

РАЗРАБОТКА СПЕЦИАЛИЗИРОВАННОЙ СРЕДЫ ПРОГРАММИРОВАНИЯ ДЛЯ ИЗУЧЕНИЯ PYTHON В УЧРЕЖДЕНИЯХ ОБЩЕГО СРЕДНЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Корнеев П.А., Соколова А.С., Черненко Д.В.

УО «Витебский государственный технологический университет», г. Витебск. Республика Беларусь (E-mail: chernenko203509@gmail.com)

На первый взгляд, выбор интегрированной среды разработки (IDE) для школьного курса программирования кажется сугубо технической задачей, сводящейся к поиску бесплатного редактора кода с подсветкой синтаксиса. Однако в контексте общего среднего образования этот выбор трансформируется в фундаментальную научно-педагогическую проблему, от которой напрямую зависит эффективность педагогического процесса, мотивация учащихся и формирование их вычислительного мышления. Правильная IDE становится не просто программой на компьютере, а дидактическим посредником [1] – инструментом, способным адаптировать сложные, абстрактные концепции программирования для восприятия школьников, делая их осязаемыми, а сложные процессы – понятными.

Актуальность настоящего исследования обусловлена знаковым переходом системы общего среднего образования Беларуси на язык Python в 2024/2025 учебном году [3-5]. Данный шаг, отражающий глобальные тренды, требует не простой замены синтаксиса Pascal на Python [1, 5], а системной методической и инструментальной трансформации всего учебного процесса. Основная научная проблема, как будет показано далее, заключается в глубоком противоречии между универсальностью профессиональных сред разработки и необходимостью создания специализированного инструмента, адаптирующего сложный мир программирования для восприятия школьников.

Когнитивная нагрузка ученика, впервые сталкивающегося с программированием, чрезвычайно высока. Ему необходимо одновременно удерживать в голове и усваивать:

1. Синтаксис языка – правила написания команд.
2. Логику алгоритма – последовательность действий и структуры управления.
3. Модель состояния программы – что хранится в переменных, как меняются данные в памяти в данный момент.
4. Механику взаимодействия со средой – как запустить код, где увидеть результат, как интерпретировать ошибку.

Программирование изначально абстрактно: переменная, область видимости, поток выполнения, стек вызовов – эти концепции существуют «внутри» машины и без специальных средств остаются для ученика чисто умозрительной теорией. Именно здесь образовательная IDE раскрывает свой главный методический потенциал, выполняя функцию не просто редактора, а научно-обоснованного инструмента визуализации.

Задача школы – не изолировать ученика в упрощенной «песочнице» с чрезмерно упрощенным инструментом, который он больше нигде не встретит, а подготовить его к дальнейшему образованию и профессиональной деятельности. Таким образом, стратегический выбор IDE – это решение, направленное на построение «моста» от учебной деятельности к профессиональной, минимизирующее будущий стресс от столкновения с реальными инструментами разработки и формирующее устойчивые компетенции.

Сравнительный анализ языков программирования в педагогическом контексте показывает, что строгость Pascal, игравшего важную историческую роль в обучении структурному программированию, в современных реалиях уступает место гибкости и

практической значимости Python. Python предлагает образованию принципиально иные возможности: чистый и читаемый синтаксис, высокую практическую значимость, огромное количество бесплатных ресурсов и библиотек. Однако, как показала практика, его богатый профессиональный инструментарий оказался малопригоден для прямого использования в классе, особенно при изучении ключевой темы – управления исполнителем «Робот». Существующие библиотеки, будучи технически корректными, создают неоправданные когнитивные барьеры, затрудняя достижение главных образовательных целей.

В ответ на выявленное противоречие авторами разработан прототип специализированной учебной среды, ядро функциональности которой ориентировано на решение актуальных задач белорусской школы. Текущий прототип уже содержит необходимый для ведения учебного процесса функционал, реализованный с учетом дидактических требований.

Базовые операции и снижение технических барьеров. Среда реализует базовые операции (открытие, редактирование, сохранение файлов с расширением .py) в интуитивно-понятной манере (Ctrl+O, Ctrl+S, Ctrl+N), что позволяет ученикам осваивать общепринятые паттерны работы с документами. Цветовое кодирование синтаксиса (ключевые слова, строки, комментарии) обеспечивает визуальное структурирование кода (рис. 1).

