Никитин Александр Николаевич
учитель
МБОУ «Буртасинская СОШ»
д. Буртасы, Чувашская Республика

DOI 10.21661/r-600300

PYTHON КАК ИНСТРУМЕНТ РАННЕЙ ПРОФЕССИОНАЛИЗАЦИИ: ОБУЧЕНИЕ ПРОГРАММИРОВАНИЮ В 5–7 КЛАССАХ

Аннотация: *в статье рассматривается потенциал языка Python как ресурса повышения качества образования на этапе 5–7 классов. Обосновывается целесообразность перехода от блочных сред к текстовому программированию с опорой на читабельность синтаксиса Python и визуальную библиотеку Turtle. Приводятся аргументы в пользу раннего формирования алгоритмического мышления и профилактики когнитивных перегрузок при последующем изучении информатики. Описывается структура курса, включающая работу с графикой, базовыми конструкциями и мини-проектами, а также рассматриваются риски и условия эффективной реализации.*

Ключевые слова: *Python, программирование в основной школе, алгоритмическое мышление, библиотека Turtle, инновационные технологии, качество образования.*

Современный этап обновления образовательного процесса характеризуется поиском технологий, способных не просто транслировать знания, но и формировать у обучающихся инструментальные компетенции, востребованные в цифровой экономике. В этом контексте обучение программированию в 5–7 классах перестаёт быть прерогативой углублённых курсов и всё чаще рассматривается как ресурс общего развития мышления. Выбор языка Python для этой возрастной группы не случаен: его синтаксис близок к естественному языку, что позволяет снизить порог входа и сосредоточить усилия учащихся на логике алгоритма, а не на борьбе с формальностями записи кода.

Content is licensed under the Creative Commons Attribution 4.0 license (CC-BY 4.0)

Одним из ключевых аргументов в пользу введения Python на раннем этапе является преемственность. Анализ экзаменационных материалов и содержания старшей школы показывает устойчивую ориентацию на данный язык. Отказ от псевдокодов и переход к реальному языку программирования уже в среднем звене позволяет выстроить последовательную траекторию: от простейших скриптов к решению прикладных задач, что в перспективе снимает необходимость переучивания и снижает когнитивную нагрузку при подготовке к итоговой аттестации. Кроме того, исследования подтверждают, что целенаправленное использование Python способствует развитию вычислительного мышления у школьников среднего звена, причём эффект сохраняется при контроле исходного уровня подготовки.

Методически наиболее эффективным инструментом адаптации пятиклассников и шестиклассников к текстовому программированию выступает стандартная библиотека Turtle. Визуализация процесса выполнения алгоритма – «черепашка», оставляющая след на экране, – превращает абстрактные циклы и ветвления в наглядный результат. Ученик видит, как несколько строк кода порождают геометрический орнамент, и это создаёт мгновенную обратную связь, столь важную для поддержания мотивации в младшем подростковом возрасте. На смену блочным средам приходит среда, где ошибка не блокирует выполнение, а становится предметом анализа: исправление синтаксической ошибки или логического несоответствия воспринимается как естественный этап работы, а не как неудача.

Содержательная структура курса для 5–7 классов может быть выстроена по принципу нарастания сложности с сохранением практико-ориентированной направленности. На первом этапе акцент делается на вывод данных, работу с переменными и ввод с клавиатуры. Уже здесь возможно создание утилитарных скриптов, понятных ребёнку: калькулятор индекса массы тела, генератор паролей, простой текстовый квест с ветвлениями. На втором этапе вводится Turtle как средство визуализации циклических алгоритмов. Рисование правильных многоугольников, снежинок и орнаментов позволяет «прочувствовать» работу

Содержимое доступно по лицензии Creative Commons Attribution 4.0 license (CC-BY 4.0)

цикла `for` и функции `range`. Третий этап предполагает переход к мини-проектам: создание интерактивной открытки, игры «Угадай число» или программы, обрабатывающей результаты небольшого опроса. Проектная форма работы не только закрепляет предметные навыки, но и формирует регулятивные универсальные учебные действия – планирование, контроль и коррекцию собственной деятельности.

Однако введение программирования в 5–7 классах сопряжено с рядом рисков, которые необходимо учитывать при проектировании образовательного процесса. Первый риск связан с разным темпом освоения материала: часть учащихся уже имеет опыт работы в Scratch или аналогичных средах, тогда как другие впервые сталкиваются с алгоритмизацией. Решением может служить дифференциация заданий по уровню сложности при сохранении общего тематического ядра. Второй риск – формализация обучения, сведение курса к заучиванию синтаксиса в ущерб развитию мышления. Преодоление этого риска требует смещения акцента с репродуктивных упражнений на задачи, допускающие несколько решений и предполагающие обсуждение выбранного способа. Третий риск – недостаточная материально-техническая база или неготовность педагога к работе с текстовыми средами разработки. В этом случае полезным оказывается использование онлайн-редакторов, не требующих установки, и опора на методические материалы, разработанные профессиональным сообществом.

С точки зрения повышения качества образования предлагаемая модель ценна не столько фактом освоения конкретного языка, сколько формируемыми метапредметными результатами. Учащийся учится разбивать сложную задачу на подзадачи, проверять граничные случаи, читать и отлаживать чужой код, аргументировать выбор решения. Эти компетенции выходят за рамки информатики и находят применение в проектной деятельности по другим предметам. Кроме того, ранний опыт успешного программирования способствует профессиональному самоопределению в сфере информационных технологий, что отвечает задачам подготовки инженерных кадров.

Content is licensed under the Creative Commons Attribution 4.0 license (CC-BY 4.0)

Таким образом, обучение Python в 5–7 классах следует рассматривать не как самоцель, а как ресурс обновления содержания и технологий образования. Сочетание доступного синтаксиса, визуальной обратной связи и проектной деятельности создаёт условия для развития алгоритмического мышления, которое является одним из ключевых результатов современного общего образования. Инновационность данного подхода заключается в отказе от отсроченного изучения программирования и в переносе акцента с формального усвоения языка на решение практических задач, посильных и интересных подростку.

Список литературы

1. Муратов, Э. Методические подходы к обучению Python для развития алгоритмического мышления в 5–7 классах / Э. Муратов // Journal of Uzbekistan's Development and Research. – 2025. – URL: https://zenodo.org/records/15587324 (дата обращения: 18.11.2026). – Текст : электронный.

2. Ибляминов, Р. Н. Программирование на Python в школе: от простых скриптов к решению задач ЕГЭ / Р. Н. Ибляминов // Учительский журнал. – URL: https://www.teacherjournal.ru/categories/6/articles/18644 (дата обращения: 18.11.2026). – Текст : электронный.

3. Sun, L. Improving Computational Thinking for Middle School Students Through Python Programming: Interaction Effect Analysis of Grade Level and Gender / L. Sun, Ju. Liu // Journal of Educational Computing Research. – 2025. – Vol. 63, No. 4. – P. 902–929. – DOI 10.1177/07356331251332751. – EDN WRXMED.

4. Черевкова, О. А. Мастер-класс «Программирование на Python. Нача-ло» / О. А. Черевкова // Видеоуроки в интернет. – URL: https://videouroki.net/razrabotki/mastier-klass-proghrammirovaniie-na-python.html (дата обращения: 18.11.2026). – Текст : электронный.

Содержимое доступно по лицензии Creative Commons Attribution 4.0 license (CC-BY 4.0)