ISSN 2522-6398 Vol. 8, Issue 10: 120-126, DOI: 10.25236/FER.2025.081018

Teaching Reform in the Programming Course Group of Ethnic Universities under New Engineering

Haiyan Zhao[1,a,*], Zinan Zhang[2,b], Yufeng Diao[1,c], Qiang Guan[1,d]

[1]*Computer Science and technology college, Inner Mongolia Minzu University, Tongliao, China*
[2]*Department of Mechanical and Electrical Engineering, Tongliao Vocational College, Tongliao, China*
[a]*zhaohaiyan@imun.edu.cn,* [b]*1339051088@qq.com,* [c]*diaoyufeng@imun.edu.cn,*
[d]*guanqiang@imun.edu.cn*
[*]*Corresponding author*

Abstract: *New engineering disciplines in China are the reform orientations of engineering education proposed under the background of the new era, and teaching reform is an important link to cultivate talents with sustainable innovation ability and meet the graduation requirements of engineering certification. Taking the courses related to programming design in the School of Computer at Inner Mongolia Minzu University as an example, this paper analyzes the teaching problems of the programming curriculum group for students at local ethnic colleges and universities, and solves the relevant problems by reconstructing the curriculum group, forming a teaching system of theoretical practice integration, strengthening the practice links, and constructing a diversified evaluation index system. In addition, in view of the difficulty in integrating ideological and political education with professional knowledge, this paper adopts a hierarchical progressive idea to construct the ideological and political system framework of the curriculum group. In the process of teaching reform, on the one hand, it emphasizes the effective connection between courses in the curriculum group to achieve the integrated training from programming foundation to complex system design ability. On the other hand, in order to give full play to the advantages of school-enterprise cooperation, this paper implements the teaching reform that meets the requirements of the new engineering disciplines through methods such as co-construction of courses, joint writing of textbooks, and construction of online smart experimental teaching platforms.*

Keywords: *New Engineering Disciplines; Local Ethnic Colleges and Universities; Curriculum Group of Programming; School-Enterprise Cooperation; Curriculum of Ideological and Political Education*

1. Introduction

Currently, the curriculum system of university talent training programs is mainly divided into two categories: theoretical teaching practical teaching. Theoretical teaching provides guidance for practical teaching, and practical teaching allows theoretical teaching to be deepened[1]. New engineering, with new technology, new industries and new economies as the background, is a major strategic choice for the reform of engineering education in China and is the new thinking and new way of developing engineering education in China[2].

The group of programming courses includes "Advanced Language Programming", "Discrete Mathematics", "Data Structure" and "Algorithm Design and Analysis". All courses of the group are the core content of computer. From the computing thinking and programming ability of C++ language to the algorithm design and problem-solving ability of courses such as Discrete Mathematics, Data structure, Algorithm Design and Analysis. The achievement of these teaching objectives requires that the course implementation should be student-centered and outcome-oriented. Under the background of new engineering, combined with the actual situation of local national colleges and universities, the curriculum group aims to cultivate students' ability to think computationally and solve complex engineering problems, master the core technology of modern software development, and adapt to the multidisciplinary cross-innovation ability of the intelligent era. In line with the engineering certification standards, through the learning of the curriculum group, students should have professional technical capabilities such as design, system development, and data processing, as well as non-technical capabilities such as teamwork, project management, and innovative consciousness. In addition, from the perspective of requirements, engineering ethics and lifelong learning education can be strengthened,

ISSN 2522-6398 Vol. 8, Issue 10: 120-126, DOI: 10.25236/FER.2025.081018

while broadening international horizons.

The Computer Science and Technology College of Inner Mongolia Minzu University has only the Science and Technology major that has passed the engineering certification, and is currently in the continuous improvement stage. Other 4 majors including Software Engineering, Network Engineering, Big Data and Data Science, and Digital Media still cannot meet the requirements of engineering certification. In fact, due to the source of students and other reasons, the theoretical and practical teaching design of the 2019 and 2023 versions of the programming curriculum group of all majors still does not match the new engineering and engineering education requirements, so it is necessary to strengthen the teaching of "student-centered, results-oriented, and continuous improvement" [3,4].

