

ОБУЧЕНИЕ ИНФОРМАТИКЕ TEACHING COMPUTER SCIENCE

DOI: 10.22363/2312-8631-2025-22-4-498-510

EDN: GOVYRO

УДК 378.1

Научная статья / Research article

Обучение программированию в российских школах на уровне основного общего образования: подходы и направления развития

К.М. Колос^{}*Школа № 2097, Москва, Российская Федерация*✉ km.kolos@s2097.ru

Аннотация. *Постановка проблемы.* В условиях стремительного развития цифровых технологий обучение программированию в школе становится не только средством формирования базовой цифровой грамотности, но и инструментом развития аналитического мышления, способности к решению сложных задач. Однако существующие подходы к обучению программированию в основной школе имеют ряд ограничений, связанных с неудовлетворительной практической направленностью, недостатком индивидуализации и низким уровнем вовлеченности учащихся. Актуальная задача – анализ существующих подходов, выявление наиболее эффективных методов, соответствующих требованиям современной образовательной среды. *Методология.* Проведены анализ нормативных документов (ФГОС, ПООП), сравнение базового и углубленного уровней обучения программированию, рассмотрены основные подходы к обучению программированию: традиционный, деятельностный, проблемный и личностно ориентированный. *Результаты.* Каждый из рассмотренных подходов имеет как преимущества, так и ограничения. Традиционный подход обеспечивает системность, но снижает мотивацию учащихся. Деятельностный и проблемный подходы развивают практические навыки, но требуют высокой квалификации преподавателя и дополнительных ресурсов. Личностно ориентированный и интерактивный подходы способствуют вовлеченности, однако их широкая реализация в школьной программе затруднена. *Заключение.* Оптимальная стратегия – интеграция различных подходов для формирования гибридного подхода, сочетающего системность традиционного обучения, практическую направленность деятельностного подхода и адаптацию к индивидуальным особенностям учащихся. Такой подход позволит повысить эффективность обучения программированию в условиях современных технологических вызовов.

© Колос К.М., 2025

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License
<https://creativecommons.org/licenses/by-nc/4.0/legalcode>

Ключевые слова: методика обучения программированию, подходы к обучению программированию, традиционный подход к обучению, деятельностный подход, проблемный подход, личностно ориентированный подход

Заявление о конфликте интересов. Автор заявляет об отсутствии конфликта интересов.

История статьи: поступила в редакцию 1 февраля 2025 г.; доработана после рецензирования 16 марта 2025 г.; принята к публикации 28 марта 2025 г.

Для цитирования: Колос К.М. Обучение программированию в российских школах на уровне основного общего образования: подходы и направления развития // Вестник Российского университета дружбы народов. Серия: Информатизация образования. 2025. Т. 22. № 4. С. 498–510. <http://doi.org/10.22363/2312-8631-2025-22-4-498-510>

Teaching programming in Russian schools at the basic general education level: approaches and development directions

Kirill M. Kolos 

School No. 2097, Moscow, Russian Federation

✉ km.kolos@s2097.ru

Abstract. *Problem statement.* In the context of rapid digital technology development, programming education in schools serves not only as a means of developing basic digital literacy but also as a tool for fostering analytical thinking and problem-solving skills. However, existing approaches to teaching programming in primary schools have several limitations, including insufficient practical orientation, a lack of individualized learning, and low student engagement. The key challenge is to analyze current approaches and identify the most effective ones that meet the demands of the modern educational environment. *Methodology.* Analyzed regulatory documents (FGOS, POOP), compares basic and advanced levels of teaching programming and examines the primary approaches to teaching programming: traditional, activity-based, problem-based, and student-centered. *Results.* Each approach has both advantages and limitations. The traditional approach ensures systematic learning but reduces student motivation. The activity-based and problem-based approaches foster practical skills but require highly qualified teachers and additional resources. The student-centered approach increases engagement; however, its broad implementation in school curricula remains challenging. *Conclusion.* The optimal strategy involves integrating various approaches to form a hybrid model that combines the systematic nature of traditional education, the practical focus of the activity-based approach, and the adaptation to individual student needs. This approach enhances the effectiveness of programming education in response to modern technological challenges.

Keywords: methodology of teaching programming, approaches to teaching programming, traditional approach, activity-based approach, problem-based approach, student-centered approach

Conflicts of interest. The author declares that there is no conflict of interest.

