Лабораторнаяработа №1, МАТРИЦА

Постановказадачи

Создайте программу, которая взависимости отвеличин N (количествострок)и M(количества столбцов) создает матрицуразмеромNxM. Программапредоставляетвозможность заполнить матрицуспомощью случайных чиселили ввести значениявручную. Кроме этого можно подсчитатьсуммуэлементов матрицы, определитьмаксимальный и минимальный элементы матрицы.

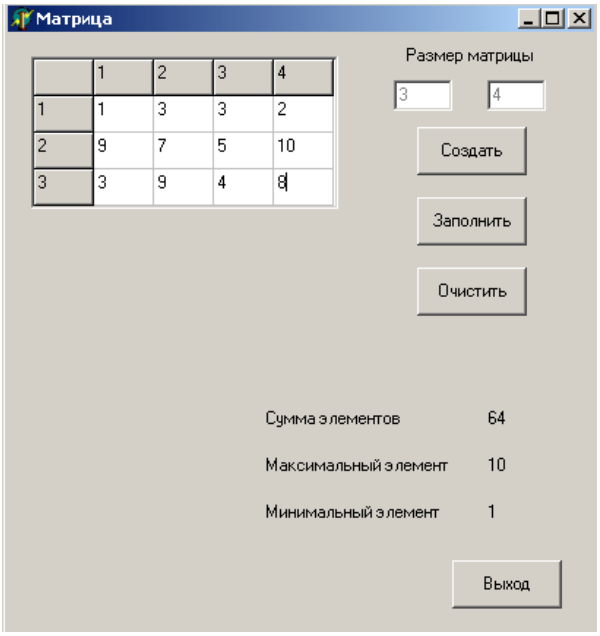


Рис.1

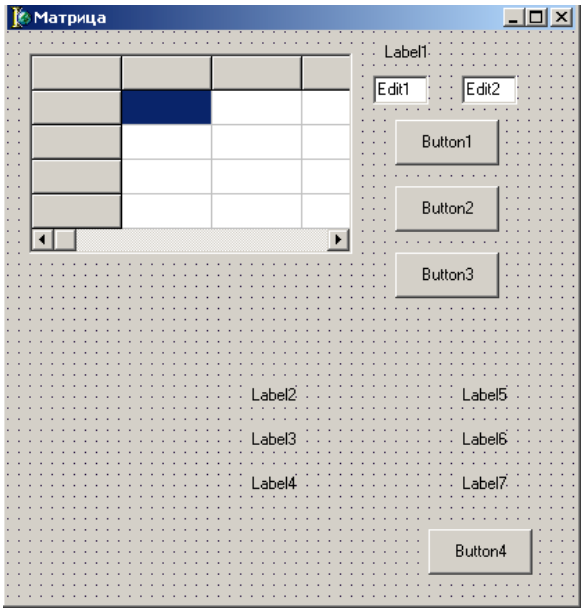


Рис.2

Новымв этой работеявляются:

- компоненттаблицастрок**StringGrid**вкладкипалитрыкомпонентов**Additional**.

Планразработки программы

1. Откройтеновыйпроект.
2. Разместите вформеобъекты всоответствиис рис.2.
3. Установитесвойства компонентнавкладке **ObjectInspector**:

Выделенный объект	Имясвойства	Значение
Label1	Caption	Размерматрицы
Edit1	Text	Удалитьназваниеобъекта
Edit2	Text	Удалитьназваниеобъекта
Button1	Caption	Создайте
Button2	Caption	Заполнить
Button3	Caption	Удалитьназваниеобъекта
Label2	Caption	Суммаэлементов
Label3	Caption	Максимальныйэлемент

Label4	Caption	Минимальныйэлемент
Label5	Caption	Удалитьназваниеобъекта
Label6	Caption	Удалитьназваниеобъекта
Label7	Caption	Удалитьназваниеобъекта
Button4	Caption	Выход
StringGrid1	Name	MATR

4. Сохранитекод программыи проектпод именами, например, **Unit_M.pas**и**Pr_M.dpr**.

Немноготеории

Компонент**StringGrid**(вкладка палитрыкомпонентов**Additional**) предназначендлясоздания таблицы.

Свойствукомпонента **Cells**соответствуетдвухмерный массивячеек, каждаяизкоторыхможет содержать произвольныйтекст. Содержание ячейки массива,находящегося напересечении столбца сномером **Col**и строки с номером**Row**определяется элементом**StringGrid1.Cells[Col,Row]**.Этосодержимоеможноредактировать.

Двумерный массив состоит из двух частей: фиксированной и рабочей.

Фиксированная часть служит дляпоказа заголовков столбцов (строк) и для ручногоуправленияих размерами.Обычно фиксированнаячастьзанимает крайний левый столбеци самуюверхнююстрокутаблицы. Спомощьюсвойств **FixedCols**и**FixedRows**можнозадатьдругоеколичество фиксированныхстолбцови строк. Еслисвойства **FixedCols**и**FixedRows**имеютзначение 0, то таблица несодержит фиксированной зоны.

Рабочаячасть – это остальная часть таблицы.Она можетсодержать произвольноеколичество столбцови строк, которое можно изменять входе выполненияпрограммы.Еслирабочаячасть не умещаетсяцеликом впределах окнакомпонента,то автоматическипоявляются полосы прокрутки.Припрокрутке рабочейобласти фиксированная область неисчезает.

ColCount	Количествостолбцовтаблицы
RowCount	Количествостроктаблицы
FixedCols	Количествостолбцовфиксированнойзоны
FixedRows	Количествострокфиксированнойзоны
DefaultRowHeight	Высотастрокитаблицы
Height	Высотавсейтаблицы
DefaultColWidth	Ширинастолбцатаблицы
Width	Ширинавсейтаблицы
Options.goEditing	Признак редактирования содержимого ячеектаблицы.True– редактированиеразрешено,False–
GridLineWidth	Ширина линий,ограничивающих ячейки таблицы
Font	Шрифт,используемыйдляотображения содержимого ячеектаблицы
Options .goEditing	Признак допустимости редактирования содержимогоячеектаблицы.True–редактирование
Options .goTabs	Разрешает(True)или запрещает(False)использование клавиши «Tab» для перемещения курсора вследующую ячейкутаблицы

5. Разместитевблокереализациипослеслова**implementation**описание переменных:

```
Var
    N,M:Integer;       //N-количествострок,M-количествостолбцов
    MATR_MAX,          //Значениемаксимальногоэлементатаблицы
    MATR_MIN,          //Значениеминимальногоэлементатаблицы
    MATR_SUM:Integer;//Значениесуммыэлементовтаблицы
```

6. Основная задача проекта – создать таблицу,размер которой будет определен послезаполненияполей**Edit1**и**Edit2**.

Создадим следующиепроцедурыобработки событий:

Выделенный объект	Имяс обытие	Текстпроцедуры
Button4 «Выход»	OnClick	procedureTForm1.Button4Click(Sender:T Object); beginC lose; end;

Form1	OnCreat	procedureTForm1.FormCreate(Sender:TObject);MA TR.Visible:=False; Button2.Enabled:=False;B utton3.Enabled:=False;
		end;
		Комментарий
		Присозданииформыустановимкомпонент**StringGrid**нев идимым (новое имяэтогокомпонента**MATR**),т.к. вначале неизвестно сколько жестрок и столбцоввтаблице. Кроме этогодо определения размера таблицыустановим недоступными кнопки«Заполнить» и «Очистить»

Button1 «Создать таблицу»	OnClick	procedureTForm1.Button1Click(Sender:T Object); VarI,J:Byte;b egin If(Edit1.Text<>’’)and(Edit2.Text<>’’)Thenbegi n Edit1.Enabled:=False;Edi t2.Enabled:=False;Button 1.Enabled:=False;Button2 .Enabled:=True;Button3.E nabled:=True; N:=StrToInt(Edit1.Text);M: =StrToInt(Edit2.Text); MATR.DefaultColWidth:=40;M ATR.RowCount:=N+1;MATR.Col Count:=M+1; MATR.Height:=(MATR.DefaultRowHeight+2)*(N+1);MA TR.Width:=(MATR.DefaultColWidth+2)*(M+1);MATR .Visible:=True; ForI:=1toNdoMATR.Cells[0,I]:=I ntToStr(I); ForJ:=1 toMdoMATR.Cells[J,0]

		end; end; **Комментарий** Данная процедурабудет выполнятьсятолькоприусловии,чтовведеныразмерыматрицы,т.е**Edit1.Text**и**Edit2.Text**неявляются«пустым символом». Вначалеустанавливаются недоступными компоненты ввода,определяющие размертаблицы, и кнопка«Создать», анедоступные ранее кнопки «Заполнить» и «Очистить»становятсядоступными. После этогопереходимк формированию таблицы.Для этогоустанавливаемзначенияпеременныхNи M, определяющих количествострок истолбцовтаблицы– переводим из текстовой информациизначенияполей **Edit1**и**Edit2**вчисловые значения.Послеэтогопрограммнозадаемсвойствакомпонента **StringGrid**(**MATR**) и устанавливаем его видимым. В

Button2 «Заполнить таблицу»	OnClick	procedureTForm1.Button2Click(Sender:T Object);
		VarI,J:Byte;b eginRandomize ;
		ForI:=1toNdoForJ: =1toMdo
		MATR.Cells[J,I]:=IntToStr(Random(N*M));
		
		MATR_MAX:=StrToInt(MATR.Cells[1,1]);M ATR_MIN:=StrToInt(MATR.Cells[1,1]);MA TR_SUM:=0;
		ForI:=1toNdoForJ: =1toMdo
		beginMATR_SUM:=MATR_SUM+StrToInt(MATR.Cells[J, I]);
		IFStrToInt(MATR.Cells[J,I])>MATR_MAXThenMA TR_MAX:=StrToInt(MATR.Cells[J,I]);
		IFStrToInt(MATR.Cells[J,I])<MATR_MINThenMA TR_MIN:=StrToInt(MATR.Cells[J,I]);
		end;Label5.Caption:=IntToStr(MATR_ SUM);Label6.Caption:=IntToStr(MATR_M AX);Label7.Caption:=IntToStr(MATR_MI N);end;
		Комментарий
		Вначале процедуры заполняются ячейки таблицы(MATR.Cells[J,I]) случайными числами в диапазоне от0до(N*M-1). Дальше определяются максимальное иминимальное числа исуммаэлементов таблицы. Вконцепроцедуры полученныезначениявыводятсянаэкран.Для этогоиспользуются компоненты**Label5**,**Label6**,**Label7**.
		Для заполнения ячеектаблицыкакположительными,так и отрицательными

		числами,например, вдиапазоне от-10 до10можнозаписать:`MATR.Cells[J,I]:=IntToStr(Random(11))-10;`

Button3 «Очистить таблицу»		`procedureTForm1.Button3Click(Sender:TObject);`
		`VarI,J:Byte;b`
		`egin`
		`ForI:=1toNdoForJ:`
		`=1toMdo`
		`MATR.Cells[J,I]:='';`
		`Label5.Caption:='';L`
		`abel6.Caption:='';La`
		`bel7.Caption:='';Edi`
		`t1.Text:='';`
		`Edit2.Text:='';`
		
		`Matr.Visible:=False;But`
		`ton2.Enabled:=False;Butt`
		`on3.Enabled:=False;Butto`
		`n1.Enabled:=True;;Edit1`
		`.Enabled:=True;Edit2.En`
		`abled:=True;end;`
		Комментарий
		Всеячейкитаблицызаполняетсяпустойстрокой, а затемочищаютсякомпоненты**Label5**,**Label6**,**Label7**,**Edit1.Text**,**Edit2.Text**.Компонент**StringGrid**(**MATR**)устанавливаетсяневидимым, кнопки«Заполнить» и «Очистить»становятсянедоступными,кнопка«Создать» икомпоненты**Edit1**и**Edit2**–доступными.

7. Программа готова.Но есливполя ввода «Размерность матрицы» мы по ошибкевместо числа введем буквуиликакой-либо другой символ,то программабудетпрервана.Длятого чтобы это избежатьсоздадим процедуры обработки события**KeyPress**длякомпонента**Edit1**.

```
procedureTForm1.Edit1KeyPress(Sender:TObject;varKey:Char);beg
in
caseKeyof'0'.
  .'9':;
  #8:       ;
  #13:Edit2.SetFocus;e
  lseKey:=Chr(0);
end;
```

Комментарий

Взависимости от нажатой клавиши программа выполнит следующие действия:

- нажаталюбаяцифроваяклавиша или клавиша«BackSpace» (код клавиши #8)– программаничего неизменит,
- нажата клавиша «Enter» (кодклавиши#13)– курсорперейдетв окно редактораввода**Edit2**,
- во всехостальных случаях – введенный символ будетудален (присвоениезначения пустогосимволаChr(0)).

Для компонента **Edit2**самостоятельносоздайтепроцедуру, обрабатывающуюввод.Она будетотличатьсятем, что принажатииклавиши «Enter», курсордолженперейтинакнопку«Создать таблицу»(**Button1**).

