



Формирование компетентности в области программирования у будущих инженеров-программистов

*В. С. Круглик, В. В. Осадчий**

*Мелитопольский государственный педагогический университет
имени Богдана Хмельницкого, г. Мелитополь, Украина,*

** osadchyi@mdp.u.org.ua*

Введение. В связи с интенсивным развитием индустрии программного обеспечения происходит постоянное обновление технологий и языков, применяемых в профессиональном программировании. Эти процессы актуализируют задачи повышения качества обучения и обеспечения формирования компетентности в области программирования у студентов специальностей, в рамках которых осуществляется подготовка специалистов для отрасли информационных технологий. Целью статьи является освещение результатов научного исследования, направленного на разработку и внедрение в высших учебных заведениях системы подготовки будущих инженеров-программистов к профессиональной деятельности.

Материалы и методы. В работе приняли участие 139 студентов двух университетов Украины. Исследование было организовано с применением метода параллельного эксперимента (сбор данных проводился в контрольной и экспериментальной группах). Для сбора данных об уровне сформированности у студентов компетентности в сфере программирования проведен анализ их успеваемости по ряду профессионально ориентированных дисциплин. Для анализа полученных показателей применялись методы группировки, построения сводных таблиц, графического представления данных. Их достоверность была проверена с использованием критериев Фишера и однородности χ^2 Пирсона.

Результаты исследования. Авторы статьи впервые разработали систему подготовки будущих инженеров-программистов к профессиональной деятельности и выполнили экспериментальную проверку эффективности формирования у студентов компетентности в области программирования в рамках данной системы. По итогам проведенного исследования авторы определили, что на констатирующем этапе более трети студентов имели низкий уровень сформированности исследуемой компетентности, что свидетельствовало о недостаточном уровне профессиональной подготовки. По результатам контрольного этапа установлено, что в экспериментальной группе уровень сформированности компетентности в области программирования существенно повысился. Данное исследование расширило представление о концептуальных основах профессиональной подготовки будущих инженеров-программистов в высших учебных заведениях.

Обсуждение и заключение. Полученные результаты представляют практическую ценность для научно-педагогических сотрудников вузов при разработке образовательных программ профессиональной подготовки будущих инженеров-программистов, в частности в сфере программирования.

Ключевые слова: будущий инженер-программист, компетентностный подход, профессиональная компетентность, компетентность в области программирования, формы, методы и средства профессиональной подготовки, педагогический эксперимент

Для цитирования: Круглик В. С., Осадчий В. В. Формирование компетентности в области программирования у будущих инженеров-программистов // Интеграция образования. 2019. Т. 23, № 4. С. 587–606. DOI: <https://doi.org/10.15507/1991-9468.097.023.201904.587-606>

© Круглик В. С., Осадчий В. В., 2019



Контент доступен под лицензией Creative Commons Attribution 4.0 License.
The content is available under Creative Commons Attribution 4.0 License.



Developing Competency in Programming among Future Software Engineers

V. S. Kruglyk, V. V. Osadchyi*

*Bogdan Khmelnytsky Melitopol State Pedagogical University,
Melitopol, Ukraine,*

* *osadchyi@mdpu.org.ua*

Introduction. Due to the intensive development of the software industry, there is a constant update of technologies and languages used in professional programming. These processes bring up to date the tasks of improving the quality of education and ensuring the development of competency in programming among students in specializations where they are trained for the information technology industry. The purpose of the article is to highlight the results of scientific research aimed at developing and implementation at higher education institutions a system of future software engineers training in professional activities.

Materials and Methods. The study involved 139 students from several Ukrainian universities. We studied the influence of the proposed forms, methods and training tools on the success of the formation of future software engineers' programming competence. The study was organized using the parallel experiment method (data collection was carried out in the control and experimental groups). To collect data on the level of students' competence in programming, their learning outcomes in a number of professionally-oriented disciplines were analysed. To analyse the data, methods of grouping, construction of pivot tables, and graphical presentation of data were used. The reliability of the data was verified using Fisher's test and Pearson's chi-square test.

Results. The authors were first who developed a system of training of future software engineers in professional activities and experimentally tested the effectiveness of students' programming competence moulding in the framework of this system. Here, presented are results of the study of formation of future software engineers' professional competency, in particular competency in the field of programming, during their college education. A brief description of the professional competence structure and the system of future software engineers training for professional activities is given. Description of the pedagogical experiment stages and its contents is given, along with quantitative data in the form of tables and bar charts and its analysis. At the decisive stage of the experiment, more than a third of the students had low levels of competence under consideration, which indicated the inadequate level of professional training. According to the results of the control stage, it was established that in the experimental group, the level of competence in programming increased significantly.

Discussion and Conclusion. The study results are of practical value for the scientific and pedagogical staff of universities for the development of educational programs for professional training of future software engineers, in particular, in the field of programming.

Keywords: future software engineer, competence-based approach, professional competence, competence in programming, forms, methods and means of professional training, pedagogical experiment

For citation: Kruglyk V.S., Osadchyi V.V. Developing Competency in Programming among Future Software Engineers. *Integratsiya obrazovaniya* = Integration of Education. 2019; 23(4):587-606. DOI: <https://doi.org/10.15507/1991-9468.097.023.201904.587-606>

Введение

Современный специалист должен быть экспертом в своей сфере деятельности, обладать умениями реагировать на события и находить новые способы решения возникающих проблем, а также соответствовать изменяющимся требованиям рынка труда [1, с. 5]. В частности, для инженера-программиста необходимы умения в сфере алгоритмизации, разработки программных продуктов и управления проектами, а также способности успешно адаптироваться к усло-

виям профессиональной деятельности, учиться в течение всей жизни, общаться и работать в команде и др.

Образовательные программы, направленные на профессиональную подготовку будущих инженеров-программистов в высших учебных заведениях (вузах), предусматривают изучение множества обязательных и выборочных курсов, в рамках которых осуществляется формирование у студентов необходимых компетентностей. Их рекомендованный перечень сформулирован и периодически-

ски пересматривается группой экспертов Association for Computing Machinery (ACM) и IEEE Computer Society (IEEE CS) [2]. В их числе выделим компетентность в области программирования как интегральное свойство личности, которое характеризуется определенным уровнем развития алгоритмического стиля мышления и способностью к усвоению искусственного языка, включает знания и умения в области алгоритмизации и программирования, проявляется в различных формах программно-алгоритмической деятельности [3, с. 72].

Проблема совершенствования профессиональной подготовки будущих инженеров-программистов является сложной в теоретическом и практическом аспектах. Теоретическая сложность связана с недостаточной изученностью структуры профессиональной компетентности и разработанностью критериев профессионализма инженеров-программистов, что объясняется разнообразием выполняемых ими профессиональных функций. Практическая сложность вызвана несовершенством образовательных и профессиональных стандартов, на которые нужно опираться в процессе профессиональной подготовки будущих инженеров-программистов. В связи с этим существует необходимость проведения исследований, направленных на поиск путей повышения качества подготовки будущих инженеров-программистов, внедрение в образовательный процесс новых методических подходов, а также на анализ и оценку полученных результатов.

Цель статьи – охарактеризовать экспериментальное исследование, ориентированное на внедрение в вузах разработанной системы подготовки будущих инженеров-программистов к профессиональной деятельности, проанализировать полученные результаты и сделать вывод об эффективности формирования у студентов компетентности в области программирования в рамках предложенной системы.

Обзор литературы

В процессе исследования мы опирались на научную и специальную литературу, посвященную проблемам программирования, обучения ИТ-специалистов, устойчивого развития общества. Профессиональная подготовка будущих специалистов в области информационных технологий, в частности инженеров-программистов, является предметом многих современных научных исследований. В них рассматриваются теоретические и практические аспекты совершенствования образовательного процесса, организации экспериментальной работы, направленной на определение эффективности предложенных подходов, приводятся методики и результаты анализа полученных данных с использованием методов дескриптивной и индуктивной статистики.

Так, роль качества высшего образования в обеспечении устойчивого развития экономики и общества исследуют Н. Д. Гуськова, С. М. Вдовин, И. Н. Краковская и Ю. Ю. Слушкина. Ученые подчеркивают, что качество высшего образования предполагает удовлетворение потребностей экономики и общества в специалистах высокой квалификации, обладающих передовыми знаниями, навыками, умениями, профессиональной и личной компетентностью и готовы применять их на практике; формирование социального капитала, который определяет конкурентные преимущества страны, региона и организации; развитие науки, техники и технологий; формирование корпоративной культуры, создание новых ценностей и способов поведения; доступ к непрерывному образованию; укрепление институтов гражданского общества [4, с. 245]. На основе анализа данных, полученных в ходе исследования, ученые выделяют несколько направлений повышения качества высшего образования в интересах устойчивого развития экономики и общества: развитие системы «непрерывного образования»; улучшение взаимодействия между заинтересованными сторонами (высшие учебные заведения – бизнес –



государственные органы – гражданское общество); использование в образовании практико-ориентированного подхода; соответствие приобретенных знаний требованиям рынка труда; соответствие структуры, объемов и форм профессиональной подготовки специалистов потребностям развития экономики и общества; формирование у обучающихся ценностей, позволяющих реализовать возрастающие требования к качеству высшего образования [4, с. 255].

