

Algorithm Course's Teaching and Practice Method Combined with ACM-ICPC for Soft Engineering

Chengzhang Qu*, Shaoqian Zhang

College of Information Engineering, Wuhan Business University, Wuhan Hubei
Email: quchengzhang@wbu.edu.cn

Received: Aug. 1st, 2018; accepted: Aug. 14th, 2018; published: Aug. 21st, 2018

Abstract

The algorithm course is the core course of software engineering, which is difficult in programming. The teaching and practice of the traditional algorithm course has certain drawbacks. This paper aims at the teaching situation of the algorithm design and analysis course of the normal university, combined with the relevant content of the ACM-ICPC program competition, and proposes reform methods of teaching and practicing for algorithm course by different student levels. Teaching and practice reform methods have employed an automatic Online Judge Platform in practice to greatly improve the practice time after the class. It also includes paying attention to students' thinking ability and computer-based algorithm evaluation methods, and cultivating students' algorithm design ability, programming skills, logical thinking ability and spirit of collaboration. The results have proved that this method can effectively improve students' programming and implementation ability, increase students' interest in learning, and have a positive effect on study atmosphere.

Keywords

ACM-ICPC, Soft Engineering, Algorithm Design and Analysis, Teaching Reform

融合ACM-ICPC的软工专业算法课程教学与实践

曲成璋*, 张少谦

武汉商学院信息工程学院, 湖北 武汉
Email: quchengzhang@wbu.edu.cn

*通讯作者。

收稿日期: 2018年8月1日; 录用日期: 2018年8月14日; 发布日期: 2018年8月21日

摘要

算法课程是软件工程专业核心课程, 属于程序设计类课程的较难内容。传统算法课程的教学、实践有一定的弊端, 本文针对二本院校的算法设计与分析课程的教学情况, 结合ACM-ICPC程序设计竞赛的相关内容, 对不同学生层次提出融合程序竞赛的算法课程教学、实践改革方法, 在实践方面引入自动在线判读平台大幅度的提高课下实践时间, 注重学生思维能力和计算机实现算法的考核方式, 培养学生的算法设计能力、编程技巧、逻辑思维能力以及良好的协作精神。实践证明, 此方法能切实提升学生的程序设计和实现能力, 提高学生学习兴趣, 并对学习风气有一定的引领规范作用。

关键词

ACM-ICPC, 软件工程, 算法设计与分析, 教学改革

Copyright © 2018 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

随着社会信息化时代的快速发展, 计算机类专业人才越来越成为社会信息化建设的核心人才, 随着大数据、人工智能、区块链等新型信息技术的爆发式发展, 对计算机类人才的需求会日益增大, 并对高水平计算机类人才的争夺会愈演愈烈。软件工程作为承载信息技术的核心基础计算机类专业, 是软件开发的核心, 不断融合最新的技术, 从技术和理念都不断更新进步, 使得此专业毕业学生在信息社会高速发展的今天有较好的发展前景。其中, 作为软件工程专业核心课程的算法课程由于其理论难度大, 实践要求高等特点, 传统的教学、实践、考核方式都亟待优化改革。本文从二本高校的软件工程专业中算法课程的教学与实践出发, 融合 ACM-ICPC 程序设计竞赛相关内容, 提出对算法课程教学与实践的改革优化方法。实践证明, 改革方法能提高学生学习算法课程的兴趣和能力, 并有效辅助提高程序设计竞赛的成绩。

