

УДК 378.146:004.5

Н. Н. Буснюк¹, В. А. Новиков²¹Белорусский государственный технологический университет²Белорусский государственный университет**КОМПЬЮТЕРНОЕ ТЕСТИРОВАНИЕ ЗНАНИЙ
С ПОЗИЦИЙ МЕТОДОЛОГИИ OLTP**

В связи с интенсивным развитием технологий компьютерного обеспечения учебного процесса появилась задача создания компьютерного теста с учетом альтернативных вариантов простоты и сервисных возможностей. В работе приводится методика компьютерного тестирования в среде Internet. Предлагается в оболочке теста практически не использовать сервер, что позволяет создавать программный продукт легко переносимый с компьютера на компьютер. Средства динамического HTML позволяют создавать программы с зашифрованным набором данных и фрагментов программного кода. Настройку теста предлагается осуществлять на языке высокого уровня C#. Рассмотрены также аспекты оценивания результатов тестирования с учетом сложности вопроса.

Ключевые слова: дистанционное обучение, тестирование, методология OLTP, вопросы, ответы, программирование, язык HTML, язык C#, язык JavaScript, среднее значение, среднеквадратичное отклонение.

N. N. Busnyuk¹, V. A. Novikov²¹Belarusian State Technological University²Belarusian State University**COMPUTER TESTING FROM OLTP METHODOLOGY POSITIONS**

In connection with the intensive development of computer technology for the educational process, the task is to create a computer test, taking into account alternative options for simplicity and service capabilities. The paper proposes a technique for computer testing in the Internet environment. It is offered in the shell of the test practically not to use the server, which allows you to create a software product that is easily portable from computer to computer. Dynamic HTML tools allow you to create programs with an encrypted set of data and code snippets. The test setting is suggested in the high-level language C#. The aspects of evaluation of testing results are also considered taking into account the complexity of the issue.

Key words: distance learning, testing, OLTP methodology, questions, answers, programming, HTML language, C# language, JavaScript language, mean value, standard deviation.

Введение. Дистанционное обучение как традиционная и перспективная форма преподавания получает все большее развитие в системе современного образования. Неотъемлемой частью дистанционного обучения являются компьютерные тесты. Компьютерное тестирование – это основная составляющая не только дистанционного обучения, но и необходимый элемент традиционной формы обучения [1].

Процесс активного внедрения компьютерных тестов в учебный процесс сопровождается не только организационными проблемами, но и чисто техническими. Последнее связано с непростым процессом подготовки теста и дальнейшего его обновления.

Основная часть. При разработке программной оболочки теста необходимо учитывать современные тенденции и рекомендации. Таким базовым руководством может быть активно пропагандируемая в настоящее время методология OLTP (Online Transaction Processing). Применительно к информационным системам OLTP-системы предназначены для ввода, струк-

турированного хранения и обработки информации (операций, документов) в режиме реального времени. Основными компонентами OLTP являются механизмы:

- транзакционного обслуживания клиентов;
- сбора статистической информации;
- подготовки и обновления базы данных.

Механизмы транзакционного обслуживания применительно к компьютерному тестированию предусматривают отсутствие задержек в процессе тестирования и простую и понятную систему интерфейсного обеспечения.

С позиций сбора статистической информации нужно учитывать только самые необходимые для ведения учебного процесса данные и не усложнять программную оболочку чрезмерным ее администрированием. На наш взгляд, сбор статистической информации можно ограничить для каждого вопроса теста количеством его появлений и числом правильных ответов на этот вопрос. Такая статистика позволяет преподавателю периодически делать коррекцию весового коэффициента вопроса. Если учитывать

программную оболочку теста, как независимую составляющую, то нет смысла вести статистику по фамилиям студентов как при обучении, так и при тестировании. Подобная статистика привлекательна, но слишком усложняет администрирование и в связи с этим обновление теста.

Самым существенным механизмом организации компьютерного теста является механизм подготовки теста и его оперативное обновление. Наиболее полно методология OLTP обеспечивается комплексным использованием технологии Internet с применением в качестве связующего элемента языков высокого уровня (например, C#).

При создании компьютерного теста надо учитывать возможность применения в нем помимо текста еще и формул, диаграмм и картинок. Наиболее полно эта возможность обеспечивается средствами Word. Оптимальным вариантом разметки является использование таблиц для формирования каждого вопроса. Необходимо предусмотреть три категории вопросов: вопрос-перебор, вопрос-эталон и вопрос-список.

Для вопроса-перебора табличка представляется, как это показано в табл. 1.