М. Бьязутти, В. Макракис, Э. Кончина, С. Фрате представили опыт повышения квалификации преподавателей университетов в рамках международного проекта Tempus, ориентированного на корректировку образовательных программ с целью подготовки студентов к решению задач устойчивого развития. Авторы отмечают, что концепция «образования для устойчивого развития» опирается на междисциплинарный и студенто-ориентированный подходы, что способствует развитию у будущих специалистов критического мышления и необходимых навыков. Кроме того, она может послужить основой для внедрения инновационных практик преподавания в высших учебных заведениях и профессионального развития преподавательского состава. На основании результатов пилотного исследования авторы делают вывод, что использование принципов «образования для устойчивого развития» способствовало совершенствованию принципов и методов преподавания, навыков разработки учебных курсов и подходов к оцениванию участников проекта, формированию их как лидеров перемен [5].

Л. М. Хилти, П. Хубер предложили ряд тематических областей, которые могут быть рассмотрены во время обучения будущих специалистов отрасли информационно-коммуникационных технологий (ИКТ) с целью привлечения их внимания к проблемам устойчивости [6]. В. В. Осадчий и соавторы представили результаты научных исследований, направленных на решение задачи установления соответствия квалификационных

характеристик специалистов с высшим образованием в странах Европы. Авторы разработали модель интеллектуальной системы для проведения сравнительного анализа квалификационных рамок [7; 8]. Веб-ориентированная программная реализация этой системы позволяет сопоставлять различные национальные рамки и является одним из инструментов выполнения потребностей специалистов в сфере академической и профессиональной мобильности [9].

Итоги опытно-экспериментального исследования по разработке и оценке электронной системы, которая оказывает помощь начинающим программистам в процессе обучения, приводит Х. Л. Фва [10]. Ее особенностью является мониторинг эмоционального состояния студентов во время решения задач по программированию и оказание им помощи с учетом полученных данных. Для оценки эффективности системы автор использовал количественные и качественные методы. Это позволило утверждать, что учет эмоциональных характеристик и предотвращение негативных реакций положительно влияет на результаты процесса обучения.

В своей работе А. Джаяль, С. Лория, А. Такер, С. Свифт представили результаты экспериментального исследования, направленного на обоснование оптимального подхода к изучению программирования на I курсе бакалавриата [11]. Авторы рассмотрели два варианта: первый – с опорой на объектно-ориентированное программирование (ООП) и язык Java, второй – с опорой на язык Python с последующим переходом к ООП и языку Java. Исследователи проанализировали количественные показатели (уровень успешности решения задач, частоту использования основных конструкций языка программирования, количество задач, выполненных с ошибками) и доказали эффективность второго подхода [11, с. 89].

В свою очередь, М. Новоставски, С. Маккалум, Д. Мишра обратились к решению проблемы повышения активности студентов в процессе изучения

дисциплин, связанных с чтением и анализом научных и технических документов [12]. Ученые отмечают, что в отличие от практических курсов по разработке программного обеспечения, такие дисциплины менее привлекательны для студентов, поэтому возникает проблема их вовлечения в учебный процесс. Для ее решения они разработали электронную игру «Чтение и обсуждение исследовательских статей», в процессе которой студенты должны обосновывать свои мнения, предлагая аргументы и контраргументы. Авторы приводят описание экспериментального исследования и первоначальные оценки эффективности использования этой игры.

Результаты экспериментального исследования, направленного на изучение особенностей эмоциональной и волевой составляющих самоконтроля будущих ИТ-специалистов в напряженных условиях деятельности, представлены в работе М. Ф. Бакунович и Н. Л. Станкевич [13]. Используя количественные (диагностические методики) и качественные (беседу, наблюдение) методы, а также статистическую обработку полученных данных, авторы установили, что в таких условиях студенты демонстрируют низкие показатели точности и производительности деятельности. Кроме того, снижаются показатели общего функционального состояния, происходит ухудшение параметров самочувствия и активности, возникают негативные эмоциональные состояния (усталость, напряженность, неуверенность в собственных действиях) [13, с. 691].

М. Коронадо, К. А. Иглесиас, А. Каррера, А. Мардоминго рассматривают возможности использования электронного помощника (ассистента) в процессе обучения студентов языку программирования Java [14]. Ученые разработали прототип такого ассистента, который поддерживает общение на естественном языке. Обработка экспериментальных данных, полученных в процессе внедрения программы в образовательный процесс, была проведена методом дисперсионного анализа. Это

позволило авторам проверить ряд статистических гипотез и сделать вывод о том, что общение студентов с программой на естественном языке с элементами социального диалога приводит к изменению их поведения, увеличивая количество взаимодействий с системой. В частности, пользователи чаще формулируют запросы к программе на естественном языке, чем на основе ключевых слов. Авторы отмечают, что эффективность системы и предложенного подхода требует дальнейшего лонгитюдного исследования. Однако полученные результаты могут служить основой для совершенствования диалоговых возможностей программного средства [14, с. 66].

Проблемы обучения будущих ИТ-специалистов исследует А. Экердаль. В своей работе она уделяет особое внимание феноменологическому анализу понимания студентами ключевых понятий объектно-ориентированного программирования [15]. Во время проведения феноменологического исследования результатов освоения начального курса ООП основная задача состоит в выяснении того, как студенты понимают, что значит научиться программировать, а также абстрактные понятия (объект, класс и др.). Используя феноменологию и теорию вариаций, А. Экердаль построила аналитическую модель связи между теорией и практикой в обучении программированию студентов-новичков. Приведены конкретные примеры того, как теория вариации и закономерности изменений могут быть использованы в процессе обучения программированию.

Значительный вклад в развитие методической системы профессионального обучения программистов сделал М. Э. Касперсен [16]. Признавая программирование одной из семи грандиозных задач в области компьютерного образования, он обосновал концептуальные основы формирования у будущих программистов навыков, необходимых для системной разработки программ. Методическая система, разработанная ученым, опирается на положения когнитивной



теории нагрузки в сочетании с лучшими практиками программной разработки. В соответствии с этой теорией лучшего результата можно достичь, если в процессе обучения полностью используется рабочая память, необходимая для эффективного приобретения знаний. Слишком малая или слишком большая нагрузка приводит к ухудшению результата обучения. Для оптимизации обучения необходимо сбалансировать когнитивную нагрузку, не сводя ее ни к минимуму, ни к максимуму. Поэтому одним из наиболее эффективных методов в профессиональной подготовке будущих программистов М. Э. Касперсен считает такой, при котором процесс обучения осуществляется через предоставление преподавателем рекомендаций и скаффолдинг (поддержка, оказываемая специалистом студентам в выполнении любой поставленной задачи). При таком подходе студенты осваивают важные аспекты программирования, сохраняя при этом когнитивные нагрузки в пределах, способствующих успешному обучению [16, с. 118].

В работе Д. Тиг освещены особенности использования метода парного программирования в образовательном процессе [17]. При написании программы один программист берет на себя роль «ведущего», т. е. набирает код и решает тактические задачи, второй – мыслит стратегически, ставит вопросы и отслеживает ошибки в коде. Преимуществами такого метода, по мнению исследователя, является то, что студенты делают меньшее количество ошибок, лучше понимают код, который становится более качественным, учатся друг у друга, овладевают навыками креативности, развивают способность быстро решать возникающие проблемы. Однако выделяются и недостатки этого метода, связанные с такими факторами, как индивидуальные различия студентов, работающих в паре, понимание студентами правил и готовность соблюдать их, посещаемость студентов, контроль со стороны преподавателей за соблюдением студентами правил. Для минимизации этих недостатков Д. Тиг предлагает

руководствоваться следующими принципами: желательно, чтобы в пару входили студенты, обладающие похожими способностями; проводить лабораторные занятия, на которых студенты работают в парах; обеспечивать удобные рабочие места для работы в паре; состав пар должен меняться в течение семестра; студенты должны понимать, что о проблемах с партнером нужно немедленно сообщить преподавателю, чтобы исправить ситуацию; программисты в паре должны работать для достижения общей цели; лучше предлагать задачи, которые могут быть выполнены на лабораторном занятии, чем ожидать, что студенты будут встречаться во внеучебное время; установить стандарт кодирования, которого должны придерживаться все учащиеся; соблюдать принцип сотрудничества, взаимного уважения и общей ответственности во взаимодействии между студентами и преподавателем; контролировать посещаемость и опоздания, чтобы студенты не оставались без партнера; осуществлять контроль баланса между индивидуальной и совместной работой студентов; преподаватели должны побудить пары самостоятельно решать задачи. Исследования, проведенные Д. Тиг, подтвердили эффективность использования метода парного программирования в обучении студентов.

