# DEVELOPMENT, VALIDATION, AND EFFECTIVENESS OF A SCRATCH-BASED INSTRUCTIONAL MEDIA FOR TEACHING LOGICAL INFORMATICS IN UZBEK SECONDARY SCHOOLS

**Khurramova Malohat Eshmumin kizi**
*International Joint Degree Program, Termez State University and Faculty of Mathematics and Physics Education, Universitas Pendidikan Indonesia*

**Abstract:** ***Logical Informatics*** *— encompassing sequencing, conditional reasoning, iteration, and algorithmic problem solving — remains one of the most challenging strands of school Informatics instruction, particularly where teaching relies on theoretical explanation rather than hands-on practice. This study addresses this challenge within the Uzbek secondary-school context by developing, validating, and evaluating a Scratch-based instructional media package through a Research and Development (R&D) approach structured around the ADDIE model (Analysis, Design, Development, Implementation, Evaluation). The media was reviewed by two experts (one Informatics content specialist and one instructional media design specialist), trialled with one Informatics teacher and 30 secondary-school students selected through purposive sampling, and evaluated using expert validation sheets, teacher and student practicality questionnaires, classroom observation, and a one-group pre-test/post-test design. Expert validation produced an overall score of 87.67% ("very valid"), with pedagogical alignment scoring highest (92%) and media design quality and usability scoring lowest (84% each). Teacher practicality was rated 89% and student practicality 87.2%, both classified as "very practical." Classroom observation indicated 85% ("very good") engagement. Mean achievement rose from 60.33 at pre-test to 83.17 at post-test, a gain of 22.84 points, with a normalized gain (N-Gain) of 0.58 (medium effectiveness); the largest indicator-level gains occurred in loops and repetition (+25) and conditional statements (+24). These findings suggest that a curriculum-aligned, block-based visual programming medium can meaningfully strengthen Logical Informatics learning in an under-researched Central Asian setting, while underscoring the need for larger-scale, controlled, and longitudinal follow-up research.*

***Keywords:*** *Scratch programming; Logical Informatics; computational thinking; instructional media development; ADDIE model; Research and Development; secondary education; Uzbekistan*

## 1. INTRODUCTION

Informatics education has become a central component of twenty-first-century schooling, closely tied to the development of critical thinking, digital literacy, and algorithmic reasoning (Wing, 2006; Grover & Pea, 2018). Within this broader field, Logical Informatics — the study of logical operators, sequencing, conditional structures, iteration, and algorithm design — functions as the cognitive foundation on which later programming instruction is built; without it, learners often struggle when they encounter syntax-heavy languages such as Python, Java, or C++ (Lye & Koh, 2014). International curricula have consequently prioritized the early development of logical reasoning skills (Yadav et al., 2021).

Despite this consensus, Informatics instruction in many contexts remains lecture-dominant and textbook-driven, emphasizing definitional recall over applied problem solving (Rahman et al., 2022). Papert's (1980) constructionist account of learning argues that programming is best learned through exploratory, playful construction of artefacts rather than passive absorption of rules — an argument echoed by decades of subsequent research into block-based visual programming environments. Among these, Scratch, developed by the MIT Media Lab, has become one of the most widely studied tools for introducing novice learners to computational thinking. By allowing students to assemble pre-structured command blocks rather than write syntactically fragile code, Scratch removes a major early barrier to programming and lets learners focus on logic itself (Resnick et al., 2009; Maloney et al., 2010). A substantial body of research links Scratch use to gains in algorithmic reasoning, debugging ability, creativity, and learner confidence (Kalelioğlu, 2015; Sáez-López et al., 2016; Durak et al., 2019; Zhang & Nouri, 2019).

This international evidence base has not yet translated into widespread classroom practice in Central Asia. Uzbekistan's national digital-transformation agenda — including the "Digital Uzbekistan 2030" initiative — explicitly calls for stronger computational literacy in schools, yet recent national studies report that Informatics classrooms remain largely theoretical, with limited access to visual programming tools, insufficient teacher training in block-based pedagogy, and few Uzbek-language, curriculum-aligned learning resources (Mukhitdinov, 2023; Rakhimova, 2024; Tursunov, 2024; Akhmedov, 2024). Consequently, although Uzbek learners appear motivated when exposed to visual programming activities, no classroom-ready Scratch-based media currently exists for the Logical Informatics component of the national curriculum.

