

Е. А. Мытникова

**РЕАЛИЗАЦИЯ ЗАДАЧНОГО ПОДХОДА ПРИ ОБУЧЕНИИ БАКАЛАВРОВ
НАПРАВЛЕНИЯ ПОДГОТОВКИ 09.03.04 «ПРОГРАММНАЯ ИНЖЕНЕРИЯ»
ОБЪЕКТНО-ОРИЕНТИРОВАННОМУ ПРОГРАММИРОВАНИЮ
В КОНТЕКСТЕ ПРОЕКТИРОВАНИЯ ПРОГРАММНОГО ПРОДУКТА**

*Чувашский государственный университет имени И. Н. Ульянова,
г. Чебоксары, Россия*

Аннотация. Современные требования к подготовке бакалавров по направлению подготовки «Программная инженерия» предполагают освоение полного цикла разработки программного продукта, включая архитектурное проектирование.

Традиционные подходы к обучению объектно-ориентированному программированию (ООП) нацелены на синтаксис языка, что не в полной мере соответствует профессиональным стандартам, требующим компетентности в области объектно-ориентированного анализа и проектирования.

В основе исследования – деятельностный и задачный подходы, синтез дидактических принципов, анализ образовательной практики и проектирование типологии задач, соответствующих этапам жизненного цикла программного продукта.

В статье описаны десять принципов реализации задачного подхода, семь критериев отбора задач по объектно-ориентированному программированию и их типология. Определены методические условия их реализации, включая многокритериальную оценку проектных решений и интеграцию с индустриальными инструментами.

Вышеизложенные методические подходы обеспечивают формирование компетентности в области проектирования программного продукта на базе объектно-ориентированного программирования, моделируют реальные профессиональные задачи. Научная новизна – интеграция задачного подхода с контекстом проектирования программного продукта. Практическая значимость заключается в разработке методических подходов к обучению будущих программных инженеров объектно-ориентированному программированию с учетом этапов жизненного цикла программного продукта.

Ключевые слова: *программная инженерия, объектно-ориентированное программирование, задачный подход, проектирование программного продукта, типология задач, принципы обучения, жизненный цикл программного продукта*

Е. А. Mytnikova

**IMPLEMENTATION OF A TASK-BASED APPROACH
IN TEACHING OBJECT-ORIENTED PROGRAMMING
TO THE 09.03.04 “SOFTWARE ENGINEERING”
BACHELOR’S DEGREE STUDENTS
IN THE CONTEXT OF SOFTWARE PRODUCT DESIGN**

I. Ulyanov Chuvash State University, Cheboksary, Russia

Abstract. Contemporary educational requirements for bachelor’s degree programs in “Software Engineering” necessitate comprehensive coverage of the entire software product development cycle, with particular emphasis on architectural design capabilities.

Conventional approaches to teaching object-oriented programming (OOP) predominantly emphasize language syntax, thereby insufficiently addressing professional standards that demand competencies in object-oriented analysis and architectural design.

The research methodology encompasses activity-based and task-based pedagogical frameworks, synthesis of didactic principles, systematic analysis of current educational practices, and development of a task typology corresponding to software product life cycle stages.

The paper presents ten principles for task-based approach implementation, seven criteria for OOP task selection, and their corresponding typology. Methodological implementation conditions are established, incorporating multi-criteria assessment of design solutions and integration with industry-standard development tools.

The proposed methodological framework facilitates the development of software product design competencies grounded in object-oriented programming principles while authentically modeling professional practice scenarios. The scientific contribution resides in the integration of the task-based approach within the software product design context. The practical value manifests in the development of methodologies for training prospective software engineers in object-oriented programming, with explicit consideration of software product life cycle phases.

Keywords: *software engineering, object-oriented programming, task-based approach, software product design, task typology, pedagogical principles, software product life cycle*

Введение. Современный этап развития Российской Федерации, характеризующийся цифровой трансформацией экономики и стратегическим курсом на технологический суверенитет, предъявляет повышенные требования к подготовке кадров для ИТ-индустрии. Указ Президента РФ от 18.06.2024 № 529 ([11]) относит создание доверенного и защищенного программного обеспечения к критическим технологиям, а направление подготовки 09.03.04 «Программная инженерия» включается в перечень приоритетных специальностей для обеспечения потребностей аккредитованных ИТ-организаций (Приказ Минцифры России № 712 от 26.09.2022 [12]). Ключевым аспектом профессиональной деятельности программного инженера, отличающим его от программиста-кодировщика, выступает проектирование программного продукта (ПП) – создание его архитектуры и управление всем жизненным циклом (ЖЦ), от анализа требований до сопровождения и рефакторинга.

