Research on the Reform of Algorithm and Programming Practice Course Based on Outcome-based Education

Yuhan Li, Yibing Kong, Xinyuan Zhang, Zhiguo Zhou
College of Information Science and Technology Northeast Normal University Chang Chun, China

Abstract—Aiming at the problems of insufficient innovativeness of basic practice courses, deviation of the content of comprehensive practice courses from the industry demand, and single means of practice teaching in computer majors of higher education institutions, we take the algorithm and programming practice course as an example and combine it with the requirements of the contemporary society for the graduates and the characteristics of the discipline, and carry out a reform of the practice course. To improve the innovativeness of practical courses in a student-centered manner, we propose a curriculum reform plan for Outcome-based education that incorporates school-enterprise cooperation and encourages students to participate in academic competitions. Through teaching practice, the proposal has proved to be effective in improving students' practical and innovative abilities, while meeting the needs of enterprises and gaining social recognition.

1. INTRODUCTION

The National Medium and Long-term Educational Reform and Development Plan (2010-2020) articulates a firm and unwavering commitment to prioritizing the cultivation of competencies, with a special emphasis on enhancing students' practical and innovative skills [1]. While there have been notable advancements in recent years toward establishing a comprehensive teaching framework for computer science education in universities and colleges, these achievements have largely been rooted in conventional educational models. These conventional approaches have, to a significant extent, hindered the development of students' practical and innovative proficiencies, especially in the context of programming courses. Consequently, many graduates often find themselves in a situation where they need to undergo an extended period of transformation and receive additional training to meet the evolving requirements of the job market [2-3]. Professor Jiang Zongli, who holds the position of Vice Chairman at the Computer Science Major Teaching Guidance Committee under the Ministry of Education, has been a strong advocate for a fundamental transformation in educational philosophy within the context of the New Engineering paradigm. This paradigm shift entails a move away from the conventional curriculum-centered approach, replacing it with a more outcome-centered model. In this revised approach, the central emphasis is on preparing students with the essential skills required to effectively address intricate engineering challenges [4-5].

Engineering-oriented thinking is of paramount importance in nurturing students' capacity to tackle intricate engineering challenges. Within the framework of the "New Engineering" initiative, numerous universities are actively overhauling their educational and instructional approaches. Among these reforms, the course "Algorithm and Programming Practice" takes on a pivotal role in the education of computer science majors. This course serves as a linchpin in students' comprehension of the entire field, facilitating mastery of computer programming languages, and establishing a robust groundwork for their forthcoming professional endeavors. Significantly, major IT companies regard proficiency in algorithmic skills as the cornerstone for assessing the competence of graduating students. The curriculum is making dedicated efforts to institute an Outcome-Based Education (OBE) reform strategy. These endeavors include forging partnerships with industry stakeholders and engaging in research competitions, all with the overarching objective of augmenting students' practical and innovative proficiencies.

2. THE ISSUES PRESENT IN THE PRACTICAL TEACHING OF THE ALGORITHM AND PROGRAM DESIGN PRACTICE COURSE

Under the dominance of traditional educational paradigms, the development of programming practical skills and innovative capabilities among computer science students is constrained [6].
2.1. Insufficient innovativeness in foundational practical courses

Laboratory courses play a vital complementary role to the theoretical aspects of the curriculum. They are meticulously crafted to not only deepen students' comprehension of theoretical concepts but also to cultivate their computational thinking abilities and practical operational skills [7]. Currently, in the curriculum framework of higher education institutions that provide computer science programs, there is a prevailing tendency to give precedence to the maturation of theory and technology in foundational practical courses. In many instances, these courses tend to concentrate primarily on delivering fundamental theories and essential knowledge, potentially overlooking the cultivation of innovative skills and the promotion of innovative awareness among students. Regrettably, instructors may not consistently delve deeply into the rapidly changing IT industry to stay abreast of the latest technological advancements. Consequently, this leads to the creation of training programs that become outdated over time [8]. Furthermore, the subject matter covered in practical courses often evolves at a sluggish pace. In certain scenarios, multiple educational institutions may offer nearly identical content in their practical courses. This redundancy significantly hampers the capacity of practical courses to effectively fulfill their intended objectives.

