

Научная статья

УДК 372.862

DOI: 10.24412/2072-9014-2025-474-116-129

ИЗ ОПЫТА ИСПОЛЬЗОВАНИЯ ТРАДИЦИОННЫХ ЗАДАЧ ПО ПРОГРАММИРОВАНИЮ ДЛЯ ОБУЧЕНИЯ ШКОЛЬНИКОВ ИНТЕРАКТИВНОЙ ВИЗУАЛИЗАЦИИ ДАННЫХ

Кирилл Михайлович Колос

Московский городской педагогический университет,
Москва, Россия
koloskm@mgpu.ru

Аннотация. Статья обосновывает интеграцию интерактивной визуализации данных (ИВД) в курс программирования 8–9-х классах. Предлагается критериальная схема оценки заданий (восемь критериев) и проводится сравнение авторских задач с материалами Л. Л. Босовой и И. Р. Высоцкого. Показано, что вероятностно-статистические задания изначально ближе к формату ИВД; задачи по информатике достигают сопоставимого уровня при условии их методической переработки. Сформулированы методы «перевода» заданий в формат, подходящий для обучения ИВД.

Ключевые слова: обучение программированию; язык программирования Python; интерактивная визуализация данных; информатика; вероятность и статистика.

Для цитирования: Колос К. М. Из опыта использования традиционных задач по программированию для обучения школьников интерактивной визуализации данных / К. М. Колос // Вестник МГПУ. Серия «Информатика и информатизация образования». 2025. № 4 (74). С. 116–129. <https://doi.org/10.24412/2072-9014-2025-474-116-129>

Original article

UDC 372.862

DOI: 10.24412/2072-9014-2025-474-116-129

**FROM THE EXPERIENCE
OF USING TRADITIONAL PROGRAMMING TASKS
TO TEACH SCHOOL STUDENTS
INTERACTIVE DATA VISUALIZATION**

Kirill M. Kolos

Moscow City University,
Moscow, Russia
koloskm@mgpu.ru

Abstract. The article substantiates the integration of interactive data visualization (IDV) into the programming course of grades 8–9. A criterion scheme for evaluating tasks (eight criteria) is proposed and the author's tasks are compared with the materials of L. L. Bosova and I. R. Vysotsky. It is shown that probabilistic and statistical tasks are initially closer to the IDV format; computer science tasks reach a comparable level provided they are methodically processed. The methods of “translating” assignments into a format suitable for teaching IVD are formulated.

Keywords: programming teaching; Python programming language; interactive data visualization; computer science; probability and statistics.

For citation: Kolos K. M. From the experience of using traditional programming tasks to teach school students interactive data visualization / K. M. Kolos // MCU Journal of Informatics and Informatization of Education. 2025. № 4 (74). P. 116–129. <https://doi.org/10.24412/2072-9014-2025-474-116-129>

Введение

Информатизация общего образования актуализирует роль работы с данными в курсе программирования 8–9-х классах. На практике фиксируется разрыв между написанием кода и пониманием свойств обрабатываемых данных, что снижает аналитическую составляющую учебной деятельности. Интерактивные визуальные средства позволяют сместить акцент к осмыслинию зависимостей, проверке гипотез и интерпретации результатов, опираясь на известную логику взаимодействия с данными «обзор – масштабирование и фильтрация – детали по требованию» [1], задающую рамки осмысленного взаимодействия с данными и поддерживающую учебные действия анализа и интерпретации.

В качестве теоретического основания принимается формулировка М. А. Гундиной: «Под интерактивной визуализацией будем понимать форму визуализации данных, при которой обеспечивается взаимодействие студента с компьютерной системой и обеспечивается возможность наблюдения за ответной реакцией

системы при изменении исходных параметров» [2, с. 126]. Исходя из этой позиции, под интерактивной визуализацией данных в обучении программированию мы будем понимать визуальные представления, непосредственно связанные с исполняемым кодом и параметрами вычислений. Учащийся изменяет входные данные или настройки программы и немедленно получает обновленный график, диаграмму либо анимацию, что обеспечивает проверку корректности алгоритмов и формирование интерпретаций [3].

