Experience of Science and Engineering Education and Teaching from Programming Competitions

Linlin Tang

Harbin Institute of Technology, Shenzhen, Guangdong, China

Abstract: As an important carrier for measuring students' computational thinking and practical abilities, programming competitions provide a unique perspective for examining and reforming science and engineering education. This article explores the interactive mode of "using competitions to promote education, and using teacher competitions" to demonstrate the key role of competition activities in comprehensively improving the quality of science and engineering education and teaching. Competitions not only stimulate students' innovative motivation, but also force the teaching system to innovate in terms of cutting-edge content and practical methods; Meanwhile, systematic teaching can provide a sustainable talent pool and knowledge foundation for competitions, forming a positive cycle of mutual empowerment between teaching and competition. This article proposes a feasible implementation path: firstly, integrating competition driven modular projects into the curriculum system to strengthen algorithm design, system construction, and team collaboration capabilities; The second is to establish a hierarchical training mechanism of "basic popularization - top-notch improvement", which combines extensive participation and elite selection through elective courses, workshops, and inter school leagues; The third is to promote teacher development, encourage and guide teachers to feed competition cases and cutting-edge technologies back into daily teaching; The fourth is to build an open training platform, integrate online question answering, project open source and other resources, and support the cultivation of normalized abilities. Practice has shown that this model can effectively bridge the gap between theoretical teaching and engineering practice, and has important promotional value for cultivating new engineering talents with innovative spirit and the ability to solve complex problems.

Keywords: Programming Competition; Promote Education through Competition; Science and Engineering Education; Reform in Education; Cultivation of Innovation Ability

1. Introduction

Currently, the new technological revolution represented by artificial intelligence is profoundly reshaping the demand for computer talents in society, and traditional programmer positions are facing transformation. As an important benchmark for measuring students' computational thinking, algorithm design, and practical abilities, programming competitions provide a unique perspective for testing and reforming science and engineering education[1]. However, there is a common contradiction between the separation of learning and application and the decoupling of competition and teaching in current educational practice. This study aims to systematically explore the interactive mode of "competition and teaching mutually promoting and coexisting", demonstrate its key role in comprehensively improving the quality of science and engineering education and cultivating innovative engineering talents, and propose a feasible path from top-level design to specific implementation[2].

Both domestic and international research and practice have fully recognized the promoting effect of competitions on education. At the theoretical level, research generally agrees that competitions can effectively enhance students' problem-solving abilities, algorithmic thinking, and stress resistance. For example, some studies have specifically designed courses that incorporate real-life competition scenarios to simulate high-pressure environments such as technical interviews and hackathons[3]. The revision of international computer curriculum guidelines (such as CS2023) also focuses on diverse teaching practices such as community

colleges, reflecting a reflection on differentiated training.

At the practical level, a common model is "competition course integration", which integrates competition questions and scoring mechanisms into daily teaching. For example, some domestic universities in China have realized the combination of full participation and elite selection by building a hierarchical resource database of "basic articles - improvement articles - advanced articles" and a progressive system of "three models in class+four levels out of class". These explorations have laid a solid foundation for constructing a systematic interactive model in this article.

2. Related Research Methods

2.1 Systematic Reform Method
Based on the practical experience of multiple universities, to deeply integrate ACM competitions with computer science teaching, the key is to break the boundary between "competition" and "teaching" from the top-level design and build a mutually reinforcing closed-loop system[4]. The common practice in universities is to carry out systematic reforms in terms of curriculum system, training mechanism, and teacher guarantee as shown below.
1) System construction: deeply embed the competition into the training plan
Course integration: Reform core courses such as "Fundamentals of Programming", "Data Structures", and "Algorithm Design and Analysis", break down classic algorithm problems and problems from ACM competitions into teaching cases and homework, and achieve "competition in class". We can develop a graded question bank consisting of "basic improvement advanced" questions.
Credit incentive: Establish credit modules for "competition practice" or "innovative practice", allowing students to exchange their high-level competition award scores for some elective courses or practical credits. Clarify the policy linking award levels with credits, awards and honors, and graduate student recommendation.
2) Teaching reform: Creating a teaching model that integrates competition and education
Content reconstruction: Reverse organize knowledge points through competitions and industry demands, collaborate with enterprises to develop cutting-edge courses or modules such as "cloud native development", and ensure that teaching content does not fall behind.
Method innovation: Project driven teaching is widely adopted in the classroom and after class, and comprehensive competition questions are designed as group projects. At the same time, an online question answering (OJ) system is introduced to enable students to receive real-time feedback after submitting their code, stimulating competition and learning enthusiasm.
3) Training support: Establish a systematic selection and training mechanism
Selection mechanism[5]: By organizing freshmen competitions and campus competitions to attract widespread participation, they will serve as the basis for selection into the school level training team. At the same time, establish a mentorship system of "old leading new" and form a stable team of team members.
Training system: Design a tiered training plan for the training team, which includes on campus exercises, inter school exchanges, and multi school joint training during winter and summer vacations. Training should not only be limited to practicing exercises, but also include post match in-depth review, simulated stress testing, and so on.
4) Evaluation and Incentives: Forming a Closed Loop for Sustainable Development
Student evaluation: Establish a comprehensive ability assessment system that not only looks at competition results, but also assesses code quality, teamwork, and innovation ability in the project.
Teacher motivation[6,7]: Recognize the workload of teachers in competition guidance and integrated curriculum construction, and associate it with teaching evaluation and professional title assessment. Support teachers to transform their guidance experience into teaching reform projects or achievements.

