IELTech Informatics Education and Learning Technologies

Vol 1, No. 1, June 2025, pp. 1–10
doi: 10.23917/ieltech.v1i1.16290

Enhancing Logic and Algorithm Learning Outcomes through Game-Based Learning with Scratch: A Quasi-Experimental Study

Naviatul Azizah[1*], Sukirman[2], Vivi Gesilanda[3]

[1,2,3] *Informatics Engineering Education, Universitas Muhammadiyah Surakarta, Indonesia*

Email: a710190044@student.ums.ac.id

Article History:

Received: 3 January 2025
Revised: 8 March 2025
Accepted: 15 April 2025
Published: 1 June 2025

Keywords:

Game-Based Learning (GBL), Scratch, Visual Programming, Logic and Algorithms, Quasi-experiment; Usability

Abstract

Logic and Algorithm (L&A) serves as a fundamental subject for fostering computational thinking in middle schools. However, novice students often struggle with abstract programming concepts, a challenge frequently exacerbated by conventional Teacher-Centered Learning (TCL) approaches. This study investigates the effectiveness and usability of a Game-Based Learning (GBL) strategy assisted by Scratch visual programming to enhance students' L&A competency. Adopting a quasi-experimental design, this research involved 108 seventh-grade students divided equally into an experimental group (n=54), which utilized the GBL-Scratch strategy, and a control group (n=54), which employed the TCL method. Efficacy was measured using pre-test and post-test assessments, while usability was evaluated using a modified USE (Usefulness, Satisfaction, and Ease of use) questionnaire. The results demonstrated a significant improvement in the experimental group, with the average score increasing from 75.70 to 84.11, compared to the control group's lower progression from 59.26 to 65.74. An independent sample t-test confirmed the statistical significance of this difference ($p < 0.05$). Furthermore, usability analysis revealed high student acceptance, indicating that Usefulness, Ease of Use, and Ease of Learning significantly influenced overall Learning Satisfaction. These findings suggest that integrating Scratch-based GBL significantly outperforms traditional methods in facilitating logic and algorithm comprehension, offering a viable pedagogical alternative for introductory programming education.

1 Introduction

Logic and Algorithm (L&A) is a fundamental subject in computer science education that serves as the cornerstone for developing computational thinking and problem-solving skills. Logic provides the principles for constructing valid arguments and reasoning to facilitate the right way of thinking (Moeis & Yunarti, 2022). Meanwhile, algorithms offer structured and systematic steps to solve problems through programming languages (Hartati et al., 2020). Mastery of these concepts is essential, as they enable students to think rationally about a problem to produce truth and approach programming tasks with a structured mindset (Bulan et al., 2018). Therefore, cultivating a strong understanding of logic and algorithms is a critical requirement for students to succeed in modern computing curriculums (Doz et al., 2022).

Despite its importance, learning Logic and Algorithms poses significant challenges for novice students, particularly in the transition from abstract concepts to practical implementation. A primary contributing factor to these difficulties is the prevalence of Teacher-Centered Learning (TCL) strategies, where students passively listen to explanations rather than actively engaging with the material (Kreinovich, 2018). This passive approach often leads to low engagement, boredom, and suboptimal learning outcomes, as students struggle to grasp complex procedural steps without hands-on practice. Consequently, learning materials should be presented interestingly so that students not only understand the material but also achieve the expected competencies properly (Timur et al., 2021). Innovative instructional strategies are urgently needed to transform abstract L&A concepts into concrete, interactive learning experiences (Ahyar et al., 2020).

Game-Based Learning (GBL) has emerged as a powerful pedagogical strategy to address engagement issues in education. By integrating game elements into the learning process, GBL stimulates the brain's ability to overcome conflicts or problems in an environment (Cózar-Gutiérrez & Sáez-López, 2016). This approach allows students to learn through interaction and immediate feedback, which significantly encourages active learning and enhances motivation compared to traditional methods (Thangjai, 2022). In the context of programming education, GBL can be implemented through three models: using game elements, employing gameplay, or learning by developing a game (Sukirman et al., 2022). This study focuses on the third model, where students learn material content by developing their own games.