Article history: received 1 February 2025; revised 16 March 2025; accepted 28 March 2025.

For citation: Kolos KM. Teaching programming in Russian schools at the basic general education level: approaches and development directions. *RUDN Journal of Informatization in Education*. 2025;22(4):498–510. (In Russ.) <http://doi.org/10.22363/2312-8631-2025-22-4-498-510>

Постановка проблемы. В эпоху стремительной цифровизации программирование занимает центральное место в системе школьного образования. Этот навык стал не только важным инструментом профессионального роста, но и необходимостью в повседневной жизни, где требуется умение структурировать и анализировать большие объемы информации. Формирование алгоритмического мышления, освоение работы с данными и базовые навыки программирования – ключевые элементы, которые помогают школьникам адаптироваться к вызовам цифрового общества.

В Российской Федерации стратегия развития школьных образовательных программ задается следующими нормативными документами:

- Федеральные государственные образовательные стандарты (ФГОС)¹;
- Примерная основная образовательная программа (ПООП)²;
- Федеральная рабочая программа (ФРП)³.

Для наглядного понимания функций и задач ключевых нормативных документов можно обратиться к табл. 1.

Эти документы формируют основу образовательного процесса, в рамках которого программирование становится важным инструментом развития следующих цифровых компетенций:

- алгоритмического мышления;
- компьютерной грамотности и информационной культуры;
- навыков анализа и обработки данных.

Таблица 1

Ключевые нормативные документы и их задачи в образовательном процессе

Нормативный документ	Основная цель	Задача
ФГОС	Определение образовательных стандартов	Определение базовых требований к образовательным результатам
ПООП	Уточнение содержания образовательного процесса	Распределение учебных часов, содержание предметов
ФРП	Детализация учебного процесса	Реализация содержания на практике

Источник: составлено К.М. Колосом.

¹ Примерная основная образовательная программа основного общего образования // Правовой сайт КонсультантПлюс. URL: https://www.consultant.ru/document/cons_doc_LAW_282455/ (дата обращения: 27.09.2024).

² Приказ Министерства просвещения РФ от 16.11.2022 № 993 «Об утверждении федеральной образовательной программы основного общего образования» // Информационно-правовой портал Гарант.ру. URL: <https://www.garant.ru/products/ipo/prime/doc/405897655/> (дата обращения: 26.09.2024).

³ Приказ Министерства просвещения РФ от 16.11.2022 № 993 «Об утверждении федеральной образовательной программы основного общего образования» // Информационно-правовой портал Гарант.ру. URL: <https://www.garant.ru/products/ipo/prime/doc/405897655/> (дата обращения: 26.09.2024).

Table 1

Key regulatory documents and their roles in the educational process

Regulatory document	Main goal	Problem
FGOS	Definition of educational standards	Establishment of basic requirements for educational outcomes
POOP	Specification of the educational process content	Allocation of study hours and subject content
FRP	Detailing of the learning process	Implementation of the content in practice

Source: compiled by Kirill M. Kolos.

Цели – анализ существующих подходов к обучению программированию в российских школах; исследование методической базы, лежащей в основе образовательных программ; выявление ограничений действующих подходов и определение направлений их совершенствования.

Основное внимание сосредоточено на анализе нормативных документов, определяющих содержание образовательного процесса, а также на изучении программ обучения программированию, представленных на базовом и углубленном уровнях. Рассматриваются как традиционные, так и инновационные подходы, включая личностно ориентированный, проблемный и др.

Вопросы методики обучения программированию активно изучаются в современной педагогической науке. Исследователи анализируют различные подходы к обучению, включая традиционный, деятельностный, проблемный и личностно ориентированный [1–4]. В рамках существующих работ особое внимание уделяется развитию алгоритмического мышления, использованию проектной деятельности и созданию условий для персонализации обучения.

Однако, несмотря на многообразие исследований, остается ряд нерешенных вопросов. В частности, отсутствует комплексный анализ эффективности сочетания различных подходов, который позволил бы выявить оптимальные стратегии обучения в условиях цифровизации образования. Кроме того, в существующих работах недостаточно изучено использование современных технологий, таких как интерактивная визуализация данных, в качестве инструмента повышения наглядности и вовлеченности учащихся.