Таблица 1

Вопрос-перебор

+	3.5
-	6
-	2
-	4
\$	Чему равно значение интеграла $\int_1^2 (x + 2) dx$?
@	Вес=1; Категория=2; Колонки_ответов=4;

Символом «@» обозначается строка параметров: весовой коэффициент вопроса, категория сложности вопроса и число колонок при выводе ответов. Символом «\$» обозначается строка вопроса. Строки ответов вопроса-перебора идентифицируются символом «+» для правильного ответа и символом «-» для неправильного ответа. В вопросе, как и в ответах, может быть любой текст, кроме таблиц (в приведенном примере это формула). В компьютерном тесте вопрос-перебор выглядит так, как это представлено на рис. 1.

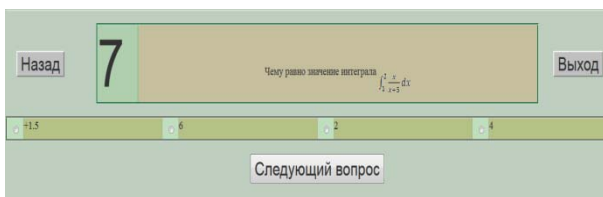


Рис. 1. Отображение вопроса-перебора в тесте

Для вопроса-эталона табличка Word и ее отображение представлены в табл. 2 и на рис. 2.

Таблица 2

Вопрос-эталон

@	Вес=1; Категория=1; Колонки_ответов=1;
\$	Кто открыл теорию относительности
*	Эйнштейн
*	Альберт Эйнштейн
*	Эйнштейн Альберт

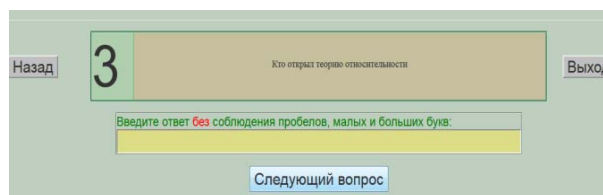


Рис. 2. Отображение вопроса-эталона в тесте

Как видно, признак вопроса-эталона определяется по вариантам правильных ответов, идентифицируемых символом «*».

Для вопроса-списка табличка Word и ее отображение представлены в табл. 3 и на рис. 3.

Таблица 3

Вопрос-список

\$	Расположите в хронологическом порядке по дате рождения писателей
1	Нарезный
2	Лев Толстой
3	Есенин
4	Булгаков

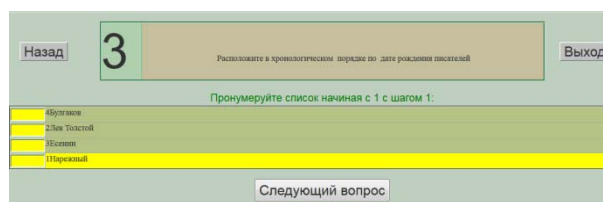


Рис. 3. Отображение вопроса-списка в тесте

Признак вопроса-списка определяется по вариантам правильного ответа в виде нумерации, начиная с цифры 1.

Для дальнейшей обработки размеченного документа его необходимо конвертировать из Word в формат HTML с фильтрацией. Дальнейшая обработка документа HTML зависит от средств организации интерфейсной части программной оболочки.

С позиций доступности компьютерного теста и обеспечения транзакционного обслуживания клиентов наиболее подходят средства организации интерактивных оболочек в Internet. Они позволяют организовывать доступный и

понятный интерфейс средствами HTML и JavaScript. С другой стороны, программирование на стороне сервера позволяет обеспечить сбор статистических данных любой сложности.

С позиций методологии OLTP компьютерный тест должен в отсутствие статистики работать на любом локальном компьютере, чем обеспечивается принцип доступности теста. Таким образом, программная оболочка теста должна обслуживаться только средствами обработки на клиентской стороне с полным исключением из этого процесса серверной обработки данных.

Изложенная концепция компьютерного теста частично реализована в статье [2]. Тест предусматривает два режима: тестирование и обучение. Режим обучения отличается от тестирования только возможностью вывода после регламентированного числа попыток правильной комбинации ответов. В режиме обучения в качестве ответа засчитывается первая попытка. Кроме режима обучения тест предусматривает два уровня тестирования: уровень 1 облегченного варианта и уровень 2 полной версии.

В конце тестирования выводится стандартный протокол тестирования (рис. 4).

