

О РАЗНОУРОВНЕВОМ ОБУЧЕНИИ ПРОГРАММИРОВАНИЮ В КУРСЕ ИНФОРМАТИКИ ОСНОВНОЙ ШКОЛЫ В УСЛОВИЯХ ДИФФЕРЕНЦИАЦИИ СОДЕРЖАНИЯ ОБУЧЕНИЯ

Л.Л. Босова, Н.Н. Самылкина, В.А. Мишин

Аннотация. В статье раскрываются значения и различные аспекты использования общедидактических понятий — «уровень усвоения», «уровень образования», «уровень изучения предмета», «индивидуализация», «дифференциация», «персонализация» и др. — применительно к изучению программирования в курсе информатики основной школы. Рассматриваются теоретические основы разноуровневого обучения в классе (группе) в условиях дифференцированного обучения (по содержанию и уровням изучения), история становления дифференцированного обучения в нашей стране, общедидактические проблемы описания качественных характеристик уровня усвоения на разных уровнях изучения предмета. Описывается реализация разноуровневого обучения программированию в курсе информатики основного общего образования с использованием активных методов обучения и современных образовательных технологий для учителей информатики, методистов по информатике и педагогов-исследователей в области теории и методики обучения информатике.

Ключевые слова: разноуровневое обучение программированию, дифференциация содержания обучения, уровни усвоения, внутренний контроль, внешний контроль, персонализация, обновленный ФГОС общего образования.

Для цитирования: Босова Л.Л., Самылкина Н.Н., Мишин В.А. О разноуровневом обучении программированию в курсе информатики основной школы в условиях дифференциации содержания обучения // Преподаватель XXI век. 2024. № 1. Часть 1. С. 253–273. DOI: 10.31862/2073-9613-2024-1-253-273

253

ON MULTILEVEL TEACHING OF PROGRAMMING IN THE BASIC SCHOOL COMPUTER SCIENCE COURSE WITHIN THE DIFFERENTIATION OF TEACHING CONTENT

L.L. Bosova, N.N. Samylkina, V.A. Mishin

Abstract. The article reveals the meanings and various aspects of using general didactic concepts such as “level of assimilation”, “level of education”, “level of learning the

© Босова Л.Л., Самылкина Н.Н., Мишин В.А., 2024



Контент доступен по лицензии Creative Commons Attribution 4.0 International License
The content is licensed under a Creative Commons Attribution 4.0 International License

subject”, “individualization”, “differentiation”, “personalization”, etc. applied to the study of programming in the basic school computer science course. Theoretical bases of multilevel learning in a class (group) in terms of differentiated learning (by content and levels of learning), the history of differentiated learning in our country, general didactic problems of describing qualitative characteristics of learning levels at different stages of learning the subject are considered. The author describes the implementation of multilevel teaching of programming in the course of computer science of basic general education using active teaching methods and modern educational technologies for teachers of computer science, methodologists in computer science and pedagogical researchers in the field of theory and methodology of computer science learning.

Keywords: *multilevel programming teaching, differentiation of teaching content, learning levels, internal control, external control, personalization, updated FSES of general education.*

Cite as: Bosova J.L., Samylkina H.N., Mishin B.A. On Multilevel Teaching of Programming in the Basic School Computer Science Course within the Differentiation of Teaching Content. *Prepodavatel XXI vek. Russian Journal of Education*, 2024, No. 1, part 1, pp. 253–273. DOI: 10.31862/2073-9613-2024-1-253-273

Актуальность реализации разноуровневого обучения программированию

В современной дидактике продолжают оставаться актуальными вопросы, связанные с разноуровневым обучением и оценением его результатов в системе общего образования. Это связано с изменившимися социальными запросами и нормативными условиями реализации образовательных программ, а также разнообразием организаций, реализующих основные и дополнительные образовательные программы. Одной из задач современного образования является формирование у обучающихся цифровых компетенций, которые принято трактовать как «знания, навыки и деятельностные установки, позволяющие в условиях цифровизации экономики и социальной сферы применять для решения задач или достижения требуемого результата информационно-коммуникационные технологии» [1]. Достаточно полный перечень цифровых компетенций, необходимых для жизни и работы в условиях цифровой экономики, представлен в европейской модели цифровых навыков для граждан [2]. Цифровые компетенции подразделяются на две группы: 1) базовые как развитые навыки, используемые для жизни; 2) специальные, необходимые для профессиональной деятельности. В обновленных федеральных государственных стандартах общего образования (ФГОС ОО) цифровые компетенции представлены не только как предметные результаты, достигаемые в процессе изучения информатики, но и как метапредметные, формируемые усилиями многих учебных предметов.

Важной частью школьного курса информатики является изучение алгоритмов и программирования; последнее в современном мире приобретает статус массово востребованной технологии. Действительно, в процессе освоения программирования формируются практически все группы цифровых навыков (информационная грамотность, общение и сотрудничество, создание цифрового контента, безопасность, решение проблем); изучение программирования способствует формированию у людей новых ценностей цифрового общества (сообществ, совместной работы, обмена знаниями); благодаря

навыкам программирования человек лучше понимает правила поведения в цифровой среде, чувствует себя в ней более уверенно и комфортно. Навыки программирования являются как базовой цифровой компетенцией, так и специальной цифровой компетенцией, необходимой для профессиональной деятельности. Очевидно, что имеются в виду разные виды деятельности в программировании, различные уровни владения технологиями программирования в самых разнообразных средах и на разных языках.

Следует отметить, что идея разделения программирования на программирование для жизни и программирование для профессиональной деятельности получила широкую поддержку среди специалистов в образовании; она является основой обучения программированию во многих странах с развитой экономикой и высоким уровнем цифровой грамотности. Например, по этому пути идут Финляндия, Великобритания, Канада, США, Япония и др. [3]. С введением обновленных ФГОС основного общего образования, предусматривающих возможность выбора базового или углубленного уровня изучения информатики, у нас также появляется возможность при изучении программирования уделять внимание формированию образовательных результатов с ориентацией на их использование для жизни (базовый уровень) или в будущей профессии (углубленный уровень). В связи с вышеизложенным актуальность приобретает разработка методики разноуровневого обучения программированию в основной школе.

Теоретические основы разноуровневого обучения в условиях дифференциации содержания обучения

Проблема дифференциации в образовании имеет многолетнюю историю в отечественной и зарубежной педагогике и дидактике. Первые идеи о потребности индивидуального подхода к учащимся (в условиях обучения в коллективе) принадлежат Яну Амосу Коменскому. В своих работах он указывал на необходимость использования индивидуальной и групповой учебной деятельности. Эти же идеи нашли отражение в трудах К.Д. Ушинского, а в начале XX века — в трудах В.П. Вахтерова, В.И. Водовозова, П.Ф. Каптерева и др. Так, П.Ф. Каптерев утверждал, что дифференциация в отечественном образовании должна опираться на различные способности, а не на различные сословия [4]. Тем не менее в дореволюционной России имела место именно сословная дифференциация в образовании, поддерживавшая интересы высших сословий.

