УДК 372.862

КЛЮЧЕВЫЕ АСПЕКТЫ РЕАЛИЗАЦИИ АКТИВНЫХ МЕТОДОВ ОБУЧЕНИЯ В ДИСЦИПЛИНЕ «ОСНОВЫ ПРОГРАММИРОВАНИЯ» В РАМКАХ ИДЕЙ МЕЖДУНАРОДНОГО ПРОЕКТА ПО РЕФОРМИРОВАНИЮ ИНЖЕНЕРНОГО ОБРАЗОВАНИЯ

Виденин С.А., Раскина А.В., Виденина М.С., Дьячук П.П. (мл.), Корнеева А.А.
 ФГАОУ ВО «Сибирский федеральный университет», Красноярск, e-mail: svidenin@sfu-kras.ru

В статье представлена концепция проектирования дисциплины «Основы программирования» в контексте инициативы CDIO и взаимосвязь ее ключевых разделов со стандартами CDIO. Описаны методические особенности применяемых педагогических технологий в реализации модулей, составляющих курс. Приводится перечень компетенций, подлежащих формированию в образовательном процессе в контексте стандартов CDIO. Особое внимание уделяется активным методам обучения и их использованию на лекционных и практических занятиях. Показана важность практической направленности всех применяемых методик обучения, которые берут свое начало из реальных практик ИТ-индустрии. Говорится о важности интеграции дисциплины со смежными и последующими дисциплинами в образовательной программе направления подготовки «Информационные системы и технологии». Приводятся примеры того, как лучшие ИТ-практики (code review, pair programming, meet up, refactoring) вовлекают студентов непосредственно в решение производственных проектных задач: предпроектный анализ требований, составление технического задания, оценку предложенных решений, проектирование и реализация проекта. Описывается опыт проведения аудиторных занятий, на которых в процессе выполнения практических работ студенты видят себя в ролях, напрямую связанных с их будущей профессией, что дает им опыт ведения проектно-внедренческой деятельности.

Ключевые слова: CDIO, проектирование образовательного процесса, активные методы обучения, case-study

KEY ASPECTS OF IMPLEMENTING ACTIVE LEARNING METHODS IN THE «FUNDAMENTALS OF PROGRAMMING» DISCIPLINE IN THE FRAMEWORK OF THE INTERNATIONAL PROJECT FOR REFORMING ENGINEERING EDUCATION

Videnin S.A., Raskina A.V., Videnina M.S., Dyachuk P.P. (jr), Korneeva A.A.
 Siberian Federal University, Krasnoyarsk, e-mail: svidenin@sfu-kras.ru

The article presents the concept of «Fundamentals of programming» discipline design in the context of the CDIO initiative and the relations between the key sections of the discipline with the CDIO standards. There are methodological features of the applied pedagogical technologies used in the course. The list of competencies to be formed in the educational process in the context of CDIO standards is given. Special attention is paid to active learning methods and their usage in lectures and practical classes. The importance of practical orientation of all applied teaching methods, which originate from the real practices of the IT industry, is shown. The importance of integrating the discipline with related and subsequent disciplines in the «Information systems and technologies» bachelor's degree program is discussed. There are some examples how the best IT practices (code review, pair programming, meet up, refactoring) involve students directly in solving production project tasks: pre-project analysis of requirements, preparation of technical requirements, evaluation of proposed solutions, design and implementation of the project. The article describes the experience of conducting classes in which students take the roles directly related to their future profession, which gives them opportunity conducting design-implement experience.

Keywords: CDIO, learning design, active learning, case-study

Задачи усиления практической направленности обучения являются первостепенными в любом образовательном направлении. Не исключением является и направление подготовки бакалавров «Информационные системы и технологии». Для решения озвученных выше задач, в связи с необходимостью повышения качества инженерного образования, в 2000 г. был создан проект, основной целью которого является реформирование базового высшего образования в области техники и технологий. Инициатива CDIO получила широкое распространение в образовательных практиках [1–3]. Проектирование образовательного процесса в контексте CDIO предполагает

усиление практической направленности обучения, а также введение системы проблемного и проектного обучения [4].

