

Jinggangshan Spirit and Algorithmic Education: Instructional Design and Practical Exploration of Ideological-Political Integration in a Java Programming Course



Daifu Zha^{1,*}, Ni Ma¹, Qingyun Zou¹, Tangshu Lv¹ & Guojun Zha²

¹Jiujiang Institute of Technology, P.R. China

²Xinyu University, P.R. China

Abstract: Ideological-political integration in curriculum represents a key approach to fostering moral character. This study proposes a pedagogical framework characterized by “algorithmic education as the foundation and ideological-political cultivation as the core,” integrating the Jinggangshan Spirit into the teaching of a Java programming course. To address the problem of rigid integration of ideological-political elements, the study systematically identifies value-oriented elements — such as commitment to national development through science and technology and the spirit of craftsmanship—within topics including the history of Java development, exception handling, and multithreading. Case-based instruction and project-oriented practice are employed to embed red cultural resources into technical teaching. Practice indicates that this model effectively enhances both students’ technical competence and value identification, facilitating a transformation from knowledge acquisition to holistic personal development. The study provides a practical reference for cultivating digital talents who possess technical expertise, humanistic commitment, and a strong sense of responsibility.

Keywords: Ideological-Political integration in curriculum, Java programming, Jinggangshan spirit, fostering moral character, instructional design

1. Introduction

“Algorithmic education as the foundation and ideological-political cultivation as the core” represents an essential orientation for computer science education in the new era. As a core compulsory course, Java Programming plays a crucial role in cultivating digital talents. The Guidelines for Ideological-Political Integration in Higher Education Curriculum emphasize that science and engineering courses should incorporate scientific and engineering ethics, foster the spirit of craftsmanship, and cultivate students’ sense of social responsibility and commitment to national development (Ministry of Education of the People’s Republic of China, 2020).

However, the implementation of ideological-political integration in Java Programming currently faces three major challenges. First, the cultivation of disciplinary knowledge and value guidance remains insufficiently integrated, often resulting in mechanical insertion and fragmentation (Liu et al., 2024). Second, the teaching lacks contextualization within China’s technological

development, with limited incorporation of locally grounded narratives (Xiong & Xu, 2024). Third, the integration remains superficial and does not extend to the algorithmic level of programming practice.

The Jinggangshan Spirit—characterized by firm ideals, perseverance, a commitment to seeking truth from facts, the courage to explore new paths, reliance on collective effort, and determination to achieve success—provides important insights for ideological-political integration in curriculum. Taking the Java Programming course as an example, this study explores an instructional design framework of “algorithmic education as the foundation and ideological-political cultivation as the core” (Zhang & Jia, 2023).

2. Reconstruction of Instructional Objectives for Ideological-Political Integration in Curriculum

The Java Programming course is typically offered in the second year of undergraduate study, a critical stage for value formation. Guided by the Outcome-Based Education (OBE) framework (Zhang & Jia, 2023), the instructional objectives are

Corresponding Author: Daifu Zha

Jiujiang Institute of Technology, Jiujiang City, P.R. China

©The Author(s) 2026. Published by BONI FUTURE DIGITAL PUBLISHING CO., LIMITED. This is an open access article under the CC BY

License(<https://creativecommons.org/licenses/by/4.0/>)

reconstructed into three dimensions.

First, knowledge objectives focus on mastering Java syntax, object-oriented programming, core APIs, and development frameworks.

Second, competency objectives emphasize the ability to analyze problems, design solutions, and develop high-quality code using Java, while cultivating innovation, engineering practice skills, and teamwork.

Third, value objectives, aligned with fostering moral character, aim to cultivate students' commitment to national development through

science and technology, engineering ethics, and the spirit of craftsmanship, facilitating a transformation from knowledge acquisition to holistic personal development (Diao & Fu, 2014).

3. Systematic Identification of Ideological-Political Elements in the Java Programming Course

Based on the insights of the Jinggangshan Spirit (Xu & Song, 2025), this study systematically identifies ideological-political integration points across six dimensions within the Java Programming curriculum.

