

ОБУЧЕНИЕ ПРОГРАММИРОВАНИЮ

УДК 378.02:37.016

DOI: 10.23951/1609-624X-2017-5-134-140

ИСПОЛЬЗОВАНИЕ СОВРЕМЕННЫХ ТЕХНОЛОГИЙ В ОБЛАСТИ ИНТЕРАКТИВНОГО ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ: ТЕНДЕНЦИИ И ПЕРСПЕКТИВЫ

М. С. Можаров

Новокузнецкий институт (филиал) Кемеровского государственного университета, Новокузнецк

Исследуется проблема совершенствования методики преподавания программирования в вузе и школе. Обосновывается необходимость новых методов и форм обучения, связанных с командной работой студентов, мультимедийным контентом, виртуальными пространствами и др. Анализируются практики преподавания программирования в ведущих зарубежных университетах и предлагаются основные пути совершенствования преподавания программирования в российских учебных заведениях. Представлен собственный опыт обучения программированию и предложено начать преобразование традиционных подходов преподавания программирования и методики обучения информатике в педагогических вузах в целях распространения прогрессивных практик преподавания из педвузов в школу.

Ключевые слова: *программирование, методика обучения, проектная деятельность, командная работа, интерактивные технологии, интернет-платформа, подготовка учителя, робототехника, мобильные приложения, алгоритм, программа, электронный образовательный ресурс.*

Современное образование в области информатики и программирования сталкивается с вызовами ускоряющегося обновления содержания, форм и методов обучения, инструментов и методологий программирования. Смена техники, операционных систем, усложнение прикладного программного обеспечения и языков программирования ставят перед системой образования новые проблемы.

Постоянное усложнение компьютерных программ заставляет изменять существующие подходы к их разработке. Актуальная практика создания программного обеспечения регламентируется множеством международных стандартов и корпоративных норм. Очевидно, что индивидуальное программирование не в состоянии обеспечивать создание сложных программных продуктов.

Командная работа над программными комплексами строится по принципам организации проектов: жизненный цикл проекта, роли участников (руководитель, дизайнер и т. д.), стадии. Каждая из ролей связана с компетенциями, которые необходимо сформировать в процессе обучения в школе и вузе.

Вместе с этим в условиях развития спроса на специалистов, способных не только разрабатывать безупречный программный код, но и участвовать в совместной проектной работе над сложными программными комплексами, преподавателям и учителям при обучении программированию необходимо

формировать у студентов, наряду с техническими знаниями и умениями, компетенции, позволяющие работать в социально-профессиональной среде [1–3]. Поэтому в образовательные программы подготовки, связанные с программированием, необходимо включать содержание обучения, обеспечивающее успешную командную работу, лидерство, командный менеджмент.

Основные структуры, используемые в программировании (линейный алгоритм, ветвление, циклы), не могут быть проблемными в повседневной жизни, постоянно присутствуя в нашей деятельности, они практически не осмысливаются человеком. И только при обучении программированию эти ситуации становятся предметом внимания студентов и требуют от них владения особенной методологией, особыми мыслительными паттернами [2].

Проведенный опрос показал, что сами студенты в качестве основных проблем при изучении программирования называют: большой объем новой информации, строгость в использовании синтаксиса языка программирования, базовую математическую подготовку, необходимость продолжительной работы в изоляции, а также длительные периоды умственного напряжения при разработке программ и др.

Традиционное обучение программированию не предоставляет возможности студентам освоить успешную работу в команде. Большинство образовательных практик направлены на изучение техни-

ческих и теоретических особенностей, при этом обучение центрировано на преподавателе (teacher-centered approach), который транслирует свои знания и опыт студентам [3].

Следует учитывать, что даже традиционное обучение теории программирования и формирование навыков создания алгоритмов – неординарная задача. Процесс создания программ требует особых качеств личности и интеллектуальных способностей, таких как анализ полисубъектных процессов, отслеживание собственной мыследеятельности, отображение внутренних процессов конструкциями алгоритмических языков.

Таким образом, необходимость формирования готовности к командной работе над программным продуктом значительно усложняет и без того непростую задачу преподавателя.

В нашей профессиональной и научно-методической работе особое внимание уделяется обучению программированию будущих учителей информатики – бакалавров педагогического образования по профилю «информатика». Это особенно важно потому, что хорошая методическая подготовка позволит впоследствии реализовать современные подходы к преподаванию программирования в школе. В целях такой подготовки целесообразно проанализировать опыт преподавания программирования в наиболее успешных учебных центрах, успехи которых в данной области очевидны.