Объектно-ориентированное программирование остается доминирующей парадигмой в современной индустрии и фундаментальной основой для моделирования предметной области и проектирования архитектуры сложных систем ([16], [17]). Однако анализ научно-методической литературы и существующей образовательной практики позволяет констатировать наличие педагогического противоречия между требованиями профессиональной деятельности и образовательных стандартов, предполагающими у выпускников направления 09.03.04 «Программная инженерия» сформированность компетенций в области объектно-ориентированного анализа, проектирования архитектуры программных систем и управления их ЖЦ, и преобладающей ориентацией традиционных методических подходов к обучению ООП на освоение синтаксиса языка программирования и решение изолированных алгоритмических задач, что не обеспечивает формирование системного мышления, способности к объектной декомпозиции сложных предметных областей, обоснованному выбору архитектурных решений и применению паттернов проектирования.

Это противоречие приводит к разрыву между академической подготовкой и реальными потребностями ИТ-индустрии: выпускники, формально владея синтаксисом ООП языков, испытывают значительные трудности при решении комплексных задач проектирования реальных ПП. В этой связи возникает проблема, заключающаяся в выявлении теоретико-методических аспектов реализации задачного подхода при обучении бакалавров направления подготовки 09.03.04 «Программная инженерия» объектно-ориентированному программированию в контексте проектирования ПП.

Таким образом, цель исследования состоит в теоретическом обосновании и разработке методической системы реализации задачного подхода при обучении бакалавров направления подготовки 09.03.04 «Программная инженерия» объектно-ориентированному программированию в контексте проектирования ПП, включающей принципы реализации задачного подхода, критерии отбора и типологию учебных задач, соответствующих этапам ЖЦ ПП.

Вопросам изучения задачного подхода в профессиональном образовании посвящены работы В. А. Голяковой [2], М. С. Горбузовой [3], И. В. Кирьяковой [4], С. М. Комаровой [5], Л. А. Кочеровой [6], Л. А. Кугеля [7], И. Р. Федоровой [13], М. В. Шингаревой [15] и др. Данный подход выступает технологической реализацией деятельностного. Учебная задача определяется как модель профессиональной проблемы, требующая нахождения нового способа деятельности ([1], [10], [14]).

Актуальность применения задачного подхода в обучении будущих программных инженеров обусловлена тем, что его применение позволяет перенести фокус с усвоения знаний о синтаксисе ООП на формирование способа деятельности – объектно-ориентированного анализа и проектирования, а также помогает моделировать условия и структуру реальной профессиональной задачи, включая рефакторинг и кросс-функциональную интеграцию.

Необходимость интеграции ООП и контекста проектирования обусловлена системностью профессиональной деятельности программиста, который должен освоить полный ЖЦ ПП.

В связи с этим вводится следующее понятие: *учебная профессионально-ориентированная задача по объектно-ориентированному программированию для бакалавров программной инженерии* – это комплексное, контекстно-аутентичное задание, моделирующее проблемную ситуацию из практики разработки ПП и требующее от обучающегося не только использования ООП, но и осуществления деятельности на различных этапах ЖЦ ПП.

Актуальность исследуемой проблемы. Российская ИТ-индустрия испытывает дефицит квалифицированных разработчиков в области проектирования сложных систем. Корень проблемы лежит в разрыве между содержанием и методами университетского обучения ООП и реальными задачами, стоящими перед программным инженером в определенном секторе экономике. Традиционный фокус на синтаксисе языка и решении алгоритмических задач не формирует у выпускников способности к объектно-ориентированному анализу и проектированию архитектуры, понимания ЖЦ ПП, навыков работы с требованиями, тестирования, рефакторинга и сопровождения. Актуальность исследования определяется социальной востребованностью в разработке теоретических моделей обучения, способных преодолеть этот разрыв и обеспечить подготовку конкурентоспособных специалистов ([8], [9]).