ACM 国际大学生程序设计竞赛(ACM-ICPC: ACM International Collegiate Programming Contest)是由国际计算机界权威组织、美国计算机协会主办的世界公认的规模最大、水平最高的国际大学生程序设计竞赛, 旨在使大学生运用计算机来充分展示自己分析问题和解决问题的能力[1] [2] [3]。中国大陆从 1996 年开始设立 ACM-ICPC 亚洲区预选赛赛区。多年来, 各个大学组织纷纷参加相应赛事并获得奖项, 从大赛中涌现出一批编程明星, 为当今大数据、人工智能行业贡献了卓越的力量。今年中国大陆区新增河南赛区, 区域赛达到了 8 个, 极大地促进了这一赛事在中国的发展。我校从 2017 年开始初步尝试参加 ACM-ICPC 相关赛事, 作为一个二本院校参加此类比赛难度较大, 对学生、老师都是不小的锻炼, 取得的竞赛成绩有待进一步加强。另一方面, 参加 ACM-ICPC 程序设计竞赛有助于提高学生编程、解决问题、理解算法的兴趣, 对《算法设计与分析》课程的教学与实践有较好的补充与帮助作用。本文即从二本院校开展《算法设计与分析》课程出发, 融合 ACM-ICPC 程序设计竞赛, 简单探讨一下课程的教学与实践方面的方法策略。

《算法分析与设计》是一门计算机专业核心课程, 是计算机科学研究的重要领域之一, 也是核心难点课程。课程是程序设计、数据结构的后续课程, 是在学生掌握了相关编程基础和数据结构知识的前提下, 培养学生分析问题和解决问题的能力, 并通过算法设计及复杂性分析来实际编程解决相关问题, 提高学生实际解决问题的和编程实现的能力[1] [2]。算法课程具有理论深度大, 实践任务多等多个特性, 教授内容综合了基础知识、计算机数学、编程实现等多门学科, 对学生计算机专业素养的提升起到核心的作用(图 1)。学生如果把这一门集应用性、创造性及实践性融为一体的综合性课程学号, 会打下坚实的计算机专业基础, 对学习其他计算机类专业课程有很好的帮助。

目前, 结合程序设计竞赛来对相关课程进行优化改革的高校也有不少。利用组织参加大学生程序设计竞赛、蓝桥杯、高校天梯赛等程序设计类竞赛, 对现有数据结构、算法等程序设计课程进行优化改革, 提高教学效果, 拓展培养人才思路, 不少学校的教师和教练都做了很多很好的工作[4]-[9]。然而, 对于一些国内非 985、211 院校, 其学生素质、自觉性和基础都无法与顶尖院校相比, 因此, 有必要针对特定的学生层次, 制定面相不同基础同学的以竞赛为驱动的课程改革方案。本文分讲授和实践两部分来提出融合 ACM-ICPC 程序设计竞赛的软工算法课程的教学方法。学生参加基于 ACM-ICPC 的程序设计类学习路线图如图 1 所示。

2. 教学方面

目前大多数学校开设该课程的教学方法主要以讲解为主, 这也是算法课程有较深的理论性所决定的, 在课堂上讲清楚、学习明白相关算法理论, 是算法课程要求的第一步, 只有在弄懂理论的前提下才能有效的进行编程实现。《算法设计与分析》课程在本校软件工程专业中的学时数是 48 学时, 其中包含 32 学时理论课和 16 学时实验课, 从本人实际的教学过程来看, 只能达到一般的教学要求, 即介绍几种典型解决问题的策略, 包括分治法、动态规划、贪心选择、回溯法、分支限界法等。对于本校大二下的软件工程专业学生而言, 前期程序设计类课程和数据结构课程虽然学过, 但缺乏大量有效实践, 学生往往是在相关理论课上学懂相关知识, 而在实验课上仅仅完成最基本的实验, 课后没有花时间精力进行实践, 致使一部分同学基础知识较薄弱, 在上课的时候不能讲的太快。

算法课程本身的难度也是导致传统以讲课为主的教学方式有弊端的客观因素之一。通常对经典算法只在数学模型和数据结构上片面地解释给学生, 在实践环节只盲目地验证算法, 而对该算法的运行效率、测试数据的规模以及实际应用场景则很少考虑。学生的学习则主要以理解和记忆的继承式学习为主, 虽然记住了大量的算法理论, 但没有“理解”和“消化”, 不能灵活运用算法。