This gap between international evidence and local implementation motivates the present study. Following a Research and Development (R&D) approach structured around the ADDIE instructional design model, this study developed a Scratch-based instructional media package for Logical Informatics and evaluated it along three widely used dimensions in educational-technology research: validity (expert judgment), practicality (teacher and student response, and classroom observation), and effectiveness (pre-test/post-test learning gain). The study was guided by four research questions: (1) How can Scratch-based learning media be developed to support Logical Informatics instruction? (2) How valid is the developed media according to expert evaluation? (3) How practical is the media in classroom use? (4) How effective is the media in improving student learning outcomes? By answering these questions, the study contributes empirical evidence from an under-represented Central Asian setting to the international literature on visual programming and computational thinking instruction.

## 2. Literature Review

### *2.1 Logical Informatics and Computational Thinking*

Logical Informatics comprises the reasoning competencies — sequencing, abstraction, conditional logic, iteration, decomposition, and pattern recognition — that underlie programming and computational problem solving more broadly (Wing, 2006; Grover & Pea, 2018). Scholars position it not merely as preparatory content for later coding instruction but as a distinct cognitive training process that strengthens systematic thinking and decision making (Angeli et al., 2016; Shute et al., 2017), with demonstrated links to later achievement in mathematics and engineering (Bocconi et al., 2018; Kazakoff & Bers, 2014). Because these skills are cumulative, researchers argue they must be taught early, through learner-centred, constructivist environments rather than passive lecture formats (Papert, 1980; Kafai & Burke, 2015; Lai & Moss, 2016).

### *2.2 Scratch as a Vehicle for Logical Reasoning*

Scratch operationalizes constructionist principles by letting learners build interactive projects from visual command blocks, eliminating syntax errors that typically frustrate novice programmers (Resnick et al., 2009; Meerbaum-Salant et al., 2013). Because logical structures such as "if-then" conditionals and "repeat" loops are rendered as manipulable, visible objects, students can observe the consequences of their logical decisions directly, which several studies link to stronger algorithmic reasoning and reduced cognitive load (Kalelioğlu, 2015; Bocconi et al., 2018; Sweller, 2011). Systematic reviews of the Scratch literature confirm consistent gains in computational thinking, abstraction, and debugging across diverse educational settings (Zhang & Nouri, 2019; Durak et al., 2019; Stewart & Baek, 2023), while also noting that outcomes depend heavily on instructional scaffolding, teacher preparedness, and contextual adaptation (Bocconi et al., 2018; Meerbaum-Salant et al., 2013).

Beyond cognitive gains, Scratch supports multimodal and collaborative learning: students encode meaning through animation, sound, and interactivity (Jewitt, 2018), share and remix projects within an international community (Brennan & Resnick, 2012), and — because block manipulation relies on visual rather than linguistic comprehension — the platform has been associated with more inclusive participation across ability levels and gender (Kafai & Burke, 2015; Kazimoglu et al., 2012). Recent regional studies from Latin America and East Asia similarly report that block-based tools such as Scratch support programming-competency development even where infrastructure and teacher training are limited, suggesting the approach generalizes beyond the contexts in which it was originally validated (Vidal-Silva et al., 2024; Piedade & Dorotea, 2023).

### *2.3 Instructional Media Development and the ADDIE Framework*

Instructional media development is most effective when pedagogically, rather than technologically, driven (Reigeluth & Carr-Chellman, 2009), and should satisfy criteria of clarity, interactivity, learner motivation, and contextual relevance (Mishra & Koehler, 2006; Mayer, 2014). The ADDIE model — Analysis, Design, Development, Implementation, and Evaluation — remains one of the most widely used frameworks for this purpose because it embeds iterative validation into the production process (Branch, 2009; Molenda, 2015), complemented by usability principles that ensure learners can navigate and operate the product independently (Nielsen, 2012). Within computer science education specifically, digital media that allow learners to observe cause-and-effect relationships and test algorithmic reasoning are associated with deeper conceptual mastery than static, text-based materials (Durak et al., 2019; Kalelioğlu, 2015).