Материал и методы исследования. Исследование опирается на деятельностный подход и теорию учебной деятельности (Л. С. Выготский, П. Я. Гальперин, А. Н. Леонтьев, С. Л. Рубинштейн, Н. Ф. Талызина), теорию учебных задач и задачный подход (Г. А. Балл, Е. И. Машбиц, Л. М. Фридман) и современные требования к образованию в области информационных технологий. Методологической базой служат системный анализ образовательных практик, синтез дидактических принципов и проектирование типологии задач, соответствующих этапам ЖЦ ПП.

Результаты исследования и их обсуждение. Основным результатом данного исследования является целостная методическая система, реализующая задачный подход в обучении объектно-ориентированному программированию бакалавров программной инженерии. Система структурно интегрирует четыре взаимосвязанных компонента: принципы реализации, критерии отбора задач, их типологию и методические условия внедрения. Рассмотрим каждый из компонентов, акцентируя внимание на их педагогическом смысле и вкладе в преодоление выявленного противоречия.

Методическая система базируется на десяти принципах, которые выступают в качестве руководящих правил для организации учебного процесса, обеспечивая его ориентацию на проектирование ПП.

Таблица 1 – Принципы реализации задачного подхода

№	Принцип	Сущность принципа
1	Принцип учета выбора предметной области	Предоставление студенту выбора предметной области для проекта (из предложенного перечня), соответствующей его профессиональным интересам
2	Принцип проблемно-исследовательской направленности задач	Задачи должны содержать открытый вопрос, требующий поиска и обоснования проектного решения
3	Принцип контекстной аутентичности задач	Задачи должны моделировать реальные проблемные ситуации из практики программной инженерии, включая работу с реалистичными предметными областями
4	Принцип инкрементального усложнения архитектурных решений	Сложность требуемых архитектурных решений возрастает постепенно: от простых задач к разработке многокомпонентных систем
5	Принцип поэтапности выполнения практических заданий	Каждое крупное задание разделено на четкие этапы, соответствующие стадиям ЖЦ ПП
6	Принцип объектной декомпозиции предметной области	Формирование умения разбивать сложную область на независимые объекты / классы до начала кодирования
7	Принцип вариативности и альтернативности решений	Задачи должны допускать множество корректных архитектурных решений, требующих их сравнения и выбора
8	Принцип обоснованности решений	Любое проектное решение должно быть подкреплено теоретическими знаниями и аргументировано перед экспертом
9	Принцип реструктуризации и улучшения кода и итеративности	Обучение включает работу с существующим кодом, его улучшение и адаптацию
10	Принцип кросс-функциональной интеграции	Обучение ООП должно быть интегрировано с другими смежными дисциплинами, моделирующими реальную рабочую среду

Рассмотрим педагогическое обоснование некоторых принципов реализации задачного подхода.

Принцип учета выбора предметной области основан на положениях теории мотивации в обучении. Предоставление студенту возможности выбора предметной области для проектирования ПП из предложенного преподавателем перечня (медицинская информационная система, система управления проектами, образовательная платформа, финансовая аналитика и т. д.) обеспечивает внутреннюю мотивацию обучающегося и позволяет связать учебную деятельность с его профессиональными интересами. Структурированный перечень предметных областей, с одной стороны, не ограничивает студента в выборе той или иной сферы, с другой стороны, позволяет преподавателю оценить разработанные студентами архитектурные решения. Это важно для будущих программных инженеров, поскольку в реальной практике им придется работать с различными предметными областями, требующими глубокого понимания специфики бизнес-процессов.

Принцип обоснованности решений отражает умение аргументировать принятые архитектурные решения. В отличие от программиста-кодировщика, который реализует готовые спецификации, программный инженер должен уметь обосновывать выбор между альтернативными решениями, опираясь на принципы SOLID, паттерны проектирования и требования к качеству ПП (производительность, масштабируемость, поддерживаемость). Этот принцип реализуется через включение в структуру учебных заданий обязательного компонента – письменного или устного обоснования проектного решения с анализом его преимуществ и недостатков, альтернативных вариантов и критериев выбора.

Принцип реструктуризации и улучшения кода признает рефакторинг неотъемлемой частью процесса разработки ПП. Традиционное обучение ООП фокусируется на создании кода «с нуля», игнорируя тот факт, что значительная часть профессиональной деятельности программного инженера связана с работой над существующей кодовой базой. Включение рефакторинга в учебный процесс формирует критическое отношение к качеству кода, что является признаком сформированной проектной культуры программного инженера. Студенты учатся не только создавать новые решения, но и систематически улучшать существующие, применяя принципы объектно-ориентированного проектирования для повышения качества ПП.