Главная задача – на основе анализа нормативной базы (ФГОС, ПООП, ФРП), содержания курсов информатики базового и углубленного уровней и современной методической литературы сопоставить традиционный, деятельностный, проблемный и личностно ориентированный подходы к обучению программированию в основной школе, выявить их ограничения и обосновать гибридную модель, дополняющую стандартный курс практико-ориентированными средствами (интерактивной визуализацией данных с использованием библиотек Python), с первичной проверкой ее применимости в школьной апробации.

Методология. Нормативная база, регулирующая обучение программированию в российских школах, опирается на два ключевых документа: федеральные государственные образовательные стандарты (ФГОС) и примерную основную образовательную программу (ПООП). Эти документы задают общую

стратегию развития школьного образования и определяют содержание учебного предмета «Информатика» для базового и углубленного уровней.

В рамках ФГОС (п. 45.5.3, 45.5.4) программирование рассматривается как важный элемент подготовки школьников. Основное внимание уделяется развитию алгоритмического мышления, умению анализировать задачи и решать их с использованием компьютерных технологий. Также акцентируется необходимость формирования навыков разбиения сложных задач на подзадачи, их формализации и создания алгоритмов для реализации решений.

ПООП, дополняя требования ФГОС, углубляет их детализацию. В документе подчеркивается значимость изучения алгоритмов, программирования и методов обработки данных.

ФГОС (п. 45.5.3, 45.5.4) и ПООП (п. 2.1.14) детализируют ожидаемые результаты освоения курса «Информатика» для базового и углубленного уровней обучения. Они охватывают как предметные, так и метапредметные аспекты, направленные на развитие у школьников универсальных навыков работы с информацией и алгоритмами.

Базовый и углубленный уровни обучения программированию в рамках школьной программы существенно различаются по содержанию и ожидаемым результатам. Базовый уровень фокусируется на изучении простых алгоритмических конструкций, таких как линейные алгоритмы, ветвления и циклы, а также на освоении основ работы с одномерными массивами и строками. Углубленный уровень охватывает более сложные темы, включая двумерные массивы, рекурсию и основы динамического программирования, с акцентом на применение языка Python для решения сложных задач [5].

Содержание школьных программ по обучению программированию разрабатывается на основе учебников, которые соответствуют требованиям ФГОС и ПООП. Наиболее популярны пособия Л.Л. Босовой и А.Ю. Босовой, включенные в Федеральный перечень учебников (ФПУ)⁴ для изучения информатики на базовом и углубленном уровнях [6].

Для более детального сравнения содержания, используемых инструментов и планируемых результатов базового и углубленного уровней [6] можно обратиться к табл. 2.

Эффективное обучение программированию в школе опирается на определенные подходы, которые в педагогической науке рассматриваются как система концептуальных принципов, задающих логику и структуру образовательного процесса. Они могут классифицироваться в зависимости от целевых установок, методов реализации и содержания обучения. Как отмечает

⁴ Приказ Министерства просвещения РФ от 21.09.2022 № 858 «Об утверждении федерального перечня учебников, допущенных к использованию при реализации имеющих государственную аккредитацию образовательных программ начального общего, основного общего, среднего общего образования организациями, осуществляющими образовательную деятельность и установления предельного срока использования исключенных учебников» // Информационно-правовой портал Гарант.ру. URL: <https://www.garant.ru/products/ipo/prime/doc/405490287/> (дата обращения: 25.09.2024).

Е.Н. Степанов, в широком смысле подходы формируют общую стратегию педагогической деятельности, а в узком – определяют конкретные методики работы преподавателя [7].

Таблица 2

Сравнение уровней обучения программированию

Критерий	Базовый уровень	Углубленный уровень
Основные темы	Линейные алгоритмы; ветвления и циклы; одномерные массивы и строки	Двумерные массивы; рекурсия; основы динамического программирования
Язык программирования	Паскаль	Python (расширенное применение для решения сложных задач)
Фокус на навыках	Развитие алгоритмического мышления; умение разбивать задачи на подзадачи; создание простых алгоритмов	Свободное владение алгоритмами различной сложности; разработка и оптимизация сложных программных решений
Планируемые результаты	Освоение базовых алгоритмических конструкций; написание и отладка простых программ	Реализация сложных алгоритмов, включая сортировки и рекурсию; применение Python в прикладных задачах
Практическая направленность	Ограничена простыми заданиями и теоретическими аспектами	Углубленные проектные задачи; применение знаний для решения прикладных проблем

Источник: составлено К.М. Колосом.

