# Practical Teaching Design of Programming Course for Automation Major of New Engineering Education

Qinghua Yu, Bin Liu, Xiaohong Xu*, and Zongtan Zhou

(College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China
yuqinghua@nudt.edu.cn, 9916794@qq.com, xxhac311@126.com, narcz@163.com.)
(* Corresponding author: X Xu, E-mail: xxhac311@126.com)

**Abstract.** Under the background of new engineering education, automation major education faces new requirements of engineering practice reform. As the basic tool course of automation major, the programming course needs to further optimize the teaching content and strengthen practical teaching. This work designs the stepped and structured practical teaching contents as a new form of practice teaching and introduces simple in-class exercises, comprehensive in-class experiments, as well as extracurricular robotics practice competitions in the program design. The course trains students to transform their theoretical knowledge into engineering skills and improves their engineering practice and engineering innovation abilities. The programming course adopts a structured assessment and scoring method, emphasizing practical assessment, and comprehensively examines students' basic knowledge and theory, programming skills, and the ability to solve engineering problems. The novel practical teaching form strengthens the tool attribute of the programming course and enhances the students' ability to analyze and solve problems using computational thinking. Therefore, the new engineering education of automation major is strongly recommended.

**Keywords:** Programming course; Practical teaching; New engineering education; Automation major.

## 1 Introduction

New engineering education is an important development strategy of China's education reform. It has become an urgent demand to promote new engineering education to actively respond to a new round of scientific and technological revolution and industrial transformation, and further, to develop the new economy and to promote the reform and innovation of higher education [1]. The new engineering faces the needs of new industries and the new economy in the future, aiming to cultivate high-quality engineers with strong practical ability, innovation ability, and international competitiveness.


A. Holl et al. (Eds.): ICHESS 2022, ASSEHR 720, pp. 3000–3008, 2022.
https://doi.org/10.2991/978-2-494069-89-3_345

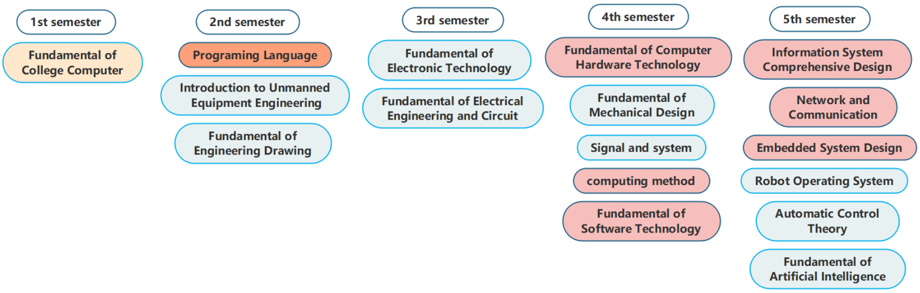
The new engineering education aims at the upgrading and transformation of industries such as intelligent manufacturing, cloud computing, artificial intelligence and robotics. It is necessary to promote the reform of higher education teaching. Since 2017, a series of construction concepts and practice models have been proposed in the practice of new engineering education, such as “Fudan Consensus” [2], “Tianda Action”[3], “Beijing Guide” [4], “Tianda Plan” [5], “Plan F” [6], “Chengdian Plan” [7], and “Tianda Plan 2.0” [8]. Today, the construction of new engineering has entered a new stage of deepening reform.

## 2 Overview of programming courses for the automation major

The training goal of the automation major is to master the basic theory, knowledge, methods, and skills in the field of automation. Graduates are expected to be competitive in demand analysis, frontier research, scheme design, theoretical calculation, technical breakthrough, project realization, system integration, experimental testing, as well as organization and management in the automation engineering domain, such as robotics engineering, unmanned systems, and artificial intelligence. Students majoring in automation should have strong abilities in system integration, reasoning and problem solving, and innovative practice.