Отечественные исследования рассматривают визуализацию и моделирование как значимые компоненты обновления содержания информатики. Работы А. В. Гриншкуна и его соавторов [4] показывают дидактический потенциал визуальных технологий в школе (в том числе AR/VR) и задают рамки методического использования наглядных средств в обучении информатике. В исследованиях О. Ю. Заславской [5] обсуждаются трансформация образовательных практик и роль интерактивных технологий как инструмента повышения наглядности и вовлеченности.

Несмотря на то что визуальные средства и инфографика уже используются в школьной практике, их применение носит в основном иллюстративный характер. Систематическое обучение приемам визуализации данных у учащихся 8–9-х классах фактически не выстроено, в том числе в связке с программированием. В результате ощущается дефицит заданий, где визуализация выступает средством мышления и исследования, а не оформлением ответа. При этом такой подход обладает высоким потенциалом для развития алгоритмического и статистического мышления.

Целью настоящей статьи является определение потенциала школьных задач по программированию в 8–9 классах для интеграции интерактивной визуализации данных.

Методы исследования

Данная работа нацелена на выстраивание методической основы интеграции интерактивной визуализации данных в обучение программированию в 8–9-х классах: предполагается разработать и обобщить критериальную схему оценивания учебных задач, сопоставить авторские материалы с заданиями из школьных учебников по информатике [6] и вероятности и статистике [7], а также предложить способы методически корректной переработки традиционных заданий под цели внедрения ИВД.

Структура исследования охватывает последовательные шаги:

- формирование операционального набора критериев (целевая роль визуализации, когнитивный уровень, межпредметность, открытость и вариативность, персонализация, дидактическая структурированность, требования к интерпретации результатов, ясность спецификации) с трехуровневой шкалой оценивания;
- переработка формулировок задач под единый шаблон;

- экспертное оценивание и сведение результатов в сопоставимую таблицу;
- содержательный сравнительный анализ блоков заданий с последующей разработкой модификаций для обучения ИВД в программировании.

Такой комплексный подход позволяет зафиксировать текущие тенденции, выявить типовые дефициты, а также сформулировать рекомендации по трансформации школьных задач в задания с ИВД, ориентированные на развитие визуально-аналитического и алгоритмического мышления.

Результаты исследования

Анализ задач требует прозрачных оснований оценивания. В целях дальнейшего сопоставления введем критериальную дифференциацию, которая задает, что именно считать качеством задания в контексте ИВД, и как это качество измерять. Для обеспечения сопоставимости будем использовать трехуровневую шкалу (0–2), которая позволит увидеть не только итоговую «сумму», но и общий профиль сильных и слабых сторон рассматриваемых заданий.

Описанные критерии со шкалой оценивания представлены в таблице 1.

Таблица 1

Критерии и уровни оценки заданий для интеграции интерактивной визуализации данных

№	Критерий	0	1	2
1	Целевая роль визуализации	Визуализация не требуется	Визуализация допустима как оформление	Визуализация является условием решения задачи
2	Когнитивный уровень (по Блуму ¹)	Воспроизведение по образцу	Применение/анализ (выбор формы, сопоставление)	Оценивание/создание (обоснование выбора, выводы, улучшения)
3	Межпредметность	Отсутствует	Тематический контекст без методов предмета	Предметные величины/статистики, корректная интерпретация
4	Открытость и вариативность	Один путь/ответ	Ограниченный выбор (тип графика, один параметр)	Свободная вариативность при соблюдении критериев качества

¹ Мурзагалиева А. Е. Учебные цели согласно таксономии Блума: сборник заданий и упражнений / А. Е. Мурзагалиева, Б. М. Утегенова. Астана: Центр педагогического мастерства АОО «Назарбаев Интеллектуальные школы», 2015. 54 с.