Принцип кросс-функциональной интеграции отражает реальность современной разработки ПП, где программный инженер должен владеть не только языком программирования, но и экосистемой инструментов и практик. Интеграция с системами контроля версий, практиками автоматизированного тестирования, работой с базами данных и использованием UML для документирования архитектуры создает у студентов целостное представление о профессиональной деятельности и снижает адаптационный период при выходе на рынок труда.

На основе вышеуказанных принципов сформулированы семь критериев отбора задач, которые выступают методологическим фильтром для учебного материала:

1. Критерий соответствия этапам ЖЦ ПП – задача должна моделировать конкретный этап или несколько этапов ЖЦ ПП.

2. Критерий демонстрации принципов и механизмов ООП – задача должна требовать применения фундаментальных принципов ООП: абстракции, инкапсуляции, наследования, полиморфизма.

3. Критерий практической значимости и контекстной аутентичности – задача должна быть основана на реальных или реалистичных проблемных ситуациях из практики разработки ПП.

4. Критерий вариативности решений – задача должна допускать несколько корректных архитектурных решений, требующих обоснования выбора.

5. Критерий масштабируемости – задача должна предусматривать возможность расширения функциональности и усложнения архитектуры.

6. Критерий использования инструментальных средств – задача должна предполагать применение профессиональных инструментов разработки.

7. Критерий комплексности (кросс-функциональной интеграции) – задача должна интегрировать ООП со смежными областями.

На основе критерия соответствия этапам ЖЦ ПП разработана типология задач, решение которых способствует формированию компетенции в области проектирования ПП на базе объектно-ориентированного программирования.

Таблица 2 – Типология задач по ООП

Тип задачи	Основная деятельность
1. Аналитические	Объектно-ориентированный анализ, выделение классов-сущностей
2. Проектировочные	Применение паттернов проектирования, создание UML-моделей
3. Реализационные	Кодирование с реализацией принципов ООП
4. Рефакторинговые	Улучшение структуры и качества существующего кода
5. Адаптационные	Расширение функциональности с применением принципов открытости / закрытости
6. Диагностические	Анализ готового кода, выявление нарушений принципов ООП
7. Сравнительно-оценочные	Сравнение альтернативных архитектурных решений, обоснование выбора

Приведем примеры некоторых задач, предлагаемых студентам при изучении ООП, иллюстрирующих реализацию разработанной типологии.

Пример 1. Аналитическая задача

Проведите объектно-ориентированный анализ предметной области «Система управления библиотекой». Выделите основные классы-сущности, определите их атрибуты и ответственности. Постройте концептуальную диаграмму классов в нотации UML. Обоснуйте выделение каждого класса с точки зрения принципа единственной ответственности. Определите типы отношений между классами (ассоциация, агрегация, композиция, наследование).

Ожидаемый результат: UML-диаграмма классов с пояснительной запиской, содержащей обоснование каждого проектного решения.

Педагогическая ценность: данная задача реализует *принцип объектной декомпозиции* и соответствует этапу анализа требований в ЖЦ ПП. Студент должен продемонстрировать умение выделять абстракции до начала кодирования, что является важным навыком для архитектора ПП. Задача формирует способность мыслить категориями объектов и их взаимодействий, а не процедур и алгоритмов.

Пример 2. Рефакторинговая задача

Дан код класса, нарушающий несколько принципов ООП:

- класс содержит более 500 строк кода;
- реализует валидацию заказа, расчет стоимости, применение скидок, отправку уведомлений и логирование;
- содержит множественные вложенные условные операторы;
- использует конкретные классы вместо абстракций.

Выполните рефакторинг кода:

1. Определите нарушенные принципы SOLID.
2. Разработайте улучшенную архитектуру с выделением отдельных классов ответственности.
3. Примените паттерны проектирования (Strategy, Decorator, Chain of Responsibility).
4. Обоснуйте каждое изменение.

Ожидаемый результат: рефакторированный код с UML-диаграммой до / после и аналитической запиской, описывающей улучшения.

Педагогическая ценность: данная задача реализует *принцип реструктуризации и улучшения кода* и моделирует типичную ситуацию из практики сопровождения ПП. Студенты учатся распознавать антипаттерны и применять систематический подход к улучшению качества кода. Такая работа развивает способность к анализу и оценке и формирует понимание того, как неправильные проектные решения приводят к проблемам сопровождения.