Инициатива CDIO предполагает формировать у студентов способность [5]:

1. Овладеть глубокими знаниями технических основ.
2. Руководить процессом создания и эксплуатации новых продуктов и систем.
3. Понимать важность и последствия воздействия научного и технологического прогресса на общество.

Проектирование образовательного процесса в контексте инициативы CDIO реализуется в соответствии с моделью *Conceive – Design – Implement – Operate*. Это означает,

что после освоения программ CDIO выпускники должны быть способны создавать новые продукты, осуществлять конструкторские работы и впоследствии внедрять полученные результаты в производство. В рамках инициативы в 2004 г. были приняты 12 стандартов, которые позволяют определить отличительные критерии программ CDIO, а также и их выпускников. В 12 стандартах CDIO раскрывается философия программы (Стандарт 1), разработка учебных планов (Стандарты 2, 3 и 4), реализация проектной деятельности и требования к рабочему пространству (Стандарты 5 и 6), методы преподавания и обучения (Стандарты 7 и 8), повышение квалификации преподавателей (Стандарты 9 и 10), а также оценка результатов обучения и программы в целом (Стандарты 11 и 12) [5].

Материалы и методы исследования

Приведем общие сведения рассматриваемой дисциплины. Дисциплина «Основы программирования» реализуется для направления подготовки 09.03.02 Информационные системы и технологии, объем дисциплины 324 ч, из них практические занятия составляют 108 акад. ч.

Определение результатов обучения способствует получению необходимой основы для успешного применения полученных знаний в будущем. Планируется, что в ходе освоения материала дисциплины «Основы программирования» у студентов будут сформированы компетенции в соответствии с ФГОС ВПО третьего поколения (плюс)), а также CDIO Syllabus, представленные в табл. 1.

Таблица 1

Планируемые результаты обучения

Планируемые результаты обучения CDIO Syllabus	Универсальные	Общепрофессиональные
4.3.1. Изучение потребностей и постановка целей	УК-2 Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий
3.1.1. Формирование эффективной команды		
4.4.1. Процесс проектирования (продукции / системы)		
4.5.3. Процесс производства программной продукции		

В данной статье рассмотрен вопрос проектирования образовательного процесса по дисциплине «Основы программирования» направления подготовки 09.03.02 «Информационные системы и технологии» в соответствии с ФГОС ВПО (плюс) в контексте следующих стандартов CDIO:

Стандарт 1. CDIO как контекст инженерного образования;

Стандарт 5. Опыт ведения проектно-внедренческой деятельности;

Стандарт 7. Интегрированный учебный план;

Стандарт 8. Активные методы обучения.

Цель данного исследования – выявить особенности реализации ряда стандартов CDIO при осуществлении образовательного процесса по дисциплине «Основы программирования» направления подготовки 09.03.02 «Информационные системы и технологии». Определить эффективные активные методы обучения, выявить интегрируемые дисциплины, а также разработать содержание дисциплины с учетом получения студентами опыта ведения проектно-внедренческой деятельности.

Применение компетентностного подхода является одним из важных концептуальных положений модернизации образования. В соответствии с этим положением образование должно формироваться «от результата», где основная ориентация направлена на достижение конкретных целей-векторов образования: обучаемость, самоопределение, самоактуализация, социализация и развитие индивидуальности [6].