№	Критерий	0	1	2
5	Персонализация	Нейтральный контент	Выбор тематики/ данных из набора	Связь с интересами учащихся, работа с собственными/ актуальными данными
6	Дидактическая структурированность («лестница» поддержки)	Без подсказок	Минимальные подсказки (фрагменты, указание библиотеки)	Лестница поддержки: пример → подсказки → требования к объяснению
7	Требования к интерпретации результатов	Формальный ответ	Краткий комментарий	Аналитический вывод: что/почему/ ограничения, сравнение альтернатив
8	Ясность спецификации	Расплывчатые требования	Частичная определенность	Четкие критерии: корректность кода, уместность визуализации, качество интерпретации

Источник: составлено автором.

Предложенные критерии позволяют оценить потенциал задач для обучения ИВД: они фиксируют место визуализации в решении, требуемый уровень мыслительной деятельности, степень открытости и персонализации, а также качество методической поддержки и интерпретации. Соответственно, критериальная матрица служит не только инструментом ранжирования, но и основой для проектирования последующих модификаций заданий под цели внедрения ИВД.

В качестве эмпирической базы рассматривается выборка из трех блоков:

1. Две авторские задачи, изначально разработанные с целью обучения ИВД.
2. Две задачи из курса информатики Л. Л. Босовой [6].
3. Две задачи из школьного курса вероятности и статистики И. Р. Высоцкого [7].

Использование учебника Л. Л. Босовой обосновано его включением в федеральный перечень учебников², что отражает массовую школьную практику и обеспечивает репрезентативность «базовой» информатики.

Блок по вероятности и статистике включен, исходя из предметной специфики: такие задания изначально ориентированы на работу с выборками, частотами и характеристиками распределений, то есть естественно переходят

² Приказ Минпросвещения России от 26.06.2025 № 495 «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 установлении предельного срока использования исключенных учебников и разработанных в комплекте с ними учебных пособий» // Официальный интернет-портал

к визуализации (гистограммы, круговые диаграммы, графики) и интерпретации результатов. Сопоставление этих трех блоков позволяет выявить, какие типы задач наиболее подходят к внедрению в ИВД и где требуются минимальные, методически оправданные доработки.

Перейдем к рассмотрению отобранных задач.

Авторские задачи

Задача № 1. Дан фрагмент программы с циклом, который имитирует подбрасывание монеты N раз и считает число «орлов» и «решек». Сейчас программа только печатает значения в консоль. Доработайте код, чтобы построить столбчатую диаграмму распределения исходов и сделать краткий вывод (какой исход встречается чаще; насколько велика разница).

Входные данные (по умолчанию):

N = 200 (количество подбрасываний)

Фрагмент программы (рис. 1):

```
import random

N = 200
heads = 0
tails = 0

for i in range(N):
    if random.randint(0, 1) == 0:
        heads += 1
    else:
        tails += 1

print("Орлы:", heads, "Решки:", tails)
```

Рис. 1. Фрагмент программы для задачи № 1

Что требуется сделать:

1. Построить столбчатую диаграмму для значений heads и tails.
2. Подписать ось Y («Количество»), добавить заголовок.
3. Написать 1–2 предложения интерпретации (например: «частоты близки» / «наблюдаются перекос»).
4. Изменить код, чтобы число N вводилось вручную. Проверить результат на одних и тех же значениях N (будет ли меняться результат?) и разных N (например: 400, 800, 10).

5. Посчитайте вероятность того, что при подбрасывании монеты выпадет «орел», для всех разных значений и выведите результат на экран. Сравните вероятности разного количества подбрасываний. Есть ли разница? Какую тенденцию можно заметить?

Задача № 2. Дано количество элементов в первом множестве А. Известно, что второе множество в k раз больше первого, а третье на p процентов меньше первого.

Что требуется сделать:

1. Вычислить размеры А, В, С.
2. Выбрать подходящий тип диаграммы для сравнения (приоритет — круговая, но допускаются другие при обосновании).
3. Построить диаграмму с подписями.
4. Сделать вывод: какая доля у каждого множества и почему выбран именно этот тип визуализации.

Учитель может менять множества в соответствии с индивидуальными интересами учащихся. Например, если учащийся интересуется спортом, то множеством может выступать количество медалей (А — золотые, В — серебряные, С — бронзовые).

i. Входные данные (пример):

$$A = 120;$$

$$k = 1.5 (B = 1.5 * A);$$

$$p = 20 (C = A * 0.8).$$

Задачи из учебника по информатике Л. Л. Босовой

Задача № 3 (рис. 2).