В советское время вопрос о дифференциации образования рассматривался в 1918 году во время подготовки таких документов, как «Декларация о единой трудовой школе» и «Положение о единой трудовой школе». В них предусматривалось разделение учебных планов в старших классах на три направления: технический, гуманитарный, естественно-математический. Тогда дифференциация обучения рассматривалась как углубленное изучение учебных предметов с учетом общественных потребностей, способностей и интересов учащихся. Далее идеи индивидуально-дифференцированного подхода в обучении находят отражение в работах П.П. Блонского, Н.К. Крупской, А.В. Луначарского и других отечественных педагогов. В 1932 году принимается постановление «Об учебных программах и режиме в начальной и средней школе», которое восстановило единые программы и требования в обучении. В 1930-е годы индивидуализация рассматривается в основном как инструмент предупреждения неуспеваемости. С этого момента почти на два десятилетия были приостановлены теоретические

исследования и практические работы в области индивидуализации и дифференциации. Скудное разнообразие реализовывалось через изучение дополнительного материала в школьных кружках и внешкольных образовательных учреждениях.

В 50–60 годах XX века число исследований, связанных с индивидуализацией обучения, связью обучения с жизнью увеличивается, создаются экспериментальные классы с производственным обучением. В это же время Н.К. Гончаровым предлагается ввести дифференциацию, выражающуюся в создании на старшей ступени школы отделений: физико-технического; химико-технического; естественно-агрономического; гуманитарного. Экспериментальная проверка нововведений в школах, проводившаяся в 1968–1969 годах, оказалась успешной. Начали появляться классы с углубленным изучением отдельных предметов, которые зарекомендовали себя как эффективные формы дифференцированного обучения в различных областях науки и техники. Следует отметить, что в основном такая дифференциация ориентировалась на школьников, у которых уже сформировался познавательный интерес и проявились определенные способности [5]. С начала 1990-х годов в нашей стране начали открываться новые типы школ (лицеи, гимназии, специализированные школы), ориентированные на изучение учащимися выбранных ими направлений, связанных с будущей профессиональной подготовкой. Закон Российской Федерации «Об образовании» (1992 г.) закрепил вариативность учебных планов, программ, учебников и разрешил организацию разнообразных типов общеобразовательных учреждений.

К началу XXI века отечественными учеными были разработаны теоретические основы дифференциации обучения (Б.Г. Ананьев, Ю.К. Бабанский, А.А. Бодалев, Л.И. Божович, К.М. Гуревич, И.В. Дубровина, А.Н. Леонтьев, Х.И. Лийметс, Е.А. Певцова, И.Э. Унт, С.Г. Шаповаленко, Н.М. Шахмаев, Г.И. Щукина и др.), исследованы многие проблемы дифференцированного содержания обучения (Л.Л. Босова, Г.В. Дорофеев, Т.Б. Захарова, Л.В. Кузнецова, В.М. Монахов, В.А. Орлов, Н.Н. Самылкина, С.Б. Суворова, С.И. Шварцбург, В.В. Фирсов и др.). При этом во многих исследованиях под дифференциацией обучения понимается «система обучения, которая обеспечивает каждому школьнику определенный минимум общеобразовательной подготовки и одновременно дает право и гарантированную возможность уделять преимущественное внимание в обучении тем направлениям, которые в наибольшей степени отвечают его склонностям и возможностям» [6].

С введением ФГОС ОО (2009 г.) происходит усиление акцентов на возможности и ценность именно индивидуального развития, появляется возможность выбора индивидуального учебного плана и различных программ (в том числе адаптированных), т. е. самостоятельного проектирования собственной образовательной траектории на всех уровнях общего образования [7]. Таким образом, происходит усиление личностно-ориентированного обучения, индивидуализации образовательного процесса посредством профильной и уровневой дифференциации.

В обновленных ФГОС общего образования представлены требования к трем взаимосвязанным группам результатов: личностным, метапредметным и предметным. Такое деление отражает общепризнанный мировым научно-педагогическим сообществом подход к распределению педагогических целей по следующим областям: когнитивная (познавательная) область, психомоторная область, активная эмоционально-ценностная область (Б. Блум, Д. Кратвиль). Аналогичные подходы отчетливо прослеживаются

и в отечественной педагогической науке. Например, О.Е. Лебедев и И.Я. Лернер в своих трудах выделяют развитие знаний (О.Е. Лебедев), знания о природе, обществе, технике, человеке (И.Я. Лернер); развитие умений и навыков (О.Е. Лебедев), опыт осуществления способов деятельности, в том числе творческий (И.Я. Лернер); развитие системы отношений (О.Е. Лебедев), эмоционально-чувственный опыт (И.Я. Лернер) [8]. Образовательные результаты по всем трем областям формируются во взаимосвязи, однако наибольшее внимание в исследованиях прошлых лет уделялось именно результатам в когнитивной области, применительно к которым во второй половине прошлого века в педагогике стал использоваться термин «таксономия».

Традиционно описание результатов образовательного процесса представляет собой структурированное множество объектов учебной деятельности, существующее в рамках определенного содержания образования. Следует учитывать, что при описании результатов интегративно используются морфологическая и функциональная структуры. Морфологическая структура описания результатов в соответствии с требованиями обновленного ФГОС общего образования представляет собой иерархию предъявления учебного содержания последовательно по годам изучения, описанную не только в терминах учебного предмета, но и в видах деятельности обучающихся. Функциональная структура описания результатов позволяет учесть функции объектов изучения и акцентировать внимание на прикладное использование этого функционала при описании видов деятельности, в которых они проявляются. Именно здесь мы подходим к уровням усвоения содержательных и деятельностных составляющих образовательной программы. Функционально-морфологической структуры описания образовательных результатов недостаточно для их использования в качестве критериев оценивания достижений обучающихся; необходимо указывать качественные характеристики проявления результатов образовательной деятельности [9].

В современной психолого-педагогической литературе выделяются качественные характеристики в овладении учебным материалом, которые разными авторами обозначаются как уровни учебной деятельности или как уровни усвоения. Существуют различные трактовки уровней усвоения учебного материала, некоторые из них представлены в табл. 1 [10].

Таблица 1

Уровни усвоения учебного материала

	Б. Блум	В.П. Беспалько	О.Е. Лебедев
1.	Знание	Ученический (узнавание)	Информированность
2.	Понимание	Алгоритмический (решение типовых задач)	Функциональная грамотность
3.	Применение	Эвристический (выбор действия)	Грамотность
4.	Анализ	Творческий (поиск действия)	Компетентность
5.	Синтез		
6.	Оценка		

На практике в отечественной школе чаще всего выделяют три уровня усвоения учебного материала: 1) восприятия, осмысления и запоминания; 2) применения знаний в сходных ситуациях, т. е. по образцу; 3) применения знаний в новой ситуации, требующей проявления творческой деятельности.

Следствием различия во взглядах на уровни усвоения учебного материала стало развитие разных подходов к решению на практике таких проблем, как конкретизация образовательных результатов в терминах внешней деятельности обучающихся и отнесение их к определенному уровню усвоения; возможность комплексной оценки сформированных образовательных результатов (компетенций) обучающихся (предметных, общеучебных и личностных) и пр.