Особенности профессиональной подготовки будущих веб-программистов представлены в работе В. В. Осадчего и В. С. Круглика [18]. Авторы проанализировали направления деятельности веб-программистов и требования к ним, обосновали содержание профессиональной подготовки в университетах, определили факторы их успешного трудоустройства.

Таким образом, в процессе теоретических и прикладных исследований накоплен значительный опыт обучения программированию будущих специалистов отрасли ИКТ, в частности инженеров-программистов. В то же время актуальной остается задача экспериментальной проверки эффективности разрабатываемых подходов с учетом

изменяющихся требований к компетентности выпускников высших учебных заведений.

Материалы и методы

С целью формирования у студентов компетентности в области программирования мы, основываясь на работах ряда исследователей (В. В. Калитина [19], М. Э. Касперсена [16], А. Экердаль [15], Л. Ма [20], Ю. Сорвы [21], Д. Тиг [17], Н. Труонг [22] и др.), считаем целесообразным использование проектного, исследовательского, эвристического, итерационного методов, стратегии когнитивного конфликта, методов ученичества, визуального моделирования программ, построения ментальных моделей, выполнения упражнений с пробелами, парного программирования.

Для проверки эффективности использования указанных методов для формирования у будущих инженеров-программистов компетентности в области программирования в процессе профессиональной подготовки в вузах нами было проведено экспериментальное исследование на базе ряда университетов Украины. При разработке и реализации плана экспериментальной работы мы опирались на положения, изложенные в трудах М. И. Грабарь и К. А. Краснянской¹, А. Колгатина [23], Д. А. Новикова², А. В. Сидоренко³.

Уровень сформированности компетентности в области программирования определялся по 100-балльной шкале (высокий – 90–100, средний – 74–89, низкий – 60–73) путем оценивания способности студентов применять знания и умения в рамках ряда профессионально ориентированных учебных курсов («Введение в специальность», «Основы алгоритмизации и программирования», «Системное программирование», «Объектно-ори-

ентированное программирование», «Веб-программирование», «Основы логического программирования»).

В процессе опытно-экспериментальной работы был использован метод параллельного эксперимента, что предусматривало выделение в составе генеральной совокупности студентов области знаний «Информационные технологии» (648 чел.) контрольной (КГ) и экспериментальной (ЭГ) групп.

Педагогический эксперимент включал три основных этапа.

1. *Констатирующий*. Основные мероприятия: формирование контрольной (67 чел.) и экспериментальной (72 чел.) групп, предварительное оценивание уровня сформированности компетентности в области программирования у участников этих групп и определение имеющихся недостатков, анализ исследуемого показателя с использованием критерия Фишера и критерия однородности χ^2 Пирсона.

2. *Формирующий*. Основные мероприятия: организация учебного процесса в экспериментальной группе с применением предложенных форм, методов и средств; наблюдение за изменениями, происходящими в ЭГ и КГ.

3. *Контрольный*. Основные мероприятия: повторная диагностика сформированности компетентности в области программирования, анализ полученных данных с применением статистических критериев, сравнительный анализ результатов констатирующего и формирующего этапов.

Оценка эффективности формирования у будущих инженеров-программистов компетентности в области программирования осуществлялась в конце IV и V курсов и сравнивалась с результатами диагностики в начале I курса для выяснения динамики процесса ее формирования.

¹ Грабарь М. И., Краснянская К. А. Применение математической статистики в педагогических исследованиях. Непараметрические методы. М.: Педагогика, 1977. 134 с.

² Новиков Д. А. Статистические методы в педагогических исследованиях (типовые случаи). М.: МЗ-Пресс, 2004. 68 с.

³ Сидоренко Е. В. Методы математической обработки в психологии. С.-Пб.: ООО «Речь», 2000. 350 с.



Результаты исследования

1. *Профессиональная компетентность инженера-программиста и ее структура.* По определению А. В. Хурторского, компетентность представляет собой «совокупность личных качеств обучаемого, необходимых и достаточных для осуществления продуктивной деятельности по отношению к определенному объекту» [24, с. 86].

Под понятием «профессиональная компетентность будущих инженеров-программистов» мы понимаем совокупность профессиональных и общих компетентностей, которые являются важными (ключевыми) для профессиональной деятельности будущих инженеров-программистов [3, с. 72].

С целью разработки структуры профессиональной компетентности будущих инженеров-программистов мы проанализировали ряд научных работ [5–29] и других информационных источников⁴ [2]. В результате был сформирован общий список компетентностей, систематизация которого дала основания для выделения общих и профессиональных компетентностей.

К общим мы отнесли умения учитывать влияние факторов внешней среды на результативность профессиональной деятельности, общаться, включая устную и письменную коммуникацию на государственном и иностранном языках, заниматься в группах по разработке сложных программных комплексов, организовать свою деятельность и эффективно управлять временем; способность применять знания на практике, способность к овладению новыми знаниями и продолжению профессионального развития, расширению научного мировоззрения.

Среди профессиональных мы выделили знания о выдающихся представителях сферы информационных

технологий; способность разрабатывать программные продукты; владение современными методами эффективного доступа к информации, ее сбора, систематизации и хранения; способность использовать методы идентификации и классификации информации на базе новых информационных технологий с помощью программных технических средств, локальных и глобальных компьютерных сетей; умения разрабатывать модели разных типов в зависимости от задач проектирования программного комплекса, строить эффективные вычислительные алгоритмы для расчетных задач, определять эффективность программ с помощью программного обеспечения, выбирать рациональные алгоритмы решения задач оптимизации и оптимального управления, разрабатывать комплексные информационные решения для предприятий; умения применять современные методы проектирования программ, принимать оптимальные решения о составе программного обеспечения, осуществлять сбор, обработку, анализ, систематизацию научно-технической информации; способность самостоятельно получать и применять в практической деятельности новые знания и умения, пользоваться современными парадигмами программирования для разработки и проектирования программных систем; знания предметной области в рамках профессии «специалист по информационным технологиям»; способность демонстрировать знания фундаментальных и смежных специальных дисциплин бакалаврской программы для решения прикладных задач; способность разрабатывать и реализовывать процессы жизненного цикла информационных систем, программного обеспечения, сервисов систем информационных технологий; способность применять методы оценки и анализа функционирования средств

⁴ Профессиональные стандарты в области ИТ [Электронный ресурс]. URL: <http://www.apkit.ru/committees/education/meetings/standarts.php>; Computer Engineering 2016 [Электронный ресурс]. Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering. – A Report in the Computing Curricula Series. – The Joint Task Force on Computing Curricula. Association for Computing Machinery (ACM) IEEE Computer Society [Электронный ресурс]. 2015. 142 p. URL: <https://www.acm.org/binaries/content/assets/education/ce2016-final-report.pdf>; European ICT Professional Profiles [Электронный ресурс]. URL: <http://www.ecompetences.eu/ict-professional-profiles>.

и систем ИТ; способность разрабатывать проектную и программную документацию в соответствии с нормативными документами, использовать международные и профессиональные стандарты в области ИТ, выполнять анализ профессиональных проблем, постановку и обоснование задач.

Выделенные знания, умения, навыки, способности и способы деятельности мы сгруппировали в 7 составляющих профессиональной компетентности будущих инженеров-программистов: компетентность в области программирования, цифровая, математическая, инженерная, коммуникативная, личностно-профессиональная и управленческая.

В составе профессиональной компетентности будущего инженера-программиста одной из важнейших мы считаем компетентность в области программирования, которая включает знания и умения в области алгоритмизации и программирования, проявляется в различных формах программно-алгоритмической деятельности, является интегральным свойством личности, которое характеризуется определенным уровнем развития алгоритмического стиля мышления и способностью к усвоению и использованию искусственного языка. Будущие инженеры-программисты должны владеть методами разработки программных комплексов, умением разрабатывать и налаживать эффективные алгоритмы и программы с использованием современных технологий программирования.

2. Система подготовки будущих инженеров-программистов к профессиональной деятельности. Процесс профессиональной подготовки будущих инженеров-программистов в высших учебных заведениях предполагает реализацию принципиально новых подходов в рамках специально сконструированной педагогической системы, которой присущи такие признаки, как многофункциональность, открытость, согласованность уровней (университет – факультет – кафедры), целостность, взаимосвязь компонентов, нацеленность на результат.

В нашем понимании система подготовки будущих инженеров-программи-

стов к профессиональной деятельности – это особая педагогическая система, основанная на мониторинге требований рынка труда к квалификации инженера-программиста, интеграции целей, содержания, методологических подходов, дидактических принципов, форм, методов и средств обучения и информационно-коммуникационных технологий в профессиональной подготовке будущих инженеров-программистов в условиях вуза, направленная на формирование целостной профессиональной компетентности будущих инженеров-программистов.

Цель, задачи, структура и организационно-педагогические условия функционирования этой системы в вузах определены нами на основе положений авторской концепции системы подготовки будущих инженеров-программистов к профессиональной деятельности.

