

Teaching Reform of the C# Programming Course for Computer Science Majors in Local Finance and Economics Universities

Hao Li*

School of Computer and Information Engineering, Anhui University of Finance and Economics, Bengbu, Anhui 233030, P.R.China

* Corresponding author: Hao Li (lhao@aufe.edu.cn)

Abstract: Aiming at the common problems in the teaching of the C# programming course in financial colleges, this study introduces the international advanced engineering education concept Conceive Design Implement Operate (CDIO), and systematically puts forward the teaching reform Measurements. The reform practice shows that by reconstructing the course content system oriented by the financial application scenario, constructing the "online-offline competition" Trinity hybrid teaching mode, implementing practical teaching throughout the whole life cycle of the project, and establishing a multi-dimensional assessment system combining process and finality, students' interest in learning can be effectively stimulated. Their computational thinking, interdisciplinary problem-solving, and innovative practice ability can be significantly improved, which provides a reference for the cultivation of applied and compound information technology talents in Financial Colleges and Universities under the background of new engineering.

Keywords: Engineering Education Accreditation, CDIO, Local Finance and Economics Universities, C# Programming, Teaching Reform.

1. Introduction

With the in-depth development of the fourth industrial revolution represented by artificial intelligence, big data, and cloud computing, China's education department has put forward the "new engineering" strategy [1]. The new engineering construction aims to cultivate interdisciplinary engineering talents with interdisciplinary integration, innovation, and practical ability, which poses a new challenge to the teaching of engineering-related courses in finance and Economics Universities, which are traditionally good at economics and management disciplines [2,3]. The construction of new engineering disciplines in finance and economics universities is not intended to turn these institutions into science and engineering universities, but to fully leverage their advantages in economics and management to foster versatile talents with expertise in both technology and finance, for which the reform of programming courses serves as the key breakthrough [4]. At the same time, the concept of "output oriented" (OBE) and continuous improvement advocated by the international engineering education certification requires that teaching should shift from knowledge transfer to the cultivation of comprehensive abilities such as solving complex problems, communication and cooperation, and the traditional mode of "emphasizing theory over practice" needs to be reformed urgently. [5]The CDIO concept is highly consistent with this. By combining curriculum practice with practical financial issues (such as financial data analysis and business intelligence), students can effectively bridge the gap between theory and industry application by "learning by doing" [6]. The three aspects jointly guide the systematic teaching reform direction of the programming course in Financial Colleges and universities.

2. Some Problems in the Teaching of C # Programming Course in Local Universities of Finance and Economics

The widespread use of C# in enterprise-level application development (especially for Windows desktop applications, Web services, and game development on the .NET platform), together with its powerful ecosystem and distinct object-oriented characteristics, has made it a key teaching language for information-technology majors in many finance and economics universities [7]. Nevertheless, as with other programming languages, the conventional teaching approach for C# has exposed many shortcomings in the new-era context [8-10].

2.1. Insufficient Integration of Teaching Content and Financial Characteristics

Under the traditional teaching mode, the content of the course is often limited to the syntax foundation, control structure, and object-oriented concepts (such as classes, inheritance, polymorphism, etc.) of the C # language, and simple application development based on Windows Forms [11]. Teaching cases often use generic examples such as "digital calculators" and "student information management systems", which are seriously disconnected from the background of finance and economics (such as accounting, finance, taxation, auditing, e-commerce). Students find it difficult to understand the significance and value of the course for their major in their studies, resulting in unclear learning objectives and insufficient motivation[12].

2.2. The Teaching Methods are Outdated and Monotonous, Can't Stimulate Students' Learning Initiative

The classroom teaching mode is a "cramming" teaching method [13]. Teachers explain the code line by line, and students passively digest and understand it. Although some question-and-answer interaction is adopted, the interactivity is still limited. This mode is not conducive to the cultivation of students' active thinking and exploration spirit, resulting in insufficient initiative of students, which is contrary to the modern educational concept centered on students and emphasizing ability cultivation [14].

2.3. Few Experimental Class Hour and Insufficient Project Practice

In the traditional course teaching arrangement, theoretical classes are often the main focus, with experimental classes as a supplement, with a 7:3 ratio. The experimental content aims to consolidate individual knowledge points and consists mostly of isolated verification exercises, lacking systematic and comprehensive training in design projects. This results in students being unable to experience the complete engineering development process, which spans all stages from requirements analysis through design and development to deployment and maintenance [13].