8. Созданнаяпрограмма не позволяет редактировать элементы таблицы. Внесемизменениявсвойства компонента**StringGrid**(**MATR**). Дляэтогонавкладке**ObjectInspector**свойству**Options.goEditing**установим значение **True**.Процедура,обрабатывающаянажатие клавишивполекомпонента **StringGrid**(**MATR**)будетвыглядетьтак:

```
procedureTForm1.MATRKeyPress(Sender:TObject;varKey:Char);VarI
,J:Byte;
begin
 caseKeyof'0'.
  .'9':       ;
  #8:         ;
  #13:ifMATR.Col<MATR.ColCount-
  1thenMATR.Col:=MATR.Col+1;elseKey:=Chr(0);
 end;
MATR_MAX:=StrToInt(MATR.Cells[1,1]);M
ATR_MIN:=StrToInt(MATR.Cells[1,1]);MA
TR_SUM:=0;
ForI:=1toNdoForJ:
 =1toMdo
  beginMATR_SUM:=MATR_SUM+StrToInt(MATR.Cells[J,I
  ]);
  IFStrToInt(MATR.Cells[J,I])>MATR_MAXThenMA
TR_MAX:=StrToInt(MATR.Cells[J,I]);
  IFStrToInt(MATR.Cells[J,I])<MATR_MINThenMA
TR_MIN:=StrToInt(MATR.Cells[J,I]);
  end;Label5.Caption:=IntToStr(MATR_
SUM);Label6.Caption:=IntToStr(MATR_M
AX);Label7.Caption:=IntToStr(MATR_MI
N);

end;
```

Комментарий

Обработканажатия допустимой клавиши рассмотрена вп.7. Отличиезаключается вдействии, которое произойдет, если нажата клавиша «Enter» - переход к следующейячейке в даннойстроке, если эта ячейка непоследняя в строке.

Дальше в процедуреидет подсчетмаксимального, минимального элемента и суммыэлементоввтаблице, после внесенияизменений.

9. Сохранитепроект окончательно,запустите и протестируйте его.

Заданиедлясамостоятельного выполнения

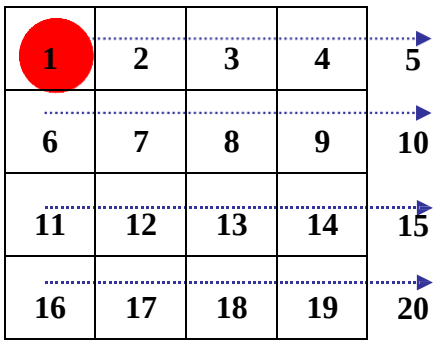
1. Дополните программу, вставив блокопределения суммы по каждомустолбцуматрицы.

Подсказка.Необходима ещеоднатаблица (компонент**StringGrid**),вкоторойбудутнаходиться подсчитанные суммы по столбцам.Формировать этутаблицунужно вмомент формирования основнойтаблицы.

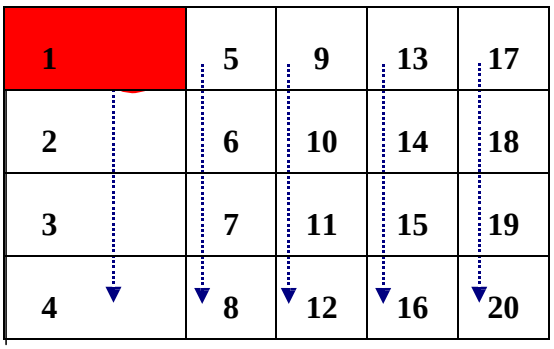
Продолжение работы

Доработатьпрограмму так, чтобыможнобылобызаполнятьматрицуразличнымиспособами:

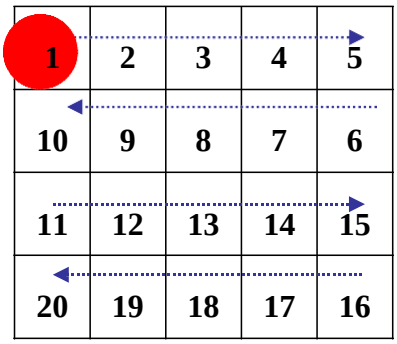
1 способ



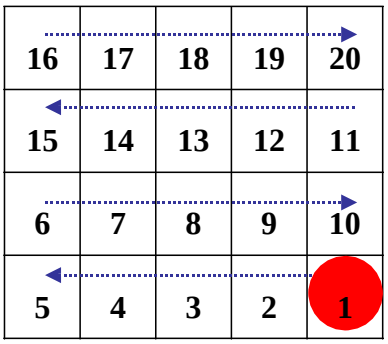
2 способ



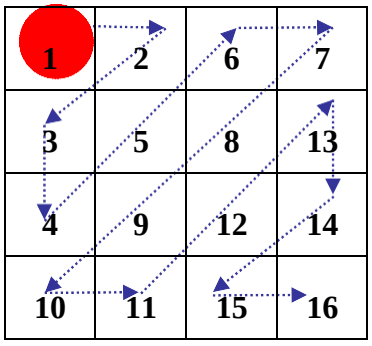
3 способ



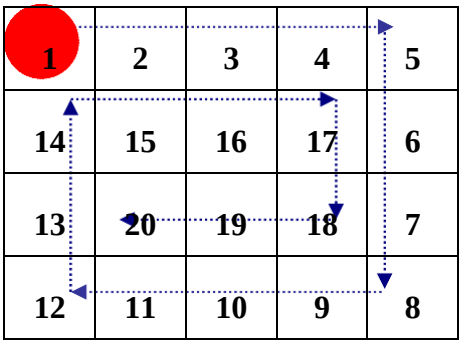
4 способ



5 способ



6 способ



Подсказка.

Удалите кнопку*Заполнитьтаблицу*, добавив отдельныепроцедурыдля заполнениякаждым способом.

Вначаленеобходимо внести изменения враздел объявления:

```
ConstN=15;M=15;       //N-количествострок,M-количествостолбцов
TypeT_Mas=Array[1..N,1..M]OfInteger;//Типмассив
Var
     Mas   :T_Mas;
     MATR_MAX,
                        //ЗначениемаксимальногоэлементатаблицыMAT
     R_MIN,
```

Примерпроцедуры для заполнения с помощью случайныхчилелможет выглядеть так:

```
ProcedureZ0(VarMAS:T_MAS);VarI
,J:Byte;
beginRandom
 ize;
 ForI:=1toNdoForJ:
  =1toMdo
   MAS[J,I]:=Random(N*M);
End.
```

Созданные вами процедуры должны располагатьсяв тексте программы сразуже послераздела объявленияпеременных.

Длявыбора способа заполнения можновоспользоваться компонентой**ListBox.**

Длязаполнения выделите объект **ListBox1**,найдите свойство **Items**,щелкните накнопке с тремя точками,расположенной справа отнего. Впоявившемся окневстроенногоредактора**StringListEditor**введитеназванияспособовзаполнения,каждый нановой строке. Например:

```
Случайными
числамиПо
горизонтали
ПовертикалиЗм
ейкой
слеваЗмейкой
справаЗигзагом
```

Комментарий

а) Свойство**Items**содержит элементы списка.

б) Список может бытьсоздан присоздании формы или во время работыпрограммы.

в)Свойство**ItemIndex**определяетномер элемента,выбранногоиз списка.Первый элементимеетномер0. Если невыбран ни один из элементов,тозначение свойства**ItemIndex**равно – «-1».

Сохраните набранный текст вфайле под именем ZAPOLN.txt. Дляэтого нажмитеправую кнопкумышии выберитережим **Save**.Длявыходаиз встроенного редакторащелкнитена кнопке **OK**.

Комментарий

Просмотретьсодержимое созданноготекстового файлаZAPOLN.txt,можноспомощью любоготекстового редактора, а такжевнести изменения в тестовыйфайл,не используявстроенный редакторDelphi.

Выполните следующиедействия:

Выделенный объект	Вкладкаокна**Object Inspector**	Имясвойства/ Имясобытия	Значение/Действие
ListBox1	Events	OnKeyPress	ifkey=#13then CaseListbox1.ItemIndexof 0:Z0(MAS); 1:Z1(MAS); 2:Z2(MAS); 3:Z3(MAS); 4:Z4(MAS); 5:Z5(MAS); 6:Z6(MAS); end; ForI:=1toNdoForJ: =1toMdo MATR.Cells[J,I]:=IntToStr(MAS[I,J]);

Лабораторнаяработа№2,Угадайслово

Постановказадачи

Разработать программу, которая реализует логическую игру«Угадай слово», известнуюкаждомус детства.

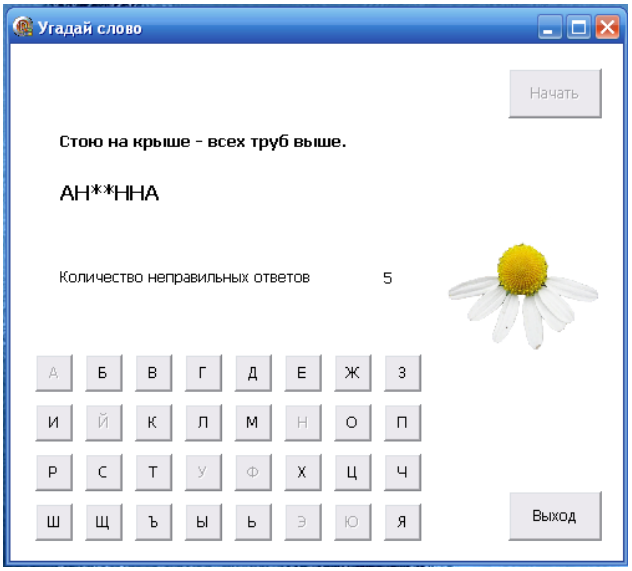


Рис.1

Правила игры

Ведущийзагадывает слово, которое надоугадать, и записываетчерточки, количествокоторых равно количествубукввзагаданномслове.

(Для подсказки можно использовать вопрос,ответомна который является загаданноеслово.) Игрокупредоставляется право выбора букв по одной,при этом он можетдопустить толькодесять ошибок (попыток).

Игрок называетбукву,которая на его взгляд может присутствоватьвслове. Есливыбранная игрокомбуква естьвзагаданномслове, товедущийзаписывает ее вместочерточки на своем месте. Если буква встречаетсявслове несколько раз,то ведущий

«открывает»всеэтибуквы. Еслибуква названная игреком отсутствует вслове, тоуменьшаетсяколичество предоставленных попыток.

Игразаканчивается,если словоугадано, илиесли исчерпаны всепредоставленныепопытки.

Новым в этой работеявляются:

- создание компонент во время выполнения программыи обработкаих событий,
- вывод иллюстраций.

Информационная постановказадачи

1. Информацияовопросахи ответах хранитсявтекстовом файле(Questions.txt).Накаждый вопрос-ответ отводится две строки: в первой строке - вопрос, а вовторой-ответ.
2. Информацияиз текстовых файловвначале программысчитывается в массив (AQ), из которогозатем случайнымобразомвыбираются текстывопросов.
3. Выбранныйвопросвыводится наэкран, а ответ записываетсявпеременную STR_R.
4. Формируетсяугадываемое слово ввиде черточек (переменная STR_N).Количествочерточек соответствует количествубукв (например,STR_N='------').

5. Длявыбораотгадываемых букв словаиспользуютсякнопки,которые создаютсявходе программы.
6. При нажатии накнопкус буквой,которая естьвслове, этабуквапоявляется нанужном месте,т.е.меняется переменнаяSTR_N (например, STR_N='-A-A--').Еслитакойбуквынет,то уменьшаетсяколичество попыток(переменнаяATTEMPT).Кнопка, содержащая нажатую букву, в дальнейшемонастановится недоступной.
7. Приугадываниислова (STR_R=STR_N) игрокупредоставляетсявозможностьначать новую игруиливыйтииз игры.
8. Прииспользованиивсехпопыток(ATTEMPT=0)наэкранепоявляетсянеугаданноеслово и предоставляется возможность начать новую игруили выйтииз игры.

Планразработки

программыПостроение

программыбудемпоэтапно.**Первый этап.**

Формированиеформыэкрана

1. Откройтеновыйпроект.

2. Разместите вформеэкземплярыкомпонентовв соответствии с рис.2.

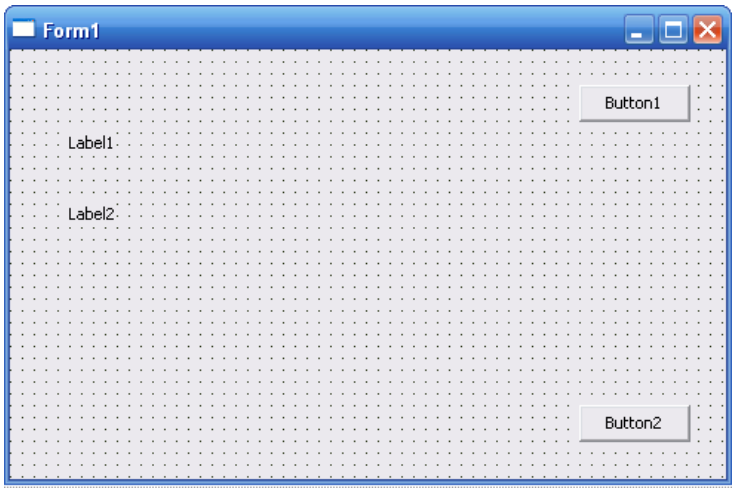


Рис.2

3. Выделите объект**Form1**, перейдите на вкладку**EventsИнспектора объектов**(**ObjectInspector**),найдитесобытие**OnCreat**,справаотнегодваждыщелкнутьле войкнопкоймыши. Попав в кодпрограммы,надо написатьследующий текст:

```
procedureTForm1.FormCreate(Sender:TObject);begin
//Формированиеэлементовформы
  WithForm1dobeg
  in
   Caption:='Угадайслово';Hei
   ght:=450;
   Width:=500;Color
   :=clWhite;
  end;
  WithButton1dobeg
  in
    Caption:='Начать';He
    ight:=40;Width:=75;
    Top:=20;Left:=400;F
    ont.Size:=9;
```

```
  end;
  WithButton2dobeg
  in
    Caption:='Выход';He
    ight:=40;Width:=75
    ;Top:=360;Left:=40
    0;Font.Size:=9;
  end;
  WithLabel1dobeg
  in
    Caption:='';H
    eight:=40;Wid
    th:=250;Top:=
    70;
    Left:=40;Font.Style:=[f
    sBold];Font.Size:=10;
  end;
  WithLabel2dobeg
  in
    Caption:='';H
    eight:=40;Wid
    th:=75;Top:=1
    10;
    Left:=40;Font.Style:=[f
    sBold];Font.Size:=12;
  end;
 Read_File;//Процедурачтенияинформацииизфайла
end;
```

Пояснение

Длякомпонент**Form1**,**Button1**,**Button2**,**Label1**,**Label2**втекстепрограммыопределенывсенеобходимыесвойства(размеры, местоположение и т.п.),поэтомуприразмещении компонентнетнеобходимости устанавливать свойствакомпонентспомощьювкладки**Properties**(Свойства)инспектораобъектов.