2. Teaching Issues in Programing Course Group

The characteristics of students in local ethnic colleges are that they come from a wide range of sources, with admission scores ranging from 570 to 300. There are also some students who have been teaching in ethnic minority languages before entering university, resulting in uneven student levels. The courses in the programming course group are closely related to mathematics and English knowledge, which leads to some students having headaches when they see incorrect information such as Tian Shu or formulas; Teachers also face the problem of not being able to speak deeply and not being able to eat enough when teaching shallowly.

According to multiple interviews and evaluation records of students at Inner Mongolia Minzu University, students generally reflect that:

Firstly, the course knowledge points are complex, scattered, and easily confused when studying related programming courses[5]. The knowledge points of programming courses are complex. Taking binary relations in Discrete Mathematics as an example, it is necessary to master the concepts and operations of binary relations, judge the properties of relations, and apply equivalence relations and partial order relations. In the context of partial order relations, students must first distinguish between relation diagrams and Hasse diagrams. Subsequently, based on Hasse diagrams, they need to understand a series of concepts including maximal elements, minimal elements, the greatest element, the least element, suprema, and infima. These concepts present challenges for students in achieving a thorough grasp and applying them practically.

Secondly, the innovation and challenge of the experiment are insufficient. Due to insufficient experimental hours in the course group, students lack learning initiative, and the course itself is also a basic course. Taking Advanced Language Programming as an example, most of the 12 experiments are confirmatory experiments, which makes it difficult to cultivate students' innovative thinking ability.

3. Construction of Course Group System

To improve learning effectiveness, a combination of multiple teaching methods is adopted. In theoretical teaching, it is important to strengthen the internal connections and overall construction between courses, so that students can form a systematic knowledge system. In practical teaching, students are encouraged to apply their learned knowledge to practical problems through project practice, course design, and other methods, in order to improve their practical and innovative abilities and meet the needs of cultivating new engineering talents for future industries such as artificial intelligence, big data, and other interdisciplinary fields.

3.1 Refactoring Program Design Course Group

The course group includes four courses: Advanced Language Programming, Discrete Mathematics, Data Structures and Algorithm Design and Analysis. They are the core courses for cultivating students' programming abilities. Among them, Advanced Language Programming is the language foundation for implementing data structures and algorithms, Discrete Mathematics is the mathematical foundation for Data Structure and Algorithm Design and Analysis, and Algorithm Design and Analysis is at the highest level of the four courses.

3.1.1 Form a teaching object matrix

Under the background of engineering certification, we construct a teaching objective matrix for the course group. The teaching objective matrix of the course group is shown in Table 1. From the matrix, it

ISSN 2522-6398 Vol. 8, Issue 10: 120-126, DOI: 10.25236/FER.2025.081018

can be seen that all four courses have been matched with engineering certification requirements in terms of knowledge, ability, and quality dimensions, forming a knowledge cultivation system of language foundation→data structure→algorithm design. The implementation of this matrix follows a quality cultivation path of project driven→team collaboration→document writing→defense presentation.

Table 1: Teaching Objective Matrix

Course name	Knowledge dimension	Ability dimension	Quality dimension
Advanced Language Programming	Conditional control/Class and object	Debugging ability/Modular	Code standard consciousness
Discrete Mathematics	Predicate logic/Set theory	Logical reasoning/Abstract ability	Logical thinking
Data Structure	Storage structure/ADT	Complexity analysis/Optimization ability	Engineering thinking
Algorithm Design and Analysis	Greedy/Dynamic programming	Algorithm analysis/Data abstraction ability	Innovative thinking

3.1.2 Refactoring the course group

In order to achieve output oriented training objectives, the courses of the programming course group are restructured according to the graduation requirements of engineering education certification. Discrete Mathematics has been adjusted from 48 hours to 64 hours, and students are encouraged to use arrays, pointers, structures, etc. for programming implementation in sections such as first-order logic, sets and graph theory; Data Structure has added course design; Algorithm Design and Analysis has been adjusted from the original optional courses to restricted courses.

3.1.3 Integrate course content

Although there is continuity in the content of the course group, there is a large degree of jumping. We integrate all course contents of the course group to maintain continuity of teaching content. For example, when discussing divide and conquer or recursive algorithms, the implementation of tree structures in Discrete Mathematics and Data Structure is integrated with the algorithm. When discussing graph algorithms such as greedy/dynamic programming, we combine graph theory knowledge of Discrete Mathematics with graph storage of Data Structure; When debugging algorithms, it is actually a combination of programming skills, algorithm implementation, and engineering practice. Only in this way can students form a systematic knowledge system, laying a foundation for subsequent project practice.