To facilitate this game development approach for middle school students, Visual Programming environments like Scratch offer a distinct advantage over text-based languages. Scratch utilizes a block-based interface that allows students to learn Logic and Algorithms through provided visual features, effectively eliminating the frustration of syntax errors commonly faced by beginners (Iyamuremye & Nsabayezu, 2022). This visual representation helps students develop the basics of programming skills faster and deal with programming material by learning from knowledge and experience (Kanaki & Kalogiannakis, 2022). Moreover, using Scratch saves programming logic from being boring by visualizing the concepts, making the learning process more enjoyable and accessible (Saltan & Kara, 2022).

While previous studies have highlighted the benefits of GBL and Scratch independently, there is a need to rigorously evaluate their combined effectiveness specifically within the L&A curriculum. Adipat et al. (2021) suggest that engaging learning modules can involve students more deeply in composing learning concepts. Although literature reveals an improvement in student performance when using different methods for teaching logic and algorithms (Tekerek & Tuğba, 2018), many implementations still lack a structured strategy that guides students from basic concepts to complex problem-solving through game creation. Successful integration requires not just a tool, but a comprehensive learning strategy that aligns theoretical content with practical game development (Sukirman et al., 2022).

This study aims to bridge this gap by implementing a GBL strategy using Scratch to facilitate the learning of Logic and Algorithms. Unlike traditional TCL approaches, this research employs a quasi-experimental design to empirically measure the effectiveness of the GBL-Scratch strategy on student learning outcomes. Additionally, it evaluates the usability of the developed learning strategy—encompassing usefulness, ease of use, ease of learning, and satisfaction—using an instrument developed from the USE questionnaire (Hariyanto et al., 2020). This comprehensive evaluation ensures that the pedagogical intervention is not only effective in improving grades but also sustainable and accepted by students for classroom adoption.

2 Method

This study employed a quasi-experimental design, specifically a non-equivalent control group design, to evaluate the effectiveness of the Game-Based Learning (GBL) strategy. The research procedure followed a systematic five-stage approach: (1) Analysis, (2) Design, (3) Development, (4) Experiment, and (5) Evaluation. In the analysis stage, a comprehensive literature review and document analysis of the syllabus and curriculum were conducted. Subsequently, the learning module and experimental design were constructed in the design stage , followed by the development of the learning module and valid research instruments. The core of this study, the experimental stage, compared two distinct instructional strategies: the experimental group utilizing the GBL strategy with Scratch, and the control group employing a TCL approach. Finally, an evaluation was performed using statistical analysis of test scores and questionnaires.

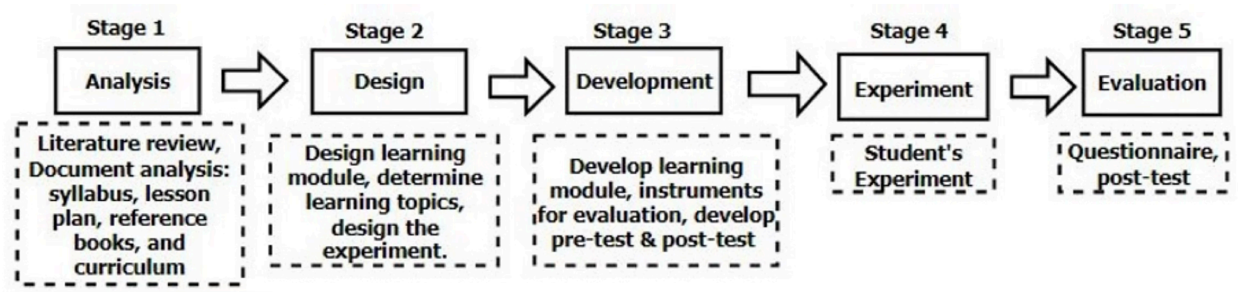

Figure 1: The five-stage research phase.