4. Разместитевблокереализациипослеслова**implementation**описание константипеременных:

```
Const
 N=100;    //Максимальноеколичествозаписейвмассиве
TYPE
 T_R=record         //Структуразаписимассива
        Que:string[250];A
        ns:string[30];
       end;R=array[1..
 N]ofT_R;


VarAQ:R;                        //МассивВопросовиОтветов
    Quetions_F:TextFile;        //Файловаяпеременная
    STR_R,STR_N:String[30];//Отгадываемоеслово
    KOL_QUE:Integer;            //Количествовопросоввфайле
```

5. Разместите после блока описания переменныхпроцедуручтенияинформацииизфайла и формирование массива вопросов **AQ**. К этой процедурепроисходит обращениевконцепроцедуры**TForm1.FormCreate**(см.п.3).

```
procedureRead_File;
//Чтениеинформацииизфайлаизаписьвмассив
VarKOL,I:Integer;beg
in
 Assignfile(Quetions_F,'Quetions.txt');Reset
 (Quetions_F);
 KOL:=1;I:=1;
 WhilenotEof(Quetions_F)dobegin
   if(KOLmod2)=1thenReadln(Quetions_F,AQ[I].QUE)
                      elseb
                      egin
                       Readln(Quetions_F,AQ[I].ANS);Inc(
                       I);
                      end;
   Inc(KOL);
  end;KOL_QUE
 :=I;
 closefile(Quetions_F);end
;
```

6. Создадим процедуру,которая обрабатывает ситуацию нажатия кнопки**Button1**Начатьигру).Дляэтого выделите объект**Button1**, перейдите на**вкладку****Events****Инспектораобъектов**(**ObjectInspector**), найдитесобытие**OnClick**,справа отнегодваждыщелкнутьлевой кнопкой мыши. Попав вкод программы, надонаписать следующий текст:

```
procedureTForm1.Button1Click(Sender:TObject);VarI,N
UMBER:integer;
beginRandomi
 ze;
 NUMBER:=Random(KOL_QUE-1)+1;
 Label1.Caption:=AQ[NUMBER].QUE;STR_
 R:=AQ[NUMBER].ANS;STR_N:='';
 forI:=1toLength(STR_R)doSTR_N:=ST
 R_N+'*';
 Label2.Caption:=STR_N;Form1.Butt
 on1.Enabled:=False;
end;
```

Пояснение

Начало игры означает, что нужно выбрать вопрос, который затем выводитьсяна экран(**Label1**),исформироватьзашифрованноеслово (**Label2**,**STR_N**),котороебудетотгадыватьигрок.Выборвопросаиз массивапроизводитсяс помощью функцииRandom.

7. Самостоятельно добавьте событие обработки кнопки**Button2** (Выход).

8. Создайтетекстовыйфайл **Quetions.txt**, записав его втужепапку,где находится ипрограмма.Текст файла может выглядеть так:

Стою накрыше - всехтруб

выше.АНТЕННА

Всех насветеобшивает, а сама не

надевает.ИГОЛКА

Твой хвостикя врукедержал, ты полетел, я

побежал.ШАРИК

Золотоерешето,черных
домиковполно.ПОДСОЛНУХ
Ах,не трогайте меня.Обожгуи без
огня!КРАПИВА

9. Сохранитепроект,запустите ипротестируйте его. После запускапрограммынаэкране видны двекнопки - **Начать**и **Выход**. Если нажать кнопку**Начать**, топоявляется текст вопроса и зашифрованное слово ответа,при этомкнопка**Начать**становитсянедоступной.Проверьте, чтобы при каждом новомзапуске программыувасвсегда выбирался новыйтекст ответа.

Второйэтап. Создание компонентво времявыполненияпрограммы иобработка ихсобытий

Для дальнейшейработойнад проектом нам необходимосоздать33 кнопки, которыебудут принимать значения букв русскогоалфавита, и обрабатывать ситуацию поискавыбраннойбуквы (нажатой кнопки).Эта задача несложная,но оченьутомительная- 33раза повторить одни и теже операции. Какупростить этотпроцесс покажем наотдельном примере.

Вданном примере понажатию кнопки**Button1**создаются32 кнопки,которыерасполагаются в три ряда. Свойства**Caption**этихкнопок принимаютзначения букврусскогоалфавита от «А» до «Я» (кроме буквы «Ё»).Понажатию каждой кнопки идетобращениекпроцедуре**TForm1.BtnClick**,котораяопределяеткакаябукварусскогоалфавита соответствуетнажатой кнопке. Дальше идетобращение кпроцедуре**Poisk**, инажатая кнопкастановится недоступной.

Поцедура**Poisk**ищетвстречаетсяли выбранная буква(нажатая кнопка) взаданномслове **STR_N**, и еслида, то заменяется в**STR_R**соответствующий символ назаданнуюбукву.

1. Откройтеновыйпроект.
2. Разместитевформеэкземплярыкомпонентов**Label1**и**Button1**.
3. Длясобытия**OnClick**компоненты**Button1**напишите такойтекст:

```
procedureTForm1.Button1Click(Sender:TObject);var
 i:integer;be
gin
fori:=1toNdo                    //Циклпосозданиюкнопокнаформе
begin
  Btn:=TButton.Create(Form1);with
  Btndo
  begin            //Определениесвойствсоздаваемыхкнопок
    Parent:=Form1;//Определениеродителя,
                        //т.е.гдесоздаютсякнопки
    Caption:=Chr(191+I);//Определениебуквыпоеекоду
    Height:=30;
    Width:=30;
    Top:=200+((I-1)div11)*40;
    Left:=((I-1)mod11)*40+30;
    Font.Style:=[fsBold];
    OnClick:=FORM1.BtnClick;        //Имяпроцедуры,
                          //обрабатывающейсобытиенажатиякнопки
  end;
```

```
 end;
  STR_R:='РОССИЯ';//Дляпримеранашеотгадываемоеслово
  STR_N:='';
  ForI:=1toLength(STR_R)doSTR_N:
  =STR_N+'*';
  Label1.Caption:=STR_N;Form1.Butt
  on1.Enabled:=False;
 end;
```

4. Процедуру**TForm1.BtnClick**(обработкасобытия нажатия созданных кнопок) мысоздаемсами.Дляэтоговставьтетекст этойпроцедурывобщий текст программыпередтем, как осуществляется обращение кпроцедуре **TForm1.BtnClick**.

```
procedureTForm1.BtnClick(Sender:TObject);
//Процедураобработкисобытиянажатиясозданныхкнопок
VarSTR_:String;be
gin
STR_:=(SenderasTButton).Caption;
//ПеременнаяSenderсодержитимя
//объекта,которомусоответствуетданноесобытие
CHAR_:=STR_[1];
POISK(STR_R,STR_N,CHAR_);//Обращениепроцедурепоискабуквы(SenderasTB
utton).Enabled:=False;//Кнопкабуквынедоступнаend;
```

5. Вразделописания процедур(перед**private**)добавьтестрокуописаниязаголовкапроцедуры:

```
procedureBtnClick(Sender:TObject);
```

6. Длятого,чтобынаша программа заработаланеобходимо внести объявлениепеременныхиописаниепроцедуры**Poisk**.

```
 ConstN=32;                  //Количествобукв-кнопок
 TypeSTR_30=String[30];Var
 Btn:TButton;                //Переменнойдлясозданиякнопок
     STR_N,STR_R:STR         //Зашифрованноеслово
     _30;,                   //Отгадываемоеслово
     CHAR_:Char;             //Нажатаябуква
   {$R*.dfm}


ProcedurePOISK(STR_R:STR_30;VarSTR_N:STR_30;CHAR_:Char);
//Поискбуквывсловеи«открытие»ее
VarI:Integer;
       Flag:Boolean;        //Флагнайденаливсловенажатаябуква
  beginFlag:=Fa
  lse;
  ForI:=1toLength(STR_R)do          //Поискбуквывслове
   IfSTR_R[I]=CHAR_thenbegi
     n
      STR_N[I]:=CHAR_;
      Flag:=True;e
     nd;
  IfFlag=TrueForm1.Label1.Caption:=STR_N;

  end;
```

7. Сохранитепроект,запустите ипротестируйте его. После запускапрограммынаэкране виднатолько одна кнопка**Button1**и закодированное слово ввиде звездочек.После нажатиянаэтукнопкунаэкране появляется три ряда кнопок с буквами(см.рис.3) икнопка**Button1**становится недоступной.

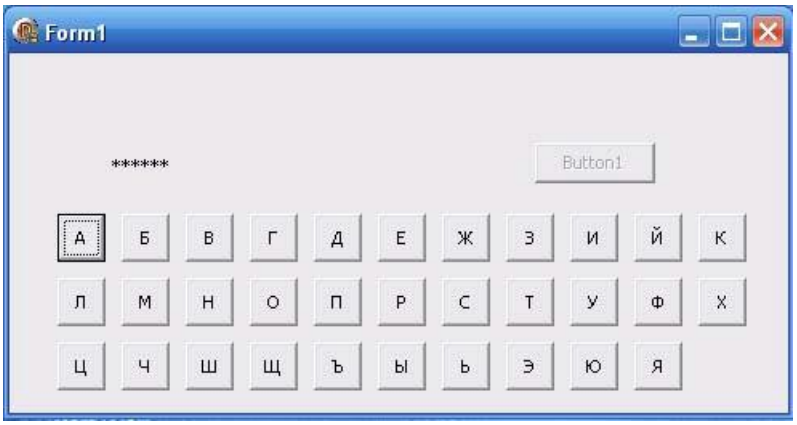


Рис.3

Намизвестно слово, которое зашифровано - это слово «РОССИЯ».Протестируйтепрограмму,введя буквы этого слова, и проверьте, чтобы ониувас«открылись»взашифрованном слове.

Третий этап. Создание кнопокперед началом игры

Выполнитеэтот этап самостоятельно, внесяизменения в основнойтекст программы,созданный навторомэтапе.При создании кнопок расположите их не в три ряда, а вчетыре(см.рис.4).Подумайте, как этосделать.

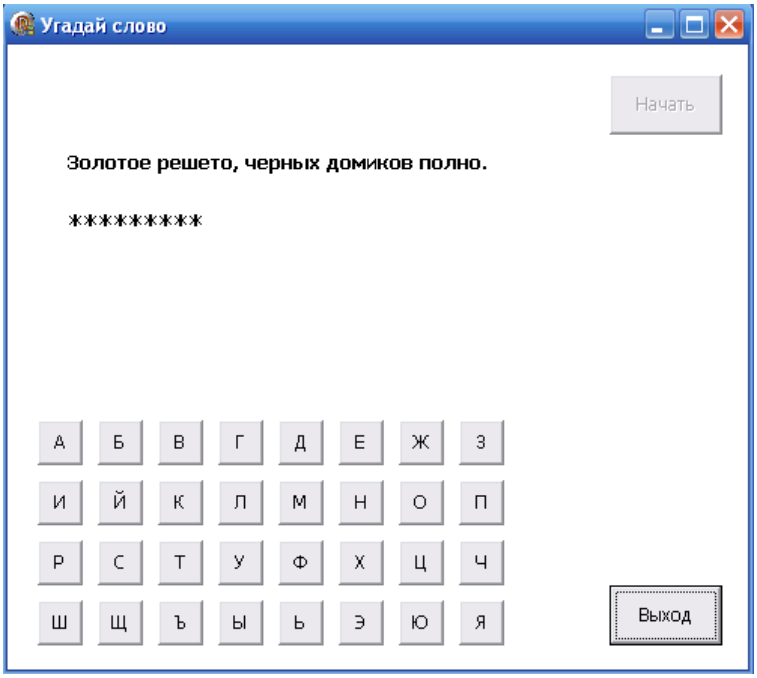


Рис.4

Четвертыйэтап.Анализ состояния игры

Необходимо внести изменения впрограмму,дляопределения закончена ли игра,т.е.угаданы всебуквы взашифрованном слове.Вместе с тем установимколичестводопустимых ошибок игрока.

1. Разместитевблоке реализации описаниедополнительныхконстант и переменных:

```
Const
  Attempt=10;//Максимальноеколичествопопыток
```

```
VarN_Attempt:Integer;      //Количествоошибок
```

2. Разместитевформеэкземплярыкомпонентов**Label3,Label4**(сообщениеоколичестве допущенных ошибок) и**Panel1**(сообщение озавершении игры).