2.2. Slow update of professional practice programs

The design of practical courses is intended to foster students' engineering thinking abilities, prompting universities to actively encourage and organize students' participation in practical training initiatives, such as college students' innovation training projects. However, many of these practice programs are either teacher-assigned or institutionally designated, lacking significant connections with enterprises. Consequently, they often fall short of truly honing students' skills and enhancing their capabilities. Conversely, there's a scarcity of genuine design, integration, and creative practice teaching. Both enterprise internships and graduation projects serve as vital components of professional practice that play a substantial role in cultivating students' engineering thinking abilities, as illustrated in Fig. 1. Currently, only a mere 20% and 15% of internship topics and graduation design topics for computer science majors at our institution are directly derived from real enterprise projects. As a result, there's a compelling need to expand and diversify these approaches to better align with the requirements of industry practice.

![Figure 1. Competencies valued by companies](image_url)

The situation described above has far-reaching consequences. It not only hampers students' personal development and leaves them with a deficiency in practical skills but also compels enterprises to invest more in training new graduates to bridge the gap between their abilities and industry requirements. Consequently, there is a predicament where graduates struggle to secure employment, while enterprises grapple with challenges in recruiting adequately skilled candidates.

3. Reform of the Algorithms and Programming Practice Course

3.1. Updating basic practice courses and developing hands-on practical skills

Practice teaching serves as a pivotal element in the cultivation of new engineering talents, offering an effective means to reinforce and deepen theoretical knowledge. It stands as a crucial component in training students to not only master subject knowledge but also to enhance their practical aptitude [9]. Specifically, in the field of computer science, the course "Algorithms and Programming Practice" is a foundational and specialized programming course. The primary objective of this course is to equip students with fundamental problem-solving skills in the realm of computers. These skills are deemed essential as a prerequisite for acquiring other computer-related competencies [10-11]. Therefore, it is imperative to ensure that students not only possess a strong foundational knowledge base but also have the ability to seamlessly integrate this knowledge and demonstrate adept programming skills.

In order to refine the curriculum development program and accentuate a competency-based methodology, it is crucial to design the teaching process in a manner that gradually leads students through a series of practical subjects. To exemplify this approach, the practical programming topic of "Calculating the Fibonacci series" serves as a case study, with the corresponding teaching framework elucidated in Fig. 2.
3.2. Following the needs of enterprises and cultivating engineering thinking skills

The nurturing of practical engineering professionals should pivot back to the core of engineering practice. This shift should encourage the adoption of a teaching concept that integrates industry and education and fosters collaborative education between educational institutions and businesses. In doing so, it becomes essential to overhaul the methods used in talent cultivation [12].

3.2.1. Modify talent development programs flexibly. Actively invite enterprise experts to come to the classroom and participate in teaching together with teachers. These experts can adapt and diversify their teaching responsibilities, leveraging their strong theoretical foundation and practical algorithm implementation expertise. They will elucidate the underlying knowledge and real-world applications, bridging the gap between industry demands and educational objectives. By aligning our teaching goals with the needs of the business world, we not only foster students’ engineering problem-solving skills but also groom them into the skilled professionals sought after by enterprises.

3.2.2. Establish a school-enterprise cooperation program. Outside the traditional classroom setting, students are steered towards active participation in corporate internships, providing them with immersive experiences that expose them to the complexities of real-world organizational requirements and dynamics.

3.3. Participate in scientific research and competitions and develop creative thinking skills

The emerging field of new engineering disciplines encompasses fresh concepts, novel structures, innovative methodologies, enhanced quality, and a redefined educational system[13]. In this context, students are required to possess not only a strong foundation in their respective subjects but also a high level of social competence, ethics, and professionalism. Scientific research and competitive activities serve as effective and vital avenues for college students to develop innovative thinking, creative skills, organizational coordination, and teamwork. They play a crucial and irreplaceable role in nurturing the spirit of innovation and entrepreneurship among college students, especially within the framework of these new engineering disciplines[14]. Following the curriculum overhaul, students are actively encouraged to participate in professional competitions, which serve to broaden their horizons and bolster their computational thinking skills. These competitions include esteemed events like the International Collegiate Programming Contest (ICPC), wherein students are introduced to advanced algorithms in the field of computer science, providing them with a comprehensive understanding of the subject.

3.3.1. Build a professional competition management system. Organize students who are interested in competitions to participate in professional competition training after school hours, and provide professional and systematic training to students.