Результаты исследования и их обсуждение

Ориентация образования на компетентностный подход означает построение процесса обучения таким образом, чтобы студент мог проявить себя как субъект обучения. С этой целью должны активно использоваться современные образовательные технологии, одной из которых является применение активного обучения [7, 8]. М. Смолкин дает следующее определение [9]: активные методы обучения – это способы активизации учебно-познавательной деятельности учащихся, которые побуждают их к активной мыслительной и практической деятельности

в процессе овладения материалом, когда активен не только учитель, но активны и ученики. В процессе реализации дисциплины «Основы программирования» активное обучение имеет практическую направленность. Формы проведения лекционных и практических занятий при освоении материала учебных модулей дисциплины «Основы программирования» в соответствии со стандартом 8 идеологии CDIO могут включать в себя:

- скринкастинг (экранное видео), на котором лектор демонстрирует пошаговое написание алгоритма, давая в ходе этого теоретический учебный материал;

- просмотр кода (ревью, Code review) – это взаимный анализ кода с целью выявить ошибки, недочеты, расхождения в стиле написания кода, в соответствии написанного кода и поставленной задачи;

- командное выполнение проекта согласно назначенным ролям: архитектор информационной системы, подрядчик (компания, выполняющая часть работ), программист, дизайнер, специалист по контролю качества программного обеспечения;

- парное программирование – техника программирования, при которой исходный код создаётся парами людей, программирующих одну задачу, сидя за одним рабочим местом;

- митап (встреча «на ногах») с целью обсуждения принятого решения в рамках выполнения учебного проекта;

- закрепление полученного материала путем проведения рефакторинга кода – изменение исходного кода программы без изменения его внешнего поведения.

Именно активные методы обучения в дисциплине «Основы программирования» вовлекают студентов непосредственно в решение производственных проектных задач: предпроектный анализ требований, составление технического задания, оценку предложенных решений, проектирование и реализация проекта. Можно говорить, что активные методы обучения являются практико-ориентированными, особенно в случаях, когда в процессе выполнения

практических работ студенты видят себя в ролях, которые напрямую связаны с их будущей профессией: проект-менеджер, руководитель команды разработчиков, ведущий программист, программист, тестировщик. Именно с этой целью, все аудиторные занятия дисциплины реализуются в соответствии со стандартом 5, который предполагает получение студентами опыта ведения проектно-внедренческой деятельности.

Итогом изучения каждого раздела дисциплины «Основы программирования» является проектирование и разработка информационной системы, реализация жизненного цикла которой осуществляется в рамках модели «планирование – проектирование – производство – применение», что соответствует 1 стандарту инициативы CDIO. Разделы дисциплины «Основы программирования» и их интеграция с другими предметами представлены в табл. 2.

В контексте реализации стандарта 7 CDIO выполнение практических занятий осуществляется в виде выполнения интегрированных исследовательских задач проектного типа, основными задачами которых является подготовка студентов к выполнению проектной деятельности, а также демонстрация контекста будущей профессиональной деятельности. Из широкого спектра имитационных методов обучения на практических занятиях мы отдаем предпочтение методу case-study. Это весьма эффективный метод активизации учебной и познавательной деятельности студентов. Пример кейса показан на рисунке.

В рамках практических занятий дисциплины «Основы программирования» мы используем следующие активные методы обучения:

- «ситуация-иллюстрация» применяется, когда мы хотим обучить студента оценивать ситуацию в целом. Мы описываем проблемную ситуацию и показываем несколько правильных вариантов ее решения. Студент должен провести анализ каждого из них, выбрать одно из них и донести видение своей точки зрения;

Таблица 2

Разделы дисциплины

Раздел дисциплины	Часы	Интегрируемые дисциплины
Процедурное программирование	8	«Алгоритмы и структуры данных», «Информатика», «Математическая логика»
Объектно-ориентированная парадигма программирования	18	«Архитектура информационных систем», «Введение в инженерную деятельность», «Проектирование информационных систем»
Визуальное программирование	10	«Основы графического дизайна», «Иностранный язык»

ПОЧТА РОССИИ: **ОЖИДАНИЕ****ОЖИДАНИЕ**

После окончания университета вы твердо решили устроиться программистом в компании «Почта Россия». Так как руководство почты обеспокоено производительностью своих сотрудников, то вам поручили разработать информационную систему, позволяющую ежеминутно выяснять, чем же занимаются их сотрудники на рабочем месте.

