ISET 10 (2024): 367-370 e-ISSN 2964-4291
International Conference on Science, Education and Technology http://proceeding.unnes.ac.id/ISET

The Effectiveness of Combining Block-Based Coding with Unplugged Activities for Learning Programming Concepts in Primary Grades

Bagus Dwi Prasetyo*, Suwito Eko Pramono, Sungkowo Edy Mulyono

Postgraduate School, Universitas Negeri Semarang, Indonesia

*Corresponding Author: bagus.prasetyo@sisschools.org

Abstract. This research study investigates the effectiveness of an integrated approach to teaching programming concepts to primary grade students, combining block-based coding exercises (BBCEs) with unplugged activities. The backdrop is the 21st-century digital era, where computational thinking and programming have become critical skills for young learners. The research was conducted with two fourth-grade classes: one received traditional BBCEs instruction (control group), and the other combined this with unplugged activities (intervention group). A post-test was used to compare the groups' performance. The results showed that the intervention group, who experienced the blended learning approach, performed better than the control group, indicating the potential benefits of the combined approach. It's important to note that the study utilized a quasi-experimental design with intact classes and a post-test-only design, which could influence the findings. Despite these limitations, the findings contribute to the growing research on effective pedagogical approaches for teaching programming concepts, suggesting that integrating unplugged activities with BBCEs can enhance student learning outcomes.

Keywords: unplugged activities, block-based programming, computational thinking, programming

INTRODUCTION

The advent of the 21st century has marked a significant shift towards a more digital world. This transformation has magnified the importance of computational thinking (CT) in the realm of education. CT, a problem-solving method and thought process integral to computer science, has become a critical skill for young students navigating through the digital age (Wing, 2006). However, introducing these intricate concepts to young learners, especially those in primary grades, presents a considerable educational challenge (Angeli & Giannakos, 2020; Li et al., 2020b).

Traditionally, the education sector has relied heavily on the use of digital tools for teaching computer programming. Among these, block-based coding environments (BBCEs) have emerged as particularly popular choices. BBCEs provide a user-friendly interface with visual blocks that represent programming constructs such as sequences, loops, and conditional statements (Resnick et al., 2009). This interactive and engaging approach has proven particularly beneficial for young learners in primary grades. BBCEs provide students with hands-on experience, which is instrumental in learning programming concepts and developing problem-solving skills (Dilmen et al., 2023; Resnick et al., 2009).

Despite the promising findings, some researchers have expressed concerns about a purely digital approach. While BBCEs make programming more accessible and interesting for young learners, they may not provide a firm foundation for understanding the more abstract and complex programming concepts (Ballard & Haroldson, 2022; Huang & Looi, 2020; Saidin et al., 2021). This viewpoint suggests that while BBCEs are an excellent tool for introducing programming, they may not provide the depth of understanding required for more advanced computational thinking (Kwon et al., 2021).

In response to these concerns, educators have developed “unplugged” activities. These activities involve physical representations of programming concepts, offering a more tangible and concrete approach to learning these abstract ideas. Numerous research studies have highlighted the effectiveness of unplugged activities in promoting a deeper understanding of core programming concepts. For instance, a study by ERÜMIT & SAHIN (2020) found that unplugged activities significantly enhanced students’ understanding of sequencing and debugging, as compared to a traditional lecture-based approach. Another study reported that unplugged activities not only improved students' problem-solving skills but also their ability to apply and transfer the learned concepts to digital environments (Chen et al., 2023; Jehan & Akram, 2023). These findings strongly suggest the potential of unplugged activities as a valuable tool in the teaching of programming.

However, relying solely on unplugged activities for a comprehensive programming education might have its limitations. BBCEs offer valuable features that are challenging to replicate in an unplugged setting. They provide immediate feedback, which is crucial for learning through trial and error, and they allow for the creation of complex programs, which can serve as a source of motivation and engagement for students (Damle et al., 2023; Dilmen et al., 2023).

Given these considerations, a promising approach might be to combine unplugged activities with BBCEs. This integrated approach has the potential to leverage the strengths of both methods, promoting a more profound and well-rounded understanding of programming concepts in young learners (Chen et al., 2023; Jehan & Akram, 2023).

The objectives of this research include evaluating the effectiveness of combining block-based coding with unplugged activities for teaching programming, identifying best practices for implementing this blended approach in primary education, and assessing whether this integrated method leads to improved student outcomes compared to using either method in isolation. This study seeks to provide valuable insights into enhancing computational thinking education for your learners.

METHODS

The methodology for this research was carefully designed with the primary objective of examining the effects of combining block-based coding with unplugged activities on the learning of programming concepts within primary grades. The research took place at SIS Semarang, Indonesia during Term 3 of the 2023-2024 academic year.