Целью этой концепции является теоретико-методологическое обоснование внедрения в вузах системы подготовки будущих инженеров-программистов к профессиональной деятельности, которая соответствует современным требованиям и может быть адаптирована к требованиям, которые будут возникать в будущем в связи с развитием ИТ-отрасли. Ее основная идея заключается в создании системы организованного и целенаправленного педагогического воздействия на профессиональную подготовку будущих инженеров-программистов в процессе их обучения в информационно-образовательной среде вуза, которая учитывает современные требования к квалификации инженеров-программистов и предусматривает:

1) поэтапное формирование профессиональной компетентности будущих инженеров-программистов с учетом результатов систематического мониторинга рынка труда и тенденций ИТ-индустрии;

2) оперативное, своевременное и систематическое обновление содержания образования (выделение ядра учебных дисциплин, контекстуализация дисциплин общей подготовки, усиление языковой подготовки, учет прпедевтического содержания учебных дисциплин);



3) сквозное внедрение разработанной схемы последовательного изучения студентами парадигм программирования на основе принципа наращиваемой сложности;

4) гармоничное сочетание очных и дистанционных, индивидуальных и групповых, традиционных и инновационных форм, общедидактических и специфических методов и средств профессиональной подготовки будущих инженеров-программистов;

5) организацию сквозного взаимодействия с работодателями для обеспечения практической подготовки будущих специалистов и их трудоустройства;

6) внедрение системы качества теоретической и практической подготовки будущих специалистов.

Цель системы подготовки будущих инженеров-программистов к профессиональной деятельности заключается в создании организационно-педагогических условий для формирования у них высокого уровня профессиональной компетентности в процессе обучения в вузе. Достижение этой цели осуществляется путем выполнения ряда задач: устранения противоречий в профессиональной подготовке будущих инженеров-программистов; создания в вузах организационно-педагогических условий для качественной профессиональной подготовки студентов на уровне, соответствующем современным стандартам ИТ-отрасли; системного использования ИКТ в процессе профессиональной подготовки будущих инженеров-программистов; поэтапного формирования профессиональной компетентности будущих инженеров-программистов и постоянного мониторинга этого процесса.

Система подготовки будущих инженеров-программистов к профессиональной деятельности может быть эффективной только в случае обеспечения следующих организационно-педагогических условий ее функционирования:

1. Организационные:

– разработка учебных планов с учетом тенденций развития ИТ-отрасли;

– создание информационно-образовательной среды профессиональной подготовки будущих инженеров-программистов;

– предоставление студентам возможности создавать персональную электронную среду для индивидуализации профессиональной подготовки как инженеров-программистов;

– обеспечение качественного материально-технического оснащения профессиональной подготовки;

– использование современных информационно-коммуникационных технологий;

– поддержание связей с работодателями отрасли;

– организация производственной и вычислительной практик, направленных на использование современной техники и информационных технологий.

2. Педагогические:

– учет методологических подходов и дидактических принципов в процессе формирования профессиональной компетентности будущих инженеров-программистов;

– разработка системы мотивации к изучению предметной области;

– определение и мониторинг групп компетентностей, входящих в структуру профессиональной компетентности будущих инженеров-программистов;

– отбор содержания профессиональной подготовки будущих инженеров-программистов;

– разработка и постоянное обновление учебно-методических материалов и информационных ресурсов для обеспечения профессиональной подготовки будущих инженеров-программистов;

– использование целесообразных форм, методов, средств, методических подходов и технологий обучения с целью формирования профессиональной компетентности будущих инженеров-программистов;

– привлечение студентов к самоуправлению, самообучению и самосовершенствованию;

– активная позиция преподавателей по повышению профессиональной квалификации.

Реализация профессиональной подготовки будущих инженеров-программистов на основе предложенной нами системы направлена на преодоление противоречий, имеющихся в практике деятельности учреждений высшего образования, способствуя повышению качества подготовки специалистов для ИТ-отрасли.

3. *Результаты педагогического эксперимента по исследованию эффективности формирования у будущих инженеров-программистов компетентности в области программирования.* На констатирующем этапе опытно-экспериментальной работы оценивание уровня сформированности компетентности в области программирования у студентов было осуществлено путем анализа их успеваемости по результатам первого периодического контроля по дисциплинам, названным выше. Необходимые данные

были получены из ведомостей учета успеваемости учащихся контрольной и экспериментальной групп.

В процессе оценивания было выявлено, что высокий и средний уровень сформированности указанной компетентности имели 43 студента (62,94 %) контрольной группы и 45 (64,28 %) – экспериментальной. Низкий уровень имели 24 студента (37,07 %) контрольной группы и 25 (35,71 %) – экспериментальной. Таким образом, более трети студентов в обеих выборках имели низкий уровень сформированности компетентности в области программирования, что свидетельствовало о недостаточном уровне профессиональной подготовки.

Данные о распределении студентов КГ и ЭГ по уровням сформированности компетентности в области программирования в разрезе проанализированных учебных курсов приведены в таблице 1.

Т а б л и ц а 1. Данные констатирующего этапа педагогического эксперимента
T a b l e 1. Data of the summative stage of the pedagogical experiment

Дисциплина / Subjects		Уровни сформированности компетентности в области программирования / Levels of formedness of competence in programming					
		Высокий / High		Средний / Average		Низкий / Low	
		КГ / CG	ЭГ / EG	КГ / CG	ЭГ / EG	КГ / CG	ЭГ / EG
Введение в специальность / Introduction to speciality	чел. / pers.	6	8	37	36	24	26
	%	8,96	11,43	55,22	51,43	35,82	37,14
Основы алгоритмизации и программирования / Fundamentals of algorithms and programming	чел. / pers.	5	6	36	40	26	24
	%	7,46	8,57	53,73	57,14	38,81	34,29
Системное программирование / System programming	чел. / pers.	4	7	35	39	28	24
	%	5,97	10,0	52,24	55,71	41,79	34,29
Объектно-ориентированное программирование / Object-oriented programming	чел. / pers.	6	7	38	38	23	25
	%	8,96	10,0	56,72	54,29	34,33	35,71
Веб-программирование / Web programming	чел. / pers.	3	5	39	39	25	26
	%	4,48	7,14	58,21	55,71	37,31	37,14
Основы логического программирования / Fundamentals of Logical Programming	чел. / pers.	3	7	41	38	23	25
	%	4,48	10,0	61,19	54,29	34,33	35,71
Среднее значение / Average value	чел. / pers.	5	7	38	38	24	25
	%	6,72	9,52	56,22	54,76	37,07	35,71



На основании данных, приведенных в таблице 1, можно утверждать, что существенных отличий между исследуемыми группами студентов нет, поэтому формирующий этап эксперимента возможно проводить на их базе.

Во время формирующего этапа педагогического эксперимента для повышения эффективности формирования у будущих инженеров-программистов компетентности в области программирования и качества их профессиональной подготовки в целом было осуществлено:

1. Обновление перечня компетенций, которые необходимо формировать у студентов в процессе их обучения в вузах. Для определения этого перечня было проведено исследование квалификационных требований работодателей к программистам [29], что позволило сместить акценты в целях подготовки будущих инженеров-программистов, четко поставив задачу по формированию устойчивой профессиональной компетентности.

2. Внедрение рассмотренной выше системы подготовки будущих инженеров-программистов к профессиональной деятельности. Это выразилось в применении целостной системы концептуальных подходов, организационно-педагогических условий и информационно-коммуникационных средств ее обеспечения в условиях вузов; в разумной автоматизации и информатизации профессиональной подготовки будущих инженеров-программистов; обеспечении формирования профессиональной компетентности, ориентированной на мировые ИТ-стандарты; в усилении практической ориентированности обучения и роли опыта и умений реализовывать знания на практике, устанавливая подчиненность знаний умениям и акцентируя внимание на результатах образования; в учете психологических особенностей, стиля мышления, интересов и предпочтений будущих инженеров-программистов.

3. Внедрение обновленного содержания профессиональной подготовки будущих инженеров-программистов, целью которого является формирование

системных знаний в их взаимосвязи и совокупности с профессиональными умениями и навыками для решения проблем в будущей профессиональной деятельности. Для формирования компетентности в области программирования основное значение имело:

1) усвоение студентами ЭГ обновленного содержания учебных дисциплин подготовки бакалавра:

– нормативные дисциплины профессионального цикла: «Основы алгоритмизации и программирования» (5 кредитов ECTS), «Введение в специальность» (6 кредитов ECTS), «Системное программирование» (4 кредита ECTS), «Объектно-ориентированное программирование» (5 кредитов ECTS), «Визуальное программирование» (5 кредитов ECTS), «Системы управления базами данных» (4 кредита ECTS), «Веб-программирование» (5 кредитов ECTS), «Основы логического программирования» (3 кредита ECTS), «Тестирование программного обеспечения» (6 кредитов ECTS), «Теория информации и кодирования» (5 кредитов ECTS);

– вариативные дисциплины общей и профессиональной подготовки: «Цифровая логика» (3 кредита ECTS), «Управление ИТ-проектами» (5 кредитов ECTS);

2) обновление содержания производственной и преддипломной практики.