2.4. The Assessment Method is Monotonous, Making it Difficult to Students' Comprehensive Abilities

Course assessment usually adopts a final exam mainly based on written tests. The content of the test paper focuses on evaluating the grammar details, program logic and understanding, as well as simple programming skills of the C# language. This "one exam determines everything" approach cannot comprehensively measure students' comprehensive abilities (including programming practice ability, system design ability, teamwork ability, and innovative ability to solve practical problems, etc.). Furthermore, this approach is prone to inducing students to adopt a pre-exam cramming learning method in pursuit of short-term test-taking effects, thus deviating from the original intention of cultivating students' qualities and comprehensive abilities [15].

3. Teaching Reform Measures of C# Course based on CDIO Concept

To address the above issues, we have carried out a reform and exploration of the C# course teaching in our school, guided by the new engineering strategy and the CDIO concept. The specific measures are as follows:

3.1. Update the Course Content, Introduce the CDIO Concept, Emphasize the Integration of Multiple Disciplines, and Focus on Cultivating Students' Engineering Capabilities

Firstly, reconstruct the course knowledge modules with financial and economic applications as the main line, and introduce the "concept-design" concept of CDIO into classroom teaching. When explaining each key point of knowledge, first introduce a related "micro-scenario"

problem in the field of finance and economics. For instance, when explaining "arrays and sets", the issue of "storage and statistics of stock trading day return sequences" can be introduced. Secondly, during the programming stage, teachers not only teach grammar but also guide students to conduct requirement analysis, data structure design, and class relationship design for the aforementioned financial and economic scenarios. This enables the natural integration of programming learning with financial expertise, enhancing the appeal of the classroom and students' sense of immersion.

3.2. Integrate Multiple Online and Offline Teaching Approaches, Break through the Limitations of Teaching Time and Space, and Support Continuous Learning in All Links of CDIO

In the offline classroom session, the core syntax of the C# language and complex technologies (such as delegation and events, LINQ, multi-threading) and other key and difficult knowledge points are elaborated in detail first. Then, case teaching, student group discussions, code reviews, and other forms are adopted to carry out relevant project design and discussions, thereby strengthening students' classroom interaction and in-depth thinking.

During the online learning session, teachers can use the school's online teaching platforms (such as Chaoxing, Touge, etc.) to upload C# knowledge points micro-videos, basic exercises, advanced and extended learning materials, etc. They can also assign homework and provide Q&A. Students can study, ask questions, and conduct extension training anytime and anywhere after class.

3.3. Take Subject Competitions as a Handle to Consolidate Students' Practical Innovation Capabilities

Actively guide students to participate in subject competitions that are in line with the characteristics of the finance and economics information technology major, such as the China Undergraduate Computer Design Contest, the "Internet Plus" College Students' Innovation and Entrepreneurship Competition, and the Financial Technology Innovation Competition. Meanwhile, the fragmented practical activities spread across the semester are integrated into the staged roll-out of the "Finance and Economics C# Comprehensive Competition Project." Corresponding point-based incentives are also offered to motivate students to actively engage in the full CDIO-based practical training process, which includes team conception, system analysis and design, coding and testing, and deployment and presentation [16]. The topics for major course assignments are prioritized based on real competition questions and the actual project demands of enterprises, enabling students to implement the CDIO educational philosophy in the high-standard and high-pressure competition preparation practice, and comprehensively temper their innovative thinking, engineering practice, and teamwork abilities. Instructors transform outstanding competition works, core technical difficulties, and optimization plans into high-quality teaching cases, which are then fed back into daily classroom teaching to build a closed-loop education system of "teaching - practice - competition - feedback" [17,18].

3.4. Reform the Traditional Teaching Evaluation Model and Establish a Diversified Assessment and Evaluation System

The details of establishing a diversified and process-oriented assessment system that matches the CDIO process and capability goals are as follows:

1. Process evaluation (70% weight)

(1) Conception and design ability (20% weight): Evaluated through project initiation reports, system design documents, and contributions to group design discussions.

(2) Implementation capability (20% weight): This includes the completion rate and quality of exercises and assignments on online platforms, the quality of code in laboratory courses, the code submitted during the project's iterative development process, and the pass rate of unit tests, etc.