3.3.2. Develop innovative projects. Academic competitions go beyond the boundaries of involvement in competitions at the national, provincial, and municipal levels organized by industry associations. They are shaped by the demands of professional training, following an educational approach that emphasizes teacher guidance and student leadership, placing students at the core of the learning experience[15]. This approach not only encourages students to explore and innovate but also empowers them to formulate their unique project topics. With the guidance of teachers, students collaborate to design their own experimental programs, step by step crafting personalized practical projects that help develop their capacity for collaborative innovation.

3.4. Multi-level process evaluation program

A curriculum assessment, centered on fostering student growth, is designed with a keen focus on student development. This comprehensive assessment places a greater emphasis on online interactions and daily grades, highlighting the progression of skill development. The assessment process incorporates the use of an Online Judge (OJ) assessment format from start to finish[16], elevating the evaluation from measuring "knowledge mastery" to evaluating the "capability to address intricate engineering challenges following knowledge internalization." The process assessment framework is detailed in Fig. 3.

4. Effectiveness of implementation

Upon the conclusion of each teaching cycle for the course, an extensive feedback mechanism is put into action. This process involves the collection of teaching evaluations from current students and actively soliciting feedback and suggestions from both graduates and enterprise employers. These combined efforts are crucial in enabling the course management team to promptly identify any issues or
shortcomings in the teaching process. By responding to these insights, the course undergoes a continuous cycle of improvement, ensuring that it remains in a perpetual state of enhancement. This relentless commitment to improvement is instrumental in continuously elevating the quality of teaching and the overall educational experience. In recent years, the continuous exploration and reform of the curriculum has basically achieved the expected results, and the following results have been achieved:

1) Students wholeheartedly embrace the competency-based teaching model, indicating a clear departure from passive absorption of knowledge to active and enthusiastic exploration. This shift reflects their eagerness to engage actively in the learning process and take a more proactive role in their education. Their keen enthusiasm for practical courses not only solidifies their theoretical understanding but also fuels a reciprocal process where their practical innovation skills are continually strengthened, creating a virtuous cycle of skill development.

2) The teaching methodology that seamlessly blends theoretical understanding with practical application serves to enrich students' computational thinking capabilities and elevate their engineering expertise. As a result, this comprehensive approach leads to improved prospects in the job market and opens doors for further academic pursuits, as students are well-equipped with a holistic skill set. Based on statistical data, graduates who have actively participated in ACM competitions exhibit significantly better employment prospects compared to their peers, along with a notably higher rate of college advancement. The destination statistics of our graduating ACM team members (undergraduates) in 2023 are shown in Fig. 4, and the further education of our graduating ACM team members (undergraduates) in 2023 is shown in Fig. 5.

In order to gather insights on teaching and learning outcomes, the course team conducted a survey of the Class of 2023 graduates regarding their post-graduation destinations. According to the survey results, 40% of the graduates have secured positions at prominent domestic Internet giants such as Alibaba and Tencent, while 18% have successfully joined renowned foreign Internet companies like Google and Microsoft. An additional 30% of the graduates have embarked on their careers with other well-established domestic firms.

Undergraduate education is the foundation of higher education, and it is of paramount importance to institute a curriculum reform program firmly grounded in outcomes and competency-driven education. This entails the establishment of a sustainable improvement mechanism and the annual evaluation of teaching and learning quality. These evaluations are systematically compared to predefined objectives, guaranteeing that the curriculum consistently revolves around the needs of students and undergoes a continuous process of refinement. The ultimate goal of this approach is to cultivate a new generation of versatile computer professionals who possess a well-rounded skill set and can make substantial contributions to society in an effective manner.

5. CONCLUSION

The ongoing process of exploration and reform within the course "Algorithm and Programming Practice," rooted in an outcomes and competency-driven approach, has successfully addressed several critical issues. It has triumphed over the limitations associated with the lack of innovation in traditional teaching methods and the misalignment of comprehensive practice course content with the ever-evolving demands of the industry. This endeavor has, in turn, resulted in a significant enhancement of students' practical and innovative proficiencies, making them better suited to meet the contemporary needs of society and the job market. Through the introduction of competence-oriented teaching mode and the teaching form of race-teaching integration, it keeps up with the needs of enterprises and integrates into school-enterprise cooperation, laying a good foundation for cultivating high-level computer professionals. At the same time, this reform also provides useful experience and reference for the future development of practice courses in higher education. In the future, more innovative teaching methods will continue to be explored in order to continuously improve the quality of teaching and to form a set of practical teaching methods that meet students' career development and social requirements.
REFERENCES