2.1 Participants

The participants in this study were 108 seventh-grade students from a middle school (SMP), divided equally into two groups: 54 students in the experimental group and 54 students in the control group. The demographic distribution of the participants based on gender is presented in Table 1.

Table 1: Demographic characteristics of participants

Group	Gender	Frequency (f)	Percentage (%)
Experiment (n=54)	Male	20	30%
	Female	34	70%
Control (n=54)	Male	22	30%
	Female	32	70%
Total		108	100%

2.2 Experimental Procedure

The learning experiment was conducted over five meetings. The experimental group utilized a developed learning module that integrated GBL with Scratch visual programming. The module content was structured into five distinct sections to guide students from basic concepts to game development, as detailed in Table 2.

Table 2: Module content components

No	*Material Focus*	*Learning Objectives*
1	*Basic Concept of Logic and Algorithms*	*Students know basic programming Logic and Algorithms*

No	Material Focus	Learning Objectives
2	*Introduction to Scratch*	*Students can explain visual Scratch programming*
3	*Logic and Algorithms on Scratch*	*Students can know the concept of Logic and Algorithms*
4	*Making a Simple Game Using Scratch*	*Students can make simple games using Scratch with Logic and Algorithm material*
5	*Logic and Algorithms with Games*	*Students can make games from Logic and Algorithm material using Scratch*

The experimental cycle began with a pre-test to measure initial knowledge. During the learning phase, the experimental group engaged in active learning by creating game projects (e.g., "The shark hunts pufferfishes") guided by the module, where they applied logic to control game mechanics. Conversely, the control group followed the TCL strategy, focusing on teacher explanations and passive exercises. The cycle concluded with a post-test and a usability questionnaire for the experimental group.

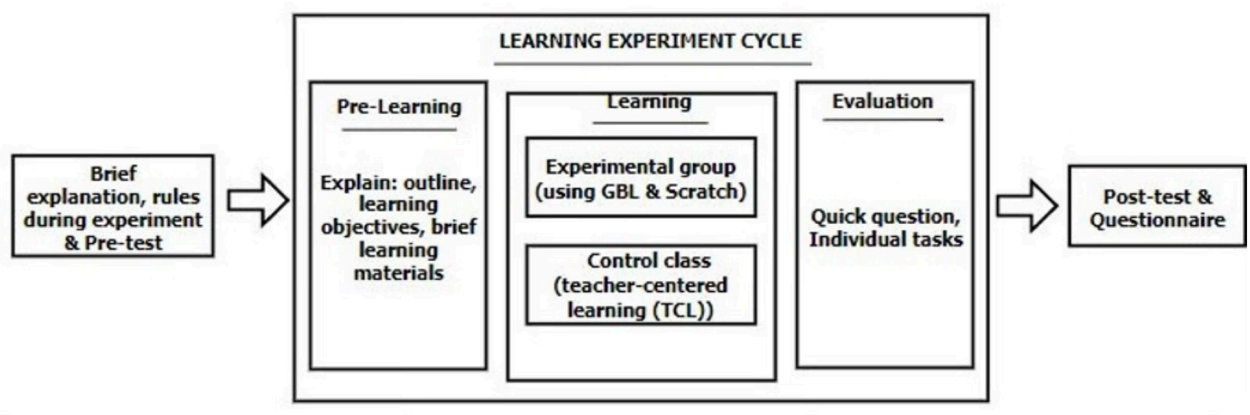

Figure 2: Experimental design and learning cycle

2.3 Instruments

To measure the variables, two types of instruments were used:

1. **Test Instruments:** A pre-test and post-test were administered to both groups to measure the improvement in learning outcomes regarding Logic and Algorithms.
2. **Questionnaire:** A usability questionnaire was distributed to the experimental group to evaluate the GBL strategy. The instrument was adapted from the USE questionnaire model (Hariyanto et al., 2020), measuring four parameters: Usefulness (GBL strategy), Ease of Use (Learning module), Ease of Learning (Scratch software), and User Satisfaction (Overall method). The conceptual framework is shown in Figure 3.