В ходе формирующего этапа опытно-экспериментальной работы студенты ЭГ теоретически и практически отработывали учебный материал в соответствии с разработанными программами, самостоятельно анализируя электронные источники и рекомендуемую литературу, закрепляли его во время производственной и преддипломной практики, участия в конкурсах и олимпиадах.

С целью формирования у студентов ЭГ компетентности в области программирования были использованы традиционные и инновационные формы, методы и средства профессиональной подготовки:

– аудиторные (лекционные, семинарские, лабораторные, практические, индивидуальные занятия) и внеауди-



торные (учебные (производственные, преддипломные) практики, участие в научно-исследовательской деятельности, курсовое проектирование и написание квалификационной работы, дистанционное обучение, олимпиады, конкурсы, мастер-классы, творческие и тематические встречи и т. д.) формы обучения;

– общедидактические методы: объяснительно-иллюстративный, репродуктивный, проблемного изложения, частично-поисковый, исследовательский;

– специфические методы подготовки инженеров-программистов к профессиональной деятельности: поэтапное усовершенствование программ (тест-управляемая разработка, рефакторинг, объектно-ориентированное моделирование), стратегии когнитивного конфликта, визуализация программ, построение ментальных моделей, интервью, визуальное моделирование программ, упражнений с пропусками, парное программирование, программное обучение, метод решения профессионально ориентированных задач;

– вербальные, невербальные, комбинированные, наглядно-иллюстративные, технические, информационные средства учебной деятельности.

На контрольном этапе опытно-экспериментальной работы оценивание уровня сформированности у студентов компетентности в области программирования было осуществлено путем анализа их успеваемости по результатам второго периодического контроля по соответствующим дисциплинам. Эти данные показали, что в контрольной группе 42 студента (63,19 %) имели высокий и средний уровни, а 25 студентов (36,82 %) – низкий уровень. В экспериментальной группе высокий и средний уровни имели 64 студента (91,43 %), а низкий – 6 студентов (8,57 %) (табл. 2).

Таким образом, можно утверждать, что в экспериментальной группе зафиксирован более высокий уровень сформированности компетентности в области программирования у студентов. Это свидетельствует об эффективности проведенной работы.

Т а б л и ц а 2. Данные контрольного этапа педагогического эксперимента

Table 2. Data of the control stage of the pedagogical experiment

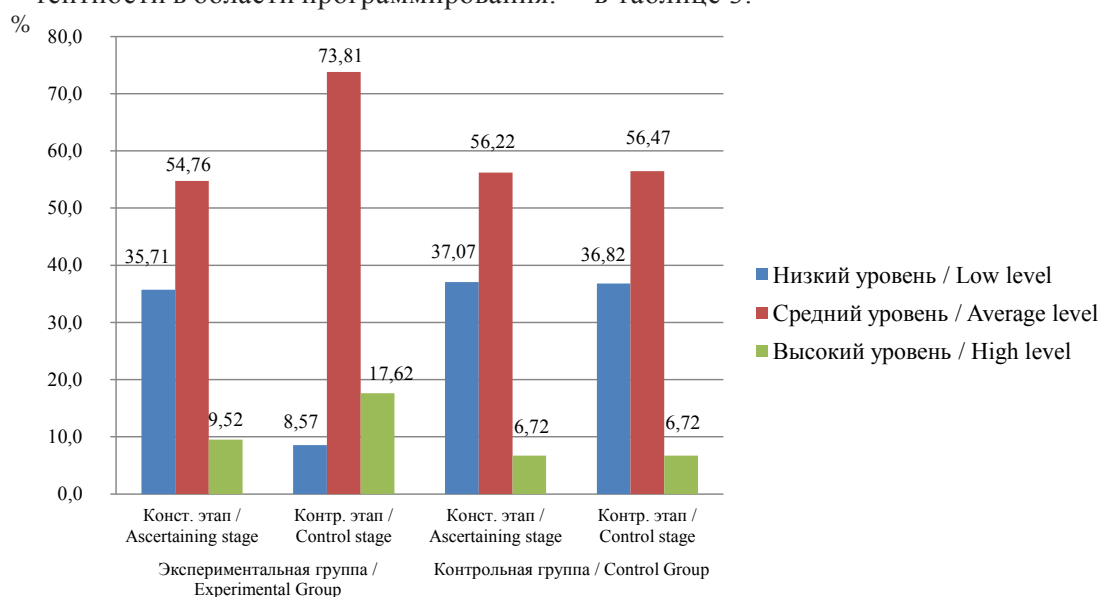
Дисциплина / Subjects		Уровни сформированности компетентности в области программирования / Levels of formedness of competence in programming					
		Высокий / High		Средний / Average		Низкий / Low	
		КГ / CG	ЕГ / EG	КГ / CG	ЕГ / EG	КГ / CG	ЕГ / EG
Введение в специальность / Introduction to speciality	чел. / pers. %	7 10,45	15 21,43	38 56,72	50 71,43	22 32,84	5 7,14
Основы алгоритмизации и программирования / Fundamentals of algorithms and programming	чел. / pers. %	5 7,46	12 17,14	37 55,22	50 71,43	25 37,31	8 11,43
Системное программирование / System programming	чел. / pers. %	4 5,97	13 18,57	35 52,24	51 72,86	28 41,79	6 8,57
Объектно-ориентированное программирование / Object-oriented programming	чел. / pers. %	4 5,97	12 17,14	39 58,21	52 74,29	24 35,82	6 8,57
Веб-программирование / Web programming	чел. / pers. %	3 4,48	11 15,71	40 59,7	53 75,71	24 35,82	6 8,57
Основы логического программирования / Fundamentals of Logical Programming	чел. / pers. %	4 5,97	11 15,71	38 56,72	54 77,14	25 37,31	5 7,14
Среднее значение / Average value	чел. / pers. %	4 6,72	12 17,62	38 56,47	52 73,81	25 36,82	6 8,57



Обобщенные результаты исследования сформированности компетентности в области программирования студентов КГ и ЭГ в начале и в конце экспериментальной работы представлены на рисунке.

В целом по результатам контрольного этапа опытно-экспериментальной работы установлено, что в контрольной группе не произошло изменений по показателю сформированности компетентности в области программирования.

В экспериментальной группе зафиксировано существенное снижение показателей низкого уровня (на 0,271), а также рост показателей среднего (на 0,19) и высокого (на 0,081) уровней. Данные, характеризующие изменение значений коэффициента показателя сформированности компетентности в области программирования у студентов контрольной и экспериментальной групп в начале и в конце эксперимента, представлены в таблице 3.



Р и с у н о к. Сводные результаты исследования, %
F i g u r e. Summary of research results, %

Т а б л и ц а 3. Динамика показателя сформированности у студентов компетентности в области программирования

T a b l e 3. Dynamics of students' competency formedness in the field of programming

Уровень сформированности / Levels of formedness	Количественные показатели (экспериментальная группа) / Quantitative Indicators (Experimental Group)		Разница / Difference	Количественные показатели (контрольная группа) / Quantitative Indicators (Control Group)		Разница / Difference
	конст. этап / summative stage	контр. этап / control stage		конст. этап / summative stage	контр. этап / control stage	
Низкий / Low	0,357	0,086	-0,271	0,371	0,368	-0,003
Средний / Average	0,548	0,738	0,190	0,562	0,565	0,003
Высокий / High	0,095	0,176	0,081	0,067	0,067	0

Достоверность статистических данных, полученных в ходе эксперимента, проверялась с помощью критерия Фишера φ и критерия однородности Пирсона χ^2 . Было установлено, что на констатирующем этапе экспериментального исследования показатели среднего и низкого уровней сформированности компетентности в области программирования в обеих группах не отличались. Этот вывод сделан на основе проверки с использованием критерия Фишера: полученные эмпирические значения критерия меньше, чем его критическое значение ($\varphi_{кр} = 1,64$). В разрезе отдельных дисциплин: «Введение в специальность» (средний уровень $\varphi_{эмп} = 0,45$, низкий уровень $\varphi_{эмп} = 0,16$), «Основы алгоритмизации и программирования» (средний $\varphi_{эмп} = 0,4$, низкий $\varphi_{эмп} = 0,55$), «Системное программирование» (средний $\varphi_{эмп} = 0,41$, низкий $\varphi_{эмп} = 0,907$), «Объектно-ориентированное программирование» (средний $\varphi_{эмп} = 0,287$, низкий $\varphi_{эмп} = 0,907$), «Веб-программирование» (средний $\varphi_{эмп} = 0,298$, низкий $\varphi_{эмп} = 0,023$), «Основы логического программирования» (средний $\varphi_{эмп} = 0,819$, низкий $\varphi_{эмп} = 0,176$).

