

**SECTION 4. Computer science, computer engineering and automation.**

**Shevtsov Alexandr Nikolayevich**  
 candidate of technical Sciences, associate  
 Professor of the Department «Applied mathematics»  
 Taraz State University named after M.Kh. Dulati,  
 Kazakhstan

**Shyrynkhanova Dinara Zhaksylykovna**  
 1 year magistr of the speciality "Information  
 systems "  
 Taraz State University named after M.Kh. Dulati,  
 Kazakhstan



**DEVELOPMENT OF ALGORITHMS AND APPLICATION  
 COMPONENT MODEL TO ANALYZE AND FIX ERRORS TEST.**

*The process of working of the Center for testing University, and includes the reception of test tasks of teachers in various disciplines, validation of recruitment and clearance for entry into the database and its further use in examinations.*

*One of the most difficult and crucial moments - is checking the correctness of test and identification of errors. This issue is devoted to this article.*

*Keywords: test automation, Delphi.*

**РАЗРАБОТКА АЛГОРИТМОВ И ПРИЛОЖЕНИЯ  
 КОМПОНЕНТНОЙ МОДЕЛИ ДЛЯ АНАЛИЗА И ИСПРАВЛЕНИЯ  
 ОШИБОК ЭКЗАМЕНАЦИОННОГО ТЕСТА**

*Процесс работы Центра тестирования университета, включает в себя – прием тестовых заданий преподавателей по различным дисциплинам, проверку правильности набора и оформления, занесение в базу данных и дальнейшее его использование на экзаменах.*

Одним из наиболее сложных и ответственных моментов – является проверка правильности оформления тестового задания, и выявление допущенных ошибок. Именно этой проблеме посвящена данная статья

*Ключевые слова: экзамен, тест, автоматизация, Дельфи.*

Разработаем программу, которая в сочетании с набором уже используемых в Центре тестирования, будет образовывать компонентную модель. А также алгоритмы для автоматической проверки теста, нахождения ошибок, и автоматического их исправления в соответствии с требованиями.

Определим правильную форму тестового задания (рис.1-2).

1. Квадратная матрица называется диагональной, если....¶ A) все элементы вне главной диагонали равны нулю¶ B) все элементы побочной диагонали равны нулю¶ C) все элементы главной диагонали равны нулю¶ D) все элементы матрицы равны нулю¶ E) все элементы вне главной диагонали равны единице¶ ¶
2. Прямоугольная матрица $A$ размеров $m \times n$ называется квадратной, если...¶ A) $m = n$ ¶ B) $m \neq n$ ¶

Рисунок 1 – Образец тестового задания.

№ вопроса	Вопрос	¶
A)	Правильный ответ	¶
B)	Ошибочный ответ	¶
C)	Ошибочный ответ	¶
D)	Ошибочный ответ	¶
E)	Ошибочный ответ	¶
		¶

Рисунок 2 – Блок схема тестового задания.

Определим наиболее типичные ошибки встречающиеся в тестовых заданиях:

- Наличие лишнего знака ¶ (Enter), два и более подряд.
- Знак ¶ (Enter) находится между вариантами ответов.
- Два правильных ответа.
- Ошибочная нумерация вопросов.
- Отсутствие точки после номера вопроса.
- Отсутствие скобки после буквы(индекса) ответа.
- Два одинаковых индекса ответов.
- Вопрос разбит на две строки знаком ¶ (Enter)
- Лишние пробелы в зоне индексов вопросов и ответов.
- Повторяющиеся вопросы в тесте.

Разработаем программу в среде «Embarcadero® RAD Studio 2010». Редактирование теста будем осуществлять непосредственно в программе Microsoft Word, посылая определенные запросы на его сервер. Для этого встроим основные серверные компоненты Word в нашу программу. Нам потребуются:

- MSWORD.OLB
- Office\_Tlb.dcu
- Office\_TLB.pas
- VBIDE\_TLB.dcu
- VBIDE\_TLB.pas
- Word\_tlb.dcu
- Word\_TLB.pas

Теперь займемся разработкой алгоритмов.

Для нахождения лишних знаков ¶ (Enter), встречающихся в тесте, будем проходить по всему тесту сравнивая две рядом стоящих строки и в случае их равенства будем удалять вторую из них (проверяя наличие постоого переноса строки).

**code:**

```
procedure TForm1.Button2Click(Sender: TObject);
Var
sss, sss1:String;
N, I, j: Integer;
v, vk: OleVariant;
begin
form1.Left:=10; form1.Top:=10;
memo1.Clear;
WordApp.Options.CheckSpellingAsYouType:=False;
```

```
WordApp.Options.CheckGrammarAsYouType:=False;
label4.Caption:=(max='+inttostr(trunc(col/7)));
ProgressBar3.Max:=trunc(col/7);

// запоминание символа Enter
WordApp.Selection.HomeKey(wdStory,EmptyParam);
WordApp.ActiveDocument.Range.Insertbefore(#13#10);

v:=WordApp.ActiveDocument.Paragraphs.Item(1).Range.start;//.select;
vk:=WordApp.ActiveDocument.Paragraphs.Item(1).Range.end;//.select;
WordApp.ActiveDocument.Range(v,vk).Select;
Enter:=WordApp.selection.text;
WordApp.selection.delete;

// Проверка на наличие двух знаков Enter и удаление второго
E2b:= true;
col1:=1;
while E2b do
BEGIN
col2:=WordApp.ActiveDocument.Paragraphs.Count;
E2;
application.ProcessMessages;
END;

// Проверка на наличие знака Enter между строк и его удаление,
исключая последнюю строку в каждом вопросе.
E2b:= true; col1:=1;
while E2b do
BEGIN
col2:=WordApp.ActiveDocument.Paragraphs.Count;
E3;
application.ProcessMessages;
END;

ProgressBar3.Position:=0;
end;
```