Пример 3. Сравнительно-оценочная задача

Для системы управления документами необходимо реализовать функциональность экспорта документов в различные форматы (PDF, DOCX, HTML, Markdown). Проанализируйте и сравните три архитектурных решения:

- Вариант А. Использование условных операторов в методе экспорта.
- Вариант Б. Применение паттерна Strategy.
- Вариант В. Применение паттерна Factory Method + Strategy.

Для каждого варианта оцените:

- соответствие принципу открытости / закрытости;
- сложность добавления нового формата экспорта;
- читаемость и поддерживаемость кода;
- возможность повторного использования компонентов.

Обоснуйте выбор оптимального решения для данной задачи.

Ожидаемый результат: сравнительная таблица с анализом каждого варианта, UML-диаграммы для каждого решения, аргументированное заключение с выбором оптимального варианта.

Педагогическая ценность: эта задача реализует *принцип вариативности и альтернативности решений* и *принцип обоснованности решений*. Студенты развивают аналитические умения и формируют способность к обоснованию архитектурных решений на основе системного анализа альтернатив. Понимание того, что правильное решение зависит от контекста и требований, является признаком сформированности умений в области архитектурного проектирования программных систем.

Выводы. Традиционные методики обучения программированию направлены в основном на освоение синтаксиса языка и реализации алгоритмов, вместе с тем специфика профессиональной деятельности выпускников по направлению подготовки «Программная инженерия» заключается в проектировании ПП с учетом его ЖЦ. Кроме того, объектно-ориентированное программирование занимает доминирующее место в современной индустрии разработки программного обеспечения, следовательно, востребованными становятся умения проектирования архитектуры программных систем. При этом специфика объектно-ориентированного программирования требует формирования не только навыков кодирования, но и системного подхода к проектированию ПП, способности к объектно-ориентированному анализу предметной области, обоснованному выбору архитектурных решений и применению паттернов проектирования. Решению этого противоречия способствует реализация задачного подхода, совершенствование которого в контексте проектирования ПП позволяет обосновать и сформулировать принципы обучения ООП будущих бакалавров программной инженерии, типологию и критерии отбора учебных задач. Это способствует подготовке конкурентоспособных, востребованных на рынке ИТ программных инженеров, компетентных в области разработки программного обеспечения.

Перспективы дальнейших исследований включают апробацию предложенной системы в рамках учебных курсов, разработку цифровых средств оценки архитектурных решений студентов и расширение модели на магистерские программы, ориентированные на архитектуру программного обеспечения и проектные паттерны.

ЛИТЕРАТУРА

1. Балл Г. А. Теория учебных задач: психолого-педагогический аспект. – М. : Педагогика, 1990. – 184 с.
2. Голякова В. А. Подготовка курсантов военного вуза к решению компетентностно-ориентированных профессиональных задач : автореф. дис. ... канд. пед. наук : 13.00.08. – Челябинск, 2019. – 25 с.
3. Горбузова М. С. Методика использования систем контекстных задач при обучении будущих учителей информационным технологиям : автореф. дис. ... канд. пед. наук : 13.00.02. – Волгоград, 2015. – 28 с.
4. Кирьякова И. В. Задачный подход в обучении основам программного обеспечения для развития продуктивного мышления будущего учителя информатики : автореф. дис. ... канд. пед. наук : 13.00.02. – Омск, 2007. – 23 с.
5. Комарова С. М. Методика обучения бакалавров педагогического образования, специализирующихся в области информационных технологий, компьютерному моделированию с использованием межпредметных задач : автореф. дис. ... канд. пед. наук : 13.00.02. – СПб., 2017. – 22 с.
6. Кочерова Л. А. Задачный подход как технология индивидуально-дифференцированного обучения сотрудников ОВД РФ : автореф. дис. ... канд. пед. наук : 13.00.08. – Махачкала, 2017. – 23 с.
7. Кугель Л. А. Обучение студентов алгоритмизации и программированию на основе структурно-алгоритмического подхода к постановке и реализации задач (на примере направления подготовки бакалавров «Прикладная информатика») : автореф. дис. ... канд. пед. наук : 13.00.02. – М., 2015. – 23 с.
8. Лавина Т. А., Ильина Л. А. Кейс-технология как средство формирования ИКТ-компетенций бакалавров по информационной безопасности при обучении информатике и языкам программирования // Вестник Чувашского государственного педагогического университета им. И. Я. Яковлева. – 2022. – № 1(114). – С. 171–176. – DOI 10.26293/chgpu.2022.114.1.022.