### *2.4 Prior Studies and the Research Gap*

Foundational work by Resnick et al. (2009) and Maloney et al. (2010) established Scratch as a legitimate academic programming environment, and subsequent quasi-experimental studies (Kalelioğlu, 2015; Sáez-López et al., 2016; Durak et al., 2019; Shute et al., 2017) confirmed measurable gains in algorithmic thinking, debugging, and conditional reasoning. Affective outcomes — motivation, reduced anxiety, and creative ownership — have also been documented (Lye & Koh, 2014; Brennan & Resnick, 2012; Harman et al., 2021). However, most of this evidence concerns general programming achievement rather than Logical Informatics as a distinct curricular domain, and comparatively few studies apply a formal R&D methodology to produce a validated, classroom-ready instructional product rather than describing an intervention post hoc (Kazimoglu et al., 2012; Bocconi et al., 2018).

Regional coverage is a further limitation. Recent systematic reviews note that Scratch research remains concentrated in Europe, North America, and parts of East Asia, with very limited empirical evidence from Central Asia or post-Soviet education systems (Zhang & Nouri, 2019; Stewart & Baek, 2023; Sun et al., 2026). Local Uzbek research confirms that Informatics instruction remains theory-dominated and lacking in interactive, curriculum-aligned resources (Mukhitdinov, 2023; Rakhimova, 2024), but no prior study has produced or empirically evaluated Scratch-based Logical Informatics media adapted to the Uzbek curriculum and language context. This study addresses that gap.

## 3. Materials and Methods

### *3.1 Research Design*

This study used a Research and Development (R&D) design structured around the ADDIE model to design, validate, and evaluate a Scratch-based instructional media package for Logical Informatics. In the Analysis stage, curriculum documents, student learning difficulties, and existing instructional practice were reviewed to define content requirements. The Design stage produced storyboards, lesson sequencing, and task outlines aligned with four Logical Informatics indicators: sequencing, conditional logic, loops, and algorithmic problem solving. During Development, the digital Scratch media was produced and submitted to expert review; revisions were made iteratively until the product met validation criteria. The Implementation stage involved classroom try-outs with the target student and teacher sample, during which practicality data were collected. Finally, the Evaluation stage assessed learning effectiveness through a one-group pre-test/post-test comparison. This design corresponds to a single-group pre-experimental format; it does not include a control or comparison group, a constraint addressed further in the Limitations section.

### *3.2 Participants*

Participants were drawn from a government secondary school in Uzbekistan using purposive sampling, appropriate for R&D research in which the goal is to evaluate product performance among authentic, information-rich users rather than to generalize to a wider population. The student sample comprised 30 secondary-level Informatics students currently studying the Logical Informatics curriculum component, with no exclusion by sex or socioeconomic background. One experienced Informatics teacher participated as a practicality evaluator and facilitated classroom implementation. Two experts — one Informatics content specialist and one instructional media design specialist — conducted the formal validation review. Participation was voluntary, and informed consent was obtained from students, guardians, and staff prior to data collection; procedures for confidentiality and withdrawal without penalty are detailed in Section 3.5.

### *3.3 Instruments and Data Collection*

Data were collected in three phases. First, in the expert validation phase, two specialists rated the media on a five-point Likert scale (1 = very poor, 5 = very good) across six aspects: content accuracy, instructional clarity, media design quality, pedagogical alignment, usability and functionality, and language appropriateness, supplemented by open-ended qualitative comments. Second, in the classroom implementation phase, practicality was assessed through separate Likert-scale questionnaires for the teacher (covering ease of use, clarity of instruction, curriculum relevance, support for engagement, and classroom-implementation suitability) and for students (covering interest, ease of use, clarity of instructions, motivation, and helpfulness for understanding logic concepts), together with a structured observation checklist recording attention, participation, Scratch navigation ability, task completion, collaboration, and problem-solving behaviour. Third, in the learning-effectiveness phase, a pre-test and post-test — comprising multiple-choice, short-answer, and logical problem-solving items aligned with the four Logical Informatics indicators — were administered before and after the intervention. Informal interviews and open-ended questionnaire items supplemented the quantitative instruments to capture student and teacher perspectives on engagement and conceptual clarity.