Table 2

Comparison of programming education levels

Criterion	Basic level	Advanced level
Main Topics	Linear algorithms; branching and loops; one-dimensional arrays and strings	Two-dimensional arrays; recursion; basics of dynamic programming
Programming language	Pascal	Python (extensively applied for solving complex tasks)
Focus on skills	Development of algorithmic thinking; ability to break tasks into subtasks; creation of simple algorithms	Proficiency in complex algorithms; development and optimization of advanced software solutions
Planned outcomes	Mastery of basic algorithmic structures; writing and debugging simple programs	Implementation of complex algorithms, including sorting and recursion; application of Python in practical tasks
Practical orientation	Limited to simple exercises and theoretical aspects	Advanced project-based tasks; application of knowledge to real-world problem-solving

Source: compiled by Kirill M. Kolos.

Следует учитывать, что в педагогике существует проблема разграничения понятий «подход» и «метод». Как указывает Е.Н. Соловова, подход отражает ведущие принципы обучения, тогда как метод представляет собой совокупность конкретных шагов и приемов их реализации. Таким образом, выбор подхода предопределяет используемые методики и инструменты, обеспечивая целостность образовательного процесса [8; 9].

Рассмотрим ключевые подходы к обучению программированию, среди которых выделены традиционный, деятельностный, проблемный и личностно ориентированный. Оценивать их будем точки зрения их влияния на развитие алгоритмического мышления, практических навыков и мотивации учащихся.

Результаты и обсуждение. Рассмотрим и проанализируем разнообразные подходы к обучению программированию в современных российских школах.

Традиционный подход, преобладающий на базовом уровне, обеспечивает теоретическую основу, но страдает низкой мотивацией учащихся [3; 10]. Деятельностный и проблемный подходы способствуют развитию прикладных навыков и аналитического мышления, однако их внедрение требует значительных ресурсов и квалифицированных преподавателей [11–14]. Личностно ориентированный подход позволяет адаптировать процесс обучения к интересам учеников, однако чаще встречается во внеурочной деятельности, а не в основной программе [15; 16].

Для наглядного представления особенностей подходов к обучению программированию обратимся к табл. 3 [2; 3; 10–16].

Таблица 3

Сравнение подходов к обучению программированию

Подход	Описание	Применение	Преимущества	Ограничения
Традиционный	Передача теоретических знаний через объяснение алгоритмов, структуры данных и базовых конструкций	Базовый уровень обучения	Формирует теоретическую базу	Низкая вовлеченность учащихся, ограниченная мотивация
Деятельностный	Практическая деятельность: проекты, реальные задачи, современные языки программирования	Углубленный уровень, проектные задания	Развивает прикладные навыки, связь с реальными задачами	Требует ресурсов и высококвалифицированных учителей
Проблемный	Задания, требующие нестандартного мышления: оптимизация процессов, работа с данными	Мотивированные учащиеся, сложные задачи	Развивает аналитическое мышление и интерес к предмету	Сложность организации, высокая подготовка учащихся
Личностно ориентированный	Адаптация материалов к интересам и уровню подготовки учащихся	Внеурочная деятельность, кружки	Индивидуальный подход, развитие проектов по интересам	Ограниченное применение в стандартной программе

Источник: составлено К.М. Колосом.

Table 3

Comparison of approaches to teaching programming

Approach	Description	Application	Advantages	Limitations
Traditional	Transmission of theoretical knowledge through explanations of algorithms, data structures, and basic constructs	Basic level of education	Establishes a strong theoretical foundation	Low student engagement, limited motivation
Activity-based	Practical activities: projects, real-world tasks, modern programming languages	Advanced level, project-based assignments	Develops applied skills, connects learning to real tasks	Requires resources and highly qualified teachers
Problem-based	Tasks requiring non-standard thinking: process optimization, data analysis	Motivated students, complex problem-solving	Enhances analytical thinking and interest in the subject	Difficult to implement, requires high student preparedness
Student-centered	Adapting materials to students' interests and proficiency levels	Extracurricular activities, coding clubs	Individualized learning, development of interest-based projects	Limited applicability in standard curricula

Source: compiled by Kirill M. Kolos.