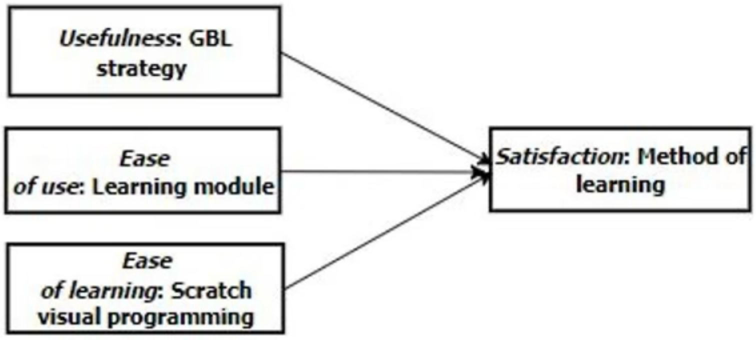

Figure 3: Conceptual framework of the USE questionnaire

The questionnaire consisted of 21 items validated by three experts with over five years of experience. The grid of the instrument is presented in Table 3

Table 3: Instruments for evaluating learning strategies (USE Model)

No	***Questions***
Usefulness: Game-Based Learning (GBL) Strategy	
1	*This applied learning strategy helps me be more effective in learning Logic & Algorithm*
2	*This applied learning strategy helps me be productive in learning Logic & Algorithm*
3	*This applied learning strategy is very useful for learning Logic & Algorithms*
4	*This applied learning strategy makes things that I want to accomplish in learning Logic & Algorithm easier*
5	*Using this applied learning strategy saves me time learning Logic & Algorithms*
6	*This applied learning strategy meets the needs of learning Logic & Algorithms*
Ease of Use: Learning Module	
7	*This learning module is easy to use for learning Logic & Algorithms*
8	*This learning module is simple to use to learn Logic & Algorithms*
9	*This learning module is easy to understand for learning Logic & Algorithms*
10	*I can use the module according to the specified command to learn Logic & Algorithm*
11	*I don't notice any inconsistencies while using the learning module*
12	*Every meeting I always manage to learn using the learning module to learn Logic & Algorithms*
Ease of Learning: Visual Scratch Programming	
13	*I learned to use Scratch visual programming quickly*
14	*I easily remember how to use Visual Scratch programming to learn Logic & Algorithm*
15	*Visual Scratch programming is easy to use to learn Logic & Algorithms*
16	*I quickly became more skilled with Visual Scratch programming to learn Logic & Algorithms*
Satisfaction: Overall Learning Method Used	
17	*I'm satisfied with the learning method used for Logic & Algorithm study*
18	*I would recommend this method of study to my teacher*
19	*This learning method is very fun to use in learning Logic & Algorithms*
20	*This learning method goes with what I want in learning Logic & Algorithm*
21	*I feel I need a learning method like this to learn Logic & Algorithm*

2.4 Data Analysis

Quantitative data were analyzed using SPSS software. The validity of the questionnaire was tested using Pearson correlation, where items were considered valid if the correlation value exceeded the r-table (0.3494). Reliability was assessed using Cronbach's Alpha, with a threshold of > 0.7 indicating acceptable reliability. To evaluate the effectiveness of the intervention, the learning outcome data were subjected to normality (Shapiro-Wilk) and homogeneity (Levene's test) checks. Finally, an Independent Sample t-test was conducted to determine the significant difference between the experimental and control groups.