Анализ образовательного контента Интернета позволяет утверждать, что наиболее широкий спектр распространения авторских подходов к преподаванию программирования, а также высокая эффективность преподавания представлены университетами США и Европы. Многие из ведущих западных университетов успешно собирают и развивают методики из других регионов мира: России, США, Китая, Индии, стран Азии и Африки.

Опыт применения интерактивных технологий в преподавании программирования. Приведем ключевые инновационные направления в области методики обучения программированию за рубежом и примеры реализации этих направлений в ведущих университетах.

Опыт коллективного обучения программированию достаточно разнообразен [3–5]. Различные подходы реализуются в университетах Европы. Например, в Испании (Complutense University of Madrid) группа преподавателей проводит командное обучение программированию в системе e-Learning NUCLEO, которая была разработана на основе социально-конструктивистского педагогического подхода для развития коммуникационной практики в проблемном обучении (Problem Based Learning). Данная компьютерная система позволяет моделировать активность взаимодействия в групповой ра-

боте, межличностную коммуникацию, конфликты и их разрешение. Принципы работы системы NUCLEO основаны на теории активности и реализованы в виртуальной среде, позволяющей изучать процессы развития личности. Для решения этой задачи педагогу необходимо моделировать поведение субъекта в социальном окружении. Исследователи предлагают использовать NUCLEO для совершенствования командной разработки программных продуктов. В NUCLEO социальное взаимодействие реализуется по двум различным схемам: соревновательности и сотрудничества, призванным повысить мотивацию и развить групповую динамику в различных игровых контекстах, сценарии которых готовят преподаватели. В системе реализована иерархия наград и различных социальных рангов, которые достигаются благодаря учебным достижениям студентов [6].

Игровая концепция обучения программированию подразумевает, например, угадывание участков программы по комбинации входных и выходных данных. Pex4Fun (<http://www.pex4fun.com>) – браузерная игра, разработанная специалистами Microsoft Research (MSR). Она может быть использована на различных компьютерных платформах и устройствах – от персональных компьютеров до смартфонов. В соревновательных ситуациях студенты могут осваивать программирование на языках C# и VisualBasic, достигая намеченных преподавателем целей. Данная игра позволяет изучать программирование на различных уровнях образования – от школьного до вузовского. К достоинствам данной игры можно отнести возможность группового участия в решении задач и обратную связь, позволяющую преподавателю менять стратегии обучения. Дидактическим преимуществом является возможность необычных комбинаций входных данных для созданных студентами алгоритмических конструкций, которые не только позволяют протестировать решение задачи, но и заставляют их задуматься над пониманием созданного фрагмента кода [5].

Одним из направлений инновационных изменений в преподавании программирования мы считаем создание виртуальных образовательных пространств, насыщенных предметным (алгоритмическим) содержанием. Например, виртуальные вселенные Second Life (<http://secondlife.com>) могут быть использованы для командного изучения программирования в вузе [7]. Second Life представляет собой расширяемую пользователями компьютерную игровую платформу, позволяющую реализовывать социальные взаимодействия, в том числе образовательные сценарии. К недостаткам данной программы относят отсутствие возможностей организовать управляемый образовательный процесс

и сложности в коммуникации между участниками обучения [8].

Учебный язык Alice (<http://www.alice.org>) предложен преподавателями Carnegie Mellon University. Он имеет интуитивно понимаемый интерфейс визуального программирования, который, по их мнению, существенно облегчает начало учебной работы и сокращает время на изучение синтаксиса и алгоритмических конструкций [4]. Alice – инновационный 3D-проект, обеспечивающий предметный контекст обучения программированию, имеющий цель – сделать процесс обучения более легким для школьников и студентов. Проект разделен на модули и позволяет изучать не только процедурное, но и объектно-ориентированное программирование.

В процессе обучения студенты создают простые анимационные фильмы и видеоигры с привлечением 3D-объектов из библиотеки Alice (люди, животные, автомобили, строения и др.). Простой интерфейс Alice основан на технологии drag and drop и избавляет обучающихся от рутинных видов работ. Инструкции языка соответствуют стандартным инструкциям в языках программирования Java, C++, and C#.