Программирование является обязательным разделом предмета «Информатика» и включено в школьную программу для учащихся 8–9 классов. Однако учебные материалы в значительной степени ориентированы на теорию и простые алгоритмические задачи. Несмотря на это, документация ФГОС и ПООП предоставляет основу для развития алгоритмического мышления, хотя, например, темы, связанные с интерактивной визуализацией данных или работой с большими данными, остаются за рамками учебной программы.

Что касается уровней обучения информатике, базовый уровень обеспечивает школьникам фундаментальные знания, достаточные для понимания основ программирования. Углубленный уровень, напротив, фокусируется на развитии более сложных навыков и готовит учащихся к решению прикладных задач. Однако ни один из уровней пока не охватывает темы, такие как интерактивная визуализация данных или работа с большими данными, которые могли бы существенно обогатить образовательный процесс.

Для дальнейшего совершенствования стандартов и программ необходимо усилить их практическую направленность, расширив содержание за счет изучения современных языков программирования, таких как Python. Этот язык предоставляет широкие возможности для решения разнообразных задач: от анализа больших данных и создания интерактивных визуализаций до работы с нейронными сетями и разработки графических интерфейсов. Однако в программах обучения зачастую не уделяется внимание этим аспектам, что ограничивает возможности учащихся для знакомства с передовыми технологиями. При этом существующие образовательные стандарты не запрещают педагогам использовать современные технологии в преподавании, что открывает возможности для адаптации учебного процесса к актуальным вызовам цифрового общества.

Дополнение курса информатики элементами, связанными с интерактивной визуализацией данных, машинным обучением и основами анализа больших данных, могло бы повысить интерес школьников к программированию. Даже краткое знакомство с этими инструментами не только разнообразит образовательный процесс, но и станет мотивационным фактором, способствующим вовлечению учащихся в углубленное изучение предмета, самостоятельную проектную деятельность и участие в олимпиадах.

Формирование алгоритмического мышления – одна из ключевых задач курса «Информатика», обозначенная в ФГОС и ПООП. Это умение позволяет учащимся анализировать задачи, формализовать их в виде последовательности шагов и эффективно применять алгоритмы на практике.

Развитие алгоритмического мышления осуществляется:

- на базовом уровне: через изучение линейных алгоритмов, ветвлений и циклов, разбиение задач на подзадачи и их реализацию в учебных заданиях;
- углубленном уровне: через освоение более сложных тем, таких как рекурсия, двумерные массивы и оптимизация алгоритмов [6].

Важным аспектом является способность школьников не только следовать готовым алгоритмам, но и создавать собственные для решения реальных задач. Это развивает логику, критическое мышление и аналитические способности, которые востребованы не только в программировании, но и в других дисциплинах.

Алгоритмическое мышление остается универсальным инструментом, помогающим школьникам адаптироваться к современным вызовам цифрового мира. Укрепление образовательного процесса за счет включения актуальных тем, таких как визуализация данных и использование Python в задачах анализа и машинного обучения, обеспечит учащимся лучшее понимание современных технологий и их практического применения.

Традиционный подход по-прежнему преобладает в обучении информатике и программированию в российской школе, особенно на базовом уровне, где важно обеспечить системное освоение основ [4]. Деятельностный и проблемный подходы находят применение на углубленных уровнях, способствуя развитию прикладных навыков и критического мышления. Личностно ориентированный подход остается редкостью, несмотря на его высокий потенциал [1; 17].

Анализ представленных подходов к обучению программированию в основной школе (табл. 3) показал, что каждый из них обладает как преимуществами, так и определенными ограничениями. Для повышения эффективности образовательного процесса в условиях стремительного технологического развития целесообразно использовать сочетание различных методик, что позволит объединить их сильные стороны и минимизировать недостатки.

Одним из ключевых аспектов модернизации является разработка гибридных подходов, включающих элементы традиционного, деятельностного, проблемного и личностно ориентированного обучения. Такая интеграция способствует повышению мотивации учащихся, формированию алгоритмического мышления и развитию практических навыков, необходимых для работы с цифровыми данными.

Определим ключевые критерии выбора наиболее перспективного подхода к обучению программированию:

- развитие алгоритмического мышления – наиболее эффективно достигается через деятельностный и проблемный подходы, которые предполагают активное участие учащихся в учебном процессе;
- повышение мотивации – достигается за счет интеграции проблемного и личностно ориентированного подходов, предполагающих использование задач, основанных на интересах школьников;
- подготовка к современным вызовам – углубленное изучение программирования с применением актуальных технологий формирует фундаментальные знания в области анализа и визуализации данных.