(3) Collaboration and communication skills (20% weight): Evaluated through group mutual assessment, project discussion records, and performance in milestone reports.

(4) Online learning performance (10% weight): This includes the completion rate of video learning, the activity level of online discussions, and the scores of online quizzes.

2. Final Class Evaluation (30% weight)

(1) Project Defense (20% weight): The project will be evaluated from aspects such as its completeness, innovation, technical implementation quality, and team operation. During the defense phase, the students' abilities to present, answer questions, and demonstrate operational effects will be assessed.

(2) Final comprehensive written test (10% weight): The written test will be appropriately retained, but the question types will focus on assessing the understanding of core concepts, architectural design thinking, and the ability to read and debug slightly more complex programs, reducing questions that require rote learning.

4. Result Analysis

We obtained a satisfaction evaluation of students' teaching effectiveness through multiple channels such as questionnaires, online teaching evaluations, teacher-student symposiums, and class observation supervision (among which the 2023 and 2024 grades implemented curriculum reforms, while the 2022 grade still adopted the traditional teaching mode). The specific results are shown in Fig.1. As can be seen from Fig.1, with the implementation of teaching reform, students' satisfaction with the "C# Programming" course has significantly increased.

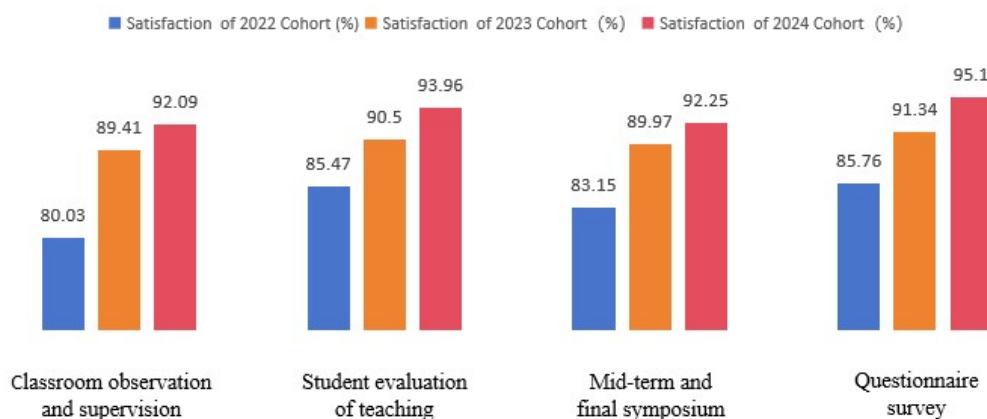


Fig. 1 Students satisfaction with the C# programming course

5. Conclusion

Under the dual impetus of new engineering construction and the professional accreditation of engineering education, the transformation and upgrading as well as teaching reform of programming courses in finance and economics universities are extremely urgent. This project tackles the inherent limitations of the existing C# course model while leveraging the CDIO engineering education philosophy to implement a holistic curriculum reform. It not only optimizes the course content and teaching methods but also brings about a fundamental shift in educational philosophy, prompting the teaching approach to transition from a single "knowledge-imparting" model to one that emphasizes the collaborative cultivation of both ability and quality, and from single-subject instruction to a multidisciplinary, cross-integrated teaching model. Teaching practice shows that this reform has effectively stimulated students' motivation for autonomous learning and significantly enhanced their comprehensive abilities in solving complex engineering problems, practical development, and team innovation and collaboration. In the future, the project intends to sustain its reform exploration, prioritizing the enhancement

of teaching staff capacity, the promotion of robust school-enterprise cooperation, and the optimization of an end-to-end teaching assessment mechanism, so as to further cement the reform outcomes and consistently improve the educational quality of its courses.

Acknowledgments

This research is supported by the 2024 Undergraduate Teaching Quality and Teaching Reform Project of Anhui University of Finance and Economics (Project No. acjyyb2024020), titled "Reform and Exploration of Programming Course Teaching in the Context of Engineering Education Accreditation and New Engineering".