Европейский проект Minerva CodeWitz (Финляндия) связан с раскрытием инновационных возможностей продуктивного обучения программированию в интерактивной объектно-ориентированной среде. Визуализация базовых алгоритмических структур и иерархическая библиотека примеров решения заданий позволяют разработать для студентов индивидуальные образовательные маршруты. К преимуществам данного проекта (перед традиционными подходами к обучению) можно отнести постоянное улучшение, обновление и расширение коллекций учебных материалов за счет участия в проекте ведущих европейских университетов. Авторы развивают концепцию небольших, автономных, переносимых и документированных учебных заданий, которые могут комбинироваться в учебном процессе и применяться без дополнительной подготовки преподавателей и студентов. CodeWitz может быть использован для организации командной работы студентов над учебными заданиями по программированию и позволяет развивать состязательные качества обучающихся [8].

Подход MOOC (Massive Open Online Course) предлагает возможности онлайн-обучения программированию и набирает популярность в современном образовательном пространстве. Открытые массовые онлайн-курсы обеспечивают открытость, мультимедийность, интерактивность и групповые формы работы. К недостаткам данной системы можно отнести отсутствие постоянной обратной связи преподавателя и студента, позволяющей оценить успешность обучения в целом.

Другим наиболее перспективным направлением организации обучения программированию в командных проектах является учебная робототехника. Доступные для образовательных учреждений устройства с программируемыми контроллерами позволяют создавать сложные роботизированные системы, на варьируемых уровнях сложности – от визуальных алгоритмических конструкций (MRDS 4, Scratch) до современных языков программирования (C++, Java).

Созданная в США платформа RobotC (<http://www.robotc.net/>) предлагает широкий спектр возможностей по программируемой робототехнике, позволяет программировать роботизированные устройства различных производителей (TETRIX, NXT, Cortex, RCX, PIC, VEX PIC, Arduino Diecimila, Duemilanove, Mega 2560, Mega 1280, Uno).

Для организации обучения программированию удобно, что RobotC имеет встроенную подсистему RobotC Virtual World, которая позволяет создавать программы для управления виртуальными роботизированными устройствами.

Важно отметить, что проектирование, создание и программирование роботов позволяют эффективно организовать командную работу над проектом, распределить роли между участниками и в процессе изучения программирования формировать требуемые личностные качества.

Еще одной инновационной тенденцией в современном обучении программированию является изучение алгоритмических конструкций и методов программирования при разработке мобильных приложений [9–11]. Смартфоны и телефоны прочно завоевали интерес и признание у современной молодежи и позволяют поддерживать устойчивый интерес при обучении. В качестве демонстрации такого подхода можно привести пример использования приложения визуальной разработки мобильных приложений AppInventor в США (Loyola University Chicago), где проводится экспериментальное обучение программированию студентов по специальностям, не связанным с получением квалификации в области программирования. Работа в малых группах по разработке мобильных приложений существенно повышает мотивацию студентов и увеличивает время их самостоятельной работы [9].

Мы привели только отдельные, концептуальные подходы, формы и методы обучения программированию, представленные в мировой образовательной практике, позволяющие сформулировать современные тенденции в области методики обучения программированию трансформации ее содержания, форм и методов. Эти тенденции можно обобщить в следующих положениях:

1) существует необходимость использования проектного подхода при обучении программирова-

нию для освоения студентами современных форм создания программного обеспечения;

2) командная работа над проектными заданиями позволяет, кроме технических компетенций, развивать личностные качества и компетенции, востребованные современным рынком труда;

3) в процессе обучения результатом деятельности студентов может быть завершённый программный продукт, позволяющий значительно повысить мотивацию к обучению;

4) в процессе обучения следует использовать графические приложения и виртуальные среды, моделирующие социальное взаимодействие обучающихся в ходе решения учебных задач;

5) в процессе обучения целесообразно использовать возможности открытых массовых онлайн-курсов и систем дистанционного обучения (СДО), обеспечивающих открытость, мультимедийность, интерактивность и групповые формы обучения;

6) прикладную составляющую и интерес студентов можно обеспечить за счёт привлечения к занятиям элементов учебной робототехники и конструирования;

7) построение курса обучения программированию на основе проектов по созданию мобильных приложений позволит повысить интерес и работоспособность обучающихся, а также обеспечить продуктивный характер обучения.

Экспериментальное обучение. Реализация приведенных положений возможна при условии использования интернет-платформы, позволяющей инкапсулировать современные интерактивные технологии, виртуализацию и мультимедиа в процессе обучения студентов программированию. При этом наиболее значимой, на взгляд автора, является задача применения интерактивных технологий обучения программированию студентов педагогических вузов.