17. Напишите программу, вычисляющую значение функции:

$$y = \begin{cases} 1 & \text{при } < 0, \\ 0 & \text{при } = 0, \\ 1 & \text{при } > 1. \end{cases}$$

Пример входных данных	Пример выходных данных
-5	y= -1
0	y= 0
5	y= 1

Рис. 2. Задача № 3 из учебника Л. Л. Босовой [6, с. 45]

Задача № 4 (рис. 3).

20. Объявлен набор в школьную баскетбольную команду. Известен рост каждого из N учеников, желающих попасть в эту команду. Составьте алгоритм подсчёта количества претендентов, имеющих шанс попасть в команду, если рост игрока команды должен быть не менее 170 см. Запишите программу на языке Python. Считайте рост претендента в команду случайным числом из диапазона от 150 до 200 см, а число претендентов $N = 50$.

Рис. 3. Задача № 4 из учебника Л. Л. Босовой [6, с. 87]

Задачи из учебника по вероятности и статистике И. Р. Высоцкого

Задача № 5 (рис. 4).

36 В течении четверти Ваня получил следующие оценки: по английскому языку — 4, 5, 5, 4, 3, 5, 4, 4, 3, 5, 5, 5; по математике — 4, 3, 5, 5, 4, 5, 5, 4.

а) Постройте круговые диаграммы распределения оценок по каждому из предметов. Сравните диаграммы.

б) Можно ли утверждать, что Ваня примерно одинаково учится по этим предметам?

Рис. 4. Задача № 5 из учебника И. Р. Высоцкого [7, с. 27]

Задача № 6 (рис. 5).

28 В таблице 17 указаны 6 лучших нападающих премьер-лиги чемпионата России по футболу сезона 2018—2019 гг. и место команды по итогам чемпионата.

Таблица 17. Лучшие нападающие

Игрок	Команда	Число голов	Место
Чалов Фёдор	ЦСКА (Москва)	15	4
Азмун Сердар	«Зенит» (Санкт-Петербург)	13	1
Классон Виктор	«Краснодар»	12	3
Дриусси Себастьян	«Зенит» (Санкт-Петербург)	11	1
Миранчук Антон	«Локомотив» (Москва)	11	2
Зе Луиш	«Спартак» (Москва)	10	5

- а) Постройте столбиковую диаграмму числа голов, забитых лучшими нападающими.
- б) Можно ли сказать, что среди нападающих есть явный лидер?
- в) Как вы думаете, есть ли связь между числом голов, забитых нападающими, и результатами их команд? Иными словами, можно ли утверждать, что чем больше голов забил игрок, тем выше место его команды в чемпионате?

Рис. 5. Задача № 6 из учебника И. Р. Высоцкого [7, с. 21]

Построим сравнительную матрицу отобранных задач по разработанным критериям (см. табл. 1), где цветом выделим количество баллов за критерии:

- 0 – белый
 - 1 – серый
 - 2 – черный

Результат представлен в таблице 2.

Таблица 2

Сравнительная критериальная матрица отобранных задач

Источник: составлено автором.

Сопоставление по критериальной схеме показало, что задачи вероятностно-статистического блока демонстрируют наивысшую готовность к интеграции интерактивной визуализации: работа с выборками и характеристиками распределений делает визуализацию инструментом анализа, а не оформлением, повышая уровень мыслительной деятельности и межпредметность при минимальной доработке. Задания по информатике также обладают значимым потенциалом при целенаправленном методическом переосмыслении: включение данных (реальных или псевдореальных), выбор корректной графической формы и требование интерпретации переводят базовые темы (циклы, ветвления, списки) в визуально-аналитический формат. Авторские задачи, изначально спроектированные в логике обучения ИВД, подтверждают применимость подхода: они устойчиво достигают высоких показателей по роли визуализации, структурированности и интерпретации, что служит «верхней планкой» для модификаций задач. В этой связи оправдано будет formalизовать процедуру перехода от традиционных формулировок к заданиям с ИВД.

