

УДК 378.147:004.42

UDC 378.147:004.42

**РАЗВИТИЕ ВЫЧИСЛИТЕЛЬНОЙ
КОМПЕТЕНТНОСТИ СТУДЕНТОВ
СРЕДСТВАМИ ЯЗЫКА
PYTHON****DEVELOPING
STUDENTS' COMPUTATIONAL
COMPETENCE USING
PYTHON****Т. Г. Алейникова,**

кандидат физико-математических наук,
доцент кафедры прикладного и системного
программирования факультета
математики и информационных
технологий Витебского государственного
университета имени П. М. Машерова
ORCID: <https://orcid.org/0000-0003-2674-5599>;

T. Aleinikova,

Associate Professor, Department
of Applied and System Programming,
Faculty of Mathematics and Information
Technology, Vitebsk State University
named after P. M. Masherov, Candidate
of Physical and Mathematical Sciences
ORCID: <https://orcid.org/0000-0003-2674-5599>;

А. И. Шербаф,

кандидат физико-математических наук,
доцент кафедры информатики
и методики преподавания информатики
Белорусского государственного
педагогического университета
имени Максима Танка
ORCID: <https://orcid.org/0009-0006-9069-5249>

A. Sherbaf,

Associate Professor, Department
of Computer Science and Methods
of Teaching Computer Science,
Belarusian State Pedagogical University
named after Maxim Tank,
PhD in Physics and Mathematics
ORCID: <https://orcid.org/0009-0006-9069-5249>

Поступила в редакцию 28.10.2025.

Received on 28.10.2025.

В статье рассматривается проблема модернизации методики преподавания дисциплины «Вычислительные методы и компьютерное моделирование» для студентов педагогических специальностей физико-математического профиля в контексте требований образовательного стандарта по специальности «Физико-математическое образование». Обосновывается актуальность интеграции среды программирования Python в учебный процесс в качестве дополнения к традиционно используемым языкам Pascal и C#. Целью предлагаемого подхода является целенаправленное формирование универсальных и базовых профессиональных компетенций будущих специалистов образования, а также развитие у них вычислительного мышления.

Ключевые слова: вычислительные методы, вычислительная компетентность, вычислительное мышление, язык программирования Python.

The article considers the problem of modernizing the teaching methodology of the discipline «Computational Methods and Computer Modeling» for students of pedagogical specialties of the physical and mathematical profile in the context of the requirements of the educational standard of the specialty “Physics and Mathematics education”. The relevance of integrating the Python programming environment into the educational process as a supplement to the traditionally used Pascal and C# languages is substantiated. The purpose of the proposed approach is the targeted formation of universal and basic professional competencies of future education specialists, as well as the development of their computational thinking.

Keywords: computational methods, computational competence, computational thinking, programming language Python.

Введение. Цифровизация образования и общества предъявляет новые требования к профессиональной подготовке педагогических кадров. Кроме обладания фундаментальными знаниями в области педагогики и предметной области, современный педагог должен владеть вычислительными компетенциями. Согласно отчету ЮНЕСКО о цифровой грамотности и компетенциях, вычислительные компетенции – это набор навыков, связанных с использованием информационно-коммуника-

ционных технологий, включая умение работать с программным обеспечением, алгоритмами, данными и цифровыми инструментами для решения практических задач, а также умение работать с образовательными платформами, создавать и редактировать электронные материалы, анализировать данные для оценки эффективности обучения, а также обеспечивать информационную безопасность и этическое использование цифровых технологий [1]. Владение этими компетенциями позволяет

педагогам успешно адаптироваться к современным требованиям образования и повышать качество преподавания и обучения. В связи с этим особую значимость приобретает переосмысление подходов к преподаванию дисциплин, формирующих вычислительные компетенции будущих специалистов. Одной из таких дисциплин является учебная дисциплина «Вычислительные методы и компьютерное моделирование», модернизация содержания которой напрямую продиктована требованиями образовательного стандарта высшего образования по специальности «Физико-математическое образование» [2]. Он предусматривает формирование у будущего специалиста знаний, умений и навыков, необходимых для развития вычислительной компетентности у будущих педагогов.

Основная часть. Изучение дисциплины «Вычислительные методы и компьютерное моделирование» создает основу для формирования вычислительного мышления, позволяющего эффективно решать сложные задачи, моделировать и автоматизировать вычислительные процессы. Вычислительное мышление представляет собой способность человека мыслить, используя аналитические и алгоритмические подходы к постановке, анализу и решению проблемы с помощью компьютерной техники и технологий [3]. Другими словами, вычислительные компетенции – это результат вычислительного мышления, выражающийся в знаниях и навыках, которые дают возможность решать задачи с использованием технологий и алгоритмов.

