

РАЗДЕЛ «ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ»

УДК 004:378

ЖУЛЬКОВСКАЯ И.И., к.т.н., доцент
ЖУЛЬКОВСКИЙ О.А., к.т.н., доцент

Днепродзержинский государственный технический университет

К ВЫБОРУ СТРАТЕГИИ ПРЕПОДАВАНИЯ ИНФОРМАТИКИ ДЛЯ ИНЖЕНЕРНЫХ СПЕЦИАЛЬНОСТЕЙ УНИВЕРСИТЕТОВ

Введение. Современный этап развития человечества характеризуется революционным переходом от индустриального к постиндустриальному (информационному) общественному устройству, смещением его материальных и энергетических приоритетов к информационным ресурсам. Стремительно развивающиеся процессы информатизации привели к глобализации экономического и политического пространства, кардинальным изменениям в экономике и политике, науке и технике, образовании и культуре, т.е. практически во всех сферах человеческой деятельности. Одним из этапов глобализации высшего образования является подписание многими странами Европы (в том числе и Украиной) так называемой Болонской декларации, направленной на создание единой системы европейского высшего образования.

Современное информационное общество в эпоху глобализации неуклонно изменяет отношение к образованию в целом, качественный уровень которого становится одним из важнейших показателей развития любого государства.

Институт ЮНЕСКО в рамках развития информационных технологий в образовании видит своей задачей укрепление национального потенциала в плане применения информационно-коммуникационных технологий (ИКТ) во всех сферах развития государств-членов в целях построения инклюзивного информационно развитого общества и создания предпосылок для устойчивого развития этих стран [1].

В этой связи ИКТ-компетенция студентов становится основой идеологии современного стиля обучения. Исследования, проведенные в Европе [2], показывают, что для 80% студентов навыки в ИКТ являются очень важными для будущего трудоустройства, карьерного роста и развития. Европейские работодатели, в свою очередь, ищут среди выпускников вузов не только профессионально компетентных в своей области специалистов, но и работников с должными базовыми навыками в применении ИКТ на практике. Другими словами, современный этап развития общества и технологий требует высококвалифицированных специалистов, имеющих наряду с должной специальной инженерной подготовкой глубокие знания в области информационных и коммуникационных технологий.

Постановка задачи. Абсолютно очевидно, что массовое внедрение динамически развивающихся информационных технологий в процесс подготовки и формирования современного специалиста невозможно без повышения эффективности и обеспечения динамики развития методик обучения информатике на всех этапах образования и, особенно, в высшей школе. Поэтому весьма актуальной становится задача анализа существующих проблем, методик и концепций преподавания информатики для инженерных специальностей высших учебных заведений Европы и США. Кроме того, следует уделить особое внимание выработке стратегии выбора языка программирования в учебных планах преподавания университетского курса информатики для указанных специальностей.

Результаты работы. В научно-педагогической среде постоянно ведется полемика по целому ряду аспектов проблематики внедрения ИКТ в высших учебных заве-

дениях. Это проблемы существующей материально-технической базы университетов (или специализирующихся на преподавании информатики кафедр), недоступность программного обеспечения, используемого в учебном процессе, несвоевременное повышение квалификации преподавательского состава кафедр информатики, недостаточное и стремительно теряющее актуальность учебно-методическое и информационное обеспечение учебного процесса, практически полное отсутствие государственной поддержки и т.п. [1-6].

Как зарубежные, так и отечественные педагоги отмечают одну из основных трудностей в преподавании основ информатики – отсутствие единого мнения о принципах отбора содержания.

Самым значительным на сегодня проектом в области стандартизации обучения информатике является создание и обновление документа Computing Curricula (рекомендации по преподаванию информатики в университетах) [7]. Этот документ является результатом совместной работы Компьютерного общества Института инженеров по электротехнике и электронике (IEEE-CS) и Ассоциации по вычислительной технике (ACM). Кроме того, во время разработки последнего отчета Computing Curricula 2001 (CC2001) был проведен целый ряд специальных семинаров на различных конференциях и совещаниях, включая Симпозиум специальной группы по образованию в области информатики (SIGCSE), Конференцию «Передний край в образовании» (FIE), Всемирный конгресс по вопросам в области информатики и образования (WCCE), а также другие специализированные совещания меньших масштабов в Европе, Азии и различных частях США.

В 2008 году появилось дополнение к CC2001 – Computer Science Curriculum 2008 (CS2008) [8]. Этот документ состоит из рекомендаций CC2001 с уточнениями и замечаниями. В настоящий момент обсуждается и готовится к выпуску очередной итоговый отчет Computer Science Curricula 2013, который также рассматривается как прогноз развития компьютерных наук на ближайшие 5-10 лет. С проектом можно ознакомиться на веб-сайте CS2013.