3 Result

3.1 Game Project Development

In the experimental group, students successfully developed game projects using Scratch based on the Logic and Algorithm (L&A) concepts taught in the module. One of the primary projects created was "The Shark Hunts Pufferfishes". In this project, students applied algorithmic thinking to control the game mechanics: the shark sprite was programmed to move and hunt, while the pufferfish sprites were programmed to evade. The construction of these mechanics required students to implement logical structures (selection and repetition) within the block-based coding environment. Figure 4 illustrates the interface and code blocks of the student projects.

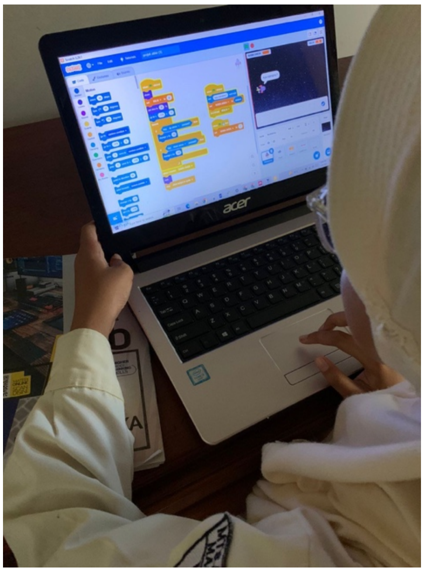

Figure 4: Student game project: Implementing logic and algorithm concepts.

3.2 Validity and Reliability of Instruments

Before analyzing the learning outcomes, the research instruments were tested for validity and reliability. The validity test using Pearson Correlation demonstrated that all 21 questionnaire items had a correlation value higher than the r-table (0.3494) at a significance level of 0.05, indicating that all items were valid. Furthermore, the reliability test using Cronbach's Alpha confirmed the consistency of the instrument, with all variables (Usefulness, Ease of Use, Ease of Learning, and Satisfaction) scoring above the 0.7 threshold. The reliability results are summarized in Table 4.

Table 4: Reliability test analysis results.

Variable	Cronbach's Alpha	N of Items	Status
Usefulness: GBL Strategy	0.765	7	Reliable
Ease of Use: Learning Module	0.778	7	Reliable
Ease of Learning: Visual Scratch	0.758	5	Reliable
User Satisfaction	0.795	6	Reliable

3.3 *Student Learning Outcomes*

The effectiveness of the GBL strategy was measured by comparing the pre-test and post-test scores of both groups. Descriptive statistics reveal that the experimental group achieved a higher improvement compared to the control group. The experimental group's average score increased from 75.70 (SD=11.36) to 84.11 (SD=10.29). In contrast, the control group, which utilized the TCL method, showed a smaller increase from 59.26 (SD=17.00) to 65.74 (SD=13.26). These results indicate an escalation of 8.41 points for the experimental group versus 6.48 points for the control group. The comparison is detailed in Table 5.

Table 5: Descriptive Statistics of Pre-test and Post-test Scores

Group	N	Minimum	Maximum	Mean	Std. Deviation
Experimental					
Pre-test	54	40	90	75.70	11.360
Post-test	54	50	100	84.11	10.296
Control					
Pre-test	54	20	90	59.26	17.009
Post-test	54	40	90	65.74	13.262

3.4 *Hypothesis Testing*

Prerequisite tests were conducted to ensure the data met the assumptions for parametric testing. The Shapiro-Wilk normality test indicated that data distributions for both groups were normal ($p > 0.05$). The Levene's test for equality of variances showed a significance value of 0.350 ($p > 0.05$), confirming that the data variance was homogeneous.

To test the hypothesis, an Independent Sample t-test was performed. The results, as presented in Table 6, show a significant difference between the two groups. The calculated t-value was 2.661 with a significance (2-tailed) of 0.004, which is less than the alpha level of 0.05. Therefore, the null hypothesis is rejected, confirming that the Game-Based Learning strategy using Scratch significantly improves students' Logic and Algorithm learning outcomes compared to the traditional TCL method.

Table 6: Independent samples t-test results.

