

6. Brackett K.E., Pal S. Creating high-impact visualisations // Applied Marketing Analytics. – 2015. Т. 1. № 4. Р. 331-341.
7. Павлов А.А. Компьютеризация инфографического моделирования // Интернет-журнал Науковедение. – 2012. – №. 2 (11).
8. Tufte, Edward R (2001) [1983], The Visual Display of Quantitative Information (2nd ed.), Cheshire, CT: GraphicsPress, ISBN 0-9613921-4-2.
9. Create Easy Infographics, Reports, Presentations | Piktochart. [Электронный ресурс]. URL: <http://piktochart.com> (дата обращения: 21.02.2016).
10. Create and share visual ideas online | Easel.ly. [Электронный ресурс]. URL: <http://www.easel.ly/> (дата обращения: 21.02.2016).
11. Create online charts & infographics | Infogr.am. [Электронный ресурс]. URL: <https://infogr.am> (дата обращения: 21.02.2016).
12. Online Diagram Software to draw Flowcharts, UML & more | Creately [Электронный ресурс]. URL: <http://creately.com/> (дата обращения: 21.02.2016).
13. Yankovskaya A., Krivdyuk N. Cognitive Graphics Tool Based on 3-Simplex for Decision-Making and Substantiation of Decisions in Intelligent System // Proceedings of the IA STED International Conference Technology for Education and Learning TEL '2013. Marina del Rey, USA, November 11-13, 2013. P. 463-469.
14. Евсюков А.А. Применение технологии OLAP для построения тематических карт / А.А. Евсюков // Сборник научных трудов VIII Международной научно-практической конференции «Современные информационные технологии и ИТ-образование». – Москва: МГУ, 2013. С. 377-384.
15. Евсюков А.А. Динамическое формирование картографических слоев в информационно-аналитических системах / А.А. Евсюков // Вестник Сибирского государственного аэрокосмического университета имени академика М. Ф. Решетнева. 2011. № 1. С. 15-20.

Application of infographic simulation methods to represent multidimensional data

Evsyukov Alexander Anatolyevich, candidate of technical sciences, major scientific researcher

Korobko Anna Vladimirovna, candidate of technical sciences, scientific researcher

Institute of Computational Modeling Siberian Branch of Russian Academy of Sciences

The paper presents the survey of existing approaches and tools of infographics, the types of infogram. The methods integrating infographic simulation and online analytical processing technology are observed.

Keywords: infographics, infogram, OLAP, graphical representation of multidimensional data.

УДК 004

МЕТОДЫ И СРЕДСТВА ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ В ВУЗЕ

Елена Викторовна Касьянова, к.ф.-м. н, доцент, старший научный сотрудник

Тел.: 8 383 330 4047, e-mail: kev@iis.nsk.su

Институт систем информатики им. А.П. Ершова СО РАН,

Новосибирский государственный университет

<http://www.iis.nsk.su>

Рассматриваются методы и средства обучения программированию, разрабатываемые в лаборатории конструирования и оптимизации программ Института систем информатики им. А.П. Ершова СО РАН и на кафедре программирования Новосибирского государственного университета.

Ключевые слова: адаптивные гипермедиа-системы, курс по программированию, облачные вычисления, параллельное программирование, практикум по программированию, функциональное программирование, язык Sisal, язык Zonnon.

Работа выполняется при частичной финансовой поддержке РФФИ, грант 15-07-02029.

Стремительное развитие информационных технологий и проникновение их во все стороны жизни общества и во все сферы производственной деятельности приводит к тому, что выпускнику вуза, чтобы стать успешным в своей дальнейшей деятельности, не достаточно освоить существующие пользовательские технологии и получить навыки поиска готовых решений, а необходимо научиться решать возникающие задачи с помощью программирования. Однако овладение умением программировать все еще остается весьма сложной задачей для многих студентов.

Системы дистанционного обучения в настоящее время активно исследуются и развиваются. Выгоды сетевого обучения ясны: аудиторная и платформенная независимости. Сетевое обучающее программное обеспечение, один раз установленное и обслуживаемое в одном месте, может использоваться в любое время и по всему миру тысячами учащихся, имеющих любой компьютер, подключенный к Интернету. Тысячи программ сетевого обучения и других образовательных приложений стали доступны в сети Интернет за последние годы. Проблема состоит в том, что большинство из них являются не более чем статичными гипертекстовыми страницами и слабо поддерживают проблемный подход к обучению. Появившиеся в последнее время адаптивные гипермедиа-системы существенно повышают возможности обучающих систем [1; 2]. Целью этих систем является персонализация гипермедиа-системы, ее настройка на особенности индивидуальных пользователей.