В CC2001 и его дополнениях описаны основные подходы к преподаванию информатики в высшей школе «с ориентацией на программирование» и три альтернативных модели организации обучения ИКТ.

Члены специальной комиссии CC2001 полагают, что модель «с ориентацией на программирование» (programming-first introduction) доказала свою исключительную жизнеспособность и будет оставаться доминирующей в обозримом будущем, несмотря на существующие недостатки. Данная концепция предполагает три реализации обучения информатике с ориентацией на программирование. Это традиционный подход с ориентацией на императивное программирование (imperative-first approach), подход с ориентацией на объектно-ориентированное программирование (objects-first approach), основанный на раннем изложении концепций объектов и объектно-ориентированного проектирования, а также подход с ориентацией на функциональное программирование (functional-first approach), представляющий алгоритмические концепции на примере языков с семантически простым функциональным синтаксисом.

В свою очередь, альтернативные модели обучения представляют:

- подход с максимальным охватом материала (breadth-first approach), который начинается с общего обзора дисциплины;
- стратегию с ориентацией на алгоритмы (algorithm-first strategy), которая фокусируется на алгоритмах, а не на синтаксисе;
- модель с ориентацией на аппаратную часть (hardware-first approach), которая начинается с электронных схем и далее постепенно продвигается вверх по всё усложняющимся уровням иерархии абстрактных машин.

Модель с максимальным охватом материала предоставляет возможность студентам перед началом курса оценить разнообразие тем учебной программы дисциплины и только после этого приступить к традиционному циклу изучения программирования. Или же наоборот – можно предложить обзор дисциплины уже после завершения вводного цикла по программированию. Такая модель предполагает трехсеместровый вводный курс информатики.

Подход «с ориентацией на алгоритмы» реализуется следующими курсами:

1. Введение в алгоритмы и приложения.
2. Методология программирования.

Первый курс начинается с обсуждения алгоритмов и их приложений, а затем переходит к изложению основ объектно-ориентированного программирования и продолжает его до конца курса. Второй курс предполагает более тщательное изучение объектно-ориентированного программирования.

Курсами, составляющими модель «с ориентацией на аппаратную часть», являются:

1. Введение в компьютер.
2. Технологии объектно-ориентированного программирования.

Первый курс цикла в мельчайших подробностях излагает устройство компьютера, чтобы во втором использовать этот базис для выработки у студентов навыков программирования и дать солидное введение в объектно-ориентированные технологии.

Таким образом, в результате даже такого глобального исследования IEEE-CS и ACM не была найдена единая, наиболее успешная стратегия преподавания информатики в высшей школе. В итоге, сделан вывод о том, что никакой из предложенных подходов не будет успешным и универсальным для всех университетов и специальностей. Причина этого заключается в том, что их вводные программы сильно различаются по своим целям, структуре, ресурсам и предполагаемой аудитории.

Так как каждый подход имеет свои сильные и слабые стороны, то комиссия считает, что нужен ряд альтернативных стратегий, проверенных на практике: «Мы должны поощрять институты, факультеты и отдельных преподавателей продолжать эксперименты в этой области. В такой динамично меняющейся науке, как информатика, для повторения успеха необходимы постоянные педагогические нововведения» [7].

Все описанные методики либо полностью посвящены изучению программирования, либо большая часть курса посвящена этой теме.

К сожалению, отчет CC2001 не содержит каких-либо конкретных рекомендаций по выбору языка программирования, изучаемого в курсе информатики. При этом делается акцент, что этот вопрос продолжает оставаться предметом дальнейших дискуссий и споров.

Как известно, постоянное совершенствование информационных технологий привело к появлению огромного числа языков программирования, эволюция которых продолжается и сейчас [9].

В Computer Science Curriculum 2008 указывается о том, что вопрос о языках и парадигмах программирования вызвал серьезные дебаты в момент окончательных общественных консультаций [8]. В итоге, после конференции SIGPLAN (Special Interest Group on Programming Languages – специальная группа в составе ACM, объединяющая специалистов по языкам программирования) в Гарварде было принято решение пересмотреть подходы к преподаванию программирования в рамках курса информатики.

В рекомендацию внесено требование, что все студенты должны научиться программировать в более чем одной парадигме. Выбор второй парадигмы (первая, как указывалось в CC2001, – объектно-ориентированная) будет зависеть от конкретных образовательных целей каждого учебного заведения. Уточняется, что для университетов,

которые стремятся подготовить студентов с позиции академических, научных исследований и передовых разработок, было бы целесообразно ввести в учебные планы по информатике функциональную парадигму программирования.

Комитет CS2008, учитывая интерес, вызываемый в педагогической среде вопросами обучения программированию, запланировал именно эти вопросы сделать центральными в своих последующих рекомендациях.