3.1.4 Integration of multiple teaching and research rooms

Advanced Language Programming is a mixture of C programming and C++programming; Discrete Mathematics is the mathematical foundation of data structures and algorithms, and these two courses belong to Computer Application Teaching and Research Office. Data Structure is the core courses of computer science majors, and Algorithm Design and Analysis is the foundation for correctly analyzing complex engineering problems and modeling relevant data. These two courses belong to the Software Engineering Teaching and Research Office. By building a programming course group, it promotes communication among teachers from different teaching and research departments and breaks down barriers between them.

3.1.5 Build and share course resources

In order to give full play to the advantages of school enterprise cooperation, curriculum resource allocation is optimized and some related resources are shared. On the one hand, through school enterprise cooperation and the joint construction platform of China Software International, we can jointly develop and optimize course resources, build high-quality core courses and shared resource libraries. On the other hand, the real software projects of the enterprise should be appropriately simplified and decomposed into case studies corresponding to classroom teaching knowledge points, and it can also enhance teachers' practical ability.

3.1.6 Establish online and offline teaching and research models

For school enterprise co construction majors, it involves the continuity of courses before and after. For the courses in the course group, relying on the "Computer Application Course Virtual Teaching and Research Office" of the Ministry of Education, online discussions are held between teachers on campus and engineers from enterprises such as China Software International to jointly discuss the development

ISSN 2522-6398 Vol. 8, Issue 10: 120-126, DOI: 10.25236/FER.2025.081018

of course outlines. During the course teaching process, on campus teachers and enterprise engineers jointly study students' learning situation and mastery, and adjust the teaching plan appropriately.

3.2 Construct a Theoretical and Practical Integrated Teaching System

On the basis of output-oriented, we strive transform from "teaching-centered" to "student-centered" active learning and autonomous learning, and design a "connotation and extension integration, multi-level, and multimodel" theoretical and practical integrated teaching system.

3.2.1 Connotation and extension integration

The connotation of the curriculum group is learner-centered and ability-cult-oriented[6], aiming to cultivate students' programming design ability. Each theoretical course is equipped with in-class experiments, and the courses in the curriculum group are connected with theoretical courses through course design or training, forming a knowledge system of programming design curriculum group, so that students can have the ability to learn programming design courses independently, and realize the connotation and extension of the curriculum group. Moreover, the concept of ideological and political education is integrated into the teaching process.

3.2.2 Multi-level theoretical and practical integrated teaching system

In class experiments and independent experimental courses adopt a multi-level training mechanism from shallow to deep, and the in-class experiments of theoretical teaching adopt case-driven practical ways, and independent and course design courses adopt project-driven practical ways, focusing on systematic design and software development capability training, emphasizing the interconnection and integration of course knowledge points. Moreover, in order to strengthen students' innovation capability, we encourage students to participate in disciplinary competitions and entrepreneurship training.

3.2.3 Multi-model theoretical and practical integrated teaching system

Teachers transform into organizers, guides, and curriculum designers. The theoretical courses in the curriculum group adopt online and offline blended teaching; experimental courses adopt multi-level heuristic practical models, and combine with a of self-learning models after class. AI teaching technologies are introduced, such as Zhihui Tree, XuexiTong, and other online learning platforms, providing students rich learning resources such as PPT, videos, and interactive communication platforms. Except for the course of Algorithm Design and Analysis, the other three courses have constructed a knowledge graph. We jointly building an online intelligent experimental teaching platform with China Software International, in order to provide students with a more personalized and efficient learning experience.

3.3 Strengthen Practical Training

Practical training is a key to improving students' programming and problem-solving abilities. Strengthening practical training is mainly carried out in the following three aspects:

3.3.1 Increasing the weight of laboratory courses

Through laboratory, students can master programming skills, data structures, and algorithm applications. In the 2023 version of the training program, a separate advanced language programming experiment course is a set as an independent experiment course, and a comprehensive experiment is set up in the first semester of the sophomore year in the form of a course design. Algorithm Design and Analysis course has increased 12 experimental hours.