Е.В. Касьянова

Всё большую популярность набирают облачные сервисы, которые предоставляют различные возможности, в том числе и для обучения. Такие облачные хостинги, как Amazon и Cloud9, предоставляют вычислительные ресурсы, не требуя дополнительной установки библиотек поддержки времени исполнения и позволяя выполнять программный код внутри браузера, причем безопасно для пользователя. Растущее количество сервисов в сети Интернет, предоставляющих различные услуги, прямо или косвенно связанные с вычислениями, говорит о популярности такого подхода. Очень привлекательно, не обладая компилятором для того или иного языка программирования, иметь возможность в любой момент зайти на соответствующую страницу в сети и выполнить интересующий код на этом языке. Подобные сервисы варьируются от совершенно аскетических, рассчитанных на исключительно короткие программы, до предоставляющих богатую среду разработки с группировкой по проектам, подсветкой синтаксиса и т. д.

Цель исследований, представленных в данной статье, — разработать методы и средства обучения программированию, которые позволят сделать процесс обучения программированию в вузе более индивидуальным, доступным и эффективным. Рассматривается практикум по программированию и вводный курс начального обучения программированию на базе языка Zonnon. Описываются исследования по разработке адаптивных методов и систем дистанционного обучения, а также визуальной облачной среды поддержки параллельного и функционального программирования.

1. Практикум на компьютере

При обучении программированию наиболее важным, на наш взгляд, является начальный этап, на котором обучаемый должен овладеть навыками точного формулирования алгоритмов на языке высокого уровня, что невозможно сделать, прочитав

лишь несколько руководств или прослушав курс лекций по программированию. Необходима практика конструирования алгоритмов, и здесь не обойтись без подходящего набора примеров и задач.

Поэтому неслучайно уже много лет на механико-математическом факультете Новосибирского государственного университета (НГУ) проводится практикум по программированию для студентов первого курса. Он проходит для студентов первого курса во втором семестре одновременно с основным курсом по программированию и на данный момент ориентируется на использование языков C / C++ и систем Visual Studio и Borland Developer Studio. Это не значит, конечно, что для решения индивидуальных заданий, выполняемых студентами в рамках общего практикума, не подходят другие реализации языков C / C++ или другие языки программирования.

Основной задачей практикума является не только и не столько обучение студентов собственно записи (кодированию) известного алгоритма на языке программирования высокого уровня, а практическое закрепление знаний, получаемых в курсе по программированию, и овладение общими методами, приемами и навыками технологии решения задач на компьютере [3]. Тематика заданий общего практикума определяется не столько конкретными областями знаний, из которых должны браться задачи для их решения на компьютере (это в большей степени имеет место в специализированных практикумах на старших курсах), а всеми видами работ, которые должен освоить студент, чтобы научиться создавать качественные (эффективные, наглядные и надежные) программы. В большинстве случаев задача, решаемая во время общего практикума, — это задача невычислительного характера, требующая разработки алгоритма, обработки сложных структур данных и создания дружественного интерфейса. Как правило, решаемая задача имеет краткую и точную (содержательную) формулировку и допускает большое разнообразие решений, из которых студент должен выбрать по возможности лучшее.

Каждое индивидуальное задание, выполняемое студентом, — это самостоятельная, как правило, комбинаторная или логическая задача с краткой и четкой формулировкой, не содержащей описания алгоритма. Тематические задачи разбиты на пять разделов: графы и системы дорог; грамматики, языки и автоматы; формулы и программы; геометрия; игры и модели. В ходе практикума каждый студент решает пять индивидуальных задач, по одной из каждого раздела. Для удобства все задания практикума в учебном пособии [3] разбиты на три группы: обычной, повышенной и пониженной сложности.