Table 1 Mapping of Ideological-Political Elements and Instructional Objectives in the Java Programming Course

Dimension	Java Knowledge Modules	Entry Points for Ideological-Political Integration	Educational Objectives	Implementation Approaches
Firm Ideals and Convictions	Technology Stack Evolution	Indigenous innovation in Java technologies (e.g., Alibaba Dubbo; Huawei Kunpeng)	Strengthening national identity and sense of mission	Lectures; video-based cases
Perseverance and Hard Work	Algorithm Optimization; Performance Tuning	Case studies on overcoming complex technical challenges in scientific research	Developing perseverance and problem-solving resilience	Project-based practice
Seeking Truth from Facts	Exception Handling; Debugging Techniques	Fault tolerance mechanisms and engineering ethics; bottom-line thinking in coding practices	Cultivating scientific spirit and rigorous thinking	Case analysis
Courage to Explore New Paths	Design Patterns; Framework Innovation	Innovation narratives in China's Internet technology development	Fostering innovative thinking and initiative	Extended readings; discussions
Reliance on Collective Effort	Team Collaboration; Open-Source Communities	Collaborative models in GitHub-based open-source development	Promoting openness, sharing, and teamwork	Group projects
Determination to Achieve Success	Complex Project Development	Participation in disciplinary competitions and tackling advanced technical challenges	Enhancing self-efficacy and striving for excellence	Innovation-driven projects

Case of Integrating Red Cultural Resources: In the multithreading module, a project titled the "Jinggangshan Scenic Area Ticketing System" is introduced to embed ideological-political integration in curriculum within algorithmic education. Students are guided to examine issues of public order and fairness in concurrent control, particularly in handling high-concurrency requests and ensuring equitable access during peak periods of red tourism (Yi et al., 2010).

4. Implementation Pathways for Ideological-Political Integration in Curriculum

4.1 Systematic planning of instructional content Case 1: Dialectical thinking in recursive algorithms

Technical focus: Recursive definitions and invocation processes.

Ideological-political focus: The concept of recursive self-calling is associated with perseverance and the intergenerational transmission of values. The

classical philosophical metaphor of Zhuang Zhou's "butterfly dream" is introduced to deepen reflective thinking (Guo & Yang, 2018).

Instructional process: Tower of Hanoi → recursive visualization → short video on the Jinggangshan context → discussion of recursive thinking in Chinese cultural traditions.

Educational outcomes: Enhancing students' appreciation of cultural heritage and strengthening value identification through engagement with the Jinggangshan Spirit.

Case 2: Bottom-Line thinking in exception handling

Technical focus: Exception types and the try-catch-finally mechanism.

Ideological-political focus: The Therac-25 accidents are used to introduce bottom-line thinking and engineering ethics.

Instructional process: Presentation of accident cases → discussion on professional ethical boundaries for programmers → explanation of safety-oriented exception handling design.

Educational outcomes: Cultivating rigorous programming habits and strengthening professional ethics.

Case 3: Rule awareness in multithreading

Technical focus: Thread synchronization, locks, and deadlocks.

Ideological-political focus: A project-based scenario—the "Jinggangshan Scenic Area Ticketing System"—is used to highlight public order and fairness in concurrent control (Xiong & Xu, 2024).

Instructional process: Simulation of concurrency conflicts → analysis of real-world system failures → debate on the fairness of ticket-grabbing software.

Educational outcomes: Promoting awareness of rules, fairness, and social responsibility in algorithmic contexts.

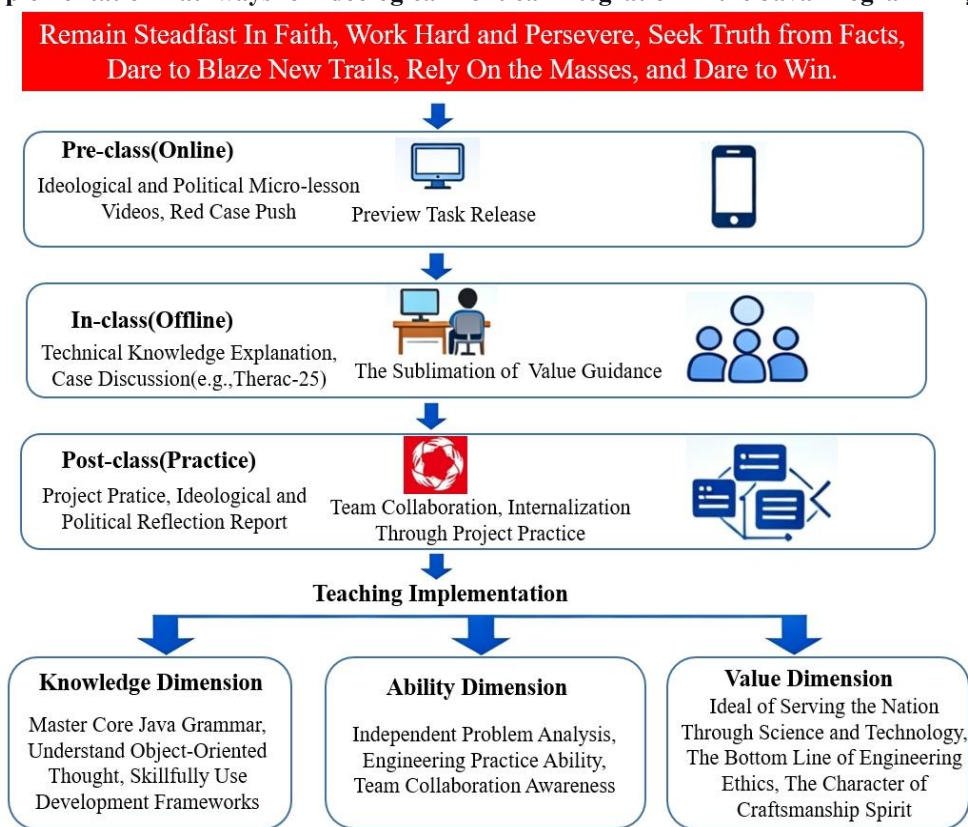
4.2 Optimization of instructional strategies

