂度、初等和高等排序、搜索、递归和分治法、动态规划法、二叉搜索树、堆、图、计算几何学、数论等与程序设计竞赛相关的算法和数据结构, 既可以作为挑战程序设计竞赛的参考书, 也可以用来引导初学者系统学习算法和数据结构的基础知识。

本书适合所有程序设计人员、程序设计竞赛爱好者以及高校计算机专业师生阅读。

-
- ◆ 著 [日] 渡部有隆
审 [日] Ozy 秋叶拓哉
译 支鹏浩
责任编辑 乐馨
执行编辑 高宇涵
责任印制 彭志环
- ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
三河市海波印务有限公司印刷
- ◆ 开本: 800×1000 1/16
印张: 26
字数: 580千字 2016年9月第1版
印数: 1-3 500册 2016年9月河北第1次印刷
著作权合同登记号 图字: 01-2016-1576号

定价: 79.00元

读者服务热线: (010)51095186 转 600 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广字第 8052号

前言

让我们一同踏上收集算法的旅程吧

■ 本书概要

本书是为攻克程序设计竞赛而撰写的参考书，书中讲解了大量竞赛相关的算法与数据结构。同时，本书又是一本入门图书，能带领初学者系统学习算法与数据结构的基础知识。

在程序设计竞赛中，优秀的数理能力是争取好名次的有力武器，但对于很多初学者而言，应用基础算法才是攻克眼前问题的最佳战略。也就是说，只要学会用“基础”解决问题，就能够提升自己的名次，在竞赛中获得更多乐趣。

另外，学习基础的过程并没有我们想象中那样痛苦乏味。将所学融会贯通、用技巧破解难题、收集算法网罗数据结构，这些乐趣都能在学习过程中体会得到。

为了让各位能在学习和解题过程中感受到上述乐趣，本书将借助在线评测（一种与程序设计竞赛系统相类似的自动评分系统）来讲解算法和数据结构。

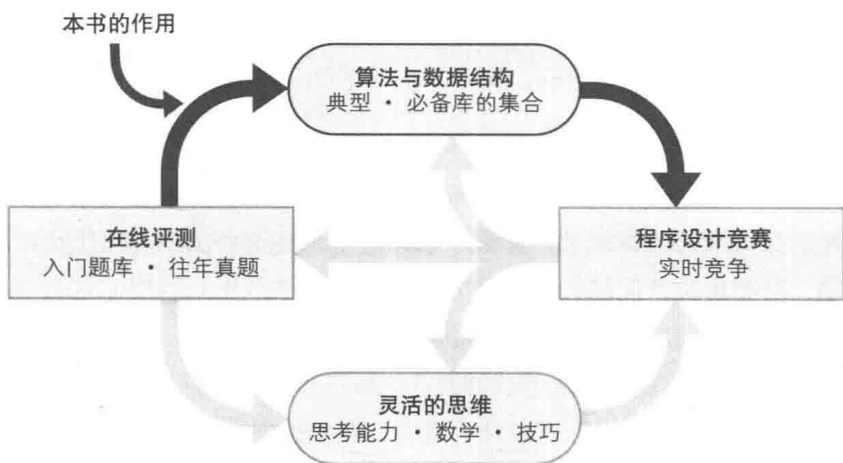


图1 本书的作用

通过在线评测掌握的算法与数据结构可以成为我们的知识储备，也可以作为库^①的一部分直接应用到程序设计竞赛之中。不过，要想在竞赛中名列前茅，还需要更加高超的算法以及灵活的思维和数理能力。本书虽然不能直接帮助各位在程序设计竞赛中跻身前列，但会在讲解在

^① 可直接调用的通用程序的集合称为库。

线评测使用技巧的同时，为各位简单介绍适用于竞赛的学习方法。

■ 致教职员工

本书从算法的概要与计算效率的概念入手，涵盖了信息处理技术人员、程序员必备的通用算法、数据结构的相关问题，以及对这些问题的讲解和参考答案。因此，本书不但是本专为程序设计竞赛服务的参考书，同时也是一本程序设计、算法与数据结构等相关科目的教材。