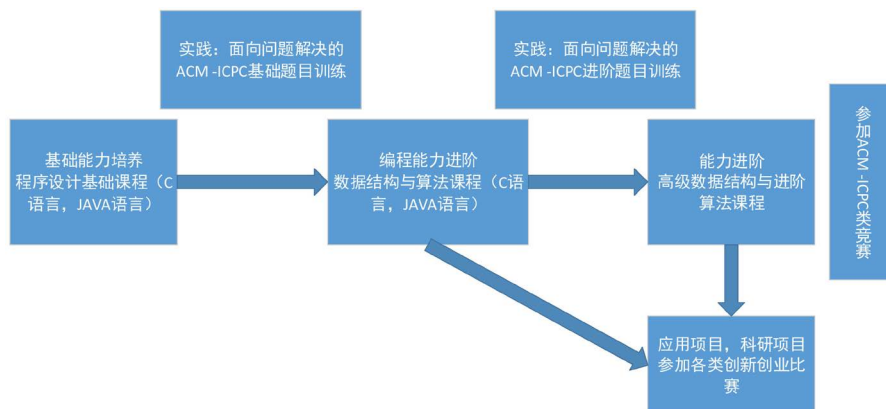


Figure 1. ACM-ICPC-based programming learning roadmap
图 1. 基于 ACM-ICPC 的程序设计类学习路线图

融合 ACM-ICPC 程序设计竞赛的算法课程教学改革中理论课部分可遵循以下原则:

1) 具体问题导向。根据教学目标, 从具体的问题自然地引出此类问题的解法。带着问题讲, 让学生带着问题学。

由于算法课程的特殊性, 其教学内容为一个个具体的算法, 而算法课程的教学目标就是要通过一个个具体的算法来体会一类算法思想和解决问题的思路和方法。所以, 在具体讲授算法课程时, 完全可以以问题导向来进行引导, 这样, 学生就能知道问题的来龙去脉, 对问题会有更深刻的见解。对问题的准确理解, 是学习掌握算法的第一步, 教师在介绍问题的时候可以联系已经学过的内容进行介绍, 这样学生就能更快的定位和理解。如: 在介绍最小生成树算法时候, 可以联系最短路径算法, 说明两类问题的区别和联系, 帮助同学理解最小生成树联通的概念, 就能更准确体会生成树算法和最短路径算法的区别。

2) 启发式提问互动。根据具体问题, 提问解决方法并进行互动。

在教学课堂上提出一个具体问题后, 往往可以给一定时间让学生自主思考一下, 可以放开思维、打开脑洞进行头脑风暴, 这样更容易集中学生的注意力, 增强学生的互动, 提高学生对所学内容的兴趣。此部分时间不宜过长, 需逐步转化到所要讲授的算法内容。

3) 简述算法思想, 概述简化的算法步骤。

通过对解决问题的算法思想进行介绍, 让学生主要领会算法解决问题的思路, 对思路的分析和掌握可以让学生对算法进行融会贯通, 而不仅仅是一个个具体算法的堆砌。如经典的分治递归算法, 可以让学生们掌握对大问题进行分解成小问题再逐个击破的问题解决思路, 此类思想也同样适用于生活中各种不同的问题。掌握了算法思想后, 要把算法过程的书面语转化成贴近生活的语言进行描述, 这样学生就能更快的理解算法的具体步骤, 即算法具体是怎么做的。此部分可以配合板书, 介绍算法的运行过程, 如动态规划类经典问题, 都需要进行填表过程的算法演示, 这样就能清晰的知道算法步骤。

4) 板书伪代码, 讲解真代码, 解决实际问题。

有的算法仅仅讲述算法思想和步骤还不足以清晰的明白算法过程, 必须通过代码的方式进行表达。如搜索算法中的深度优先和宽度优先搜索。这里, 就需要首先用伪代码对算法过程进行描述。伪代码不同于真实代码, 忽略了编程语言的具体语法规则, 突出了算法的具体运行过程, 使学生在理解了思想和步骤后能更清晰的明白算法应该如何实现。讲解完伪代码后, 可以引导学生进行运行算法, 对算法伪代码单步执行, 明白算法伪代码的执行细节。最后可以讲解、演示真实代码, 具体运行出问题结果可以提高学生的学习兴趣, 对问题导向的教学实现闭环。