2.2 Classic Case Analysis
Many universities at home and abroad have achieved significant results through characteristic models, and their core experience lies in building an ecosystem that deeply integrates competition and teaching, scientific research, and industry.
The "Top notch Innovation" Model of Top Universities: Represented by the ACM Class of Shanghai Jiao Tong University, its philosophy is to cultivate computer scientists[8, 9]. Through quality education of "thought method practice"

and scientific research training of "laboratory classroom society", it has achieved outstanding results, including the ACM-ICPC World Champion. Harbin Institute of Technology (Shenzhen), Southern University of Science and Technology, and the School of Software Engineering at Sun Yat sen University all adopt a research-oriented full-time mentorship system to guide students to deeply participate in cutting-edge research and high-level competitions. Undergraduate students have a high proportion of publishing papers and participating in research projects at top conferences. Among them, Harbin Institute of Technology (Shenzhen) won the only special prize in the paper competition in Guangdong Province in the recently concluded "Challenge Cup" competition for Chinese university students in 2025.

Guangdong University of Technology is one of outstanding institutions in this field among local colleges. The university takes high-level competitions as a guide and promotes teaching reform to improve overall teaching quality and reputation. The outstanding performance of the ACM team directly demonstrates the teaching strength of the college in the fields of algorithms and programming, becoming a concentrated reflection of the achievements of undergraduate talent cultivation[10]. This benchmark effect inspires a broader group of teachers and students. On the other hand, the school takes this as an opportunity to promote the updating of teaching content: competition questions often involve cutting-edge algorithms and complex problem-solving, which puts higher demands on teaching content[11]. By guiding students in competitions, teachers can incorporate cutting-edge and practical cases and technologies into their daily teaching of core courses such as Data Structures and Algorithm Design, promoting continuous iteration of course content[12]. The four-dimensional integration model of "industry education competition creation" guides course updates based on industry demand, uses competitions as drivers, and uses real enterprise projects as teaching "ballast stones", significantly enhancing students' engineering practice ability and employment competitiveness.

2.3 Experience Research

Overall, universities with good competition education have not focused their resources solely on a few top players, but have implemented a talent cultivation model of "competition education integration" at the whole school level.

Widely participated "Taki": The school promotes the "one major, one competition" model through brand activities such as "Luban Festival", with the goal of allowing every undergraduate student to participate in at least one academic and technological competition during their university years. This has reserved a huge interest group and reserve force for high-end competitions such as ACM.

The 'tower body' of ability cultivation: Through various school level and provincial-level competitions as well as innovation and entrepreneurship projects, potential students are trained and selected.

The elite elite "tower": The ACM training team is the top of this system, undertaking the selection of top talents from a large number of students and conducting high-intensity specialized training.

The "booster" for deepening the integration of industry and education: Schools host or collaborate with enterprises to hold high-level competitions in multiple fields (such as engineering practice competitions, "Jingdiao Cup" graduation design competitions), which often originate from the real needs of enterprises and promote the deep integration of teaching and industry needs.

3. Design Schemes

Based on the analysis of various cases mentioned above, we can conclude that classified training strategies should be adopted based on differences in source quality, training objectives, and resource endowments. Table 1 below presents a basic strategy and approach.

This comparative plan reflects a differentiated development positioning and resource allocation logic: 985 strong universities focus on systematic, high-end, and international training paths, while general universities pay more attention to the gradual improvement of popularization, practicality, and breakthrough. Both types of universities can form a sustainable ACM competition development model based on their own conditions and corresponding strategic frameworks. The following figure 1gives a cultivation plan diagram to visually demonstrate the core strategies and processes of the interactive mode of "competition promotes teaching, teaching promotes competition".