■ 有效运用在线评测

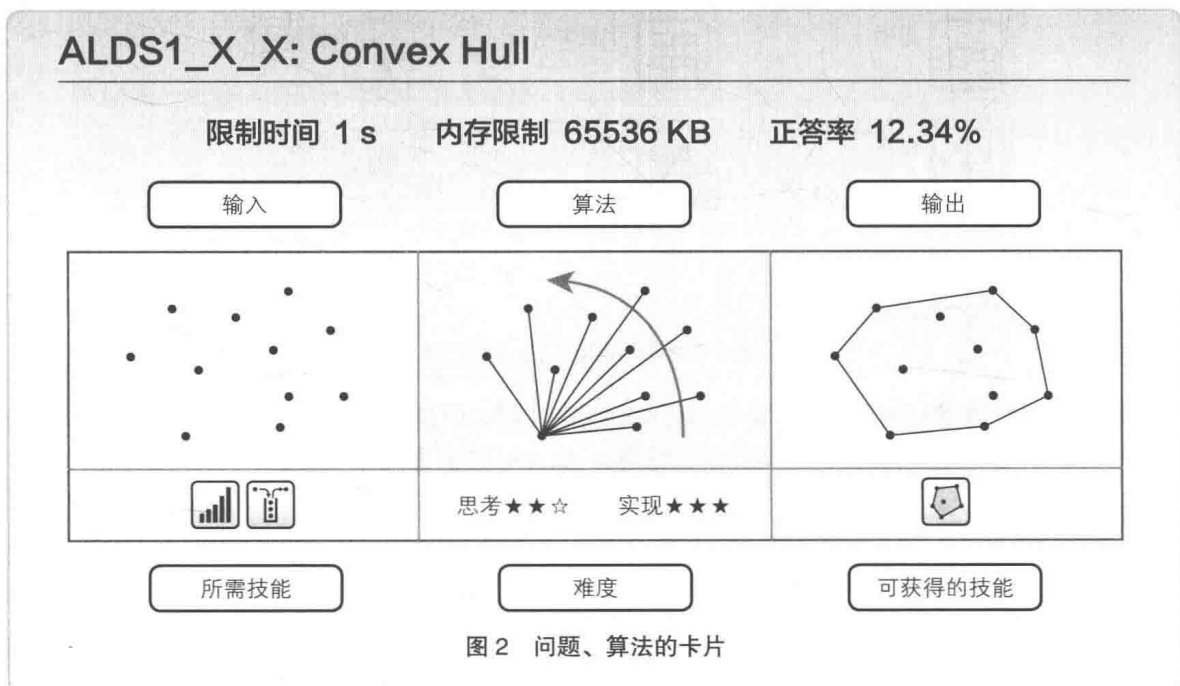
算法和数据结构与单纯的知识不同，只靠阅读并不能将其转化为自己的东西。因此，我们需要将例题的答案（代码）落实到环境中，通过实际运行来检验其正误与性能。不过，我们根据算法编写的程序往往会含有 Bug，或者算法效率达不到规格要求。这种情况下，在线评测会根据严格的测试数据（虽然我们也能亲自编写类似的测试数据，但这会浪费大量时间与精力）检查程序是否有缺陷，帮助我们在“正确实现程序”的过程中学习算法与数据结构。此外，反复学习还能给我们带来以下好处。

- ▶ 作为一名信息处理技术人员或程序员，必备的基础算法与数据结构的相关知识涉及面极广，而反复学习能帮助我们掌握这些知识
- ▶ 反复学习能帮助我们掌握程序员必备的技巧，具体说来就是准确理解文章要义，编写程序时能忠实于需求且极力避免 Bug。此外，在设计和编写程序时还能够兼顾计算机资源、计算效率和内存使用量等