3. Дополнитетекстпроцедуры**TForm1.FormCreate**описаниемновыхкомпонент:

```
With Label3 dobeg
  in
    Caption:='';H
    eight:=40;Wid
    th:=75;Top:=1
    80;
    Left:=40;Font.Si
    ze:=9;
  end;
  With Label4 dobeg
  in
    Caption:='';H
    eight:=40;Wid
    th:=75;Top:=1
    80;Left:=300;
    Font.Style:=[fsBold];Fon
    t.Size:=10;
  end;
  With Panel1 dobeg
  in
    Caption:='';H
    eight:=45;Wid
    th:=185;Top:=
    15;Left:=110;
    Font.Style:=[fsBold];Fon
    t.Size:=10;Visible:=Fal
    se;
  end;
```

4. Вначалепроцедуры**TForm1.Button1Click**добавьтеследующийтекст:

```
 Form1.Panel1.Visible:=False;Label3.Caption:='Количество
 неправильныхответов';N_Attempt:=0;
 Label4.Caption:=IntToStr(N_Attempt);
```

5. Внесите изменениявтекст процедуры поискаPOISKдляанализа результата.Процедураможетвыглядеть следующим образом:

```
ProcedurePOISK(STR_R:STR_30;VarSTR_N:STR_30;CHAR_:Char);
//Поискбуквывсловеи«открытие»ее
VarI:Integer;
Flag:Boolean;//Флагнайденаливсловенажатаябуква
beginFlag:=Fa
lse;
ForI:=1toLength(STR_R)dobegin//Поискбуквывслове
  IfSTR_R[I]=CHAR_thenbegi
   n
    STR_N[I]:=CHAR_;
    Flag:=True;e
   nd;
```

```
 end;
IfFlag=FalseThen
 beginN_Attempt:=N_Attempt+1;Form1.Label4.Caption
  :=IntToStr(N_Attempt);
 end
 ElseForm1.Label2.Caption:=STR_N;
IfSTR_N=STR_Rthen       //Играокончена-отгадановсеслово
 begin
 WithForm1.Panel1dobegi
  n
   Visible:=True;Font.Color:=
   clBlack;Caption:='Вывыигра
   ли!';
  end;Form1.Button1.Enabled:=True;
  end;
 IfN_Attempt=Attemptthen
//Играокончена-количествоошибокравноколичествупопыток
 begin
  WithForm1.Panel1dobegi
   n
    Visible:=True;Font.Color:=clR
    ed;Caption:='Выпроиграли...';
   end;Form1.Button1.Enabled:=True
  ;end;
end;
```

6. Сохранитепроект,запустите ипротестируйте его.

Пятый этап.Вывод иллюстраций

Длятого, чтобыпрограммабылаболее привлекательной, добавим вывод измененныхизображений взависимости отколичества допущенных ошибок.Ряд изменениязображенийможетвыглядеть так:




0 ошибок 1 ошибка 2 ошибки 3 ошибки и т.д.

Осталосьтолько добавитьэти изображениявпрограмму, но сначаласоздадимсамостоятельную программу,которая будетвыводить изображениявзависимости отнажатия кнопки.

Немноготеории

Длявыводаиллюстрации используется компонент **Image**, которыйнаходится навкладке **Additional**палитры компонентов.

Втаблице приведены основныесвойствакомпонента**Image**.

Имясвойства	Значение
Name	Имякомпонента

Picture	Свойство,являющееясяобъектом типа**Tbitmap**. Определяетвыводимуюкартинку
Left	Расстояние от левого краяформыдолевой границыобластикартинки
Top	Расстояние отверхнейграницыформыдоверхнейграницыобласти картинки
Height	Высотакартинки
Width	Ширинакартинки
Stretch	Признак автоматического сжатия или растяжения картинки таким образом,чтобы она была виднаполностью в области, размер которой задансвойствами**Width**и**Height**
AutoSize	Признак автоматического изменениянразмеракомпонента в соответствии среальнымразмеромкартинки. Еслизначение свойства**AutoSize**равно **True**, то приизменении значениясвойстваPictureавтоматическименяетсяразмеробластивыводаилл юстрации так, чтобыбыла виднався картинка. Еслизначение свойства**AutoSize**равно**False**,аразмеркартинкипревышает размер

Картинку,отображаемую вобласти**Image**, можно задатьво времясоздания формы или вовремяработы программы:

- Во времясоздания формы картинка задаетсяустановкой значения свойства**Picture**.
- Вовремяработыпрограммы-

применениемметода**LoadFromFile**.Например, для вывода иллюстрации,

находящейся в файле**dog.bmp**:**Image1.Picture.LoadFromFile('dog.bmp');**

Припроектированииформыможножесткозадатьпредельныйразмериллюстрации.Иеслире альныйразмериллюстрациипревышаетразмеробласти,выделеннойдляеевывода,тонеобхо димовычислитькоэффициентмасштабированияиустановитьмаксимальновозможные,про порциональныеширинеивысотеиллюстрации,значениясвойств**Width**и**Height**областивыв одаиллюстрации.Аеслиразмериллюстрациименьше области вывода,то можнопропорционально увеличить картинку.

Реальныеразмерыиллюстрации,загруженнойвобласть**Image**,можнополучитьизсвойств:

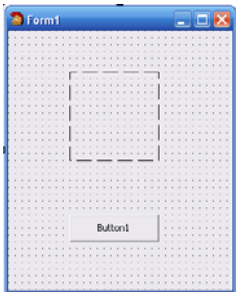
Image1.Picture.Bitmap.WidthI

mage1.Picture.Bitmap.Height.

Пример программы выводакартинок

Программапозволяет последовательно выводить на экран при нажатии накнопку**Button1**трикартинки, находящиесяв файлах **1.bmp**,**2.bmp**,**3.bmp**. После последнейкартинкибудет выведенаопять первая-**1.bmp**.При выводе на экранразмеркартинкимасштабируется взависимости отзаданного размеракомпанента**Image**.

Откройтеновыйпроект. Разместите на формеэкземпляры компонентов**Image**иButton.



Исходная форма

Выполнение программы

```
unitUnit1;in

terface

uses
  Windows,Messages,SysUtils,Variants,Classes,Graphics,Controls,Forms,
  Dialogs,StdCtrls,ExtCtrls;

type
  TForm1=class(TForm)Imag
    e1:TImage;Button1:TB
    utton;
    procedureButton1Click(Sender:TObject);proced
    ureFormCreate(Sender:TObject);
  private
    {Privatedeclarations}publi
  c
    {Publicdeclarations}end;

var
  Form1:TForm1;

implementationVa
r
   iw,ih:integer;
                        //ПервоначальныйразмеркомпонентаImageN_Image:
   integer;//Номервыводимойкартинки
   AFile:String;        //Имякартинки

{$R*.dfm}
//измененииеразмераобластивыводаиллюстрациипропорционально
//размеруиллюстрации
ProcedureScaleImage;
//Масштабированиеизображения
var
    pw,ph:integer;
                                //Размериллюстрацииscal
    eX,scaleY:real;//МасштабпоХиYscale:real;
                                //Масштаб
begin
      //Иллюстрацияужезагружена,получаемееразмеры
      pw:=Form1.Image1.Picture.Width;ph:=Fo
      rm1.Image1.Picture.Height;scaleX:=iw/
      pw;
      scaleY:=ih/ph;
```

```
       //Выбераемнаименьшийкоэффициент
       if   scaleX<scaleY
            thenscale:=scaleXelsesc
            ale:=scaleY;

       //ИзменяемразмеробластивыводаиллюстрацииForm1.Image1.Height:=Ro
und(Form1.Image1.Picture.Height*scale);Form1.Image1.Width:=Round(Form1
.Image1.Picture.Width*scale);
//таккакStrech=Trueиразмеробластипропорционален
//размерукартинки,токартинкамасштабируетсябезискажений
end;

procedureTForm1.Button1Click(Sender:TObject);begin
 IfN_Image=4ThenN_Image:=1;
 //Формированиеименикартинки
 AFile:=IntToStr(N_Image)+'.bmp';
 //УстановказначениясвойстваPictureедлявыводакартинки
 //вовремяработыпрограммы
 Form1.Image1.Picture.LoadFromFile(AFile);
 //МасштабированиекартинкиScal
 eImage;N_Image:=N_Image+1;
end;

procedureTForm1.FormCreate(Sender:TObject);begin
 Image1.AutoSize:=False;
 Image1.Stretch:=True;          //Разрешиммасштабирование

   //Запомнимпервоначальныйразмеробластивыводаиллюстрации
   iw:=Image1.Width;ih:=i
   mage1.Height;

   N_Image:=1;    //Начальноезначениеномеравыводимойкартинки
end;
end.
```

Шестой этап. Заключительный

Самостоятельно внеситеизменения впрограмму,добавив вывод картиноквзависимости от количествадопущенныхошибок.

Протестируйтеполученнуюпрограмму.

Программаимеетещеодин недостаток - после окончанияигры (вывод панелиссообщением об оконачии игры), если нажимать кнопки сбуквами,то программа можетаварийнозавершиться.Чтобы этого не быловначале процедуры Poiskвыпонитепроверку:

If (N_Attempt<Attempt)and (STR_N<>STR_R) then

Т.е.выполнять поиск нажатойбуквы тольков случае,если неисчерпаны всепопытки ине отгадано все слово.

Листингпрограммы

```
unitUnit1;

interfaceuses
  Windows,Messages,SysUtils,Variants,Classes,Graphics,Controls,Forms,
  Dialogs,StdCtrls,ExtCtrls;

type
  TForm1=class(TForm)Label1:
    TLabel;Label2:TLabel;Bu
    tton1:TButton;Button2:T
    Button;Label3:TLabel;La
    bel4:TLabel;Panel1:TPan
    el;Image1:TImage;
    procedureFormCreate(Sender:TObject);procedureBut
    ton1Click(Sender:TObject);procedureBtnClick(Send
    er:TObject);procedureButton2Click(Sender:TObject
    );
  private
    {Privatedeclarations}public
    {Publicdeclarations}end;

var
  Form1:TForm1;implementa
  tion
Const
 ABC=32;       //Количествобукв
 N=100;        //Максимальноеколичествозаписейвмассиве
 Attempt=10;//Максимальноеколичествопопыток
TYPE
T_R=record
                    //СтруктуразаписимассиваQue:
        string[250];//ВопросAns:string[30];
                    //Ответ
      end;
R=array[1..N]ofT_R;     //МассивВопросовиОтветов
STR_30=String[30];

VarAQ:R;
    Quetions_F:TextFile;//ФайловаяпеременнаяSTR_R,STR_N:S
    TR_30;//ОтгадываемоесловоCHAR_:Char;
                          //БуквадляпоискаKOL_QUE:Integer;
                          //КоличествовопросоввфайлеBtn:T
    Button;
    N_Attempt:Integer;     //Количествоошибок
    Iw,Ih:integer;       //
    первоначальныйразмеркомпонентаImageAfile:string;      //Имяфайла-
    картинки

{$R*.dfm}

procedureRead_File;
```

```
//Чтениеинформациииизфайлаизаписьвмассив
VarKOL,I:Integer;begin
 Assignfile(Quetions_F,'Quetions.txt');Reset(Quetions_F);
 KOL:=1;I:=1;
 WhilenotEof(Quetions_F)dobegin
   if(KOLmod2)=1thenReadln(Quetions_F,AQ[I].QUE)
                    elsebegin
                     Readln(Quetions_F,AQ[I].ANS);Inc(I)
                     ;
                    end;
   Inc(KOL);
  end;KOL_QUE
 :=I;
 closefile(Quetions_F);end;

ProcedureScaleImage;
//Масштабированиеизображения
//изменениеразмераобластивыводаиллюстрациипропорционально
//размеруиллюстрации
var
    pw,ph:integer;
                               //размериллюстрацииscal
    eX,scaleY:real;//масштабпоХиYscale:real;
                               //масштаб
begin
      //иллюстрацияужезагружена
      //получимееразмеры
      pw:=Form1.Image1.Picture.Width;ph:=For
      m1.Image1.Picture.Height;scaleX:=iw/pw
      ;
      scaleY:=ih/ph;
      //выберемнаименьшийкоэффициент
      if   scaleX<scaleY
           thenscale:=scaleXelsescale
           :=scaleY;
      //изменимразмеробластивыводаиллюстрацииForm1.Image1.Height:=Round(Fo
rm1.Image1.Picture.Height*scale);Form1.Image1.Width:=Round(Form1.Image1.Pic
ture.Width*scale);
//таккакStrech=Trueиразмеробластипропорционален
//размерукартинки,токартинкамасштабируетсябезискажений
end;

ProcedurePOISK(STR_R:STR_30;VarSTR_N:STR_30;CHAR_:Char);
//Поискбуквывсловеи«открытие»ее
VarI:Integer;
Flag:Boolean;//Флагнайденаливсловенажатаябуква
begin
If(N_Attempt<Attempt)and(STR_N<>STR_R)thenbegin
Flag:=False;
ForI:=1toLength(STR_R)dobegin//Поискбуквывслове
IfSTR_R[I]=CHAR_thenbegin
STR_N[I]:=CHAR_;
Flag:=True;
```

```
end;end;
IfFlag=FalseThen
 beginN_Attempt:=N_Attempt+1;Form1.Label4.Caption:=Int
  ToStr(N_Attempt);AFile:=IntToStr(N_Attempt+1)+'.bmp
  ';Form1.Image1.Picture.LoadFromFile(AFile);ScaleIm
  age;
 end
 ElseForm1.Label2.Caption:=STR_N;
IfSTR_N=STR_Rthenbegin
 WithForm1.Panel1dobegin
   Visible:=True;Font.Color:=clBl
   ack;Caption:='Вывыиграли!';
  end;Form1.Button1.Enabled:=True;en
  d;
 IfN_Attempt=Attemptthenbegin
  WithForm1.Panel1dobegi
   n
    Visible:=True;Font.Color:=clRed;
    Caption:='Выпроиграли...';end;
  Form1.Button1.Enabled:=True;end;
 end;end;

procedureTForm1.BtnClick(Sender:TObject);
//Процедураобработкисобытиянажатиясозданныхкнопок
VarSTR_:String;begin
STR_:=(SenderasTButton).Caption;
//ПеременнаяSenderсодержитимя
//объекта,которомусоответствуетданноесобытие
CHAR_:=STR_[1];POISK(STR_R,ST
R_N,CHAR_);
(SenderasTButton).Enabled:=False;end;
procedureTForm1.Button1Click(Sender:TObject);VarI,NU
MBER:integer;
beginForm1.Panel1.Visible:=False;
 Label3.Caption:='Количествонеправильныхответов';N_Attempt:=0;
 Label4.Caption:=IntToStr(N_Attempt);Randomize;
 NUMBER:=Random(KOL_QUE-1)+1;
 Label1.Caption:=AQ[NUMBER].QUE;STR
 _R:=AQ[NUMBER].ANS;STR_N:='';
 forI:=1toLength(STR_R)doSTR_N:=STR_N
 +'*';
 Label2.Caption:=STR_N;
```

```
 Form1.Button1.Enabled:=False;
 fori:=1toABCdo//Циклпосозданиюкнопокнаформе
  beginBtn:=TButton.Create(Form1);wi
   thBtndo
    begin//Свойствасоздаваемыхкнопок
     Parent:=Form1;Caption:=Ch
     r(191+I);Height:=30;
     Width:=30;
     Top:=250+((I-1)div8)*40;
     Left:=((I-1)mod8)*40+20;
     Font.Size:=9;
     OnClick:=FORM1.BtnClick;end;
   end;
  AFile:=IntToStr(N_Attempt+1)+'.bmp';Form1.Image1.Pi
  cture.LoadFromFile(AFile);ScaleImage;
end;
procedureTForm1.Button2Click(Sender:TObject);begin
Close;end;
procedureTForm1.FormCreate(Sender:TObject);begin
  WithForm1dobegin
   Caption:='Угадайслово';Height:
   =450;
   Width:=500;Color:=clWhite
   ;
     end;
  WithButton1dobegin
    Caption:='Начать';He
    ight:=40;Width:=75;
    Top:=20;Left:=400;F
    ont.Size:=9;
  end;
  WithButton2dobegin
    Caption:='Выход';Heig
    ht:=40;Width:=75;Top
    :=360;Left:=400;Font
    .Size:=9;
  end;
  WithLabel1dobegin
    Caption:='';Height:=
    40;Width:=250;Top:=
    70;
    Left:=40;Font.Style:=[fs
    Bold];
```

```
    Font.Size:=10;end;
  WithLabel2dobegin
    Caption:='';Height
    :=40;Width:=75;Top
    :=110;
    Left:=40;Font.Style:=[f
    sBold];Font.Size:=12;
  end;
  WithLabel3dobegin
    Caption:='';Height
    :=40;Width:=75;Top
    :=180;
    Left:=40;Font.Si
    ze:=9;
  end;
  WithLabel4dobegin
    Caption:='';Height
    :=40;Width:=75;Top
    :=180;Left:=300;
   //Font.Style:=[fsBold];Font.S
    ize:=10;
  end;
  WithPanel1dobegin
    Caption:='';Height
    :=45;Width:=185;To
    p:=15;Left:=110;
    Font.Style:=[fsBold];Fo
    nt.Size:=10;Visible:=F
    alse;
  end;
  WithImage1dobegin
    AutoSize:=False;
    Stretch:=True;//разрешиммасштабирование
    //запомнимпервоначальныйразмеробластивыводаиллюстрации
    Iw:=Width;Ih:
    =Height;Heigh
    t:=140;Width:
    =140;Top:=120
    ;Left:=350;
  end;Read_F
  ile;
end;end

.
```