Далее предлагается компактный набор методических приемов, которые помогают переформулировать задания для обучения ИВД.

Перевод традиционных школьных задач в формат ИВД опирается на следующую установку: визуализация служит инструментом мышления и проверки гипотез. Соответственно, формулировка задания должна связать три компонента — код, данные и интерпретацию. Ниже описаны приемы, выстраивающие эту связь и позволяющие довести задачи до требуемого уровня методической проработки.

• **Работа с данными.** Вводится источник данных — реальный (оценки, спорт, опрос) или псевдореальный, генерируемый программой. Определяются сущности и соответствие переменных элементам будущих диаграмм.

• **Визуализация как условие решения.** В тексте задания задается тип диаграммы, опорные элементы (пороговая линия, подписи, диапазоны осей) и роль визуального результата в принятии решения: по диаграмме требуется выполнить расчет, сравнение или классификацию.

• **Параметризация и варьирование.** Вводятся изменяемые параметры (порог, размер выборки, фильтр). Допускается ввод через `input()`, простую замену значения или добавления ползунка (слайдера). Ученик сравнивает не менее двух конфигураций и фиксирует изменения на графике и в показателях.

• **Обязательная интерпретация.** В ответ включается краткий аналитический абзац: что видно; почему так с учетом данных и параметров; каковы ограничения. Для сравнений — что изменилось и чем это объясняется.

• **Ступенчатая поддержка.** Сначала мини-пример построения нужного типа графика на микроданных, затем фрагменты кода с пропусками, затем самостоятельная часть с вариативными параметрами. Переход от образца к частичной опоре и далее к самостоятельности удерживает фокус на анализе, а не на синтаксисе.

• **Персонализация и межпредметность.** Разрешается менять контекст (спорт, учебные предметы, экология) при сохранении инвариантной структуры данных и единых требований к визуализации и интерпретации. Это повышает мотивацию и не нарушает сопоставимость результатов.

Рассмотрим пример переработки заданий из учебника по информатике и вероятности и статистике. Переформулируем по одному из заданий и оценим их переработанные версии по выделенным критериям.

Дополним и переработаем **задачу № 4** [6, с. 87].

Основной текст задачи остается неизменным. Дополнение к задаче:

Задание:

1. Сгенерируй данные роста N претендентов.
2. Построй горизонтальную столбчатую диаграмму по претендентам.
3. Проведи красную вертикальную линию на отметке 170 см как порог отбора.
4. Подсчитай и выведи, сколько претендентов выше или равны порогу.
5. Сделай вывод: какой процент прошел отбор? Что можно сказать о распределении роста?

Дополнительно:

- А) Позволь пользователю вводить порог с клавиатуры.
- Б) Покажи на графике подпись с числом прошедших.
- В) Напиши программу, в которой вводятся ученики твоего класса и их рост. Сколько человек прошли в команду?

Фрагмент кода для заполнения (см. рис. 6).

Далее переработаем **задачу № 5** [7, с. 27] из учебника по вероятности и статистике. Основная формулировка остается неизменной. Дополним задачу.

```

import random
import matplotlib.pyplot as plt

# Количество претендентов и порог
N = ___ # укажи количество
threshold = ___ # укажи порог

# Генерация роста от 150 до 200 см
heights = [random.randint(___, ___) for i in range(N)]

# Подсчёт прошедших
passed = 0
for h in heights:
    if h ___ threshold:
        passed += 1

# Построение диаграммы
plt.barh(range(1, N + 1), heights, color='skyblue', edgecolor='black')

# Вертикальная линия-порог
plt.axvline(пороговая переменная, color='red', linestyle='--', linewidth=2)

# Подписи и оформление
plt.title(f'Отбор по росту: прошли {___}/{___} ({___/___*100:.1f}%)')
plt.xlabel('Рост, см')
plt.ylabel('Номер претендента')
plt.xlim(___, ___) # границы диаграммы

# Подписи оси Y
plt.yticks(range(1, N + 1))

plt.show()

```

Рис. 6. Фрагмент кода для переработанной задачи № 4

Задание:

1. Постройте круговые диаграммы распределения оценок по каждому предмету. Используйте Python и библиотеку matplotlib. Подпишите диаграммы, добавьте легенду.
2. Сравните диаграммы. Сделайте вывод: по какому предмету Ваня учится стабильнее?
3. Вычислите статистические показатели по каждому предмету:
 - *средний балл* — сумма всех оценок, поделенная на их количество;
 - *медиана* — «средняя» оценка в отсортированном списке;
 - *стандартное отклонение* — показывает, насколько сильно оценки отличаются от среднего (чем меньше, тем стабильнее).