Model	F	Sig.	t	df	Sig. (2-tailed)	Mean Diff.
Equal variances assumed	0.280	0.865	2.661	106	0.004	7.370

3.5 Usability Evaluation

The usability of the implemented strategy was evaluated based on the USE questionnaire responses from the experimental group. The analysis, categorized by aspect, indicates a very strong acceptance of the method. Students reported high confidence in expressing opinions (81%) and strong knowledge comprehension regarding programming materials (80%). Furthermore, the skills aspect, which encompasses the GBL strategy and L&A mastery, achieved the highest percentage at 82%, categorized as "Very Good".

Table 7: Usability assessment results based on questionnaire aspects.

Aspect	Indicator	Percentage	Criteria
Confident	Express opinions	81%	Very Good
Knowledge	Understanding programming material	80%	Good
	Understanding learning models	—	—
Skills	GBL Strategy	82%	Very Good
	Logic and Algorithms	—	—

4 Discussion

The primary objective of this study was to evaluate the effectiveness of a GBL strategy using Scratch visual programming in enhancing students' L&A learning outcomes. The statistical analysis revealed a significant difference between the experimental and control groups. The experimental group, which engaged in developing game projects like "The Shark Hunts Pufferfishes," demonstrated a superior improvement in post-test scores compared to the control group taught via the TCL method. This finding aligns with the hypothesis that GBL significantly impacts students' comprehension of abstract programming concepts. The substantial increase in the experimental group's mean score (from 75.70 to 84.11) confirms that active engagement in game creation fosters a deeper understanding of logic structures compared to passive listening in TCL environments.

The effectiveness of this strategy can be attributed to the unique affordances of Scratch as a visual programming tool. By transforming complex algorithmic syntax into drag-and-drop visual blocks, Scratch reduces the cognitive load associated with syntax errors, allowing students to focus on the logic itself. This observation supports the findings of Saltan and Kara (2022), who argued that visual programming saves the learning of logic from being boring by making it visual and interactive. In this study, students were able to immediately visualize the output of their logic—such as the movement of a shark or the evasion of a pufferfish—which provided instant feedback and reinforcement. This process transforms the abstract nature of algorithms into concrete, observable actions, thereby facilitating better retention and skill acquisition (Kanaki & Kalogiannakis, 2022).

In contrast, the lower performance of the control group highlights the limitations of the TCL approach in teaching L&A. As noted by Kreinovich (2018), TCL models often result in students being less actively involved, leading to boredom and a lack of focus. Without the practical application provided by game development, students in the control group struggled to conceptualize the dynamic nature of algorithms. The GBL strategy addresses this gap by creating an immersive learning environment. Adipat et al. (2021) emphasized that engaging learning modules can involve students more deeply in the learning process. By integrating L&A materials into a game project, students in the experimental group were not just learning to code; they were "playing" with logic, which significantly boosted their motivation and creative thinking (Timur et al., 2021).

Furthermore, the usability evaluation using the USE questionnaire corroborates the quantitative findings. The high percentage scores in Usefulness (81%) and User Satisfaction indicate that students perceived the GBL strategy as both effective and enjoyable. The regression analysis implies that Ease of

Use and Ease of Learning are critical predictors of student satisfaction. When students find the learning module and the Scratch interface easy to navigate, their satisfaction with the learning method increases, which in turn correlates with better learning outcomes. This supports the notion that a well-structured learning module is essential for the successful implementation of GBL (Sukirman et al., 2022). Therefore, the synergy between the GBL strategy, the visual nature of Scratch, and a structured module creates a robust pedagogical framework for introductory programming education.

5 Conclusion

The empirical results of this study demonstrate that the implementation of Game-Based Learning (GBL) utilizing Scratch visual programming significantly enhances students' Logic and Algorithm learning outcomes compared to traditional Teacher-Centered Learning (TCL). This is evidenced by the substantial increase in the experimental group's average score from 75.70 to 84.11, whereas the control group showed a modest improvement from 59.26 to 65.74. The independent sample t-test ($p < 0.05$) confirms that this difference is statistically significant. Furthermore, the usability evaluation reveals a high level of student acceptance, with Usefulness, Ease of Use, and Ease of Learning identified as key factors influencing overall satisfaction. These findings suggest that integrating Scratch-based GBL is a viable and effective pedagogical alternative for teaching abstract programming concepts in middle schools. Future research is recommended to expand the scope to other STEM subjects and involve a larger demographic to validate the scalability of this strategy.