3.3.2 Creating a multi-level innovative training mechanism

Through increasing project-driven practice and assigning multi-level innovative experiments and training tasks, a multi-level innovative training mechanism is created. In the first semester, a series of case practice tasks closely related to the content of Advanced Language Programming are designed. In the second semester, students apply C++ language to complete project-driven course design tasks, such as student management systems. In the third semester, students implement basic data structure problems in combination with data structure course teaching, and extend this basis. For example, based on linked list graph data structures, we design corresponding algorithms for graph traversal and shortest path solution. In the fourth semester, students are encouraged to participate competitions such as Blue Bridge Cup, National Collegiate Programming Contest. In the fifth semester, through school-enterprise cooperation, students are guided to participate in real projects from enterprises or innovative

ISSN 2522-6398 Vol. 8, Issue 10: 120-126, DOI: 10.25236/FER.2025.081018

entrepreneurship training. For example, in partnership with a company, students are involved in designing and optimizing the product recommendation algorithm for an online shopping platform.

3.3.3 Writing relevant practical textbooks

To build a suitable program design course group for computer science at ethnic local universities better, and to promote the implementation of basic experiments and practice, the research team is collaborating with from Zhongruan International to write a textbook. The textbook's name "C++ Programming Practice", and it is focused on Advanced Language Programming, with the core concept of " consolidating the foundation, strengthening practice, and cultivating algorithmic thinking". The content design closely follows the examination points of programming contests such as the Blue Bridge Cup and the National Collegiate Programming Contest, taking into account both procedural and object-oriented ideas. This helps students master the core skills of C++ language and improve their ability to solve practical problems. Through three modules: in-class experiments, competition algorithm training, and project development, a complete learning path from basic syntax to engineering practice is constructed. The in-class experiment part provides a complementary lab and after-class self-practice for Advanced Language Programming; the basic competition and algorithm part includes training for sorting, searching, greedy algorithm, and other content suitable for the Blue Bridge Cup and other programming contests, which is beneficial the expansion of algorithmic thinking; and the comprehensive project development is used as reference content for course design.

3.4 Diverse Evaluation Indicator System Construction

During the course of teaching, process and result evaluations are collected from four perspectives: course assessment, student, peer evaluation and supervisor evaluation. Through quality monitoring and assessment, problems are identified, feedback is provided in a timely manner, and continuous improvement is made to the course group. In this process, course assessment includes not only routine homework but also adds pre-class tests, post-class tests, unit tests, and other components. For courses with experiments, experiment content is evaluated through classroom performance, lab reports, and online exams, and the degree of achievement of course objectives is analyzed. The course design is conducted by enterprise instructors using multidimensional assessment mechanisms such as peer mutual evaluation, process evaluation, project outcome defense, and innovation points. At the same time, the work of graduate tracking and enterprise feedback is strengthened to a continuous improvement loop of student evaluation→ graduate tracking→enterprise feedback→course iteration.

4. Strengthen the Construction of Course Ideology and Politics

4.1 Difficulties in the Integration of Course Ideology and Politics Professional Knowledge

Engineering education focuses on classroom teaching, supplemented by a small amount of experiments, internships, and other practical teaching links. Teachers lack the motivation to innovate methods, and practical teaching often focuses on skill verification and proficiency improvement. Students are mostly in a passive state of acceptance in the classroom. However, course ideology and politics need to diversified and interactive teaching methods to stimulate students' initiative in thinking. There are the following problems in promoting the construction of course ideology and politics in the current programming course group:

4.1.1 Shallow Element Mining

Some teachers' mining of ideological and political elements stops at the surface level, such as simply adding patriotic slogans in code comments and making a comparison of "China High-speed Railway Speed" when explaining sorting algorithms. There is a lack of deep correlation with the essence of knowledge, forming a "labeling" phenomenon.

4.1.2 Lack of Course Coordination

The four courses in the course group should form a progressive chain of ideological and political education, but in reality, they often operate independently. For example, the "Graph Theory Basis" in Discrete Mathematics could naturally connect the "Graph Storage" in Data structure and the "Shortest Path" in algorithms, and integrate a case of independent innovation of China's Beidou navigation system to form a through ideological and political main line, but it is often fragmented in actual teaching.

ISSN 2522-6398 Vol. 8, Issue 10: 120-126, DOI: 10.25236/FER.2025.081018

4.2 Constructing a Hierarchical and Progressive Ideological and Political System Framework

In response to the difficulty of integrating ideological and content with professional knowledge, it is necessary to construct a multi-dimensional and systematic framework for the implementation of ideological and political education in courses, break down course barriers, and achieve deep of ideological and political elements with the knowledge system. The program design course group can adopt a hierarchical and progressive approach to construct the ideological and political system framework:

Firstly, some ideological and political elements are embedded into the key knowledge points of each course.