Лабораторнаяработа№3,ПРОГРАММИРУЕМЫЙТЕСТ

1. Постановказадачи

Написатьпрограммутестирования,котораяудовлетворяет следующимтребованиям:

1. Количествовопросов теста можетварьироватьот5 до20.
2. Для каждого вопроса должно бытьпредусмотрено четыревариантаответов.
3. В результатетестирования должна быть выставлена оценка(неудовлетворительно,удовлетворительно, хорошо, отлично) взависимости от количества правильныхответов.
4. Вопросы иответы теста должнынаходиться вфайле.
5. Впрограмме должна быть заблокирована возможность возврата к предыдущемувопросу.

Новыми в этой программеявляются:

- использованиетипизированныхфайлов,
- построение форм в ходевыполненияпрограммы,
- использованиекомпонента**SpinEdit**(редактор числа)состраницы палитрыкомпонентов**Samples**.

2. Информационнаяпостановказадачи

Вусловии задачи сказано, что дляхранения используется файл.BDelphi поддерживаеттриразныхподхода для работы с файлами,для разного типа информации желательноиспользовать наиболее подходящий подход.

2.1. Немноготеории

Текстовыефайлы – в файле находится текстовая информация (набор строк изсимволов).

Текстовая информациясостоит не обязательно из букв, вней могутсодержатьсялюбыесимволы из таблицы ASCII.

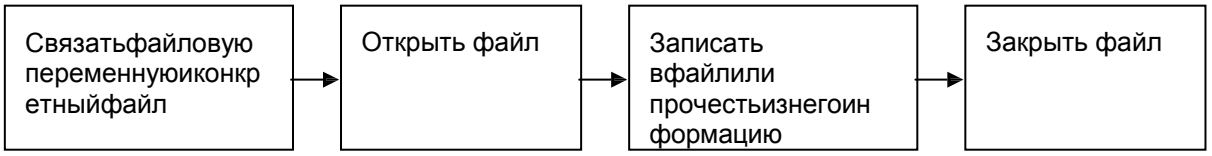
Типизированныефайлы – вфайле находитсяинформациялюбого рода. Ноструктура такой информации обязательно должна повторяться. Названиетипизированный файл получилиз-затого, что внем хранитсяоднотипнаяинформация.Однастрокатипа называетсязаписьютипизированного файла.

> **Примечание**
>
> Типизированныефайлыоченьчастоиспользуютсядляописаниякаких-либосписковинформации,например,телефонныхсправочников.Каждаязаписьданногоспискаможетсостоять,например,изномерателефонабонента,имя,адрес.

Нетипизированныефайлы – файл содержитпоследовательностьбайтбез какой-либо структуры.Например,этоможет бытьзакодированный или сжатый блокинформации.

Приработе с такимфайломвся информация рассматривается какнабор отдельныхкусков по N байт, где значение Nзадается программистомв диапазоне от1 до65535.

Впроцессеработы для любогофайланужновыполнять последовательностьследующихдействий:



Менятьместами пункты нельзя.

Для каждогофайла есть понятие – текущее положение вфайле, т.е.этотоположение, в которомна данный момент идет обработка информации.Например, втипизированном файлемыможемпрочестьпервую запись, затем вторую и такдалее. В такомслучае текущимположением вфайле сначала будет перваязапись,потомвторая, затем третья.

2.2. Структураинформации

Внашейзадаче единицейинформации будет являться вопрос, который в свою очередьбудет состоять из текста вопроса,четырех вариантов ответа и четырехуказателейправильностиответа. Таким образом, можноввеститип,описывающий вопрос.

```
TypeTTEST =Record

      TEXT:String[250];//Текст Вопроса

      OTV_1:String[100];//Вариантответа
      №1OTV_2:String[100];//Вариантответа
      №2OTV_3:String[100];//Вариантответа
      №3OTV_4:String[100];//Вариантответа №4

      REZ_1:Byte//Признакправильностиответа
      №1REZ_2:Byte//Признакправильностиответа
      №2REZ_3:Byte//Признакправильностиответа
      №3REZ_4:Byte//Признакправильностиответа №4

          end;
```

TEXT,**OTV_1**,**OTV_2**,**OTV_3**,**OTV_4**содержаттекстовуюинформацию,а**REZ_1**,**REZ_2**, **REZ_3**,**REZ_4**будутсодержатьчисловуюинформацию:0– ответнеправильный,1 – ответ правильный.Можно былобыдля «Признакаправильностиответа» ввести тип*Boolean*, номыввели числовоезначение, т.к. входе выполнения программы будемсуммировать признакиправильности для ответов,которые выбирает тестируемый. Врезультате мы получимколичествоправильныхответов.

Впредложенную структуруможновнести изменения, которые позволятвдальнейшемсократить текст программы.Дляэтого определим типмассива для данных «Вариантответа» и «Признак правильностиответа».Тогдаописаниетипабудет выглядеть:

```
TypeTTEST =Record

      TEXT:String[250];                                  //Текст Вопроса

      OTV:Array[1..4]ofString[100];//Вариантыответов

      REZ:Array[1..4]ofByte//Правильныйответ

          end;
```

Непроизвольно мыподошли ктому, что при работебудемиспользоватьтипизированныйфайл.

Дляработы программытестирования нам будет нужнаещеинформация о названиитестаи количествевопросов в тесте. Количество вопросов в тесте(количество записейвфайле)можноузнатьс помощьюфункции **FileSize.** Авотназваниетеста необходимопоместитьвфайл. Поэтомуинформация в файле будет располагаться следующимобразом:

Первая запись содержит названиетеста,все последующие записи – информациюовопросах. В результатеколичество записейв файле будетнаоднубольше, чем количествовопросов в тесте.

Для формированиятеста мы будемиспользоватьмассив:

```
ConstKOL= 21;                                    // Махколичествовопросов

Var     TEST_:Array[1..KOL]ofTTEST;               //МассивТЕСТА
```

2.3. Описаниефайла

Приобъявлениифайловой переменной типизированныхфайловуказывается тип данных,содержащихся в файле.

```
VarFFile:Fileof TTEST;                           //Описаниефайла
```

Для чтения или записифайлов такого типа возможно использованиетолько операторы**Read**и**Write**.**ReadLn**и**WriteLn**использоватьнельзя, т.к. данныевфайле этоне строкитекста,т.е. нети линий.

Данные втипизированном файлехранятся вовнутреннеммашинном виде. Поэтомсоздаватьфайл и читатьинформацию из него можно только с помощью программы,которая «знает»структуруданных.Дляэтого мы создадимдве программ. Первая будетотвечать за создание теста, а вторая – тестирование.

3. Программа созданиетеста

Откройтеновый проекти разместите вформе компоненты в соответствии с рис.29.

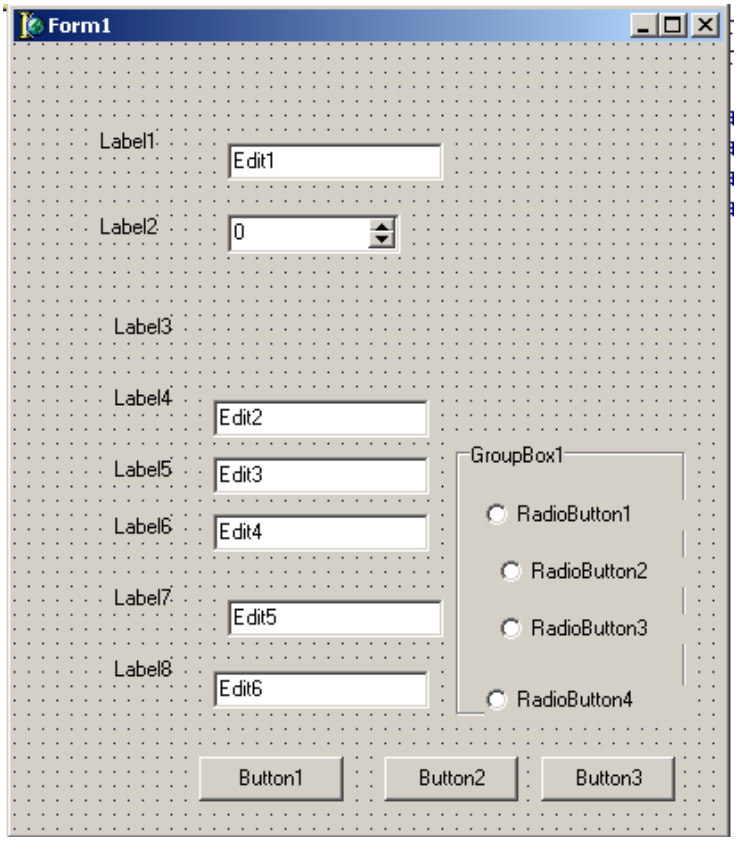


Рис.29

Нам понадобятся следующиекомпоненты:

№ п/п	Наименование компонента	Страница	Длячегопредназначен
1	Label1	Standard	Выводтекста«Названиетеста»
2	Label2	Standard	Выводтекста«Количествовопросов» в тесте
3	Label3	Standard	Выводтекста«Вопрос №»
4	Label4	Standard	Выводтекста«Текст»вопроса
5	Label5	Standard	Выводтекста«Ответ № 1»
6	Label6	Standard	Выводтекста«Ответ № 2»
7	Label7	Standard	Выводтекста«Ответ № 3»
8	Label8	Standard	Выводтекста«Ответ № 4»
9	Edit1	Standard	Окновводаназванияетеста
10	Edit2	Standard	Окновводатекставопроса
11	Edit3	Standard	Окновводавариантаответа № 1
12	Edit4	Standard	Окновводавариантаответа № 2
13	Edit5	Standard	Окновводавариантаответа № 3
14	Edit6	Standard	Окновводавариантаответа № 4
15	SpinEdit1	Samples	Выбор количествавопросов в*Тесте*. **Примечание** *SpinEdit*– редакторчисла. Обеспечивает отображениеиредактированиецелогочисла.Этоткомпоне нтсодержитдве кнопкисо стрелками,используемые дляувеличенияили уменьшениязначениячисла. Диапазон значенийчислазадаетсясвойствами *Min*и*Max*.Этисвойства определяют,соответственно, минимально имаксимальновозможныезначениячисла. Шаг изменениязначенияпринажатиикнопоксострелками содержится в свойстве *Increment*. Поумолчанию значениеэтогосвойстваравно
16	GroupBox1	Standard	Панель*GroupBox*позволяет объединитькомпоненты*RadioButton*длятого,чтоб ыкнопкивзаимодействовали другс другом в группе(может быть нажататолько одна кнопкаизчетырех).
17	RadioButton1	Standard	Выборправильногоответа
18	RadioButton2	Standard	
19	RadioButton3	Standard	
20	RadioButton4	Standard	
21	Button1	Standard	Кнопка перехода напредыдущий вопрос
22	Button2	Standard	Кнопка перехода наследующий вопрос
23	Button3	Standard	Запись готовоготестана диск (заполнены всевопросы)

Сохранитепроект. Впоследующем сохраняйтепроект ипроверяйте егоработоспособность после каждогоизменения.