(Подсказка: можно использовать функции mean(), median(), stdev() из модуля statistics для проверки себя. Но изначальный подсчет делайте по алгоритму.)

4. Сделайте текстовый вывод: сравните результаты и диаграммы. Можно ли сказать, что Ваня учится по обоим предметам примерно одинаково?
5. Разрешите пользователю (себе) ввести собственные оценки по любым школьным предметам (через input()) и визуализируйте их.

Дополнительно:

- А) Постройте гистограмму вместо круговой диаграммы и сравните, какая форма нагляднее.

Б) Добавьте вычисление количества «пятерок» и их долю в процентах для каждого предмета.

Построим сравнительную матрицу по переработанным задачам и проверим, как изменился результат (табл. 3).

Таблица 3

Сравнительная критериальная матрица переработанных задач по информатике и вероятности и статистике

Критерий \\ Задача	1	2	3	4	5	6	7	8
№ 4 (до)								
№ 4 (после)								
№ 5 (до)								
№ 5 (после)								

Источник: составлено автором.

Сравнение «до/после» демонстрирует устойчивый прирост качества: визуализация становится условием решения, появляются требования к интерпретации, растет открытость за счет параметров и допускается персонализация без потери сопоставимости. Наибольший эффект преобразования наблюдается у задач по информатике; вероятностно-статистические задания изначально ближе к целевому состоянию.

Следует отметить, что в рамках обучения ИВД не предполагается, что каждое задание должно стремиться к максимальным показателям по всем критериям. Дидактически обосновано распределять задачи по этапам: на стартовом уровне — простые постановки для освоения техники построения графиков и базовой параметризации; на поздних — проектные задания с обязательной интерпретацией и вариативностью. Описанные методы задают целевую модель, к которой целесообразно выводить задания по мере роста подготовленности. Критериальная схема в этом случае выполняет не только оценочную, но и навигационную функцию, позволяя выстраивать траекторию от тренировочных к исследовательским задачам и обеспечивать переход к визуально-аналитическому мышлению.

Заключение

Разработанная критериальная схема позволила сопоставимо оценить задания из трех источников и выделить типологические различия: вероятностно-

статистические задачи изначально ближе к целевому формату ИВД, тогда как задачи по информатике достигают сопоставимого уровня при целенаправленной переработке. Представленные примеры модификаций показали воспроизводимый прирост качества по ключевым параметрам — роли визуализации, параметризации и интерпретации — без потери дидактической ясности.

Практический результат состоит в наборе методов «перевода» традиционных постановок в формат, подходящий для обучения ИВД, и в навигационной функции критериальной схемы, позволяющей выстраивать траекторию от тренировочных к исследовательским заданиям. Ограничения исследования связаны с объемом выборки и экспертным характером оценивания; дальнейшая работа будет направлена на расширение выборки заданий, уточнение весов критериев и более объективную оценку по критериям за счет увеличения количества экспертов.