Необходимо отметить, что одним из критериев выбора языка программирования может служить индекс TIOBE (TIOBE programming community index) [10]. Голландская компания TIOBE Software BV, как известно, является автором регулярно рассчитываемого рейтинга популярности языков программирования. Кстати, большинство сотрудников компании – известные специалисты в области разработки компиляторов. http://ru.wikipedia.org/wiki/TIOBE_Index_-_cite_note-3

Рейтинг TIOBE публикуется, начиная с июня 2001 года; измерения проводятся не менее десяти раз в месяц. TIOBE ориентируется только на полные по Тьюрингу языки программирования. Исходя из этого, популярность XML, HTML или базового SQL, к примеру, не исследуется. С другой стороны, такие расширения SQL, как PL/SQL и Transact-SQL, считаются языками программирования. Средства разработки и технологии, такие как ASP, ASP.NET, Rails, AJAX и т.п. также не считаются языками программирования. Наконец, из рассмотрения исключены языки типа ассемблера, хотя они и полны по Тьюрингу. Диалекты одного и того же языка программирования объединены под общим названием. При этом в рейтинг включается не суммарный вес диалектов, а рейтинг диалекта, показавшего наилучший результат. Кроме Тьюринг-полноты, авторы индекса требуют от исследуемого языка наличие соответствующей статьи в Wikipedia, где было бы четко указано, что язык является языком программирования. По последнему критерию в индекс не были включены: Ruby on Rails, Excel, Android, Boost, Cocoa, ASP, AJAX и некоторые другие.

Для формирования индекса используется поиск на нескольких, наиболее посещаемых (по данным Alexa.com), порталах: Google, Blogger, Wikipedia, YouTube, Baidu, Yahoo!, Bing, Amazon. Такой же поиск проводится на Google по группам новостей и блогам за последние 12 месяцев.

Полученные результаты нормализуются по некоторой формуле, которая и определяет место языка в рейтинге.

Если считать первые 50 языков программирования за 100%, то рейтинг TIOBE в числовом выражении показывает долю, занимаемую каждым языком.

Кроме этого, авторы рейтинга делят языки на две категории: А и В (хотя существуют А- и А--). Они рекомендуют при выборе языка для разработки или изучения (с учётом карьерных перспектив) ориентироваться на языки категории А, поскольку для языков категории В гораздо меньше квалифицированных специалистов, значительно меньше выбор качественных инструментов для разработки, меньше доступных библиотек и т.п.

Авторы индекса считают, что он может быть полезен при принятии стратегических решений. При этом, естественно, делается оговорка о необходимости учитывать сферу применения.

На индекс TIOBE ориентируется большинство авторов в научных публикациях при сравнении популярности языков программирования, несмотря на его косвенные, проприетарные методики и платность набора исходных данных [11]. Хотя такой метод вряд ли можно назвать научным, тем не менее, на сегодняшний день он является наиболее популярным.

Расчет индекса происходит ежемесячно. Текущая информация предоставляется на сайте бесплатно.

Для примера в табл.1 представлено пятнадцать первых позиций рейтинга ТЮВЕ за февраль 2013 года в сравнении с данными за февраль 2012 года.

Таблица 1 – Индекс ТЮВЕ за февраль 2013 года

Позиция февр. 2013	Позиция февр. 2012	Изменение в позиции	Язык программирования	Индекс февр. 2013	Изменение февр. 2012	Категория
1	1	=	Java	18.387%	+1.34%	A
2	2	=	C	17.080%	+0.56%	A
3	5	↑↑	Objective-C	9.803%	+2.74%	A
4	4	=	C++	8.758%	+0.91%	A
5	3	↓↓	C#	6.680%	-1.97%	A
6	6	=	PHP	5.074%	-0.57%	A
7	8	↑	Python	4.949%	+1.80%	A
8	7	↓	(Visual) Basic	4.648%	+0.33%	A
9	9	=	Perl	2.252%	-0.68%	A
10	12	↑↑	Ruby	1.752%	+0.19%	A
11	10	↓	JavaScript	1.423%	-1.04%	A
12	16	↑↑↑↑	Visual Basic .NET	1.007%	+0.21%	A
13	13	=	Lisp	0.943%	+0.04%	A
14	15	↑	Pascal	0.932%	+0.12%	A
15	11	↓↓↓↓	Delphi/Object Pascal	0.886%	-1.08%	A

Кроме того, начиная с 2003 года авторами ТЮВЕ выбирается язык года (Programming Language of the Year):

- 2012 Objective-C;
- 2011 Objective-C;
- 2010 Python;
- 2009 Go;
- 2008 C;
- 2007 Python;
- 2006 Ruby;
- 2005 Java;
- 2004 PHP;
- 2003 C++.