Не стремитесь располагать компоненты на свои места. Это мысделаем впрограмме присоздании формы. Для этоговыделите объект**Form1**,перейдите на вкладку**EventsИнспекцияобъектов**(**ObjectInspector**),найдитесобытие**OnCreat**,справаотнегодважды щелкнуть левой кнопкой мыши. Попав в кодпрограммы,надо написатьследующийкод:

```
VarLeft_N :Integer;           //Отступслева верхнейчасти
       Top_N    :Integer;      //Отступсверху
       Left_NN:Integer;        //Отступслева для разделаответов
       Top_NN : Integer;       //ОтступсверхудляRadioButtonK, I
                :Integer;
   ...
Left_N:=30;Top_N:=50;Left_
NN:=60;

// Формированиеэлементовформы

Form1.Width:=740;Form1.Height:=540;Form1.Ca
ption:='Созданиетеста';

Label1.Left:=Left_N;Label1.Top:=Top_N;Label
1.Font.Style:=[fsBold];Label1.Font.Size:=10
;Label1.Caption:='Названиетеста';

Edit1.Text:='';Edit1.Top:=Top_N;
Edit1.Left:=Left_N+Label1.Width+10;Edit1.Width:
=300;

Top_N:=Top_N+40;

Label2.Left:=Left_N;Label2.Top:=Top_N;Label2.Fo
nt.Style:=[fsBold];Label2.Font.Size:=10;Label2.
Caption:='Количествовопросов';

SpinEdit1.Left:=Left_N+Label2.Width+10;SpinEdit1.
Top:=Top_N;SpinEdit1.MinValue:=5;SpinEdit1.MaxVal
ue:=20;SpinEdit1.Text:='5';SpinEdit1.Width:=40;

Top_N:=Top_N+60;

Label3.Left:=Left_N;Label3.Top:=Top_N;La
bel3.Font.Style:=[fsBold];Label3.Font.Si
ze:=10;Label3.Caption:='Вопрос № 1';

Top_N:=Top_N+40;Label4.Left:=Left_NN;
```

```
Label4.Top:=Top_N;Label4.Fon
t.Style:=[fsBold];Label4.Fon
t.Size:=9;Label4.Caption:='T
екст';

Edit2.Text:='';Ed
it2.Top:=Top_N;
Edit2.Left:=Left_NN+Label4.Width+10;
Edit2.Width:=600;Left_NN:=Left_NN+La
bel4.Width+10;

Top_N:=Top_N+40;

GroupBox1.Left:=620;GroupBox1.Top:
=Top_N+20;GroupBox1.Width:=90;Grou
pBox1.Height:=180;GroupBox1.Captio
n:='Правильный';GroupBox1.Font.Siz
e:=8;
```

Мыустановилисвойствакомпонентам**Label1**,**Edit1**,**Label2**,**SpinEdit1**,**Label3**,**Label4**,**Edit2**,**GroupBox1**. Всеониобладаютразнымисвойствами.Нокомпоненты,которыеописывают4 варианта ответов, можносгруппировать–**Label**+**Edit**+**RadioButton**(для каждого ответа).Компоненты каждой строки будут отличатьсятолько значением свойства**Top**исвоими номерами.Компоненты распределятся построкам следующимобразом:

1-аястрока-Label5,Edit3,RadioButton12-аястрока–
Label6,Edit4,RadioButton23-аястрока–
Label7,Edit5,RadioButton34-аястрока–
Label8,Edit6,RadioButton4

Поэтомувозникаетвопрос: «Каким образомможно перечислять впрограмме именакомпонентовпоследовательно,например, переходяот**Label5**к**Label6**,потомк**Label7**?»

Это можно сделать,если воспользоваться методом **FindComponent**.Он возвращаетуказатель экземпляра компоненты, о которой известно.Допустим,что форма содержитэкземплярkомпоненты**TLabel**сименем**Label2**. Чтобыполучитьуказательна экземпляр**Label2**иизменитьсвойство**Caption**,можнозаписать:

```
K:=2;

TLabel(FindComponent('Label'+IntToStr(K))).Caption:='МЕТКА';
```

Следовательно, для формирования строк вариантов ответов в программумы можемдобавитьтекст:

```
//  ФормированиеразделаОТВЕТОВ
K:=2;Top_NN:=-20;
For I:=1To 4
  dobegin
  Top_N:=Top_N+40;Top_NN:=Top_NN+40;

  TLabel(FindComponent('Label'+IntToStr(K+I+2))).Left:=Left_NN;
  TLabel(FindComponent('Label'+IntToStr(K+I+2))).Top:=Top_N;TLa
  bel(FindComponent('Label'+IntToStr(K+I+2))).Font.Size:=9;TLab
  el(FindComponent('Label'+IntToStr(K+I+2))).Caption:='Ответ№'
+IntToStr(I);
```

```
  TEdit(FindComponent('Edit'+IntToStr(K+I))).Text:='';TE
  dit(FindComponent('Edit'+IntToStr(K+I))).Top:=Top_N;TE
  dit(FindComponent('Edit'+IntToStr(K+I))).Left:=200;TEd
  it(FindComponent('Edit'+IntToStr(K+I))).Width:=400;

 TRadioButton(FindComponent('RadioButton'+IntToStr(I))).Left:=40;TR

adioButton(FindComponent('RadioButton'+IntToStr(I))).Top:=Top_NN;TR

adioButton(FindComponent('RadioButton'+IntToStr(I))).Caption:='';
 TRadioButton(FindComponent('RadioButton'+IntToStr(I))).Width:=10;

  end;
```

Ив конце добавляем блок формирования трех кнопок.

```
//  Формированиенижнейчасти формы

Top_N:=Top_N+70;

Button1.Caption:='Предыдущая';
Button1.Left:=Left_NN+100;Butt
on1.Top:=Top_N;Button1.Width:=
100;Button1.Height:=30;Button1
.Enabled:=False;

Button2.Caption:='Следующая';
Button2.Left:=Left_NN+250;But
ton2.Top:=Top_N;Button2.Width
:=100;Button2.Height:=30;

Button3.Caption:='Записать';
Button3.Left:=Left_NN+500;Bu
tton3.Top:=Top_N;Button3.Wid
th:=100;Button3.Height:=30;B
utton3.Enabled:=False;




STEP:=1;   //Номер вопроса =1
```

В результате, после запускапроекта,вы увидитеформу,представленную нарис.30.

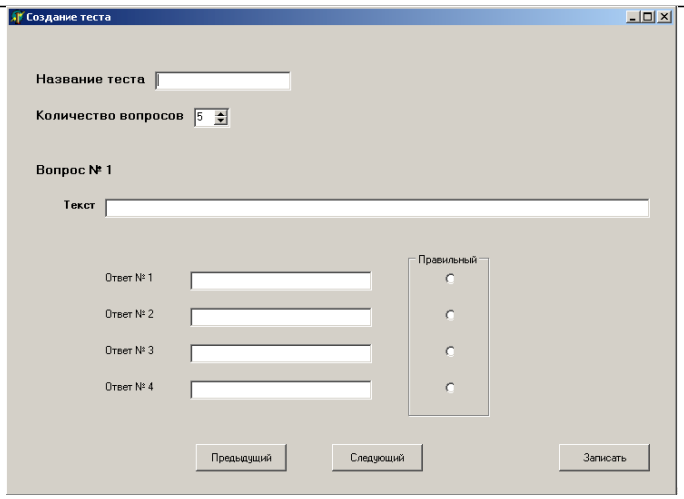


Рис.30

Наша задача – сформировать файл тестовыхвопросов, которыйпотом можноиспользовать дляпроведения тестирования.Дляэтого необходимоподумать о том, какбудуторганизованыданные. Из общеговида **Form1**можносделатьвывод, чтокаждыйвопрос состоит из текставопроса, четырех вариантов ответови четырех вариантовпризнаков правильностиответа.Дляудобства воспользуемся типом данных **Запись(Record).** Соднойстороны, записьможнорассматривать какединоецелое, с другойстороны – какнабор отдельных элементовразного типа.Для нашейзадачи подойдетследующееописание типа:

```
TypeTTEST  = Record
     TEXT       : String[250];                  // Текст Вопроса
     OTVR       : Array[1..4]of  String[40];//   ВариантыответовП
     EZ         : Array[1..4]of  Byte           // равильныйответ
        end;
```

Решить нашузадачу можноследующим образом:

1. Ввести название теста и количествовопросов.
2. Ввестивсевопросы с вариантамиответов,записывая их в массив.

   Примечание.
   Этопозволит просматривать(вперед, назад)вопросы,вноситьизменениядо записиихвфайл.

3. Сохранитьвсесформированные вопросывфайле.

Для этого наш массивбудет иметь тип записи**TTEST.**Враздел описаниявключимописание файла и описаниемассива.

```
ConstKOL = 21;                              // Махколичество
вопросов
VarFFile:Fileof TTEST;                      // Описаниефайла
    TEST_:Array[1..KOL]of TTEST;            // МассивТЕСТА
```

При построении массивапервый элемент массива (**TEST_[1]**) будетсодержатьинформацию о тесте:**TEST_[1].TEXT**– названиетеста

Начиная совторого элемента массива,будет располагаться непосредственно информацияо вопросах теста.

Впроцедуру**TForm1.FormCreate**необходимодобавитьещеодинблокподготовкимассива **TEST_**к работе.

```
For I:=1to KOL
  dobeginTEST_[I].TE
  XT:='';
  for K:=1to 4 do
    beginTEST_[I].OTV[K]
    :='';TEST_[I].REZ[K]
    :=0
    end;
  end;
```

Создадим процедуру,которая обрабатываетситуацию нажатиякнопки **Button3**(записьсозданногомассива в файл).Дляэтоговыделитеобъект **Button3**, перейдитена**вкладкуEventsИнспектора объектов**(**ObjectInspector**),найдите событие **OnClick**, справаотнего дважды щелкнуть левой кнопкой мыши. Попав вкод программы, надонаписатьследующийкод:

```
procedureTForm1.Button3Click(Sender:TObject);Va
rI       :Byte;
    STROK:TTEST;
beginAssignFile(FFILE,'TEST.t
xt');Rewrite(FFILE);
Button1.Enabled:=False;   //Недоступнывсе три кнопки
Button2.Enabled:=False;
Button3.Enabled:=False;

 For I:=1to
              STEP_N+1dob
  egin
   STROK:=TEST_[I];Wri
   te(FFILE,STROK)
  end;CloseFile(F
 FILE)
end;
```

Впредставленнойпроцедуре файловой переменной**FFILE**ставитсявсоответствие имяфайла, т.е.массивмызапишем вфайл **TEST.txt.** Затем открываетсяфайл (в нашемслучаеон создается) для записи. Вфайл переписываются все элементы массива**TEST_**.Ихколичествосоответствует введенномупараметру«Количествовопросов в тесте»плюс 1,т.к. добавленодин элемент (первый),который содержит названиетеста.

Устанавливаются недоступными кнопки«Предыдущая»,«Следующая»,«Запись», а затеммассив записывается в файл. В конце файл закрывается.

Перейдем ксозданию процедурыобработки**Button2**(«Следующая» – переходксследующейзаписи). Дляэтого выделите объект **Button2,**перейдите навкладку**EventsИнспектораобъектоВ**(**ObjectInspector**),найдитесобытие**OnClick**,справао тнегодважды щелкнуть левой кнопкой мыши. Разберем кодпроцедуры почастям.

Часть №1

```
procedureTForm1.Button2Click(Sender:TObject);Va
r I:Byte;
begin
If
    (RadioButton1.Checked=False)and(
   RadioButton2.Checked=False)and(R
   adioButton3.Checked=False)and(Ra
   dioButton4.Checked=False)
Thenbe
 gin
 ShowMessage('Не выбранправильныйответ');
 Exit
 end;
```

Вначале процедуры необходимо проверить выбранли правильныйответ (должна бытьнажата одна из кнопок **RadioButton**). Еслизначение свойства**CheckedyRadioButton**равно**False**, то будет выдано сообщение«Не выбран правильный ответ» ипроцедуразакончит свою работу.

Часть №2

```
If
 STEP=1Thenbeg
 in
  TEST_[1].TEXT:=Edit1.Text;
  Edit1.Enabled:=False;
  STEP_N:=StrtoInt(SpinEdit1.Text);
                                        //Количествовопросоввтес
те
  SpinEdit1.Enabled:=False;
```

Этот кусокпроцедурынеобходим,т.к. вмассиве элементы с индексом 1 содержатинформацию о названии теста. Поэтому, еслиунас первый вопрос (**STEP**=1), тоосуществляетсязаписьвмассивинформацииотесте иустанавливаютсяобъекты**Edit1** и**SpinEdit1**недоступными.Переходим к формированию вопросов.