Авторы считают целесообразным на практике использовать методику А. А. Самарского [4], которая предусматривает переход от постановки задачи к математической модели, затем к алгоритму ее решения и его компьютерной реализации с последующей оценкой полученных результатов. Под компьютерным моделированием понимается процесс построения компьютерной модели, представляющей объект, систему, понятие в форме, отличной от реальной, но приближенной к алгоритмическому описанию. А. А. Самарский подчеркивал, что компьютерное моделирование включает в себя:

- построение математической модели системы, которая отражает ее основные свойства и поведение;
- реализацию этой модели в виде алгоритма, пригодного для вычислений;

– использование компьютера для проведения расчетов, экспериментов и анализа полученных результатов.

Другими словами, компьютерное моделирование является важнейшим инструментом для исследования сложных систем в случае, когда экспериментальные методы затруднены или невозможны, а также средством для получения новых знаний и оптимизации процессов.

В соответствии с учебными программами дисциплин, связанных с программированием, традиционный подход к преподаванию вычислительных методов базируется на использовании языков Pascal и C#. Этот подход, безусловно, способствует формированию у студентов фундаментальных алгоритмических навыков и понимания структурного и объектно-ориентированного программирования. Следует подчеркнуть, что ориентация исключительно на Pascal и C# концентрирует внимание студентов на технические аспекты кодирования, зачастую оставляя в тени исследовательскую, экспериментальную сущность вычислительной математики. Студент учится реализовывать алгоритм, но испытывает затруднения при попытке использовать его в качестве инструмента для исследования, анализа и получения новых знаний.

В связи с этим возникает проблема несоответствия традиционного подхода к преподаванию вычислительных методов, основанного на языках Pascal/C#, современным требованиям образовательного стандарта к формированию исследовательских компетенций и развитию вычислительного мышления студентов. Решение данной проблемы видится в методически обоснованной интеграции в учебный процесс современного инструментария, а именно – языка программирования Python с его научным стеком библиотек.

Остановимся на некоторых аспектах интеграции языка Python в изучение вычислительных методов и компьютерного моделирования, которую следует рассматривать не как механическую замену одного языка программирования другим, а как переход на новый, более высокий уровень абстракции, дополняющий уже имеющиеся у студентов знания. Проведем сравнительный анализ трех подходов к изучению вычислительных методов. Предполагается, что к моменту изучения дисциплины «Вычислительные методы и компьютерное моделирование» студенты освоили основы алгоритмизации и программирования на языках Pascal и C#.

Алгоритмический подход. При изучении вычислительных методов с использованием языка Pascal на практических занятиях от студентов требуется детальная пошаговая реализация построенных алгоритмов. Такой подход особенно ценен на начальном этапе обучения, поскольку способствует формированию фундаментального понимания логики работы вычислительного метода, структуры данных и управляющих конструкций. Строгая типизация и процедурная модель Pascal развивают логическое мышление, побуждая студентов к тщательному планированию, строгому описанию типов данных и контролю над логикой ветвлений и циклов. Программируя вручную, например, метод Ньютона или метод Гаусса, студент глубоко погружается в детали алгоритма, преобразуя математическую модель в исполняемый код.

Несмотря на очевидные достоинства, алгоритмический подход имеет ряд недостатков. Во-первых, внимание студента переключается с сущности метода на технические детали реализации, он сосредотачивается на синтаксисе и отладке кода, вместо того чтобы анализировать устойчивость, сходимость и точность реализуемого метода. Во-вторых, использование устаревших языков программирования, таких как, например, Pascal, затрудняет использование современных вычислительных библиотек и инструментов визуализации. Все это снижает мотивацию студентов и усложняет решение прикладных задач. Кроме того, ручная реализация алгоритмов требует значительных временных затрат, что не всегда оправдано при обучении методам, которые уже реализованы в готовых библиотеках.

Объектно-ориентированный и событийно-управляемый подход. При переходе к использованию языка C# и технологии WinForms студент учится объектно-ориентированному и событийно-управляемому программированию. Основная проблема, над которой работает студент, – это «Как построить приложение для решения поставленной задачи?». Вычислительный метод инкапсулируется в класс, а взаимодействие с пользователем реализуется через элементы графического интерфейса. Обработка событий позволяет связать вычисления с действиями пользователя и способствует развитию навыков проектирования программного обеспечения и визуализации данных, необходимых будущим педагогам для создания интерактивных обучающих приложений.

Так же, как и при алгоритмическом подходе, студент сосредотачивается на технических аспектах реализации, разработке интерфейса и архитектуре приложения. Значительные усилия затрачиваются на проектирование визуальных компонентов и обработку событий, что отвлекает его от анализа реализуемого вычислительного метода, его точности, устойчивости и области применения. Кроме того, на ранних этапах обучения сложность среды разработки и необходимость освоения принципов объектно-ориентированного программирования могут препятствовать восприятию теоретического материала.