The starting and ending points of the diagram clearly indicate that the driving goal of the entire plan is to establish a virtuous cycle of "teaching and competition mutually empowering", ultimately leading to the core value of cultivating innovative talents. This is a clear "screening improvement" process, corresponding to the original text's "basic popularization first-class improvement" hierarchical training mechanism. It starts with widespread participation in popular education, natural selection and distribution through competition activities, and ultimately focuses on systematic training of elite athletes. Overall, the diagram clearly presents the complete logical chain from concept to implementation, and then to evaluation and feedback, highlighting the mutual support and dynamic cyclic relationship between each link, and fully interpreting the strategy system described in the original text.

Table 1. Basic Strategy and Approach

Strategic Dimension	Strong Schools	General Colleges and Universities
Quality and Selection of Student Sources	Select top talents through multiple channels such as independent enrollment, summer camps, and freshman competitions; Establish competition experimental classes or specialized classes for early identification and tracking.	Widely promote and attract interested students; Explore potential players through campus competitions; Emphasize the cultivation of basic abilities and the stimulation of interests.
Training Objectives and Paths	The core goal is to strive for gold medals in ICPC regional competitions/world finals; Give equal importance to systematic training and research on high-order algorithms; Emphasize the connection between innovation capability and scientific research.	The main goal is to enhance provincial competition awards and break through regional competitions; Focus on basic algorithm training and practical simulation; Encourage students to enhance their employment competitiveness through competitions.
Training System and Content	Hierarchical training (foundation improvement special topic simulation); Equipped with dedicated coaches and academic mentors; Regularly organize multi school joint training and overseas exchanges.	Conducting periodic training through clubs or courses as carriers; Relying on previous outstanding contestants for peer guidance; Strengthen inter school cooperation to obtain training resources.
Resource Investment and Support	Specialized training facilities, high-performance computing equipment, and special funding support; Co building laboratories or establishing competition funds with enterprises; Provide policy support such as research recommendation and international exchange.	Strive for basic funding and venue support from universities; Actively seeking corporate sponsorship or cooperation; Strive for recognition of campus policies and resource allocation through award-winning achievements.
Incentive mechanism and development linkage	Incorporate competition results into the recommendation and scholarship evaluation system; Priority recommendation to top universities or enterprises both domestically and internationally; Promote the transformation of competition results into scientific research projects.	Strengthening the direct support of awards for excellence evaluation and employment; Establish internship recommendation channels with local enterprises; Emphasize the support of competitive ability for long-term career development.

This competition education framework design demonstrates excellent systematicity, foresight, and practical feasibility. Its core advantage lies in construction of a dynamic circular ecosystem driven by "mutual promotion of teaching and competition", with "screening improvement" as internal logic, and ultimately serving the core value of "cultivating innovative talents".

Firstly, the strategic height of this design is particularly prominent. It does not simply regard competition as an additional evaluation link after teaching, but rather positions it as a core driving mechanism deeply embedded and mutually reinforcing with the teaching process. This makes the entire system surpass the drawbacks of "separation of learning and application" or "competition for the sake of competition" that often occur in traditional education, forming a virtuous loop of "promoting learning through competition and striving for excellence through learning", ensuring the vitality and directionality of educational activities.

Secondly, its hierarchical promotion mechanism of "basic popularization level improvement elite

cultivation" is highly inclusive and scientific. From the starting point, it achieves fairness in educational opportunities and universal cultivation of innovative soil through extensive participation in mass education. This is not only a basic guarantee for talent discovery, but also an important educational process itself; Subsequently, natural screening and diversion were carried out through competitions, rather than cruel elimination, forming a performance-based "diagnosis feedback" system that provides clear and personalized pathways for learners with different traits and stages, enabling resource allocation and training guidance to accurately match student needs, achieving a dialectical unity of efficiency and fairness.
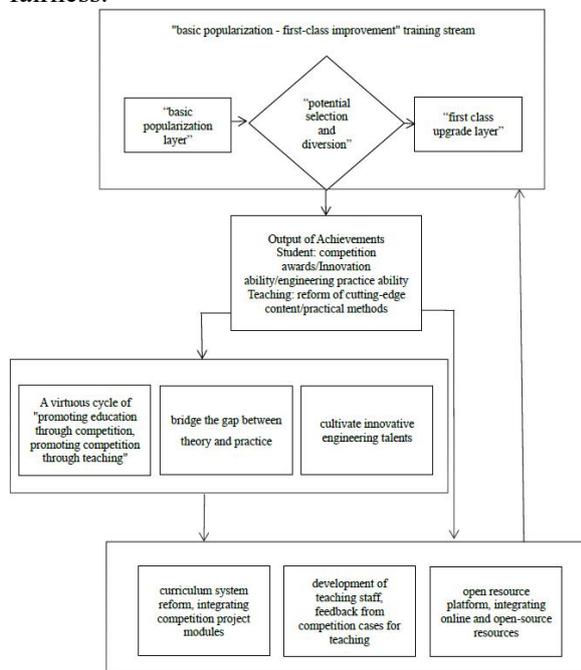

Figure 1. Training Plan Diagram

Furthermore, the framework clearly outlines the complete logical chain from concept to implementation and evaluation feedback, with each link interconnected and mutually supportive. In particular, the clear evaluation and feedback mechanism ensures that the system has the ability to self reflect and continuously optimize, making it a "smart" educational organism that can adapt to changes and evolve, rather than a rigid static structure.