5) 算法分析、改进与后续。

讲解运行完算法后, 可以对算法进行时间复杂度和空间复杂度的分析, 对学生强调算法一定是通过反思和修改得到的, 可以降低学生的畏难情绪。事实也是如此, 只有对问题和算法进行反复思考, 才可能得到时间、空间效率较高的算法。更进一步, 可以介绍算法的方法应用和后续研究思路等, 提高学生的研究兴趣。

综合上述五步, 可以对算法课程的理论教学课进行一定的改革优化。要求基础较差的同学事先预习, 鼓励基础较好的同学可以学习同课程内容的更深的知识, 可参照 ACM-ICPC 程序设计竞赛中相关算法课程内容的更难的部分。由于理论课时间有限, 要兼顾大多数同学的学习需求, 所以针对不同基础同学的算法课程改革主要体现在算法实践部分。

3. 实践方面

算法课程除了培养学生分析问题和解决问题的能力, 更主要的还是让学生掌握算法设计的基本方法并能熟练设计实现和运用一些常用算法。所以, 算法设计和实现能力是算法课程实践部分的重点。在算

法实践过程中, 采用如下两种方式展开:

- 1) 课上实验 + 讲解;
- 2) 课下做题 + 讨论的方式。

具体的, 在有限的实验课上, 根据实验大纲布置实验题, 通过与学生互动, 掌握学生做实验题的情况, 根据具体情况对实验题进行引导与讲解。课上部分是传统课程教学的必备部分, 更重要的是课下的部分。结合 ACM-ICPC 程序竞赛的组织方式, 采用分小组、分任务的方式结合在线自动判读平台(Online Judge system)来开展算法课程课下实践部分。

在线自动判读平台(Online Judge system) (简称 OJ) 一般指在 ACM/ICPC 程序设计竞赛中用来评测参赛选手的程序的正确性与时空效率的程序以及评测程序所依托的网络环境。一般包括以下几部分: Web 服务器、数据库服务器、编译程序以及评测程序。用户可以通过网络浏览服务器上的网页来选择题目, 通过网页提交代码, 由服务器端调用编译程序对用户提交的程序进行编译, 然后由评测程序进行评判并返回相应的结果到网页上供用户查看[10][11]。这种用户自己提交题目相应代码给服务器自动编译测试的方式, 大大提高了传统由人工来判读程序正确性的效率, 学生可以实时的得到正确的反馈, 测试数据是由有经验的教练或者学生根据题目测试目标提供的较完备的数据, 涵盖了一些代码实现中的细节问题, 对学生代码的判断提供了标准统一的测试依据, 使学生能更快的得到反馈, 或和其他学生进行比较, 从而激发实践算法问题的积极性。另一方面, OJ 还能对算法的运行时间和所需空间进行检测, 使学生能按照算法题目要求实践出合乎复杂度的代码, 养成写完代码测试通过后的算法分析与反思的良好习惯。

在具体执行上, 仿照 ACM-ICPC 程序竞赛的 3 人组队方式, 课下实践部分可以 3~5 人自由组队进行实践练习。组内成员可以相互讨论学习, 加强了学生的团队协作能力, 一定程度上防止了仅为了完成课下练习而网上直接搜索拷贝提交的现象。

实践内容具体包括: 分治法、动态规划、贪心法、回溯法、分支限界法。具体做法可以在 OJ 上布置每章节题目 5~10 题, 选择 ACM-ICPC 程序设计竞赛中较经典的题目, 能有效的提高学生练习算法的积极性。题目数量不宜过多, 难度选取应当采取均匀分布, 属于程序竞赛的基本难度。这样, 对于基础较差的学生, 能多练习提高算法课程的内容要求; 对于基础较好的同学, 能解决 OJ 中较难题增强信心的同时, 还能提高学习兴趣, 对外校 OJ 上的题目进行自行探索实践, 如 Codeforce、POJ、HDUOJ 等。

OJ 的另一个优势体现在能实时、准确的提供学生完成练习的情况, 题目通过与否一清二楚, 是哪个用户在什么时候提交的, 提交了多少次才通过, 每次提交的代码是怎样的都在 OJ 系统中记录的非常清晰, 这就方便教师能在课下掌握学生的课下实践情况, 并在实验课上进行反馈与督促。