另外，每当我们独立解决一个问题，看到在线评测给出“正确”的评判时，都会自然而然地感到些许喜悦。保持人们的积极性，寓教于乐，这也是在线评测的魅力所在。通过累积经验来解决新的问题，再将解决新问题的经验化为武器进一步挑战更高难度，这与游戏并没有什么两样。在这一过程中，我们可以把学到的技巧转变为自己的收藏品，久而久之，程序设计便会成为一种爱好。

■ 本书涉及的问题

本书的例题以在线评测系统中的各问题为原型，如图 2 所示，以卡片形式提出。



这种卡片中包含了问题的概要，具体由以下信息组成。

- ▶ **基本信息。**显示在线评测的问题 ID、题目、CPU 和内存限制、正答率等基本信息。其中限制时间是指所提交程序的执行时间
- ▶ **输入与输出。**用插图简要说明程序所需的输入以及输出结果
- ▶ **算法。**用插图简要说明解题所需的算法。标为“?”时需要各位自行拟定算法，各位可以将这个过程视为一种思考的乐趣
- ▶ **难度。**思考、实现的难度最高为 5 颗★。★记 1 分，☆记 0.5 分。“思考”分数越高算法越难，“实现”分数越高代码量越大。不过，代码量是指算法和数据结构中不使用标准库时的量
- ▶ **所需技能（图标）。**表示解决该问题所必需的前提技能。本书虽然不要求各位严格按照顺序解题，但还是推荐各位在掌握各问题所需技能之后再行挑战
- ▶ **可获得的技能（图标）。**解决该问题后可以学到的技能。在后面的问题中，这些技能都会出现在“所需技能”之中。通过本书可以获得的技能一览表见附录 1

除“变量”和“四则运算”这两个编程基础之外，阅读本书首先要具备如图 3 所示的七种技能。



图 3 阅读本书所需的前提技能

只要学习过任何一门编程语言（C/C++ 或 Java 等）的基础知识，这些技能就应该不在话下了。如果对这方面没有自信，建议各位先找一些入门书籍来预习一番。

本书涉及的都是信息处理中基础且通用的问题，其中一些问题在很多语言中已经被纳入程序库。不过，学习标准库的内部结构有助于我们对其性能和动作（能做什么不能做什么）有进一步了解，这对于使用库的程序员来说意义重大。

另一方面，本书将在各章节中适时穿插 C++ 标准模板库（STL）通用算法和数据结构的介绍，并将其与基本算法和数据结构相结合，以挑战一些略有难度的问题。

如何有效运用本书

本书各章的组成项目如图 4 所示（根据每章情况不同，第 1 项和第 4 项可能被省略）。

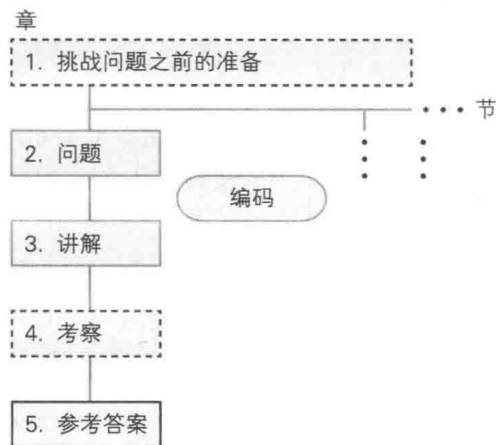


图 4 如何有效运用本书

在各章的开头部分，我们会先简要说明与该章主题相关的用语及概念，同时介绍一些基础算法与数据结构的概要。之后的各小节则分别包含下述项目。

“问题”就是实际挑战的例题，“讲解”是对解题时所用算法的细节以及实现方法的详细说明。各位请将这两部分视为“一对儿”，然后根据自己的情况决定何时开始编写代码。比如，