Здесь вводятся дополнительно две подпрограммы **E2** и **E3**, которые как раз и обрабатывают команды для сервера, находят и исправляют отмеченные ошибки. Рассмотрим их более подробно:

**code:**

```
procedure TForm1.E2; // два Enter
```

```
Var
sss, sss1:String;
I: Integer;
  v, vk: OleVariant;
begin
for i := col1 to col2-1 do
BEGIN
v:=WordApp.ActiveDocument.Paragraphs.Item(i).Range.start;
vk:=WordApp.ActiveDocument.Paragraphs.Item(i).Range.end;
WordApp.ActiveDocument.Range(v,vk).Select;
sss:=WordApp.selection.text;

v:=WordApp.ActiveDocument.Paragraphs.Item(i+1).Range.start;
vk:=WordApp.ActiveDocument.Paragraphs.Item(i+1).Range.end;
WordApp.ActiveDocument.Range(v,vk).Select;
sss1:=WordApp.selection.text;

  if sss=sss1 then
begin
WordApp.selection.delete;
col1:= i-3;
exit;
end;

  application.ProcessMessages;
if i=col2-1 then  E2b:=false;
END;
end;

procedure TForm1.E3; // удаляем лишний Enter в строке
Var
sss, sss1:String;
I: Integer;
  v,vk: OleVariant;
begin
for i := col1 to col2 do
BEGIN
v:=WordApp.ActiveDocument.Paragraphs.Item(i).Range.start;
vk:=WordApp.ActiveDocument.Paragraphs.Item(i).Range.end;
WordApp.ActiveDocument.Range(v,vk).Select;
sss:=WordApp.selection.text;

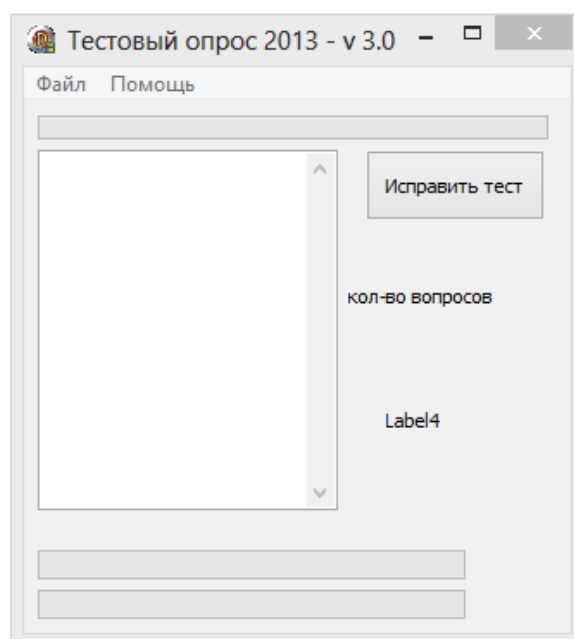
  if sss=Enter then
```

```
if i mod 7 <>0 then
  begin

    WordApp.selection.delete;
coll:= i-3;
exit;
end;
application.ProcessMessages;

if i=col2 then  E2b:=false;
END;
end;
```

Получим следующую программу (рис.3) работающую на основе OLE технологии и реализующую компонентную модель. Загрузка данных из файла производится посредством Drag&Drop.



**Рисунок 3 – Программа.**

Проверим ее функциональность на тесте (рис. 4-5), как видим после автоматического исправления – тест соответствует требованиям. Разработанная программа исправляет пропущенные переносы на другую строку, а также удаляет лишние.

1. Квадратная матрица называется диагональной, если.....  
A) все элементы вне главной диагонали равны нулю  
B) все элементы побочной диагонали равны нулю  
C) все элементы главной диагонали равны нулю  
D) все элементы матрицы равны нулю  
E) все элементы вне главной диагонали равны единице

2. Прямоугольная матрица  $A$  размеров  $m \times n$  называется квадратной, если.....  
A)  $m = n$   
B)  $m \neq n$   
C)  $m < n$   
D)  $m > n$   
E)  $m \leq n$

3. При умножении матрицы на число умножаются.....  
A) все элементы матрицы

Рисунок 4 – Тест до исправления.

1. Квадратная матрица называется диагональной, если.....  
A) все элементы вне главной диагонали равны нулю  
B) все элементы побочной диагонали равны нулю  
C) все элементы главной диагонали равны нулю  
D) все элементы матрицы равны нулю  
E) все элементы вне главной диагонали равны единице

2. Прямоугольная матрица  $A$  размеров  $m \times n$  называется квадратной, если.....  
A)  $m = n$   
B)  $m \neq n$   
C)  $m < n$   
D)  $m > n$   
E)  $m \leq n$

3. При умножении матрицы на число умножаются.....  
A) все элементы матрицы  
B) все элементы какого-нибудь столбца

Рисунок 5 – Тест после автоматического исправления.