Поэтому на кафедре теории и методики преподавания информатики Новокузнецкого института (филиала) ФБГОУ ВО «Кемеровский государственный университет» в 2015 г. в рамках обновления основных образовательных программ (направления: 44.03.01 «Педагогическое образование», профиль «информатика», 44.03.05 «Педагогическое образование», профили – «информатика и физика» и «математика и информатика») в содержание основных образовательных программ и рабочие программы дисциплин «Программирование», «Методика обучения информатике», «Практикум на ЭВМ», «Решение задач по программированию повышенной сложности» были добавлены специальные разделы «Использование интерактивных форм и методов обучения».

Главная идея экспериментального обучения заключалась в комплексном подходе к усилению ин-

терактивности в преподавании программирования на протяжении всего обучения будущих учителей информатики и обеспечению преемственности между теоретическими дисциплинами и методикой обучения информатике.

В качестве экспериментальной интернет-платформы был выбран образовательный портал НФИ КемГУ (<http://moodle.dissw.ru/>) на базе СДО Moodle.

В СДО, в первом модуле дисциплины «Программирование», были использованы учебные элементы, обеспечивающие игровую деятельность, например, при изучении стандартных типов данных студенты угадывали тип переменных по составу выражений. При изучении основных алгоритмических конструкций (линейный алгоритм, ветвление, цикл с параметром, цикл с предусловием и цикл с постусловием) подбирались задания, содержательно учитывающие второй профиль образовательной программы (физика или математика). Организация самостоятельной работы студентов носила интерактивный характер за счёт использования учебных элементов с тестированием, а также возможностей форума по дисциплине, предусмотренного в СДО. В форуме при решении задач студенты и преподаватель участвовали в обсуждении возникших сложностей, предлагали собственные алгоритмы и доказывали их эффективность.

При завершении лабораторных работ преподаватель фиксировал индивидуальные достижения студентов, сложность и количество решённых задач в рейтинге группы и публиковал их в СДО – в разделе дисциплины, поэтому в процессе решения проблемных заданий студенты соревновались друг с другом.

Во втором модуле дисциплины студентам предлагалось выполнить групповое задание: в соответствии с предложенной технологией создать простейший программный продукт, демонстрирующий теоретическую модель объекта или явления. Эти задания, как правило, были связаны со вторым профилем подготовки (математика или физика). Задания и требования к разработке программ также размещались в СДО, там же реализовывались коммуникации между студентами – участниками малых групп, а также общение с преподавателем. Каждый этап создания программ был закреплён за отдельным учебным элементом курса в СДО и требовал не только освоения определённого вида деятельности, но и прохождения теста по теории. Непосредственно программированием студенты занимались в среде Lazarus.

В процессе изучения дисциплины «Программирование» студенты осваивали специальные профессиональные компетенции, связанные с профилем «информатика», а также компетенции, обеспечивающие проектный вид деятельности в соответ-

ствии с ФГОС ВО. Взаимодействуя в малых группах, студенты самостоятельно распределяли между собой отдельные виды работы (анализ предметной области, поиск и обоснование математического аппарата, разбиение задачи на подзадачи, разработка подпрограмм, проектирование интерфейса, тестирование программ и т. д.), а в условиях групповой деятельности преодолевали проблемы межличностного характера.

В курсе «Методика обучения информатике» студенты использовали программы Scratch и RobotC. В первом модуле осваивали технологию программирования в онлайн-среде Scratch, создавали программы с использованием визуальных графических средств, применяли компоненты «движение», «внешность», «звук», «данные», «управление» и др. Затем разбивались на группы по 3–4 человека для разработки серии уроков по основам программирования в начальной школе и среднем звене. Для каждого урока каждая группа разрабатывала технологическую карту, презентацию, рабочую тетрадь и методические рекомендации учителю. Одним из требований к разработке уроков было использование интерактивных и игровых возможностей Scratch (оформление рабочего поля, использование нестандартных объектов – спрайтов, звука и графики, датчиков движения и системы событий).