我们根据难度和经验构思了以下两种学习顺序。

► 问题→讲解→编码→考察和参考答案

本书涉及大量基础问题（帮助读者了解基础算法知识的问题），因此初学者往往很难在阅读问题之后立刻开始编写代码。这时不必勉强自己写出答案，可以在阅读过“问题”与“讲解”之后再挑战。如果无论如何都通不过在线评测，不妨看一下参考答案。另外，即便已经做出正确答案，也要记得与“考察”和“参考答案”进行对比，看看自己的实现方法有没有需要改善的地方。

► 问题→编码→讲解→考察和参考答案

阅读“问题”之后，如果觉得自己能够在没有任何提示的情况下解开问题，便可以直接尝试编码。如果能顺利通过在线评测，证明你已经掌握了这部分知识。不过，通过评测后也不要忘记查看“讲解”和“考察”，因为本书所给的解法或许能给各位带来新的发现。

“考察”部分中，我们会一同考察示例解法的复杂度，以及算法中值得一提的特征和需要注意的地方。

“参考答案”中将提供一份能实际通过在线测评的 C 或 C++ 语言代码，供各位读者参考。不过各位要清楚，每个程序设计问题的解法都不是唯一的，参考答案并不一定是最优秀的实现方法。

本书的支持网站

该网站上公布了本书的补充内容和勘误，请根据需要参考。

<http://www.ituring.com.cn/book/1739>

- 本书内容可灵活运用会津大学的 Aizu Online Judge（会津大学在线评测，简称 AOJ）
Aizu Online Judge（AOJ）<http://judge.u-aizu.ac.jp>

- AOJ 可免费使用，但是需要注册。详情请参考本书第 1 章中的内容

Aizu Online Judge Version 1.0 ©2004 - 2015 AIZU Competitive Programming Club,
Database Systems Lab. University of Aizu

- 本书中的内容仅以提供信息为目的。在编写本书的过程中，我们力求准确表述，但在本书的使用过程中，著者和 Mynavi 出版社、人民邮电出版社及译者均不对本书内容作任何保证，对于因对本书内容运用不当而导致的任何问题均不负责。对本书的使用由个人负责，请知悉
- 本书中的报道、产品名称和 URL 等均为截止至 2014 年 12 月的内容。在使用过程中，这些内容可能会有变更，请知悉
- 本书中的公司名称和产品名称等多为各公司注册商标或商标。正文中将省略 ©、®、™ 等符号

目录

第1部分 [准备篇]攻克程序设计竞赛的学习方法	1
第1章 有效运用在线评测系统	3
1.1 攻克程序设计竞赛的学习方法.....	3
1.2 什么是在线评测.....	7
1.3 用户注册.....	9
1.4 浏览问题.....	10
1.5 解答问题.....	12
1.6 个人页面.....	18
1.7 如何运用本书.....	19
第2部分 [基础篇]为程序设计竞赛做准备的算法与数据结构	21
第2章 算法与复杂度	23
2.1 算法是什么.....	23
2.2 问题与算法示例.....	23
2.3 伪代码.....	25
2.4 算法的效率.....	26
2.5 入门问题.....	28
第3章 初等排序	33
3.1 挑战问题之前——排序.....	33
3.2 插入排序法.....	35
3.3 冒泡排序法.....	40
3.4 选择排序法.....	44
3.5 稳定排序.....	48
3.6 希尔排序法.....	52
第4章 数据结构	57
4.1 挑战问题之前——什么是数据结构.....	57
4.2 栈.....	59
4.3 队列.....	64
4.4 链表.....	70
4.5 标准库的数据结构.....	77
4.6 数据结构的应用——计算面积.....	86