Сложность описания образовательных результатов для их однозначного оценивания была показана многими учеными также, как и необходимость их описания в качестве критериев оценивания и постоянного обновления с учетом внутренних (личностные особенности психофизического развития) и внешних (изменения потребностей социума) факторов. По мнению академика А.А. Кузнецова, «качество знаний рассматривается как важнейшая, наиболее существенная характеристика учебной подготовки, интегральный показатель усвоения учебного материала. Если под качеством результатов обучения в самом общем виде понимать особенности усвоения знаний, обусловленные учебной деятельностью учащихся, то указание на эти особенности усвоения является необходимой формой конкретизации планируемых результатов обучения» [5, с. 106]. Качество знаний обучающихся должно быть описано на предметно-содержательном уровне (полнота и системность); на содержательно-деятельностном уровне (прочность и действенность); на содержательно-личностном уровне (активность, самостоятельность, продуктивность, гибкость, осознанность мыслительной деятельности и др.) [9]. Качества результатов усвоения учебного материала выступают в качестве критериев оценки образовательных достижений учащихся. При этом под оценкой уровня усвоения учебного содержания следует понимать указание на качество усвоения знаний и формирования умений, которое формируется на основании сопоставления предполагаемых и достигнутых результатов обучения.

В современных условиях в основном общем образовании для таких учебных предметов, как математика, информатика, физика, химия и биология заданы два уровня изучения — базовый и углубленный, представление образовательных результатов на которых происходит путем описания качественных характеристик уровней усвоения — предметно-содержательного, содержательно-деятельностного и содержательно-личностного. При конкретизации предметных результатов по годам обучения учитываются потребности и реальные учебные возможности обучающихся определенного возраста, уровня учебной подготовки, особенностей здоровья и развития и т. д. Такое описание соответствует компетентностному подходу, принятому в международном образовательном пространстве, обеспечивая вместе с тем учет национальных приоритетов.

Особо обратим внимание на необходимость различать кажущиеся синонимичными понятия «уровень усвоения» в общедидактическом аспекте, описанном выше, и «уровень обучения» или «уровень изучения предмета» — нормативно закрепленное понятие для выстраивания образовательной траектории обучающегося при реализации дифференцированного обучения. По ФГОС ООО на каждом уровне изучения предмета — базовом и углубленном — описываются образовательные результаты на всех уровнях

усвоения. Понятие «уровень обучения» (по контексту — это базовый и углубленный уровни изучения предмета) также не следует подменять понятием «уровень образования» — законодательно введенным понятием, означающим освоение образовательных программ определенного уровня (начальное общее образование, основное общее образование и т. д.).

Уточним сущность еще несколько понятий, которые зачастую употребляются педагогами как синонимичные, хотя и не являются таковыми. Это «дифференцированный подход», «дифференцированное обучение», «индивидуализация», «разноуровневое (многоуровневое) обучение» и «персонализация».

Дифференцированное обучение в общем образовании реализуется на базе организационно-управленческих, правовых аспектов функционирования организации как предоставленная обучающимся выбрать предметы и уровень их изучения (базовый или углубленный) [11]. Дифференцированный подход следует понимать как реализацию индивидуализации в образовании, т. е. учета индивидуальных запросов, потребностей, способностей при получении образования. Так, на уровне начального общего образования применяется именно «дифференцированный подход», поскольку здесь в отличие от основного общего и среднего общего образования нет возможности выбора предметного содержания и уровня изучения выбранного предмета. На уровне начального общего образования средством реализации индивидуального подхода выступает только разноуровневое (многоуровневое) обучение на уроке внутри класса (группы).

Используя понятие «разноуровневое (многоуровневое) обучение» на уроке, надо отчетливо представлять аспект его применения. Мы говорим об изучении одинакового предметного содержания на одном из выбранных уровней изучения предмета, но использовании при этом контента, детализированного в разной степени подробности, заданий разной трудности, различных дополнительных материалов, требующих разной степени погружения в него. Внешняя нормативно закрепленная дифференциация в виде возможности выбора предметов для изучения на определенном уровне создает благоприятные предпосылки для целенаправленной индивидуализации обучения, но не решает ее до конца. Сформированные классы (группы) по принципу примерной «одинаковости», т. е. сходной обученности, учебных возможностей и целевых установок после начала обучения очень быстро становятся неоднородными в своих учебных возможностях. Учителю необходимо использовать различный по трудности выполнения учебный материал и регулировать активность обучающихся на уроке. В исследовании В.К. Шишмаренкова используется понятие «разноуровневость в обучении», когда в любой группе, даже специально отобранной по гомогенному признаку, в короткий промежуток времени происходит выделение учащихся с разными возможностями усвоения учебного материала (сильные, средние, слабые). Этот фактор требует учета при организации процесса обучения, поэтому «разноуровневое обучение» — это организация образовательного процесса с учетом разноуровневости в обучении» [12, с. 4]. Если дифференциация позволяет создавать группы или классы для изучения выбранного содержания на выбранном уровне, то разноуровневое (многоуровневое) обучение предмету на уроке внутри таких групп позволяет ориентироваться на индивидуальные возможности и потребности обучающихся в освоении учебного материала.

Возникает необходимость уточнения сущности понятия «индивидуализация» уже как обобщающего понятия. «Индивидуализация» рассматривается в качестве глобальной

цели функционирования системы образования, поскольку представлена в качестве основного принципа государственной политики через «адаптивность системы образования к уровню подготовки, особенностям развития, способностям и интересам человека» [13, с. 3, ч. 1, п. 8]. Индивидуализация обеспечена многообразием реализуемых образовательных программ, видов и форм обучения, а также используемыми образовательными технологиями, к которым иногда относят разноуровневое обучение в группах.

В международной образовательной практике понимание разноуровневого обучения сходно с пониманием дифференциации обучения и оценивания, которое реализуется через усвоение определенного контента; обработки, конструирования или осмысления идей и разработки учебных материалов и оценочных мер, чтобы все учащиеся в классе могли эффективно учиться независимо от различий в их способностях [14].

Таким образом, в системе общего образования на уровне начального общего образования используется дифференцированный подход. Изучение обязательных предметов реализуется на одном нормативно закреплённом уровне обучения, но с использованием разноуровневых теоретических и практических заданий на уроке для учета индивидуальных возможностей обучающихся. При этом индивидуальные интересы и потребности обучающихся реализуются возможностью выбора ими внеурочной деятельности и программ дополнительного образования. На следующих уровнях образования (основном общем и среднем общем образовании) реализуется дифференцированное обучение на двух уровнях изучения предмета — базовом и углубленном. При этом на любом из уровней изучения предмета можно использовать разноуровневое (многоуровневое) обучение на уроке в группе обучающихся с разными возможностями для освоения предметного содержания. Оценивание образовательных результатов производится во внутришкольном мониторинге. В обновленном ФГОС общего образования результаты освоения основной образовательной программы представлены интегративно (метапредметные и предметные) на двух уровнях изучения с возможностью их поэтапного достижения в ходе разноуровневого обучения.

