# Поддержка электронного образовательного ресурса по изучению информатики студентами специальности «математика» в ПетрГУ

## Н. А. Будникова, Н. В. Прокопович, Ю. А. Федюк

Петрозаводский государственный университет

Петрозаводск

budnikova@psu.karelia.ru, \_ufdk@mail.ru

Студенты специальности «Математика» ПетрГУ в рамках курса «Информатика» начинают изучать программирование на I курсе. На сегодня в качестве базового языка осуществлен переход на язык программирования Си. Для изучения этого, достаточно сложного для начинающих, языка программирования на кафедре информатики и математического обеспечения разработан учебный ресурс, позволяющий реализовывать различные стратегии обучения, содержащий объемный комплекс упражнений по каждому разделу. Одной из составляющих учебного ресурса является модуль электронного экзамена по информатике. При компьютерном контроле знаний организация контроля направлена, во-первых, на то, чтобы облегчить работу преподавателя, освободив его от рутины проверки знаний и, во-вторых, на повышение объективности их оценки.

Экзамен представляет собой тест с определенным фиксированным числом вопросов. Каждый студент выполняет одинаковую цепочку заданий без возможности возврата к предыдущим вопросам. Время, отводимое для ответов, не ограничено, но фиксируется для каждого задания. В тест включены задания трех типов: на переключатель, на выбор варианта и ввод ответа. Задания подразделяются на понятия и задачи. Все задания имеют уровень сложности и значимости (важности).

Понятия проверяют знания студентов по соответствующим материалам, умение выполнять деятельность по узнаванию, воспроизведение по памяти ранее усвоенной информации, а задачи требуют самостоятельной работы. Понятиям приписан минимальный уровень сложности, так как знание понятий является ключевым аспектом в изучении любого предмета, и выполнение таких заданий не должно вызывать особых трудностей у студентов, прошедших курс обучения. В то же время правильное выполнение подобных заданий является очень важным для определения уровня знаний студента, поэтому понятия имеют максимальный уровень значимости. Сложность задачи зависит от числа существенных операций, содержащихся в ней.

Первоначально при выполнении студентами полного теста ими затрачивалось достаточно много времени, ближе к концу возрастало время обдумывания, нарастала усталость, ослабевало внимание, увеличивалось число ошибок. В связи с этим в следующем экзамене тест был значительно сокращен, при этом все студенты, и сильные, и слабые, выполняли одни и те же задания. Время выполнения сократилось, но многие ранее составленные задания не участвовали в процессе тестирования, тем самым дидактическая цепочка прерывалась (например, из двух-трех задач оставлялась одна, средняя по числу существенных операций). Вследствие этого сокращались возможности более детального и всестороннего изучения модели обучаемого и соответствующей диагностики его состояния. Сильный студент не мог раскрыться в полной мере, а слабый зачастую не дотягивал до минимального уровня оценивания.

Очевидно, назрела необходимость разработки и реализации адаптивного теста. При создании подобного программного продукта важна разработка механизма адаптации. С этой целью был проведен детальный анализ результатов ранее проведенных тестирований. Анализ выполнялся в целом по аудитории, по отдельным обучаемым и отдельным тестовым заданиям, а также блокам взаимосвязанных заданий. При этом основной информацией является правильность ответа (или степень решенности задачи) и время, затрачиваемое на решение задания.

Анализ результатов показал следующее:

* Выяснилось, что тест представлен в весьма сжатом виде, переход от одного задания к следующему означает значительное увеличение числа существенных операций, т.е. усложнений, с которыми даже средний ученик не справляется. Выявлены участки траекторий, где можно добавить промежуточные задания, содержащие меньшее количество существенных операций. Характерные признаки таких участков – резкое увеличение доли неверных ответов при переходе от задания к заданию.
* Знание понятий не является информативным признаком для направления адаптивного тестирования: и сильные, и слабые ученики могут иметь значительный разброс по этому признаку.
* У некоторых студентов явно отслеживается преобладание практических знаний над теоретическими. В частности, некоторые студенты не справились с теоретическими вопросами, но показали хорошие знания в практических заданиях.
* Даже самые сильные ученики имеют «выпадение» по отдельным темам, о чем свидетельствуют несколько неверных идущих подряд ответов.
* Среди неверных ответов встречаются случайные (необъяснимые) и закономерные. Отличительная особенность закономерных неверных ответов – их объяснимость в данном контексте, а также наличие не менее двух-трех одинаковых неверных.
* Среди заданий следует выделить опорные (базовые) задачи, по ответу на которые можно судить о степени овладения студентом темой. Такие задачи, как правило, содержат некоторый набор простых существенных операций.
* В тесте содержатся наиболее показательные задачи, правильное выполнение которых демонстрируют глубокое знакомство с темой.
* Для многих заданий требуется разработка дополнительного множества вспомогательных, уточняющих и «наталкивающих» подзаданий.

