

## **ПРАКТИЧЕСКИЕ АСПЕКТЫ ОПТИМИЗАЦИИ ЭЛЕКТРОННОГО ТЕСТИРОВАНИЯ**

*Аннотация. Описывается опыт разработки и внедрения в учебный процесс программы Unitest 1.1, предназначенной для тестирования студентов с целью контроля знаний.*

*Ключевые слова: тестирование, контроль знаний, программный продукт, тестовая система.*

*Дисциплины: информатика, педагогика.*

Тестирование как форма проверки знаний достаточно широко используется в современном образовании. Однако чтобы располагать системой, способной адекватно оценивать освоенные знания, необходимо обеспечить возможность проверки с использованием всех доступных способов. Унификация системы тестирования предполагает, что созданный продукт будет соответствовать всем требованиям, предъявляемым к тестированию.

Представление математической символики в рамках тестовых заданий является очень сложной задачей. Наиболее известный компонент, позволяющий отображать формулы, — Mathlabe. Данный модуль требует использования специальных библиотек символов и освоения процесса создания математической формулы. Наряду с этим решением большинство систем тестирования предлагают внедрение объектов Microsoft Equation 3.0 в качестве OLE-объектов. Этот путь реализации задачи ставит в зависимость вновь создаваемые программные продукты от уже имеющихся приложений и создает дополнительные трудности размещения данных.

В качестве альтернативных решений можно:

1. использовать в качестве поля создания вопроса компонента TRichedit (в рамках программного продукта Delphy 7.0), что позволит идти стандартным путем создания формул с помощью символов текстового редактора Word (с последующим их копированием через буфер обмена), при этом осуществляя необходимое форматирование. Вновь созданным программным приложением сможет с успехом пользоваться потребитель с базовыми знаниями компьютера;
2. воспользоваться (согласно авторской разработке Ю. К. Карповой, А. В. Ежова — Unitest 1.1) встроенным графическим редактором и сохранить созданную формулу в качестве рисунка в случае, если первый вариант внедрения объектов по каким-либо причинам окажется недоступным;
3. сохранить в качестве изображения математические формулы с использованием графического планшета, т. е. путем начертания их вручную на поверхности с помощью пера.

Несмотря на серьезные разработки по проблемам тестирования, программный продукт Unitest 1.1 выгодно отличается:

1. высокой степенью универсальности, поскольку позволяет преподавателю с минимальными пользовательскими навыками работы на компьютере быстро и легко преобразовать как внешнее, так и внутреннее содержание тестирования;
2. максимальной приближенностью к реальному процессу тестирования, так как существует возможность заранее определить или же изменить критерии оценки знаний;
3. возможностью поддерживать более десяти различных типов тестирования: выбор одного правильного ответа из многих, многих из многих, использование одного или нескольких полей ввода, соответствие, упорядочение, выбор области на изображении, однозначный или развернутый ответ и т. д.;
4. возможностью импорта итоговых результатов в табличный редактор Excel для ведения статистики при небольшом объеме самого программного продукта (менее 50 Мб), что является актуальным в электронных учебных пособиях на дисковых носителях;
5. гибкой системой формирования проверочных заданий путем случайной выборки в равных соотношениях из предложенных разделов (количество заданий для итогового тестирования также может быть задано самим преподавателем предварительно).

Предлагаемая вашему вниманию программа разработана для Центра дистанционного обучения Института экономики, управления и права (г. Казань). Основной целью было создание продукта, который бы позволил корректировать сложность предложенных студенту в ходе тестирования заданий. Программа состоит из двух модулей, один из которых предназначен для редактирования содержания и формы тестовых заданий. Этот модуль защищен паролем, чтобы исключить возможность просмотра правильных ответов в ходе тестирования. Преподаватель может сам указывать уровень сложности представленных в тесте заданий (высокий, низкий, средний). Определение уровня сложности в программе строится по следующей схеме:

$$\frac{\text{уровень сложности (\%)} \cdot \text{количество правильных ответов}}{\text{количество вопросов} \cdot 20}$$