### *3.4 Data Analysis*

Expert validation, teacher practicality, student practicality, and observation data were analysed descriptively as percentage scores relative to each instrument's maximum possible score, and classified using standard interpretive categories (e.g., "very valid," "very practical," "very good") consistent with prior instructional-media validation research (Sáez-López et al., 2016; Harman et al., 2021). Learning effectiveness was analysed by comparing pre-test and post-test means, standard deviations, and gain scores, and by computing the normalized gain (N-Gain) using the standard formula:

N-Gain values were interpreted using conventional thresholds (high $\geq 0.7$; medium 0.3–0.7; low $< 0.3$). Indicator-level gains were also computed separately for sequencing, conditional statements, loops and repetition, and algorithmic problem solving to identify differential effectiveness across Logical Informatics sub-domains.

### *3.5 Ethical Considerations*

The study was conducted with approval from the school administration and participating teacher, and informed consent was obtained from students and guardians prior to involvement. Participation was voluntary, with no academic or evaluative consequence for withdrawal or for providing critical feedback on the media, and teachers were not involved in recruiting specific student participants in order to avoid coercion. Data were anonymized, coded, and stored securely, and results are reported only in aggregated form. All Scratch projects used in the media were original or appropriately attributed to respect intellectual property rights.

## 4. Results

### *4.1 Expert Validation*

The developed media received an overall expert validation score of 87.67% (263 of 300 possible points), placing it in the "very valid" category (Table 1). Pedagogical alignment received the highest rating (92%), indicating that the Scratch activities were judged strongly aligned with Logical Informatics learning objectives. Media design quality and usability and functionality received the lowest ratings (84% each) — still within the valid range, but pointing to specific opportunities to simplify navigation and improve visual consistency.

*Table 1. Expert Validation Results (n = 2 experts)*

| Validation Aspect | Score Obtained | Maximum Score | Percentage | Category |
|---|---|---|---|---|
| Content accuracy | 45 | 50 | 90% | Very valid |
| Instructional clarity | 43 | 50 | 86% | Very valid |
| Media design quality | 42 | 50 | 84% | Valid |
| Pedagogical alignment | 46 | 50 | 92% | Very valid |
| Usability and functionality | 42 | 50 | 84% | Valid |
| Language appropriateness | 45 | 50 | 90% | Very valid |
| Overall | 263 | 300 | 87.67% | Very valid |

### *4.2 Teacher and Student Practicality*

Both teacher and student practicality ratings fell into the "very practical" category (Table 2). The teacher rated curriculum relevance highest (95%) and clarity of instruction and classroom-implementation suitability lowest (85% each), suggesting the media was well aligned to curricular goals but required some initial orientation. Students (n = 30) rated helpfulness in understanding logic concepts highest (92%) and clarity of instructions lowest (82%), indicating that while the media supported conceptual understanding, some learners needed more explicit step-by-step guidance, consistent with their limited prior exposure to Scratch.

*Table 2. Teacher and Student Practicality Results*

| Respondent / Aspect | Score | Maximum | Percentage | Category |
|---|---|---|---|---|
| Teacher — Ease of use | 18 | 20 | 90% | Very practical |
| Teacher — Clarity of instruction | 17 | 20 | 85% | Practical |
| Teacher — Curriculum relevance | 19 | 20 | 95% | Very practical |
| Teacher — Student engagement support | 18 | 20 | 90% | Very practical |
| Teacher — Classroom-implementation suitability | 17 | 20 | 85% | Practical |
| Teacher — Overall | 89 | 100 | 89% | Very practical |
| Students — Interest in learning | 132 | 150 | 88% | Very practical |
| Students — Ease of using Scratch | 126 | 150 | 84% | Practical |
| Students — Clarity of instructions | 123 | 150 | 82% | Practical |
| Students — Motivation to learn | 135 | 150 | 90% | Very practical |
| Students — Helpfulness for understanding logic | 138 | 150 | 92% | Very practical |

| Respondent / Aspect | Score | Maximum | Percentage | Category |
|---|---|---|---|---|
| Students — Overall | 654 | 750 | 87.2% | Very practical |

### *4.3 Classroom Observation*

Observation data corroborated the practicality findings, with an overall score of 85% ("very good"; Table 3). Student attention (90%) and collaboration with classmates (86.67%) were the strongest-rated behaviours. The ability to use Scratch blocks received the lowest score (80%, "good"), reflecting that some students required initial guidance before operating the interface independently, though most completed assigned tasks successfully after practice.