第 5 章 搜索	89
5.1 挑战问题之前——搜索	89
5.2 线性搜索	91
5.3 二分搜索	94
5.4 散列法	98
5.5 借助标准库搜索	102
5.6 搜索的应用——计算最优解	106
第 6 章 递归和分治法	109
6.1 挑战问题之前——递归与分治	109
6.2 穷举搜索	111
6.3 科赫曲线	114
第 7 章 高等排序	119
7.1 归并排序	120
7.2 分割	125
7.3 快速排序	129
7.4 计数排序	133
7.5 利用标准库排序	137
7.6 逆序数	139
7.7 最小成本排序	143
第 8 章 树	147
8.1 挑战问题之前——树结构	148
8.2 有根树的表达	150
8.3 二叉树的表达	154
8.4 树的遍历	159
8.5 树遍历的应用——树的重建	163
第 9 章 二叉搜索树	167
9.1 挑战问题之前——二叉搜索树	168
9.2 二叉搜索树——插入	169
9.3 二叉搜索树——搜索	174
9.4 二叉搜索树——删除	177
9.5 通过标准库管理集合	182
第 10 章 堆	189
10.1 挑战问题之前——堆	190
10.2 完全二叉树	191
10.3 最大 / 最小堆	193
10.4 优先级队列	197

10.5 通过标准库实现优先级队列	201
第 11 章 动态规划法	203
11.1 挑战问题之前——动态规划法的概念	203
11.2 斐波那契数列	204
11.3 最长公共子序列	208
11.4 矩阵链乘法	211
第 12 章 图	217
12.1 挑战问题之前——图	218
12.2 图的表示	221
12.3 深度优先搜索	224
12.4 广度优先搜索	232
12.5 连通分量	237
第 13 章 加权图	241
13.1 挑战问题之前——加权图	242
13.2 最小生成树	244
13.3 单源最短路径	249
第 3 部分 [应用篇]程序设计竞赛的必备程序库	261
第 14 章 高等数据结构	263
14.1 互质的集合	264
14.2 范围搜索	269
14.3 其他问题	278
第 15 章 高等图算法	279
15.1 所有点对间最短路径	280
15.2 拓扑排序	284
15.3 关节点	290
15.4 树的直径	295
15.5 最小生成树	299
15.6 其他问题	303
第 16 章 计算几何学	305
16.1 几何对象的基本元素与表现	306
16.2 直线的正交 / 平行判定	312
16.3 投影	314
16.4 映象	316
16.5 距离	317

16.6	逆时针方向	321
16.7	判断线段相交	324
16.8	线段的交点	326
16.9	圆与直线的交点	328
16.10	圆与圆的交点	331
16.11	点的内包	333
16.12	凸包	335
16.13	线段相交问题	339
16.14	其他问题	343
第 17 章	动态规划法	345
17.1	硬币问题	346
17.2	背包问题	349
17.3	最长递增子序列	353
17.4	最大正方形	357
17.5	最大长方形	360
17.6	其他问题	364
第 18 章	数论	367
18.1	质数检验	368
18.2	最大公约数	372
18.3	幂乘	376
18.4	其他问题	378
第 19 章	启发式搜索	381
19.1	八皇后问题	382
19.2	九宫格拼图	386
19.3	十六格拼图	391
附录		399
	通过本书可以获得的技能	400
	挑战以往的程序设计竞赛真题!	402
	参考文献	404

第 1 部分
[准备篇]
攻克程序设计竞赛的学习方法

在准备篇中，本书将为入门者以及准备挑战程序设计竞赛的读者介绍一些学习方法，以帮助各位提升名次。

- ▶ 本书作为一本算法与数据结构的教科书，与在线评测系统有着紧密的联系，这在市面上的同类书籍中是很少见的。亲手实现程序→接受在线评测系统的自动检测→通过评测后继续挑战下一问题，这一流程能有效帮助读者扎实地打好基础（秋叶）
- ▶ 我刚接触程序设计那阵子，基本上所有东西都要靠摸索，浪费了很多时间。随着在线评测系统的问世，人们学习新知识和技术并掌握其实现方法的过程得到了大幅缩短。相信本书运用的学习方法会成为今后的主流（Ozy）