260

В настоящее время процессы цифровой трансформации характеризуются переходом от индивидуализации к персонализации образовательного процесса, предусматривающем субъектность обучающегося (включенность во все аспекты собственного образования), использование персонализированного контента, траекторию и цифровые инструменты его освоения, а также персонализированное оценивание как оценивание собственных приращений в обучении относительно своих прежних достижений. По сути мы получаем ориентацию на персональные нормы конкретного учащегося, его образовательные возможности в текущий момент времени. Оценивается темп усвоения, объем усвоенного материала по сравнению со стартовым уровнем, качественные характеристики в овладении материалом. Все это позволит учителю реализовать разноуровневое обучение в группе (классе) и при этом выйти на нормативные требования к результатам освоения образовательной программы на выбранном уровне изучения предмета. Учащийся ориентируется на предметные и метапредметные результаты основных образовательных программ соответствующего уровня, выстраивает собственный рейтинг достижений. У учащегося есть возможность выбора своих итоговых результатов, которые не могут быть ниже описанных для выбранного им уровня изучения предмета в рамках соответствующего

уровня образования, но верхняя планка у каждого своя. Система внутреннего мониторинга образовательной организации призвана предусматривать оценку динамики учебных достижений обучающихся во всем многообразии образовательной деятельности. Стоит обратить внимание на то, что метапредметные результаты буквально «встроены» в предметные результаты. Таким образом, разноуровневое обучение в группах и внутреннее оценивание учитывает особенности образовательного процесса, потребности каждого учащегося и позволяет выйти на достижение предметных и метапредметных результатов согласно требованиям ФГОС ООО, контролируемым во внешнем оценивании (ОГЭ, ВПР).

Описание образовательных результатов по тематическому разделу «Алгоритмы и программирование» для уровня основного общего образования

Принципиальное различие в требованиях к предметным результатам изучения информатики в основной школе состоит в следующем:

- на базовом уровне речь идет, как правило, о формировании общих представлений об изучаемых понятиях и методах, о воспроизведении нескольких базовых алгоритмов, о практических навыках использования программного обеспечения;
- углубленный уровень характеризуется свободным оперированием понятиями, алгоритмами, методами; освоением обучающимися более широкого содержания, связанного с представлением информации, элементов математической логики, теории графов и компьютерного моделирования; существенно большим учебным временем, выделяемым на освоение программирования.

Тематический раздел «Алгоритмы и программирование» направлен на развитие алгоритмического мышления, разработку алгоритмов, формирование навыков реализации программ на языках программирования высокого уровня [15]. В требованиях ФГОС ООО, связанных с освоением тематического раздела «Алгоритмы и программирование» (табл. 2), отчетливо видно, что базовый уровень изучения соответствующего материала ориентирован на формирование представлений о программировании как базовой цифровой компетенции (программирование для жизни); цель углубленного уровня — формирование навыков программирования как основы специальной цифровой компетенции, необходимой для профессиональной деятельности.

Таблица 2

Требования к базовому и углубленному уровню изучения тематического раздела «Алгоритмы и программирование»

№п/п	Базовый уровень	Углубленный уровень
1	Развитие алгоритмического мышления как необходимого условия профессиональной деятельности в современном обществе; понимание сущности алгоритма и его свойств	Наличие развитого алгоритмического мышления как необходимого условия профессиональной деятельности в современном обществе; свободное оперирование понятиями «исполнитель», «алгоритм», «программа», понимание разницы между употреблением этих терминов в быденной речи и в информатике; умение выбирать подходящий алгоритм для решения задачи

Таблица 2. Окончание

№п/п	Базовый уровень	Углубленный уровень
2	Умение составлять, выполнять вручную и на компьютере несложные алгоритмы для управления исполнителями (Черепашка, Чертежник); создавать и отлаживать программы на одном из языков программирования (Python, C++, Паскаль, Java, C#, Школьный алгоритмический язык), реализующие несложные алгоритмы обработки числовых данных с использованием циклов и ветвлений; умение разбивать задачи на подзадачи, использовать константы, переменные и выражения различных типов (числовых, логических, символьных); анализировать предложенный алгоритм, определять, какие результаты возможны при заданном множестве исходных значений;	Свободное оперирование понятиями: переменная, тип данных, операция присваивания, арифметические и логические операции, включая операции целочисленного деления и остатка от деления; умение создавать программы на современном языке программирования общего назначения: Python, C++ (JAVA, C#), реализующие алгоритмы обработки числовых данных с использованием ветвлений, циклов со счётчиком, циклов с условиями, подпрограмм (алгоритмы проверки делимости одного целого числа на другое, проверки натурального числа на простоту, разложение на простые сомножители, выделение цифр из натурального числа, поиск максимумов, минимумов, суммы числовой последовательности и т. п.); владение техникой отладки и выполнения полученной программы в используемой среде разработки.

В федеральных рабочих программах по информатике предметные результаты по разделу «Алгоритмы и программирование» конкретизируются в описании качественных характеристик уровней усвоения в деятельностном формате. В табл. 3 представлены предметные результаты по 7–9 классам.

Таблица 3

Предметные результаты изучения тематического раздела «Алгоритмы и программирование» по классам основного общего образования

Базовый уровень	Углубленный уровень
<i>7 класс</i>	
	<ul style="list-style-type: none"> ▪ раскрывать смысл понятий «исполнитель», «алгоритм», «программа», понимая разницу между употреблением этих терминов в обыденной речи и в информатике; ▪ описывать алгоритм решения задачи различными способами, в том числе в виде блок-схемы; ▪ разбивать задачи на подзадачи; составлять, выполнять вручную и на компьютере несложные алгоритмы с использованием ветвлений, циклов и вспомогательных алгоритмов для управления исполнителями, такими как Робот, Черепашка, Чертежник;
<i>8 класс</i>	
<ul style="list-style-type: none"> ▪ раскрывать смысл понятий «исполнитель», «алгоритм», «программа», понимая разницу между употреблением этих терминов в обыденной речи и в информатике; 	

Таблица 3. Продолжение

Базовый уровень	Углубленный уровень
<i>8 класс</i>	
<ul style="list-style-type: none"> ▪ описывать алгоритм решения задачи различными способами, в том числе в виде блок-схемы; ▪ составлять, выполнять вручную и на компьютере несложные алгоритмы с использованием ветвлений и циклов для управления исполнителями, такими как Робот, Черепашка, Чертёжник; 	
	<ul style="list-style-type: none"> ▪ уметь выбирать подходящий алгоритм для решения задачи;
<ul style="list-style-type: none"> ▪ использовать константы и переменные различных типов (числовых, логических, символьных), а также содержащие их выражения; использовать оператор присваивания; 	<ul style="list-style-type: none"> ▪ свободно оперировать понятиями: переменная, тип данных, операция присваивания, арифметические и логические операции, включая операции целочисленного деления и остатка от деления; ▪ использовать константы и переменные различных типов (числовых — целых и вещественных; логических; символьных), а также содержащие их выражения; использовать оператор присваивания;
<ul style="list-style-type: none"> ▪ использовать при разработке программ логические значения, операции и выражения с ними; 	<ul style="list-style-type: none"> ▪ записывать логические выражения на изучаемом языке программирования;
<ul style="list-style-type: none"> ▪ анализировать предложенные алгоритмы, в том числе определять, какие результаты возможны при заданном множестве исходных значений; 	<ul style="list-style-type: none"> ▪ анализировать предложенные алгоритмы, в том числе определять, какие результаты возможны при заданном множестве исходных значений; определять возможные входные данные, приводящие к определённому результату;
<ul style="list-style-type: none"> ▪ создавать и отлаживать программы на одном из языков программирования (Python, C++, Паскаль, Java, C#, Школьный алгоритмический язык), реализующие несложные алгоритмы обработки числовых данных с использованием циклов и ветвлений, в том числе реализующие проверку делимости одного целого числа на другое, проверку натурального числа на простоту, выделения цифр из натурального числа. 	<ul style="list-style-type: none"> ▪ создавать и отлаживать программы на современном языке программирования общего назначения (Python, C++, Java, C#), реализующие алгоритмы обработки числовых данных с использованием ветвлений (нахождение минимума и максимума из двух, трёх и четырёх чисел; решение квадратного уравнения, имеющего вещественные корни); ▪ создавать и отлаживать программы на современном языке программирования общего назначения из приведённого ниже списка, реализующие алгоритмы обработки числовых данных с использованием циклов с переменной, циклов с условиями (алгоритмы нахождения наибольшего общего делителя двух натуральных чисел; проверки натурального числа на простоту; разложения натурального числа на простые сомножители; выделения цифр из натурального числа); ▪ создавать и отлаживать программы на современном языке программирования общего назначения из приведённого ниже списка, реализующие алгоритмы обработки потока данных (вычисление количества, суммы, среднего арифметического, минимального и максимального значений элементов числовой последовательности, удовлетворяющих заданному условию);