Список источников

1. Shneiderman B. The eyes have it: a task by data type taxonomy for information visualizations / B. Shneiderman // Proceedings 1996 IEEE Symposium on Visual Languages. Boulder (CO), 1996. Р. 336–343.
2. Гундина М. А. Интерактивная визуализация задач комбинаторики в системе Wolfram Mathematica при управляемой самостоятельной работе студентов / М. А. Гундина, Н. А. Кондратьева, О. В. Юхновская // Информатизация образования и методика электронного обучения: цифровые технологии в образовании: материалы V Междунар. науч. конф. (Красноярск, 21–24 сентября 2021 г.). Красноярск: Сибирский федеральный университет, 2021. Ч. 1. С. 125–129.
3. Колос К. М. Интеграция интерактивной визуализации данных в процесс обучения программированию на языке Python в основной школе / К. М. Колос // Трансформация механико-математического и ИТ-образования в условиях цифровизации: материалы II Междунар. науч.-практ. конф. (Минск, 22–24 апреля 2025 г.). Минск: Белорусский государственный университет, 2025. С. 226–230.
4. Гриншкун А. В. Обучение и использование технологии дополненной реальности в курсе информатики основной школы / А. В. Гриншкун, И. В. Левченко. М.: Образование и Информатика, 2022. 140 с.
5. Заславская О. Ю. Использование инструментов визуализации в процессе обучения студентов педагогического вуза / О. Ю. Заславская // Вестник МГПУ. Серия «Информатика и информатизация образования». 2019. № 3 (49). С. 17–23.
6. Босова Л. Л. Информатика. 8–9 классы. Начала программирования на языке Python. Дополнительные главы к учебникам / Л. Л. Босова, Н. А. Аквилянов, И. О. Кочергин [и др.]. М.: БИНОМ. Лаборатория знаний, 2020. 96 с.
7. Высоцкий И. Р. Математика. Вероятность и статистика: 7–9-е классы: базовый уровень: учебник: в 2 ч. / И. Р. Высоцкий, И. В. Ященко; под ред. И. В. Ященко. М.: Просвещение, 2025. 176 с.

References

1. Shneiderman B. The eyes have it: a task by data type taxonomy for information visualizations / B. Shneiderman // Proceedings 1996 IEEE Symposium on Visual Languages. Boulder (CO), 1996. Р. 336–343.

2. Gundina M. A. Interactive visualization of combinatorics problems in Wolfram Mathematica in guided independent work of students / M. A. Gundina, N. A. Kondratyeva, O. V. Yukhnovskaya // Informatization of Education and Methods of E-Learning: Digital Technologies in Education: Proceedings of the V International Scientific Conference. (Krasnoyarsk, September 21–24, 2021). Krasnoyarsk: Siberian Federal University, 2021. Part 1. P. 125–129.
3. Kolos K. M. Integration of interactive data visualization into teaching programming in Python in lower secondary school / K. M. Kolos // Transformation of Mechanical-Mathematical and IT Education under Digitalization : Proceedings of the II International Scientific and Practical Conference (Minsk, April 22–24 2025). Minsk: Belarusian State University, 2025. P. 226–230.
4. Grinshkun A. V. Teaching and using augmented reality technology in the lower-secondary informatics course / A. V. Grinshkun, I. V. Levchenko. M.: Education and Computer Science, 2022. 140 p.
5. Zaslavskaya O. Yu. The use of visualization tools in the training of pedagogical university students / O. Yu. Zaslavskaya // MCU Journal of Informatics and Informatization of Education. 2019. № 3 (49). P. 17–23.
6. Bosova L. L. Informatics. Grades 8–9. Fundamentals of programming in Python. Additional chapters to the textbooks / L. L. Bosova, N. A. Akvilyanov, I. O. Kochergin [et al.]. M.: BINOM. Laboratory of Knowledge, 2020. 96 p.
7. Vysotsky I. R. Mathematics. Probability and Statistics: Grades 7–9: basic level: textbook: in 2 p. / I. R. Vysotsky, I. V. Yashchenko; ed. I. V. Yashchenko. M.: Prosvetshchenie, 2025. 176 p.

Статья поступила в редакцию: 12.08.2025;
одобрена после рецензирования: 22.09.2025;
принята к публикации: 01.10.2025.

The article was submitted: 12.08.2025;
approved after reviewing: 22.09.2025;
accepted for publication: 01.10.2025.

Информация об авторе / Information about the author:

Кирилл Михайлович Колос — аспирант, Институт цифрового образования, Московский городской педагогический университет, Москва, Россия.

Kirill M. Kolos — Postgraduate Student, Institute of Digital Education, Moscow City University, Moscow, Russia.

koloskm@mgpu.ru, <https://orcid.org/0009-0001-2125-3918>