Требования к программе:

Информационная система должна задавать всем сотрудникам отделения три вопроса:

- «А вас как зовут?»
- «А что вы делаете?»
- «И давно вы тут работаете?»

Смоделируйте процесс опроса сотрудников, реализовав следующую модель классов.

Базовый класс `Employee` содержит: авто-свойство `string Name` (имя сотрудника), авто-свойство `DateTime DateOfEmployment` (дата трудоустройства), метод `Say ()` (отдает `Name` сотрудника в ответ на вопрос: «А вас как зовут», все классы-наследники получают базовую реализацию метода `Say` и не переопределяют его), виртуальный метод `WorkTime ()`, который должен отдавать число на вопрос «И давно вы тут работаете?» (в базовом классе метод должен возвращать 0), виртуальный метод `WhatYouDo ()`, дает ответ на вопрос, а что же делает каждый из сотрудников.

У класса `Employee` три класса-наследника: `Cashier` (кассир), `Operator` (оператор) и `Postman` (почтальон).

На первый вопрос все отвечают одинаково – называют свое имя.

На второй вопрос все отвечают, по-разному (переопределяют метод `WhatYouDo`):

Кассиры: «Пополняю транспортные карты»;

Операторы по работе с клиентами: «Ищу посылку»;

Почтальон: «Разношу почту, не мешайте».

На третий вопрос, классы-наследники переопределяют метод `WorkTime`:

- В классе `Cashier` метод возвращает количество дней.
- В классе `Operator` метод возвращает количество месяцев.
- В классе `Postman` метод возвращает количество лет.

Разработайте также класс `PostOffice`, который содержит коллекцию `List <Employee>` `Employees` (авто-свойство) и метод `Poll()`, который и задает три вопроса всем сотрудникам из коллекции `Employees` (вызывая у каждого объекта этой коллекции методы `Say` и `WorkTime`).

Продемонстрируйте работу вашей системы на 8-ми сотрудниках.

Практическое задание в виде кейса

– «ситуация-проблема» используется, когда условием задачи мы ставим студента в уже сложившуюся проблемную ситуацию, в которой ему необходимо найти решение в рамках поставленных ограничений;

– «ситуация-упреждение» применяется, когда мы хотим обучить студента процессу выбора решений на реальных примерах. Другими словами, показать готовые кейсы нахождения правильного решения. Данный метод применяется для закрепления ранее полученных знаний.

Основным способом донесения теоретических знаний по-прежнему остается лекция. Можно выделить несколько видов лекций, используемых нами:

– «проблемная лекция» – в ней моделируются противоречия реальной жизни. Позволяет стимулировать познавательный интерес студентов. Используется, когда необходимо показать междисциплинарные связи изучаемого материала;

– «лекция-диалог». Строится на серии взаимосвязанных вопросов к студенту, которые опираются на ранее полученные знания, преподаватель помогает синтезировать новые;

– «лекция вдвоем» читается в паре с приглашенным ИТ-специалистом. Позволяет наглядно и моментально показать прикладное значение теоретического материала;

– «лекция с ошибками» – т.е. лекция, которая призывает студентов к постоянному контролю предлагаемой информации путем поиска ошибок;

– «лекция-консультация» – это лекция, основной частью которой является ответ лектора на вопросы студентов. В конце занятия возможна дискуссия, которая подводит итоги.

Применение активных методов обучения позволяет увеличить результативность учебной деятельности посредством повышения эмоциональной включенности обучающихся и их творческой активности на занятиях. Обязательное непосредственное общение обучающихся между собой и преподавателем формирует коллективную сплоченность и интенсифицирует процесс обучения. Таким образом, можно утверждать, что активные методы обучения подходят для преподавания ИТ-дисциплин, так как будущие ИТ-специалисты должны уметь работать в команде под руководством ведущего разработчика, аналогично ролям студентов и преподавателей в процессе обучения.