Under the background of the new engineering, the education program for automation major requests that all the professional courses should deepen the reform of teaching content and innovate teaching methods. The education plan of the automation major in NUDT (National University of Defense Technology) points out that it is necessary to strengthen the teaching and training of students' innovative practical ability and build practical content. The accumulated class hours of practical teaching shall not be less than one-third of the total teaching hours. It also requires the curriculum to optimize the design of experimental teaching content and increase the proportion of comprehensive experiments. The proportion of the latter in the total experimental courses should not be less than 80%.

In order to adapt to the development of new engineering education, the teaching reform of the automation major has achieved some results [9]. A batch of courses aiming at high-quality and compound new engineering education have been built, which have greatly improved the practical ability and innovation ability of automation students. As a basic tool course of the automation major, programming courses play a cohesive role in a series of professional courses, as shown in Fig. 1. With the improvement of the requirements for students' practical ability in the series of professional courses, the programming course, as a basic tool course, needs to further optimize the practical teaching content, to reform the teaching mode, so as to improve students' practical ability and innovation ability.




（figure of author-drawn）

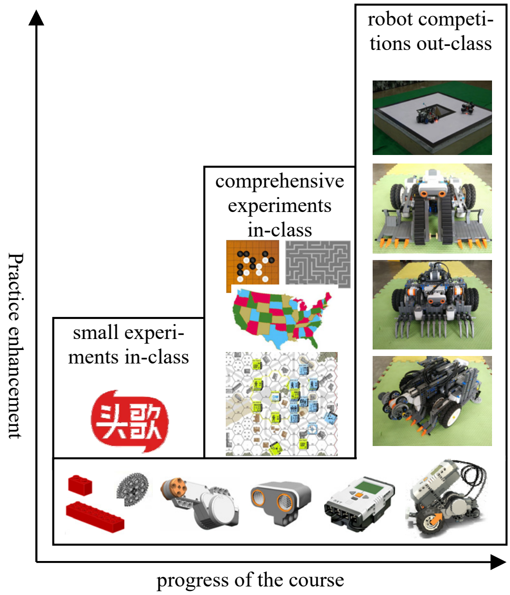
**Fig. 1.** Courses for automation major (excerpt)

Programming courses for the automation major aim to cultivate students' logical thinking ability and programming ability to solve practical engineering problems. In order to deepen the teaching reform of the new engineering education, the programming courses shall adhere to the student-centered concept, and practice the teaching form of "perceptual inspiration, practical strengthening, and ability improvement". Based on the CDIO (Conceive, Design, Implement, Operate) projective and cooperative learning mode, the courses inspire students to carry out project conceptualization, team formation, project analysis and design, as well as project implementation and assessment. Students' abilities of independent learning, team cooperation, and practical problems solving can be greatly improved.

## 3 Stepped and structured practical teaching design

As a tool course, programming not only teaches students to learn a programming language but also develops students' ability to use programming to solve problems. Therefore, practical teaching is very important to this course and should be used throughout the course. It is also necessary to set more practical tasks with engineering characteristics to constantly cultivate students' computer thinking and ability to solve practical engineering problems.

The programming course is open to automation major students in the first academic year. At this time, the students still maintain the passive learning habit in high school. Engineering practice ability and engineering innovation ability need to be gradually cultivated. Therefore, it is not advisable to set up practice projects that are too difficult at the beginning. In this work, the course has set up a stepped and structured experiment and practice content, as shown in Figure 2. Course practice includes small exercises in class, comprehensive experiments in class, and practical robot competitions outside class. Students will work through more difficult and even challenging exercises as the course progresses.