第 1 章

有效运用在线评测系统

在线评测是一种 Web 系统，我们向其提交程序后，它能够自动检测该程序的正确性以及运行效率。使用它，我们就可以按照自己的步调，随时连接到网络上进行解题练习。本章将以本书使用的 Aizu Online Judge 为中心，为各位介绍在线评测系统，讲解其使用方法。

1.1 攻克程序设计竞赛的学习方法

■ 程序设计竞赛简介

程序设计竞赛种类繁多，其中最具代表性的有“根据某个主题开发应用，比拼创意与技术实力的竞赛”“周期较长的、制作游戏 AI（人工智能）的竞赛”以及“短时间内求解指定问题的竞赛”等。本书所涉及的程序设计竞赛以解题类竞赛为主。

程序设计竞赛大致以如下形式进行。

- ▶ 在限制时间内解答多个指定问题
- ▶ 根据解答正确的问题数或总分决定名次
- ▶ 正确数或得分相同时，答错较少或用时较短的团队名次更靠前

这种竞赛的题目难度覆盖面较广，入门者也可以参与挑战。另外，明确的评比规则使得该类竞赛排名更加精确直观。从教育方面来说，这类活动能有效地培养参与者的编程技术、思考能力、算法知识以及团队协作，教育效果十分显著。

首先介绍几个具有代表性的程序设计竞赛。

下述三个是以学生为对象的赛事。

- ▶ 国际信息学奥林匹克竞赛

<http://www.ioinformatics.org/>

国际科学奥林匹克竞赛之一，是面向中学生（个人）的国际大赛。各国经由国内竞赛选拔代表选手。信息学奥林匹克竞赛需要选手设计出十分优秀的算法，因此对数理能力要求极高，属于巅峰级的赛事。

▶ 电脑甲子园编程部门

<http://web-ext.u-aizu.ac.jp/pc-concours/>

日本面向高中生以及职高学生（三年级以下）的全国大赛，每年在会津大学召开。比赛为两人一组的团队战，由于每组只配备一台电脑，因此团队配合十分重要。该大赛题量甚至超过了国际信息学奥林匹克竞赛，问题大多考验选手对典型算法的应用以及实现能力。

▶ ACM-ICPC（国际大学生程序设计竞赛）

<http://icpc.baylor.edu/>

由计算机科学领域最具影响力的学会之一 ACM 主办，各国大学在此较量程序设计实力。比赛为三人一组的团队战，由于每组只配备一台电脑，因此团队配合十分重要。选手在各地举办的亚洲区预选赛中取得优异成绩便可参加世界大赛。该赛事不仅要求高超的算法设计技术，还十分考验实现能力和团队配合。

以下两个网站会定期举办竞赛，且没有参赛资格的限制。

▶ TopCoder

<http://www.topcoder.com/>

提供程序设计竞赛和软件众包^①服务的网站。竞赛分为几个类型，其中 SRM（Single Round Match）会定期举行，要求用户在一小时内解开算法相关问题。网站会纪录成绩，同时依照成绩给用户评级并分配相应颜色。如今已有 TopCoder 相关的攻略书出版，甚至还配备了相应的学习环境。

▶ AtCoder

<http://atcoder.jp/>

定期举办程序设计竞赛的日本网站，其中包含专门面向初学者的竞赛，刚刚接触程序设计的人也可轻松参加。另外，该网站上也不乏企业和学生志愿者举办的高端赛事。

这些竞赛中的问题涉及各个领域，并且全都可以通过编程来解决。选手将已完成的程序提交给裁判（评测系统）来判定对错。评测系统会通过测试代码严格检验算法的效率与正确性。这类竞赛要求参赛者更快更准确地完成下述几项工作。

- ▶ 理解问题（需求）
- ▶ 设计效率足够高的算法
- ▶ 编写程序
- ▶ 修正 Bug（尽量编写出没有 Bug 的程序）

^① 指企业将软件开发任务以竞赛形式发布，公开募集参与者的行为。