Выполнение задания включает следующие виды работ: анализ условия задачи и выработку подхода к ее решению; пошаговую разработку (на основании выбранного подхода) алгоритма решения и его описание; обоснование алгоритма; выбор и обоснование представления для входных, выходных и промежуточных данных; кодирование алгоритма, т. е. его запись на языке C; выбор и обоснование набора тестов, на которых будет проверяться программа; отладку программы и демонстрацию ее правильной работы на выбранном наборе тестов. Это разбиение условно в том смысле, что фактически некоторые виды работ тесно переплетаются и выполнение их обычно составляет единый процесс. Например, строить набор тестов удобнее одновременно с построением самого алгоритма, а обосновывать правильность работы алгоритма удобно путем детальной демонстрации процесса его построения. Практические рекомендации по выполнению всех перечисленных работ приведены в учебном пособии и составляют большую его часть.

2. Вводный курс программирования на базе языка Zonnon

Zonnon – это новый универсальный язык программирования в семействе языков Паскаль, Модула-2 и Оберон, работа над которым ведется в Цюрихском институте информатики [4, 5]. Он сохраняет стремление к простоте, ясному синтаксису и независимости концепций, а также уделяет внимание параллельности и легкости композиции и

выражения. Унификация абстракций является стержнем проектирования языка Zonnon, и она отражается в его концептуальной модели, основанной на модулях, объектах, определениях и реализациях. Язык Zonnon содержит такие новые черты, как активность в объектах, основанный на межобъектном взаимодействии диалог, перегрузка операций и обработка исключительных ситуаций. Он специально разрабатывается как платформенно-независимый язык. Поскольку язык Zonnon задуман как дальнейшая эволюция языка Оберон, авторы стремятся сохранить такие важные черты языка Оберон и его преемников, как компактность языка, ясность, недвусмысленность и ортогональность его основных понятий. Вместе с тем, чтобы создать современную альтернативу языку Оберон, авторы внесли в язык ряд изменений, основными из которых являются: более развитая модульная структура языка; продвинутая и одновременно простая и ясная объектная модель; концепция активных объектов.

«Общеалгоритмическая» часть языка Zonnon практически полностью повторяет соответствующие части Оберона, поэтому данный язык можно рассматривать как естественную замену Оберону и Паскалю там, где они традиционно используются для обучения программированию. Компилятор и среду программирования для языка Zonnon можно скачать с сайта разработчиков языка [5]. Там же всегда можно найти последние новости о языке, его реализации и применении, а также обратиться за консультацией к разработчикам.

Нами разработан вводный курс по программированию, который предназначен для обучения основным методам построения корректных, эффективных и надежных программ на базе языка Zonnon и платформы Microsoft.NET [6]. Курс ориентирован на широкий круг лиц, обучающихся методам программирования, в первую очередь, на студентов НГУ, других вузов и средних учебных заведений, а также школьников, желающих углубить свои знания по программированию. Он предназначен главным образом для тех учебных заведений, в которых в настоящее время используется язык Паскаль в качестве языка начального обучения программированию и есть желание перейти к более современному курсу программирования, охватывающему концепции языков программирования нового поколения, таких как Java и C#, но осуществить этот переход плавно, без резкого изменения сложившегося стиля преподавания программирования.

Курс опирается на опыт преподавания основного курса по программированию для студентов механико-математического факультета НГУ [7]. Это семестровый курс, который предполагает еженедельно шесть часов аудиторных занятий: два часа лекционных занятий, два часа семинарских занятий у доски и два часа практических занятий за компьютером.

В основу курса положен принцип концентрического изложения материала. Предполагается, что с первых занятий студенты начинают упражняться в составлении программ, которые могут реально выполняться на доступных компьютерах, и постепенно овладевают навыками разработки на языке Zonnon линейных, ветвящихся и итеративных алгоритмов, алгоритмов с массивами и процедурами. Также постепенно, одновременно с расширением класса решаемых задач, студенты углубляют свои знания о языке Zonnon. Другими основными методическими и технологическими принципами, на которых базируется разработанный курс, являются следующие.

1. Принцип обучения конструированию программ на подробно комментированных образцах решения тщательно подобранных задач. Назначение примеров — не только дать образцы и описать основные схемы алгоритмов, но и на сравнительном анализе разных решений одной и той же задачи познакомить студента с такими понятиями, как эффективность, наглядность и надежность решения.

2. Принцип доказательного программирования, когда программа строится вместе с доказательством ее правильности. Для этого в курсе вводятся понятия промежуточных утверждений и инвариантов, а в разрабатываемых алгоритмах решения задач такие утверждения записываются в форме программных комментариев.

3. Принцип пошаговой разработки программ, когда программа строится из формальной спецификации задачи с помощью мелких формально проверяемых шагов преобразования.