С учетом требований современной образовательной среды наиболее перспективными представляются деятельностный и проблемный подходы. Они

обеспечивают баланс между теоретической подготовкой и практическими навыками, что делает процесс обучения программированию более осмысленным и прикладным. Однако другие подходы также играют важную роль и остаются неотъемлемой частью образовательного процесса.

Традиционный подход необходим для систематизации и глубокого освоения теоретических основ, что позволяет формировать фундаментальные знания, без которых невозможно дальнейшее изучение программирования. Личностно ориентированный подход, в свою очередь, способствует развитию интереса учащихся, учитывает их индивидуальные особенности, повышает мотивацию к обучению, что немаловажно для успешного освоения сложных концепций.

Таким образом, для успешной подготовки школьников к вызовам цифрового общества важно расширить использование деятельностных и проблемных подходов, интегрируя их в повседневное обучение и делая процесс изучения программирования более мотивирующим и практико-ориентированным [2; 3]. Наиболее эффективным является не отказ от существующих методик, а их разумная интеграция, при которой традиционные элементы сочетаются с инновационными практиками, обеспечивая комплексное развитие учащихся.

Предложенный гибридный подход был апробирован в рамках преподавания учебного предмета «Информатика» среди учащихся 9 «А» класса Школы № 2097 г. Москвы. Это арт-класс, в учебном плане которого предусмотрено увеличенное количество часов информатики (2 часа в неделю). Однако стандартные методики обучения программированию вызывали у учащихся ограниченный интерес, что требовало внедрения новых подходов для повышения вовлеченности и мотивации.

В ходе апробации проведено четыре пробных занятия, направленных на изучение инструментов визуализации данных в языке программирования Python и применение их в проектной деятельности. Основной задачей занятий стало создание учащимися собственных визуализаций данных на основе сценариев, сгенерированных нейросетью. Генерируемые искусственным интеллектом данные и описания ситуаций служили основой для построения графиков, диаграмм и других способов представления информации. Итогом работы становился анализ визуализированных данных и формулирование выводов на их основе.

Работа носила ознакомительный характер, в связи с чем педагог оказывал учащимся поддержку на каждом этапе проектной деятельности. Несмотря на это, применение нейросетей для генерации данных и возможность их последующего визуального представления непосредственно в среде программирования без использования сторонних приложений вызвали у школьников высокий уровень заинтересованности. По результатам анкетирования, проведенного после завершения занятий, 78 % учащихся положительно оценили свой опыт работы с инструментами визуализации и выразили желание использовать их в дальнейшем изучении программирования.

Заключение. Несмотря на очевидные достижения, программы обучения программированию в российских школах все еще сталкиваются с ограничениями, которые снижают их прикладную ценность. Основной акцент по-прежнему сделан на теоретической части, что ограничивает развитие практических навыков. Задачи преимущественно сводятся к выполнению базовых алгоритмов (циклы, ветвления, массивы), реальные кейсы, требующие анализа данных или решения комплексных проблем, не учитываются.

В программах отсутствуют актуальные темы: работа с большими данными, интерактивная визуализация, машинное обучение и нейронные сети, что лишает школьников возможности освоить востребованные в современных отраслях инструменты. Интеграция этих направлений, особенно с использованием Python, может повысить интерес учащихся к углубленному изучению программирования.

Для повышения эффективности образовательных программ важно:

- усилить прикладной характер обучения, включая задачи, связанные с реальными данными из науки, экономики и технологий;
- развивать проектную деятельность, позволяя учащимся применять знания на практике, разрабатывать приложения или анализировать данные.

Будущее программ обучения программированию должно строиться на сбалансированном гибридном подходе, где теория подкрепляется практикой. Внедрение современных технологий и проектных задач позволит учащимся адаптироваться к вызовам цифрового мира и развить навыки, востребованные на рынке труда.