Traditional one-way lecturing is replaced by diversified interactive approaches, including group discussions and code reviews (see Figure 1) (Du & Qiang, 2024). Narratives of China's independent software innovation are incorporated to ensure that knowledge acquisition and value cultivation are mutually reinforcing.

The principle of "educators cultivating themselves before cultivating others" is emphasized, with instructors modeling craftsmanship through

punctuality, integrity, and patient guidance. The Jinggangshan Spirit is systematically embedded throughout the teaching process: establishing firm ideals through introductory sessions on China's software development; fostering perseverance during programming practice; cultivating a truth-seeking attitude through debugging; encouraging innovation through multiple solution strategies; reinforcing collective collaboration through pair programming; and promoting determination to achieve success through participation in competitions (Xu & Song, 2025).

Figure 1 Implementation Pathways for Ideological-Political Integration in the Java Programming Course



4.3 Construction of a scientific assessment and evaluation mechanism

At the student level, a balanced evaluation system is established, consisting of process-based assessment (50%) and outcome-based assessment (50%). The process-based component includes code quality and coding standards (10%), teamwork (10%), ideological-political reflection (10%), classroom participation and discussion (10%), and literature review presentations (10%). The outcome-based component includes project evaluation, in which the “social value” of the project accounts for 15%, as well as written examinations incorporating open-ended ethical questions (Du & Qiang, 2024).

At the teacher level, instructional quality is evaluated through student feedback and peer observation.

For continuous improvement, the course design and implementation are systematically reviewed and refined on an annual basis to ensure alignment with both algorithmic education objectives and ideological-political integration in the curriculum.

5. Analysis of Implementation Effectiveness

During the 2024–2025 academic year, an experimental class (n = 45) adopted the ideological-political integration model, while a

control class (n = 43) followed a traditional instructional approach.

5.1 Improved discriminative power of assessment

Following the instruction on recursive algorithms, the discrimination index increased from 0.25 to 0.39, representing a 56% improvement. In addition, student performance in exception handling and multithreading modules was significantly higher in the experimental group compared to the control group.

5.2 Value-Oriented shift in project selection

In the experimental group, 42.2% of student projects focused on socially oriented themes (e.g., “Jinggangshan Red Tourism Guidance System”), compared to only 11.6% in the control group. Students reported an increased awareness that “technology should serve fairness and justice” (Diao & Fu, 2014), indicating a strengthened alignment between technical learning and value cultivation.

5.3 Positive student feedback

Survey results are presented in Table 2, indicating overall positive student perceptions of the integration model.

Table 2 Survey Results on the Effectiveness of Ideological-Political Integration in the Java Programming Course

Survey Items	Experimental Group (Agreement Rate)	Control Group (Agreement Rate)
The course enhanced my understanding of programmers' social responsibility	93.3%	67.4%
I began to consider the social impact of technology during programming	91.1%	60.5%
The course strengthened my sense of national identity	88.9%	55.8%
I paid greater attention to coding standards and documentation practices	95.6%	79.1%
I am more willing to contribute to open-source communities	84.4%	46.5%

Qualitative Feedback: Qualitative feedback further supports the effectiveness of ideological-political integration in the curriculum. One student noted that “the lesson on recursion helped me understand the connection between algorithms and traditional Chinese culture” (Student Li). Another student commented that “the course not only taught us programming skills but also guided us in personal development, from coding standards to teamwork” (Student Wang).

5.4 Teacher development and reflective practice

Effective ideological-political integration in curriculum requires sustained teacher engagement and reflective practice. The metaphor of “treating a course as one would nurture a child” underscores the need to thoughtfully embed value-oriented elements into disciplinary teaching. In this sense, instructors are expected to internalize and seamlessly integrate ideological-political elements—like “salt dissolved in soup”—within professional knowledge domains. This requires teachers to systematically explore the value implications embedded in technical content and to align algorithmic education with fostering moral character (Yi et al., 2010).

6. Conclusion and Implications

6.1 Key findings

This study yields four main conclusions.