References

- [1] Hu, B., Feng, H., Han, W., et al. (2017). Accelerating New Engineering construction and promoting engineering education reform and innovation: A review of the "Symposium on Engineering Education Development Strategies in Comprehensive Universities". *Fudan Education Forum*, 15(2), 20–27+2. <https://doi.org/10.13397/j.cnki.fef.2017.02.004>
- [2] Pang, P., Feng, S., & Wu, J. (2025). Construction of pathways for enhancing science and technology talent cultivation

- objectives under the background of “Four New” construction: From the perspective of information literacy education. *Journal of Library and Information Science*, 10(1), 65–71.
- [3] Wang, X. (2022). Exploration of “Power Electronics Technology” teaching method for New Engineering. *Frontiers in Educational Research*, 5(22). <https://doi.org/10.25236/FER.2022.052219>
- [4] Liu, Y., Wang, A., Xuan, Q., et al. (2019). Analysis of the construction of data science and big data technology major under the New Engineering background: A case study of local finance and economics universities. *Journal of Jilin Business and Technology College*, 35(6), 106–109. <https://doi.org/10.19520/j.cnki.issn1674-3288.2019.06.021>
- [5] He, Y., Zhang, W., & Yan, J. (2021). The design of the college English curriculum from the perspective of international engineering education accreditation—A case study from the Beijing Institute of Petrochemical Technology. *Journal of Curriculum and Teaching*, 10(1). <https://doi.org/10.5430/JCT.V10N1P18>
- [6] Wu, Y. (2025). Research and practice on the reform of analog electronic technology curriculum system based on the OBE CDIO concept. *International Journal of New Developments in Education*, 7(2). <https://doi.org/10.25236/IJNDE.2025.070214>
- [7] Zhang, Q., Mao, N., Li, J., et al. (2025). Design of the top level code framework for Unity based on C# language. *Industry Science and Engineering*, 2(1). <https://doi.org/10.62381/I255108>
- [8] Wang, S., Liu, J., Li, B., et al. (2025). Research and practice of teaching reform in “C++ Programming and Algorithm Implementation” under the New Engineering education background. *Journal of Higher Education Research*, 6(5). <https://doi.org/10.32629/JHER.V6I5.4575>
- [9] Ahat, A., & Alimjan, G. (2025). Teaching reform of the C programming language course integrating OBE concept and AI assistance under the New Engineering background. *Journal of Contemporary Educational Research*, 9(7), 197–202. <https://doi.org/10.26689/JCER.V9I7.11350>
- [10] Liu, G., Zhang, X., & Yan, H. (2022). Exploration and practice of interdisciplinary integration cultivation of talents majored in intelligent manufacturing engineering under the background of New Engineering education. *Curriculum and Teaching Methodology*, 5(15). <https://doi.org/10.23977/CURTM.2022.051517>
- [11] Gao, X. (2026). AI knowledge graph empowers teaching reform of “C# Programming”. *Office Automation*, 31(11), 51–53.
- [12] Yu, S., & Jia, S. (2025). Practice of OBE concept in teaching “C# Programming”. *Journal of Shazhou Professional Institute of Technology*, 28(4), 29–33.
- [13] Tian, X., He, Y., & Wang, S. (2025). Teaching innovation of C# programming driven by Unity practical combat. *Computer Education*, (2), 150–154. <https://doi.org/10.16512/j.cnki.jsjy.2025.02.017>
- [14] Shi, Y., & Lu, H. (2024). Innovation and practice of “C# Programming” course reform integrating OBE CDIO concepts under the New Engineering background. *Journal of Anhui Vocational College of Electronics & Information Technology*, 23(1), 64–69.
- [15] Qasem, A. A. (2025). The effectiveness of project based learning in development of C# programming language skills among education technology students. *Discover Education*, 4(1), 223. <https://doi.org/10.1007/s44217-025-00633-0>
- [16] Guo, X., & Hu, X. (2023). Teaching reform based on OBE object oriented programming language (Java) course. In *Proceedings of 2023 3rd International Conference on Internet Technology and Educational Informatization* (pp. 198–210). Quanzhou University of Information Engineering Software College.
- [17] Wu, J., & Ma, C. (2026). OBE oriented reform of Java programming course in higher education: A case study integrating PBL and project driven teaching model. *Asian Journal of Education and Social Studies*, 52(7), 733–746. <https://doi.org/10.9734/AJESS/2026/V52I73203>
- [18] Chen, Y. (2026). Theoretical framework and practical implementation of the OBE PBL integrated teaching model. *Education Reform and Development*, 8(6), 82–92. <https://doi.org/10.26689/ERD.V8I6.15096>