В конце экспериментального исследования (на контрольном этапе) показатели среднего и низкого уровней сформированности компетентности в области программирования в ЭГ и КГ отличались (полученные эмпирические значения критерия превышают его критическое значение $\varphi_{кр} = 1,64$). В разрезе отдельных дисциплин: «Введение в специальность» (средний уровень $\varphi_{эмп} = 1,8$, низкий уровень $\varphi_{эмп} = 3,98$), «Основы алгоритмизации и программирования» (средний $\varphi_{эмп} = 1,98$, низкий $\varphi_{эмп} = 3,65$), «Системное программирование» (средний $\varphi_{эмп} = 2,7$, низкий $\varphi_{эмп} = 4,57$), «Объектно-ориентированное программирование» (средний $\varphi_{эмп} = 2,007$, низкий $\varphi_{эмп} = 4,025$), «Веб-программирование» (средний $\varphi_{эмп} = 2,563$, низкий $\varphi_{эмп} = 4,534$), «Основы логического программирования» (средний $\varphi_{эмп} = 2,019$, для низкого $\varphi_{эмп} = 4,025$).

Оценивание достоверности обобщенных данных с использованием критерия Фишера подтверждает результаты, приведенные выше. На констатирующем этапе показатели среднего и низкого уровней в обеих группах не имели статистически значимых отличий (средний уровень $\varphi_{эмп} = 0,287 < \varphi_{кр}$, низкий уровень $\varphi_{эмп} = 0,012 < \varphi_{кр}$). На контрольном этапе зафиксированы статистически значимые отличия между контрольной и экспериментальной группами (средний $\varphi_{эмп} = 2,182 > \varphi_{кр}$, низкий $\varphi_{эмп} = 4,207 > \varphi_{кр}$).

Сравнение показателей сформированности компетентности в области программирования в разрезе отдельных дисциплин с использованием критерия Пирсона также подтверждает сформулированный выше вывод. На констатирующем этапе эксперимента различия между исследуемыми выборками отсутствовали ($\chi^2_{эмп} < \chi^2_{кр} = 5,99$). На контрольном этапе между ними были выявлены существенные различия, о чем говорит $\chi^2_{эмп} > \chi^2_{кр} = 5,99$. Для отдельных дисциплин: «Введение в специальность» (на констатирующем этапе $\chi^2_{эмп} = 0,4$, на контрольном $\chi^2_{эмп} = 15,9$), «Основы алгоритмизации и программирования» (на констатирующем этапе $\chi^2_{эмп} = 0,399$, на контрольном $\chi^2_{эмп} = 14,2$), «Системное программирование» (на констатирующем этапе $\chi^2_{эмп} = 1,4$, на контрольном $\chi^2_{эмп} = 20,21$), «Объектно-ориентированное программирование» (на констатирующем этапе $\chi^2_{эмп} = 0,17$, на контрольном $\chi^2_{эмп} = 17,4$), «Веб-программирование» (на констатирующем этапе $\chi^2_{эмп} = 0,54$, на контрольном $\chi^2_{эмп} = 20,25$), «Основы логического программирования» (на констатирующем этапе $\chi^2_{эмп} = 1,87$, на контрольном $\chi^2_{эмп} = 17,96$). Для обобщенных данных: на констатирующем этапе $\chi^2_{эмп} = 0,37$, на контрольном $\chi^2_{эмп} = 18,62$.

Таким образом, в процессе педагогического эксперимента выявлены достоверные значимые отличия между показателями компетентности в сфере программирования среди учащихся контрольной и экспериментальной групп. Полученные данные свидетельствуют о том, что внедрение разработанной



системы подготовки будущих инженеров-программистов к профессиональной деятельности способствует становлению студентов как высококвалифицированных ИТ-специалистов.

Обсуждение и заключение

Эффективность формирования у будущих инженеров-программистов компетентности в области программирования обеспечивается сочетанием методов и средств обучения, спроектированных с учетом современного состояния ИТ-отрасли и требований к их подготовке, индивидуальных особенностей познавательных процессов, необходимости вовлечения в социальное взаимодействие и др.

В рамках нашего исследования были использованы аудиторные и внеаудиторные формы обучения. Для формирования компетентности в области программирования особое значение имело применение специфических методов обучения (поэтапное усовершенствование программ, стратегия когнитивного конфликта, построение ментальных моделей, интервью, визуальное моделирование программ, упражнения с пропусками, парное программирование, метод решения профессионально ориентированных задач, проектный подход и т. д.).

Нами было проведено экспериментальное исследование эффективности формирования у будущих инженеров-программистов компетентности в области программирования в процессе обучения в высших учебных заведениях. В нем приняли участие 139 студентов двух университетов Украины. Опытно-экспериментальная работа включала три этапа: констатирующий, формирующий и контрольный. На констатирующем этапе была проведена первичная диагностика уровня сформированности компетентности в области программирования. Цель формирующего этапа заключалась в обновлении содержания подготовки будущих инженеров-программистов, а также внедрении ряда форм, методов и средств обучения программированию. На контрольном

этапе осуществлена вторичная диагностика, анализ полученных данных и их сравнение с исходным уровнем сформированности компетентности в области программирования.

Результатом мероприятий, проведенных в экспериментальной группе на формирующем этапе исследования, стало повышение уровня сформированности компетентности в области программирования у студентов: доля студентов с высоким уровнем возросла на 0,081, со средним – на 0,19, а с низким уровнем уменьшилась на 0,271. Значительной динамики показателей сформированности компетентности в области программирования у студентов контрольной группы не зафиксировано. Полученная статистически значимая разница уровней сформированности компетентности в области программирования у студентов контрольной и экспериментальной групп (на констатирующем этапе $\chi^2_{\text{эмп}} = 0,37$, что меньше, чем $\chi^2_{\text{кр}} = 5,99$, на контрольном этапе $\chi^2_{\text{эмп}} = 18,62$, что больше, чем критическое значение критерия) показала эффективность экспериментальной работы.

По результатам проведенного педагогического эксперимента формулируем рекомендации по обеспечению качества подготовки будущих инженеров-программистов по программированию:

- модернизировать содержание и методы обучения программированию ИТ-специалистов в соответствии с международными стандартами;
- разработать вариативные модули специализаций с учетом стандартов, требований и потребностей современного рынка труда;
- осуществлять мониторинг качества подготовки по программированию будущих инженеров-программистов на уровнях «университет – факультет – кафедра»;
- осуществлять мониторинг рынка труда с целью определения требований работодателей к подготовке по программированию выпускников университетов и в соответствии с этим корректировать ее содержание.

Полученные результаты представляют практическую ценность для научно-педагогических сотрудников вузов. Они могут быть использованы при проектировании содержания образовательных программ, направленных на подготовку специалистов в области ИКТ, профессиональная деятельность

которых связана с программированием. Среди возможных направлений дальнейших исследований выделим разработку индивидуальных траекторий обучения программированию и программ тьюторского консультирования в университетах, в том числе с применением современных интеллектуальных обучающих систем.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. KEYCIT 2014: Key Competencies in Informatics and ICT / T. Brinda, N. Reynolds, R. Romeike, A. Schwill (eds.). Potsdam: Universitätsverlag Potsdam, 2015. 446 p. URL: <https://publishup.uni-potsdam.de/opus4-ubp/frontdoor/deliver/index/docId/7032/file/cid07.pdf> (дата обращения: 03.03.2019).
2. Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science. New York, NY, USA: ACM, 2013. 518 p. DOI: <https://doi.org/10.1145/2534860>
3. *Круглик В. С., Осадчий В. В.* Структура професійної компетентності майбутнього інженера-програміста // Педагогічний дискурс. 2016. Вип. 21. С. 69–74. URL: <https://clck.ru/K8oTx> (дата обращения: 03.03.2019).
4. The Quality of Education as a Primary Concern of the Sustainable Development / N. D. Guskova [et al.] // European Research Studies Journal. 2016. Vol. XIX, Issue 3 (part B). Pp. 239–257. URL: <https://www.ersj.eu/journal/574> (дата обращения: 03.03.2019).
5. Educating Academic Staff to Reorient Curricula in ESD / M. Biasutti [et al.] // International Journal of Sustainability in Higher Education. 2018. Vol. 19, No. 1. Pp. 179–196. DOI: <https://doi.org/10.1108/IJSHE-11-2016-0214>
6. *Hilty L. M., Huber P.* Motivating Students on ICT-Related Study Programs to Engage with the Subject of Sustainable Development // International Journal of Sustainability in Higher Education. 2018. Vol. 19, Issue 3. Pp. 642–656. DOI: <https://doi.org/10.1108/IJSHE-02-2017-0027>
7. A Mathematical Model of an Intelligent Information System for a Comparative Analysis of European Qualification Standards / V. S. Ereemeev [et al.] // Global Journal of Pure and Applied Mathematics. 2016. Vol. 12, No. 3. Pp. 2113–2132. URL: http://www.ripublication.com/gjpmv16/gjpmv12n3_14.pdf (дата обращения: 03.03.2019).
8. *Osadchyi V., Osadcha K., Ereemeev V.* The Model of the intelligence System for the Analysis of Qualifications Frameworks of European Countries // International Journal of Computing. 2017. Vol. 16, Issue 3. Pp. 133–142. URL: <http://computingonline.net/computing/article/view/896> (дата обращения: 03.03.2019).
9. Порівняння національних рамок кваліфікацій засобами веб-орієнтованої інтелектуальної інформаційної системи / В. В. Осадчий [и др.] // Інформаційні технології і засоби навчання. 2016. Т. 56, № 6. С. 121–136. DOI: <https://doi.org/10.33407/ilt.v56i6.1493>
10. *Fwa H. L.* An Architectural Design and Evaluation of an Affective Tutoring System for Novice Programmers // International Journal of Educational Technology in Higher Education. 2018. Vol. 15. DOI: <https://doi.org/10.1186/s41239-018-0121-2>
11. Python for Teaching Introductory Programming: A Quantitative Evaluation / A. Jayal [et al.] // Innovation in Teaching and Learning in Information and Computer Sciences. 2011. Vol. 10, Issue 1. Pp. 86–90. DOI: <https://doi.org/10.11120/ital.2011.10010086>
12. *Nowostawski M., McCallum S., Mishra D.* Gamifying Research in Software Engineering // Computer Applications in Engineering Education. 2018. Vol. 26, Issue 5. Pp. 1641–1652. DOI: <https://doi.org/10.1002/cae.21994>
13. *Бакунович М. Ф., Станкевич Н. Л.* Самоконтроль как базовый элемент профессиональной компетентности будущих IT-специалистов // Интеграция образования. 2018. Т. 22, № 4. С. 681–695. DOI: <https://doi.org/10.15507/1991-9468.093.022.201804.681-695>
14. A Cognitive Assistant for Learning Java Featuring Social Dialogue / M. Coronado [et al.] // International Journal of Human-Computer Studies. 2018. Vol. 117. Pp. 55–67. DOI: <https://doi.org/10.1016/j.ijhcs.2018.02.004>