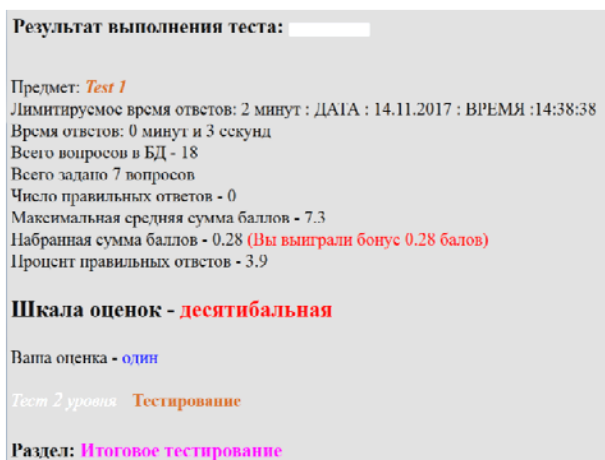


Рис. 4. Протокол тестирования

В режиме сбора статистики протокол тестирования сопровождается фиксацией заданных вопросов и правильных ответов на них. Структурировано тест предусматривает наличие нескольких разделов по предмету с возможностью тестирования по каждому разделу. Каждый раздел может содержать несколько тем с возможностью обучения по каждой теме. При тестировании по разделу число вопросов берется равномерным по каждой теме. Итоговое тестирование сводится к равномерному выбору числа вопросов по разделу, а внутри раздела – и по темам.

Обеспечение сервиса и настройку теста наиболее оптимально организовывать на языке

высокого уровня, которым является Visual Studio C#. В первую очередь необходимо конвертировать документ HTML каждой темы, полученный при подготовке теста в Word, в спецификацию данных на JavaScript. Для вопросов и ответов данные хранятся в ступенчатом массиве JavaScript. Строкового типа элемент массива представляет собой HTML-фрагмент, который и отображается в окне браузера.

Кроме конвертации на языке C# несложно создать и настройку теста с использованием механизма создания на C# соответствующих фрагментов JavaScript-кода. В этот сервис входят фиксация разделов, привязка темы к разделу, формирование критериев выставления оценок и необходимых режимов типа проверки теста или режима вывода зависимых переключателей для ответов.

Непростой задачей при тестировании является определение критериев оценивания по каждому вопросу и затем по всем вопросам [3]. Анализ имеющихся точек зрения показал, что наиболее подходящим является зачет правильного ответа на вопрос, если выставлены правильно все флажки ответов. Для учета комбинаторной сложности ответов в вопросе-переборе предлагается, например, для числа вариантов $s = C_8^4$ удваивать вес вопроса ($k = 2$). Вес вопроса – это количество баллов, засчитываемое за правильный ответ. Для меньших комбинаций C_n^m при $m < 4$ и $n < 8$ вес χ берется прямо пропорционально между значением $C_2^1 = 2$ и s :

$$\chi = \frac{s - 2k + C_n^m}{s - k}.$$

Другой задачей оценивания результатов при случайном выборе вопросов темы является учет возможности появления комбинации легких вопросов, с одной стороны, или тяжелых вопросов, с другой стороны. Можно с позиций объективности оценивания знаний предложить следующий алгоритм учета этой ситуации. Предположим, тест содержит N вопросов, а выводятся случайно K вопросов. Отметка выставляется по десятибалльной шкале, исходя из процентного отношения суммы баллов правильных ответов S к суммарному числу баллов по выводимым вопросам KW_* , где W_* – средний вес вопроса среди K штук. Например, при проценте правильных ответов $100S / KW_* = 65\%$ выставляется отметка 6 (шесть). Фиксируется среди N вопросов среднеарифметический вес вопроса W_0 , а при отображении – среднеарифметический вес W_k вопроса по числу K выводимых вопросов. Если для тестирования выведены вопросы с $W_k < W_0$, то берется $W_* = W_k$. Если выведены сложные вопросы с $W_k > W_0$, то берется $W_* = W_0$.

С другой стороны, можно предложить алгоритм тестирования при заданном количестве баллов и с плавающим числом вопросов.

Прежде всего, вычисляется среднее арифметическое значение W_0 среди всех вопросов теста по формуле

$$W_0 = \frac{1}{N} \sum_i w_i,$$

где w_i – весовой коэффициент вопроса.

Далее вычисляется среднее квадратичное отклонение σ :

$$\sigma = \sqrt{\frac{1}{N-1} \sum_i (w_i - W_0)^2}.$$

По заданному преподавателем числу выводимых для тестирования вопросов K вычисляется среднее значение суммы баллов Σ , которую тестируемый должен набрать:

$$\Sigma = KW_0.$$

Теперь при тестировании число задаваемых вопросов K_0 оказывается плавающим и определяется величиной Σ . При выводимых легких вопросах величина K_0 будет больше K , а при тяжелых вопросах K_0 будет меньше K . Но если ограничиться только этим алгоритмом, то могут быть ситуации задания слишком малого $K = K_{\min}$ или слишком большого $K = K_{\max}$ числа вопросов. В связи с этим величины K_{\min} и K_{\max} ограничиваются по значению σ :

$$K_{\min} = \frac{\Sigma}{W_0 + \sigma}; \quad K_{\max} = \frac{\Sigma}{W_0 - \sigma}.$$

Таким образом, при тестировании число задаваемых вопросов зависит от величины Σ , но не может быть меньше K_{\min} и больше K_{\max} . Так алгоритм позволяет исключить необъективность оценивания при случайном выборе сложных вопросов.

