

## ОБ ОДНОЙ ПРОГРАММЕ КОМПЬЮТЕРНОГО ТЕСТИРОВАНИЯ

*Макалада компьютердин жардамы менен тесттөөчү жана окутуучу программанын бири келтирилет.*

*В статье рассматривается одна программа компьютерного тестирования и обучения.*

*A program for computer testing and teaching is considered in this article.*

В настоящее время на мировом рынке имеются тысячи обучающих программ по разным дисциплинам. Только перечень фирм, разрабатывающих обучающие программы, занимает десятки страниц. Значение обучающих программ с растёт каждым годом, ибо они используются не только в учебных аудиториях и классах, но и в системе дистанционного обучения, роль которого в связи с широким распространением Интернета по всему миру становится незаменимой /1, 2/.

В данной статье обсуждаются структура и возможности одной простой обучающей программы, которую можно использовать как в учебных аудиториях, так и в среде Интернет. Благодаря своей простой структуре она может быть легко установлена в локальных сетях и использоваться и как программа тестового контроля, и как обучающая программа без преподавателя. Она универсальна, т.е. не ориентирована на определенную предметную область, обладает определенными игровыми свойствами и, думается, имеет шансы широко использоваться как в системе традиционного, так и дистанционного обучения.

Комплекс программ «Электронная обучающая система» состоит из двух программ, работающих с единой базой данных, в составе которой имеются три таблицы:

1. Программа ввода тестовых данных “Vvod”.
2. Программа сеанса обучения по введенным тестовым данным “Prov”.

Обе программы написаны на языке Object Pascal в среде Delphi и используют интерфейс BDE для установления связи с таблицами базы данных.

1. Программа ввода тестовых данных в таблицы “Kluch” и “Test” или “Kluch1” и “Test1”. Первая пара таблиц предназначена для ввода тестовых данных по языку Лисп, вторая – по языку Пролог.

Данные вводятся с электронной или отсканированной с бумажного документа версии тестовых данных. Предусмотрен полуавтоматический ввод вводимых данных с указанных источников данных посредством выделения текста в источнике, записи его в буфер ввода и с буфера непосредственно в окно редактирования на форме с последующей записью в таблицы данных.

Ввести данные можно либо с языка Лисп, либо с языка Пролог с установлением нужного режима ввода.

Таблицы “Kluch” и “Test” или “Kluch1” и “Test1” попарно имеют идентичные структуры. Для примера приводим структуры таблиц “Kluch” и “Test”.

Таблица “Kluch” состоит из трех полей:

1. Nv – числовое поле, содержащее номер вопроса теста (ключевое поле).
2. Npo – числовое поле, содержащее номер правильного варианта ответа.
3. Tv – текстовое поле, содержащее текст вопроса теста.

Таблица “Test” состоит из трех полей:

1. Nv – числовое поле, содержащее номер вопроса теста.
2. Vr – числовое поле, содержащее номер варианта ответа.
3. Vorp – текстовое поле, содержащее текст варианта ответа.

Ключом данной таблицы служит набор полей Nv и Vr.

Форма интерфейса программы состоит из двух разделов “Ввод тестовых вопросов” и “Ввод ключей”:

1. Раздел “Ввод тестовых вопросов” служит для ввода данных для таблицы “Kluch”.
2. Раздел “Ввод ключей” служит для ввода данных для таблицы “Test”.

При вводе текста с электронного документа используется следующий прием: после выделения фрагмента текста в документе с помощью комбинации клавиш Ctrl+C выделенный текст копируется в буфер обмена. Запись его в конкретное поле, используя промежуточное поле редактирования, производится согласно, например, следующему фрагменту программы:

```
Edit2.Clear;  
Edit2.PasteFromClipboard;  
Table2.Fields[2].Value:=Edit2.Text;
```

Интерфейс ввода во время работы программы показан на рис. 1.

Ввод значений других числовых полей производится обычным образом с помощью компонент DBNavigator1 и DBNavigator2.

Программа сеанса обучения по введенным тестовым данным “Prov” использует таблицы “Kluch”, “Test” и “Rab”. Описание таблиц “Kluch” и “Test” приведены при описании программы “Vvod”.