Таблица 3. Окончание

Базовый уровень	Углубленный уровень
	<ul style="list-style-type: none"> ▪ создавать и отлаживать программы на современном языке программирования общего назначения из приведённого ниже списка, реализующие алгоритмы обработки символьных данных (посимвольная обработка строк, подсчёт частоты появления символа в строке; использование встроенных функций для обработки строк); ▪ создавать и отлаживать программы, реализующие типовые алгоритмы обработки одномерных числовых массивов, на одном из языков программирования из приведённого ниже списка: заполнение числового массива случайными числами, в соответствии с формулой или путём ввода чисел; линейный поиск заданного значения в массиве; подсчёт элементов массива, удовлетворяющих заданному условию; нахождение суммы, минимального и максимального значений элементов массива;
<i>9 класс</i>	
<ul style="list-style-type: none"> ▪ разбивать задачи на подзадачи; составлять, выполнять вручную и на компьютере несложные алгоритмы с использованием ветвлений, циклов и вспомогательных алгоритмов для управления исполнителями, такими как Робот, Черепашка, Чертёжник; 	<ul style="list-style-type: none"> ▪ разбивать задачи на подзадачи; создавать и отлаживать программы на современном языке программирования общего назначения (Python, C++, Java, C#), реализующие алгоритмы обработки числовых данных с использованием подпрограмм (процедур, функций);
<ul style="list-style-type: none"> ▪ составлять и отлаживать программы, реализующие типовые алгоритмы обработки числовых последовательностей или одномерных числовых массивов (поиск максимумов, минимумов, суммы или количества элементов с заданными свойствами) на одном из языков программирования (Python, C++, Паскаль, Java, C#, Школьный алгоритмический язык). 	<ul style="list-style-type: none"> ▪ составлять и отлаживать программы на современном языке программирования общего назначения, реализующие несложные рекурсивные алгоритмы; ▪ составлять и отлаживать программы на современном языке программирования общего назначения из приведённого выше списка, реализующие алгоритмы сортировки массивов, двоичного поиска в упорядоченном массиве; ▪ составлять и отлаживать программы на современном языке программирования общего назначения из приведённого выше списка, реализующие основные алгоритмы обработки двумерных массивов (матриц): заполнение двумерного массива случайными числами и с использованием формул; вычисление суммы элементов, максимального и минимального значений элементов строки, столбца, диапазона; поиск заданного значения; ▪ составлять и отлаживать программы на современном языке программирования общего назначения из приведённого выше списка, реализующие простые приёмы динамического программирования.

Даже беглый взгляд на предметные результаты изучения тематического раздела «Алгоритмы и программирование» учебного предмета «Информатика» позволяет сделать следующие выводы:

- на углубленном уровне изучение раздела начинается раньше; его содержание значительно шире;
- на базовом уровне предпочтение отдается работе с исполнителями, язык управления которыми может быть избран в качестве единственного языка программирования, предлагаемого ученикам для знакомства; на углубленном уровне речь идет об изучении основ одного или нескольких языков программирования, широко используемых профессионалами в сфере информационных технологий;
- результаты углубленного уровня изучения тематического раздела «Алгоритмы и программирование» не просто включают в себя соответствующие результаты базового уровня, но требуют от обучающегося принципиально иного уровня освоения содержания обучения.

Подкрепим последнее утверждение примером, рассмотрев следующие сходные требования к предметным результатам базового и углубленного уровней. Одним из результатов изучения курса информатики на базовом уровне является формирование у обучающихся умения «использовать при разработке программ логические значения, операции и выражения с ними»; на углубленном уровне освоения предмета соответствующий результат звучит как умение «записывать логические выражения на изучаемом языке программирования».

Для проверки предметного результата базового уровня ученикам может быть предложено задание 1, в котором требуется использовать подходящую логическую операцию в готовом логическом выражении, записанном на изучаемом языке программирования.

Задание 1. Дан код программы на Python. Какую логическую операцию необходимо использовать в логическом выражении вместо знака #? Дополните код логической операцией. Получите результат выполнения программы.

Чтобы определить существование треугольника по заданным сторонам, необходимо проверить истинность условия «любая сторона треугольника меньше суммы двух других»:

```
print('A, B, C — стороны треугольника')
X = int(input('Введите A: '))
Y = int(input('Введите B: '))
Z = int(input('Введите C: '))
print('Каждая сторона треугольника меньше суммы двух других')
if X < Y + Z # Y < X + Z # Z < X + Y:
print('Треугольник существует')
else:
print('Треугольник не существует')
Ответ: and
```

265

Задание 2 относится к проверке предметного результата углубленного уровня; здесь обучающемуся предстоит составить логическое выражение самостоятельно.

Задание 2. Дан код программы на Python. Составьте логическое выражение, которое необходимо использовать вместо знаков #. Дополните код логическим выражением. Получите результат выполнения программы.

Для проверки равнобедренности треугольника достаточно равенства двух любых сторон, поэтому все возможные случаи равенств связываем в условной конструкции с помощью оператора or.

```

print(A, B, C — стороны треугольника)
X = int(input('Введите A: '))
Y = int(input('Введите B: '))
Z = int(input('Введите C: '))
print('В равнобедренном треугольнике есть две равные
стороны')
if #####
print('Треугольник равнобедренный')
else:
print('Треугольник не равнобедренный')
Ответ: X == Y or Y == Z or Z == X:

```

При оценивании результатов выполнения заданий, предполагающих разный уровень изучения предмета, используются критерии, учитывающие разные уровни усвоения учебного материала (см. табл. 4).

Таблица 4

Пример критериев оценивания заданий на разных уровнях изучения информатики

№	Базовый уровень	Баллы	Углубленный уровень	Баллы
1.	Правильно выбрана логическая операция. Программа работает корректно.	5	Составлено правильно логическое выражение. Программа работает корректно. Неточности в программе устранены самостоятельно.	5
2.	Правильно выбрана логическая операция. Для устранения неточностей в программе требуется помощь.	4	Составлено правильно логическое выражение. Для обнаружения неточностей/ошибок в программе требуется помощь.	4
3.	Выбрана неправильная логическая операция. При выполнении кода программы выбор логической операции исправлен, но требуется помощь в запуске программы.	3	Вызывает затруднение использование логических выражений в программе. Требуется помощь при запуске и корректировке кода программы.	3
4.	Задание вызывает затруднение.	0	Задание вызывает затруднение.	0

266

Реализация разноуровневого обучения программированию в курсе информатики основного общего образования

Реализация разноуровневого обучения программированию в 7–9 классах предполагает использование широкого диапазона активных методов обучения и образовательных технологий на их основе, а также встроенных в образовательный процесс оценочных процедур (прежде всего формирующего оценивания).