由此可见, 通过在算法课程的实践环节融入在线判读平台, 有效的延伸了算法实验课的时间, 使学生能在课下方便的对算法实验进行实践。另一方面, 通过在 OJ 在线判读平台上加入 ACM-ICPC 相关内容的题目, 可从算法课程内容上进行有效补充, 提升学生学习相关算法的兴趣, 加深对相关算法的理解和掌握。综合上述, 融合 ACM-ICPC 程序设计竞赛的算法课程实践环节可以很大程度上弥补计算机实践教学存在的不足。

4. 相关实践结果

本人通过以上教学和改革的初探, 在算法课程教学和 ACM-ICPC 类程序比赛有一定实践结果。在算法课程上, 学生能掌握算法设计的基本思想和策略, 更多的学生能实际看懂算法和实际设计算法并成功运行。通过 2015 级和 2016 级两届软工专业学生的算法课程成绩可以初步看出学生掌握算法知识和实践算法的能力加强了, 如图 2 所示。从图 2 可以看出, 在考核难度不变的情况下学生成绩整体有所提升, 但仍有若干不及格的学生, 还有更进一步提升的空间。

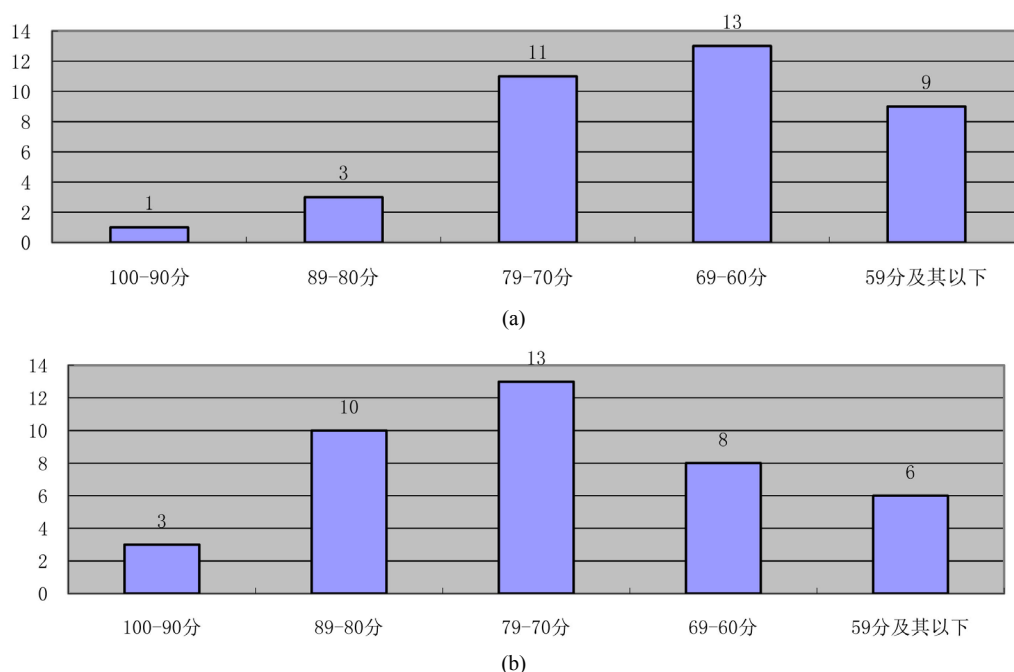


Figure 2. Score distribution chart of soft engineering students. (a) Score distribution chart of 2015 grade students in soft engineering; (b) Score distribution chart of 2016 grade students in soft engineering

图 2. 成绩分布图。(a) 2015 级软工专业某班成绩分布图; (b) 2016 级软工专业某班成绩分布图

在 ACM-ICPC 程序竞赛方面, 由于开展时间短, 对二本学校来说难度较大, 在 ACM-ICPC 中国区域赛上还没有获得成绩, 不过通过组建 ACM 学生社团, 已经招募了 20 人左右的学生团队, 涌现出多名对程序设计竞赛很有兴趣的学生, 通过有效练习和讨论, 逐步培养自学方法和获取知识的能力, 在教练的引导下培养创新意识, 掌握受益终身的思维方法。相信在不久将来在程序设计竞赛方面会有一些的成绩。