Таблица “Rab” состоит из трех полей:

1. Nv – числовое поле, содержащее номер вопроса теста (ключевое поле).
2. Npo – числовое поле, содержащее номер правильного варианта ответа.
3. No – числовое поле, содержащее номер ответа, заданного пользователем.

В форме программы (рис. 2) можно задавать количество вопросов сеанса по заданному тесту. Режим ввода предусматривает тесты по языку Лисп или по языку Пролог.

Программа “Prov” имеет следующие обработчики событий:

1. procedure Vvod(i: integer);

2. procedure Nseans;
3. procedure RadioButton1Click(Sender: TObject);
4. procedure RadioButton2Click(Sender: TObject);
5. procedure FormActivate(Sender: TObject);
6. procedure Button1Click(Sender: TObject);
7. procedure Button2Click(Sender: TObject);
8. procedure Button4Click(Sender: TObject);
9. procedure Edit3KeyPress(Sender: TObject; var Key: Char);

1. Процедура Vvod выполняет действия по выводу очередного тестового вопроса на экран.

На заголовок метки Label5 выводится # очередного вопроса. На заголовок метки Label6 выводится # вопроса по таблице “Kluch”. Затем осуществляется поиск записи с заданным тестовым вопросом в таблице “Kluch”. При нахождении искомой записи текст вопроса записывается на заголовок метки Label7. Если длина текста вопроса больше 81, остальная часть текста записывается на заголовок метки Label8.

Затем осуществляется поиск номера вопроса по таблице “Kluch”. По нахождении искомой записи номера вариантов и варианты ответов выводятся на экран на заголовки меток Label9, Label10, Label11 и Label12.

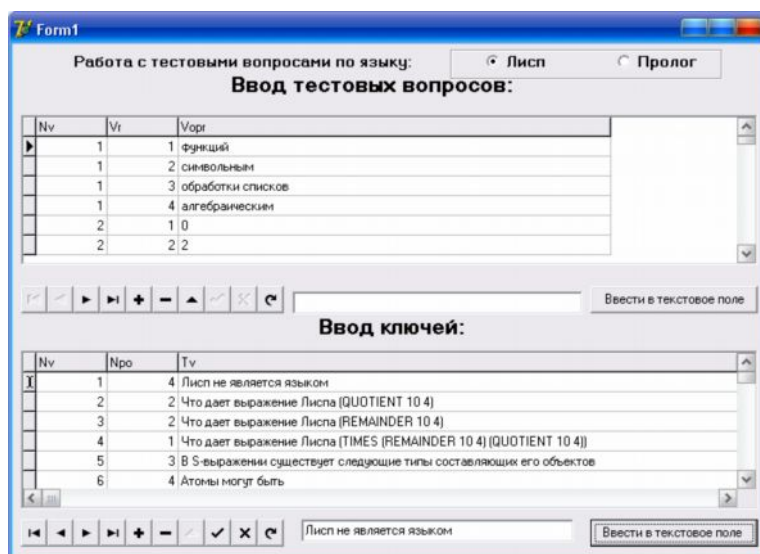


Рис. 1. Интерфейс программы “Vvod ” при вводе данных

2. Процедура Nseans, используя генератор случайных чисел, определяет случайным образом номера предлагаемых тестовых вопросов и записывает в массив для использования в дальнейшем.
3. Процедура RadioButton1Click – выбор режима языка Лисп, вывод количества записей.
4. Процедура RadioButton2Click – выбор режима языка Пролог, вывод количества записей.
5. Процедура FormActivate – установка начальных значений некоторых переменных. Установка по умолчанию режима языка Лисп, вывод количества записей.

6. Процедура Button1Click – подготовка к новому сеансу работы (кнопка «Начало сеанса»).

Очистка таблицы 'Rab' для нового сеанса работы, проверка правильности количества введенных вопросов, выполнение процедуры Nseans; определяющих случайным образом номера предлагаемых тестовых вопросов и записывающих их в массив для использования в дальнейшем.

7. Процедура Button2Click – ввод и обработка принятого ответа (кнопка «Принять»).