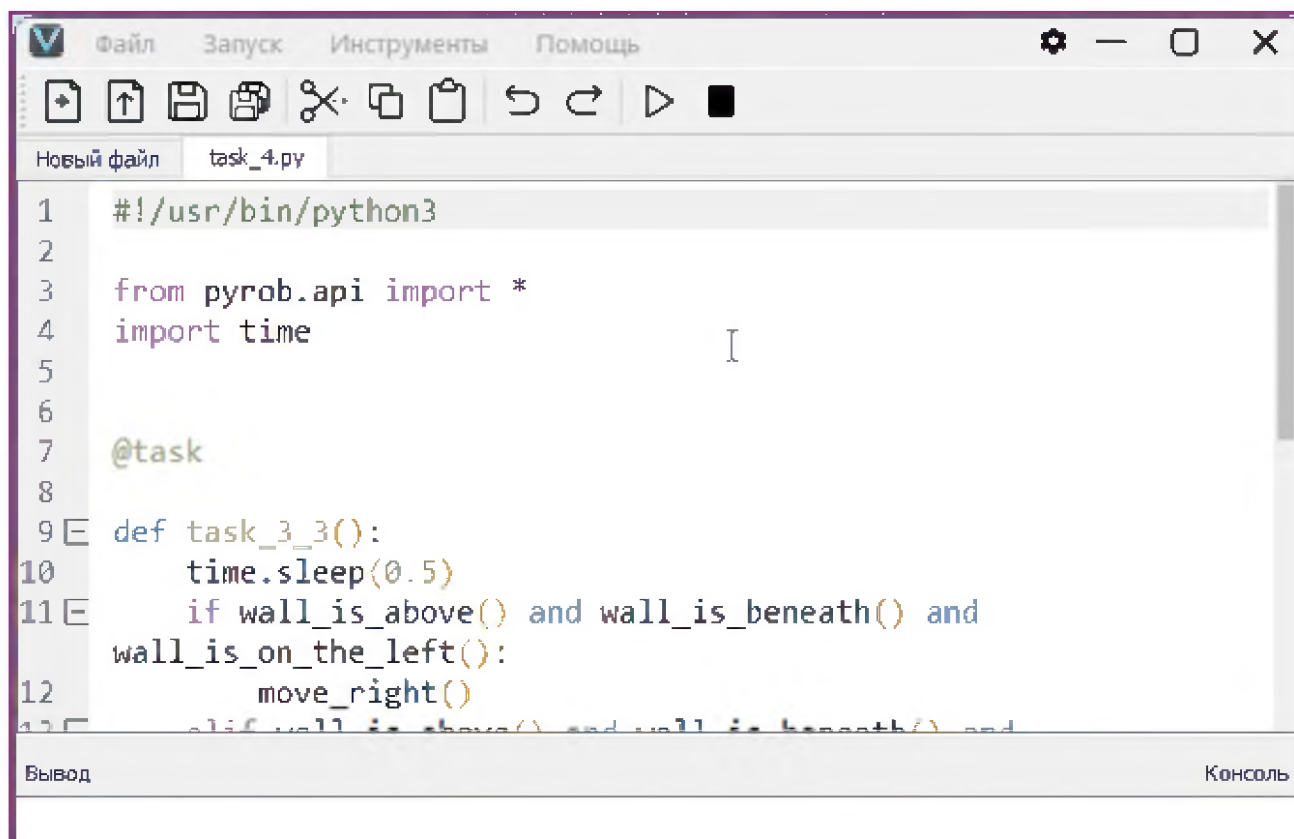


Рис. 1 Цветное выделение кода

Педагогическая ценность встроенного интерпретатора заключается в его «бесшовной» интеграции: он не требует установки Python на систему, работает «из коробки» и представляет собой гарантированно стабильную, изолированную среду выполнения. Это позволяет избежать демотивирующей ситуации, когда результат выполнения команды `print("Привет, мир!")` теряется в мгновенно закрывающемся окне командной строки, заменяя её понятным и стабильным выводом в консоль самой IDE.

Интеллектуальная помощь и диагностика ошибок. Ключевым научно-методическим результатом является реализация модуля справки и диагностики ошибок. Встроенная

локальная документация (рис. 2) содержит русскоязычные примеры, адаптированные для школьников, и позволяет осуществлять поиск по ошибкам с объяснением на русском языке. Это не просто справочная система, а инструмент формирования навыка самостоятельной диагностики и отладки кода. Функция автозаполнения кода (интеллектуальные подсказки) (рис. 3) снижает когнитивную нагрузку, позволяя ученику концентрироваться на логике алгоритма, а не на механическом запоминании названий методов.