Исследовательско-ориентированный подход. При использовании на практических занятиях языка Python и его экосистемы научных библиотек (NumPy, SciPy, Matplotlib, SymPy и др.) процесс решения поставленных перед студентом задач кардинально меняется. Выбор из большого количества эффективных реализаций вычислительных методов требует от студента их изучения и исследования, что, в свою очередь, предполагает освоение соответствующего теоретического материала. Использование Python в сочетании с современными средами (например, Anaconda, Jupyter Notebook и др.) позволяет поэтапно выполнять код, визуализировать результаты, комментировать каждый шаг и оперативно вносить изменения. Все это способствует глубокому пониманию метода, его параметров, устойчивости, точности и области применения. У студента формируются навыки анализа, сравнения и визуализации различных вычислительных алгоритмов, а также моделирования реальных задач и проведения полноценных вычислительных экспериментов.

Отметим некоторые недостатки исследовательско-ориентированного подхода. Во-первых, он требует от студента знания основ Python, базовых навыков в работе с библиотеками и средами программирования. Приобретение таких знаний и навыков может быть затруднительно для студента, ранее работавшего с предыдущими поколениями средств разработки программ. Во-вторых, «скрытая» за готовыми функциями реализация вычислительного алгоритма не способствует развитию вычислительного мышления, без которого студент рискует формально воспринимать вычислительные методы.

Принимая во внимание преимущества и недостатки перечисленных подходов, авторы считают, что оптимальным решением может

стать *комбинированный подход*, при котором студент сначала осваивает алгоритмическую реализацию изучаемого вычислительного метода, затем переходит к его визуализации и, наконец, к анализу полученного решения в интерактивной среде.

Использование языка Python рассматривается не как замена языкам Pascal и C#, а как логическое дополнение и инструмент более высокого уровня. Python позволяет осуществить переход от этапа «кодирования алгоритмов» к этапу «проведения вычислительных экспериментов», что в полной мере соответствует задаче формирования исследовательских компетенций студента и развитию его вычислительного мышления.

Переход к исследовательской парадигме реализуется через комплекс лабораторных работ, в которых базовое задание на реализацию метода дополняется обязательными исследовательскими задачами, рассчитанными на применение готовой функции из научной библиотеки.

К базовым мы относим задания, широко представленные в [5]. Например:

- реализовать алгоритм вычислительного метода для решения задачи с заданной точностью;
- проверить выполнение условия сходимости;
- проверить решение, сравнив с полученным с помощью математического пакета и т. п.

Использование для программирования методов языка Python дает возможность воспользоваться преимуществами его экосистемы и дополнить практикум заданиями исследовательского характера. Например:

- сравнить эффективность и точность нескольких методов для одной задачи, строя графики зависимости погрешности от размера шага или числа итераций;
- исследовать сходимость метода, варьируя входные данные и визуализируя траектории приближений;
- исследовать устойчивость решения при наличии малых возмущений правой части системы линейных алгебраических уравнений;
- построить аппроксимирующие экспериментальные данные кривые, оценить влияние погрешности исходных данных на качество приближения функции;
- сравнить классические (полиномиальные) и современные (нейросетевые) методы аппроксимации функции.

Подобного рода задания готовят студентов к решению научно-практических задач средствами компьютерного моделирования и вычислительного эксперимента, тем самым развивая у них навыки аналитического мышления, системного подхода и умения работать с современными программными инструментами. Такой подход способствует формированию у студентов умения формулировать гипотезы, проводить виртуальные эксперименты, анализировать полученные результаты и делать обоснованные выводы. Студенты приобретают компетенции, необходимые для проведения исследований в различных областях науки и техники, а также для разработки инновационных решений в реальных практических условиях.

Реализация комбинированного подхода требует особого внимания к изучению языка программирования Python. Однако внедрение полноценного модуля по освоению этого языка в рамках стандартной учебной программы часто сталкивается с проблемой нехватки аудиторных часов. Эффективным решением этой проблемы представляется организация под руководством преподавателя самостоятельной работы студента, которая направлена на освоение базовых элементов синтаксиса и ключевых конструкций языка Python и соответствует современным методическим требованиям.

Важность знания языка Python для профессиональной деятельности будущих учителей физики, математики и информатики трудно переоценить. Этот язык программирования в настоящее время активно интегрируется в школьную программу. Python позволяет создавать интерактивные модели, симуляции и визуализации сложных физических и математических процессов, что делает обучение более наглядным и понятным. Использование Python помогает развивать навыки программирования, логического мышления и анализа данных, что важно для педагогов, стремящихся внедрять современные методы обучения. Знание Python способствует развитию инновационных методов обучения и подготовке учащихся к современному миру технологий.