Часть №3

```
TEST_[STEP].TEXT:=Edit2.Text;

For I:=1to 4
 dobegin
TEST_[STEP].OTV[I]:=TEdit(FindComponent('Edit'+IntToStr(I+2))).Text;
IfTRadioButton(FindComponent('RadioButton'+IntToStr(I))).Checked=TrueThenT
EST_[STEP].REZ[I]:=1;
 end;
```

Втретьейчасти элементам массива с индексом **STEP**присваивается значениесоответствующих полей. Еслинажата**RadioButton**,тосоответствующемуэлементумассива **REZ[I]**(*Результат ответа*) присваивается значение 1.

Часть №4

```
Button1.Enabled:=True;
If STEP>STEP_NThen
 beginButton2.Enabled:=Fa
  lse;Button3.Enabled:=Tr
  ue;
 end
Else
 beginSTEP:=ST
 EP+1;
 Label3.Caption:='Вопрос № '+IntToStr(STEP-
 1);Edit2.Text:=TEST_[STEP].TEXT;
 For I:=1to 4
 dobegin
TEdit(FindComponent('Edit'+IntToStr(I+2))).Text:=TEST_[STEP].OTV[I];
  If TEST_[STEP].REZ[I]= 1 Then
   TRadioButton(FindComponent('RadioButton'+IntToStr(I))).Checked:=  True
   else
   TRadioButton(FindComponent('RadioButton'+IntToStr(I))).Checked:=  False
 end
ende
 nd;
```

Взаключительной части сначалаустанавливается доступнойкнопка**Button1**(«Предыдущая»),т.к. уже есть одна записьи мы, после выполненияпроцедуры,переходим навторую.Дальше идет проверка– являетсяли наша запись последняя,т.е.номертекущейзаписи(**STEP**) больше количества вопросов втесте(**STEP_N**). Еслимызаписаливмассив все вопросы,токнопка**Button2**(«Следующая») устанавливаетсянедоступной, а кнопка**Button3**(«Запись») – доступной.

1. Еслинедостигликонцатеста,то идет подготовка кследующему вопросу:увеличиваетсяпеременная**STEP**исвойствам компонентэкрана присваиваютсязначения следующей записимассива.
2. Если массив не заполнендо конца,тоэто будут пустые записи.
3. Если массив былзаполнен, и мыпереходилиотоднойзаписи к другой спомощью кнопок **Button1**(«Предыдущая») и**Button2**(«Следующая»),тонаэкранбудутвыведены значения соответствующейзаписи.

Перейдем ксозданию процедурыобработки**Button1**(«Предыдущая» – переход кпредыдущей записи).Дляэтого выделите объект **Button1**, перейдите навкладку**EventsИнспектораобъектов**(**ObjectInspector**),найдитесобытие**OnClick**,справао тнегодважды щелкнуть левой кнопкой мыши. Попав в кодпрограммы,надо написатьследующийкод:

```
procedureTForm1.Button1Click(Sender:TObject);Va
r I: Byte;
beginSTEP:=ST
EP-1;
Label3.Caption:='Вопрос № '+IntToStr(STEP-
1);Edit2.Text:=TEST_[STEP].TEXT;
For I:=1to 4
 dobegin
TEdit(FindComponent('Edit'+IntToStr(I+2))).Text:=TEST_[STEP].OTV[I];
If TEST_[STEP].REZ[I]= 1 Then
```

```
    TRadioButton(FindComponent('RadioButton'+IntToStr(I))).Checked:=  True
    Else
    TRadioButton(FindComponent('RadioButton'+IntToStr(I))).Checked:=  False
 end;
If
 STEP=2thenButton1.Enabl
 ed:=False;
Button2.Enabled:=True;
end;
```

Вначале переходим кпредыдущемувопросу (уменьшаем значениепеременной**STEP**), азатем выводимнаэкран содержимое вопроса. В концепроцедуры проверяем,если вопросявляетсяпервым (номерстрокимассива**STEP**=2), то делаем недоступнойкнопку*Предыдущая* и влюбом случае должнабыть доступна кнопка*Следующая*.

Сохранитепроект окончательно, запустите ипротестируйте его.

Задание

Внесите изменение в программутак, чтобыможнобылобы просматривать ужесозданныетесты.

4. Программатестирования

Программатестирования будет состоять из двух форм:титульнойформы,накоторойосуществляется выбор тестаиз справочника тестов (текстовый файл) и форма формынепосредственнотеста.

Взависимости отправильности ответов навопросы теста, подсчитываетсярезультативыставляется оценка.

4.1. Созданиетитульнойформы

Титульнаяформа будет выполнять следующиефункции:

- Поиск справочника тестов (текстовый файл **STEST.txt**), чтение информации отестах-файлах и выводперечня тестов вкомпонент**RadioGroup1**.
- Взависимости от выбранного тест-файла,формирование рабочегофайлатестов.
- Вызов формы непосредственного тестирования.
- Удаление рабочегофайла тестов при окончании работы программы.

Справочник тестов – текстовый файл, которыйможносоздатьв любомтекстовомредакторе. Каждая строка файла содержит наименование теста. Может иметь,например,такое содержание:

```
Информация и информационные
процессыАрхитектуракомпьютера
Измерение
информацииОсновы
```

Каждомутестусоответствует файл-тест,который имеетимя **TEST_N.txt**, где **N**– номерстроки вфайле.

Откройтеновый проекти разместите вформе компоненты в соответствии с рис.31.

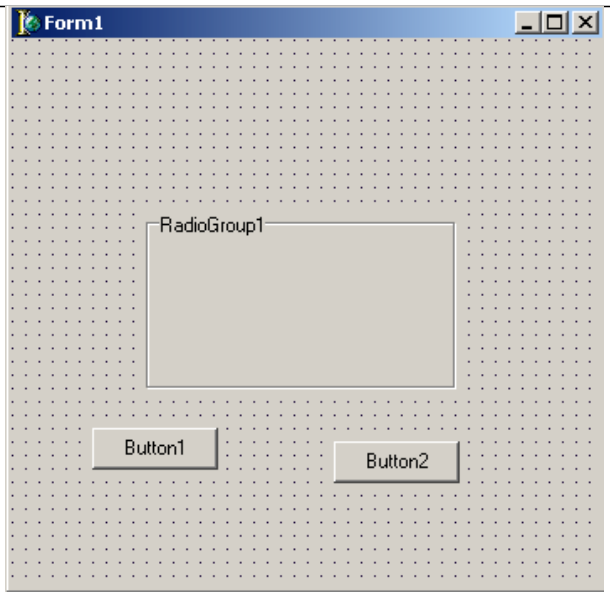


Рис.31

№ п/п	Наименование компонента	Страница	Длячегопредназначен
1	RadioGroup1	Standard	Перечень возможных тестов (из справочникатестов)
2	Button1	Standard	Кнопка дляперехода ктестированиюпослевыбора теста
3	Button2	Standard	Кнопкавыхода из программы тестирования

Вразделе**implementation**разместите описание типа иданных,которыемыбудемиспользовать при работе.

```
TypeTTEST = Record
        TEXT        : String[250];                      //Текствопроса
        OTV         : Array[1..4]ofString[40];//Вариантыответов
        REZ         :Array[1..4]ofByte                  //Правильныйответ
            end;
VarSFILE :TextFile;             //Справочниктестов
       FFile:Fileof TTEST;//Файлтестов
       FF_R : Fileof TTEST;//Рабочийфайлтестов
     KOL     :Byte;             //Количествотестов в справочникетестов
       TEST_:TTEST;             //Строказаписифайла
```

Выделитеобъект**Form1**, перейдитена вкладку **EventsИнспектора объектов**(**ObjectInspector**),найдите событие **OnCreat**, справаотнегодваждыщелкнутьлевой кнопкоймыши.Попав в кодпрограммы,надо написать следующий код:

```
procedure TForm1.FormCreate(Sender:TObject);
VarISF     : Byte;
    T_TEXT : String;
    Left_N : Array[1..10]of String[50];
    Top_N  : Integer;    //отступслева
begin      : Integer;    //отступсправа
```

```
KOL:=1;
AssignFile(SFILE,'STEST.txt');
{$I-}Reset(SFILE);{$I+}
If IOResult =0
 thenbegin
  Repeat
    ReadLn(SFILE,SF);
    T_TEXT[KOL]:=SF;K
    OL:=KOL+1;
  until(Eof(SFILE))or
  (KOL>10);CloseFile(SFILE);

  Left_N:=30;
  Top_N:=50;

  Form1.Caption:='Тестирование';Form
  1.Width:=400;Form1.Height:=Top_N+K
  OL*25 + 120;

  RadioGroup1.Top:=Top_N;RadioGroup1.Lef
  t:=Left_N;RadioGroup1.Width:=350;Radio
  Group1.Height:=KOL*25;RadioGroup1.Capt
  ion:= ' Выберитест ';For I:=1to KOL-
  1do
   RadioGroup1.Items.Add(T_TEXT[I]);
  RadioGroup1.Font.Size:=11;

  Button1.Caption:='Тест';Button
  1.Left:=Left_N;Button1.Top:=To
  p_N+KOL*25+40;Button1.Width:=1
  00;Button1.Height:=30;

  Button2.Caption:='Выход';Butto
  n2.Left:=Left_N+250;Button2.To
  p:=Top_N+KOL*25+40;Button2.Wid
  th:=100;Button2.Height:=30;
 end
  else
   ShowMessage('Отсутствуетсправочниктестов');en
d;
```

Немноготеории

Директива компилятора{$I+}/{$I-}включаетиливыключаетавтоматическуюпроверкуошибок ввода/вывода.Когдадирективавключена и возникает ошибкаввода/вывода, товыполнениепрограммы завершается с сообщением опроизошедшейошибке. Но можноотключить аварийноезавершение

программыпри возникновении ошибки. Вэтом случае программистберет насебя обработкуаварийной ситуации.Длятого, чтобы узнать произошла ли ошибка при открытиифайла,можновоспользоваться функцией *IOResult*. Она возвращаеткод отличныйотнуля, если произошла ошибка иноль, еслиоперация выполниласьуспешно.

Типичнойошибкой ввода/вывода является отсутствие файла.Поумолчаниюдирективакомпиляторавключена –{$I+}.

Вначале процедуры открывается файл справочникатестов **STEST.txt**.

Еслифайл существует,то информация из него переписываетсявмассив**T_TEXT**иподсчитываетсяколичествозаписей(**KOL**).

Впрограмме установленоограничениенаколичествоэлементов массива**T_TEXT**– неболее 10. По желанию это можно изменить. После устанавливаютсясвойствакомпонентыформа.

Список значений свойства**Items**компоненты **RadioGroup1**будем формировать входепрограммывзависимости отколичества элементов вмассиве справочника тестов.Дляэтогоксвойству**Items**применяемметод**Add**:

```
For I:=1to 4 do RadioGroup1.Items.Add(T_TEXT[I]);
```

Еслифайлсправочника тестов отсутствует,тобудет выдано сообщение*Отсутствуетсправочниктестов.*

В результатеформабудет выглядеть так же как нарис.32.

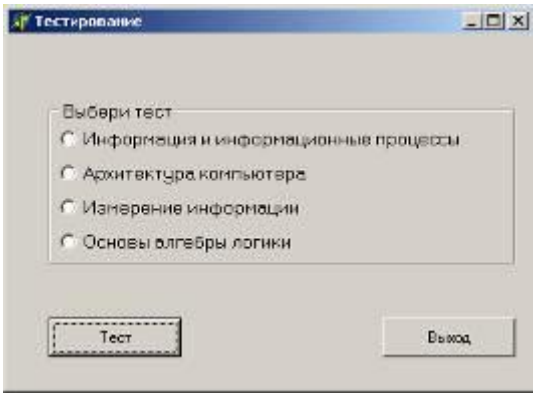


Рис.32

Создайте новуюформу**Form2**исохраните подименем **Unit2.pas**.Подсоединитесозданнуюформу ктитульнойформе. Для этого в**Unit1.pas**вразделе**interface**встрокуспискаподсоединенныхмодулейдобавьте **Form2**.

Создадим процедуру,которая обрабатываетситуацию нажатиякнопки **Button1**(*Тест*).Дляэтого выделите объект **Button1**, перейдите навкладку**EventsИнспектораобъектов**(**ObjectInspector**),найдитесобытие**OnClick**,справа отнегодваждыщелкнутьлевойкнопкоймыши. Попав в кодпрограммы,надо написатьследующий код:

```
procedureTForm1.Button1Click(Sender:TObject);Va
r N:Byte;
beginN:=RadioGroup1.ItemInd
ex+1;
AssignFile(FFILE,'TEST_'+IntToStr(N)+'.txt');//Тестсномером
N
{$I-}Reset(FFILE);{$I+}
If IOResult =0
 thenbegin
  AssignFile(FF_R,'TEST_R.txt');    //Рабочийфайлтестов
  Rewrite(FF_R);
  Repeat
    Read(FFILE,TEST );
```

```
    Write(FF_R,TEST_);u
  ntil(Eof(FFILE));Clos
  eFile(FFILE);CloseFil
  e(FF_R);Form2.ShowMod
  al;
 end
 elseShowMessage('Отсутствуетфайлтестов');

end;
```

Вначале процедурыформируетсяимя теста взависимости отномера выбраннойстрокикомпонента**RadioGroup1**.

> **Примечание**
>
> Индексперовогопереключателяравен0,амассивсправочникасодержитиндексперовогоэлемент аравный1.

Дальше открываетсявыбранный файл-тест.Еслиданный файл отсутствует,то выдаетсясообщение «Отсутствует файл тестов». Еслипри открытии файла-тестаневозниклоошибки, тоинформацияизнего, переписываетсяврабочийфайл тестов (**TEST_R.txt**). Взавершении открываетсяформа какмодальная.Этоозначает, чтоуправление передаетсяновой форме, и пользователь неможет передать фокус другойформеданного приложениядо тех пор,пока он не закроет модальную форму.

