# Evaluating ChatGPT's Impact on Java Programming: Experimental Insights and Educational Implications

Zhihao Cao[a], Menglin Liu[*]

College of Information Science and Engineering, Zaozhuang University, Zaozhuang, China

[a]zhhacao@gmail.com
[*]jssxss@163.com

**Abstract.** This study examines the impact of ChatGPT on the education industry, specifically focusing on Java programming instruction. Using the Java Programming Design course at Zaozhuang University as a case study, it analyzes ChatGPT's capabilities from a programming perspective. The analysis highlights ChatGPT's high efficiency in answering questions and its strong programming abilities, while also acknowledging its limitations, such as the inability to interpret charts. The study concludes by proposing the integration of ChatGPT as a supplementary tool alongside traditional teaching methods, utilizing its strengths to optimize various stages of the teaching process.

**Keywords:** Java Programming; ChatGPT; Experimental Teaching; Educational reform

## 1 Introduction

As a foundational course for computer science major, Java programming is widely offered at universities. Currently, more than 80% of Chinese universities offer Java programming courses, and it has become a core component of science and engineering training programs [1-2]. However, due to differences in students' abilities, ensuring that all students master the Java language remains a major challenge in teaching.

In the past two years, ChatGPT, with its extensive knowledge base and capabilities in contextual understanding and memory, has been able to perform various complex tasks. It has demonstrated significant potential in the field of computer science by assisting in writing articles, speeches, and even generating codes based on natural language instructions. Furthermore, ChatGPT can aid in code optimization and provide suggestions for system development. These abilities help researchers in scientific writing, thereby accelerating the pace of innovation. With the advent of ChatGPT, its ability to quickly respond to requirements and generate code through conversational models has sparked extensive discussions about changes in the computer programming industry and concerns over AI potentially replacing programmers [3-4].

In this paper, we examined the impact of ChatGPT on the Java Programming, and analyzed ChatGPT's capabilities from a programming perspective.


D. Hu et al. (eds.), *Proceedings of the 2024 5th International Conference on Modern Education and Information Management (ICMEIM 2024)*, Atlantis Highlights in Social Sciences, Education and Humanities 29,
https://doi.org/10.2991/978-94-6463-568-3_42

## 2 Experimental Analysis of ChatGPT's Abilities

This section presents the results of experiments conducted with ChatGPT-4, followed by discussions with programming professionals, ultimately concluding with an analysis of its capabilities in Java programming, particularly in code generation.

### 2.1 Design of Experiment

Firstly, eight final exam questions from the 2024 Java Programming Design course at Zaozhuang University were selected. These questions present a smooth difficulty curve, and the scores of over 500 students exhibit a approximately normal distribution. Detailed information on the eight programming tasks is provided in Table 1. The difficulty levels in Table 1 range from 1 to 5, where 1 is the easiest and 5 is the hardest. Specifically, tasks with difficulty levels of 1-2 are considered easy, 3 as medium, and 4-5 as difficult. Each question is evaluated using 5 test cases, with each test case worth 20 points. A perfect score of 100 is awarded only if all test cases are passed.

**Table 1.** Brief Overview of the Eight Programming Problems Used in the Experiment.

| Number | Difficulty | Problem Characteristics |
|---|---|---|
| (1) | 1 | Basic concept problem, straightforward and easy |
| (2) | 2 | Basic concept problem, requiring attention to detail |
| (3) | 2 | Basic concept problem, requiring attention to detail |
| (4) | 3 | Simulation problem, straightforward and easy to understand |
| (5) | 3 | Simulation problem, straightforward and easy to understand |
| (6) | 4 | Complex application, involving numerous formulas |
| (7) | 4 | Complex application, involving numerous formulas |
| (8) | 5 | Complex mathematical operations, with extensive diagrams and examples |

Secondly, ChatGPT-4, a widely recognized model, was selected in the programming experiment to respond the eight questions. ChatGPT assists with Java programming tasks by leveraging its pre-trained natural language processing capabilities to interpret user requirements. Initially, it analyzes the user's query to determine specific programming needs and objectives. Then, based on these requirements and its understanding of Java's structure and syntax, it generates the corresponding code. Throughout this process, ChatGPT considers Java programming conventions and best practices to ensure that the generated code meets user requirements while being readable and efficient.

Thirdly, after ChatGPT generates the programming code, its accuracy is first evaluated using an Online Judge System [5], and then three expert evaluators assess its compliance with coding standards.