The population for this study consisted of primary students at SIS Semarang. From this population, two intact classes were selected: Grade 4A and Grade 4B. These classes were chosen based on their academic calendar and availability during the data collection period. Grade 4A served as the control group, receiving traditional instruction on programming concepts using a block-based coding environment (BBCE) on code.org. These students followed the standard curriculum of 11 lessons offered by the platform, without any additional enhancements. Grade 4, the intervention group, received the same BBCE instruction on code.org, but with the addition of unplugged activities incorporated throughout the curriculum. These unplugged activities were meticulously designed to provide a solid, concreate foundation for the programming concepts being taught in the BBCE lessons.

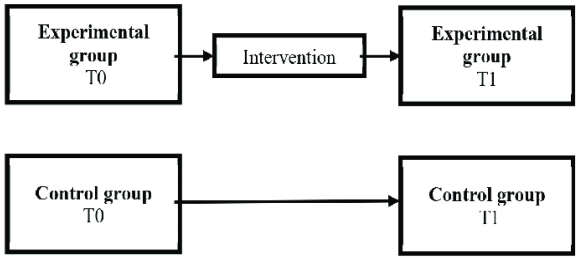

Figure 1. Quasi experimental research design

The data collection involved several key steps. Although a pre-test was not administered, all participants were assumed to have the same basic knowledge about coding based on their results from previous lessons. During the instruction phase, both groups received their respective instructional methods over the course of the term. The control group followed the BBCE curriculum on code.org without any modifications, while the intervention group incorporated unplugged activities alongside the BBCE lessons. At the end of the intervention, a post-test was administered to both groups. The post-test consisted of block-based coding tasks on code.org, selected to be similar in format to the lessons in the curriculum to ensure consistency with the learning environment.

The final step was the analysis of post-test data. The dependent variable in this study was the student performance on the post-test. The data was meticulously analysed using appropriate statistical methods to compare the performance of the control and intervention groups. Depending on the normality of the data, this might involve independent samples t-tests or non-parametric alternatives (Field, 2009). This stage was pivotal in determining the effectiveness of the unplugged activities in enhancing the learning of programming concepts.

RESULTS AND DISCUSSION

The primary objective of this study was the explore the effectiveness of combining block-based coding with unplugged activities in teaching programming concepts. The results, derived from meticulously conducted tests, provided a clear and concise answer to the research questions.

First, the normality of data distribution in both the control and intervention groups was assessed

to validate the subsequent statistical tests. The Kolmogorov-Smirnov test was employed for this purpose. The results indicated normal distributions for both groups, with significance values greater than 0.05: the control group at $p = 0.097$ and the intervention group at $p = 0.200$. This aligns with previous research, which has established the reliability of using statistical test to validate data distributions in educational studies (Kwon et al., 2021; Li et al., 2020a; Saidin et al., 2021). Ensuring normality before proceeding with further statistical analysis is crucial and consistent with established best practices in educational research (Chen et al., 2023; Damle et al., 2023; Sun et al., 2021).

Next, the post-test performance of both groups was evaluated. The control group had an average score of 74.81 on the block-based coding post-test. In contrast, the intervention group, exposed to a combination of block-based coding and unplugged activities, achieved a higher average score of 90.41. These findings are consistent with previous studies that have shown the effectiveness of incorporating unplugged activities to enhance understanding and retention of programming concepts (Ballard & Haroldson, 2022; Chen et al., 2023; Gleasman & Fegely, 2019). The substantial increase in scoresfor the intervention group highlights the potential benefits of a blended learning approach, which combines digital and physical learning activities to reinforce programming concepts (Jehan & Akram, 2023; Namli & Aybek, 2022; Resnick et al., 2009).

To further validate these findings, hypothesis testing was performed. Levene's Test for Equality of Variance indicated homogeneity of variance between the control and intervention groups ($p = 0.013$), allowing an independent-sample t-test to be conducted assuming equal variances. The results revealed a statistically significant difference between the groups ($p = 0.023$. This statistically significant difference is supported by earlier research, which has demonstrated that combining different instructional methods can lead to improved educational outcomes (Jehan & Akram, 2023; Namli & Aybek, 2022; Resnick et al., 2009). The integration of unplugged activities with block-based coding not only enhances engagement but also provides a more holistic understanding of programming concepts, as evidenced by the improved performance of students in the intervention group (Li et al., 2020b; Saidin et al., 2021; Toma, 2023).

This difference suggests that the intervention group, which received combined instruction of block-based coding exercises (BBCEs) with unplugged activities, performed better on the post-test than the control group, which received only traditional BBCE instruction. This finding indicates the potential benefits of the combined approach and corroborates prior research that emphasizes the importance of incorporating multiple strategies to enhance learning (Damle et al., 2023; Li et al., 2020b; Toma, 2023). By integrating unplugged activities, students are provided with tangible experiences that bridge the gap between abstract programming concepts and real-world applications, thereby deepening their comprehension and retention (Jehan & Akram, 2023; Namli & Aybek, 2022; Resnick et al., 2009).

