

*Мосьяков Владимир Евгеньевич*

канд. техн. наук, доцент

ФГБОУ ВО «Российский экономический  
университет им. Г.В. Плеханова»

г. Москва

## **МЕТОДИЧЕСКИЕ АСПЕКТЫ ПРЕПОДАВАНИЯ ДИСЦИПЛИНЫ «ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ»**

*Аннотация:* в статье рассмотрены методические вопросы обучения технологии программирования (ТП). Методики технологии направлены на обучение инструментарию программирования и формирование у студентов навыков по использованию таких средств в профессиональной деятельности. По мнению автора статьи, это позволяет формировать определённые компетенции, такие как владение основными методами, способами и средствами создания программных систем, способность осуществлять сбор, анализ и обработку данных, необходимых для решения поставленных профессиональных задач, умение использовать для решения аналитических и исследовательских задач современные технические средства и ТП.

*Ключевые слова:* компетенция, информатика, технология, система, программно-аппаратная система.

Сегодня задачи реформирования системы высшего профессионального образования сводятся в основном к решению проблемы содержательного характера. Повсеместное активное использование информационных технологий требуют определённого пересмотра основ образовательной системы. Стремительное развитие современных информационных технологий внедряющихся в профессиональную и повседневную деятельность меняют требования, предъявляемые к обучению соответствующих дисциплин.

В данной работе предлагается проводить обучение технологиям программирования в форме деловой игры включающей три собирательных образа: Программист (студент), Руководитель (преподаватель) и Заказчик (преподаватель).

Деловая игра включает следующие объекты (рис. 1):



Рис. 1. Объекты обучения

На первом этапе обучения предлагается сформировать представление студентов о программировании, как процесса создания и сопровождения программ [5]. Этот процесс включает анализ и постановку задачи, проектирование программы, построение алгоритмов, разработку структур данных, написание кода программы, отладку, тестирование, документирование, модификацию и сопровождение.

Надо обратить внимание студентов на то, что цель работы программиста – создание и сопровождение законченных программно-аппаратных систем. Каждый человек, участвующий в разработке или сопровождении системы должен представлять себе работу всей системы и особенно хорошо знать подпрограммы, взаимодействующие с его программой. Знание системы во всей её полноте, умение общаться с ней и варьировать её компоненты – вот что является целью работы программиста.

Вторым важным шагом при проведении занятий со студентами является определение интегрированной среды разработки (ИСР), как системы программных средств, призванных помогать программисту. ИСР обычно включает в себя текстовый редактор, компилятор, интерпретатор, средства автоматизации разработки и сборки программного обеспечения и отладчик. Иногда также содержит средства для интеграции с системами управления версиями и разнообразные инструменты для упрощения конструирования графического интерфейса пользователя. Многие современные среды разработки также включают окно просмотра программных классов, инспектор объектов и диаграмму иерархии классов – для использования при объектно-ориентированной разработке программ. Большинство современных ИСР предназначены для разработки программ на нескольких языках программирования одновременно и включают в себя возможность визуального редактирования интерфейса программы.

Текстовый редактор используется для ввода исходного кода в ИСР и ориентирован на работу с последовательностью символов в текстовых файлах. Редакторы ИСР (редакторы кода) обеспечивают выделение синтаксических конструкций текста с использованием различных цветов, шрифтов и начертаний, сортировку строк, шаблоны, конвертацию кодировок, показ кодов символов.

Важная часть ИСР – отладчик (модуль среды, предназначенный для поиска ошибок в программе), позволяющий выполнять пошаговую трассировку, отслеживать, устанавливать или изменять значения переменных в процессе выполнения программы, устанавливать и удалять контрольные точки или условия остановки.

При совместной разработке проектов средней и высокой степени сложности для облегчения работы с изменяющейся информацией в ИСР используется система управления версиями. Она позволяет хранить несколько версий одного и того же документа, возвращаться к более ранним версиям и многое другое.

Примерами универсальных ИСР являются Microsoft Visual Studio, Eclipse, KDevelop.

Студенты должны понимать, что программирование как вид деятельности имеет два уровня: кодирование элементарных операций, что требует очень точной скрупулёзной работы, и уровень охвата глобальных информационно-программных структур.