Активные методы обучения характеризуются высокой степенью включенности обучаемого в учебный процесс, т. е. деятельность обучаемого носит продуктивный, творческий, поисковый характер. Это происходит за счет целенаправленного создания и погружения обучающегося в условия, когда он становится активным участником образовательного процесса длительное время, в течение всего занятия или выполнения проекта.

Существует множество классификаций активных методов обучения; наиболее часто используется классификация по ведущей деятельности обучающихся. В ней предполагается разделение активных методов на четыре группы: дискуссионные (эвристическая беседа, дискуссия, дебаты, форсайт-сессия), игровые (деловые игры, конкурсы), рейтинговые (мозговой штурм, эстафета, хакатон, проект) и тренинговые (кейс-технологии, тренинги).

Перечисленные выше активные методы обучения лежат в основе следующих образовательных технологий, наиболее успешно применяемых учителями:

1) технология геймификации (когда используются игровые элементы вне игрового контекста, результатом может быть любой продукт, в том числе и игра);

2) кейс-технология (когда на основе анализа и обсуждения конкретных ситуаций вырабатываются различные стратегии решения проблемы);

3) проектная технология (когда детальную разработку проблемы (предпочтительно в группе) завершают практическим результатом);

4) смешанное обучение (когда есть возможность гибкого изменения форматов деятельности обучающихся: малые группы, фронтальная работа, сочетание традиционного и дистанционного обучения).

При изучении информатики в 5–6 классах рекомендуется использовать среды визуального программирования, например, Scratch, ПиктоМир, Kodu, Blockly Game, Minecraft Education и др. Подобные среды имеют свои преимущества и недостатки, но общее у них одно — атмосфера игры. Такой эффект достигается, прежде всего, за счет использования блочного программирования, когда текст программы не пишется на языке программирования, а составляется с помощью имеющихся заготовок — графических блоков. Обучающиеся, создавая простейшие программы, управляют картинками, анимацией, звуком. Итогом самых первых занятий в вышеперечисленных средах программирования могут стать созданные своими руками мультфильм или игра.

Ощутимый результат, простота составления программ делают данные среды эффективным инструментом раннего обучения программированию, способствуя развитию алгоритмического, логического и системного мышления обучающихся, становления творческого подхода к решению задач; формированию культуры пользования информационными и коммуникационными технологиями, умений и навыков проектной и исследовательской деятельности; воспитанию интереса к программированию как к ключевой технологии XXI века, стремления использовать полученные знания, умения и навыки в учебной деятельности и в повседневной жизни [16].

После работы в средах визуального программирования в 6–7 классах целесообразен переход к знакомству со средами текстового программирования (Школьный алгоритмический язык, Python) при сохранении элементов технологии геймификации, позволяющих сохранить повышенную мотивацию к процессу обучения. Именно игровые элементы (правила игрового пространства, баллы, бейджи, рейтинг-листы) позволяют

поддерживать высокую мотивацию длительный период, пока тема не будет освоена достаточно полно. При использовании игровых элементов в программировании создается специальное общее игровое пространство, накапливаются очки, подсчитываются рейтинги участников [1]. При этом совсем не обязательно конечной целью ставить создание готового программного продукта; на данном этапе обучения достаточно вовлечения обучающихся в деятельность по созданию кода для решения задачи доступными средствами.

При изучении нового материала целесообразно использование дискуссионных (проблемных, эвристических) методов обучения, предполагающих обмен мнениями участников по рассматриваемой теме, при необходимости — управление со стороны учителя процессами выработки и принятия группового решения обучающимися. Лучше всего использовать эвристическую беседу, проблемную дискуссию или поисковый диспут, но не весь урок, а в строго ограниченный временной период по заранее подготовленным вопросам или опережающему домашнему заданию. Такое обсуждение нового материала формирует также «мягкие» навыки: деловой коммуникации, анализа ситуации, эмоционального интеллекта и настрой на выработку коллективного решения проблемы, которые ценятся в профессиональной среде программистов. При углубленном изучении информатики возможно проведение круглого стола в виде форсайт-сессии или дебатов отдельным уроком или его основной частью, если предусматривается решение задач или практическая работа в конце урока или новый материал в начале урока. Различаются они тем, что форсайт-сессия предполагает взгляд в будущее, его прогнозирование с опорой на лучшие решения/практики, а в дебатах отстаивают противоположные точки зрения на возможные варианты развития событий.

Основное внимание изучению тематического раздела «Алгоритмы и программирование» и на базовом, и на углубленном уровне изучения информатики уделяется в 8 классе. И в том, и в другом случае рассматриваются базовые алгоритмические конструкции, формируются навыки создания и отладки программ на изучаемом языке программирования. Временные ресурсы углубленного курса информатики в основной школе позволяют задействовать в учебном процессе не отдельные задачи, а их специальным образом сформированные последовательности [17]. При этом не обязательно решать с учениками всю последовательность задач сразу, одну задачу за другой; к задачам можно возвращаться на протяжении изучения нескольких тем (например, при программировании линейных, разветвляющихся и циклических алгоритмов). Главное — создать условия, чтобы обучающиеся, получив минимальный опыт, имели возможность опираться на него, «узнавать» знакомые алгоритмы и применять их при решении других задач; не просто получать и накапливать знания, а самостоятельно изобретать алгоритмы, развивать свое мышление, совершенствовать свои способности.

При решении задач разными способами, разборе наиболее трудных задач на уроках можно реализовать соревновательные преимущества рейтинговых методов обучения. При поиске ошибок в кодах или дописывании «утраченной» части кода можно использовать эстафету, которая поможет удержать темп урока и внимание всех его участников до конца работы. Хорошим помощником учителя для реализации индивидуальных образовательных запросов обучающихся в процессе освоения программы

рования может стать искусственный интеллект, который уже сегодня позволяет генерировать разнообразные задания, консультировать учеников и проверять правильность программного кода.

Углубленный уровень изучения информатики позволяет использовать при освоении программирования кейсы, представляющие собой описание определённой проблемной ситуации, подготовленной для образовательных целей. С помощью кейса формируются навыки анализа информации, ее обобщения, выявления и формулирования проблемы и выработки различных альтернатив её решения. Кейсы можно использовать как для обучения, так и для диагностики функциональной грамотности или компетенций в определенной сфере. При подготовке ситуаций кейса можно сразу ориентироваться на заданные уровни сформированности функциональной грамотности. К первому уровню относятся задания кейса, в которых анализируется ситуация и ее решение. Ученику требуется определить, подходит ли это решение, возможно ли использовать более рациональное решение. Ко второму уровню относятся задания кейса, в которых проблема определена в явном виде; надо найти самостоятельно решение проблемы и обосновать его. К третьему уровню относятся кейсы с описанием ситуации, где проблеме надо сформулировать явно, а затем найти ее решение, возможно не одно. В условиях ограниченного времени на изучение информатики на базовом уровне возможно комплексное использование кейсов. С использованием кейса можно изучить материал, а дополнительные вопросы к ситуации помогут диагностировать его усвоение. Так проявляются стимулирующее, корректирующее или развивающее воздействие на личность и поведение участников [18].