## 2.2 Programming Experiment and Analysis of Results

Three researchers completed the code generation and evaluation for eight Java programming questions using ChatGPT-4. The highest score for each task and the corresponding minimum number of interactions needed to achieve that score were summarized, as shown in Table 2.

**Table 2.** Summary of ChatGPT-4's Responses.

| Number | Difficulty | Highest Score | Minimum Interactions |
|---|---|---|---|
| (1) | 1 | 100 | 5 |
| (2) | 2 | 100 | 4 |
| (3) | 2 | 100 | 4 |
| (4) | 3 | 100 | 8 |
| (5) | 3 | 100 | 9 |
| (6) | 4 | 20 | 8 |
| (7) | 4 | 0 | - |
| (8) | 5 | 0 | - |

The results indicate that ChatGPT excelled in simple problems, but as the difficulty increased, its performance experienced a steep decline, with some problems yielding no score at all. Out of a total possible score of 520, ChatGPT ranked 189th among over 500 students, placing it in the top 40%. This shows that, based on accuracy alone, ChatGPT performs at the level of an intermediate beginner in Java programming, making it particularly useful for students with below-average or average performance.

Code quality also includes adherence to coding standards. The standardization assessment was based on the perfect-score code generated by ChatGPT, with professional programmers serving as evaluators. They assessed the codes in three criteria: naming conventions, indentation standards, and symbol usage. The scores are summarized in Table 3.

**Table 3.** Results of the Code Standardization Analysis.

| Number | Naming Conventions | Indentation Standards | Symbol Usage |
|---|---|---|---|
| (1) | 100 | 100 | 100 |
| (2) | 100 | 100 | 100 |
| (3) | 100 | 100 | 100 |
| (4) | 90 | 100 | 100 |
| (5) | 90 | 100 | 80 |
| (6) | 90 | 100 | 80 |
| (7) | 90 | 100 | 80 |
| (8) | 70 | 100 | 80 |

ChatGPT's adherence to coding standards remained consistent across problems of varying difficulty, consistently maintaining a high level of quality. The low score in naming conventions was attributed to the frequent use of ambiguous variable names.

The low score in symbol usage was due to the inconsistent use of single and double quotes for characters and strings. Overall, its coding standards were recognized by professionals, indicating that ChatGPT is well-suited for improving code quality.

ChatGPT's programming limitations mainly arise from variations in code accuracy across problems of different difficulty levels.

(1) Simple problems (basic concept problems): These problems are straightforward, and ChatGPT can quickly generate high-scoring code after a brief explanation. Compared to textbooks and online resources, ChatGPT provides faster and more precise solutions.

(2) Intermediate problems (simulation problems): These typically focus on a single knowledge point but require more time to process, and multiple prompts are often needed to guide ChatGPT. In this case, the accuracy of the code varies significantly depending on the user's skill level.

(3) Difficult problems (complex applications): These problems involve multiple knowledge points and real-world application scenarios, which are difficult for most people to understand, and even more challenging for ChatGPT. Due to unclear problem interpretation, it often generates incorrect results and is almost unable to solve these problems.

(4) Innovative problems (complex mathematical operations): These problems feature a large number of images and formulas. Since ChatGPT is currently effective only in text-based contexts (some versions can understand images, but this capability is beyond the scope of this paper due to general applicability), it cannot communicate effectively and is unable to provide solutions.

## 3 Discussion of Experimental Results

From the results of the experiments, it is evident that ChatGPT's programming capabilities derive from large language models trained on large datasets. While these models can simulate human conversation and exhibit logical consistency, they lack true understanding of programming. As a result, ChatGPT sometimes produces responses that appear logical but are, in fact, incorrect. In addition to its communication limitations, ChatGPT also faces challenges in programming. It can generate incorrect code due to ambiguous prompts, and for problems that are difficult to describe in text, effective communication can be nearly impossible. Thus, the idea that "ChatGPT could replace traditional programming education" is untenable.

### 3.1 The Teacher is Irreplaceable

In practical Java programming instruction, teachers guide students using a systematic method, often utilizing knowledge graphs to build understanding from basic concepts to a comprehensive knowledge framework. However, ChatGPT currently cannot replicate this method using knowledge graphs, partly because expressing knowledge graphs through text is challenging and partly because refining such knowledge sys-

tems requires specialized expertise. As a result, ChatGPT cannot fully replace teachers in helping students grasp the entirety of programming knowledge.