Secondly, a chain of connections which is around ideological and political themes is established between courses. For example, the theme of "independent innovation in technology" can be integrated into the discussion of Chinese postman problem in graph theory in Discrete Mathematics, the Beidou navigation route optimization algorithm; in Data Structure, the domestic improvement scheme of graph storage structure such as Huawei's index optimization; in Algorithm Design and Analysis, case of independent breakthrough of the shortest path algorithm such as Gaode map real-time navigation optimization; and in the process of curriculum design, develop a traffic network optimization simulation system and the scenario of rural revitalization.

Thirdly, we design comprehensive projects that integrate ideological and political goals in project-based practice based on the concept of engineering education. For example, we can design a data dictionary through hash storage of Data Structure or implement a dynamic planning prediction model through the related algorithm of Algorithm Design and Analysis.

Finally, to build an education ecosystem which is composed of teacher collaboration, resource collaboration and platform collaboration, we integrate the resources of course group. To finish it, we organize professional course teachers, ideological and political teachers, and enterprise mentors to work together. For example, we guide the design of the "algorithm ethics" module with the teachers from our school's Marxism College; we establish some thematic resource libraries such as "algorithm case library" and "the development history of domestic technology". In addition, we rely on virtual teaching and research offices to achieve sharing of ideological and political resources among schools.

5. Conclusions

This paper first analyzes the teaching issues of programming courses in the context of new engineering, and then completes the construction of the curriculum system through four aspects: reconstructing the programming curriculum group, constructing a teaching system with integrated theory and practice, strengthening the practice links, and perfecting the diversified evaluation system. Finally, by constructing a multi-dimensional and systematic framework for the implementation of curriculum ideology and politics, the barriers of courses are broken, and the deep integration of ideological political elements and the knowledge system is realized. In the process of curriculum group construction, it is insisted on following the educational philosophy of new engineering and engineering education certification, giving full play the advantages of school-enterprise cooperation, strengthening the organic connection between courses, and fully integrating curriculum ideology and politics, multi-level innovation training, and subject competitions into the teaching reform so as to achieve the integrated training from programming basics to complex system design capabilities, and enhance the innovation and practical abilities of students in local ethnic colleges and universities. However, the reform is a continuous improvement process. In the future, we will continue to explore more effective teaching methods and means, such as establishing a dynamic capability portrait tracking system, developing a hierarchical case library, and strengthening the pertinence and effectiveness of curriculum ideology and politics, so as to further improve the teaching quality of computer programming courses in local ethnic colleges and universities.

Acknowledgements

14th Five-Year Plan for Education Science Research in Inner Mongolia Autonomous Region 2024 Annual Project NGJGH224242 and NGJGH2024236; General Project of the Teaching and Education Research Project of Inner Mongolia Minzu University YB204005; Innovation and Entrepreneurship Start-up Support Plan for Returned Overseas Students in Inner Mongolia Autonomous Region 2024LXC003.

ISSN 2522-6398 Vol. 8, Issue 10: 120-126, DOI: 10.25236/FER.2025.081018

References

[1] Shen Y. Cultivating new engineering talents who are practical and future-oriented[N]. China Education Daily, 2024-11-26.
[2] Shi Y, Mao Z Y, Li H Q. Exploration and practice of new education mode in local colleges and universities based on accreditation of engineering education and "New Engineering"[J]. Theory and Practice of Contemporary Education, 2022(01):88-92.
[3] Zhong D H. Connotation and action for establishing the emerging engineering education[J]. Higher engineering education research, 2017(03): 1-6.
[4] Yang K F, Shi Z Y. Innovative construction of digital learning mode based on OBE concept and multiplatform integration[J]. Computer Education, 2022(11): 87-91.
[5] Zhang C, Lv W G, Li J X. Reform of mixed teaching mode for core courses in colleges and universities driven by information technology[J]. Computer Education, 2021(07): 188-192.
[6] Dai H M, Dai H L. Research on software engineering application-oriented talent training program based on CDIO concept under the background of new engineering[J]. Computer Education, 2020(01): 64-67.