15. *Eckerdal A.* Novice Programming Students' Learning of Concepts and Practise: Thesis for the degree of Doctor of Philosophy. Uppsala, 2006. 76 p. URL: <http://user.it.uu.se/~annae/FullAvh-Spikenheten.pdf> (дата обращения: 03.03.2019).
16. *Caspersen M. E.* Educating Novices in the Skills of Programming: Thesis for the PhD Degree. Aarhus, 2007. 311 p.
17. *Teague D.* Pedagogy of Introductory Computer Programming: A People-First Approach: Thesis for the Degree of Master of Information Technology (Research). Queensland, 2011. 129 p. URL: http://eprints.qut.edu.au/46255/1/Donna_Teague_Thesis.pdf (дата обращения: 03.03.2019).
18. *Осадчий В. В., Круглик В. С.* Эффективная организация содержания профессиональной подготовки для повышения уровня квалификаций будущих веб-программистов // Образовательные технологии и общество. 2015. Т. 18, № 4. С. 540–558. URL: <https://readera.ru/14062644> (дата обращения: 03.03.2019).
19. *Калитина В. В.* Методика ментального обучения программированию студентов информационных направлений подготовки // Вестник Красноярского государственного педагогического университета им. В. П. Астафьева. 2015. № 1 (31). С. 45–48.
20. *Ma L.* Investigating and Improving Novice Programmers' Mental Models of Programming Concepts: Thesis for the Degree of Doctor of Philosophy. Glasgow, 2007. 208 p. URL: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.721.8479&rep=rep1&type=pdf> (дата обращения: 03.03.2019).
21. *Sorva J.* Visual Program Simulation in Introductory Programming Education: Doctoral Thesis for the degree of Doctor of Science in Technology. Aalto, 2012. 422 p. URL: <http://lib.tkk.fi/Diss/2012/isbn9789526046266/isbn9789526046266.pdf> (accessed 03.03.2019).
22. *Truong N.* A Web-Based Programming Environment for Novice Programmers: Thesis for the Degree of Doctor of Philosophy. Queensland, 2007. 286 p. URL: http://eprints.qut.edu.au/16471/1/Nghi_Truong_Thesis.pdf (дата обращения: 03.03.2019).
23. *Kolgatin O.* Computer-Based Simulation of Stochastic Process for Investigation of Efficiency of Statistical Hypothesis Testing in Pedagogical Research // Інформаційні технології в освіті. 2016. Vol. 2 (27). Pp. 7–14. DOI: <https://doi.org/10.14308/ite000582>
24. *Хуторской А. В.* Методологические основания применения компетентностного подхода к проектированию образования // Высшее образование в России. 2017. № 12 (218). С. 85–91. URL: <https://vovr.elpub.ru/jour/article/view/1228/1047> (дата обращения: 03.03.2019).
25. *Гришко Л. В.* Концептуальні підходи до навчання основ програмування у вищій школі // Науковий часопис НПУ імені М.П. Драгоманова. Сер. № 2: Комп'ютерно-орієнтовані системи навчання. 2004. № 1 (8). С. 134–148.
26. *Еришов А. П.* О человеческом и эстетическом факторах в программировании // Кибернетика. 1972. № 5. С. 95–99.
27. *Мустафина Д. А., Мустафина Г. А., Матвеева Т. А.* Процесс формирования конкурентоспособности будущих инженеров-программистов // Международный журнал, прикладных и фундаментальных исследований. 2009. № 5. С. 51–55. URL: http://www.applied-research.ru/pdf/2009/05/2009_05_09.pdf (дата обращения: 03.03.2019).
28. *Орел Е. А.* Особенности интеллекта профессиональных программистов // Вестник Московского университета. Сер. 14: Психология. 2007. № 2. С. 70–79. URL: http://msupsy.j.ru/pdf/vestnik_2007_2/vestnik_2007-2_70-79.pdf (дата обращения: 03.03.2019).
29. *Kruhlyk V.* Satisfaction of Qualification Requirements of Employers Applied to Software Engineers in the Process of Training at Higher Educational Institutions // Інформаційні технології в освіті. 2017. № 1 (30). Pp. 71–80. DOI: <https://doi.org/10.14308/ite000620>

Поступила 11.03.2019; принята к публикации 02.09.2019; опубликована онлайн 31.12.2019.

Об авторах:

Круглик Владислав Сергеевич, профессор кафедры информатики и кибернетики Мелитопольского государственного педагогического университета имени Богдана Хмельницкого (72312, Украина, г. Мелитополь, ул. Гетманская, д. 20), доктор педагогических наук, доцент, **ORCID: <https://orcid.org/0000-0002-5196-7241>**, **Scopus ID: 56006715100**, **Researcher ID: V-3861-2017**, kryglikvlad@gmail.com

Осадчий Вячеслав Владимирович, заведующий кафедрой информатики и кибернетики Мелитопольского государственного педагогического университета имени Богдана Хмельницкого (72312, Украина, г. Мелитополь, ул. Гетманская, д. 20), доктор педагогических наук, профессор, **ORCID: <https://orcid.org/0000-0001-5659-4774>**, **Scopus ID: 57190217440**, **Researcher ID: I-8586-2016**, osadchyi@mdpu.org.ua

Заявленный вклад авторов:

Круглик Владислав Сергеевич – проведение исследования; анализ и обобщение результатов; содержательная характеристика данных и формулировка выводов.

Осадчий Вячеслав Владимирович – научное руководство и консультирование на всех этапах исследования; постановка и конкретизация проблемы исследования; проведение критического анализа.

Все авторы прочитали и одобрили окончательный вариант рукописи.