Представленные таблицы рейтингов позволяют сориентироваться преподавательскому составу специализированных на информатике кафедр, а также другим заинтересованным специалистам в выборе соответствующего языка программирования при насыщении и актуализации учебных планов с целью повышения эффективности университетского курса информатики.

Выводы. Как показывает проведенный анализ, в настоящий момент существует несколько стратегий обучения информатике, каждая из которых имеет свои как сильные, так и слабые стороны. Поэтому в педагогической среде продолжается формирование альтернативных стратегий обучения. Предложенный критерий выбора языка программирования на основе ТЮВЕ programming community index позволяет сделать более эффективным, современным и прогрессивным университетский курс информатики.

ЛИТЕРАТУРА

1. Применение ИКТ в высшем образовании стран СНГ и Балтии: текущее состояние, проблемы и перспективы развития: аналитический обзор / СПб.: ГУАП, 2009.–

- 160с.– Режим доступа: <http://iite.unesco.org/pics/publications/ru/files/3214561.pdf>.
2. A Comparison of ICT Skills and Students Across Europe / J.Haywood, D. Haywood, H. Macleo [et al.] // Journal of eLiteracy. – 2004. – Vol.1. – p.69-81.– Режим доступа: http://www.jelit.org/33/01/JeLit_7.pdf.
 3. Козлакова Г.О. Науково-методична підтримка розвитку вищої природно-математичної освіти у технічних університетах / Г.О.Козлакова, Т.В.Ковалюк // Педагогічні науки: теорія, історія, інноваційні технології. – 2010. – №2(4). – С.309-317. – Режим доступа: http://www.nbu.gov.ua/portal/soc_gum/pednauk/2010_2/309.pdf.
 4. Козлакова Г.О. Впровадження інформаційно-комунікаційних технологій у навчальний процес вищої школи: аналіз стану, проблеми, перспективи / Г.О.Козлакова, Т.В.Ковалюк // Вісник Національного технічного університету України «Київський політехнічний університет». Філософія. Психологія. Педагогіка. – 2009.– № 3(27), Ч.2. – С.102-107. – Режим доступа: http://www.nbu.gov.ua/portal/soc_gum/VKPI_fpp/2009-3-2/07_Kozlakova.pdf.
 5. Tedre M. Know Your Discipline: Teaching the Philosophy of Computer / M.Tedre. // Journal of Information Technology Education. – 2007. – Vol.6. – P.105-122. – Режим доступа: <http://www.jite.org/documents/Vol6/JITEv6p105-122Tedre266.pdf>.
 6. Matthíasdóttir Á. How to teach programming languages to novice students? Lecturing or not? / Á.Matthíasdóttir // CompSysTech'06: Proceedings of the International Conference on Computer Systems and Technologies, 15-16 June 2006. – University of Veliko Tarnovo, 2006. – p.13.1-13.6. – Режим доступа: <http://ecet.ecs.ru.acad.bg/cst06/docs/cp/siv/iv.13.pdf>
 7. Рекомендации по преподаванию информатики в университетах; пер. с англ.– СПб., 2002. – 372с. – Режим доступа: <http://se.math.spbu.ru/SE/cc2001r.pdf>.
 8. Обновление рекомендаций по преподаванию информатики 2008 (CS2008 Curriculum Update). – 2008. – 108р. – Режим доступа: <http://www.acm.org/education/curricula/ComputerScience2008.pdf>.
 9. Грэм П. Языки программирования через сто лет / КомпьютерраOnline. – Режим доступа: <http://www.computerra.ru/hitech/35042/>.
 10. TIОBE Programming Community Index for February 2013. – Режим доступа: <http://www.tiobe.com>.
 11. Жульковський О.О. До принципів формування навчальних програм з дисциплін, що вивчають основи програмування / О.О.Жульковський // Проблеми математичного моделювання: міждерж. науково-метод. конф., 13-15 червня 2012р.: тези доп. – Дніпродзержинськ, 2012. – С.162-163.

Поступила в редколлегию 28.02.2013.

УДК 061.66:004.65

ДРАНИШНИКОВ Л. В., д.т.н., профессор
ПАМФИЛОВ О.И., магистр

Днепродзержинский государственный технический университет

РАЗРАБОТКА WEB–ОРИЕНТИРОВАННОЙ СИСТЕМЫ ЭЛЕКТРОННОГО ДОКУМЕНТООБОРОТА

Введение. Количество и объёмы используемых в современном мире документов растут. Причём соотношение электронных и бумажных документов со временем меняется в пользу первых. На данный момент, согласно статистическим данным, объём корпоративной электронной текстовой информации каждые три года удваивается.