Успешное самостоятельное освоение Python студентами обеспечивается целым рядом факторов. Прежде всего, язык отличается низким порогом вхождения благодаря своему простому, лаконичному и интуитивно понятному синтаксису, что особенно важно для студентов, уже имеющих базовые знания в области программирования. Студент, знакомый с Pascal или C#, уже обладает алго-

ритмическим мышлением и пониманием фундаментальных концепций программирования, таких как переменные, циклы, условия и функции. Все это позволяет ему осваивать Python через аналогии, концентрируясь на особенностях синтаксиса, а не на базовых принципах программирования.

Значительно облегчает процесс самостоятельного обучения обилие качественных образовательных материалов. Существует множество бесплатных интерактивных курсов, учебников, видеоуроков и документации, которые могут быть использованы студентами в процессе самообразования. Важную роль играет доступность облачных платформ разработки. Современные сервисы, такие как Google Colaboratory, полностью решают проблему установки и настройки программного обеспечения. Для начала работы студенту достаточно иметь браузер и доступ в интернет, что позволяет сосредоточиться на решении задач, а не на технических аспектах настройки рабочего окружения.

Роль преподавателя в этом процессе заключается в правильном направлении и мотивации студентов. Преподаватель должен предоставить им структурированный план обучения, подборку рекомендованных ресурсов и вводную лабораторную работу, демонстрирующую преимущества языка Python.

Сравнение результатов разных подходов в преподавании вычислительных методов позволяет авторам, исходя из их опыта, сделать обоснованный вывод об эффективности интеграции Python как инструмента для перехода от репродуктивной деятельности к исследовательской и, как следствие, для более качественного формирования ключевых компетенций будущего специалиста.

Заключение. Рассмотренный авторами подход к интеграции среды программирования Python в изучение учебной дисциплины «Вычислительные методы и компьютерное моделирование» является своевременным и педагогически целесообразным ответом на вызовы современной системы образования.

В ходе учебного процесса развивается вычислительное мышление студента, ведь он осваивает алгоритмизацию, декомпозиции, абстракции, анализ данных, обобщение результатов, что является ключевыми компонентами вычислительного мышления. Выпускник, освоивший такой подход, будет не просто учителем, способным объяснить учащемуся базовые понятия программирования, но и специалистом, готовым к профессиональной деятельности в условиях цифровой трансформации образования, способным ставить и решать исследовательские задачи и прививать эти навыки учащимся.

ЛИТЕРАТУРА

1. Цели образования в интересах устойчивого развития: задачи обучения. – ЮНЕСКО, 2017. – 62 с. – URL: https://unesdoc.unesco.org/ark:/48223/pf0000247444_rus (дата обращения: 22.03.2023).
2. Образовательный стандарт общего высшего образования по специальности 605-0113-04 «Физико-математическое образование», утвержденный постановлением Министерства образования Республики Беларусь 02.08.2023 № 225.
3. Aleynikova T. Computational thinking as a new reality of the educational process / T. Aleynikova, A. Sherbaf // Global science and innovations 2019: Central Asia: материалы VI Международной научно-практической конференции, Нур-Султан, Казахстан, май 2019 г. – Т. 4. – С. 80–84.
4. Самарский, А. А. Математическое моделирование. Идеи. Методы. Примеры / А. А. Самарский, А. П. Михайлов. – 2-е изд., испр. – М. : Физматлит, 2001. – 320 с.
5. Алейникова, Т. Г. Вычислительные методы : практикум / Т. Г. Алейникова, А. И. Шербаф – Витебск : ВГУ имени П. М. Машерова, 2020. – 98 с.

REFERENCES

1. Celi obrazovaniya v interesah ustojchivogo razvitiya: zadachi obucheniya. – YuNESKO, 2017. – 62 s. – URL: https://unesdoc.unesco.org/ark:/48223/pf0000247444_rus (data obrashcheniya: 22.03.2023).
2. Obrazovatel'nyj standart obshchego vysshego obrazovaniya po special'nosti 605-0113-04 «Fiziko-matematicheskoe obrazovanie», utverzhdenyj postanovleniem Ministerstva obrazovaniya Respubliki Belarus' 02.08.2023 № 225.
3. Aleynikova T. Computational thinking as a new reality of the educational process / T. Aleynikova, A. Sherbaf // Global science and innovations 2019: Central Asia: materialy VI Mezhdunarodnoj nauchno-prakticheskoy konferencii, Nur-Sultan, Kazakhstan, maj 2019 g. – Т. 4. – S. 80–84.
4. Samarskij, A. A. Matematicheskoe modelirovanie. Idei. Metody. Primery / A. A. Samarskij, A. P. Mihajlov. – 2-e izd., ispr. – M. : Fizmatlit, 2001. – 320 s.
5. Alejnikova, T. G. Vychislitel'nye metody : praktikum / T. G. Alejnikova, A. I. Sherbaf – Vitebsk : VGU imeni P. M. Masherova, 2020. – 98 s.