Второй модуль дисциплины был связан с изучением возможностей инструментальной среды программирования систем учебной робототехники – RobotC. Структура практических и лабораторных занятий была заимствована из первого модуля. На первых занятиях студенты осваивали технологии среды программирования и особенности управления роботом (управление приводами колес, поворот, торможение, использование датчиков и т. д.), а затем разрабатывали уроки, обеспечивающие обучение школьников созданию алгоритмических конструкций, позволяющих управлять исполнителем в режиме реального времени. Уроки разрабатывались с учетом двух возможностей – использования реальных роботов LEGO MINDSTORMS EV3, а также возможностей виртуальной среды программирования RobotC Virtual Worlds, позволяющей реализовать управление роботами в режиме графической среды, меняя ее параметры: ракурсы наблюдения, начальные конфигурации полей и роботов.

Разработанные уроки студенты использовали в процессе прохождения педагогической практики. Для школьников на базе экспериментальной интернет-платформы образовательного портала НФИ КемГУ студенты создавали специальные электронные курсы, содержащие интерактивные задания, учебные элементы и тесты.

Дисциплина «Практикум на ЭВМ» была представлена в СДО и формировала у обучающихся готовность к разработке программных средств учебного назначения. В процессе освоения дисциплины будущие учителя разрабатывали завершённые программы, реализуя ключевые этапы проектов по созданию программных продуктов: от исследования предметной области и технического задания до отладки и тестирования программы. Студенты работали в малых группах, а основной идеей, обеспечивающей повышение уровня интерактивности обучения программированию, было использование MS Project для управления проектом на уровне задач, ресурсов и календаря.

Дисциплина «Решение задач по программированию повышенной сложности» требовала от студентов знания теории программирования, практики использования стандартных алгоритмов и структур данных. В СДО для студентов был подготовлен образовательный маршрут, подразумевающий использование традиционных интерактивных форм организации обучения, например, таких, как мозговой штурм. В одном из учебных модулей студенты использовали ранее описанную систему интерактивного обучения программированию **Рex4Fun, разработанную в MSR, в игровой форме «разгадывая» сложные алгоритмические загадки с потерянными строками программы и анализируя алгоритмы на языке Visual Basic.**

В заключение необходимо отметить, что еще рано говорить о завершении экспериментального обучения. Ежегодно образовательные программы обновляются и улучшаются, в рабочие программы включаются новые интерактивные средства, новые формы организации занятий и здания к практическим работам. Важным направлением повышения эффективности обучения программированию будущих учителей информатики мы считаем организацию педагогической практики и осмысление студентами интерактивных технологий обучения как инструмента дальнейшей профессиональной деятельности.

Несмотря на незавершенность экспериментального обучения, уже на данном этапе можно утверждать, что использованный комплекс интерактивных форм и методов обучения значительно повысил интерес студентов к изучению программирования, обеспечил повышение абсолютной и качественной успеваемости по ряду сложных дисциплин подготовки, повысил уровень самооценки готовности к педагогической деятельности и обеспечил мотивацию студентов к использованию интерактивных технологий в процессе педагогической практики.

Литература

1. Dann W., Copper S., Pausch R. Learning to program with Alice. Upper Saddle River, NJ: Prentice Hall, 2006.
2. Kelleher C., Pausch R. Lowering the barriers to programming: A taxonomy of programming environment and languages for novice programmers // ACM Computing Surveys. 2008. № 37 (2). P. 83–137.
3. Strijbos J.-W. The effect of roles on computer-supported collaborative learning: doctoral dissertation. Heerlen, The Netherlands: Open University of the Netherlands, 2014.
4. Daly T. Minimizing to maximize: An initial attempt at teaching introductory programming using Alice // Journal of Computer Science in Colleges. 2011. № 26 (5). P. 23–30.
5. Tillmann N., Halleux J.D., Xie T. Pex4Fun // Teaching and Learning Computer Science via Social Gaming, in 2012 IEEE 25th Conference on Software Engineering Education and Training (CSEE&T). Nanjing, 2012. P. 90–91.
6. Pilar S.-T., Rubén F.-F. Learning teamwork skills in university programming courses // Computers & Education. 2009. № 53. P. 517–531.
7. Liu M., Williams D., Pedersen, S. Alien rescue: A problem-based hypermedia learning environment for middle school science // Journal of Educational Technology Systems. 2002. № 30 (3). P. 255–270.
8. Kirsti Ala-Mutka. Problems in learning and teaching programming - a literature study for developing visualizations in the Codewitz-Minerva project // Institute of Software Systems, Tampere University of Technology. Finland, 2008.
9. William L. Honig Teaching and Assessing Programming Fundamentals for Non Majors with Visual Programming // Computer Science: Faculty Publications and Other Works Faculty Publications. Loyola University Chicago. 2013. № 7.
10. Jordine T., Liang Y., Ihler E. A mobile-device based serious gaming approach for teaching and learning Java programming // IEEE Frontiers in Education Conference (FIE). Madrid, 2014. P. 1–5.
11. Werner M. Teaching graphics programming on mobile devices // Journal of Computing Sciences in Colleges. pp. 2013. P. 125–131.