*Table 3. Classroom Observation Results*

| Observation Aspect | Score | Maximum | Percentage | Category |
|---|---|---|---|---|
| Student attention | 27 | 30 | 90% | Very good |
| Participation in activities | 26 | 30 | 86.67% | Very good |
| Ability to use Scratch blocks | 24 | 30 | 80% | Good |
| Task completion | 25 | 30 | 83.33% | Good |
| Collaboration with classmates | 26 | 30 | 86.67% | Very good |
| Problem-solving behaviour | 25 | 30 | 83.33% | Good |
| Overall | 153 | 180 | 85% | Very good |

***4.4 Learning Effectiveness***

Descriptive statistics for the pre-test and post-test are reported in Table 4. Mean achievement increased from 60.33 (SD = 8.21) at pre-test to 83.17 (SD = 6.74) at post-test, an average gain of 22.84 points across all 30 students, with every participant showing improvement. The minimum score rose from 45 to 70, indicating that students with the weakest initial performance also benefited from the intervention, and the reduction in standard deviation suggests achievement became more homogeneous following the intervention. The normalized gain (N-Gain) was 0.58, classified as medium effectiveness.

*Table 4. Descriptive Statistics for Pre-Test and Post-Test Scores (n = 30)*

| Measure | Pre-Test | Post-Test |
|---|---|---|
| Minimum score | 45 | 70 |
| Maximum score | 75 | 95 |
| Mean score | 60.33 | 83.17 |
| Standard deviation | 8.21 | 6.74 |
| Mean gain | — | 22.84 |
| N-Gain (category) | — | 0.58 (Medium) |

Gains were not uniform across the four Logical Informatics indicators (Table 5). The largest improvement occurred in loops and repetition (+25 points), followed by conditional statements (+24), sequencing (+23), and algorithmic problem solving (+20), suggesting that Scratch's visual, immediate-feedback representation of iterative and conditional structures was particularly effective for concepts that are otherwise difficult to visualize through text-based explanation alone.

*Table 5. Learning Indicator Achievement*

| Learning Indicator | Pre-Test Mean | Post-Test Mean | Improvement |
|---|---|---|---|
| Sequencing | 65 | 88 | +23 |

| Learning Indicator | Pre-Test Mean | Post-Test Mean | Improvement |
|---|---|---|---|
| Conditional statements | 58 | 82 | +24 |
| Loops and repetition | 56 | 81 | +25 |
| Algorithmic problem solving | 62 | 82 | +20 |

## 5. Discussion

This study asked how a Scratch-based instructional media package for Logical Informatics could be developed, and how valid, practical, and effective such a package would be in an Uzbek secondary-school setting. The findings offer converging evidence across all three evaluation dimensions.

With respect to validity, the overall expert score of 87.67% indicates that the ADDIE-based development process — moving systematically through analysis, design, development, and iterative expert review — produced media judged appropriate in content, structure, and pedagogy (Branch, 2009; Molenda, 2015). The particularly strong pedagogical-alignment rating (92%) supports the theoretical premise, articulated by Wing (2006) and operationalized by Resnick et al. (2009), that visual programming environments can translate abstract logical structures — sequencing, conditionals, iteration — into concrete, manipulable representations. The comparatively lower scores for design quality and usability (84%) are informative rather than disqualifying: they point to specific, addressable refinements (navigation, visual consistency) rather than to a fundamental pedagogical shortcoming, and are broadly consistent with prior findings that Scratch implementations benefit from iterative, context-specific interface refinement (Bocconi et al., 2018).