Создадим процедуру,которая обрабатываетситуацию нажатиякнопки **Button2**(«Выход»).Дляэтого выделите объект **Button2**, перейдите навкладку**EventsИнспекцораобъектов**(**ObjectInspector**), найдитесобытие**OnClick**,справа отнегодваждыщелкнутьлевой кнопкой мыши. Попав вкод программы, надонаписать следующий код:

```
procedureTForm1.Button2Click(Sender:TObject);be
gin
Close;
end;
```

4.2. Созданиеформытестирования

Сделайте активной**Form2** и разместите на ней компонентывсоответствиис рис.33.

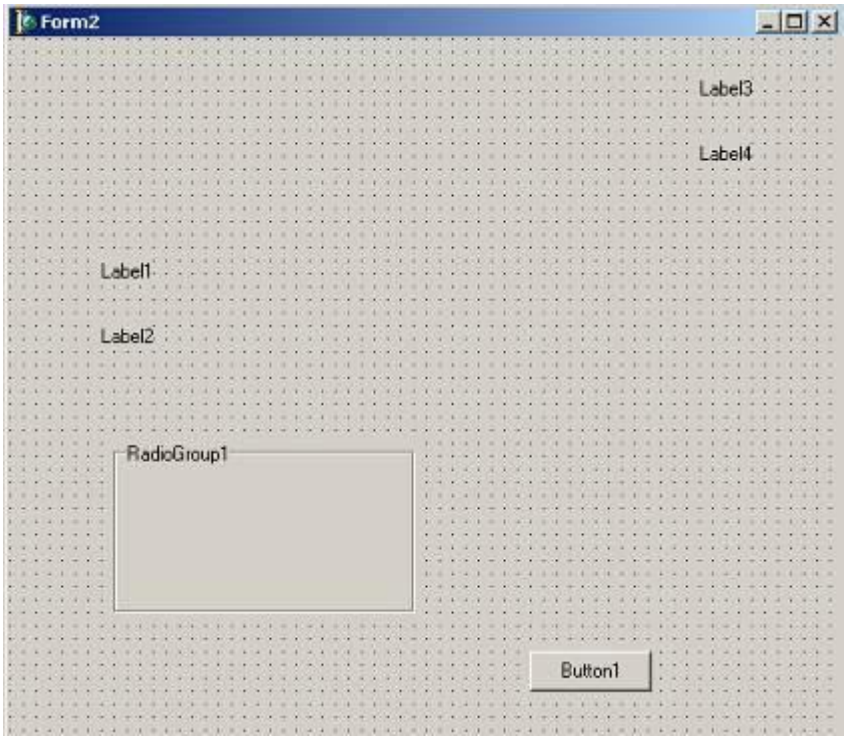


Рис.33

№ п/п	Наименование компонента	Страница	Длячегопредназначен
1	Label1	Standard	Выводтекста«Вопрос № …»
2	Label2	Standard	Выводтекста«Текст»вопроса
3	Label3	Standard	Выводтекста «Всеговопросов…»
4	Label4	Standard	Выводтекста«Правильныхответов…»
5	RadioGroup1	Standard	Выводвариантовответов
6	Button1	Standard	Кнопка дляперехода кследующемувопросу,показурезультата и

Вразделе**implementation**разместите описание типа иданных,которыемыбудемиспользовать при работе.

```
TypeTTEST  = Record
     TEXT      : String[250];                 // Текст вопроса
     OTVR      : Array[1..4]of  String[100];//  ВариантыответовП
     EZ        : Array[1..4]of  Byte          // равильныйответ
        end;
Var FF_R : Fileof  TTEST; //Рабочийфайл тестов
    KOL  : Byte;B         //Количествовопросов в тесте
    KOL_N: yte;TT         //Текущийномер вопроса
    TEST_: EST;           //Строказаписифайла
    REZ_N: Byte;          //Количествоправильныхответов
```

Дляобъекта**Form2**, перейдитена вкладку **EventsИнспектораобъектов**(**ObjectInspector**),найдите событие **OnActivate**,справа отнегодваждыщелкнуть левойкнопкоймыши.Попав в кодпрограммы,надо написать следующий код:

```
procedureTForm2.FormActivate(Sender:TObject);Va
rI: Byte;
beginAssignFile(FF_R,'TEST_R.t
xt');Reset(FF_R);
Read(FF_R,TEST_);
KOL:=FileSize(FF_R)-
1;KOL_N:=1;
REZ_N:=0;

Label1.Visible:=True;Radio
Group1.Visible:=True;

Form2.Caption:=TEST_.TEXT;
Form2.Height:=420;Form2.Wi
dth:=850;

Label3.Caption:='Всеговопросов -
'+IntToStr(KOL);Label3.Top:=20;
Label3.Left:=700;
Label3.Font.Size:= 8;

Label4.Caption:='Правильныхответов -
'+IntToStr(REZ_N);Label4.Top:=40;
```

```
Label4.Left:=700;
Label4.Font.Size:=
8;Label4.Font.Color:=clBlack;

Button1.Caption:='Следующий';
Button1.Top:=350;Button1.Left
:=720;Button1.Width:=100;Butt
on1.Height:=30;

Read(FF_R,TEST_);

Label1.Caption:='Вопрос №
'+IntToStr(KOL_N);Label1.Font.Style:=
[fsBold];Label1.Font.Size:= 11;
Label1.Height:=50;Label1.Wi
dth:=400;Label1.Left:=40;La
bel1.Top:=70;

Label2.Caption:=TEST_.TEXT;
Label2.Font.Size:=
12;Label2.Height:=50;Label2
.Width:=500;Label2.Left:=60
;Label2.Top:=100;Label2.Wor
dWrap:=
True;Label2.AutoSize:=
False;

RadioGroup1.Top:=160;Rad
ioGroup1.Left:=60;RadioG
roup1.Width:=750;
RadioGroup1.Height:=150;RadioGroup1.Fon
t.Size:=11;RadioGroup1.Caption:= '
Выбериответ ';RadioGroup1.Items.Clear;
 For I:=1to 4
  doRadioGroup1.Items.Add(TEST_.OTV[I]
  );

end;
```

Работаданной процедурыначинаетсяс того,что открывается рабочий файл тестов**TEST_R.txt**и считываетсяпервая запись,которая содержит название теста. Спомощьюфункции **FileSize**определяем количество записейвфайле, а количество вопросовбудетнаединицуменьше.

Дальшенастраиваемкомпоненты**Form2**,**Label3**,**Label4**,**Button1**.Читаемизрабочегофайла тестов еще однузапись– первыйвопрос. После этого настраиваем компоненты**Label1**,**Label2**,**RadioGroup1**.Приописании компонента**RadioGroup1**мыдобавляем

```
RadioGroup1.Items.Clear;
```

Этастрока позволяетнам очистить списокзначений свойства**Items**компоненты**RadioGroup1**, которыйформируется входе программы.Применениеметода**Clear**намнеобходимо при повторномтестировании – очистить списокот предыдущихвариантовответов.

В результатеформабудет выглядеть так же как нарис.34.

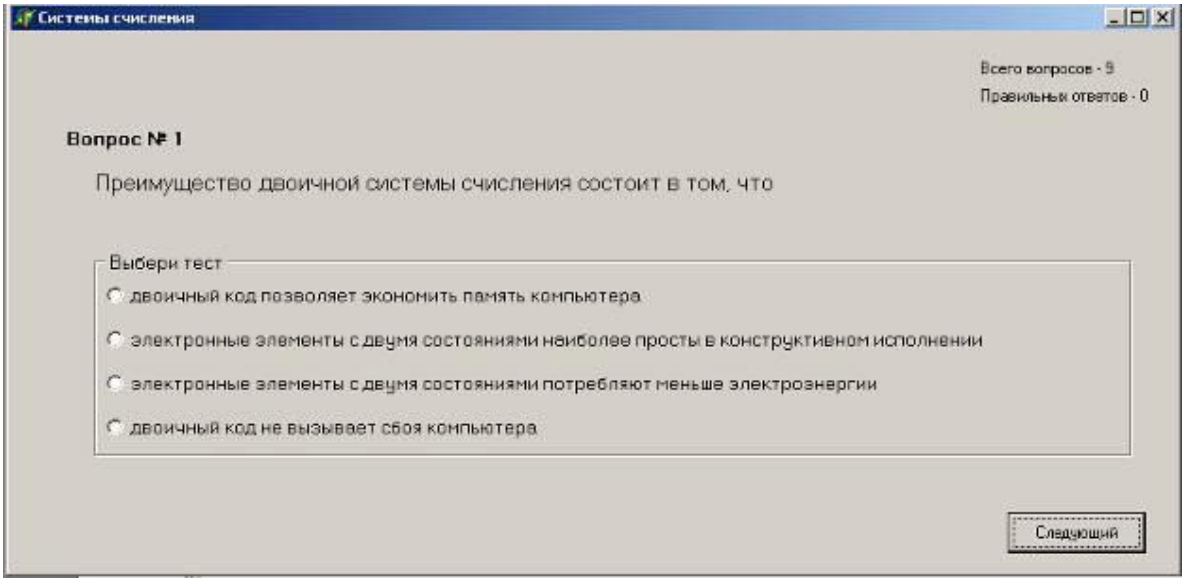


Рис.34

Создадим процедуру,которая обрабатываетситуацию нажатиякнопки **Button1**.Дляэтоговыделите объект**Button1**, перейдите на вкладку**EventsИнспектора объектов**(**ObjectInspector**),найдите событие **OnClick**,справа отнегодваждыщелкнуть левой кнопкоймыши.Попав в кодпрограммы,надо написать следующий код:

Часть №1

```
procedureTForm2.Button1Click(Sender:TObject);Va
r N, I     :Byte;
    OZENKA : Byte;
begin

If Button1.Caption='Результат'
 Thenbegin
  OZENKA:=Round(REZ_N/KOL*5);
  Button1.Caption:='Выход';La
  bel1.Visible:=False;RadioGr
  oup1.Visible:=False;

  Label2.Font.Size:=
  12;Label2.Left:=250;Lab
  el2.Top:=150;
  Label2.Caption:='Количествоправильныхответов-
  '+IntToStr(REZ_N)+'из'+IntToStr(KOL);

  Label3.Font.Size:=
  14;Label3.Font.Style:=
  [fsBold];Label3.Left:=300;Labe
  l3.Top:=200;Label3.Caption:='O
  ценка';

  Label4.Font.Size:=
  14;Label4.Left:=350;Lab
  el4.Top:=200;
  CaseOZENKAof
```

```
  1..2:Label4.Font.Color:=clPurple;
  3    : Label4.Font.Color:=clMaroon;
  4    : Label4.Font.Color:=clGreen;
  5    :
  Label4.Font.Color:=clNavy;end;Label4.Ca
  ption:=IntToStr(OZENKA);
 endelse
 If Button1.Caption='Выход' Thenbegin
  Close;CloseFile(FF_R);
  end
 elsebegi
 n
  N:=RadioGroup1.ItemIndex+1;REZ_N
  :=REZ_N+TEST_.REZ[N];KOL_N:=KOL_
  N+1;

  If KOL_N>KOLThenbegin
   Label4.Caption:='Правильныхответов -
   '+IntToStr(REZ_N);Button1.Caption:='Результат'
  end
   elsebegin
     Label4.Caption:='Правильныхответов -
     '+IntToStr(REZ_N);Read(FF_R,TEST_);
     Label1.Caption:='Вопрос №
     '+IntToStr(KOL_N);Label2.Caption:=TEST_.TEXT;RadioGroup1.Ite
     ms.Clear;
     For I:=1to 4 doRadioGroup1.Items.Add(TEST_.OTV[I]);
    end;end;
end;
```

Входе программы кнопка**Button1**может иметь надписи «Следующая»,«Результат»,

«Выход».

Если кнопка имеет надпись «Результат» и вы кликнулина ней, то в этомслучаеопределяетсязначениепеременной**OZENKA**по пяти бальной системе взависимости отколичества правильныхответов**REZ_N**и общего количества вопросов **KOL**. Затемустанавливаетсянадпись кнопки«Выход», формируетсяновое изображение формы,выводитсяитоговая оценка, делаются невидимыми компоненты**Label1**и **RadioGroup1**(поэтомувпроцедуре**TForm2.FormActivate**устанавливаютсяданныекомпонентывидимыми).

Если кнопка имеет надпись «Выход» и вы кликнулинаней,то в этомслучаезакрываетсярабочий файл тестов изакрывается форма.

Если кнопка имеет надпись «Следующий» и вы кликнулина ней, то в этомслучаеопределяется какой былвыбран ответ навопрос, результатскладываетсяс переменной**REZ_N**.Увеличиваетзначениепеременной**KOL_N**(номертекущеговопроса).Еслиновый текущий номербольше количествавопросов,то изменяется надпись накнопке

(«Результат»), впротивном случаепереходимкчтению нового вопроса из рабочегофайлатестов и формируем вывод нового вопроса.

Все, программа готова.Осталосьтолько ее протестировать.

Заданиедлясамостоятельного выполнения

	Задание
1	Написать программу, которая осуществляет конвертирование информации изтекстовогофайла в типизированный файл, для файла с тестовыми
2	Предусмотреть возможность выбораслучайнымобразомтестового задания изобщего списказаданий.
3	Создать файл, в которомнакапливается информация опрохожденииитестирования.Информация, которая может находиться в нем следующая:фамилия, имя,класс,номервопрос, номер ответа учащегося,количествоправильных ответов,количество неправильныхответов,времяначалатестирования, времяокончания