Currently, ChatGPT functions similarly to a well-informed yet somewhat limited upper-level student. It holds vast knowledge but sometimes lacks deep understanding, confidently providing answers that might occasionally be incorrect. In contrast, traditional education depends on teachers who not only impart knowledge but also carefully ensure its accuracy and improve it based on feedback, a level of rigor and precision that ChatGPT cannot match.

Additionally, the convenience of obtaining answers from ChatGPT has led to a widespread reliance among students for completing assignments, a trend so pervasive that institutions like New York University and the University of Tübingen have prohibited its use. This over-reliance poses a significant risk, potentially diminishing students' curiosity and stalling educational progress.

In contrast, teachers bear the crucial responsibility of education, emphasizing hands-on learning and personal interaction to instill moral values and underscore the importance of authentic learning. They keep students focused and committed to their educational journey, preventing the pitfalls of over-reliance on tools like ChatGPT. Therefore, the role of teachers as mentors and guides remains irreplaceable in the educational landscape.

### 3.2 New Model for Programming Education

While ChatGPT has its limitations, it also brings numerous advantages for enhancing Java programming education. For both students and teachers, it serves as a highly valuable tool.

ChatGPT provides clear and concise explanations of various concepts, which aids beginners in memorization. Its high-quality answers, combining concepts, examples, and key points, help students quickly grasp unfamiliar topics and review previously learned material. Additionally, its explanations of coding logic foster the development of programming skills, while the high standardization of the code promotes good coding habits through careful reading and analysis.

Leveraging its vast database of articles and programming exercises, ChatGPT can rapidly generate personalized study plans and curated lists of practice problems based on the needs of the student, supporting an efficient and structured learning path from foundational to advanced levels. Thus, ChatGPT represents a significant improvement over traditional online search methods, acting as a powerful tool to enhance the learning process and substantially boost learning efficiency.

For teachers, ChatGPT can quickly source relevant articles, videos, and other course materials. It also helps clarify unfamiliar concepts or jargon from other disciplines, speeding up lesson preparation and refining lecture content to improve teaching quality. Moreover, ChatGPT's ability to promptly answer routine questions drastically reduces the time needed for Q&A, allowing teachers to focus on delivering core content and optimizing their lesson plans.

In terms of post-class assignments, ChatGPT offers access to problem sets from major programming platforms, enabling teachers to generate high-quality exercises

quickly by using targeted keywords. Initial estimates suggest that with ChatGPT's assistance, the average time to create a practice problem can be reduced from 30 minutes to just 5 minutes. As long as educators embrace this evolving technology, their teaching efficiency will continue to improve with this indispensable tool.

## 4 Conclusions

The analysis of the results shows that the claim that it will revolutionize the programming industry is overstated and, at least for now, unfounded. However, ChatGPT does offer impressive programming capabilities, fostering students' interest in learning and enhancing their sense of social responsibility. It improves students' ability to apply knowledge in practical settings, cultivates professional talent with strong innovation and independent learning skills, and provides valuable insights for the reform of other specialized courses.

## References

1. Ouh, E. L., Gan, B. K. S., Jin Shim, K., & Wlodkowski, S. (2023, June). ChatGPT, Can You Generate Solutions for my Coding Exercises? An Evaluation on its Effectiveness in an undergraduate Java Programming Course. In Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 1 (pp. 54-60). https://doi.org/10.1145/3587102.3588794.
2. Hopkins, B. (2024). Using ChatGPT As Your Java Pair-Programmer. In ChatGPT for Java: A Hands-on Developer's Guide to ChatGPT and Open AI APIs (pp. 25-56). Berkeley, CA: Apress. https://doi.org/10.1007/979-8-8688-0116-7_2.
3. Kashefi, A., & Mukerji, T. (2023). ChatGPT for programming numerical methods. Journal of Machine Learning for Modeling and Computing, 4(2). https://doi.org/10.1615/ JMachLearnModelComput.2023048492.
4. Biswas, S. (2023). Role of ChatGPT in Computer Programming. Mesopotamian Journal of Computer Science, 2023, 9-15. https://doi.org/10.58496/MJCSC/2023/002.
5. Kuo, J. Y., Wen, Z. J., Hsieh, T. F., & Huang, H. X. (2023). A Study on the Security of Online Judge System Applied Sandbox Technology. Electronics, 12(14), 3018. https://doi.org/10.3390/electronics12143018.