With respect to practicality, both teacher (89%) and student (87.2%) ratings were high, aligning with prior evidence that Scratch increases engagement and lowers the affective barriers associated with early programming instruction (Sáez-López et al., 2016; Brennan & Resnick, 2012). The teacher's highest rating — curriculum relevance (95%) — suggests the media succeeded in its core design goal of aligning visual programming tasks with existing Logical Informatics learning objectives rather than functioning as a disconnected add-on. The comparatively lower ratings for clarity of instruction (teacher: 85%; students: 82%) and for students' ease of using Scratch (84%) converge with the observation data, where the ability to use Scratch blocks was the weakest-rated behaviour (80%). Read together, these results suggest that most implementation friction stemmed not from the instructional content itself but from students' unfamiliarity with the Scratch interface — a first-exposure effect also reported by Meerbaum-Salant et al. (2013) and one that a brief orientation session could plausibly mitigate.

With respect to effectiveness, the increase from a pre-test mean of 60.33 to a post-test mean of 83.17 (N-Gain = 0.58) indicates a medium, educationally meaningful improvement, consistent with the broader literature linking Scratch-based instruction to gains in computational thinking and algorithmic reasoning (Kalelioğlu, 2015; Durak et al., 2019; Shute et al., 2017). The indicator-level pattern — with the largest gains in loops and conditional statements — is theoretically coherent: these are the structures for which Scratch's visual, immediatelyobservable block behaviour most directly substitutes for the abstract mental simulation that text-based explanation typically demands (Meerbaum-Salant et al., 2013; Sweller, 2011). That the N-Gain fell in the medium rather than high category is plausibly attributable to two factors evident in the data itself: many students were first-time Scratch users, and the implementation period was comparatively short, both of which are known moderators of programming-instruction effectiveness (Lye & Koh, 2014).

Collectively, these results support the broader argument that block-based visual programming can reduce the cognitive load associated with syntax acquisition and thereby free cognitive resources for logical reasoning itself (Sweller, 2011; Resnick et al., 2009). Within the Uzbek context specifically, where Informatics instruction has been described as predominantly theoretical and under-resourced in digital learning tools (Mukhitdinov, 2023; Rakhimova, 2024), these findings provide direct, localized empirical support for the value of curriculum-aligned Scratch media — extending an evidence base that has, to date, been concentrated in European, North American, and East Asian settings (Zhang & Nouri, 2019; Stewart & Baek, 2023) and only recently begun to include comparable developing-country contexts (Vidal-Silva et al., 2024).

Two interpretive caveats are warranted. First, the pre-experimental one-group design means that the observed gains, while consistent with a genuine instructional effect, cannot be fully isolated from maturation, testing, or novelty effects in the absence of a control group; the result should therefore be read as evidence of a promising intervention rather than a definitively causal effect. Second, because the sample was drawn from a single school through purposive sampling, the practicality and effectiveness findings are best interpreted as evidence of feasibility and initial efficacy within a comparable Uzbek secondary-school context rather than as population-level estimates.

## 6. Conclusion

This study developed a Scratch-based instructional media package for teaching Logical Informatics in Uzbek secondary schools using an ADDIE-structured Research and Development approach, and evaluated it for validity, practicality, and effectiveness. The media was judged very valid by expert reviewers (87.67%), very practical by both the participating teacher (89%) and students (87.2%), and effective in improving learning outcomes, with mean achievement rising from 60.33 to 83.17 (N-Gain = 0.58, medium effectiveness) and the strongest gains occurring in loops, repetition, and conditional reasoning. Together, these findings indicate that curriculum-aligned, block-based visual programming media can help transform abstract Logical Informatics content into interactive, learner-centred experience, offering a practical, evidence-based tool for Informatics teachers and a contribution to the still-limited empirical literature on computational thinking instruction in Central Asia.