Заключение

Реализация дисциплины «Основы программирования» в соответствии с идеологией CDIO позволяет сформировать у студентов первого курса ключевые компетенции в рамках подготовки по направлению «Информационные системы и технологии». Освоение данной дисциплины должно носить практико-ориентированный характер, что делает актуальным применение идеологии CDIO, которая позволяет не только повысить практическую направленность обучения, но и сформировать необходимые компетенции будущих выпускников.

В ходе реализации в дисциплине «Основы программирования» концепций, заложенных в CDIO, преподавателями курса были подобраны соответствующие стандарты, которым должна соответствовать дисциплина, планируемые результаты обучения, а также выбраны активные методы обучения. Следует отметить, что уже сегодня преподавание данной дисциплины ориентируется на рассматриваемую идеологию, что демонстрирует повышение обучаемости студентов, их социализацию и развитие индивидуальных качеств в профессиональной сфере.

Для каждого раздела дисциплины преподавателями курса сформулирована своя технология обучения, направленная на проектную деятельность и работу в команде. Большое значение при этом имеют применяемые в ходе проведения занятий активные методы обучения, подробно описанные в настоящей работе. Немаловажную роль здесь играет интеграция с другими дисциплинами, в частности с дисциплиной «Введение в инженерную деятельность», изучение которой проходит параллельно с курсом «Основы программирования». Данная интеграция позволяет сформировать у студентов навыки создания эффективной команды. Интеграция с дисциплиной «Иностранный язык» позволяет расширить кругозор студентов в профессиональной сфере производства программных продуктов.

Таким образом, можно сказать, что применение идеологии CDIO к дисциплине «Основы программирования» является обоснованным и позволит повысить эффективность обучения по направлению «Информационные системы и технологии».

Список литературы

1. Кроули Э.Ф., Малмквист Й., Остлунд С., Бродер Д.Р., Эдстрем К. Переосмысление инженерного образования. Подход CDIO. М.: ВШЭ, 2015. 504 с.
2. Гафурова Н.В., Осипова С.И. Металлургическое образование на основе идеологии CDIO // Высшее образование в России. 2013. № 12. С. 137–139.
3. Бутакова С.М., Братухина Н.А., Арасланова М.Н., Кубикова Н.Б. Проектирование образовательного процесса по математике в контексте стандартов CDIO // Фундаментальные исследования. 2014. № 6. С. 1497–1503.
4. Всемирная инициатива CDIO. Планируемые результаты обучения (CDIO Syllabus): информационно-методическое издание / Пер. с англ. и ред. А.И. Чучалина, Т.С. Петровской, Е.С. Кулокиной. Томский политехнический университет. Томск: Изд-во Томского политехнического университета, 2011. 22 с.
5. Всемирная инициатива CDIO. Стандарты: информационно-методическое издание / Пер. с англ. и ред. А.И. Чучалина, Т.С. Петровской, Е.С. Кулокиной. Томский политехнический университет. Томск: Изд-во Томского политехнического университета. 2011. 17 с.
6. Зеер Э.Ф. Компетентный подход к образованию // Образование и наука. 2005. № 3. С. 27–40.
7. Вербицкий А.А. Активное обучение в высшей школе: контекстный подход. М.: Высшая школа, 1991. 207 с.
8. Активные и интерактивные образовательные технологии (формы проведения занятий) в высшей школе: учебное пособие / Сост. Т.Г. Мухина. Н. Новгород: ННГАСУ, 2013. 97 с.
9. Антропова М.А. Активные методы обучения при проведении занятий в высших учебных заведениях // Проблемы обеспечения безопасности при ликвидации последствий чрезвычайных ситуаций. 2016. № 1–2 (5). С. 207–208.