In the end, all links are closely anchored to the ultimate value of "cultivating innovative talents", giving technical and process oriented design a soul and destination, ensuring that educational innovation is not alienated by means and always serves the comprehensive development and outstanding growth of people. In summary, this framework successfully elevates the educational function of competitions from a single selection tool to a complete strategic system that integrates large-scale universal education, precise process diagnosis, elite system training, and sustainable ecological construction. It provides China with an excellent blueprint for innovative talent cultivation that combines theoretical height and operational path.

4. Conclusion

This article takes programming competitions as a starting point to explore their core role in promoting the reform of science and engineering education and cultivating innovative talents. The core argument of the paper is to construct a bidirectional and mutually reinforcing cycle model of "using competition to promote teaching and teaching to support competition", aiming to bridge the gap between traditional theoretical teaching and cutting-edge engineering practice. The article first establishes the position of programming competitions as a key carrier for measuring computational thinking and practical abilities, pointing out that they provide a unique and effective perspective for examining and reforming science and engineering education. The author thoroughly demonstrates the positive relationship of "mutual empowerment" between competition activities and systematic teaching: on the one hand, competition, through its challenge and cutting-edge nature, can effectively stimulate students' intrinsic learning and innovation motivation, and "force" the teaching system to iteratively update its content and methods; On the other hand, solid and systematic curriculum teaching provides a stable talent reserve and a profound knowledge foundation for the sustainable development of competitions. To achieve this interactive mode, the paper proposes a specific implementation framework that includes four dimensions. This model effectively promotes the integration of theory and practice through the deep coupling of competition and teaching, and has important practical value and promotion significance for cultivating new engineering talents with innovative spirit and the ability to solve complex problems. The logic of the entire text is clear, from theoretical argumentation to path design, systematically elaborating an operable and promotable educational reform plan.

References
[1] Desai, P., & Hiremath, P. G. S. (2024). Integrated Project Development through Combined Theory and Practices of Core Courses: A Case Study of Software Engineering. Journal of Engineering Education Transformations, 37(Special Issue): 550-558.
[2] Zahn, E., Kuechler, B., & Böhmann, T. (2023). Computational Thinking for Design Science Researchers - A Modular Training Approach. In Proceedings of the 18th International Conference on Design Science Research in Information Systems and Technology (DESRIST 2023) : 365-374.
[3] Uriawan, W., Zambrano, R., & Wong, D. (2024). Eureka: Question Answer Course Discussion using Agile Methods. Preprints, 2024061507.
[4] Porter, L., Guzdial, M., McDowell, C., & Simon, B. (2013). Success in introductory programming: what works? Communications of the ACM, 56(8): 34-36.
[5] Bunde, D. P., Stehlik, M., & Klingbeil, K. (2020). The effect of competitive programming on students' motivation and performance. In Proceedings of the 51st ACM Technical Symposium on Computer Science Education (SIGCSE '20): 682-688.
[6] Li, Q. (2010). Research and Design of Network Teaching Platform Based on B/S Mode (Master's thesis). Southwest Jiaotong University.
[7] Wang, T., & Su, X. (2015). Teaching programming skills in primary school classrooms: A case study of implementing the CPBL model. Computer Science Education, 25(2): 169-186.
[8] Luxton-Reilly, A., Simon, Albluwi, I., Becker, B. A., Giannakos, M., Kumar, A. N., Ott, L., Paterson, J., Scott, M. J., Sheard, J., & Szabo, C. (2018). Introductory programming: a systematic literature review. In Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE 2018): 55-106.
[9] Papert, S. (1980). Mindstorms: Children, Computers, and Powerful Ideas. Basic Books.
[10] Falkner, K., Sentance, S., Vivian, R., Barksdale, S., Busuttil, L., Cole, E., Liebe, C., Maiorana, F., McGill, M. M., & Oates, G. (2019). A framework for digital competencies for teachers. UNESCO.
[11] Hattie, J. (2009). *Visible Learning: A Synthesis of Over 800 Meta-Analyses Relating to Achievement*. Routledge.
[12] Vihavainen, A., Vikberg, T., Luukkainen, M., & Pärtel, M. (2014). How to study programming on mobile touch devices: interactive Python code exercises. In Proceedings of the 14th Koli Calling International Conference on Computing Education Research (Koli Calling '14): 51-58.
.