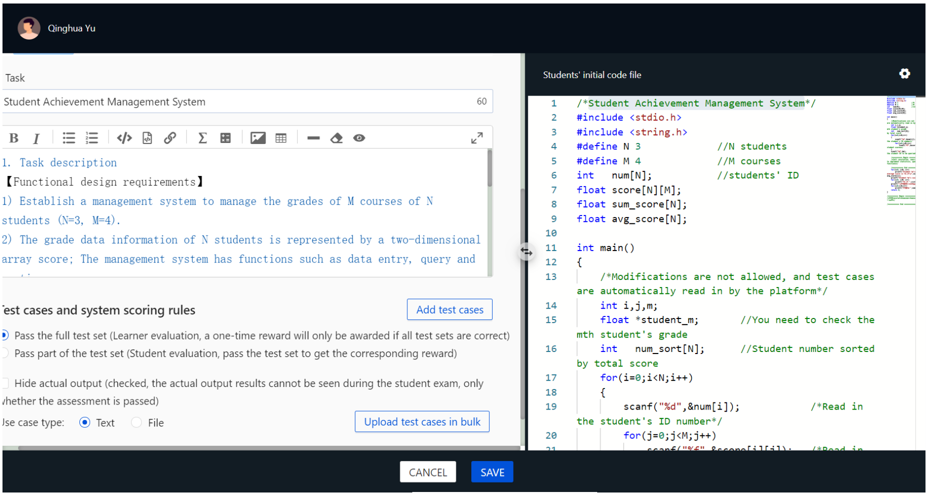

（figure of author-drawn）

**Fig. 2.** The stepped and structured practical teaching design

### 3.1 Small in-class exercises based on EduCoder

EduCoder is a practical teaching platform that provides online practical teaching services. It provides an innovative environment and has been widely used by universities and enterprises to improve their practical and innovation capabilities. It is the official cooperation platform of the MOOC Practical Teaching Working Committee of the China University Computer Education MOOC Alliance and the National Artificial Intelligence Vocational Education Group Practical Teaching Working Committee. Programming courses can be based on the EduCoder practical teaching platform for small exercises.

As the course progresses, small exercises will be released in the corresponding teaching session, and students will write codes in class to complete the assigned tasks. The difficulty of the small exercises is very flexible, usually completed in 5 to 10 minutes. A demonstration of a small exercise based on EduCoder is shown in Figure 3.




（figure of author-drawn）

**Fig. 3.** The demonstration of a small exercise based on EduCoder

Based on the real-time statistical feedback of the experimental results from the EduCoder platform, teachers can quickly and accurately analyze the learning effects, and then adjust the teaching speed during the teaching process.

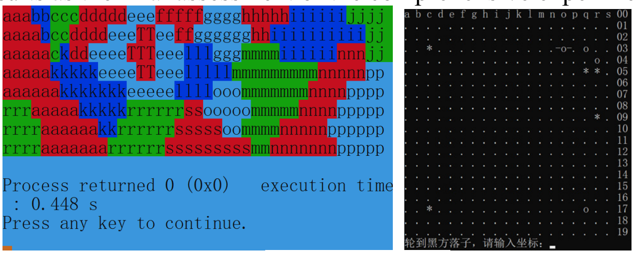
Students can practice the knowledge learned in the classroom through practical small exercises, improve the learning effect, and promote the transformation of knowledge into ability.

### 3.2 Comprehensive experiments with programming software projects

As a tool course, the programming course should set up a comprehensive practice project as a major experiment of the course to test the students' learning effect and practice ability. The comprehensive experiment of the course is arranged at the last session of the course, which takes 12 class hours.

The comprehensive experiment is designed with four optional projects, and students can choose one of them and complete the project in groups. The first project is a simulation exercise of military force deployment and confrontation. The second project is the analysis and coloring of military maps and confrontation. The third project is mobile robot exploration in an unknown environment. The fourth project is a simple game software of Go. Completion of the four projects requires a combination of all the knowledge and skills learned in the course. The objective of the projects is clear, but the approach is open-ended, which requires students to be open-minded and innovative in designing their own programs. Each project also provides human-computer confrontation as an optional supplementary practice content as a high-level challenge.