Из практических работ, выполняемых обучающимися в процессе освоения тематического раздела «Алгоритмы и программирование», за счет новых заданий к выполненной работе, использования дополнительных параметров в заданиях, подключения или перепрограммирования дополнительных устройств могут быть «выращены» различные учебные или исследовательские проекты. Важно учитывать, что проекты, которые могут быть реализованы в основной школе, имеют, прежде всего, формирующее значение, они обеспечивают накопление и развитие личностного социального опыта ребенка, что одинаково актуально как для углубленного, так и для базового уровня изучения информатики. В первом случае в проектную деятельность могут быть вовлечены все обучающиеся, во втором — отдельные, наиболее мотивированные школьники. Роли обучающихся основной школы в большинстве своем исполнительские, с ограниченным функционалом. Например, в межпредметных информационно-экологических проектах могут быть статистики (собирающие данные), аналитики (делающие выводы), инженеры (реализующие новые решения). Но по мере накопления опыта проектной деятельности могут выделяться явные лидеры, претендующие на роль менеджера всего проекта. Возникают иные взаимоотношения внутри группы, новые стимулы для работы.

Учитывая специфику возраста в основной школе, можно использовать все преимущества групповой проектной работы. Групповая работа позволяет учитывать индивидуальные особенности каждого обучающегося, производить разделение труда и распределение ролей, а достигаемый результат будет значительно выше у группы, чем у каждого в отдельности. При групповой организации работы формируются необходимые коммуникативные компетенции. К тому же при такой организации формируется коллективная

ответственность и обеспечивается взаимопомощь как со стороны одноклассников, так и со стороны педагога. Результаты проектной деятельности обязательно представляются на внутреннее и внешнее оценивание.

Углубленный уровень изучения информатики подразумевает широкое участие обучающихся в олимпиадном движении. Практика подготовки к олимпиадам по информатике показала, что программистские навыки целесообразно оттачивать в различных индивидуальных и групповых соревнованиях, в том числе в хакатонах по программированию, помня о том, что в профессиональных сообществах также существуют рейтинги (индивидуальные и командные), есть правила участия, похожие на правила игрового пространства, где можно набирать или терять баллы (очки). В качестве командных соревнований и тренировок внутри класса, школы или города наиболее приемлемым является мини-хакатон. Для организации хакатона нет возрастных ограничений. Возраст обучающихся надо учитывать при постановке командного задания, выборе программного пакета и времени работы. Чем ниже возраст, тем меньше времени на создание кода и тем более интерактивной должна быть среда программирования.

Выводы

Разноуровневое обучение предмету приобрело особую актуальность в условиях дифференциации содержания обучения, декларированной обновленными федеральными государственными образовательными стандартами общего образования. На примере тематического раздела «Алгоритмы и программирование» мы представили возможные подходы к реализации разноуровневого формирования навыков программирования как цифровой компетенции: «для жизни» при базовом уровне и специальной на углубленном уровне изучения учебного предмета «Информатика». Методика разноуровневого обучения представляет интерес для всех педагогов, поскольку позволяет понять, как формировать необходимые предметные и метапредметные результаты для их оценивания при переходе к персонализированному обучению, какие методы обучения и образовательные технологии следует использовать для достижения наилучшего результата.

СПИСОК ИСТОЧНИКОВ И ЛИТЕРАТУРЫ

1. Мишин, В.А. Использование платформы Minecraft: Education Edition на уроках информатики // Информатика в школе. 2021. № 3 (166). С. 50–59.
2. Kluzer, S., Priego, L.P. DigComp into Action: Get Inspired, Make It Happen. S. Carretero, Y. Punie, R. Vuorikari, M. Cabrera, and O'Keefe, W. (Eds.). JRC Science for Policy Report, EUR 29115 EN. Publications Office of the European Union, Luxembourg, 2018. 50 p.
3. Босова, Л.Л. Цифровые навыки современного школьника и возможности их формирования в школьном курсе информатики // Информатика в школе. 2020. Т. 1. №. 7. С. 5–9.
4. Кантрев, П.Ф. Избранные педагогические сочинения / под ред. А.М. Арсеньева. М.: Педагогика, 1982. 704 с.
5. Кузнецов, А.А., Захарова, Т.Б., Захаров, А.С. Общая методика обучения информатике: учебное пособие. Ч. 1. М.: Прометей, 2016. 300 с.
6. Захарова, Т.Б. Профильная дифференциация обучения информатике на старшей ступени школы: монография. М.: Б. и., 1997. 212 с.

7. Приказ Министерства образования и науки РФ от 17 декабря 2010 г. № 1897 «Об утверждении федерального государственного образовательного стандарта основного общего образования». URL: <https://www.garant.ru/products/ipo/prime/doc/55070507/> (дата обращения: 21.09.2023).
8. Самылкина, Н.Н. Современные средства оценивания результатов обучения. М.: Бином. Лаборатория знаний, 2007. 172 с.
9. Требования к знаниям и умениям школьников: дидактико-методический анализ / под ред. А.А. Кузнецова. М.: Педагогика, 1987.
10. Беспалько, В.П. Слагаемые педагогической технологии. М.: Педагогика, 1989. 192 с.
11. Самылкина, Н.Н. Организация углубленного обучения информатике на основе интегративного подхода: монография. М.: МПГУ, 2020. 346 с.
12. Шишмаренков, В.К. Теория и практика дифференцированного обучения в средней школе: автореф. дис. ... д-ра пед. наук. Челябинск, 1996.
13. Федеральный закон от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации». URL: <https://base.garant.ru/70291362/> (дата обращения: 13.09.2023).
14. Томлинсон, К. Как дифференцировать обучение в классах со смешанными способностями. Александрия, Вирджиния: Ассоциация по надзору и разработке учебных программ, 2001.
15. Приказ Министерства просвещения Российской Федерации от 31.05.2021 № 287 «Об утверждении федерального государственного образовательного стандарта основного общего образования» (Зарегистрирован 05.07.2021 № 64101). URL: https://edsoo.ru/Normativnie_dokumenti.htm (дата обращения: 28.05.2023).
16. Босова, Л.Л., Босова, А.Ю., Филитов, В.И. «Программируем, учимся и играем!». Программа курса внеурочной деятельности для учащихся III–VI классов // Информатика в школе. 2021. № 6 (169). С. 1–15.
17. Босова, Л.Л. Программирование как инструмент формирования вычислительного мышления обучающихся // Информатика в школе. 2020. № 10 (163). С. 4–10.
18. Босова, Л.Л., Самылкина, Н.Н. Система оценки достижений планируемых предметных результатов освоения учебного предмета «Информатика»: методические рекомендации. М.: Институт стратегии развития образования, 2023. 83 с.

