

О СОСТАВЛЕНИИ И РЕАЛИЗАЦИИ УЧЕБНЫХ ПЛАНОВ ПО КИБЕРНЕТИЧЕСКИМ ДИСЦИПЛИНАМ В ВЫСШЕЙ ШКОЛЕ

А.П. Полищук, С.А. Семериков
г. Кривой Рог, Криворожский государственный педагогический
университет
ss@kpi.dp.ua

Любой технологический процесс выполняется “в прямом направлении” – от сырья к конечному продукту. Однако любая технология разрабатывается (проектируется) в обратном порядке – от требований к готовому продукту к характеристикам промежуточных до входных сырьевых включительно и к средствам обеспечения этих характеристик на всех стадиях переработки. Учебный процесс не исключение – при составлении учебных планов вначале должны быть определены требования к знаниям и навыкам выпускников и сформированы учебные планы заключительных предметных курсов, определен уровень подготовки слушателей, необходимый для восприятия заключительного и, очевидно, наиболее сложного материала – в том числе по поддерживающим дисциплинам, не относящимся непосредственно к выпускающим кафедрам. Затем должны формироваться планы более ранних этапов обучения и т.д. вплоть до требований к уровню подготовки абитуриентов по профильным дисциплинам. Сказанное достаточно тривиально, но учебные планы по кибернетическим дисциплинам для связанных с ними специальностей (или специализаций) во многих вузах характеризуются эклектизмом как по номенклатуре, так и последовательности преподавания.

В последнее время предпринимаются попытки устранить непоследовательности в учебных планах различного рода организационными мерами. Так, преподаватели всех вузов сейчас обязаны предоставить в учебную часть рабочие (плюс рабочие учебные) программы читаемых курсов (что в принципе не вызывает возражений); первые пункты этих программ должны содержать цель и задачи курса, а также перечень знаний и умений студента после его освоения. Эти пункты явно не по адресу

– квалифицированный специалист может прекрасно прочитать курс, не зная цели, с которой он был включен в учебный план. Например, нелегко обосновать наличие в учебных планах для специальности “Информатика и экономика” (готовит специалистов по математическим методам решения экономических задач) предмета “Микропроцессорные устройства и системы”, читаемого вдобавок без предварительной базовой подготовки студентов по промышленной электронике. Даже очень хорошо поставленный курс, уложенный на неподготовленную почву, не даст ожидаемого результата.

Но алогичность просматривается уже в последовательности действий: сначала верстается учебный план как перечень названий без хотя бы краткой расшифровки предполагаемого содержания, а затем по каждому его пункту составляется рабочая программа. Обе работы выполняют специалисты различной квалификации, учебный план – работником административного профиля, программа – исполнителем и они зачастую по-разному понимают скрытое под нерасшифрованным названием содержание. Нормально было бы вначале рассмотреть подробную рабочую программу курса (авторскую преподавателя или взятую в качестве образца уже поставленного ранее в текущем или другом вузе) и по ее содержанию принимать решение о включении или не включении курса в учебный план.

В требованиях к содержанию учебных программ нет перечня знаний и умений, необходимых для восприятия студентами содержания курса – это означает отсутствие требований к поддерживающим дисциплинам, которые должны быть прочитаны на предшествующих этапах обучения и порождает одну из причин упомянутого эклектизма учебных планов.

Конечно, ключевая проблема организации и обеспечения качества обучения – кадровая. Острая нехватка квалифицированных в области кибернетических наук преподавательских кадров была всегда, конъюнктура лишь усилила спрос, не обеспечив его удовлетворение. Квалифицированный программист или электронщик из промышленности, академической или прикладной науки (особенно без ученой степени и звания) не кандидат в преподаватели хотя бы по чисто материальным соображениям. В результате кибернетические предметы (как, к слову, и экономиче-

ские) зачастую преподают люди, не имеющие ни малейшего практического опыта работы в этой области, с дипломами самых разных специальностей. А специфика кибернетических дисциплин требует наличия опыта: непрограммирующий преподаватель программирования – нонсенс, быстро найти ошибку в программе студента (в компьютерном классе их от 12 до 17), помочь в решении даже чисто технической проблемы он не сможет. Вузам остается оснащаться кадрами из внутренних ресурсов – использовать специалистов собственного приготовления и здесь круг замыкается.