Comprehensive experiments are mainly completed in class. Students work in pairs and use personal computers for projects. The practice project is divided into three steps: opening defense, group implementation, and final assessment. The opening defense mainly reports the technical plan and organizational work of the project; after that, the team members carried out programming as planned to complete the experimental project. Furthermore, the final assessment introduces the main technologies and shows the project results as the final assessment of the comprehensive experiment.




（figure of author-drawn）

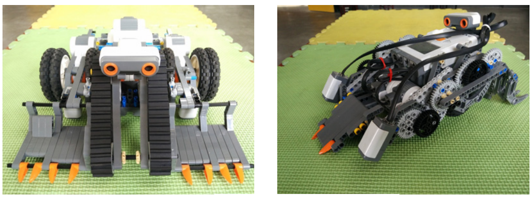
**Fig. 4.** Some project results of the practice project

Through the comprehensive experiment of the programming projects, the theoretical knowledge learned in class is comprehensively tested, the students' programming ability is improved through practice, and the ability to use computer programming to solve practical engineering problems is improved.

### 3.3 Practical robot competitions

The confrontational robot competition is introduced into the programming course as a comprehensive extracurricular practice project. As a practical activity of "combining software and hardware", robot competition is an important supplement to pure software programming in the classroom and an important embodiment of the new engineering education. The implementation of robot competition runs through the whole course and is an important supplement to classroom theoretical teaching.

The hardware platform used for the robotics competition is the Lego Mindstorms robot platform. Students need to independently design and assemble the robot system using the platform and program the software in the Bricxcc IDE. The participating robots should use sensors to percept the surrounding environment, use the robot controller to process the sensor information, and control the robot motors to perform the corresponding actions. Two participating robots in a game move independently in the arena according to the program. If the opponent robot falls out of the field or loses the ability to move, one robot wins. The robots designed and assembled by some students are shown in Figure 5.



（figure of author-drawn）

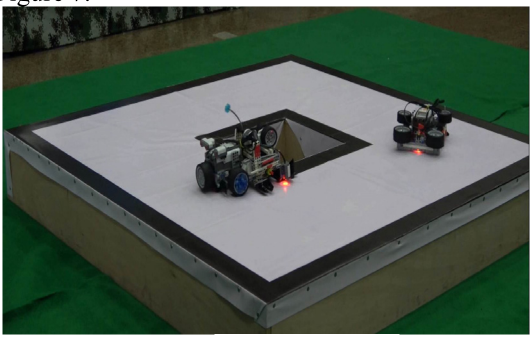
**Fig. 5.** The demonstrations of robots designed and assembled by the students

The practical teaching of robot competition adopts the mode of "task-driven, cooperative learning, and competitional assessment". Robot competition tasks will be arranged at the beginning of the course, and the tasks may drive knowledge learning. Students work in groups of three to enhance teamwork. Finally, the practice effect is evaluated through the competition.


（figure of author-drawn）

**Fig. 6.** The practical teaching mode of the robot competition

Students receive robot lab equipment at the beginning of the course. As the course progresses, they gradually complete the hardware design and software programming of the robot. Students continuously optimize and improve the functional performance of the robot and compete against each other at the end of the course. Team competition and elimination competition will be played against each other. The robot competition stage is shown in Figure 7.



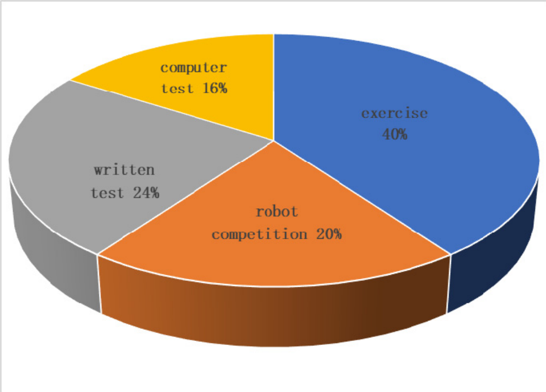
（figure of author-drawn）

**Fig. 7.** The stage of the robot competition

Through the engineering practice training of robot competition, students can deeply understand the professional characteristics of the combination of software and hardware in automation. Through the comprehensive practice of combining hardware and software, students' ability to use theory for practice is improved, and the ability to use programming to solve practical engineering problems is exercised.