Список литературы

- [1] Босова Л.Л. Как учат программированию в XXI веке: отечественный и зарубежный опыт обучения программированию в школе // Информатика в школе. 2018. № 6(139). С. 3–11. EDN: XZOOJV
- [2] Гребнева Д.М. Обзор методических подходов к обучению программированию в школе // Научное обозрение. Педагогические науки. 2016. № 3. С. 13–27.
- [3] Садыкова О.В. Методические подходы к обучению программированию в школьном курсе информатики // Традиции и инновации в образовательном пространстве России, ХМАО-Югры, НВГУ : материалы IV Всероссийской науч.-практ. конф., Нижневартовск, 24 марта 2015 г. / отв. ред. М.В. Худжина. Нижневартовск : Нижневартовский государственный университет, 2015. С. 79–82. EDN: UFEQVD
- [4] Ярвилянин Е. В. Факторы, препятствующие развитию творческого мышления учащихся при традиционном подходе к обучению информатике // Качество. Инновации. Образование. 2008. № 8(39). С. 27–28. EDN: JTKGOL
- [5] Информатика. Примерные рабочие программы. 5–9 классы : учеб.-метод. пособие / сост. К.Л. Бутягина. 2-е изд., стереотип. М. : БИНОМ. Лаборатория знаний, 2018. 224 с.
- [6] Информатика. 7–9 классы. Методическое пособие / Л.Л. Босова, А.Ю. Босова, А.В. Анатольев, Н.А. Аквилянов. 3-е изд., перераб. М. : БИНОМ. Лаборатория знаний, 2019. 512 с.
- [7] Степанов Е.Н., Лузина Л.М. Педагогу о современных подходах и концепциях воспитания. 2-е изд., перераб. и доп. М. : Сфера, 2008. EDN: QWJCMX

- [8] *Марданова С.* Соотношение понятий «подход к обучению» и «метод обучения» // Русский язык в XXI веке: исследования молодых : материалы VII Междунар. науч. студ. конф., Нур-Султан, 6–7 февраля 2020 г. / отв. ред. Е.А. Журавлёва. Нур-Султан : Евразийский национальный университет им. Л.Н. Гумилёва, 2020. С. 65–67. EDN: BGTGEW
- [9] *Соловова Е.Н.* Методика обучения иностранным языкам : базовый курс лекций : пособие для студентов пед. вузов и учителей. М. : Просвещение, 2005. 239 с.
- [10] *Коменский Я.А.* Великая дидактика Я.А. Коменского. Санкт-Петербург : Симашко, 1875. XIV, 8, II, 282 с.
- [11] *Vihavainen A., Airaksinen J., Watson Ch.* A systematic review of approaches for teaching introductory programming and their influence on success // Proceedings of the tenth annual conference on international computing education research (ICER'14). New York : Association for Computing Machinery, 2014. P. 19–26. <https://doi.org/10.1145/2632320.2632349>
- [12] *Насырова Д.М., Очилова М.Р., Кадырова З.Б.* Проблемный метод обучения, как активный метод // Наука. Мысль: электронный периодический журнал. 2014. № 6.
- [13] *Давыдов В.В.* Виды обобщения в обучении: логико-психологические проблемы построения учебных предметов. 2-е изд. М. : Педагогическое общество России, 2000. 480 с.
- [14] *Махмутов М.И.* Проблемное обучение : основные вопросы теории. М. : Педагогика, 1975. 367 с.
- [15] *Сазонова Т.А.* Искусство давать урок // Актуальные проблемы педагогики и психологии. 2021. Т. 2. № 6. С. 26–30. EDN: LNYTOV
- [16] *Якиманская И.С.* Технология личностно-ориентированного образования. М. : Сентябрь, 2000. 176 с.
- [17] Актуальные проблемы методики обучения информатике в современной школе : материалы Междунар. науч.-практ. интернет-конф., Москва, 24–26 апреля 2018 г. / под ред. Л.Л. Босовой, Н.К. Нателаури. М. : МПГУ, 2018. 222 с. URL: <https://znanium.com/catalog/product/1020611> (дата обращения: 10.12.2025).