В наиболее трудном положении оказываются периферийные вузы “непрестижного” профиля, в частности “трудные” физико-математические факультеты педагогических университетов, потребляющие абитуриентов (выпускников обычных городских и сельских школ) по остаточному принципу – с низким в среднем уровнем школьной подготовки по физике и математике, с неразвитым логическим (алгоритмическим) мышлением, ориентированные на добросовестное заучивание и повторение шаблонных действий при решении задач. Неудовлетворительное качество сырья предопределяет низкое качество выпускаемого продукта, который поступает в школу и готовит на свое место новую волну слабо подготовленных абитуриентов.

Одно из забавных (но не безобидных) следствий кадровой проблемы – терминологические неточности. Введенный в свое время в обиход термин “информатика”, который является всего лишь другим названием кибернетики (науки об оптимальном управлении и математических методах обоснования управленческих решений), породил немало путаницы – еще недавно можно было встретить в университетах параллельно действующие кафедры, например, экономической информатики и экономической кибернетики с полностью совпадающими направлениями работы. К информатике сегодня могут отнести любые компьютерные приложения – от компьютеризированной пишущей машинки (текстовый редактор + принтер) до различных интерактивных оболочек уровня первичной компьютерной грамотности и даже компьютерных игр, – а за подменой понятий следует подмена содержания преподаваемых курсов и соответствующее снижение уровня подготовки специалистов [2].

Возможно, когда-нибудь ключевая проблема преподавательских кадров для кибернетических специальностей будет решена, если преподавать будут параллельно с основной работой активно действующие работники промышленности, академической и прикладной науки и “выработавшие ресурс” специалисты с большим накопленным в практической работе багажом знаний и навыков. Пока же остаются попытки хотя бы маленьких организационных усовершенствований для пусть незначительного повышения результативности преподавательской работы.

Первое предложение в этом направлении нами уже изложено: каждый курс в рабочей программе должен содержать подлежащие выполнению в учебном плане требования к уровню подготовки на предыдущих этапах обучения.

Рассмотрим это предложение на примере условной кибернетической специальности, заключительные курсы которой предполагают изучение методов формирования оптимальных управляющих воздействий и методов идентификации (математического моделирования) управляемых систем (так называемые дуальные кибернетические задачи). Решение обеих задач предполагает (для проверки адекватности модели и оптимальности рассчитанного управления) многократное решение третьей основной кибернетической задачи – задачи анализа, сводящейся, в свою очередь к решению систем дифференциальных (интегро-дифференциальных) уравнений.

Если считать, что десятый заключительный семестр отводится под подготовку выпускной (дипломной) работы, в девятом изучается сопровождаемый лабораторными работами курс теории оптимального управления, в седьмом и восьмом семестрах – методы идентификации дискретных и непрерывных управляемых систем соответственно, то не позднее шестого семестра должны быть завершены следующие поддерживающие курсы:

- методы машинных вычислений, включая методы аппроксимации функций, решения дифференциальных уравнений, линейная алгебра с методами решения проблемы собственных значений (5-й и 6-й семестры);
- теория вероятностей и статистические методы обработки результатов экспериментов (4-й семестр);
- теория планирования экспериментов (5-й семестр);

– общая теория систем автоматического управления и регулирования (6-й семестр);

– математическое программирование, включая теорию массового обслуживания для поддержки методов идентификации дискретных систем (5-й семестр);

В соответствии с этим планом не позднее 4-го семестра должны быть завершены изучением курсы, поддерживающие методы машинных вычислений:

– общий курс математического анализа;

– объектно-ориентированного программирования;

– архитектура компьютеров и операционные системы (2-й семестр);

– фундаментальные алгоритмы и структуры данных (3-й семестр);

– алгоритмические языки и техника программирования (1, 2 семестры);

Мы ограничились планированием общих для всех компьютерных специальностей курсов, не затрагивая многие специальные курсы, зависящие от специализации, например, компьютерные сети, цифровая схемотехника, системное программирование и пр. Не рассматривали также проблемы, связанные с первичной компьютерной грамотностью и офисные пакеты – для них в любом случае не остается ресурса времени, они должны быть решены в довузовской подготовке.

Дефицит времени для определяющих профессиональную подготовку курсов порожден “наступлением” с двух сторон – сокращением общего числа учебных часов в неделю до 30 для создания “щадящего” режима для студентов и перегруженностью учебных планов дисциплинами гуманитарного и общеобразовательного направления, в педагогических университетах дополнительно – психолого-педагогическим комплексом и курсами различных методик преподавания. Все эти дополнения из благих намерений, но все они – за счет снижения уровня профессиональной, предметной подготовки. Трудно оценить эффективность курсов по культуре семейных отношений, валеологии, социологии, делового украинского языка, безопасности жизнедеятельности и многих других, читаемых, естественно, за счет урезания времени на чисто профессиональную подготовку и вполне дос-

тупных для самостоятельного освоения всеми желающими. Все же высокоморальное и гуманное общество может быть построено только на основе материального благополучия, а материальные ценности создаются специалистами, в том числе в области высоких технологий.

Один из источников временного дефицита в подготовке специалистов кибернетического профиля объясняется огромной инерционностью, свойственной системе образования. Наиболее ярко он выражен в преподавании компьютерного программирования. Около 15 лет в школе и вузах начала программирования преподаются на учебном языке Паскаль. Как правило, под это отводится пара семестров, затем приходит очередь объектно-ориентированного и системного программирования и оказывается, что Паскаль для этого слишком неудобен, в профессиональном прикладном и системном программировании не используется вообще, что базовым языком для всех операционных систем является язык Си – более простой и наиболее эффективный [1], лежащий в основе всех современных языков сетевого и интернет-программирования, а в прикладном программировании преобладает объектно-ориентированный язык C++. Не меньше половины отведенного для системного и прикладного объектно-ориентированного программирования времени (тоже не более 2-х семестров в сумме) приходится отводить под изучение С и C++, после чего для изучения собственно предметной области времени не остается – результат легко предсказуем. Мы не против Паскаля как средства любительского программирования “для себя”, но использовать его в профессиональной подготовке специалистов в области информатики (кибернетики) с неизбежным последующим отбрасыванием как ненужного – в нынешних условиях слишком расточительно.

Как хороший филолог должен владеть несколькими языками, так и специалист в области кибернетики должен владеть несколькими языками программирования. Дав в качестве базового простой язык Си, не стоит жестко определять, какие еще языки должен знать будущий специалист – выбор инструмента (языка) определяется особенностями предметной областью, а не нормативными актами (учебными планом). Так, в курсе архитектуры компьютеров и операционных систем целесообразно дать один

из языков ассемблера, курс объектно-ориентированного программирования проиллюстрировать примерами из Смоллток, при изучении систем искусственного интеллекта куда естественней дать Лисп и Пролог, чем использовать язык общего назначения, а в курсе сетевого программирования можно познакомиться с Явой.

Литература:

1. Полищук А.П., Семериков С.А., Грищенко Н.В. О выборе языка программирования для начального обучения. / Збірник наукових праць «Комп'ютерне моделювання та інформаційні технології в природничих науках». – Кривий Ріг, 2000.
2. Семериков С.О. Методичні основи вивчення теми “Операційні системи” у підготовці майбутнього вчителя // Рідна школа. – 2003. – №1. – С. 44-45.