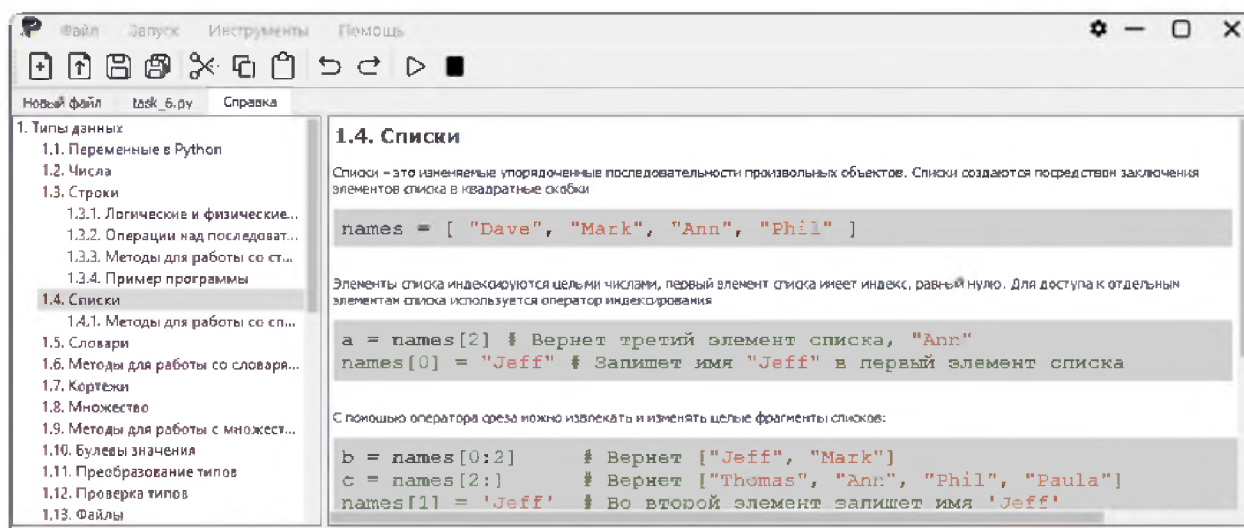


Рис. 2 Краткая справка по python

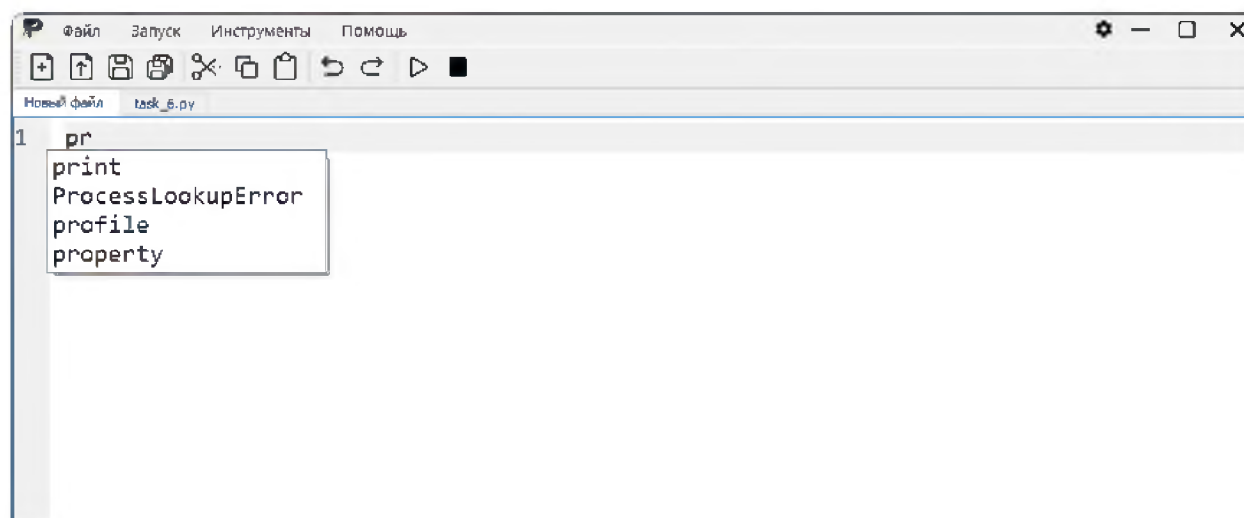


Рис. 3 Интеллектуальная помощь: автозаполнение кода

Реализация исполнителя «Робот» как дидактической лаборатории. Особое место в разработке занимает реализация исполнителя «Робот» (Рис. 4). В отличие от технически сложных библиотек, реализация в разработанной IDE превращает Робота из абстрактного объекта в «живую лабораторию алгоритмического мышления». Визуальная модель вычислительного процесса позволяет ученику непосредственно наблюдать связь между кодом и его исполнением в реальном времени. Это превращает отладку в естественную часть исследования, а не в дополнительную техническую сложность. Исполнитель становится центральным элементом интерактивного учебного пространства, где алгоритмическое мышление формируется через непосредственное манипулирование объектами, а творчество и исследование поощряются встроенными инструментами.