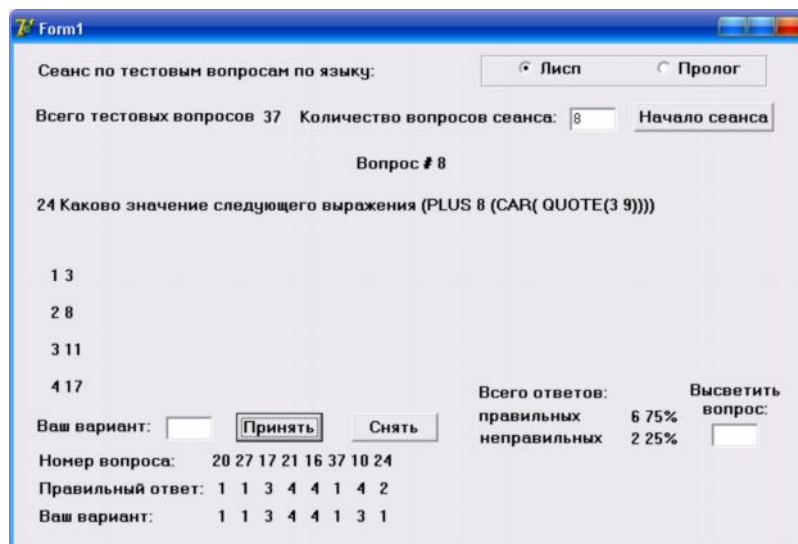


Рис. 2. Интерфейс программы “Prov ” при выполнении

Процедура осуществляет проверку номера введенного варианта на правильность, затем создает новую запись в таблице 'Rab' с номером тестового вопроса номером правильного варианта ответа и номера, введенного пользователем варианта ответа. Если количество вопросов не исчерпано, программа переходит к вводу очередного тестового вопроса, в противном случае на заголовки меток Label19, Label20, Label21 выводятся тексты: 'Номер вопроса:', 'Правильный ответ:', 'Ваш вариант: ' соответственно. Поскольку записи в таблицу 'Rab' вставляются в обратном к естественному порядку, значения номера вопроса, правильного ответа, предлагаемого пользователем варианта выводятся начиная с конца таблицы. Наконец, на заголовок метки Label15 выводится количество и % правильных ответов, а на метку Label1 – количество и % неправильных ответов.

8. Процедура Button4Click – отменяет введенные данные (кнопка “Снять”).

Чистит окна редактирования Edit1 и Edit2. Закрывает таблицы “Kluch” и “Test”.

9. Процедура Edit3KeyPress(Sender: TObject; var Key: Char); – высвечивает заданный пользователем вопрос на экране.

Осуществляет вывод на экран требуемого пользователем вопроса после нажатия им на клавишу Enter.

Имеются следующие общие (глобальные) переменные:

Массив целых чисел A из 30 элементов – предназначен для хранения номеров введенных тестовых вопросов.

Целые переменные Kv, Kz содержат количество введенных вопросов, количество записей в таблице 'Kluch' соответственно.

Целые переменные i, Npo, No: содержат номер текущего вопроса, номер правильного ответа, номер введенного пользователем варианта ответа.

Программной единицей (unit), содержащей код программы, является T1.

Программа Pgov обладает следующими свойствами.

- Программа является универсальной в том смысле, что может работать с любым исходным тестовым набором, отличным от указанных выше.
- Свои знания пользователь может проверить на любом подмножестве исходного набора тестовых вопросов и получить исчерпывающую информацию об уровне своих знаний по предмету.
- Пользователь может высветить любой вопрос из заданного набора тестовых вопросов для уточнения правильности или неправильности своего ответа.
- Набор тестовых вопросов выбирается программой в случайном порядке, что исключает заведомо формальный правильный ответ на заданный вопрос. Пользователем задается лишь количество вопросов.
- Программа может использоваться как для тестового контроля, так и в качестве обучающей программы, так как позволяет пользователю проверять и уточнять свои знания по предмету на любом наборе вопросов любое количество раз, начиная с малого, кончая полным набором тестовых вопросов.

## СПИСОК ЛИТЕРАТУРЫ

1. Компьютерные и информационные технологии обработки и анализа данных: Сборник научных статей / Под ред. С.С. Садыкова. – Муром: МИ ВлГУ, 2001. – с. 78-83.

2. Таннинг Ж.Ф. Технология и структура программного обеспечения для дистанционного обучения // Дистанционное обучение и новые технологии в образовании: Материалы региональной НТК / Владим. гос. ун-т. – Владимир, 2001. – С. 45-47.