However, any scientific study must be interpreted within the context of its limitations. The use of a quasi-experimental design with intact classes introduces the possibility of pre-existing group differences that could have influenced the findings. Additionally, a post-test only design was employed. Future research using a more robust methodology, such as random assignment of participants and including a pre-test, would allow for a more in-depth evaluation of the intervention's effectiveness. These limitations are consistent with those noted in other studies on educational interventions, which have called for more rigorous experimental designs to validate findings (Angeli & Giannakos, 2020; Saidin et al., 2021; Sun et al., 2021). Addressing these limitations in future studies could further substantiate the efficacy of combining block-based coding with unplugged activities (Damle et al., 2023; Li et al., 2020b; Toma, 2023).

CONCLUSION

In conclusion, the research objectives aimed to evaluate the effectiveness of combining block-based coding environments (BBCEs) with unplugged activities for teaching programming concepts to primary grade students. The study's findings confirmed that this integrated approach significantly enhances student learning outcomes compared to using BBCEs or unplugged activities in isolation. The post-test results demonstrated that students who experienced the combined instructional method performed better than those who only engaged with BBCEs. This suggests that the tangible, hands-on experiences provided by unplugged activities complement the immediate feedback and complexity offered by BBCEs, resulting in a more comprehensive understanding

of programming concepts. These findings underscore the importance of adopting a blended learning strategy in primary education to foster computational thinking skills effectively. By leveraging the strengths of both digital and physical learning environments, educators can create a more engaging and robust educational framework that better prepares young learners for the digital age. Future research should continue to explore this integrated approach, addressing the study's limitations and further refining the methodology. This will ensure that the educational practices and policies developed are both effective and inclusive, ultimately contributing to the advancement of computational thinking education for primary grade students.

REFERENCES

Angeli, C., & Giannakos, M. (2020). Computational thinking education: Issues and challenges. In *Computers in Human Behavior* (Vol. 105).

Ballard, E. D., & Haroldson, R. (2022). Analysis of Computational Thinking in Children's Literature for K-6 Students: Literature as a Non-Programming Unplugged Resource. *Journal of Educational Computing Research*, *59*(8).

Chen, P., Yang, D., Metwally, A. H. S., Lavonen, J., & Wang, X. (2023). Fostering computational thinking through unplugged activities: A systematic literature review and meta-analysis. In *International Journal of STEM Education* (Vol. 10, Issue 1).

Damle, P., Bull, G., Watts, J., & Nguyen, N. R. (2023). Automated Structural Evaluation of Block-based Coding Assignments. *SIGCSE 2023 - Proceedings of the 54th ACM Technical Symposium on Computer Science Education*, *2*.

Dilmen, K., Kert, S. B., & Uğraş, T. (2023). Children's coding experiences in a block-based coding environment: a usability study on code.org. *Education and Information Technologies*, *28*(9).

ERÜMIT, A. K., & SAHIN, G. (2020). Plugged or Unplugged Teaching: A Case Study of Students' Preferences in the Teaching of Programming. *International Journal of Computer Science Education in Schools*, *4*(1).

Gleasman, C., & Fegely, A. (2019). Building a Foundation for K-12 Computer Science Teacher Education Using Block-Based Coding. *Learntechlib.Org*.

Huang, W., & Looi, C. K. (2020). A critical review of literature on "unplugged" pedagogies in K-12 computer science and computational thinking education. In *Computer Science Education.*

Jehan, S., & Akram, P. (2023). Introducing Computer Science Unplugged in Pakistan: A Machine Learning Approach. *Education Sciences*, *13*(9).

Kwon, K., Ottenbreit-Leftwich, A. T., Brush, T. A., Jeon, M., & Yan, G. (2021). Integration of problem-based learning in elementary computer science education: effects on computational thinking and attitudes. *Educational Technology Research and Development*, *69*(5).

Li, Y., Schoenfeld, A. H., diSessa, A. A., Graesser, A. C., Benson, L. C., English, L. D., & Duschl, R. A. (2020a). Computational Thinking Is More about Thinking than Computing. In *Journal for STEM Education Research* (Vol. 3, Issue 1).

Li, Y., Schoenfeld, A. H., diSessa, A. A., Graesser, A. C., Benson, L. C., English, L. D., & Duschl, R. A. (2020b). On Computational Thinking and STEM Education. In *Journal for STEM Education Research* (Vol. 3, Issue 2).

Namli, N. A., & Aybek, B. (2022). An Investigation of The Effect of Block-Based Programming and Unplugged Coding Activities on Fifth Graders' Computational Thinking Skills, Self-Efficacy and Academic Performance. *Contemporary Educational Technology*, *14*(1).

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., & Kafai, Y. (2009). Scratch: Programming for all. *Communications of the ACM*, *52*(11).

Saidin, N. D., Khalid, F., Martin, R., Kuppusamy, Y., & Munusamy, N. A. P. (2021). Benefits and challenges of applying computational thinking in education. *International Journal of Information and Education Technology*, *11*(5).

Sun, D., Ouyang, F., Li, Y., & Zhu, C. (2021). Comparing learners' knowledge, behaviors, and attitudes between two instructional modes of computer programming in secondary education. *International Journal of STEM Education*, *8*(1).

Toma, R. B. (2023). Measuring Acceptance of Block-Based Coding Environments. *Technology, Knowledge and Learning*, *28*(1).