Формирование итогового количества баллов относительно 100 % правильных ответов проведено исходя из стандартной пятибалльной системы оценки знаний. Число двадцать в знаменателе означает индекс отношения 100%-го уровня сложности к пяти баллам оценки. Алгоритм вычисления содержится в специально созданном модуле Unittesting.pas. Рекомендуемые уровни сложности регулируются с помощью подвижной линейки в диапазоне от 100 до 120 %, однако существует возможность установить любое значение в специально отведенной ячейке программы. Например, если преподаватель установил уровень сложности высокий, то при ответе на шесть вопросов из десяти студент

$$\frac{100\% \cdot 6}{10 \cdot 20} = 3$$

получит оценку «3» ( ). При установке преподавателем низкого уровня сложности, в котором расчет предлагается вести от 120 %, этот же студент получит оценку «4». Подобный подход позволяет максимально приблизить реальную систему тестирования к

автоматизированной, а также учитывать возможность индивидуального подхода, на котором базируется современная педагогика.

Разработанный программный продукт успешно апробирован в системе дистанционного обучения Института экономики, управления и права (г. Казань), занял первое место на городском конкурсе «Мастер-класс по информационным технологиям» в 2006 году в номинации «Учебная программа».

В процессе разработки программы в ее структуру был внесен ряд изменений, что позволило повысить первоначальную версию Unitest 1.0 до Unitest 1.1. В частности, скорректирована обработка создания базы данных результатов тестирования, сокращено число используемых переменных, разработана система автоматической замены данных расположения конфигурационного файла. Среди внесенных изменений особого внимания заслуживает возможность сравнения исходных и вносимых значений в ходе тестирования. В программе используется схема непосредственного сопоставления ответов, что обеспечивает универсальность. Однако необходимо учесть, что форма создания ответа студентом может быть различной, т.е. в ходе заполнения итогового поля могут быть использованы случайные пробелы или не прописанные заранее специальные символы. Например, в вопросе на сопоставление значений, состоящем из двух категорий по три значения в каждом, запись ответа может иметь вид 1-3, 2-2, 3-1. В указанном случае, тестируемым в различных комбинациях могут быть использованы как пробелы, так и другие знаки препинания, кроме запятой. Конечно, ограничения начертания создаются путем создания шаблонной формы, однако это нагружает интерфейс программы и требует предварительного изучения разработанных шаблонов. Нами предложена стратегия идентификации, широко используемая в штрих-кодировании. То есть перед сравнением значений из ряда правильного ответа и ответа тестирования будут искусственно удаляться все посторонние символы и пробелы, что обеспечит точность в проведении идентификации без перегрузки интерфейса. Следует отметить необходимость приведения полученных комбинаций ответов к единому виду. Допустим, что при ответе на вопрос теста испытуемый задаст другую последовательность ответа 2-а, 1-б, 3-с. Анализ результатов тестирования может в этом случае быть представлен в виде схемы:



**Схема.** Последовательность обработки результатов тестирования

Таким образом, несмотря на значительное разнообразие взглядов на организацию тестирования, можно создавать универсальные элементы с устойчивым показателем работоспособности.

### **Список использованной литературы**

1. **Тагиров Р. Т.** Новые информационные технологии как фактор эффективности современного образования / Р. Т. Тагиров, З. К. Тагирова // Материалы Республик. науч.-практ. конф. «Информатизация образования в РТ: опыт, проблемы, перспективы». Ч. 1. Казань, 2006. С.306.
2. **Косарева Н. И.** Проблемы стартового этапа дистанционного обучения в учебном заведении / Н. И. Косарева, А. Ф. Сук // Образование и виртуальность – 2004 (ВИРТ – 2004). Харьков : Ялта: УАДО, 2004. С. 250—255.