Подходящим инструментом реализации подобного рода программного продукта является технология экспертных систем.

Одной из главных компонент экспертной системы является база знаний. Для разработки базы знаний предстояло выявить схему рассуждений эксперта-преподавателя, предъявляющего задания одно за другим, с учетом предыстории процесса общения, и при этом наблюдающего за характером работы ученика: как быстро тот выполняет задание, какие ошибки допускает, какой наводящий вопрос следует задать или что подсказать; и на основании этой информации сформулировать, какое задание может быть следующим. Подобные знания предстояло вербализовать, структурировать, формализовать и перевести во внутреннюю форму представления с целью управления процессом тестирования.

На основе выявленных особенностей заданий были составлены правила. На основании этих правил была разработана база знаний, которая отражает всю логику теста. Подобные модели получили название логико-лингвистических.

Примеры таких правил:

* *ЕСЛИ* <студент не выполнил Базовое задание> *ТО* <студент, возможно, плохо представляет Базовое понятие>,
* *ЕСЛИ* <студент плохо представляет Базовое понятие> *И* <студент не ответил на дополнительный вопрос сложности 1 на узнавание по Базовому понятию> *ТО* <студент не владеет Базовым понятием>,
* *ЕСЛИ* <обучаемый успешно справился с группой заданий на узнавание по выбранной теме> *И* <частично справился с заданием на воспроизведение> *ТО* <следующим должно быть задание на уточнение воспроизведения>.

Правила позволяют гибко выявлять знания обучаемого. Достигается эта гибкость за счет составления дополнительного множества заданий по отдельным вопросам теста. Чтобы тест «не расползался», предусмотрена система метаправил. Такие метаправила на самом деле присутствуют и в рассуждениях эксперта. Примеры некоторых метаправил, использующихся для продолжения процесса тестирования:

* *ЕСЛИ* <ученик хорошо отвечает> *ТО* <делать скачки по дидактической цепочке>,
* *ЕСЛИ* <ученик отвечает очень плохо> *ТО* <выбирать задания на узнавание>.

Все рассуждения эксперта-преподавателя хранятся в базе знаний в предикатной форме в синтаксисе языка Пролог. Записаны эти высказывания не на естественном языке, а в символьной форме, что сокращает объем памяти и упрощает работу с базой. Между словесной и символьной формой существует взаимно однозначное соответствие.

Была выбрана подходящая архитектура экспертной системы. В ходе программной реализации пришли к необходимости разработки двух экспертных систем. Первая экспертная система создана для определения следующего вопроса и функционирует в ходе работы адаптивного теста, она оценивает текущую ситуацию в данный момент. Это означает, в частности, что если студент плохо делает задания по какой-либо теме, тест не прекращается, но меняется стратегия, выясняется, какие все-таки знания у обучаемого имеются.

Кроме того, первая экспертная система осуществляет подсчет текущей эвристической оценки на каждом шаге тестирования. Также система заносит в отдельную базу данных информацию об уровне знаний учащегося: если студент отвечает на вопрос частично и этот ответ является обоснованным (распознанным), то этот факт сохраняется, а программа предлагает дополнительный вопрос, чтобы точнее определить степень знаний. По ходу тестирования собираются отдельные дополнительные факты, характеризующие работу ученика. Например, можно предусмотреть случаи, когда ученик прибегал к отгадыванию. Также может быть зафиксирован факт быстрого выполнения сложной нетривиальной задачи.

Вторая экспертная система запускается после завершения тестирования. К этому моменту в системе собран материал по конкретному обучаемому. Здесь подводятся итоги, проводятся возможные обобщения накопленных фактов, выполняется диагностика с учетом сильных и слабых сторон, генерируется список достоинств и недостатков знаний обучаемого, проводится обобщение эвристической оценки и формируется результирующая оценка, а также оцениваются промежуточные результаты по темам.

Обе экспертные системы реализованы на диалекте языка Пролог под названием swi-prolog. Интерфейс электронного экзамена составляет модуль, написанный на языке РНР. Все задания, а также их характеристики хранятся в базе данных MySQL. На каждом шаге тестирования в php-файле реализуется запуск первой экспертной системы, работа которой осуществляется с использованием общей базы данных. Вызов кода swi-prolog выполняется из php-скрипта. Перевод сгенерированного списка диагностики на естественный язык осуществляется в интерфейсном php-скрипте.

Практическое применение модуля адаптивного тестирования показало, что программа способна решить проблемы из данной предметной области. База знаний, составленная на Прологе, позволяет накапливать опыт эксперта-преподавателя со всеми нюансами. Прототип можно наращивать до тех пор, пока он не достигнет желаемого уровня детальности. Для целей применения это существенно, хотя достигается это свойство дорогой ценой создания громоздкой базы знаний. Тем не менее применение логико-лингвистической модели позволяет использовать опыт эксперта-преподавателя со всеми тонкостями, которые обычно пропадают при управлении процессом тестирования, например в виде алгоритма.