REFERENCES

1. Brinda T., Reynolds N., Romeike R., Schwill A. (ed.). KEYCIT 2014: Key Competencies in Informatics and ICT. Potsdam: Universitätsverlag Potsdam; 2015. Available at: <https://publishup.uni-potsdam.de/opus4-ubp/frontdoor/deliver/index/docId/7032/file/cid07.pdf> (accessed 03.03.2019). (In Eng.)
2. Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science. New York: ACM; 2013. (In Eng.) DOI: <https://doi.org/10.1145/2534860>
3. Kruglyk V.S., Osadchyi V.V. Structure of Professional Competence of Future Software Engineers. *Pedahohichnyi dyskurs = Pedagogical Discourse*. 2016; (21):69-74. Available at: <https://clck.ru/K8oTx> (accessed 03.03.2019). (In Ukr., abstract in Eng.)
4. Guskova N.D., Vdovin S.M., Krakovskaya I.N., Slushkina Yu.Yu. The Quality of Education as a Primary Concern of the Sustainable Development. *European Research Studies Journal*. 2016; 19(3B): 239-257. Available at: <https://www.ersj.eu/journal/574> (accessed 03.03.2019). (In Eng.)
5. Biasutti M., Makrakis V., Concina E., Frate S. Educating Academic Staff to Reorient Curricula in ESD. *International Journal of Sustainability in Higher Education*. 2018; 19(1):179-196. (In Eng.) DOI: <https://doi.org/10.1108/IJSHE-11-2016-0214>
6. Hilty L.M., Huber P. Motivating Students on ICT-Related Study Programs to Engage with the Subject of Sustainable Development. *International Journal of Sustainability in Higher Education*. 2018; 19(3): 642-656. (In Eng.) DOI: <https://doi.org/10.1108/IJSHE-02-2017-0027>
7. Ereemeev V.S., Osadchyi V.V., Gulynina E.V., Doneva O.V. A Mathematical Model of an Intelligent Information System for a Comparative Analysis of European Qualification Standards. *Global Journal of Pure and Applied Mathematics*. 2016; 12(3):2113-2132. Available at: http://www.ripublication.com/gjpm16/gjpmv12n3_14.pdf (accessed 03.03.2019). (In Eng.)
8. Osadchyi V., Osadcha K., Ereemeev V. The Model of the Intelligence System for the Analysis of Qualifications Frameworks of European Countries. *International Journal of Computing*. 2017; 16(3): 133-142. Available at: <http://computingonline.net/computing/article/view/896> (accessed 03.03.2019). (In Eng.)
9. Osadchyi V.V., Yermieiev V.S., Sharov S.V., Osadcha K.P., Koniukhov S.L. Comparison of National Qualifications Frameworks by Means of Web-Oriented Intelligent Information System. *Informatsiini tekhnologii i zasoby navchannia = Information Technologies and Learning Tools*. 2016; 56(6):121-136. (In Ukr., abstract in Eng.) DOI: <https://doi.org/10.33407/itlt.v56i6.1493>
10. Fwa H.L. An Architectural Design and Evaluation of an Affective Tutoring System for Novice Programmers. *International Journal of Educational Technology in Higher Education*. 2018; 15. (In Eng.) DOI: <https://doi.org/10.1186/s41239-018-0121-2>
11. Jayal A., Lauria S., Tucker A., Swift S. Python for Teaching Introductory Programming: A Quantitative Evaluation. *Innovation in Teaching and Learning in Information and Computer Sciences*. 2011; 10(1):86-90. (In Eng.) DOI: <https://doi.org/10.11120/ital.2011.10010086>
12. Nowostawski M., McCallum S., Mishra D. Gamifying Research in Software Engineering. *Computer Applications in Engineering Education*. 2018; 26(5):1641-1652. (In Eng.) DOI: <https://doi.org/10.1002/cae.21994>
13. Bakunovich M.F., Stankevich N.L. Self-Control as a Core Component of Professional Competence of it Students. *Integratsiya obrazovaniya = Integration of Education*. 2018; 22(4):681-695. (In Russ., abstract in Eng.) DOI: <https://doi.org/10.15507/1991-9468.093.022.201804.681-695>
14. Coronado M., Iglesias C.A., Carrera Á., Mardomingo A. A Cognitive Assistant for Learning Java Featuring Social Dialogue. *International Journal of Human-Computer Studies*. 2018; 117:55-67. (In Eng.) DOI: <https://doi.org/10.1016/j.ijhcs.2018.02.004>
15. Eckerdal A. Novice Programming Students' Learning of Concepts and Practise: Thesis for Degree of Doctor of Philosophy. Uppsala; 2006. Available at: <http://user.it.uu.se/~annae/FullAvh-Spikenheten.pdf> (accessed 03.03.2019). (In Eng.)
16. Caspersen M.E. Educating Novices in the Skills of Programming: Thesis for Degree of Doctor of Philosophy. Aarhus; 2007. (In Eng.)



17. Teague D. Pedagogy of Introductory Computer Programming: A People-First Approach: Thesis for Degree of Master of Information Technology (Research). Queensland; 2011. Available at: http://eprints.qut.edu.au/46255/1/Donna_Teague_Thesis.pdf (accessed 03.03.2019). (In Eng.)
18. Osadchyi V.V., Kruglyk V.S. [Effective Organization of the Content of Training to Improve the Skills of Future Web Programmers]. *Obrazovatelnyye tehnologii i obschestvo* = Educational Technologies and Society. 2015; 18(4):540-558. Available at: <https://readera.ru/14062644> (accessed 03.03.2019). (In Russ.)
19. Kalitina V.V. [The Technique of Mental Training in Programming for Students of Informatics Majors]. *Vestnik Krasnoyarskogo gosudarstvennogo pedagogicheskogo universiteta im. V.P. Astafeva* = Bulletin of Krasnoyarsk State Pedagogical University. 2015; (1):45-48. (In Russ.)
20. Ma L. Investigating and Improving Novice Programmers' Mental Models of Programming Concepts: Thesis for Degree of Doctor of Philosophy. Glasgow; 2007. Available at: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.721.8479&rep=rep1&type=pdf> (accessed 03.03.2019). (In Eng.)
21. Sorva J. Visual Program Simulation in Introductory Programming Education: Thesis for degree of Doctor of Science in Technology. Aalto; 2012. Available at: <http://lib.tkk.fi/Diss/2012/isbn9789526046266/isbn9789526046266.pdf> (accessed 03.03.2019). (In Eng.)
22. Truong N. A Web-Based Programming Environment for Novice Programmers: Thesis for Degree of Doctor of Philosophy. Queensland; 2007. Available at: http://eprints.qut.edu.au/16471/1/Nghi_Truong_Thesis.pdf (accessed 03.03.2019). (In Eng.)
23. Kolgatin O. Computer-Based Simulation of Stochastic Process for Investigation of Efficiency of Statistical Hypothesis Testing in Pedagogical Research. *Informacijni tehnologii v osviti* = Informational Technologies in Education. 2016; 2:7-14. (In Eng.) DOI: <https://doi.org/10.14308/ite000582>
24. Khutorskoy A.V. Methodological Foundations for Applying the Competence Approach to Designing Education. *Vyssheye obrazovaniye v Rossii* = Higher Education in Russia. 2017; (12):85-91. Available at: <https://vovr.elpub.ru/jour/article/view/1228/1047> (accessed 03.03.2019). (In Russ., abstract in Eng.)
25. Hryshko L.V. Conceptual Approaches to Teaching the Basics of Programming in High School. *Naukovyi chasopys NPU imeni M.P. Drakhomanova. Seriya № 2. Kompiuterno-orijentovani systemy navchannia* = Scientific Journal of the Dragomanov National Pedagogical University. Series 2: Computer-Based Learning Systems. 2004; (1):134-148. (In Ukr.)
26. Ershov A.P. [On the Human and Aesthetic Factors in Programming]. *Kibernetika* = Cybernetics. 1972; (5):95-99. (In Russ.)
27. Mustafina D.A., Mustafina G.A., Matveeva T.A. The Formation Process of the Future Part-Programming Engineers' Competitiveness. *Mezhdunarodnyy zhurnal, prikladnyh i fundamentalnyh issledovaniy* = International Journal of Applied and Fundamental Research. 2009; (5):51-55. Available at: http://www.applied-research.ru/pdf/2009/05/2009_05_09.pdf (accessed 03.03.2019). (In Russ., abstract in Eng.)
28. Orel E.A. [Intelligence Features of Professional Programmers]. *Vestnik Moskovskogo universiteta. Seriya 14. Psihologiya* = Moscow University Bulletin. Ser. 14. Psychology. 2007; (2):70-79. Available at: http://msupsyj.ru/pdf/vestnik_2007_2/vestnik_2007-2_70-79.pdf (accessed 03.03.2019). (In Russ.)
29. Kruhlyk V. Satisfaction of Qualification Requirements of Employers Applied to Software Engineers in the Process of Training at Higher Educational Institutions. *Informacijni tehnologii v osviti* = Journal of Information Technologies in Education. 2017; (1):71-80. (In Eng.) DOI: <https://doi.org/10.14308/ite000620>

Submitted 11.03.2019; revised 02.09.2019; published online 31.12.2019.

About the authors:

Vladyslav S. Kruglyk, Professor of Chair of Informatics and Cybernetics, Bogdan Khmelnytsky Melitopol State Pedagogical University (20 Hetmanska St., Melitopol 72312, Ukraine), Dr. Sci. (Pedagogy), Associate Professor, **ORCID: <https://orcid.org/0000-0002-5196-7241>**, **Scopus ID: 56006715100**, **Researcher ID: V-3861-2017**, kryglykvlad@gmail.com

Viacheslav V. Osadchyi, Head of Chair of Informatics and Cybernetics, Bogdan Khmelnytsky Melitopol State Pedagogical University (20 Hetmanska St., Melitopol 72312, Ukraine), Dr. Sci. (Pedagogy), Professor, **ORCID: <https://orcid.org/0000-0001-5659-4774>**, **Scopus ID: 57190217440**, **Researcher ID: I-8586-2016**, osadchyi@mdpu.org.ua

Contribution of the authors:

Vladyslav S. Kruglyk – conducting research; analyzing and summarizing the results; informative description of the data and the formulation of conclusions.

Viacheslav V. Osadchyi – scientific management and counseling at all stages of the study; formulation and specification of the research problem; conducting critical analysis.

All authors have read and approved the final manuscript.