## 7. Limitations

This study has several limitations that should inform the interpretation of its findings and guide future research. The sample was limited to 30 students, one teacher, and two experts drawn from a single school, constraining the generalizability of the practicality and effectiveness results. The one-group pre-test/post-test design, without a control or comparison group, limits causal inference regarding the specific contribution of the Scratch media relative to other factors such as maturation or repeated testing. The implementation period was relatively short, which may partly explain the medium (rather than high) N-Gain and precludes any conclusion about longer-term retention. Many students were using Scratch for the first time, which may have constrained early engagement independent of the instructional content itself. The study addressed only introductory Logical Informatics content (sequencing, conditionals, loops, and basic algorithmic problem solving) and did not extend to variables, functions, or transition toward text-based programming languages, nor did it compare Scratch against other block-based platforms such as Blockly, Alice, or App Inventor. Finally, data collection relied primarily on validated quantitative instruments and structured observation rather than extended qualitative methods (e.g., in-depth interviews, project-based analysis, or video coding), which could offer richer insight into students' reasoning processes. Future research should address these constraints through larger, multi-site, controlled, and longitudinal designs; comparative evaluation against alternative visual programming platforms; broader instructional content; and richer qualitative analysis of student reasoning.

## REFERENCES:

Akhmedov, U. (2024). Teacher readiness toward programming curriculum in Uzbekistan. Journal of Central Asian Education Research, 11(2), 44–58. https://doi.org/10.5678/jcaer.2024.11204

Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., & Zagami, J. (2016). A K–6 computational thinking curriculum framework. Journal of Educational Technology & Society, 19(3), 47–57.

Bers, M. U. (2019). Coding as another language: The role of programming in literacy development. Computers & Education, 145, 103728. https://doi.org/10.1016/j.compedu.2019.103728

Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., Engelhardt, K., Kampylis, P., & Punie, Y. (2018). Developing computational thinking in compulsory education. Journal of Research on Technology in Education, 50(2), 149–176. https://doi.org/10.1080/15391523.2018.1445428

Branch, R. M. (2009). Instructional design: The ADDIE approach. Springer. https://doi.org/10.1007/978-0-387-09506-6

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. Proceedings of the American Educational Research Association Conference.

Durak, H., Yilmaz, F., & Yilmaz, R. (2019). Scratch-based programming instruction improves abstraction skills. Education and Information Technologies, 24(2), 913–928. https://doi.org/10.1007/s10639-018-9803-7

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review. Educational Researcher, 42(1), 38–43. https://doi.org/10.3102/0013189X12463051

Grover, S., & Pea, R. (2018). Design thinking and coding integration. Journal of Computer Science Education, 28(4), 345–362. https://doi.org/10.1080/08993408.2018.1457897

Harman, S., O'Leary, H., & Fitzsimmons, J. (2021). Scratch programming and creative engagement. Computers in Schools, 38(3), 230–249. https://doi.org/10.1080/07380569.2021.1882749

Jewitt, C. (2018). Multimodality and literacy learning. Journal of Language and Education, 4(3), 5–23. https://doi.org/10.17323/2411-7390-2018-4-3-5-23

Kafai, Y. B., & Burke, Q. (2015). Constructionist gaming and identity. Educational Psychologist, 50(4), 313–334. https://doi.org/10.1080/00461520.2015.1122532

Kalelioğlu, F. (2015). A quasi-experimental Scratch instruction study. Computers & Education, 88, 1–12. https://doi.org/10.1016/j.compedu.2015.04.011

Kazakoff, E. R., & Bers, M. U. (2014). Algorithm learning through visual coding. Computers in Human Behavior, 41, 93–100. https://doi.org/10.1016/j.chb.2014.09.001

Kazimoglu, C., Kiernan, M., Bacon, L., & Mackinnon, L. (2012). Scratch game for math problem solving. Computers & Education, 63, 407–422. https://doi.org/10.1016/j.compedu.2012.12.004

Lai, K., & Moss, G. (2016). Logical thinking and Scratch. Computers in Education Journal, 26(3), 73–90.

Lye, S. Y., & Koh, J. H. L. (2014). Review of programming and computational thinking. Computers in Human Behavior, 41, 51–66. https://doi.org/10.1016/j.chb.2014.09.012

Maloney, J., Resnick, M., Rusk, N., Silverman, B., & Eastmond, E. (2010). Scratch design model. Communications of the ACM, 52(11), 60–67. https://doi.org/10.1145/1592761.1592779

Mayer, R. E. (2014). Cognitive theory of multimedia learning. Journal of Educational Psychology, 106(2), 598–610. https://doi.org/10.1037/a0035836

Meerbaum-Salant, O., Armoni, M., & Ben-Ari, M. (2013). Roles of Scratch in learning loops and conditionals. Computers & Education, 63, 23–36. https://doi.org/10.1016/j.compedu.2012.11.005

Mishra, P., & Koehler, M. J. (2006). Technological pedagogical content knowledge: A framework for teacher knowledge. Teachers College Record, 108(6), 1017–1054.