REFERENCES

1. Mishin, V.A. Ispolzovanie platformy Minecraft: Education Edition na urokah informatiki [Using the MinecrAft Platform: Education Edition in Computer Science Lessons], *Informatika v shkole = Informatics in School*, 2021, No. 3 (166), pp. 50–59. (in Russ.)
2. Kluzer, S., Priego, L.P. DigComp into Action: Get Inspired, Make It Happen. S. Carretero, Y. Punie, R. Vuorikari, M. Cabrera, and O’Keefe, W. (Eds.). *JRC Science for Policy Report, EUR 29115 EN*. Publications Office of the European Union, Luxembourg, 2018, 50 p.
3. Bosova, L.L. Cifrovye navyki sovremennogo shkolnika i vozmozhnosti ih formirovaniya v shkolnom kurse informatiki [Digital Skills of a Modern Schoolboy and the Possibilities of Their Formation in a School Computer Science Course], *Informatika v shkole = Informatics in School*, 2020, vol. 1, No. 7, pp. 5–9. (in Russ.)
4. Kapterev, P.F. *Izbrannye pedagogicheskie sochineniya* [Selected Pedagogical Works], ed. by A.M. Arseniev. Moscow, Pedagogika, 1982, 704 p. (in Russ.)

5. Kuznecov, A.A., Zaharova, T.B., Zaharov, A.S. *Obshchaya metodika obucheniya informatike. Ch. 1* [General Methods of Teaching Computer Science: Textbook, part 1]. Moscow, Prometej, 2016, 300 p. (in Russ.)
6. Zaharova, T.B. *Profilnaya differenciaciya obucheniya informatike na starshej stupeni shkoly* [Profile Differentiation of Computer Science Education at the Senior School Level: Monograph]. Moscow, 1997, 212 p. (in Russ.)
7. *Prikaz Ministerstva obrazovaniya i nauki RF ot 17 dekabrya 2010 g. № 1897 “Ob utverzhdenii federalnogo gosudarstvennogo obrazovatel'nogo standarta osnovnogo obshchego obrazovaniya”* [Order of the Ministry of Education and Science of the Russian Federation Dated December 17, 2010 No. 1897 “On Approval of the Federal State Educational Standard of Basic General Education”]. Available at: <https://www.garant.ru/products/ipo/prime/doc/55070507/> (accessed: 21.09.2023). (in Russ.)
8. Samylkina, N.N. *Sovremennye sredstva ocenivaniya rezultatov obucheniya* [Modern Means of Evaluating Learning Outcomes]. Moscow, Binom, Laboratoriya znaniy, 2007, 172 p. (in Russ.)
9. *Trebvaniya k znanijam i umeniyam shkolnikov: didaktiko-metodicheskij analiz* [Requirements for the Knowledge and Skills of Schoolchildren: Didactic and Methodological Analysis], ed. by A.A. Kuznetsov. Moscow, Pedagogika, 1987. (in Russ.)
10. Bepalko, V.P. *Slagaemye pedagogicheskoy tekhnologii* [The Components of Pedagogical Technology]. Moscow, Pedagogika, 1989, 192 p. (in Russ.)
11. Samylkina, N.N. *Organizaciya uglublennogo obucheniya informatike na osnove integrativnogo podhoda* [Organization of Advanced Computer Science Education Based on an Integrative Approach: Monograph]. Moscow, Moskovskij pedagogicheskij gosudarstvennyj universitet, 2020, 346 p. (in Russ.)
12. Shishmarenkov, V.K. *Teoriya i praktika differencirovannogo obucheniya v srednej shkole* [Theory and Practice of Differentiated Education in Secondary School]: Extended Abstract of ScD Dissertation (Pedagogy). Chelyabinsk, 1996. (in Russ.)
13. *Federalnyj zakon ot 29 dekabrya 2012 g. № 273-FZ “Ob obrazovanii v Rossijskoj Federacii”* [Federal Law No. 273-FZ Dated December 29, 2012 “On Education in the Russian Federation”]. Available at: <https://base.garant.ru/70291362/> (accessed: 13.09.2023). (in Russ.)
14. Tomlinson, K. *Kak differencirovat obuchenie v klassah so smeshannymi sposobnostyami* [How to Differentiate Learning in Classes with Mixed Abilities]. Aleksandriya, Virdzhiniya, Associaciya po nadzoru i razrabotke uchebnyh programm, 2001. (in Russ.)
15. *Prikaz Ministerstva prosveshcheniya Rossijskoj Federacii ot 31.05.2021 № 287 “Ob utverzhdenii federalnogo gosudarstvennogo obrazovatel'nogo standarta osnovnogo obshchego obrazovaniya” (Zaregistririvan 05.07.2021 № 64101)* [Order of the Ministry of Education of the Russian Federation Dated 05.31.2021 No. 287 “On Approval of the Federal State Educational Standard of Basic General Education” (Registered 07.05.2021 No. 64101)]. Available at: https://edsoo.ru/Normativnie_dokumenti.htm (accessed: 28.05.2023). (in Russ.)
16. Bosova, L.L., Bosova, A.Yu., Filippov, V.I. “Programmiruem, uchimsya i igraem!”. Programma kursa vneurochnoj deyatel'nosti dlya uchashchihsya III–VI klassov [“Programming, Learning and Playing!”. The Program of the Extracurricular Activities Course for Students of Grades III–VI], *Informatika v shkole* = Informatics in School, 2021, No. 6 (169), pp. 1–15. (in Russ.)
17. Bosova, L.L. Programmirovaniye kak instrument formirovaniya vychislitel'nogo myshleniya obuchayushchihsya [Programming as a Tool for the Formation of Computational Thinking of Students], *Informatika v shkole* = Informatics at School, 2020, No. 10 (163), pp. 4–10. (in Russ.)

18. Bosova, L.L., Samylkina, N.N. *Sistema ocenki dostizhenij planiruemyh predmetnyh rezultatov osvoeniya uchebnogo predmeta “Informatika”*: metodicheskie rekomendacii [The System for Evaluating the Achievements of the Planned Subject Results of Mastering the Educational Subject “Informatics”: Methodological Recommendations]. Moscow, Institut strategii razvitiya obrazovaniya, 2023, 83 p. (in Russ.)

Босова Людмила Леонидовна, член-корреспондент, Российская академия образования, доктор педагогических наук, профессор, кафедра теории и методики обучения математике и информатике, Московский педагогический государственный университет, akulll@mail.ru

Ludmila L. Bosova, Corresponding Member, Russian Academy of Education, ScD in Education, Professor, Methods of teaching Informatics and Mathematics Department, Moscow Pedagogical State University, akulll@mail.ru

Самылкина Надежда Николаевна, доктор педагогических наук, доцент, профессор, кафедра теории и методики обучения математике и информатике, Московский педагогический государственный университет, nsamylkina@yandex.ru

Nadezhda N. Samylkina, ScD in Education, Associate Professor, Professor, Methods of teaching Informatics and Mathematics Department, Moscow Pedagogical State University, nsamylkina@yandex.ru

Мишин Вадим Андреевич, аспирант, кафедра теории и методики обучения математике и информатике, Московский педагогический государственный университет, vadyusha-mishin@mail.ru

Vadim A. Mishin, Postgraduate Student, Methods of teaching Informatics and Mathematics Department, Moscow Pedagogical State University, vadyusha-mishin@mail.ru

Статья поступила в редакцию 20.11.2023. Принята к публикации 22.12.2023

The paper was submitted 20.11.2023. Accepted for publication 22.12.2023