First, the reconstruction of instructional objectives is foundational, particularly through the establishment of a three-dimensional framework encompassing knowledge, competencies, and values (Zhang & Jia, 2023).

Second, the systematic identification of ideological-political elements is critical for meaningful integration (Liu et al., 2025).

Third, pedagogical innovation—such as case-based and project-oriented approaches—serves as an essential mechanism for implementation (Du & Qiang, 2024).

Fourth, assessment reform functions as a guiding force in reinforcing both technical and value-oriented learning outcomes.

Overall, the proposed model enhances students' learning engagement and strengthens their sense of social responsibility and commitment (Liu et al., 2024).

6.2 Limitations and future directions

Despite its contributions, this study faces several limitations. These include variability in teachers' pedagogical competencies, insufficient depth of integration at the algorithmic level, the lack of long-term evaluation mechanisms, and limited resource sharing across institutions.

Future research may focus on the following directions:

- (1) leveraging artificial intelligence to support ideological-political integration in curriculum;
- (2) strengthening collaboration between academia and industry for talent cultivation;
- (3) promoting the digitalization of red cultural resources, particularly through regionally grounded cases (e.g., Jiangxi-based contexts) (Xu & Song, 2025);
- (4) conducting longitudinal studies to assess the sustained impact of the proposed model.

7. Concluding Remarks

The transformative influence symbolized by the Jinggangshan Spirit continues to resonate in contemporary education. Within the Java Programming course, educators act as facilitators of

both technical knowledge and value transmission, enabling algorithms to carry meaning beyond computation and embedding ethical awareness within coding practices. The ultimate goal is to cultivate digital talents who demonstrate both technical competence and moral integrity.

As emphasized by Vasily Sukhomlinsky in *My Heart I Give to Children*, education should not merely produce technically skilled individuals, but well-rounded persons with rich spiritual lives, ideals, moral character, and a sense of responsibility toward others and society (Sukhomlinsky, 1984).

Conflict of Interest

The authors declare that they have no conflicts of interest in this work.

Acknowledgement

This research was supported by 1. Phase 4 of the Employment Internship Base Project between Jiangxi SF Express Co., Ltd. and Jiujiang Institute of Technology (Project No. 2025033138434). 2. Science and Technology Research Project of the Education Department of Jiangxi Province (Project No. GJJ2502102). 3. Teaching Reform Project of the Education Department of Jiangxi Province (Project No. JXJG-20-20-4).

References

- Diao, L., & Fu, R. (2014). Innovative practice of ideological-political teaching in new engineering courses under the background of integrated higher education: A case study of artificial intelligence education. *Higher Education Journal*, (13), 59–63.
- Du, H., & Qiang, Y. (2024). Innovation and practice in the construction of ideological-political teaching resources and platforms in computer science education. *Computer Education*, (8), 30–35.
- Guo, Y., & Yang, J. (2018). Flipped classroom design for programming courses based on the ARCS motivation model: A case study of recursive functions. *Computer Education*, (6), 140–144.
- Liu, S., Ma, X., & Jia, Y. (2024). Exploration of teaching reform in Java programming courses oriented toward ideological-political integration in curriculum. *Modern Business Trade Industry*, (24), 266–268.
- Liu, Y., Zhang, L., & Lü, L. (2025). Exploration and practice of ideological-political integration in

computer network laboratory courses. *Computer Education*, (5), 44–48.

Ministry of Education of the People’s Republic of China. (2020, May 28). *Guidelines for ideological-political integration in higher education curriculum*. Ministry of Education official document.

Vasily Sukhomlinsky. (1984). *Suggestions for teachers*. Beijing: Educational Science Publishing House.

Xiong, C., & Xu, Y. (2024). Design and practice of ideological-political integration in higher vocational computer courses: A case study of Java Web application development. *Computer Knowledge and Technology*, 20(6), 133–135.

Xu, W., & Song, G. (2025). Integrating the Jinggangshan Spirit into ideological and belief education for university counselors: Methods and pathways. *Journal of Dalian University*, 45(4), 96–101.

Yi, H., Xu, Z., & Zhang, H. (2010). Reform and innovation in the teaching of programming fundamentals. *Computer Education*, (8), 46–49.

Zhang, X., & Jia, M. (2023). Instructional design and implementation based on “OBE + ideological-political integration” under the background of new engineering education: A case study of Java programming. *Journal of Nanyang Normal University*, 22(1), 60–63.

How to Cite: Zha, D., Ma, N., Zou, Q., Lv, T., & Zha, G. (2026). Jinggangshan Spirit and Algorithmic Education: Instructional Design and Practical Exploration of Ideological-Political Integration in a Java Programming Course. *Contemporary Education and Teaching Research*, 07(05), 139-144.
<https://doi.org/10.61360/BoniCETR262020090502>