Molenda, M. (2015). In search of the ADDIE model. Educational Technology, 55(2), 4–13.

Mukhitdinov, B. (2023). Digital transformation in Uzbek Informatics classrooms. Uzbek Journal of ICT Education, 9(4), 22–35.

Nielsen, J. (2012). Usability theory and digital interface design. Journal of Web Interaction, 7(2), 118–134.

Papert, S. (1980). Mindstorms: Children, computers, and powerful ideas. Basic Books.

Piedade, J., & Dorotea, N. (2023). Effects of Scratch-based activities on 4th-grade students' computational thinking skills. Informatics in Education, 22(3), 499–523. https://doi.org/10.15388/infedu.2023.19 [Suggested addition]

Rahman, R., Yusuf, M., & Aziz, H. (2022). Teaching logical reasoning in schools. International Journal of Education, 14(1), 66–77.

Rakhimova, S. (2024). Challenges in Uzbek Informatics curriculum delivery. Central Asian ICT Studies, 12(3), 87–101.

Reigeluth, C. M., & Carr-Chellman, A. A. (2009). Instructional-design theories and models. Routledge. https://doi.org/10.4324/9780203873005

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Silverman, B., & Eastmond, E. (2009). Scratch programming environment. Communications of the ACM, 52(11), 60–67. https://doi.org/10.1145/1592761.1592779

Sáez-López, J. M., Román-González, M., & Vázquez-Cano, E. (2016). Scratch and algorithmic thinking. Computers & Education, 103, 114–127. https://doi.org/10.1016/j.compedu.2016.09.003

Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). CT assessment framework. Educational Research Review, 22, 1–30. https://doi.org/10.1016/j.edurev.2017.09.003

Stewart, W., & Baek, K. (2023). Analyzing computational thinking studies in Scratch programming: A review of elementary education literature. International Journal of Computer Science Education in Schools, 6(1), 35–58. https://doi.org/10.21585/ijcses.v6i1.156 [Suggested addition]

Sun, J., Wang, R., & Wei, B. (2026). Teaching computational thinking skills through Scratch in K–12 education: A systematic review. Journal of Computer Assisted Learning. Advance online publication. https://doi.org/10.1002/jcal.70230 [Suggested addition]

Sweller, J. (2011). Cognitive load theory and instruction. Psychological Review, 118(2), 284–302. https://doi.org/10.1037/a0022684

Tursunov, A. (2024). ICT teacher competency study in Uzbekistan. International Journal of Modern Education, 6(2), 55–70. https://doi.org/10.2224/ijme.2024.62055

Vidal-Silva, C., Cardenas-Cobo, J., Tupac-Yupanqui, M., Serrano-Malebran, J., & Sanchez Ortiz, A. (2024). Developing programming competencies in school students with block-based tools in Chile, Ecuador, and Peru. IEEE Access, 12, 118924–118936. https://doi.org/10.1109/ACCESS.2024.3449228 [Suggested addition]

Wing, J. M. (2006). Computational thinking overview. Communications of the ACM, 49(3), 33–35. https://doi.org/10.1145/1118178.1118215

Wing, J. M. (2010). Progress in computational thinking. Communications of the ACM, 54(4), 25–28. https://doi.org/10.1145/1785414.1785430

Yadav, A., Gretter, S., Good, J., & McLean, T. (2021). Computational thinking integration in schools. TechTrends, 65(2), 1–15. https://doi.org/10.1007/s11528-020-00571-y

Zhang, L., & Nouri, J. (2019). A systematic review of learning computational thinking through Scratch in K–9. Computers & Education, 141, 103607. https://doi.org/10.1016/j.compedu.2019.103607 [Citation present in original thesis text but missing from its reference list; restored here]