4. Принцип модульного программирования, позволяющий проектировать, разрабатывать и собирать программу по частям и с использованием библиотек уже готовых частей.

5. Принцип объектно-ориентированного программирования, позволяющий разработчикам программ легко создавать все более сложные приложения с помощью инкапсуляции, наследования и полиморфизма.

3. Адаптивные методы и системы дистанционного обучения

Класс адаптивных гипермедиа-систем состоит из всех таких гипертекстовых и гипермедиа-систем, которые отражают некоторые особенности пользователя в его модели и применяют эту модель для адаптации различных видимых для пользователя аспектов системы [1; 2]. Таким образом, каждый пользователь имеет свою собственную картину и индивидуальные навигационные возможности для работы с адаптивной гипермедиа-системой.

Способности к адаптации стали особенно важны для Web-систем дистанционного обучения с тех пор, как обучаемые стали учиться в основном самостоятельно (обычно из дома). Интеллектуальное и личное содействие, которое могут дать учитель или студент-сокурсник при обычном (аудиторном) обучении, при дистанционном обучении нелегко достижимо. Адаптивность важна для программного обеспечения дистанционного обучения еще и потому, что оно должно использоваться намного более разнообразным множеством студентов, чем любое “однопользовательское” учебное приложение. Сетевое программное обеспечение, разработанное для одного класса пользователей (с определенным складом ума и с одним уровнем знаний), может совсем не подойти другим обучаемым.

Выделяются следующие характеристики пользователя обучающей системы, важные для ее адаптации: цель (или задача) пользователя, уровень его знаний, уровень его подготовки, имеющийся опыт работы пользователя с данной гипермедиа-системой, набор (система) предпочтений пользователя, личностные характеристики пользователя и характеристики пользовательской среды.

Адаптация в адаптивной гипермедиа может состоять в настройке содержания очередной страницы (адаптация на уровне содержания) или в изменении ссылок с очередной страницы, индексных страниц и страниц карт (адаптация навигации). Основные цели (методы) адаптации на уровне содержания – это дополнительные объяснения, предварительные объяснения, сравнительные объяснения, варианты объяснений и сортировка. Для достижения целей адаптации: на уровне адаптации разработаны такие техники, как условный текст, эластичный текст, варианты страниц и варианты фрагмента, и технология, основанная на фреймах. Основные цели (методы) адаптации навигации – это глобальное руководство, локальное руководство, поддержка локальной ориентации, поддержка глобальной ориентации, управление индивидуализированными представлениями, а основные технологии адаптивной навигационной поддержки – это полное руководство, адаптивная сортировка ссылок, адаптивное сокрытие ссылок, адаптивное аннотирование ссылок, адаптивное генерирование ссылок и адаптация карты.

Нами разработан проект адаптивной среды дистанционного обучения WAPE, поддерживающей активное индивидуальное обучение программированию в рамках проблемного подхода и соединяющей возможности адаптивных гипермедиа-систем и интеллектуальных обучающих систем [9; 10]. Помимо студентов система поддерживает таких пользователей, как инструкторы, лекторы и администраторы. Все они осуществляют доступ к системе через стандартный Web-браузер, но различаются по предоставляемым им возможностям. Система поддерживает следующие три уровня процесса обучения. Это — изучение студентом теоретического материала, тестирование систе-

мой знаний студента и выполнение студентом учебных задач (заданий и упражнений). Третий уровень рассматривается нами как основной в использовании системы. Изучить курс – это значит выполнить набор индивидуальных задач.

Многие адаптивные системы отслеживают движение пользователя по гиперкнижке. Хотя это оправданный подход, его недостатком является трудность измерения знания, приобретенного пользователем во время чтения Web-страницы. Вместо этого мы используем для обновления модели знаний только успехи и неудачи студента, проявленные им при решении задач: тестов, заданий и упражнений.

Другое важное отличие нашего подхода от традиционного состоит в том, что мы не пытаемся оценивать успех студента в изучении курса на основании уровня знаний в его модели. Поэтому в нашей системе достижение определенного уровня знаний студентом не позволяет ему завершить изучение курса с определенной оценкой, а лишь дает студенту возможность приступить к решению той или иной задачи из его индивидуального набора. Все это мотивировано проблемным подходом к обучению, который поддерживается системой WAPE.