5. 结论

本文对融合 ACM-ICPC 的软工专业算法课程教学和实践方面的改革策略进行了分析和描述, 在理论课教学上提出了 5 点教学该给建议, 有效提高了学生在课堂上的注意力和学习效率, 为实践打下坚实基础, 在实践阶段, 通过自主搭建在线判读平台(OJ)来有效弥补实验课时间不足的劣势, 使学生能课下方便快捷的提交代码并得到正确反馈, 提高了学生实践算法的兴趣, 并方便了教师对学生课下练习的管理。另一方面, 融合了 ACM-ICPC 程序竞赛的内容, 也对更有效的提高人才培养, 获得竞赛荣誉有很大帮助。二本院校学生基础较弱, 学习风气不如 985、211 高校, 开展基于竞赛的算法课程改革一方面提高了课程教学效果, 一方面为较强学生的高层次人才培养提供了有效途径, 对整个专业的学风也有一定的积极影响。今后在具体实践中, 应进一步广泛学习理解 ACM-ICPC 程序竞赛的相关经典内容, 为自主 OJ 提供高效、经典的题目, 进一步结合算法课程优化教学效果。

基金项目

武汉市教育局高校教学研究资助项目(课题编号: 2017032)。

参考文献

- [1] 林金珠, 倪天伟. 基于 ACM-ICPC 竞赛的 C 语言课程教学实践[J]. 安庆师范大学学报(自然科学版), 2017, 23(1): 102-104, 119.

- [2] 向程冠, 熊世桓, 熊伟程, 等. 基于 ICPC 问题驱动的《数据结构》教学改革研究[J]. 科技资讯, 2017, 15(20): 179, 181.
- [3] 孙宣东, 路璐, 明俊峰, 等. 从 ACM/ICPC 看计算机算法设计教学改革[J]. 广东工业大学学报(社会科学版), 2005(S1): 237-238.
- [4] 王建芳. 计算机软件类竞赛人才培养模式的探索与实践[J]. 大学教育, 2016(6): 153-154.
- [5] 梁冰, 冯林. 面向 ACM-ICPC 竞赛的计算机人才培养教学与实践方法[J]. 实验室科学, 2017, 20(4): 229-231.
- [6] 郑秋生, 王璐, 王文奇, 等. 普通本科程序设计课程的泛在学习应用研究[J]. 计算机教育, 2018(1): 55-58.
- [7] 赵磊. 融入 ACM/ICPC 竞赛的程序设计类课程教学的改革与探讨[J]. 电脑知识与技术, 2015, 11(25): 116-117.
- [8] 侯向辉, 徐卫, 李琰琰. 以 ACM/ICPC 竞赛促进计算机专业实践教学研究[J]. 计算机时代, 2018(7): 94-96.
- [9] 毛玉萃, 赵宏伟, 邱少明. 由 ACM-ICPC 引起的教与学的变化[J]. 中国校外教育, 2012(24): 162.
- [10] 吴舜歆, 梁宝兰. Online Judge 在 C++程序设计课程实践教学中的应用[J]. 高教学刊, 2016(9): 164, 167.
- [11] 何移. 基于 Online Judge 的算法与编程实训平台建设[J]. 产业与科技论坛, 2016, 15(12): 64-65.

知网检索的两种方式:

1. 打开知网页面 <http://kns.cnki.net/kns/brief/result.aspx?dbPrefix=WWJD>
下拉列表框选择: [ISSN], 输入期刊 ISSN: 2331-799X, 即可查询
2. 打开知网首页 <http://cnki.net/>
左侧“国际文献总库”进入, 输入文章标题, 即可查询

投稿请点击: <http://www.hanspub.org/Submission.aspx>
期刊邮箱: ces@hanspub.org