Applying Artificial Intelligence to Build Game Programs in Java

ICTU Student 1
Thai Nguyen University of Information and Communication Technology
Thai Nguyen University, Thai Nguyen, Vietnam

***Abstract*—This thesis explores the application of algorithms in game development, focusing on the development of a Java-based game for solving the Ta-cach puzzle. The author discusses the use of A* algorithm for solving the game, along with the implementation of MiniMax and alpha-beta pruning techniques. The thesis provides a detailed explanation of Java programming and its application in game development, highlighting the stability and ease of understanding of the developed program. However, the author notes that the game's performance could be improved, particularly in terms of the slow movement of puzzle pieces and the need for a more complex environment. The study also covers the theoretical foundations of Java and artificial intelligence, emphasizing the importance of these technologies in modern game development. The author suggests that future work should focus on enhancing the game's performance and expanding its functionality to support more complex scenarios.**

***Index Terms*—Java, A*, MiniMax, Alpha-Beta Pruning, Game Development, Ta-cach Puzzle**

I. Introduction

The thesis introduces the field of knowledge and theory, particularly focusing on Java programming and artificial intelligence. It explains the development of personal computers and the increasing role of software games in entertainment. The author chose Java for game development due to its support for image and sound processing, making it suitable for creating a perfect game. The thesis also discusses the reasons for choosing the Ta-cach puzzle game and the Auto Puzzle game for development. The author highlights the historical context of these games and their significance in the field of game development. The introduction also outlines the challenges faced in developing a game, such as the need for efficient algorithms and user-friendly interfaces. [1]

II. Methods

The methods section covers the implementation of the A* algorithm for solving the Ta-cach puzzle game. The author describes the initialization of the game state and the process of expanding the state space. The MiniMax and alpha-beta pruning techniques are also applied to enhance the game's decision-making process. The author explains the use of Java Development Kit (JDK) and the various components of the Abstract Window Toolkit (AWT) to create the game's graphical user interface. The implementation details include the use of event handling and window listeners to manage user interactions. The author also discusses the challenges of integrating these algorithms into the game, such as optimizing the movement of puzzle pieces and ensuring the stability of the game's performance. [1]

Supervisor: ICTU Teacher 1, Thai Nguyen University of Information and Communication Technology.

III. Results

The developed game was implemented using Java, providing a stable and user-friendly interface. The A* algorithm was effective in finding solutions, but the slow movement of puzzle pieces made the game's completion process relatively time-consuming. The MiniMax and alpha-beta pruning techniques were successfully applied, improving the game's decision-making process. However, the author notes that the game's performance could be further improved, particularly in terms of the speed of puzzle piece movement and the complexity of the game environment. The author also mentions that the game's current implementation does not fully support network environments, which limits its potential for multiplayer functionality. [1]

IV. Discussion

The author discusses the challenges faced during the development, such as the slow movement of puzzle pieces and the need for a more complex environment. The thesis also highlights the potential for future development, including the expansion of the game to a network environment and the development of a program for players to assemble puzzles. The author suggests that future work should focus on enhancing the game's performance and expanding its functionality to support more complex scenarios. The discussion also addresses the limitations of the current implementation, such as the lack of support for network environments and the need for more sophisticated algorithms to improve the game's efficiency. [1]

V. Conclusion

The thesis concludes that the application of search algorithms in game development is effective, but there is room for improvement. The author suggests future work should focus on enhancing the game's performance and expanding its functionality to support more complex scenarios. The conclusion also reiterates the importance of using Java for game development due to its support for image and sound processing. The author emphasizes the need for further research and development to address the limitations of the current implementation and to explore new possibilities in game development. [1]

References

[1] ICTU Student 1, "Applying Artificial Intelligence to Build Game Programs in Java," undergraduate thesis (original in Vietnamese), Thai Nguyen University of Information and Communication Technology, n.d.

[2] Đinh Mạnh Tường, Giáo trình trí tuệ nhân tạo, Nhà xuất bản khoa học kỹ thuật, Năm 2002

[3] Đoàn Văn Ban, Lập trình hướng đối tượng với Java, Nhà xuất bản khoa học kỹ thuật, Năm 2005

[4] Phương Lan, Java, Nhà xuất bản thống kê, Năm 2004

[5] Trang web: http://www.blogthuthuat.com.vn

[6] Trang web: http://www.vietgo.wetpaint.com

[7] Trang web: http://www.free4vn.org.vn

[8] Trang web: http://www.vn.myblog.com

[9] Trang web: http://www.javapassipm.com

[10] Trang web: http://www.ebook.edu.vn