References

- [1] Bosova LL. How programming is taught in the 21st century: domestic and foreign experience of teaching programming in schools. *Informatics in School*. 2018;(6):3–11. (In Russ.) EDN: XZOOJV
- [2] Grebneva DM. Review of methodological approaches to teaching programming in school. *Scientific Review. Pedagogical Sciences*. 2016;(3):13–27. (In Russ.)
- [3] Sadykova OV. Methodological approaches to teaching programming in the school course of informatics. In: Xudzhina MV. (ed.) *Traditions and Innovations in the Educational Space of Russia, KhMAO-Ugra, NVSU: Proceedings of the IV All-Russian Scientific and Practical Conference, 24 March 2015, Nizhnevartovsk*. Nizhnevartovsk: Nizhnevartovsk State University Publ.; 2015. p. 79–82. (In Russ.) EDN: UFEQVD
- [4] Yarovilyanin EV. The factors hindering the development of creative thinking in students when using the traditional approach to teaching informatics. *Quality. Innovation. Education*. 2008;(8):27–28 (In Russ.) EDN: JTKGOL
- [5] Butyagina KL. *Informatics. Model Syllabi. Grades 5–9: Teaching-Methodical Manual*. 2nd ed. Moscow: BINOM. Knowledge Laboratory Publ.; 2018. 224 p. (In Russ.)
- [6] Bosova LL, Bosova AY, Anatolyev AV, Akvilyanov NA. *Informatics. Grades 7–9. Methodological Manual*. 3rd ed. Moscow: BINOM. Knowledge Laboratory Publ.; 2019. 512 p. (In Russ.)
- [7] Stepanov EN, Luzina LM. *Pedagogy of Modern Approaches and Concepts of Education*. 2nd ed. Moscow: Sfera Publ.; 2008. (In Russ.) EDN: QWJCMX

- [8] Mardanova S. The relationship between the concepts ‘approach to teaching’ and ‘teaching method’. In: Zhuravleva EA. (ed.) *Russian Language in the 21st Century: Research by Young Scientists: Proceedings of the VII International Scientific Student Conference, 6–7 February 2020, Nur-Sultan*. Nur-Sultan: L.N. Gumilyov Eurasian National University Publ.; 2020. p. 65–67. (In Russ.) EDN: BGTGEW
- [9] Solovova EN. *Methodology of Teaching Foreign Languages: Basic Course of Lectures: A Manual for Students of Pedagogical Universities and Teachers*. Moscow: Prosveshchenie Publ.; 2005. 239 p. (In Russ.)
- [10] Comenius JA. *The Great Didactic of John Amos Comenius*. Saint Petersburg: Simashko Publ.; 1875. XIV; 8, II, 282 p. (In Russ.)
- [11] Vihavainen A, Airaksinen J, Watson Ch. A systematic review of approaches for teaching introductory programming and their influence on success. In: *Proceedings of the Tenth Annual Conference on International Computing Education Research (ICER'14)*. Association for Computing Machinery, New York, USA; 2014. p. 19–26. <https://doi.org/10.1145/2632320.2632349>
- [12] Nasyrova DM, Ochilova MR, Kadyrova ZB. Problem-based learning as an active method. *A Science. Thought: Electronic Periodical Magazine*. 2014;(6). (In Russ.)
- [13] Davydov VV. *Types of Generalization in Instruction: Logical-Psychological Problems of Constructing School Subjects*. 2nd ed. Moscow: Pedagogical Society of Russia Publ.; 2000. 480 p. (In Russ.)
- [14] Makhmutov MI. *Problem-Based Learning: Main Issues of Theory*. Moscow: Pedagogika Publ.; 1975. 367 p. (In Russ.)
- [15] Sazonova TA. The art of giving a lesson. *Actual Problems of Pedagogy and Psychology*. 2021;2(6):26–30 (In Russ.) EDN: LNYTOV
- [16] Yakimanskaya IS. *Technology of Personality-Oriented Education*. Moscow: Sentyabr Publ.; 2000. 176 p. (In Russ.)
- [17] Bosova LL, Natelaury NK (ed.). *Current Problems of the Methodology of Teaching Computer Science in Modern School. Materials of the International Scientific-Practical Online Conference, 24–26 April 2018, Moscow*. Moscow: Moscow Pedagogical State University Publ.; 2018. (In Russ.) Available from: <https://znanium.com/catalog/product/1020611> (accessed: 10.12.2025).

Сведения об авторе:

Колос Кирилл Михайлович, аспирант, учитель, Школа № 2097, Российская Федерация, 125363, Москва, ул. Аэродромная, д. 9. ORCID: 0009-0001-2125-3918; SPIN-код: 3260-4235. E-mail: km.kolos@s2097.ru

Bio note:

Kirill M. Kolos, PhD Student, Teacher, School No. 2097, 9 Aerodromnaya St, Moscow, 125363, Russian Federation. ORCID: 0009-0001-2125-3918; SPIN-code: 3260-4235. E-mail: km.kolos@s2097.ru