Можаров Максим Сергеевич, кандидат педагогических наук, Новокузнецкий институт (филиал) Кемеровского государственного университета (ул. Циолковского, 23, Новокузнецк, Россия, 654041). E-mail: root@nbikemsu.ru

Материал поступил в редакцию 15.02.2017.

DOI: 10.23951/1609-624X-2017-5-134-140

APPLICATION OF MODERN TECHNOLOGIES IN THE FIELD OF INTERACTIVE LEARNING PROGRAMMING TRENDS AND PERSPECTIVES

M. S. Mozharov

Novokuznetsk Institute (Branch) of Kemerovo State University, Novokuznetsk, Russian Federation

This paper analyses the problem of improving the methods of teaching and learning programming in university and school. The authors justify the need for new methods and forms of education related to the teamwork of students, multimedia content, and other virtual spaces. The article analyzes the programming of teaching practice in leading foreign universities and proposes the main ways of improving the teaching of programming in Russian schools. The article provides an overview of the educational content of the Internet, which allows us to state that the most wide spread spectrum of creative approaches to teaching programming, as well as high efficiency of teaching are the United States and European universities. Many of the leading Western universities successfully collect and develop methods from other regions of the world: Russia, USA, China, India, Asia and Africa. The authors analyze their own experience in teaching programming and suggest starting the transformation of traditional approaches to teaching programming and methods of teaching computer science in pedagogical universities, hoping that progressive teaching practices from teachers' universities will spread to the school.

Key words: *programming, teaching methods, project activity, teamwork, interactive technologies, Internet platform, teacher training, robotics, mobile applications, algorithm, program, electronic educational resource.*

References

1. Dann W., Copper S., Pausch R. Learning to program with Alice. Upper Saddle River, NJ: Prentice Hall, 2006.
2. Kelleher C., Pausch R. Lowering the barriers to programming: A taxonomy of programming environment and languages for novice programmers. ACM Computing Surveys, 37(2), pp. 83–137, March 28, 2008.
3. Strijbos J.-W. The effect of roles on computer-supported collaborative learning. Doctoral dissertation. Heerlen, The Netherlands: Open University of the Netherlands, 2014.
4. Daly T. Minimizing to maximize: An initial attempt at teaching introductory programming using Alice // Journal of Computer Science in Colleges, 26 (5), pp. 23–30., May, 2011.

5. Tillmann N., Halleux J.D., Xie T. Pex4Fun // Teaching and Learning Computer Science via Social Gaming, in 2012 IEEE 25th Conference on Software Engineering Education and Training (CSEE&T), Nanjing, pp. 90–91, 2012.
6. Pilar S-T., Rubén F-F. Learning teamwork skills in university programming courses // Computers & Education 53, pp. 517–531, 2009.
7. Liu M., Williams D., Pedersen, S. Alien rescue: A problem-based hypermedia learning environment for middle school science // Journal of Educational Technology Systems, 30 (3), pp. 255–270, 2002.
8. Kirsti Ala-Mutka. Problems in learning and teaching programming a literature study for developing visualizations in the Codewitz-Minerva project // Institute of Software Systems, Tampere University of Technology, Finland, 2008.
9. William L. Honig Teaching and Assessing Programming Fundamentals for Non Majors with Visual Programming // Computer Science: Faculty Publications and Other Works Faculty Publications, Loyola University Chicago, 7, 2013.
10. Jordine T., Liang Y., Ihler E. A mobile-device based serious gaming approach for teaching and learning Java programming // IEEE Frontiers in Education Conference (FIE), pp. 1–5, Madrid, 2014.
11. Werner M. Teaching graphics programming on mobile devices // Journal of Computing Sciences in Colleges, pp. 125–131, 2013.

Mozharov M. S., Novokuznetsk Institute (Branch) of Kemerovo State University (ul. Tsiolkovskogo, 23, Novokuznetsk, Russian Federation, 654041). E-mail: root@nbikemsu.ru