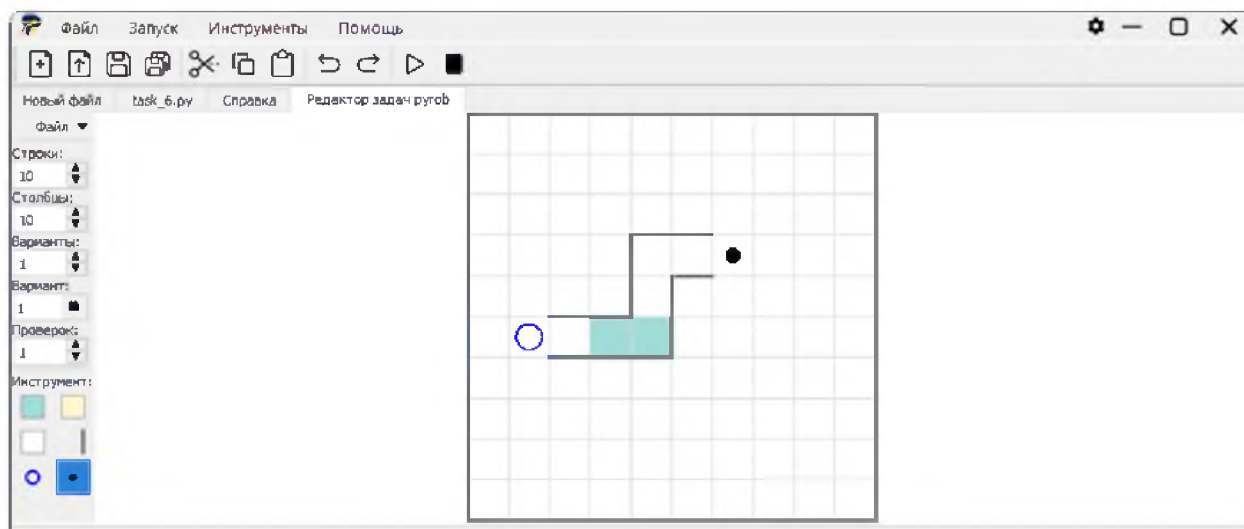


Рис. 4 Изучение исполнителя "Робот"

Заключение

Переход на Python в школьном образовании Беларуси – это прогрессивный и своевременный шаг, успех которого напрямую зависит от адекватности инструментария. Текущая реализация учебной IDE решает ключевую научно-методическую проблему: она снижает когнитивные барьеры, визуализирует абстрактные понятия и предоставляет учителю выверенные дидактические средства. Создание такой среды – это стратегическая инвестиция в качество IT-образования, которая окупится повышенной мотивацией учеников, снижением нагрузки на учителей и формированием более глубоких компетенций у будущих специалистов. Время, которое сегодня тратится на борьбу с техническими сложностями, будет направлено на главное – обучение мыслить алгоритмически и решать реальные задачи.

Научная и практическая значимость работы подтверждена успешным прохождением университетского отборочного тура республиканского конкурса «100 идей для Беларуси», а также наличием акта о внедрении результатов НИОК(Т)Р в государственном учреждении образования «Средняя школа №2 г. Дубровно имени Ю. Смирнова» (от 25.11.25). Дальнейшие направления исследований будут связаны с расширением инструментов визуализации состояний программы и развитием адаптивных модулей обучения.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Сорокина, Н. А. Python как основной язык программирования в средней школе / Н. А. Сорокина // Молодой ученый. – 2019. – № 5(243). – С. 15-16.
2. Котов, В. М. Информатика: учебное пособие для 6-го класса учреждений образования, реализующих образовательные программы общего среднего образования, с русским языком обучения и воспитания / Котов В. М., Макарова Н. П., Лапо А. И., Войтехович Е. Н. – Минск: Нар. асвета, 2024. – 183 с.
3. Котов, В. М. Информатика: учебное пособие для 7-го класса учреждений образования, реализующих образовательные программы общего среднего образования, с русским языком обучения и воспитания / В. М. Котов, А. И. Лапо. – 2-е издание, пересмотренное и дополненное. – Минск: Адукацыя і выхаванне, 2025. – 192 с.
4. Учебная программа по учебному предмету «Информатика» для VI–IX классов учреждений образования, реализующих образовательные программы общего среднего образования, с русским языком обучения и воспитания. Постановление Министерства образования Республики Беларусь 29.07.2025 № 132.

5. Обучение школьников программированию в рамках предмета «Информатика»: проблемы и возможные решения / О. А. Козлов, И. В. Барышева, Е. В. Малкина, Н. В. Шестакова // Информатика в школе. – 2023. – № 5(184). – С. 67-73.