References

Adipat, S., Laksana, K., Busayanon, K., Ausawasowan, A., & Adipat, B. (2021). Engaging Students in the Learning Process with Game-Based Learning: The Fundamental Concepts. *International Journal of Technology in Education*, *4*(3), 542–552. https://doi.org/10.46328/ijte.169

Ahyar, H., Maret, U. S., Andriani, H., Sukmana, D. J., Mada, U. G., Hardani, S.Pd., … & Istiqomah, R. R. (2020). *Buku Metode Penelitian Kualitatif & Kuantitatif*. Yogyakarta: CV. Pustaka Ilmu.

Bulan, R., Isroqmi, A., & Nopriyanti, T. D. (2018). Pengaruh penerapan algoritma terhadap pembelajaran pemrograman komputer. *Jurnal Pendidikan Matematika*, *2*(2), 126–135.

Cattell, L. (2021). Identifying Students At Risk Using Prior Performance Versus a Machine Learning Algorithm. *Journal of Learning Analytics*.

Cózar-Gutiérrez, R., & Sáez-López, J. M. (2016). Game-based learning and gamification in initial teacher training in the social sciences: An experiment with MinecraftEdu. *International Journal of Educational Technology in Higher Education*, *13*(1). https://doi.org/10.1186/s41239-016-0003-4

Doz, D., Felda, D., & Cotic, M. (2022). Assessing Students' Mathematical Knowledge with Fuzzy Logic. *Education Sciences*, *12*(4). https://doi.org/10.3390/educsci12040266

Duman, E. Z., & Aslan, G. (2019). An Investigation into the Logic Textbooks Written in Accordance with the 2009 Logic Curriculum. *Asian Journal of Education and Training*, *5*(1), 166–175. https://doi.org/10.20448/journal.522.2019.51.166.175

Frana, P. L. (2022). Using Algorithmic Imaginaries and Uncanny Pedagogy to Facilitate Interdisciplinary Research and Digital Scholarship. *Journal of Interactive Technology and Pedagogy*.

Graham, K. M., & Kokensparger, B. (2021). Blended Instruction in Symbolic Logic. *Journal of the Scholarship of Teaching and Learning*, *21*(2), 1–14. https://doi.org/10.14434/josotl.v21i2.27666

Hariyanto, D., Triyono, M. B., & Köhler, T. (2020). Usability evaluation of personalized adaptive e-learning system using USE questionnaire. *Knowledge Management & E-Learning*, *12*(1), 85–105.

Hartati, S. J. (2022). Design of Learning Model of Logic and Algorithms Based on APOS Theory. *International Journal of Evaluation and Research in Education (IJERE)*, *3*(2), 109–118. https://doi.org/10.11591/ijere.v3i2.5743

Hartati, S. J., Vitianingsih, A. V., & Kurniati, N. (2020). Identifying the Factor of Mathematical Reasoning That Affects the Ability to Programming Algorithm. *International Education Studies*, *13*(5), 44–56.

Iyamuremye, A., & Nsabayezu, E. (2022). Mathematics and Science Teacher's Conception and Reflection on Computer Programming with Scratch: Technological and Pedagogical Standpoint. *International Journal of Education, Training and Learning*, *6*(1), 11–16. https://doi.org/10.33094/ijetl.v6i1.488

Jarrah, A. M., Almassri, H., Johnson, J. D., & Wardat, Y. (2022). Assessing the impact of digital games-based learning on students' performance in learning fractions using (ABACUS) software application. *Eurasia Journal of Mathematics, Science and Technology Education*, *18*(10). https://doi.org/10.29333/ejmste/12421

Kamisli, H. (2019). On Primary School Teachers' Training Needs in Relation to Game-Based Learning. *International Journal of Curriculum and Instruction*, *11*(2), 285–296.