4. Визуальная облачная среда для поддержки функционального и параллельного программирования

Параллельные вычисления в настоящее время являются одной из главных парадигм современного программирования и охватывают чрезвычайно широкий круг вопросов разработки программ. Ввиду значительно более сложной природы параллельных вычислений по сравнению с последовательными большое значение приобретают методы автоматизации разработки параллельного программного обеспечения, основанные на применении техники формальных моделей, спецификаций и преобразований параллельных программ.

Используя традиционные языки и методы, очень трудно разработать высококачественное, переносимое программное обеспечение для параллельных компьютеров. В частности, параллельное программное обеспечение не может быть разработано с малыми затратами на последовательных компьютерах и потом перенесено на параллельные вычислительные системы без существенного переписывания и отладки. Поэтому высококачественное параллельное программное обеспечение может разрабатываться только небольшим кругом специалистов, имеющих прямой доступ к дорогостоящему оборудованию.

Однако, используя языки программирования с неявным параллелизмом, такие как функциональный язык Sisal [10], можно преодолеть этот барьер и предоставить широкому кругу прикладных программистов, не имеющих достаточного доступа к параллельным вычислительным системам, но являющихся специалистами в своих прикладных областях, возможность быстрой разработки высококачественных переносимых параллельных алгоритмов на своем рабочем месте. Функциональная семантика языков программирования с неявным параллелизмом гарантирует детерминированные результаты для параллельной и последовательной реализации то, что невозможно гарантировать для традиционных языков, подобных языку Фортран. Пропадает необходимость переписывания исходного кода при переносе его с одного компьютера на другой. Гарантировано, что программа с неявным параллелизмом, правильно исполняющаяся на персональном компьютере, будет давать те же результаты при ее исполнении на высокоскоростном параллельном или распределенном вычислителе. Более того, по сравнению с императивными языками (подобными языку Фортран) функциональные языки, такие как Sisal, упрощают работу программисту. В функциональной программе программист должен только специфицировать результаты вычислений и может переложить большую часть работ по их организации на компилятор, который отвечает за отображение алгоритма на определенную архитектуру вычислителя (включая планирование команд, передачу данных, синхронизацию вычислений, управление памятью и т.

д.). По сравнению с другими функциональными языками язык Sisal поддерживает типы данных и операторы, присущие научным вычислениям, такие как циклы и массивы.

Создаваемая в лаборатории конструирования и оптимизации программ Института систем информатики им. А.П. Ершова СО РАН (ИСИ СО РАН) облачная интегрированная визуальная среда параллельного программирования использует язык Cloud Sisal, который базируется на языке Sisal 3.2 [11], и внутреннее теоретико-графовое представление функциональных и параллельных программ, ориентированное на их визуальную обработку [12]. Цель проекта [13] – дать возможность широкому кругу лиц, находящихся в удаленных населенных пунктах или в местах с недостаточными вычислительными средствами, но имеющих выход в Интернет, дистанционно и без установки дополнительного программного обеспечения на своих недорогих вычислительных устройствах в визуальном стиле создавать и отлаживать переносимые параллельные программы на языке Cloud Sisal, а также затем дистанционно (в облаке) осуществлять эффективное решение своих задач, исполняя на некоторых супервычислителях, доступным им по сети, созданные и отлаженные переносимые Cloud-Sisal-программы, предварительно адаптировав их под используемые супервычислители с помощью облачного оптимизирующего кросс-компилятора, предоставляемого средой.

Заключение

В статье рассмотрены методы и средства обучения программированию, разрабатываемые в лаборатории конструирования и оптимизации программ ИСИ СО РАН и на кафедре программирования НГУ. Часть из них уже успешно применяется в учебном процессе, в частности на механико-математическом факультете НГУ. Другие – пока только ждут своего применения, чтобы привести в учебный процесс вузов современные технологии дистанционного обучения и облачных вычислений.

Автор благодарен своим коллегам по кафедре программирования НГУ и лаборатории конструирования и оптимизации программ ИСИ СО РАН, особенно доктору физ.-мат. наук, проф. В.Н. Касьянову.