## 4 Structured assessment of the course

The programming course adopts a structured method to assess the teaching performance, including formative assessment and summative assessment. Formative assessment includes exercise results, comprehensive experiment results, and robot competition rankings. The final assessment includes written test results and computer test results. The proportion of each part of the structured evaluation to the total score is shown in Fig. 8.




（figure of author-drawn）

**Fig. 8.** The proportion of the structured assessment of each part

The written test is organized according to the traditional examination form and scored objectively. The computer test is organized online based on the EduCoder platform. Students program on personal computers with limited network access. Computer test assessment scores are automatically scored online by the EduCoder platform.

The structured assessment of the programming course comprehensively evaluates the students' basic knowledge and theory, programming skills, and ability to solve engineering problems. Structured assessment is a comprehensive test of the students' learning effects of programming courses in the context of new engineering education.

## 5 Conclusion

New engineering education is an important development strategy of China's educational reform, which puts forward new requirements for the programming courses of the automation major. It is necessary to deepen the reform of teaching content, innovate teaching forms, and strengthen innovative practice. In this work, the programming course adheres to the student-centered teaching philosophy. The stepped and structured practical teaching form is proposed. Small in-class exercises, comprehensive in-class experiments, and extracurricular practical robot competitions are introduced to gradually transform their theoretical knowledge into engineering skills. Through structured assessment, the proportion of practical assessment is increased, and students' basic knowledge and theory, programming skills, and ability to solve engineering problems are comprehensively assessed.

The new practical teaching form proposed in this work strengthens the instrumental nature of the programming course, enhances students' ability to use computational thinking to analyze and solve problems, and strongly supports the new engineering education for the automation major.

## 6 References

1. Aihua Wu, Qiubo Yang, Jie Hao. The Innovation and Reform of Higher Education under the Leadership of Emerging Engineering Education ［J］. Research in Higher Education of Engineering, 2019(1):1-7.
2. Fudan Consensus of “New Engineering” [J]. Research in Higher Education of Engineering, 2017(1):10-11.
3. Route of “New Engineering” (“Tianda Action”) [J]. Research in Higher Education of Engineering, 2017(2):24-25.
4. Guide of “New Engineering” (“Tianda Action”) [J]. Research in Higher Education of Engineering, 2017(4):20-21.
5. Jianxing Yu, Ying Ji, Yang Yu, et al. The Key Reformation and Innovative Practice of Talent Training under the Emerging Engineering Education——Based on the Analysis of Tianjin University [J]. Journal of National Academy of Education Administration, 2020(3):71-77.
6. Song Gao. Implement Emerging Engineering Education F-Plan and Cultivate Engineering Leaders [J]. Research in Higher Education of Engineering, 2019(4):19-25.
7. Yong Zeng, Yan Huang, Guijun Xiang, et al. Design and Practice of UESTC New E3 Plan for Emerging Engineering: Starting from the Freshman Project Courses [J]. Research in Higher Education of Engineering, 2020(1):14-19.
8. Jia Liu, Kun Liu, Xiaoyan Liu. The “Tianda Plan” 2.0 for the new engineering education was released to further deepen and expand the new engineering construction [N]. Tianjin Education Daily, 2020-6-19(01).
9. Ming XU, Huimin LU, Xiaohong XU, et al. Comprehensive Practical Teaching System Based on the Whole Process of Engineering Training for “Unmanned” Majors [J]. Journal of Higher Education Research, 2022(1):98-101.