Предлагаемая методология организации программной оболочки теста позволяет полно-

стью автоматизировать процесс его подготовки. Кроме этого, самая сложная задача разметки теста сводится к набору текста в Word с простой системой разметки вопросов и ответов. Применение в качестве буфера программной оболочки на C# позволяет выполнять предварительную настройку теста и его сервиса любой сложности. Средства Internet используются только для организации удобного интерфейса и сбора необходимой статистики.

С позиций организации всего комплекса дистанционного обучения тест легко интегрируется в систему выдачи дидактического материала. Более того, при создании на основе предлагаемой методологии электронных учебников тест несложно монтируется в систему обеспечения сервиса таких документов. Он из электронного учебника позволяет войти в оболочку, минуя систему обычного доступа к теме. Также он дает возможность выдавать на сервер через элементы формы фамилию тестируемого и его отметку, причем фамилия поступает на тест из электронного учебника через командную строку. Подобный механизм обеспечивает синхронизацию многопользовательского режима при обслуживании электронного учебника.

Заключение. Учет рекомендаций, основанных на методологии OLTP, дает возможность избежать слишком запутанной системы навигации по интерфейсу программной оболочки и обеспечить простую технологию наполнения базы данных вопросами и ответами теста. Учитывая, что компьютерный тест – тоже специфическая база данных, использование прогрессивной технологии структурирования данных дает возможность создания самой простой процедуры подготовки теста средствами, непосредственно предназначенными для такого структурирования и отображения [4]. Средства динамического HTML позволяют шифровать данные и фрагменты программного кода JavaScript с использованием языка высокого уровня типа C#.

Литература

1. Никифоров О. Ю., Кокшарова Е. И. Комплекс признаков классификации систем компьютерного тестирования // *Современные научные исследования и инновации*. 2013. № 6. С. 31–36.
2. Новиков В. А., Шипулина Л. Г. Универсальный тест на базе Internet // *Образовательно-инновационные технологии: теория и практика*. Кн. 9. Воронеж, 2009. С. 43–54.
3. Звонников В. И., Челышкова М. Б. *Современные средства оценивания результатов обучения: учеб. пособие для студентов высш. учеб. заведений*. М.: Академия, 2007. 224 с.
4. Сиренко С. Н. Компьютерное тестирование в вузе: преимущества, особенности, опыт внедрения // *Педагогические измерения*. 2007. № 4. С. 67–74.

References

1. Nikiforov O. Yu., Koksharova E. I. Complex of signs classification of computer testing systems. *Sovremennyye nauchnye issledovaniya i innovatsii* [Modern scientific research and innovation], 2013, no. 6, pp. 31–36 (In Russian).

2. Novikov V. A., Shipulina L. G. Universal test on the basis of the Internet. *Obrazovatel'no-innovatsionnye tekhnologii: teoriya i praktika* [Educational-innovative technologies: theory and practice, Book 9]. Voronezh, 2009. Pp. 43–54.

3. Zvonnikov V. I., Chelyshkova M. B. *Sovremennye sredstva otsenivaniya rezul'tatov obucheniya: uchebnoye posobiye dlya studentov vysshikh uchebnykh zavedeniy* [Modern means of assessing learning outcomes: Textbook allowance for students supreme training institutions]. Moscow, Akademiya Publ., 2007. 224 p.

4. Sirenko S. N. Computer testing in the university: advantages, features, experience of implementation. *Pedagogicheskiye izmereniya* [Pedagogical measurements], 2007, no. 4, pp. 67–74 (In Russian).

Информация об авторах

Буснюк Николай Николаевич – кандидат физико-математических наук, доцент, доцент кафедры информационных систем и технологий. Белорусский государственный технологический университет (220006, г. Минск, ул. Свердлова, 13а, Республика Беларусь). E-mail: busnnn@belstu.by

Новиков Василий Алексеевич – кандидат технических наук, доцент, доцент кафедры экономической информатики. Белорусский государственный университет (220004, г. Минск, пр-т Независимости, 4, Республика Беларусь). E-mail: vanovikov@tut.by

Information about the authors

Busnyuk Nikolay Nikolaevich – PhD (Physics and Mathematics), Associate Professor, Assistant Professor, the Department of Information Systems and Technology. Belarusian State Technological University (13a, Sverdlova str., 220006, Minsk, Republic of Belarus). E-mail: busnnn@belstu.by

Novikov Vasily Alekseevich – PhD (Engineering), Associate Professor, Assistant Professor, the Department of Economical Informatics. Belarusian State University (4, Nezavisimosti Ave., 220004, Minsk, Republic of Belarus). E-mail: vanovikov@tut.by

Поступила 28.11.2017