9. Лавина Т. А., Мытникова Е. А., Ярускина Е. Т. Подготовка по программированию бакалавров направления «Программная инженерия» с учетом концепции комплексного подхода к инженерному образованию // *Современные наукоемкие технологии*. – 2023. – № 7. – С. 160–166. – DOI 10.17513/snt.39712.
10. Машибиц Е. И. Психологический анализ учебной задачи // *Советская педагогика*. – 1973. – № 2. – С. 58–65.
11. Об утверждении приоритетных направлений научно-технологического развития и перечня важнейших наукоемких технологий: указ Президента Российской Федерации от 18.06.2024 № 529 [Электронный ресурс]. – URL : <http://publication.pravo.gov.ru/document/0001202406180018> (дата обращения: 01.11.2025).
12. О рекомендованном перечне приоритетных специальностей и направлений подготовки высшего образования для обеспечения основных потребностей аккредитованных организаций, осуществляющих деятельность в области информационных технологий, и операторов связи в квалифицированных кадрах: приказ Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации от 26 сентября 2022 г. № 712 [Электронный ресурс]. – URL : https://www.consultant.ru/document/cons_doc_LAW_427459/2ff7a8c72de3994f30496a0ccb1ddafdad518/ (дата обращения: 01.11.2025).
13. Федорова И. Р. Формирование общих компетенций студентов профессиональных образовательных организаций средствами учебных задач : автореф. дис. ... канд. пед. наук : 13.00.08. – Томск, 2020. – 23 с.
14. Фридман Л. М. Логико-психологический анализ школьных учебных задач. – М. : Педагогика, 1977. – 208 с.
15. Шингарева М. В. Проектирование компетентностно-ориентированных задач по учебным дисциплинам вуза : автореф. дис. ... канд. пед. наук : 13.00.08. – М., 2012. – 23 с.
16. Rangisetti A. K. The Importance of Object-Oriented Programming // *Hands-On Object-Oriented Programming: Mastering OOP Features for Real-World Software Systems Development*. – Berkeley, CA : Apress, 2024. – P. 1–41. – DOI 10.1007/979-8-8688-0524-0_1.
17. Singh N., Chouhan S. S., Verma K. Object oriented programming: Concepts, limitations and application trends // 2021 5th International Conference on Information Systems and Computer Networks (ISCON). – IEEE, 2021. – P. 1–4. – DOI 10.1109/ISCON52037.2021.9702463.