Сегодня умение писать программы по заданным спецификациям недостаточно – требуется умение создавать программный продукт и сопровождать его. Помимо этого, деятельность программиста, рассматриваемая с точки зрения экономики, требует снижения себестоимости (в основном за счёт затрат на сопровождение). Отсюда следует, что программный продукт надо сразу делать сразу начисто без черновых вариантов. Для реализации такого подхода требуются квалифицированные программисты, умеющие изучить систему в целом и подробности взаимодействия программ внутри системы.

Следующим шагом методики является создание надёжных программ.

Стандартный словарь терминов программного инжиниринга определяет надёжность программного обеспечения как способность системы или компонента выполнять требуемые функции в заданных условиях на протяжении указанного периода времени [4]. Надёжность программ достигается правильной организацией их проектирования и тщательным тестированием.

Здесь необходимо обратить внимание студентов на то, что надёжность и стоимость сопровождения оказываются тесно связанными характеристиками. Ненадёжную программу необходимо дорабатывать в процессе сопровождения (а иногда и полностью отказаться от её эксплуатации), что сильно увеличивает затраты на программную систему (ПС).

Проектировании ПС обязательно включает в себя два аспекта:

- 1) разработка системы в деталях;
- 2) организация системы в целом.

Детальная разработка вопроса применения компьютера – это творческая деятельность программиста.

Что касается системы, то она должна иметь иерархическую структуру и быть организованной по модульному принципу [6].

Процесс разработки и эксплуатации программной системы (жизненный цикл) можно разбить на несколько этапов [1]:

- 1) постановка задачи;
- 2) составление алгоритмов;
- 3) кодирование;
- 4) отладка;
- 5) тестирование;
- 6) сопровождение.

Как правило, длительность и трудоёмкость этапов 1 и 6 (как вместе, так и по отдельности) намного превышает те же показатели для этапов 2–5.

Этап кодирования наименее трудоёмкий среди всех видов программирования. Согласно теории структурного программирования, для записи любого алгоритма достаточно всего три оператора и ещё два оператора из соображений удобства записи [3].

Тесно связанные между собой этапы 2–4 можно объединить в особый вид деятельности – проектирование программных систем. Этим достигается цель существенного облегчения этапов 5 и 6 (тестирование и сопровождение). Основная идея методов проектирования заключается в итеративной организации работ.

Создание больших программных систем [6] заключается не только и не столько в составлении алгоритмов, а в организации данных, т.е. организации признаков, условий, предикатов и других данных, необходимых для решения задачи [2].

Именно здесь необходимо искусство программиста для понимания и эффективного использования больших информационно-логических структур. Алгоритм программы строится на базе неявной части обрабатываемой информации, содержащей нужные для реализации каждой функции признаки и критерии.

Таким образом всё вышеизложенное позволяет обучить студентов не только технологиям программирования, но и основам проектирования больших информационно-логических структур, а также, позволяет в интерактивной форме освоить основные виды профессиональной деятельности в расчетно-экономической,

аналитической и научно-исследовательской областях и сформировать следующие общекультурные и профессиональные компетенции:

– владеет основными методами, способами и средствами получения, хранения, переработки информации, имеет навыки работы с компьютером как средством управления информацией, способен работать с информацией в глобальных компьютерных сетях;

– способен осуществлять сбор, анализ и обработку данных, необходимых для решения поставленных экономических задач;

– способен использовать для решения аналитических и исследовательских задач современные технические средства и информационные технологии.

### ***Список литературы***

1. Брукс Ф.П. (мл.). Как проектируются и создаются программные комплексы / Ф.П. Брукс. – М.: Наука, 1979. – 128 с.

2. Вирт Н. Алгоритмы + структуры данных = программы / Н. Вирт. – М.: Мир, 1985. – 406 с.

3. Йодан Э. Структурное проектирование и конструирование программ / Э. Йодан. – М.: Мир, 1979. – 415 с.

4. Майерс Г. Надёжность программного обеспечения / Г. Майерс. – М.: Мир, 1980. – 360 с.

5. Мосьяков В.Е. Информатика: Учеб. пособие для студентов вузов / В.Е. Мосьяков, Е.Б. Попкова. – М.: Федер. агентство по образованию; Рос. гос. торгово-экон. ун-т; Ин-т безопасности экон. развития, 2006.

6. Мосьяков В.Е. зарождение и основные компоненты коллективного разума / И.В. Мосьяков, В.Е. Мосьяков //Альманах современной науки и образования. – 2014. – №7 (85). – С. 99–102.