Литература

1. Brusilovsky P. Adaptive hypermedia // User Modeling and User-Adapted Interaction. – 2001. Vol. 11. P. 87-110.
2. Касьянов В.Н., Касьянова Е.В. Адаптивные системы и методы дистанционного обучения // Информационные технологии в высшем образовании. 2004. Т. 1. N. 4. С. 40-60.
3. Касьянов В.Н., Касьянова Е.В. Практикум по программированию. – Новосибирск: НГУ, 2013. 198 с.
4. Zonnon programming language & compiler. Официальный сайт разработчиков языка. – [Электронный ресурс]. URL <http://www.zonnon.ethz.ch>.
5. Касьянов В.Н., Касьянова Е. В. Язык программирования Zonnon. – Новосибирск, НГУ, 2010. 120 с.
6. Сайт русскоязычной библиотеки учебных курсов академической программы MSDN Academic Alliance. – [Электронный ресурс]. URL <http://www.microsoft.com/Rus/Msdnaa/Curricula/Default.aspx>.
7. Касьянов В.Н. Курс программирования на Паскале в заданиях и упражнениях. – Новосибирск: НГУ, 2001. 448 с.
8. Касьянова Е.В. Адаптивные методы и средства поддержки дистанционного обучения программированию. – Новосибирск, ИСИ СО РАН, 2007. 170 с.
9. Kasyanov V.N., Kasyanova E.V. WAPE – a system for distance learning of programming // IFIP International Federation for Information Processing. – Boston: Springer, 2008. V. 261. P. 355-356.
10. Gaudiot J.-L., DeBoni T., Feo J., et al The Sisal project: real world functional programming // Lecture Notices in Computer Science. 2001. V. 1808. P. 84-72.
11. Kasyanov V.N. Sisal 3.2: functional language for scientific parallel programming // Enterprise Information Systems. 2013. V. 7. N. 2. P. 227-236.

12. Касьянов В.Н., Касьянова Е.В. Визуализация информации на основе графовых моделей // Научная визуализация. 2014. Т. 6. № 1. С. 31-50.

13. Kasyanov V.N., Kasyanova E.V. Cloud system of functional and parallel programming for computer science education // Proceedings of 2015 2nd International Conference on Creative Education (ICCE 2015), June 27-28, 2015, London, UK. – SMSSI, 2015. P. 270-275.

Methods and tools of teaching programming in the institutes of higher education

Elena Kasyanova, Ph.D, Associate Professor, Chief Researcher

The methods and tools for teaching programming developed by the Program Construction and Optimization Laboratory of the A.P. Ershov Institute of Informatics Systems and by the Programming Department of Novosibirsk State University are considered.

Keywords – adaptive hypermedia-systems, course of programming, cloud computing, parallel programming, practice of programming, functional programming, Sisal language, Zonnon language

УДК 004.42

ФОРМИРОВАНИЕ ЗАДАНИЙ В УЧЕБНО-ИССЛЕДОВАТЕЛЬСКОЙ СИСТЕМЕ КОНСТРУКТОРА БОРТОВОЙ АППАРАТУРЫ

Андрей Юрьевич Колдырев, инженер

Тел.: 8 913 198 4994, e-mail: Raventus@icm.krasn.ru

Евгений Андреевич Грузенко, м.н.с.

Тел.: 8 391 290 7452, e-mail: Gruzenko@icm.krasn.ru

*Институт вычислительного моделирования СО РАН
icm.krasn.ru*

В статье рассматривается решение проблемы формирования заданий в обучающих системах для развития как теоретических, так и практических навыков работы конструкторов в стандартных и нестандартных ситуациях. Представлена учебно-исследовательская система для конструкторов бортовой аппаратуры командно-измерительной системы космического аппарата.

Ключевые слова: учебно-исследовательская система, имитационная модель, бортовая аппаратура

Введение

Разработка систем обучения для конструкторов наукоёмких устройств – сложный итеративный процесс. Разработчикам обучающей системы необходимо учесть множество факторов



А.Ю. Колдырев

проблемной области для повышения квалификации обучаемых специалистов [1]. Важным условием подтверждения квалификации специалиста наряду с теоретическими знаниями является умение работать как в стандартных, так и нестандартных (аварийных) ситуациях. В то же время большинство существующих систем обучения направлены на передачу лишь теоретических знаний, а не на формирование практических навыков [2].

В статье рассмотрена проблема формирования заданий в учебно-исследовательской



Е.А. Грузенко

системе для специалистов-конструкторов командно-измерительной системы космического аппарата. Для выработки умений и навыков работы конструктора в стандартных и нестандартных ситуациях система оснащена программно-тренажёрными имитационными инструментами, позволяющими имитировать ситуации, в которые может попадать специалист в процессе работы [3].