Статья поступила в редакцию 09.11.2025

REFERENCES

1. Ball G. A. *Teoriya uchebnykh zadach: psikhologo-pedagogicheskij aspekt*. – М. : Pedagogika, 1990. – 184 s.
2. Golyakova V. A. *Podgotovka kursantov voennogo vuza k resheniyu kompetentnostno-orientirovannykh professional'nykh zadach* : avtoref. dis. ... kand. ped. nauk : 13.00.08. – Chelyabinsk, 2019. – 25 s.
3. Gorbuzova M. S. *Metodika ispol'zovaniya sistem kontekstnykh zadach pri obuchenii budushchikh uchitelej informacionnym tekhnologiyam* : avtoref. dis. ... kand. ped. nauk : 13.00.02. – Volgograd, 2015. – 28 s.
4. Kir'yakova I. V. *Zadachnyj podkhod v obuchenii osnovam programmnoho obespecheniya dlya razvitiya produktivnogo myshleniya budushchego uchitelya informatiki* : avtoref. dis. ... kand. ped. nauk : 13.00.02. – Omsk, 2007. – 23 s.
5. Komarova S. M. *Metodika obucheniya bakalavrov pedagogicheskogo obrazovaniya, specializiruyushchikhsya v oblasti informacionnykh tekhnologij, komp'yuternomu modelirovaniyu s ispol'zovaniem mezhpredmetnykh zadach* : avtoref. dis. ... kand. ped. nauk : 13.00.02. – SPb., 2017. – 22 s.
6. Kocherova L. A. *Zadachnyj podkhod kak tekhnologiya individual'no-differencirovannogo obucheniya sotrudnikov OVD RF* : avtoref. dis. ... kand. ped. nauk : 13.00.08. – Makhachkala, 2017. – 23 s.
7. Kugel' L. A. *Obuchenie studentov algoritmizacii i programmirovaniyu na osnove strukturno-algoritmicheskogo podkhoda k postanovke i realizacii zadach (na primere napravleniya podgotovki bakalavrov «Prikladnaya informatika»)* : avtoref. dis. ... kand. ped. nauk : 13.00.02. – М., 2015. – 23 s.
8. Lavina T. A., Il'ina L. A. *Kejs-tekhnologiya kak sredstvo formirovaniya IKT-kompetencij bakalavrov po informacionnoj bezopasnosti pri obuchenii informatike i yazykam programmirovaniya* // *Vestnik Chuvashskogo gosudarstvennogo pedagogicheskogo universiteta im. I. Ya. Yakovleva*. – 2022. – № 1(114). – S. 171–176. – DOI 10.26293/chgpu.2022.114.1.022.
9. Lavina T. A., Mytnikova E. A., Yarus'kina E. T. *Podgotovka po programmirovaniyu bakalavrov napravleniya «Programmная inzheneriya» s uchetom koncepcii kompleksnogo podkhoda k inzhenernomu obrazovaniyu* // *Sovremennye naukoemkie tekhnologii*. – 2023. – № 7. – С. 160–166. – DOI 10.17513/snt.39712.
10. Mashbic E. I. *Psikhologicheskij analiz uchebnoj zadachi* // *Sovetskaya pedagogika*. – 1973. – № 2. – С. 58–65.
11. Об утверждении приоритетных направлений научно-технологического развития и перечня важнейших наукоемких технологий: указ Президента Российской Федерации от 18.06.2024 № 529 [Электронный ресурс]. – URL : <http://publication.pravo.gov.ru/document/0001202406180018> (дата обращения: 01.11.2025).

12. O rekomendovannom perechne prioritetnykh special'nostej i napravlenij podgotovki vysshego obrazovaniya dlya obespecheniya osnovnykh potrebnostej akkreditovannykh organizacij, osushchestvlyayushchikh deyatelnost' v oblasti informacionnykh tekhnologij, i operatorov svyazi v kvalificirovannykh kadrakh: prikaz Ministerstva cifrovogo razvitiya, svyazi i massovykh kommunikacij Rossijskoj Federacii ot 26 sentyabrya 2022 g. № 712 [Elektronnyj resurs]. – URL : https://www.consultant.ru/document/cons_doc_LAW_427459/2ff7a8c72de3994f30496a0ccbb1ddafdaddf518/ (data obrashcheniya: 01.11.2025).

13. Fedorova I. R. Formirovanie obshchikh kompetencij studentov professional'nykh obrazova-tel'nykh organizacij sredstvami uchebnykh zadach : avtoref. dis. ... kand. ped. nauk : 13.00.08. – Tomsk, 2020. – 23 s.

14. Fridman L. M. Logiko-psihologicheskij analiz shkol'nykh uchebnykh zadach. – M. : Pedagogika, 1977. – 208 s.

15. Shingareva M. V. Proektirovanie kompetentnostno-orientirovannykh zadach po uchebnym disciplinam vuza : avtoref. dis. ... kand. ped. nauk : 13.00.08. – M., 2012. – 23 s.

16. Rangiseti A. K. The Importance of Object-Oriented Programming // Hands-On Object-Oriented Programming: Mastering OOP Features for Real-World Software Systems Development. – Berkeley, CA : Apress, 2024. – P. 1–41. – DOI 10.1007/979-8-8688-0524-0_1.

17. Singh N., Chouhan S. S., Verma K. Object oriented programming: Concepts, limitations and application trends // 2021 5th International Conference on Information Systems and Computer Networks (ISCON). – IEEE, 2021. – P. 1–4. – DOI 10.1109/ISCON52037.2021.9702463.

The article was contributed on November 9, 2025

Сведения об авторе

Мытникова Екатерина Анатольевна – старший преподаватель кафедры компьютерных технологий Чувашского государственного университета имени И. Н. Ульянова, г. Чебоксары, Россия, <https://orcid.org/0000-0001-7974-5520>, amaliaamalia@yandex.ru

Author Information

Mytnikova, Ekaterina Anatolyevna – Senior Lecturer of the Department of Computer Technologies, I. Ulyanov Chuvash State University, Cheboksary, Russia, <https://orcid.org/0000-0001-7974-5520>, amaliaamalia@yandex.ru