Kanaki, K., & Kalogiannakis, M. (2022). Assessing Algorithmic Thinking Skills in Relation to Age in Early Childhood STEM Education. *Education Sciences*, *12*(6), 44–59. https://doi.org/10.3390/educsci12060380

Kreinovich, V. (2018). The Logic in Computer Science Column. *Bulletin of EATCS*.

Lavasani, M. G., & Khandan, F. (2022). Effectiveness of game-based learning on students' motivation and performance. *Cypriot Journal of Educational Sciences*, *2*(5), 61–74.

Logachev, M. S. (2022). Information System for Monitoring and Management of the Quality of Educational Programs: Development of Functioning Algorithms. *Journal of Open Innovation: Technology, Market, and Complexity*, *15*(3), 429–450.

Moeis, D., & Yunarti, S. (2022). Pelatihan Logika dan Algoritma Pemrograman Bagi Siswa/i SMAN 3 Makassar. *Jurnal Pengabdian Kepada Masyarakat*, *6*, 1008–1012.

Nijenhuis, J., Bayram-Jacobs, D., Meijer, P. C., & Barendsen, E. (2022). Recognizing Algorithmic Concepts in New Contexts: An Analysis of Students' Reasoning. *Informatics in Education*, *21*(3), 541–568. https://doi.org/10.15388/infedu.2022.16

Rose, S. P., Habgood, M. P. J., & Jay, T. (2020). Designing a Programming Game to Improve Children's Procedural Abstraction Skills in Scratch. *Journal of Educational Computing Research*, *58*(7), 1372–1411. https://doi.org/10.1177/0735633120932871

Saltan, F., & Kara, M. (2022). ICT Teachers' Acceptance of "Scratch" as Algorithm Visualization Software. *Higher Education Studies*, *6*(4), 146. https://doi.org/10.5539/hes.v6n4p146

Sukirman, Pramudita, D. A., Afiyanto, A., & Utaminingsih. (2022). Block-Based Visual Programming as a Tool for Learning the Concepts of Programming for Novices. *International Journal of Information and Education Technology*, *12*(5), 365–371. https://doi.org/10.18178/ijiet.2022.12.5.1628

Sun, L., & Chen, X. (2021). Digital Game-based Pedagogical Activities in Primary Education: A Review of Ten Years' Studies. *International Journal of Educational Technology in Higher Education*, *16*, 78–92.

Tekerek, M., & Tuğba, A. (2018). The effect of Scratch environment on student's achievement in teaching algorithm. *World Journal on Educational Technology*, *6*(2), 132–138.

Thangjai, N. (2022). Developing Inquiry Learning Characteristics of Grade 7 Students Using Integrated 5E's of Inquiry-Based Learning and Game-Based Learning. *Journal of Educational Issues*, *8*(1), 137–150. https://doi.org/10.5296/jei.v8i1.19547

Timur, S., Timur, B., Güvenç, E., Us, İ., & Yalçinkaya-Önder, E. (2021). Pre-Service Pre-School Teachers' Opinions About Using Block-Based Coding / Scratch. *Acta Didactica Napocensia*, *14*(2), 299–317. https://doi.org/10.24193/adn.14.2.22

Wijaya, A., Elmaini, & Doorman, M. (2021). A learning trajectory for probability: A case of game-based learning. *Journal on Mathematics Education*, *12*(1), 1–16. https://doi.org/10.22342/JME.12.1.12836.1-16

Yildiz, M., & Börekci, C. (2020). Predicting Academic Achievement with Machine Learning Algorithms. *Journal of Educational Technology and Online Learning*, *3*(3). https://doi.org/10.31681/jetol.773206